BAYESIAN, GRADIENT-FREE, AND MULTI-FIDELITY SUPERVISED DIMENSION REDUCTION METHODS FOR SURROGATE MODELING OF EXPENSIVE ANALYSES WITH HIGH-DIMENSIONAL INPUTS

A Dissertation
Presented to
The Academic Faculty

By

Raphaël H. Gautier

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Engineering
Department of Aerospace Engineering

Georgia Institute of Technology

May 2022

BAYESIAN, GRADIENT-FREE, AND MULTI-FIDELITY SUPERVISED DIMENSION REDUCTION METHODS FOR SURROGATE MODELING OF EXPENSIVE ANALYSES WITH HIGH-DIMENSIONAL INPUTS

Thesis committee:

Professor Dimitri Mavris, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Professor Lakshmi Sankar
School of Aerospace Engineering
Georgia Institute of Technology

Professor Graeme Kennedy
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Sayan Ghosh
Probabilistic Design and Optimization
GE Research

Dr. Chung Lee
School of Aerospace Engineering
Georgia Institute of Technology

Date approved: April 20, 2022

ACKNOWLEDGMENTS

The completion of the research undertaken in this thesis would not have been possible without the support I have had the chance to receive both before and during this journey. I would first like to warmly thank my advisor, Professor Mavris, for his unfaltering support and granting me access to unparalleled research opportunities during the years I have spent at ASDL. I would also like to thank Professor Sankar and Professor Kennedy for accepting to serve on my committee and contribute their time in addition to their regular obligations. The formulation of the methods I propose and the experiments I conduct have greatly benefited from the guidance of Dr. Sayan Ghosh and Dr. Piyush Pandita, with whom I truly enjoyed working during my formative internship at GE Research. My research meetings would not have been the same without the wit and old-school references of Dr. Chung Lee, to whom I also express my gratitude for his support. I would like to acknowledge my colleagues and friends Dr. Christian Perron and Dr. Dushhyanth Rajaram for the support they provided by sharing their data with me, and thank them for the stimulating discussions we have had together about our research ideas. I would like to thank Adrienne Durham and Tanya Ard-Smith for helping me navigate all administrative procedures with constant helpfulness, kindness, and patience. I would like to acknowledge Georgia Tech's Partnership for an Advanced Computing Environment (PACE) for the computing resources made available for my research. Despite the distance that separates us, my parents, my brother, and Oriane have always remained close, thoughtful, and extremely supportive on a daily basis during the completion of my thesis, and they are also responsible for nurturing and making possible the aspirations that eventually led me to pursue my graduate studies at Georgia Tech. The last months of my thesis were brightened by the birth and news of my nephew Corentin growing up, whom I expect to grasp the intricacies of Bayesian statistics in no time. Of course, Coline's kindness, patience, and support during our shared PhD journey was invaluable, and I am looking forward to our next adventures together.

TABLE OF CONTENTS

Acknowledgments iii

List of Tables x

List of Figures xii

List of Acronyms xxv

Summary xxviii

Chapter 1: Motivation and Definition of Research Objectives 1

1.1 Foreword 1

1.1.1 Document Organization 1

1.1.2 Section Organization 2

1.2 Motivation 3

1.2.1 Context: Aerospace conceptual design 3

1.2.2 Many-Query Applications 5

1.2.3 The Need for Surrogate Modeling 8

1.2.4 Sparse Training Data 9

1.2.5 High-Dimensional Inputs 10

1.2.6 Gap Identification 11

1.3 Background for Scope Definition 12
1.3.1 Overview of Surrogate Modeling 13
1.3.2 Data-Fit Surrogate Modeling 14
1.3.3 Reduced-Order Methods 18
1.3.4 Multi-Fidelity Surrogate Modeling 20
1.3.5 Discussion 20
1.3.6 Adaptive Sampling 21
1.3.7 Scope Definition: Three Surrogate Modeling Scenarios 24
1.4 Definition of the Research Objectives 26
1.4.1 The Curse of Dimensionality 26
1.4.2 Impact of the Curse of Dimensionality on Considered Surrogate Modeling Scenarios 29
1.4.3 Research Framing 31
1.5 Summary 34

Chapter 2: Fully Bayesian Approach to Approximation by Ridge Functions . . . 36

2.1 Refinement of Research Question 1 36
2.1.1 Inapplicability of Unsupervised Dimension Reduction Strategies . . 36
2.1.2 Existing Approaches for Surrogate Modeling with High-Dimensional Inputs 44
2.1.3 Approximation by Ridge Functions 50
2.1.4 Conclusion 58
2.2 Fully Bayesian Approach to Approximation by Ridge Functions 60
2.2.1 Background and Research Objective 60

2.2.2 Proposed Method 68
2.2.3 Setup of Experiment 1.1 78
2.2.4 Preliminary Results: In-Depth Walk-Through 95
2.2.5 Results of Experiment 1.1 106
2.2.6 Conclusion 118
2.3 Noise Variance as Dimension Selection Metric 119
2.3.1 Background and Research Objective 119
2.3.2 Proposed Method 125
2.3.3 Setup of Experiment 1.2 129
2.3.4 Results of Experiment 1.2 138
2.3.5 Conclusion 154
2.4 Sequential Approach to Building the Feature Space 157
2.4.1 Background and Research Objective 157
2.4.2 Proposed Method 159
2.4.3 Setup of Experiment 1.3 165
2.4.4 Results of Experiment 1.3 169
2.4.5 Conclusion 187
2.5 Conclusion 190
2.5.1 Summary 190
2.5.2 Contributions 192
2.5.3 Next Steps 193

Chapter 3: Multi-Fidelity Extension 194

3.1 Multi-Fidelity Extension . . . 194
3.1.1 Background and Research Objective . . . 194
3.1.2 Proposed Method . . . 200
3.1.3 Setup of Experiment 2.1 . . . 205
3.1.4 Results of Experiment 2.1 . . . 220
3.1.5 Conclusion . . . 227
3.2 Different Low- and High-Fidelity Feature Spaces . . . 228
3.2.1 Background and Research Objective . . . 228
3.2.2 Formulation of the Alternative Approaches . . . 230
3.2.3 Setup of Experiment 2.2 . . . 233
3.2.4 Results of Experiment 2.2 . . . 236
3.2.5 Conclusion . . . 239
3.3 Conclusion . . . 240
3.3.1 Summary . . . 240
3.3.2 Contributions . . . 242
3.3.3 Next Steps . . . 242

Chapter 4: Sampling Strategies Leveraging the Feature Space . . . 243

4.1 Design of Experiment Leveraging the Feature Space . . . 245
4.1.1 Background and Research Objective . . . 245
4.1.2 Proposed Method . . . 247
4.1.3 Setup of Experiment 3.1 . . . 250
4.1.4 Results of Experiment 3.1 . . . 254

4.1.5 Conclusion 261
4.2 Adaptive Sampling Leveraging the Feature Space 262
4.2.1 Background and Research Objective 262
4.2.2 Proposed Method 264
4.2.3 Setup of Experiment 3.2 266
4.2.4 Results of Experiment 3.2 272
4.2.5 Conclusion 277
4.3 Conclusion 282
4.3.1 Summary 282
4.3.2 Contributions 283
4.3.3 Next steps 284

Chapter 5: Validation of the Proposed Methodology and Conclusion 285
5.1 Validation of the Proposed Methodology 286
5.1.1 Proposed Methodology 286
5.1.2 Setup of Experiment 4 297
5.1.3 Results of Experiment 4 301
5.1.4 Conclusion 304
5.2 Conclusion 305
5.2.1 Thesis Summary 305
5.2.2 Contributions 308
5.2.3 Summary of Findings 309
5.2.4 Future Work 311

Appendices 315

Appendix A: Additional Results for the Noise Variance Study 316

Appendix B: Additional Results for the Sequential Approach Study 344

References 387

LIST OF TABLES

2.1 Summary of benchmark datasets 88

2.2 Experiment 1.1 – Summary of the parameters varied in the parametric study 94

2.3 Summary of test datasets used in experiment 1.2 135

2.4 Experiment 1.2 – Summary of the parameters varied in the parametric study 136

2.5 Complete List of Alternatives 163

2.6 Experiment 1.3 – Summary of the parameters varied in the parametric study 167

3.1 Experiment 2.1 – Summary of the parameters varied in the parametric study 217

3.2 Experiment 2.1 – feature space (FS) dimension considered in the parametric study for each dataset. 218

3.3 Experiment 2.1 – Analysis budget (in high-fidelity (HF) samples) as a function of the analysis input dimension for the elliptic PDE datasets. 218

3.4 Experiment 2.1 – Analysis budget (in HF samples) as a function of the analysis input dimension for the RAE 2822 datasets. 219

3.5 Experiment 2.2 – Summary of the parameters varied in the parametric study 235

3.6 Experiment 2.2 – Analysis budget (in HF samples) as a function of the analysis input dimension. 236

4.1 Experiment 3.1 – Summary of the parameters varied in the parametric study 253

4.2 Experiment 3.1 – Analysis budget (in HF samples) as a function of the analysis input dimension. 254

4.3 Experiment 3.2 – Summary of the parameters varied in the parametric study 269

4.4 Experiment 3.2 – Analysis budget (in HF samples) as a function of the analysis input dimension. 269

4.5 Experiment 3.2 – HF adaptive budget fraction as a function of the HF design of experiments (DOE) budget fraction. 270

5.1 Summary of the parametric study conducted in experiment 4 299

5.2 Experiment 4 – Analysis budget (in HF samples) as a function of the analysis input dimension. 300

LIST OF FIGURES

1.1 Notional illustration of many-query applications nesting to achieve a particular task 7

1.2 Notional workflow diagrams for four of the main surrogate modeling scenarios 25

1.3 Illustration of the partition of the $[0, 1]^d$ hypercube into smaller hypercubes of side length $1/m$ when $d = 2$ and $m = 5$. 27

1.4 Evolution the volume of individual hypercubes (top, left axis), the minimum distance between sampling locations (top, right axis), and the number of hypercubes required to cover $[0, 1]^d$, as a function of the number d of dimensions when $m = 10$ segments are used for each dimension. 28

2.1 Generic process for the first considered surrogate modeling scenario, single-fidelity surrogate modeling. 37

2.2 Inapplicability of unsupervised dimension reduction – example response . . 40

2.3 Inapplicability of unsupervised dimension reduction – uniformly distributed inputs 41

2.4 Example of application of principal components analysis (PCA) on a correlated input distribution. This time, there is a principal component, that captures the dominant direction in which *inputs* are varying. 42

2.5 Projection of the function values on 1) the first principal component (left) and 2) the second principal component (right). Selecting the component using PCA would lead to trying to creating a surrogate for the noisy observations on the left, leading to a poorer model than if the observations on the right had been used. 43

2.6 3D view and expression of the analytical function used for illustration purposes (left) and corresponding contour plot with gradients represented as red arrows (right). 43

2.7 The application of PCA on the *gradient samples* yields the active subspace (AS). 44

2.8 The directions obtained when applying Active Subspace are much more suitable to being used for regression than those obtained using PCA 45

2.9 Illustration of Bayesian linear regression 64

2.10 Markov-Chain Monte-Carlo chains of the model parameters 97

2.11 Values of the split Gelman-Rubin statistics of the Markov chain Monte-Carlo (MCMC) chains as a function of the FS dimension m for two fixed numbers of training samples (a, b), and as a function of the number n of training samples for two fixed FS dimensions (c, d). Parameter names are indicated to the left of the plot. 98

2.12 Prior (orange) and posterior (blue) distributions. Parameter names are indicated to the left of the plot. Histograms are normalized such that their respective areas equal 1. 100

2.13 Comparison of the training data (Actual) and the model predictions (Predicted) for the ONERA M6 dataset for four different combinations of FS dimension and number of training samples. Vertical bars indicate the 95% confidence interval. 101

2.14 Training duration as a function of the FS dimension m for two fixed numbers of training samples (a, b), and as a function of the number n of training samples for two fixed FS dimensions (c, d). 102

2.15 Comparison of the actual dataset and the model predictions for validations points, i.e. observations not used during training. Comparisons are shown for four different combinations of FS dimension and number of training samples. Vertical bars indicate the 95% confidence interval. 103

2.16 Values of the coefficient of determination R^2 as a function of the FS dimension m for two fixed numbers of training samples (a, b), and as a function of the number n of training samples for two fixed FS dimensions (c, d). . . 104

2.17 Values of the mean log pointwise predictive density as a function of the FS dimension m for two fixed numbers of training samples (a, b), and as a function of the number n of training samples for two fixed FS dimensions (c, d). 105

2.18 Evolution of the mean first subspace angle (MFSA) between the predicted and actual active subspaces for training sets varying in size n from one to five times the number of input dimensions for quadratic function datasets. Plots are organized by increasing input dimension d from top to bottom and by increasing AS dimension m from left to right. 107

2.19 Evolution of the validation coefficient of determination (R^2) for training sets varying in size n from one to five times the number of input dimensions for quadratic function datasets. Plots are organized by increasing input dimension d from top to bottom and by increasing AS dimension m from left to right. 109

2.20 Evolution of the mean log pointwise predictive density (MLPPD) for training sets varying in size n from one to five times the number of input dimensions for quadratic function datasets. Plots are organized by increasing input dimension d from top to bottom and by increasing AS dimension m from left to right. 111

2.21 Evolution of the training time (TT) for training sets varying in size n from one to five times the number of input dimensions for quadratic function datasets. Plots are organized by increasing input dimension d from top to bottom and by increasing AS dimension m from left to right. 113

2.22 Evolution of the first subspace angle (FSA) between the predicted and actual active subspaces for training sets varying in size n from one to five times the number of input dimensions for all four science and engineering datasets. 114

2.23 Evolution of the validation coefficient of determination (R^2) for training sets varying in size n from one to five times the number of input dimensions for all four science and engineering datasets. 115

2.24 Evolution of the mean log pointwise predictive density (MLPPD) for training sets varying in size n from one to five times the number of input dimensions for all four science and engineering datasets. 116

2.25 Evolution of the training time (TT) for training sets varying in size n from one to five times the number of input dimensions for all four science and engineering datasets. 117

2.26 Notional illustration of the expected noise variance decay and corresponding increase in coefficient of determination (R^2) 137

2.27 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 25 input variables and a 1D feature space . . 140

2.28 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 25 input variables and a 2D feature space . . 141

2.29 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 25 input variables and a 5D feature space . . 142

2.30 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the ONERA M6 drag dataset with 50 input variables 145

2.31 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the Elliptic PDE (y_{short}) dataset with 100 input variables 147

2.32 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the NACA0012 lift dataset with 18 inputs 148

2.33 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the CRM subsonic drag dataset with 50 input variables 150

2.34 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the CRM subsonic z-moment dataset with 50 input variables 152

2.35 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the RAE2822 drag dataset with 51 input variables 153

2.36 Comparison of the MCMC chains for selected model parameters obtained for the CRM subsonic drag dataset ($RS = 867$) for different FS dimensions (top: 1D, middle: 5D, bottom: 10D) and number of training samples (left: 100, right: 250). 155

2.37 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 25 input variables and a 1D feature space . . 170

2.38 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 25 input variables and a 1D feature space 171

2.39 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 25 input variables and a 2D feature space . . 172

2.40 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 25 input variables and a 2D feature space 173

2.41 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 25 input variables and a 5D feature space . . 174

2.42 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 25 input variables and a 5D feature space 175

2.43 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the ONERA M6 drag dataset with 50 input variables 177

2.44 Evolution of the training duration as a function of the number of training samples for the ONERA M6 drag dataset with 50 input variables 178

2.45 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the Elliptic PDE (y_{short}) dataset with 100 input variables 179

2.46 Evolution of the training duration as a function of the number of training samples for the Elliptic PDE (y_{short}) dataset with 100 input variables 180

2.47 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the NACA0012 lift dataset with 18 inputs 181

2.48 Evolution of the training duration as a function of the number of training samples for the NACA0012 lift dataset with 18 inputs 182

2.49 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the CRM subsonic drag dataset with 50 input variables 184

2.50 Evolution of the training duration as a function of the number of training samples for the CRM subsonic drag dataset with 50 input variables 185

2.51 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the CRM subsonic z-moment dataset with 50 input variables 186

2.52 Evolution of the training duration as a function of the number of training samples for the CRM subsonic z-moment dataset with 50 input variables . . 187

2.53 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the RAE2822 drag dataset with 51 input variables 188

2.54 Evolution of the training duration as a function of the number of training samples for the RAE2822 drag dataset with 51 input variables 189

3.1 Generic process for the second considered surrogate modeling scenario, multi-fidelity surrogate modeling. 195

3.2 Probabilistic graphical model for the proposed multi-fidelity model to approximation by ridge function. 202

3.3 Comparison of the high- and low-fidelity (LF) data for the Elliptic PDE dataset with $\beta = 1.0$ and input dimensions 10, 25, 50, and 100. 209

3.4 Comparison of the high- and LF data for the Elliptic PDE dataset with $\beta = 0.01$ and input dimensions 10, 25, 50, and 100. 210

3.5 Comparison of the high- and LF data for the RAE 2822 dataset with input dimensions 15, 25, and 51. 212

3.6 Multi-fidelity extension – R^2 vs. LF allocation ration for elliptic PDE with $\beta = 0.01$ 221

3.7 Multi-fidelity extension – R^2 vs. LF allocation ration for elliptic PDE with $\beta = 1.0$ 224

3.8 Multi-fidelity extension – R^2 vs. LF allocation ration for RAE2822 dataset . 226

3.9 Probabilistic graphical model for *different* alternative. 230

3.10 Probabilistic graphical model for the *related* alternative. 231

3.11 LF-HF FS relation – R^2 vs. LF allocation ration – elliptic PDE dataset with $\beta = 0.01$ 237

3.12 LF-HF FS relation – R^2 vs. LF allocation ration – elliptic PDE dataset with $\beta = 1.0$ 238

4.1 Generic process for the third considered surrogate modeling scenario, adaptive sampling in the multi-fidelity context. 244

4.2 FS-based DOE (full budget) – R^2 comparison – elliptic PDE dataset with $\beta = 0.01$ and 3D FS 256

4.3 FS-based DOE (full budget) – R^2 comparison – elliptic PDE dataset with $\beta = 0.01$ and 5D FS 257

4.4 FS-based DOE (full budget) – R^2 comparison – elliptic PDE dataset with $\beta = 1.0$ and 3D FS 259

4.5 FS-based DOE (full budget) – R^2 comparison – elliptic PDE dataset with $\beta = 1.0$ and 5D FS 260

4.6 Adaptive sampling – R^2 vs. adaptive sampling budget fraction – elliptic PDE dataset with $\beta = 0.01$ and 3D FS 273

4.7 Adaptive sampling – R^2 vs. adaptive sampling budget fraction – elliptic PDE dataset with $\beta = 0.01$ and 5D FS 274

4.8 Adaptive sampling – R^2 vs. adaptive sampling budget fraction – elliptic PDE dataset with $\beta = 1.0$ and 3D FS 275

4.9 Adaptive sampling – R^2 vs. adaptive sampling budget fraction – elliptic PDE dataset with $\beta = 1.0$ and 5D FS 276

4.10 Adaptive sampling – R^2 vs. spent HF budget fraction for different HF adaptive budget fractions – elliptic PDE dataset with $\beta = 0.01$ and 3D FS . 278

4.11 Adaptive sampling – R^2 vs. spent HF budget fraction for different HF adaptive budget fractions – elliptic PDE dataset with $\beta = 0.01$ and 5D FS . 279

4.12 Adaptive sampling – R^2 vs. spent HF budget fraction for different HF adaptive budget fractions – elliptic PDE dataset with $\beta = 1.0$ and 3D FS . . 280

4.13 Adaptive sampling – R^2 vs. spent HF budget fraction for different HF adaptive budget fractions – elliptic PDE dataset with $\beta = 1.0$ and 5D FS . . 281

5.1 Flowchart for the proposed method 287

5.2 Comparison of R^2 Between Deep multi-fidelity (MF) Gaussian process (GP) and Proposed Approach – elliptic PDE dataset with $\beta = 0.01$ 302

5.3 Comparison of R^2 Between Deep MF GP and Proposed Approach – elliptic PDE dataset with $\beta = 1.0$ 303

A.1 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 10 input variables and a 1D feature space . . 317

A.2 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 10 input variables and a 2D feature space . . 318

A.3 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 10 input variables and a 5D feature space . . 319

A.4 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 50 input variables and a 1D feature space . . 320

A.5 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 50 input variables and a 2D feature space . . 321

A.6 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 50 input variables and a 5D feature space . . 322

A.7 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 100 input variables and a 1D feature space . 323

A.8 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 100 input variables and a 2D feature space . 324

A.9 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 100 input variables and a 5D feature space . 325

A.10 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the Elliptic PDE (y_{long}) dataset with 100 inputs 327

A.11 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the HIV (y_{3400}) dataset with 27 inputs 328

A.12 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the NACA0012 drag dataset with 18 inputs 329

A.13 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the ONERA M6 lift dataset with 50 inputs 330

A.14 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the subsonic CRM lift dataset with 50 inputs 332

A.15 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the subsonic CRM x-moment dataset with 50 inputs 333

A.16 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the subsonic CRM y-moment dataset with 50 inputs 334

A.17 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the subsonic CRM sideforce dataset with 50 inputs 335

A.18 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM drag dataset with 50 inputs 336

A.19 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM lift dataset with 50 inputs 337

A.20 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM x-moment dataset with 50 inputs 338

A.21 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM y-moment dataset with 50 inputs 339

A.22 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM z-moment dataset with 50 inputs 340

A.23 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM sideforce dataset with 50 inputs 341

A.24 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the RAE2822 lift dataset with 51 inputs 342

A.25 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the RAE2822 z-moment dataset with 51 inputs 343

B.1 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 10 input variables and a 1D feature space . . 345

B.2 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 10 input variables and a 2D feature space . . 346

B.3 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 10 input variables and a 5D feature space . . 347

B.4 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 50 input variables and a 1D feature space . . 348

B.5 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 50 input variables and a 2D feature space . . 349

B.6 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 50 input variables and a 5D feature space . . 350

B.7 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 100 input variables and a 1D feature space . 351

B.8 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 100 input variables and a 2D feature space . 352

B.9 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the quadratic test function with 100 input variables and a 5D feature space . 353

B.10 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 10 input variables and a 1D feature space 354

B.11 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 10 input variables and a 2D feature space 355

B.12 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 10 input variables and a 5D feature space 355

B.13 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 50 input variables and a 1D feature space 356

B.14 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 50 input variables and a 2D feature space 356

B.15 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 50 input variables and a 5D feature space 357

B.16 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 100 input variables and a 1D feature space 357

B.17 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 100 input variables and a 2D feature space 358

B.18 Evolution of the training duration as a function of the number of training samples for the quadratic test function with 100 input variables and a 5D feature space 358

B.19 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the Elliptic PDE (y_{long}) dataset with 100 inputs 360

B.20 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the HIV (y_{3400}) dataset with 27 inputs 361

B.21 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the NACA0012 drag dataset with 18 inputs 362

B.22 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the ONERA M6 lift dataset with 50 inputs 363

B.23 Evolution of the training duration as a function of the number of training samples for the Elliptic PDE (y_{long}) dataset with 100 inputs 364

B.24 Evolution of the training duration as a function of the number of training samples for the HIV (y_{3400}) dataset with 27 inputs 365

B.25 Evolution of the training duration as a function of the number of training samples for the NACA0012 drag dataset with 18 inputs 365

B.26 Evolution of the training duration as a function of the number of training samples for the ONERA M6 lift dataset with 50 inputs 366

B.27 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the subsonic CRM lift dataset with 50 inputs 368

B.28 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the subsonic CRM x-moment dataset with 50 inputs 369

B.29 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the subsonic CRM y-moment dataset with 50 inputs 370

B.30 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the subsonic CRM sideforce dataset with 50 inputs 371

B.31 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM drag dataset with 50 inputs 372

B.32 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM lift dataset with 50 inputs 373

B.33 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM x-moment dataset with 50 inputs 374

B.34 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM y-moment dataset with 50 inputs 375

B.35 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM z-moment dataset with 50 inputs 376

B.36 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the transonic CRM sideforce dataset with 50 inputs 377

B.37 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the RAE2822 lift dataset with 51 inputs 378

B.38 Evolution of σ_n^2 and R^2 as a function of the number of FS dimensions for the RAE2822 z-moment dataset with 51 inputs 379

B.39 Evolution of the training duration as a function of the number of training samples for the subsonic CRM lift dataset with 50 inputs 380

B.40 Evolution of the training duration as a function of the number of training samples for the subsonic CRM x-moment dataset with 50 inputs 381

B.41 Evolution of the training duration as a function of the number of training samples for the subsonic CRM y-moment dataset with 50 inputs 381

B.42 Evolution of the training duration as a function of the number of training samples for the subsonic CRM sideforce dataset with 50 inputs 382

B.43 Evolution of the training duration as a function of the number of training samples for the transonic CRM drag dataset with 50 inputs 382

B.44 Evolution of the training duration as a function of the number of training samples for the transonic CRM lift dataset with 50 inputs 383

B.45 Evolution of the training duration as a function of the number of training samples for the transonic CRM x-moment dataset with 50 inputs 383

B.46 Evolution of the training duration as a function of the number of training samples for the transonic CRM y-moment dataset with 50 inputs 384

B.47 Evolution of the training duration as a function of the number of training samples for the transonic CRM z-moment dataset with 50 inputs 384

B.48 Evolution of the training duration as a function of the number of training samples for the transonic CRM sideforce dataset with 50 inputs 385

B.49 Evolution of the training duration as a function of the number of training samples for the RAE2822 lift dataset with 51 inputs 385

B.50 Evolution of the training duration as a function of the number of training samples for the RAE2822 z-moment dataset with 51 inputs 386

LIST OF ACRONYMS

R^2 coefficient of determination

AIC Akaike information criterion

ANN artificial neural network

ANOVA analysis of variance

ARD automatic relevance determination

ARGP autoregressive Gaussian process

AS active subspace

ASDL aerospace systems design laboratory

B-GP Bayesian Gaussian process

B-R-DMF-GP Bayesian ridge deep multi-fidelity Gaussian process

B-R-GP Bayesian ridge Gaussian process

BIC Bayesian information criterion

BO Bayesian optimization

CCA canonical correlation analysis

CFD computational fluid dynamics

DMF-GP deep multi-fidelity Gaussian process

DOE design of experiments

DSE design space exploration

DSM design-structure matrix

EI expected improvement

FFD free-form deformation

FS feature space

GP Gaussian process

GPR Gaussian process regression

HDMR high-dimensional model representation

HF high-fidelity

HMC Hamiltonian Monte-Carlo

HPC high-performance computing

i.i.d. independent and identically distributed

IC information criterion

IMSE integrated mean square error

LDA linear discriminant analysis

LF low-fidelity

LHS Latin-hypercube sampling

MAE maximum absolute error

MC Monte-Carlo

MCMC Markov chain Monte-Carlo

MDA multi-disciplinary analysis

MDAO multi-disciplinary analysis and optimization

MF multi-fidelity

MFSA mean first subspace angle

MLE maximum likelihood estimation

MLPPD mean log pointwise predictive density

MO-AS manifold optimization-based active subspace

NN neural network

NUTS no-U-turn sampler

PCA principal components analysis

PCE polynomial chaos expansion

PCR principal components regression

PDE partial differential equation

PGM probabilistic graphical model

PLS partial least squares

POD proper orthogonal decomposition

PPR projection pursuit regression

RANS Reynolds-Averaged Navier-Stokes

RBF radial basis function

RDP relative distance plane

RHMC Riemannian Hamiltonian Monte-Carlo

RMSE root-mean-square error

ROM reduced-order model

SA Spalart-Allmaras

SVD singular value decomposition

SVI stochastic variational inference

TD training duration

UQ uncertainty quantification

WAIC widely applicable information criterion

SUMMARY

Modern approaches to engineering design rely on decision-support tools such as design space exploration, engineering optimization, or uncertainty quantification, to make better-informed design decisions. Such approaches typically rely on physics-based analyses that model the aspects of the system-of-interest that are relevant to the design task. As they operate by repeatedly evaluating their underlying analyses, carrying out these so-called "many-query applications" may become prohibitively expensive. Surrogate models act as enablers by replacing the online cost of evaluating analyses with a smaller offline cost spent to gather data used to train a free-to-evaluate mathematical model. Two current trends however make the generation of surrogate models more challenging and may therefore hinder the application of modern approaches. First, analyses of higher fidelity and greater computational cost are increasingly used to gather more detailed and accurate design knowledge early on in the design process, leading to the availability of fewer training observations under a constant analysis budget. Second, higher-dimensional parameter spaces are being considered, for example motivated by a more thorough exploration of the design space, the investigation of novel vehicle configurations, or the desire to retain design freedom longer, leading to surrogate models with high-dimensional inputs whose training suffers from the curse of dimensionality. In this thesis, we propose to investigate methods that address the impacts of these two trends on the generation of surrogate models: we seek new methods better suited for the creation of surrogate models with high-dimensional inputs and using only relatively few training observations. In particular, we focus on three surrogate modeling scenarios that map to the three research areas structuring this thesis: 1) single-fidelity surrogate modeling, 2) multi-fidelity surrogate modeling, and 3) active sampling in the multi-fidelity context.

The methods proposed in this thesis rely on approximation by ridge functions to alleviate the curse of dimensionality. It consists in first projecting the original high-dimensional

inputs onto a low-dimensional feature space, followed by a traditional regression. Accordingly, training such approximations consists in 1) determining a relevant projection, and 2) training the regression model. Multiple contributions were made in this thesis, starting in the single-fidelity context with a fully Bayesian and gradient-free formulation of approximation by ridge functions. Compared to existing approaches, the proposed method enables a full quantification of epistemic uncertainty due to limited training data, both in the regression parameters as well as the low-dimensional projection. Through a thorough study conducted on multiple datasets originating from science and engineering applications, it was shown to outperform existing state-of-the-art methods. Alternate methods for determining the dimension of the low-dimensional feature space, that aim to address shortcomings of existing methods, were then proposed and assessed. These advancements were then brought to the multi-fidelity context by altering a deep multi-fidelity Gaussian process model to include an initial projection of its inputs and a fully Bayesian approach to its training. Under certain conditions, this approach was shown to make better use of a given analysis budget compared to relying on a single fidelity. The relationship between the projections used for the low- and high-fidelity parts of the model was then investigated. Two approaches to sampling leveraging the feature space were formulated and assessed. The proposed approach to experimental design for selecting the location of high-fidelity observations was shown to outperform a traditional design of experiments in the original input space, but the proposed active sampling approach did not yield any additional improvement. Finally, we assembled a coherent approach to multi-fidelity modeling that leverages the knowledge of the low-dimensional feature space to assist the selection of expensive, high-fidelity observations and showed that it outperforms the state-of-the-art deep multi-fidelity Gaussian process method.

CHAPTER 1
MOTIVATION AND DEFINITION OF RESEARCH OBJECTIVES

1.1 Foreword

1.1.1 Document Organization

This section presents the organization of the document as a whole.

In the current chapter, we start by introducing the overall problem being addressed, leading to the overarching research question that motivated the development of the different methods proposed in this thesis. Starting from this broad objective, we refine the scope of the thesis in the subsequent sections. Defining the scope then allows us to introduce the three research areas that are the focus of this thesis along with the corresponding research questions.

The following chapters focus on the three research areas that make up this thesis:

1. developing a fully Bayesian approach to approximation by ridge functions and detecting the dimension of the feature space (chapter 2);

2. extending the proposed approach to the multi-fidelity context (chapter 3);

3. leveraging the feature space to assist design of experiments and adaptive sampling (chapter 4).

In the final chapter 5, we first assemble a coherent approach to the creation of multi-fidelity surrogate models with high-dimensional inputs. This method is the culmination of the research efforts undertaken in this thesis, as it leverages nearly all individual developments previously presented. We then compare it to a state-of-the-art method. Finally, we summarize the contributions made in this work and propose avenues for related future work.

The sections of the chapters 2 to 5 all follow a common structure. Starting from the research question of the corresponding research area, we first aim at identifying a capability gap. This is done by contrasting capability needs with existing methods identified through a literature review. The gap prompts a new research question that is more focused than the original research question. We return to literature in search of options for solving the new research question, and a testable hypothesis is formulated as to which option is most appropriate to reach the desired goal. An experiment is designed to test this hypothesis, and the corresponding results are presented. Based on these results, the validity of the hypothesis is discussed.

This dissertation is a relatively long document. The details provided would hopefully allow an interested practitioner to fully understand the underlying line of thought and, along with the openly available code repositories, to reproduce all experimental results. Other readers may however rightfully wish for a more cursory reading, and we sought to offer them this option. To this end, the summary section that concludes every chapter aims at reproducing the logical flow of the chapter in a condensed format. Reading these summaries along with the final chapter of the dissertation should hopefully allow to develop a good understanding of this work as a whole within a reasonable amount of time.

1.1.2 Section Organization

This chapter aims at exposing the motivation for the methods developed in this thesis.

In section 1.2, we start with an overview of the motivation for this thesis. We start by recalling the importance of surrogate modeling in modern approaches to engineering design (sections 1.2.1 to 1.2.3). Then, we introduce two situations that make training surrogate models particularly challenging: the sparsity of training data when analyses are expensive to evaluate (section 1.2.4) and high-dimensional inputs (section 1.2.5). Based on these discussions, we introduce the main gap motivating a more detailed review of existing methods in section 1.2.6.

Having a better idea of the motivation, section 1.3 aims at refining the scope of the thesis. Starting from surrogate modeling in general (sections 1.3.1 to 1.3.3), we then focus on surrogate modeling methods specifically tailored to situations in which the analysis budget is limited, resulting in few available training observations: multi-fidelity methods (section 1.3.4) and active sampling (section 1.3.6). This leads in section 1.3.7 to the presentation of the three surrogate modeling scenarios that define the scope of this thesis.

Finally, in section 1.4, we discuss the curse of dimensionality (section 1.4.1) and more specifically its impact on the three surrogate modeling scenarios under the scope of this research (section 1.4.2). The three main research areas follow from this discussion (section 1.4.3). Finally, section 1.5 contains a summary of this chapter.

1.2 Motivation

This section gives a high-level presentation of the motivation for this thesis in order to briefly introduce the challenges that it aims to address. Here, we consider the general problem of surrogate modeling, and the scope will be refined to three more specific surrogate modeling scenarios in the following section. These later sections will also more thoroughly present some of the topics alluded to in this section and thus pinpoint issues to be addressed in a more specific manner.

1.2.1 Context: Aerospace conceptual design

The aim of this section is to recall the main objectives of conceptual design, along with the means employed to reach them. In particular, having a clear view of the central role held by many-query applications within this process is primordial to understanding the necessity to leverage surrogate modeling.

Conceptual design is a complex decision-making process that aims at progressively defining the main system characteristics such that the resulting design satisfies top-level design requirements. The system characteristics resulting from the conceptual design phase

are then used to derive more detailed requirements that drive later design stages, and as such have extensive cost implications [1, 2, 3, 4]. Because early decisions will directly impact all subsequent design stages, it is essential to ensure that all conceptual design decisions are well-informed.

There is no unique definition of the outcome expected from conceptual design. In fact, [4] even states that main design characteristics are determined ”within a fuzzy latitude”. [1] defines the outcome of conceptual design as the ”single best concept” out of ”a number of possible solutions”. [2] underlines the fact that this outcome is meant to be passed on to later stages of design: “The end product of all this will be an aircraft design that can be confidently passed to the preliminary design phase [...], major revisions will not occur if the conceptual design effort has been successful.” They also introduce the idea that revisions, also referred to as reworks or design iterations [1], should not be necessary if conceptual design was successfully carried out.

The outcome of conceptual design is the result of a series of standard problem-solving methodology [1]:

1. Definition of the problem
2. Gathering of information
3. Generation of alternative solutions
4. Evaluation of alternatives and decision-making
5. Communication of the result

While steps were enumerated in series, the resulting process is rarely linear but rather iterative, which results in certain of these steps being repeated. As Raymer puts it: ”Design is Iterative - You Never Build the Dash-One” [2]. In the context of aerospace design, evaluating an alternative involves the following activities [2]: ”initial analysis (Aerodynamics, Weights, Propulsion, Stability & Control, Structures, Cost, Subsystems, etc.)” and ”sizing

& performance optimization". We will have a closer look at the activities carried out during conceptual design in the following paragraphs.

1.2.2 Many-Query Applications

Decision-support tools such as multi-disciplinary analysis and optimization, design space exploration, sensitivity analysis, and uncertainty quantification play a crucial role in informing the decisions made throughout conceptual design.

Multidisciplinary Analysis Analyses carried out in the context of aerospace design are multidisciplinary in essence, i.e., the different disciplines can sometimes be coupled at a very fundamental level. For example, the tight coupling necessary to model flexible wings associates structures and aerodynamics, leading to the study of aeroelasticity [5]. Even when they are not fundamentally coupled, the fact that all disciplines operate on the same design usually creates coupling. As an example, all subsystems contribute to the total vehicle weight, but their own design (e.g., the wing surface area) may be influenced by the total vehicle weight, thus creating a circular dependency.

This leads to the notion of multi-disciplinary analysis (MDA) [6]. Some disciplines need to satisfy the laws of physics at the subsystem level to be in a "valid" state. For example, in the context of aeroelasticity, the results obtained for the wing loads and deflections do not have a physical meaning as long as the converged solution has not been reached. In general, all coupling variables between disciplines need to match for the complete system to be in a "valid" state. The process of matching coupling variables is iterative [6]. As a result, underlying analyses need to be run many times until convergence is met.

Multidisciplinary Analysis and Optimization Optimization is also at the heart of conceptual design. When optimization is carried out in a multi-disciplinary setting, it is referred to as multi-disciplinary analysis and optimization (MDAO) [2, 6]. In addition to the coupling between disciplines being matched for the overall system to be in a valid state, ob-

jectives must be optimized under specified constraints. If optimization leverages an MDA, the number of cumulated evaluations of the underlying models rises sharply. Many architectures have been developed over the years to carry out MDAO in an organization context [6, 7, 8, 9, 10, 11, 12].

Design Space Exploration Design trade studies are also a big part of conceptual design, as they are important decision-making support tools, and decisions are ultimately made by stakeholders. [2] lists three main categories:

- Design trades: what is the impact of a change in a design variable on a performance or mission requirement?
- Requirements trades: how is the design affected by changes in requirements?
- Growth sensitivities: what is the impact on parameters on the aircraft weight?

Parametric studies are widely used as they allow to observe all these effects in real-time [13].

Uncertainty Quantification Robust design is an example of a design approach leveraging uncertainty quantification (UQ). In addition to meeting requirements, robust design dictates that the system's performance should be robust to different factors, such as variations in operations conditions, manufacturing variability, or lack of knowledge about certain aspects of the design [14, 15, 16]. In order to achieve this, sources of uncertainty, whether aleatory or epistemic, need to be identified and their corresponding level of uncertainty quantified. UQ is the process by which the uncertainty originating from these different sources is propagated to the system responses of interest [14].

Monte-Carlo methods are usually employed for this step. Estimation error in traditional Monte-Carlo methods presents a slow rate of decay [15], thus requiring a large number of forward model evaluations. Similarly to MDAO, UQ may be carried out on top of another many-query application. For example, UQ may leverage an underlying MDA or even MDAO process , making it a particularly costly process.

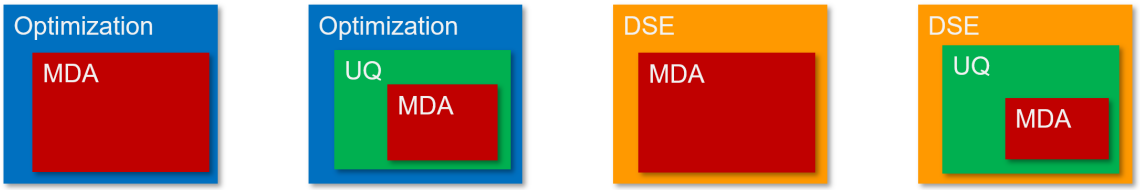


Figure 1.1: Notional illustration of many-query applications nesting to achieve a particular task

Observation An activity carried out as part of the design process may combine multiple of these many-query applications, as conceptually illustrated in fig. 1.1. For example, in MDAO, individual disciplinary analyses are wrapped together within an MDA, which provide the objective and constraint functions then used by an optimization process.

As indicated by the "many-query" denomination, an aspect that all these activities have in common is the large number of times that underlying analyses need to be evaluated. In optimization, the analysis is evaluated along the optimization path. For gradient-based methods, when gradients are not accessible through another method (e.g., analytically or using the adjoint), additional evaluations may additionally be necessary to estimate gradient values. In MDA, the convergence of the system is ensured by a mathematical solver that keeps evaluating underlying analyses until pre-defined convergence criteria are met. In UQ, analyses may be evaluated multiple times with different input settings in order to estimate statistics of the outputs, as in Monte-Carlo methods, or may be queried many times to carry out Bayesian inference of model parameters.

This discussion leads to the following observations:

Observations

- Many-query applications such as MDA, MDAO, or UQ, and real-time design space exploration (DSE) trade studies are essential to informing decision-making during conceptual design.
- Many-query applications require a large number of evaluations of the underlying analyses.

1.2.3 The Need for Surrogate Modeling

Physical experiments and numerical analyses underlie all many-query applications, such that decisions are eventually informed based on physical or simulated data. Performing the experiment or running a simulation incurs a cost. Time, computational resources (in the numerical case), or actual monetary budget and material resources (in the case of physical experiments), need to be allocated.

Conceptual design is not carried out in a vacuum: it must be considered in the context of an industrial organization that operates under constraints and that needs to reduce cycle time to face an increasingly competitive environment [1]. As a result, in a realistic organizational context, material and computational resources, as well as time allocated to a project, are limited and the use of organizational resources needs to be informed in order to make best use of the limited available resources.

Surrogate models allow to limit the cost of running these many-query applications by substituting a *one-time offline cost* to a *recurring online cost*. The underlying analyses are evaluated in order to gather a number of observations allowable under the analysis budget fixed by organizational constraints. Then, these observations are used to create a mathematical model that mimics the input-output relationship of the analysis. As we will see in the next section, there are many ways to achieve this in practice, with methods varying in complexity, flexibility, features, and cost. Once the surrogate model has been created, it

can replace the original analysis when conducting many-query applications. Thanks to its mathematical nature, additional evaluations of the surrogate model incur no or negligible computational cost.

This leads to the following observations:

Observations

- By replacing physical models with cheap-to-evaluate mathematical functions, surrogate models allow the practical implementation of many-query applications.
- Surrogate models are essential enablers to modern approaches to engineering design.

1.2.4 Sparse Training Data

High-fidelity numerical simulations have been increasingly employed in the design process. Thanks to simultaneous improvements in physical modeling and computing power, numerical analyses have reached a high degree of fidelity and practicality [1, 17, 18, 19]. An increasing number of unconventional designs are being investigated [20], that call for the use of physics-based models to compensate for the lack of available historical data, while allowing a broader exploration of the parameter space. That fact is compounded by the paradigm shift towards design for affordability [13]. In order to achieve affordability through the reduction of design reworks, a higher confidence in the outcome of conceptual design is required, which motivates the usage of higher-fidelity models. As a consequence, high-fidelity computer analyses have now become central to conceptual design, while they had previously been confined to the preliminary and detailed stages [18]. As the fidelity of the analyses is directly correlated to their computational cost [17, 18, 19], the number of analysis observations that can be gathered under a fixed analysis budget may be severely limited. As we will see in the next chapter, sparse training data is a challenge to the creation of accurate surrogate models.

This leads to the following observations:

Observations

- Expensive-to-evaluate computer and physical experiments are increasingly employed during the design process.
- As a result, only a relatively low number of analysis observations can be obtained offline to train a surrogate model when analysis budget is limited.

1.2.5 High-Dimensional Inputs

In the conceptual stage of design, the main features of the design are still being defined, which results in a large design space. [18] differentiates *model complexity*, that defines the level of detail with which the system under consideration is modeled and *analysis complexity*, that refers to the fidelity of the analysis being carried out on the system-of-interest. In practice, these two notions of complexity are correlated: as analysis fidelity increases, so do the dimensions of their parameter and response spaces. Finally, the increasing level of detail in conceptual design, that was previously reserved to preliminary and detailed design, opens up dimensions that would otherwise have only be considered later. As a practical consequence, the independent parameter spaces used to carry out many-query applications have increasingly high dimensions.

Observation

Increasingly more independent parameters are being considered simultaneously in the context of many-query applications, requiring surrogate models with high-dimensional inputs.

The so-called *curse of dimensionality* remains a major obstacle in the way of developing surrogate models with high-dimensional inputs. The consequences of the curse of dimensionality on surrogate modeling will be exposed in greater details in section 1.2.5.

In a nutshell, in order to get a coverage of the input space similar to what is traditionally expected in low-dimensional spaces to reach sufficient predictive accuracy, the number of training data would need to increase exponentially with the number of dimensions. This quickly becomes impractical under a limited analysis budget. In these conditions, the applicability of traditional sampling methods that aim to have a uniform coverage of the input space is challenged.

Observations

- Models with increasingly high-dimensional parameter spaces are being leveraged in conceptual design.
- The creation of surrogate models with high-dimensional inputs is made challenging by the curse of dimensionality.

1.2.6 Gap Identification

In this section, we started by recalling why surrogate modeling is a enabler to modern approaches to aerospace conceptual design. By modern approaches, we mean the reliance on so-called "many-query applications" that assist decision-making, such as engineering optimization, DSE, or MDA. Leveraging analyses within these applications in an *online* fashion is generally too costly. Instead, analyses are replaced by surrogate models that only incur an *offline* training cost. Without surrogate models, modern approaches used to inform decision-making would generally be impractical.

Then, we identified two trends that are challenging for surrogate modeling. First, increasingly higher-fidelity and expensive analyses are being relied on for conceptual design, which limits the number of training observations that are affordable under a finite analysis budget. Second, increasingly more independent parameters are being considered simultaneously in the context of many-query applications, leading to surrogate models with higher-dimensional inputs that are challenging to train because of the curse of dimension-

ality.

These two challenges compound with each other, making the generation of surrogate models under these conditions particularly hard. High-dimensional inputs calls for more training data, while expensive analyses result in fewer allowable analysis observations. This leads to the identification of the following gap:

Overarching Gap

The inability to create accurate surrogate models of expensive physical models with high-dimensional inputs under a limited computational budget severely hinders the applicability of modern techniques used to inform design decisions.

In the next two sections, we will be taking a closer look at 1) surrogate modeling and 2) the challenges with high-dimensional input spaces. This will first allow us to substantiate this gap in a more specific manner. It will also allow to scope the problem-at-hand, from surrogate modeling in general to three scenarios that are commonly used when working with expensive analyses under a limited analysis budget. By having a more thorough look at the existing literature, we will down-select the classes of methods that are deemed most promising in this situation. These discussions will then allow us to derive the research objective for the thesis.

1.3 Background for Scope Definition

In this section, we conduct a short literature review of surrogate modeling techniques, with an emphasis on strategies used for expensive analyses. This serves to narrow down our scope to three main surrogate modeling scenarios: 1) the generation of a single-fidelity surrogate model, 2) the creation of a multi-fidelity surrogate model, and 3) the use of adaptive sampling in the multi-fidelity context to carefully select observations sites.

1.3.1 Overview of Surrogate Modeling

Terminology Surrogate models lie at the intersection of mathematics and engineering. As a consequence, ambiguous language may sometimes be encountered in literature. In this document, we tried to adhere to the terminology presented in this section.

We abstract the analyses that we are trying to replace with a surrogate model as a mapping f whose domain is X and codomain is Y:

$$\begin{aligned} f : X &\to Y \\ x &\mapsto y \end{aligned} \tag{1.1}$$

X and Y may each be multidimensional spaces. We refer to the set X as the *input* or *parameter space* of f and Y as its *output* or *response space*. Accordingly, we will usually refer to x as the *inputs* or, when it does not create ambiguity, as the *independent parameters* of f. We will avoid the terms *design space* and *design variable* sometimes used in the aerospace design literature on surrogate modeling since 1) they may be ambiguous (they sometimes refer to a space of independent variables and in other instances to a space of dependent ones), and 2) surrogate models might accept inputs that are not strictly design-related, such as flight conditions for an aerodynamic simulation. We will refer to y as the *outputs* or *responses* of f.

Output Dimension The numerical outputs of analyses, and accordingly of their surrogates, may be real numbers or, alternatively, vector quantities. Multiple options exist to treat vector outputs. The simplest option is to treat all outputs as independent. If the output is n-dimensional, this reverts to the problem of creating n independent surrogates whose outputs are real-valued. Other solutions take advantage of the expected correlation between the different components of the vector output, such as to effectively reduce the output dimension to a number m with $m \ll n$. The problem then reverts to creating m surrogate models whose outputs are real numbers for each of the dimensions of the reduced space.

Contributions made to improve surrogate modeling with real-valued outputs can therefore generally be transposed to the context of vector-valued outputs. For this reason, we will be focusing on real-valued surrogate models.

General Taxonomy The following classification of surrogate model methods was introduced in [21] and reused recently by [22, 23]:

- Data-fit/data-based models, discussed in section 1.3.2
- Physics-based reduced-order models (ROMs), discussed in section 1.3.3
- Hierarchical models and multi-fidelity models, discussed in section 1.3.4
- Hybrid approaches

The fourth category was added by [23] to account for recent advances in surrogate modeling. In order to better understand the challenges posed by sparse training data and high-dimensional inputs, we will briefly present some of these classes of methods. The next three sections cover the first three classes. Hybrid methods will not be discussed as they simply consist in combining different types of methods together.

1.3.2 Data-Fit Surrogate Modeling

Overview Data-fit surrogate modeling is a general class of methods that aim at replacing a computer code or capture the responses of a system under experiment with a mathematical model. While they can be used to a variety of ends, such as the transmission of proprietary codes between different organizations, surrogate models are widely used to replace expensive computer or physical experiments with a cheap-to-evaluate mathematical function that can be evaluated a large number of times without incurring a significant computational cost.

Data-fit surrogate models are *de facto* enablers for many-query applications [18, 19]. Even when models have an affordable one-time computational cost, e.g., evaluation time of

the order of the second or minute, evaluating them many times in the context of one of the many-query applications discussed previously may become prohibitive. Data-fit surrogate models mitigate this issue since their evaluation is usually quasi-instantaneous with recent hardware.

Interpolation and Regression When creating surrogate models of deterministic computer codes, interpolation methods are usually preferred to regression, i.e., the resulting surrogate model's predictions perfectly match observations made at training locations. However, if the data is noisy, e.g., a physical quantity measured in-field using a sensor, then regression techniques that account for noise may be preferred. In the context of this thesis, where we mostly focus on numerical analyses, interpolation may first appear as the best route. However, as we will see in chapter 2, the dimension reduction techniques that we propose trade dimensions for some level of noise, making regression methods more suited in that particular case.

Observation

Interpolation methods are usually used when creating data-fit surrogate models of noiseless computer code outputs. However, regression techniques are required as soon as the data becomes noisy.

Local and Global Surrogates Surrogate models may be global or local depending on the end applications [18]. Global surrogates, e.g., radial basis function (RBF) or Gaussian process regression (GPR), may be used in the context of DSE for assisting trade studies, in the context of MDAO to perform global optimization [18], or simply in the context of MDA to speed up convergence. Local surrogate models are mostly used for optimization [19] with the intent to be accurate in the vicinity of an input location, and therefore use very simple first or second order approximations, such as simple polynomials. In trust-region-based methods [18], the surrogate is only ephemeral and is updated as soon as its validity

conditions are violated. As opposed to global methods for which accuracy guarantees generally do not exist without *a priori* knowledge of the response, local methods can be provably convergent (to at least a local extremum) under particular assumptions [18].

This leads to different performance metrics being used for assessing the validity of a surrogate model [18]. For surrogate models that aim to emulate the response over the whole design space, tools such as k-fold cross-validation, or metrics such as root-mean-square error (RMSE) [19] or maximum absolute error (MAE) [24] are employed. For surrogate models used for optimization in a local region, the priority is rather to ensure that the surrogate model conserves trends, but not necessarily that it minimizes the error of the domain of validity [19].

Observation

While local surrogate models play an important role in particular computational engineering methods, global surrogate models are required in the context of many-query applications such as DSE, MDAO, or UQ.

Parametric and Non-Parametric Models Surrogate modeling borrows methods from the statistical learning and machine learning communities. In general, regression or interpolation models are classified as either parametric or non-parametric models.

Parametric models include linear models, such as linear regressions, and more generally the models based on a linear combination of basis function (e.g., polynomial regression that leverages polynomial basis functions), as well as more complex non-linear methods (e.g., neural networks (NNs)). The parameters of NNs are the weights of the mode, that are optimized for the network predictions to match the training data.

Parametric methods that use basis functions are often too rigid for many applications and therefore not adapted to the wide variety of responses observed in engineering problems. The selection of the right basis functions for a given problem makes these methods impractical beyond problems with more than 20 input variables [19]. NNs on the other

hand are very flexible, in particular their deep variants [25, 26]. However, applications of such methods generally employ large datasets. As stated previously, computationally expensive models increasingly used in conceptual design result in sparse training data under the constraint of a realistic analysis budget. Therefore, NNs are generally ill-fitted in this context. Moreover, in a high-dimensional setting, parametric approximation suffers from over-fitting, and the many model parameters that need to be trained lead to high variance [19]. Such parametric methods are therefore impractical in the context of surrogate modeling of expensive analyses with high-dimensional inputs.

Non-parametric methods do not directly rely on parameters in the models. Examples of non-parametric methods include nearest-neighbor methods or Gaussian process regression (GPR). Prediction with a non-parametric method leverages the whole training set, not just the set of learned parameters. Non-parametric approximation are themselves classified into two classes: full-dimensional models and lower-dimensional models.

Non-parametric full dimensional models approximate a d-dimensional function with a d-dimensional estimate. This is what is usually done with parametric regression, piecewise-parametric, local parametric, or roughness penalty approaches. However, these methods "become unreliable when there are many variables" [27], which may be explained by two reasons further discussed in section 1.4.1: 1) the concept of locality is lost in high dimensions, and 2) the number of training observations required to retain the same density increases exponentially with the number of input dimensions. Now, the complexity of training a GPR model, one of the most common non-parametric approaches, grows as $O(N^3)$, where N is the number of training points. This makes these methods impractical when the training set becomes excessively large.

Non-parametric lower-dimensional models approximate a d-dimensional function using a set of low-dimensional functions, i.e., functions whose input spaces is of dimension lower than d. For example, additive models and high-dimensional model representation (HDMR) represent the original high-dimensional function as a sum of lower-dimensional

functions, while retaining the original design variables. Adaptive computations, such as projection pursuit as well as classification and regression trees, use particular ways to obtain low-dimensional functions that are then combined to approximate the full-dimensional function.

Observation

Non-parametric methods are better suited to the situation where training data is sparse, which is the case when expensive analyses are used.

1.3.3 Reduced-Order Methods

ROMs were successfully applied to many problems, in particular in the aerospace engineering literature [28, 22, 29, 30, 31, 32]. The motivation behind ROMs is to speed up the resolution of high-scale possibly non-linear partial differential equations (PDEs) that predict high-dimensional responses, such as numerical fields over a surface. Instead of solving PDEs on a very high-dimensional geometric mesh, a transformation is carried out to solve the PDE on a lower-dimensional basis of functions. To achieve this, the original large scale state vector consisting of every spatial location of the field quantity under study, such as the pressure coefficient C_p for aerodynamic analyses, is replaced by a more compact representation using unsupervised dimension reduction techniques, such as PCA. This reduces the complexity and thus the computational cost of solving the PDE, possibly at the expense of added uncertainty in the result.

Contrasting data-fit surrogates and ROMs The terminology used to refer to surrogates of field responses varies in the literature: the term ROM is sometimes employed for any type of surrogate outputting vector quantities, even if it does not result from the reduction of a PDE. In addition to ROMs in the sense used in the previous paragraph, purely data-fit approaches have indeed been developed that do not require any knowledge or access to the underlying equations or solver's residuals. They only consider the code of interest as

a black-box function whose inputs can be varied and producing a vector output, with no knowledge of the inner model functioning. In [22], the term *equation-free model reduction* is used to refer to those methods. An advantage of ROMs over other surrogate modeling techniques, which explains their success despite their intrusive nature, is the fact that they remain *physics-based*. Governing equations are certainly simplified, but if the snapshots used to constructed lower-dimensional bases are carefully chosen, the solutions to the simplified equations are in practice very close to those of the original equations. There is no such guarantee in a purely data-fit approach where equations are not used: the practicality of non-intrusive methods comes at the cost of losing the physics-based nature of the method.

ROMs are in essence *intrusive* methods in the sense that they require a modification of the analysis itself. Depending on their level of intrusiveness, they may be classified as either intrusive or semi-intrusive.

Intrusive ROMs By intrusive, we mean for example that access to the original solver code is required, along with the ability and resources to modify it in order to implement specific methods [22]. Moreover, if PCA is employed as the dimension reduction technique, a set of responses obtained via the full-scale model (referred to as snapshots) is required. Since all solutions to the PDE will be linear combinations of the snapshots, the latter need to be representative of the set of the PDE solutions. Obtaining such a well-representative set of snapshots may be complex in a high-dimensional input space.

Semi-intrusive ROMs By semi-intrusive, we mean that methods require the trajectory of the solver solutions to infer the underlying equations, such as the method proposed in [33]. Even though they do not require direct modification of the solver code, these semi-intrusive methods 1) require knowledge and access to the internals of the analysis, and 2) they make assumptions regarding the type of equations governing the model.

1.3.4 Multi-Fidelity Surrogate Modeling

Multi-fidelity methods [19, 34, 35, 36] aim at leveraging an auxiliary source of information, e.g., a simplified physics model, that is cheaper to evaluate than the original model. Low-fidelity samples can thus be gathered to fill gaps in input space. Multi-fidelity surrogate modeling methods enhance predictions in the empty regions by making use of this denser auxiliary data.

Multi-fidelity surrogate models are used in situations where the original model's evaluation cost is high. They leverage other sources of information, e.g., lower-fidelity models of the same response or of a correlated response, in order to enhance predictions [18, 19, 35, 37]. Multi-fidelity methods make up a vast field of study that encompasses very different techniques. They will be reviewed in greater details in chapter 3.

1.3.5 Discussion

Observation While ROMs aim at addressing the high computational cost by solving the equations on a smaller-dimensional grid than the original geometrical mesh, they do not address the issues raised by a high-dimensional input space, as the dimension of the input space remains unchanged. Moreover, given the diversity of models used in conceptual design, the necessity to 1) have access to the solver's code, 2) have the knowledge and resources to modify the code, 3) spend resources developing speed-up solutions applicable to a single code, make such approaches challenging to apply in an industrial organizational context.

This leads to the following observation:

Observation

The intrusive or semi-intrusive nature of physics-based ROMs conflicts with their applicability to the wide variety of models used in conceptual design.

We will therefore focus on data-fit and multi-fidelity methods.

Additional strategies Surrogate models are enablers for using analyses inside many-query applications, in that they dramatically reduce the *online* cost of running an analysis and replace it with a usually more affordable *offline* cost. Surrogate modeling may bring benefits by replacing analyses whose standalone evaluation may usually not be deemed computationally expensive, as the prohibitive cost may solely be the consequence of the sheer number of times that the analysis must be evaluated. In the case of computationally expensive analyses, the offline cost of training the surrogate may in turn become prohibitive, as it still requires evaluating the underlying analysis to gather training data. Equivalently, if we assume a fixed analysis budget that keeps the offline training cost affordable, this may result in only being able to afford a sparse training dataset. In order to still be able to benefit from surrogate models as enablers in these conditions, approaches were developed in order to ease the creation of surrogate models when analysis budget is low relatively to the cost of evaluating the analysis [17, 23]. These approaches are not surrogate modeling approaches *per se*, so they do not fit in the taxonomy presented earlier in section 1.3.1. One of the most widespread approaches in that case is adaptive sampling, that is briefly discussed in the next section.

1.3.6 Adaptive Sampling

Adaptive sampling techniques [18, 19, 38] aim at carefully selecting new evaluation points based on an acquisition criterion tailored to the goal being pursued. Possible goals include optimization and global metamodeling, where *metamodeling* is used as a synonym for surrogate modeling. When performing adaptive sampling, it is assumed that 1) an usually sparse training dataset is initially available, and 2) the expensive analysis may be further evaluated at new arbitrary input sites until depletion of the analysis budget. The initial sparse dataset is then optimally filled with respect to the task-at-hand, where optimality is defined with respect to the objective criterion.

Contrast with Traditional DOEs DOEs may be used to select the sample locations when constructing a surrogate model [18], [19]. These methods mostly focus on finding the best arrangement of a fixed number of points within the design space by optimizing some specified space-filling criterion. However, the mathematical ground for these criteria is only valid under the assumption of certain surrogate model forms [39, 40, 41, 42], e.g., polynomial approximation. Additionally, since the decisions on which points to sample are entirely made *a priori*, the knowledge gathered throughout the sampling process about the response of interest is not leveraged. In contrast, adaptive sampling aims at leveraging this knowledge to tailor the selection of new observations based on the current state of knowledge of the response.

Quantification of Epistemic Uncertainty While the acquisition criteria used in adaptive sampling reflect different objectives, they all require a quantification of predictive uncertainty of the response at unobserved input locations. Uncertainty is generally be classified as either aleatory, that originates from a source of uncertainty over which one has no control, and epistemic uncertainty, that is merely due to a lack of knowledge and that can be reduced by gathering additional data [43]. The predictive uncertainty encountered with surrogate models is epistemic in nature, as it may be reduced by making additional observations of the original analyses.

In the case of optimization where the expected improvement (EI) criterion may be used, the quantification of the predictive uncertainty allows to balance exploitation and exploration behaviors. Exploitation directs the selection of new observations towards relatively well-determined extrema, in the sense that their predictive uncertainty is small, while exploration leads towards regions of the input space displaying higher predictive uncertainty, in which new extrema may be uncovered. When adaptive sampling is used to refine a surrogate model, criteria directly leverage the predictive variance, either on a point basis (maximum variance criterion) or as an integrated quantity (integrated mean square error

(IMSE) criterion) [44].

Probabilistic Methods Specific methods are required to capture the epistemic uncertainty introduced by surrogate modeling. The quantification of epistemic uncertainty is – at least in part – built-in for certain surrogate modeling techniques, such as GPR. In GPR, outputs at different input locations are modeled as random variables that follow a joint Gaussian distribution whose covariance matrix is a function of the distance between their respective locations in the input space. This follows from the intuitive idea that the closer points are located to each other in the input space, the more similar we expect the values of their respective outputs to be. More details on GPR will be given in chapter 2.

In order to enable the quantification of epistemic uncertainty when, unlike GPR, they do not have a built-in mechanism to quantify it, surrogate models can be trained in a Bayesian manner. The Bayesian framework offers a principled way of accounting for epistemic uncertainty and is therefore well-adapted to its quantification [45]. In particular, it seamlessly allows to account for epistemic uncertainty arising from a limited set of training samples, which is exactly the situation faced when relying on computationally expensive codes. In the Bayesian framework, the inherent uncertainty of models is captured using tools from probability theory.

During the training phase, the uncertainty of the generated model is inferred from the data. This contrasts with surrogate modeling methods based on optimization, such as the least squares or maximum likelihood estimation methods, whose objectives in the training phase are respectively to minimize the discrepancy between observed and predicted data and to maximize the likelihood of the training set under the predictive model. Instead, a Bayesian surrogate model would recognize that many models or many values or a model's parameters may – to a different extent – be able to explain the observed data. Put simply, it would distribute larger "weights" to the models or parameter values that are the most likely to explain observations.

During the prediction phase, the previously quantified model uncertainty can be propagated to the prediction. This accounts for the fact that the model used to make predictions is inherently uncertain, so predictions made using the model should also carry this uncertainty. The knowledge about the expected function value as well as the uncertainty in the prediction are both leveraged to perform adaptive sampling.

More details on the Bayesian framework will be discussed in chapter 2,

1.3.7 Scope Definition: Three Surrogate Modeling Scenarios

In section 1.2, we presented surrogate modeling as an enabler for the many-query applications that are essential to decision-making in the context of aerospace design. We introduced at a high level the challenges caused by using expensive analyses to create surrogate models with high-dimensional inputs. In this section, we have been looking at surrogate modeling in greater details. We justified a focus on data-fit surrogate models instead of ROMs as they appear to be the most versatile solution to handle the variety of models used in conceptual design. We also introduced two approaches that are common to help in situations where the analysis budget is low relative to the analysis cost: multi-fidelity methods that allow to combine observations of multiple fidelities, and adaptive sampling that consists in carefully selecting new evaluation points. Both approaches aim at making better use of a limited analysis budget. This allows us to introduce three surrogate modeling scenarios that allow us to 1) reduce the research scope, and 2) within this reduced research scope, pinpoint the specific difficulties encountered with high-dimensional inputs.

We will focus on the three following scenarios:

1. Single-fidelity data-fit surrogate modeling;
2. Multi-fidelity data-fit surrogate modeling;
3. Adaptive sampling in a multi-fidelity context.

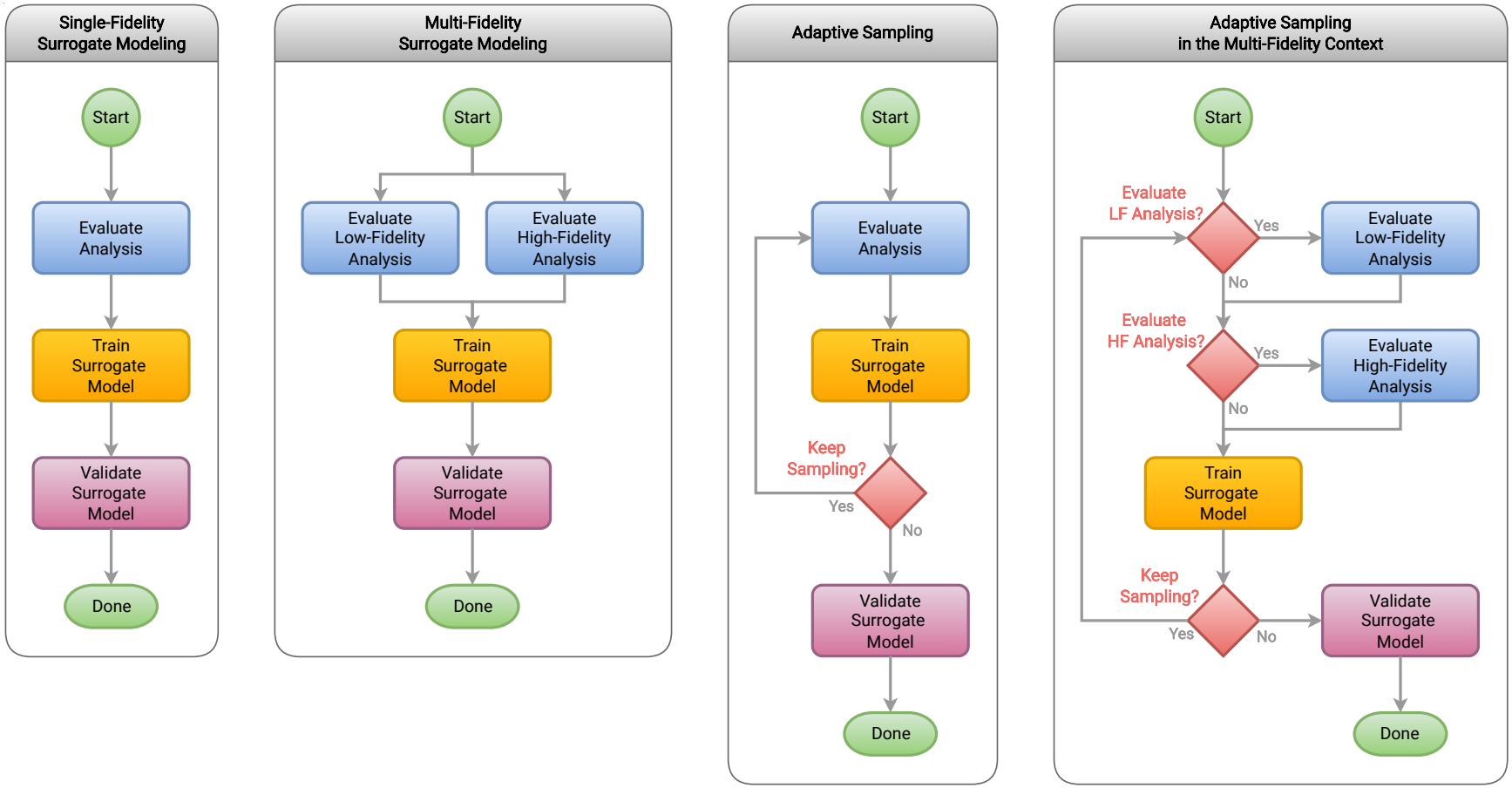


Figure 1.2: Notional flow diagrams for four of the main surrogate modeling scenarios. For ease of representation, we assume two levels of fidelity.

The process corresponding to each scenario presented in fig. 1.2. Single-fidelity surrogate modeling is the simplest. It consists of 1) evaluating the analysis to gather training data, 2) train the model, and 3) validate it. Multi-fidelity surrogate modeling differs by the fact that observations from multiple fidelities can be used as training data. Adaptive sampling introduces a loop in the process. Instead of evaluating the underlying analysis once and for all, we start by spending only a portion of the analysis budget to to build an initial training set used to train a first predictive model. The predictions of this predictive model are then used to assist the selection of subsequent observations. A new predictive model is created that leverages all available observations, and the process repeats until the analysis budget has been spent in its entirety. Adaptive sampling in a multi-fidelity context adds another layer of complexity, as decisions on whether to evaluate each fidelity level must be made at every iteration.

In the next section, we will focus on presenting the challenges that appear when the input space if high-dimensional. For each surrogate modeling scenario, we will seek to identify the steps of the process that are impacted by the high-dimensional inputs, such as to further and more precisely substantiate the gap first raised in section 1.2 and identify the different research areas of this thesis.

1.4 Definition of the Research Objectives

1.4.1 The Curse of Dimensionality

The term *curse of dimensionality* was first introduced by Bellman in [46] in the context of multi-stage decision processes. For example, if M decisions must be made at each of the N stages of the process, then a total of $M \times N$ decisions must be made, with the outcome of earlier decisions possibly affecting later ones. As a result, exploring all possible combinations of decisions is not tractable, motivating the new class of dynamic programming methods introduced by Bellman. Since then, the term was adopted in the literature to describe any problem in which traditional methods lose tractability due to high dimen-

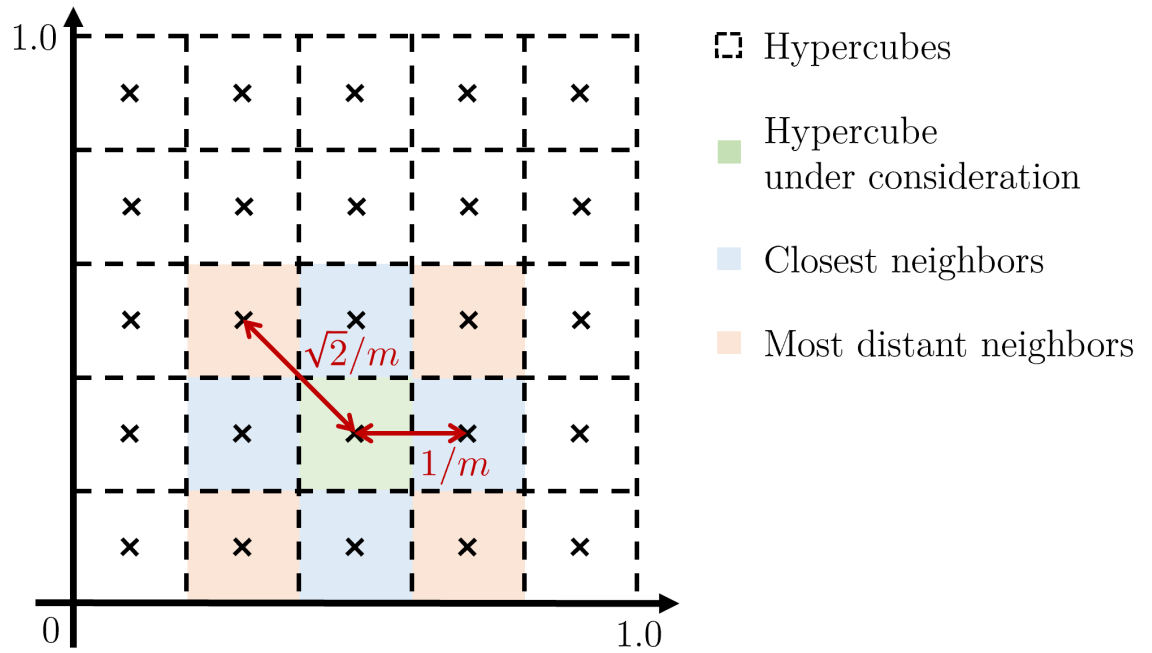


Figure 1.3: Illustration of the partition of the $[0,1]^d$ hypercube into smaller hypercubes of side length $1/m$ when $d = 2$ and $m = 5$.

sions [24, 47, 48, 49]. The following paragraphs detail several practical consequences of the curse of dimensionality on sampling and regression.

Full-factorial sampling Let us first illustrate the effect of the curse of dimensionality on a dense sampling scheme, such as the full-factorial sampling with m levels in d dimensions. Let us consider the case of independent continuous inputs and use the d-dimensional hypercube $[0,1]^d \subset R^d$ as input space, since continuous inputs can easily be normalized. Following this sampling scheme, every dimension is assigned m levels between 0 and 1 and a sample point is placed at each possible combination of d levels. Such a scheme requires $N = m^d$ samples in total. It can be alternatively be visualized as segmenting the input space into hypercubes of edge length $1/m$ and placing a sample at the center of each of them. Figure 1.3 depicts the situation for $d = 2$ and $m = 5$.

The volume of each hypercube is then $(1/m)^d$. Each sample has $3^d - 1$ direct neighbors, in the sense that their corresponding hypercubes share at least one vertex or corner. The maximum distance to one of these neighbors then varies as $\sqrt{\frac{d}{m}}$. Figure 1.4 depicts these

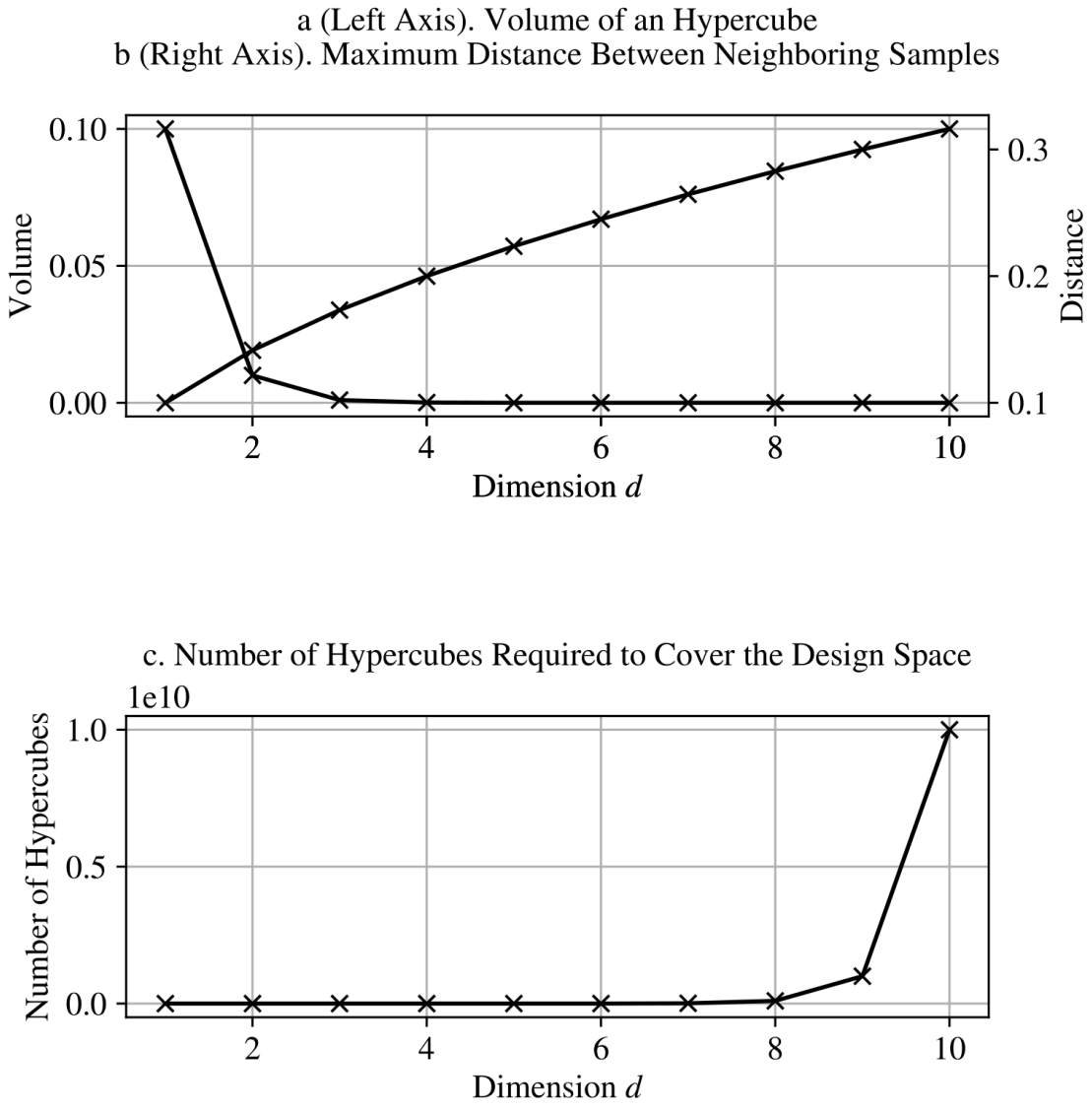


Figure 1.4: Evolution the volume of individual hypercubes (top, left axis), the minimum distance between sampling locations (top, right axis), and the number of hypercubes required to cover $[0, 1]^d$, as a function of the number d of dimensions when $m = 10$ segments are used for each dimension.

results for $m = 10$ and d varying from 1 to 10.

The volume of each hypercube decreases sharply as the number of dimensions increases, while the distance between two neighboring hypercubes increases. The number of hypercubes required to cover the whole design space increases exponentially with the number of dimensions.

Altered notion of neighborhood Motivated by the simplicity of function approximations methods based on local neighborhoods, such as k-nearest-averaging, [24] considers the evolution of the size of an hypercubical neighborhood necessary to capture a fraction denoted r of all observations as the number of dimensions increases, and the interested reader is invited to refer to this reference, and in particular its figure 2.6 that offers a clear depiction of the situation. In this figure, we observe that, as expected, the size of the neighborhood increases from 0 to 1 as the fraction r increases. The interesting effect of dimensionality is on the rate of increase: in 10 dimensions, in order to capture 1% to 10% of the hypercubical volume, one would need an hypercube whose side lengths measure between 63% and 80% of each axis dimensions. This means that two points within that "neighborhood" are actually located very far from each other.

1.4.2 Impact of the Curse of Dimensionality on Considered Surrogate Modeling Scenarios

The illustrations of the curse of dimensionality presented in section 1.4.1 all impact the surrogate modeling scenarios of interest depicted in fig. 1.2.

All scenarios, whether single-fidelity, multi-fidelity, or adaptive sampling, start with a step in which initial observations are selected using a DOE. In the case of multi-fidelity surrogate modeling, such DOEs must be made for each individual fidelity. For adaptive sampling, an initial model is necessary to adaptively select new observations, thus requiring an initial DOE. The full-factorial sampling example showed that a set of observations will be inherently sparse in a high-dimensional design space, and any attempt to make it dense

would require an exponential increase in sampling density. The full-factorial sampling makes for a clear-cut illustration of the problem, but the same challenges plague other space-filling DOE techniques. Therefore, the selection of training observations quickly becomes impractical as the input dimension increases (as shown in fig. 1.4), even when model evaluations are relatively cheap.

Observation

Space-filling DOE methods become impractical due to the curse of dimensionality, impacting the initial selection of training observations in all considered surrogate modeling scenarios.

The second example showed that it becomes harder to reason on distances when dealing with high-dimensional spaces. For parametric methods, the number of model parameters generally increases with the input space dimension due to the necessity to model the dependence on the extra input variables, but also on the interactions between input variables. The more flexible the model, the more parameters are needed to model each dependence, again increasing the total number of required parameters. The more parameters are used, the harder it becomes to identify them. In particular, if too few observations are used for training a model relative to the number of its parameters, we incur the risk of over-fitting. For non-parametric methods, the reliance on the notion of distance between points through length scale hyperparameters means that methods such as RBF or GPR may encounter issues in high-dimensional spaces [38, 50]. This also impacts multi-fidelity methods that rely on GPs, such as the Bayesian calibration framework of Kennedy and O'Hagan [51] and its derivatives [35]. This impacts the training step of all considered scenarios.

Observation

The curse of dimensionality creates challenges for the training of both parametric and non-parametric surrogate modeling techniques, affecting all considered surrogate modeling scenarios.

Adaptive sampling requires the optimization of an acquisition function when selecting new observations. Because this optimization problem is solved in the input space, it is made more difficult when it becomes high-dimensional [38, 50]. First because of the exponential increase in the volume of the solution space, requiring more iterations and, when applicable, more restarts. Second because of the added computational complexity, for example during the direction-finding step. In gradient-based approaches to optimization, when direct gradient evaluations are not available, approximation methods such as finite differences are used. Such methods may incur a prohibitive cost when the inputs are high-dimensional. These difficulties compound with an already challenging optimization problem, as the acquisition criteria generally exhibits a highly multimodal behavior.

Observation

The curse of dimensionality hinders the adaptive selection of new observations, affecting the surrogate modeling scenarios that use adaptive sampling.

1.4.3 Research Framing

Overarching Research Objective

We have seem that the curse of dimensionality that arises from high-dimensional input spaces hinders all three surrogate modeling scenarios. In section 1.2, we have seen that high-dimensional parameter spaces are increasingly common in the context of aerospace design, as this enables the designer to potentially retain design freedom longer in the design process, or consider larger solution spaces to find better designs. Simultaneously, increasingly expensive analyses are being used to maximize the design knowledge acquired early on in the design process. Under these conditions, only a relatively low number of analysis observations are available to create surrogate models, further complicating the process of creating them. This is problematic to the successful completion of design activities, which are informed by many-query applications such as design space exploration or optimization

since they are enabled by surrogate models. If surrogate models with high-dimensional inputs cannot be created when only a few analysis observations are available, then the many-query applications that are crucial to the design process may not be carried out. This motivates the overarching research question driving this thesis:

Overarching Research Question

How can we alleviate the impact of the curse of dimensionality on the three surrogate modeling scenarios under consideration when only few training analysis observations are affordable?

Research Area 1: Single-Fidelity Surrogate Modeling

We have seen that single fidelity surrogate modeling suffers from the curse of dimensionality in multiple ways. First, the space-filling DOEs carried out to gather training samples are effectively sparse in high dimensions. Second, whether the mathematical models underlying the surrogate are parametric or not, their training is made harder by the high number of dimensions. Moreover, the single-fidelity scenario underpins the other two, and advancements made on this case would hopefully benefit the other two. Therefore, we will focus the first research area on this scenario, driven by the following research question:

Research Question 1

How can we alleviate the impact of the curse of dimensionality on single-fidelity surrogate modeling when only few training analysis observations are available?

Research Area 2: Multi-Fidelity Modeling

We saw that multi-fidelity modeling is impacted by the curse of dimensionality in much the same way as the single-fidelity case: both the initial DOE and the training of such models are affected. Now, we are hoping to be able to transpose the solutions found in the first research area to this research area, since single- and multi-fidelity surrogate modeling

methods have a lot in common. Still, we expect that the extension of methods proposed to address research question 1 would require further development to be used in the multi-fidelity case, warranting the second area motivated by research question 2. We use the formulation "low analysis budget" instead of "few analysis observations" to account for the fact that, in a multi-fidelity context, we are usually working with an analysis budget that is split between observations of analyses of different levels of fidelity.

Research Question 2

How can we alleviate the impact of the curse of dimensionality on multi-fidelity surrogate modeling for low analysis budgets?

Research Area 3: Sampling Strategies for High-Dimensional Inputs

As explained previously, multiple mechanisms exist for selecting training points when creating surrogate models. DOEs are systematically employed to gather initial training points, and adaptive sampling may be leveraged to select additional training observations after an initial predictive model has been generated. Both approaches suffer from the curse of dimensionality. As we will see, a byproduct of the approach proposed in chapter 2 is a low-dimensional subspace of the original input space that hopefully concentrates most of the variation of the function under consideration. The third research area focuses on leveraging this subspace to perform DOEs and adaptive sampling in the hope to lessen the impact of the curse of dimensionality.

Research Question 3

How can we alleviate the impact of the curse of dimensionality on DOEs and adaptive sampling for low analysis budgets?

1.5 Summary

In this chapter, we started by presenting the high-level motivation for the research on surrogate modeling conducted in this thesis. Modern approaches, such as design space exploration, engineering optimization, or uncertainty quantification, are crucial to engineering design because they allow stakeholders to make informed design decisions. These approaches have in common their need for evaluating analyses online and often, quickly becoming prohibitively expensive. By trading the large cost of evaluating analyses online for a smaller one-time offline cost, surrogate modeling makes the practical implementations of these approaches possible, and can therefore be considered to be an enabler to modern approaches to engineering design. The process of generating surrogate models is however challenged by two current trends. The first is the increase in analysis fidelity and – accordingly – evaluation cost. Under organizational constraints which limit the budget allocated to running analyses, this results into a limited set of observations being available for the offline training phase of surrogate models. The second is the increase in the number of independent parameters considered in engineering design trade studies. Using more independent parameters may be motivated by the desire to explore larger design spaces to reach new optima, consider unconventional configurations, or retain design freedom longer in the design process. The curse of dimensionality then comes into play, complicating all phases of the construction of surrogate models. These observations led to the main gap motivating this thesis: if adequate surrogate models cannot be created with high-dimensional inputs when only few observations are available, then the methods used to inform design decisions can in turn not be leveraged.

In the second part of this chapter, we first aimed at refining the scope of this thesis to then be able to derive narrower research objectives. After a review of surrogate modeling techniques, we identified three main surrogate modeling scenarios on which to focus. First, the "base" case of single-fidelity surrogate modeling. Then, we introduced two scenarios

specifically tailored to situations in which analysis observations are expensive to acquire and additional strategies are needed beyond traditional regression methods: multi-fidelity surrogate modeling and adaptive sampling in the context of multi-fidelity surrogate modeling. The curse of dimensionality that plagues computational methods involving the exploration of high-dimensional spaces was then illustrated and its impact on the three surrogate modeling scenarios under consideration were underlined. This led to the introduction of the overarching research question and the three research areas that structure this thesis and directly map to the three surrogate modeling scenarios delineated earlier. Beyond the logical split between the scenarios, the ordering of the research areas also comes naturally. The single-fidelity case is treated first in chapter 2 as it allows us to build the foundation for further developments. The method proposed in the multi-fidelity case subsequently builds on the single-fidelity approach and is presented in chapter 3. Finally, the sampling strategies proposed in chapter 4 leverage contributions made in the two previous chapters.

CHAPTER 2
FULLY BAYESIAN APPROACH TO APPROXIMATION BY RIDGE FUNCTIONS

2.1 Refinement of Research Question 1

In chapter 1, the discussion of the impact of the curse of dimensionality on single-fidelity surrogate modeling led to the following research question:

Research Question 1

How can we alleviate the impact of the curse of dimensionality on single-fidelity surrogate modeling when only few training analysis observations are available?

The generic process corresponding to single-fidelity surrogate modeling is recalled in fig. 2.1. The objective of this section and the next is to refine this research question by clearly identifying a capability gap, i.e., a needed capability that existing methods lack to provide. In this section, we will start by looking into greater details at existing methods to surrogate modeling for high-dimensional inputs, identifying those that are most promising in the context of this thesis, and pinpointing their limitations. We will see that although many methods address the curse of dimensionality, they require a large amount of training data or access to gradient evaluations. In the next section, we will investigate ways to overcome these limitations. In particular, we will look at Bayesian approaches which enable robustness against the sparsity of training data. There, we will identify a more specific capability gap that will accordingly allow us to introduce a more focused version of research question 1.

2.1.1 Inapplicability of Unsupervised Dimension Reduction Strategies

In this section, we take a quick detour to discuss other contexts in which high dimensions may be encountered, beside the input space. In particular, high-dimensional state or output

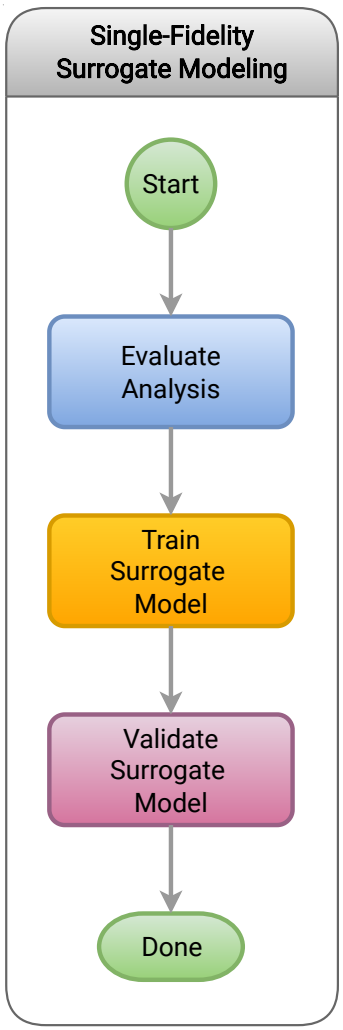


Figure 2.1: Generic process for the first considered surrogate modeling scenario, single-fidelity surrogate modeling.

spaces are widespread in numerical simulations. Methods to handle these have been developed based on unsupervised dimension reduction techniques and are now commonly used in research and industry. We will explain and show by example why these unsupervised methods are not applicable to address the curse of dimensionality that plagues the creation of surrogate models with high-dimensional inputs.

Dimension Reduction for Data-Fit Field Surrogate Models

While the present work focuses on the impact of the curse of dimensionality on the *input* space of surrogate models, it also affects their *state* and *output* spaces. Responses predicted by surrogate models are generally scalar quantities. For example, in the aerospace engineering context, these scalars might be integrated quantities, such as lift, drag, and moment coefficients, vehicle weight, or fuel required. However, surrogate models may also be used to predict more complex types of responses, such as vector quantities, sometimes referred to as *fields* in the engineering literature. For example, in the context of an aerodynamic analysis, outputs of interest may include the pressure distribution, or C_p distribution, over the surface of the wing. Compared to integrated scalar quantities, the C_p distribution carries more information that may be of interest to aerodynamic or structural engineers during design. Not only can lift and drag coefficient be derived from the C_p distribution, it can also be integrated in an aerostructural analysis, in which aerodynamic loads are passed to a structural analysis to compute an elastic wing's deflections. While surrogates of field responses were first obtained using intrusive ROMs in the literature, full data-fit approaches have been developed and are now being used.

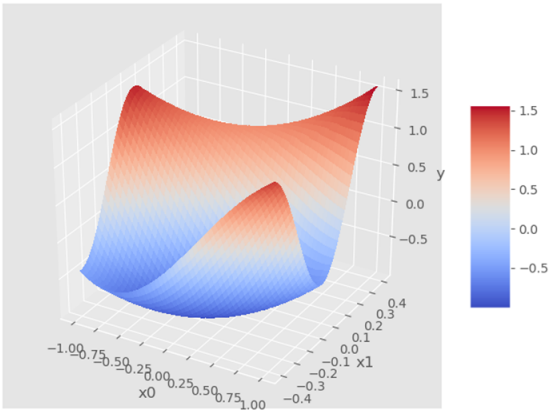
Observation

Surrogate models of field responses boast advantages over traditional scalar surrogates, which explains their increased usage to support engineering applications.

Data-fit surrogate modeling methods only yield scalar or vector outputs. Numerical

field quantities are in general discretized, e.g., when they originate from solving a governing equation over a mesh and can therefore be represented by a high-dimensional vector. However, it becomes cumbersome and inefficient to train a surrogate model with high-dimensional multivariate outputs. To overcome this, we can reduce the dimension of the response. This can be done using *unsupervised* dimension reduction methods, such as PCA [52], deep autoencoder-decoder networks [53], or diffusion maps [54]. The prevailing method is to construct that basis using PCA, also called proper orthogonal decomposition (POD), using fields from a sample set called snapshots [28, 55, 56, 57, 58]. PCA seeks a basis (in the sense of vector spaces) in which the field outputs can be described in a more compact manner. This is achieved by building the basis such that it concentrates the variance of the training dataset in as few directions as possible. As a consequence, the resulting basis can be lower-dimensional than the original basis with negligible effect on the model's predictive accuracy. The coordinates of the training fields projected in this reduced space are substituted to the original outputs. Otherwise, the process carried out to train the surrogate model remains unchanged.

Addressing the high dimensionality of the state and output spaces has been the subject of extensive research in the fields of reduced-order modeling and equation-free model reduction [29]. As a result, data-fit and physics-based surrogates of high-dimensional responses are now widespread in the literature. Such approaches are even integrated in commercial software, e.g., SAS' JMP [59] or in proprietary software, e.g., Airbus' PODRacer [60]. While dimension reduction techniques enable the generation of data-fit field surrogates, it is important to make note of the two following observations.



$$y = 0.5 \times (0.2x_1 + x_0)^2 + \sin\big(2.5\pi(-0.2x_0 + x_1)\big)$$

Figure 2.2: Shape and expression of the analytical function used for the purpose of illustrating the shortcomings of applying unsupervised dimension reduction techniques for reducing the input space dimension.

Observations

- Unsupervised dimension reduction methods applied to the response or state fields do not alter the number of input parameters.
- The off-line phase of dimension reduction is computationally expensive: since the reduced basis is only as good as the samples used to build it, it requires many observations of the field output of the underlying analysis.

Graphical Example

In this section, we will discuss why unsupervised dimension reduction techniques are inappropriate to reduce the input space dimension and illustrate the problems raised here on a simple graphical example. The analytical function used for the purpose of illustration is depicted in fig. 2.2.

Using PCA, we seek the directions of maximal variance of a given dataset $\mathbf{X}$, i.e., the directions in which the data varies the most. In fact, if $\mathbf{X}$ follows a multivariate Gaussian distribution, then PCA can be used as a density estimation technique [61]. These methods effectively leverage the dependence, and in the case of PCA the correlation, between

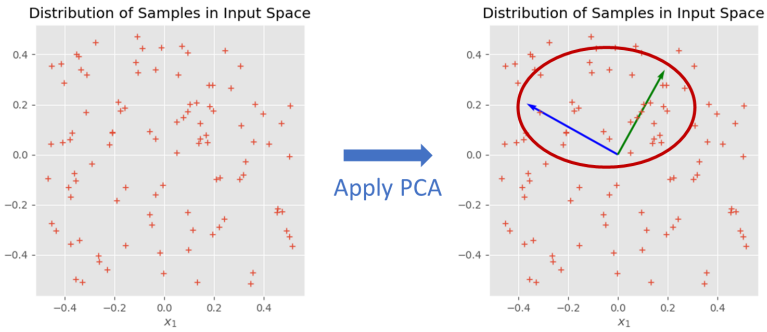


Figure 2.3: Example of application of PCA on a uniform input distribution. Both resulting PCA components approximately have the same amplitude, and no dimension reduction is possible.

different components. These methods are however not directly applicable to the problem of high-dimensional *input* spaces, as they do not take the input-output relationship into account when learning the dimension reduction transformation.

The dependence assumption underlying unsupervised dimension reduction methods contrasts with assumptions usually encountered in the context of surrogate modeling, where each input variable is assumed to uniformly vary within known bounds and inputs are assumed to be independent. Dependent, or even correlated, input variables are rarely encountered. This leads to the vector of input variables usually being uniformly distributed within an hypercube. Because there is no dependence to leverage, or correlation in the case of PCA, unsupervised dimension reduction methods would simply fail at their task. We illustrate this situation in fig. 2.3: both components obtained using PCA have the same amplitude and no effective dimension reduction is possible.

If inputs happened to be be dependent, this dependence may not be informative to correctly model the input-output relationship of interest. On the contrary, such an approach may be detrimental to the predictive accuracy of the surrogate model because it may discard information that is useful for predicting the response of interest. Reduction of the input space dimension demands a thorough understanding of the relationship between the inputs and the underlying function. As recognized in [62], *supervised* techniques bring an advantage over unsupervised methods when applied in the context of surrogate modeling.

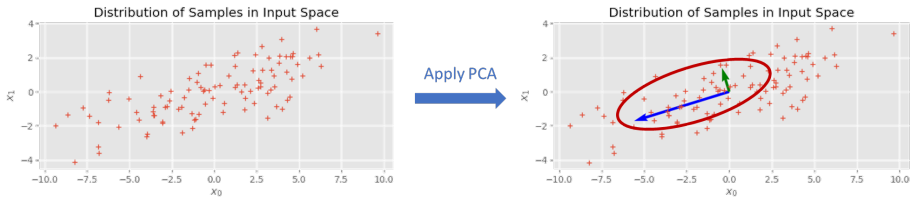


Figure 2.4: Example of application of PCA on a correlated input distribution. This time, there is a principal component, that captures the dominant direction in which *inputs* are varying.

What we are really after are directions or subspaces of the input space that contribute the most to the variability of the response. We illustrate this situation in fig. 2.4. PCA is applied to the input locations shown on the left hand side, resulting in two principal components with clearly different amplitudes. This time, dimension reduction of the input space using PCA is possible, although not relevant as demonstrated next.

Figure 2.5 shows the projection of the output values on both these directions. If the PCA methodology is followed and the first dimension is selected, we would be in the situation depicted in the bottom left. The next step would be to create a surrogate model for these noisy observations, which would probably lead to a poor predictive accuracy. On the other hand, if we had chosen the second dimension, we would be left with creating a surrogate for the input/output pairs depicted to the left, which although noisy, appear more reasonable. This leads to the following observation:

Observation

Unsupervised dimension reduction methods are generally not applicable to reduce the input space dimension.

Although we defer the discussion of supervised dimension reduction methods to the next section, we will go one step further and visualize what had happened if we had applied one such method to the current analytical problem. In fig. 2.6, the original response is represented on the left-hand side in 3D and using contours on the right hand side. Red

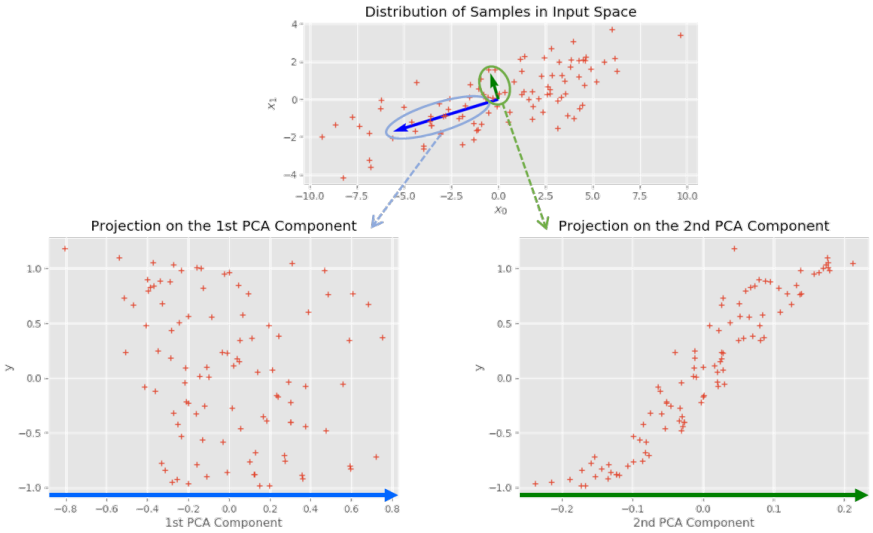
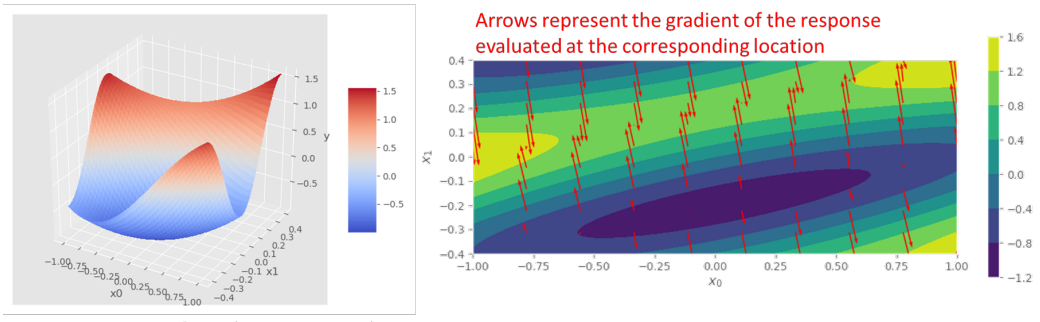


Figure 2.5: Projection of the function values on 1) the first principal component (left) and 2) the second principal component (right). Selecting the component using PCA would lead to trying to creating a surrogate for the noisy observations on the left, leading to a poorer model than if the observations on the right had been used.



$y = 0.5 \times (0.2x_1 + x_0)^2 + \sin(2.5\pi(-0.2x_0 + x_1))$

Figure 2.6: 3D view and expression of the analytical function used for illustration purposes (left) and corresponding contour plot with gradients represented as red arrows (right).

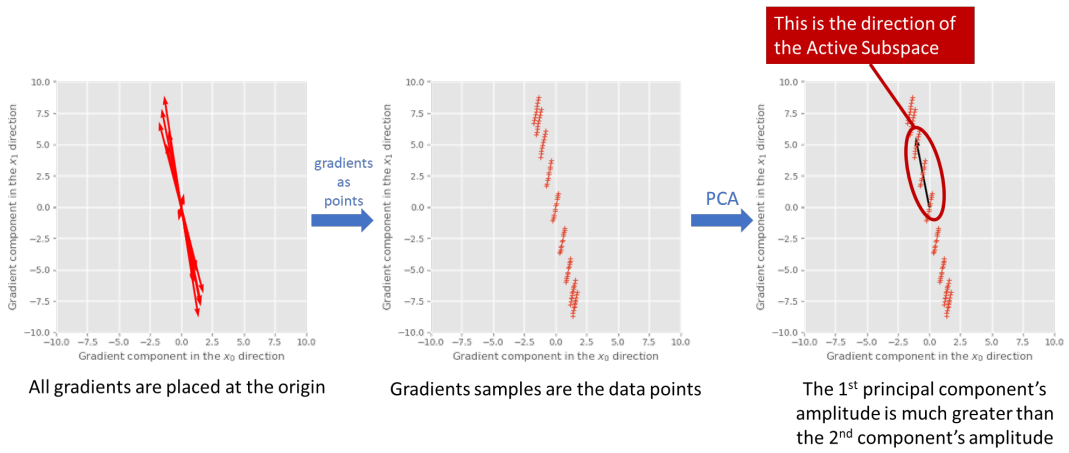


Figure 2.7: The application of PCA on the *gradient samples* yields the AS.

arrows represent gradient samples, which are necessary to apply the active subspace (AS) method.

In fig. 2.7, these gradient samples are treated as data points on which PCA is applied. The principal components obtained when applying PCA to the gradient samples form the so-called *active subspace*.

Finally, fig. 2.8 depicts the result obtained when projecting data onto the AS in the bottom right corner. Some noise was introduced due to the projection (there is still some variation along the so-called *inactive dimension* orthogonal to the AS). However, the resulting data is significantly less noisy and more suited to regression than the projected data previously obtained when naively applying PCA directly to the distribution of the inputs (top-right corner of fig. 2.8).

2.1.2 Existing Approaches for Surrogate Modeling with High-Dimensional Inputs

Since we have seen that the widely used *unsupervised* dimension reduction are not suitable to addressing the curse of dimensionality encountered when creating surrogate models with high-dimensional methods, we now turn to the more relevant class of *supervised* approaches to dimension reduction. This section focuses on introducing these methods, understanding their limitations, and identifying the most promising among them based on

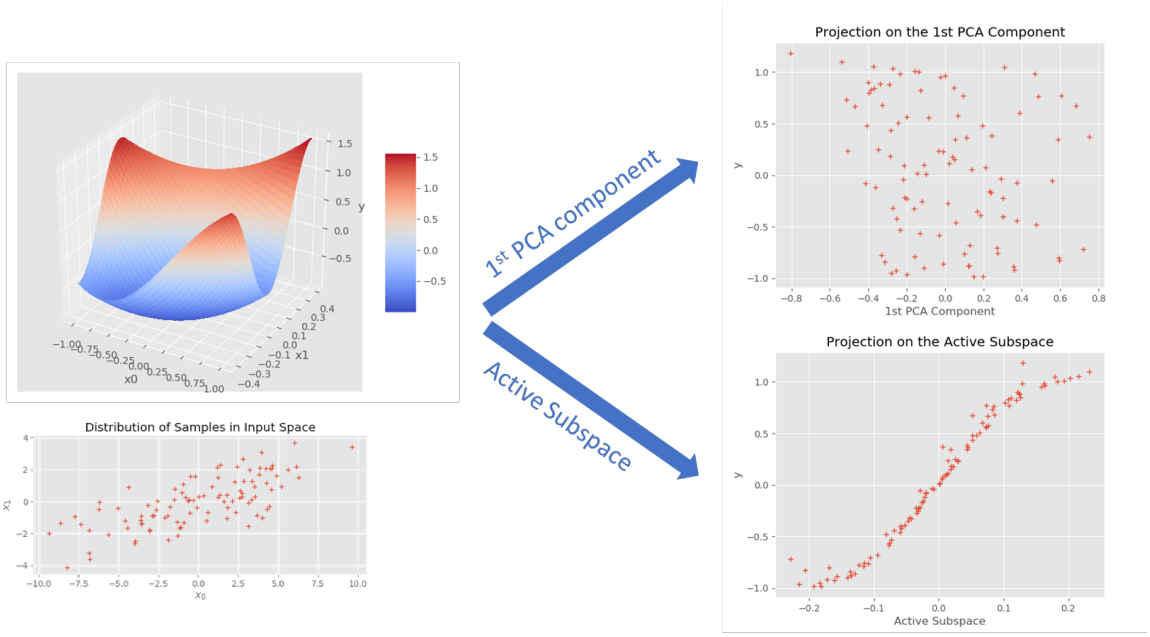


Figure 2.8: The directions obtained when applying Active Subspace are much more suitable to being used for regression than those obtained using PCA

a review of existing literature. In section 2.1.2, we first present an overview of high-level strategies available to handle high-dimensional inputs. In section 2.1.2, we focus on specific surrogate modeling methods designed to work with high-dimensional inputs. Finally, in section 2.1.2, we summarize our findings and discuss the next steps.

High-Level Strategies

[63] proposed to classify strategies to handle high-dimensional inputs in five categories: 1) decomposition, 2) screening, 3) mapping, 4) space reduction, and 5) visualization. We briefly discuss each category in the following paragraphs. The interested reader may refer to table 1 in [63] for a more thorough exposition to these methods.

Decomposition These methods "reformulate an original problem into a set of independent or coordinated sub-problems of smaller scale" [63]. They rely on matrices, such as the design-structure matrix (DSM) widely used in the MDAO context. Advantages of such methods include the reduction of the number of dimensions in each sub-problem compared

to the original problem, the possibility to solve these independent or loosely coupled sub-problems in parallel, and an improved coordination between the different sub-problems. Drawbacks of decomposition methods include the fact that boundaries used for the decomposition are usually subjective and these boundaries may not even exist if, e.g., we are considering a single discipline that cannot easily be segmented into several sub-problems.

Screening A screening method "identifies and retains important input variables and interaction terms, whereas removes less important ones or noises in the problems of interest so that the complexity or dimensionality of the problems is reduced to save computational cost" [63]. Examples of this class of methods include analysis of variance (ANOVA), as well as local and global sensitivity analysis. These methods generally leverage the 80/20 principle according to which most of the variability of most responses encountered in engineering applications can be explained by a few variables. Drawbacks of screening methods include the loss of modeling accuracy due to discarding dimensions, the possible inability to reduce the number of dimensions to a practical level (too many dimensions remain after screening), the reduced effectiveness when simultaneously applied to multiple responses, and the rapid increase in cost when input dimension increases [63].

Mapping These methods include "projection, non-linear mapping, parameter space transformations" [63]. Mapping techniques include methods such as PCA, ANOVA mapping, or relative distance plane (RDP) mapping. However, as discussed previously, these examples only rely on an existing correlation between parameters and are therefore not applicable when the parameter space is for example uniformly sampled. Other instances are non-linear mappings, such as artificial neural networks (ANNs). Space mapping is used when fine and coarse models are used concurrently in an optimization setting while they have different input spaces. Mapping techniques are identified by the author of the review [63] as one of the promising routes for dimensionality reduction because they actually address the curse of dimensionality directly by mapping the original high-dimensional parameter

space to a lower-dimensional alternate parameter space. The challenge is to find a space that maintains the desirable function properties of the underlying analysis, such as extrema in an optimization setting, or an overall accuracy in the case of global metamodeling.

Space Reduction These methods aim at the "reduction of ranges of design variables excluding the reduction of the number of variables" [63]. They are not applicable in the context of surrogate models where the design variable bounds are fixed *a priori* by the practitioner.

Visualization These include "techniques for multidimensional data visualization". These strategies are irrelevant to the present context.

Surrogate Modeling Methods

While the previous section focused on high-level strategies for handling high-dimension inputs, this section focuses on specific methods. In [63], it is argued that the number of dimensions, nonlinearity, interaction among variables, and the relative importance of interaction terms should be used to characterize black-box functions and that this characterization should drive the selection of an appropriate surrogate modeling method. For data-fit techniques, the authors identified methods used for sampling, model structure selection, model fitting, and determining the sample size as essential. In short, they propose to solve the issue raised by the conflicting requirements of high sampling density and high number of dimensions by leveraging "strategies such as decomposition, additive modeling, mapping, etc.", which should be tailored to "the nature of the underlying function". In the case of high-dimensional inputs, [63] argue that non-parametric lower-dimensional methods are most appropriate. Non-parametric lower-dimensional models for data-fit surrogate modeling can be classified as additive models (linear additive models, generalized additive models, HDMR) or adaptive computation (projection pursuit, classification and regression tree), and both will be discussed in this section.

Additive Models Example of additive models include the class of HDMR [64] methods that leverage the ANOVA decomposition to model a function as a sum of main effects and interactions. They are motivated by many empirical observations that high-order interactions are negligible in most scientific and engineering problems. As a consequence, the terms corresponding to these higher-order interactions may effectively be dropped from the decomposition. This simplifies the regression task, which is reduced to the identification of a set of low-dimensional functions that are not, or at least to a lesser extent, affected by the curse of dimensionality.

The methods however suffer from multiple shortcomings. First, they heavily depend on the additivity property of the underlying analysis. When few observations are available, this may be hard to determine. Second, another challenge arises with specific methods within the HDMR family, such as cut-HDMR, that call for particular sampling schemes. Because these sampling schemes are particular to the model form assumed by those methods and often not space-filling, the observations gathered in this manner may not be easily reusable to create other surrogate models. As a result of applying such schemes, we may incur the risk to waste part of the allocated analysis budget. Third, because they still use the original untransformed inputs parameters, additive methods are still susceptible to the curse of dimensionality, although to a lesser extent. For example, a purely additive model structure for a function with 100 inputs leads to 100 individual 1-dimensional functions that need to be identified. This quickly gets worse if interactions between variables are considered: there are $\binom{d}{2}$ two-way interactions, which leads for example to having to identify $\binom{100}{2} = 4950$ 2-dimensional functions. For these reasons, additive methods that operate on the original parameters do not appear as a scalable solution.

Projection Pursuit Regression Projection pursuit regression (PPR) is a precursor to ANNs [24] that seeks to automatically construct new explanatory variables by means of transformations of the original inputs. These derived dimensions are meant to be more

informative than the original input dimensions, and it is expected that significantly fewer derived dimensions would be sufficient to accurately represent the function under consideration.

In a sense, PPR combines the ideas of additive models and mapping strategies. The objective is to find a set of directions $\{\theta_{\mathbf{1}}, \theta_{\mathbf{2}}, \ldots, \theta_{\mathbf{m}}\}$ with $m \ll d$ such that $\forall \mathbf{x} \in \mathcal{X}$:

$$f(\mathbf{x}) \approx g_1({\theta_{\mathbf{1}}}^T \mathbf{x}) + g_2({\theta_{\mathbf{2}}}^T \mathbf{x}) + \cdots + g_m({\theta_{\mathbf{m}}}^T \mathbf{x}) \tag{2.1}$$

It is computationally costly because it requires optimization of the projection parameters whose length is the original number of input variables. Such methods are expected to successfully handle non-linear behaviors because the individual functions with 1D inputs can be fitted using any kind of flexible data-fit surrogate, such as GPR [24]. PPR can actually be shown to be an universal approximator [24], i.e., any continuous function in $\mathbb{R}^d$ can be approximated using PPR if the selected number m of individual functions is large enough.

Projection pursuit regression has never been widely adopted as it required important computational capabilities that were not available at the time of its inception, and once they became available, PPR was directly supplanted by modern artificial neural networks. In contrast with ANN that perform a series of non-linear transformations of the original input variables in order to identify latent features that are the most informative for prediction, PPR only performs linear transformations. Non-linear features are only captured by the regression functions g_i. Due to the many connections between nodes in ANN, many parameters need to be estimated. This is possible if many training samples are available. If the process by which training samples are obtained is costly to evaluate, as it is for high-fidelity physics-based models in engineering applications, then this becomes problematic.

Conclusion

Among possible routes for alleviating the impact of the curse of dimensionality on the input space, the two deemed most promising in literature are: a) mapping strategies that

take advantage of the dependence of the response on a low-dimensional subspace instead of the original high-dimensional input space, and b) lower-dimensional models, and in particular additive or partially additive models.

Observation

Among all considered strategies to handle high-dimensional inputs, mapping strategies appear as the most promising class of methods to alleviate the curse of dimensionality.

In the next section, we will accordingly be looking at methods based on approximation by ridge functions, a large class of methods within mapping strategies.

2.1.3 Approximation by Ridge Functions

Overview

As argued previously, mapping strategies are the most promising class of methods to deal with high-dimensional inputs, as they intend to extract useful directions or subspaces in parameter space instead of simply discarding variables. While mapping strategies is a broad denomination proposed in [63] in the context of engineering design, similar methods actually exist under other names in other fields:

- approximation by ridge functions in the function approximation literature [65, 66, 67, 68, 69, 22, 70, 71, 72, 73, 74, 75]
- sufficient dimension reduction [76, 77, 78, 79, 80]
- spectral methods, sometimes referred to as supervised dimension reduction or dimension reduction for supervised learning [81, 82, 83, 84, 85, 86, 87, 88, 89, 90]
- probabilistic approaches [91, 92]

These methods all seek similar objectives while leveraging different mathematical tools. It is also interesting to note that while we are seeking explicit supervised dimensionality

reduction methods that provide an explicit two-way mapping between the low- and high-dimensional spaces, the success of recent deep learning techniques can be attributes to an implicit dimensionality reduction, referred to as feature extraction [93].

Observation

Approximation by ridge functions is a principled way to reduce the dimension of the input space.

The idea to identify a function f using the form $f(\mathbf{x}) = g(\mathbf{W}^T\mathbf{x})$, where $\mathbf{W}$ is a tall rectangular matrix and g is referred to as *link function* whose input space is lower-dimensional than f, has been proposed several decades ago in the literature [70]. The term *ridge function* is usually used to refer to those functions and extensive literature on this topic can be found [65, 66, 67, 68, 69, 22, 70, 71, 72, 73, 74, 75].

Recent methods were proposed to identify ridge functions, that either rely on sparse matrix approximations [75, 94] or low-rank approximations [72, 74]. The sparse approximation is hard to justify, as there is no theoretical argument to back the fact that the covariance matrix of the gradient is sparse. Low-rank approximation methods are interesting but they can in the end be interpreted as approximating finite-difference gradients, which is very ineffective and hinders exploration of the full parameter space by requiring small variations of samples about points.

In the sufficient dimension reduction literature [78, 95], methods such as sliced inverse regression [95] have been extensively used. However, as they assume elliptical distribution for the parameters, they do not fit the needs encountered when generating surrogate models, where the input space is usually uniformly sampled.

Spectral methods [81, 82, 83, 84, 85, 86, 87, 88, 89, 90] to supervised dimensionality reduction are numerous. They include canonical correlation analysis (CCA), linear discriminant analysis (LDA), principal components regression (PCR), partial least squares (PLS), supervised principal components [24], along with the corresponding non-linear kernelized

versions. Hessian-based approaches have also been proposed to identify low-dimensional manifolds in the input space [96].

Observation

Conventional approaches to approximation by ridge functions require large amounts of data or make impractical assumptions

In the next section, we will turn to a more recent addition to the family of methods relying on approximation by ridge functions, the AS method.

Active Subspace

This section focuses on active subspace (AS), a state-of-the-art supervised dimensionality reduction technique developed recently that has quickly gathered momentum in literature. We will see that, while it might not be the best-suited method in the context of this thesis due to its restrictive need for response gradients, its underlying concepts offer a solid ground on which new gradient-free methods can developed.

The AS method addresses challenges raised by functions of high-dimensional inputs and the resulting curse of dimensionality that hinders the use of such functions in numerical activities such as uncertainty quantification, surrogate modeling, or numerical optimization. In simple terms, the method seeks directions in input space that contribute in average the most to the variation of the output. Those directions form the basis of a low-dimensional subspace of the original input space, the so-called *active subspace*. The curse of dimensionality is alleviated by substituting the active subspace to the original high-dimensional input space, followed by the identification for an approximate mapping whose input space is effectively lower than the original. It is a general-purpose method in the sense this alternate mapping can be used to assist any of the aforementioned numerical applications.

The AS method comes within the general scope of *approximation by ridge functions* [70] that seek to approximate the function of interest f with a mapping of the sort $f(\mathbf{x}) =$

$g(\mathbf{W}^T\mathbf{x})$ where $\mathbf{W}$ is a tall projection matrix onto a subspace of the input space $\mathcal{X}$ and g is a mapping whose input space is thus lower-dimensional than f's. The same idea was pursued independently in [97]. Other very similar methods were also independently proposed, in particular at the aerospace systems design laboratory (ASDL) [98, 99, 100]. While the AS method does not necessarily yield the subspace leading to the optimal ridge approximation [101], where optimality is defined with respect to the squared prediction error, it has been shown to result in useful predictive models based on approximation by ridge functions for numerous engineering applications [102, 103, 104, 105, 106, 107]. Extensions to the method are the subject of current research, such as multivariate outputs [108] and multi-fidelity setting [109].

The following paragraphs recall the major results pertaining to the AS method. Interested readers may find more details in [110]. In the context of this method, we equip the input variables $\mathbf{x}$ of f with a probability distribution $p(\mathbf{x})$. The matrix $\mathbf{C}$, which is average of the outer product of the gradient with itself, plays a central role in the AS method:

$$\mathbf{C} = \int (\nabla_{\mathbf{x}} f)(\nabla_{\mathbf{x}} f)^T p(\mathbf{x})\, d\mathbf{x} \tag{2.2}$$

This real symmetric matrix is diagonalized as $\mathbf{C} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$ where $\mathbf{Q} = [\mathbf{w_1}, \ldots, \mathbf{w_d}]$ is the matrix containing the normalized eigenvectors $\{\mathbf{w_i} \mid i = 1, \ldots, d\}$ of $\mathbf{C}$ and $\mathbf{\Lambda}$ is a diagonal matrix whose diagonal contains the eigenvalues $\{\lambda_i \mid i = 1, \ldots, d\}$ of $\mathbf{C}$. We assume the eigenvalues to be sorted in descending order such that $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$.

The following relationship links the eigenvalues with the projections of the function's gradient onto the corresponding eigenvectors [110]. For $i = 1, \ldots, d$:

$$\lambda_i = \int \left((\nabla_{\mathbf{x}} f)^T \mathbf{w_i} \right)^2 p(\mathbf{x})\, d\mathbf{x} \tag{2.3}$$

The higher the λ_i, the greater the variations in f in the direction $\mathbf{w_i}$. The active subspace is defined as the subspace spanning the first $m \leq d$ directions $\{\mathbf{w_i} \mid i = 1, \ldots, m\}$. In this subspace, the variations of f are in average greater than in its orthogonal complement, referred to as the *inactive subspace*.

Because they are the eigenvectors of a real symmetric matrix, the vectors $\{\mathbf{w_i} \mid i = 1, \ldots, d\}$ form an orthonormal basis of the input space. We can then arrange them into two matrices:

$$\mathbf{W} = [\mathbf{w_1}, \ldots, \mathbf{w_m}] \tag{2.4}$$

$$\mathbf{W_i} = [\mathbf{w_{m+1}}, \ldots, \mathbf{w_d}] \tag{2.5}$$

$\mathbf{W}$ is the $d \times m$ projection matrix onto the *active* subspace while $\mathbf{W_i}$ is the $d \times (d - m)$ projection matrix onto the *inactive* subspace. The original mapping of interest can then be rewritten as $f(\mathbf{x}) = f(\mathbf{W}\mathbf{z} + \mathbf{W_i}\mathbf{z_i})$ where $\mathbf{z} = \mathbf{W}^T\mathbf{x}$ is the component of the inputs in the AS and $\mathbf{z_i} = \mathbf{W_i}^T\mathbf{x}$ is the component of the inputs in the inactive subspace. Starting from this decomposition, a series of approximations can be made to obtain a practical AS-assisted surrogate modeling approach. Given the conditional probability $p(\mathbf{z_i}|\mathbf{z})$ of the inactive variables given the active variable, we start by defining the *link function* g as the conditional expectation of f over the inactive subspace given a position in the active subspace:

$$g(\mathbf{z}) = \mathbb{E}_{\mathbf{z_i}|\mathbf{z}}\left[f(\mathbf{W}\mathbf{z} + \mathbf{W_i}\mathbf{z_i})\right] \tag{2.6}$$

By assuming that the variations of f caused by variations of input variables in the inactive subspace are substantially less than those caused by variations in the inactive subspace, we obtain the following approximation:

$$f(\mathbf{x}) \approx g(\mathbf{W}^T\mathbf{x}) \tag{2.7}$$

Since the exact integration in (2.6) would be either too costly or simply not possible, a Monte-Carlo approximation $\hat{g}$ of g is used:

$$f(\mathbf{x}) \approx \hat{g}(\mathbf{W}^T\mathbf{x}) \tag{2.8}$$

The Monte-Carlo integration is shown to have good convergence properties in [110] since f does not by construction greatly vary in the inactive subspace. The function $\hat{g}$ is itself approximated by a surrogate model $\tilde{g}$ based on a limited number of model observations:

$$f(\mathbf{x}) \approx \tilde{g}(\mathbf{W}^T\mathbf{x}) \tag{2.9}$$

Finally, an approximation $\hat{\mathbf{W}}$ of $\mathbf{W}$ is obtained by replacing the integral with a finite sum in the computation of $\mathbf{C}$ in equation (2.2):

$$f(\mathbf{x}) \approx \tilde{g}(\hat{\mathbf{W}}^T\mathbf{x}) \tag{2.10}$$

AS-assisted surrogate modeling methods rely on equation (2.10) to approximate the original function f [110] by following a two-step approach. An approximation $\hat{\mathbf{W}}$ of the projection matrix $\mathbf{W}$ onto the AS is first computed using a sample of gradient evaluations. An approximation $\tilde{g}$ of the link function g is then constructed using traditional surrogate modeling techniques by substituting the projected design matrix $\mathbf{Z} = \mathbf{XW}$ to the original design matrix $\mathbf{X}$ in the training process.

Conclusion on AS The gradient-based AS method [111] is among the most recent additions to the mapping or projection-based strategies. It has shown considerable promise on challenging engineering applications [102, 110, 104, 105, 106, 107]. Extensions of the method to multivariate outputs [108, 112] and to the multi-fidelity setting [109] are the subject of ongoing research. Similar ideas were also independently pursued [113, 99]. While the AS method is in practice restricted to problems for which adjoint derivatives can be obtained, alternative formulations have been proposed to identify the low-dimensional structure of the input space using direct function evaluations only. Through its applications, AS proves that mapping strategies have potential in assisting to solve high-dimensional problems, and in particular the task of interest in this thesis: the generation of surrogate models with high-dimensional inputs.

However, it also has shortcomings. First, the AS method is in practice enabled by adjoint methods or other means by which gradient evaluations can be obtained at a negligible additional computational cost. Without such methods, access to gradients in a high-dimensional setting is prohibitively expensive. Given 1) the variety of models that are used in conceptual design, 2) the proprietary nature of codes, 3) the fact that adjoints are only available for PDE-based solvers, and 4) the effort required to add adjoint capabilities to a

solver, we argue that the requirement to have gradients may be excessive in certain organizational contexts. Second, it requires an initial batch run of the high-fidelity model in the high-dimensional space in order to estimate the covariance matrix $\mathbf{C}$ of the gradient. This step is necessary because there is no prior knowledge of the function under consideration. However, it means that computationally expensive simulations are being performed without guarantee that they actually bring useful information for the generation of the surrogate model. However, once the active subspace is known, then very few high-fidelity samples are required. If there was another way to obtain these directions, without gradients and without running the high-fidelity code, this would tremendously decrease the offline cost of training the surrogate.

Observation

While the need of the AS method for direct access to gradient evaluations strongly restricts its applicability, the various examples from science and engineering on which it has been applied demonstrate that knowledge of a response's AS is often an effective way to alleviate the curse of dimensionality.

In the next section, we will discuss recent developments in the field of approximation by ridge functions, which were in part motivated by the AS method, but that do not require gradients.

Recent Developments

Gaussian Processes with Built-In Dimensionality Reduction When direct or cheap gradient evaluations are not available, the AS method presented in section 2.1.3 may not be practically applicable, as gradient values would first need to be estimated, e.g., using finite differences, and such schemes require a large number of direct function evaluations when the input dimension d is high.

An alternative approach proposed in [91] is to simultaneously train the approximate

link function $\tilde{g}$ and the approximate projection matrix $\hat{\mathbf{W}}$ onto the AS. The underlying predictive model combines aspects of the original AS method with Gaussian processes: the original inputs are projected onto a low-dimensional subspace that serves as the alternate, low-dimensional input space for a GP. This leads to the following generative model:

$$\mathbf{y}|\theta, \mathbf{W} \sim \mathcal{N}\left(0, K\left(\mathbf{XW}, \mathbf{XW}; \theta\right) + \sigma_n^2 \mathbf{I}\right) \tag{2.11}$$

This model can be broken down into two steps. An initial *projection step* where the original design matrix $\mathbf{X}$ is projected in the AS to obtain a lower-dimensional design matrix $\mathbf{Z} = \mathbf{XW}$ followed by a *regression step* in which the lower-dimensional space is substituted to the original high-dimensional input space.

Compared to traditional GPR, this approach leads to the effective dimension reduction of GP's input space from the original d inputs to only m inputs. As a consequence, the number of GP hyperparameters is also decreased from $d + 2$ to $m + 2$ when using the automatic relevance determination (ARD) kernel. However, the projection matrix $\mathbf{W}$ is introduced as a new model parameter that must be determined in the training process. Previous methods have leveraged matrix manifolds, later discussed in greater details in section 2.2.2, to handle the projection matrix: , [91, 114] use the Stiefel manifold while [115] uses the Grassmann manifold. While they differ by their specifics, these methods all rely on optimization algorithms in manifolds, for which theory is well-established [116] and numerical implementations are readily available, such as *Pymanopt* [117] used in [115].

Conclusion

Surrogate modeling methods relying on approximation by ridge functions are promising to alleviate the curse of dimensionality. They act by replacing the original high-dimensional input space with an alternate lower-dimensional subspace in which the curse of dimensionality does not apply. However, we made the following observation.

Observation

Existing surrogate modeling methods that handle high-dimensional input spaces require access to gradients or many observations.

Gradients are rarely directly accessible, and the high-dimensional inputs make gradient approximation techniques impractical. Under realistic organizational constraints that limit the total available computational budget, the high cost of individual model evaluations limits the number of times the expensive model can be queried. The resulting sparsity of model observations is the main barrier to the application of approximation by ridge functions to expensive models.

Observation

Without the ability to use methods based on approximations by ridge functions when gradients are unavailable and training data is sparse, it is challenging to create surrogate models with high-dimensional inputs in these conditions.

2.1.4 Conclusion

The curse of dimensionality may be alleviated in the context of surrogate modeling using approximations by ridge functions, which effectively substitute a *low-dimensional feature space* to the original *high-dimensional input space*. However, these approaches become challenging to apply when evaluation costs are high because they rely on the availability of either 1) a large number of model evaluations or 2) gradient evaluations or approximations thereof.

Gap 1

Surrogate modeling methods relying on approximations by ridge functions provide an effective way to alleviate the curse of dimensionality for high-dimensional input spaces. However, they either require gradients or a large number of model observations, making them inapplicable when direct gradients evaluations are not available and only few analysis observations are affordable due to high analysis costs.

This leads to the refined version of research question 1:

Refined Research Question 1

How can we enable surrogate modeling methods based on approximation by ridge functions without access to gradients and when only few analysis observations are available?

Predictive modeling methods can be made more robust to the lack of data by working in a Bayesian framework. Traditional training methods based on the optimization of a mathematical model's parameters, such as maximum likelihood estimation (MLE), suffer from over-fitting when training data is sparse, leading do to predictive models that do not generalize well to unseen data. In a Bayesian approach, we seek posterior distributions for the model parameters instead of point-based optimal values. In practice, such approaches avoid over-fitting the model parameters to the limited number of available observations by acknowledging that a range of model parameter values may be adequate to explain the training observations. This generally leads to models that generalize better when validated against data not used for training. By applying a Bayesian approach to training surrogate models based on approximation by ridge functions, we expect to take advantage of these benefits and eventually increase predictive accuracy compared to traditional training approaches, specifically in situations where only a low number of analysis observations are available for training. The Bayesian framework is the focus of the next section.

2.2 Fully Bayesian Approach to Approximation by Ridge Functions

2.2.1 Background and Research Objective

In section 2.1, we have established that approximation by ridge functions is a promising way to overcome the curse of dimensionality. Methods such as AS have been shown to successfully assist the creation of surrogate models with high-dimensional inputs on multiple engineering-related examples. However, we also observed that existing methods either require many samples or access to gradients. The need for many samples is problematic in situations where the analysis cost is high and only few analysis observations are affordable. In the context of high-dimensional inputs, access to gradients is dependent on the availability of analytical expressions or computational tools such as adjoint method, since approximations such as finite differences are prohibitively expensive. In this section, we are going to discuss methods that may help in overcoming these limitations. In the section "Overview of the Bayesian Framework", we will introduce the Bayesian framework and explain how it results into surrogate models that are more robust to the lack of analysis observations by enabling a full quantification of epistemic uncertainty due to limited training data. We will see that the application of Bayesian methods on approximation by ridge functions is however not directly possible, as orthogonal matrices are needed to model the projection onto the lower-dimensional subspace, whereas available Bayesian inference frameworks work with real-valued parameters. This leads to the capability gap presented in section 2.2.1 and the corresponding research question. We then return to the literature to seek and discuss methods relevant to filling this gap. We will at that point be equipped to formulate the proposed method in the next section, leading to the conclusion of the section.

The formulation and results presented in this section of the thesis have been the subject of a journal publication by the author [118].

Overview of the Bayesian Framework

The Bayesian framework uses probability theory as a way to quantify uncertainty, as opposed to the traditional frequentist approach in which probabilities are viewed as "the frequencies of random, repeatable events" [48]. Multiple references are available to the reader willing to develop a better understanding of the Bayesian framework, such as [48, 119, 120]. In this section, we will limit ourselves to a short introduction to the concept such as to underline how it may help overcoming the challenges of training surrogate models based on approximation by ridge functions when the number of analysis observations is low. Then, in the next section, "Example: Bayesian Linear Regression", we will discuss a Bayesian approach to linear regression. Finally, in the section "Practical Implementation: Markov-Chain Monte-Carlo Samplers", we will briefly present computational methods used to carry out the practical implementation of Bayesian approaches, as we will see that they have some limitations when it comes to the nature of the parameters encountered in approximation by ridge functions.

As its name implies, Bayes' formula is at the core of Bayesian statistics. Here, we place ourselves in the context of supervised learning to highlight the driving principles. Let us assume that we are building a model parameterized by a vector θ of parameters, using observations denoted as $\mathcal{D}$. In the supervised learning context, observations would typically consist of a set of input-output pairs. In this context, "training" the model consists in inferring the posterior probability $p(\theta \mid \mathcal{D})$ of the model's parameters given the observed data. This is in contrast with traditional training where a point value $\theta_\mathbf{0}$ for the parameters θ is sought. The posterior probability can be expressed using Bayes' formula:

$$p(\theta \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \theta)p(\theta)}{p(\mathcal{D})} \tag{2.12}$$

The quantity $p(\theta)$ is the prior distribution for the parameters. It encodes the prior belief about the values of the model parameters and its choice is left to the model's designer. $p(\mathcal{D} \mid \theta)$ is referred to as the likelihood function: it is the probability of observing the

training data given a specific value of the model parameters. Because the observed data is fixed, $p(\mathcal{D} \mid \theta)$ is usually thought of as a function of the model parameters θ: as we vary the model's parameters, the probability of observing the training data varies. In a maximum likelihood estimation (MLE) approach, this is the quantity that is maximized to select a point-based value for the parameters θ. $p(\mathcal{D})$ is usually thought of a normalization constant instead of a meaningful quantity. Although it is usually not necessary to compute it in practical approaches to Bayesian inference, it may be obtained by marginalizing the model's parameters:

$$p(\mathcal{D}) = \int p(\mathcal{D} \mid \theta) p(\theta) d\theta \tag{2.13}$$

For simple models, such as the Bayesian linear regression discussed in the next section, $p(\theta \mid \mathcal{D})$ is tractable, in the sense that a closed-form expression can be derived. As model complexity increases, Bayesian models quickly become intractable, and approximate inference methods are needed, such as MCMC that is discussed in the following section.

Example: Bayesian Linear Regression

We illustrate the process of training a predictive model in a Bayesian manner on the linear regression of a straight line. This is in contrast to the more general of linear regression where explanatory variables may be chosen as basis functions that are function of the original input parameters. The example is – on purpose – very simple and idealized, as we assume extensive knowledge of the mapping for which we are creating a surrogate model. Namely, we assume that observations are generated from a process of the form $y = a\,x + b + \epsilon$ where:

- x and y are the real-valued inputs and outputs of the mapping that we are observing;
- a and b are unknown real-valued parameters to be inferred;
- ϵ is a zero-mean noise of known variance σ_n^2: $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$

The objective of the training is to determine the probability distributions for the parameters a and b given a set of observed N input-output pairs $\mathbf{x}, \mathbf{y}$. Let us introduce the vector of parameters $\theta = [a\ b]^T$. We equip θ with a prior distribution that reflects our initial belief, or lack thereof, about the relationship between x and y: $\theta \sim \mathcal{N}(0, \mathbf{I_{2\times 2}})$.

Under these assumptions, the posterior distribution for the model parameters may be computed in closed-form [48]:

$$\theta \mid \mathbf{x}, \mathbf{y} \sim \mathcal{N}(\mathbf{m}, \mathbf{\Sigma}) \tag{2.14}$$

With:

$$\phi = \begin{bmatrix} x_0 & 1 \\ \vdots & \vdots \\ x_N & 1 \end{bmatrix} \tag{2.15}$$

$$\mathbf{m} = \frac{\mathbf{\Sigma}\phi^T\mathbf{y}}{\sigma_n^2} \tag{2.16}$$

$$\mathbf{\Sigma} = \left(\mathbf{I_{2\times 2}} + \frac{\phi^T\phi}{\sigma_n^2}\right)^{-1} \tag{2.17}$$

Figure 2.9 is a graphical depiction of this example. Each row corresponds to a different number of training observations, starting with none in the first row and increasing towards the bottom. The left column depicts the likelihood function at each stage. Accordingly to the explanation made above, it is shown as a function of the model parameters a and b. The middle column depicts the posterior distribution for the parameters, which combines the prior parameters distribution with the evidence from the observations. Finally, the right-most column depicts, in "data space", 1) the training observations, 2) the actual underlying linear function, and 3) the model prediction and its uncertainty.

We can see that the more observations become available for training the model, the better the posterior probability distribution for the model parameters is defined, and, in turn, the better and less uncertain the prediction data space. This is consistent with an intuitive understanding of the situation: the more training data is at our disposal, the more

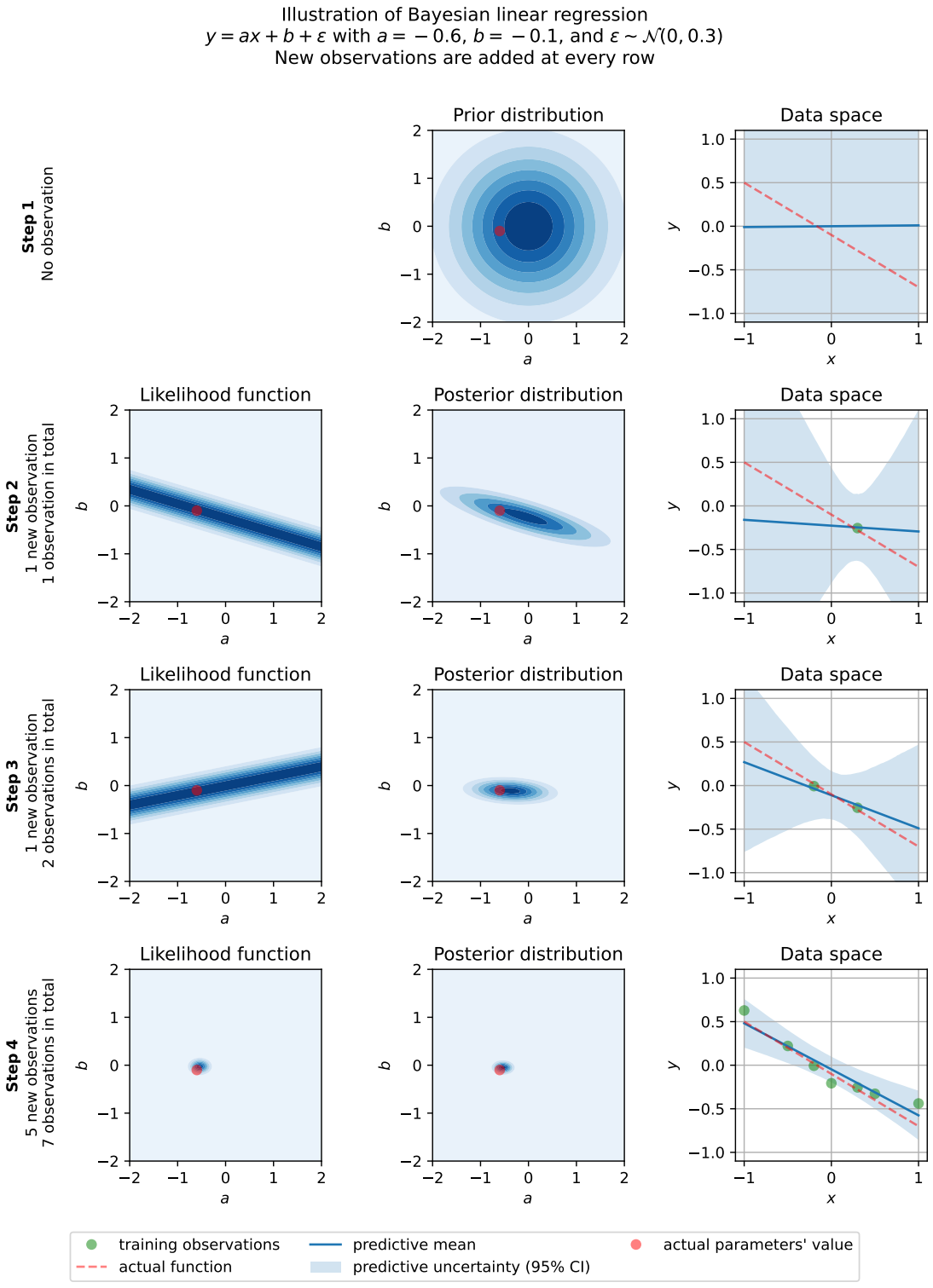


Figure 2.9: Illustration of Bayesian linear regression

confident we are in our estimation of the model parameters. In particular, when little data is available, the Bayesian approach recognizes that many parametrizations of the linear model, i.e., specific values for a and b, may be compatible with the data being observed. In contrast, a conventional approach to training would have selected a single value for these parameters despite only being provided with a limited amount of evidence. In that sense, a Bayesian approach makes the training of the model more robust to the limited number of observations.

While the posterior distribution of the model's parameters was tractable in this simple example, it is generally not. The next section briefly introduces MCMC, a computational approach to approximating the posterior distribution when it is not tractable.

Practical Implementation: Markov-Chain Monte-Carlo Samplers

In situations where the posterior distribution $p(\theta \mid \mathcal{D})$ is not tractable, an alternative approach is to use MCMC to draw a chain of samples from it [121]. MCMC methods rely on an iterative sampling scheme in which successive draws are either accepted, and added to the final chain, or rejected, leading to repeated samples in the final chain. The process by which samples are either accepted or rejected leads to draws of the resulting chain to follow the desired posterior distribution. These methods only require the computation of a quantity proportional to the posterior distribution $p(\theta \mid \mathcal{D})$, thus not requiring the computation of the constant evidence term $p(\mathcal{D})$. Only the likelihood $p(\mathcal{D} \mid \theta)$ of the model and prior distributions $p(\theta)$ for its parameters need to be specified. Existing MCMC methods usually operate on unconstrained real-valued parameters.

Observations The discussions from this section lead to the following observations:

Observations

- The Bayesian framework offers a principled manner to quantify the uncertainty of a model's parameters due to limited training data.
- Quantifying uncertainty due to limited training data grants robustness against the sparseness of analysis observations.

A Bayesian approach therefore appears as a promising way to alleviate the challenges raised due to the curse of dimensionality. In the next section, we will be considering how a Bayesian approach could be undertaken for training a surrogate model based on approximation by ridge functions.

Research Question 1.1

The motivation for considering the Bayesian framework was to make existing models to approximation by ridge functions more robust to limited analysis observations by enabling full quantification of epistemic uncertainty. Even though not fully Bayesian, probabilistic approaches to supervised dimension reduction that are relevant in the context of this thesis can be found in the literature; we briefly present them in this section.

[91] combines a linear projection with a GP and includes the coefficients of the projection matrix as additional hyperparameters to the GP covariance matrix. Hyperparameters are then optimized using a maximum likelihood approach. This approach benefits from a partial quantification of the epistemic uncertainty as a GP is employed, but the uncertainty due to the linear projection's parameters is not tracked. This method will be discussed in greater details in the first section. [92] leveraged geodesic MCMC in order to adopt a fully Bayesian approach to inferring the probability distribution of the projection matrix. polynomial chaos expansion (PCE) are employed as link functions instead of GPs, as their end goal is to perform uncertainty quantification. As a result, epistemic uncertainty due to limited data is not captured.

To that end, some methods combine the projection onto the AS with a probabilistic surrogate model such as GP regression [91, 114, 112]. However, they only partially quantify epistemic uncertainty due to limited data, as the projection matrix on the AS results from an optimization process instead of a Bayesian inference process. While similar approaches have been proposed for other kinds of surrogates [122, 92, 62], the fully Bayesian inference of a predictive model combining a projection onto a lower-dimensional subspace and a Gaussian process (GP) has not yet been attempted. Such an approach would open the door to efficient metamodeling for high-dimensional problems using GPs, including a full quantification of epistemic uncertainty introduced by limited data.

Including an orthonormal projection matrix in the predictive model however complicates the Bayesian inference process, as readily available MCMC samplers only operate with real-valued model parameters. This leads to the following capability gap:

Gap 1.1

Without the ability to perform MCMC on orthogonal matrices using turn-key MCMC samplers, it is impractical to train surrogate models based on approximation by ridge functions in a Bayesian fashion.

Accordingly, we introduce the following research question:

In the next section, we seek potential solutions to this research question in methods for Bayesian inference of orthonormal matrices. We will see that, to address the Bayesian inference of orthonormal matrices, it is useful to recognize that orthonormal matrices and linear subspaces are sets that can be equipped with a manifold structure, respectively forming the Stiefel and Grassmann manifolds. The mathematical framework of manifolds grants a principled way of dealing with sets of relatively complex mathematical entities, such as orthonormal matrices, with the same tools that are routinely employed in Euclidean spaces, such as differentiation. We will see that this is key to finding methods to fill the capability gap identified here.

2.2.2 Proposed Method

This section focuses on the development and presentation of the proposed approach. We will start by recalling the background on the existing techniques that underlie the proposed method before detailing its specifics.

Stiefel and Grassmann Manifolds

The previous section highlighted the central role played by orthogonal projection matrices in approximation by ridge function. The presence of an orthonormal projection matrix in the predictive model may be dealt with using real-valued model parameters and additional sets of constraints. However, satisfying the orthonormality constraint adds to the computational burden of training the predictive model. Instead, we recognize that the set of orthonormal matrices may be equipped with a manifold structure. A detailed explanation of the mathematical concepts surrounding manifolds is out of the scope of this paper, and a clear introduction to those can be found in [116], from which we adapted the definitions given in this section. A main benefit of working in manifolds is the well-defined transposition of differential calculus operations routinely made with real-valued parameters to orthonormal matrices.

The set of orthonormal matrices forms the Stiefel manifold [116], whose definition, adapted from [116], is given below.

Definition 1 (Stiefel Manifold)**.** Let $\mathrm{St}(p, n)$ $(p \leq n)$ denote the set of all $n \times p$ orthonormal matrices

$$\{\mathbf{X} \in \mathbb{R}^{n \times p} : \mathbf{X}^T \mathbf{X} = \mathbf{I_p}\} \tag{2.18}$$

where I_p denotes the $p \times p$ identity matrix. Endowed with its manifold structure, the set $\mathrm{St}(p, n)$ is called the Stiefel manifold.

A related manifold is the Grassmann manifold, defined below.

Definition 2 (Grassmann Manifold)**.** Let $\mathrm{Gr}(p, n)$ be the set of all p-dimensional subspaces of $\mathbb{R}^n$. Endowed with its manifold structure, the set $\mathrm{Gr}(p, n)$ is called the Grassmann manifold

By these definitions, given p and n, we note that for every element $g \in \mathrm{Gr}(p, n)$ of the Grassmann manifold, we may find infinitely many elements $\{s \in \mathrm{St}(p, n) : \mathrm{span}(s) = g\}$ in the Stiefel manifold whose span is g, all being orthonormal bases of the subspace g that only differ by a rotation within g. As such, going from an orthonormal projection matrix in the Stiefel manifold to a subspace in the Grassmann manifold may be interpreted as retaining the information regarding the subspace spanned by this projection matrix, but losing the information regarding the exact orientation of the coordinate axes described by the matrix within that subspace.

When formulating our method in the next section, we will discuss the relevance of Stiefel against Grassmann manifold in the context of approximation by ridge functions, when the low-dimensional subspace is used as an alternate input space for a link function. Now that we have recognized that orthogonal matrices may be handled as these particular mathematical entities, it opens the door to new methods for including them as part of a probabilistic model while still and still carry out Bayesian inference. These methods are discussed in the next section.

MCMC for Orthonormal Matrices

Methods to apply MCMC when some model parameters belong to these manifolds have been developed following two distinct strategies: Riemannian Hamiltonian Monte-Carlo (RHMC) techniques [123, 124] and reparametrization techniques [125, 126, 127, 128]. The former modify the leapfrog steps of the original Hamiltonian Monte-Carlo (HMC) algorithm to ensure that new proposals remain on the relevant manifold. For the Stiefel manifold, this means that the orthogonality constraint is satisfied by construction. The latter simply use parameterization techniques, such as Householder reflections [126], polar

angles [127], the Givens representation [128], or simply a Gram-Schmidt orthogonalization scheme [129] to transform a set of real-valued parameters into an orthogonal matrix.

While RHMC methods are theoretically elegant and lie on strong mathematical foundations, they require the implementation of specialized MCMC samplers [130]. "Turn-key" MCMC algorithms that do not require extensive tuning have not yet been developed for RHMC. As a consequence, implementations and usages of these methods are still at a very early stage of research. The method recently proposed in [131] is representative of such approaches. On the other hand, reparameterization approaches have been successfully leveraged using openly-available MCMC samplers in instances where orthogonal matrices were part of the probabilistic model [125, 126, 127, 128]. However, they have not yet been applied to the fully Bayesian inference of a low-dimensional input subspace in the context of supervised learning. This is the approach taken in this work, eventually allowing us to leverage existing MCMC algorithms to fully quantify uncertainty in the predictive model. By that, we mean both the uncertainty in the projection matrix onto a low-dimensional subspace as well as the uncertainty in the hyperparameters of the Gaussian process. In the following, because it can be thought of as a reduced set of relevant input features, we will refer to the low-dimensional linear subspace of the original input space that we are seeking as the feature space.

These methods, while having not yet been applied in the context of a fully Bayesian approach to approximation by ridge functions, fulfill our needs: they effectively enable to include orthonormal matrices in a probabilistic and still carry out Bayesian inference through Markov chain Monte-Carlo (MCMC) using turn-key samplers. The next step is therefore to formulate the proposed method, which is the focus of the next section.

Notation

We briefly introduce the notation used needed to formalize the presentation of the proposed method. Let f be the mapping through which we collect data about the underlying physical

process of interest. We assume that f is a scalar-valued function of d variables:

$$\begin{aligned} f : \mathcal{X} \subseteq \mathbb{R}^d &\longrightarrow \mathcal{Y} \subseteq \mathbb{R} \\ \mathbf{x} &\longmapsto y = f(\mathbf{x}) \end{aligned} \tag{2.19}$$

The input vector $\mathbf{x} \in \mathbb{R}^d$ while the output response $y \in \mathbb{R}$. The proposed approach targets high-dimensional input spaces, i.e., large values of d.

We assume that a total of n input-output pairs, or model observations, have been obtained by evaluating the model f at the input sites $\mathbf{x_1}, \ldots, \mathbf{x_n}$ and are available to support the creation of the surrogate model. Following standard practice, we partition model observations into a training set $\mathcal{D}$ of size p and a validation set $\mathcal{D}_*$ of size q such that $p + q = n$. In the context of surrogate modeling, the observations in $\mathcal{D}$ are used for training the model while those in $\mathcal{D}_*$ allow to assess the performance of the predictive model.

$$\mathcal{D} = \{(\mathbf{x_i}, y_i) \mid i = 1, \ldots, p\} \tag{2.20}$$

$$\mathcal{D}_* = \{(\mathbf{x_i^*}, y_i^*) \mid i = 1, \ldots, q\} \tag{2.21}$$

Let $\mathbf{X} = [\mathbf{x_1}, \ldots, \mathbf{x_p}]^T$ and $\mathbf{X}_* = \left[\mathbf{x_1^*}, \ldots, \mathbf{x_q^*}\right]^T$ respectively be the $p \times d$ and $q \times d$ design matrices corresponding to the training and validation input sites. Accordingly, let $\mathbf{y}$ and $\mathbf{y}_*$ be the size p and q training and validation response vectors. We focus on the creation of a surrogate model $\hat{f}$ of f, i.e. such that $\hat{f}$ can be evaluated in lieu of f.

We work within the probabilistic framework to quantify uncertainty in the surrogate model predictions. As such, we are seeking a generative model, i.e., a model for the joint probability distribution $p(\mathbf{X}, \mathbf{y})$. Probabilistic predictions for points in the validation set can then be made by using the conditional predictive distribution $p(\mathbf{y}_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y})$. In the following, we denote a multivariate normal distribution with mean $\mathbf{m}$ and covariance matrix $\mathbf{V}$ as $\mathcal{N}(\mathbf{m}, \mathbf{V})$, and the identity matrix, whose size can be deduced from context, as $\mathbf{I}$.

Gaussian Process Regression

GPR is a popular surrogate modeling method that grants access to predictive uncertainty. We briefly recall the main idea behind GPR, mostly adopting the reference notation introduced in [132]. The GPR model relies on the assumption that the prior distribution for the underlying mapping f can be modeled as a GP: for a set of latent, unobserved, model responses made at input sites $\mathbf{X}$ and arranged in the vector $\mathbf{f}$, there exist a vector μ and a matrix $\mathbf{\Sigma}$ such that $\mathbf{f} \sim \mathcal{N}(\mu, \mathbf{\Sigma})$.

Additionally, it is assumed that observations $\mathbf{y}$ of $\mathbf{f}$ are independently affected by a zero-mean Gaussian random variable of variance σ_n^2 such that $\mathbf{y}|\mathbf{f} \sim \mathcal{N}(\mathbf{f}, \sigma_n^2\mathbf{I})$. This assumption is usually employed to model the effect of measurement noise on experimental results. In the context of numerical simulations, measurement noise is irrelevant. However, the process of reducing the input space dimension by projecting it onto a low-dimensional subspace introduces artificial noise corresponding to the variations of the output due to variations of the inputs in the orthogonal complement of the low-dimensional subspace. In this work, this artificially introduced noise is accounted for by assuming noisy observations.

Those assumptions enable the analytical marginalization of the latent vector $\mathbf{f}$, leading to the following prior distribution for the observations: $\mathbf{y} \sim \mathcal{N}(\mu, \mathbf{\Sigma} + \sigma_n^2\mathbf{I})$. Multiple options for constructing the mean vector μ and the covariance matrix $\mathbf{\Sigma}$ exist and lead to different GPR variants. Common assumptions are made in this work. First, we assume the model observations to be centered and the mean of the prior GP to be zero, i.e., $\mu = [0, \ldots, 0]^T$. Then, a kernel function k is used to construct $\mathbf{\Sigma}$ by encoding the correlation structure of the GP, i.e.. the correlation between responses y and y' at input input locations $\mathbf{x}$ and $\mathbf{x}'$. We use the widespread automatic relevance determination (ARD) kernel:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \prod_{i=1}^{d} \exp\left(\frac{(x_i - {x_i}')^2}{2\ell_i^2}\right) \tag{2.22}$$

where σ_f^2 is the signal variance and $\ell = [\ell_i, ..., \ell_d]$ are characteristic length scales. For convenience, we denote as K the generalization of the kernel function k to design matrices.

For two design matrices $\mathbf{X}$ and $\mathbf{X}'$ respectively containing n and n' points, we have:

$$K(\mathbf{X}, \mathbf{X}') = \begin{bmatrix} k(\mathbf{x_1}, \mathbf{x'_1}) & \dots & k(\mathbf{x_1}, \mathbf{x'_{n'}}) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x_n}, \mathbf{x'_1}) & \dots & k(\mathbf{x_n}, \mathbf{x'_{n'}}) \end{bmatrix} \tag{2.23}$$

Using that notation, the GP covariance matrix is then computed as $\mathbf{\Sigma} = K(\mathbf{X}, \mathbf{X})$. We gather all hyperparameters into the vector $\theta = [\sigma_n, \sigma_f, \ell_1, \dots, \ell_d]$. Rewriting the generative model with those assumptions and explicitly including parameters, we obtain equation (2.24). Varying the hyperparameters θ effectively leads to different generative models.

$$\mathbf{y}|\theta \sim \mathcal{N}(0, K(\mathbf{X}, \mathbf{X}; \sigma_f, \ell) + \sigma_n^2 \mathbf{I}) \tag{2.24}$$

In a MLE approach, training the GPR model then consists in estimating the values of the hyperparameters θ leading to the generative model that is most in agreement with the training data. This is achieved by selecting θ that maximizes the likelihood $p(\mathbf{y}|\mathbf{X}, \theta)$. A closed-form equation for the likelihood $p(\mathbf{y}|\mathbf{X}, \theta)$ is made possible by the GP assumption. In practice, the log-likelihood $\log p(\mathbf{y}|X, \theta)$ is used for numerical stability, Given training data $(\mathbf{X}, \mathbf{y})$ and denoting $\mathbf{K} = K(\mathbf{X}, \mathbf{X})$:

$$\log p(\mathbf{y}|X, \theta) = -\frac{1}{2}\mathbf{y}^T \left(\mathbf{K} + \sigma_n^2 I\right)^{-1} \mathbf{y} - \frac{1}{2} \log \det \left(\mathbf{K} + \sigma_n^2 I\right) - \frac{n}{2} \log 2\pi \tag{2.25}$$

In a Bayesian approach, hyperparameters are equipped with a prior distribution $p(\theta)$ and the full posterior distribution of the hyperparameters $p(\theta|\mathbf{y}, \mathbf{X})$ is inferred by leveraging Bayes' rule:

$$p(\theta|\mathbf{y}, \mathbf{X}) \propto p(\mathbf{y}|\mathbf{X}, \theta) p(\theta) \tag{2.26}$$

Predictions $\mathbf{y}_*$ at validation locations $\mathbf{X}_*$ can be made by recalling that the underlying process is assumed to be a GP, therefore the training and test outputs are distributed according to the following joint probability distribution:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I} & K(\mathbf{X}, \mathbf{X}_*) \\ K(\mathbf{X}_*, \mathbf{X}) & K(\mathbf{X}_*, \mathbf{X}_*) + \sigma_n^2 \mathbf{I} \end{bmatrix}\right) \tag{2.27}$$

The posterior predictive distribution is obtained by conditioning the joint distribution with respect to the training data. Once again, the GP assumption allows to derive this joint distribution analytically:

$$\mathbf{y}_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y}, \theta \sim \mathcal{N}(\mu_*, \mathbf{\Sigma}_*) \tag{2.28}$$

with:

$$\mu_* = K(\mathbf{X}_*, \mathbf{X}) \left(K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I\right)^{-1} \mathbf{y} \tag{2.29}$$

$$\mathbf{\Sigma}_* = K(\mathbf{X}_*, \mathbf{X}_*) - K(\mathbf{X}_*, \mathbf{X}) \left(K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I\right)^{-1} K(\mathbf{X}, \mathbf{X}_*) \tag{2.30}$$

In the fully Bayesian approach to GPR, the hyperparameters need to be marginalized out using their posterior distribution to make predictions:

$$p(\mathbf{y}_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y}) = \int p(\mathbf{y}_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y}, \theta) p(\theta|\mathbf{X}, \mathbf{y})\, d\theta \tag{2.31}$$

Proposed Fully Bayesian Approach

From now on, we will refer to the low-dimensional subspace of the input space used for dimension reduction as the feature space (FS) to recognize that it may be different from the AS. The optimization-based approaches to approximation by ridge functions relying on GPR discussed previously do not enable a full quantification of epistemic uncertainty due to limited analysis observations. After projection of the design matrix onto the FS, predictive uncertainty only originates from the assumption that the link function is modeled as a GP. However, neither the uncertainty in the GP hyperparameters nor in the projection matrix $\mathbf{W}$ are quantified. The probabilistic model used in the proposed approach is similar to the one in equation 2.11. However, we adopt a fully Bayesian approach to training its parameters such that full posterior probability distributions for both the GP hyperparameters θ and the projection matrix $\mathbf{W}$ are obtained. As a result, the predictive uncertainty obtained when querying the surrogate model at unobserved input locations accounts for all uncertain model parameters.

In view of existing methods [91, 114, 112], the choice of the relevant manifold for the projection matrix $\mathbf{W}$ remains. The Grassmann manifold may appear as the natural choice, since we are seeking a low-dimensional subspace to substitute to the original input space. However, the GPR model with an ARD kernel that was chosen to model the link function g grants different length scales to the different input directions of the GP. As such, the particular choice of basis for a given subspace matters when using the ARD kernel, not just the subspace. In other words, the directional information that is retained for elements of the Stiefel manifold but lost for those in the Grassmann manifold is required. For this reason, we set $\mathbf{W} \in \mathrm{St}(m, d)$.

As opposed to optimization in manifolds, Bayesian inference in manifolds has been developed more recently [123, 130]. Numerical implementations of those algorithms are not yet mature, thus restricting practical Bayesian inference to parameters defined in Euclidean spaces [133, 95]. In order to accommodate these limitations, reparametrization techniques, that map a set of real-valued parameters to a point on a manifold, have been used [126]. In this manner, tools operating with real-valued parameters may be leveraged to perform Bayesian inference in manifolds. Because we chose to work in the Stiefel manifold, the reparametrization mapping we use associates a vector $\theta_{\mathbf{p}}$ of k real parameters to a $d \times m$ orthonormal matrix $\mathbf{W}$:

$$\begin{aligned} \mathcal{P} : \mathbb{R}^k &\longrightarrow \mathrm{St}(m, d) \\ \theta_{\mathbf{p}} &\longmapsto \mathbf{W} = \mathcal{P}(\theta_{\mathbf{p}}) \end{aligned} \tag{2.32}$$

The choice of $\mathcal{P}$ is driven by the need to equip $\mathbf{W}$ with a meaningful prior distribution while recalling that the matching distribution on the parameters $\theta_{\mathbf{p}}$ is the one that must be specified when implementing the probabilistic model. In other words, along with $\mathcal{P}$, the distribution of $\theta_{\mathbf{p}}$ that results in the desired distribution for $\mathbf{W}$ is needed. The prior distribution placed on $\mathbf{W}$ must convey the prior belief that any set of orthonormal directions are *a priori* equally probable candidates, i.e., $p(\mathbf{W})$ should be a uniform distribution on $\mathrm{St}(m, d)$.

Algorithm 1: $\mathcal{H}$: orthonormal matrix parametrization through Householder transformations

input : parameters $\theta_{\mathbf{p}} \in \mathbb{R}^k$
output: projection matrix $\mathbf{W} \in \mathrm{St}(m, d)$

$\mathbf{Q} \leftarrow \mathbf{I} \in \mathbb{R}^{d \times d}$
$l \leftarrow 0$
for $i \leftarrow 1$ **to** m **do**
  $k \leftarrow l$
  $l \leftarrow k + d - i$
  $\mathbf{v} \leftarrow (\theta_{p,k}, \ldots, \theta_{p,l})^T$
  $\mathbf{u} \leftarrow \dfrac{\mathbf{v} + \mathrm{sgn}(v_1)||\mathbf{v}||\mathbf{e_1}}{||\mathbf{v} + \mathrm{sgn}(v_1)||\mathbf{v}||\mathbf{e_1}||}$
  $\hat{\mathbf{H}} \leftarrow -\mathrm{sgn}(v_1)(\mathbf{I} - 2\mathbf{u}\mathbf{u}^T)$
  $\mathbf{H} \leftarrow \begin{pmatrix} \mathbf{I} & 0 \\ 0 & \hat{\mathbf{H}} \end{pmatrix}$
  $\mathbf{Q} \leftarrow \mathbf{H}\mathbf{Q}$
end
$\mathbf{W} \leftarrow (\mathbf{Q_1}, \ldots, \mathbf{Q_m}) \in \mathbb{R}^{d \times m}$

The Stiefel manifold can be endowed with a uniform measure that is a Haar measure, i.e., it remains unchanged by the application of orthogonal transformations: $p(\mathbf{W}) = p(\mathbf{Q}\mathbf{W})\ \forall \mathbf{Q} \in O(d)$ where $O(d)$ is the orthogonal group [126]. As shown in [126], if the projection parameters $\theta_{\mathbf{p}}$ are independent and identically distributed (i.i.d.) Gaussian random variables and the Householder parametrization $\mathcal{H}$ detailed in algorithm 1 is used for the reparametrization mapping $\mathcal{P}$ such that $\mathbf{W} = \mathcal{H}(\theta_{\mathbf{p}})$, then $\mathbf{W}$ is a random orthogonal matrix with distribution given by the Haar measure in $\mathrm{St}(m, d)$, which is the desired prior distribution for $\mathbf{W}$.

Compared to other reparametrization methods, the Householder transformation has the advantage of not requiring a computationally burdensome change of measure [126]. The number k of real-valued parameters $\theta_{\mathbf{p}}$ is $k = md - m(m-1)/2$, which is larger than the actual dimension of the Stiefel manifold: $\dim(\mathrm{St}(m, d))) = md - m(m+1)/2$. Doing without a change of measure comes at the cost of m additional parameters. This is not a significant penalty in practice since dimension reduction methods seek a low-dimension

Algorithm 2: Proposed Fully Bayesian Model

$\theta_{p,i} \sim \mathcal{N}(0,1) \quad \forall i = 1, \ldots, k$
$\mathbf{W} = \mathcal{H}(\theta_{\mathbf{p}}) \quad (\text{algorithm } 1)$
$\mathbf{Z} = \mathbf{X}\mathbf{W}$
$\log \theta_j \sim \mathcal{N}(0,1) \quad \forall j = 1, \ldots, m+2$
$\mathbf{y} \sim \mathcal{N}(0, K(\mathbf{Z}, \mathbf{Z}; \theta) + \sigma_n^2 \mathbf{I})$

space such that $m \ll d$.

Prior distributions for the remaining model parameters must also be specified: a log-normal distribution is used as prior for the GP hyperparameters. This results the proposed generative model detailed in algorithm 2.

The training of the proposed model consists in inferring the joint posterior distribution $p(\theta, \theta_{\mathbf{p}}|\mathbf{X}, \mathbf{y})$ of the model parameters by conditioning the generative model shown in algorithm 2 with respect to the training data $\mathcal{D} = (\mathbf{X}, \mathbf{y})$ and performing probabilistic inference using MCMC.

Once posterior distributions are obtained, the model can be used to make predictions. In contrast to traditional GPR (equation (2.31)), fully Bayesian predictions in the proposed approach require marginalizing over both the GP hyperparameters and the projection parameters:

$$p(\mathbf{y}_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y}) = \iint p(\mathbf{y}_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y}, \theta, \theta_{\mathbf{p}}) p(\theta, \theta_{\mathbf{p}}|\mathbf{X}, \mathbf{y}) \, d\theta \, d\theta_{\mathbf{p}} \tag{2.33}$$

where equation (2.28) is used to compute the term $p(\mathbf{y}_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y}, \theta, \theta_{\mathbf{p}})$ after projecting the input design matrix $\mathbf{X}$ onto the feature space.

In addition to the prediction of the response $\mathbf{y}_*$ at unobserved locations $\mathbf{X}_*$, the posterior distribution $p(\mathbf{W}|\mathbf{X}, \mathbf{y})$ of the projection matrix $\mathbf{W}$ can be readily obtained by application of the Householder parametrization detailed in algorithm 1 on the MCMC samples of the marginal distribution $p(\theta_{\mathbf{p}}|\mathbf{X}, \mathbf{y})$. Instead of a point-based estimation, the proposed method therefore grants access to a full posterior probability distribution of the projection matrix $\mathbf{W}$.

We will refer from now on refer to the approach presented in this section as Bayesian ridge Gaussian process (B-R-GP). We are now in a position to formulate hypothesis 1:

Hypothesis 1.1

When inputs are high-dimensional and only few analysis observations are affordable, the proposed fully Bayesian approach to training a surrogate model based on approximation by ridge function leads to higher predictive accuracy than 1) MLE-based training because it allows a better quantification of epistemic uncertainty due to limited training data, and 2) a fully Bayesian method trained using the original methods because it reduces the dimension of the inputs.

In the next section, we will discuss the experiment designed to test this hypothesis.

2.2.3 Setup of Experiment 1.1

Need for Experiment 1.1

From the literature, it is established that a Bayesian approach generally leads to better models when training data is limited thanks to the better quantification of epistemic uncertainty. This comes from the fact that better quantifying epistemic uncertainty in turn allows to avoid over-fitting the model parameters to the specific training data, as we illustrated it using the Bayesian linear regression example in the previous section. However, this effect has not been shown on the practical Bayesian approach to approximation by ridge functions that we proposed: this is the point of hypothesis 1.1 that we aim to support through experiment 1.1.

The two parts of hypothesis 1.1 call for two comparisons. First, to support the first part of the hypothesis, we need to compare the predictive accuracy of surrogate models obtained with the proposed approach to surrogate models obtained with an MLE-based approach and check that both predictive accuracy and the quantification of epistemic uncertainty are improved: while improving predictive accuracy is the end goal, a better quantification of

epistemic uncertainty is the enabling mechanism. Second, to support the second part of the hypothesis and check that the Bayesian approach alone is not sufficient to handle situations where inputs are high-dimensional when only few training samples are available, we need to compare the predictive accuracy obtained with the proposed approach to the predictive accuracy obtained when using a Bayesian approach, that does not include a mechanism for supervised dimension reduction. These two comparisons should confirm that it is indeed the combination of 1) the approximation by ridge functions, and 2) the fully Bayesian approach, that allows to reach a better predictive accuracy.

Experiment Design

The discussion held above calls for two benchmark methods: one that includes supervised dimension reduction but not the Bayesian approach, and another that follows a Bayesian approach without a mechanism for supervised dimension reduction. We use the recently developed manifold optimization-based active subspace (MO-AS) method [112] for the first because it only differs from the proposed fully Bayesian approach by the training mechanism: instead of using MCMC, it uses an optimization-based approach. Otherwise, the rest of the formulation is identical. For the second, we use a fully Bayesian Gaussian process, which corresponds to removing the projection of the inputs from the method that we propose. Both methods are discussed in greater details below, in the section *Benchmark Methods*.

Since the proposed approach is aimed at surrogate modeling, the comparisons discussed previously bear on the ability of the proposed method to accurately replicate the results obtained using computer analyses. Therefore, an analysis, or a set of analyses, is needed to make the comparisons possible. In the context of this thesis, we are developing a general-purpose method rather than focusing on a specific target application. Therefore, we work on a set of analytical functions and analyses that are representative of the type of analyses carried out in the context of engineering design. We introduce the datasets used in the

context of this experiment in the section *Test Datasets*.

As explained above, there are two aspects to the comparison that we need to make: predictive accuracy, and quantification of epistemic uncertainty. Each aspect calls for a different metric. For predictive accuracy, we follow standard practice and use the out-of-sample coefficient of determination (R^2). In this context "out-of-sample" is meant in the sense that it is computed using analysis observations that were not used for training the model. To assess the quality of the quantification of epistemic uncertainty, we use mean log pointwise predictive density (MLPPD), which quantifies the probability of observing the actual value of the analysis under the posterior predictive distribution provided by the surrogate model. The definition of each metric is detailed thereafter in the section *Benchmark Metrics*. While they are not strictly needed to support the hypothesis, we also keep track of two other metrics: 1) the training duration, as we expect the Bayesian approach to require a computational cost higher than an MLE-based approach, and 2) when possible, a measure of the similarity between the FS obtained using the proposed method and the gradient-based AS method, to provide additional insights about the proposed approach.

Given a dataset, the experimental process then consists in 1) selecting training observations within the dataset observations, 2) use this training data to train surrogate models using the proposed method and the two benchmark methods, and 3) compute the benchmark metrics for the trained model. The comparison needed to support hypothesis 1.1 can then be drawn based on the metrics. The way we present and interpret numerical results is discussed in the section *Interpretation of the Results* below.

Benchmark Methods

The two state-of-the-art methods used as benchmarks are presented thereafter. We recall that the proposed fully Bayesian method discussed in 2.2.2 is referred to as Bayesian ridge Gaussian process (B-R-GP).

MO-AS The first benchmark method was proposed in [115] and is based on the *Gaussian processes with built-in dimensionality reduction* method proposed in [91] and discussed in section 2.1.3. The benchmark method from [115] modified two aspects of the method in [91]: a) a state-of-the-art manifold optimization library [117] was employed, and b) optimization is performed in the Grassmann manifold instead of the Stiefel manifold. Here, we only retain the first modification (the state-of-the-art manifold optimization algorithm) but use the Stiefel manifold instead of the Grassmann manifold for the same reasons that we use the Stiefel manifold in our proposed fully Bayesian approach (see discussion in section 2.2.2): the directional information retained when working in the Stiefel manifold, which matters for the GP's ARD kernel, is lost when working in the Grassmann manifold. This benchmark method is referred to as MO-AS in the rest of this section. In the same spirit as our proposed method, it aims at simultaneously identifying an orthogonal projection onto a low-dimensional input subspace and the mapping from this subspace to the output space. However, it does not include any mechanism for quantifying uncertainty in the identified subspace.

B-GP The second benchmark method is based on the method proposed in [134] and we refer to it as Bayesian Gaussian process (B-GP) in this section. In this method, a GPR model is first built on the original, full-dimensional input space. Instead of gathering gradient samples and using a Monte-Carlo (MC) approximation as described in section section 2.1.3, the matrix $\mathbf{C}$ can be analytically derived based on the assumptions of the GPR model. We recall from section 2.1.3 that the AS can be obtained by performing an singular value decomposition (SVD) of this matrix. If Bayesian inference is used to train the GPR model and posterior distributions for its hyperparameters are obtained, uncertainty can then be propagated using MC to obtain a distribution on the AS.

The computation of the matrix $\mathbf{C}$ implemented in the context of this study is only semi-analytical. GP gradient evaluations are made analytically once the posterior distribution of

the GP hyperparameters has been inferred, but an MC estimator is used to approximate $\mathbf{C}$ with 1,000 gradient samples instead of the fully analytical scheme from [134]. This was done because an exact reproduction of the process in [134] brought excessive complexity and long runtimes. Given the high number of gradient samples used for the MC approximation, this modification is not expected to alter results. Improving on the methodology proposed in the original paper, we carry out exact inference of the GP hyperparameters distribution using MCMC instead of approximate inference.

Implementation Details The B-R-GP and B-GP methods were implemented using the probabilistic programming language *numpyro* [133, 135], which uses *JAX* [136] as computational backend. The MO-AS method was implemented using a version of the *Pymanopt* framework [117] modified to use *JAX*. Using the same computational backend across all three methods helps in ensuring a level playing field for the comparative study such that differences in training time can be linked to the methods themselves instead of implementation specifics. For the Bayesian methods, MCMC is used for inference, leveraging the no-U-turn sampler (NUTS) as implemented in *numpyro*. Four parallel chains are sampled, each consisting of 1,000 samples from the joint posterior probability distribution on model parameters and initialized using 500 warmup samples. For MO-AS, 500 restarts of the manifold optimization algorithm are used, as in [115]. The implementation of the three models used to produce the results presented in this paper are available online[1].

Benchmark Datasets

The comparison of the proposed method's performance with benchmark method is drawn based on datasets generated using analytical functions and on datasets originating from science and engineering.

[1] https://gitlab.com/raphaelgautier/thesis_experiments_part1/

Analytical Functions Analytical functions are chosen to be quadratic functions featuring a dependence on a low-dimensional input subspace by construction. As before, d and m are respectively the dimensions of the input space and feature space. The quadratic functions f are defined as:

$$\mathbf{z} = \mathbf{W}^T\mathbf{x} \tag{2.34}$$

$$f(\mathbf{x}) = \mathbf{z}^T\mathbf{A}\mathbf{z} + \mathbf{b}\mathbf{z} + c + \epsilon \tag{2.35}$$

where $\mathbf{x}$ and $\mathbf{z}$ are respectively the input vector and the vector of projected coordinates in the FS. $\mathbf{W}$ is the $d \times m$ projection matrix onto the FS. $\mathbf{A} \in \mathbb{R}^{m\times m}$, $\mathbf{b} \in \mathbb{R}^m$, and $c \in \mathbb{R}$ are parameters of the quadratic mappings. Given d and m, their coefficients are sampled from standard normal distributions to obtain randomly generated mappings. Since the domains of the quadratic mappings are restricted to their respective FS by construction, an additive centered Gaussian noise ϵ of standard deviation 5×10^{-2} is added to simulate the variation of the response due to variations of the inputs in the inactive subspace. In the context of the present study, eight quadratic functions were generated, that combine an input space dimension d of 10, 25, 50, or 100 with an FS dimension of 2 or 5. For these datasets, we fix the FS dimension to their actual, known, value.

Four datasets originating from scientific and engineering applications are also used to assess the proposed method. These datasets were previously used in studies related to the AS method [110, 137, 138, 104] and include gradient evaluations in addition to input-output pairs. Gradient evaluations are used to estimate the actual AS using the original AS method, which is needed to assess the ability of the proposed method to uncover the AS. Those datasets cover a wide range of input space dimensions: the NACA 0012 [110], HIV [137], ONERA-M6 [138], and Elliptic PDE [104] datasets respectively have 18, 27, 50, and 100 input dimensions. At the moment, we fix the FS dimension at 3 for these datasets, and the selection of the FS dimension will be the subject of the next two sections of this chapter.

NACA0012 The NACA0012 dataset was introduced in [110] to illustrate the AS method. In the following, we summarize the main features of this dataset. The data originates from the computational fluid dynamics (CFD) simulation of a NACA0012 airfoil using the SU2 solver. The simulation computes the 2D velocity and pressure fields surrounding the airfoil in transonic compressible flow conditions, an illustration of the resulting flow field is shown in figure 5.10 of reference [110]. The inputs that are varied to generate the different observations in the dataset correspond to alterations to the original airfoil geometry, in the form of 18 parameters that set the height of Hicks-Henne bump functions. The height parameters are varied uniformly within the hypercube $[-0.01, 0.01]^{18}$, which corresponds to valid geometries (e.g., the surface boundaries remain separated). These inputs have then been normalized to the hypercube $[-1, 1]^{18}$. The dataset outputs are the airfoil lift and drag coefficients, that are computed based on the CFD flow solutions. In the context of aerospace design, such an analysis could be used to perform aerodynamic shape optimization or in the context of uncertainty quantification, for example to assess the impact of manufacturing defects.

ONERA M6 The ONERA M6 dataset was introduced in [138], also in the context of the AS method. The data also originates from a CFD simulation carried out using SU2's Euler solver, but this time for the full ONERA M6 wing instead of an airfoil. Accordingly, the simulation computes the 3D flow field, for which an illustration is available in figure 5.10 of reference [110].. The simulation assumed a transonic Mach number $M = 0.8395$ and the angle of attack was set at $\alpha = 3.06°$. The inputs varied to generate the dataset also correspond to geometric alterations, this time relying on free-form deformation (FFD). An FFD box encloses the original wing surface geometry and 50 control points are defined and used to deform the the box, in turn smoothly deforming the wing geometry. This approach allows to modify the main geometrical features of the wing, such as thickness, sweep, or twist, in a flexible and controlled manner that ensures the smoothness of the

resulting surface. The FFD parameters are varied within the hypercube $[-0.05, 0.05]^{50}$, then normalized to the hypercube $[-1, 1]^{50}$. The dataset outputs are the lift and drag of the wing, which are computed based on the CFD flow solutions. In the context of aerospace design, such an analysis could be used in the context of aerodynamic shape optimization.

HIV The HIV dataset was introduced in [137], also in the context of the AS method. This dataset originates from a long-term model of the HIV disease dynamics within a host, in which the evolution of the count of various types of cells is predicted on a scale of several years, as a function of 27 parameters whose value depend on each individual patient. The model is composed of a system of seven ordinary differential equations [137]. The dataset inputs are the 27 model parameters, and the scalar output chosen here is the count of $CD4^+$ T-cells at $t = 3400$ days. The motivation that was put forward for creating a surrogate model of such an analysis was to speed up the calibration of the model to a new patient. While the dataset does not relate to engineering design, its target application – model calibration – is commonly carried out when, for example, physics-based models require the tuning of some of their parameters to match the results of experimental simulations.

Elliptic PDE The elliptic PDE example was introduced in the context of the Active Subspace method by [104], from which the formulation is reproduced.

We consider a 2D square-shaped spatial domain whose coordinates are denoted as $\mathbf{s}$. Accordingly, we set $\mathbf{s} \in [0, 1]^2$. We introduce the m-dimensional vector $\mathbf{x} \in [-1, 1]^m$ of input parameters, whose impact is detailed below. We are solving for the quantity $u = u(\mathbf{s}, \mathbf{x})$ that satisfies:

$$-\nabla_{\mathbf{s}} \cdot (a(\mathbf{s}, \mathbf{x}) \nabla_{\mathbf{s}} u(\mathbf{s}, \mathbf{x})) = 1 \tag{2.36}$$

The four boundaries are defined as:

$$\begin{aligned}
\Gamma_1 &= \{\mathbf{s} \in [0,1]^2 \mid s_2 = 0\} \quad \text{(bottom)} \\
\Gamma_2 &= \{\mathbf{s} \in [0,1]^2 \mid s_1 = 0\} \quad \text{(left)} \\
\Gamma_3 &= \{\mathbf{s} \in [0,1]^2 \mid s_2 = 1\} \quad \text{(top)} \\
\Gamma_4 &= \{\mathbf{s} \in [0,1]^2 \mid s_1 = 1\} \quad \text{(right)}
\end{aligned} \tag{2.37}$$

The boundary conditions are $u = 0$ if $\mathbf{s} \in \Gamma_1, \Gamma_2, \Gamma_3$ and $\mathbf{n} \cdot (a \nabla_{\mathbf{s}} u) = 0$ if $\mathbf{s} \in \Gamma_4$. For a given position $\mathbf{s}$ and input value $\mathbf{x} = (x_1, \ldots, x_m)$, the coefficient $a(\mathbf{s}, \mathbf{x})$ is defined as:

$$\log\left(a(\mathbf{s}, \mathbf{x})\right) = \sum_{i=1}^{m} \sqrt{\sigma_i} \phi_i(\mathbf{s}) x_i \tag{2.38}$$

The scalar σ_i and spatial field ϕ_i are respectively the eigenvalues and eigenvectors of the following spatial correlation function:

$$C(\mathbf{s_1}, \mathbf{s_2}) = \exp\left(-\frac{\|\mathbf{s_1} - \mathbf{s_2}\|_1}{\beta}\right) \tag{2.39}$$

The parameter β accounts for the length scale of the input fields: the smaller the β, the greater the variations of the corresponding fields $(\phi_i)_{i=1,\ldots,m}$. Finally, for a given $\mathbf{x}$, the output of interest $f(\mathbf{x})$ is the integral of the quantity $u(\mathbf{s}, \mathbf{x})$ over the right boundary Γ_4:

$$f(\mathbf{x}) = \int_{\Gamma_4} u(\mathbf{s}, \mathbf{x}) \, \mathrm{d}\mathbf{s} \tag{2.40}$$

The dataset for this analysis is generated using the original implementation provided by [104], in which the problem is solved on a 100×100 grid.

While remaining simple and fast to evaluate, this example goes beyond simple analytical functions. In addition, it can be modified to change its 1) input dimension, 2) "complexity", and 3) level of fidelity. Varying the input dimension m simply amounts to retaining the first m eigenpairs $(\sigma_i, \phi_i)_{i=1,\ldots,m}$ of the correlation function given in eq. (2.39). By "complexity", we mean the complexity of creating a surrogate model for the analysis, which can be tuned by the scalar parameter β. As illustrated in [139] using the directions of the AS, for a "large" value for β, such as $\beta = 1.0$, the output quantity $f(\mathbf{x})$ effectively depends

on only few of the input directions x_i. When β is set to a lower value, such as $\beta = 0.01$, then all input dimensions play a role in the variation of $f(\mathbf{x})$. Finally, changing the level of fidelity of the analysis can be simulated by varying the size of the grid on which the problem is discretized and numerically solved. While this is not of particular interest for the present experiment, this capability will be leveraged in chapters 3 and 4.

Summary Table 2.1 summarizes the main features of the datasets used in the context of this experiment and presented in this section. "QF" stands for "quadratic function". "Observations in Dataset" indicates the total number of observations of the underlying analysis contained in the dataset. We only use a fraction of these observations for training since we are focusing on the low-data regime. The rest of the dataset is used to compute the validations metrics discussed previously. "Input Space Dimension" indicates the original dimension of the inputs, whose nature has been detailed above. "Feature Space Dimension" indicates the number at which we set the FS dimension. Analytical quadratic functions by construction have an intrinsic FS dimension, and we use an FS dimension of 3 for the science and engineering datasets, which is expected to be sufficient based on previous AS studies conducted on these datasets. The last column, "Reference", references the publication that introduced the dataset, when applicable.

Benchmark Metrics

The comparative study focuses on the four following aspects: a) deterministic predictive capability, b) probabilistic predictive capability, c) computational cost, and d) similarity of the uncovered subspace with the AS. To each aspect corresponds a numerical metric enabling the quantitative comparison of the proposed method with both benchmark methods: a) coefficient of determination, b) mean log pointwise predictive density, and c) training duration, and d) subspace angles. These metrics are briefly defined in the following paragraphs.

Table 2.1: Summary of benchmark datasets

		Observations in Dataset	**Input Space Dimension**	**Feature Space Dimension**	**Reference**
Analytical Quadratic Functions (QF)	QF 10/2	1000	10	2	–
	QF 10/5	1000	10	5	–
	QF 25/2	1000	25	2	–
	QF 25/5	1000	25	5	–
	QF 50/2	1000	50	2	–
	QF 50/5	1000	50	5	–
	QF 100/2	1000	100	2	–
	QF 100/5	1000	100	5	–
Science and Engineering	NACA0012 (lift)	1756	18	3	[110]
	HIV at $t = 3400$	1000	27	3	[137]
	ONERA M6 (lift)	297	50	3	[138]
	Elliptic PDE	1000	100	3	[104]

Coefficient of Determination The deterministic predictive capability refers to the quality of point predictions made by the model and is quantified using the R^2 metric defined below.

Definition 3 (coefficient of determination (R^2))**.** Let n test points $\{\mathbf{x_1^*}, ..., \mathbf{x_n^*}\}$ distributed according to the distribution $p(\mathbf{x})$, f the underlying function of interest, and $\hat{f}$ its approximation, the coefficient of determination (R^2) is:

$$1 - \frac{\sum_{i=1}^{n} \left(f(\mathbf{x_i^*}) - \hat{f}(\mathbf{x_i^*}) \right)^2}{\sum_{i=1}^{n} \left(f(\mathbf{x_i^*}) - \bar{f} \right)^2} \tag{2.41}$$

with $\bar{f} = \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{x_i^*})$.

For fully Bayesian methods, the point estimate $\hat{f}(\mathbf{x}^*)$ at a new location $\mathbf{x}^*$ is chosen as the median of the posterior predictive distribution $p(y_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y})$ introduced in equation (2.33). The median is approximated as follows. Samples from the joint distribution $p(y_*, \theta, \theta_\mathbf{p}|\mathbf{x}_*, \mathbf{X}, \mathbf{y})$ are drawn using the fact that:

$$p(y_*, \theta, \theta_\mathbf{p}|\mathbf{x}_*, \mathbf{X}, \mathbf{y}) = p(y_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y}, \theta, \theta_\mathbf{p}) p(\theta, \theta_\mathbf{p}|\mathbf{X}, \mathbf{y}) \tag{2.42}$$

Samples from $p(\theta, \theta_\mathbf{p}|\mathbf{X}, \mathbf{y})$ are byproducts of the MCMC process and $p(y_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y}, \theta, \theta_\mathbf{p})$ is a known Gaussian distribution. For every draw from $p(\theta, \theta_\mathbf{p}|\mathbf{X}, \mathbf{y})$, multiple draws from $p(y_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y}, \theta, \theta_\mathbf{p})$ are performed. This results into a collection of draws from the joint distribution $p(y_*, \theta, \theta_\mathbf{p}|\mathbf{x}_*, \mathbf{X}, \mathbf{y})$. Information regarding θ and $\theta_\mathbf{p}$ is dropped from these samples, effectively marginalizing out those parameters, and yielding draws from the desired posterior predictive distribution $p(y_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y})$. The median of $p(y_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y})$ is approximated by the sample median of these draws.

Mean Log Pointwise Predictive Density The probabilistic predictive capability refers to the quality of the probability distributions outputted by the model. Because the computer models under consideration are deterministic mappings, the likelihood of observing the actual value output under the posterior predictive distribution can be used as a metric. The log pointwise predictive density metric [119] can be used for this purpose. To account for

different numbers of validation samples across cases, the metric is normalized by taking the mean over the observations used for its computation. We refer to the resulting metric as mean log pointwise predictive density (MLPPD), it is defined below.

Definition 4 (mean log pointwise predictive density (MLPPD))**.** Let n test points $\{\mathbf{x_1^*}, ..., \mathbf{x_n^*}\}$ be distributed according to the distribution $p(\mathbf{x})$, the mean log pointwise predictive density (MLPPD) is:

$$\frac{1}{n}\sum_{i=1}^{n} \log p(y_i^*|\mathbf{x_i^*}, \mathbf{X}, \mathbf{y}) \tag{2.43}$$

The expression for the posterior predictive distribution was recalled in equation (2.28). With μ_i^* and σ_i^{*2} respectively computed using equations (2.29) and (2.30), we obtain the following expression:

$$\log p(y_i^*|\mathbf{x_i^*}, \mathbf{X}, \mathbf{y}) = -\frac{1}{2}\frac{(y_i^* - \mu_i^*)^2}{\sigma_i^{*2}} - \log \sigma_i^* - \frac{\gamma}{2}\log 2\pi \tag{2.44}$$

where $\gamma = m$ for the MO-AS and B-R-GP methods and $\gamma = d$ for the B-R-GP method.

For fully Bayesian methods, marginalization with respect to the hyperparameters is necessary:

$$\log p(y_i^*|\mathbf{x_i^*}, \mathbf{X}, \mathbf{y}) = \log\left(\iint p(y_i^*|\mathbf{x_i^*}, \mathbf{X}, \mathbf{y}, \theta, \theta_\mathbf{p}) p(\theta, \theta_\mathbf{p}|\mathbf{X}, \mathbf{y})\, d\theta_\mathbf{p}\, d\theta\right) \tag{2.45}$$

Computation of the double integral in equation (2.45) relies on a Monte-Carlo approximation that uses the samples from the joint posterior distribution $p(\theta, \theta_\mathbf{p}|\mathbf{X}, \mathbf{y})$ obtained through MCMC.

Training Duration

Definition 5 (Training duration (TD))**.** training duration is measured as the wall-clock time elapsed between the start and the end of the model training.

Mean First Subspace Angle R^2 and MLPPD allow to assess the performance of the proposed approach with respect to its end application, namely creating a surrogate model

that is an accurate image of the original mapping of interest and that provides accurate quantification of epistemic uncertainty due to limited data. However, they do not provide any insight regarding the inner workings of the proposed approach. In particular, the feature space on which original inputs are projected to serve as low-dimensional inputs to a GP is hidden by those metrics. A direct assessment of the subspace is not straightforward, as methods seeking a ridge approximation of the form $f(\mathbf{x}) = g(\mathbf{W}^T\mathbf{x})$ may yield different projection matrices $\mathbf{W}$ and subspaces [101]. Despite those facts, the comparison of the uncovered feature space with the subspace yielded by benchmark methods, and in particular the AS method, has proven to be insightful. In many instances, we show that the proposed approach eventually recovers the AS once a sufficient training number of training data is used. In those instances, the comparison of the uncovered feature space with the AS enables to link the poor predictive performance with smaller training sets to the inability of the method to uncover an adequate subspace.

Directly comparing projection matrices $\mathbf{W}$ does not allow to draw conclusions regarding the subspaces they span, as infinitely many orthogonal bases may be obtained through rotations within the subspace. Instead, principal angles between subspaces, or simply subspace angles, provide a similarity measure between subspaces that does not depend on the particular choice of bases for these subspaces. The definition of subspace angles from [140] is reproduced in definition 6 below.

Definition 6 (Definition 2.1. in [140])**.** Let $\mathcal{X} \subset \mathbb{C}^n$ and $\mathcal{Y} \subset \mathbb{C}^n$ be subspaces with $\dim(\mathcal{X}) = p$ and $\dim(\mathcal{Y}) = q$. Let $m = \min(p, q)$. The principal angles

$$\Theta(\mathcal{X}, \mathcal{Y}) = [\theta_1, \ldots, \theta_m], \text{where } \theta_k \in [0, \pi/2], k = 1, \ldots, m,$$

between $\mathcal{X}$ and $\mathcal{Y}$ are recursively defined by

$$s_k = \cos(\theta_k) = \max_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} \left| x^H y \right| = \left| x_k^H y_k \right|,$$

subject to

$$\|x\| = \|y\| = 1, x^H x_i = 0, y^H y_i = 0, i = 1, \ldots, k-1.$$

The vectors $\{x_1, \ldots, x_m\}$ and $\{y_1, \ldots, y_m\}$ are called the principal vectors.

By this definition, the first subspace angle is greater than the following angles. It is therefore an upper bound for all subspace angles. For this reason and because using a single numerical value eases comparison between methods, the first subspace angle will be used as metric.

The reference AS is computed using the original AS method [110] presented in section 2.1.3. This is possible because all test datasets include gradient evaluations. For each dataset, all available gradient samples are used to estimate the reference AS in an effort to maximize the quality of the MC estimator $\sum_{i=0}^{n} \nabla f(\mathbf{x_i}) \nabla f(\mathbf{x_i})^T$.

The probability distribution for the projection matrix $\mathbf{W}$ that we obtain using Bayesian inference leads to a distribution on the subspace angles between the recovered feature space and the AS computed using gradients. To ease the presentation of the results, we use its mean as metric instead of the full distribution and we refer to it as the mean first subspace angle (MFSA).

Sensitivity Study

We expect the number of training observations, as well as the particular choice of training observations, to have an impact on the metrics tracked in this experiment, and accordingly to possibly have an impact on the outcome of the comparison between the three surrogate modeling methods.

In general, we expect predictive accuracy to increase with the number of training observations. For training sets on the larger size, we do not necessarily expect to observe a large difference between the MLE-based approach and the proposed approach, as the Bayesian training is mostly expected to bring a benefit when training data is sparse. Therefore, we will be conducting a sensitivity study in which we vary the number of observations selected in each dataset to train the different surrogate models. When it comes to GPR, a rule of thumb is to use a number of training observations equal or greater than five times the num-

ber of inputs. Since we focus on small training sets, we will be focusing on training sets whose size ranges from one to five times the number of inputs, by increment of one time the number of inputs. For instance, for an analysis with 50 inputs, we would consider training sets respectively containing 50, 100, 150, 200, and 250 observations.

In addition to the number of training samples, we expect the actual selection of those training samples to have an impact on predictive accuracy. The outcome of the training phase directly depends on the training observations. As an example, we would generally expect better predictive accuracy from a training set that fills the input space rather than a training set whose points are concentrated in a limited region of the input space. In the latter case, there is not much we can learn about the behavior of the underlying analysis outside of the covered region. Therefore, we will be carrying out five repetitions for every number of training samples, by selecting the observations randomly within the datasets presented above.

Finally, the number of input dimensions of the analysis for which we are creating a surrogate model is also expected to have an impact, as we expect the effect of the curse of dimensionality to increase with higher-dimensional inputs. The datasets used in this experiment have been either designed (for the quadratic functions) or chosen (for the science and engineering datasets) to span a variety of input dimensions: quadratic functions have between 10 and 100 input dimensions, while science and engineering datasets have input dimensions ranging from 18 to 100.

Table 2.2 summarizes the parameters being varied in the context of experiment 1.1.

Presentation and Interpretation of the Results

In section 2.2.4, we will first present an in-depth walk-through of multiple aspects of the results, beyond those discussed previously, on a single dataset (ONERA-M6). This should allow to develop a first understanding of how the proposed method operates.

Then, we will move on to the results of the parametric study that was constructed

Table 2.2: Experiment 1.1 – Summary of the parameters varied in the parametric study

Parameter	**Parameter Values**
Datasets	see table 2.1
Number of Training Samples (times number of inputs)	[1, 2, 3, 4, 5]
Number of Training Repetitions	5 repetitions with fixed random seeds
Surrogate Modeling Method	[MO-AS, B-GP, B-R-GP]

throughout this section. In this paragraph, we discuss the graphical representations that we will be using to interpret the results. For each dataset considered, we will visualize the evolution of the metrics of interest (R^2, MLPPD, training time, and MFSA) as a function of the number of training observations for all three considered surrogate modeling methods. Because training is repeated for every considered number of training samples, we obtain multiple values for each metric. The distribution of these values is represented using a box plot, also called box-and-whisker plot. The box represents the quartiles of the distribution while the "whiskers" cover the rest of the distribution. Values that are deemed "outliers" based on the inter-quartile distance, are represented as diamonds. For each number of training samples, there will be three distributions of different colors, that correspond in order to 1) the proposed method (B-R-GP), 2) B-GP, and 3) MO-AS.

For results to support the hypothesis, the proposed method should – consistently across training repetitions – exhibit higher deterministic and predictive predictive accuracies, as measured by R^2 and MLPPD respectively. As the number of training observations grows, we expect these metrics to converge for the B-R-GP and MO-AS methods, as discussed above. In the lower-dimensional cases, such as a 10D input space, we expect B-GP to display competitive predictive accuracy, as these are conditions in which a Bayesian GP should perform well.

We expect the training time of the proposed approach to generally be higher than the benchmark methods as a result of the added computational complexity introduced by the

Bayesian approach and the projection of the inputs, thus introducing a trade-off between accuracy and cost. Finally, when we look at the MFSA metric, we are looking to see whether it converges towards 0, which would indicate that the FS determined using the proposed method matches with the low-dimensional subspace found using the gradient-based AS method. These last two metrics, training time and MFSA, are used in a more exploratory context, in order to better understand the proposed method, but they are not strictly necessary to support hypothesis 1.1.

2.2.4 Preliminary Results: In-Depth Walk-Through

This section presents a detailed walk-through of the application of the proposed method on an engineering use case. After a brief presentation of the problem, we examine the training phase of the model, and then present an assessment of the surrogate's predictive performance. We focus on a single of the benchmark datasets presented in the previous section, namely the ONERA-M6 dataset from [138]. In this problem, the authors sought to predict the impact of shape deformations encoded by 50 FFD control points on the lift produced by an ONERA-M6 wing.

The quality of the model parameters' inference and the final model's predictive performance are affected by multiple factors, including the dimension of the FS and the number of observations on which the model is conditioned. These parameters have been varied in our experiments to understand their impact. The dimension of the FS was varied from 1 to 5 and the the number of training samples has been varied from 1 to 5 times the number of input dimensions. For conciseness, we will only present results for notional "low" and "high" data regimes, respectively corresponding to 100 and 250 training samples for the current 50-dimensional problem.

The results presented for fixed numbers of FS dimension and number of training samples have been produced with 15,000 draws-long MCMC chains preceded by after a 5,000 draws-long warmup phase. These correspond to the results presented in figs. 2.10, 2.12,

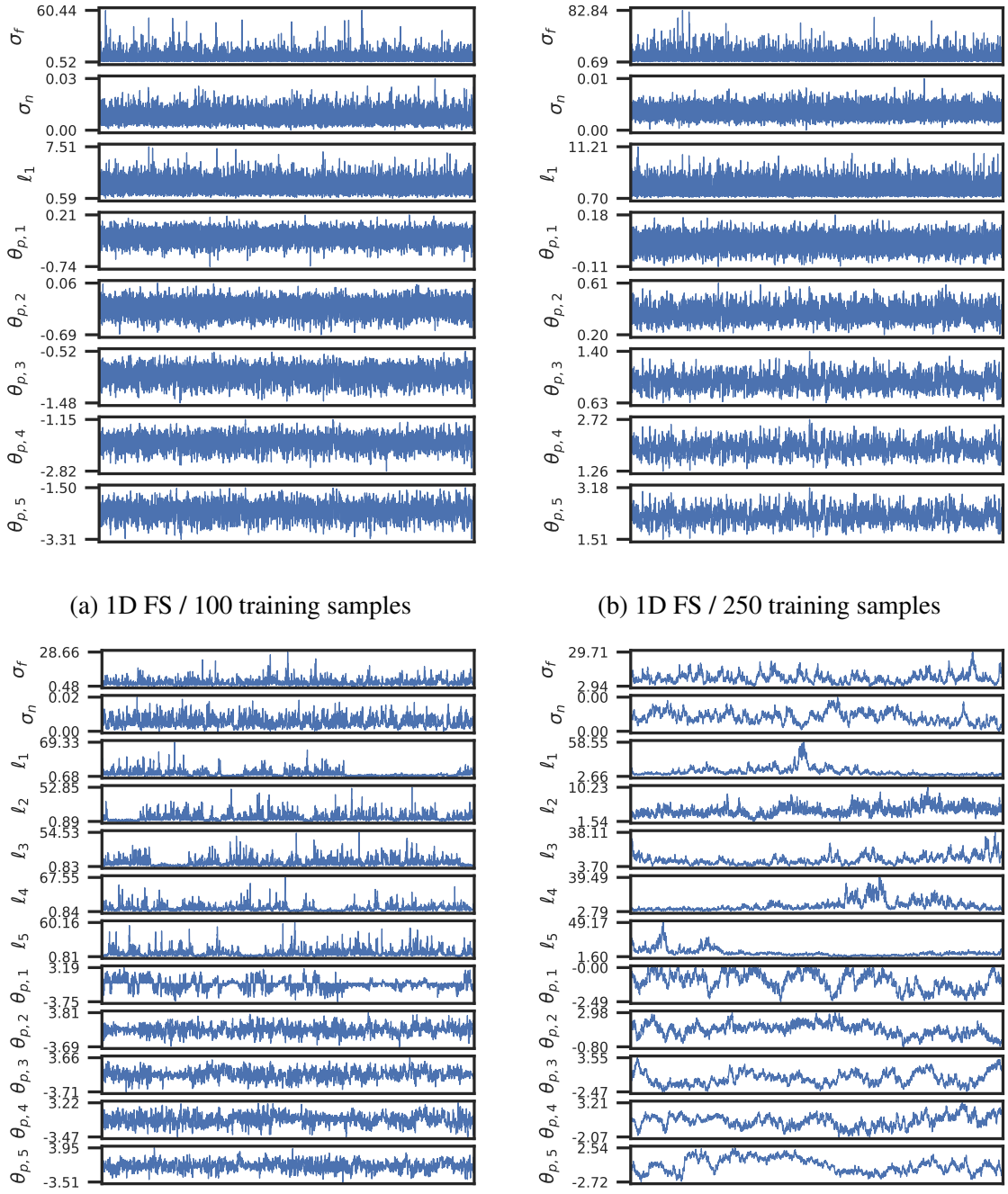
2.13 and 2.15. Due to the high computational cost incurred during model inference, shorter 1,000 draws-long chains preceded by a 500 draws-long warmup phase are used for parametric studies, such as those shown in figs. 2.11, 2.14, 2.16 and 2.17.

Statistics Pertaining to Model Training

In this section, we are assessing the validity of the training approach detailed in section section 2.2.2, that consists in inferring posterior distributions for all model parameters (parameters of the projection parameters as well as GP hyperparameters) using MCMC. The quality of the MCMC inference process can be assessed by observing the posterior chains (fig. 2.10) as well as MCMC-specific statistics, such as the split Gelman-Rubin statistic (fig. 2.11) [141].

The MCMC chains are displayed in fig. 2.10 for two different FS dimensions (1 and 5) and two different number of training samples (100 and 250). Only the first five projection parameters $\theta_{p,i}$ are shown, as it would be impractical to show all 50 (1D case) or 240 (5D case) of them. We observe that increasing the FS dimension and the number of training samples both lead to poor mixing of the chains. While adequate mixing occurs in the 1D/100 samples cases for all parameters, increasing the number of training samples leads to poor mixing of the projection parameters, while the quality of the GP hyperparameters chains remains satisfactory. When increasing the number of FS dimensions, the quality of all chains deteriorates. An increase of the FS dimension leads to an increase of the number of projection parameters, which may explain the poor mixing observed when the FS dimension increases. An increase of the number of observations leads to a more sharply peaked posterior distribution, which is accordingly more challenging to sample, and may explain the poor mixing observed as the number of training samples is increased.

The evolution of the split Gelman-Rubin statistic shown in fig. 2.11 confirms previous observations made on the MCMC chains. Both the number of FS dimensions and the number of observations lead to an increase in the value of the statistic, indicating a poor



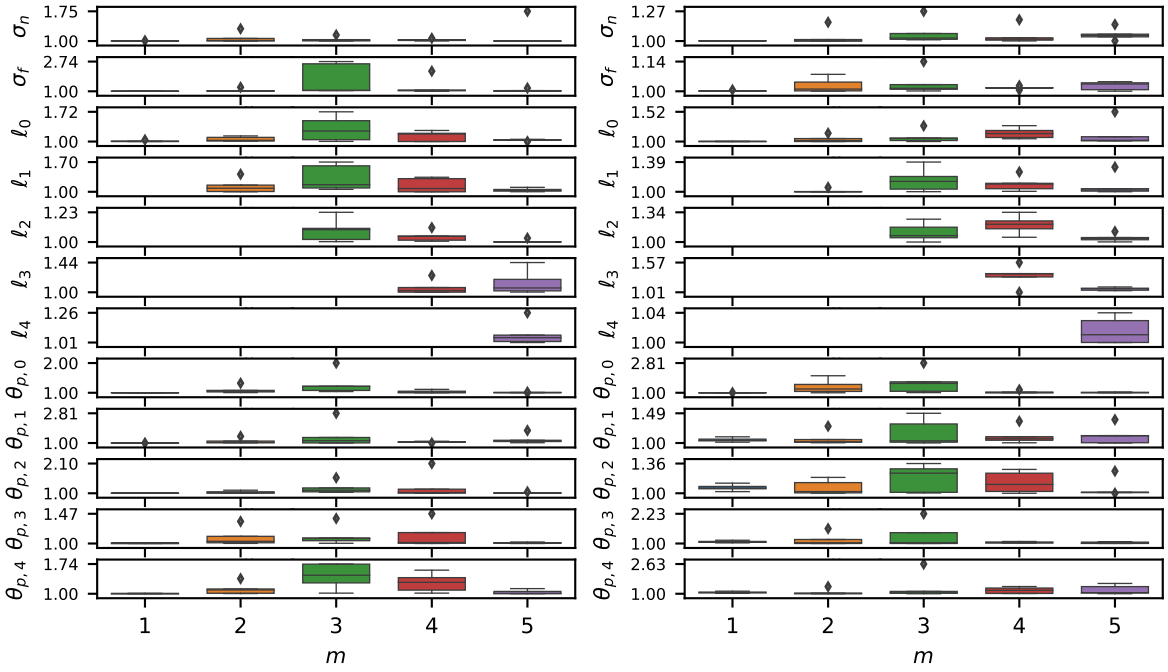
(a) 1D FS / 100 training samples

(b) 1D FS / 250 training samples

(c) 5D FS / 100 training samples

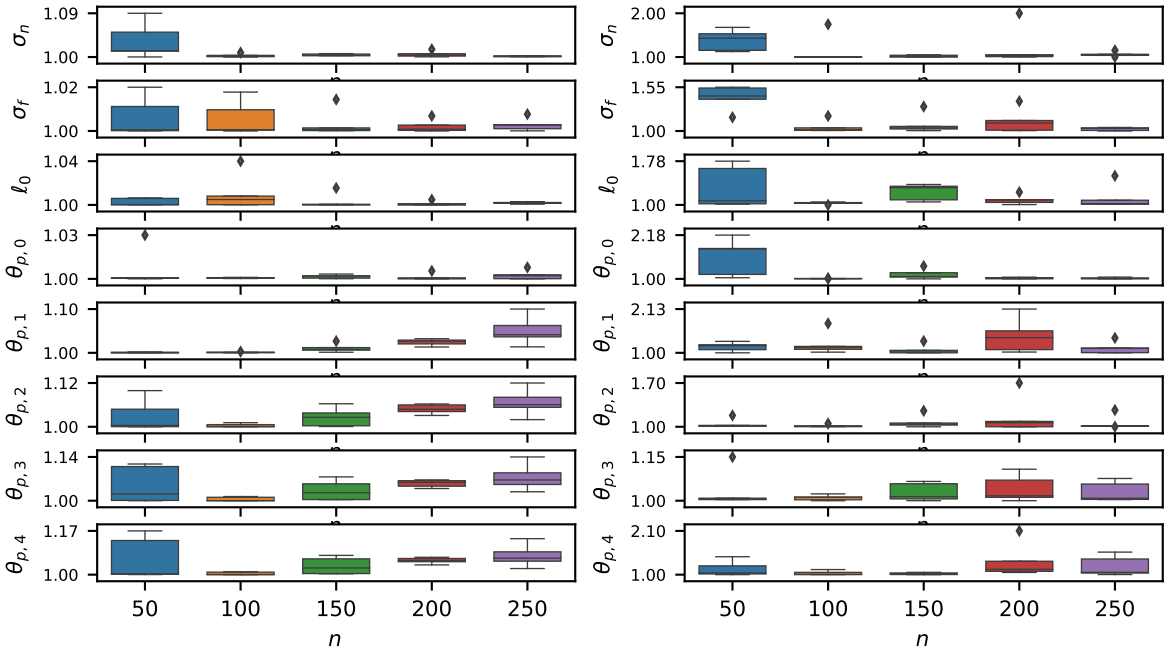
(d) 5D FS / 250 training samples

Figure 2.10: Markov-Chain Monte-Carlo chains of the model parameters for four different combinations of FS dimension and number of training samples. Each chain contains 1,000 draws. Only the first five projection parameters are shown for brevity.



(a) Number of training samples fixed to 100, x-axis is number of FS dimensions.

(b) Number of training samples fixed to 250, x-axis is number of FS dimensions.



(c) FS dimension fixed to 1, x-axis is number of training samples.

(d) FS dimension fixed to 5, x-axis is number of training samples.

Figure 2.11: Values of the split Gelman-Rubin statistics of the MCMC chains as a function of the FS dimension m for two fixed numbers of training samples (a, b), and as a function of the number n of training samples for two fixed FS dimensions (c, d). Parameter names are indicated to the left of the plot.

approximation of the posterior distribution. We have seen the MCMC inference is made increasingly harder as both the number of training samples and the FS dimension increase.

Figure 2.12 shows a comparison of the prior and posterior parameter distributions. We recall that the prior distributions were given in algorithm 2. As before, plots have been restricted to the first five projection parameters. While projection parameter distributions are closely concentrated around well-defined values in the 1D case, we observe that they are spread in the 5D case.

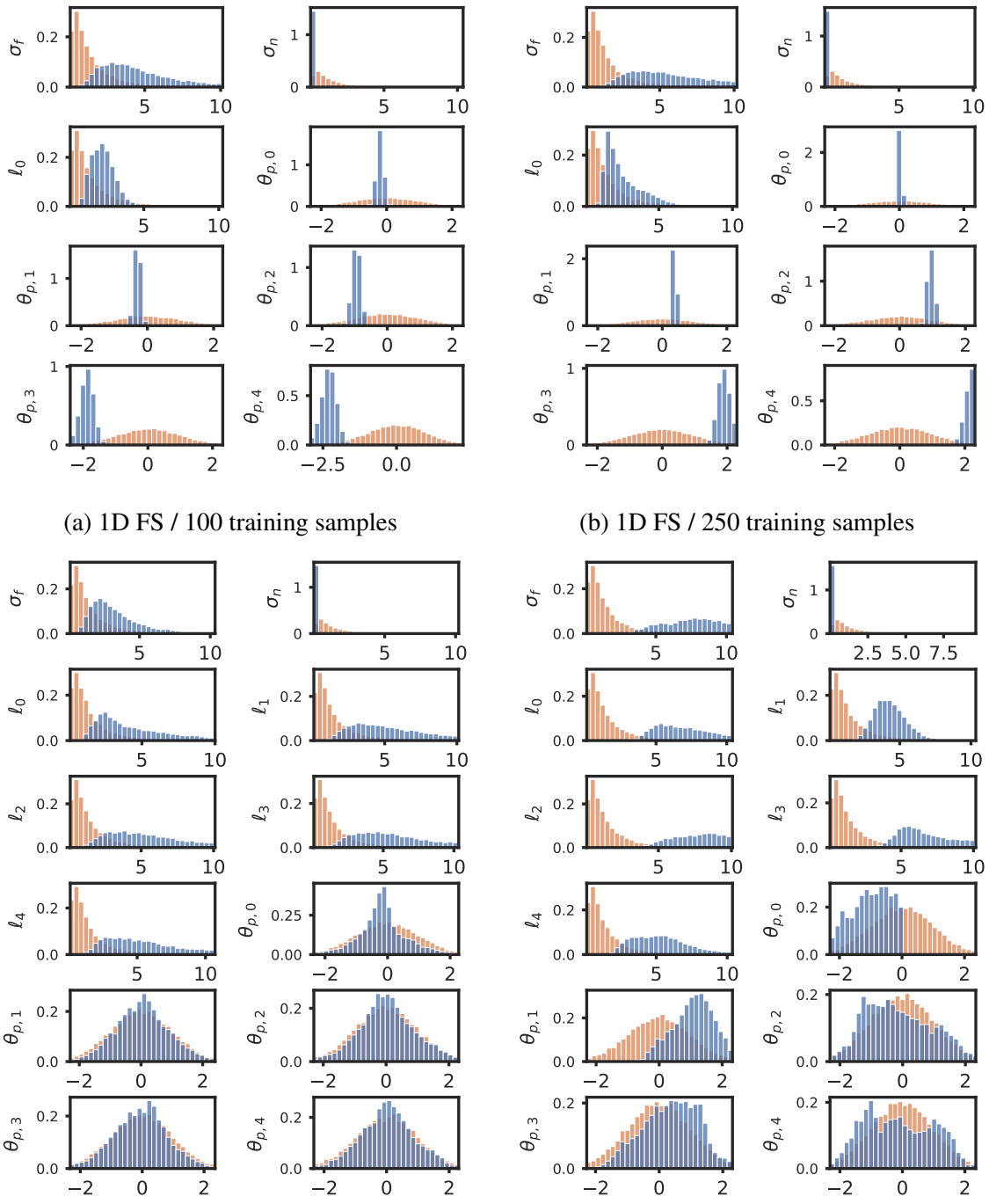
Figure 2.13 depicts a comparison of the actual and predicted lift values for the observations used to train the model. We can see that the model successfully fits the data it was provided, despite the poor mixing observed in the MCMC chains.

Figure 2.14 shows the evolution of the training duration as a function of the FS dimension and number of training samples. As expected since the model relies on GPR, training duration increases as a power-law as a function of the number of training samples. The impact of the FS dimension is mostly visible as it increased from 1 to 2, after which no clear upward or downward trend can be observed.

We can see that despite poor MCMC mixing, the model successfully fits the training data and there is effective learning, as shown by a shift from the prior to the posterior model parameter distributions. In the next section, we will assess the predictive performance of the model.

Predictive Performance

Figure 2.15 depicts the comparison of the actual and predicted lift values for the validation points. We observe that the model satisfactorily generalized to the prediction of points that were not used to train the model, demonstrating the utility of the proposed approach to generate a surrogate model. While fig. 2.15 does not allow to make clear-cut observations pertaining to the effect of the FS dimension or the number of training samples, we study the impact of these factors on global predictive accuracy metrics in the next two figures.



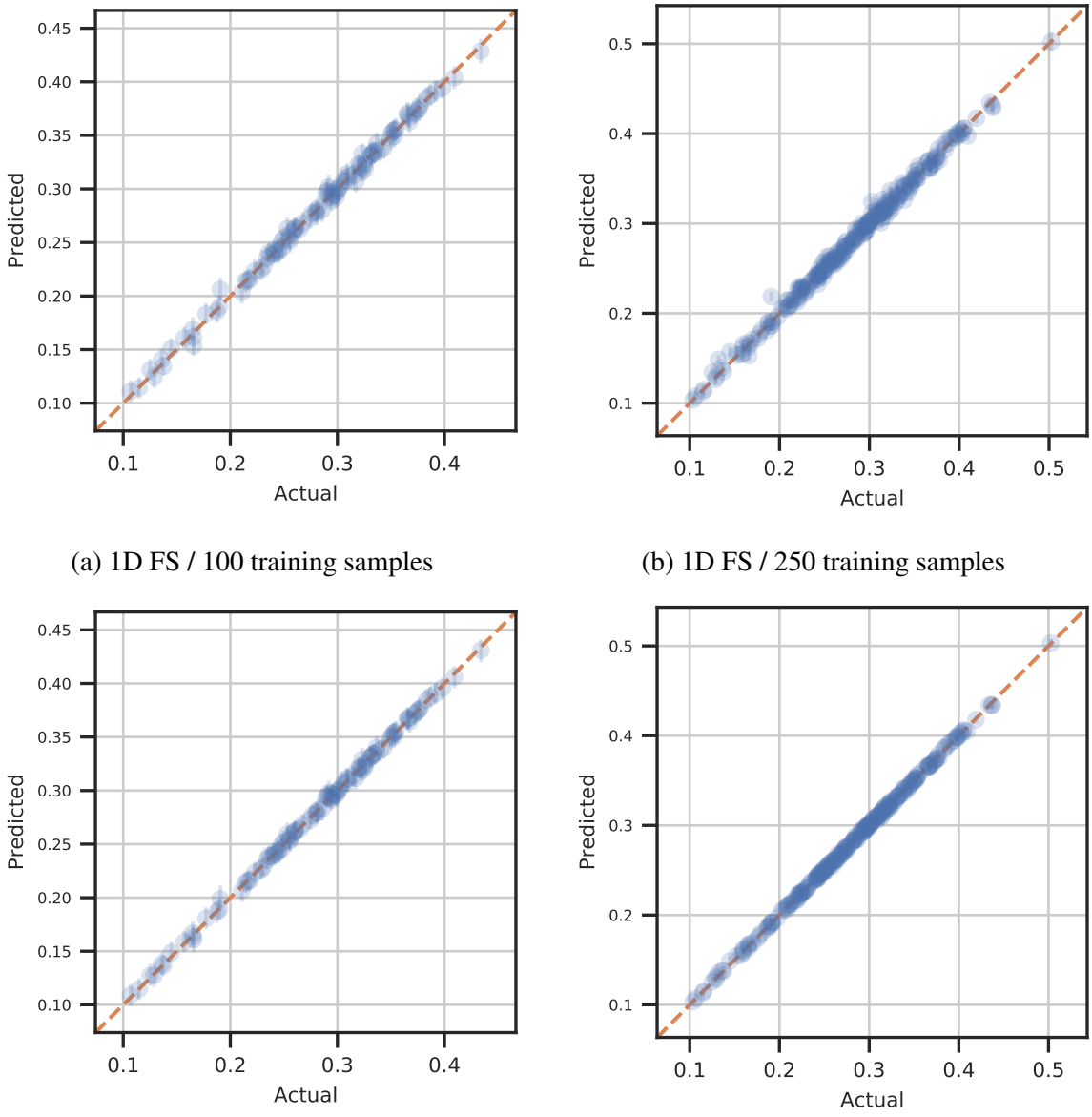
(a) 1D FS / 100 training samples

(b) 1D FS / 250 training samples

(c) 5D FS / 100 training samples

(d) 5D FS / 250 training samples

Figure 2.12: Prior (orange) and posterior (blue) distributions. Parameter names are indicated to the left of the plot. Histograms are normalized such that their respective areas equal 1.



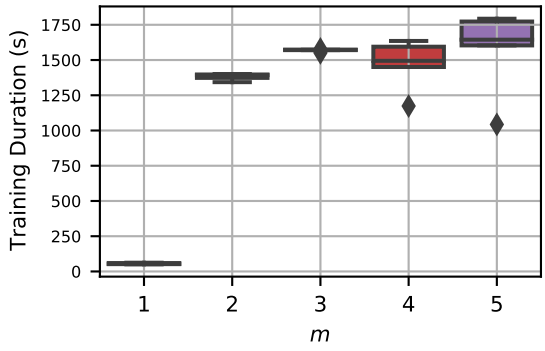
(a) 1D FS / 100 training samples

(b) 1D FS / 250 training samples

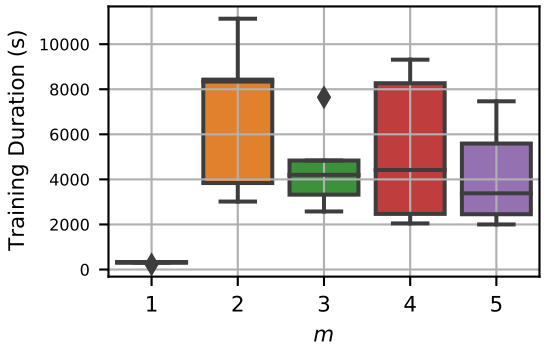
(c) 5D FS / 100 training samples

(d) 5D FS / 250 training samples

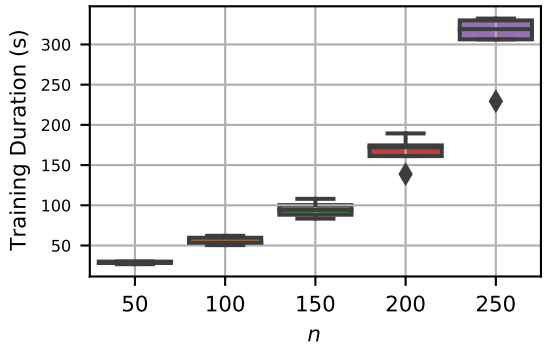
Figure 2.13: Comparison of the training data (Actual) and the model predictions (Predicted) for the ONERA M6 dataset for four different combinations of FS dimension and number of training samples. Vertical bars indicate the 95% confidence interval.



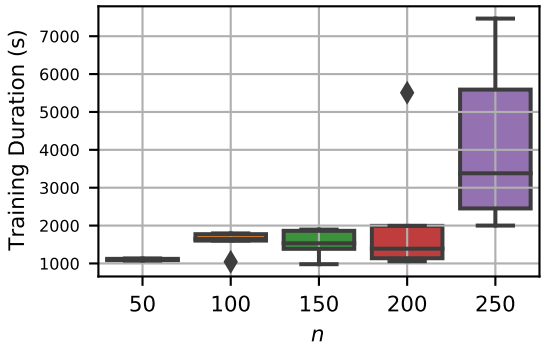
(a) Number of training samples fixed to 100, x-axis is number of FS dimensions.



(b) Number of training samples fixed to 250, x-axis is number of FS dimensions.

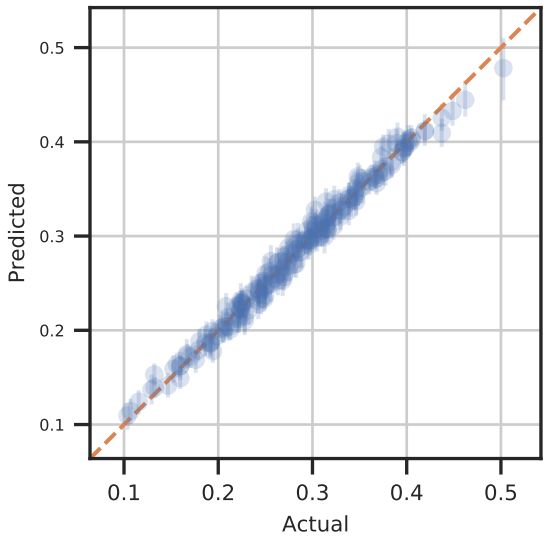


(c) FS dimension fixed to 1, x-axis is number of training samples.

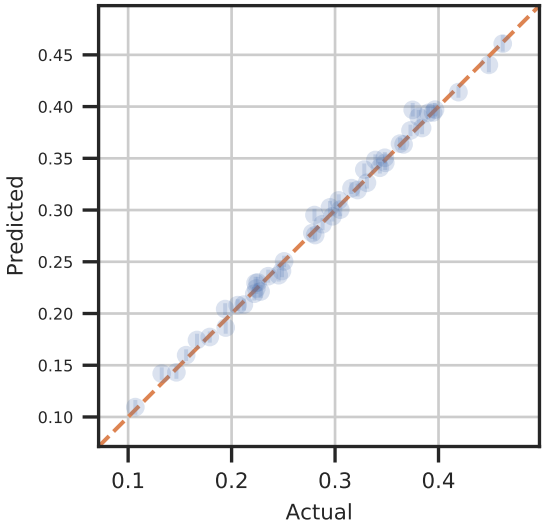


(d) FS dimension fixed to 5, x-axis is number of training samples.

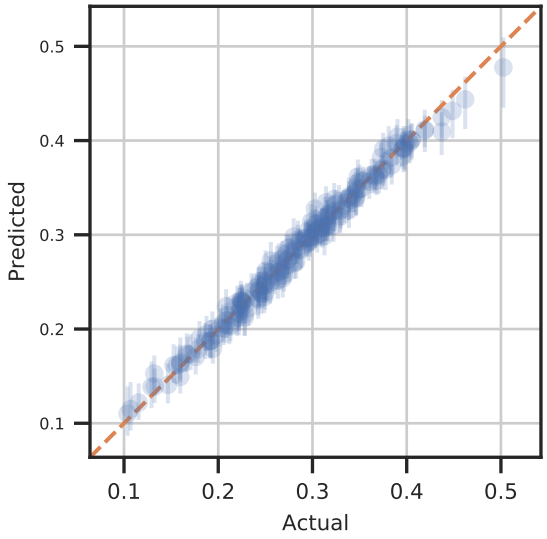
Figure 2.14: Training duration as a function of the FS dimension m for two fixed numbers of training samples (a, b), and as a function of the number n of training samples for two fixed FS dimensions (c, d).



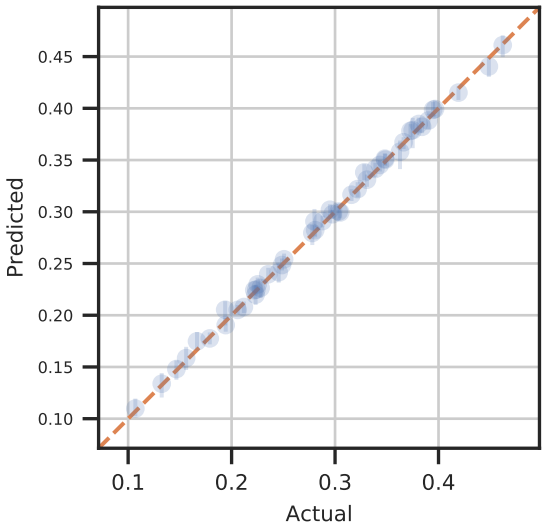
(a) 1D FS / 100 training samples



(b) 1D FS / 250 training samples

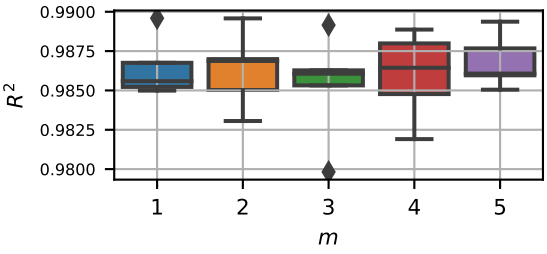


(c) 5D FS / 100 training samples

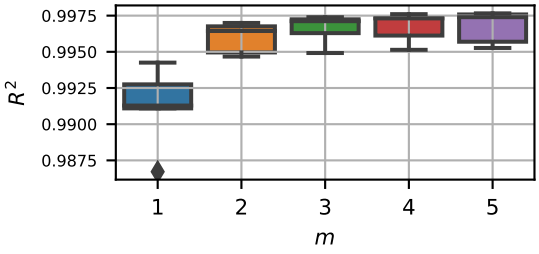


(d) 5D FS / 250 training samples

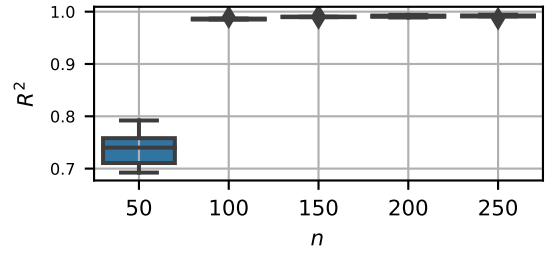
Figure 2.15: Comparison of the actual dataset and the model predictions for validations points, i.e. observations not used during training. Comparisons are shown for four different combinations of FS dimension and number of training samples. Vertical bars indicate the 95% confidence interval.



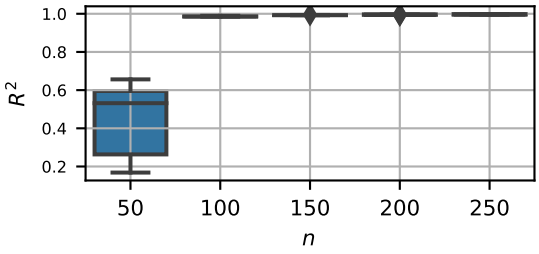
(a) Number of training samples fixed to 100, x-axis is number of FS dimensions.



(b) Number of training samples fixed to 250, x-axis is number of FS dimensions.

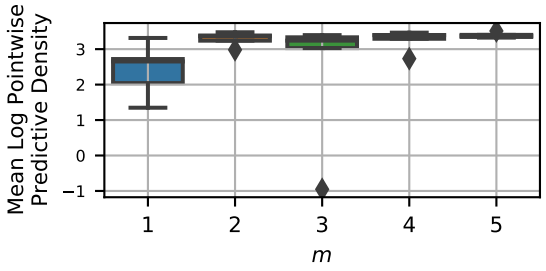


(c) FS dimension fixed to 1, x-axis is number of training samples.

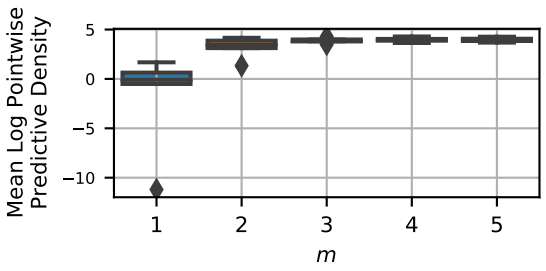


(d) FS dimension fixed to 5, x-axis is number of training samples.

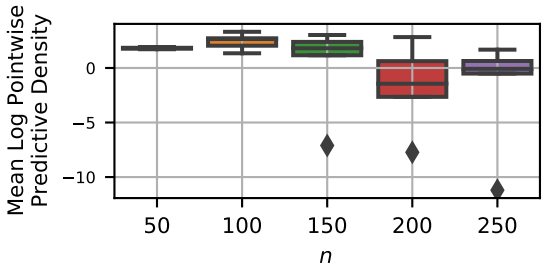
Figure 2.16: Values of the coefficient of determination R^2 as a function of the FS dimension m for two fixed numbers of training samples (a, b), and as a function of the number n of training samples for two fixed FS dimensions (c, d).



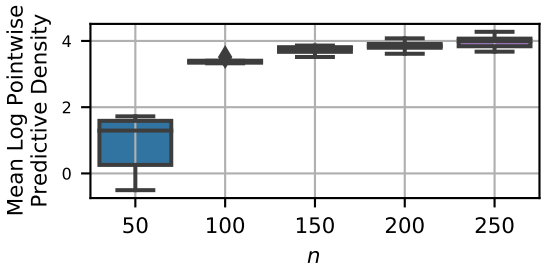
(a) Number of training samples fixed to 100, x-axis is number of FS dimensions.



(b) Number of training samples fixed to 250, x-axis is number of FS dimensions.



(c) FS dimension fixed to 1, x-axis is number of training samples.



(d) FS dimension fixed to 5, x-axis is number of training samples.

Figure 2.17: Values of the mean log pointwise predictive density as a function of the FS dimension m for two fixed numbers of training samples (a, b), and as a function of the number n of training samples for two fixed FS dimensions (c, d).

Figure 2.16 shows the evolution of the coefficient of determination R^2 as a function of both the number of FS dimensions and training samples. An interesting observation can be made from the comparison of a) and b), that respectively correspond to the low and high numbers of training samples. In the former case, increasing the FS dimension does not have a significant impact on the model's predictive accuracy while it does in the latter case. When only sparse observations are available, a single direction may then be sufficient to capture the observed variability while more dimensions become necessary as more numerous, and diverse, observations become available. Figures c) and d) show that irrespective of the number of FS dimensions, the number of training samples has the expected positive impact on the model's predictive accuracy.

Figure 2.17 shows the evolution of the mean log pointwise predictive density as a function of both the number of FS dimensions and training samples. Trends are slightly dif-

ferent from those observed with the coefficient of determination. Here, irrespective of the number of training samples, the MLPPD always tends to increase with the number of FS dimensions. While the MLPPD does increase with the number of training samples when the FS dimension is 5, it tends to decrease for a single-dimensional FS, indicating that the quality of probabilistic predictions is reduced.

2.2.5 Results of Experiment 1.1

A comparative study is carried out to characterize the performance of the proposed method. This section starts with a presentation of the benchmark methods used for comparison. The next two sections present results for each group of datasets: analytical functions first and datasets from science and engineering afterwards. For both groups, results on all four metrics of interest are presented and discussed.

Results on Quadratic Functions

Active Subspace Recovery Figure 2.18 depicts the evolution of the MFSA as a function of the number of samples used to train the model. A low subspace angle indicates that the uncovered FS and the AS are nearly aligned, corresponding to a successful recovery of the AS.

For the cases featuring a 2D FS ($m = 2$), trends seem to indicate that a minimum number of training samples is required to successfully recover the AS, which is indicated by the drop of the subspace angle metric. For all eight benchmark quadratic functions, we observe that B-GP consistently exhibits worse AS recovery capabilities than the other two methods. The performance of B-GP also decreases with the number of input dimensions. While the drop in first subspace angle, characteristic of successful AS recovery, is indeed visible within the range of training set sizes under study when $d = 10$ and $d = 25$, no such drop is visible when $d = 50$ and $d = 100$. Actively seeking the AS by adapting the form of the predictive model to incorporate a projection onto a lower-dimensional subspace

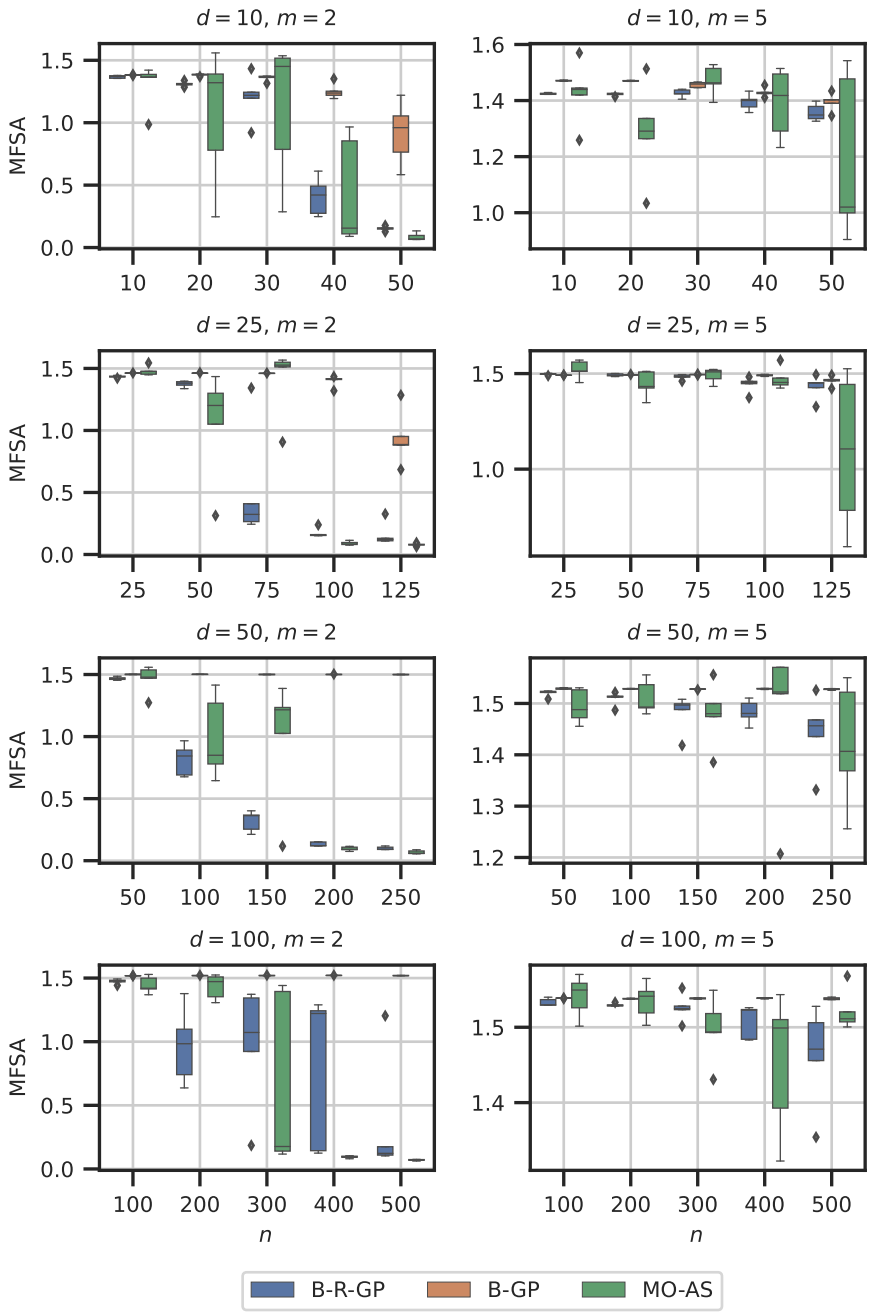


Figure 2.18: Evolution of the mean first subspace angle (MFSA) between the predicted and actual active subspaces for training sets varying in size n from one to five times the number of input dimensions for quadratic function datasets. Plots are organized by increasing input dimension d from top to bottom and by increasing AS dimension m from left to right.

therefore appears to drastically improve the ability of surrogate-based methods to detect the AS when a limited number of training samples is available. Both the B-R-GP and MO-AS have similar AS recovery capabilities. As noted earlier, the distributions for the B-R-GP method have wider spread due to the Bayesian nature of the method: instead of only seeking the most likely AS, the proposed B-R-GP gives access to the full posterior distribution of the AS. Directions that are less likely given the model observations are therefore retained and given a smaller weight. We will see that it enables better quantification of epistemic uncertainty. We also note the skewness of some of those distributions, such as for $d = 25$, $m = 1$, and 50 or 75 training samples: while the distribution almost spans the complete interval of subspace angle values, most of its weight is concentrated on small angle values. Both methods consistently enable the detection of the AS a number of training samples smaller than five times the number of dimensions for 25- to 100-dimensional quadratic functions.

For the cases featuring a 5D FS ($m = 5$), we always observe high subspace angle values which indicate that the FS and AS are misaligned in at least one direction. As we will see however, this does not necessarily significantly impact the predictive accuracy of the model. This may be explained by the fact that even though not all AS directions have been identified, a subset of them may have actually been properly recovered.

Deterministic Predictive Capability Figure 2.19 depicts the evolution of R^2 as a function of the number of samples used to train the predictive model. We recall that, for both Bayesian approaches, the median is used for point-based prediction.

The deterministic predictive capability of B-GP for high-dimensional input spaces is consistent with its poor ability to detect the AS: it nearly always scores worse than both other methods. The relative drop in performance with an increasing number of input dimensions first observed with the first subspace angle is here confirmed: when $m = 2$, the deterministic predictions of B-GP do not improve over the studied range of training

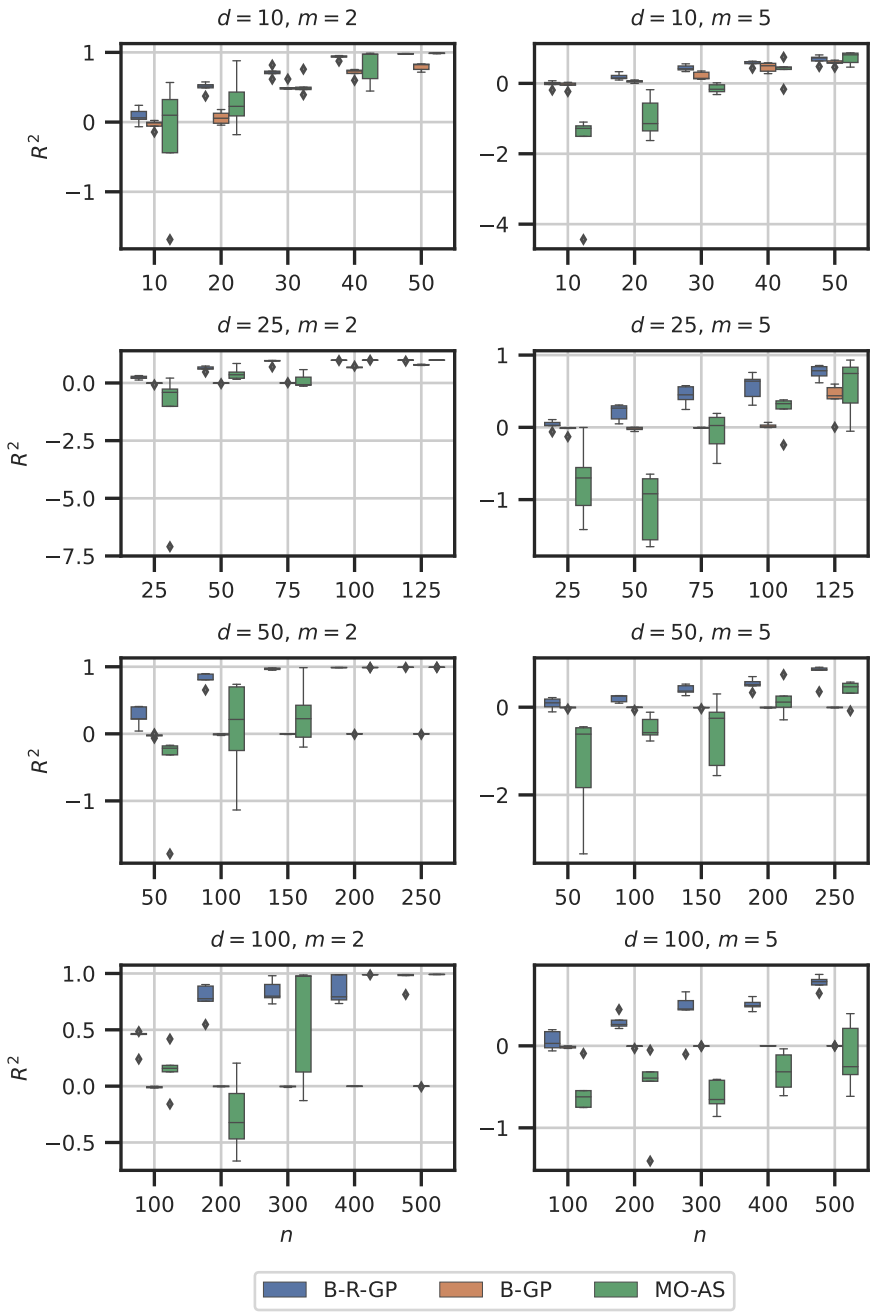


Figure 2.19: Evolution of the validation coefficient of determination (R^2) for training sets varying in size n from one to five times the number of input dimensions for quadratic function datasets. Plots are organized by increasing input dimension d from top to bottom and by increasing AS dimension m from left to right.

samples neither for $d = 50$ nor for $d = 100$.

The comparison of the B-R-GP and MO-AS methods indicates that both methods perform equally well for a number of training samples ranging from three to five times the number of input space dimensions when $m = 2$. For very small training sets and for a higher-dimensional FS ($m = 5$) however, B-R-GP invariably displays much higher R^2 values than the other two methods. This is consistent with the expected superiority of Bayesian methods when few model observations are available. In the light of these results, the proposed B-R-GP approach therefore appears to improve upon the MO-AS approach: at worst it reaches the same performance as MO-AS when a relatively high number of training samples are available and the FS dimension is low, and it significantly increases predictive performance in the sparse data regime and for higher-dimensional FSs.

Probabilistic Predictive Capability Figure 2.20 shows the evolution of the MLPPD as the number of training samples is increased. This metric reveals the quality of the probabilistic prediction: the higher the MLPPD, the most likely it is to observe the actual responses of the validation dataset under the model's predictive distribution.

As expected, both Bayesian methods consistently score better than MO-AS with respect to this metric. MO-AS actually scores so low that the corresponding values were filtered out of the plots to enable readability. This can be explained by the fact that epistemic uncertainty in the MO-AS model is only partially captured: while the GP effectively captures part of the uncertainty, neither the uncertainty in the GP hyperparameters nor in the projection matrix parameters are quantified. Except for the most challenging cases ($d = 50, m = 5$ and $d = 100, m = 5$), the proposed B-R-GP method generally leads to the highest MLPPD values. Even though B-GP scores better in terms of MLPPD for $d = 50, m = 5$ and $d = 100, m = 5$, we recall that R^2 was nearly zero for B-GP in these cases, which indicated very poor point-based predictive accuracy. Conclusions can therefore not be drawn regarding the probabilistic predictive accuracy of the two methods using

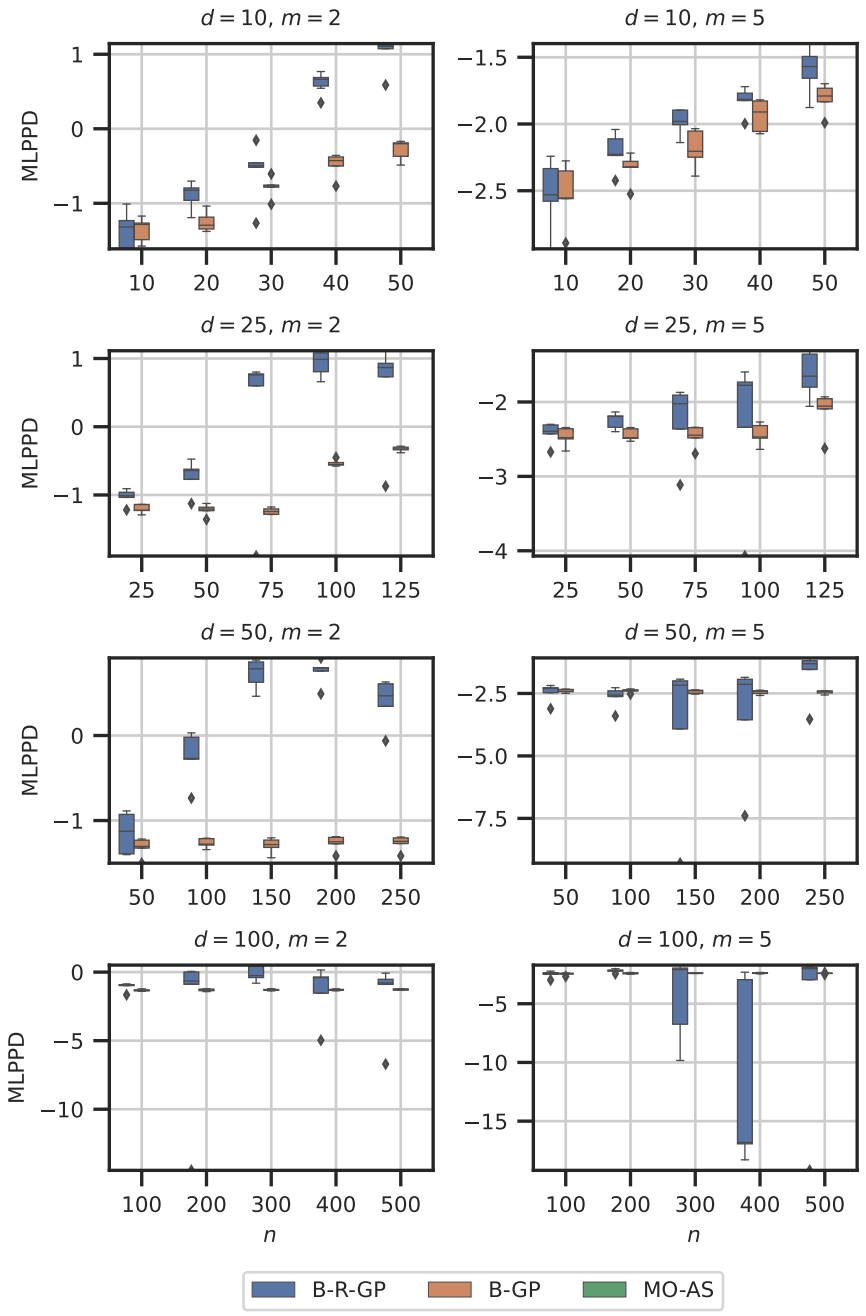


Figure 2.20: Evolution of the mean log pointwise predictive density (MLPPD) for training sets varying in size n from one to five times the number of input dimensions for quadratic function datasets. Plots are organized by increasing input dimension d from top to bottom and by increasing AS dimension m from left to right.

MLPPD in these cases.

Training Duration Figure 2.21 depicts the evolution of the time required to train all three types of models for the different quadratic functions under study. Training time always increases with the number of training samples. This is expected as the cost of computing the inverse of the sample covariance matrix during the GP likelihood computation increases as the matrix size increases. For most figures, a linear trend in log-scale is clearly visible, that corresponds to a power-law scaling in linear scale. Such a behavior is expected for B-GP, for which the $\mathcal{O}(n^3)$ scaling is well-known, where n is the number of training samples.

The B-GP models are consistently faster to train than both projection-based methods. This is expected since those methods require additional computations during the evaluation of the model likelihood and increase the number of parameters to be inferred or optimized without affecting the size of the sample covariance matrix, which is the bottleneck of GP training.

While the proposed B-R-GP model trains faster than the MO-AS model for low input space dimensions and low number of training samples, its training duration becomes larger when dimension and training samples increase. At worst, B-R-GP and MO-AS training times are of the same order. This may be explained by considering the differences between the two methods. On the one hand, in MO-AS, the number of restarts of the gradient-based optimization is fixed but the number of steps during one optimization run (and therefore the number of likelihood evaluations) may vary because a dynamic stopping criterion is used to terminate optimization. On the other hand, in B-R-GP, the MCMC chain length is fixed but the size of the leapfrog steps of the NUTS sampler are adaptively chosen during warmup based on the shape of the likelihood function. In practice, we observed greater variability in the MCMC step size than we did in the number of optimization steps. As the number of input dimensions increases and the likelihood function becomes more challenging to sample, the step size chosen by the NUTS algorithm decreases, thus leading to more HMC

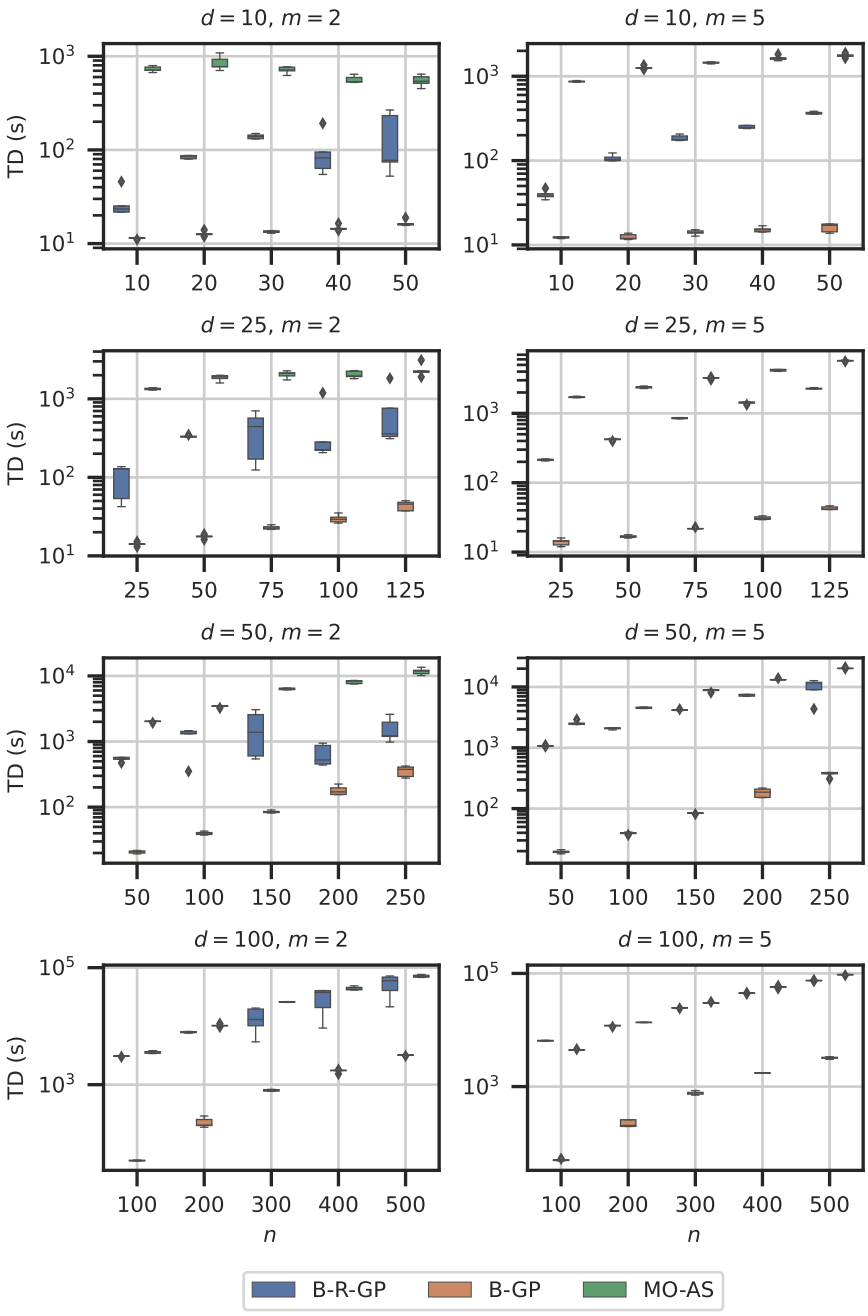


Figure 2.21: Evolution of the training time (TT) for training sets varying in size n from one to five times the number of input dimensions for quadratic function datasets. Plots are organized by increasing input dimension d from top to bottom and by increasing AS dimension m from left to right.

leapfrog steps and increased total sampling times.

Results on Science and Engineering Datasets

This section mirrors the preceding section by presenting results for all four metrics of interest, this time on the benchmark science and engineering datasets. While the quadratic functions may be representative of simple functions encountered in practical engineering applications, the datasets we are working with in this section were generated using actual analyses encountered in scientific or engineering practice.

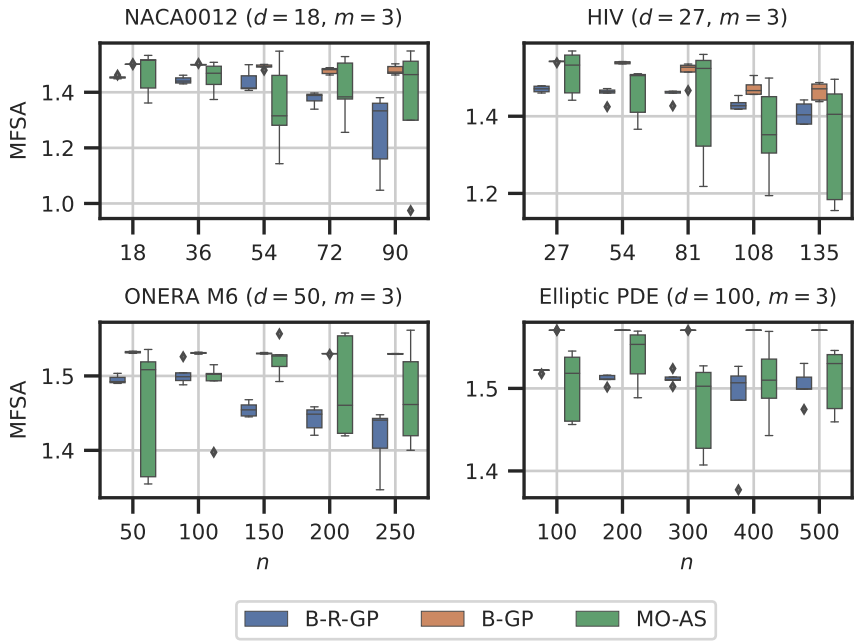


Figure 2.22: Evolution of the first subspace angle (FSA) between the predicted and actual active subspaces for training sets varying in size n from one to five times the number of input dimensions for all four science and engineering datasets.

Active Subspace Recovery Figure 2.22 presents the AS recovery results obtained for science and engineering datasets. We observe that the recovered FSs never correspond to the true AS. As noted before, we will again see that this does not necessarily translate into a poor predictive accuracy.

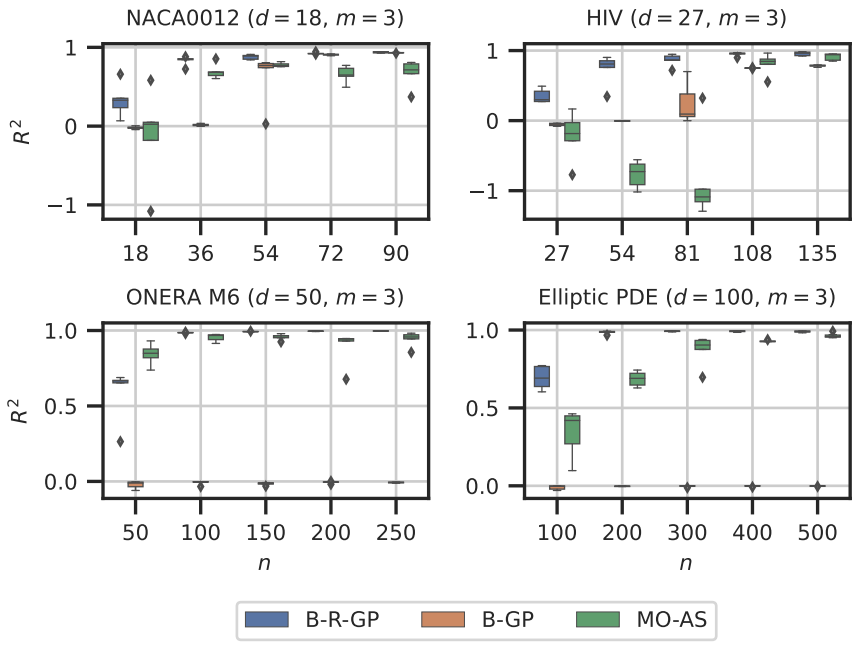


Figure 2.23: Evolution of the validation coefficient of determination (R^2) for training sets varying in size n from one to five times the number of input dimensions for all four science and engineering datasets.

Deterministic Predictive Capability Figure 2.23 shows the evolution of R^2 as a function of the number of training samples for the science and engineering datasets. B-R-GP outperforms both other methods on all four datasets. Where B-GP fails to produce a useful model for any number of training samples within the domain of study for both the ONERA M6 and elliptic PDE datasets, B-R-GP yields well-performing models with as little as a number of training samples corresponding to twice the number of input dimensions. For the NACA0012 and HIV datasets where B-GP eventually leads to satisfactory models, the proposed B-R-GP approach gives access to better models with significantly fewer training samples.

Probabilistic Predictive Capability Figure 2.24 depicts the evolution of the validation MLPPD for different numbers of training samples. Those graphs have been truncated because the MLPPD values for the MO-AS method are consistently significantly lower than the other two methods and compromise the readability of those graphs. As opposed to the

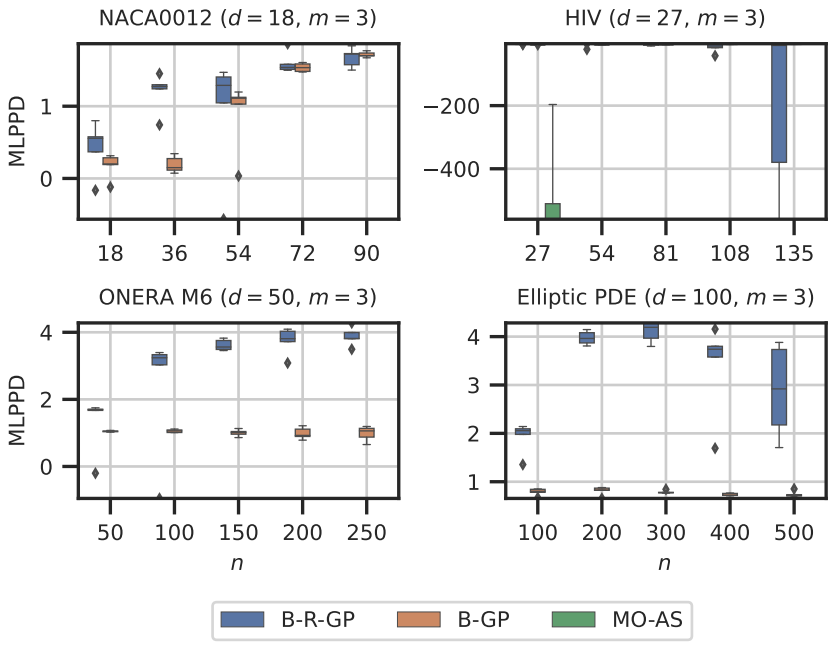


Figure 2.24: Evolution of the mean log pointwise predictive density (MLPPD) for training sets varying in size n from one to five times the number of input dimensions for all four science and engineering datasets.

results shown in fig. 2.20 that were mostly consistent across different numbers of input space and active subspace dimensions, the corresponding results for science and engineering datasets seem to be highly problem-dependent.

For the relatively low-dimensional NACA0012 dataset, MO-AS displays the poorest probabilistic predictive capabilities. B-R-GP exhibits better performance than B-GP for small training sets, but is slightly outperformed by B-GP for larger training sets. This is an illustration of the trade-off occurring when reducing the dimensionality of the inputs. While B-GP operates on the full input space, the proposed method always operates on one of its low-dimensional subspaces. As a result, as the number of training samples increases, B-GP can start capturing variability in the response due to variations of the inputs in the inactive subspace while B-R-GP is limited to capturing variations of the response due to variations of the inputs in the active subspace only, and variations in the inactive subspace are captured as noise.

The 27-dimensional HIV dataset appears to be particularly challenging for both the B-

R-GP and MO-AS methods, as extremely low values of MLPPD are reached. This behavior seems to be highly problem-dependent as it is not encountered for any other of the datasets.

The performance of the proposed B-R-GP method significantly improves for the two highest-dimensional datasets, ONERA M6 and elliptic PDE, with consistently higher MLPPD than both other methods across all training set sizes.

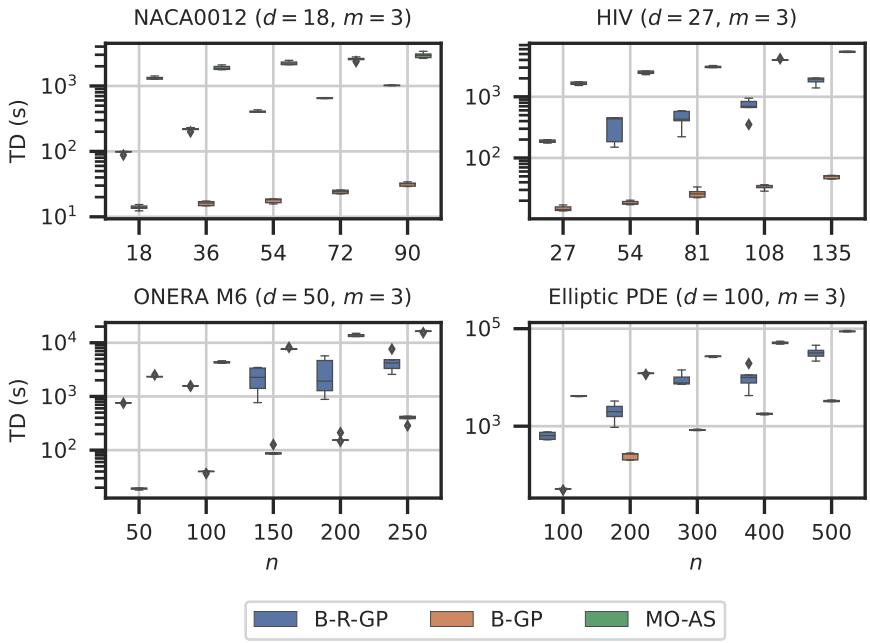


Figure 2.25: Evolution of the training time (TT) for training sets varying in size n from one to five times the number of input dimensions for all four science and engineering datasets.

Training Duration Figure 2.25 shows the evolution of the training time as a function of the number of training samples. B-GP models are consistently faster to train, as fewer model parameters need to be identified compared to projection-based methods. On those four datasets, the training duration is consistently inferior for B-R-GP than it is for MO-AS, even in the case of the elliptic PDE featuring a 100-dimensional input space. Beyond the expected scaling with the number of training samples and the number of inference parameters, training time appears to be highly dependent on the problem-at-hand.

2.2.6 Conclusion

The proposed B-R-GP method was designed to assist the creation of surrogate models of computationally expensive analyses with high-dimensional input spaces and for which access to gradients is not available. It enriches the family of projection-based methods for supervised dimension reduction with a gradient-free and fully Bayesian alternative. In this section, we carried out comparative study of the proposed method with two other state-of-the art methods, MO-AS and B-GP, that focused on four aspects: recovery of the active subspace, deterministic prediction accuracy, probabilistic prediction accuracy, and training time.

The study was carried out on eight analytical functions (25 to 100 inputs) and four science and engineering datasets (18 to 100 inputs) and showed the proposed method to be superior to previously introduced methods mainly due to its improved probabilistic predictive ability. Where optimization-based methods confidently make wrong predictions, the proposed method adequately estimates the uncertainty in its predictions thanks to the fully Bayesian approach. The explicit incorporation of a projection onto a lower-dimensional subspace within the form of the surrogate model was shown to ease the identification of the AS and in turn to improve the predictive capabilities of the resulting surrogate model, as opposed to the surrogate-based approaches to AS, such that B-GP, that aim at first constructing a full-dimensional surrogate model and then using it to find the AS.

From the results of experiment 1.1, we conclude that combining a fully Bayesian approach with a surrogate model based on approximation by ridge function allows to increase the predictive accuracy of the resulting surrogate model when training data is sparse, compared to an MLE-based approach. We confirmed that the better predictive accuracy was associated with the better quantification of epistemic uncertainty that reduces over-fitting. As expected from a Bayesian approach, we found that the benefits of the better uncertainty quantification fade as the size of the training set grows, such that after a point that is dataset-dependent, an MLE-based approach would be sufficient. As was expected, the

benefits of reducing the dimension of the input space grow with the original input space dimension: for the lower-dimensional cases, a Bayesian GP without dimension reduction is usually sufficient.

Conclusion on Hypothesis 1.1

In a study carried out on eight analytical functions (25 to 100 inputs) and four science and engineering datasets (18 to 100 inputs), the proposed approach leveraging Householder transformations to enable the Bayesian training of a GP-based approximation by ridge functions was shown to exhibit higher probabilistic predictive accuracy than two gradient-free state-of-the-art methods to approximation by ridge functions.

2.3 Noise Variance as Dimension Selection Metric

In the previous section, we focused on enabling approximation by ridge functions when only few analysis observations were available by using a Bayesian approach to training. This was achieved by fixing the FS dimension, and although we alluded to the fact that the FS dimension needed to be selected, we did not discuss a process to assist the selection. The selection of the FS dimension is the focus of the current section. We will first review existing approaches to estimating the dimensions of low-dimensional spaces of the input space and identify their shortcomings. Then, we will discuss the proposed our approach that relies on the distribution of the noise variance of the low-dimensional GP.

2.3.1 Background and Research Objective

While the dimension of the AS was fixed throughout the previous study, this is not the case in practice when being confronted to a new dataset. Existing methods to assess the AS dimension have been proposed, notably alongside the two benchmark methods used in this study. In [91], the authors propose to successively train the model assuming different AS

dimensions and select the dimension based on the Bayesian information criterion (BIC). This method may be deemed unsatisfactory as it requires multiple costly training runs to determine the FS dimension. Moreover, it relies on an information criterion (IC) whose assumptions are not valid in the low data regime and that are impractical when carrying out a fully Bayesian approach. In [142], the original AS methodology for selecting the number of active dimensions can be carried out since a surrogate model in the full-dimensional input space is built. However, as shown in this study, significantly more samples are required to obtain a good model in the full-dimensional input space compared to methods explicitly incorporating the projection onto a low-dimensional subspace. This leads to the following gap:

Gap 1.2

Existing methods used to select the FS dimension in the context of approximation by ridge functions either rely on 1) information criteria that are irrelevant in a fully Bayesian approach, or 2) in the AS method, on the decay of the eigenvalues of the uncentered covariance matrix of the gradient, that is not available without access to gradients.

This leads to the following research question:

Research Question 1.2

As part of the proposed approach, what alternative metric to IC can be used to guide the selection of the FS dimension?

The search of the feature space dimension is driven by the desire to have a representation of the input space as compact as possible, that is such that the number of input dimensions is minimized. Because the impact of the curse of dimensionality increases with the number of input dimensions, we minimize its impact by minimizing the number of input dimensions. The dimension selection problem was previously approached as a model selection problem [91, 112]. The goal of model selection is to choose, among a discrete list

of model options, the one that is expected to perform the best when confronted to new data not used for training [143]. In literature, a model's ability to correctly predict unseen data is referred to as its *generalization* capability. While model selection may be performed in various ways, approaches based on IC were previously used in the context of feature space dimension selection. Some IC attempt to estimate the out-of-sample error, such as Akaike information criterion (AIC) or widely applicable information criterion (WAIC) [119, 144, 145]. Others aim at approximating likelihood ratios, such as BIC, as these ratios may be used to select a model [143].

Model selection is used to pick out the model that has the best generalization potential. In other words, model selection attempts to avoid over-fitting when selecting among competing models. Most of the IC discussed above actually work by adding a penalization term to a goodness of fit term in order to account for the fact that, even though a model may better fit the training data, it may perform worse on new data if it becomes over-specialized. In those approaches, the value of the penalization term is computed based on the *flexibility* of the model. By flexibility, we mean the ability to fit a wide variety of training data. Flexibility may be linked to the number of model parameters: the more parameters are available to tune the model, the more flexible it is, and the more it would be penalized. The number of model parameters may either be taken as the actual number of available tuning parameters, as in BIC, or as an effective number of parameters, as in AIC or WAIC. In the context of supervised dimension reduction, the flexibility of the model increases with the number of feature space dimensions. As a result, it is expected that over-fitting may happen if the number of considered feature space dimensions becomes too large in comparison with the number of training samples, and this is what motivated the use of IC-based methods for the selection of the feature space dimension.

However, we do not expect over-fitting to be a problem when using GPR for low-dimensional regression. Indeed, the GP already has built-in mechanisms to automatically perform a trade-off between accuracy and complexity, and the fully Bayesian approach

followed here only adds up to these mechanisms. We will illustrate these mechanisms in the following paragraphs. We will first discuss how the likelihood function of a GP incorporates the trade-off. Then, we will discuss the addition of an additive noise to the GPR model, and how noise level may automatically be assessed. Finally, we will recall how the Bayesian approach helps in reducing over-fitting.

The GP's built-in mechanism for alleviating over-fitting may be understood by observing the likelihood function associated with the standard GPR model. Its expression is recalled in eq. (2.46).

$$\log p(\mathbf{y}|X,\theta) = -\frac{1}{2}\mathbf{y}^T\mathbf{K}^{-1}\mathbf{y} - \frac{1}{2}\log\det\mathbf{K} - \frac{n}{2}\log 2\pi \tag{2.46}$$

Each term of the expression in eq. (2.46) may be associated to a different effect [132]. The first term account for data fit term and decreases as the model gets less flexible. On the other hand, the second term is a complexity penalty whose value decreases as the model gets more flexible. This leads to the maximum of the likelihood function, or more generally the models that lead to the highest likelihoods of observing the training observations, to stand somewhere in the middle between complex and simple models. In that sense, the structure of the GP itself enables an automatic trade-off between models of different extreme complexities and offers a safeguard against overfitting.

As discussed in chapter 2, in GPR, the choice of the kernel is left to the model's designer. The kernel plays a central role in the model as it defines the similarity metric used to compare prediction sites. Informally, this means that, if two points are similar by this metric, then their corresponding outputs should also be similar. On the one hand, isotropic kernels do not differentiate between the different input directions when computing the similarity metric. On the other hand, anisotropic kernels grant different weights to the different input directions. By doing so, they recognize the fact that some inputs may lead to greater variability of the inputs than others. The ARD [146] kernel is an instance of anisotropic kernel based on the squared exponential kernel in which every dimension is equipped with

its own length scale parameter. The expression for the ARD kernel is:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \prod_{i=1}^{d} \exp\left(\frac{(x_i - {x_i}')^2}{2\ell_i^2}\right) \tag{2.47}$$

where $\mathbf{x}$ and $\mathbf{x}'$ are input locations, σ_f^2 is the signal variance and $\ell = [\ell_i, ..., \ell_d]$ are characteristic length scales.

As a result, the distance metric used for each input direction is independently modulated. It was shown that the ARD kernel may be used to perform *feature selection* when higher length scale values are selected, effectively leading to disregarding the corresponding directions. The selection of the length scales is driven by the same accuracy/complexity trade-off discussed previously: a model with fewer effective input dimensions, even when leading to a lower goodness of fit, may still be favored due to its lower complexity.

Another mechanism that counters over-fitting is the addition of a model for noise as part of the probabilistic predictive model, i.e., the discrepancy between observed and predicted values. It introduces another lever in the accuracy/complexity trade-off by making less complex models likelier even though they do not perfectly match observed data. This paragraph recalls the motivation for including noise as part of the probabilistic model and briefly explains how it may impact the complexity trade-off discussed previously. GPR may either be used in the context of interpolation or regression. In the case of interpolation, we force the predicted values to match observations. When dealing with computer simulations, this is usually the desired behavior since observations are deterministic: if we were to run the same computer simulation again, we would obtain the exact same results. In contrast, physical experiments may incorporate some noise, e.g., the impact of environmental factors on the experiment's outcome that are outside the experimenter's control. In that latter case, the GP may be used as a regressor by introducing a noise model, which describes the expected form of the distribution for the discrepancy between observed and "true" values. Parameters of the noise model are added to the other model parameters. They are either optimized in an MLE approach or inferred in a Bayesian inference approach to training. In the absence of further *a priori* knowledge, observation noise is usually modeled as an inde-

pendent zero-mean Gaussian distribution, and this is the choice we made in the proposed model. As a consequence, a single extra model parameter is introduced for the variance of this Gaussian distribution, the noise variance. We incorporated noise in the proposed Bayesian model even though we are working with computer models because, as discussed in section 2.2.2, discarding input dimensions effectively leads to adding noise in the response. The proposed approach may actually be thought of as trading input dimensions for noise. The effect of adding a model for the noise is an additional degree of freedom in the likelihood function. The updated expression for the likelihood function for a GPR model incorporating a Gaussian noise of variance σ_n^2 is recalled in eq. (2.48).

$$\log p(\mathbf{y}|X,\theta) = -\frac{1}{2}\mathbf{y}^T\left(\mathbf{K}+\sigma_n^2 I\right)^{-1}\mathbf{y} - \frac{1}{2}\log\det\left(\mathbf{K}+\sigma_n^2 I\right) - \frac{n}{2}\log 2\pi \qquad (2.48)$$

In addition to model complexity that is controlled through the values of the length scales, as discussed in the previous paragraph, the value of the noise variance is another way to vary the value of the likelihood function. Increasing noise variance allows the model to not perfectly interpolate the observed points, but still introduces a penalty when predictions do not match observations. Non-zero noise variance may be favored in cases where it enables an increase of the complexity term while only minimally decreasing the fit term. We note that usually, length scale and noise variance distributions are negatively correlated: the higher short-scale variability is in the model, the easier it can interpolate, so introducing noise variance in those cases is is not as necessary as it is for less complex models.

The last mechanism that the proposed approach leverages to avoid over-fitting is its reliance on Bayesian inference instead of MLE for the training of the predictive model. Benefits of this procedure were already highlighted in the results of experiment 1: the quality of the probabilistic predictions is consistently improved compared to the same model trained using MLE, especially in the low-data regime. When following a fully Bayesian approach, we obtain samples from the joint posterior distribution of model parameters instead of point values as with MLE. The posterior parameters distribution account for the fact that many possible values of the model parameters may lead to predictions that sat-

isfactorily match observations, although to different extents. This has a particularly high impact in the low-data regime: while observations may be more likely under an overfitting model, which would therefore be the one picked up by an MLE-based approach, a fully Bayesian inference would recognize that observations have comparable likelihood under other models. This would be directly noticeable by looking at the posterior distribution of the parameters.

2.3.2 Proposed Method

Given that multiple mechanisms to avoid over-fitting are already built in the proposed modeling approach, we propose to follow a dimension selection approach that does not rely on IC. Instead, by analogy to widespread methods used in unsupervised dimension reduction , we propose to base the decision on the amount of variance lost in the dimension reduction process. The next paragraph gives a high-level overview of the main ideas underlying the proposed approach. The following paragraphs will then dive into more details and offer more careful explanations motivating the proposed approach.

When applying PCA to find a more compact representation of a dataset, the selection of the number of dimensions is traditionally performed by observing the decay of the eigenvalues of the dataset's covariance matrix. Each eigendirection's eigenvalue is equal to the variance of the dataset in that particular direction. Therefore, by discarding only those directions with the smallest eigenvalues, we ensure that the retained variance is maximized. As discussed in section 2.1.1, unsupervised methods are not applicable when reducing the dimension of the input space in a supervised learning context. In turn, their approach to selecting the subspace dimension is not directly applicable either. However, the same idea of *retaining as much variance as possible* may be transposed to the supervised context. In contrast to the unsupervised approach however, we seek to quantify the variance of the function's output that is due to the variations of the inputs in certain directions, not simply the variance of a dataset in some directions.

Some background concepts are useful to developing a more thoughtful justification for the proposed approach, starting with a precise definition of the meaning of the variance of a function's output. The decomposition of a function output's variance based on its inputs is not as straightforward as the decomposition of a dataset's variance based on its features. It may be achieved using the Sobol' variance-based decomposition [147]. This decomposition is fundamental to the field of global sensitivity analysis and forms the basis for Sobol' global sensitivity indices.

The decomposition is as follows:

$$f(\mathbf{x}) = f_0 + \sum_{i=1}^{d} f_i(x_i) + \sum_{i<j} f_{i,j}(x_i, x_j) + \cdots + f_{1,2,\ldots,d}(x_1, x_2, \ldots, x_d) \tag{2.49}$$

Let us equip the input variables with a probability definition $p(\mathbf{x})$ defined over their domain $\mathcal{X}$ and let E denote the expectation operator. We use the shortcut notation $\mathrm{E}_{\sim x_i, x_j}$ to denote the expectation taken with respect to all components of $\mathbf{x}$ *except* x_i and x_j. Using these notations, the individual terms $f_0, f_i, f_{i,j}, \ldots$ of the decomposition are successively defined as:

$$f_0 = \mathrm{E}_{\mathbf{x}}\left[f(\mathbf{x})\right] \tag{2.50}$$

$$f_i(x_i) = \mathrm{E}_{\sim x_i}\left[f(\mathbf{x})|x_i\right] - f_0 \qquad \forall i:\ 1 \leq i \leq d \tag{2.51}$$

$$f_{i,j}(x_i, x_j) = \mathrm{E}_{\sim x_i, x_j}\left[f(\mathbf{x})|x_i, x_j\right] - f_i - f_j - f_0 \qquad \forall i, j:\ 1 \leq i < j \leq d \tag{2.52}$$

Higher-order terms are defined similarly by subtracting all lower-order terms that include all of the term's variables as well as f_0. Based on this definition of the component functions, it follows that, under the condition that f is L^2, which is generally the case for functions encountered in an engineering setting, the variance of f may be decomposed as the sum of the individual component functions. Let V denote the variance operator, then:

$$\mathrm{V}[f] = \sum_{i=1}^{d} \mathrm{V}[f_i] + \sum_{i<j} \mathrm{V}[f_{i,j}] + \cdots + \mathrm{V}[f_{1,2,\ldots,d}] \tag{2.53}$$

Equation (2.53) shows that we can rigorously tell apart the contributions of each input variable and group of input variables to the variance of a function's output of interest.

We can now leverage this decomposition in order to better understand the impact of neglecting input dimensions, which is exactly what we seek in the context of supervised dimension reduction. Let us start by recalling the decomposition of the input space first used in section 2.1.3 when introducing the AS method. Let $\mathbf{W}$ is the $d \times m$ projection matrix onto the *active* subspace while $\mathbf{W_i}$ is the $d \times (d-m)$ projection matrix onto the *inactive* subspace. We are reusing the terminology from the AS method but the argument made here more generally applies to any feature space of interest that is not necessarily the active subspace, and to the complement of this feature space, which in turn may not be the inactive subspace. The original mapping of interest can then be rewritten as $f(\mathbf{x}) = f(\mathbf{W}\mathbf{z} + \mathbf{W_i}\mathbf{z_i})$ where $\mathbf{z} = \mathbf{W}^T\mathbf{x}$ is the component of the inputs in the FS and $\mathbf{z_i} = \mathbf{W_i}^T\mathbf{x}$ is the component of the inputs in the orthogonal complement of the FS. We introduce the alternative mapping h such that $h(\mathbf{z}, \mathbf{z_i}) = f(\mathbf{W}\mathbf{z} + \mathbf{W_i}\mathbf{z_i})$. h is simply an alternate version of f whose domain is a rotated version of f's domain.

We may now apply the variance decomposition from eq. (2.53) to h and gather the component variables in two groups. We include all component functions that are only function of the feature space variables $\mathbf{z}$ in the first group. The second group contains the rest of the component functions, that is all those that incorporate at least one variable in $\mathbf{z_i}$. By "projecting" the original inputs onto the feature space, we are effectively neglecting all the component functions in the second group. The proposed approach models these neglected component functions as a zero-mean Gaussian noise. The zero-mean assumption is coherent with the definition of the component functions of the Sobol' decomposition. The variance of the inferred Gaussian distribution grants access to an estimate of the variance lost due to neglecting those component functions. Instead of attempting to accurately model the functional dependence on the variables $\mathbf{z_i}$ and interactions between $\mathbf{z}$'s and $\mathbf{z_i}$'s, we aggregate then as a random noise such as to be able to still assess roughly their high-level impact on the response.

To better understand how the proposed approach to selecting the feature space dimen-

sion in the supervised context relates to how it is usually carried out in unsupervised learning, we briefly recall how the the feature space dimension is selected in PCA. PCA may be thought of as fitting a dataset to a multivariate Gaussian distribution. The eigenvectors of the covariance matrix correspond to the coordinate frame in which the Gaussian is aligned with the axes, i.e., the covariance matrix is diagonalized, and the respective eigenvalues correspond to the magnitude of the variance in each direction. A dimension is selected by observing the decay of these eigenvalues: the retained variance increases as we keep accumulating variance from additional directions until a pre-determined threshold is reached. As a side note, we draw attention to a limitation of the PCA approach that will also affect the proposed approach. The whole process described previously is limited by the data available for training the model. In particular, if this subset of data is not representative of the original dataset or phenomenon under consideration, then important directions may be left out.

Unlike the eigenvalues leveraged when applying PCA, a direct measure of the retained variance is not directly available in the supervised context. This is the role of the Gaussian noise integrated to the proposed GP-based model. In analogy to the dimension selection process for PCA, we propose to train successive models by increasing the dimension and compare the distributions of the noise variance parameter. By comparing the noise variance distributions of the successive models, we are comparing the variance that the model believes it is leaving on the table. A distinction with the unsupervised case is that the estimation of the variance lost due to the dimension reduction process relies on the mapping from the reduced inputs in the feature space to the output obtained through supervised learning. Therefore, we expect the proposed approach to struggle when training data is too sparse to obtain a credible surrogate model for the link function. As in the unsupervised case, the estimation of this variance is approximate due to the fact that we are only working with a limited number of observations. Therefore, if the available data does not contain sufficient information to notice the importance of a particular input direction, it may be left

out even if it actually contributes significantly to the variance of the original mapping.

We propose the following approach to selecting the feature space dimension:

1. train successive models assuming different feature space dimensions within a predefined range according to the approach previously proposed and validated in section 2.2;
2. graph the evolution of the noise variance distribution;
3. select the feature dimension that corresponds to the threshold between the regime where noise variance is decreasing and the regime where noise variance is approximately constant or re-increasing;
4. if the noise variance remains approximately constant for all considered feature space dimensions, then use a 1D feature space.

We formulate the following hypothesis motivated by the discussion of section 2.3.1:

Hypothesis 1.2

The decay of the noise variance provided by the model can be used to select a relevant feature space dimension, as measured by out-of-sample R^2, because it provides an appropriate estimate of the response variance discarded when using the low-dimensional feature space instead of the original input space.

2.3.3 Setup of Experiment 1.2

Need for Experiment 1.2

In the previous section, we discussed the purpose of the zero-mean Gaussian term introduced in the proposed predictive model: capture the variance of the response not already captured by the low-dimensional GP. The proposed approach to dimension selection using the decay of the noise variance relies on the assumption that the captured noise corresponds

to variations of the response due to variations in the inputs outside the FS. This was motivated by the fact that, if we had unlimited access to the underlying function's analytical form or unlimited evaluations, we could exactly break down the variations of the function using the Sobol' decomposition and tell whether they are due to input variations in the feature space or outside of it. In the setting of surrogate modeling however, we work with a limited number of analysis observations. Because of the limited number of observations, the resulting surrogate model may account for noise in different manners: the predictive model may either attempt to interpolate it, in which case the noise would be captured by the GP instead of the Gaussian noise, or it may actually capture it as part of the Gaussian noise. This is a form of over-fitting to which the GP is susceptible despite its built-in quantification of epistemic uncertainty. This is however usually alleviated when adopting a fully Bayesian approach that effectively recognizes the trade-off between a pure interpolator (shorter length scale, lower noise variance) and a regressor that accounts for the presence of noise (longer length scale, higher noise variance). It therefore remains to check that the proposed approach effectively incorporates the variations outside the feature space as noise in the model, leading to the noise variance decay that would enable the selection of a relevant FS dimension.

Experiment Design

In order to check that the noise variance effectively decays with the number of FS dimensions and that an appropriate FS dimension can be selected based on this decay, we need to train successive models with different FS dimensions. We will vary the FS dimension within a pre-defined range and a different surrogate model will be trained for every considered FS dimension. Since the distribution of the noise variance is a byproduct of training the models, once the models corresponding to all FS dimensions have been trained, we will be able to visualize its evolution as a function of the number of FS dimensions. By visualizing its evolution, we will be able to check whether the decaying behavior is indeed

observed, with two regions: an initial decrease followed by a plateau.

In addition to the decaying behavior, we also need to check that the FS dimension determined using this method is appropriate. By appropriate, we mean that it allows to accurately represent the analysis under study. As in the previous experiment, we will use the out-of-sample coefficient of determination (R^2) to measure predictive accuracy. Therefore, along with the noise variance, we will be examining the evolution of R^2 and we will be checking that the FS dimension selected using the proposed approach leads to the highest R^2 values across the considered FS dimensions. In fact, we expect the noise variance and R^2 to have a symmetric evolution: as we capture more of the variance through the GP instead of the noise as the FS dimension is increased, the predictive accuracy should accordingly increase. This comparison will allow us to check that we are neither underestimating the FS dimension (in which case we would see a low noise variance with a low R^2), or overestimating it (in which case we would see a high noise variance with a high R^2).

The proposed approach to selecting the FS dimension is a component of the proposed surrogate modeling method based on approximation by ridge functions. As such, it operates on an underlying analysis and aims at accurately replicating it. Because we do not have a particular target analysis but are rather building a general-purpose tool for supporting engineering design activities, we test the proposed approach on a set of representative analyses. The datasets, which include those used in experiment 1.1 (see section 2.2.3) are reused and complemented with additional engineering datasets. They are discussed in the subsection *Test Datasets* below.

The same model assumptions and training parameters as in experiment 1 and detailed in section 2.2.3 are used in this experiment. The prior distributions for the various model parameters are the same as in experiment 1 and exposed in section 2.2.2. Due to the high computational cost incurred during model inference and the relatively large number of model trainings required to vary the number of training samples and make repetitions, the length of the MCMC chains is again limited to 1,000 draws, preceded by a 500 draws-long

warmup phase. All computational tasks are performed on the same hardware.

In the subsection *Sensitivity Analysis* below, we will discuss the parameters varied as part of the experiment, leading to the creation of a DOE allowing to assess their effects. For every case of the DOE, the following steps are followed to run the experiment and gather results: 1) select training data, 2) train surrogate model, 3) validate surrogate model, and 4) save results. The source code used to generate the results presented thereafter has been made publicly available[2].

Test Datasets

The motivation for using analytical functions when real-life engineering datasets are available is the ability to fine-tune some of their features. In the current experiment, we are focusing on the dimension of the underlying feature space and resorting to analytical functions allows us to artificially choose a feature space whose dimension is known. We picked a quadratic form in the feature space of those functions because it is a form readily encountered in a real engineering setting, even though they may be deemed simple to replicate from a function approximation standpoint. The set of quadratic functions used in experiment 1.2 has been enriched compared to those in experiment 1 by adding 5D feature spaces.

Engineering datasets allow to assess the applicability of the proposed approach on realistic responses. As a result, they allow to bring more confidence in the method's ability to perform well in real engineering situations, as well as develop a more nuanced understanding of the method's limitations. Compared to the science and engineering datasets from experiment 1, the datasets studied in experiment 1.2 include 1) additional responses, such as drag for both the NACA0012 and ONERA M6 datasets, and 2) new complete additional aerodynamic datasets for the CRM wing in the subsonic and transonic regimes and for the RAE2822 airfoil in the transonic regime that each contain multiple responses. The two extra datasets (RAE2822 and CRM) introduced in this experiment are discussed below.

[2]https://gitlab.com/raphaelgautier/thesis_experiments_part2

The complete list of datasets used in experiment 1.2 to test hypothesis 1.2 is presented in table 2.3. Like in experiment 1, datasets include both analytical functions as well as datasets originating from science and engineering.

RAE2822 The RAE2822 dataset was created by fellow researchers at the aerospace systems design laboratory (ASDL) first for their own studies [112, 148] and later shared with the author. Data was generated using a CFD analysis, specifically SU2's Reynolds-Averaged Navier-Stokes (RANS) solver along with the Spalart-Allmaras (SA) model of turbulence. The Mach number is set at $M = 0.725$. The 51 input parameters correspond to the angle of attack, that is varied uniformly between -2° and $+2^\circ$ along with 50 control points governing the deformation of an FFD box enclosing the airfoil shape, that are uniformly varied within the hypercube $[-0.03, 0.03]^{50}$. The outputs considered are the lift, drag, and moment. Compared to the NACA0012 dataset, the RAE2822 dataset 1) features significantly more inputs (51 instead of 18), 2) takes the angle of attack as an additional input parameter beyond the shape parameters, and 3) has an additional scalar output. When it comes to creating a surrogate for the analysis, the advantage of including the angle of attack as an input parameter is that the surrogate can be reused for the analysis of different flight conditions instead of having to regenerate a new surrogate every time.

CRM Like the RAE2822 dataset, the CRM dataset was created by fellow researchers at ASDL first for their own studies [112, 148] and later shared with the author. Data was also generated using SU2's RANS solver along with the SA model of turbulence, but this time it simulates the flow over a complete 3D wing instead of a 2D airfoil. The flow was simulated for both a subsonic Mach number $M = 0.3$ and a transonic Mach number $M = 0.85$. In both cases, the angle of attack is kept fixed at $\alpha = 2^\circ$. The CFD relies on an unstructured grid of 450,000 cells made available by the University of Michigan's MDO Lab [149]. The 50 input parameters all correspond to control points of an FFD box used to smoothly deform the wing geometry. They vary within the hypercube $[-0.05, 0.05]^{50}$. The outputs

are integrated quantities computed from the flow solutions: the three forces lift, drag, and side force, and the three moments around each of the three axes. Compared to the ONERA M6 dataset, the CRM dataset features 1) different flight conditions, and 2) additional output quantities.

Sensitivity Study

As in experiment 1.1, we expect multiple parameters to have an impact on the applicability of the proposed approach, and therefore on the validity of hypothesis 1.2.

The characteristics of the datasets are expected to have an effect. As discussed above, multiple datasets with high-dimensional inputs, generated using analytical functions as well as engineering simulations, are used to assess the proposed approach. Their input dimensions range from 10 to 100, and they have varying levels of complexity.

The number of training observations is also expected to have an impact on the ability to build an accurate model, and therefore the ability to detect the noise introduced when reducing the dimension of the input space. Because we focus on the low-data regime, the number of training observations is kept relatively low, as we vary it between one and five times the number of inputs. More details are provided in the subsection *Selection of the Training Set* below.

As discussed previously, we are interested in visualizing the evolution of the noise variance as a function of the number of FS dimensions. The value of the FS dimension is therefore varied as part of the experiment. As opposed to experiment 1.1 in which we were comparing the proposed fully Bayesian approach to other methods, the focus here is dimension selection using the proposed approach. Therefore, the surrogate modeling method is fixed; we use the B-R-GP proposed and studied previously in section 2.2.

We define ranges for numerical parameters and generate a full-factorial DOE to study the variations of all parameters (see table 2.4).

Table 2.3: Summary of test datasets used in experiment 1.2. QF X/Y refers to the quadratic function with X inputs and a Y-dimensional feature space.

Group	**Dataset**	**Response**	**Total Number of Observations**	**Input Dim.**	**FS Dim.**	**Ref.**
Analytical Quadratic Functions	QF 10/1	y	1000	10	1	–
	QF 10/2	y	1000	10	2	–
	QF 10/5	y	1000	10	5	–
	QF 25/1	y	1000	25	1	–
	QF 25/2	y	1000	25	2	–
	QF 25/5	y	1000	25	5	–
	QF 50/1	y	1000	50	1	–
	QF 50/2	y	1000	50	2	–
	QF 50/5	y	1000	50	5	–
	QF 100/1	y	1000	100	1	–
	QF 100/2	y	1000	100	2	–
	QF 100/5	y	1000	100	5	–
Active Subspace	NACA0012	lift	1756	18	–	[110]
		drag	1756	18	–	[110]
	HIV	$y_{t=3400}$	1000	27	–	[137]
	ONERA M6	lift	297	50	–	[138]
		drag	297	50	–	[138]
	Elliptic PDE	y_{short}	1000	100	–	[104]
		y_{long}	1000	100	–	[104]
Aerodynamics	Subsonic CRM	drag	2001	50	–	[148, 115]
		lift	2001	50	–	[148, 115]
		x-moment	2001	50	–	[148, 115]
		y-moment	2001	50	–	[148, 115]
		z-moment	2001	50	–	[148, 115]
		sideforce	2001	50	–	[148, 115]
	Transonic CRM	drag	2001	50	–	[148, 115]
		lift	2001	50	–	[148, 115]
		x-moment	2001	50	–	[148, 115]
		y-moment	2001	50	–	[148, 115]
		z-moment	2001	50	–	[148, 115]
		sideforce	2001	50	–	[148, 115]
	RAE2822	drag	3000	51	–	[148, 115]
		lift	3000	51	–	[148, 115]
		z-moment	3000	51	–	[148, 115]

Table 2.4: Experiment 1.2 – Summary of the parameters varied in the parametric study

Parameter	**Parameter Values**
Datasets	see table 2.3
FS dimension	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Number of Training Samples (times number of inputs)	[1, 2, 3, 4, 5]
Number of Training Repetitions	5 repetitions with fixed random seeds
Surrogate Modeling Method	fixed, B-R-GP

Selection of the Training Set The proposed approach aims at improving surrogate modeling in the low-data regime where observations are sparse. We therefore consider relatively small training sets whose sizes range from one to five times the number of input dimensions of the corresponding mapping. As in section 2.2.3, we will vary the number of training samples by increments equal to the number of input dimensions such as to study five different levels of training set sizes.

The datasets used in this study were pre-generated. As such, the sampling locations were already fixed and sampling approaches such as a DOE were not applicable in this context. Instead, training samples are randomly selected among the dataset rows using a uniform distribution. While it is not part of the current study, the impact of the DOE and adaptive sampling methods will be studied in chapter 4.

In order to account for the particular choice of training points, training will be repeated five times using different splits of the datasets between training and validation points. To ensure reproducibility, the uniform distribution used for selecting training points is initialized using a random seed that is being saved, and the presentation of the results will include the random seeds' values. The parametric study is summarized in table 3.1.

Presentation and Interpretation of the Results

In this experiment, we aim at verifying that a noise variance decay is indeed observed as the number of FS dimensions increases, accordingly with the expectation that the low-dimensional GP captures increasingly more of the variance of the response. As opposed to the visualization of R^2 that was aggregated into a distribution over the training repetitions in the previous experiment, such an approach is not applicable here because we already obtain a distribution for the noise variance in the Bayesian approach. Therefore, the plots that we will be visualizing correspond to a set of individual model training runs in which only the FS dimension is varied, and different repetitions will be displayed in different plots. As discussed above, the number of training observations is also varied as part of the sensitivity study.

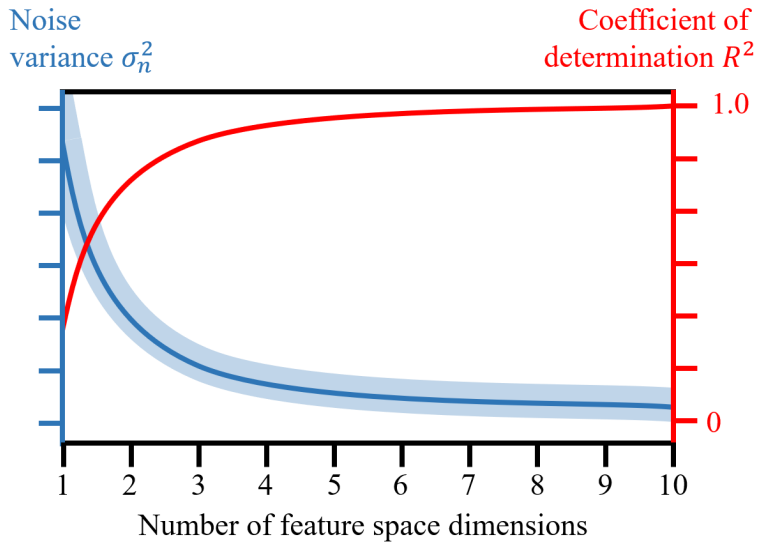


Figure 2.26: Notional illustration of the expected noise variance decay and corresponding increase in R^2

As a result, given a dataset, we will be organizing the noise variance decay plots as a grid, in which the number of training observations increases from top to bottom, and each column corresponds to a different training repetition. A notional plot for the expected behavior of the noise variance decay, which uses the same color scheme as for the results, is shown in fig. 2.26. Within each plot, the x-axis corresponds to the number of FS dimen-

sions. Two separate y-axes are used: the left axis shown in blue corresponds to the noise variance, while the red axis to the right corresponds to R^2. Accordingly, the noise variance is shown as a blue line for the mean of its distribution and a shaded area for its confidence interval, and the evolution of R^2 is shown as a red line.

For results to support the hypothesis, the noise variance should decay until the dataset's underlying FS dimension if the FS dimension is known and sufficiently many training data are available. Otherwise, when the FS dimension is not known or for low amounts of training data, the noise variance decay should point to the FS dimension leading to the best possible approximation as measured by out-of-sample R^2.

2.3.4 Results of Experiment 1.2

As discussed in the previous section, we expect noise variance to decrease as the number of feature space dimensions is increased until the true feature space dimension is reached, at which point the noise variance should remain approximately constant. As noise variance decreases, R^2 should accordingly increase, since the extra dimensions progressively added to the model should enable the predictive model to capture increasingly more of the response's variation. When noise variance becomes constant, we also expect R^2 to remain constant.

In the case of the analytical functions, the feature space was artificially created and its dimension is therefore set. We therefore expect the contrast between the decreasing and constant noise variance regimes to be clear-cut. For engineering datasets on the other hand, the transition between the two regimes may be blurrier as the mappings under consideration have not been designed with a feature space.

We also expect the number of training samples to have an impact on the applicability of the proposed approach. We indeed saw in section 2.2 that models with poor predictive performance are obtained when the number of training samples is excessively limited. In those situations, the proposed approach to the selection of the feature space dimension

is expected to struggle as the reasoning behind it assumes that a meaningful mapping is retrieved.

We will start by discussing the results obtained on the analytical quadratic functions first, then the results obtained using the AS datasets, and finally the aerodynamic datasets (RAE2822 airfoil and CRM wing).

Analytical Functions

The results obtained for the 25-input quadratic function with 1D, 2D, and 5D feature spaces are respectively presented in figs. 2.27 to 2.29. For brevity of the main document body, results for the quadratic functions with respectively 10-, 50-, and 100-dimensional input spaces are presented in appendix A.

The following observations can be made regarding figs. 2.27 to 2.29.

Predictive performance is invariably poor for the lowest numbers of training samples. This is consistent with the results obtained in chapter 2. In turn, this leads to erratic trends for the evolution of the noise variance. This results was expected, the number of training samples simply being too low. The Bayesian approach does provide a way to detect such situations: the spread of the posterior distribution of the noise variance is greater in those cases, which indicates a lack of confidence in the value of the noise variance. When more training samples are available, the value of the noise variance can be determined with greater certainty and the spread of the posterior distribution becomes smaller.

For the highest levels of training samples, we do observe the expected trend. For the 1D feature space (fig. 2.27) using 75 to 125 training samples, noise variance remains fairly constant, except for higher-dimensional feature spaces. For the 2D feature space (fig. 2.28) using 100 and 125 training samples, we do observe the expected drop when transitioning from a 1D to a 2D FS, followed by a constant noise variance as the FS dimension is further increased. The drop in noise variance is more progressive and not so clear-cut in the 5D case (fig. 2.29), even when using 100 or 125 training samples. The results are still satisfactory,

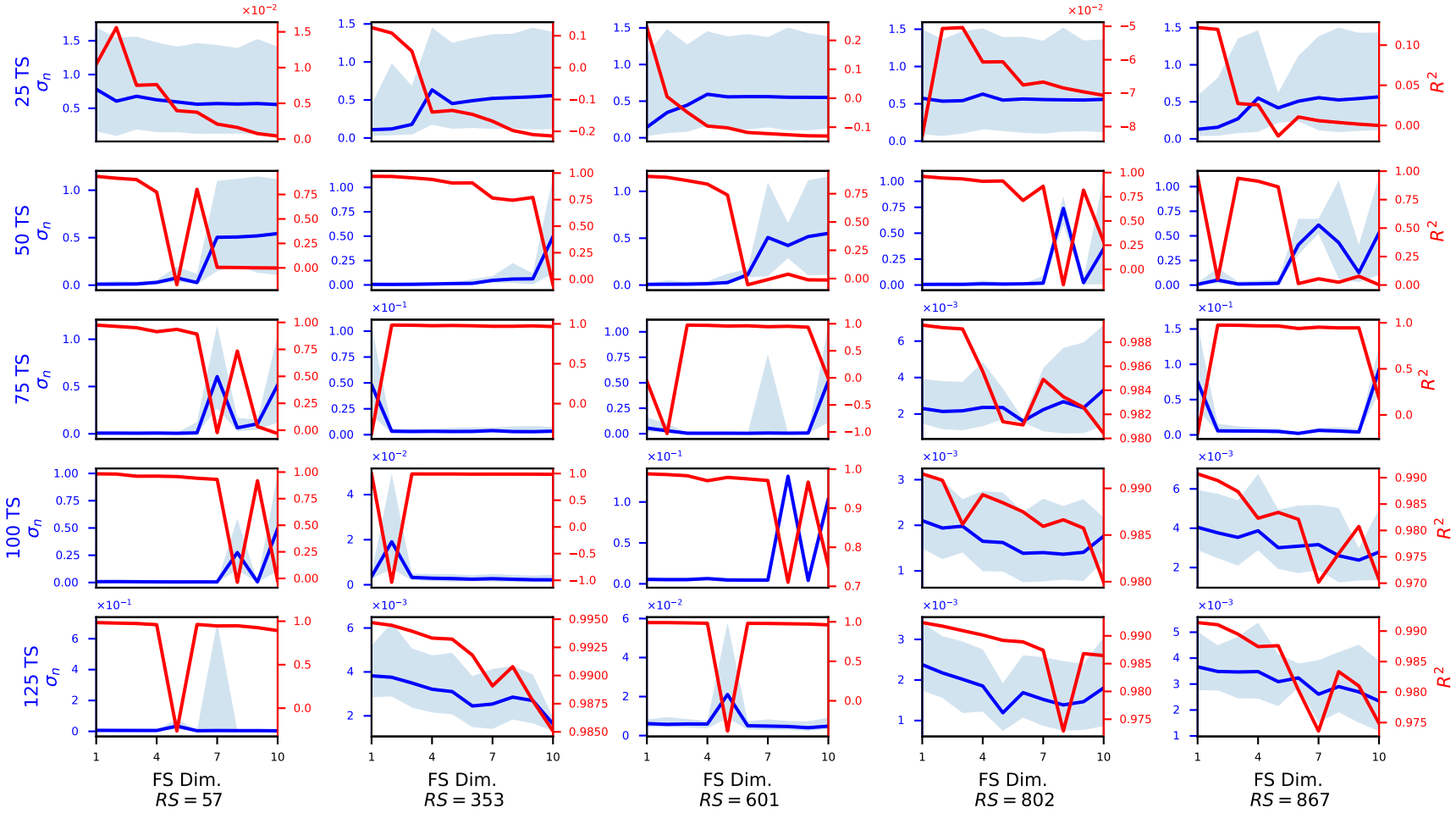


Figure 2.27: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 25 input variables and a 1D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

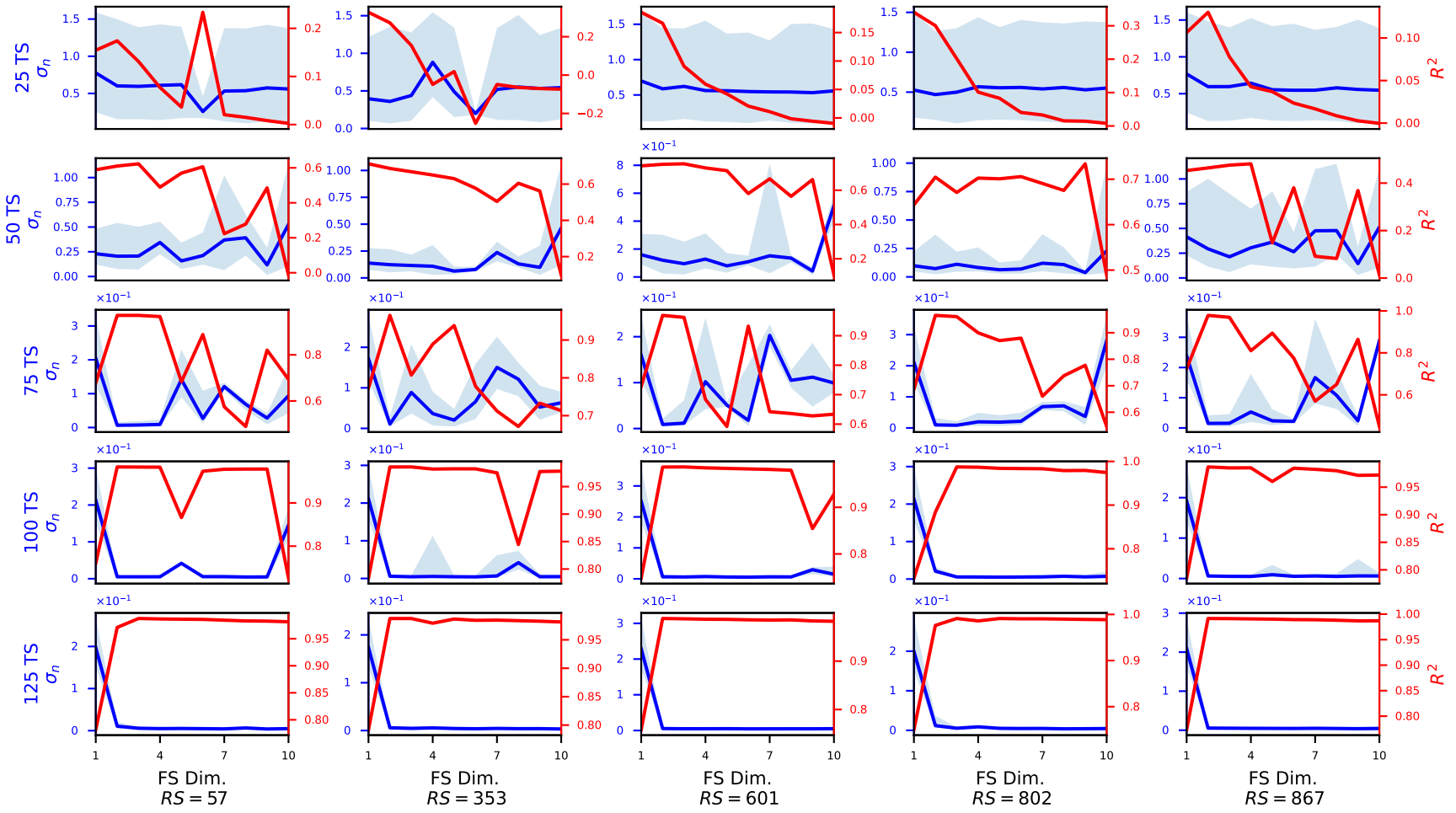


Figure 2.28: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 25 input variables and a 2D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

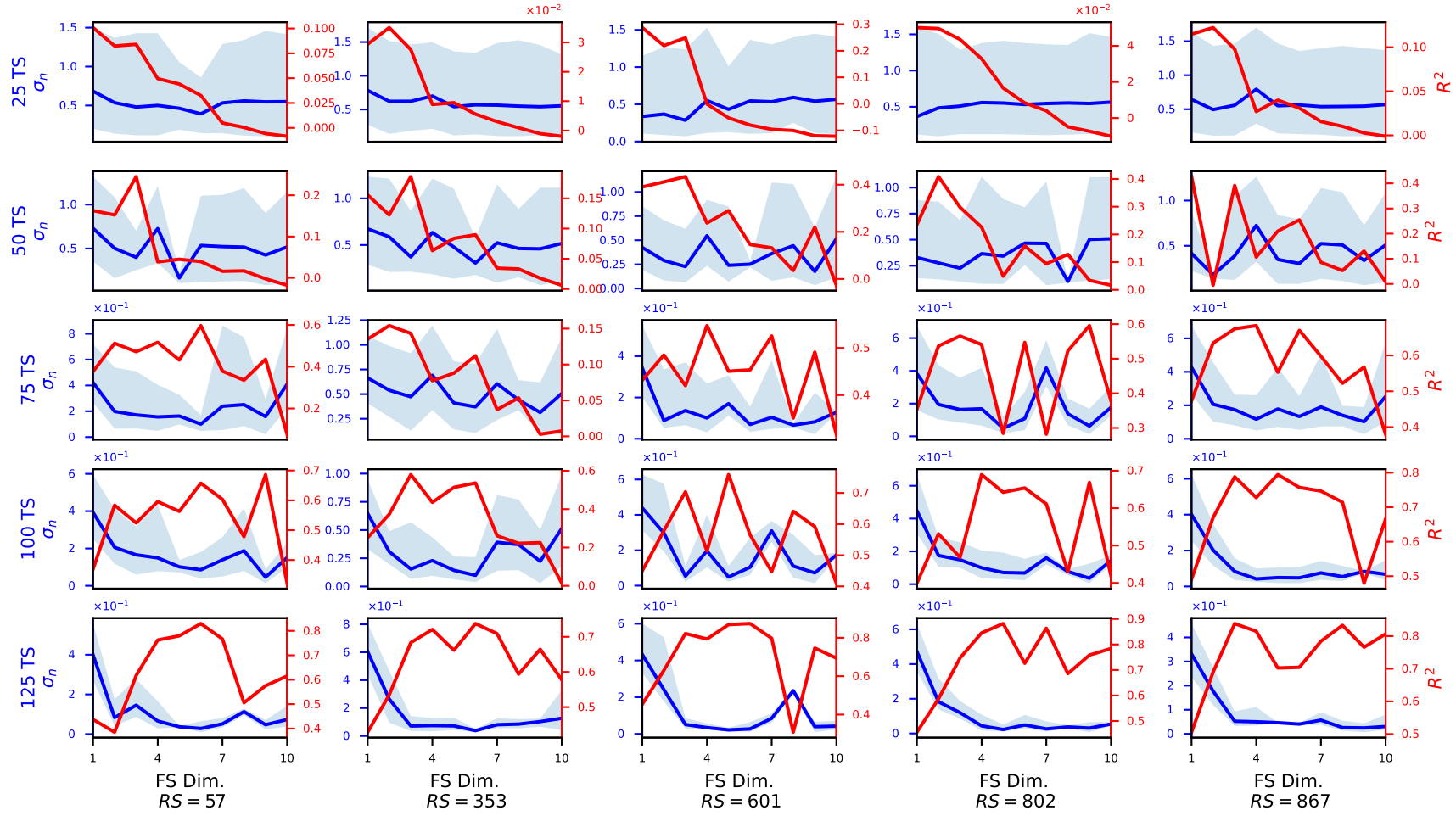


Figure 2.29: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 25 input variables and a 5D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

as the observed noise variance trends would generally lead to the selection of a model with high R^2. Seemingly erratic evolutions of the noise variance and R^2 are however observed, especially when higher-dimensional ($> 5D$) FS are considered.

We demonstrated in section 2.2 that the proposed fully Bayesian approach succeeded in building better predictive models of mappings with high-dimensional inputs with fewer training samples than benchmark methods. However, the proposed approach still struggles when the number of model parameters is disproportionately large compared to the number of training observations. This may happen in two instances: when the number of training observations is extremely low (in the order of the number of input dimensions) or, for larger training sets, when the dimension of the feature space is high. When the number of training observations is low (1 to 2 times the number of input dimensions), R^2 values remain low in this data regime and, as expected, the proposed approach to the selection of the feature space dimension is not applicable. When the number of training observations is intermediate (2 to 3 times the number of input dimensions), the expected decay of the noise variance is only observed until some feature space dimension threshold. The increase in feature space dimension translates into an increase in the number of model parameters: for every new FS dimension added to the model, increasingly more parameters are needed to parameterize the projection matrix onto the FS. In agreement with this explanation, the situation generally improves as the number of training observations becomes relatively high (5 times the number of input dimensions), thus reducing the imbalance between the number of model parameters and the number of training observations. In this situation, we do generally observe the decay of the noise variance throughout the range of considered feature space dimensions.

Similar observations can be made regarding results presented in appendix A for the remaining quadratic functions: 1) surrogate models exhibit poor predictive performance for the lowest numbers of training samples and the proposed approach is therefore challenging to apply in these conditions and 2) the evolution of the noise variance behaves as expected

as long as the number of feature space dimensions is not too high, at which point poor predictive performance is again observed due to the high number of model parameters.

This study on analytical functions confirms the validity of the proposed approach and allows to get a first grip at its behavior on well-behaved functions with known feature space dimensions before moving on to the more realistic and diverse engineering datasets in the next two sections.

Active Subspace Datasets

Results for the ONERA M6 drag, Elliptic PDE y_{short}, and NACA0012 lift datasets are presented in figs. 2.30 to 2.32. Remaining results for the active subspace datasets are included in appendix A.

Results for the ONERA M6 drag dataset in fig. 2.30 are consistent with those obtained for the quadratic functions in the previous section. While poor predictive accuracies are reached for the first two levels of training samples, thus leading to erratic evolution and large uncertainty in the noise variance distribution, the situation starts getting better with an intermediate number of training samples. With 150 training samples, we indeed start seeing the expected noise variance decay trends, although the model struggles for higher-dimensional feature spaces. This is again consistent with what was first observed with the analytical functions and may be attributed to the increasingly large number of model parameters relative to the number of training samples. For the two higher levels of training samples, we observe both the expected trends and obtain models with good predictive accuracy as soon as sufficiently many feature space dimensions are employed. In the case of this dataset, it would seem that a 3-dimensional feature space is sufficient to capture most of the variance in the response. The results for the lift response of the ONERA M6 dataset shown in appendix A seem to indicate that lift is easier to model than drag, as fewer training samples are sufficient in order to obtain the expected trends, and higher R^2 values are obtained early on. In the case of lift, it would seem that a 2D feature space is sufficient

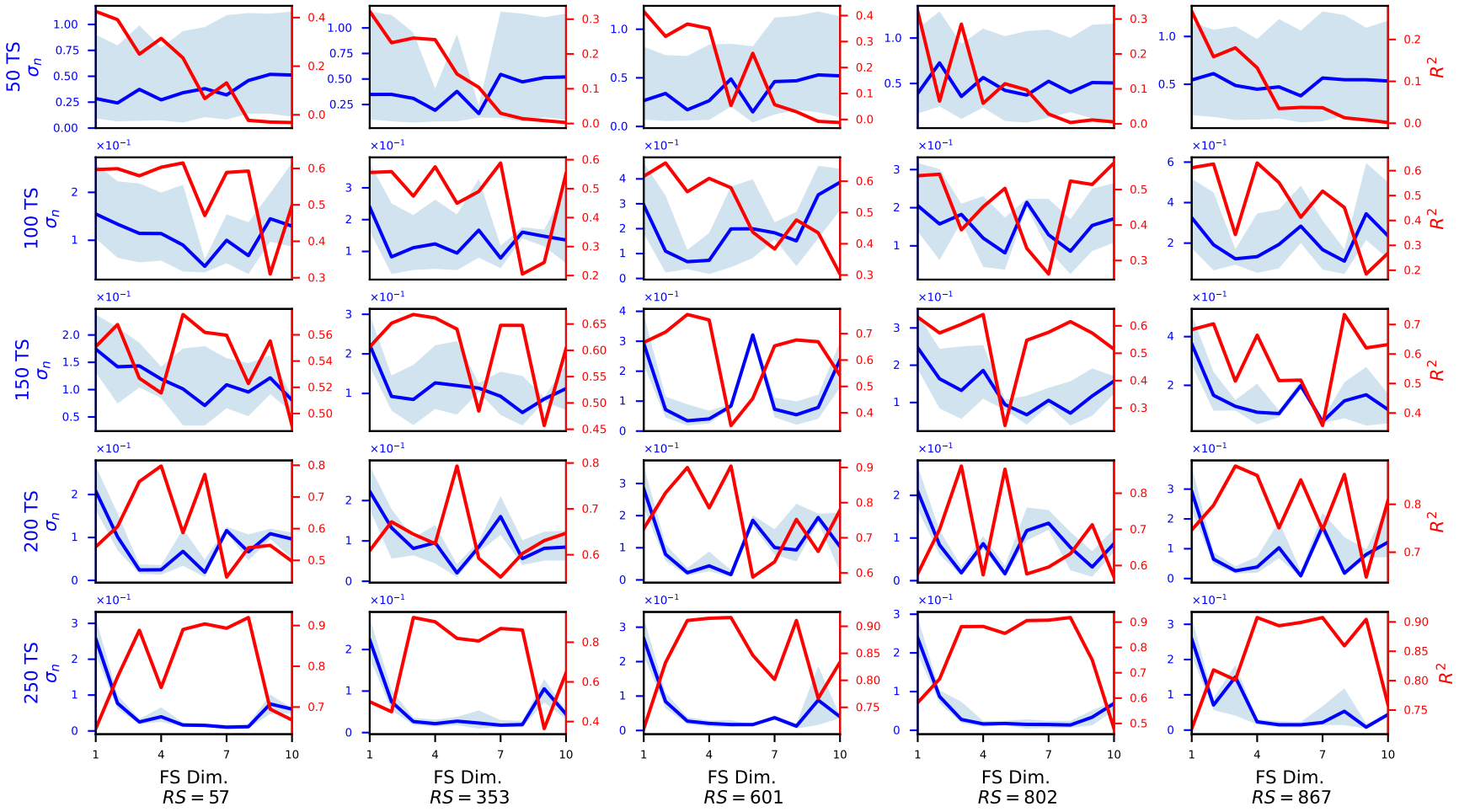


Figure 2.30: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **ONERA M6 drag dataset with 50 input variables**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

instead of a 3D feature space for drag.

The results for the Elliptic PDE y_{short} response shown in fig. 2.31 depict a different situation. It would first seem that the response is generally easier to capture as better predictive performance is reached even with few training samples. Instead of the sharp drop in noise variance observed for the ONERA M6 drag and lift responses, we observe a rather slow decay of limited amplitude that indicates that a 1D feature space is sufficient to satisfactorily model the response. The corresponding R^2 values seem to be consistent with this observation: although they do vary, a closer look at the y-axis scale shows that R^2 actually remain essentially the same throughout the range of considered FS dimensions.

An interesting observation is that the optimal FS dimension may not be the same depending on the number of available training samples. In the case of the NACA0012 lift dataset whose results are presented in fig. 2.32 for example, there does not seem to be a great variation in noise variance or R^2 values when using the intermediate value of 54 training samples. However, when the maximum number of 90 training samples is employed, then the trends rather seem to suggest that a 3-dimensional feature space would be more appropriate. This observation is not surprising: when few data are available, the mechanisms built in the proposed model to counter over-fitting do successfully kick in and favor a simpler model rather than a more complex model that would be too specific to the limited available data. As data becomes more abundant, a more complex model becomes increasingly likely under the probabilistic model.

Observations similar to those made in the previous paragraphs can be made regarding the results for the remaining datasets not shown here in the main text but included in appendix A. The Elliptic PDE y_{long} dataset exhibits good predictive accuracy early on and with a single-dimensional FS. The proposed approach seems to struggle with the HIV $y_{t=3400}$ dataset in which limited noise variance variations seem to indicate a 1D FS, although the many failed training cases may also suggest that more training samples would be necessary in this case. The NACA0012 drag dataset exhibits typical results: expected trends for

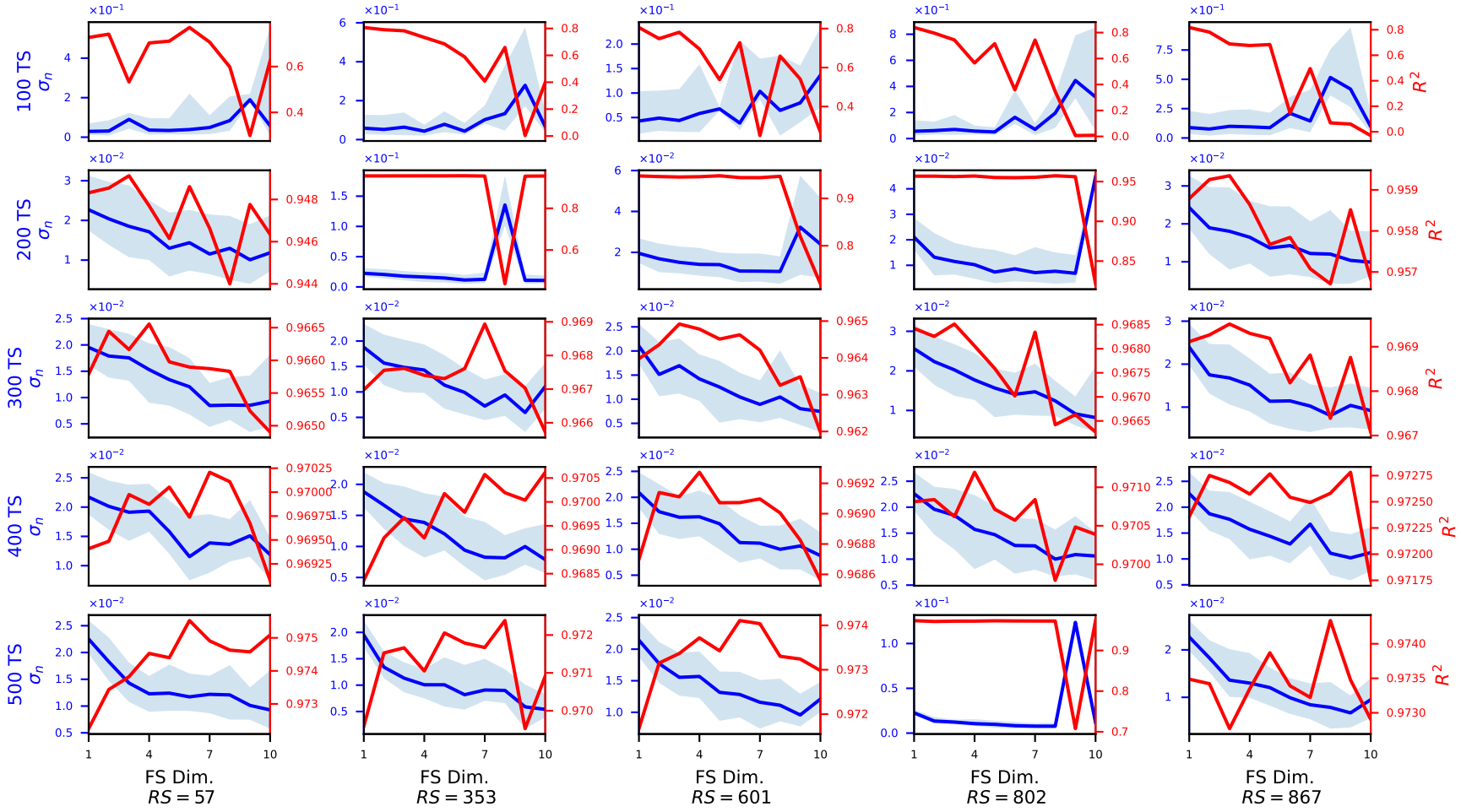


Figure 2.31: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **Elliptic PDE (y_{short}) dataset with 100 input variables**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

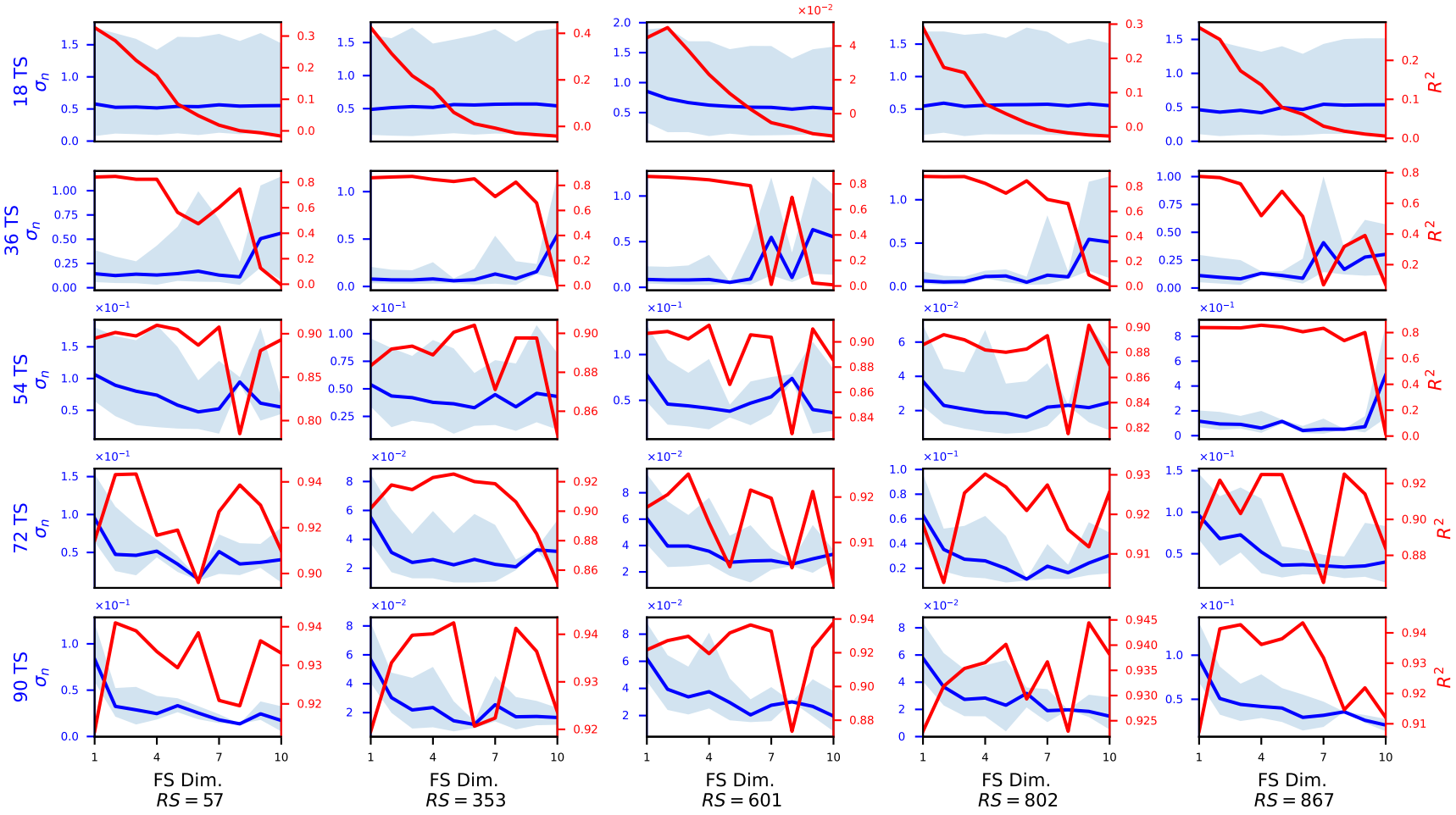


Figure 2.32: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **NACA0012 lift dataset with 18 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

the highest numbers of training samples and struggle with the lowest, as well as when the considered feature space is too high-dimensional.

Aerodynamics Datasets

Results for the subsonic CRM drag and z-moment responses as well as transonic RAE2822 drag response are presented in figs. 2.33 to 2.35. Remaining results for the subsonic CRM, transonic CRM, and transonic RAE2822 datasets are included in appendix A.

Many of models built on the both subsonic and transonic CRM datasets yield a very good predictive accuracy with relatively few training samples and exhibit the expected decaying behavior for the noise variance: lift, x-moment, y-moment, and sideforce. For those, an R^2 greater than 0.99 is reached even at intermediate levels of training samples. This suggests that those responses are less challenging to model. The results for these are relegated to appendix A and we focus here on discussing the seemingly more challenging to model responses.

The results for the subsonic CRM drag dataset are shown in fig. 2.33. They nearly perfectly match expectations. As expected, for the lowest two levels of training samples, models exhibit poor predictive accuracy and we observe relatively large uncertainty in the estimated noise variance value. When using 150 training samples, which is deemed an intermediate level for this example with a 50-dimensional input space, we observe the expected noise variance decay coordinated with an increase in R^2, and the uncertainty in the estimated noise variance decreases. For the highest two levels of training samples, we observe an archetypal version of the expected noise variance decay behavior for all except one training repetition, and the uncertainty in the estimation of the noise variance parameter has all but vanished. When looking at the results obtained with 250 training samples, the noise variance decay suggests a 4-dimensional feature space, which is corroborated by the matching R^2 maximum obtained using this dimension. Models using only 150 training samples suggest a lower-dimensional feature space, which is consistent with results

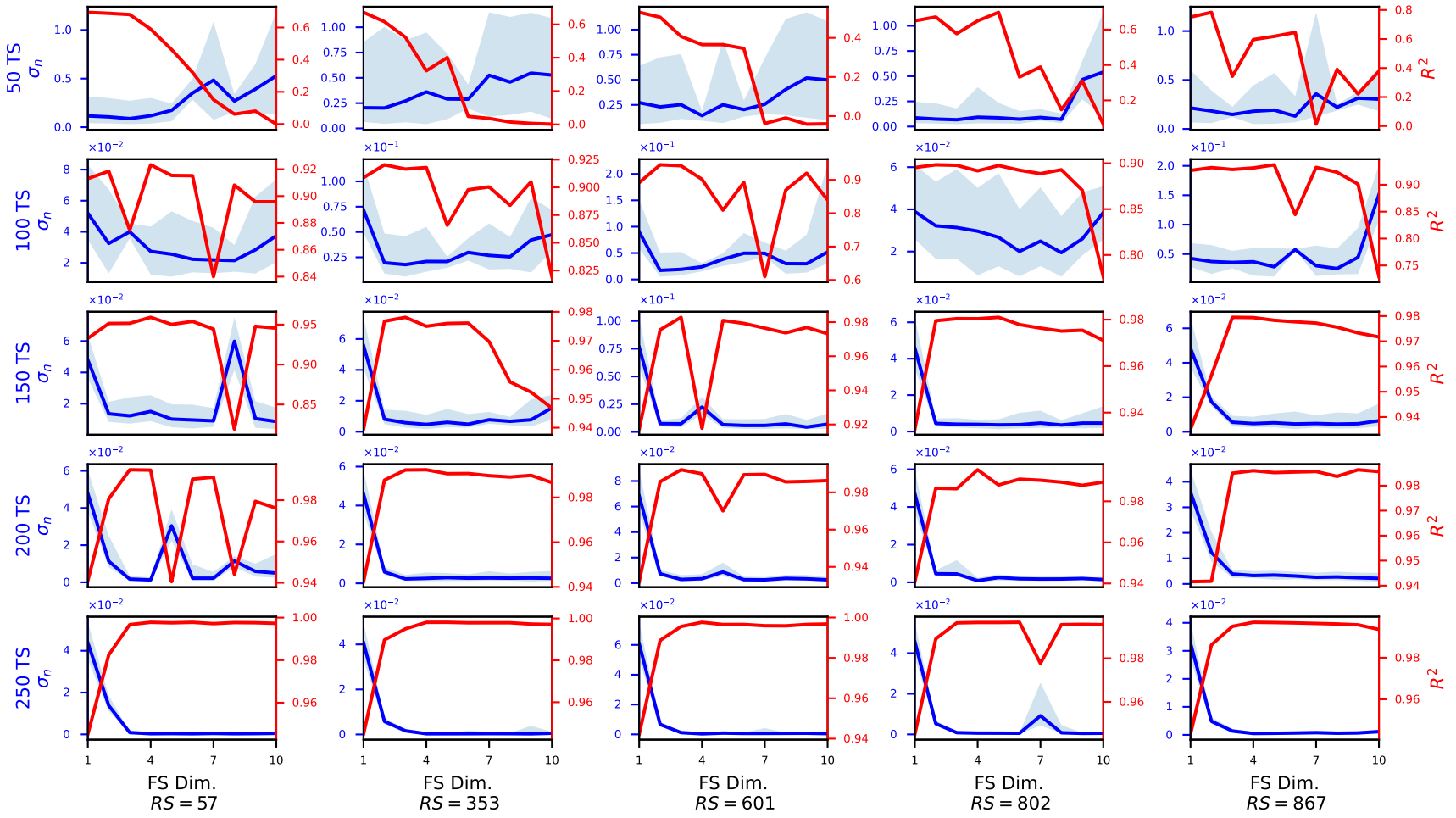


Figure 2.33: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **CRM subsonic drag dataset with 50 input variables**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

obtained previously. This hints at the fact that mechanisms avoiding overfitting are indeed at play here: complex models using many feature space dimensions are not deemed more likely than simpler ones using fewer feature space dimensions.

Results for the subsonic CRM z-moment response presented in fig. 2.34 are similar to those obtained for the drag dataset. The increase in the number of model parameters seems to have a greater impact than on the drag dataset since dives in R^2 accompanied by jumps of the noise variance distributions are observed for some repetitions even with the highest number of training samples. Still, results obtained on this dataset strengthen the argument for the validity of hypothesis 1.2 in the context of realistic engineering datasets.

Results obtained for the transonic CRM datasets and presented in appendix A are similar to those obtained in the subsonic case. The lift, x-moment, y-moment, and sideforce datasets lead to models with high predictive accuracy with relatively few training samples. The drag and z-moment datasets also exhibit the expected noise variance decay behavior, although with less reliably than in the subsonic case. Specifically, more dips in R^2 values are visible for the highest feature space dimensions. This is coherent with the added physics complexity introduced in the transonic regime compared to the subsonic regime. As shocks and high-speed phenomena appear, we indeed expect local modifications of the wing shape to have a more complex and non-linear impact on the aerodynamics of the wing.

Results for the transonic RAE2822 drag dataset shown in fig. 2.35 paint a slightly different picture. Irrespective of the number of training samples within the considered $1\times$ to $5\times$ range, we observe poor predictive accuracies with R^2 values ranging from 0.5 to 0.7. This suggests that this mapping is significantly more challenging to capture and that more training samples are needed to obtain adequate surrogate models. Since surrogate models do not succeed at properly mimicking the true mapping, it is not surprising that the expected noise variance decay is not visible in those results. Results obtained with this dataset again shows that the proposed approach to the feature space dimension selection relies on the assumption that the mapping under consideration needs to be, at least to some

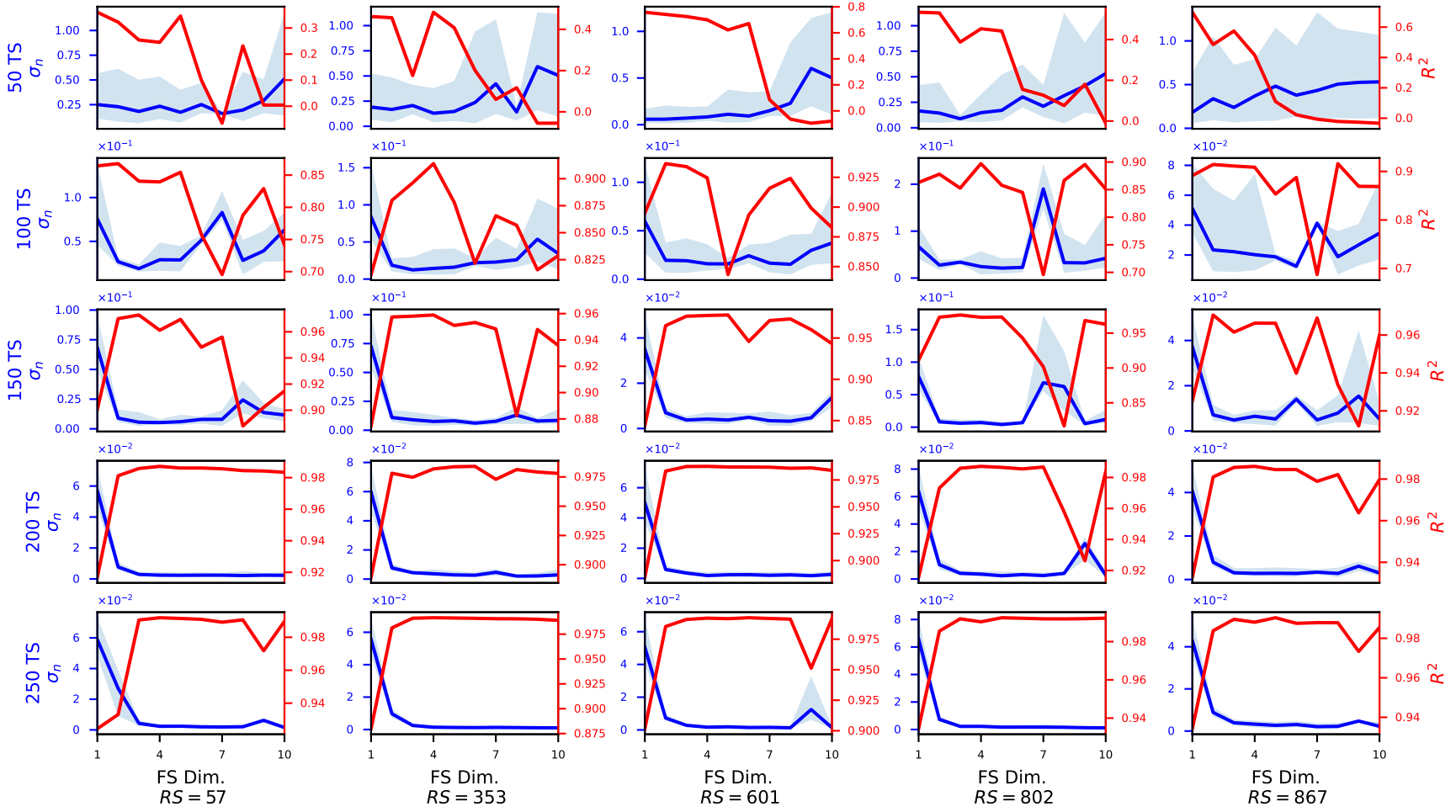


Figure 2.34: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **CRM subsonic z-moment dataset with 50 input variables**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

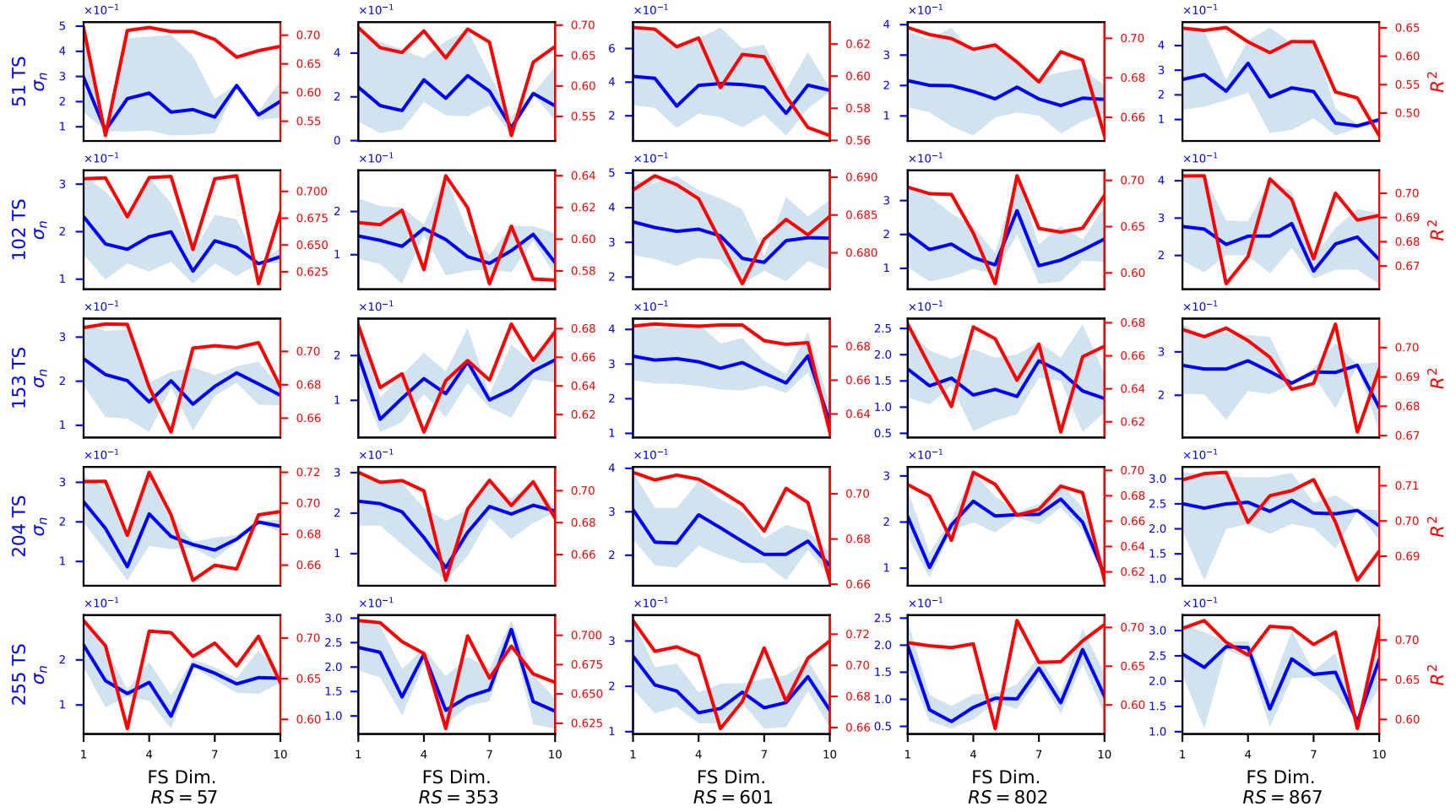


Figure 2.35: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **RAE2822 drag dataset with 51 input variables**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

extent, successfully captured by the predictive model. They also illustrate the fact that the proposed method is not universally applicable although good results were obtained for the subsonic CRM datasets. Results obtained for the remaining RAE2822 datasets (lift and z-moment) are similar to those obtained for the drag dataset.

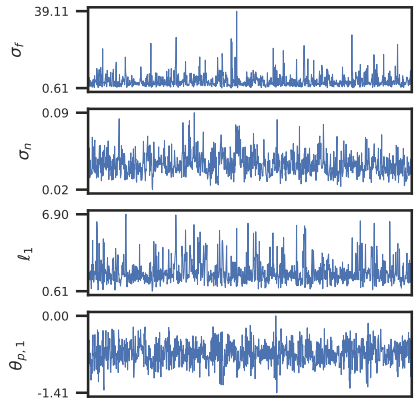
MCMC Chains

In fig. 2.36, we show selected MCMC chains obtained for the CRM drag dataset for different FS dimensions and number of training observations. We observe that the quality of the MCMC chains deteriorates with the number of FS dimensions. This observations was already made previously in fig. 2.10 in the context of experiment 1.1. However, there is no notable difference due to the number of training observations: for both 100 and 250 training samples, a similar deterioration of the MCMC chains is visible.

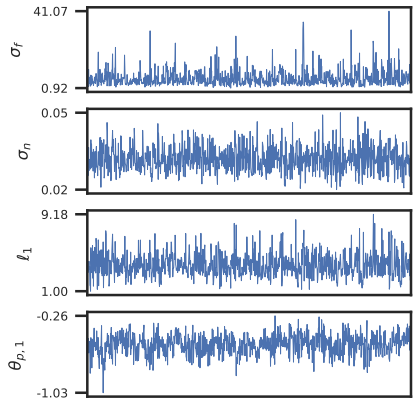
These selected training runs correspond to the results shown in fig. 2.33, in the rightmost column (repetition with random seed $RS = 867$), second and fifth rows (respectively 100 and 250 training observations). There, we observed the expected decaying behavior with 250 training samples while the noise variance increased for high FS dimensions when only 100 training observations were used. Since the deterioration of the MCMC chain is similar in both cases for a 10D FS, this suggests that the absence of the decaying behavior is not due to an MCMC convergence problem. As discussed previously, it rather appears that this is due to the imbalance between very high-dimensional model parameters and sparse training observations, as we observed that the behavior of the noise variance usually improves as the number of training observations is increased.

2.3.5 Conclusion

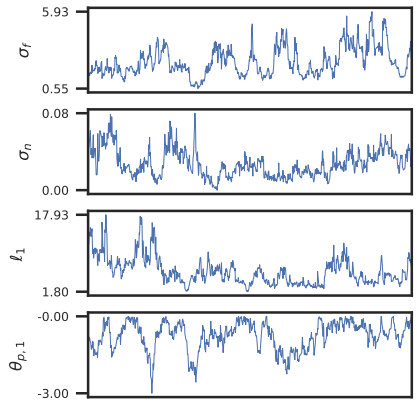
For the most part, results presented in the previous section matched the expected behavior for the noise variance decay. In the case of the quadratic functions, we could see that, if sufficiently many samples were available, using the noise variance decay to select the



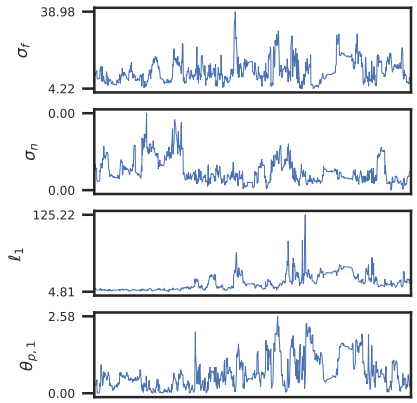
(a) 1D FS / 100 training samples



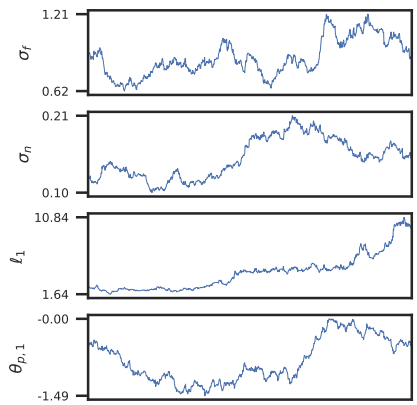
(b) 1D FS / 250 training samples



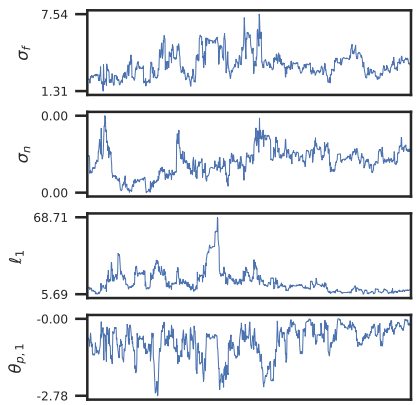
(c) 5D FS / 100 training samples



(d) 5D FS / 250 training samples



(e) 10D FS / 100 training samples



(f) 10D FS / 250 training samples

Figure 2.36: Comparison of the MCMC chains for selected model parameters obtained for the CRM subsonic drag dataset ($RS = 867$) for different FS dimensions (top: 1D, middle: 5D, bottom: 10D) and number of training samples (left: 100, right: 250).

dimension selection did lead to the right decision. For engineering datasets, under the same condition regarding the number of training samples, the noise variance decay was consistent with the out-of-sample predictive performance of the model and would therefore lead to a correct feature space dimension selection. Some responses were however notably harder to predict, thus leading to the proposed approach struggling to assist in the dimension selection process. For such datasets, the complexity of the response may call for additional training samples or a surrogate model form other than a ridge function.

We make the following general observations based on the results discussed in the previous section:

- the expected decay behavior is only observed when the number of training samples is sufficiently high,
- in many cases, even for the highest number of training samples within the considered range, it is only observed for the lower half of considered feature space dimensions,
- the amplitude and spread of the noise variance distribution can be used as indicators of poor performing models that would require additional training samples,
- the applicability of the proposed approach is case-dependent and relies on the assumption that the underlying mapping is to some extent captured by the surrogate model.

It is important to keep that last point in mind as to not overestimate the capabilities of the proposed approach. An adequate surrogate model generally becomes increasingly challenging to obtain as the number of model parameters becomes higher relative to the number of training samples [143]. This may occur 1) when the number of training samples is low in absolute terms, or 2) when the dimension of the feature space is high, leading to a large projection matrix, and thus a large number of model parameters.

The sensitivity to the number of training samples and feature space dimensions is also dataset-dependent, as simpler mappings may be recovered using fewer samples than more

complex mappings. In particular, different predictive performance may be observed for different output quantities of the same physical phenomenon, such as lift and drag for an airfoil or wing, as each individual mapping has unique properties. In the extreme, we observe near-perfect predictive performance for the simplest mappings irrespective of the feature space dimension even for the lowest levels of training samples.

Conclusion on Hypothesis 1.2

Under the condition that the number of training samples is sufficiently high relative to 1) the complexity of the mapping under consideration, and 2) the number of feature space dimensions, then an appropriate FS dimension is selected when the noise variance distribution is used as selection metric.

The current section addressed the first problem raised in section 2.3.1, in the next section we will focus on the second problem: reducing the cumulative training time incurred when performing the dimension sweep needed to determine the FS dimension.

2.4 Sequential Approach to Building the Feature Space

2.4.1 Background and Research Objective

We previously identified two shortcomings of the current approaches to the selection of the FS dimension:

1. the lack of a metric suitable to models trained using Bayesian inference and in the low-data regime
2. the need to train a model for each considered FS dimension

Using the noise variance as a selection metric addressed the first problem. The current section aims at tackling the second issue, namely the need to perform a sweep of all FS dimension under consideration in order to choose the appropriate dimension.

We have seen in chapter 2 that training models relying on approximations by ridge functions, whether the proposed fully Bayesian approach or an MLE-based approach is employed, incurs a relatively large computational cost. Therefore, repeating the training of such models for every considered FS dimension may become impractical under organizational constraints such as limited computational budgets or surrogate model development times. Existing approaches restart training from scratch for every new FS dimension [112, 91] and do not make use of the directions found using previous training rounds.

Gap 1.3

The process of selecting an appropriate FS dimension requires repeating the training of the surrogate model for each considered FS dimension, considerably increasing the computational cost of obtaining the surrogate model.

As the FS dimension increases, the number of model parameters that need to be optimized or inferred increases accordingly. In practice, this means that training time increases as dimension increases. One way to reduce cumulative training time would be to reduce the dimension of the optimization or inference problem that needs to be solved at every step. If previous directions were reused and only a single new direction was sought every time the FS dimension is incremented, then the number of model parameters to determine at every step would be roughly constant instead of increasing, hopefully leading to an overall decrease of the training time accumulated over all considered FS dimensions.

Reusing the previously found directions is however not straightforward in the context of the fully Bayesian approach. If point values for the directions were determined at every step, we could simply fix those directions and only seek an additional direction at every step, with the added constraint that the new direction be orthogonal to the previous ones. However, in the context of the fully Bayesian approach, we obtain a distribution for the FS directions instead of a point value. While it is unclear which mechanism could be used to reuse the distribution when determining the distribution for the next direction, a point

value could be extracted. But this would defeat the purpose of the fully Bayesian approach, and raise the problem of the significance of the resulting parameters' distributions. In this section, we therefore investigate a way to extend the proposed fully Bayesian approach to reuse information with the goal to reduce the cumulative training time of the dimension sweep needed to select the FS dimension.

Research Question 1.3

How can we enable the reuse of information from previously trained models in order to speed up the process used to select an appropriate FS dimension?

2.4.2 Proposed Method

In this section, we will be investigating promising means by which information could be reused to speed up the FS dimension sweep required in the proposed approach. We will start by introducing a first distinction between *joint* and *separate* methods depending on whether the determination of the FS and the low-dimensional regression are carried out simultaneously. We then propose to make a second distinction between *all-at-once* and *sequential* approaches depending on whether all directions of the FS or only a subset of them are sought for a given FS dimension. The theoretical limitations of sequential approaches will be discussed, along with the benefits they are expected to bring. These categorizations will then allow us to consider a list of alternative approaches to solve the problem-at-hand. Finally, a discussion will allow us to down-select the most promising approaches and form hypothesis 1.3.

We first propose to make a distinction between approaches in which the determination of the FS directions and the low-dimensional regression are treated as a single large problem – we will refer to these as *joint* approaches – and approaches in which they are treated as two smaller problems – we will refer to these as *separate* approaches. In a joint approach, the determination of the FS and the low-dimensional regression are treated as a

single problem. In this sense, the fully Bayesian approach introduced in chapter 2 is a joint approach, as both the projection matrix $\mathbf{W}$ and the link function g are inferred at the same time using MCMC. Other examples of joint approaches from the literature include [91, 112, 131]. In a separate approach, two different problems are solved: first determine the FS, and then use this FS to determine the link function. This is for example the approach proposed in [150] or when using the AS method to find a FS for regression [110]. Separate approaches provide flexibility: while a class of methods, such as the gradient-based AS method may be used to determine a suitable FS, a completely different class of methods, such as any supervised learning method, may be used to model the low-dimensional mapping. Joint approaches provide a more straightforward way by which the supervised learning problem can be addressed for mappings with high-dimensional input spaces instead of having to piece together different methods. Moreover, they may exhibit desirable features: for example, the proposed joint fully Bayesian approach does not require gradients, while a separate approach leveraging the AS method would.

When it comes to the learning mechanism, we restrict ourselves to the MLE and Bayesian frameworks. In a joint approach, both the feature space determination and the low-dimensional would follow the same learning mechanism. In a separate approach, each step may be carried out using a different learning mechanism, although some combinations may bring additional difficulties. We will discuss these difficulties when analyzing alternative approaches in the next few paragraphs.

The second distinction we propose to make concerns the determination of the directions that make up the FS. If we denote by m the dimension of the FS, as we perform the dimension sweep to select the FS dimension, we successively set $m = 1$, then $m = 2$, until reaching $m = m_{max}$ where m_{max} is the maximum considered FS dimension. In the fully Bayesian approach proposed in chapter 2 and reused in the study in section 2.3, all m directions of the FS are sought every time the FS dimension is varied. As a consequence, as m is increased, the number of model parameters that need to be optimized or inferred

keeps increasing. We propose to refer to these approaches as *all-at-once* since all directions of the FS are found together. On the other hand, we could retain $k, k < m$ dimensions such that only $m - k$ directions need to be found at the m^{th} step. If we choose $k = m - 1$, then only single direction needs to be determined at every step, and the number of model parameters to infer or optimize at every step would remain roughly the same. Therefore, we would expect cumulative training time to increase slower than when all directions are sought for every new FS dimension. We propose to refer to those approaches as *sequential* since directions of the FS are successively found.

While we expect a sequential approach to reduce cumulative training time, it is also important to understand some of the theoretical shortcomings of such an approach. Let us recall the variance decomposition introduced in eq. (2.53):

$$\mathrm{V}[f] = \sum_{i=1}^{d} \mathrm{V}[f_i] + \sum_{i<j} \mathrm{V}[f_{i,j}] + \cdots + \mathrm{V}[f_{1,2,\ldots,d}] \tag{2.54}$$

Let us assume that we are optimizing directions in an MLE-based approach. A similar reasoning would hold in the case of Bayesian inference. Using this decomposition, let us contrast what happens when we optimize for $m = 1$ and then $m = 2$ first in the case of an all-at-once approach, and then in the case of a sequential approach.

In an all-at-once approach, when $m = 1$, the problem can be thought of as selecting the direction $\mathbf{d_1^1}$ in input space such as to maximize $\mathrm{V}[f_{\mathbf{d_1^1}}]$, where $f_{\mathbf{d_1^1}}$ would be the component function in that direction. When $m = 2$, the problem becomes selecting two orthogonal directions $\mathbf{d_1^2}$ and $\mathbf{d_2^2}$ in the input space such as to maximize the sum $(\mathrm{V}[f_{\mathbf{d_1^2}}] + \mathrm{V}[f_{\mathbf{d_2^2}}] + \mathrm{V}[f_{\mathbf{d_1^2},\mathbf{d_2^2}}])$ where $f_{\mathbf{d_1^2}}$, $f_{\mathbf{d_2^2}}$, and $f_{\mathbf{d_1^2},\mathbf{d_2^2}}$ are the component functions corresponding to these two directions. In a situation where there exist two input space directions such that 1) the variance $\mathrm{V}[f_{\mathbf{d_1^2},\mathbf{d_2^2}}]$ of the interactions is greater than the variance $\mathrm{V}[f_{\mathbf{d_1^1}}]$ of the main effect function picked when $m = 1$, and 2) the direction $\mathbf{d_1^1}$ is orthogonal to the plane made up of the directions $\mathbf{d_1^2}, \mathbf{d_2^2}$, then the 1-dimensional FS found in the first step may be discarded and the 2-dimensional FS direction $\mathbf{d_1^1}$ may be made up of two completely new directions. A function with such Sobol' indices can easily be constructed, starting from the

decomposition of an arbitrary L^2 function, computing its Sobol' decomposition, retaining only the relevant terms, and adding a multiplicative factors such as to satisfy the variance inequalities discussed above.

In the case of the sequential approach, the objective when $m = 1$ is the same. However, when $m = 2$, the first direction is fixed and the objective may be thought of as finding the direction $\mathbf{d_2^2}$ that maximizes the sum $(\mathrm{V}[f_{\mathbf{d}_1^1}] + \mathrm{V}[f_{\mathbf{d}_2^2}] + \mathrm{V}[f_{\mathbf{d}_1^1,\mathbf{d}_2^2}])$. Notice how this problem is different from the problem solved at $m = 2$ in the all-at-once approach. In the latter, two directions are free to be varied whereas in the former, only the second direction is not fixed. If we go back to the example discussed in the previous paragraph, we see that the outcome would be different in the sequential case: it would not be possible to select two completely new directions even though they would lead to capturing more variance, and accordingly lead to a more accurate surrogate, since we chose to retain the first dimension. However, the optimization problem that needs to be solved is simpler: instead of having to determine two orthogonal directions, only a single direction needs to be found.

Now, if we summarize the last few paragraphs, we can assemble a list of alternative approaches for successively training models with different FS dimensions. The list of alternatives is presented in table 2.5. The next step will be to discuss the coherence and feasibility of each alternative and down-select the most promising subset.

Alternative 1 was the benchmark used in chapter 2 [112, 91]. Alternative 2 is the proposed fully Bayesian approach. Alternative 3 is a variation of 1 in which each dimension would be selected at once. Now, alternative 4 would be a variation of 2 in which each dimension would be selected at once. However, it is unclear how an FS distribution determined using Bayesian inference could be fixed in order to independently carry out inference on the next direction. We therefore deem this alternative infeasible. Alternative 5 is redundant with alternative 1. Alternative 6 uses directions found using MLE to create a Bayesian link function. Alternative 7 suffers from the same limitations as alternative 4 and is therefore deemed unfeasible. Alternative 8 is redundant with alternative 2. Alternative 9

Table 2.5: Complete List of Alternatives

	Overall Approach	Directions Finding	Feature Space Determination	Low-Dimensional Regression
1	Joint	All-at-Once	MLE	
2	Joint	All-at-Once	Bayesian	
3	Joint	Sequential	MLE	
4	Joint	Sequential	Bayesian	
5	Separate	All-at-Once	MLE	MLE
6	Separate	All-at-Once	MLE	Bayesian
7	Separate	All-at-Once	Bayesian	MLE
8	Separate	All-at-Once	Bayesian	Bayesian
9	Separate	Sequential	MLE	MLE
10	Separate	Sequential	MLE	Bayesian
11	Separate	Sequential	Bayesian	MLE
12	Separate	Sequential	Bayesian	Bayesian

is redundant with alternative 3. Alternative 10 is feasible, because reusing point estimations of the FS is possible, and a practical implementation is discussed thereafter. Alternative 11 is deemed unfeasible for the same reason as alternatives 4 and 7. Finally, alternative 12 is somewhat redundant with alternative 4.

This leaves us with alternatives 1, 2, 3, 6, and 10. Alternatives 1 and 2 were studied previously, and we are aiming to find an alternative method that leads to a shorter cumulative training time. Alternatives 3 and 10 would be good candidates: they are both feasible approaches that adopt a sequential strategy. In practice, carrying out training using a method based on alternative 10 would amount to 1) train a model using a method based on alternative 3, 2) retain the determined FS, and 3) discard the low-dimensional mapping found using MLE and replace it with a Bayesian GP. We do not expect replacing the MLE-based low-dimensional regression with a Bayesian GP to have an adverse effect on predictive accuracy. On the contrary, if anything, we would expect a Bayesian approach to yield a higher predictive accuracy. Therefore, we select alternative 10 over alternative 3 as a candidate approach. This leaves us with alternative 6, which is not a sequential approach and we therefore do not expect it to reduce cumulative training time. However, because it may

be thought of as an intermediary approach between alternatives 2 and 10, we choose to include it in the next study. By referring to it as an intermediary approach, we mean that the distinction between alternatives 2 and 6 is that directions-finding was switched from a Bayesian approach to an MLE approach, while keeping a Bayesian approach for low-dimensional regression. The distinction between alternatives 6 and 10 is that an all-at-once determination of the FS directions using MLE was swapped for a sequential determination of the FS directions, but still using MLE. It might therefore help us determine whether differences in behavior are to be explained by the switch 1) from a Bayesian to an MLE-based approach, or 2) from an all-at-once to a sequential approach.

In summary, the approach we deem most promising to achieve the desired cumulative training time reduction seeks FS directions sequentially using MLE and handles the low-dimensional regression separately using a Bayesian GP. Let us now discuss the specifics of this approach, particularly when it comes to ensuring orthogonality of the sequentially determined directions. To achieve this, we proceed as follows. At $m = 1$, the first dimension is unconstrained. Starting with $m = 2$, we start by building a basis for the orthogonal complement of the current FS. In practice, this is done using the Gram-Schmidt orthogonalization process by successively subtracting the projections of the previous directions starting with an initially random matrix. Then, the problem of finding the m^{th} direction is equivalent to seeking a $(d - m)$-dimensional direction in the orthogonal complement, which is straightforward once a basis exists. The previous discussion leads to hypothesis 1.3 shown below.

Hypothesis 1.3

The proposed sequential training method reduces cumulative training time while leading to the selection of a similar feature space dimension because the MLE-based approach used for determining directions allows reusing information from previously trained models while the noise variance provided by the Bayesian GP used for the link function still provides a sufficiently good approximation of the variance discarded when reducing the input space dimension.

2.4.3 Setup of Experiment 1.3

Need for Experiment 1.3

Motivated by the reduction of the total time needed to train all the surrogate models used to select the FS dimension, we proposed an alternate approach in which information from previously trained models is reused. In order to make the reuse of information possible, we have proposed compromises that we know may have an impact on the resulting surrogate models, and possibly on the process used to select an appropriate FS dimension. We made two modifications to the original approach in order to enable the reuse of information: 1) we adopted an MLE-based method for directions-finding instead of a fully Bayesian approach, and 2) we proposed to fix previously found directions. While they enable the reuse of information, and therefore are expected to reduce cumulative training time, we have previously noted that MLE-based approaches can have lower accuracy and that fixing directions may be suboptimal. In this experiment, we need to check that despite these shortcomings, we will still be able to observe the decay of the noise variance that we use as a basis for selecting an appropriate FS dimension. In addition, we also need to track and compare training time to ensure that the sequential approach is indeed faster.

Experiment Design

The experimental approach is very similar to experiment 1.2: we also need to be keeping track of the evolution of the noise variance as a function of the number of FS dimensions. However, this time, the goal will be to ensure that the decay of the noise variance observed using the alternate – and hopefully faster – methods discussed in the previous sections are similar to the decay observed with the original fully Bayesian method in which all FS directions are sought all-at-once, such that they lead to the selection of the same FS dimension.

We will be using the same datasets and the same metrics as in experiment 1.2. In addition to the results obtained in experiment 1.2 for the original approach, we will add results for the two alternative training approaches discussed previously: alternative 6, the separate approach where the FS is found using an all-at-once MLE approach and the low-dimensional regression uses a Bayesian GP, and alternative 10, the proposed sequential approach. We will be comparing 1) the noise variance decays across methods in order to ensure that they would lead to the same decision regarding the FS dimension, and 2) the cumulative training time in order to ensure that time savings are effectively realized. In addition, we will be keeping an eye on R^2 in order to estimate the predictive accuracy of the sequential approach. However, the predictive accuracy of the alternate method is not our main concern since the purpose of the sequential approach is to assist the selection of the FS dimension, not to make predictions. Once the FS dimension has been determined, the fully Bayesian approach whose predictive accuracy has been demonstrated in chapter 2 can be applied.

As in previous experiments, we do not focus on a particular dataset but rather aim at assessing it on a number of datasets that are representative of analyses carried out in engineering design. The datasets used as a basis for comparison are the same as in experiment 1.2. The source code used to generate the results presented thereafter has been made pub-

licly available[3].

Sensitivity Study

As in experiment 1.2, the FS dimension is varied between 1 and 10 for each considered dataset. As it is expected to impact the metrics of interest, the number of training observations is varied, and training is repeated five times to account for the particular choice of training observations, using a random seed to ensure reproducibility.

We define ranges for numerical parameters and generate a full-factorial DOE to study the variations of all parameters (see table 2.6).

Table 2.6: Experiment 1.3 – Summary of the parameters varied in the parametric study

Parameter	**Parameter Values**
Datasets	see table 2.3
FS dimension	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Number of Training Samples (times number of inputs)	[1, 2, 3, 4, 5]
Number of Training Repetitions	5 repetitions with fixed random seeds
Alternative Methods (see table 2.5)	[2, 6, 10]

Presentation and Interpretation of the Results

We will use the same representation of the results as in experiment 1.2, except that we will be adding the noise variance decay corresponding to the two alternative methods considered here: the sequential approach and the approach combining joint MLE for directions-finding and a Bayesian GP for the low-dimensional regression. Accordingly, we will also visualize grid plots in which the number of training samples increases from top to bottom, and each column corresponds to a different training repetition. Within each plot, we will be visualizing the evolution of both the noise variance (solid lines) and R^2 (dotted lines) as a function

[3]https://gitlab.com/raphaelgautier/thesis_experiments_part2

of the number of FS dimensions. The lines corresponding to the original, joint Bayesian method, are shown in red. The other two methods, joint MLE + B-GP and sequential, are respectively shown in green and blue. The observation of the green plot corresponding to the intermediate method allows to pinpoint whether behavior discrepancies originate from adopting an MLE-based approach or whether it comes from the sequential determination of the FS directions in which previous directions are retained. In these plots, the lines are surrounded with a shaded area, corresponding to the 95% confidence interval of the noise variance's posterior distribution, as its value is inferred using MCMC.

For each dataset, we will also be visualizing the evolution of the training time as a function of the number of training samples in a separate graph. The colors match the ones used in the noise variance decay grid plots. In this plot, the uncertainty in the training time originates from the training repetitions.

In order for the results to support the hypothesis, two conditions should be met. First, the decay of the noise variance obtained with the sequential approach (in blue) should match the decay observed using the original, fully Bayesian approach (in red). Specifically, both lines should lead to the selection of the same FS dimension, where the FS dimension is selected based on the number of dimensions at which a plateau is reached for the noise variance. Second, the training time of the sequential approach should be shorter than the original fully Bayesian approach.

Given a dataset, a number of training observations, and a repetition seed, we visualize the evolution of σ_n^2 and R^2 as a function of the FS dimension for all three training methods under study. For results to support the hypothesis, the noise variance should decay observed using the proposed sequential approach should match the decay observed with the original approach, thus leading to the selection of the same FS dimension.

2.4.4 Results of Experiment 1.3

Analytical Functions

Results for three of the analytical functions are presented in figs. 2.37, 2.39 and 2.41. These are the same functions for which results were presented when studying the applicability of the noise variance for dimension selection in section 2.3. Previous results are here supplemented by results obtained when applying the proposed sequential training approach as well as the joint MLE-based approach paired with a B-GP for the low-dimensional regression. Results for the remaining analytical functions are included in appendix B.

Results for the quadratic function with 25 inputs and a 1D feature space are presented in fig. 2.37. When the number of training samples is relatively low, the sequential approach appears to suffer less than the joint approach for the highest-dimensional FS. While models with poor predictive performance are obtained for FS with dimensions above 5 when using only 50 training samples when using the joint Bayesian approach, the sequential approach yields predictive models with consistently satisfactory performance. This would suggest that, in this case, optimization suffers less than Bayesian inference from the increase in dimension of the model parameters. When the number of training samples is higher however, we observe in multiple instances a decaying behavior that is not what we expect in the case of the 1D FS. Instead, the noise variance should remain approximately constant over the range of considered FS dimensions since we know the underlying FS to be one-dimensional. Cases using 100 and 125 training samples for repetitions $RS = 353$, $RS = 802$, and $RS = 867$ present a sharp drop in noise variance for FS dimensions ranging from 1 to 3, which would lead to the selection of a 3-dimensional FS when following the proposed approach. The over-estimation of the FS dimension may be deemed problematic as it lessens the impact of dimension reduction. However, in the context of this dataset, it does not appear to hinder the predictive capabilities of the model, as shown by the evolution of R^2. As discussed previously, we expect the sequential approach to enable

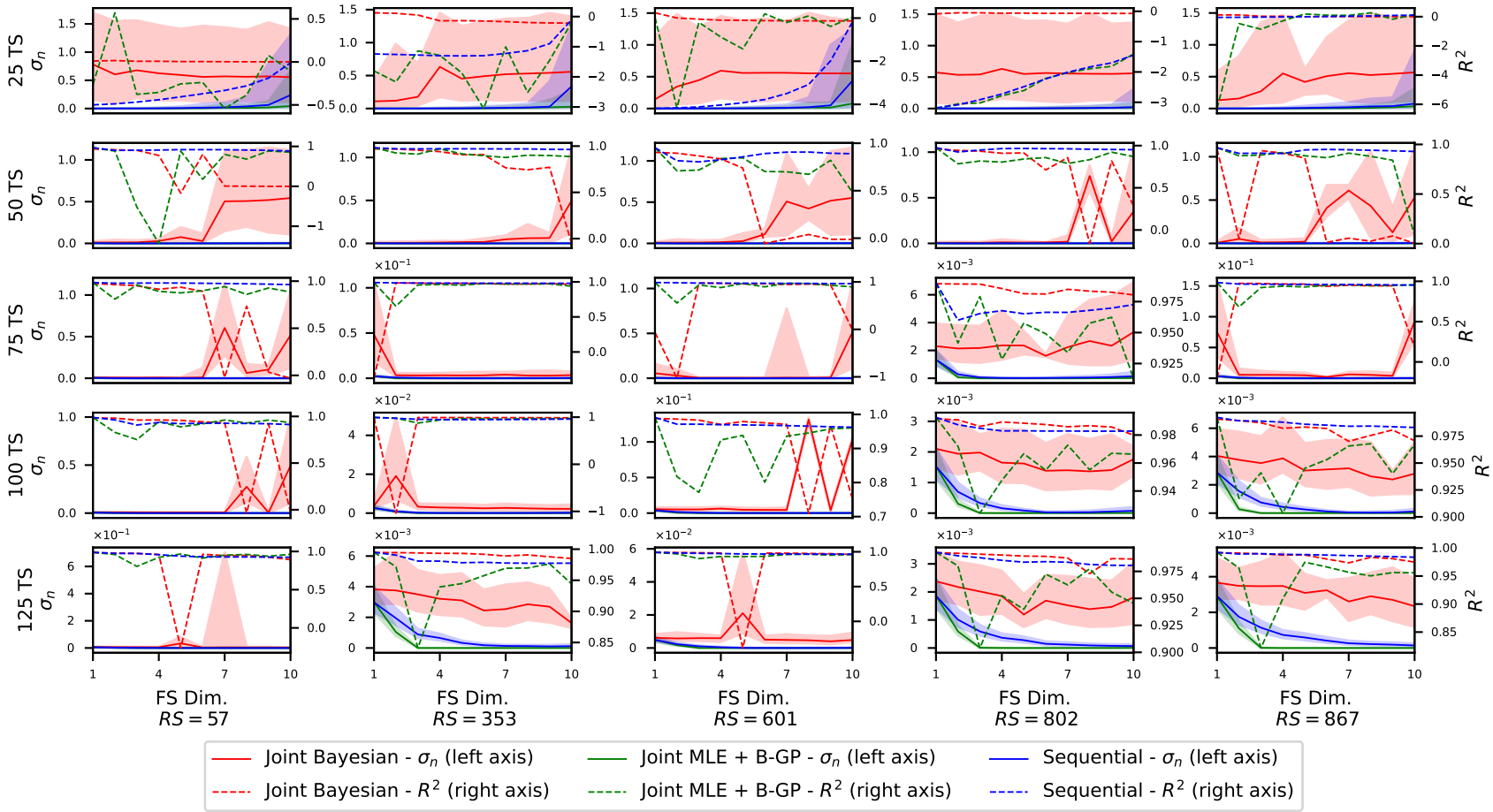


Figure 2.37: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 25 input variables and a 1D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

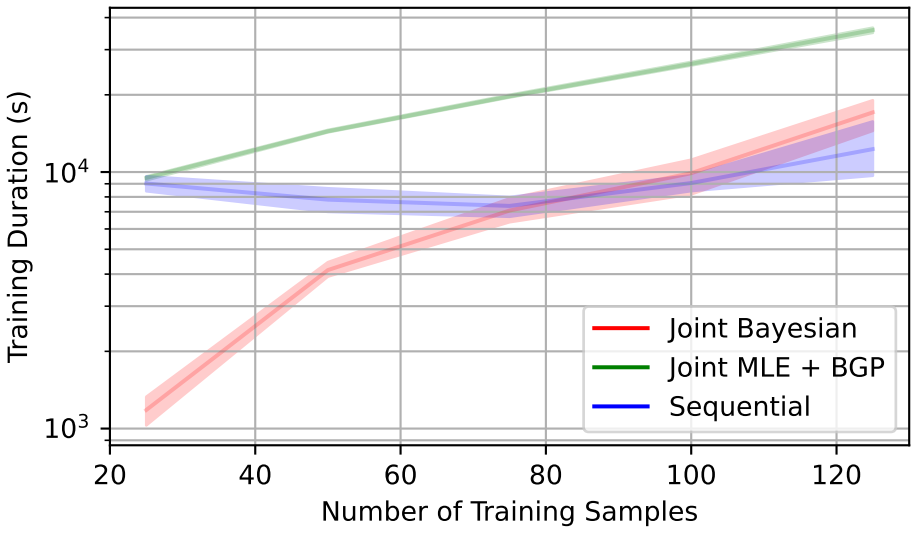


Figure 2.38: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 25 input variables and a 1D feature space.

shorter cumulative training times such as to make the sweep over FS dimensions needed to select the dimension more practical. The comparison of the training durations displayed in fig. 2.38 however shows that the sequential approach does not bring an advantage in that case. Training times are approximately equivalent for the highest numbers of training samples but favor the joint Bayesian approach for the smallest training sizes.

Results for the quadratic function with 25 inputs and a 2D feature space are presented in fig. 2.39. For the first two levels of training samples, poor predictive accuracies are observed irrespective of the model used, and the noise variance remains constant over the whole considered range. A distinction between the joint Bayesian approach and the sequential one is the uncertainty in the value of the noise variance parameter. The confidence intervals for the noise variance value remains high in the case of the joint Bayesian approach, indicating poor confidence in the values of the model parameters Despite using a Bayesian GPR for the low-dimensional link function, the confidence intervals for the noise variance are small despite the poor R^2 values. As a consequence, the uncertainty in the value of the noise variance parameter may not be used as an indicator that more training samples are needed when using the sequential approach, whereas it was shown to be

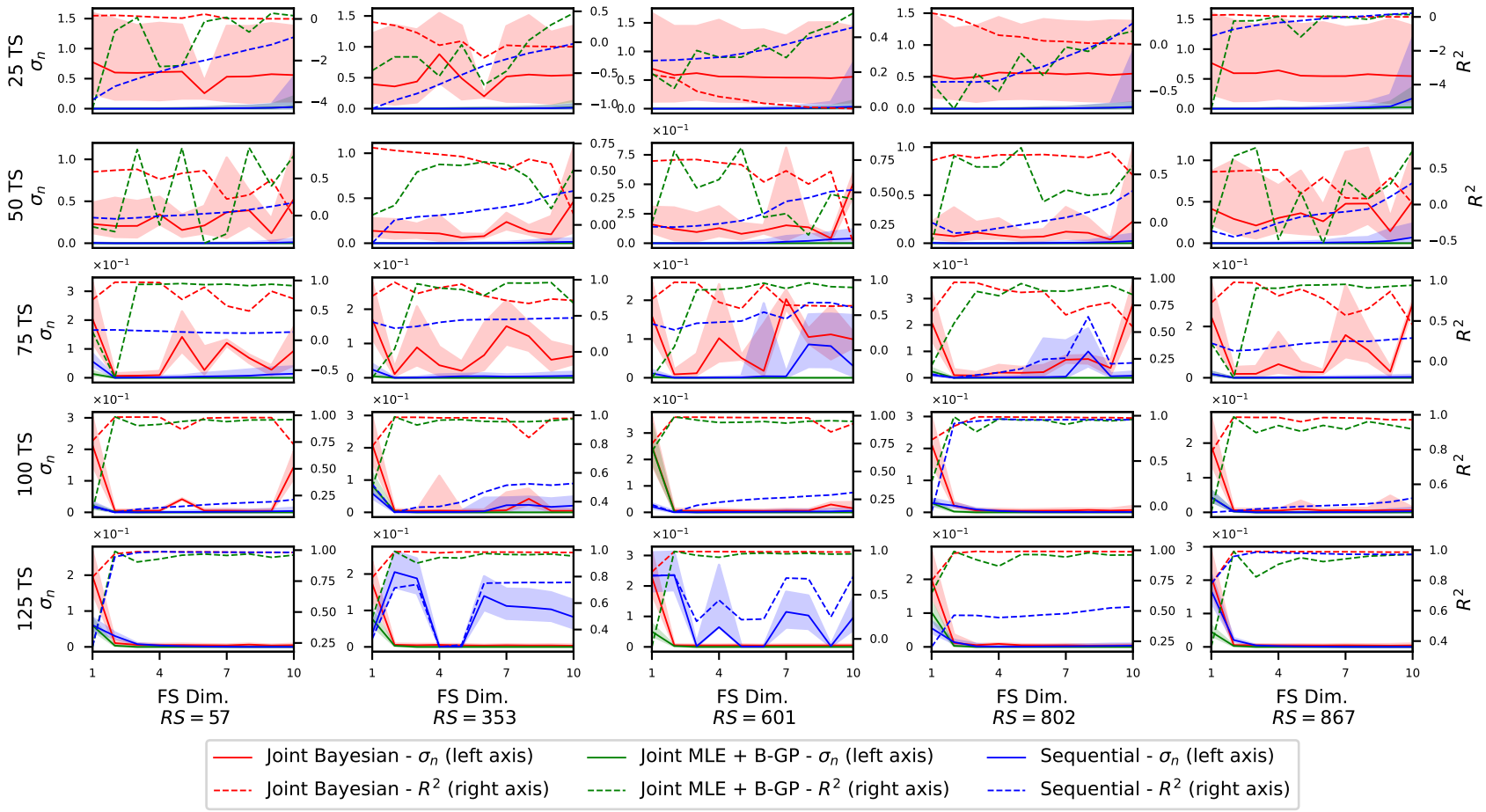


Figure 2.39: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 25 input variables and a 2D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

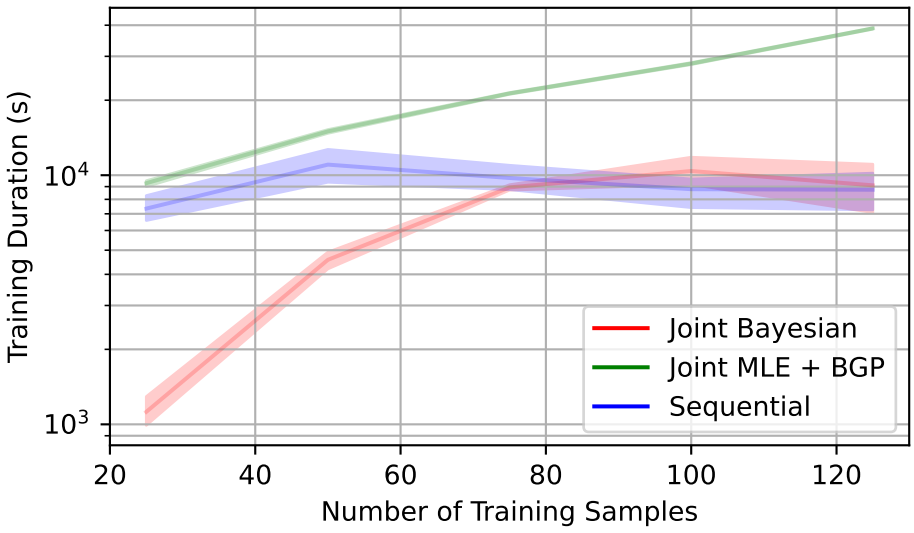


Figure 2.40: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 25 input variables and a 2D feature space.

possible with the joint approach. Predictive performance starts increasing when using 75 training samples, but the joint Bayesian approach still performs consistently better than the sequential approach as measured by R^2. For the two highest levels of training samples, the expected drop in noise variance value between the 1- and 2-dimensional FS is generally observed, except for the sequential approach in two of the five repetitions using 125 training samples. In these cases, models for some FS dimensions exhibit mediocre predictive accuracy for a few dimensions and the resulting evolution of the noise variance for the sequential model does not allow to draw a clear conclusion as to the FS dimension. When it comes to cumulative training times, the picture is the same as for the previous function with a 1D FS: the sequential approach does not bring any noticeable advantage. On the contrary, it is an order of magnitude slower for smaller training sets.

Results for the quadratic function with 25 inputs and a 5D feature space are presented in fig. 2.41. In section 2.3, we already noted that the joint approach struggled with the detection of the 5D FS: the noise variance decay made it clear that more than three dimensions were required, but the need for a 5D FS was not always evident. This is exacerbated in the case of both MLE-based approaches. In addition, the predictive accuracies obtained

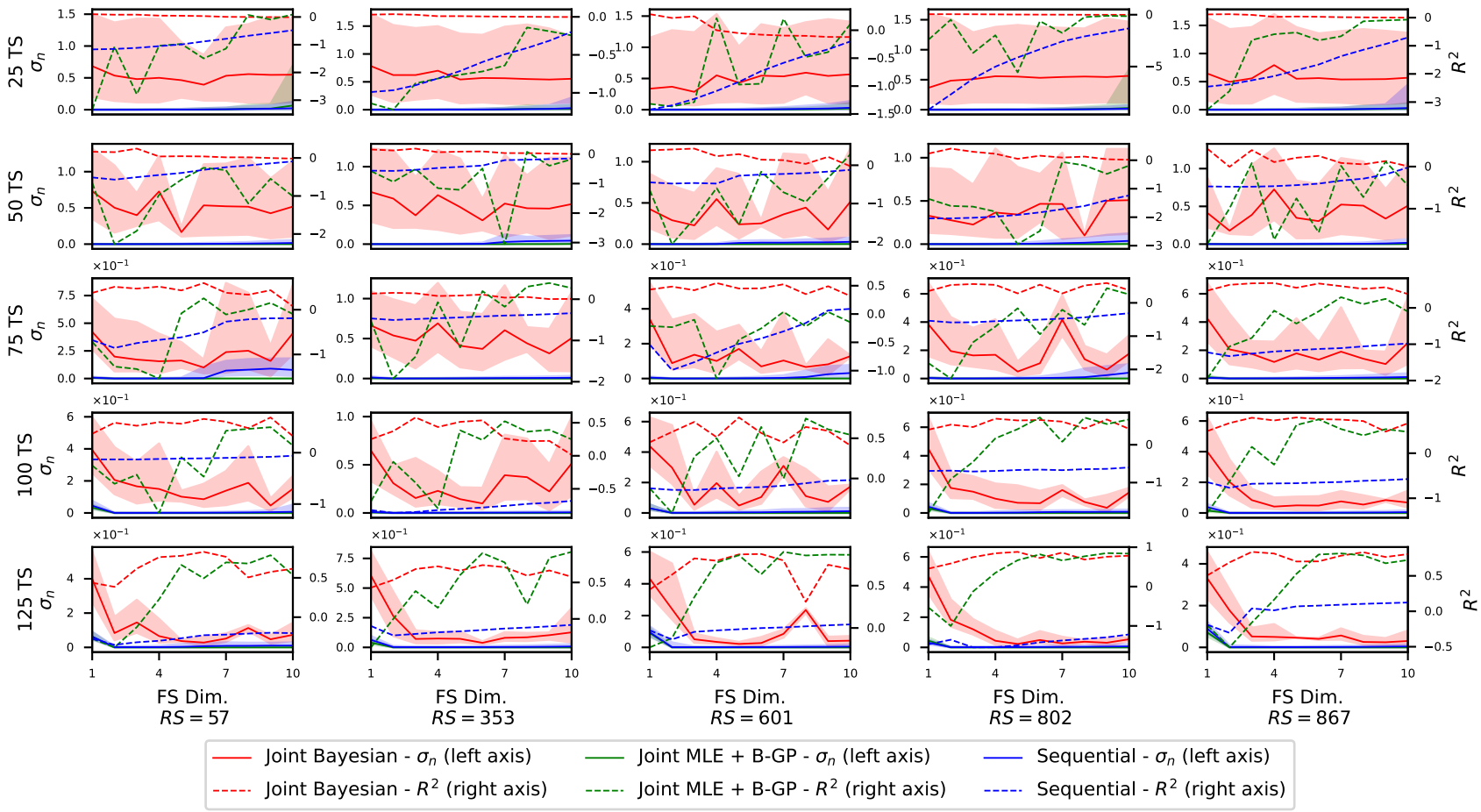


Figure 2.41: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 25 input variables and a 5D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

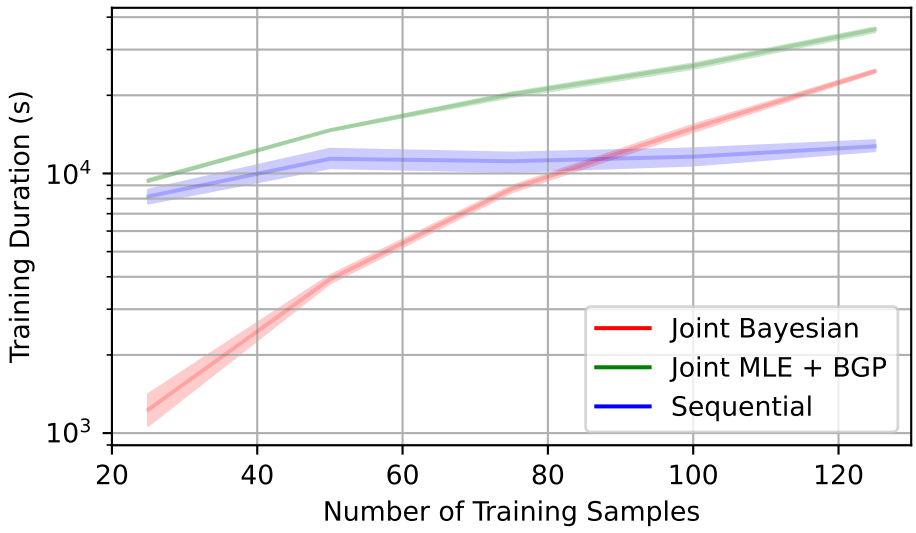


Figure 2.42: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 25 input variables and a 5D feature space.

with the sequential approach remain far below those of the two other approaches, even the latter were already mediocre. In summary, while all methods do seem to struggle with this analytical function, the sequential approach performs even worse, making the proposed selection for the FS dimension based on the noise variance decay inapplicable in this case. This time however, the sequential approach is measurably beneficial in terms of reducing training time for the largest training sets. However, it is still significantly slower than the joint Bayesian approach for the smaller training sets.

The same general comments can be made regarding the results for the remaining quadratic functions whose results are relegated to appendix B. Predictive performance is consistently inferior or at best in par with the joint joint Bayesian approach and in a number of instances, the expected noise variance decay is not observed. A reduction in training duration is not systematically observed when using the sequential approach instead of the joint Bayesian approach: in fact, we have shown in the previous paragraphs than this approach is more often than not significantly slowing down training in the case of the 25-dimensional quadratic function. When looking at the training times for the higher-dimensional quadratic functions however, we observe the opposite to be true: the duration needed to perform the FS

dimension sweep is systematically lower when using the sequential approach, and the time savings increase from the 50- to the 100-dimensional analytical function. This seems to suggest that the expected gains in training time are more likely to be realized for higher-dimensional input spaces. For relatively low input spaces, relying directly on the joint Bayesian approach may be faster.

For higher-dimensional input spaces for which the sequential approach does bring an edge in terms of training time, the issue of the inferior predictive accuracy can be addressed using a two-step hybrid approach. First, select the FS dimension using the sequential approach. Then, train a model according to the joint Bayesian approach using the FS dimension found using the sequential approach. In this manner, the time savings brought by the sequential approach for higher-dimensional input spaces may be exploited while still utilizing the highest predictive accuracy associated with the joint Bayesian approach. As a trade-off, following this hybrid approach comes with a higher risk of selecting an inadequate FS dimension since it relies on the results of the sequential approach that have been shown to be variable.

Active Subspace Datasets

The results obtained for some of the active subspace datasets are shown in figs. 2.43, 2.45 and 2.47. As for the analytical functions, we selected the same datasets as in section 2.3 to make it easier to contrast the results obtained with the sequential approach to those obtained previously using the joint Bayesian approach. Results for all other active subspace datasets are shown in appendix B in order to help keeping the main dissertation body shorter but will still be briefly commented on.

The results for the ONERA M6 drag dataset are shown in fig. 2.43. As expected, all approaches struggle for the three lowest numbers of training samples. As discussed in the previous section, the joint Bayesian approach exhibits the expected trends when 200 or 250 training samples are employed, and a 3-dimensional FS appears to be adequate. The same

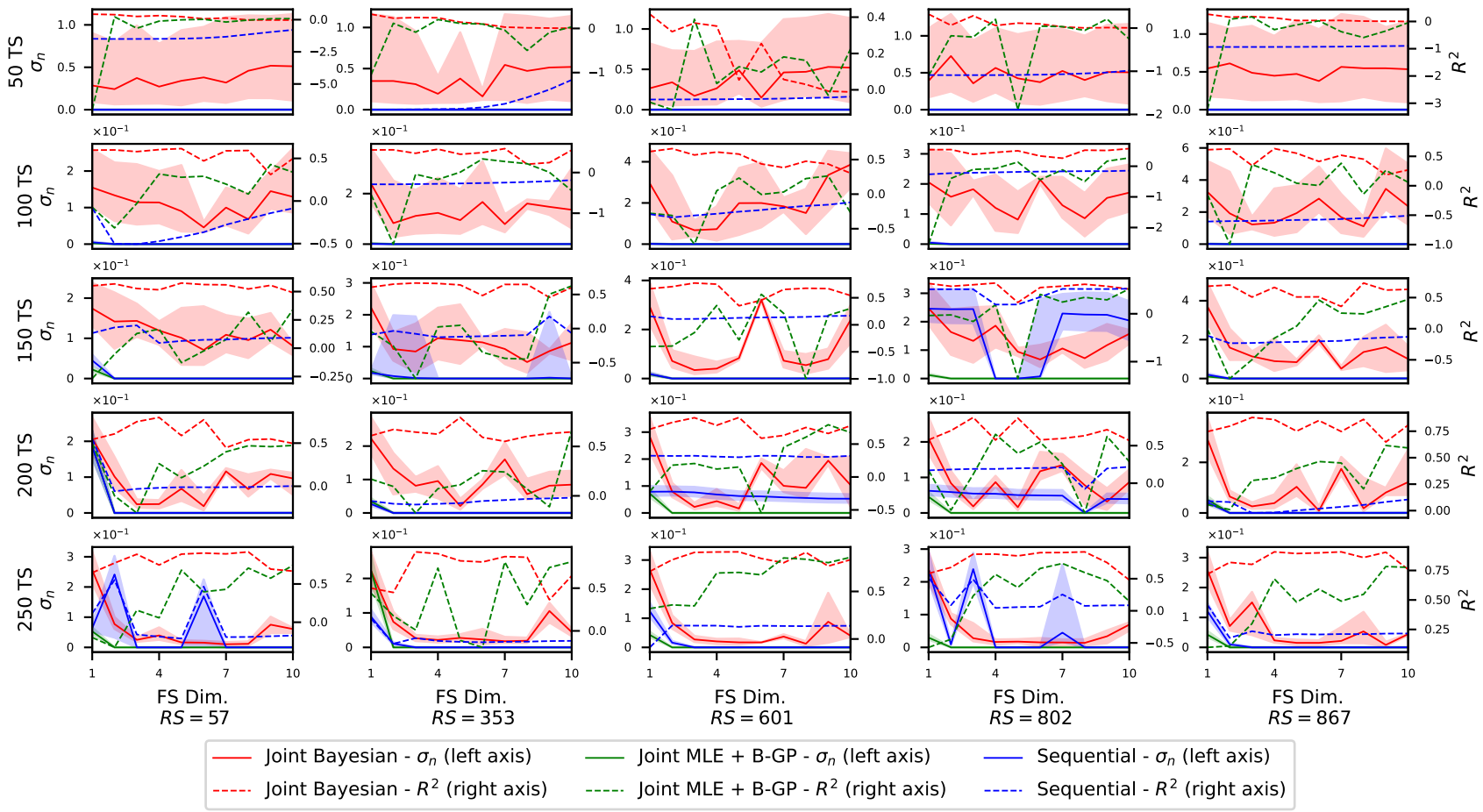


Figure 2.43: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **ONERA M6 drag dataset with 50 input variables**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

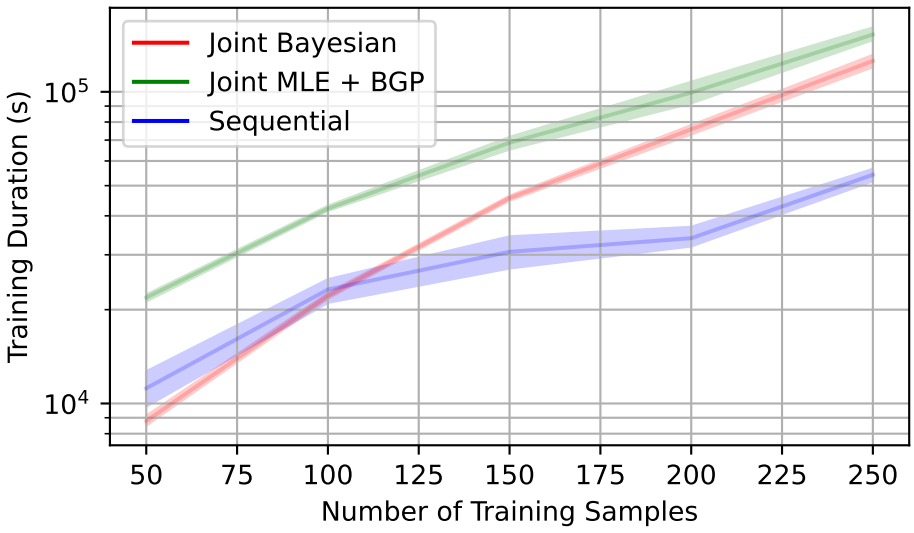


Figure 2.44: Evolution of the training duration as a function of the number of training samples for the ONERA M6 drag dataset with 50 input variables.

cannot be said of the sequential approach, for which few cases actually exhibit a clear noise variance decay. When using 200 training samples, only three out of the five repetitions display a decay, which would rather lead to believe that a 2-dimensional FS is sufficient. The other two repetitions exhibit a flat noise variance curve and poor predictive accuracy all the way from 1- to 10-dimensional FS. When 250 training samples are employed, the expected noise variance decay is observed in three of the five repetitions. In the other two, the evolution of the noise variance with the FS dimension is erratic. Again, predictive accuracies are consistently poor. Here, we can see that although the joint Bayesian approach succeeds at producing models with adequate predictive accuracy, the sequential approach fails: R^2 values remain consistently very low across the board, leading to the absence of the expected noise variance decay. For training sets larger than 100 samples, the FS dimension sweep is faster using the sequential approach rather than starting over training at each dimension. However, for smaller training sets, the sequential and joint Bayesian approaches have similar cumulative training time, with a slight advantage for the joint approach.

Results for the elliptic PDE y_{short} dataset are shown in fig. 2.45. In this instance, we observe different trends for the joint Bayesian approach and the other two approaches. As

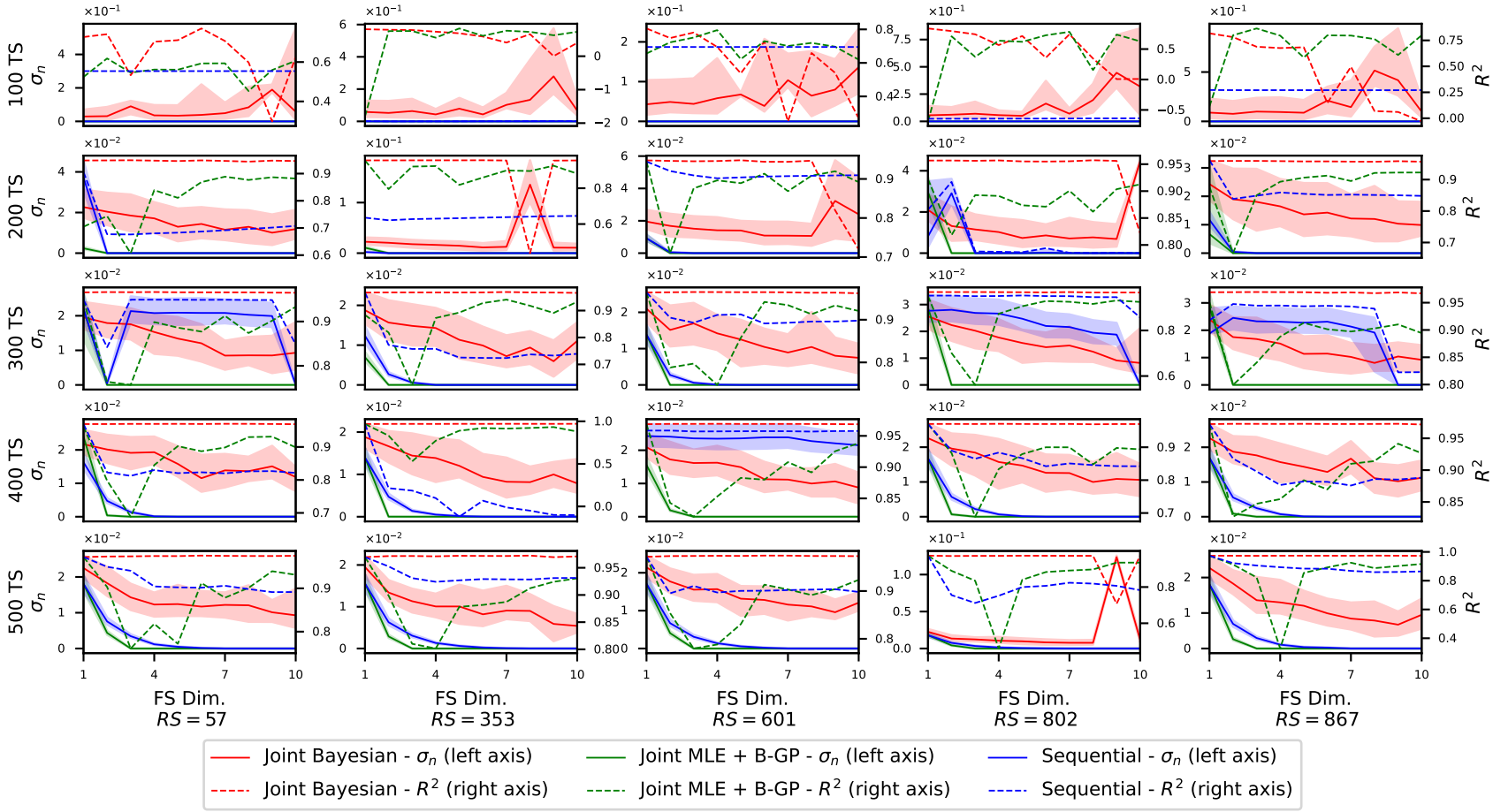


Figure 2.45: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **Elliptic PDE (y_{short}) dataset with 100 input variables**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

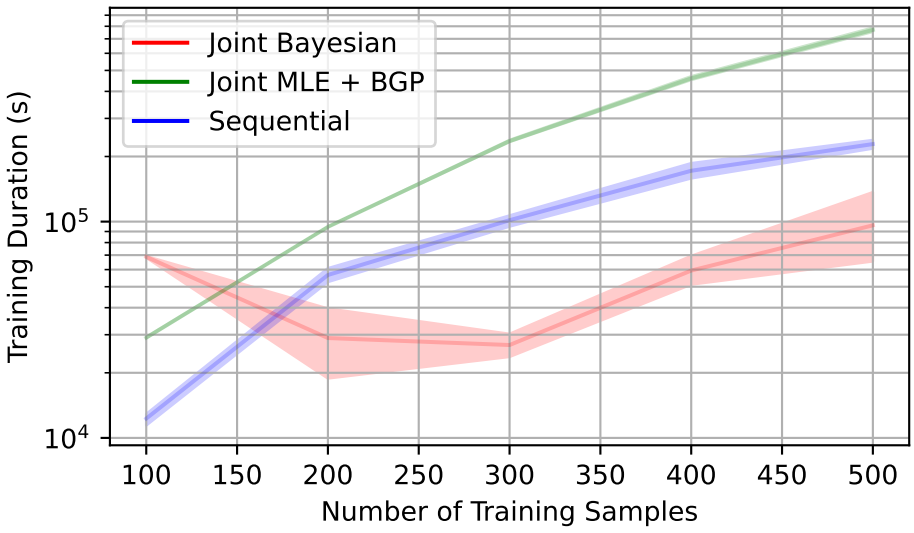


Figure 2.46: Evolution of the training duration as a function of the number of training samples for the Elliptic PDE (y_{short}) dataset with 100 input variables.

discussed in section 2.3.4, the relatively flat evolution of the noise variance displayed by the joint Bayesian approach suggested that a 1-dimensional FS would be sufficient in this case. The MLE-based methods however exhibit a clear noise variance decay that would rather indicate a 3- or 4-dimensional FS. One can however observe the discrepancy between the evolution of the noise variance and R^2 for the two MLE-based two methods. We expect the noise variance to decrease as R^2 increases: as the model captures more of the response's total variance, its accuracy should accordingly increase. Here, on the contrary, we observe that the predictive accuracies of the models decrease as the FS dimension increases. This is problematic because it suggests that the noise variance decay, when using the sequential approach, is not tied to a more accurate model, as opposed to what was observed previously with some regularity with the joint Bayesian approach. As opposed to most other datasets, the sequential approach is much slower than the joint Bayesian approach, especially for larger training sets.

Results for the NACA0012 lift dataset are shown in fig. 2.47. Similarly to the elliptic PDE y_{short} dataset, the evolution of the noise variance for the sequential method mirrors the one for the joint Bayesian method. However, we also observe the same discrepancy

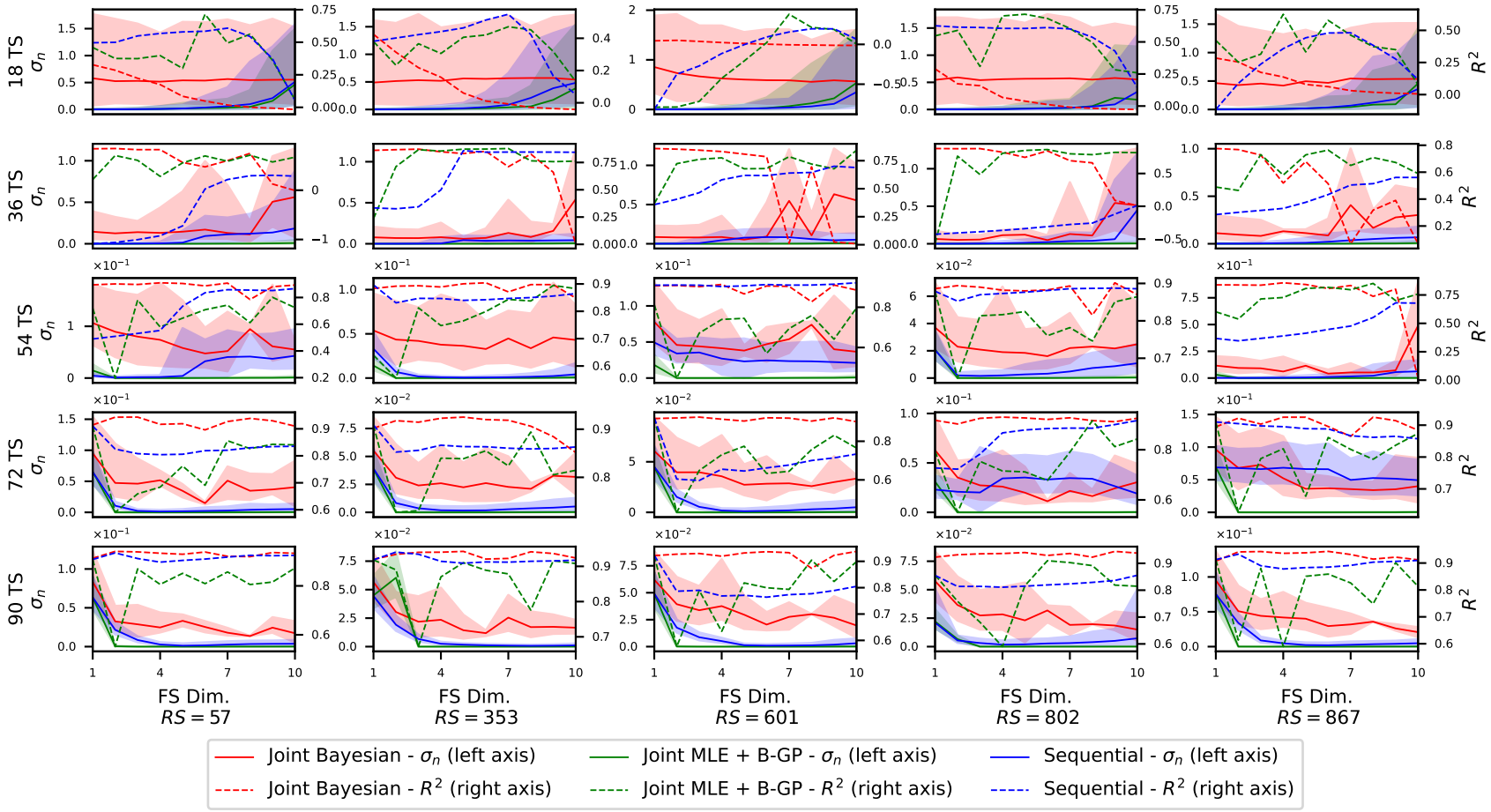


Figure 2.47: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **NACA0012 lift dataset with 18 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

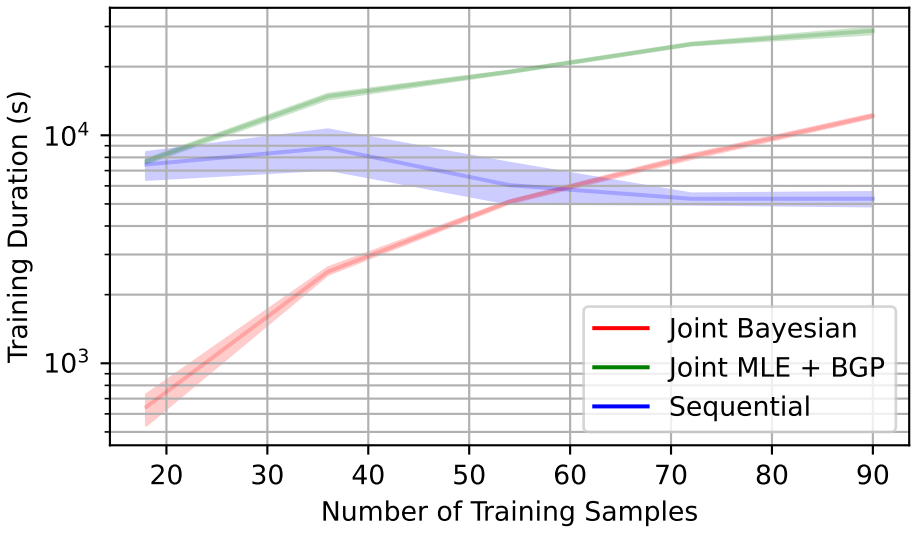


Figure 2.48: Evolution of the training duration as a function of the number of training samples for the NACA0012 lift dataset with 18 inputs.

between the noise variance and R^2: in most cases, as the FS dimension increases, predictive accuracy as measured by R^2 decreases instead of increasing. In the case of the joint Bayesian method, we did observe such decreasing R^2 behavior, but only for large FS dimension values corresponding to large numbers of model parameters. In the case of the joint Bayesian approach, we suggested that poor predictive performance could be the result of a more challenging MCMC inference due to the higher-dimensional parameter space. Here however, the dimension of the parameter space cannot be used as a justification for the poor model's predictive performance. As in other cases, the sequential approach only becomes advantageous with respect to cumulative training time after some training set size threshold is met: for smaller datasets, the original joint approach is significantly faster than the sequential approach.

The remaining active subspace datasets exhibit similar behavior. The noise variance decay using the sequential approach follows the one observed when applying the joint Bayesian approach. However, predictive accuracy is consistently lower when using the sequential approach, and unexpectedly tends to decrease as the FS dimension increases. Similarly to the datasets discussed here, the cumulative training time of the sequential

approach may be lower or higher than the joint approach; results are case-dependent.

Aerodynamics Datasets

Results for a subset of the aerodynamics datasets are presented in figs. 2.49, 2.51 and 2.53. We chose the same subset as in the previous noise variance study conducted in section 2.3 and results for all other aerodynamics datasets are shown in appendix B for completeness.

Results for the CRM subsonic drag dataset are presented in fig. 2.49. We observe that all methods struggle for the lowest two levels of training set sizes. Noise variance decay starts being visible when using 150 training samples, and are similar for all methods, even though the sequential approach yields models with significantly lower predictive accuracy. For the highest two levels of training set sizes, the noise variance decay of the sequential approach mirrors the one of the joint Bayesian approach and would therefore lead to the selection of the same or similar FS dimension. The cumulative training times displayed in fig. 2.50 show that the sequential approach however does not bring a significant advantage: while training duration is lower for the largest training sets, it remains on the same order of magnitude as the joint Bayesian approach.

Results for the CRM subsonic z-moment dataset are presented in fig. 2.49. The same comments as those made for the subsonic CRM drag dataset apply here. The only distinction is that, while remaining low relative to the joint Bayesian approach, predictive accuracies attained using the sequential approach are higher than for the drag response in absolute terms. Figure 2.52 depicts the training duration for the models as a function of the size of the training set. We can see that for the smaller training sets, the joint Bayesian approach is faster. However, the gain expected by using the sequential approach is realized for training sets greater than 100 samples.

Results for the RAE2822 drag dataset are presented in fig. 2.53. This dataset was shown to be challenging in the previous section: even with the largest considered training sets, the joint Bayesian approach yielded models with sub-par predictive accuracy, and the expected

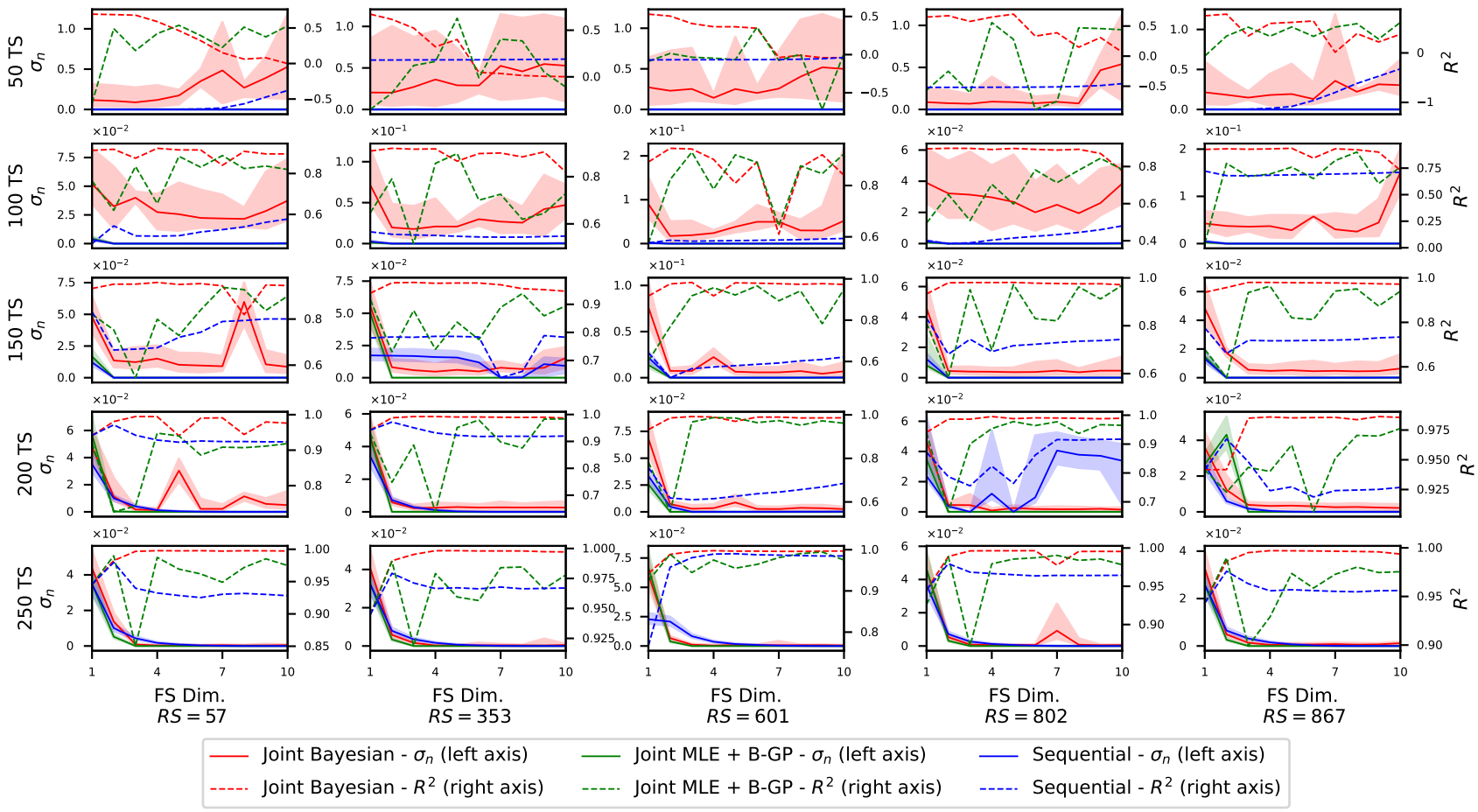


Figure 2.49: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **CRM subsonic drag dataset with 50 input variables**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

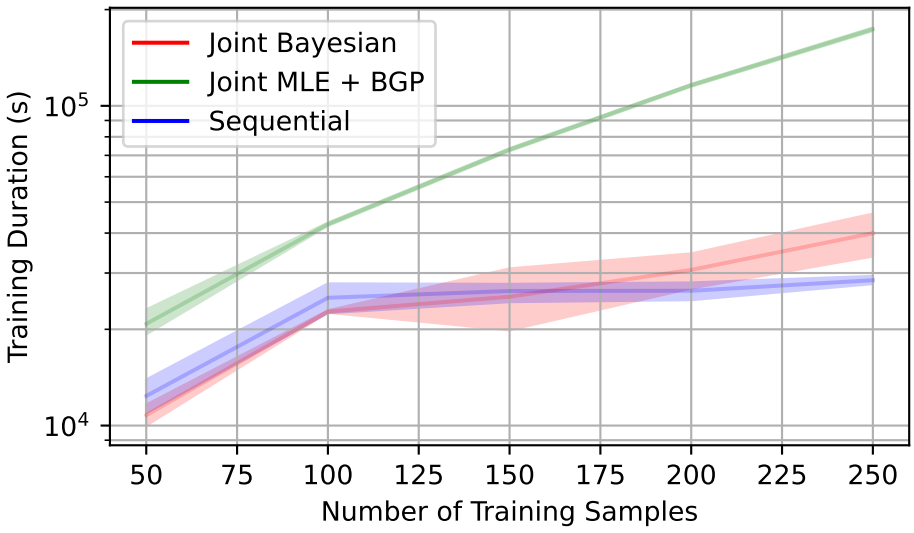


Figure 2.50: Evolution of the training duration as a function of the number of training samples for the CRM subsonic drag dataset with 50 input variables.

noise variance decay behavior was not observed. This is confirmed by the results added here for the other two methods under consideration. While a sharp noise variance drop is visible from the 1D to the 2D FS, the associated R^2 values give very little confidence in the results. The results are not surprising since we expect the sequential approach to perform worse than the joint Bayesian approach, as we trade accuracy for reduced training time. In this case however, training duration is actually longer for the sequential approach than the joint Bayesian approach, as shown in fig. 2.54. This is also the case for the other two responses of the RAE2822 dataset.

Results for the rest of the subsonic and transonic CRM datasets are more promising: not only are better predictive accuracies consistently attained with the sequential approach, but the noise variance decay is also consistent across the three methods. In all of these instances, using the sequential approach would lead to the selection of the same FS dimension as the one that would be selected using the joint Bayesian approach. Results for the remaining two RAE2822 lift dataset are inconsistent: only in rare cases do the noise variance decays match between the different methods, even when using the largest training sets. This is in contrast with the joint Bayesian method that consistently exhibited the expected

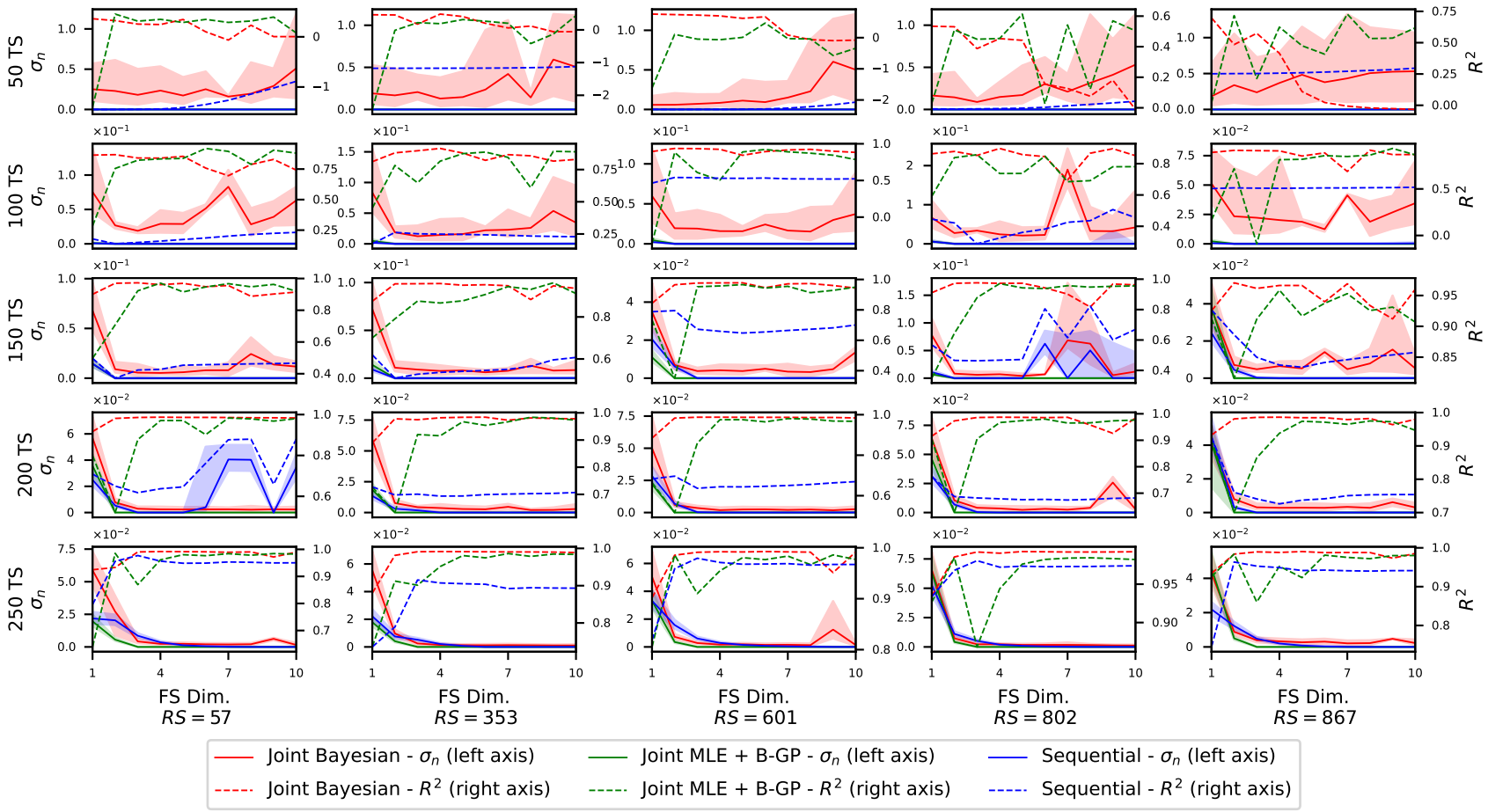


Figure 2.51: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **CRM subsonic z-moment dataset with 50 input variables**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

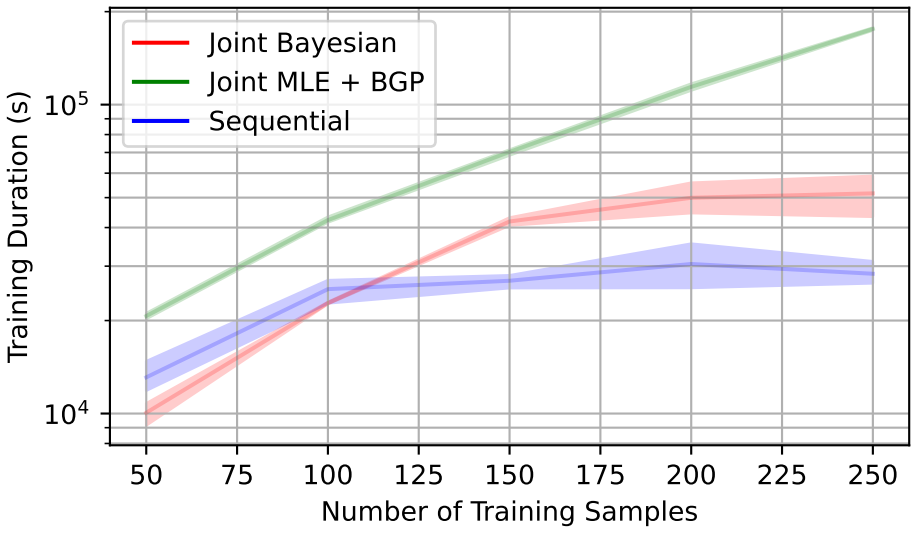


Figure 2.52: Evolution of the training duration as a function of the number of training samples for the CRM subsonic z-moment dataset with 50 input variables.

behavior when applied to the same RAE2822 lift dataset. In the case of the RAE2822 z-moment dataset, the noise variance decay in the sequential approach more closely follows the one from the joint Bayesian approach and would therefore lead to the selection of a similar FS dimension.

2.4.5 Conclusion

Since the proposed approach has been shown to struggle with the smallest considered datasets, the following discussion focuses on the comparison of the methods' performance for the largest training sets within the considered, low-data regime. Out of the 34 datasets studied in this experiment, we found that the noise variance decays match between the joint approach and the sequential approach for 21 of them. Among these 21 datasets, 16 lead to a lower cumulative training time than the joint Bayesian approach. While these results do not show an overwhelming advantage of the sequential approach, they demonstrate that under certain circumstances, the sequential approach does indeed bring the expected advantages.

The reliance of the sequential approach on MLE instead of Bayesian inference for discovering the FS provides levers for further speeding it up. For example, in hindsight, the

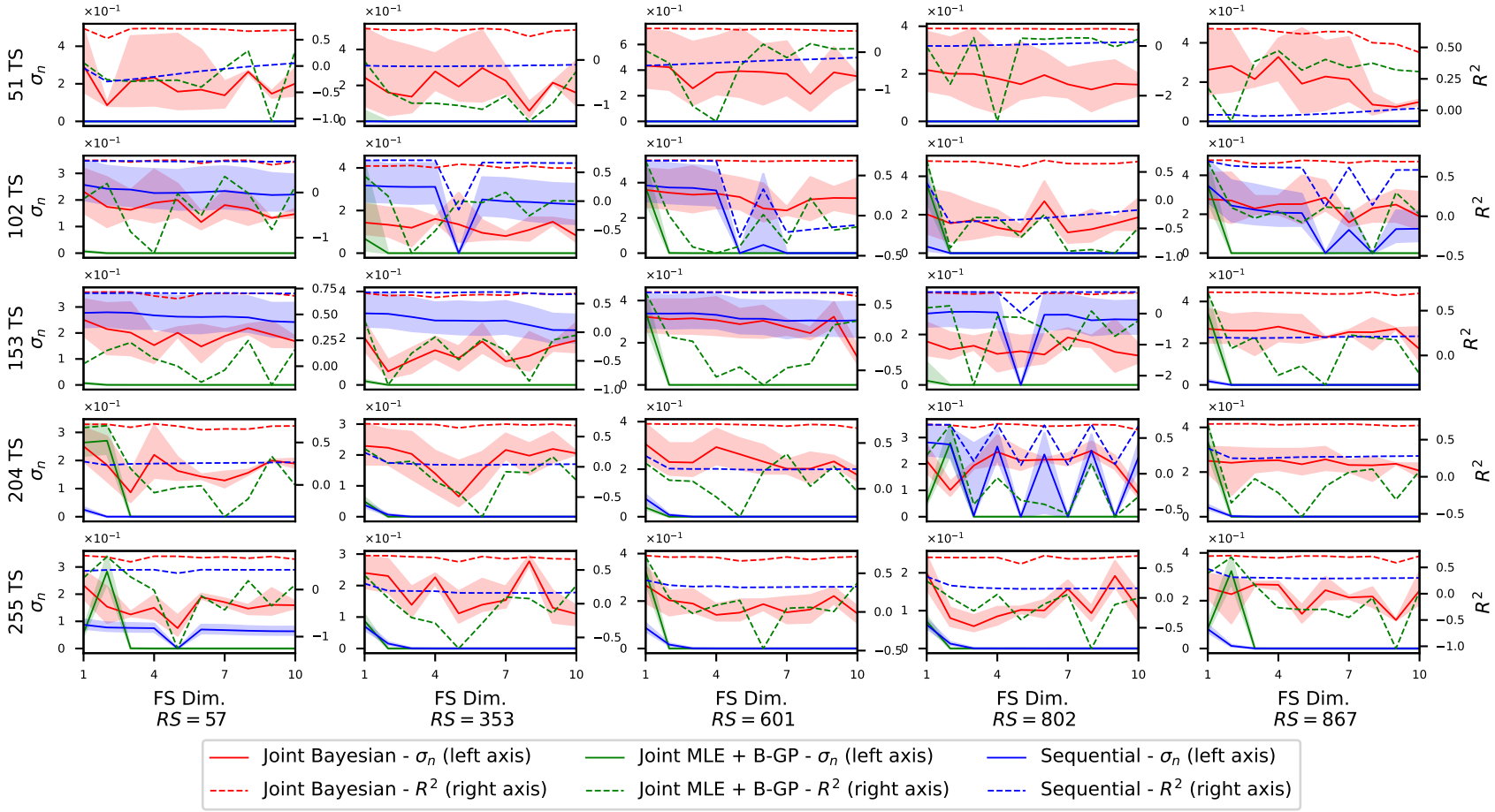


Figure 2.53: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **RAE2822 drag dataset with 51 input variables**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

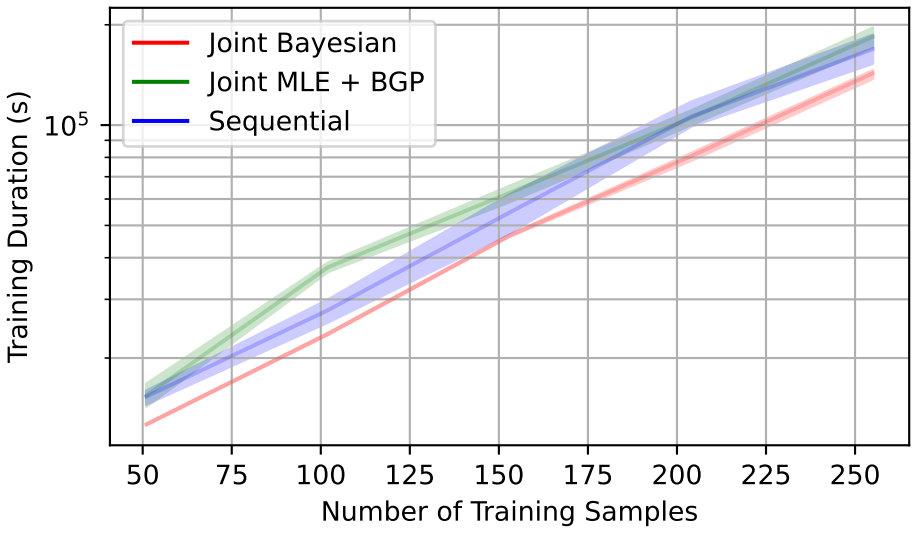


Figure 2.54: Evolution of the training duration as a function of the number of training samples for the RAE2822 drag dataset with 51 input variables.

number of optimization repetitions was set to a relatively high number in the context of this experiment but could be reduced with minimal change in the results. On the other hand, MCMC inference was already set up to use relatively short chains in this experiment, and if speeding it up is possible, it would require more than straightforward changes. As discussed above, the problem of the reduced predictive accuracy of models created using the sequential approach is not of concern: the primary end of the sequential approach is to assist in the selection of the FS dimension. Once the dimension has been selected, the joint fully Bayesian approach that has been shown to exhibit higher predictive accuracy can be applied.

Despite its limitations, the sequential approach did yield promising results on a significant portion of the datasets under study, most of them real-life engineering datasets. Although we recognize that further research is needed regarding the sequential approach in order to enable broader applicability, we consider that the results presented previously do partially support hypothesis 1.3.

Conclusion on Hypothesis 1.3

For a subset of the studied datasets and under the condition that the training set is sufficiently large, we showed that:

1. the sequential approach significantly reduced the cumulative training time of the models needed to select the FS dimension,
2. the noise variance decay obtained with the sequential approach would lead to the selection of the same FS dimension as the joint Bayesian approach.

2.5 Conclusion

2.5.1 Summary

In this chapter, we introduced the proposed fully Bayesian approach to approximation by ridge functions and showed that it outperforms state-of-the-art approaches in the single-fidelity surrogate modeling scenario, where the underlying analysis has high-dimensional inputs and only few observations are available for training.

We first refined the first research question first introduced in chapter 1. We started by recalling why unsupervised dimension reduction commonly used in machine learning are not applicable in the context of the input space. Then, we took a look at existing methods that actually address high dimension of the input space. Among these, projection-based surrogate modeling methods were identified as most promising based on a literature review, and we dived in more details into the corresponding field of approximation by ridge functions. There, we identified the challenges faced by existing approaches: either the gradient is needed, as in the AS method, or many observations are required, due to the sheer number of model parameters that need to be found. The refined version of research question was then proposed, with a scope reduced to the enablement of surrogate modeling methods based on approximation by ridge functions when gradient evaluations are not available and

only few analysis observations are affordable.

Seeking a solution addressing the newly introduced research question, we turned to a presentation of the Bayesian framework and recalled that, in the context of surrogate modeling, it can be used to quantify epistemic uncertainty in the model's parameters due to limited observations of the underlying analysis. We showed that a benefit of quantifying epistemic uncertainty when only few observations are available is to alleviate over-fitting that plagues traditional optimization-based methods, leading to improved predictive accuracy. Adopting a Bayesian approach therefore appeared as a promising way to enable the training of surrogate models relying on approximation by ridge functions with relatively few training samples. However, we observed that the concrete implementation was challenging, as some of the parameters involved in approximation by ridge functions are orthogonal matrices used to project the original input space onto a low-dimension subspace, that we refer to as the feature space. We considered multiple options options for handling these orthogonal matrices while performing Bayesian inference, and identified Householder transformations as the most promising. We formalized the proposed approach as a probabilistic model, hypothesized that it would outperform existing state-of-the-art methods for problems with high-dimensional inputs when few observations are available, and designed an experiment to gather evidence in favor of this hypothesis. On multiple examples coming from science and engineering, we showed that the proposed approach allowed to reach better predictive accuracy than state-of-the art approaches, thus validating the proposed approach in these instances.

While the dimension of the AS was fixed throughout the first study, this is not the case in practice when being confronted to a new dataset. Existing methods to assess the AS dimension have been proposed, notably alongside the two benchmark methods used in the first study. In [91], the authors propose to successively train the model assuming different AS dimensions and select the dimension based on the BIC. This method may be deemed unsatisfactory as it requires multiple costly training runs to obtain a single model. In [142],

the original AS methodology for selecting the number of active dimensions can be carried out since a surrogate model in the full-dimensional input space is built. However, as shown in this study, significantly more samples are required to obtain a good model in the full-dimensional input space compared to methods explicitly incorporating the projection onto a low-dimensional subspace. Since existing methods for determining the FS dimension have shortcomings or are not applicable in the context of the proposed fully Bayesian approach, the next two sections of the chapter focused on strategies for detecting the dimension of the FS. First, we proposed and studied the effectiveness of a method relying on the evolution of the predictive model's noise variance to select the FS dimension. Second, we studied alternate ways of training the proposed surrogate model that could reuse information from previous training such as to speed up the FS dimension sweep used when selecting an FS dimension.

2.5.2 Contributions

Multiple contributions were made in this chapter. The first contribution is a fully Bayesian and gradient-free formulation for meta-modeling of functions with high-dimensional inputs drawing inspiration from the AS method. Under this formulation, all model parameters are considered uncertain, including those associated with the projection onto the lower-dimensional input subspace. The second contribution is a set of algorithms that enable the practical implementation of the proposed formulation using "turn-key" MCMC samplers. Those additional implementation specifics, based on Householder transforms, are needed to simultaneously accommodate traditional GP hyperparameters defined in Euclidean space and an orthonormal projection matrix belonging to the Stiefel manifold as model parameters during probabilistic inference. The third contribution is a thorough comparative study of the model's performance with two recently proposed methods as seen from four different perspectives. The fourth contribution is an approach for detecting the FS dimension using the evolution of the noise variance parameter The fifth contribution is an exploration of

alternate training strategies that accelerate the FS dimension sweep required when attempting to detect the FS dimension. The last contribution are two software repositories, written in the Python programming language and made openly available, that implement the proposed algorithms and benchmark methods using state-of-the-art probabilistic programming languages, manifold optimization libraries, and computational backends, enabling reproducibility and allowing interested researchers to develop extensions to the proposed method.

2.5.3 Next Steps

When working under a limited analysis budget and expensive analyses, only a few analysis observations are available to train a surrogate model. Although the fully Bayesian approach proposed in this chapter yields better results than benchmark methods in these conditions, it is still challenging to work with very few training observations, especially when it comes to the identification of the low-dimensional FS. One of the observations made in this chapter was that once a relevant FS has been identified (by comparison to the subspace obtained using the AS method), the predictive accuracy usually follows. When analyses of different fidelities are available, we can increase the number of available observations for the same total analysis budget by relying on the evaluation of a cheaper, lower-fidelity model. If these additional observations could help identify the FS earlier, we would then expect to reach a better predictive accuracy while keeping the analysis budget the same. In order to take advantage of the observations of different fidelities, a multi-fidelity approach is however needed. Its development will be the subject of the next chapter.

CHAPTER 3

MULTI-FIDELITY EXTENSION

3.1 Multi-Fidelity Extension

3.1.1 Background and Research Objective

In chapter 1, we selected multi-fidelity surrogate modeling as one of the three surrogate modeling scenarios under the scope of this thesis, as it is one of the main strategies at the modeler's disposal when analysis budget is limited and multiple analysis fidelity are available. In section 1.4.2, we observed that the curse of dimensionality may impact multi-fidelity surrogate modeling in multiple ways, leading to the introduction of the second research area in section 1.4.3. This research area is driven by research question 2:

Research Question 2

How can we alleviate the impact of the curse of dimensionality on multi-fidelity surrogate modeling for low analysis budgets?

The generic process corresponding to multi-fidelity surrogate modeling is recalled in fig. 3.1. In section 1.4.2, we identified the specific ways in which the curse of dimensionality may impact multi-fidelity surrogate modeling, which are the same as for single fidelity surrogate modeling. These challenges include the relative sparsity of the training samples due to the exponential increase in the "size" of the input space, the necessity to include many parameters for parametric methods, and the altered notion of neighborhood for non-parametric methods. All these difficulties are compounded by the relatively low number of observations available when the underlying analysis is expensive. In fact, these are much of the same challenges encountered with single-fidelity modeling in the same context, which was the focus of the first research area.

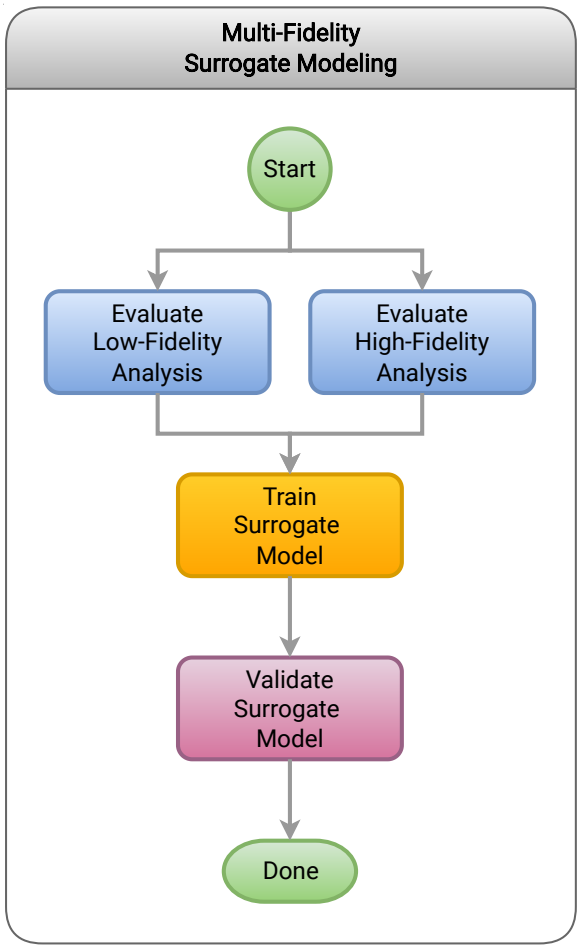


Figure 3.1: Generic process for the second considered surrogate modeling scenario, multi-fidelity surrogate modeling.

Observation

The curse of dimensionality affects multi-fidelity surrogate modeling in the same ways as single-fidelity surrogate modeling.

Single- and multi-fidelity surrogate modeling methods share much of their theoretical foundation. Distinctions lie in the specifics of their underlying mathematical model. While single-fidelity models are simply interpolators or regressors, multi-fidelity models are not only designed to model multiple fidelities individually, but also to model the dependence between fidelities [34]. Multiple strategies exist to achieve this, which are covered in the review conducted in [34] and briefly summarized in the next section. In the following section, we will focus on existing multi-fidelity approaches to AS.

High-Level Strategies for Multi-Fidelity Surrogate Modeling

Analyses of varying levels of fidelity are sometimes available to model the same underlying phenomenon. In this situation, multi-fidelity (MF) methods allow to leverage all levels of fidelity at once to reach greater performance than if levels of fidelity had been used individually by taking advantage of the statistical dependence that is expected to exist between them. MF methods can be leveraged for a variety of many-query applications, sometimes referred to as *outer-loop applications*, such as uncertainty propagation, optimization, or statistical inference.

In their broad review of MF methods, [34] classify multi-fidelity models as being either based on an adaptation, fusion, or filtering strategy. Adaptation strategies "enhance the low-fidelity model with information from the high-fidelity model while the computation proceeds" [34]. Strategies based on fusion "evaluate low- and high-fidelity models and then combine information from all outputs" [34]. In those based on filtering, "the high-fidelity model is invoked following the evaluation of a low-fidelity filter" [34].

Multi-Fidelity Approaches to AS

A few multi-fidelity approaches to AS have recently been proposed in the literature.

In [151], the authors aim at improving the convergence of a multi-fidelity control variate Monte-Carlo estimator by leveraging the gradient-based AS for selecting new observations. A first batch of observations of the LF and HF analyses are selected in the original high-dimensional input space. The AS of both LF and HF analyses are computed using their respective gradient evaluations, and new observation locations are selected to be normally distributed within each AS. Then, the remaining observations are selected at the same coordinates in their respective AS, seeking to increase the correlation between these observations. Here, the multi-fidelity aspect comes from the application towards which the AS method is applied, i.e. multi-fidelity control variate estimation, rather than the method itself. While some aspects of the method are similar to approaches proposed in this thesis, this method is not intended for surrogate modeling and requires the availability of gradient evaluations.

In [109], LF and HF gradient samples are used together within a control variate Monte-Carlo estimator to improve the approximation of the uncentered covariance matrix of the gradient used to derive the AS. Unlike [151], this time the method to determine the AS itself leverages information from both available analysis fidelities. While this method may be leveraged in the context of surrogate modeling [152], it requires the availability of gradient evaluations.

In [153, 154], a multi-fidelity surrogate modeling based on a non-linear multi-fidelity GPs is proposed. The AS is found using a gradient-free method based on maximum likelihood using LF observations only. The projection of the original inputs onto this AS are then used in lieu of the original high-dimensional inputs in the multi-fidelity GP. While this approach shares some aspects with the proposed approach, it differs from it by two aspects: 1) the lack of a full Bayesian treatment to training the model and 2) the exclusive use of LF observations to compute a low-dimensional FS instead of both LF and HF observations.

In [155, 156], a multi-fidelity surrogate modeling approach is proposed in which the gradient-based AS is leveraged to create a LF model. The AS is first derived using HF observations. Then, a regression of the HF observations against the first active dimension is created. This regression is then used as the low-fidelity component of a non-linear multi-fidelity GP. The approach does not leverage an existing LF alternative to the analysis under consideration but rather build such a LF approximation as part of the method. Moreover, it requires access to gradient evaluations.

MF methods have been shown to improve the predictive accuracy of surrogate modeling for a given analysis budget by relying on observations at multiple levels of fidelity [34]. More specifically in context of approximation by ridge functions, MF approaches to the AS method have been recently proposed and shown to be effective at uncovering a low-dimensional subspace of the input space. While these methods are not all directly aimed at surrogate modeling and require the availability of gradient evaluations, this nevertheless suggests that LF models may carry information relevant to identify a low-dimensional subspace also suitable to the HF model. Identifying a suitable low-dimensional subspace of the original high-dimensional input space is a crucial and challenging part in the process of creating surrogate models relying on approximation by ridge functions. If this part of the process could be improved by leveraging the alternate LF analysis, we would expect a significant impact on the resulting surrogate model. However, existing methods to multi-fidelity AS either require gradient evaluations, or are based on MLE approaches that are challenging to apply when only few analysis observations are available. This leads to the following capability gap:

Gap 2.1

Without a multi-fidelity surrogate modeling approach suitable to high-dimensional inputs when the analysis budget is limited and without access to direct gradient evaluations, we cannot take advantage of the availability of cheaper alternate low-fidelity analyses.

In chapter 2, we have shown that the proposed fully Bayesian approach to approximation by ridge functions effectively alleviates the impact of the curse of dimensionality on single-fidelity surrogate modeling. This was achieved by projecting the inputs on a low-dimensional feature space, a mechanism introduced by the field of approximation by ridge functions. Our contribution was to enable such approaches when only few analysis observations are available *via* a Bayesian approach to training whose application was complicated by the nature of some of the model's parameters. Since the curse of dimensionality affects multi-fidelity modeling in much the same ways as single-fidelity surrogate modeling, the single-fidelity approach proposed in chapter 2 appears to be a promising starting point to address the shortcomings of the methods proposed in the literature, leading to the following scoped-down research question:

Research Question 2.1

How can we extend the proposed fully Bayesian approach to training surrogate models based on approximation by ridge functions to the multi-fidelity context?

In [131], a multi-fidelity surrogate model based on GPR and featuring dimension reduction of the input space is proposed. The autoregressive Gaussian process (ARGP) scheme from [37] that assumes a linear relationship between the LF and HF mappings is used for multi-fidelity modeling. To sample the posterior distribution of the models' parameters, including the projection matrix, it relies on a custom geodesic MCMC scheme. Although, this method shares many aspects with the proposed approach, as it will be visible in the next section, multiple distinctions can be made. First, the proposed approach to sampling the

Bayesian posterior uses the same mechanism as in chapter 2, thus allowing the use of existing and future turn-key MCMC samplers. This added flexibility leaves the possibility to take advantage of future improvements in MCMC sampling, but also eases the integration of the projection in more complex models that benefit from existing MCMC frameworks. Second, the proposed approach builds on deep MF GPs instead of ARGP for MF modeling and is therefore compatible with non-linear relationships between the different fidelity levels.

3.1.2 Proposed Method

In this section, we are building and presenting the approach to multi-fidelity surrogate modeling that we propose to alleviate the impact of the curse of dimensionality. It attempts at bridging the capability gap presented in the previous section and address research question 2.1.

Multi-Fidelity Approach

In the previous section, we looked at existing approaches to multi-fidelity surrogate modeling and the AS method. We saw that existing approaches either require gradients or rely on an MLE-based training. As such, they are not suitable to situations where the analysis budget is low and only few observations of the analyses are available. We therefore proposed to extend the fully Bayesian approach shown to perform well in the single fidelity context to the multi fidelity context such as to benefit from the introduced improvements. We are now seeking the multi-fidelity approach on which the fully Bayesian approach could be applied. Two main MF surrogate modeling options are identified, that appear as natural options for the foundation of the MF extension because they are also based on GPR: the autoregressive Gaussian process (ARGP) [37] model and the deep multi-fidelity Gaussian process (DMF-GP) model [157].

The ARGP model that was introduced in [51] and later extended [158] uses an adapta-

tion strategy in which a first GP is used to model the LF code and a second GP is used to model the discrepancy between the LF and HF observations. This model has been abundantly used in literature, and it has also been adopted in the industrial setting as illustrated by its integration into Sandia National Lab's popular Dakota design toolkit [159].

Multiple deep approaches have been proposed to multi-fidelity surrogate modeling based on GPs [160, 157]. Here, we focus on the one proposed in [157]. [157] proposed a complete formulation in which levels of fidelity are recursively modeled by sparse GPs whose inputs are a combination of 1) the original inputs and 2) the outputs of the previous level's GP, alongside the necessary results for an stochastic variational inference (SVI)-based training approach.

Among the two considered options, we choose to use the deep multi-fidelity Gaussian process (DMF-GP) model as its modeling of the relationship between the LF and HF analyses is more generic than the ARGP model.

Notation

In the following, we assume that we only have two levels of fidelity (low-fidelity (LF) and high-fidelity (HF)). We differentiate the corresponding quantities using the subscripts "LF" and "HF": $\mathbf{X_{LF}}$ and $\mathbf{X_{HF}}$ refer to the training locations for each fidelity, $\theta_{\mathbf{LF}}$ and $\theta_{\mathbf{HF}}$ refer to the model parameters of each layer. Superscripts are used as shortcuts – instead of a proper functional notation – to denote the input at which an output has been evaluated, e.g., $\mathbf{y_{LF}^{HF}}$ instead of $\mathbf{y_{HF}}(x_{LF})$, hopefully making the notation more compact while retaining the same level of clarity. This is useful for distinguishing analysis evaluations made at LF training locations, HF training locations, and new prediction locations (marked with $\star$ superscript).

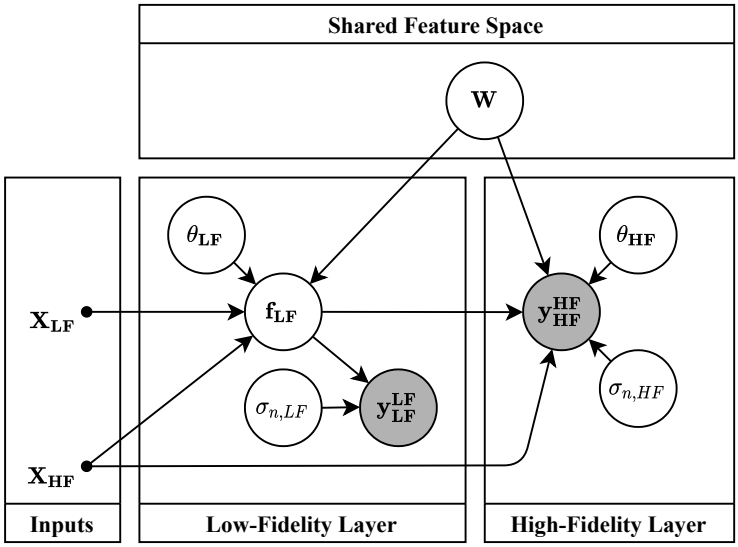


Figure 3.2: Probabilistic graphical model for the proposed multi-fidelity model to approximation by ridge function.

Probabilistic Model

Compared to the DMF-GP, we propose the following alterations. First, since our effort is focused in the low-data regime where the number of training samples is assumed to be relatively small, we drop the sparse approximation. Second, since we are aiming for a fully Bayesian approach, we drop the use of approximate inference using SVI and rather use MCMC to sample from the joint posterior distribution of the model's parameters.

A graphical model for the proposed model is shown in fig. 3.2. We start by presenting the core equations of the probabilistic model, then discuss the covariance structure employed for both the LF and HF layers of the model, and finally show the prior distributions used for the model parameters.

First, we note that we assume the FS, and therefore the projection matrix $\mathbf{W}$, to be shared between the LF and HF layers of the proposed model. This is consistent with existing approaches in the literature, and it is motivated by the fact that we are expecting the more abundant LF observations to drive the discovery of a relevant FS that may be useful for the HF prediction. However, alternate ways to model this are possible and will be explored in the next research question.

The LF layer of the model is a traditional GP in which we assume the training samples have been normalized such that they have zero mean.

$$\mathbf{f_{LF}} \sim \mathcal{N}(0, \Sigma_{LF}) \tag{3.1}$$

As in chapter 2, the ARD kernel computed with inputs projected onto the input space is used to encode the correlation structure of the GP, and $\theta_{\mathbf{LF}}$ accordingly contains the length scales and signal variance parameters. As explained in chapter 2, even though they may originate from a deterministic computer experiment, we model observations as noisy because the projection because of the projection of the observations on a low-dimensional subspace.

$$\mathbf{y_{LF}^{LF}} \sim \mathcal{N}(\mathbf{f_{LF}}, \sigma_{n,LF}^2 \, \mathbf{I_{N_{LF} \times N_{LF}}}) \tag{3.2}$$

Where $\sigma_{n,LF}^2$ is the noise variance for the LF layer.

Instead of feeding the noisy predictions to the HF layer of the model, we found that using $\mathbf{f_{LF}}$ led to better predictive capabilities. The HF layer is also modeled as a zero-mean GP:

$$\mathbf{y_{HF}^{HF}} \sim \mathcal{N}(0, \Sigma_{HF}) \tag{3.3}$$

Let $\mathbf{x_1}$ and $\mathbf{x_2}$ be two points in the input space. Let $\mathbf{z_1} = \mathbf{W}^T\mathbf{x_1}$ and $\mathbf{z_2} = \mathbf{W}^T\mathbf{x_2}$ be the projections of these inputs onto the FS. Let us denote by $\mathbf{x_1'} = [\mathbf{z_1} \, f_{LF}(\mathbf{x_1})]$ and $\mathbf{x_2'} = [\mathbf{z_2} \, f_{LF}(\mathbf{x_2})]$ the "augmented" inputs of the HF layer. The correlation structure of the HF layer is therefore a function of $\mathbf{x_1'}$ and $\mathbf{x_2'}$ It is also more complex as it not only models the correlation across the FS, but also across the augmented inputs. We chose to use the same composite correlation structure as [157] reproduced in eq. (3.4).

$$\begin{aligned} k(\mathbf{x_1'}, \mathbf{x_2'}) = & \, k^{\rho}(\mathbf{z_1}, \mathbf{z_2}; \theta_{\rho}) \, (\sigma_l^2 \, f_{LF}(\mathbf{x_1})^T f_{LF}(\mathbf{x_2}) \\ & + k^{LF}(f_{LF}(\mathbf{x_1}), f_{LF}(\mathbf{x_2}); \theta_{\mathbf{LF}})) + k^{\delta}(\mathbf{z_1}, \mathbf{z_2}; \theta_{\delta}) \end{aligned} \tag{3.4}$$

Where k^{ρ}, k^{LF} and k^{δ} are covariance kernel functions. This covariance structure was designed to capture both linear and non-linear correlation in the space of the augmented inputs

as well as the correlation in the original input space. In our case, we select k^{ρ}, k^{LF} and k^{δ} to be ARD kernel functions and equip them with the relevant length scale and signal variance parameters.

As a resulting of experimenting with different priors, we selected half-normal priors with the value of the scale parameter set to 1 for the noise and signal variance parameters, as well as the inverse of the length scale parameters. Using half-normal distributions for the inverse of the length scale parameters favors discarding irrelevant input dimensions, except when the training data provides strong evidence otherwise. Using the half-normal distribution as prior for the noise and signal variance empirically led to more consistent than other considered alternatives.

We gather all model parameters and latent variables and denote them as θ to ease notation: $\theta = (\theta_{\mathbf{LF}}, \theta_{\mathbf{HF}}, \sigma_{n,LF}, \sigma_{n,HF}, \mathbf{W}, \mathbf{f_{LF}})$. When "training" the model, we use MCMC to draw samples from the joint posterior distribution $p_{\text{post}}(\theta)$ of all model parameters and latent variables $\mathbf{f_{LF}}$:

$$p_{\text{post}}(\theta) = p(\theta_{\mathbf{LF}}, \theta_{\mathbf{HF}}, \sigma_{n,LF}, \sigma_{n,HF}, \mathbf{W}, \mathbf{f_{LF}} | \mathbf{X_{HF}}, \mathbf{y_{HF}^{HF}}, \mathbf{X_{LF}}, \mathbf{y_{LF}^{LF}}) \tag{3.5}$$

Prediction

For a given training set $\mathbf{X_{HF}}, \mathbf{y_{HF}^{HF}}, \mathbf{X_{LF}}, \mathbf{y_{LF}^{LF}}$ and prediction sites $\mathbf{X}^*$, the predictive distribution $p(\mathbf{y_{HF}^*} | \mathbf{X}^*, \mathbf{X_{HF}}, \mathbf{y_{HF}^{HF}}, \mathbf{X_{LF}}, \mathbf{y_{LF}^{LF}})$ is obtained by marginalizing out the model parameters and the latent LF predictions as shown in eq. (3.6).

$$\begin{aligned} p^* &:= p(\mathbf{y_{HF}^*} | \mathbf{X}^*, \mathbf{X_{HF}}, \mathbf{y_{HF}^{HF}}, \mathbf{X_{LF}}, \mathbf{y_{LF}^{LF}}) \\ &= \iiint p(\mathbf{y_{HF}^*}, \mathbf{y_{LF}^{HF}}, \mathbf{y_{LF}^*}, \theta_{\mathbf{LF}}, \theta_{\mathbf{HF}} | \mathbf{X}^*, \mathbf{X_{HF}}, \mathbf{y_{HF}^{HF}}, \mathbf{X_{LF}}, \mathbf{y_{LF}^{LF}}) \, \mathrm{d}\mathbf{y_{LF}^{HF}} \, \mathrm{d}\mathbf{y_{LF}^*} \, \mathrm{d}\theta \\ &= \int p_{\text{post}}(\theta) \left(\iint p_{\text{LF}}^*(\mathbf{y_{LF}^{HF}}, \mathbf{y_{LF}^*}) p_{\text{HF}}^*(\mathbf{y_{HF}^*}) \, \mathrm{d}\mathbf{y_{LF}^{HF}} \, \mathrm{d}\mathbf{y_{LF}^*} \right) \mathrm{d}\theta \end{aligned} \tag{3.6}$$

Where the following quantities have been introduced:

- p_{post} is the joint posterior distribution of all model parameters and latent variables:

$$p_{\text{post}}(\theta) = p(\theta_{\mathbf{LF}}, \theta_{\mathbf{HF}}, \sigma_{n,LF}, \sigma_{n,HF}, \mathbf{W}, \mathbf{f_{LF}} | \mathbf{X_{HF}}, \mathbf{y_{HF}^{HF}}, \mathbf{X_{LF}}, \mathbf{y_{LF}^{LF}}) \tag{3.7}$$

- $p^*_{\text{LF}}(\mathbf{y}^{\mathbf{HF}}_{\mathbf{LF}}, \mathbf{y}^*_{\mathbf{LF}})$ is the predictive distribution of the LF Gaussian process regression model at the locations of the HF training samples $\mathbf{X_{HF}}$ and the prediction sites $\mathbf{X}^*$:

$$p^*_{\text{LF}}(\mathbf{y}^{\mathbf{HF}}_{\mathbf{LF}}, \mathbf{y}^*_{\mathbf{LF}}) = p(\mathbf{y}^{\mathbf{HF}}_{\mathbf{LF}}, \mathbf{y}^*_{\mathbf{LF}} | \mathbf{X}^*, \mathbf{X_{HF}}, \mathbf{X_{LF}}, \mathbf{y}^{\mathbf{LF}}_{\mathbf{LF}}, \theta_{\mathbf{LF}}) \tag{3.8}$$

- $p^*_{\text{HF}}(\mathbf{y}^*_{\mathbf{HF}})$ is the predictive distribution of the HF Gaussian process regression model at the locations the prediction sites $\mathbf{X}^*$:

$$p^*_{\text{HF}}(\mathbf{y}^*_{\mathbf{HF}}) = p(\mathbf{y}^*_{\mathbf{HF}} | \mathbf{X}^*, \mathbf{y}^*_{\mathbf{LF}}, \mathbf{X_{HF}}, \mathbf{y}^{\mathbf{HF}}_{\mathbf{LF}}, \mathbf{y}^{\mathbf{HF}}_{\mathbf{HF}}, \theta_{\mathbf{HF}}) \tag{3.9}$$

In practice, eq. (3.6) is estimated using two nested Monte-Carlo estimates: the posterior chains obtained from running the MCMC for the outer integration and, given the value of the model's parameters, draws from the predictive distribution of the LF Gaussian process regression model for the inner integration.

This leads to the following hypothesis:

Hypothesis 2.1

For low analysis budgets and when an alternate low-fidelity analysis is available, a surrogate model with better predictive accuracy can be obtained using the proposed multi-fidelity approach rather than the single-fidelity approach because it can leverage the cheaper – and therefore more abundant – low-fidelity observations.

3.1.3 Setup of Experiment 2.1

Need for Experiment 2.1

Multi-fidelity approaches in general, and more specifically the deep Gaussian process leveraged by the proposed formulation, have been shown to lead to better predictive accuracy than single-fidelity methods for the same analysis budget. This comes from the fact that, although they are less accurate, LF observations are also cheaper to obtain. Therefore, by spending the same analysis budget, more LF observations can be obtained, which enables

a better coverage of the input space, and allows to learn features of the underlying function that could not have been learnt from sparse HF observations. Thanks to the similarity between the LF and HF analyses, this extra knowledge is expected to be relevant to the prediction of the HF analysis.

To support hypothesis 2.1, we need to check that the proposed multi-fidelity approach also provides such benefits: although we have substantiated the proposed approach, it remains a novel formulation that needs to be put to the test. Hypothesis 2.1 hinges on the ability of the multi-fidelity method to increase the predictive accuracy compared to using a single-fidelity model. Therefore, as opposed to experiment 1.1, we do not seek to compare the proposed method to another multi-fidelity method from the literature. Instead, we seek to compare it to the single-fidelity method previously developed and validated in section 2.2. By doing so, we will quantify the impact of the multi-fidelity approach on the predictive accuracy.

Experiment Design

Based on the previous discussion, experiment 2.1 should consist in a comparison between 1) the predictive accuracy of models trained using the proposed multi-fidelity approach and 2) the predictive accuracy of models obtained using the single-fidelity approach from chapter 2. The single-fidelity method was already implemented in the context of experiment 1.1. Here, we implemented the proposed multi-fidelity approach based on the formulation provided in the previous section. More details regarding the implementation of the methods are given in the subsection *Implementation Details* below.

Because we are assessing a surrogate modeling technique, analyses are required to test the performance of the proposed approach to surrogate modeling. Since we are developing a general-purpose tool, we are using a set of analyses that are representative of analyses used in the context of aerospace design. The studies conducted in chapter 2 were conducted on a relatively large number of high-dimensional datasets, which allowed us to

demonstrate a certain consistency in the improvement of predictive accuracy enabled by the proposed approach, but also to recognize limits of applicability of the proposed approach. The present study requires multi-fidelity datasets, that is observations from both LF and HF analyses modeling the same underlying phenomenon. As a result of this added requirement, only a subset of the datasets used in the previous studies are available: the elliptic PDE dataset and the RAE2822 dataset, that have been adapted to the multi-fidelity context as explained in the subsection *Test Datasets* below.

We expect multiple parameters to have an impact on the outcome of the comparison between the multi-fidelity and single-fidelity approaches. In addition to the parameters whose impact was studied in previous experiments, the multi-fidelity introduces new ones, such as the partition of the analysis budget into LF on one side and HF observations on the other side. This calls for a sensitivity study allowing to assess the effect of those parameters in order to better understand the conditions of applicability of the proposed approach, further discussed in the section *Sensitivity Study* below.

Like in other experiments, for every case of the DOE enabling the sensitivity study, the general workflow consists in 1) selecting training observation within the dataset, 2) training the model, and 3) computing the comparison metrics, here R^2. Once this is done for every case of the DOE created for the sensitivity study, the results can be plotted and interpreted. The creation of the graphs and the way they will be interpreted are discussed in the section *Presentation and Interpretation of the Results* below.

Test Datasets

The single-fidelity elliptic PDE and RAE2822 datasets were already discussed in the context of experiment 1.1 in section 2.2.3. The discussion mostly focuses on the changes made to the underlying analyses to obtain a low-fidelity version, and the resulting differences between LF and HF observations.

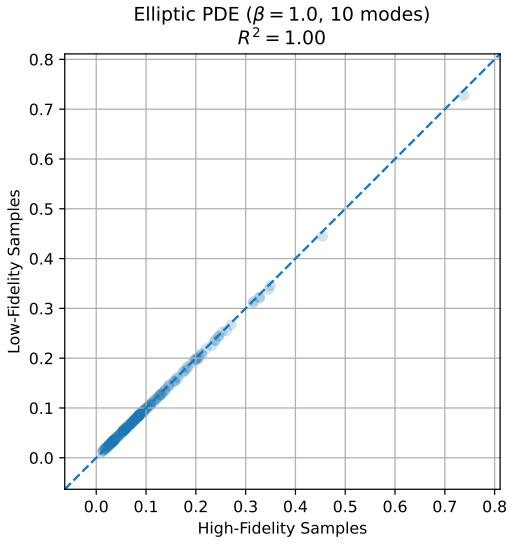
Elliptic PDE The analysis used to generate the elliptic PDE datasets in previous chapter has been adapted to to adjust its 1) level of fidelity, 2) input dimension, and 3) analysis "complexity", as discussed in section 2.2.3 when the elliptic PDE dataset was first introduced.

We adjusted the level of fidelity by varying the size of the grid on which the PDE underlying this analysis solved. Since the quantity of interest used as output is an integrated quantity, this does not raise further issues in the MF context. The analysis domain being a square, a single parameter corresponding to the length of the grid side has been introduced to adjust fidelity. The original analyses were carried out with a length of 100, resulting in a grid of 10,000 elements. For the LF version of the analysis, we aimed to approximately reduce the number of elements 100-fold in order to observe sufficient discrepancy between the LF and HF observations, which resulted in using a side length of 32 for the LF versions of the model.

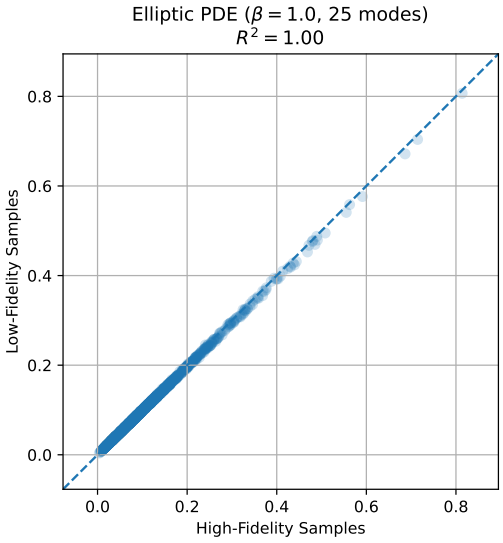
We also varied the number of inputs and "complexity" of the analysis. The input dimension can be varied by selecting the number of POD modes retained as input dimensions. The analysis complexity was varied using the physical length scale parameter β, a smaller β leading to the scalar outputs $f(\mathbf{x})$ depending on all input variables $\mathbf{x}$ instead of depending on a small subset of the x_i's. As in previous studies, the parameter β is set at two values of 1.0 (long length scale, less complex) and 0.01 (short length scale, more complex).

Comparisons of the LF and HF observations are shown in 3.3 and 3.4 for respectively the long and short length scales. This plot allows us to estimate the similarity between LF and HF observations, which, as discussed previously, is expected to have an impact on the performance of the proposed method. Points on the graph correspond to different locations in input space. For every point, its x-coordinate corresponds to the value of the HF observation (grid of side length 100), while its y-coordinate corresponds to the value of the LF observation (grid of side length 32).

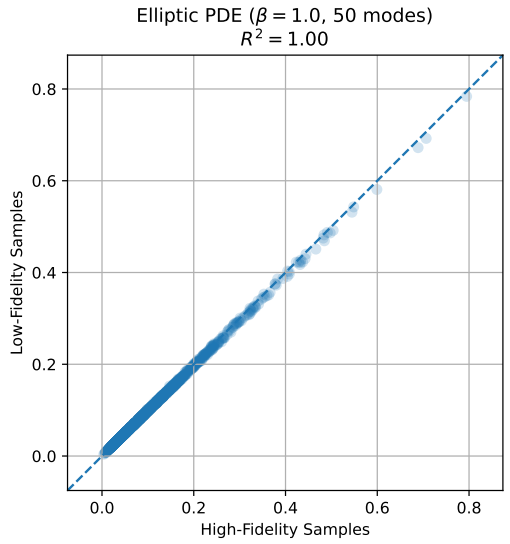
The short and long elliptic PDE datasets actually offer varying levels of similarity.



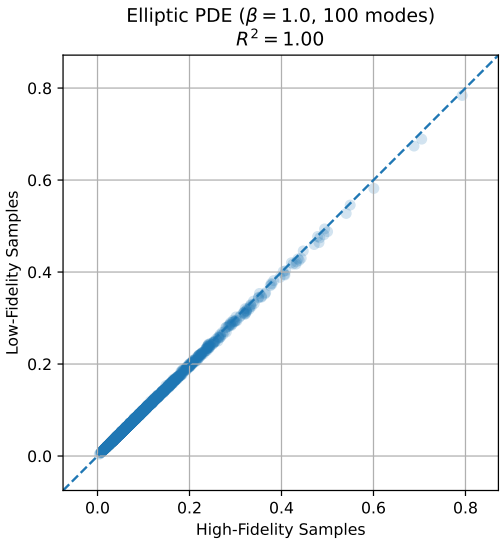
(a) Input dimension 10



(b) Input dimension 25

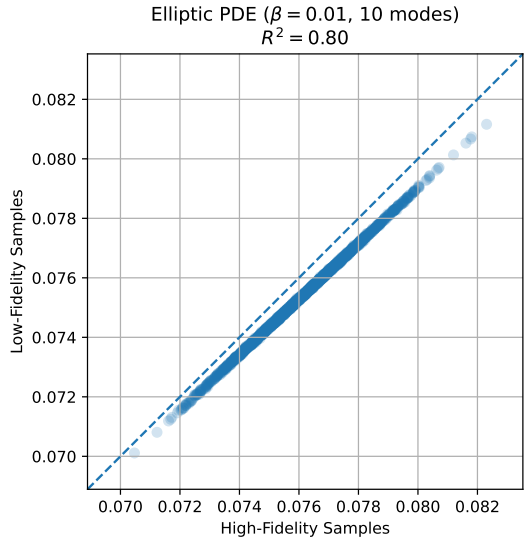


(c) Input dimension 50

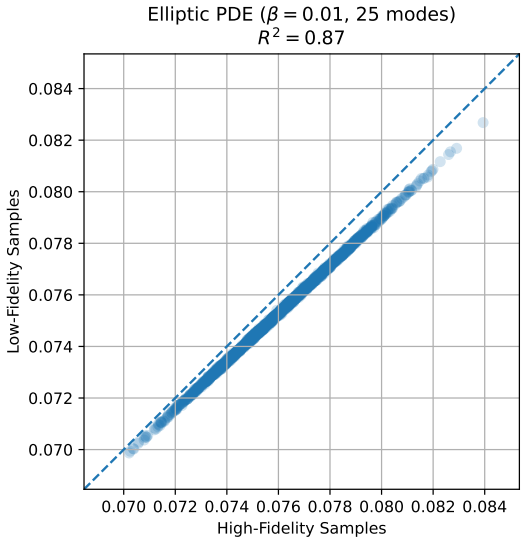


(d) Input dimension 100

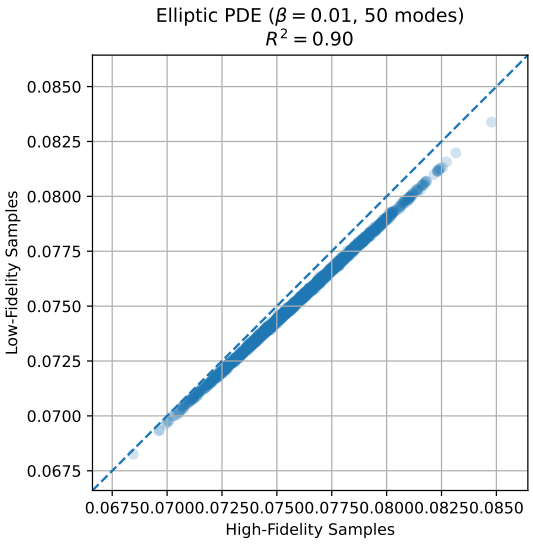
Figure 3.3: Comparison of the high- and LF data for the Elliptic PDE dataset with $\beta = 1.0$ and input dimensions 10, 25, 50, and 100.



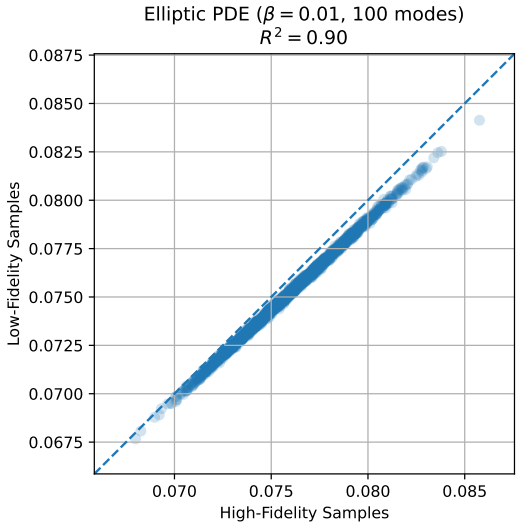
(a) Input dimension 10



(b) Input dimension 25



(c) Input dimension 50



(d) Input dimension 100

Figure 3.4: Comparison of the high- and LF data for the Elliptic PDE dataset with $\beta = 0.01$ and input dimensions 10, 25, 50, and 100.

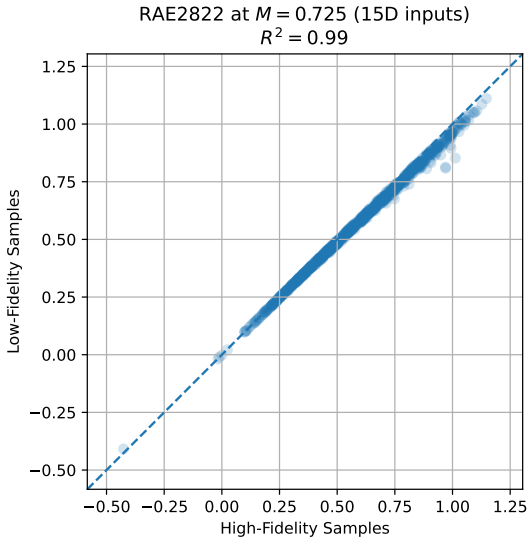
When rounded to the nearest two decimal points, the coefficients of determination for the long case are 1.00: the LF and HF observations are nearly identical, and the observations are accordingly almost aligned on the 45-degree line. This high level of similarity is important to keep in mind when interpreting results. On the other hand, the LF and HF observations are more dissimilar in the case of the short length scale, leading to coefficients of determination ranging from 0.80 to 0.90. We note that the relationship between these remains linear, although the proposed approach was selected to also allow non-linear relationships between the LF and HF observations.

RAE 2822 The RAE2822 datasets used in the context of this analysis use the same underlying analysis as those previously used and detailed in experiment 1.2. This data was generated using the SU2 CFD simulation software [161]. To create the LF version of the analysis, the size of the underlying mesh is reduced: the "baseline" HF version uses 41,796 CFD cells whereas the LF version uses only 10,560 CFD cells. The specifics regarding the generation of this dataset were discussed in section 2.3.3. Alternate versions of the analysis were also added, in which only 14 and 24 FFD control points are used instead of 50. All other analysis details remain the same, including the angle of attack being varied between $-2°$ and $+2°$. This allows to study the performance of the proposed MF approach on more intermediate input dimensions.

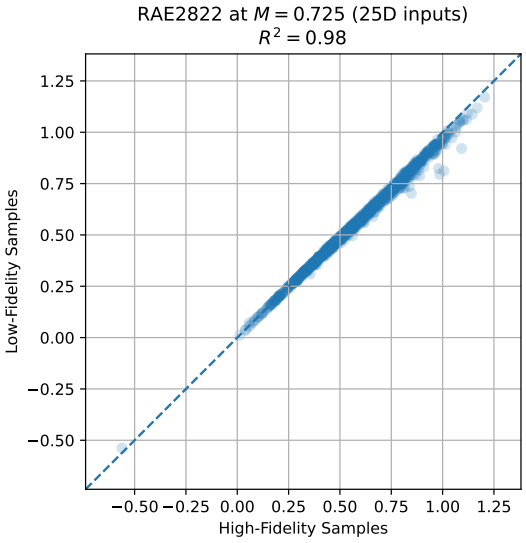
The comparison of the LF and HF observations are shown in fig. 3.5. For dimensions 10 and 25, the LF and HF observations are very similar but they differ considerably for the 51-dimensional case.

Implementation Details

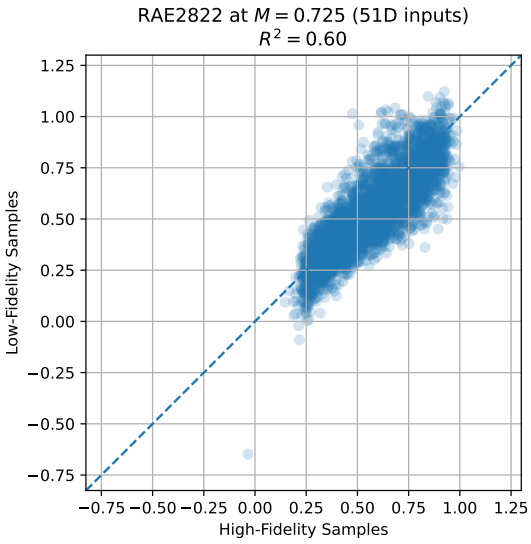
Multi-Fidelity Model As in previous experiments, the predictive model specified in section 3.1.2 is implemented using the numpyro framework [135, 133]. The built-in NUTS algorithm was used to perform MCMC, using an acceptance ratio of 0.8, a discarded chain



(a) Input dimension 10



(b) Input dimension 25



(c) Input dimension 51

Figure 3.5: Comparison of the high- and LF data for the RAE 2822 dataset with input dimensions 15, 25, and 51.

of 500 draws for initialization, and a posterior chain consisting of 1,000 draws. As for the previous experiments, these numbers have been used to keep the training time of the models reasonable in order to perform the parametric study. The prediction routines that take advantage of the known conditioned densities of the multivariate Gaussian distribution were implemented using Google's JAX framework [136].

Much of the code from the previous experiments was reused, as the MF model is a composition of multiple GPR models. Modifications include the chaining of the two GPs, which effectively results into uncertain inputs for the second HF GP. This has an impact on the implementation of both the probabilistic model used for training and the prediction routines, as this effectively adds an inner loop to the process. Other alterations include the more complex covariance structure, as detailed in section 3.1.2.

Construction of the Training Set The RAE2822 datasets were generated previously by ASDL colleagues for their own study and shared with the author. The elliptic PDE datasets were generated offline when initiating this experiment. Training observations are selected from the dataset during case execution using a random seed to ensure reproducibility. Different random seeds are used for training repetitions. Within a dataset, all samples that are not used for training are used to perform out-of-sample validation of the resulting surrogate model. The exact number of LF and HF training points is dictated by the LF allocation ratio discussed below (how much of the budget is allocated to LF vs HF observations), rounding down such that the actual analysis cost remains at or under the allowed analysis budget.

HPC and Other Details As for the previous experiments, great care has been taken to ensure the reproducibility of the results. The code was organized as a Python library with scripts leveraging this library that implement the actual experimental workflow. The resulting repository has been made available online[1]. Training duration for the proposed models ranges from minutes for the analyses with lower-dimensional inputs when using few train-

[1] https://gitlab.com/raphaelgautier/thesis_experiments_part3

ing observations to hours for the analysis with the higher-dimensional inputs when using many training observations. Since the proposed parametric study consists of thousands of cases, we leveraged the Georgia Tech's high-performance computing (HPC) environment [162] to run cases in parallel. This required the development of additional mechanisms to efficiently run the training and validation of the generated models, such as the ability for cases to continuously save their progress. This enables resuming jobs that were interrupted while running on the HPC cluster instead of restarting them from scratch, resulting in significant time gains. The workflows from previous experiments were adapted to the specifics of this experiment and improved when applicable. Case results are stored in individual file-based HDF5 databases.

Sensitivity Study

We do not expect the proposed multi-fidelity method to perform equally well in all situations. We aim to gather sufficient evidence that there exist situations in which it outperforms the single-fidelity approach, such that, if faced with a similar situation, we may consider it as a surrogate modeling option. In order to perform this assessment, we seek to account for the factors that are expected to have an impact on the performance of the proposed approach. These factors originate from 1) the underlying analysis, 2) the training set, and 3) the tuning of the method itself.

When it comes to the analysis, we expect its 1) input space dimension, 2) "complexity", and 3) degree of similarity between LF and HF versions to impact the performance of the proposed approach. Accordingly, the elliptic PDE dataset is used with two complexity settings discussed previously, as well as four different input dimension values (10, 25, 50, and 100 inputs). Multiple instances of the RAE2822 dataset were also created with different input dimensions (15, 25, and 51 inputs). As discussed previously, the LF and HF versions of these analyses exhibit different relationships: some are very similar while others display little correlation.

Regarding the training set, we expect multiple factors to play a role: 1) the total available analysis budget, 2) the relative budget allocation between LF and HF allocations, and finally 3) the particular choice of training locations. Because we expect to reach a better predictive accuracy with a lower analysis budget thanks to the multi-fidelity approach, the total analysis budgets considered in this experiment are lower than in the single-fidelity case. Instead of ranging from one to five times the number of inputs, we will rather focus on a range of one to two times the number of inputs, when expressing the budget in terms of equivalent HF observations. In previous experiments, the analysis budget could implicitly be defined as the number of training samples since a single analysis fidelity was used. In contrast, an additional degree of freedom comes with the introduction of a second level of fidelity: the ratio we use to split the training set between LF and HF observations. More details about how the analysis budget is handled in the multi-fidelity context is given in the paragraph *Analysis Budget, Allocation Ratio, and Analysis Cost* below. Finally, training is repeated five times to account for the particular choice of training observations, using a random seed to ensure reproducibility.

Finally, the parameters of the probabilistic method itself are expected to impact the outcome of the comparison. We have already identified the fact that 1) the FS dimension has a significant impact on the performance of the proposed approach in section 2.3. In addition, we know that 2) the choice of prior distributions has an impact for Bayesian methods. The approach proposed in section 2.3, where all possible FS dimensions are swept, is not practical in the context of a large parametric study such as the one conducted here. Therefore, only a limited number of alternate FS dimensions will be considered in this experiment. The prior distributions for the model parameters will however not be varied. They have been selected based on results from previous experiments and minimal hand-tuning. A complete study of their impact is deemed out-of-scope for this work, and would also prohibitively increase the cost of the parametric study. More details are included in section 3.1.3.

Ranges are defined for numerical parameters and a full-factorial DOE to is generated to study the impact of their variations. Table 3.1 summarizes the parameters varied in the parametric study conducted as part of experiment 2.1.

Analysis Budget, Allocation Ratio, and Analysis Cost As discussed previously, we focus on situations where the budget allocated to analyses is low relative to the analysis cost, resulting in few observations available for training the surrogate models. In the single-fidelity context, the whole budget is spent on the evaluation of the same analysis. In contrast, in the multi-fidelity context, the budget is split between the LF and HF observations. The relative allocation of the budget between the two levels of fidelity introduces a new degree of freedom. We will use the LF allocation ratio to represent this degree of freedom. We define it as the fraction of the budget allocated to LF evaluations, the rest of the budget being allocated to the HF observations. A LF allocation ratio of 0 corresponds to spending the entire budget on HF observations while a LF allocation ratio of 1 means exclusively using LF observations. Any number between 0 and 1 leads to a mix of LF and HF observations in different proportions.

We expect this ratio to have a significant impact on the accuracy of the resulting predictive model. Using only HF samples reverts to the original, single-fidelity, problem. Using only LF samples means that the prediction error would at best be the prediction error between the LF and HF models. Except under certain circumstances, such as very similar or dissimilar LF and HF observations, we do not expect any of these extremes to yield the best predictive accuracies. We also expect that unbalanced ratios (too few LF or HF training samples) will also lead to poor results as the GPR models that make up the multi-fidelity model individually require sufficiently many training points to make adequate predictions. Therefore, we expect to obtain the best results for relatively balanced mixes of LF and HF observations.

We also expect the optimal ratio to vary depending on the dataset and, for a fixed

dataset, on the total analysis budget. The determination of the optimal allocation ratio is out-of-scope of this work. Instead of setting it to a fixed value, which may lead to a misrepresentation of the predictive accuracy, we will be varying this parameter between 0 and 1 by increments of 0.2. We will therefore be able to crudely determine a near-optimal allocation ratio and assess the predictive performance of the proposed approach in this condition. As a byproduct, we will also be able to visualize the actual impact of the allocation ratio on predictive performance and check whether the expected behavior exposed in this paragraph is actually observed.

In the single-fidelity case, we simply expressed this budget as a number of observations, without the need to take into account the cost of analyses. In order express the allocation ratio in the multi-fidelity case, we need to introduce a measure of the analysis cost. We adopt the following approach to estimate it. During the generation of the validation set, which is done offline and only once, we record the time elapsed during each analysis run. We then compute the mean of the elapsed time and use this as a measure of analysis cost.

Summary of the Sensitivity Study Table 3.1 summarizes the parameters varied in the parametric study conducted as part of experiment 2.1.

Table 3.1: Experiment 2.1 – Summary of the parameters varied in the parametric study

Group	**Parameter**	**Parameter Values**
Analysis	Dataset	[elliptic PDE ($\beta = 1.0$), elliptic PDE ($\beta = 0.01$), RAE2822]
	Input Dimension	see tables 3.3 and 3.4
Training Set	Analysis Budget	see tables 3.3 and 3.4
	LF Allocation Ratio	[0, 0.2, 0.4, 0.6, 0.8, 1.0]
	Training Repetitions	5 repetitions with fixed random seeds
Method	FS dimension	see table 3.2

Table 3.2: Experiment 2.1 – FS dimension considered in the parametric study for each dataset.

Dataset	**FS dimension**
elliptic PDE ($\beta = 1.0$)	fixed at 3
elliptic PDE ($\beta = 0.01$)	[3, 5]
RAE2822	[1, 3, 5]

Table 3.3: Experiment 2.1 – Analysis budget (in HF samples) as a function of the analysis input dimension for the elliptic PDE datasets.

Input Dimension	**Analysis Budget (in HF samples)**
10	[4, 8, 12, 16, 20]
25	[10, 15, 20, 25, 30]
50	[20, 30, 40, 50, 60]
100	[70, 80, 90, 100, 110]

Presentation and Interpretation of the Results

As in previous experiments, the comparison of the predictive accuracy is quantified using the coefficient of determination (R^2), and this is the metric on which the comparison between the single-fidelity and multi-fidelity approaches will be based. For a given dataset, multiple parameters will be varied, making it challenging to use a single plot. One plot per dataset and FS dimensions is created. In each plot, the effect of both the LF allocation ratio and overall budget can be compared in terms of R^2.

For a single of these plots, i.e., given a dataset and an FS dimension, the LF allocation ratio varies along the x-axis. To the left of the plot, an LF allocation ratio of 0 means that we are considering a single-fidelity surrogate model trained exclusively using HF observations. To the right of the plot, an LF allocation ratio of 1.0 means that we are considering a single-fidelity surrogate model trained exclusively using *LF* observations. In-between, we are considering multi-fidelity surrogate models trained using different ratios of LF to HF

Table 3.4: Experiment 2.1 – Analysis budget (in HF samples) as a function of the analysis input dimension for the RAE 2822 datasets.

Input Dimension	**Analysis Budget (in HF samples)**
15	[10, 15, 20, 25, 30, 40, 50, 60]
25	[10, 15, 20, 25, 30, 70, 80, 90, 100]
51	[20, 30, 40, 50, 60, 100, 150, 200]

observations.

On a single plot, the different colored lines show the results for different total analysis budgets. The budget, expressed in terms of equivalent HF samples, is indicated in the legend. Moving on the line from left to right means increasing the portion of the budget allocated to LF observations. Since the total analysis budget remains constant, this effectively means "trading" HF observations for the corresponding number of LF observations.

Because training is repeated to account for the specific choice of training observations, multiple values for R^2 are obtained, given an LF allocation ratio and a total analysis budget. The spread of the R^2 is represented using a shaded region that represents the 95% confidence interval of the distribution of the R^2 values.

For results to support the hypothesis, there should exist an LF allocation ratio at which the predictive accuracy of the multi-fidelity model is greater than either of the single-fidelity models – consistently across training repetitions. Visually, this means that we expect to see a bell-shaped curve. Going from left to right, the predictive accuracy should first increase as we start adding LF observations to the training set, until a point where the optimal LF to HF allocation ratio is reached, and then decrease until the training set is only made up of LF observations.

3.1.4 Results of Experiment 2.1

The results are presented in figs. 3.6 to 3.8 for the different test datasets. In these graphs that depict the evolution of the coefficient of determination R^2 as a function of the LF allocation ratio, each colored line corresponds to a different analysis budget. As explained in the previous section, analysis budgets are expressed in equivalent HF samples: under an analysis of 10 HF samples, 10 HF samples would be used for an LF allocation of 0.0. As the LF allocation ratio increases, HF samples are "traded" for more LF ones. The shaded areas in the plots correspond to the 95% confidence interval over the training repetitions.

Elliptic PDE Datasets - $\beta = 0.01$

Figure 3.6 depicts the evolution of the coefficient of determination R^2 as a function of the LF allocation ratio for different analysis budgets for the elliptic PDE datasets with $\beta = 0.01$. Only a 3-dimensional feature space has been considered in this case. The expected evolution of the R^2 is visible in all instances, even though the 100-input case exhibits a more erratic behavior.

Let us start by discussing the 10, 25-, and 50-dimensional cases that exhibit similar results. The smallest analysis budgets exhibit low predictive accuracies that monotonously increase with the LF allocation ratio. This corresponds to a situation where the analysis budget is so low that the limited analysis budget is better spent using LF samples only. As the analysis budget increases, the situation improves: even though it an LF-only training set may lead to higher predictive accuracies, we observe a peak for an intermediate LF allocation ratio, and this peak generally allows to reach a better predictive accuracy than if HF samples had been exclusively used. Finally, for the highest considered analysis budgets, which remain low relative to the dimensions of the problems considered, we observe the expected behavior: the maximum predictive accuracy is obtained for an intermediary allocation ratio, and the value of R^2 is higher than if a single-fidelity model had been used.

As discussed previously, we also observe a dip in the predictive accuracy as we switch

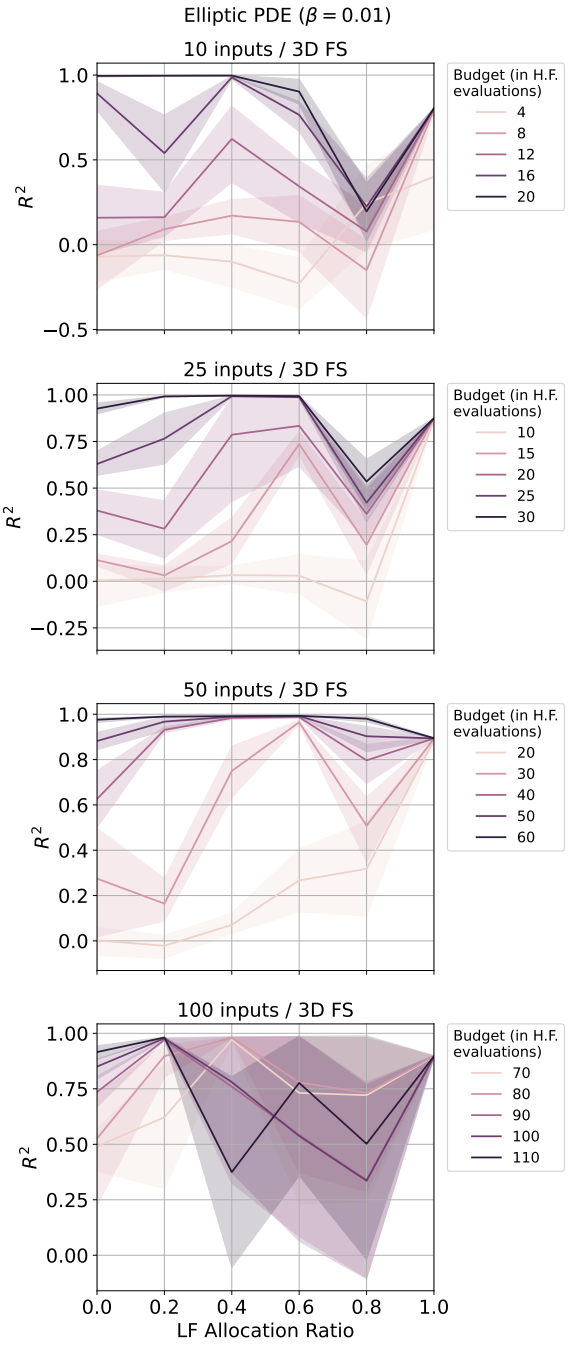


Figure 3.6: Evolution of the coefficient of determination R^2 as a function of the LF allocation ratio for different analysis budgets for the elliptic PDE with $\beta = 0.01$. Individual plots map to a combination of input space and FS dimensions. Input space dimension increases from top to bottom. Feature space dimension increases from left to right.

from a single-fidelity model to a multi-fidelity model. This can be observed when switching from a HF-only to a multi-fidelity model for the following cases:

- 10 inputs / 3D FS: budgets of 12 and 16 HF samples,
- 25 inputs / 3D FS: budgets of 15 and 20 HF samples,
- 50 inputs / 3D FS: budgets of 30 HF samples.

All of these drops in accuracy happen at an LF allocation ratio of 0.2, corresponding to a rather unbalanced dataset. In a symmetric fashion, predictive accuracy sinks for an an LF allocation ratio of 0.8, corresponding again to an unbalanced dataset, but this time more in favor of the LF samples. This happens because both components of the Bayesian ridge deep multi-fidelity Gaussian process (B-R-DMF-GP) individually need enough training data. All models coalesce to the same point with very little variation across training repetitions when the LF allocation ratio equals 1.0, corresponding to a single-fidelity model exclusively trained using LF models. While it may be surprising that this happens irrespective of the analysis budget, we recall that each LF observation consumes a lesser fraction of the analysis budget, and spending the whole budget on LF observations accordingly results in a large dataset: even small budgets in terms of HF samples lead to a relatively large LF-only training dataset As a consequence, The predictive accuracy that we observe here is therefore very close to the R^2 displayed in fig. 3.4: we consistently obtain an accurate surrogate model of the inaccurate LF process.

The case of the 100-dimensional input space exhibit significantly different results. The first obvious observation is the amplitude of the variations among the training repetitions. While the previous cases did exhibit some variation, it remained quite limited. Variation occurs for allocation ratios greater than 0.2. As already observed for the single-fidelity model in previous chapters, the training of the model becomes increasingly difficult with the input space dimension, as the number of model parameters increases accordingly. High accuracy variations across repetitions would be consistent with this explanation: it's not

that accuracy is always low, but rather that it is erratic. Despite these large variations for allocation ratios greater than 0.2, we still observe a maximum in predictive accuracy: it is reached at an allocation ratio of 0.4 for the budgets of 70 and 80 HF samples while it is reached at an allocation ratio of 0.2 for budgets at and above 90 HF samples.

The problem remains that we cannot tell a priori what is a good LF allocation ratio, and we expect it to vary from one dataset to another: we can actually already see that the evolution of the accuracy at the maximum is different for the different input dimensionalities. In some cases, the peak is well-defined, while in other instances, it is flatter. For intermediate analysis budgets, it also appears that the optimal LF allocation ratio increases with the input space dimension.

Elliptic PDE Datasets - $\beta = 1.0$

Figure 3.7 depicts the evolution of the coefficient of determination R^2 as a function of the LF allocation ratio for different analysis budgets for the elliptic PDE datasets with $\beta = 1.0$. This time, two possible feature space dimensions were considered: $m = 1$ and $m = 3$, accordingly with the results shown in chapter 2. This time, it is clearly visible that a 1-dimensional feature space is inappropriate: the predictive accuracy for the multi-fidelity model generally remains below the predictive accuracy that is obtained using any of the low and high fidelities only. We will therefore focus on the discussion of the results obtained using a 3-dimensional feature space.

Although the results appear similar to those obtained previously when $\beta = 0.01$, there are visible distinctions. We recall that the case the parameters β sets the correlation length for the input perturbations used, therefore a smaller value of β corresponds to more rapid variations, leading to physics that are more complex to capture. The main difference with the previous case is the higher similarity between LF and HF observations: this was first observed in fig. 3.3, where the R^2 values equal 1.00. As a consequence, this is challenging to do better using multi-fidelity samples than it is using LF samples only. The fact that the

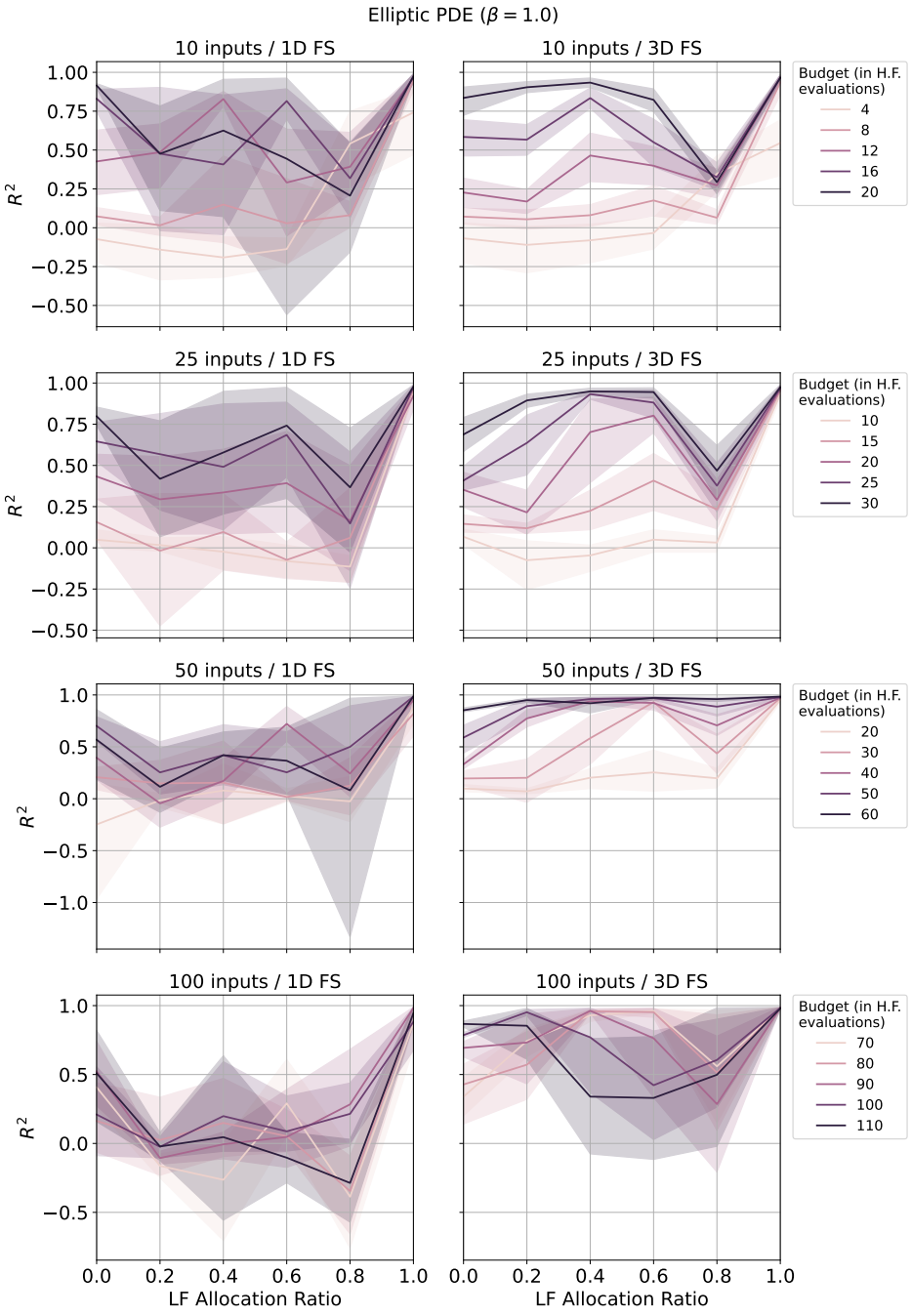


Figure 3.7: Evolution of the coefficient of determination R^2 as a function of the LF allocation ratio for different analysis budgets for the elliptic PDE with $\beta = 1.0$. Individual plots map to a combination of input space and FS dimensions. Input space dimension increases from top to bottom. Feature space dimension increases from left to right.

optimal LF allocation lies somewhere between 0.0 and 1.0 hints at the trade-off between the lower cost, and therefore greater abundance, of LF samples, and the higher accuracy of the HF samples. If the LF and HF observations are too similar, there is no trade-off: we are better off using LF samples only. This is what we are observing here, even though we do observe a extremum for intermediate LF allocation ratios: the peak value of R^2 always remains inferior to the R^2 using LF samples only.

Beyond this fact, most of the remarks made previously apply here. Predictive accuracy consistently increases with analysis budget, except in the 100-dimensional case. Unbalanced multi-fidelity datasets still result in a dive in terms of predictive accuracy Significantly more variations across training repetitions is observed for the 100-dimensional case, especially for the larger analysis budgets.

RAE 2822 Dataset

Figure 3.8 depicts the evolution of the coefficient of determination R^2 as a function of the LF allocation ratio for different analysis budgets for the RAE 2822 dataset. This time, 1-, 3-, and 5-dimensional feature spaces were considered and a more analysis budgets were considered, especially in the higher range in order to tentatively observe an asymptotic behavior. Overall, the proposed multi-fidelity approach does not bring any improvement over the single-fidelity approach for these datasets, but the reason for it is different depending on the dataset.

In the case of the 15- and 25-dimensional input spaces, the predictive accuracy of an LF-only model remains always higher than any multi-fidelity model, even in instances where an extremum is visible in the multi-fidelity region. For the highest analysis budgets, we see that the multi-fidelity approach may even become detrimental: R^2 decreases monotonously with the LF allocation ratio until it rebounds for the LF-only model. As a result, multi-fidelity models perform worse than both the LF-only and HF-only cases. This is to be expected: if the analysis budget is high enough, then sufficiently many HF training samples

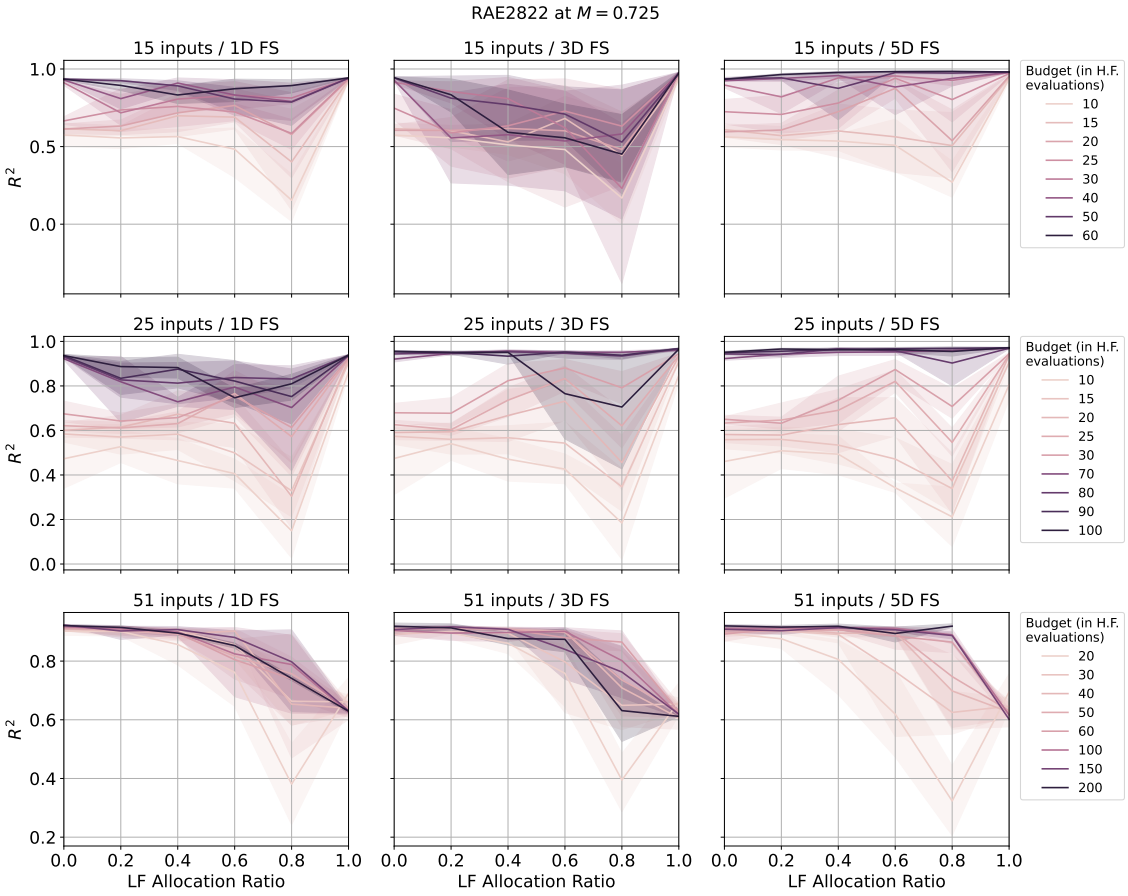


Figure 3.8: Evolution of the coefficient of determination R^2 as a function of the LF allocation ratio for different analysis budgets for the RAE2822 dataset. Individual plots map to a combination of input space and FS dimensions. Input space dimension increases from top to bottom. Feature space dimension increases from left to right.

can be obtained that lead to a single-fidelity model with adequate predictive accuracy. The multi-fidelity approach was only expected to bring an edge when the analysis budget is so low that HF samples only would not lead to an adequate HF model. For intermediate analysis budgets, we do see an improvement over HF-only models, but not over the LF-only ones. The same reason as for the elliptic PDE with $\beta = 1.0$ may be used here: the LF and HF observations are similar enough to each other, as shown in fig. 3.8, such that it is more beneficial to use LF training samples only.

In the case of the 51-dimensional input space, we observe a new behavior: R^2 decreases monotonously as the LF allocation ratio increases from 0.0 to 1.0, and the accuracy rebound for the LF-only model is not systematic. The situation observed here is opposite to the one exhibited by lower-dimensional datasets discussed previously: the LF and HF may be too dissimilar from each other, and including LF samples only makes the predictive accuracy of the multi-fidelity model worse.

3.1.5 Conclusion

The results discussed in this section show that the performance of the proposed approach greatly differs from one dataset to another. We hypothesize that this may be explained by the relationship between the LF and HF observations. For the elliptic PDE dataset with $\beta = 1.0$, as well as the RAE2822 datasets with 15 and 25 inputs, we observed that better results were obtained when using LF samples only. This is consistent with the similarity between the two fidelities: when LF and HF observations are virtually identical, we expect the best model to be obtained when the most observations are used to train the model, that is when all the budget is allocated to LF observations. On the other end of the spectrum, we saw that the LF/HF relationship in the case of the RAE2822 dataset with 51 inputs was particularly noisy. In that case, the best results are obtained when only HF samples are used for training, as the LF observations are uninformative. The benefit of the multi-fidelity approach was observed for the elliptic PDE dataset with $\beta = 0.01$, where the LF and HF

observations are visibly distinct but correlated. In this case, leveraging both LF and HF samples appears to be beneficial.

We also observed the importance of selecting the right FS dimension: while the proposed MF approach may be beneficial when the relevant dimension is used, it may become detrimental otherwise. Another observation concerns the selection of the LF allocation ratio: when the conditions for the LF/HF and the FS dimension are met, it still remains to select a suitable mix of LF and HF observations. The right fraction appears to be case-dependent, as the maximum predictive accuracies were reached at different LF allocation ratios depending on the input dimension and the total analysis budget. These observations lead to the following conclusion regarding hypothesis 2.1:

Conclusion on Hypothesis 2.1

For the same total analysis budget, the multi-fidelity surrogate models obtained using the proposed approach have higher predictive accuracy than the corresponding LF-only and HF-only single-fidelity models if:

- an appropriate FS dimension is used;
- an appropriate LF allocation ratio is used;
- the LF and HF analyses are neither too similar or too dissimilar;
- the analysis budget is neither too low nor too high.

3.2 Different Low- and High-Fidelity Feature Spaces

3.2.1 Background and Research Objective

In section 3.1, the projection matrix onto the FS was shared by both the LF and HF layers of the B-R-DMF-GP model. The choice of sharing the FS was motivated by the notion of "transfer learning": we expect that the relative abundance of low-fidelity observations

allows to identify a FS that is also well-suited to model the HF mapping. Based on the results obtained in section 3.1.4, this approach yielded encouraging results. However, other options may be available that allow transfer learning to happen without being as restrictive.

Previous approaches to multi-fidelity AS discussed in section 3.1 have different takes on the FS's at different fidelity levels. In [151], individual AS's are computed for each fidelity: the multi-fidelity aspect only comes into play in the target control variates application. In [109], both LF and HF samples are used together to derive a shared AS for both fidelities. This approach is also used in [152]. In [153, 154], the AS is determined using LF observations only using a gradient-free method, and used for both fidelities in the multi-fidelity surrogate model. In [155, 156], the AS is found using gradient evaluations of the HF analysis only to create an approximation of the HF analysis by a 1D ridge function that is used as a low-fidelity model. In [131], the same FS is used for all fidelities as part of their modified ARGP scheme. Most relevant literature therefore uses the same FS for both levels of fidelity. To our knowledge, the question of how the LF and HF FS's should be related has therefore not been considered in previous works, raising the following gap:

Gap 2.2

The relationship between the FS respectively used for the LF and HF parts of an MF model based on approximation by ridge functions is a modeling choice whose impact has not been studied in the literature.

This leads to the following research question:

Research Question 2.2

How should the respective FS's of the LF and HF layers of the proposed MF model be related?

In this section, we propose to investigate alternative strategies for handling FS's that are nto necessarily identical in the LF and HF layers of the B-R-DMF-GP.

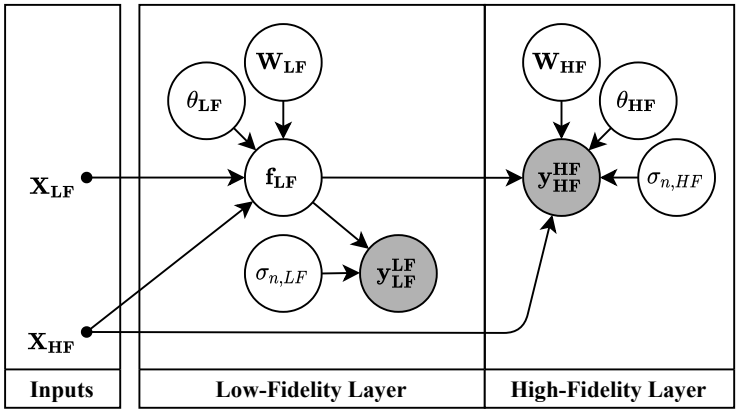


Figure 3.9: Probabilistic graphical model for *different* alternative.

3.2.2 Formulation of the Alternative Approaches

The first alternative to be considered is the one previously shown in fig. 3.2: the same projection matrix $\mathbf{W}$ is used by both layers of the B-R-DMF-GP. Beyond the motivation of "transfer learning", this approach is beneficial in that it minimizes the number of model parameters to be inferred. As initially discussed in chapter 2, the projection matrix is parameterized by many parameters, and Bayesian inference becomes increasingly difficult as the number of parameters increases. We refer to this alternative as the *shared* approach.

A second natural alternative is to use completely different FS's for each fidelity level. In this case, we do not expect any transfer learning between the two fidelities to happen through the knowledge of the FS, and it increases two-fold the number of projection parameters. We do not expect this alternative to perform well, but rather see it as a way to confirm that sharing the FS indeed plays a role in the improved performance of the proposed MF approach. We equip each individual projection matrix with the same uniform prior distribution in the Stiefel manifold as used previously. The probabilistic graphical model (PGM) for this alternative is shown in fig. 3.9 and we refer to this alternative as the *different* approach.

In the third and last alternative we consider, we propose a mechanism to enforce some dependence between the FS's of the LF and HF layers. In a Bayesian framework, it is

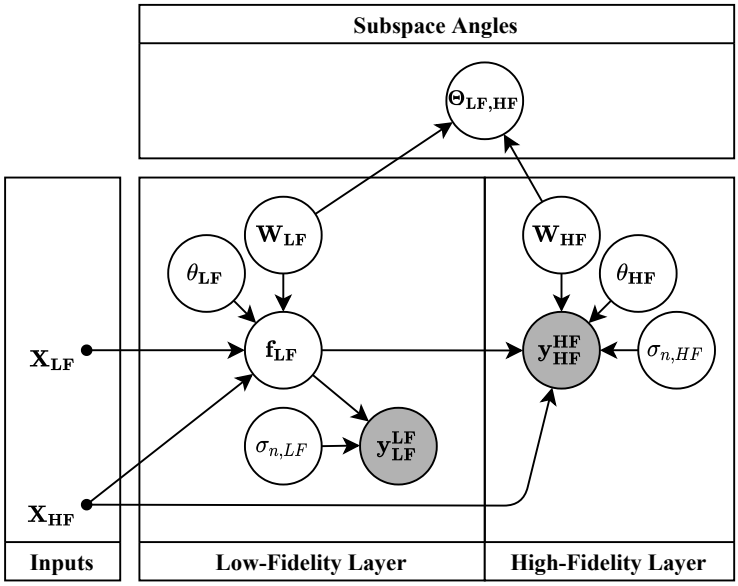


Figure 3.10: Probabilistic graphical model for the *related* alternative.

challenging to simply constrain model parameters to certain values, as their distribution is obtained from the structure of the probabilistic model and observed data. However, the Bayesian framework also offers a way to encode prior belief inside the model through the use of prior distributions. Here, we aim to translate the belief that the LF and HF FS's are not too different. In chapter 2, we used subspace angles metric to measure the similarity between different subspaces, ans we propose to leverage this metric here. In particular, we propose to encode the belief that the subspace between the LF and HF FS's is small. We achieve this by adding the subspace angles as a latent variable denoted $\mathbf{\Theta_{LF,HF}}$ in the probabilistic model and equipping it with a half-normal distribution with scale parameter of 0.4, thus *a priori* favoring small subspace angle values between FS, except if strong evidence from the data suggests otherwise. The resulting PGM for this approach is shown in fig. 3.10 and we refer to this alternative as the *related* approach.

Using two different FS for each layer is a generalization of the approach in which the FS is shared. In other words, the shared FS is a special case of the more general case in which different FS are used for each layer. Therefore, with sufficient training data, we would expect the predictive model to perform at least as well when different FS are used

compared to when a single shared FS is used. However, because the model has been made more flexible, this also increases the difficulty of determining meaningful model parameters when training data is sparse, which is the setting in which we operate. The following paragraph explains exactly what makes the training of the model more challenging when two different FS are used instead of a single one.

As illustrated in the PGM shown in fig. 3.2, when the FS is shared across the layers of the model, then varying the shared FS has an impact on the likelihoods of both LF and HF observations. In other words, the process by which we learn the posterior distribution of the FS is affected by both LF and HF observations. Therefore, all available observations are used to learn about the underlying structure of the analyses under consideration via the FS. On the other hand, as illustrated in figs. 3.9 and 3.10 for the two alternative approaches considered in this experiment, varying the HF FS would only impact the likelihood of the HF observations. Similarly, varying the LF FS would mainly impact the likelihood of the LF observations. To a lesser extent, varying the LF FS also has an impact on the likelihood of the HF observations, since LF outputs are passed as inputs to the HF layer. However, we expect this relationship to have less impact on the likelihood of HF observations than the direct impact of varying the HF FS. In summary, we moved from a situation in which all observations were used to learn a shared FS to a situation where 1) two FS have to be learnt, therefore doubling the number of model parameters being inferred from the same data, and 2) each FS is only (for the HF FS), or mostly (for the LF FS), learnt using the training data of the corresponding fidelity, instead of fully benefiting from the training data at both levels of fidelity.

In terms of training difficulty, we expect two corresponding consequences: 1) more training data will be needed in order to identify relevant FS, due to the added number of model parameters, and 2) it will be become more crucial to balance the number of LF and HF observations to be able to learn their respective FS, since each fidelity's FS is mostly learnt using the corresponding fidelity's training data. In the context of this thesis, we focus

on the low-data regime, in which analysis budget is limited and training observations are accordingly sparse. We expect the added flexibility of the model to be mostly detrimental in this setting. However, if more training data is available, as discussed above, we would expect the case in which different FS are used to lead to at least as good surrogate models as when the shared FS is used. The stage at which training data would become sufficient to properly train the model is a priori unknown, and it may or may not lie within the range of – relatively low – analysis budgets considered in the context of this experiment. A study that would verify the expected behavior and determine the amount of training data required to reach comparable or better predictive accuracy using the "different" approach than what was achieved using the "shared" approach is out-of-scope of the present work that focuses on the low-data regime.

Hypothesis 2.2

Under a limited computational budget, better predictive accuracy is obtained when using the same feature space for both layers of the proposed multi-fidelity model rather than different feature spaces for each layer because it introduces fewer model parameters and combines observations from all levels of fidelity to learn the feature space.

3.2.3 Setup of Experiment 2.2

Need for Experiment 2.2

We have shown that, under certain conditions, the proposed MF approach allows to reach higher predictive than if all the budget was spent building a single-fidelity surrogate model using exclusively LF or HF observations. However, we had not considered the case where different feature spaces are used for each layer of the surrogate model. By using different feature spaces for each layer of the model, we effectively make it more flexible, thus better able to fit training observations. However, adding flexibility also adds complexity

to the model, in the form of additional training parameters that need to be identified with the limited training data at our disposal in the low-data regime. Moreover, this structural modification has an another impact: while observations of both fidelities came into play into the process of learning the shared FS, the learning of the separate FS's is mostly driven by observations of the corresponding fidelity level only. In the low-data regime, this is expected to be detrimental. Experiment 2.2 aims at confirming this. To this end, we need to show that the predictive accuracy of the resulting surrogate is higher when sharing the feature space.

Experiment Design

We have laid out different options for handling the FS's of the different levels of the MF model and experiment 2.2 aims at verifying that, in the context of the low-data regime, the option initially selected and studied in experiment 2.1 was the right one. Therefore, we will be comparing the different options. As in previous experiments, because our end goal is to produce a predictive model that accurately reproduces the original analysis, this comparison will be carried out on the basis of predictive accuracy, as measured by R^2.

Because we are operating in the same context as experiment 2.1, the rest of the experiment design is similar. We will also be working on a set of representative analyses. In order to enable the comparison, surrogate models are trained according to each of the methods under consideration and compute the corresponding out-of-sample R^2, thus making the quantitative comparison between the different options possible.

The source code used to generate the results presented thereafter has been made publicly available[2].

[2]https://gitlab.com/raphaelgautier/thesis_experiments_part3

Sensitivity Study

The sensitivities studies are the same as in experiment 2.1. The elliptic PDE datasets with two levels of complexity and four input dimensions are used to assess the alternative types of FS relationships under consideration. As it is expected to impact the surrogate model's accuracy, the total analysis budget is varied, and training is repeated five times to account for the particular choice of training observations, using a random seed to ensure reproducibility. Other parameters that are expected to have an impact and are varied are the LF allocation ratio and the FS dimension. The different FS relationships discussed previously are considered: shared, related, and different FS's. A full-factorial DOE is generated to study the variations of all parameters (see table 3.5).

Table 3.5: Experiment 2.2 – Summary of the parameters varied in the parametric study

Group	**Parameter**	**Parameter Values**
Analysis	Input Space Dimension	[10, 25, 50, 100]
	Complexity (parameter β)	[0.01, 1.0]
Training Set	Analysis Budget	see table 3.6
	LF Allocation Ratio	[0, 0.2, 0.4, 0.6, 0.8, 1.0]
	Number of Training Repetitions	5 repetitions with fixed random seeds
Method	FS dimension	[3, 5]
	LF/HF FS relationship	[shared, related, different]

Presentation and Interpretation of the Results

We need to be able to compare the predictive accuracy obtained when the FS is shared and when it is different between the two layers of the multi-fidelity model. Like in experiment 2.1, we noted that the total analysis budget and the LF allocation ratio are still factors that we expect to have an impact on the performance of each method considered here. There-

Table 3.6: Experiment 2.2 – Analysis budget (in HF samples) as a function of the analysis input dimension.

Input Dimension	**Analysis Budget (in HF samples)**
10	[4, 8, 12, 16, 20]
25	[10, 15, 20, 25, 30]
50	[20, 30, 40, 50, 60]
100	[70, 80, 90, 100, 110]

fore, we will be using the same kinds of plots as in experiment 2.1, in which the x-axis is used to vary the LF allocation ratio and different lines correspond to different total analysis budgets. As in the previous experiment, shading is also used to represent the variability across training repetitions. The results for the three types of LF/HF relationships will be shown side-to-side: the "shared" alternative on the left, the "related" alternative in the middle, and the "different" alternative to the right. For compactness, the plots corresponding to the same complexity parameter β will be grouped together, with the input dimension increasing from top row down to bottom row.

For a given dataset and a given training budget, for the results to support hypothesis 2.2, the predictive accuracy in the "shared" case should be larger than in both other cases. In concrete terms, this means that the line representing the evolution of R^2 in the shared case (left column) should remain above the line of the same color in the other other cases (center and right columns).

3.2.4 Results of Experiment 2.2

Figures 3.11 and 3.12 depict the evolution of the coefficient of determination R^2 as a function of the LF allocation ratio for the elliptic PDE datasets with respectively $\beta = 0.01$ and $\beta = 1.0$. Each individual plot corresponds to a different input dimensions of the elliptic PDE dataset and to a different FS relation. In the following, we discuss the results for all test datasets at once since they all exhibit the same behavior. Because we are focusing on

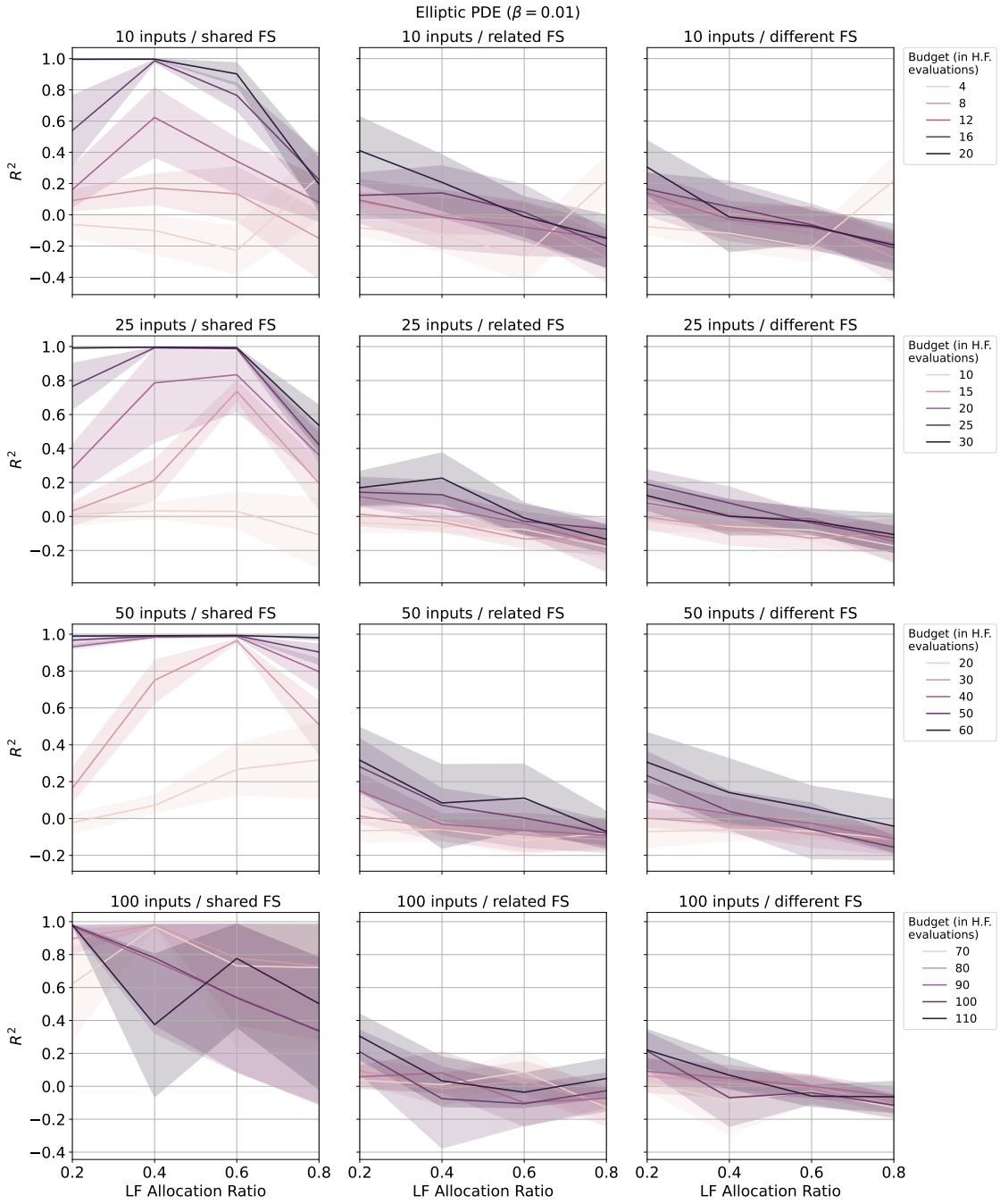


Figure 3.11: Evolution of the coefficient of determination R^2 as a function of the low-fidelity allocation ratio for different analysis budgets for the elliptic PDE dataset with $\beta = 0.01$. Individual plots map to different combinations of input space dimension and FS relation. Input space dimension increases from top to bottom. Leftmost column corresponds to shared FS, middle column to "related" FS, and rightmost to different FS.

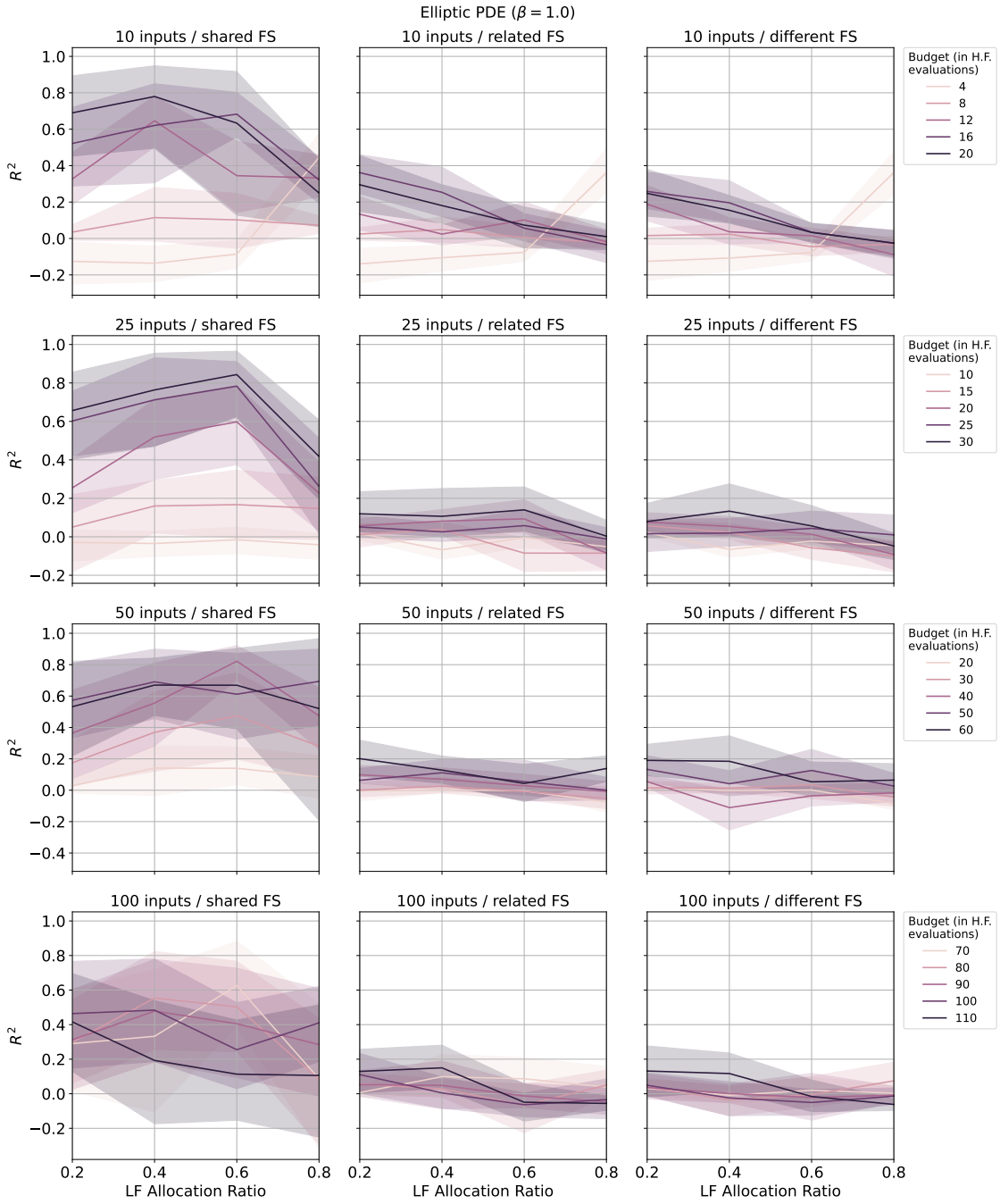


Figure 3.12: Evolution of the coefficient of determination R^2 as a function of the low-fidelity allocation ratio for different analysis budgets for the elliptic PDE dataset with $\beta = 1.0$. Individual plots map to different combinations of input space dimension and FS relation. Input space dimension increases from top to bottom. Leftmost column corresponds to shared FS, middle column to "related" FS, and rightmost to different FS.

the relationship between the FS and HF feature spaces, we only consider truly multi-fidelity models. Therefore, LF allocation ratios of 0 and 1.0 have been omitted from the plots. We observe very low predictive accuracies over the board for the alternate FS relations under consideration in this experiment, and the expected bell-shaped behavior is also not visible. The results clearly show that the options considered for handling different FS for the LF and HF models are inadequate, and sharing the FS still appears as the best option.

These observations are compatible with the explanation given previously. Because of each FS is for the most part learnt using the corresponding fidelity's training data, and because we are operating under very limited training budgets (up to twice the number of inputs for the 10D case, but only up to 1.2 times the number of inputs in the other cases, when the budget is expressed in terms of equivalent HF evaluations), we do not encounter a situation in which both levels of fidelity simultaneously have enough training data to learn their respective FS. A study that would determine the analysis budget needed to reach a predictive accuracy at least as good as when the FS is shared is out-of-scope, as we are specifically on low analysis budgets.

3.2.5 Conclusion

In section 3.1, we observed that the proposed multi-fidelity approach improves predictive accuracy for a given analysis budgets under certain circumstances when the same feature space is used for both models. Using the same feature space for both fidelities as a first step appears natural: it is simple and it forces sharing information across the fidelities. Other options for handling the possibly different FS had not yet been investigated. In this section, we have been considering two of these alternate solutions. The motivation for using different FS is that it gives more flexibility to the model: if the LF and HF FS do not match in reality, this would allow them to differ. However, added flexibility is also expected to have an impact on the training stability: the more parameters need to be inferred, the higher the chance for over-fitting, and the harder the MCMC inference. The motivation behind the

"related" option was to offer an intermediate solution: providing some flexibility without forcing both FS to be the same. Given the high number of new parameters introduced by either alternate approach, we thought more likely that these approaches would decrease predictive accuracy. Therefore, the hypothesis we have been testing is that using the same feature space is beneficial.

In the considered test cases, we saw that the models that do not share the FS between their LF and HF components perform poorly: their coefficient of determination remains below 0.6 irrespective of the analysis budget within the considered range and irrespective of the LF allocation ratio. There was no significant difference in behavior between the "different" and "related" under consideration. This may be explained by the fact that, in this case, the added flexibility was more detrimental than beneficial to the model. This leads to the validation of hypothesis 2.2:

Conclusion on Hypothesis 2.2

Using the same feature space for both the low- and high-fidelity components of the proposed multi-fidelity approach leads to better predictive models.

Some avenues for future work follow from the observations made in this section. Only two options for handling the different FS have been considered here. As future work, other options may be found that succeed at bringing some flexibility with limited detrimental effects to model training.

3.3 Conclusion

3.3.1 Summary

We started this chapter with a brief review of multi-fidelity surrogate modeling, and, in particular, multi-fidelity approaches to approximation by ridge functions. However, we observed that existing methods either require gradient evaluations, or are based on MLE approaches that are challenging to apply when only few analysis observations are available.

This led to the identification of the gap motivating the research question 2.1 regarding the extension of the method proposed in section 2.2.2. Because the proposed single-fidelity approach relied on GPs, we returned to the literature to look at existing GP-based approaches to surrogate modeling that could serve as a foundation. We identified two options, ARGP and DMF-GP. The comparison of the two methods led to the selection of DMF-GP as a base method since it allows for a more flexible relationship between the LF and HF versions of the analyses. From there, we introduced the proposed MF method, B-R-DMF-GP, that combines approximation by ridge functions with deep multi-fidelity Gaussian process regression and that is trained in a fully Bayesian fashion. After presenting the experimental setup employed to assess the proposed method, we discussed the results obtained, and showed that, in certain circumstances, the proposed approach can yield better results than a single-fidelity approach for the same total analysis budget.

In the second part of this chapter, we focused on one specific aspect of the proposed multi-fidelity approach: the relationship between the projection matrices, and therefore FS, used in the LF and HF layers of the proposed MF method. In the initial assessment made in section 3.1, the matrices were chosen to be identical, motivated by the notion of "transfer learning": we expect the more numerous LF observations to accelerate the identification of a relevant FS and therefore help reaching higher predictive accuracies with a smaller overall training budget. We considered two alternative approaches to handling the LF and HF FS's: 1) using totally different projection matrices, or 2) enforcing some similarity between the matrices using a prior distribution on their subspace angles. The motivation for these alternate approaches is to allow some flexibility in the search of the different subspaces. However, we expected both approaches to reduce the amount of transfer learning, possibly to an extent that would be detrimental to the resulting surrogate model's predictive accuracy. This is what we observed: the added flexibility made training these surrogate significantly harder that was not compensated for by the added flexibility. Incidentally, this study confirmed the initial choice of using the same FS for both layers of the proposed

B-R-DMF-GP model.

3.3.2 Contributions

The first contribution of this chapter is the new multi-fidelity approach to surrogate modeling for analysis with high-dimensional inputs. Compared to methods previously proposed in the literature, it does not require access to gradients and follows a fully Bayesian approach, thus enabling a full quantification of epistemic uncertainty and making the training of the model more robust to the availability of few training observations. We showed that it may lead to higher predictive accuracy than than the single-fidelity approach introduced in chapter 1 for the same analysis budget. The second contribution of this chapter is the study of the different approaches to model the LF and HF FS. The few multi-fidelity approaches to approximation by ridge functions found in the literature do not address this question. Finally, as in the previous chapter, all codes used to run experiments 2.1 and 2.2 have been made openly available online to enable the reproduction of the results presented in this thesis, as well as allow future use of the proposed methods.

3.3.3 Next Steps

This chapter focused on enabling multi-fidelity surrogate modeling of analyses with high-dimensional inputs. We have shown that by extending the proposed approach to the multi-fidelity context, we are able to reach higher predictive accuracies for the same analysis cost when both low- and high-fidelity observations are available. In the next chapter, we will be focusing on adaptive sampling, another strategy for assisting the creation of surrogate models of expensive analyses. Its application is also made harder by high-dimensional inputs, and we will investigate ways to adapt it to that context.

CHAPTER 4

SAMPLING STRATEGIES LEVERAGING THE FEATURE SPACE

In chapter 1, we identified sampling strategies for high-dimensional inputs as the third research area and introduced the third research question for this thesis:

Research Question 3

How can we alleviate the impact of the curse of dimensionality on DOEs and adaptive sampling for low analysis budgets?

This chapter addresses this research question and therefore focuses on the third surrogate modeling scenario presented in chapter 1, adaptive sampling in the multi-fidelity context, for which a generic process is recalled in fig. 4.1.

The curse of dimensionality impacts this surrogate modeling scenario in two main ways: 1) traditional surrogate modeling techniques are inadequate for high-dimensional input spaces, therefore the creation of the surrogate model itself is challenging, and 2) prior to the creation of the surrogate model, training observations must be selected, and traditional strategies for selecting these training observations are also ill-suited to high-dimensional input spaces. Chapters chapter 2 and chapter 3 already focused on the first impact. We have proposed different predictive models based on approximation by ridge functions and trained in a fully Bayesian manner, first in the single fidelity context and then in the multi-fidelity context, that have been shown to help alleviate the curse of dimensionality. Even though we assume the availability of analyses of multiple fidelities, the single fidelity approach will be of interest as we will adopt a strategy in which the LF analysis will be exclusively evaluated at first, resulting in an initially single fidelity surrogate model.

In this chapter, we will focus on addressing the second impact of the curse of dimensionality in this scenario. There are two stages of the process in which the selection of

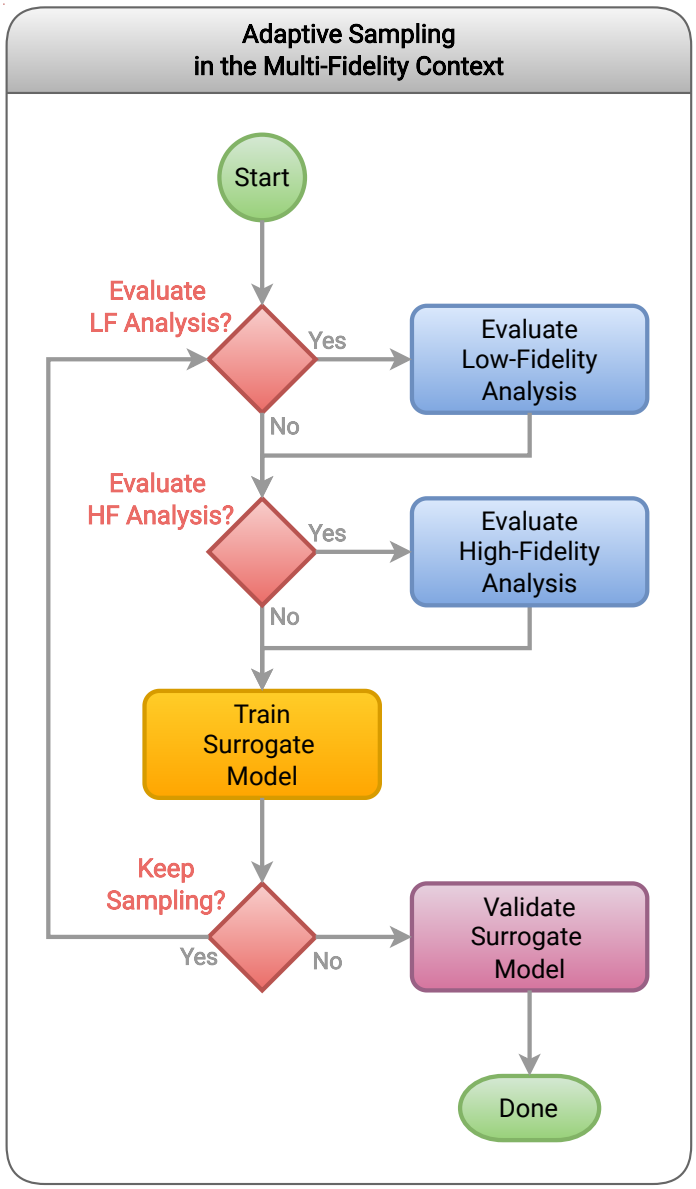


Figure 4.1: Generic process for the third considered surrogate modeling scenario, adaptive sampling in the multi-fidelity context.

new observations occurs and that are impacted by the curse of dimensionality: 1) *a priori* sampling using DOEs and 2) adaptive sampling. DOEs are made challenging by the sheer size of a high-dimensional space that unavoidably renders space-filling DOEs sparse for reasonably attainable amounts of training data. Alleviating the impact of the curse of dimensionality on high-dimensional DOEs will be the focus of the first section. In addition to the inherent sparsity of observations, adaptive sampling is made more complex by the high-dimensional input space in which the optimization problem solved for selecting new samples must be solved. This will be the focus of the second section, in which we propose and assess a strategy seeking to improve adaptive sampling for high-dimensional inputs.

4.1 Design of Experiment Leveraging the Feature Space

4.1.1 Background and Research Objective

Even when active sampling is used, an initial set of analysis observations must be obtained in order to create the surrogate model then used to evaluate the selection criterion. In this section, we focus on the creation of this initial set of observations. As we will see in the next section, many approaches have been developed to improve active sampling when the input space is high-dimensional by leveraging a low-dimensional subspace, mostly in the context of Bayesian optimization (BO). However, we have not found any method focusing on the initial selection of training points in the MF context. Among the multi-fidelity methods discussed in the previous chapters, some were gradient-based and others gradient-free. We do not expect the particular selection of training observations to have a great impact on the gradient-based methods [151, 109, 155], as gradients already contain much information about the directions of greater variations of the function of interest. Among the gradient-free approaches [157, 153, 131], the initial selection of training points is either based on random sampling or unspecified in the method formulation. This is problematic as DOEs are plagued by the curse of dimensionality, as discussed in chapter 1. This leads to the following gap:

Gap 3.1

Existing approaches to multi-fidelity AS either 1) are less affected by the curse of dimensionality thanks to gradient information, or 2) do not address its impact on the selection of initial training observations.

Existing multi-fidelity approaches to AS rely on an initial sample of both LF and HF observations. However, as depicted in fig. 4.1, in an active sampling method in the multi-fidelity context, we are left with the choice of evaluating only the LF analysis, only the HF one, or both at every step of the process, including initially. A motivation for delaying the evaluation of the HF analysis would be the availability of additional information about the structure of the underlying analysis that could be used when selecting the HF samples. This information could be acquired using LF observations only. In chapter 3, we have clearly observed the advantage of sharing the same FS across the LF and HF layers of the proposed MF model. When we experimented with different FS for every level of fidelity, the knowledge about the low-dimensional structure of the analysis of interest was not transferred from the relatively cheap and abundant LF training observations to the HF layer of the model, leading to poor predictive accuracy of the resulting MF model. The LF FS is therefore of value for modeling the HF analysis and it can be obtained using LF observations only.

This motivates an approach in which the LF analysis is first exclusively evaluated and the observations are used to create an initial single-fidelity LF surrogate model using the proposed fully Bayesian approach to approximation by ridge functions. The LF FS being a byproduct of the surrogate model, it can then be leveraged to inform the selection of the initial HF training samples. Although the AS has been used on multiple occasions to assist BO, we did not identify an existing approach to leveraging it to perform a DOE. This discussion motivates the following research question:

Research Question 3.1

In a multi-fidelity context, how can we leverage the feature space obtained as a byproduct a single-fidelity surrogate model exclusively trained using observations of the low-fidelity analysis to inform the selection of the initial high-fidelity training observations?

4.1.2 Proposed Method

Many methods exist for performing traditional DOEs an the interested may refer to [17] for a thorough exposition to these techniques. We are particularly interested in space-filling designs since they are the most suited to training global surrogate models.

Three main challenges arise when trying to use the FS to perform a DOE. First, in the proposed approach, we obtain a distribution for the FS, not just a single FS. Second, even though the original input space is an hypercube, the FS has more complex bounds that need to be defined and enforced in the DOE. Third, the mapping from the FS back to the original input space, which is required to evaluate the analysis, is ill-defined.

Multiple options are available to address the first challenge. The uncertainty could be dropped by either selecting a single FS among the sampled FS, or by taking an average over the posterior draws of the projection parameters. However, it is unclear 1) how the single FS draw should be selected in the first case, and 2) what the relevance of a "mean FS" would be. Instead, DOEs based on an experimental design criterion such as *maximin* offer a way to account for the uncertainty in the FS: the criterion can be computed for every FS and then averaged over them. In this case, the significance of the resulting quantity is clearer than if we had taken the mean of the FS, as it may be interpreted as an estimate of the experimental design criterion that is robust against the epistemic uncertainty in the FS.

The second challenge was highlighted in [110] in which the bounds of a 2D AS of a 3D input space are represented in figure 4.1. More generally, the bounds of an AS form a polytope [163, 164, 165] whose shape can become quite complex for high-dimensional

input spaces. In the context of BO, [163] proposes to ensure that points selected in the plane of the FS belong to the initial bounded input space by adding a set of linear constraints to the optimization of the selection criterion. However, as noted in [164], the enforcement of these constraints complicates sampling approaches. In order to avoid having to incorporate a large number of constraints in the optimization problem, we propose to work in the original input space, and to alleviate the impact of the curse of dimensionality by designing an experimental design criterion that effectively mostly varies when input locations are varied within the input space. In this manner, by using a gradient-based optimization algorithm for optimizing the experimental design criterion, the search directions and consequently the regions of the input space searched while carrying out optimization, will effectively be low-dimensional.

The third challenge comes from the fact that the projection onto the FS is not injective (i.e., multiple points in the input space may be mapped to an identical point in the FS). As a consequence, the inverse transformation is ill-posed: the coordinates of the point in the so-called *inactive* subspace remain to be specified. One option would be to set these coordinates to zero, but this would highly constrain the explored regions of the input space. Moreover, if we constrain new observations to the span of the current FS, we will hinder the refinement of the FS determination that we hope to gain from new observations. Another possibility is to randomly choose the coordinates of new sampling locations in the inactive subspace. In order to achieve this and not face the same issues than those for sampling the feature space, we propose to add a second component in the proposed experimental design criterion that favors exploration of the whole input space.

Based on this discussion, we observe that using fixed DOEs would be challenging due to the particular constraints originating from working in a bounded subspace of the original input space. Instead, building a criterion resulting from the optimization of an experimental design criterion would allow us to address these constraints. We propose to base our criterion on a widespread design, the *maximin distance design*, and adapt it based on the

discussion above to take advantage of our knowledge of the FS, resulting in th criterion c given in eq. (4.1).

$$c(\mathcal{D}) = \frac{1}{N_{post}} \sum_{i=1}^{N_{post}} \left(K \min_{\mathbf{x_1}, \mathbf{x_2} \in \mathcal{D}} \frac{\|\mathbf{W_i}^T \mathbf{x_1} - \mathbf{W_i}^T \mathbf{x_2}\|}{\sqrt{d_{FS}}} + \min_{\mathbf{x_1}, \mathbf{x_2} \in \mathcal{D}} \frac{\|\mathbf{x_1} - \mathbf{x_2}\|}{\sqrt{d_{\mathcal{X}}}} \right) \tag{4.1}$$

Where $\mathcal{D}$ is the experimental design, N_{post} is the length of the chain of posterior draws obtained through MCMC, $\mathbf{W_i}$ is the projection matrix using the i^{th} draw from from the posterior chain of projection parameters, d_{FS} is the FS dimension, $d_{\mathcal{X}}$ is the input space dimension, and K is a weighting factor.

The first part of the expression in eq. (4.1) corresponds to a space-filling criterion in the FS while the second corresponds to a space-filling criterion over the original, high-dimensional input space. Each component was normalized as the square root of the dimension since this corresponds to the greatest Euclidean distance between two points inside a unit hypercube of the corresponding dimension (length of the longest diagonal). The weighting factor K allows to adjust the relative importance of the two components in the proposed criterion. In our case, we mostly care about obtaining a good design in the FS, and the exploration of the high-dimensional input space is secondary, therefore we select $K = 100$.

The experimental design is obtained by maximizing the criterion c over experimental designs $\mathcal{D}$ of the desired size: $\mathcal{D}^\star = \max_{\mathcal{D} \subset \mathcal{X}} c(\mathcal{D})$. In practice, this is achieved using the scipy's implementation of the L-BFGS-B algorithm with 100 restarts to account for local optima.

This leads to the following hypothesis:

Hypothesis 3.1

A multi-fidelity surrogate model with better predictive accuracy is obtained when selecting high-fidelity training observations using the proposed experimental design criterion rather than a space-filling DOE in the original high-dimensional input space because the low-fidelity feature space can be used as an approximation of the high-fidelity feature space.

4.1.3 Setup of Experiment 3.1

Need for Experiment 3.1

Although we expect the proposed DOE criterion to perform better than a DOE in the original high-dimensional input space as it focuses on a subspace in which the variations of the function-of-interest are greater in average, and whose low dimension is less affected by the curse of dimensionality, it has not yet been put to the test. In order to test hypothesis 3.1, the proposed approach should be compared to a DOE method corresponding to current practice. In the context of surrogate modeling, space-filling designs are used, in which points are chosen to cover the original input space of the analysis, since current methods would not leverage the knowledge of a feature space. Among available space-filling methods, Latin-hypercube sampling (LHS) DOEs are commonly used in standard practice, and they will therefore be used as a basis for comparison.

Experiment Design

The options considered in this experiment only differ by the manner in which their training observations are selected: on the one hand, they are selected using a traditional space-filling LHS DOE, and on the other hand they are selected using the proposed experimental design criterion. Otherwise, the same approach to MF surrogate modeling is used to train the models.

Because this experiment and the next deal with the selection of training samples, we need the ability to query the test analyses online while running the experiment cases. This is not possible with most of the datasets that we worked with in previous experiments, for either one of the following reasons. First, some of the datasets used in previous experiments were obtained from openly available sources published online without the analysis used to generate the samples. Second, other datasets were generated using complex physics-based analysis requiring large amounts of computational power, making them challenging to integrate in the parametric study carried out here: the experiment already includes multiple thousands cases, and each of these cases would possibly entail tens of evaluations of the underlying analysis. This leaves us with the elliptic PDE analysis, already extensively used in the context of the multi-fidelity approach, as we were able to modify its input dimension, complexity, and level of fidelity. Although it is well representative of expensive analyses used in an engineering context, its evaluation is simple and cheap enough to be integrated as part of running experiment cases.

Since we are working with the probabilistic models tested in previous experiments, their implementations were simply reused. New developments specific to this experiment include 1) the computation of the criterion for the proposed FS-based space-filling DOE and 2) the optimization of the criterion. The expressions for computing the proposed criterion were presented in the previous section. In order to optimize it, we used the L-BFGS-B [166] gradient-based method, as implemented by the *scipy* Python library and we perform 100 restarts to account for local optima. The gradient evaluations are automatically computed via automatic differentiation by the *JAX* computational backend. First, the number of new HF observations is computed based on the allocated budget. Then, all their locations are optimized at once.

The source code used to generate the results presented thereafter has been made publicly available[1].

[1] https://gitlab.com/raphaelgautier/thesis_experiments_part4

Sensitivity Study

As in previous experiments, the proposed approach will be tested on representative analyses and a parametric study will be performed on multiple parameters, either to understand their impact on the performance of the proposed approach, or because they are a free parameter whose determination is out-of-scope of this thesis. As before, we expect the FS and input dimensions, as well as the total analysis budget, to have an impact on performance, and these parameters will be varied. The training is also repeated multiple times using different random seeds, for both the input space and FS DOE.

Although the LF allocation ratio is expected to have an impact on predictive accuracy, as seen in experiment 2.1, it is kept fixed in this experiment and the next in order to limit the number of experimental cases needed to be run. In experiment 2.1, we observed that the allocation of the analysis budget to LF and HF observations should not be too "unbalanced", otherwise leading to poor predictive models. This corresponds to models trained with either very few LF or HF observations, leading to either the LF or HF layer of the DMF-GP to lack sufficient training data. The need for a balanced MF training set is expected to be independent of the method used to select observations, i.e., whether they are selected using a DOE in the original input space or using the FS-based DOE. Based on this, the LF allocation ratio is fixed at 0.5, corresponding to an equal share of the budget being allocated to LF and HF observations.

The other half of the budget is then used to select HF observations, using either 1) an LHS DOE in the original input space (benchmark method) or 2) the proposed approach to FS-based DOE. Eventually, in experiment 3.2, the budget will be split three-way instead of two-way as here, because a fraction of the budget will be allocated to the adaptive selection of HF observations. Training is repeated five times to account for the particular choice of training observations, using a random seed to ensure reproducibility.

Ranges are defined for numerical parameters and a full-factorial DOE is generated to study the variations of all parameters (see table 3.5).

Table 4.1: Experiment 3.1 – Summary of the parameters varied in the parametric study

Group	**Parameter**	**Parameter Values**
Analysis	Input Space Dimension	[10, 25, 50, 100]
	Complexity (parameter]β)	[0.01, 1.0]
Training Set	Analysis Budget	see table 4.2
	LF Allocation Ratio	fixed at 0.5
	HF DOE Allocation Ratio	fixed at 0.5
	HF DOE Space	[original input space, feature space]
	Number of Training Repetitions	5 repetitions with fixed random seeds
Method	FS dimension	[3, 5]
	LF/HF FS relationship	fixed (shared FS)

Presentation and Interpretation of the Results

In this experiment, we are comparing the predictive accuracy of the surrogate model obtained when training observations are selected using 1) a traditional space-filling DOE in the original high-dimensional input space, and 2) the proposed approach to DOE leveraging the feature space. Based on the experimental setup discussed previously, this corresponds to half the analysis budget being spent on the initial selection of LF observations, and the other half on the HF observations. To be meaningful, the comparison should be made given a dataset, a total budget, and an FS dimension. This only remaining source of variation stemming from the training repetitions, a simple box plot is then sufficient to compare the performance of each method. Each plot will show the predictive accuracy distributions for 1) the DOE in the input space on the left and 2) the FS-based DOE on the right. For compactness, results will be presented as a grid of plots for each complexity β and FS dimension: input dimension will increase from top to bottom, and analysis budget will increase from left to right.

Table 4.2: Experiment 3.1 – Analysis budget (in HF samples) as a function of the analysis input dimension.

Input Dimension	**Analysis Budget (in HF samples)**
10	[5, 10, 15, 20, 25]
25	[10, 20, 30, 40, 50]
50	[20, 40, 60, 80, 100]
100	[50, 90, 130, 170, 210]

For the results to support hypothesis 3.1, the distribution of predictive accuracy obtained using the DOE leveraging the FS should be higher than the one corresponding to the input space DOE, indicating an average increase in predictive accuracy. The FS selection method proposed in section 2.3 is not used because it is not suited to being automated. Instead, it is assumed that the FS dimension leading to the highest predictive accuracy would be selected. Therefore, the higher predictive accuracy should be observed for at least one of the two FS dimensions considered for each dataset. We expect the predictive accuracies to generally increase from left to right of the grid plots, i.e., as the total analysis budget increases.

4.1.4 Results of Experiment 3.1

In this section, we will be looking at the impact of leveraging the FS for gathering HF observations used to build the multi-fidelity model, instead of simply performing a DOE in the original input space. We will focus on the case where the full HF analysis budget was spent on the DOE, as this allows for a more synthetic look at the results. A global viewpoint of the problem is adopted, in which summary statistics are used to assess the accuracy of the predictive model, and we will study trends as the input dimension and total analysis budget vary.

In this section, we will use figs. 4.2 to 4.5 to discuss the impact of performing the HF DOE in the FS. They depict a comparison the coeffcient of determination R^2 for different

values of the input space (plots organized from top to bottom) and for different values of the total analysis budget (from left to right). Individual plots show the R^2 value for 1) the DOE in the input space on the left-hand side, and 2) the FS DOE on the right-hand side. Colors correspond to training repetitions. An upward progression shows improvement of the predictive accuracy when using the proposed approach, while a downward progression shows deterioration.

Figure 4.2 shows the results obtained with the elliptic PDE data with $\beta = 0.01$ when the feature space dimension is set to 3. In the 10-dimensional input case, the proposed approach has a positive impact in all but one training repetition. We observe that the impact tends to decrease as the analysis budget increases. This makes sense, as more training samples make it easier to obtain a good coverage of the original input space. The most dramatic improvement happen for intermediate analysis budget values, such as for a budget of 11 HF samples. Performing the DOE in the original input space leads to models with R^2 randing from as low as 0.2 up to 0.75, while leveraging the FS to carry out the DOE leads to R^2 from 0.7 to 0.9. When the analysis budget is too low, the proposed approach does not bring much advantage.

In the case of the 25- and 50-dimensional input spaces, similar results are obtained. Poor models are obtained for the lowest analysis budget irrespective of the method. As the analysis budget increases, the approach relying on the FS allows to reach higher predictive accuracies. After that, for larger analysis budgets, the difference between the two approaches is not significative.

Results look different in the 100-dimensional case. For the lowest analysis budget, the proposed approach significantly and consistently increases the predictive accuracy of the model. However, as soon as the analysis budget increases, the impact first decreases (for a budget of 67 HF samples), and the proposed approach tends to have a detrimental effect for the highest three analysis budgets. The cause for this is unclear, and increasing the FS dimension makes this problem disappear.

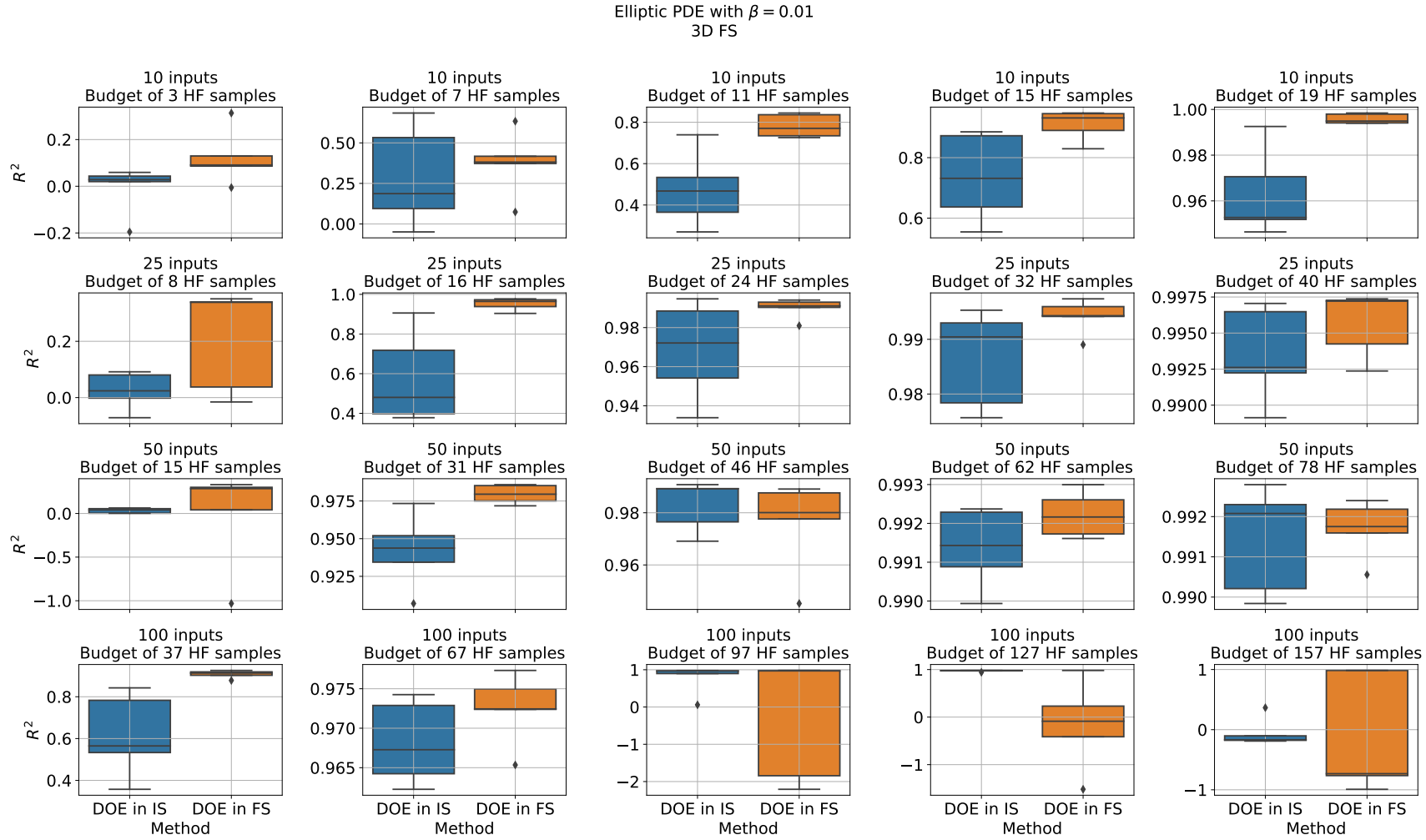


Figure 4.2: Comparison of the coefficient of determination R^2 of predictive models trained using points sampled using 1) a DOE in the original input space (left) or 2) a DOE leveraging the feature space (right), where the full HF analysis budget was spent on the DOE, for the elliptic PDE dataset with $\beta = 0.01$ and 3D FS. Colors map to training repetition. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

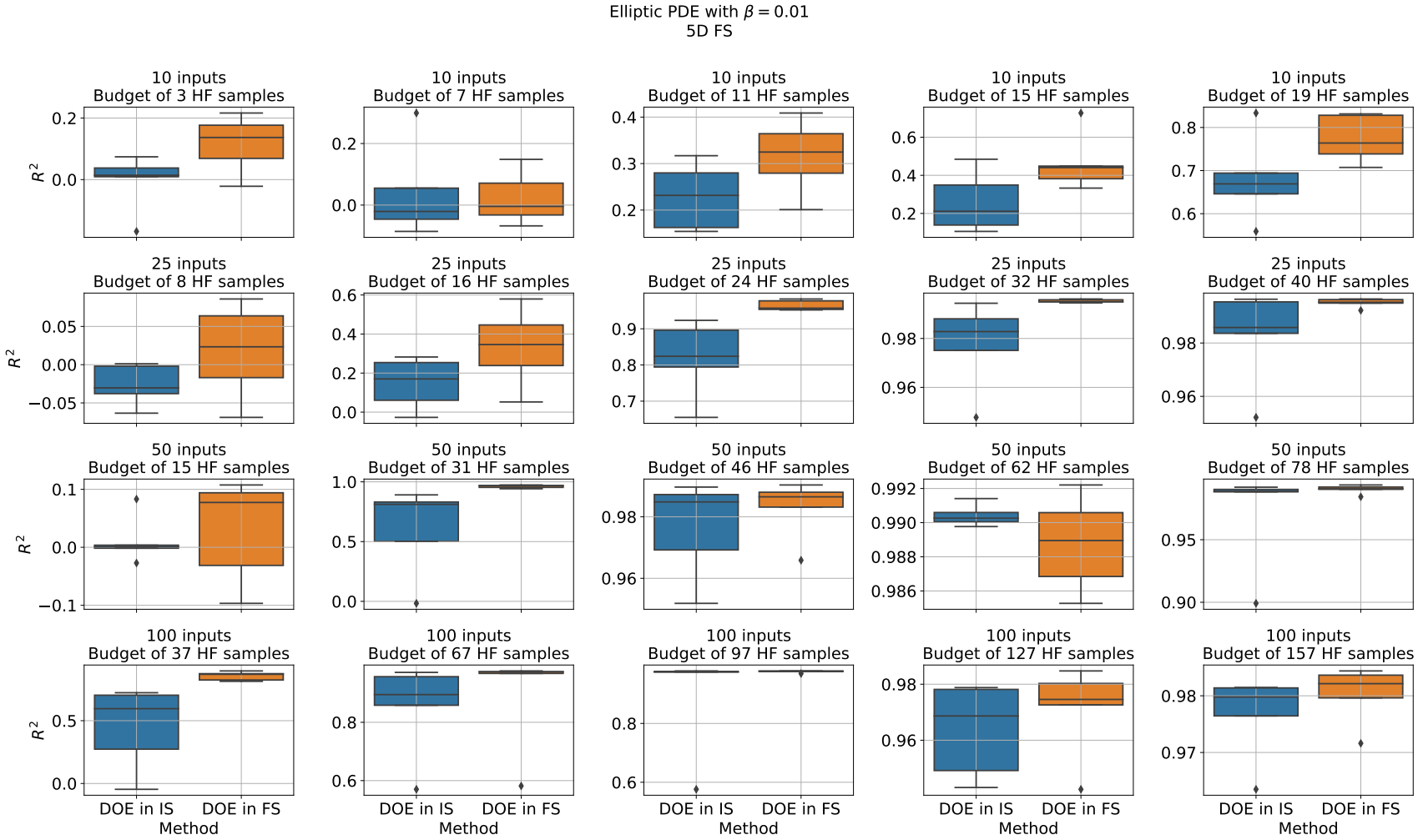


Figure 4.3: Comparison of the coefficient of determination R^2 of predictive models trained using points sampled using 1) a DOE in the original input space (left) or 2) a DOE leveraging the feature space (right), where the full HF analysis budget was spent on the DOE, for the elliptic PDE dataset with $\beta = 0.01$ and 5D FS. Colors map to training repetition. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

We now focus on Figure 4.3, in which a 5-dimensional feature space was used instead of the previously used 3D FS. The first observation we can make is that things remain very similar for the cases with 25D and 50D inputs: the proposed approach allows to significantly increase the predictive uncertainty for intermediate values of the analysis budget while having less impact when the budget is either too small, or very large. In the case the 10D inputs, while the proposed approach still has a positive impact, it does not reach R^2 values as high as when a 3D FS was used. This suggests that a 3D FS was more appropriate than a 5D one in this case. The other difference with the results discussed previously is visible in the 100D inputs case. While the FS-based sampling approach had a negative impact in certain cases, in this case it has at worse no significant impact. This suggests that a 5-dimensional FS is more appropriate in this case. The increase in predictive accuracy displayed in the case of the lowest analysis budget (37 HF samples) remains significant and consistent across training repetitions.

Let us now focus on the results obtained for the elliptic PDE with $\beta = 1.0$ shown in figs. 4.4 and 4.5. They are similar to the results obtained with the $\beta = 0.01$ dataset. In the case of the 10D inputs, the proposed approach either has insignificant impact when the analysis budget is low, or a positive impact for larger analysis budgets. Once again, higher predictive accuracies are obtained with the lower-dimensional FS. This is proably due to the fact that, in this case, the 3D FS is sufficient, and the extra parameters brought by the 5D FS only tend to make the predictive model worse. In the case of the 25D inputs, 25- and 50-dimensional inputs cases, the same observations can be made, except for a 3D FS for the highest considered analysis budget, where the proposed approach tends to decrease predictive accuracy. This is however improved in the 5D FS case, where the proposed approach tends to have a positive impact. In the case of the 100-dimensional inputs, the FS-based approach significantly increases predictive accuracy for the lowest considered analysis budget with both 3D and 5D FS. Results are more mixed for alrger analysis budgets. In the 3D FS case, results are inconsistent across training samples, and the

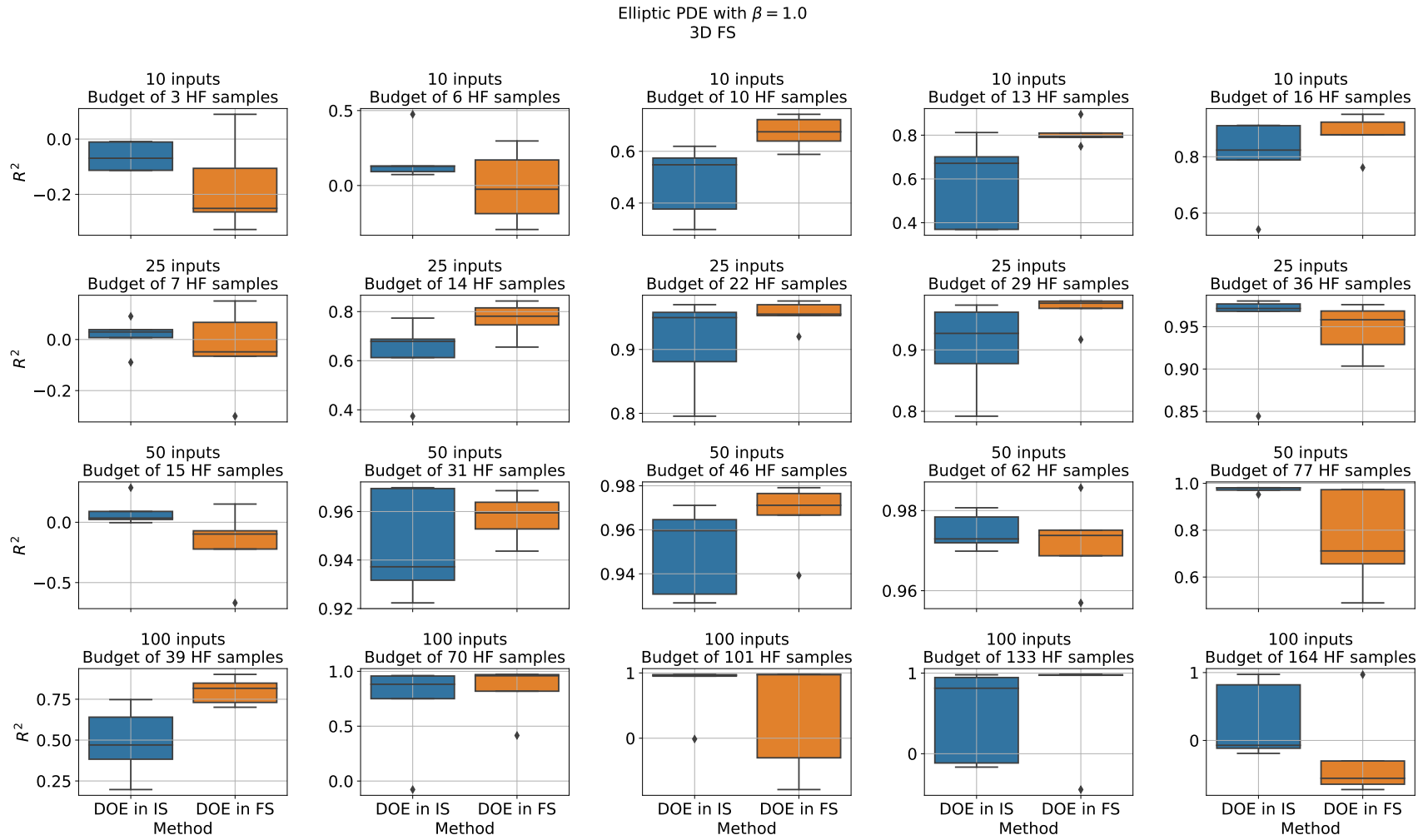


Figure 4.4: Comparison of the coefficient of determination R^2 of predictive models trained using points sampled using 1) a DOE in the original input space (left) or 2) a DOE leveraging the feature space (right), where the full HF analysis budget was spent on the DOE, for the elliptic PDE dataset with $\beta = 1.0$ and 3D FS. Colors map to training repetition. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

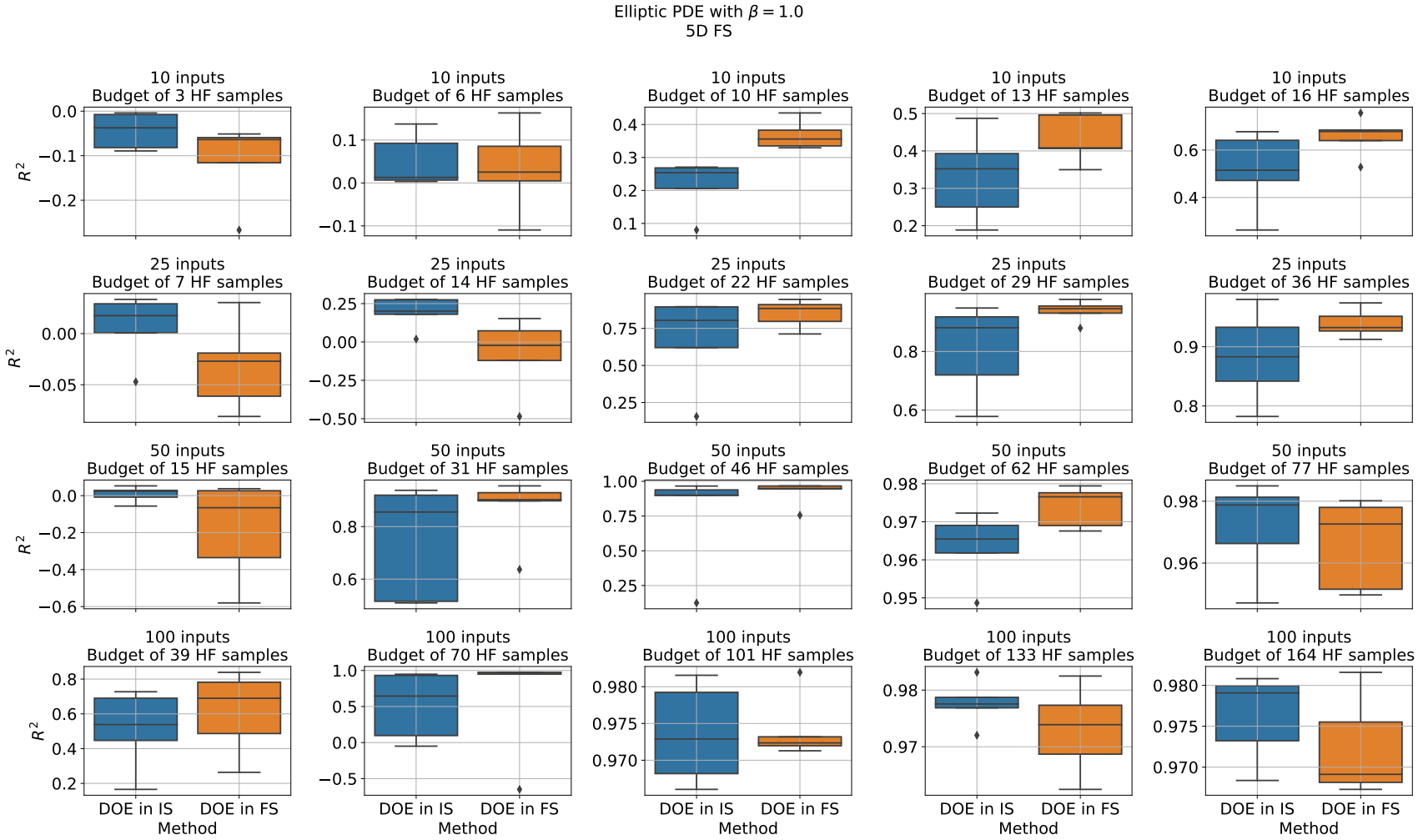


Figure 4.5: Comparison of the coefficient of determination R^2 of predictive models trained using points sampled using 1) a DOE in the original input space (left) or 2) a DOE leveraging the feature space (right), where the full HF analysis budget was spent on the DOE, for the elliptic PDE dataset with $\beta = 1.0$ and 5D FS. Colors map to training repetition. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

proposed approach may in some instances lead to drastic reductions in predictive accuracy. In the 5D FS case however, although the FS-based approach sometimes leads to reductions in predictive accuracies, these are not significant: this can be seen by observing the scales of the y-axis in the plots. In that case, the FS-based approach tends to have a positive impact for the smallest considered analysis budgets and insignificant impacts for larger analysis budgets.

4.1.5 Conclusion

Overall, the results obtained here are in line with expectations. When the analysis budget is too small compared to the complexity of the problem under consideration, the proposed method is not sufficient to obtain an adequate surrogate model. In cases where the analysis budget is high enough such that a traditional DOE is sufficient to obtain good predictive accuracy, the proposed approach does not bring significant advantage. Some cases are problematic, in which the proposed approach actually leads to a decreased predictive accuracy for relatively high analysis budgets. As we have seen, this is improved by selecting a different FS dimension. Although this was not carried out due to the necessity of automating the parametric study carried out here, the dimension selection developed in section 2.3 may be used to select the most appropriate dimension. The setting in which the proposed DOE method leveraging the FS brings the most advantage is for those intermediate analysis budgets. In these conditions, we have seen on all considered examples that carrying out a DOE in the input space directly leads to poor models, whereas leveraging the FS leads to predictive models that, although not perfect, may do a much better job at identifying meaningful trends in the underlying analysis.

Conclusion on Hypothesis 3.1

Selecting the HF observations using a DOE obtained by optimizing the proposed experimental design criterion that leverages the feature space obtained as a byproduct of the initial LF surrogate model leads to a MF surrogate model with higher predictive accuracy than if the DOE had been performed in the original input space.

4.2 Adaptive Sampling Leveraging the Feature Space

4.2.1 Background and Research Objective

In the previous section, we proposed a method to leverage information about the FS obtained by using LF observations only, thus delaying the evaluation of the expensive HF model until knowledge about the mathematical structure of the underlying analysis has been acquired. In this section, we focus on the next step in the adaptive selection surrogate modeling scenario in the multi-fidelity context: use the MF surrogate model to inform the acquisition of new analysis observations. Since we made the choice to initially spend all budget allocated to LF observations in the initial set of observations, active sampling only concerns the selection of *HF* observations. In the following paragraphs, we are exploring the literature for existing approaches that may fulfill our specific requirements: our objective is global metamodeling and, as for the initial DOE, we wish to leverage the FS obtained as a byproduct of our surrogate modeling approach.

Existing approaches to active sampling may have multiple objectives, the two most common being 1) global metamodeling or 2) global optimization. Recently, many methods have been developed to enable the use of BO, the prevailing approach to active sampling for global optimization, in the context of high-dimensional input spaces. Like the procedures proposed in this thesis, it relies on projections on low-dimensional subspaces of the high-dimensional input space, also referred to as *linear embeddings* [167, 168, 169, 163, 170, 171, 172, 164, 173, 165, 174, 175, 176, 153]. However, in the context of this thesis,

our end goal is to create global surrogate models that may be used for a variety of many-query applications. Our objective is therefore to maximize the predictive accuracy of the surrogate model, as measured by its R^2 or MLPPD, and we will focus therefore focus on the first category of active sampling approaches, i.e., those targeting global metamodeling.

Gap 3.2

Existing approaches to active sampling for high-dimensional inputs are aimed at Bayesian optimization and do not leverage the knowledge of a low-dimensional subspace when already available.

Nonetheless, the methods targeting BO address some challenges that are also relevant to active sampling for global metamodeling when a low-dimensional FS is known. In the following, we will be taking a closer look at approaches to active sampling specifically designed for metamodeling to see how they may be leveraged in the present context.

[38] classifies adaptive sampling strategies for global metamodeling as either based on 1) variance, 2) query-by-committee, 3) cross-validation, or 4) gradient. Approaches based on variance leverage the predictive uncertainty provided by a probabilistic predictive model to select new observations. Approaches based on query-by-committee also leverage a measure of variance that is computed from the discrepancy between multiple predictive models of the same underlying analysis instead of being directly made available by a probabilistic model. In approaches based on cross-validation, an estimate of the prediction error is used using cross-validation, i.e., repeating training with different subsets of the complete training set, and this estimate is used to drive the selection of new observations. Approaches based on gradients leverage the availability of gradients to focus sampling in regions of greater function's variations. Among these options, variance-based approaches appear as the natural choice given that the proposed approach to surrogate modeling is in essence probabilistic. In fact, one of the main contributions of the single-fidelity fully Bayesian approach to approximation by ridge functions is the complete quantification of epistemic

uncertainty due to limited training data.

The next step is to select a particular selection criterion, i.e. the quantity that leverages the variance quantification and that is optimized to select the location of the next observations. We find ourselves in a similar situation as in section 4.1 where methods exist to address the objective-at-hand in a low-dimensional subspace such as the FS but they cannot be naively transposed to the FS because of the particular constraints stemming from 1) the probabilistic nature of our determination of the FS and 2) the geometrical nature of the FS. This motivates the following research question:

Research Question 3.2

How can we adapt existing approaches to active sampling for global metamodeling to the context of high-dimensional input spaces when the knowledge of a low-dimensional FS is available?

4.2.2 Proposed Method

We now aim at designing an active sampling selection criterion leveraging the FS. Multiple variance-based selection criteria for global metamodeling have been proposed in the literature, based on mean square error [177], integrated mean square error [178], maximum entropy [179, 180], or mutual information [181]. Beyond these, other approaches have been devised to account for the fact that usual GP covariance kernels are stationary, leading to new selected observations to be space-filling (i.e., fill the empty spaces in input space between previous observations) [38].

As in the previous section, we cannot simply transpose one of these active sampling criteria to the FS. The constraints are similar to those previously encountered when proposing an alternate experimental design criterion in the previous section: it needs to 1) account for the uncertainty in the FS, 2) allow for the complete specification of the coordinates of the next observation, and 3) account for the particular shape of the FS. As we expect benefits to

come from working with the FS rather than the specifics of the underlying active sampling criterion, we aim for simplicity and do not particularly seek a complex active sampling method, and we will therefore simply use the mean square error criterion as a basis for the proposed criterion. Using the mean square error criterion, we simply select the point in input space with highest predictive variance.

In order to account for both the intrinsic GP uncertainty and the uncertainty in the model parameters θ, we use the law of total variance:

$$\mathrm{Var}(y^\star \mid \mathbf{x}^\star) = \mathbb{E}_\theta \left[\mathrm{Var}\left(y^\star \mid \mathbf{x}^\star, \theta\right)\right] + \mathrm{Var}_\theta \left(\mathbb{E}\left[y^\star \mid \mathbf{x}^\star, \theta\right]\right) \tag{4.2}$$

Where:

- $\mathrm{Var}(y^\star \mid \mathbf{x}^\star)$ is the variance of the prediction at input site $\mathbf{x}^\star$ due to both the intrinsic GP uncertainty and uncertainty in the model parameters;
- θ is a vector of all model parameters (length scales, signal and noise variance for both LF and HF layers, as well as projection parameters);
- $\mathrm{Var}\left(y^\star \mid \mathbf{x}^\star, \theta\right)$ is the predictive variance of the GP at input location $\mathbf{x}^\star$ for a given value of its hyperparameters;
- $\mathbb{E}\left[y^\star \mid \mathbf{x}^\star, \theta\right]$ is the expected value of the GP at input location $\mathbf{x}^\star$ for a given value of its hyperparameters.

The last two quantities are obtained by fixing the GP's parameters and running a Monte-Carlo to propagate the uncertainty in the output of the LF layer through the HF layer. This is repeated for all values of the model parameters in the MCMC chain to estimate the expectation $\mathbb{E}_\theta$ and variance Var_θ. The estimation of $\mathrm{Var}(y^\star \mid \mathbf{x}^\star)$ therefore consists in running two nested Monte-Carlo simulations. By using this measure of variance, we account for the uncertainty in the FS.

This leads to the following hypothesis:

Hypothesis 3.2

Selecting high-fidelity training observations using a maximum variance selection criterion in complement of the proposed DOE will lead to a surrogate model with higher predictive accuracy than if all training observations had been selected using the proposed DOE because it leverages the current knowledge of the response to focus the selection of new training observations in locations of the input space where predictive uncertainty is the highest.

4.2.3 Setup of Experiment 3.2

Need for Experiment 3.2

In different contexts, adaptively selecting new observations by choosing input locations in input space where predictive variance is maximized has been shown to lead to better improvements in predictive accuracy than using a pre-determined DOE that does not leverage current knowledge of the response. However, the efficiency of the approach is expected to depend on the form of the surrogate model, and it has not been attempted using the proposed surrogate modeling form. In order to test hypothesis 3.2, we need to check whether the maximum variance criterion computed using this particular model effectively leads to higher predictive accuracy than the previously proposed DOE.

Experiment Design

As the previous experiments, experiment 3.2 is designed to select the best option for carrying out a part of the process used to create a surrogate model with high-dimensional inputs when only few observations are available. In this case, we are focusing on the selection of the samples and we wish to determine whether it is better in terms of the predictive accuracy of the resulting surrogate model to select training observations adaptively by maximizing predictive variance or to use the FS-based DOE studied in experiment 3.1. Since we have

previously shown that the proposed approach to DOE leveraging the FS outperforms a traditional DOE in the input space, we will be using the proposed DOE approach as benchmark in this experiment. Namely, we expect a better predictive accuracy when adaptively sampling the training observations using the FS than when they are selected *a priori* using a DOE leveraging the FS.

In experiment 3.1, the analysis budget was split only two-way: 1) LF observations selected using a DOE and 2) HF observations selected using a DOE. In this experiment, the budget is split three-way: 1) LF observations selected using a DOE, 2) HF observations selected using the FS-based DOE, and 3) HF observations selected using the proposed adaptive sampling approach. We respectively refer to the corresponding budget fractions as 1) the "LF DOE budget fraction", 2) the "HF DOE budget fraction", and 3) the "HF adaptive budget fraction".

Similarly to experiment 3.1, the locations of the observations are determined online during the experiment, which means that access to the analysis is required. Pre-computed datasets used in experiments 1.1 to 2.2 are therefore not adapted. For this reason, the same analyses that were used in experiment 3.1 are used here. Additional implementation needed for this experiment included the computation and optimization of the proposed adaptive sampling selection criterion. The same optimization process that was used in experiment 3.1 for the custom DOE criterion is here used to optimize the proposed adaptive sampling criterion. Since this experiment only affects the selection of training observations, the implementations of the predictive models from previous experiments required no modification.

The source code used to generate the results presented thereafter has been made publicly available[2].

[2]https://gitlab.com/raphaelgautier/thesis_experiments_part4

Sensitivity Study

As in experiment 3.1, the parameters of the elliptic PDE analysis are varied to account for different input space dimensions and analysis complexity. Because the proposed approach to selecting the FS dimension is not suited to being automated, only two different values for the FS dimension are considered.

When it comes to the training set, we are still performing five training repetitions to account for the particular choice of training observations. Multiple values for the total budgets are also considered, as it has a significant impact on the predictive accuracy of the resulting surrogate models.

The partition of the analysis budget differs in this experiment. As mentioned previously, the total analysis budget is now split three-way as follows: 1) a fraction for LF DOE observations, 2) a fraction for HF DOE observations, and 3) a fraction for HF adaptive observations. The fraction of the analysis budget spent on adaptively selecting points is a new parameter, and it is expected to have an impact on predictive accuracy. As in experiment 3.1, we keep the LF DOE budget fraction constant to keep the number of experiment cases manageable. Unlike experiment 3.1 where the HF DOE budget was kept constant at half the total analysis budget, we will be varying it in this experiment. Once the LF DOE fraction and the HF DOE fraction are set, if any remains, the rest of the analysis budget can be allocated to HF adaptive observations.

We will work in 10% budget increments: at every stage, we select a number of new HF samples corresponding to 10% of the analysis budget. Therefore, in addition to the final results obtained when the complete analysis budget has been spent, we will also have access to predictive accuracies corresponding to situations in which only a fraction of the full budget has been used. In other words, we will be able to visualize the evolution of the models' predictive accuracy as the analysis budget is being progressively spent. Although this is not necessary to discuss the validity of hypothesis 3.2, this provides additional insights about the method.

Table 4.3 summarizes the parameters varied in the parametric study conducted as part of experiment 3.2.

Table 4.3: Experiment 3.2 – Summary of the parameters varied in the parametric study

Group	**Parameter**	**Parameter Values**
Analysis	Input Space Dimension	[10, 25, 50, 100]
	Complexity (parameter β)	[0.01, 1.0]
Training Set	Analysis Budget	see table 4.4
	LF Allocation Ratio	fixed at 0.5
	HF DOE Allocation Ratio	[0.1, 0.2, 0.3, 0.4, 0.5]
	HF Adaptive Allocation Ratio	see table 4.5
	Number of Training Repetitions	5 repetitions with fixed random seeds
Method	FS dimension	[3, 5]
	LF/HF FS relationship	fixed (shared FS)

Table 4.4: Experiment 3.2 – Analysis budget (in HF samples) as a function of the analysis input dimension.

Input Dimension	**Analysis Budget (in HF samples)**
10	[5, 10, 15, 20, 25]
25	[10, 20, 30, 40, 50]
50	[20, 40, 60, 80, 100]
100	[50, 90, 130, 170, 210]

Presentation and Interpretation of the Results

In order to support hypothesis 3.2, we seek to compare the predictive accuracy of the surrogate models respectively obtained when a fraction of the HF budget is spent on adaptively selected observations and when all the HF budget is allocated to the previously proposed

Table 4.5: Experiment 3.2 – HF adaptive budget fraction as a function of the HF DOE budget fraction.

HF DOE Allocation Ratio	HF Adaptive Allocation Ratios
0.1	[0, 0.1, 0.2, 0.3, 0.4]
0.2	[0, 0.1, 0.2, 0.3]
0.3	[0, 0.1, 0.2]
0.4	[0, 0.1]
0.5	0

FS DOE. To discuss the validity of the hypothesis, making the comparison once all the budget has been spent is sufficient, and this corresponds to the first set of plots that we will be showing in the next section. In order to better understand the limits of the proposed approach, we will also look at a second set of plots, in which we will look at the evolution of R^2 as the budget is being progressively spent.

For the first set of plots comparing the final predictive accuracy, we will use grid plots organized like in experiment 3.1: for a given dataset, the input dimension increases from top to bottom, and the total analysis budget increases from left to right. Within each plot, we will be comparing the distribution of predictive accuracy as a function of the fraction of the HF budget fraction spent on adaptive sampling using box-and-whisker plots. We recall that the first half of the budget is spent on LF observations. A fraction of 0.0 means that all observations were selected using the FS-based DOE, and it only goes up to 0.4 as some of the budget always has to be spent on the FS-based DOE to obtain an initial MF surrogate model before being able to carry out adaptive sampling. We are considering all these different HF adaptive budget fractions since we do not know a priori which HF adaptive budget fraction leads to the best results. As in other experiments, the fact that we are displaying distributions instead of point values corresponds to repetitions of the observations selection process, leading to different training sets. For the results to support hypothesis 3.1, in at least one of the cases where adaptive sampling is used, the distribution of the predictive

accuracy should be higher than when the FS-based DOE is used exclusively.

Although the plots discussed in the previous paragraph are sufficient to discuss the validity of hypothesis 3.2, we also wish to develop a better understanding of how predictive accuracy varies as the budget is being consumed. For every dataset and FS dimension, plots are organized as a grid in which the input dimension increases from top to bottom and the total analysis budget increases from left to right. Each plot displays the evolution of R^2 as a function of the consumed fraction of the HF budget. As opposed to the plots discussed previously, that corresponded to cases in which the *complete* analysis budget was consumed, these plots show the evolution of predictive accuracy *as the analysis budget is being consumed*. The x-axis corresponds to the consumed HF budget fraction: it starts at 0.1 since at least 10% of the budget is used to carry out the HF DOE. At that point, 40% of the analysis budget still remains to be spent. It increases up to 0.5, which corresponds to having consumed the whole analysis budget, since the LF budget fraction was fixed at 0.5.

The thick blue line corresponds to an HF DOE budget fraction of 0.5, meaning that adaptive sampling is not used. It shows the evolution of R^2 as more of the budget is spent selecting points using the proposed FS-based DOE. Then, each of the colored dotted lines corresponds to results obtained when performing adaptive sampling after having spent different budget fractions on the HF DOE. Accordingly, each of the colored line branches off the thick blue line at a different HF budget fraction. For example, the green line corresponds to an HF DOE budget fraction of 0.2. This means that half the budget is first spent on LF observations (not visible in the plots), then 20% of the budget is spent on the HF DOE (identical to the thick blue line), and only then are the last 30% of the budget spent adaptively selecting HF observations using the maximum-variance criterion (the dotted green line branches off the thick blue line for an HF budget fraction of 0.2).

4.2.4 Results of Experiment 3.2

In this section, we will be discussing the results using the experimental setup described previously. As discussed above, multiple parameters have been varied in order to get a better understanding of the performance of the proposed approach. In particular, adaptive sampling had been dome in a stepwise manner, where the analysis budget was progressively spent. At first, in section 4.2.4, we will be looking at the results obtained when the full analysis budget has been spent. Then, in section 4.2.4, we will be looking more closely at what happens as the analysis budget is being consumed.

Full Analysis Budget

In this section, we will be looking at bar plots for the coefficient of determination R^2 as a function of the budget fraction used for adaptively sampling the HF training samples, assuming that the whole analysis budget has been spent. For example, this means that an HF adaptive sampling budget of 0.1 corresponds to using 0.5 of the budget for LF samples (fixed in the context of this experiment), 0.4 of the budget for HF samples chosen using the LF-based DOE, and the remain 0.1 of the budget for adaptively sampled HF samples. Within the figures presented in this section, each individual plot corresponds to a different value for the input dimension and the total analysis budget.

Results for the different datasets and FS dimensions under consideration are shown in figs. 4.6 to 4.9. Since they are similar, we will discuss all results together. We observe that the proposed approach to adaptive sampling at best performs as well as using the FS-based DOE, but generally performs worse, leading to predictive models with lower accuracy. This observation is consistent across all considered datasets, analysis budgets, and FS dimensions. Looking at more detailed results in the next section will confirm this observation.

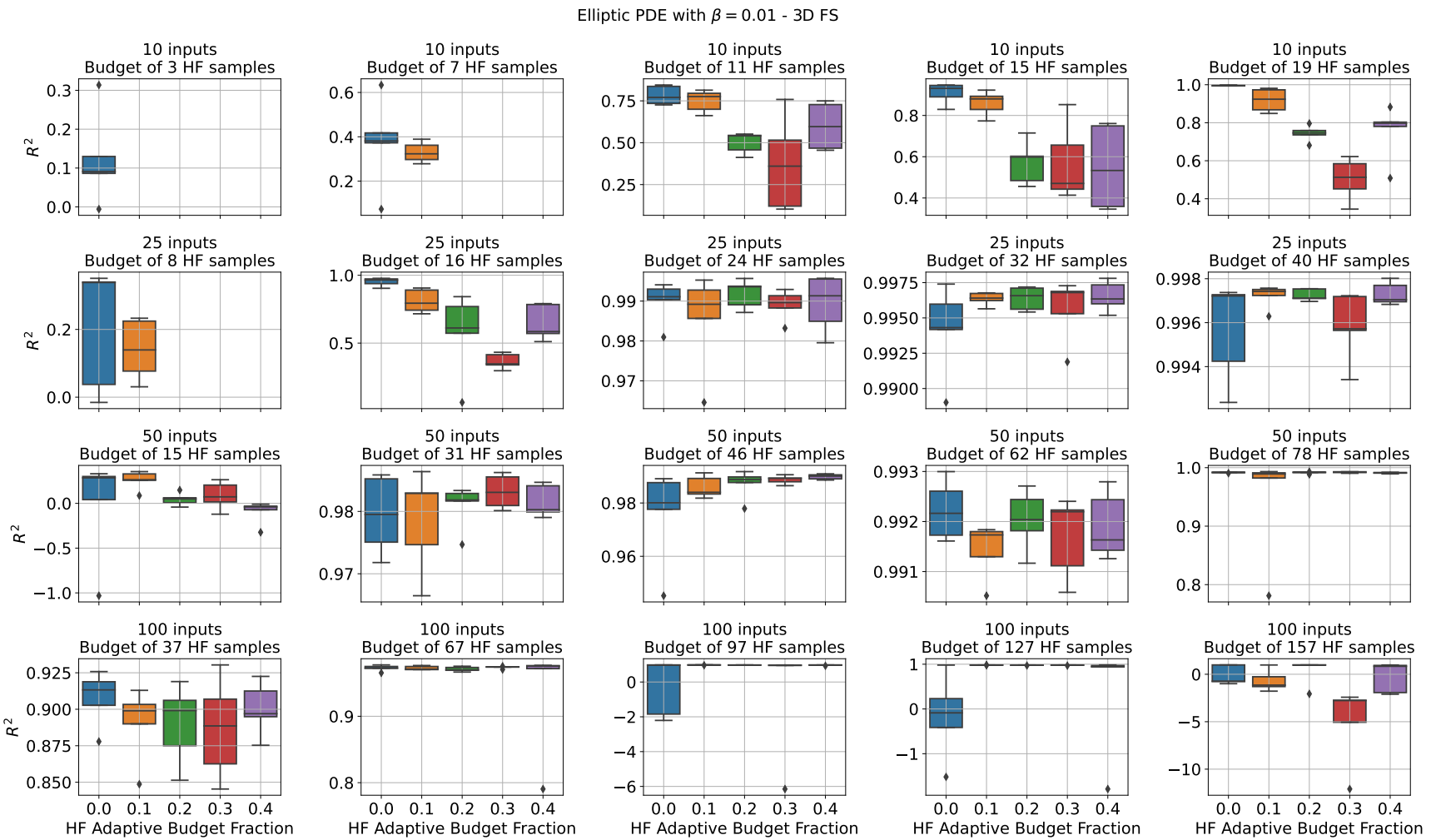


Figure 4.6: Evolution of the coefficient of determination R^2 as a function of the budget fraction used for adaptive sampling of HF observations using the proposed approach for the elliptic PDE dataset with $\beta = 0.01$ and 3D FS dataset. A budget fraction of 0 corresponds to selecting all points using the FS-based HF DOE. of predictive models trained using different points sampled using 1) a DOE in the original input space (left) or 2) a DOE leveraging the feature space (right), where the full HF analysis budget was spent on the DOE. Colors map to training repetition. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

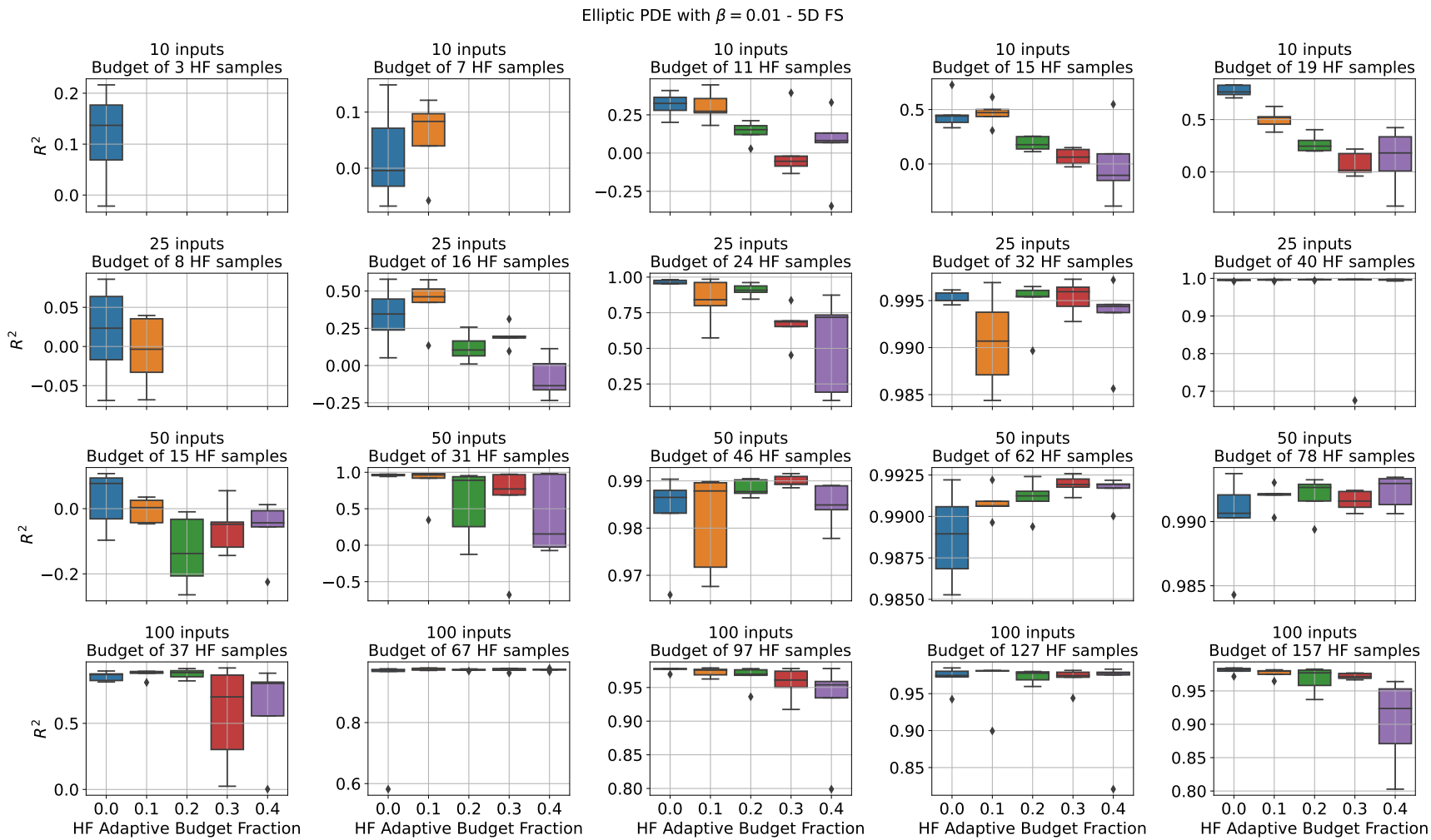


Figure 4.7: Evolution of the coefficient of determination R^2 as a function of the budget fraction used for adaptive sampling of HF observations using the proposed approach for the elliptic PDE dataset with $\beta = 0.01$ and 5D FS dataset. A budget fraction of 0 corresponds to selecting all points using the FS-based HF DOE. of predictive models trained using different points sampled using 1) a DOE in the original input space (left) or 2) a DOE leveraging the feature space (right), where the full HF analysis budget was spent on the DOE. Colors map to training repetition. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

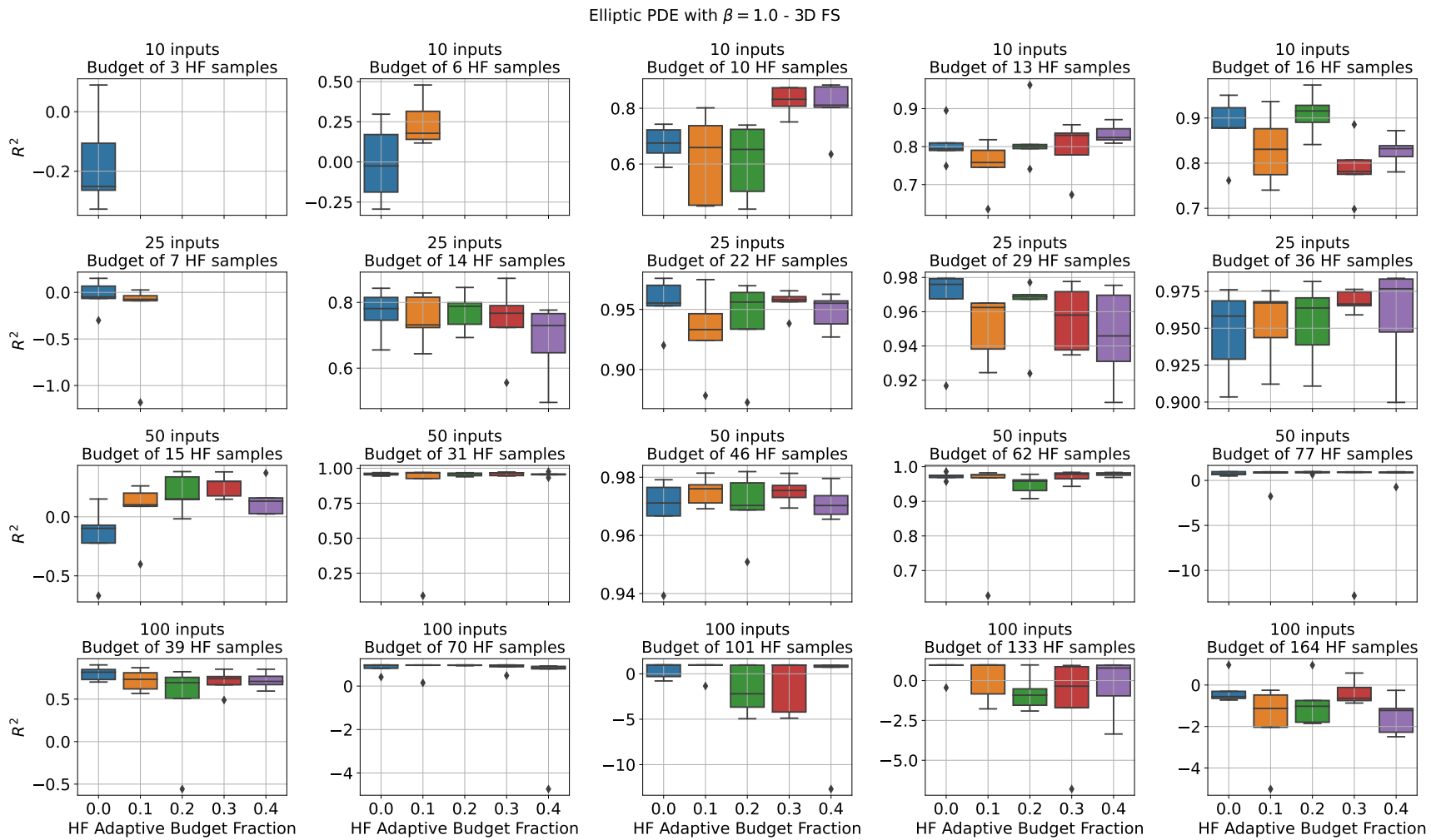


Figure 4.8: Evolution of the coefficient of determination R^2 as a function of the budget fraction used for adaptive sampling of HF observations using the proposed approach for the elliptic PDE dataset with $\beta = 1.0$ and 3D FS dataset. A budget fraction of 0 corresponds to selecting all points using the FS-based HF DOE. of predictive models trained using different points sampled using 1) a DOE in the original input space (left) or 2) a DOE leveraging the feature space (right), where the full HF analysis budget was spent on the DOE. Colors map to training repetition. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

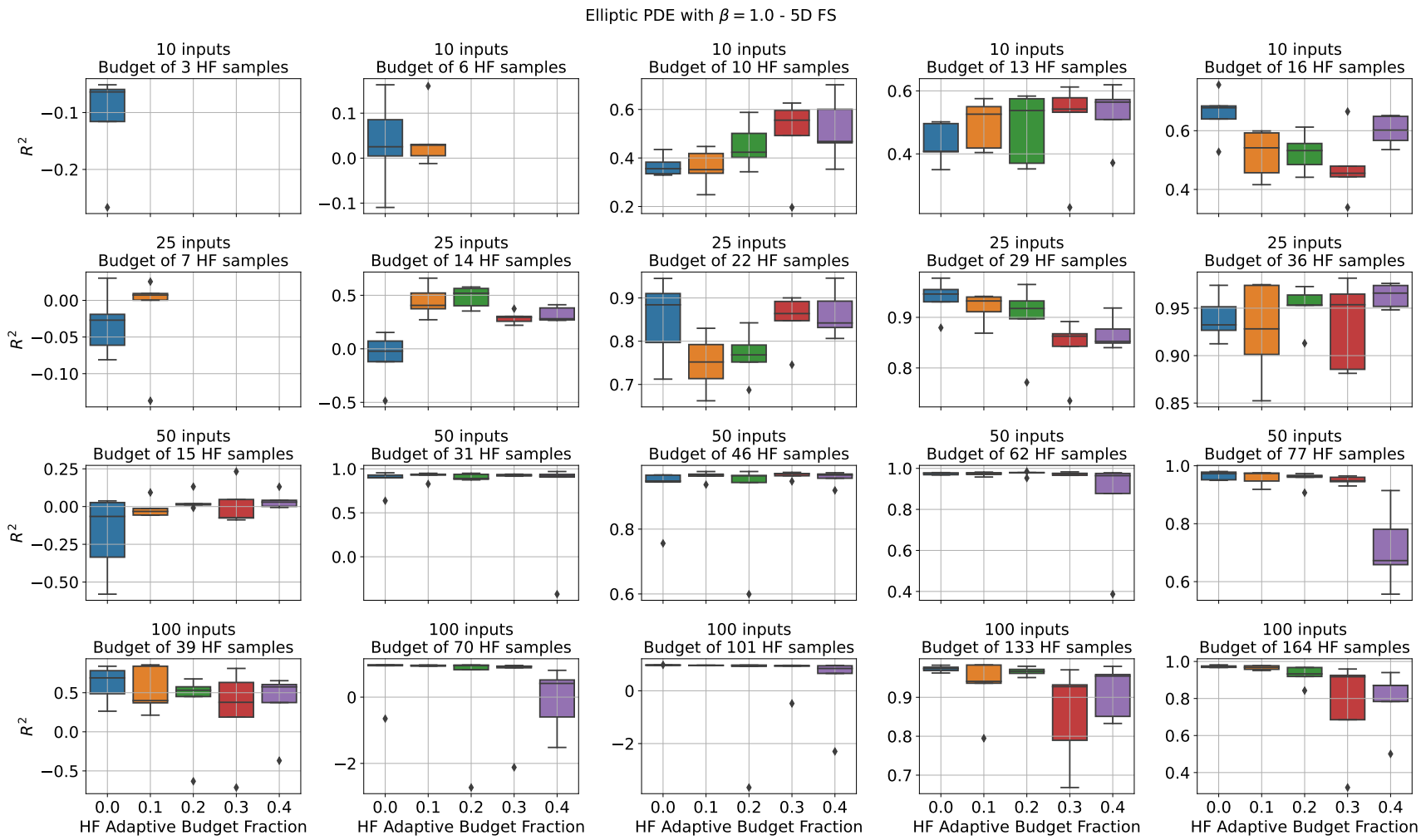


Figure 4.9: Evolution of the coefficient of determination R^2 as a function of the budget fraction used for adaptive sampling of HF observations using the proposed approach for the elliptic PDE dataset with $\beta = 1.0$ and 5D FS dataset. A budget fraction of 0 corresponds to selecting all points using the FS-based HF DOE. of predictive models trained using different points sampled using 1) a DOE in the original input space (left) or 2) a DOE leveraging the feature space (right), where the full HF analysis budget was spent on the DOE. Colors map to training repetition. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

Partial Analysis Budget

Figures 4.10 to 4.13 paint a more detailed picture than the plots observed previously. Instead of simply looking at the R^2 after the complete analysis budget is spent, we look at its evolution *as the analysis budget is being spent*. The blue line corresponds at the FS-based HF DOE. R^2 increases monotonously across all cases in this case: as we had more samples selected using the DOE to the training set, the predictive accuracy of the predictive model increases. In the previous section, we had shown that, under the condition that the FS dimension is correctly is selected, the FS-based DOE outperforms a DOE directly made in the original input space. Now, compared to the previous section, we added the colored dotted lines that correspond to the evolution of the R^2 when adaptive sampling is used. Each color correspond to a different HF DOE budget fraction. Since we first carry out the DOE, and then carry on with adaptive sampling, the dotted lines "branch out" from the main solid blue line.

The bar plots discussed in section 4.2.4 only showed us that the proposed adaptive sampling approach underperformed the DOE when the full budget was spent. In addition, these plots show us that adaptive sampling almost always underperforms the DOE from the moment we start using it. In the plots, this is visible by the fact that, as soon as a dotted line branches out from the main line, it consistently remains below it, or at best at the same level. These observations confirm the fact that the proposed approach to leveraging the FS for adaptively sampling the HF samples after an initial FS-based DOE is not beneficial to the quality of the resulting surrogate model. On the contrary, it generally decreases its accuracy.

4.2.5 Conclusion

The results discussed in the previous section clearly show the inferiority of the proposed active sampling approach compared to the DOE approach proposed earlier. The poor performance may be explained by several factors. First, the training sets used during the

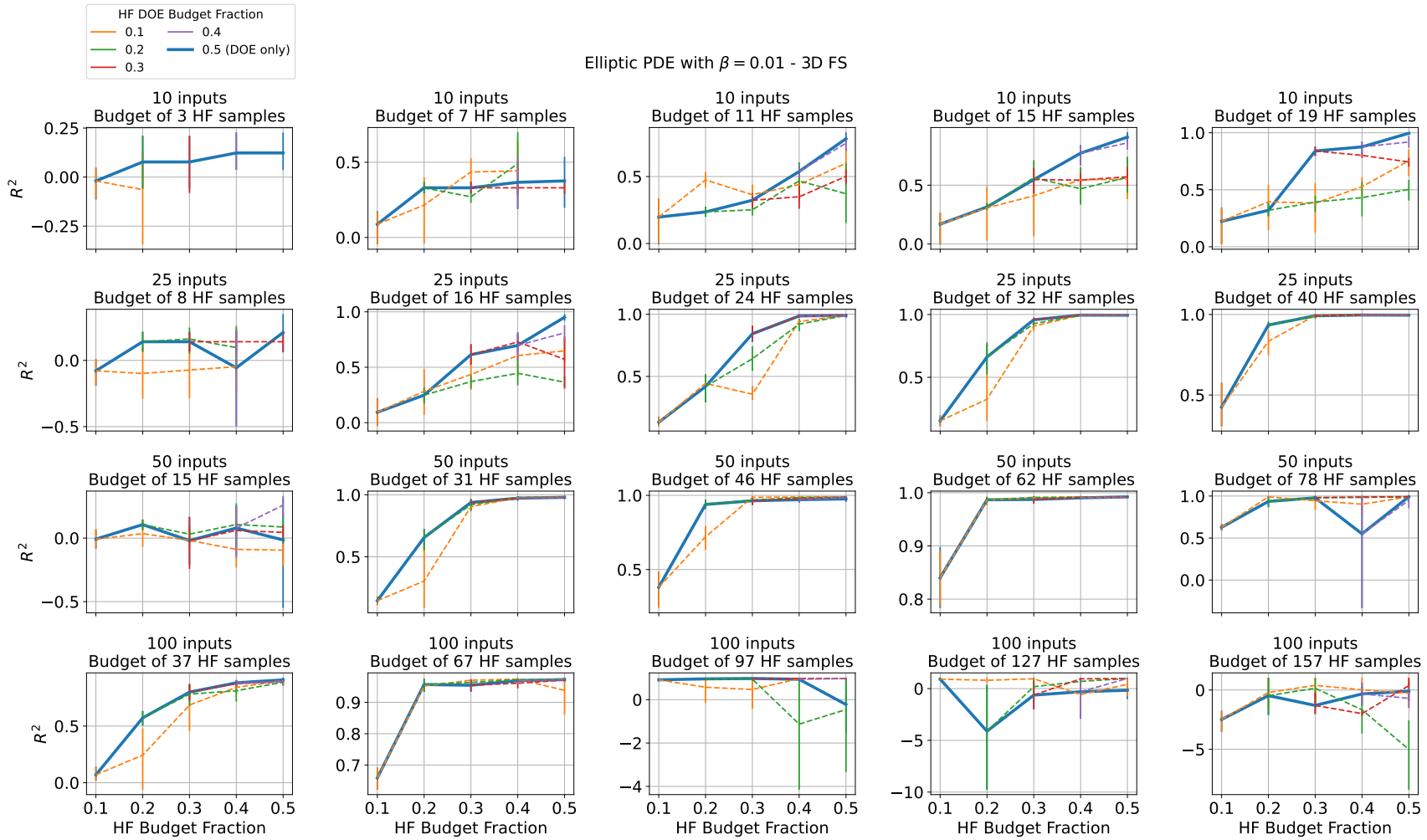


Figure 4.10: Evolution of the coefficient of determination R^2 as a function of the spent HF budget fraction for the elliptic PDE dataset with $\beta = 0.01$ and 3D FS dataset. The solid blue line corresponds to the FS-based HF DOE, while the colored dashed lines correspond to HF adaptive sampling with different budget fractions. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

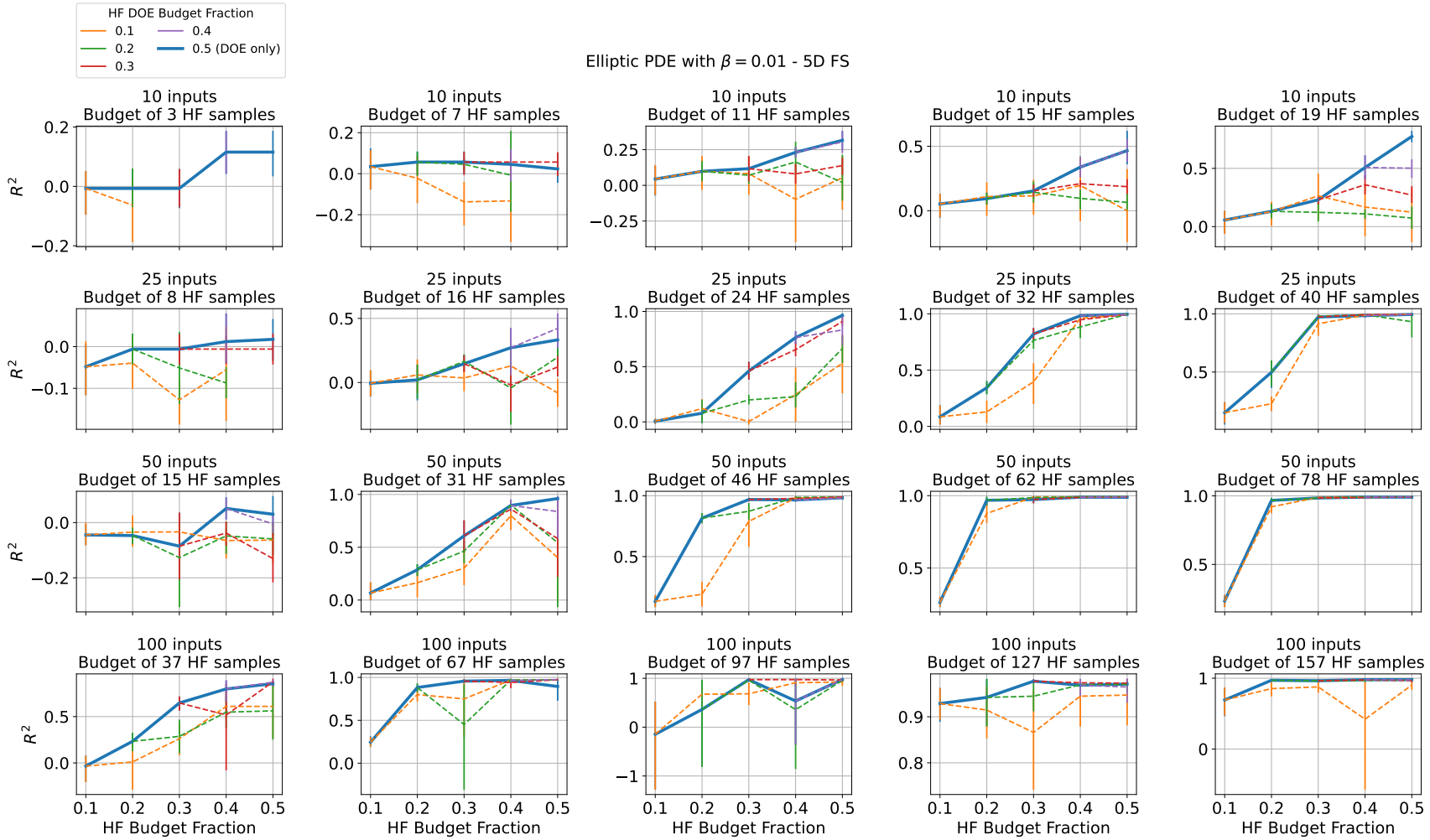


Figure 4.11: Evolution of the coefficient of determination R^2 as a function of the spent HF budget fraction for the elliptic PDE dataset with $\beta = 0.01$ and 5D FS dataset. The solid blue line corresponds to the FS-based HF DOE, while the colored dashed lines correspond to HF adaptive sampling with different budget fractions. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

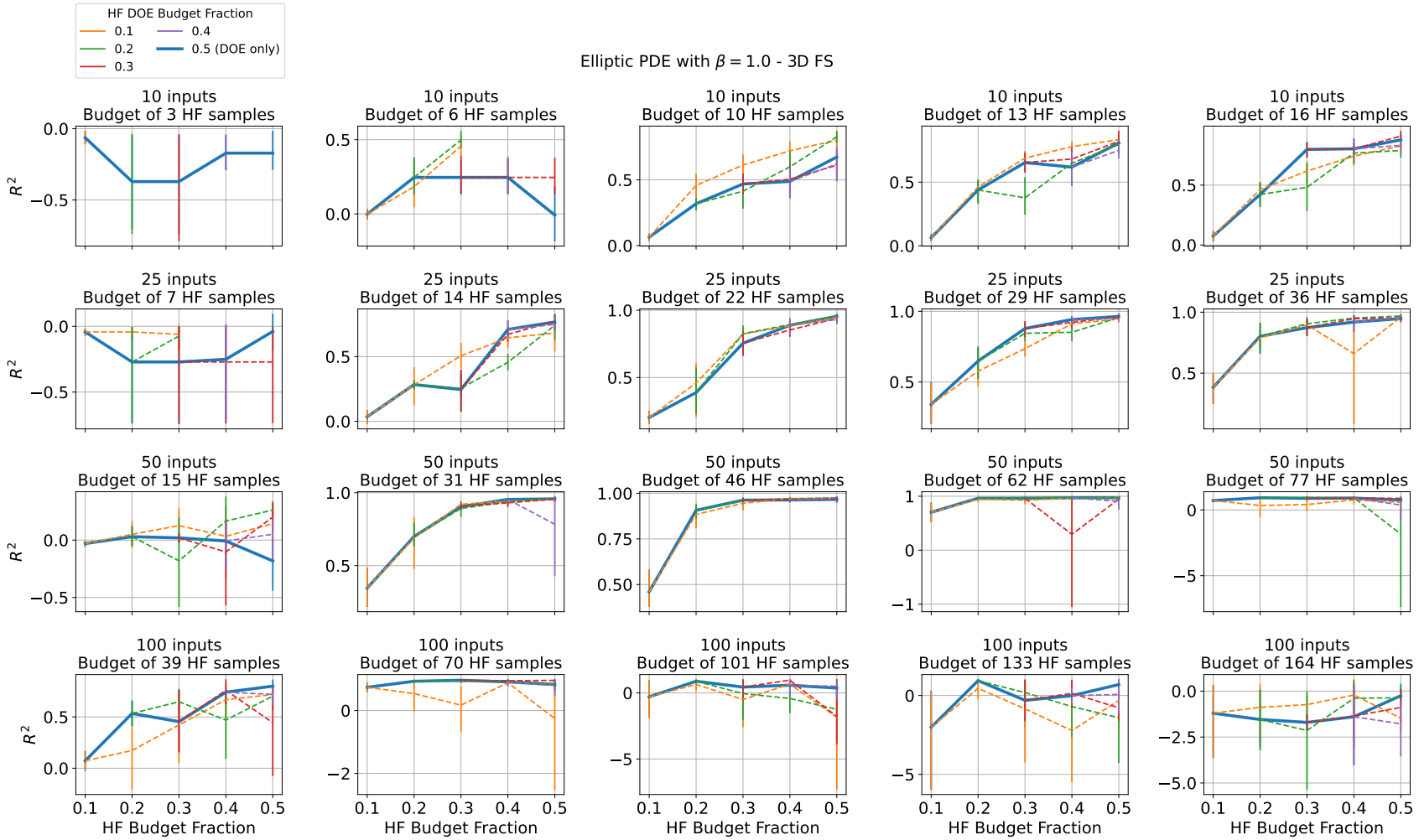


Figure 4.12: Evolution of the coefficient of determination R^2 as a function of the spent HF budget fraction for the elliptic PDE dataset with $\beta = 1.0$ and 3D FS dataset. The solid blue line corresponds to the FS-based HF DOE, while the colored dashed lines correspond to HF adaptive sampling with different budget fractions. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

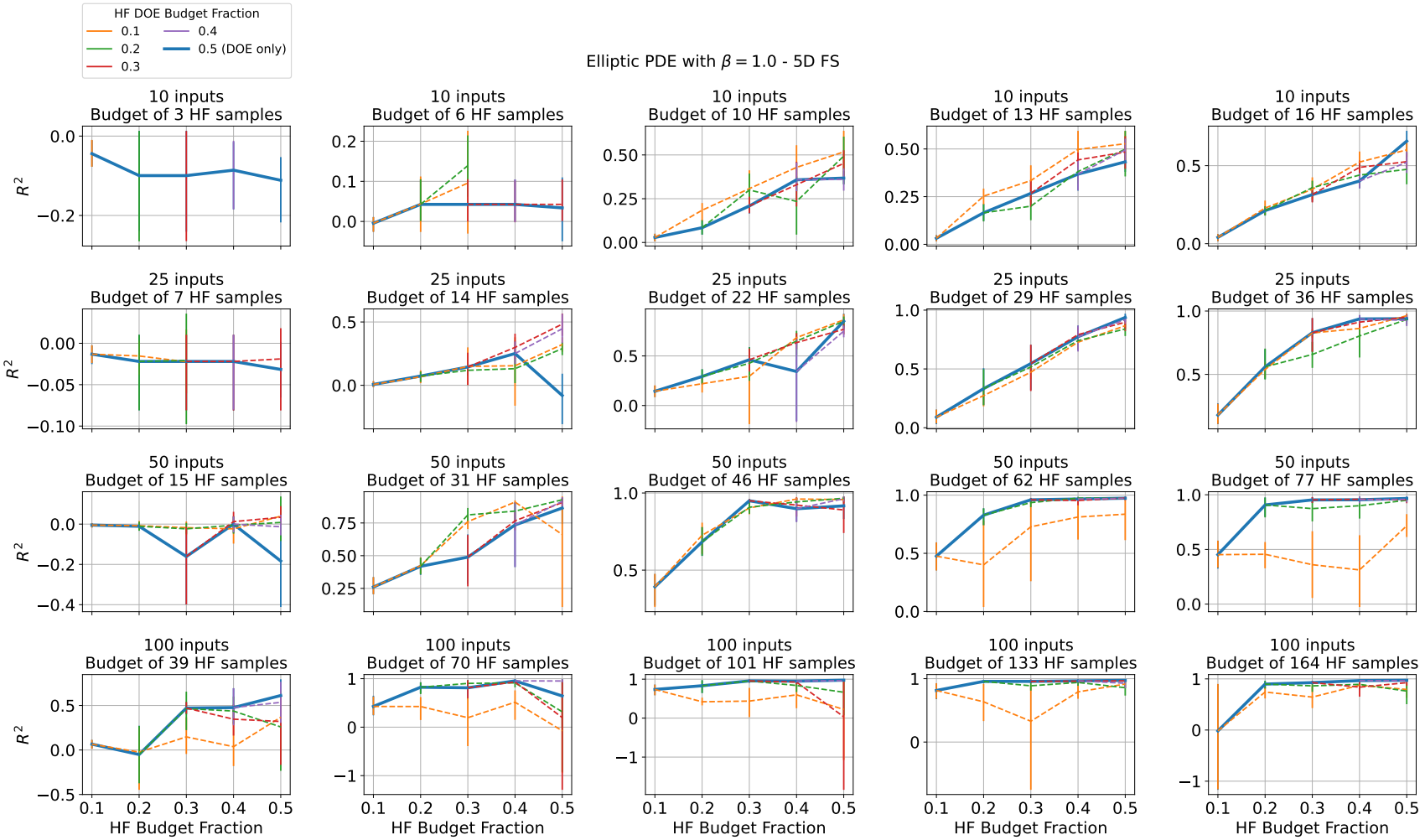


Figure 4.13: Evolution of the coefficient of determination R^2 as a function of the spent HF budget fraction for the elliptic PDE dataset with $\beta = 1.0$ and 5D FS dataset. The solid blue line corresponds to the FS-based HF DOE, while the colored dashed lines correspond to HF adaptive sampling with different budget fractions. Individual plots map to different combinations of input dimension and analysis budget: input dimension increases from top to bottom and analysis budget increases from left to right.

adaptive sampling phase are mostly unbalanced, in the sense that they contain many LF samples and few HF samples. In the previous chapter, we have seen that this leads to poor predictive accuracy. Since the selection criterion leverages the predictive capabilities of the intermediate multi-fidelity models, this may in turn impact the selection process. Second, the sampling criterion itself may be inadequate to the situation, and future work may include the adaptation of alternate selection criteria to the FS, such as IMSE.

Conclusion on Hypothesis 3.2

The results of experiment 3.2 do not support hypothesis 3.2. On the contrary, it appears that solely using the previously proposed DOE approach that leverages the LF FS leads to surrogate models with better predictive accuracy than the proposed active sampling approach.

4.3 Conclusion

4.3.1 Summary

In this chapter, we studied ways to leverage the FS that is obtained as a byproduct of the proposed method based on approximation by ridge functions to assist with the selection of new observations. In chapter 1, we had explained the impact of the curse of dimensionality when it comes to selecting training observations: in order to maintain the same sampling density, an exponential increase in the number of observations is necessary, and space-filling approaches are doomed by their inability to cover the large volume of high-dimensional spaces. Here, we started by identifying the capability gap motivating our research: existing multi-fidelity to approximation by ridge functions are either less affected by the curse of dimensionality because they leverage gradient information, or they simply do not address its impact on the initial selection of training observations. This motivated the investigation of a new DOE method to leverage the FS to initially select HF samples. We introduced a new design criterion that leverages the FS distribution obtained after hav-

ing initially trained the LF model. After having presented the experiment used to assess the proposed approach, we discussed the results and showed that using the proposed DOE instead of performing a space-filling DOE in the original input space yields significant and consistent improvement in the resulting surrogate model's predictive accuracy.

In the second part of the chapter, we attempted to follow a similar approach for active sampling: this time, instead of selecting a set of new observations, we aimed at progressively sampling the regions of the feature space in which the current surrogate model displayed the largest uncertainty. Put to the test, this approach did not yield any positive results: the predictive accuracies of the surrogate models trained using this approach were consistently lower than if the DOE approach proposed previously was used. We discussed multiple explanations for the poor behavior, including the possible unsuitability of our specific choice of LF allocation ratio or selection criterion.

4.3.2 Contributions

The main contribution of this chapter is the demonstration of the effectiveness of leveraging – in a multi-fidelity context – the FS found using a LF analysis to select the locations of the observations of the HF analysis. Beyond leading to surrogate models with higher predictive accuracy, the FS may also be used to assist sampling. The second contribution is the study of leveraging the FS for active sampling this time, which did not however yield any improvement in the accuracy of the surrogate model. Even though the considered approach will not be included in the proposed overall approach, this study can serve to inform future attempts at leveraging the FS for active sampling. Finally, as in previous chapters, all codes developed to implement experiments 3.1 and 3.2 have been made openly available and can be used to reproduce the results shown here as well as extend the approaches proposed in this work.

4.3.3 Next steps

In this chapter, we have focused on the last of the three surrogate modeling scenarios under consideration. Now, based on all the pieces developed throughout this thesis, we will propose a coherent approach to MF surrogate modeling based on approximation by ridge functions tailored to high-dimensional inputs when few analysis observations are available. To demonstrate its utility, we will be benchmarking the proposed method against a competing method previously proposed in the literature.

CHAPTER 5

VALIDATION OF THE PROPOSED METHODOLOGY AND CONCLUSION

In chapter 1, the following overarching research question was established:

Overarching Research Question

How can we alleviate the impact of the curse of dimensionality on the three surrogate modeling scenarios under consideration when only few training analysis observations are affordable?

This overarching research question was then broken down into three research areas, each corresponding to each surrogate modeling scenario:

Research Question 1

How can we alleviate the impact of the curse of dimensionality on single-fidelity surrogate modeling when only few training analysis observations are available?

Research Question 2

How can we alleviate the impact of the curse of dimensionality on multi-fidelity surrogate modeling for low analysis budgets?

Research Question 3

How can we alleviate the impact of the curse of dimensionality on DOEs and adaptive sampling for low analysis budgets?

In chapter 2, we addressed research question 1 by showing that the proposed fully Bayesian method can outperform state-of-the-art methods when only few analysis observations are available. In chapter 3, we addressed research question 2 by showing that the proposed multi-fidelity method may perform better than its single-fidelity counterpart when

analyses of multiple levels of fidelity are available. In chapter 4, we evaluated two mechanisms for informing the selection of HF observations: using the FS to 1) perform a DOE for the HF observations, and 2) perform active sampling. We could support the claim that the DOE approach was beneficial for the resulting surrogate, but not the active sampling approach. It still remains to construct a coherent method that addresses research question 3 and the overarching research question. Constructing this method will be the focus of the first part of this chapter: we will be putting together all the elements of the method that were developed and validated throughout the thesis. In the second part of the chapter, we will be concluding the thesis.

5.1 Validation of the Proposed Methodology

5.1.1 Proposed Methodology

In this section, we start by introducing the overall structure of the proposed methodology. Then, we present the various assumptions necessary to its application along with practical recommendations aimed at practitioners that would be interested in applying it to new problems or use it as a basis for future research. Then, we cover each step of the process in greater details by detailing their formulations and practical implementations. Finally, we summarize how the proposed methodology addresses the various impacts that the curse of dimensionality has on the scenario of adaptive sampling in multi-fidelity context, that were first discussed in chapter 1 and that motivated the research undertaken in this thesis.

Overview of the Methodology Workflow

The proposed methodology is shown in fig. 5.1. It incorporates individual improvements to the adaptive sampling scenario in the multi-fidelity context that were proposed and studied throughout the thesis. In this subsection, we only introduce the overall structure of the proposed methodology, as assumptions, recommendations, and further details regarding the formulation and implementation of each step of the process will be detailed in the

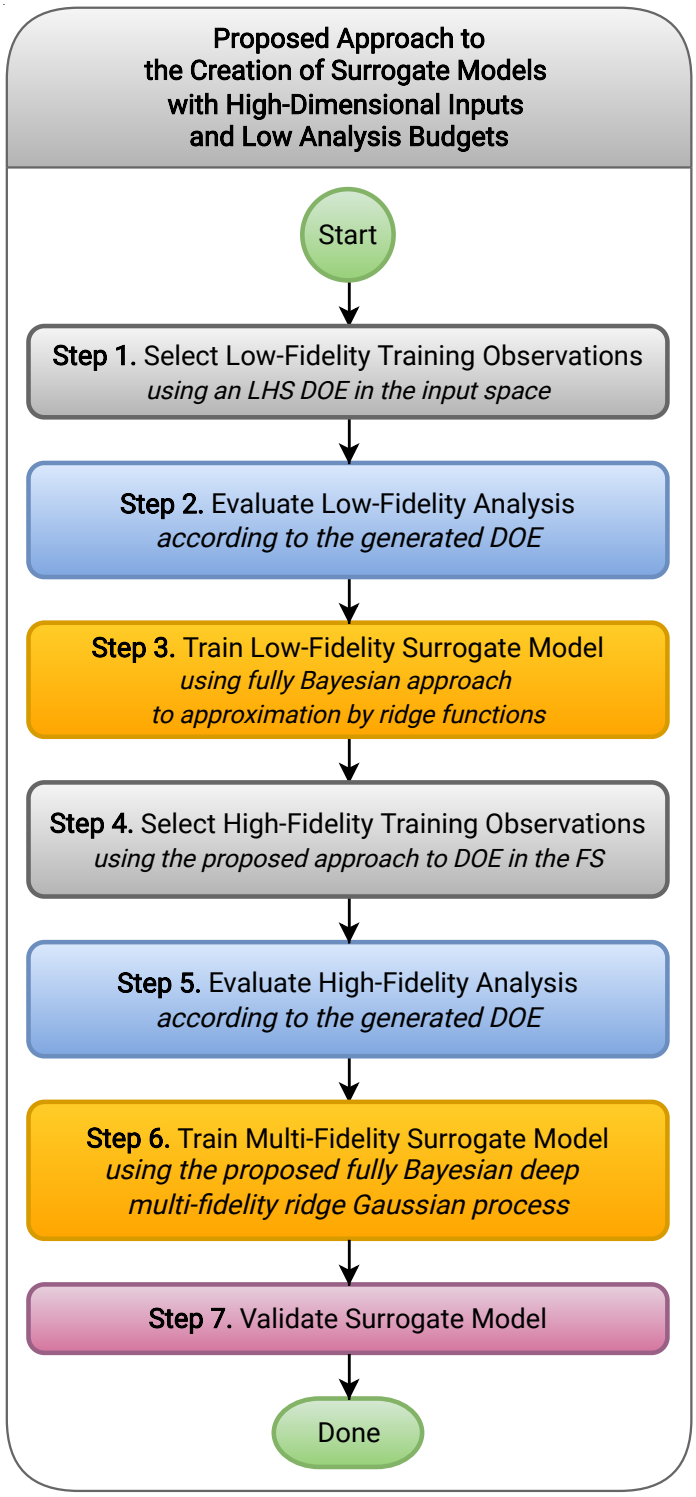


Figure 5.1: Flowchart for the proposed method

following sections.

The first step of the proposed process is to select the LF training observations. Since we have no information about the underlying analysis at that point, we use all the budget allocated to the LF analysis to perform a traditional DOE. Because the LF samples are cheaper than their HF counterparts, given the same fraction of the total analysis budget, a LF training set is significantly more abundant than a HF training set. Then, we evaluate the LF analysis at these locations and gather the initial training observations. These training observations are then used to create an initial single-fidelity model. The purpose of this model is to learn about the structure of the problem, via the knowledge of the FS. This knowledge is then leveraged in the next step in which we apply the proposed approach to DOE that leverages the LF FS. Once the new evaluation locations are known for the HF analysis, it can be evaluated. At that point, both LF and HF observations are available, making the training of the multi-fidelity model possible. The last step of the process is to validate the model before using it to make predictions.

Assumptions and Recommendations

As a starting point, we assume that two analyses of the same physical phenomenon but of different fidelities can be evaluated. These two levels of fidelity are assumed to have the same input and output variables. A generalization to different inputs and outputs is not within the scope of this thesis, but an interesting direction for future work and is accordingly discussed in 5.2.4.

In section 3.1.4, we have observed that the degree of similarity between the two levels of fidelity has an impact on the benefits brought by the MF method. If the LF and HF analyses are highly similar, then using the LF is preferred under a restricted analysis budget. On the other hand, if the LF and HF analyses are excessively dissimilar, then there is not much to learn about the HF mapping from the LF mapping, and better results are obtained when using the HF only. A quantification of the similarity between the LF and HF analyses

required for the successful application of the proposed methodology is out-of-scope of the present research. In section 5.2.4, we discuss this as a possible direction for future work, since traditional metrics used to assess accuracy, such as R^2, are not adapted in this case. Despite the lack of exact quantification, a preliminary study should be conducted, in which the level of similarity is qualitatively assessed, before applying the proposed methodology. The MF datasets presented and discussed in the context of experiment 2.1 (section 3.1.3) can be used as guidance for how similar the LF and HF analyses should be.

Moreover, it is assumed that the total analysis budget is fixed and split *a priori* between the two levels of fidelity. Linking the budget to an actual number of, respectively, LF and HF analysis evaluations, requires an estimate of the evaluation cost for each level of fidelity. If analyses are always run on the same hardware, this can be as simple as estimating their runtime duration in seconds. This also assumes that the runtime for each level of fidelity is sufficiently consistent such that the cost of each level of fidelity can meaningfully be summarized in a single number. In cases where the evaluation cost at a given level of fidelity may vary significantly from one evaluation to another, then a more elaborate cost prediction and allocation scheme should be used,but this lies outside the scope of this thesis.

The determination of a reasonable total analysis budget is highly problem-dependent, as it varies greatly based on the "complexity" of the underlying functions, that may depend on its degree of non-linearity, non-monotonous nature, frequency of variations, heteroscedastic behavior, or other singular features that make the creation of surrogate models more challenging. As a general guideline, based on the experimental results obtained in this thesis, benefits of a multi-fidelity approach have been observed for total budgets ranging from 0.5 to 1.5 times the number of inputs, when the budget is expressed in number of equivalent high-fidelity analysis evaluations. Lower budgets are expected to lead to poorly fitting surrogates, while the computational cost of the fully Bayesian approach may not be justified when a larger budget is available, as acceptable surrogate models may be used using more straightforward MLE-based approaches. Beyond the dependence on the par-

ticular problem-at-hand, the applicability of the method also depends on what is deemed a "satisfactory" surrogate. While some target applications require near-perfect predictions, others may already benefit from the identification of high-level trends. In that case, the ability to uncover elementary input-output relationships – which is particularly challenging with high-dimensional inputs – could be considered as a positive outcome.

Once the total analysis budget is fixed, the way in which it should be split between LF to HF observations, or in other words the determination of the optimal ratio of LF to HF evaluations, remains an open problem. While solving this problem was not in the scope of this thesis, some recommendations can be given based on the results of the experiments that were carried out. In particular, we have observed that highly unbalanced MF training sets, for example allocating only 10% of the budget to LF observations and the rest to HF observations, or the other way around, systematically led to poor predictive accuracies as one layer of the MF model or the other is not properly trained. Based on experimental results presented in this thesis, allocating 30% to 60% of the budget to LF observations generally led to best results.

Accordingly with standard machine learning practices, the training datasets should be properly pre-processed before applying the proposed methodology. The input locations of the training observations are assumed to be centered and have unit variance. For observations with uncorrelated input locations, this can simply be achieved by subtracting the mean and dividing the centered locations by the standard deviation. For observations that have correlated input locations, which can for example happen if analysis evaluations at the edge of the design space have failed, then they should be de-correlated before applying the proposed methodology. PCA can be used for this purpose, by retaining all dimensions, and simply using the eigenvectors to effectively rotate the inputs to a frame of reference in which they are uncorrelated. Accordingly with the zero-mean assumptions of the GPR model used, the outputs of the observations should be centered. To ensure compatibility with the prior distributions of the GP's hyperparameters, in particular the signal and noise

variances, the outputs should also be scaled. In the experiments carried out in this thesis, outputs were divided by the standard deviation of the training dataset to obtain unit variance.

Formulation and Implementation of the Proposed Methodology

In this section, we discuss the formulation and implementation of the individual steps that make up the overall process pictured in fig. 5.1. The goal of the process is to create a surrogate model for an expensive high-fidelity analysis, for which an alternate low-fidelity analysis is available, under a limited analysis budget that severely restricts the number of HF observations that could be obtained. We recall that the Python implementation of this process has been made openly available online. The assumptions discussed in the previous section should be satisfied at the start of the process.

Step 1 The first step of the process is to start gathering training observations. We have seen that the curse of dimensionality impacts space-filling DOEs, making them effectively sparse. Therefore, opting for a space-filling DOE in the first step of the process may appear contradictory. However, at this stage, without having gathered any knowledge about the underlying analysis, and without prior knowledge or making assumptions regarding the mathematical structure of the analysis (e.g., additivity, dependence on low-dimensional manifolds etc.), no other option is available to us.

As discussed previously, multiple options are however available to us when it comes to which analysis should be evaluated: in the MF context, we have the choice of which level of fidelity to evaluate. Moreover, we also have the choice of whether the budget allocated to each fidelity should be spent all-at-once initially or in multiple stages following, e.g., an adaptive sampling approach. Here, we propose to start by evaluating the LF analysis only and to spend all the budget that was allocated to it all-at-once. The motivation for this approach is to extract as much information as possible about the structure of the un-

derlying analysis early on, such that this information can then be used to select the training locations for the more costly HF analysis. In the context of approximation by ridge functions, the structural information that we wish to extract using the LF model is a relevant feature space. As we found in experiment 3.1, leveraging the feature space to select HF observations leads to surrogate models with higher predictive accuracy than if both LF and HF observations had all been selected at the same time using a DOE. The implementation of this step is straightforward, as LHS is one of the most common DOE techniques. In our implementation, we use the LHS generator provided by the Python package `pydoe2`.

Step 2 The next step consists in evaluating the LF analysis using the locations provided the the generic LHS design generated at the previous step. Although the computational cost of this step depends on the analyses under consideration, this is a straightforward step of the process. The implementation of this step is highly dependent on the nature of the analysis under consideration. We simply assume that the outcome of this step is a data table that maps all the input locations of the LHS DOE to the corresponding LF response value. If cases have failed, the recommendations given previously in section 5.1.1 should be followed to ensure that the training data is uncorrelated.

At that point, we have an initial training set made up of LF observations only. As discussed previously, the motivation for gathering these observations was to extract structural information regarding the input-output relationship under consideration in form of a low-dimensional feature space. Now, the feature space is obtained as a byproduct of the proposed method for approximation by ridge functions. Indeed, since the first step of the process is to project the original inputs onto a lower-dimensional feature space, the projection matrix becomes a parameter of the predictive model, which is accessible after training. More precisely, because we adopted a Bayesian approach such that the method could be applicable to small training sets, the posterior distribution for the projection matrix is obtained, i.e., the probability distribution of these parameters *given the observed training*

data.

Step 3 The next step of the process is therefore to apply the single-fidelity version of the fully Bayesian approach to approximation by ridge function. The formulation and implementation for this method was already discussed in details in section 2.2.2 and will therefore not be reproduced here. In summary, the formulation consists in the definition of a set of conditional probability distributions that, when combined with prior distributions for all the model parameters, fully define a probabilistic model. As opposed to a deterministic predictive model, probability distributions for the model parameters are sought instead of point values. Assuming that the recommended pre-processing of the training data has been carried out as detailed in section 2.2.2, the implementation of this step of the process consists in expressing the probabilistic model into a probabilistic programming language, and using an MCMC sampler to draw from the posterior distribution. In the specific implementation used in this thesis, the `numpyro` framework was used, that internally relies on the `JAX` computational backend, and the NUTS sampler was used for MCMC. The knowledge of a probabilistic programming language, along with the equations, are sufficient to the implementation of this step. In addition, our implementation was made available and can be used for further reference, see the link given above. The outcome of this step is a trained model, or in more concrete terms, draws from the joint posterior distribution of all model parameters.

Step 4 In the two-step approach to building the training set that we adopted, it is now time to select the HF analysis observations. Unlike our initial situation, in which we had no knowledge about the underlying analysis, building a LF surrogate model based on approximation by ridge functions now allows us to extract information about mathematical structure of the problem that we can leverage to inform the selection of the HF observations. This was exactly the point of the DOE approach studied in experiment 3.1: instead of a space-filling design that is bound to be sparse in the original high-dimensional input

space, the proposed design criterion mainly aims at filling the *low-dimensional FS*.

In terms of implementation, the first step is to determine the number N_{HF} of HF observations that make up the experimental design. This can be obtained based on the budget allocated to the HF observations as well as the estimate for the HF evaluation cost. These two quantities were assumed to be available prior to applying the methodology, see section 5.1.1. From there, the implementation is identical to what was detailed in the context of experiment 3.1. The design criterion introduced in eq. (4.1) is optimized using a gradient-based solver, with the gradients automatically computed using automatic differentiation provided by the JAX computational backend. Here, scipy's implementation of the L-BFGS-B algorithm to perform the optimization. All implementation details needed to ensure reproducibility are available in the openly available code. The outcome of this stage is an experimental design for the HF analysis, i.e., a list of locations in input space at which the HF analysis will be evaluated.

Step 5 Now that an experimental design for the HF observations is available, the HF analysis can be evaluated. As in step 2 for the low-fidelity model, both the implementation and the computational cost of this step are highly analysis-dependent. We only assume that evaluating the HF analysis yields a data table that associates the input locations to the corresponding response values. As for the LF analysis, recommendations previously given should be followed to ensure that inputs are uncorrelated in case, e.g., analysis evaluations have failed.

Step 6 At this stage, training observations for both the LF and HF analyses have been gathered, and a MF model can be created. As the single-fidelity case, this step consists in defining the conditional probability and prior distributions that make up the probabilistic model. Then, the model is implemented using a probabilistic programming language, which was numpyro in our case. Then, the NUTS sampler provided by numpyro was used to draw from the joint posterior distribution of the MF model parameters. Details on the

MF model were previously given in section 3.1.2. The outcome of this step is the trained MF model that can be used to make predictions, or more precisely draws from the joint posterior distribution of the multi fidelity model's parameters.

Step 7 Before using the trained model to make predictions, it should be validated using analysis observations that were not used to train the model in order to ensure that the predictions generalize well. This step is highly dependent on the target application, as acceptable predictive accuracy may highly depend on the task-at-hand. Standard machine learning practices suggest to use 80% of available observations for training the model, and retaining 20% for validating the model. Alternatively, more elaborate methods may also be used, such as k-fold cross-validation. Validation approaches are not specific to the proposed methodology, they are rather a generic step of machine learning workflows, and each organization may have different requirements for the validation of these workflows. Therefore, we are not specifying a detailed validation process here, and organizational protocols regarding the validation of machine learning models should be followed.

Summary of the Mechanisms Used to Alleviate the Impact of the Curse of Dimensionality

In chapter 1, we had discussed the ways in which the curse of dimensionality impacts the different surrogate modeling scenarios considered in this thesis, and in particular adaptive sampling in the multi-fidelity context (fig. 4.1). In this subsection, we summarize how these impacts are alleviated in the proposed methodology.

The first impact concerns the training phase of the surrogate models: the curse of dimensionality leads to a rapid increase in the number of parameters for parametric methods, or distorted notion of distance and neighborhood for non-parametric methods, which are only amplified when the available number of training observations is low. This was addressed, first in chapter 2 in the single-fidelity case, then in chapter 3 for the multi-fidelity case, by the proposed fully Bayesian approach to approximation by ridge functions. By

projecting inputs onto a lower-dimensional subspace of the original input space, i.e. the feature space, we effectively reduce the dimension of the inputs. Because the dimension of the inputs has been reduced, the impact of the curse of dimensionality is accordingly diminished. The fully Bayesian approach allowed to extend the applicability of such supervised dimension methods, which usually require many training samples, to situations where analysis budget – and therefore the number of affordable training observations – is limited. The benefits of the fully Bayesian approach to approximation by ridge functions were demonstrated in experiments 1.1 and 2.1.

The second impact concerns the selection of training observations, either during the initial DOEs or during the application of adaptive sampling This was addressed in chapter 4, where we first proposed an approach to 1) acquire knowledge about the underlying analysis using the LF analysis, and 2) use this knowledge – through the FS – to improve the DOE used to select initial HF observations. In simple terms, the proposed approach can be thought of as performing a DOE in the reduced feature space instead of the original input space. While it does not fit the mold of traditional active sampling methods based on a selection criterion leveraging some estimate of the surrogate model's predictive variance, the proposed approach to DOE may be considered an *adaptive* strategy in the sense that it leverages an existing surrogate model in order to assist the selection of the training observations used to produce the next iteration of the model. Although the more traditional approach to active sampling leveraging the FS did not yield satisfactory results, it was observed that this FS-based DOE approach brings significant performance gains compared to a traditional DOE, helping to alleviate the impact of the curse of dimensionality on the selection of new training observations in a high-dimensional input space.

Overarching Hypothesis

We have already validated the individual elements that make up this method. Now, we will be validating that, once put together, they outperform existing methods, leading to the

following overarching hypothesis:

Overarching Hypothesis

When inputs are high-dimensional and analysis budget is low, the proposed approach to multi-fidelity surrogate modeling leads to better predictive accuracy than a state-of-the-art multi-fidelity surrogate modeling method because it combines 1) a reduction of the dimension of the input space and 2) a selection of high-fidelity observations leveraging the feature space to address the impact of the curse of dimensionality, and 3) a fully Bayesian approach to address the lack of training observations.

In the next section, we will move on to the presentation of the experimental setup used to check the overarching hypothesis.

5.1.2 Setup of Experiment 4

Need for Experiment 4

In the previous experiments, we have shown the different parts that make up the proposed methodology bring improvements to the predictive accuracy when taken individually. Now, we need to check that these benefits also exist when these individual elements are brought together as part of the proposed methodology, which is the point of this experiment. In order to test the overarching hypothesis, we need to compare the predictive accuracies of 1) a surrogate model created using the proposed methodology to 2) a surrogate model using a state-of-the-art method that does not include any of the improvements developed in this thesis. Since the proposed method is novel, an experiment is required to gather the information used as a basis for comparison.

Experiment Design

In order to test the overarching hypothesis, we need to compare the predictive accuracy of the proposed approach to a state-of-the-art method that has none of these features. The

deep multi-fidelity Gaussian process appears as a valid candidate, as this is a recently developed method, that has been shown to perform better than traditional multi-fidelity surrogate modeling techniques such as the ARGP, but 1) it does not include a mechanism for reducing the dimension of the input space, 2) it only follows an approximate inference approach relying on SVI instead of exact inference, and 3) the training observations are selected using a traditional LHS DOE in the original input space.

DMF-GP has already been discussed in chapter 3 since it is used as basis for the proposed B-R-DMF-GP method. The coefficient of determination (R^2) will be used as a metric for predictive accuracy.

The elliptic PDE analysis already used in previous experiments will also be used here. It is representative of engineering analyses encountered in the aerospace context, while being easier to work with. The setup of this analysis allows us to modify both the dimensions of its inputs and its "complexity" in a straightforward manner. As before, we will be considering two levels of complexity, set by the length scale parameter: $\beta = 0.01$ (short length scale, more complex), and $\beta = 1.0$ (long length scale, less challenging). Input spaces of dimensions 10, 25, 50, and 100 are considered. These parameters were discussed in section 2.2.3.

For cases corresponding to the benchmark DMF-GP method, the following steps are followed: 1) generate LF and HF training data, 2) train the DMF-GP model, 3) validate surrogate model, and 4) save results.

For cases corresponding to the proposed B-R-DMF-GP method, the following steps are followed: 1) generate initial LF training data, 2) train the single-fidelity LF model, 3) perform the FS-based DOE to select HF training data, 4) train the B-R-DMF-GP model, 5) validate surrogate model, and 6) save results.

The source code used to generate the results presented thereafter has been made publicly available[1].

[1] https://gitlab.com/raphaelgautier/thesis_experiments_part4

Sensitivity Study

As for other experiments, we expect multiple parameters to have an impact on predictive accuracy. The analysis-related parameters, input dimension and complexity, were already discussed above. Parameters related to the training set include: the total analysis budget, LF and HF allocation ratios, and the number of training repetitions. The same values are used as for experiments 3.1 and 3.2. The analysis budget depends on the input dimension and is detailed in table 5.2. The LF and HF allocations ratios are kept equal to avoid situations where unbalanced multi-fidelity training sets lead to poor predictive accuracy. Five training repetitions are performed for every case. We use two settings for the FS dimension (3 and 5) instead of performing a full sweep in order to reduce the computational cost of this experiment. Since it was shown to perform best, we use the same FS for both the LF and HF layers of the B-R-DMF-GP model.

The proposed B-R-DMF-GP method is compared to the DMF-GP method from the literature. We define ranges for numerical parameters and generate a full-factorial DOE to study the variations of all parameters (see table 5.1).

Table 5.1: Summary of the parametric study conducted in experiment 4

Group	**Parameter**	**Parameter Values**
Analysis	Input Space Dimension	[10, 25, 50, 100]
	Complexity (parameter β)	[0.01, 1.0]
Training Set	Analysis Budget	see table 5.2
	LF Allocation Ratio	fixed at 0.5
	HF Allocation Ratio	fixed at 0.5
	Number of Training Repetitions	5 repetitions with fixed random seeds
Method	FS dimension	[3, 5]
	LF/HF FS relationship	fixed (shared FS)

Table 5.2: Experiment 4 – Analysis budget (in HF samples) as a function of the analysis input dimension.

Input Dimension	**Analysis Budget (in HF samples)**
10	[4, 8, 12, 16, 20]
25	[10, 15, 20, 25, 30]
50	[20, 30, 40, 50, 60]
100	[70, 80, 90, 100, 110]

Presentation and Interpretation of the Results

As in the previous experiments, a comparison of predictive accuracy is made using R^2 as a metric. For each level of complexity of the elliptic PDE dataset, a grid plot will be used, in which the input dimension increases from top to bottom and the analysis budget increases from left to right. A box plot is used to account for the distribution of R^2 obtained in the repeated training runs. Similarly to what was done in previous experiments, the dimension selection method developed in chapter 2 will not be used, as it is not suited to being automated. Instead, two settings for the FS dimension are considered (3 and 5) and it is assumed that the FS dimension leading to the highest predictive accuracy would be selected. Each plot will therefore display three distributions: first the predictive of predictive accuracy using the benchmark method (deep multi-fidelity Gaussian process) to the left, and then the distributions obtained when using the proposed method with a 3D FS (in the center) and a 5D FS (to the right). For the results to support the overarching hypothesis, for at least one of the two considered FS dimensions, the predictive accuracy obtained when applying the proposed approach should be higher than when using the original deep MF GP, consistently across training repetitions.

Given a dataset and an analysis budget, we visualize the distribution of R^2 as a function of the surrogate modeling method used. For results to support the hypothesis, the proposed B-R-DMF-GP approach should lead to a predictive model exhibiting higher predictive ac-

curacy than the benchmark DMF-GP method.

5.1.3 Results of Experiment 4

Figures 5.2 and 5.3 offer a comparison of the R^2 values for the benchmark DMF-GP method as well as the proposed B-R-DMF-GP method with 3- and 5- dimensional FS for the elliptic PDE datasets and for various combinations of the input dimension (increasing from top to bottom) and analysis budget (increasing from left to right).

Let us first focus on the results obtained when $\beta = 0.01$ shown in fig. 5.2. We observe that the proposed approach outperforms DMF-GP in all situations under the condition that the correct FS dimension is selected. In particular, in the higher-dimensional cases with 50 and 100 inputs, DMF-GP leads to models with negative R^2 irrespective of the number of training samples, whereas the proposed approach leads to models with R^2 values close to 1 with only intermediately sized training sets. These results recall the importance of appropriately selecting the FS dimension, as we observe that it greatly affects the predictive performance of the proposed approach. In the cases with 10 inputs and budgets of 7 to 15 HF samples, we see that choosing a 5D FS instead of a 3D one leads to R^2 values of the same order as the benchmark method. In that case, a lower-dimensional FS is more appropriate. This makes sense, as we are working with a relatively low-dimensional input space, and few training samples. The situation is reversed in the higher-dimensional cases with 100 inputs and budgets between 97 and 157 HF samples: in these instances, selecting a 3-dimensional FS leads to large variations in the predictive performance, while using a 5D FS consistently leads to models with high predictive accuracy. Once again, this makes sense, as 3 dimensions may be insufficient to capture the variation of the original 100 dimensions.

Let us now turn to the case where $\beta = 1.0$ shown in fig. 5.3. As opposed to the previous dataset, DMF-GP outperforms the proposed approach in the 10-dimensional case. This may be explained by the facts that 1) the physics behind the $\beta = 1.0$ case are sim-

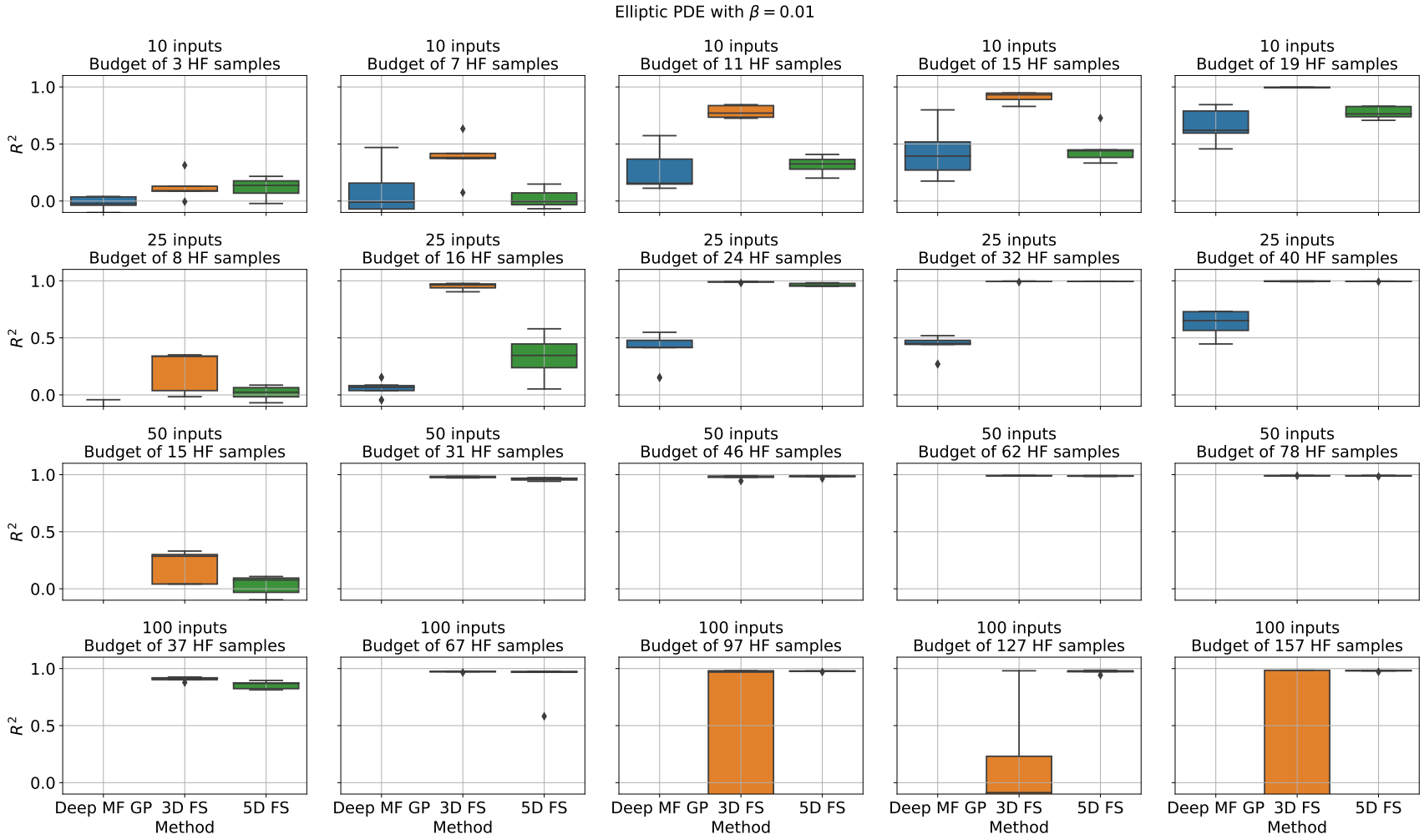


Figure 5.2: Comparison of the coefficient of determination R^2 obtained with 1) the benchmark deep MF GP method (left) and 2) the proposed deep MF ridge GP leveraging the FS-based HF DOE for the elliptic PDE dataset with $\beta = 0.01$ dataset. The y-axis bound were set to the interval $[0, 1]$ for readability.

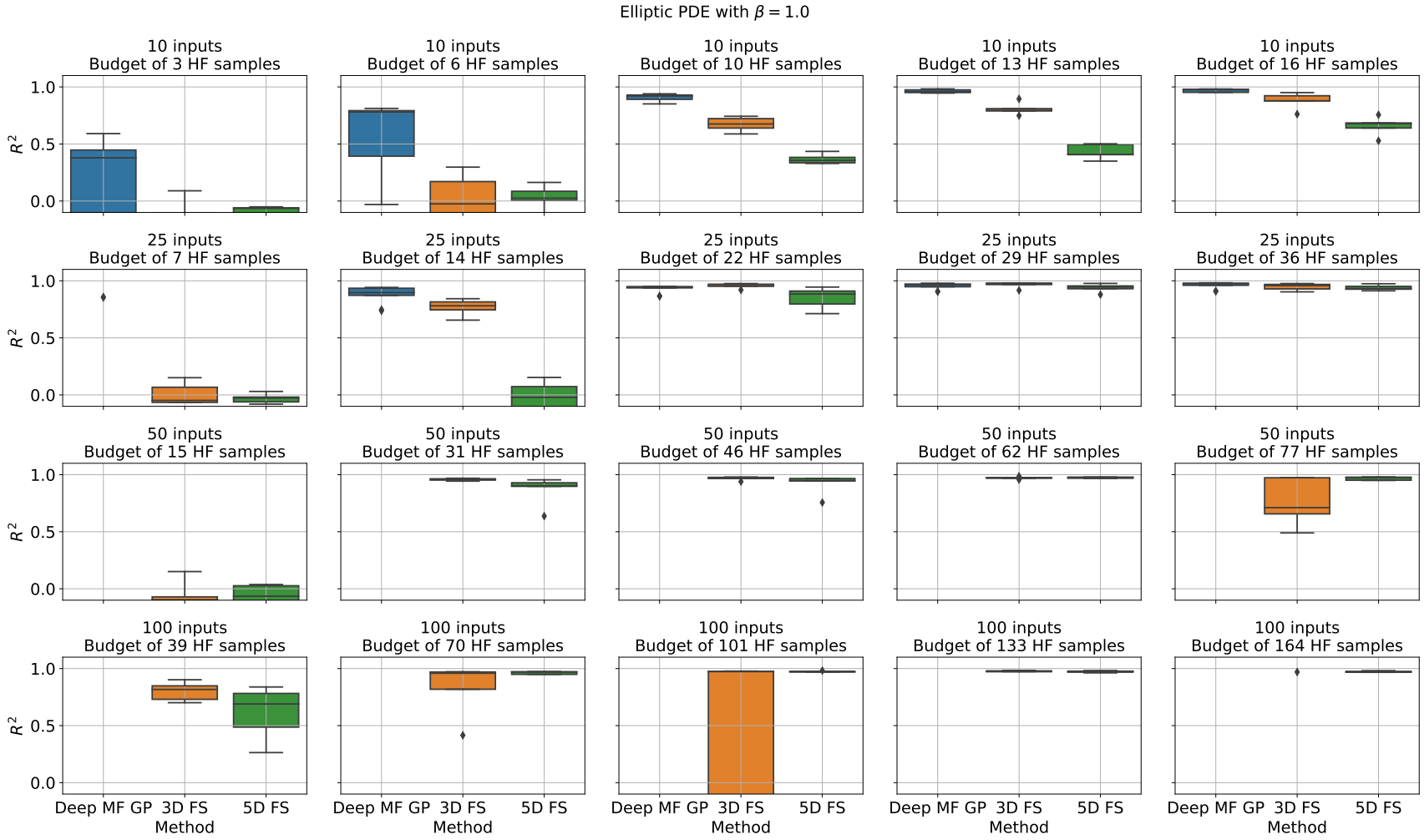


Figure 5.3: Comparison of the coefficient of determination R^2 obtained with 1) the benchmark deep MF GP method (left) and 2) the proposed deep MF ridge GP leveraging the FS-based HF DOE for the elliptic PDE dataset with $\beta = 1.0$ dataset. The y-axis bound were set to the interval $[0, 1]$ for readability.

pler than for $\beta = 0.01$, and 2) the input space is relatively low-dimensional. Under these conditions, we expect surrogate modeling methods that do not specifically aim to address high-dimensional inputs to perform relatively well. In the 25-dimensional case, DMF-GP outperforms the proposed approach for a budget of 14 HF training samples, but performance of the different methods is very similar for larger analysis budgets. In the two highest-dimensional cases, DMF-GP still struggles, leading to negative R^2 values, whereas the proposed approach eventually leads to R^2 values near 1. The impact of the FS dimension is still visible in this case, with multiple cases where an inappropriate selection leads to lower R^2 values or large performance variations.

5.1.4 Conclusion

Except for relatively low-dimensional inputs and the least complex of the considered analyses, the proposed approach outperforms the state-of-the-art deep multi-fidelity Gaussian process method. In particular, for input dimensions of 50 and 100, the proposed approach consistently allows to reach high predictive accuracies if the correct FS dimension is used while DMF-GP fails to make relevant predictions. For relatively lower input dimensions, the proposed method grants significant accuracy improvements in the case of the more complex analysis ($\beta = 0.01$), by reaching R^2 values close to 1.0 while DMF-GP remains at or below 0.5. However, for simpler analyses ($\beta = 1.0$), the proposed approach is outperformed by DMF-GP. This may be due to the projection of the inputs becoming detrimental in this case, or the fact that another FS dimension may have be more appropriate. This leads to the following conclusion:

Conclusion on the Overarching Hypothesis

For high-dimensional inputs or complex problems, the proposed approach outperforms the deep multi-fidelity Gaussian process when the right FS dimension is selected.

5.2 Conclusion

5.2.1 Thesis Summary

In chapter 1, we laid out the motivation for this thesis. Surrogate modeling was first presented as an enabler for modern approaches to design, such as design space exploration, engineering optimization, or uncertainty quantification: we explained that while these approaches typically required the costly online evaluation of the underlying analyses, a one-time offline evaluation cost was only needed when working with surrogate models. We then presented the two-fold challenges that they increasingly face as the needs of engineering design evolves. We first noted that increasingly expensive analyses were used to inform decisions earlier on in the design process, impacting the amount of training data available for training surrogate models under a limited analysis budget. We then observed that high-dimensional parameter spaces were increasingly used, motivated by a more thorough exploration of the design space, the consideration of novel configurations, or the desire to retain design freedom, impacting our ability to train accurate surrogate models due to the curse of dimensionality. From these challenges, we identified the main gap that motivates the research undertaken in this thesis: without the ability to create surrogate models for expensive analyses with high-dimensional inputs, we could not carry out the higher-level many-query applications used to inform design decisions. From there, we scoped down our focus to three surrogate modeling scenarios that we identified as being suitable for expensive analyses: single-fidelity surrogate modeling, multi-fidelity surrogate modeling, and active sampling in the multi-fidelity context. This chapter ended with the definition of the three research areas of the thesis, that mapped to each of the surrogate modeling scenarios and that would conveniently and progressively build on each other.

In chapter 2, we focused on the first research area, single-fidelity surrogate modeling. After a literature review of existing methods to address surrogate modeling with high-dimensional inputs, with a focus on methods relying on approximation by ridge functions,

we observed that existing methods were limited by their need for many observations or gradient evaluations. This motivated the search for an alternative way of training such models, and we discussed the benefits brought by working in the Bayesian framework to make existing approaches more robust to the sparsity of training data. However, we explained why the transposition of existing methods to this framework was not straightforward and proposed to address it. Specifically, we leveraged methods recently proposed in the literature on MCMC to include orthogonal matrices as parameters when using turn-key MCMC algorithms, enabling the practical implementation of the proposed fully Bayesian formulation. We showed that the proposed approach performed better than existing competing approaches on a range of analytical, scientific, and engineering applications. Then, in the rest of the chapter, motivated by limits of existing methods, we turned to a specific aspect of the supervised dimension reduction, the selection of the dimension of the low-dimensional FS. First, we justified and demonstrated the use of the noise variance as a metric for selecting a suitable dimension. Then, we aimed to reduce the cost of the dimension selection process by enabling the reuse of information when multiple FS dimensions are being considered, and showed that the approach was indeed beneficial under certain conditions.

In chapter 3, we tackled the second research area, multi-fidelity surrogate modeling. We started by pointing out the limits of the few existing multi-fidelity approaches to approximation by ridge functions, motivating the development of a new methods that does not require access to gradients and performs better under a limited analysis budget. Among different options, we chose to use the deep multi-fidelity Gaussian process model as a foundation for extending the single-fidelity approach to the multi-fidelity context. After formulating the probabilistic model of the proposed approach and discussing the specifics of its implementation, we conducted a thorough study that showed that, under a fixed analysis budget, it may under certain conditions lead to predictive models with higher predictive accuracy than if a single fidelity has been used, thus enabling to make better use of a limited computational budget. Then, we turned to a specific aspect of the proposed multi-fidelity

probabilistic model, namely the approach to modeling the FS for the LF and HF layers of the proposed approach. The study that we conducted confirmed the initial choice of sharing the FS across the layers of the model in order to enable the transfer of information from the more numerous LF observations.

In chapter 4, we turned to the third research area, sampling strategies in the multi-fidelity context. A review of existing literature pointed out that most active sampling strategies proposed previously focus on BO instead of global metamodeling. In particular, no approach tackles the initial selection of the HF observations, that are particularly costly. Based on this, we proposed a multi-step approach to the selection of the initial LF and HF observations, in which LF observations are first used to construct an initial surrogate model whose byproduct if an estimation of a relevant low-dimensional FS. In the second step, the selection of the HF observations is assisted by the characterization of this low-dimensional FS. We designed and conducted a study that showed the superiority of the proposed approach compared to directly selecting all points using a DOE in the original high-dimensional input space. Then, we turned to the problem of active sampling for global metamodeling in the context of approximation by ridge functions, which has received less attention than BO in literature. We proposed an active sampling selection criterion meant to take advantage of the current knowledge of a low-dimensional FS obtained as a byproduct of the multi-fidelity surrogate model. However, the study showed this approach led to poorer predictive accuracies than the FS-based DOE.

Finally, in chapter 5, we leveraged the individual contributions made throughout the thesis to propose a coherent process for creating multi-fidelity surrogate models with high-dimensional inputs under a limited analysis budget. We laid out the steps of the proposed approach and showed that its outperforms the state-of-the-art deep multi-fidelity Gaussian process method, especially for input spaces with more than 50 dimensions, thus validating the approach.

5.2.2 Contributions

Multiple contributions were made in each of the three research areas of this thesis.

In section 2.2.2, we 1) proposed a fully Bayesian and gradient-free formulation for surrogate modeling of analyses with high-dimensional inputs, 2) presented a practical implementation of the approach, and 3) conducted a study of the impact of multiple parameters and test datasets on the performance of the proposed approach. In section 2.3, we 1) proposed an alternate approach to selecting the FS dimension suitable to out fully Bayesian approach, and 2) conducted a study to show the applicability of the proposed approach. In section 2.4, we 1) formulated alternate strategies for speeding up the training of the proposed model when a range of FS dimensions are being considered, and 2) conducted a study to evaluate their applicability.

In section 3.1, we 1) proposed a fully Bayesian and gradient-free formulation to multi-fidelity approximation by ridge functions, 2) presented the practical aspects necessary to the implementation and usage of the method, and 3) demonstrated its potential effectiveness on multiple multi-fidelity engineering datasets. In section 3.2, we 1) proposed alternate strategies for handling LF and HF FS in the context of multi-fidelity approximation by ridge functions, and 2) presented a study that contrasted these approaches on the accuracy of the resulting surrogate model.

In section 4.1, we 1) formulated an approach to DOE that leverages the knowledge of an uncertain FS obtained using LF observations to assist the selection of new HF observations, 2) detailed a practical implementation of the approach, and 3) conducted a study of the impact of multiple factors and datasets on its effectiveness. In section 4.2, we 1) formulated an approach to active sampling that leverages the knowledge of the uncertain FS obtained asa byproduct of the proposed multi-fidelity approach to approximation by ridge functions, and 2) conducted a study to assess its effectiveness in multiple conditions.

Additionally, for all aforementioned experiments, the code used to automate the experimental workflow has been made openly available. The first part of the research also led to

a journal publication at the *International Journal for Uncertainty Quantification* [118].

5.2.3 Summary of Findings

The findings summarized here are based on the experimental results obtained throughout this thesis. As is common in machine learning, it is important to acknowledge that the performance of the proposed approach is dataset-dependent. This motivated the use of multiple datasets in experiments, that are representative of various types of analyses, to try to develop an understanding of the sensitivity of the proposed approach to different conditions. While this diversity of datasets allows to start generalizing our findings to a small extent, these findings are still very much specific to these datasets, and hence the proposed methods should be assessed when applied to new classes of problems.

In the single-fidelity context, we found that combining a fully Bayesian approach with a surrogate model based on approximation by ridge function allowed to systematically obtain a better quantification of predictive uncertainty. In turn, we confirmed that quantifying predictive uncertainty allowed to avoid over-fitting when training data is sparse, which eventually leads to surrogate models with higher predictive accuracy. As expected from a Bayesian approach, we found that the benefits of the better uncertainty quantification fade as the size of the training set grows, and after a point that is dataset-dependent, an MLE-based approach would be sufficient. As was expected, the benefits of reducing the dimension of the input space grow with the original input space dimension: for the lower-dimensional cases, a Bayesian GP without dimension reduction is usually sufficient.

When it comes to the problem of selecting the feature space dimension, we found that the decay of the noise variance could be used as a selection mechanism under the conditions that 1) the training set is sufficiently large (3 to 5 times the number of inputs) and 2) the feature space dimension remains low (up to 5 dimensions). We also found that, even though some datasets have intrinsic feature spaces with a fixed dimension, using a lower-dimensional feature space when only very few observations are available may lead to a

better out-of-sample predictive accuracy. This is consistent with the general guideline to balance model flexibility, or complexity, with the available amount of training data in order to avoid over-fitting. The approach proposed to speed up the successive training of models with different feature space dimensions by using a sequential approach in which previous directions are reused was found to yield the expected benefits – lower cumulative training time while leading to the same feature space dimension – under two conditions: 1) larger training sets (3 to 5 times the number of input dimensions) and 2) higher-dimensional inputs (50 to 100). This corresponds to situations in which the fully Bayesian approach becomes particularly computationally expensive and slow, in which case the sequential approach allows to speed it up.

In the multi-fidelity context, we found that several conditions must be met in order for the proposed approach to lead to a better predictive accuracy than the single-fidelity approach. First, benefits are only observed for intermediately sized datasets. Although the exact number is dataset-dependent, based on the results obtained with the studied datasets, the number of equivalent HF observations should be between 0.5 to 1.5 times the number of inputs. Smaller training sets lead to poor predictive accuracies, while larger training sets lead to sufficiently good predictive accuracy with a single-fidelity model. Second, the LF and HF analyses should have some degree of similarity, otherwise the proposed approach fails at learning useful features from the LF analysis, and spending the whole budget on HF observations leads to better results. While determining the optimal ratio of LF to HF observations was not part of the experimental goals, we found that balanced MF training sets were also a condition for achieving better predictive accuracy than a single-fidelity model. In practice, for the datasets under study, this corresponds to a range of 30% to 60% of the budget allocated to LF observations. When attempting to use different feature spaces for each layer of the MF model, we confirmed that sharing the same feature space is crucial to a MF approach to approximation by ridge functions under a restrictive analysis budget (less than 1.5 times the number of inputs).

Regarding the selection of HF observations by leveraging the knowledge of the LF feature space, we found that the proposed DOE approach, which mainly aims at spreading observations over the feature space – or rather over possible feature spaces – systematically leads to better predictive accuracy than performing a space-filling DOE in the original high-dimensional input space. This finding is in line with expectations: while the curse of dimensionality makes it impossible to properly fill a high-dimensional input space with training samples, focusing on a low-dimensional subspace that is especially tailored to the input-output mapping under consideration allows to alleviate it. However, we also found that using a maximum-variance adaptive sampling criterion instead of the proposed DOE reduced the predictive accuracy of the surrogate models. This finding suggests that the local variance criterion may not be adapted in this context and motivates the future investigation of future criteria.

Finally, we found that the combination of the different methods proposed throughout this thesis into the overall methodology presented earlier in this chapter may allow to create surrogate models with higher predictive accuracy than the state-of-the-art deep multi-fidelity GP when inputs are high-dimensional and analysis budget is limited, under the condition that the correct FS dimension is selected. In fact, the proposed approach leads to well-fitting surrogate models with analyses of up 50- to 100-dimensional inputs while the deep MF GP fails at creating meaningful surrogate models for such high-dimensional inputs. In cases where an unadapted FS dimension is selected, the deep MF GP may yield better results. This corresponds to situations in which too much of the output variance was discarded (if too few dimensions are retained), or prone to over-fitting (if too many dimensions are retained).

5.2.4 Future Work

The research presented in this thesis may be extended in two main directions: the improvement of the proposed methods and their application to additional challenging engineering

applications.

As the dimensions of the analysis inputs and feature space increase, so does the size of the projection matrix onto the FS, and the number of parameters whose distributions are inferred when carrying out MCMC. When walking through the application of the single-fidelity method in section 2.2.4, we pointed out the decreasing quality of the MCMC chains due to the increase in number of model parameters. A solution is to continue drawing more samples, which was excessively expensive in the context of the experiments carried out in this thesis. Exploring options for reducing the computational cost of the MCMC sampling process, thus allowing the practical sampling of longer MCMC chains, should be the focus of future work, in order to ensure that the distributions obtained are indeed representative of the actual posterior distributions of the model parameters. This could be for example achieved by selecting an alternate type of surrogate model for the low-dimensional regression, for which the computation of the likelihood is not as costly as it is for a GP. Approximate approaches to Bayesian inference, such as variational inference, may be investigated as an alternative way of training the proposed models instead of MCMC. In addition, reducing the computational cost of the proposed method, which is justified when the predictive model is meant to replace particularly costly analyses, would widen its applicability.

Further research could be carried out on the impact of the specific covariance kernels and prior distributions employed in the proposed models. While such a study lied outside the scope of this work as the sensitivity to other parameters was already being considered, we expect these to have an impact on predictive accuracy.

While a GP regression model was used to model the link function throughout this work, the key contribution that consisted in leveraging parametrization techniques to enable the Bayesian inference of the projection matrix using off-the-shelf MCMC samplers is not specific to GPs. As such, it could be generalized to enable a fully Bayesian treatment of approximation by ridge functions using other types of predictive models to model the link function, such as Bayesian neural networks [48]. In certain circumstances, these might

bring advantages over GPR, such as reducing the training time.

The proposed multi-fidelity approach was introduced assuming two levels of fidelity. To handle situations where more levels of fidelity are available, a generalized formulation of the proposed multi-fidelity approach could be proposed and its performance investigated.

While we have observed that the degree of similarity between the LF and HF analyses has an impact on the benefits of the proposed MF approach, an exact quantification of the required degree of similarity was out-of-scope of the present work. Such a quantification is challenging because the degree of similarity cannot simply be reduced to traditional metrics used for assessing predictive accuracy, such as R^2. For example, a linear relationship may exist between the responses of the LF and HF analyses, in other words they may be correlated. Despite the fact that such a relationship may lead to a poor R^2 value, we expect the MF method to perform well in such a situation, as illustrated in the results of experiment 2.1. Because we are using a deep multi-fidelity Gaussian for to model the link function, we also expect non-linear – but somewhat predictable – relationships between the LF and HF analyses to be sufficient to benefit from the MF approach. An approach for quantifying the degree of similarity between the LF and HF analyses required to apply the proposed methods should be able to account for these situations.

Moreover, the determination of the optimal ratio of LF to HF analysis observations was out-of-scope of this research and remains an open problem. An investigation enabling the *a priori* determination of the ratio in which the different levels of fidelity should be evaluated would be of great practical interest for practitioners willing to apply the proposed MF method.

In order to extend the proposed approach to situations where the LF and HF models have different inputs, an approach could be investigated, in which observations of different fidelities are projected onto a common feature space using distinct projection matrices. This approach would differ from the "different" approach studied in section 3.2, where different FS were considered for the LF and HF *layers* of the model. Instead of the projection being

tied to layer of the model, the projection applied would depend on the input space of origin of the observation (during training) or prediction (when using the trained model).

When it comes to adaptive sampling, other approaches could be considered, as the maximum variance criterion investigated in section 4.2 did not yield any improvement over the proposed DOE approach leveraging the knowledge of the feature space studied in section 4.1. In particular, adaptive sampling approaches that account for the reduction in the predictive variance over the complete input space, such as the IMSE criterion [178], or information-theoretic approaches, such as using mutual information [181], may prove more effective than the maximum variance criterion that only aims at reducing variance locally. Other adaptive sampling objectives may also be considered or used in complement to accuracy-based goals, such as the information gain in the feature space, similarly to [134].

In this work, we have mostly focused on estimating and comparing the predictive accuracy of the proposed approaches since it allows an application-independent assessment. However, it is the enabling nature of surrogate modeling that motivated this thesis, and the application of the proposed single- or multi-fidelity methods to assist design tasks such as uncertainty quantification, design space exploration, or engineering optimization in the presence of high-dimensional parameters would allow to further demonstrate their practical utility.

Appendices

APPENDIX A

ADDITIONAL RESULTS FOR THE NOISE VARIANCE STUDY

This appendix contains all the noise variance decay plots that were not shown in section 2.3 for brevity of the main text. These results are discussed in the corresponding subsection of section 2.3.4 (“Analytical Functions”, “Active Subspace Datasets”, and “Aerodynamic Datasets”).

A.1 Analytical Functions

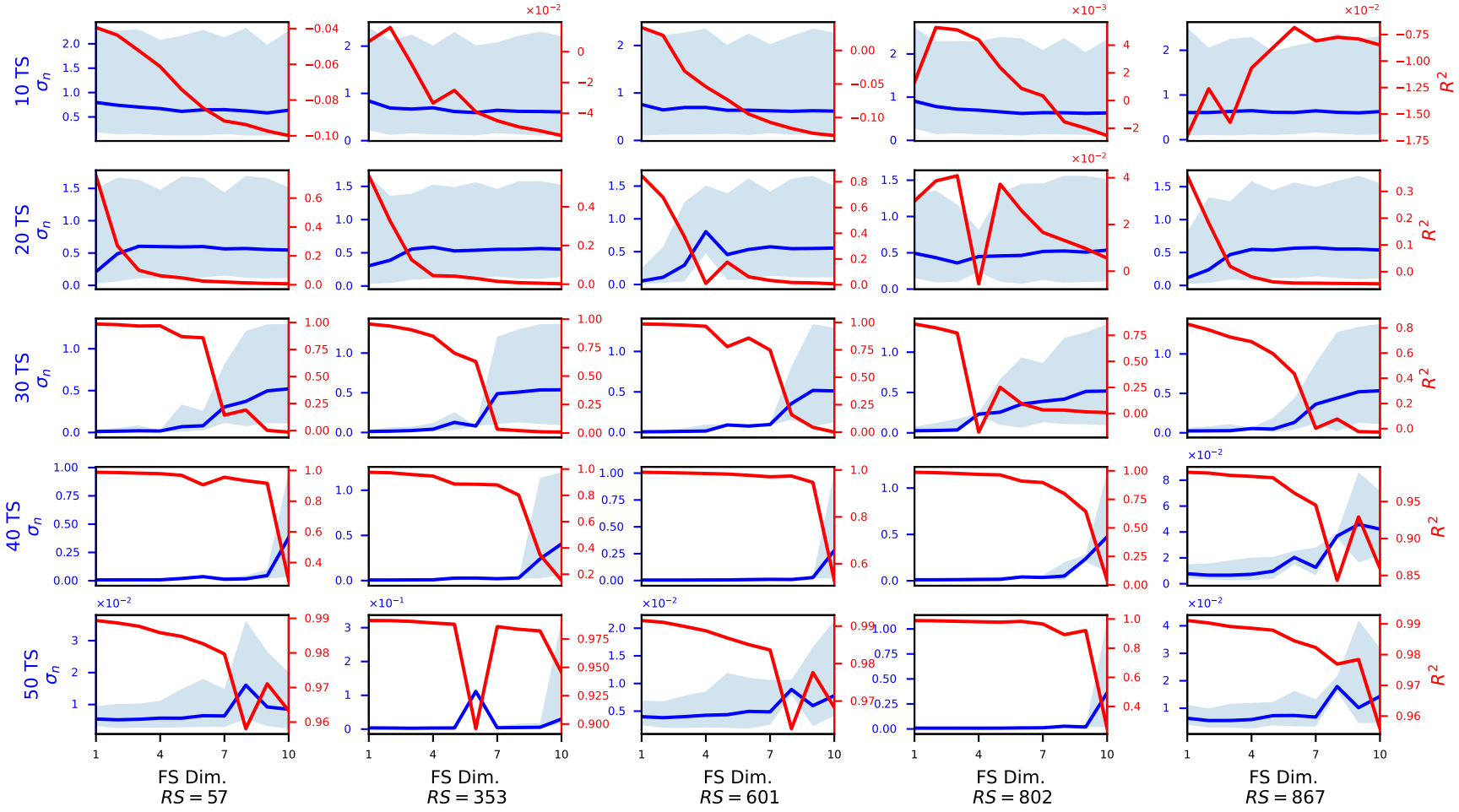


Figure A.1: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 10 input variables and a 1D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

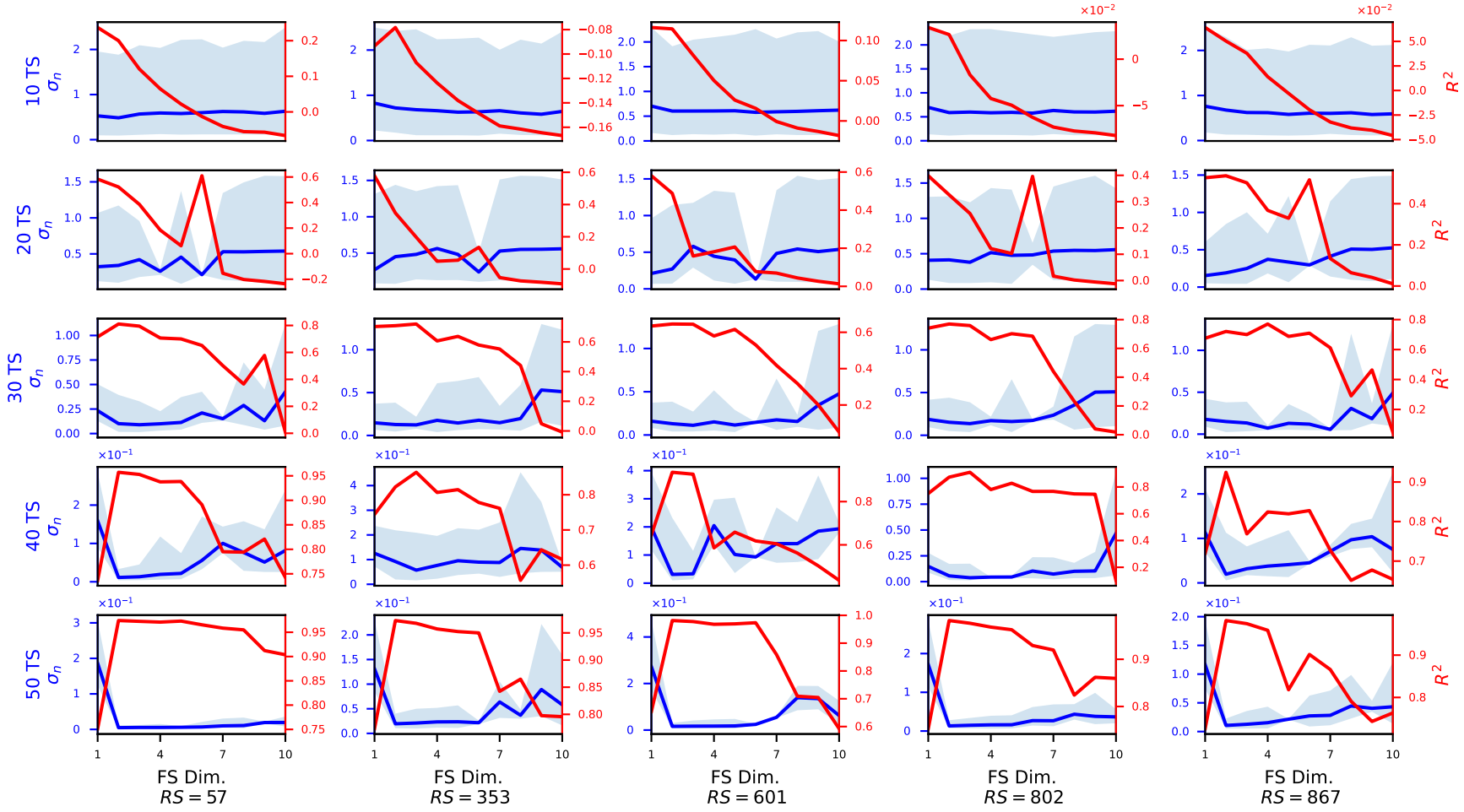


Figure A.2: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 10 input variables and a 2D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

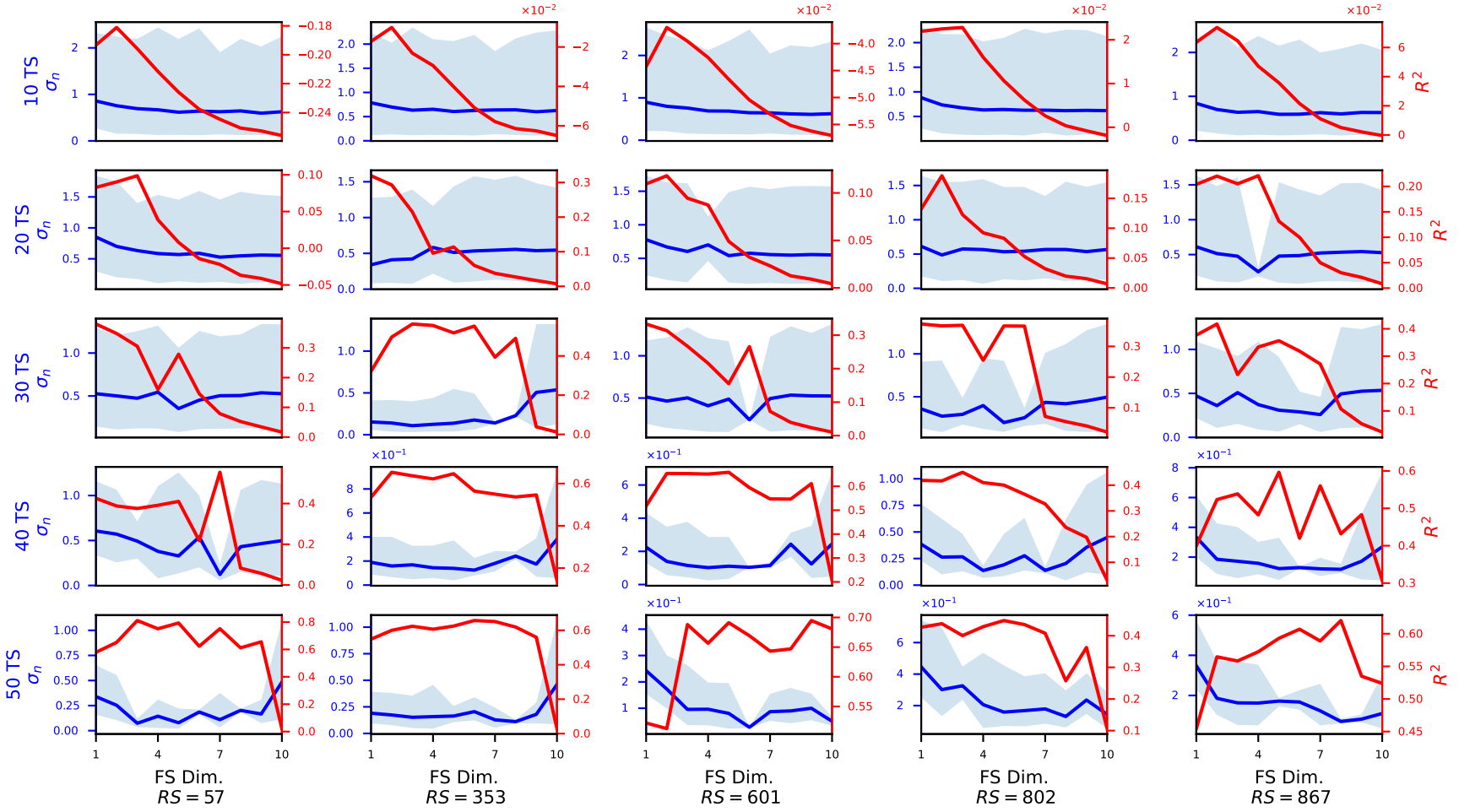


Figure A.3: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 10 input variables and a 5D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

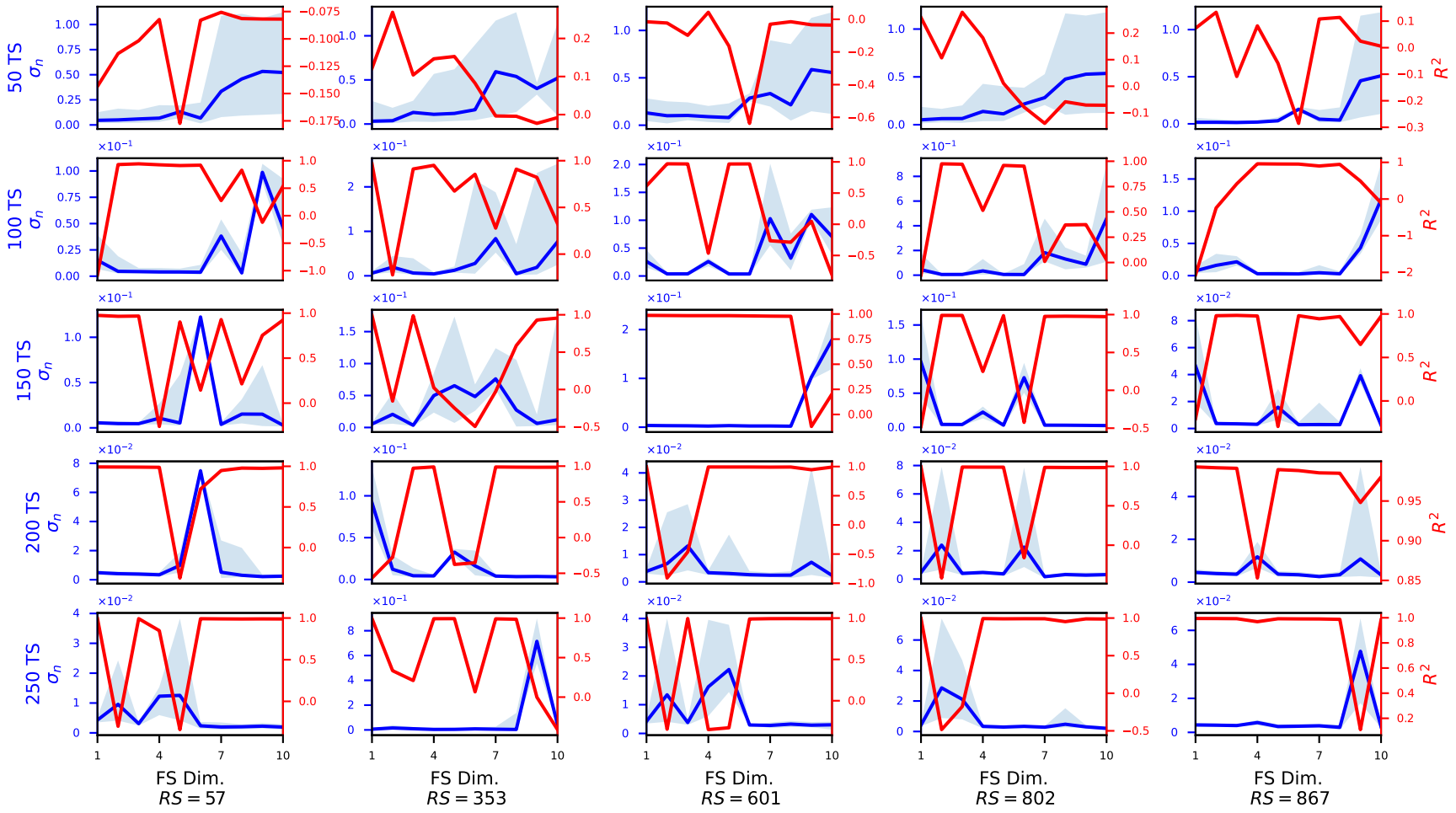


Figure A.4: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 50 input variables and a 1D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

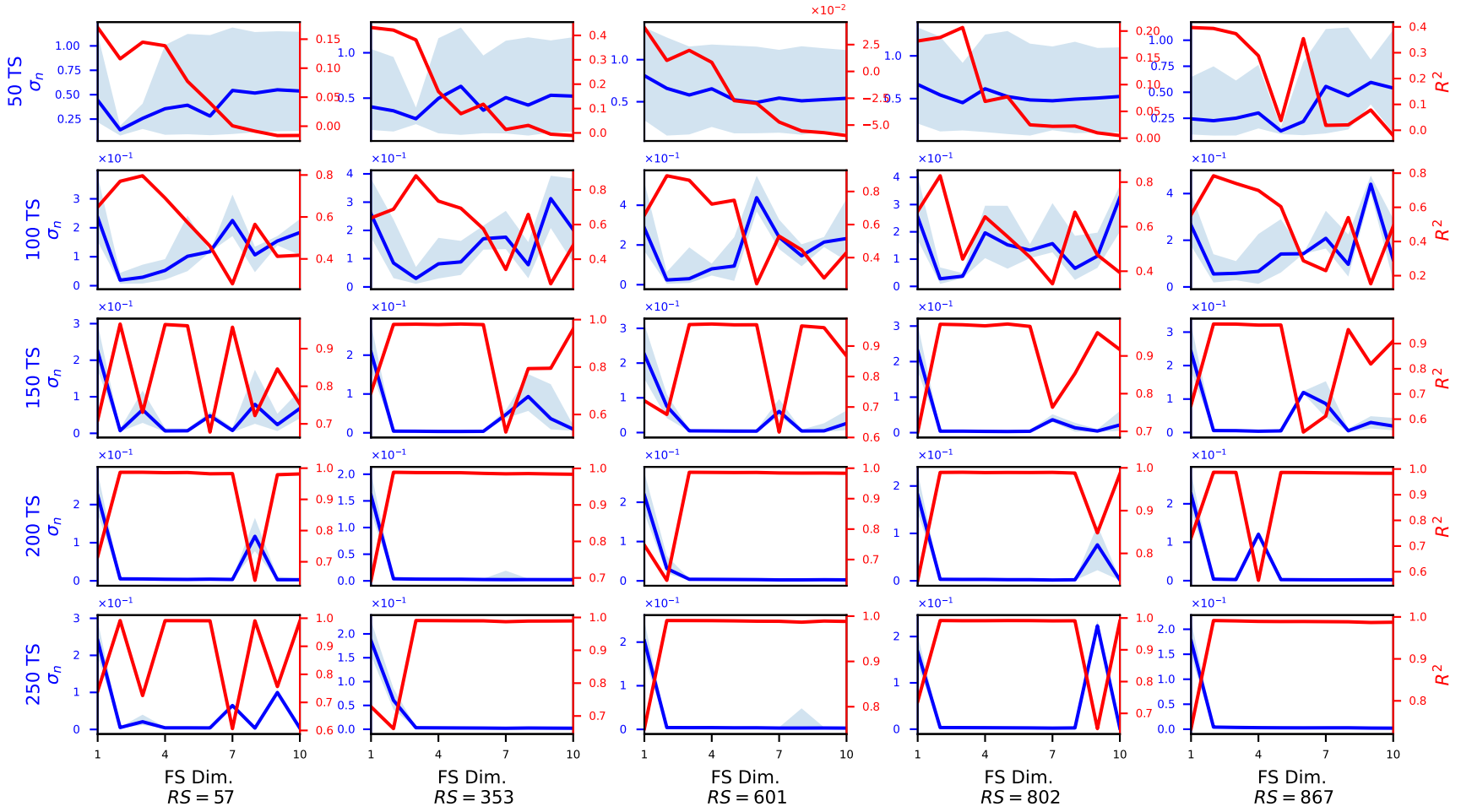


Figure A.5: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 50 input variables and a 2D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

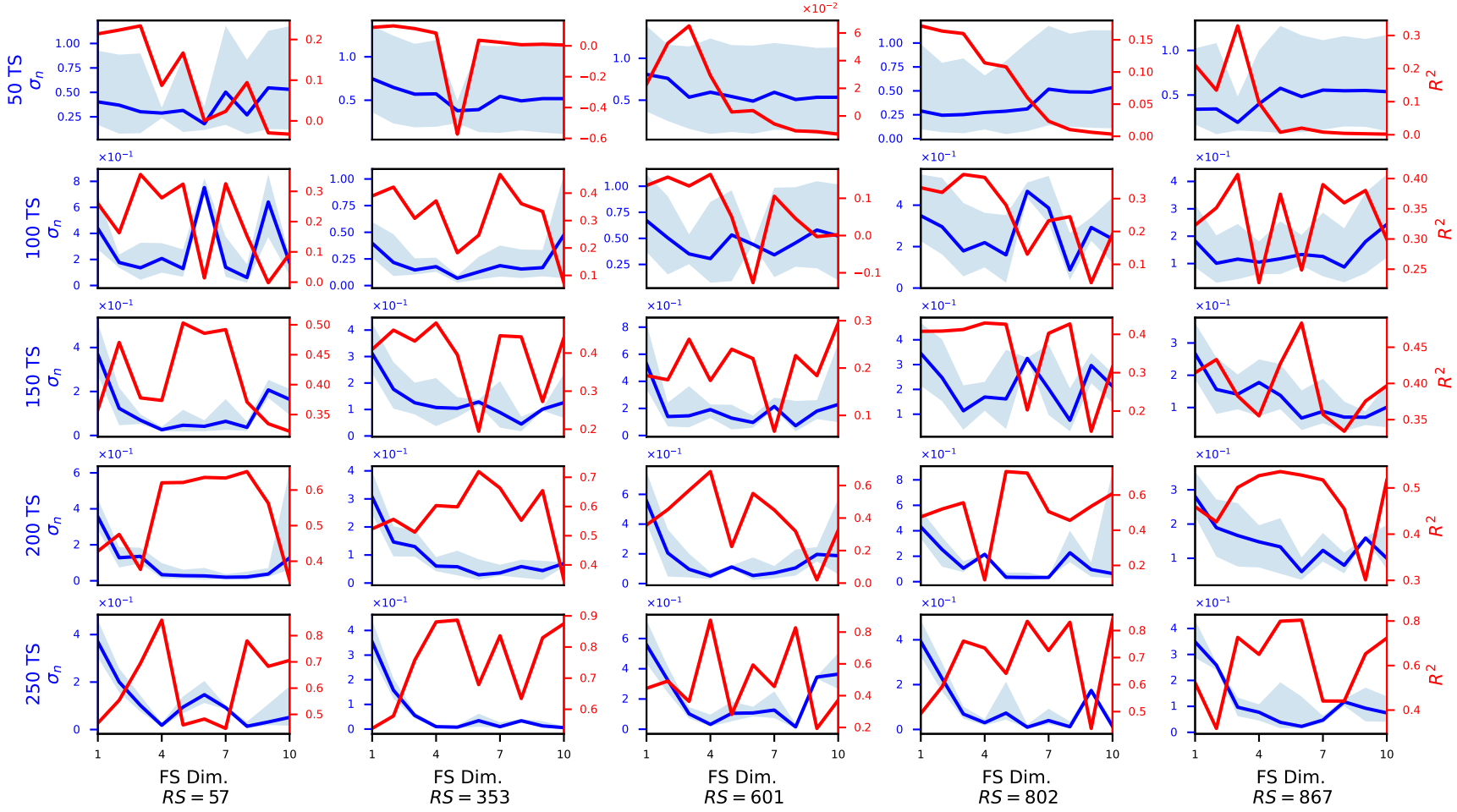


Figure A.6: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 50 input variables and a 5D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

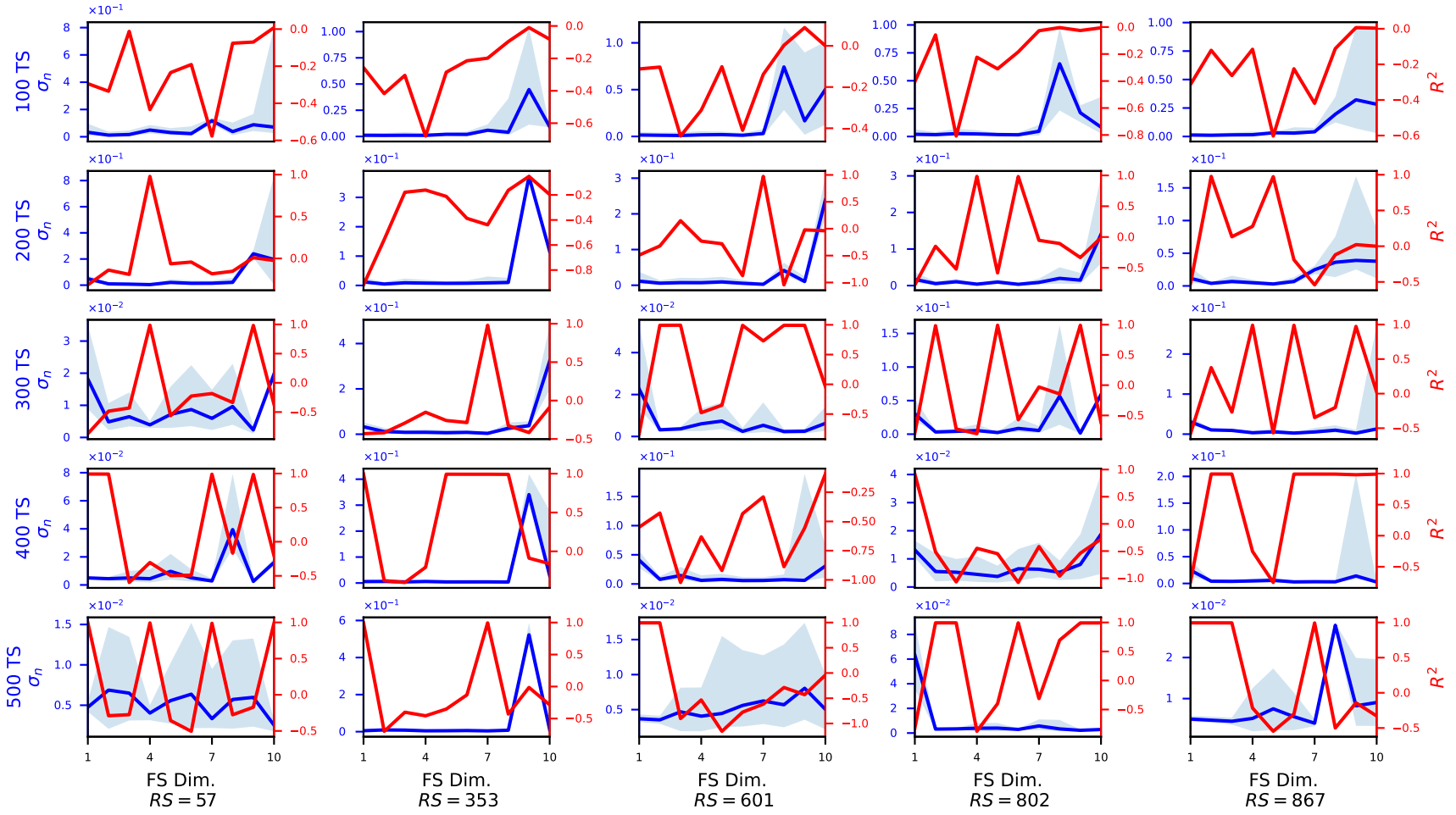


Figure A.7: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 100 input variables and a 1D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

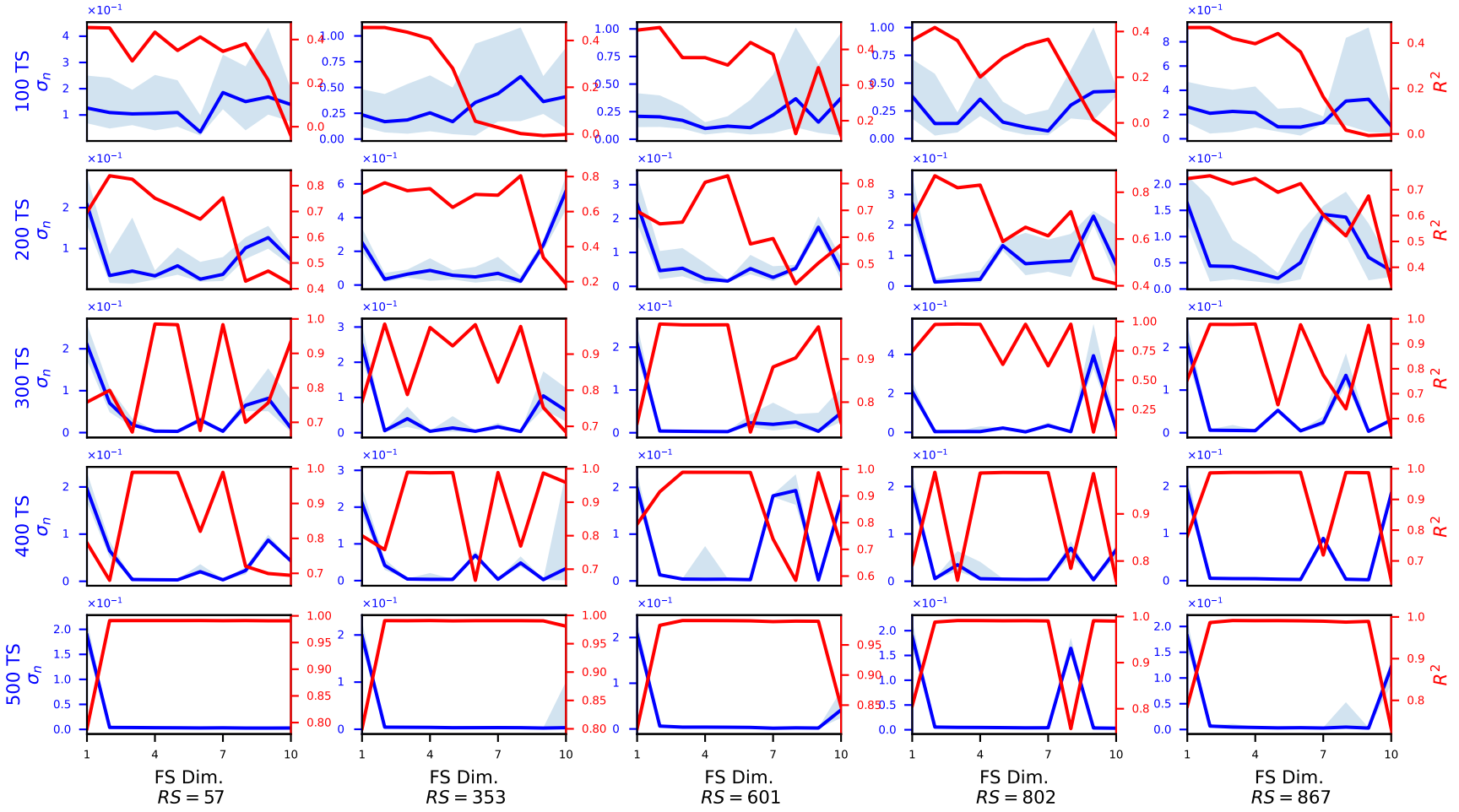


Figure A.8: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 100 input variables and a 2D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

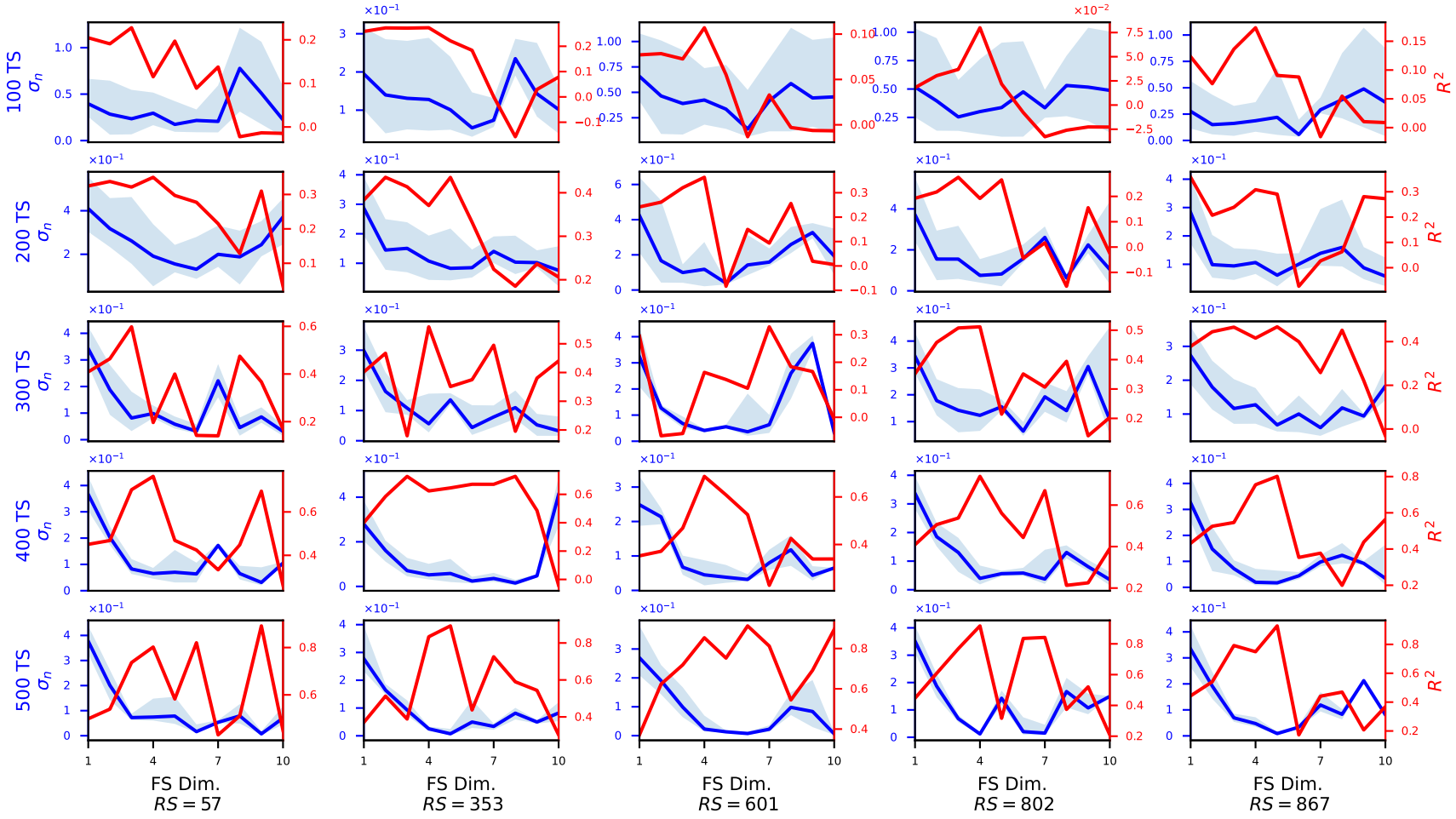


Figure A.9: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 100 input variables and a 5D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

A.2 Active Subspace Datasets

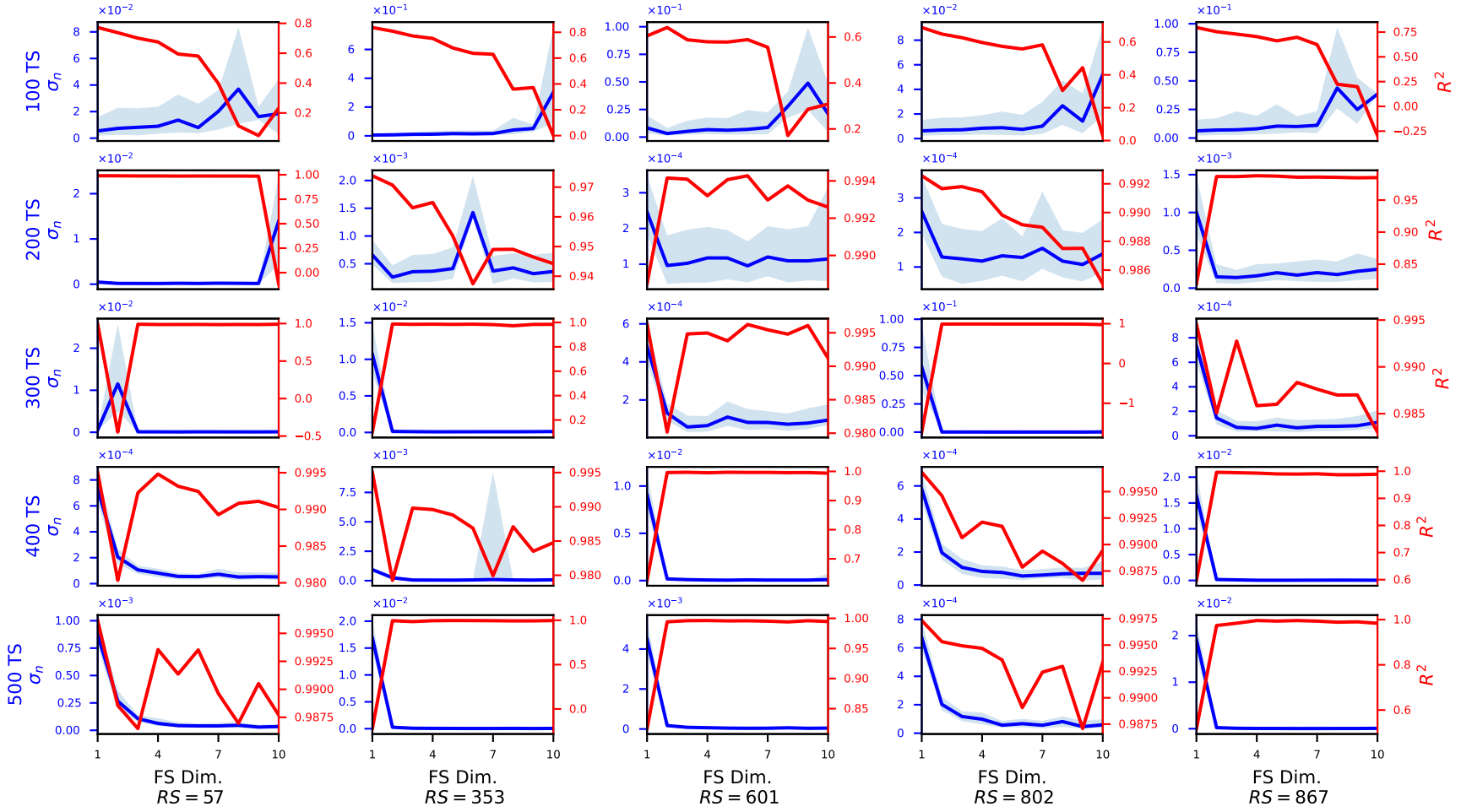


Figure A.10: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **Elliptic PDE (y_{long}) dataset with 100 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

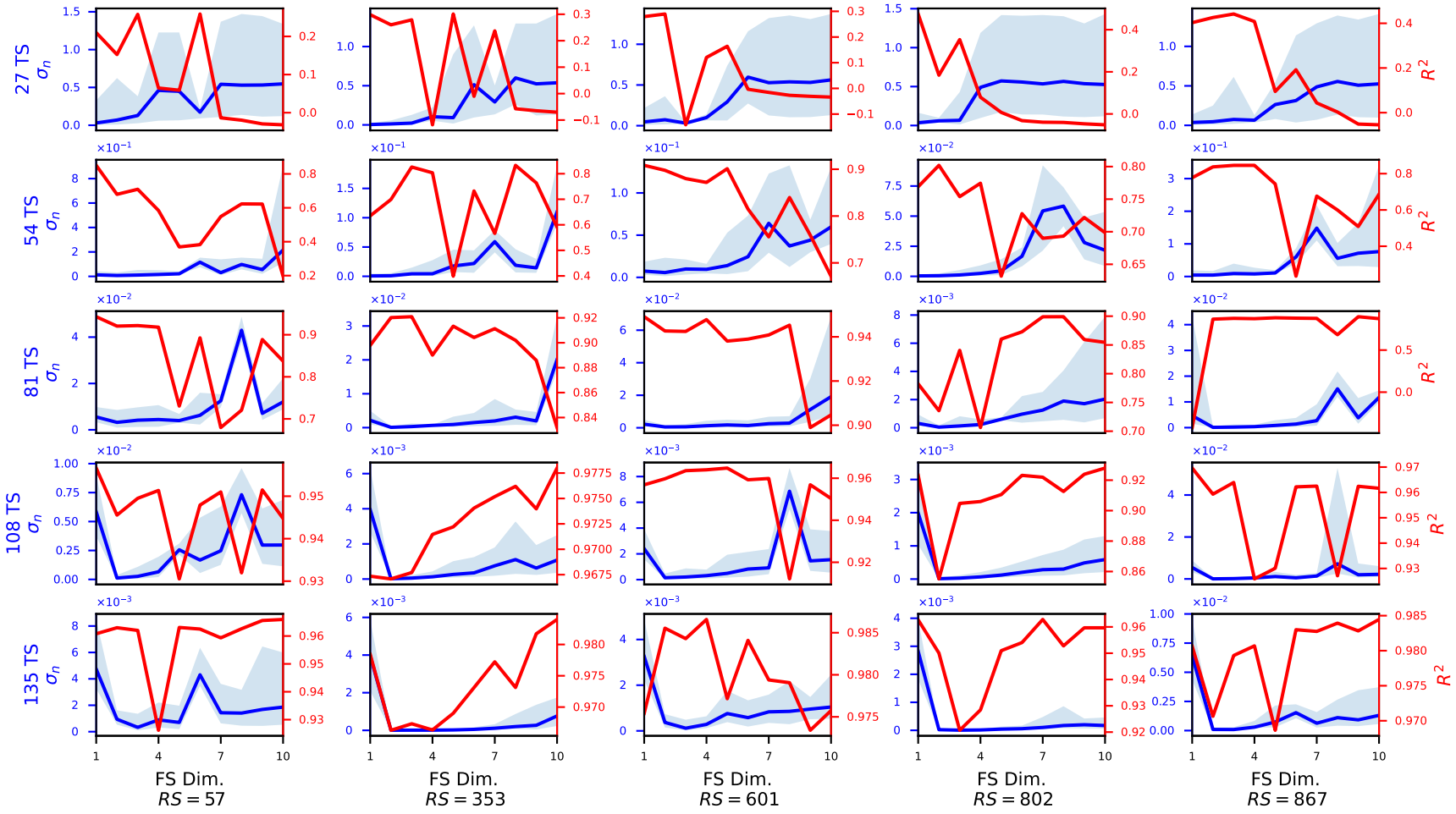


Figure A.11: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **HIV (y_{3400}) dataset with 27 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

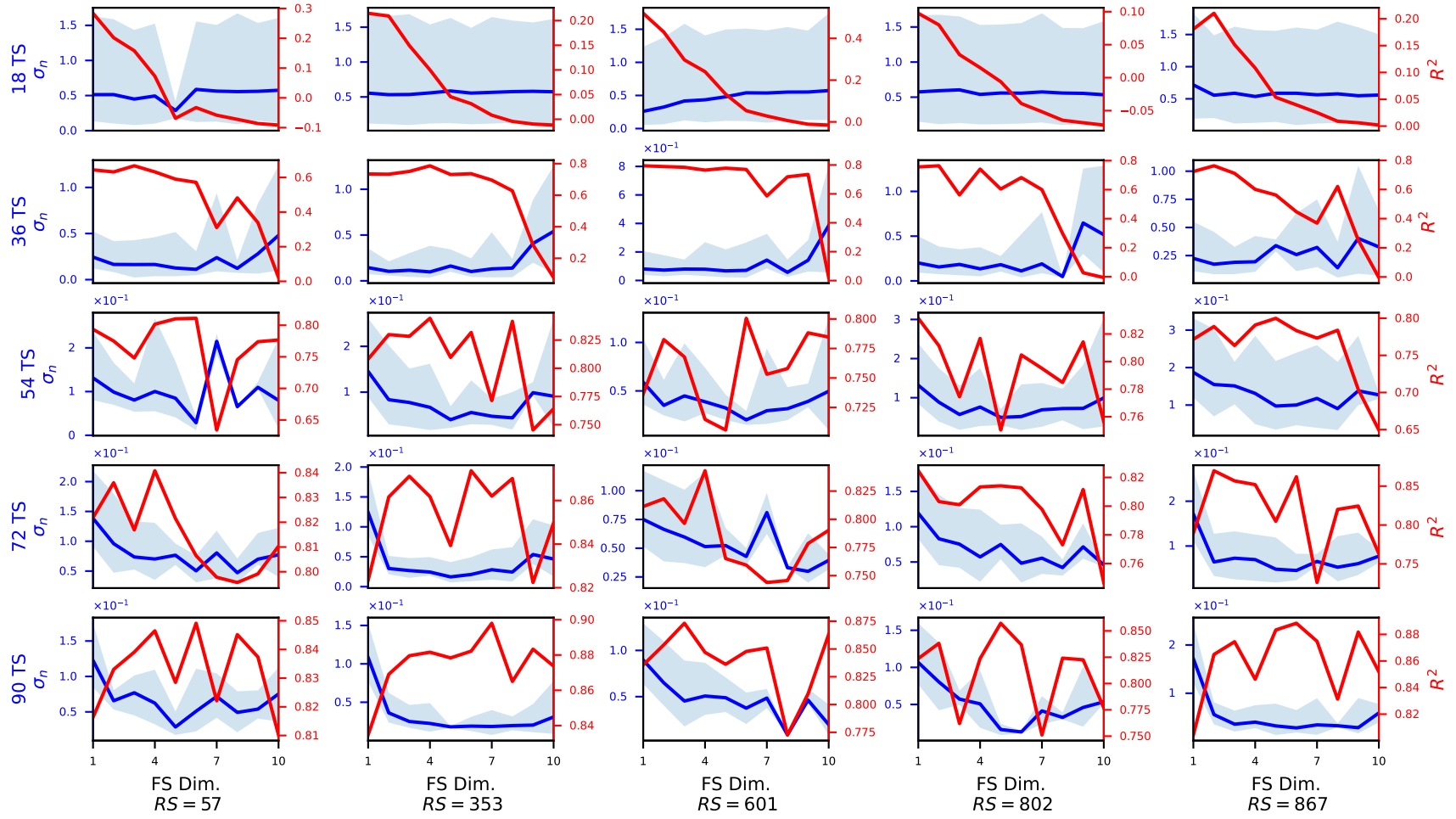


Figure A.12: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **NACA0012 drag dataset with 18 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

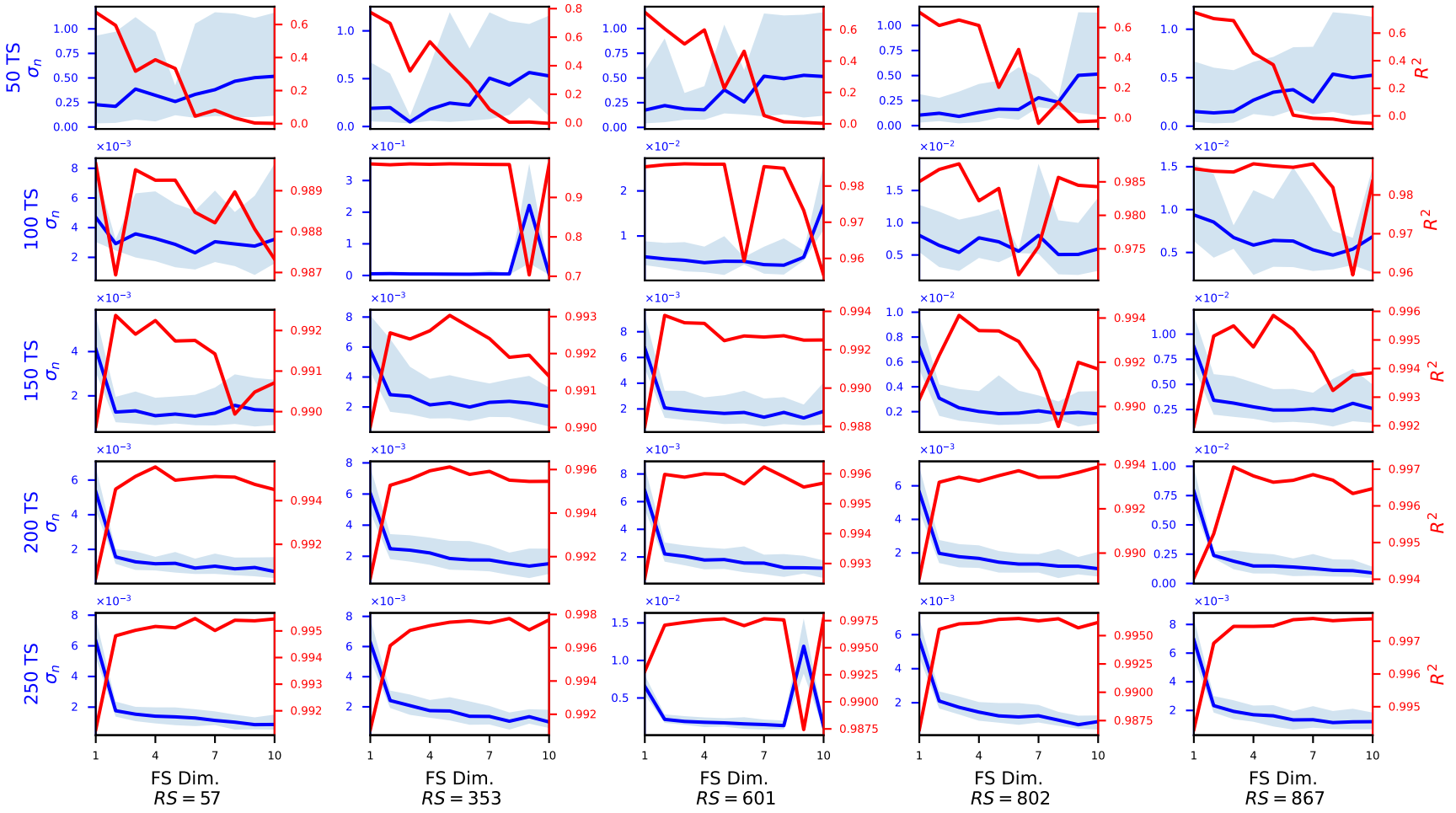


Figure A.13: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **ONERA M6 lift dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

A.3 Aerodynamic Datasets

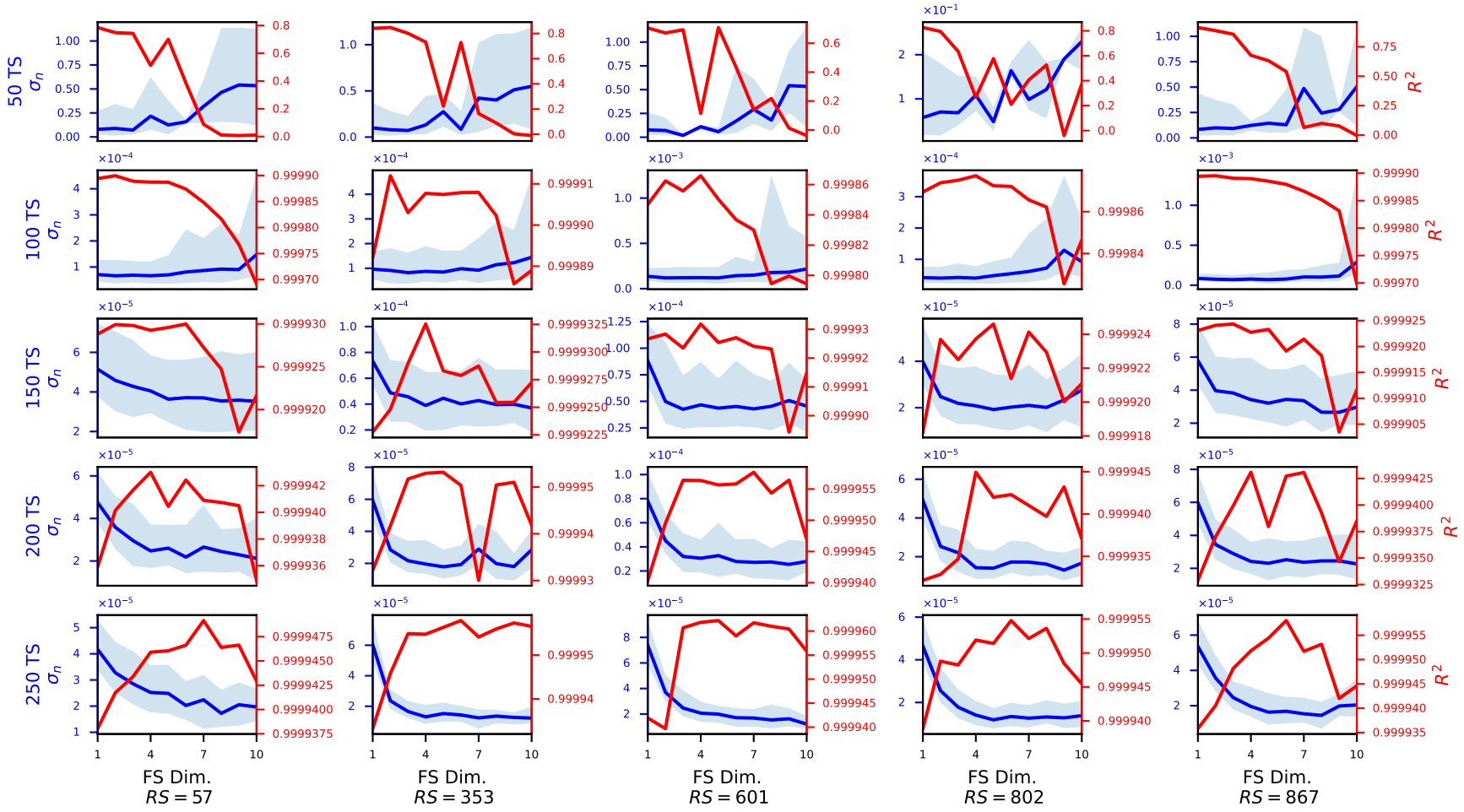


Figure A.14: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **subsonic CRM lift dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

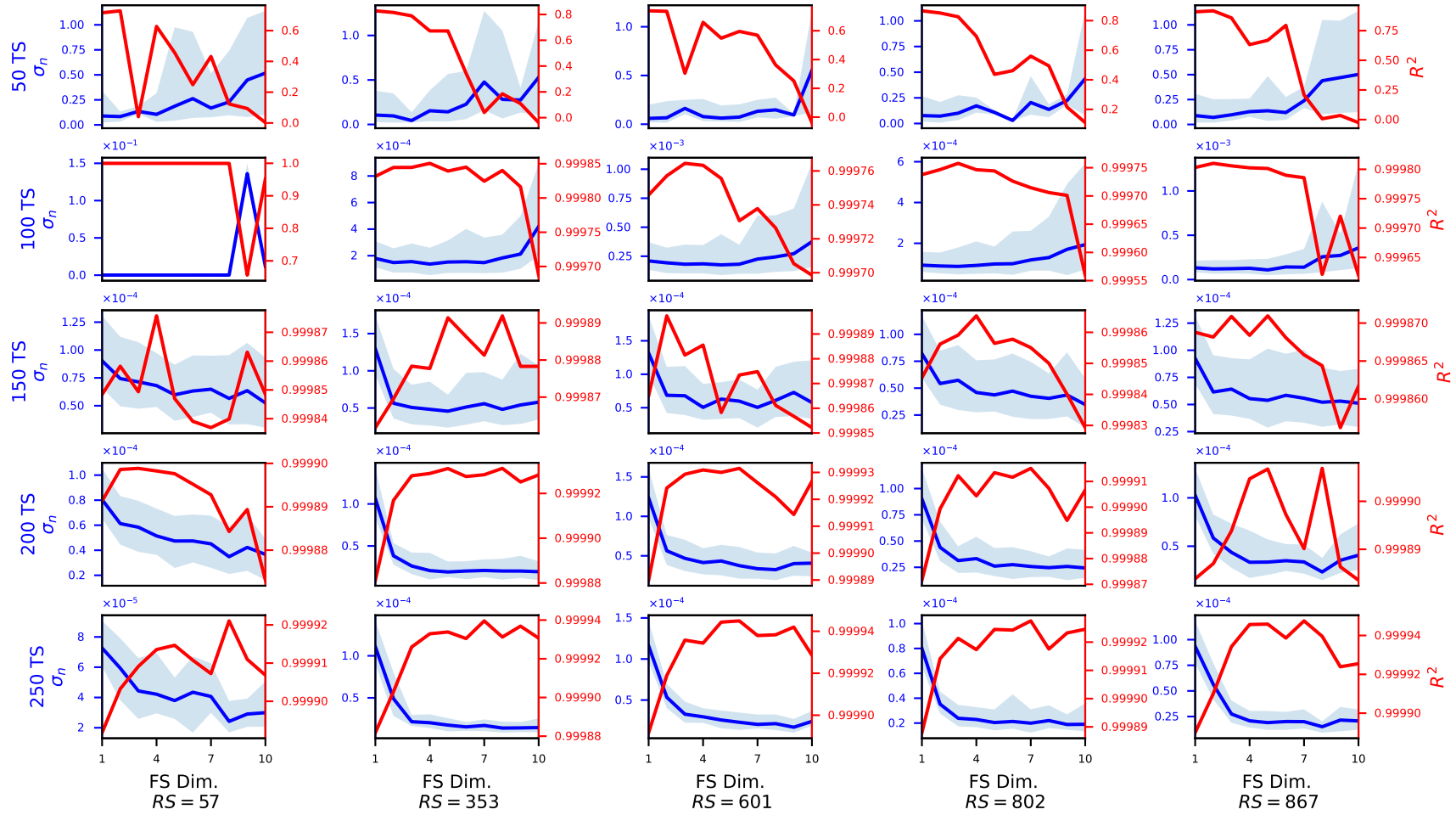


Figure A.15: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **subsonic CRM x-moment dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

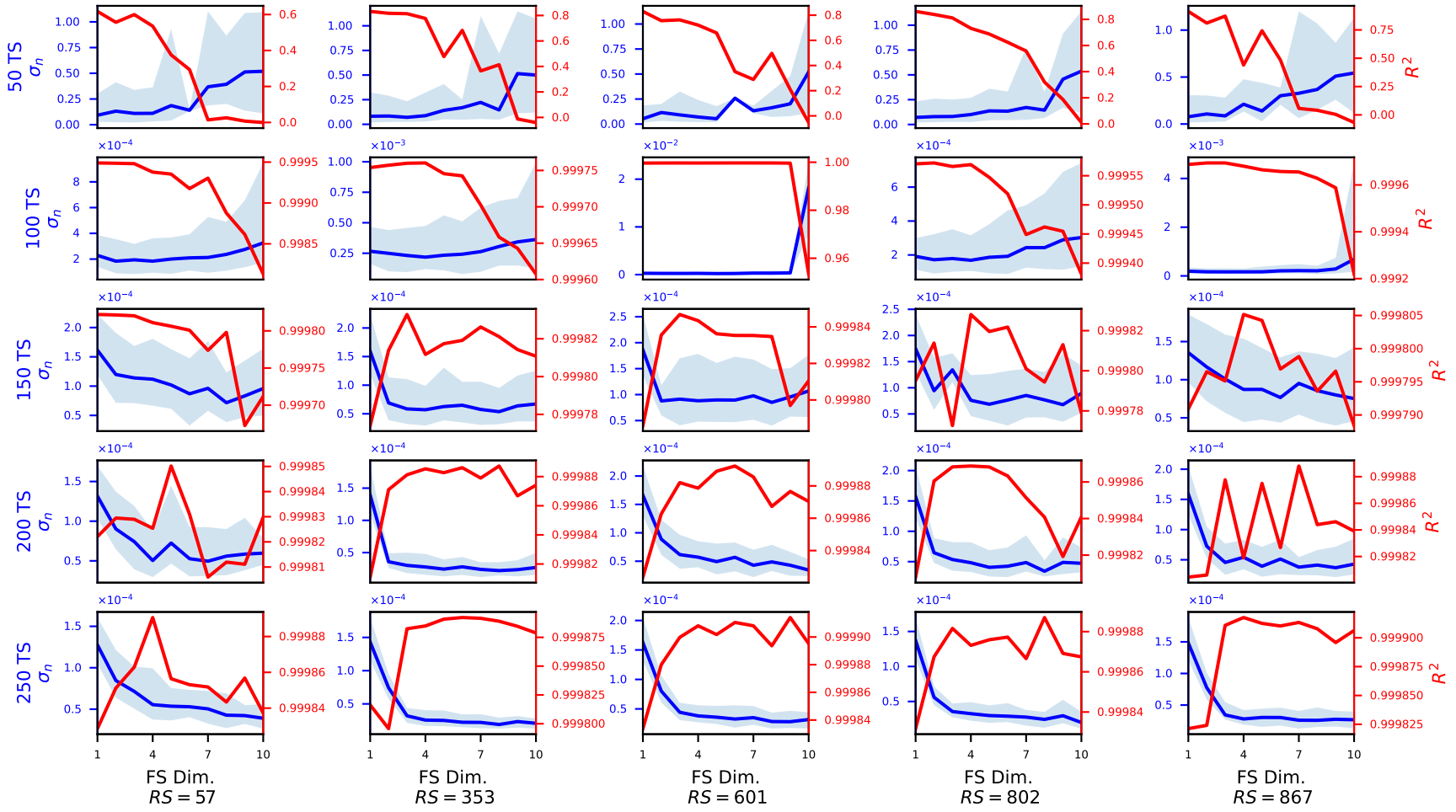


Figure A.16: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **subsonic CRM y-moment dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

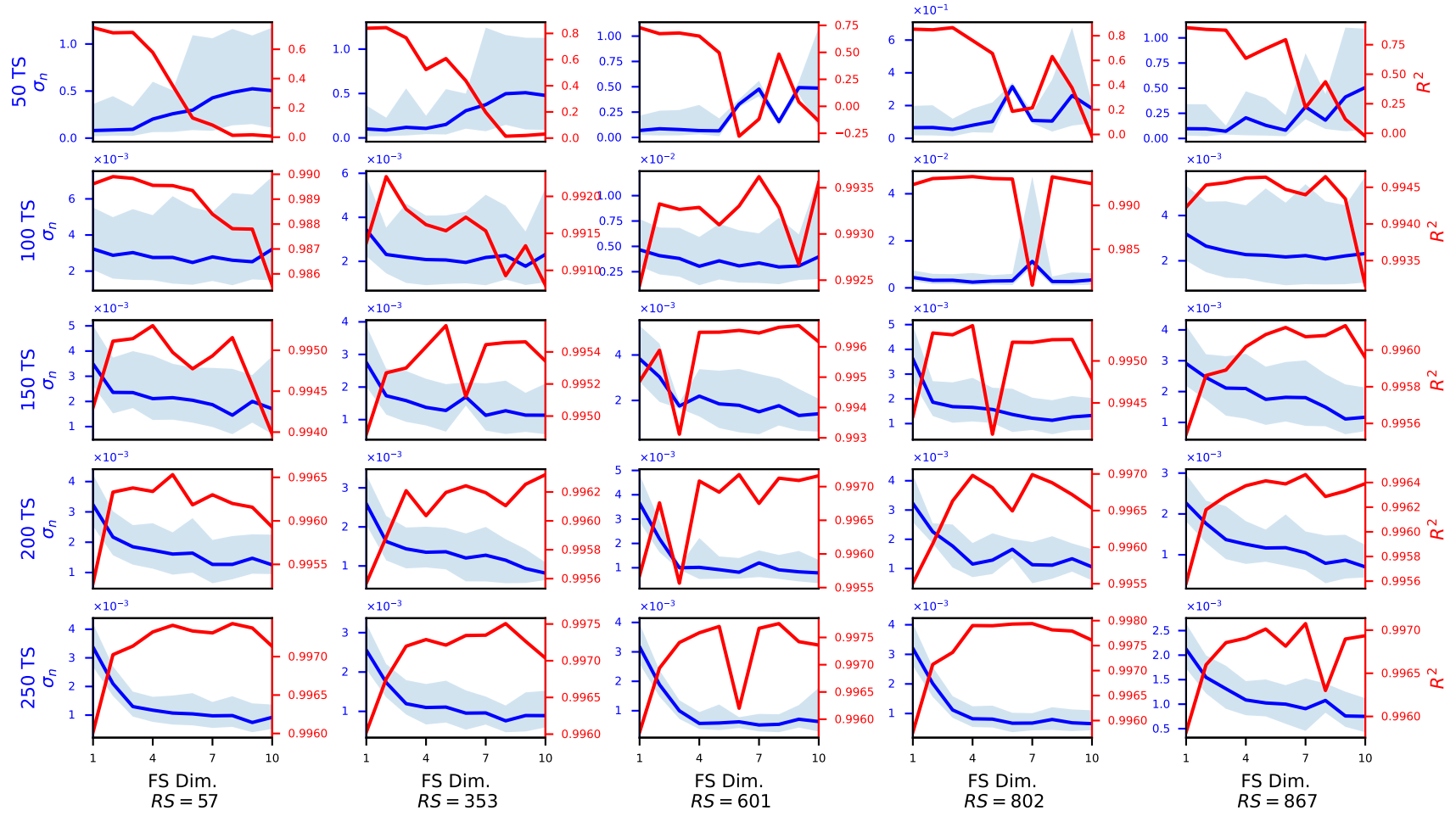


Figure A.17: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **subsonic CRM sideforce dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

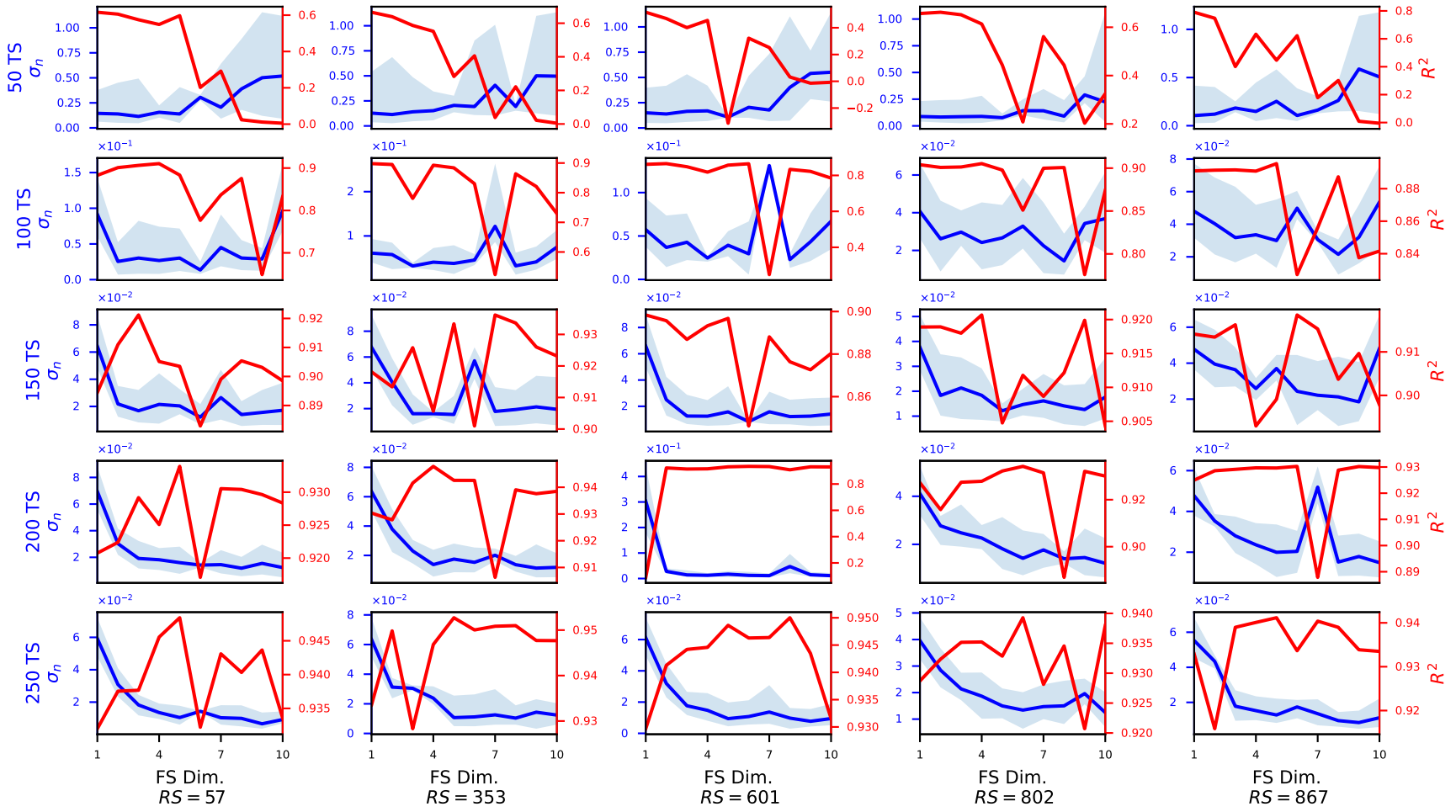


Figure A.18: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM drag dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

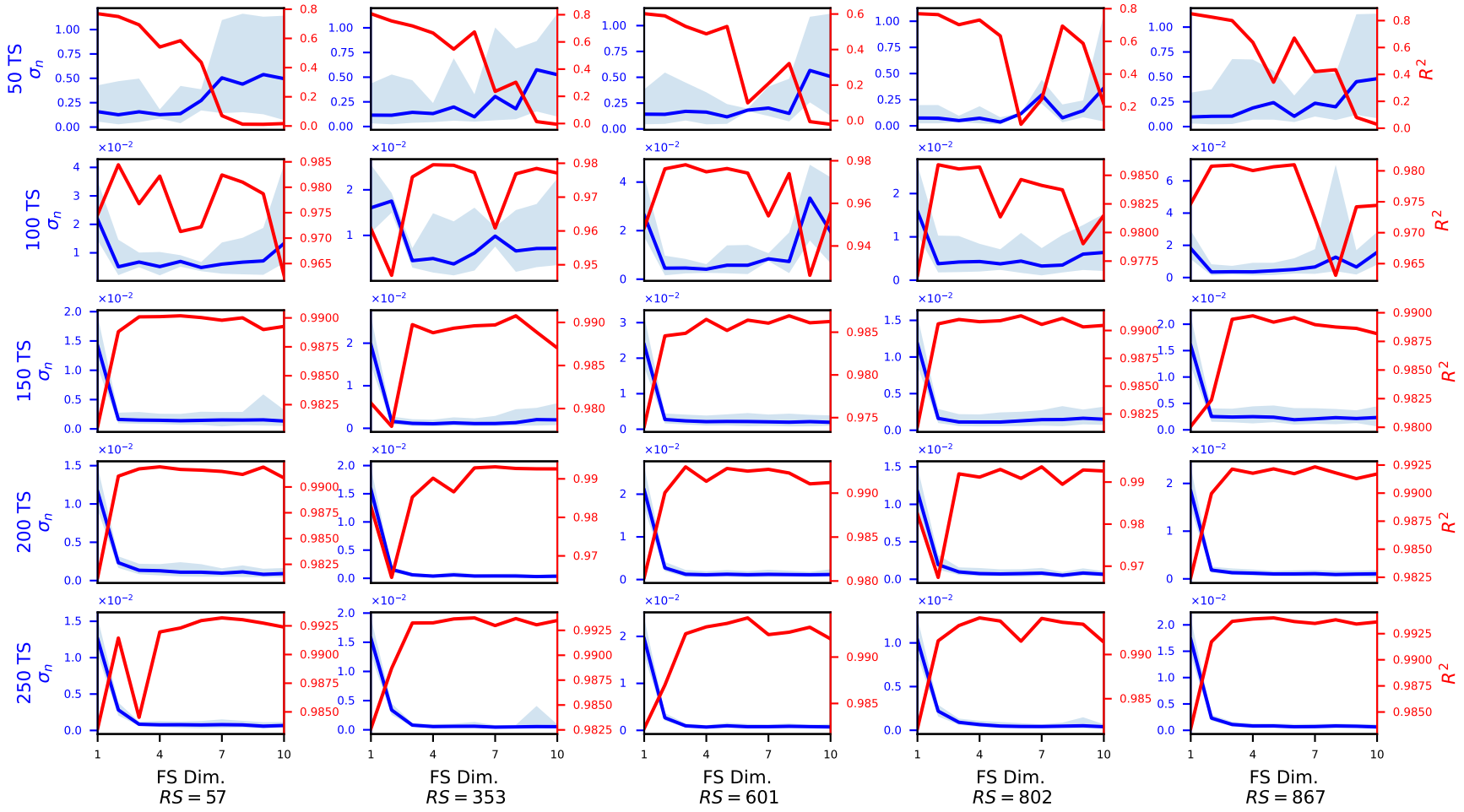


Figure A.19: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM lift dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

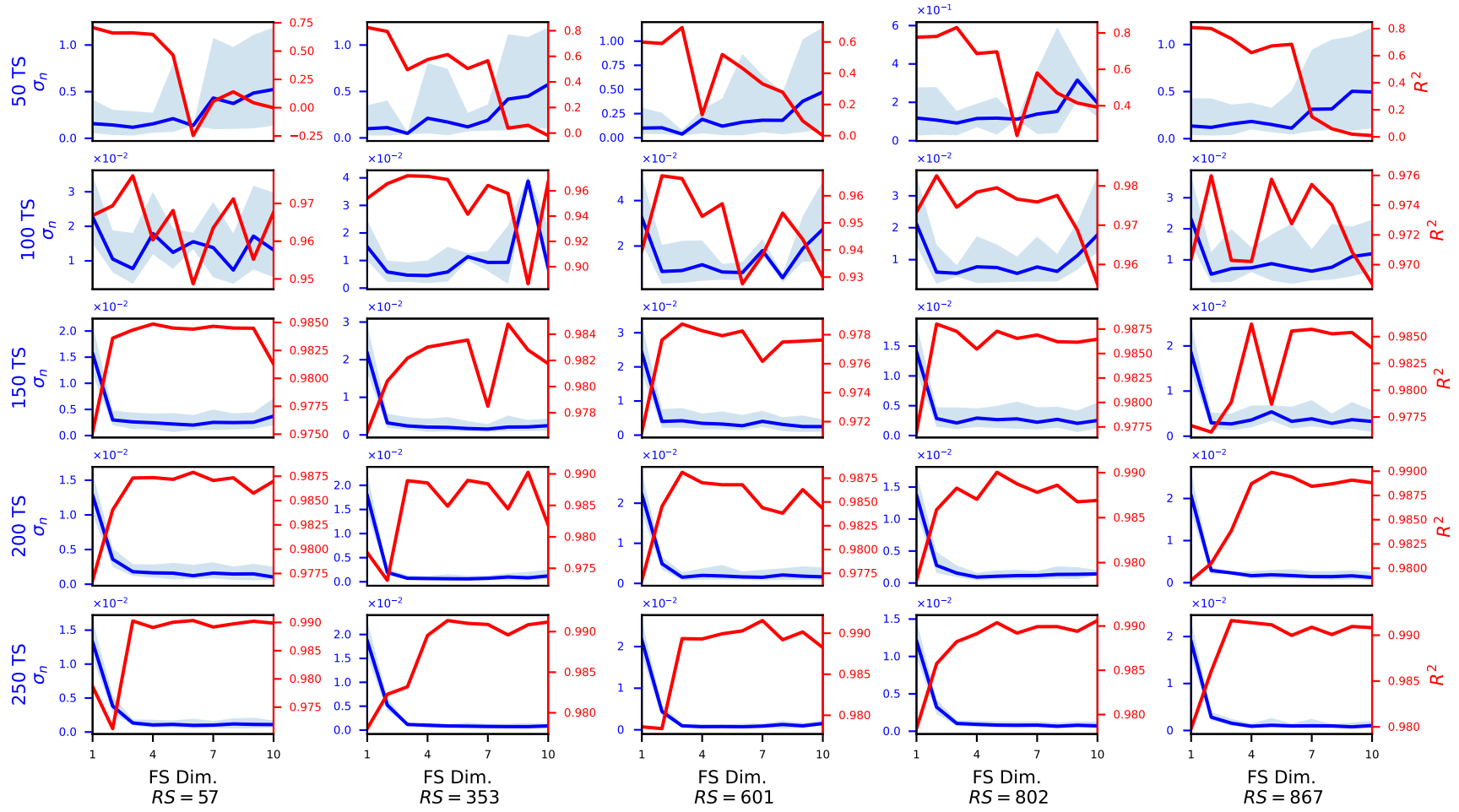


Figure A.20: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM x-moment dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

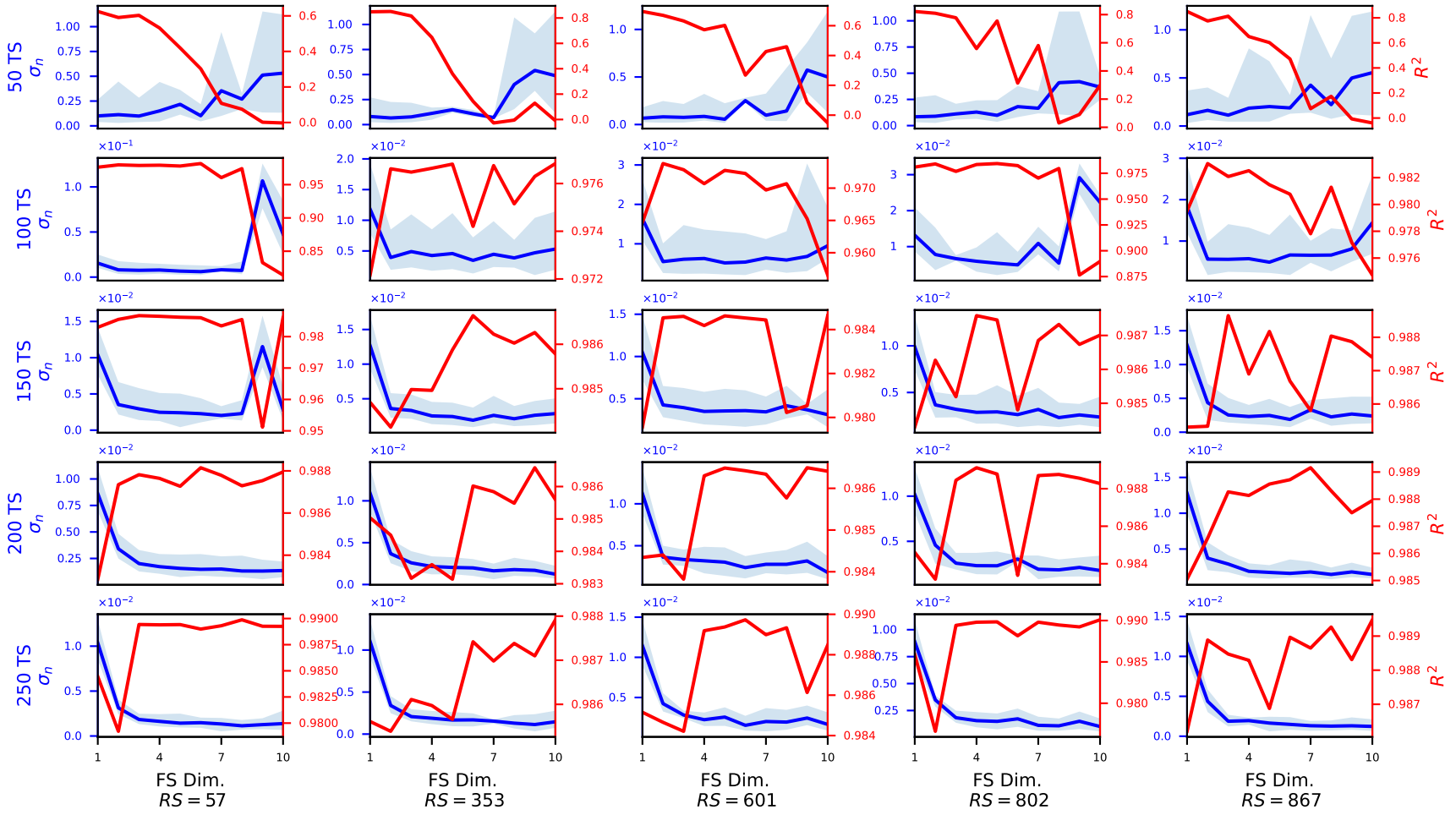


Figure A.21: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM y-moment dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

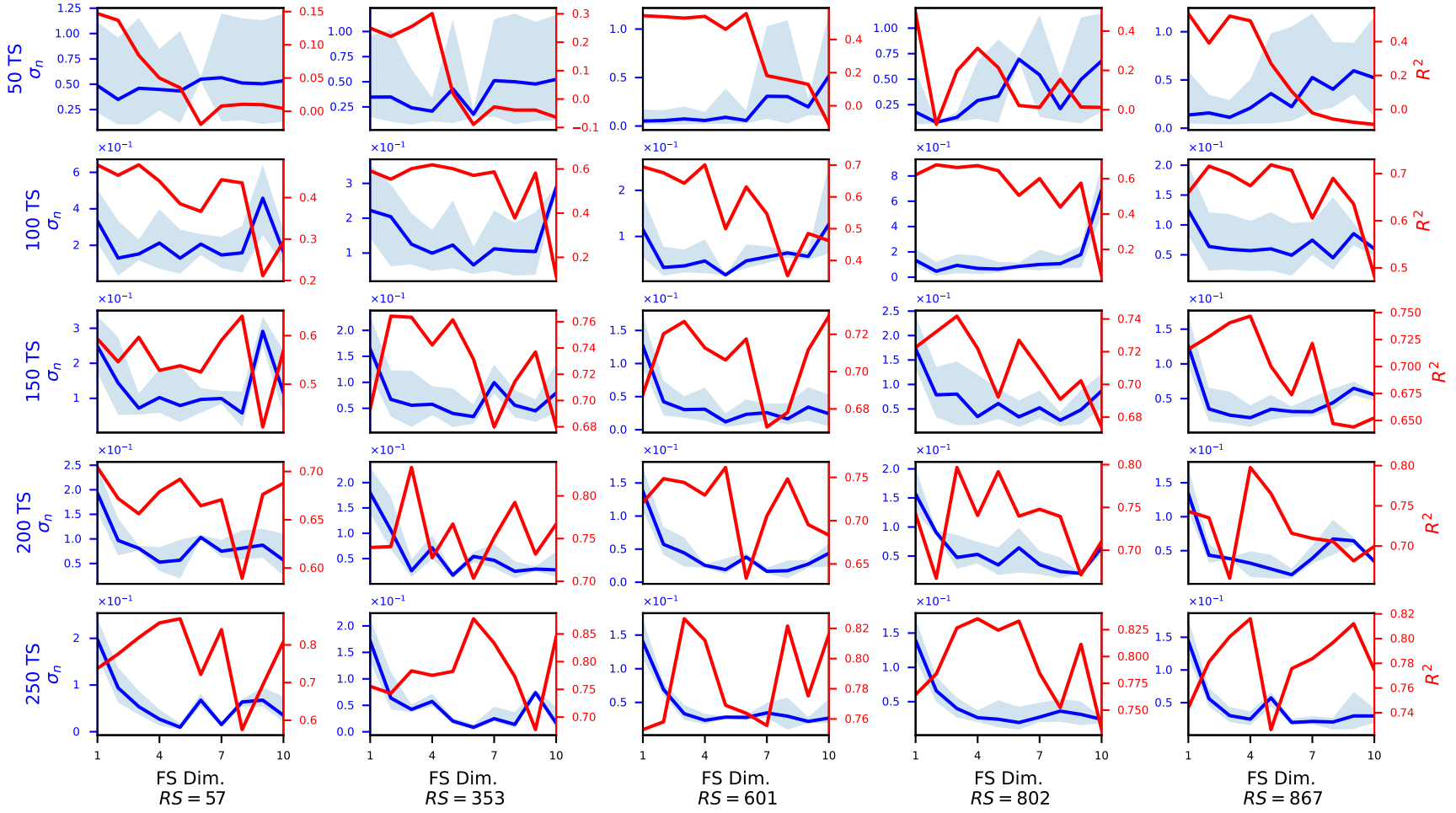


Figure A.22: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM z-moment dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

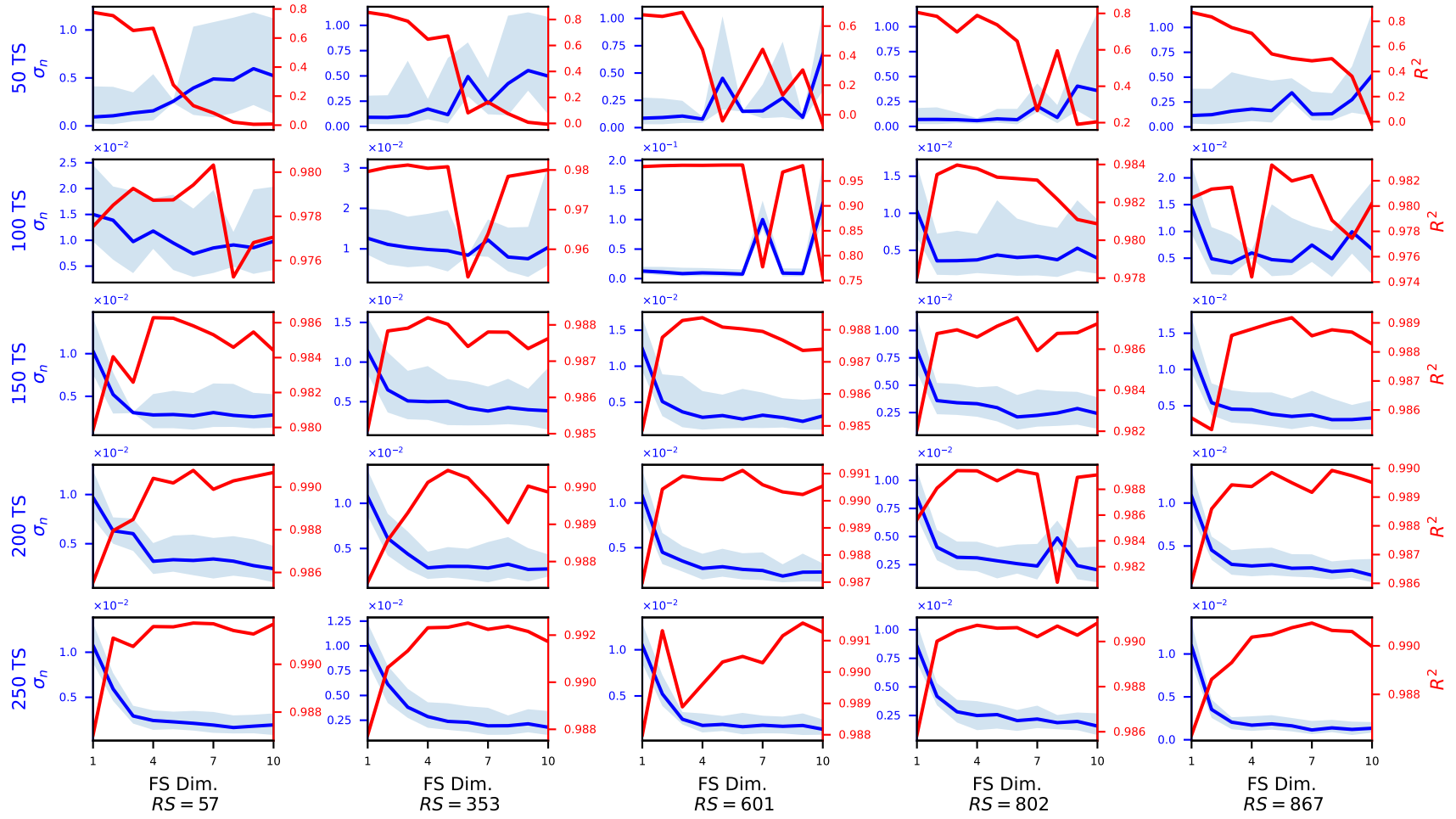


Figure A.23: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM sideforce dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

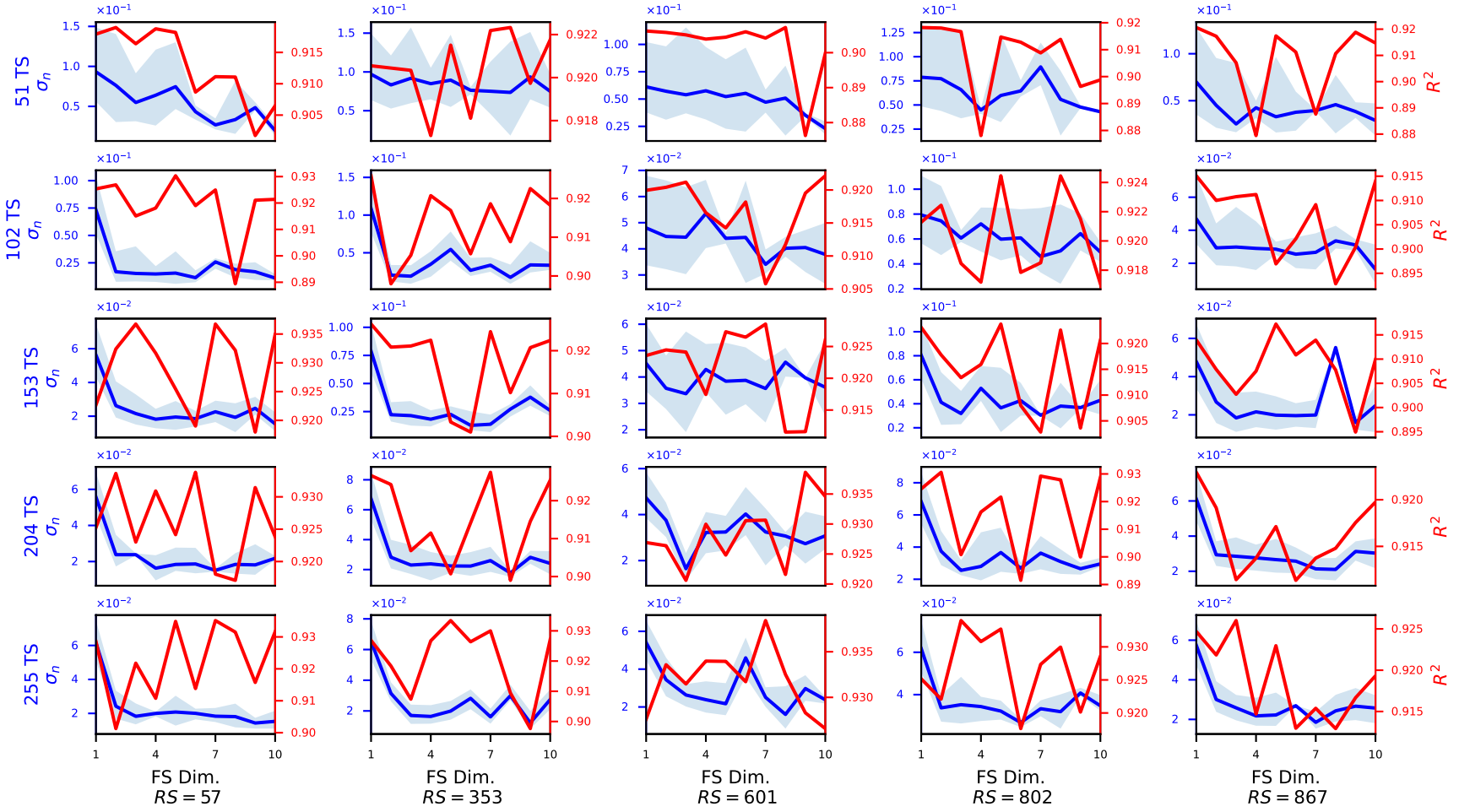


Figure A.24: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **RAE2822 lift dataset with 51 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

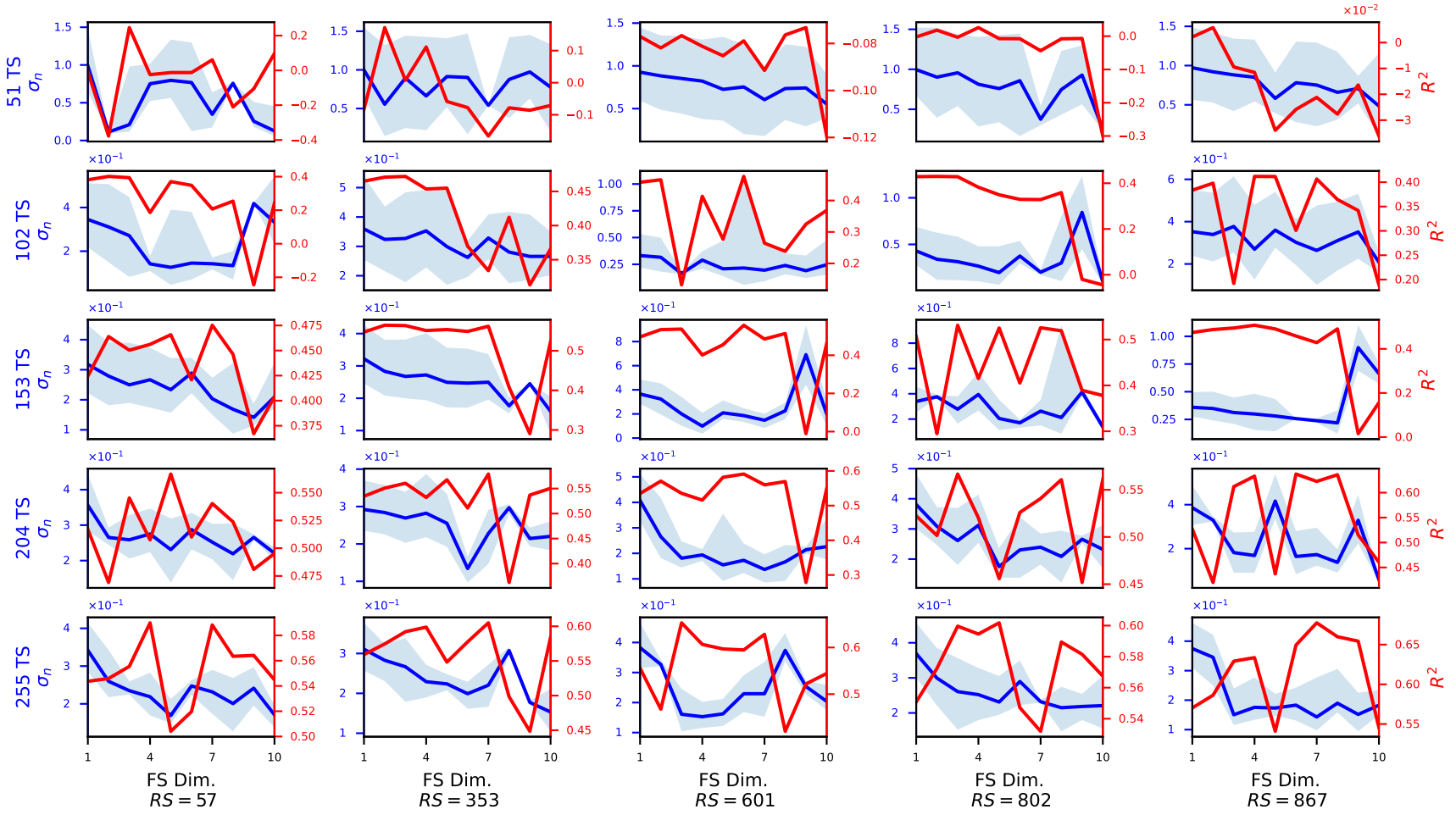


Figure A.25: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **RAE2822 z-moment dataset with 51 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

APPENDIX B

ADDITIONAL RESULTS FOR THE SEQUENTIAL APPROACH STUDY

This appendix contains the plots used to compare the noise variance decay and training time of the different approaches considered to speed up the selection of the FS dimension in section 2.4.4 and that were not shown for brevity of the main text. These results are discussed in the corresponding subsection of section 2.4.4 ("Analytical Functions", "Active Subspace Datasets", and "Aerodynamic Datasets").

B.1 Analytical Functions

B.1.1 Comparison of the Noise Variance Decay

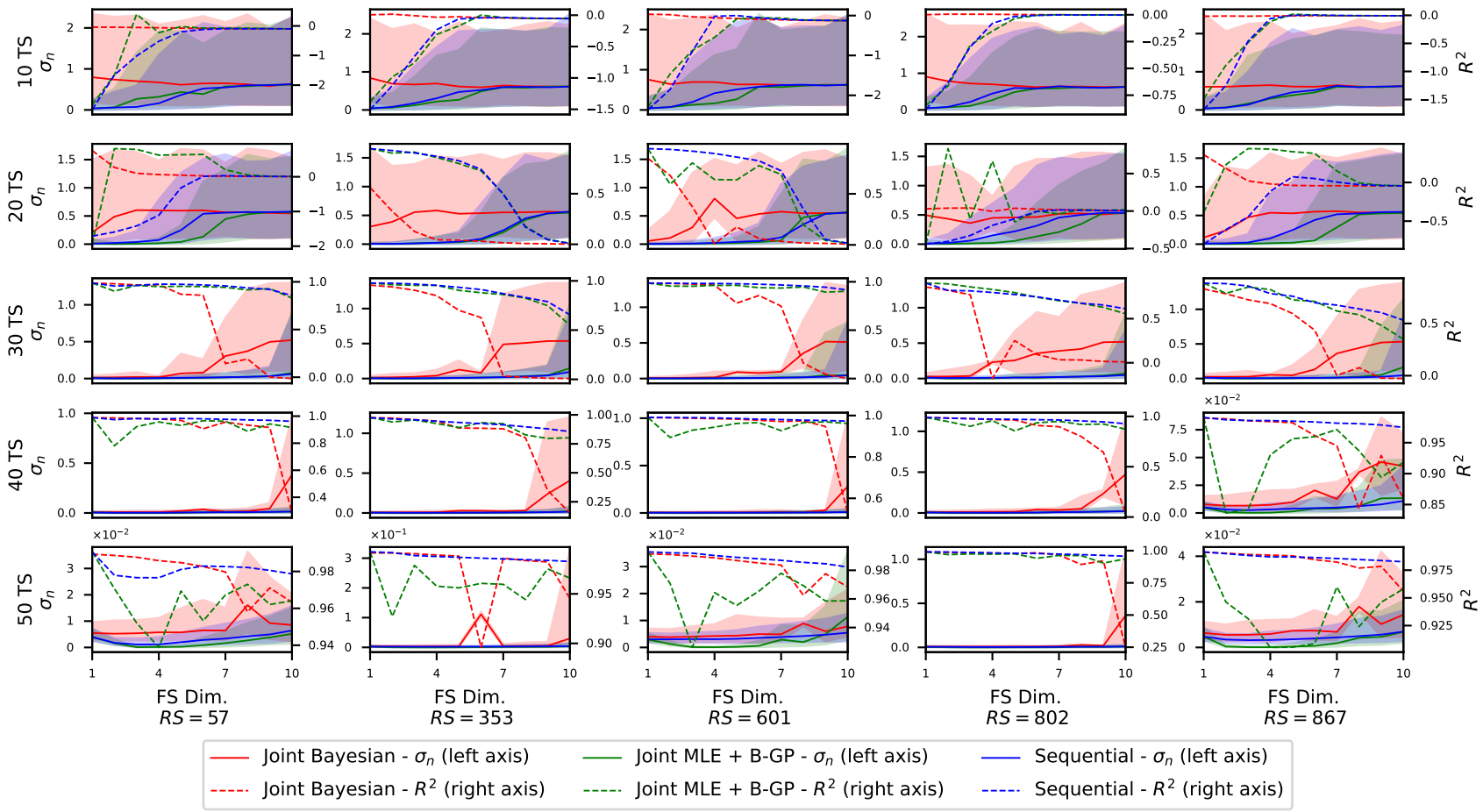


Figure B.1: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 10 input variables and a 1D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

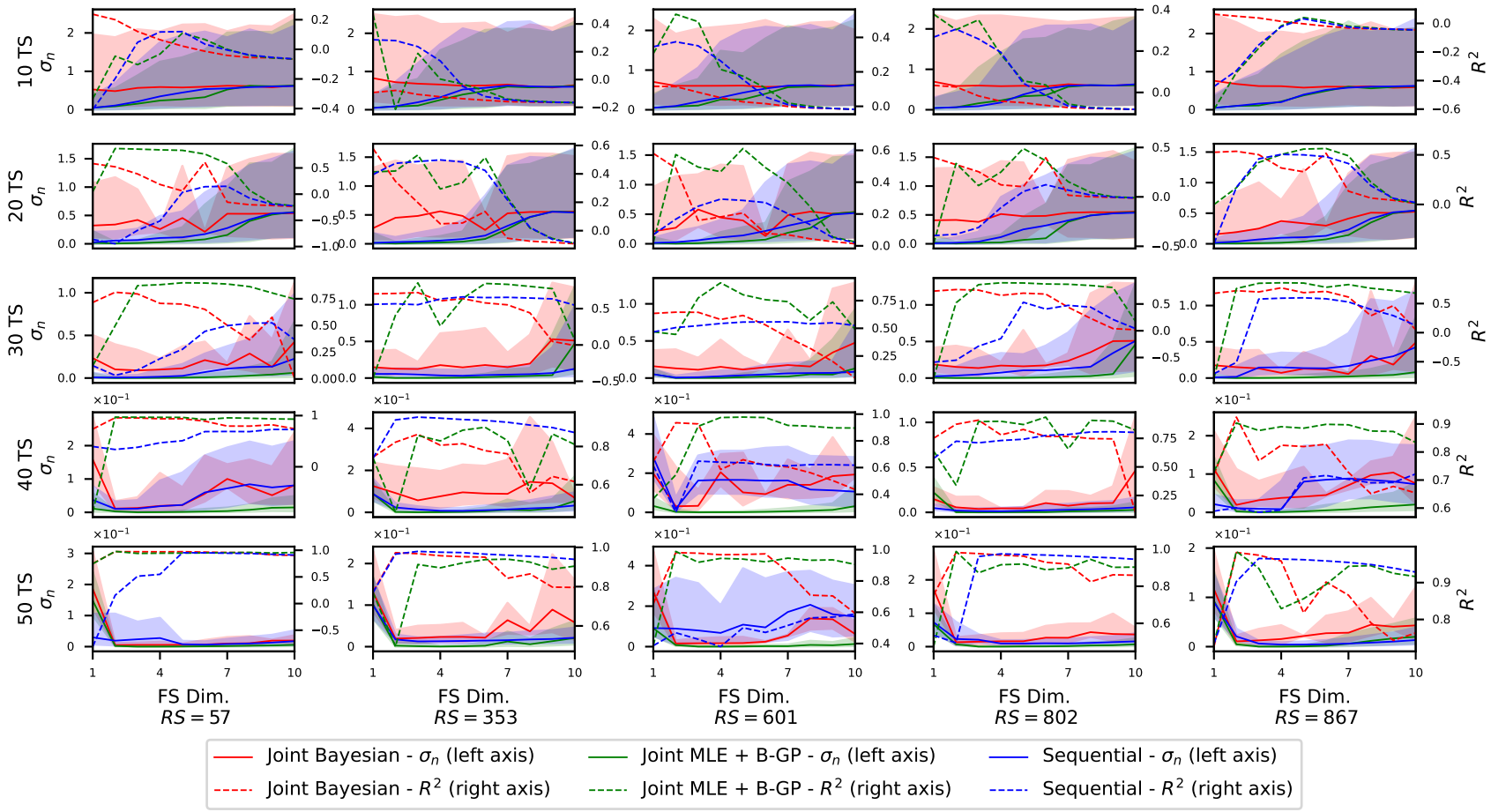


Figure B.2: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 10 input variables and a 2D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

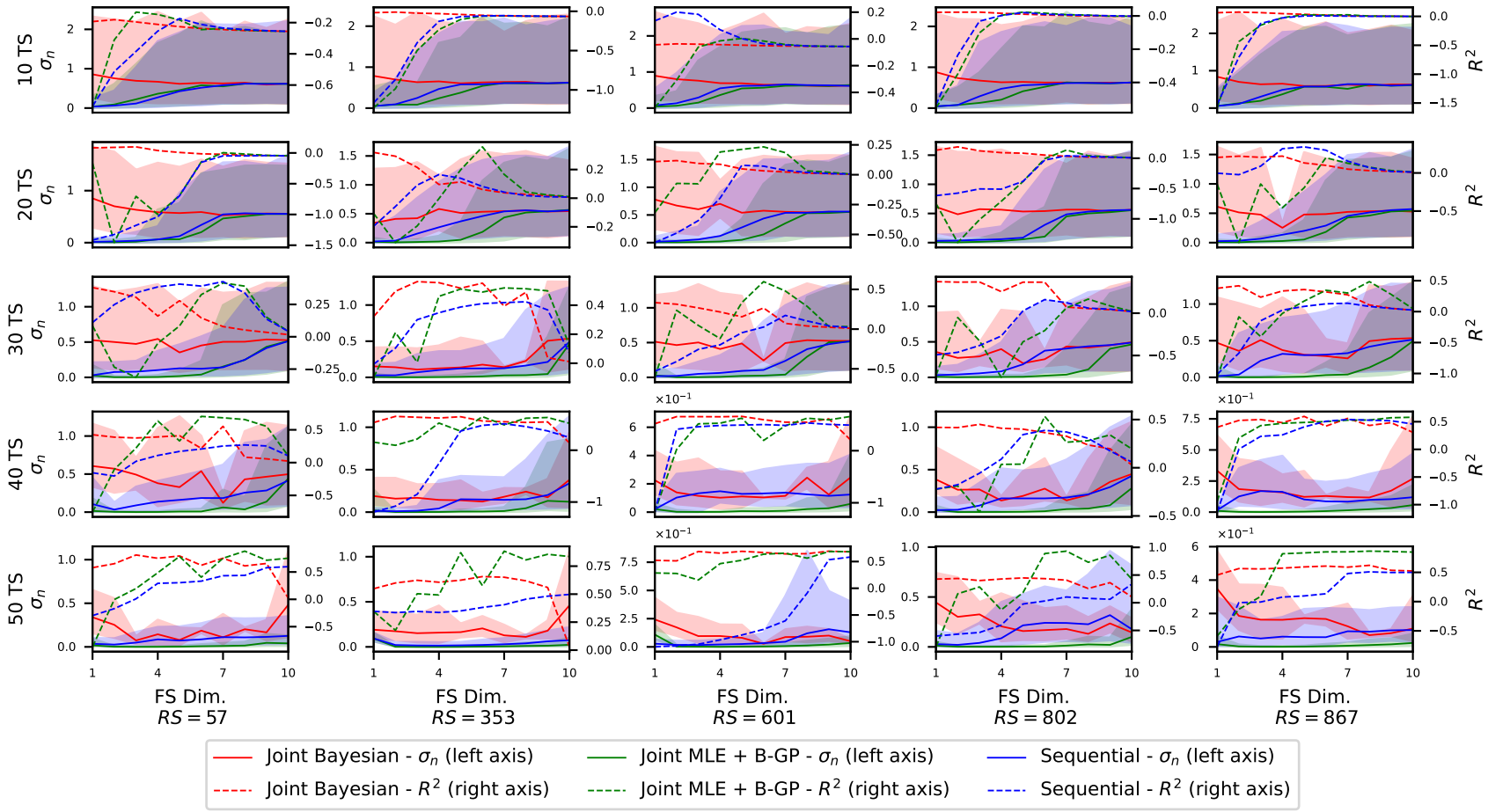


Figure B.3: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 10 input variables and a 5D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

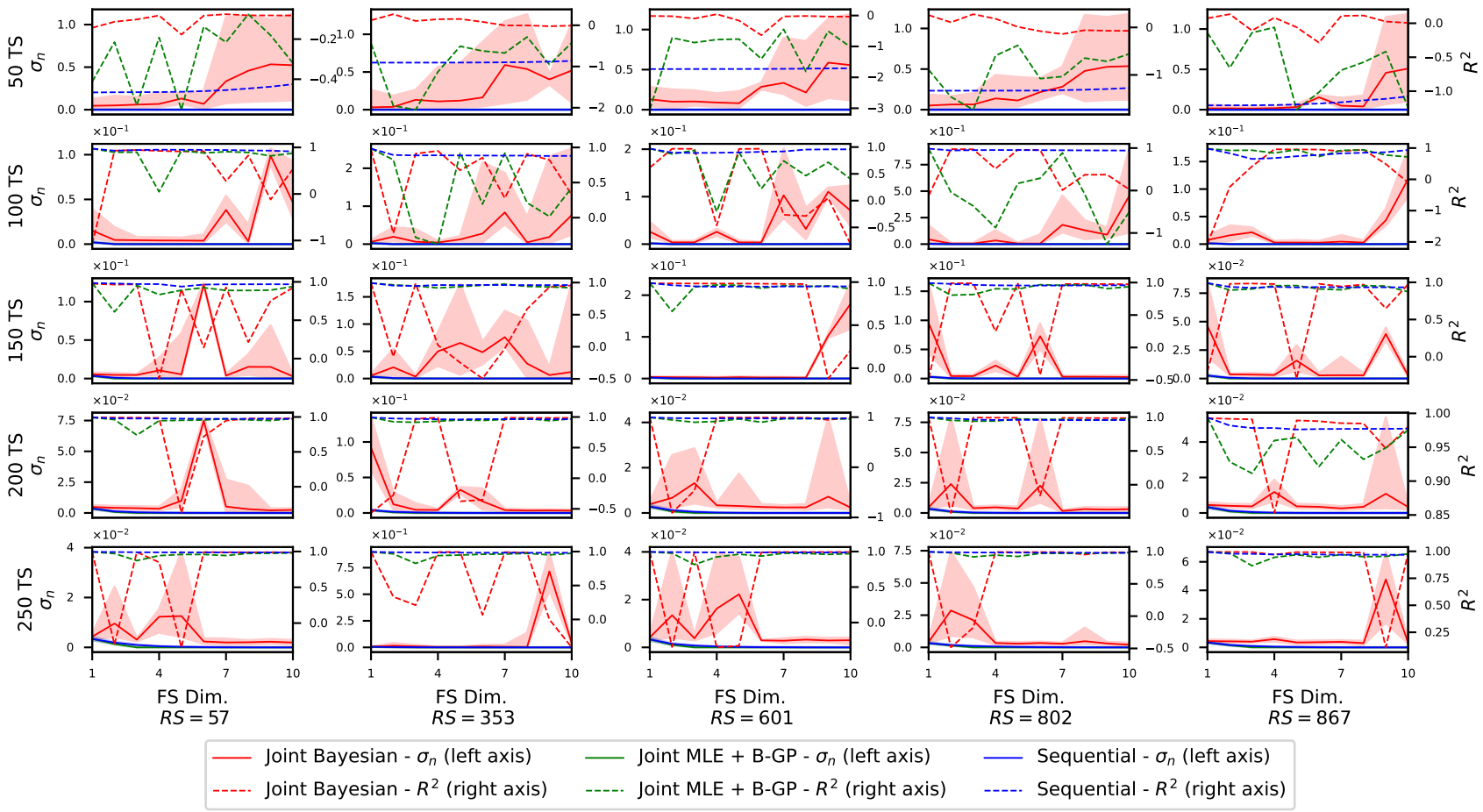


Figure B.4: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 50 input variables and a 1D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

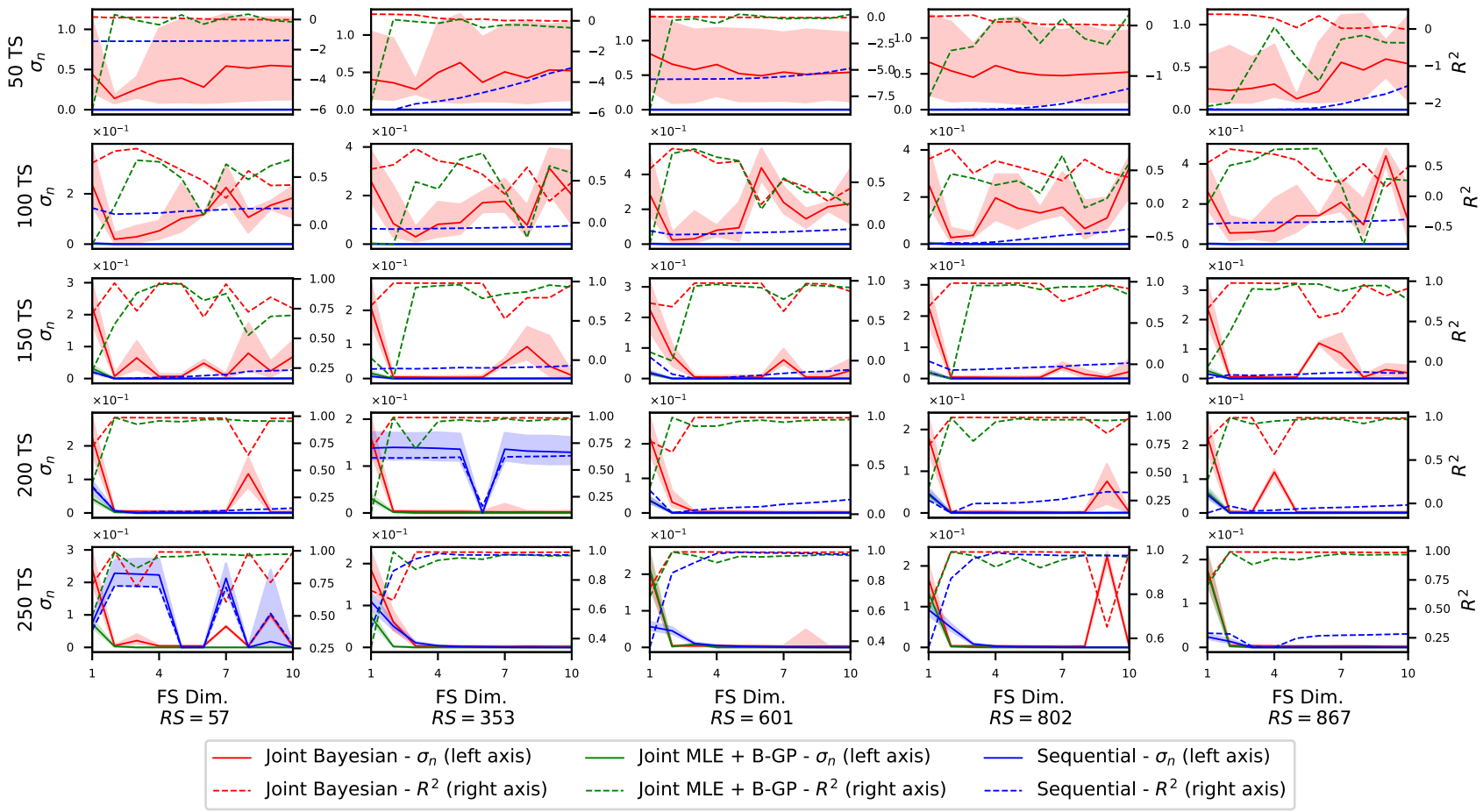


Figure B.5: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 50 input variables and a 2D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

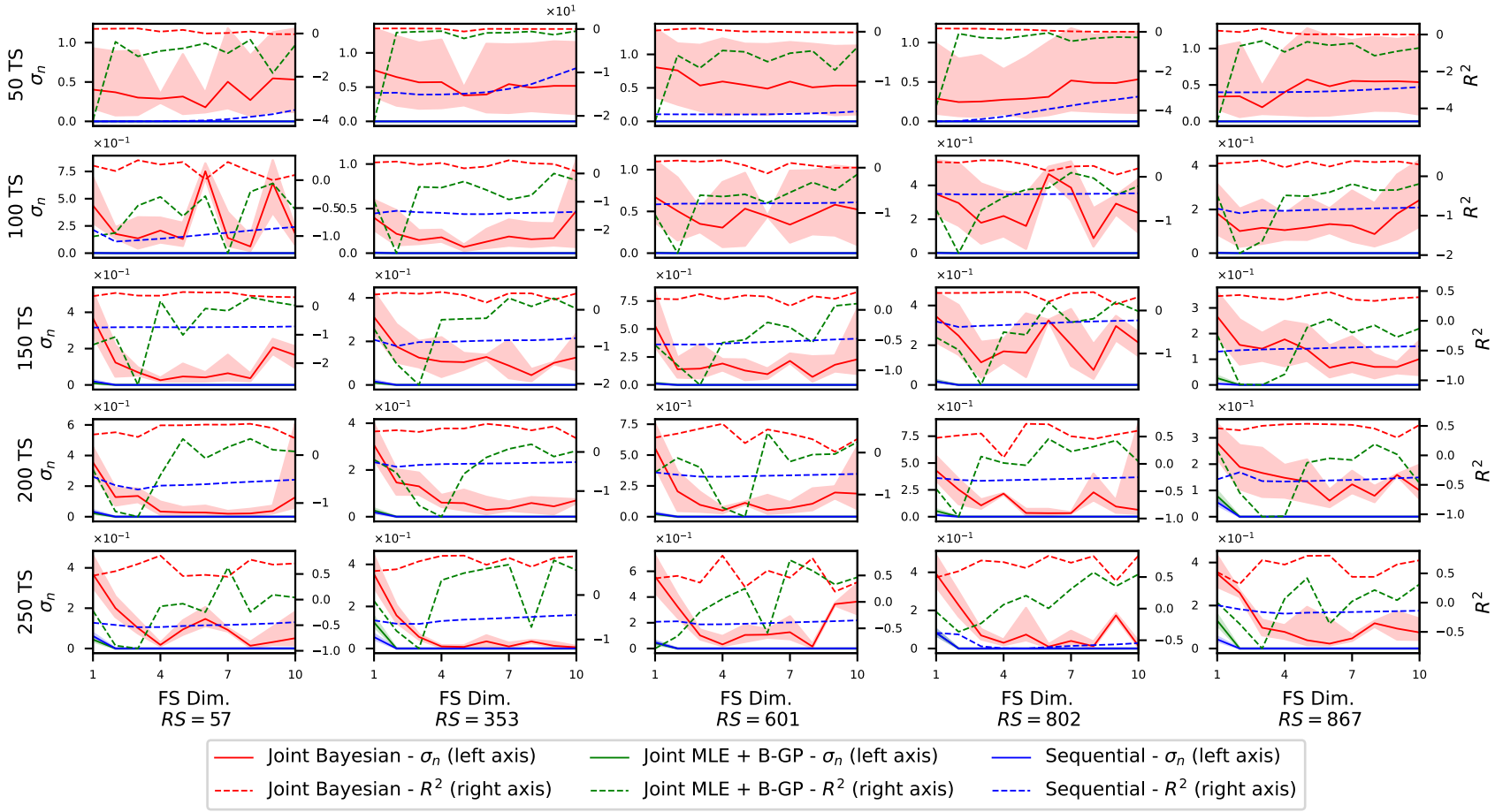


Figure B.6: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 50 input variables and a 5D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

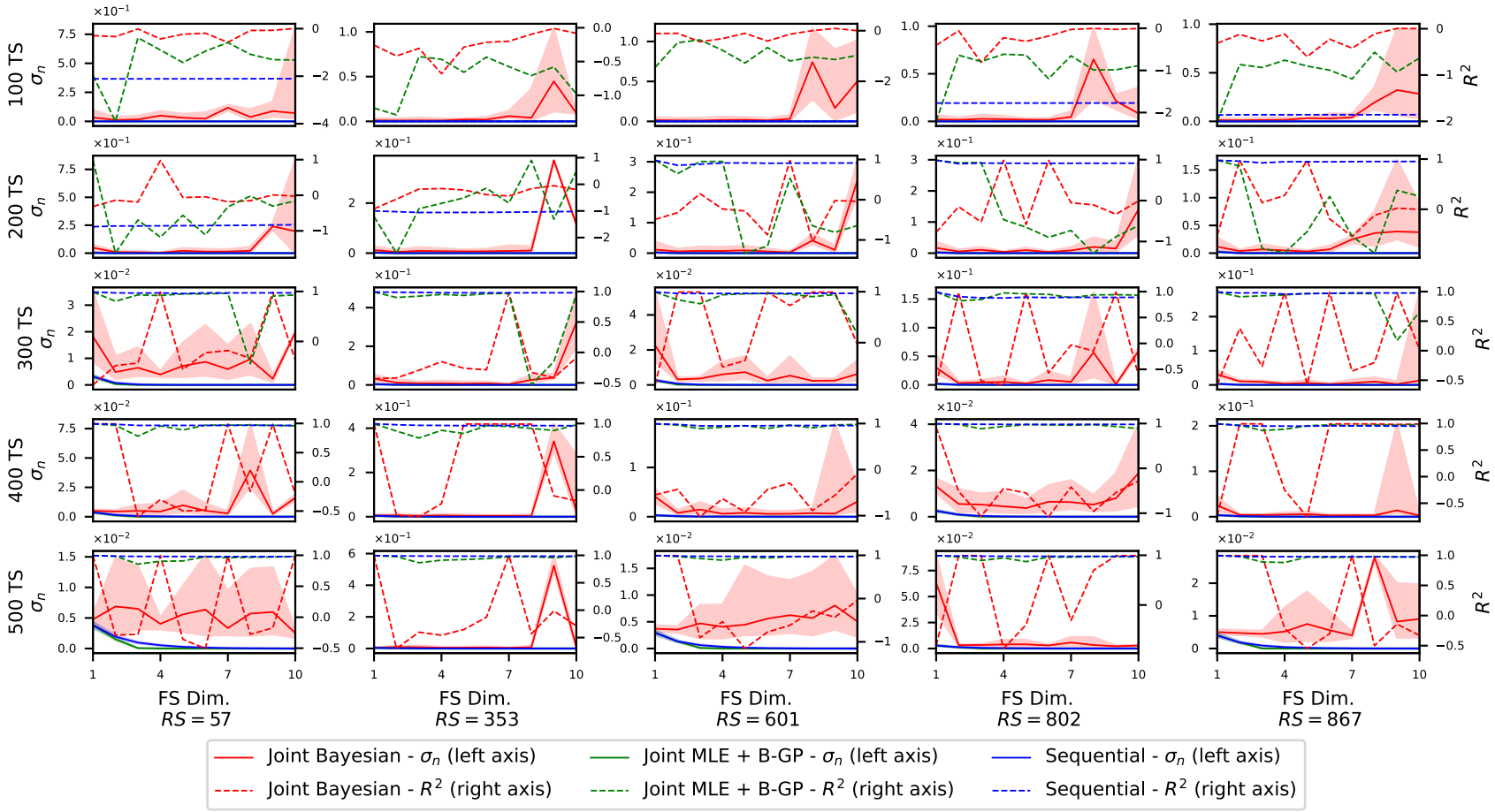


Figure B.7: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 100 input variables and a 1D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

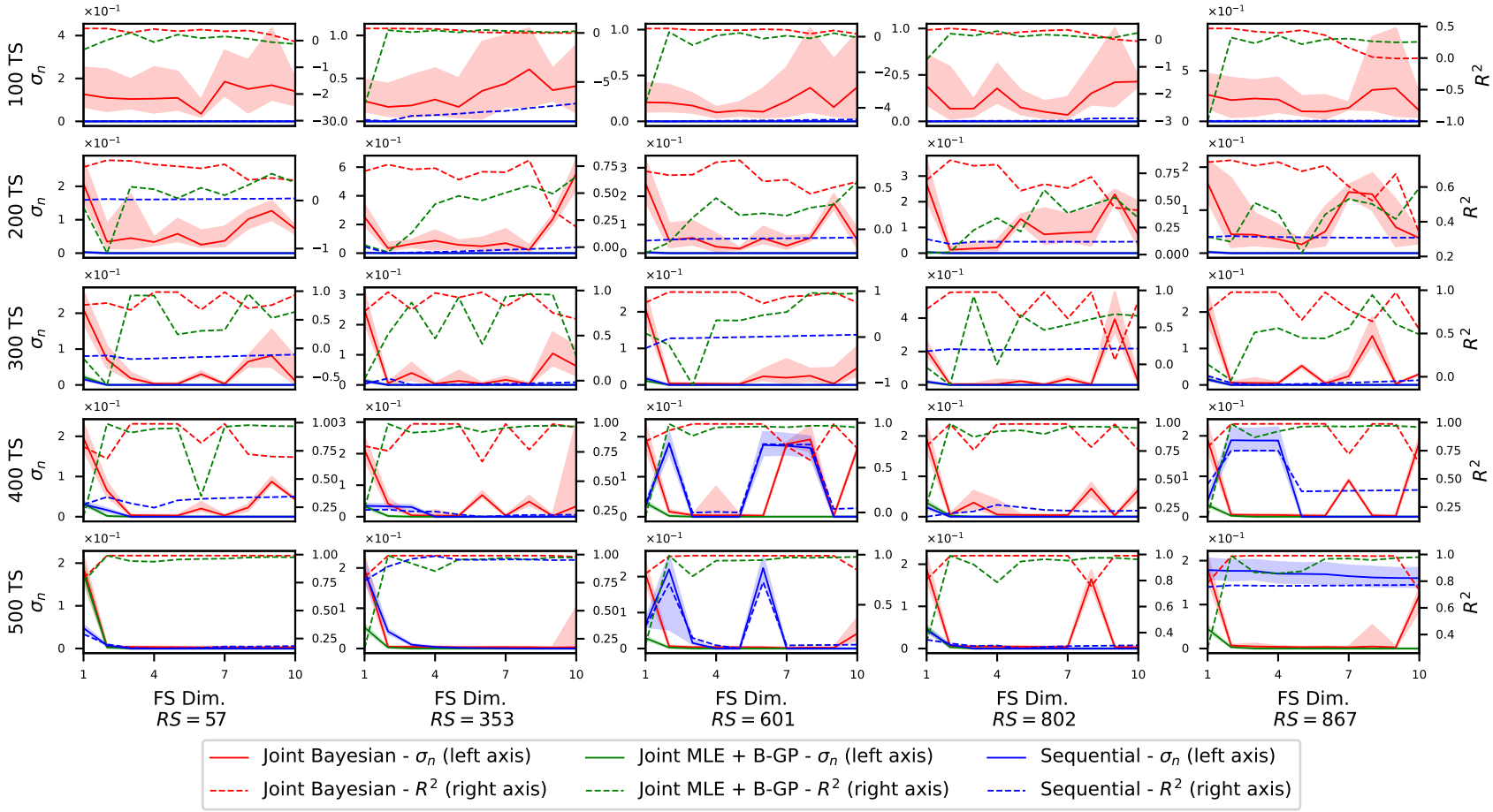


Figure B.8: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 100 input variables and a 2D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

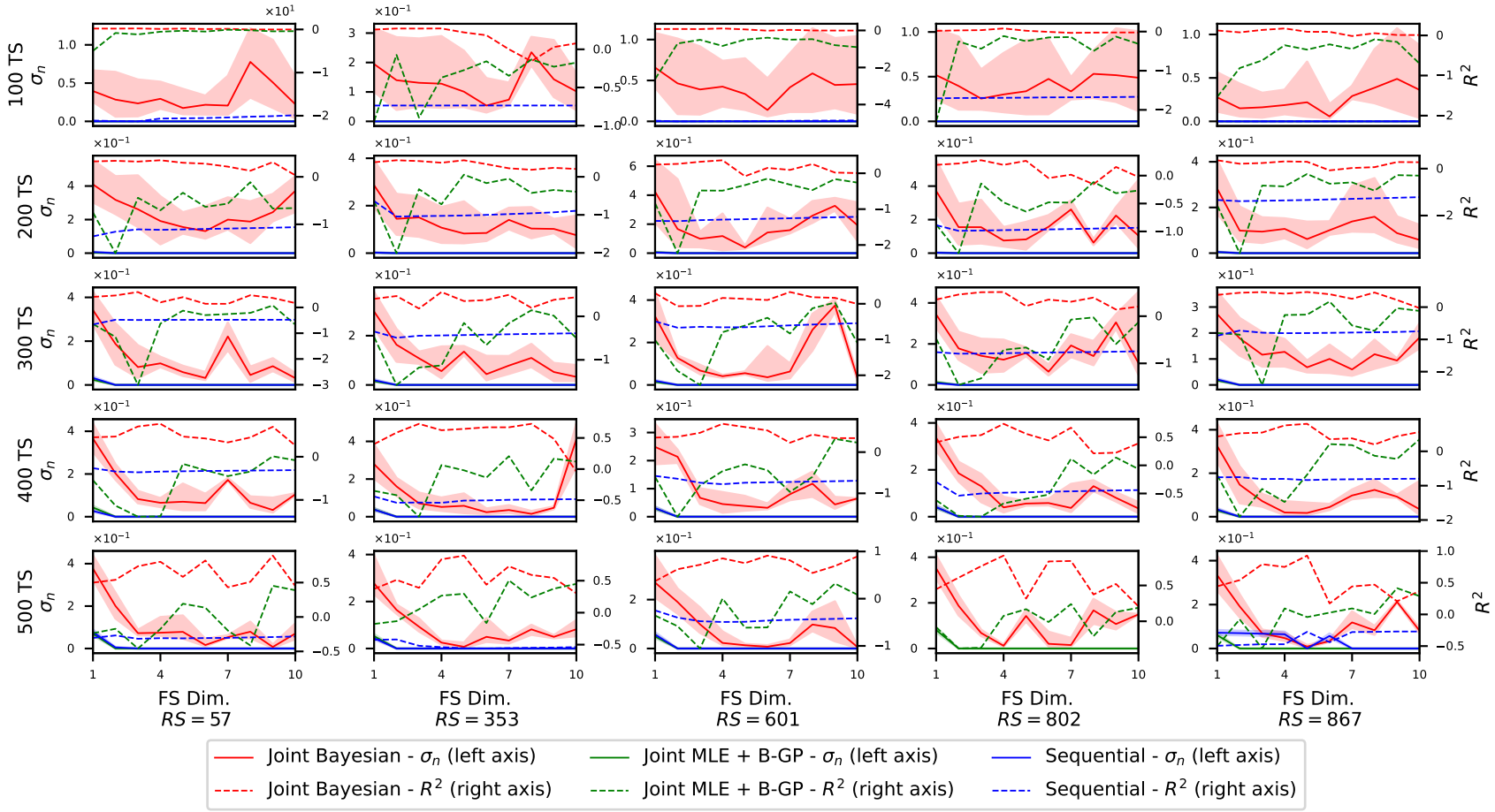


Figure B.9: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **quadratic test function with 100 input variables and a 5D feature space**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

B.1.2 Comparison of the Training Durations

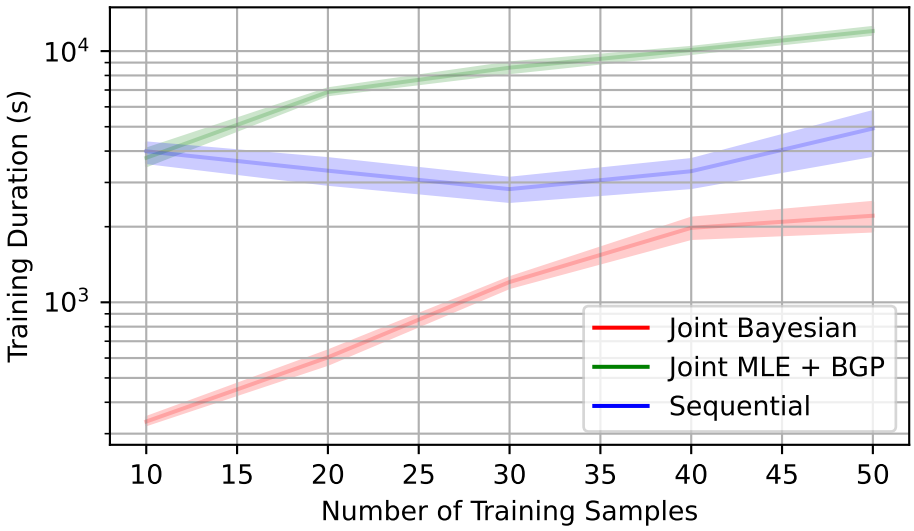


Figure B.10: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 10 input variables and a 1D feature space.

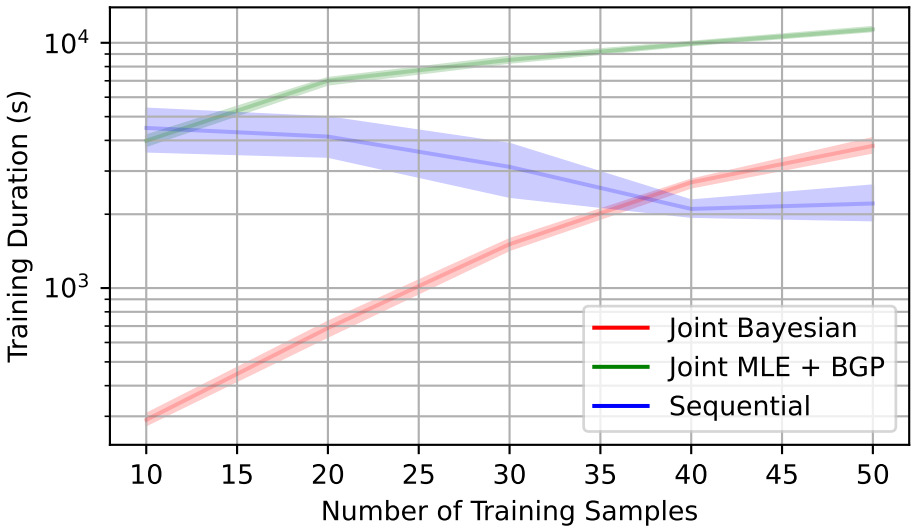


Figure B.11: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 10 input variables and a 2D feature space.

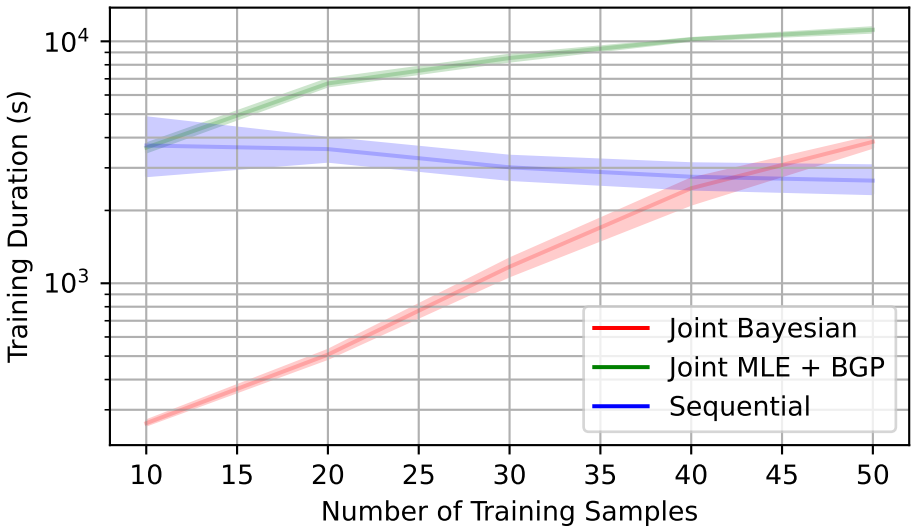


Figure B.12: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 10 input variables and a 5D feature space.

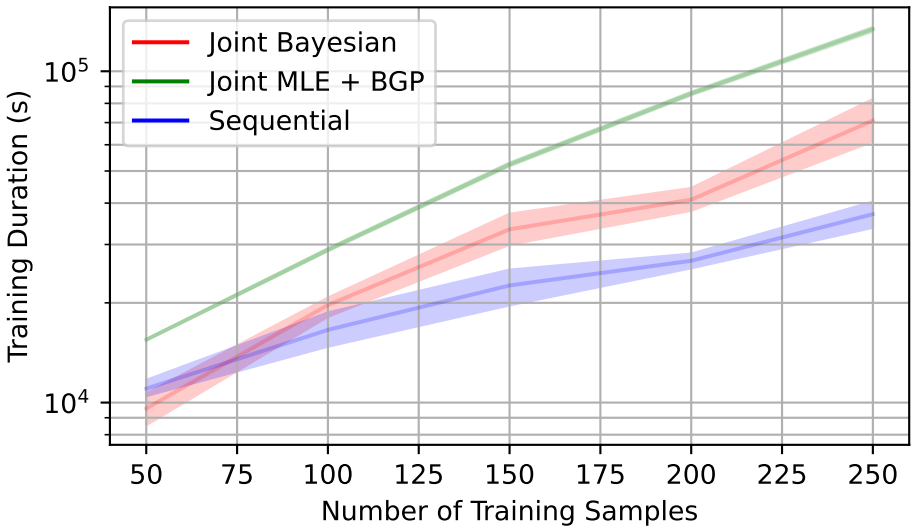


Figure B.13: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 50 input variables and a 1D feature space.

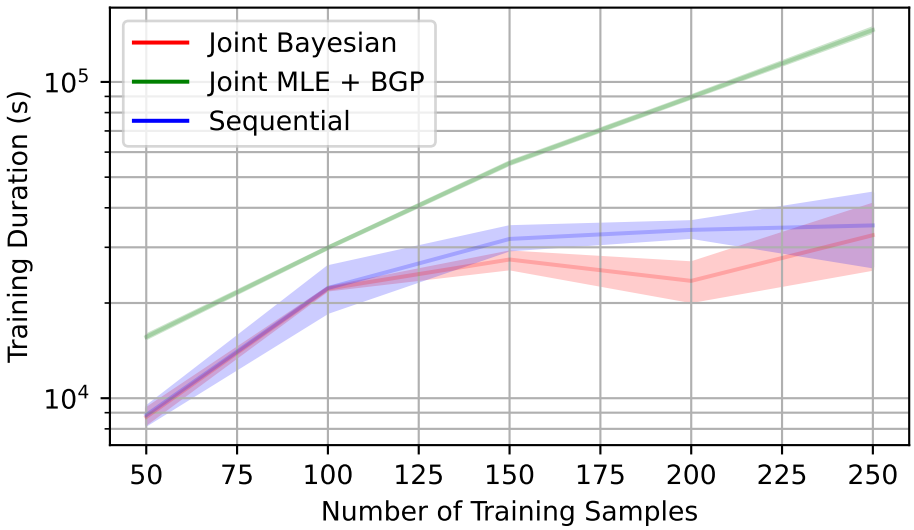


Figure B.14: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 50 input variables and a 2D feature space.

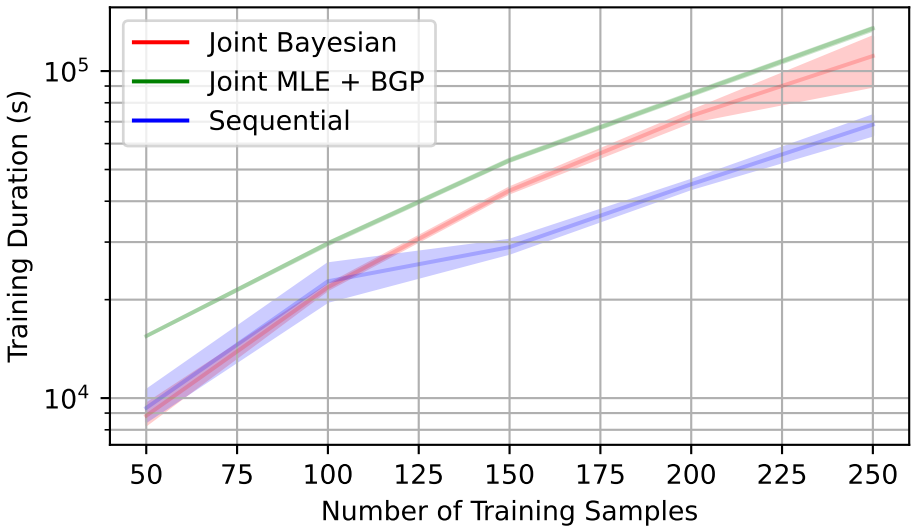


Figure B.15: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 50 input variables and a 5D feature space.

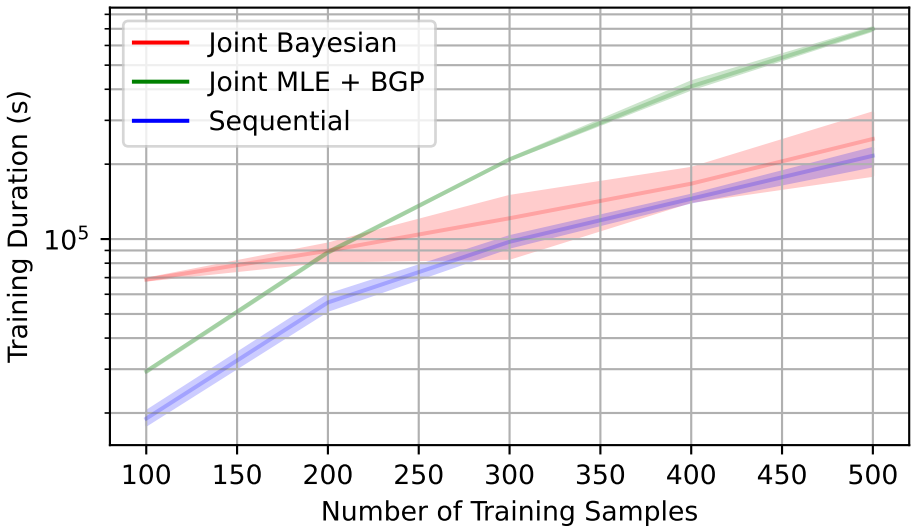


Figure B.16: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 100 input variables and a 1D feature space.

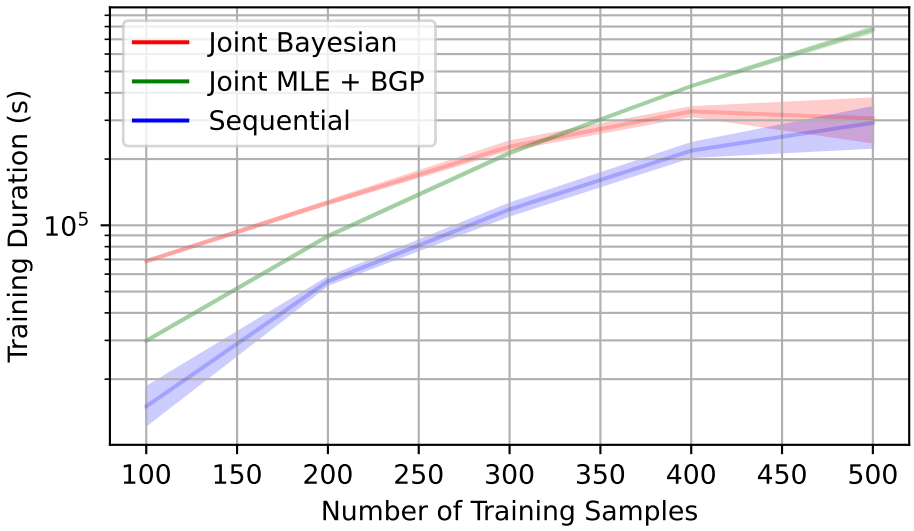


Figure B.17: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 100 input variables and a 2D feature space.

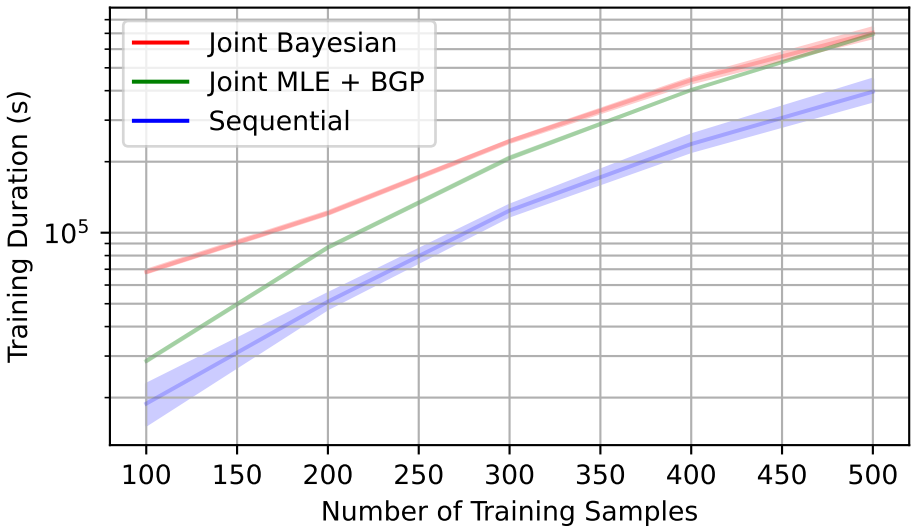


Figure B.18: Evolution of the training duration as a function of the number of training samples for the quadratic test function with 100 input variables and a 5D feature space.

B.2 Active Subspace Datasets

B.2.1 Comparison of the Noise Variance Decay

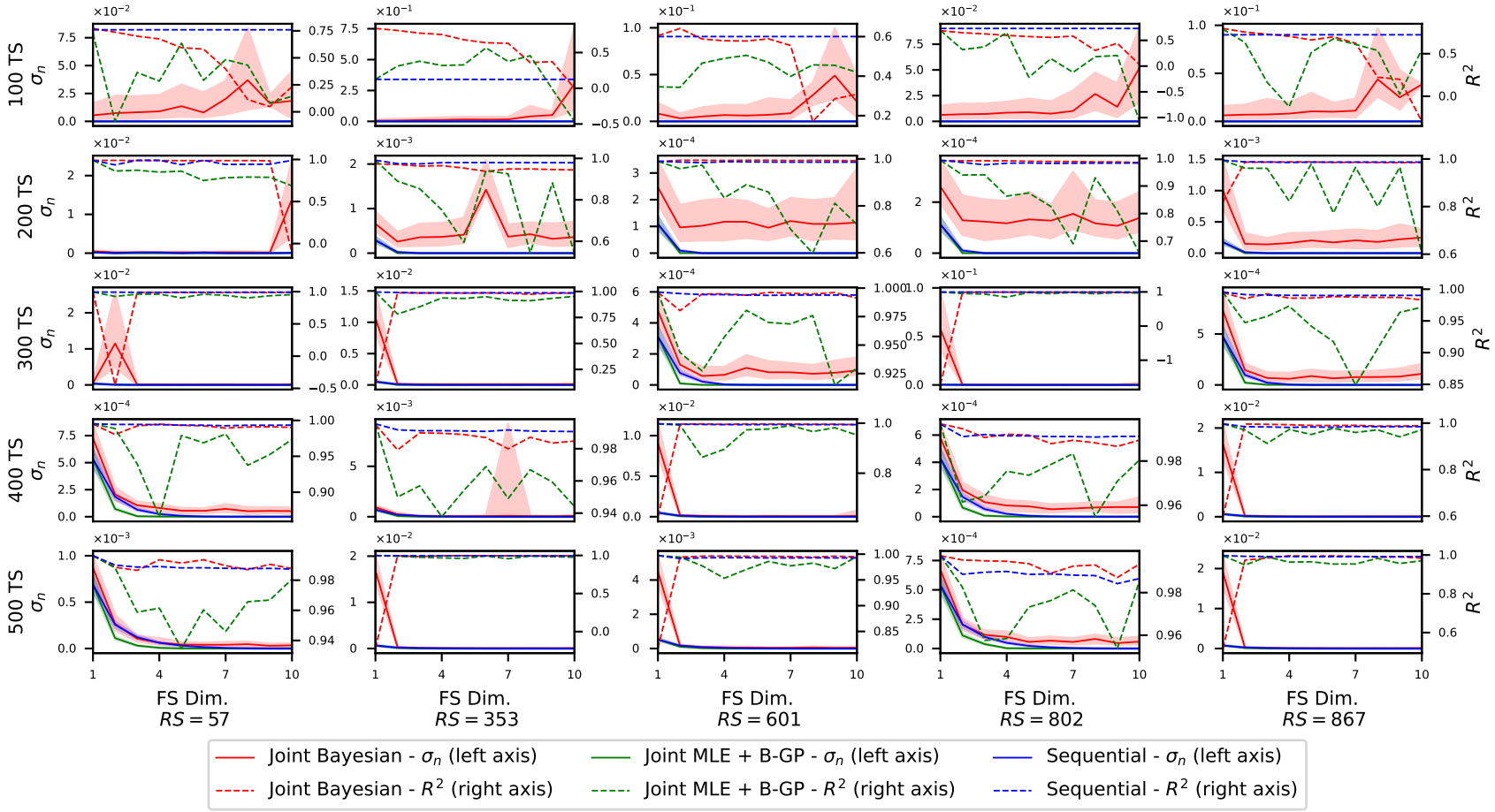


Figure B.19: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **Elliptic PDE (y_{long}) dataset with 100 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

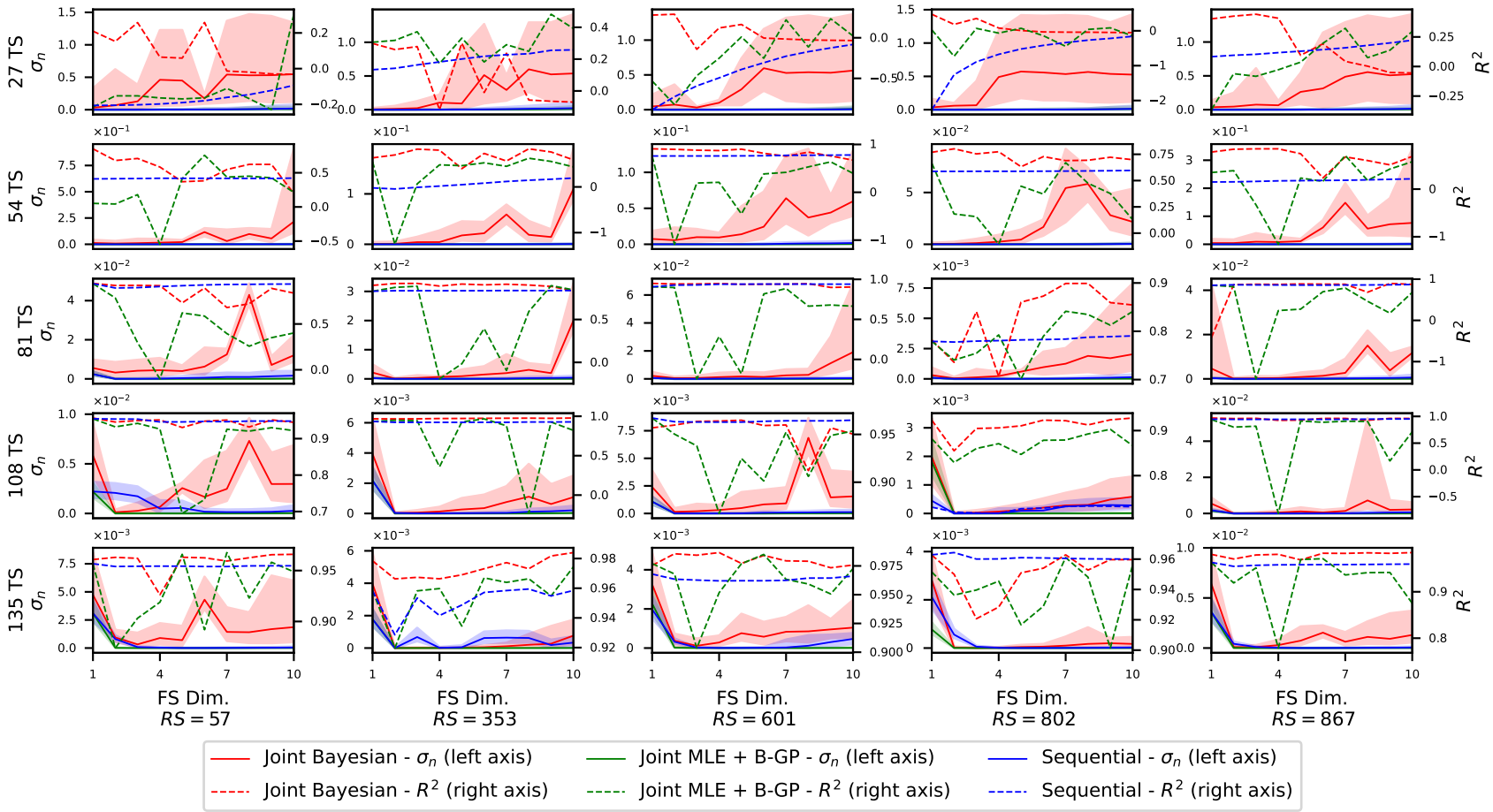


Figure B.20: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **HIV (y_{3400}) dataset with 27 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

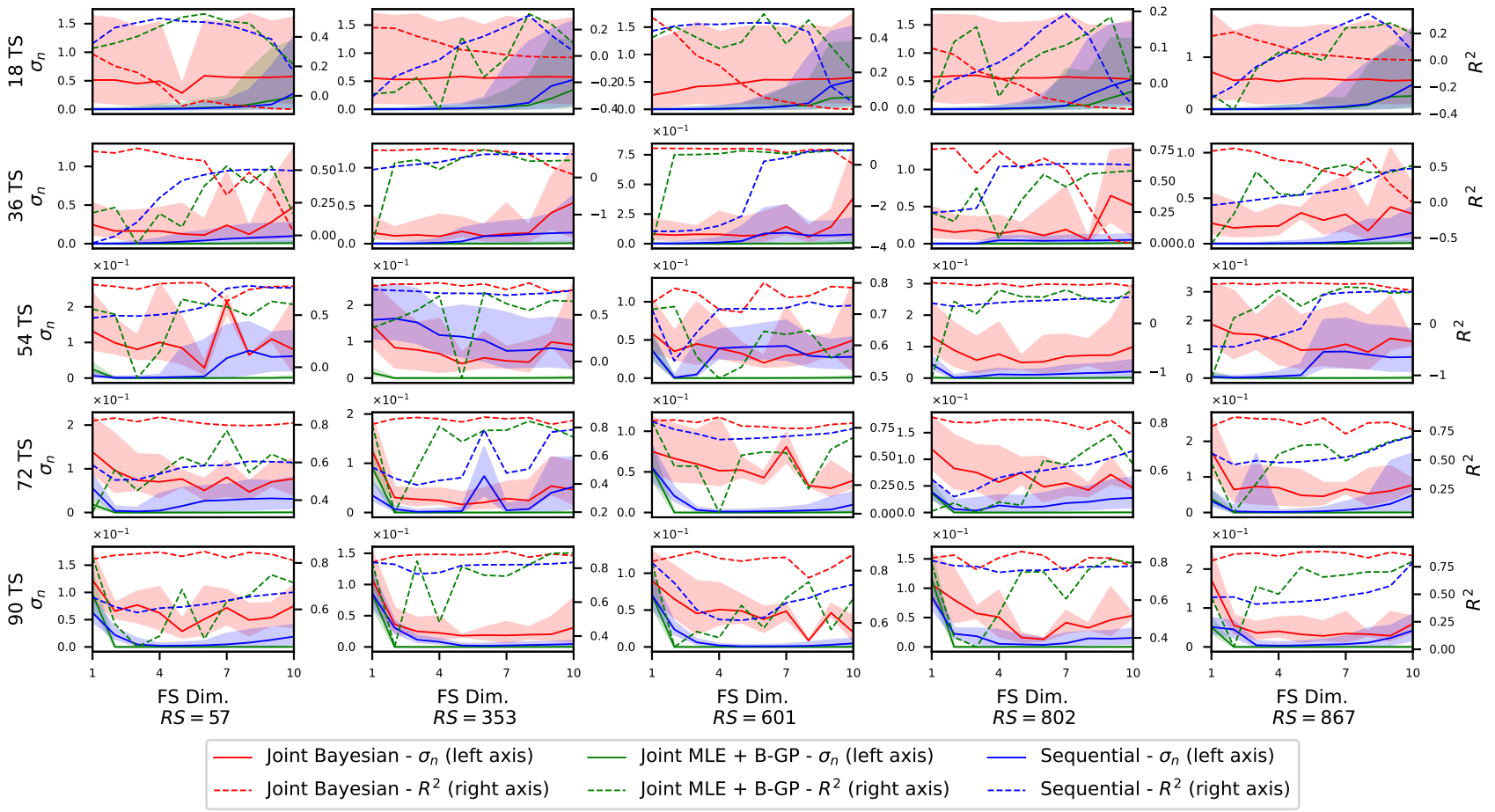


Figure B.21: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **NACA0012 drag dataset with 18 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

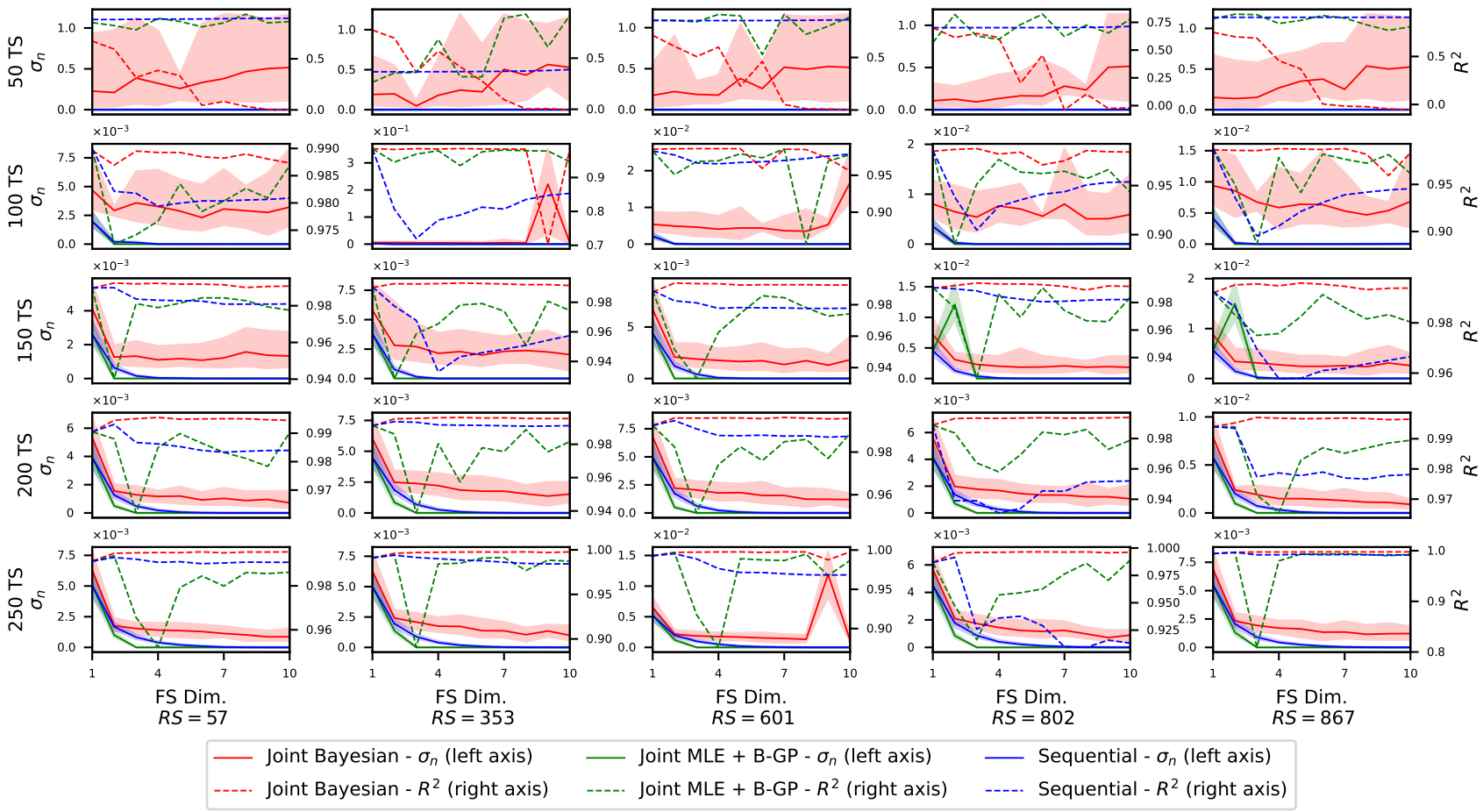


Figure B.22: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **ONERA M6 lift dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

B.2.2 Comparison of the Training Durations

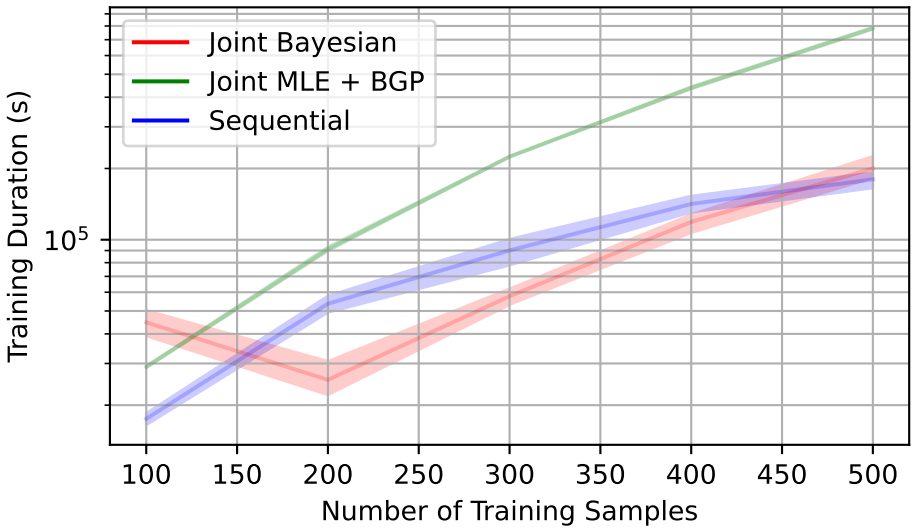


Figure B.23: Evolution of the training duration as a function of the number of training samples for the Elliptic PDE (y_{long}) dataset with 100 inputs.

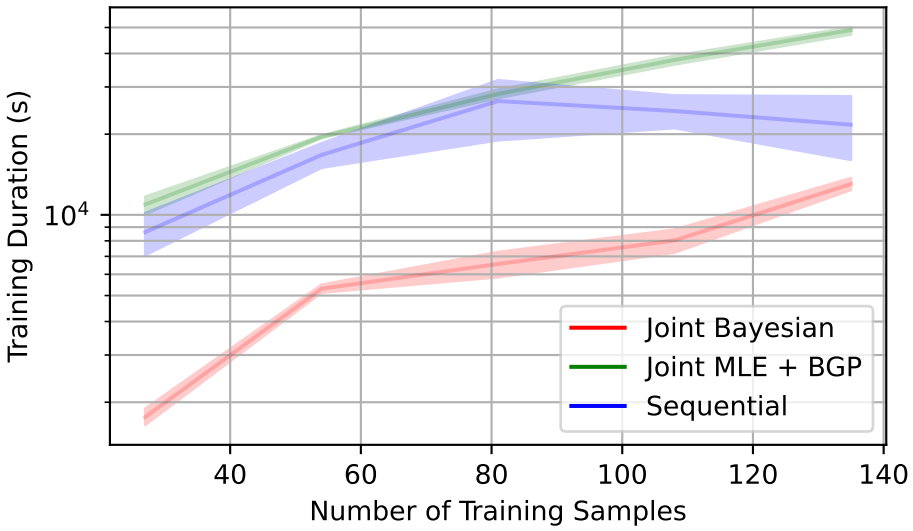


Figure B.24: Evolution of the training duration as a function of the number of training samples for the HIV (y_{3400}) dataset with 27 inputs.

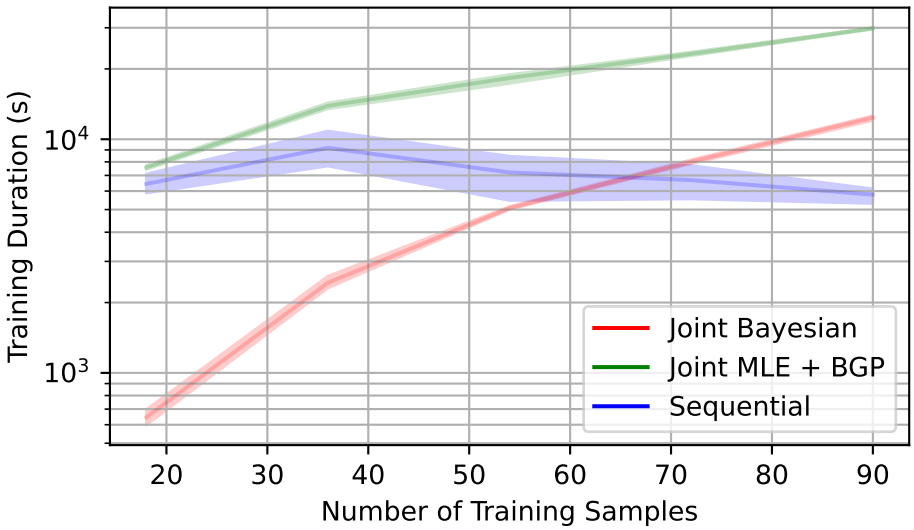


Figure B.25: Evolution of the training duration as a function of the number of training samples for the NACA0012 drag dataset with 18 inputs.

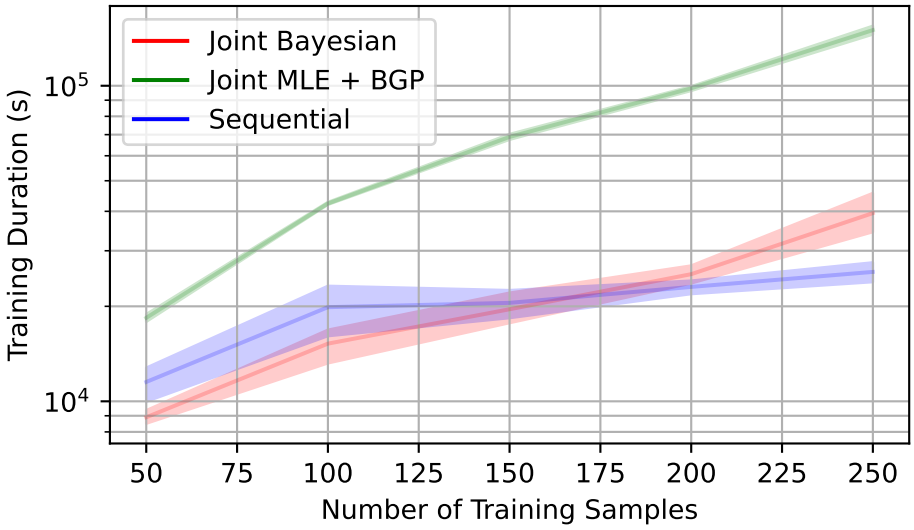


Figure B.26: Evolution of the training duration as a function of the number of training samples for the ONERA M6 lift dataset with 50 inputs.

B.3 Aerodynamic Datasets

B.3.1 Comparison of the Noise Variance Decay

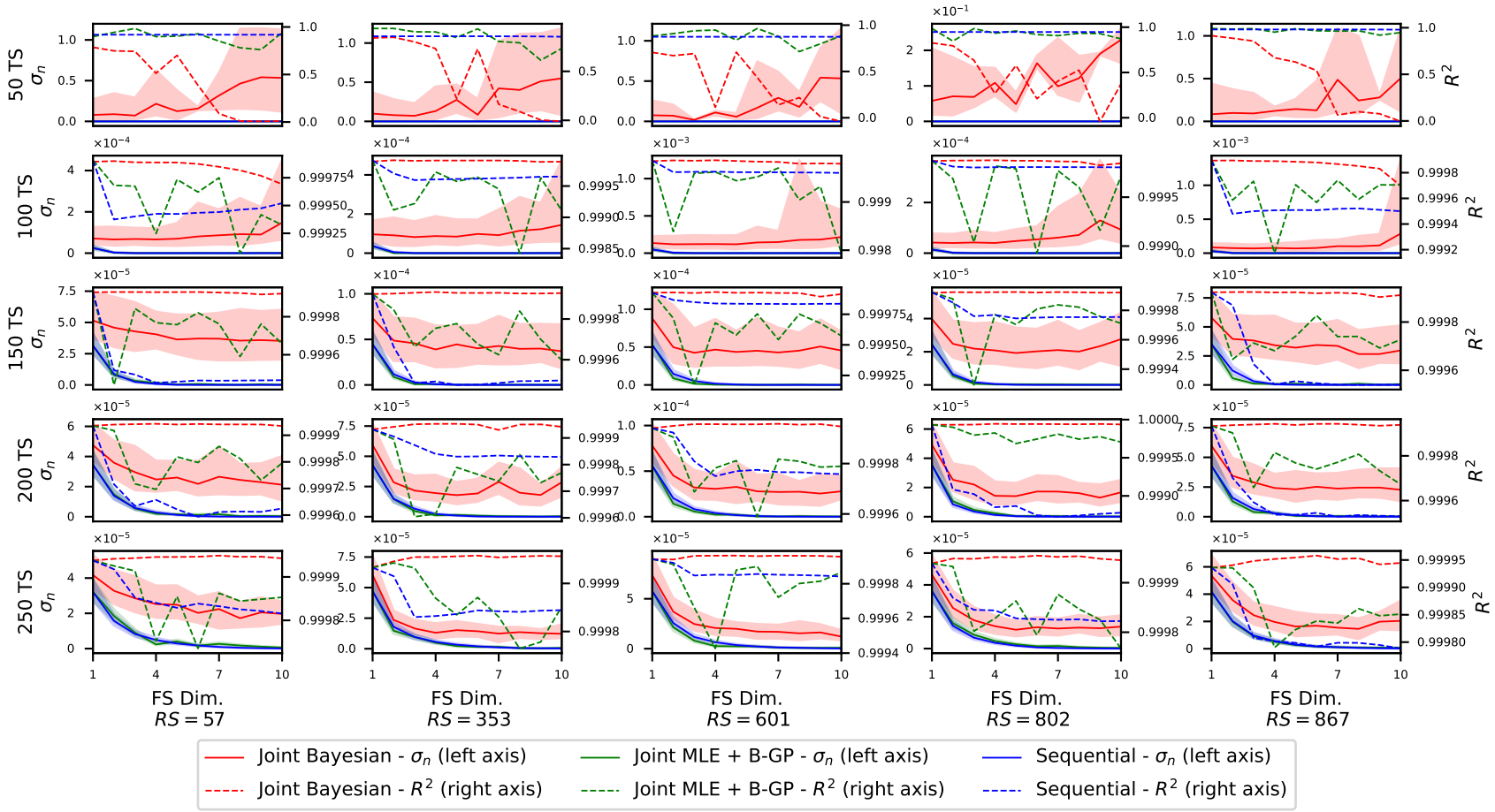


Figure B.27: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **subsonic CRM lift dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

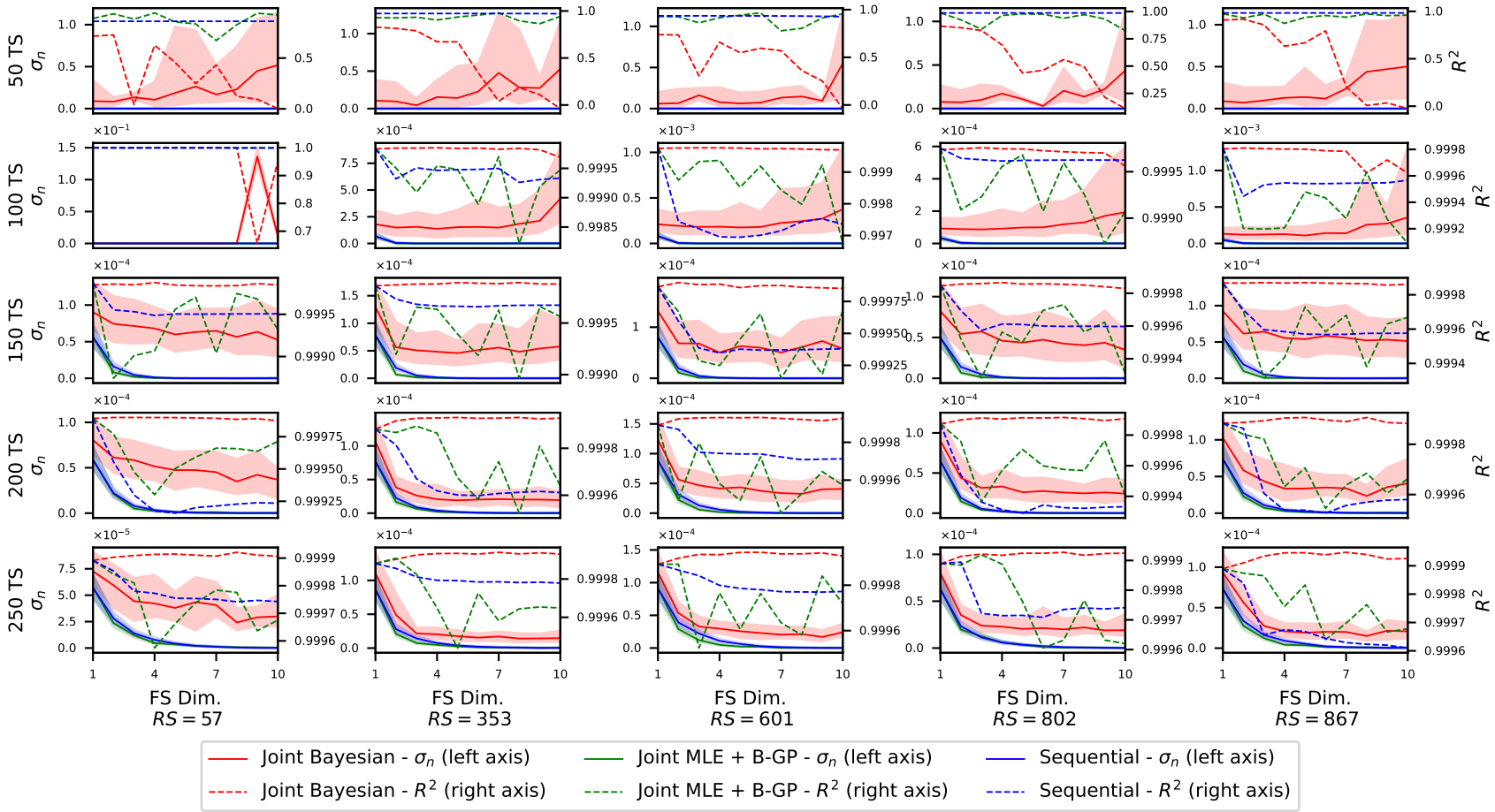


Figure B.28: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **subsonic CRM x-moment dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

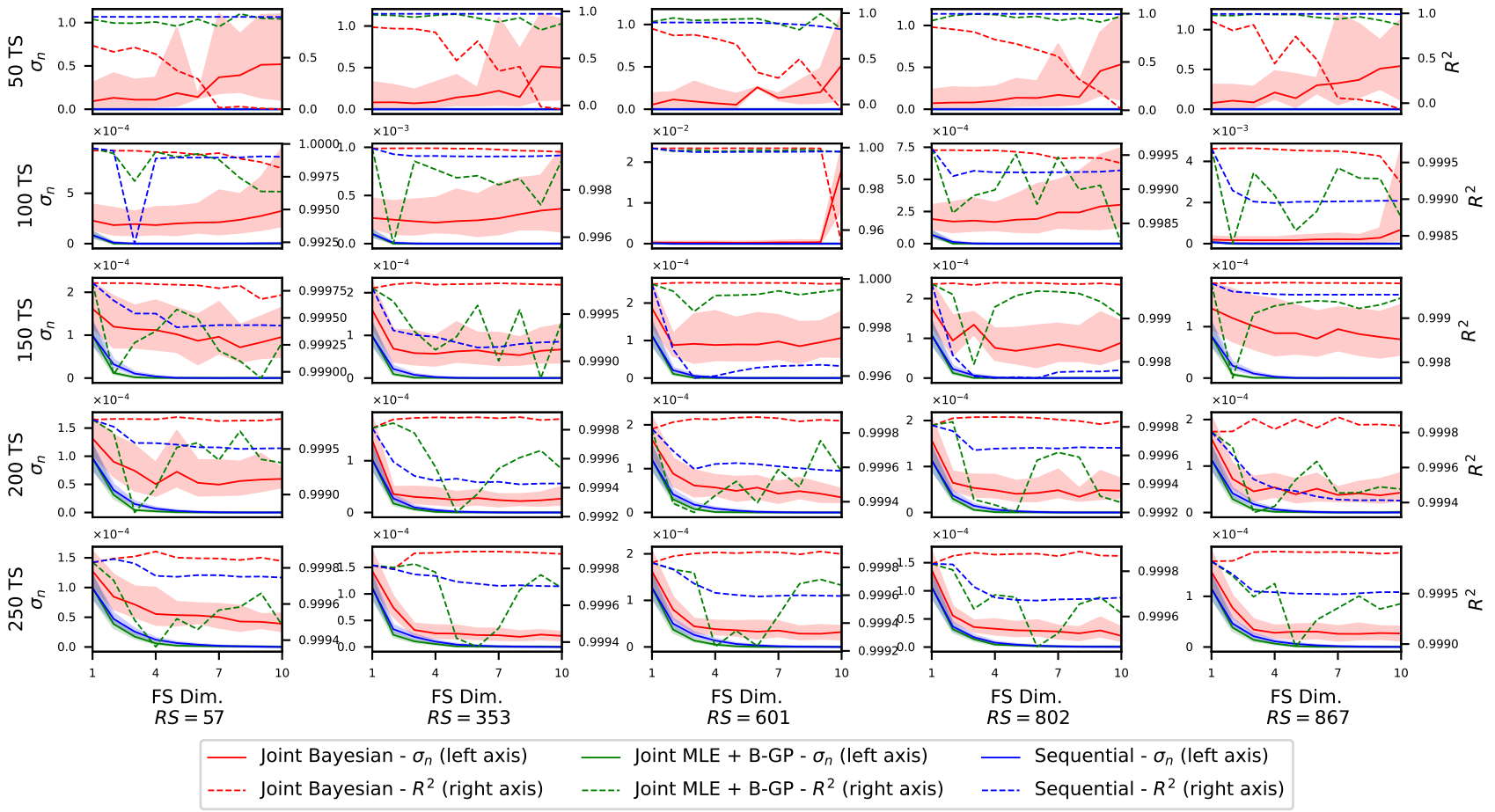


Figure B.29: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **subsonic CRM y-moment dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

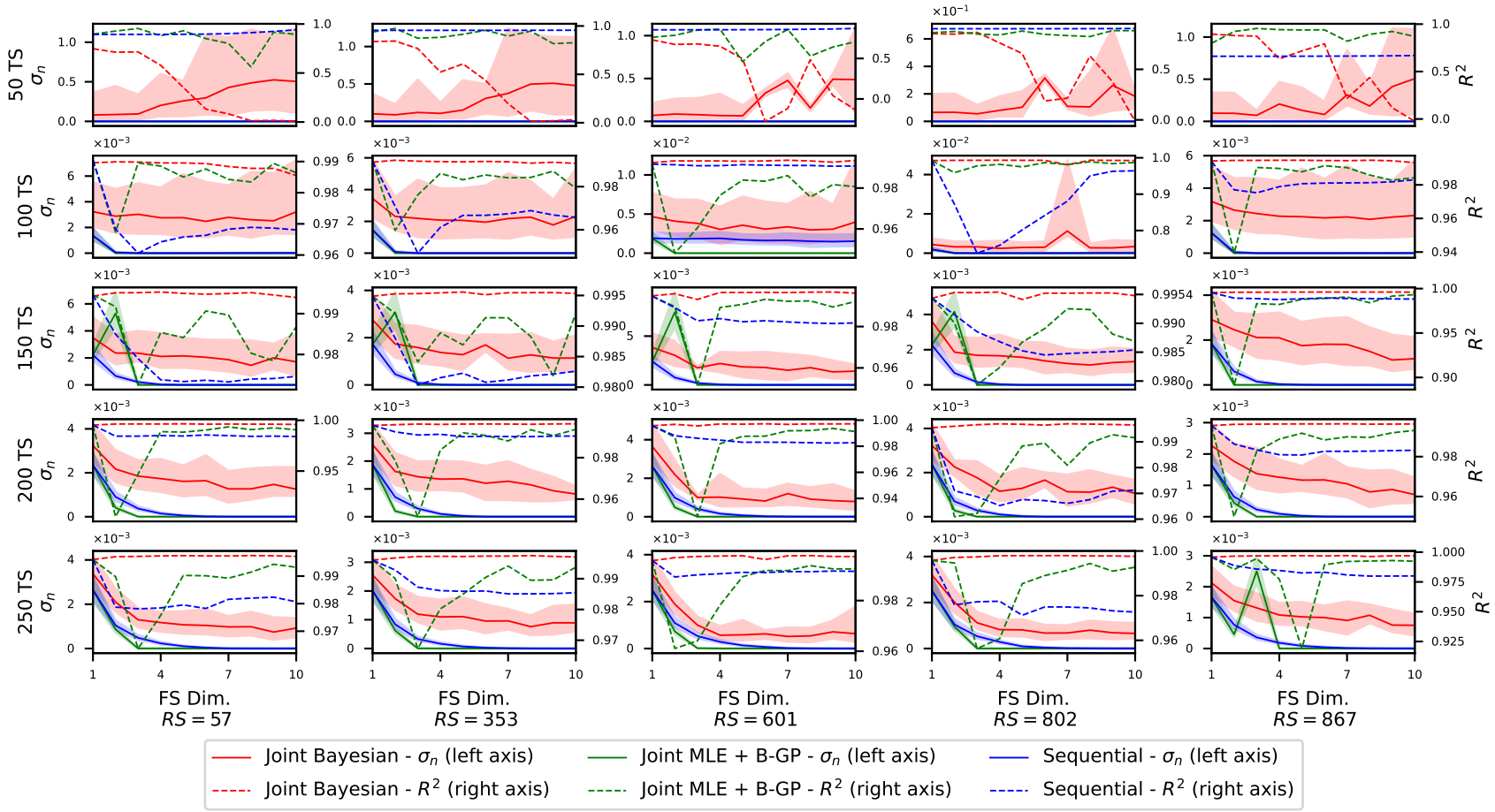


Figure B.30: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **subsonic CRM sideforce dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

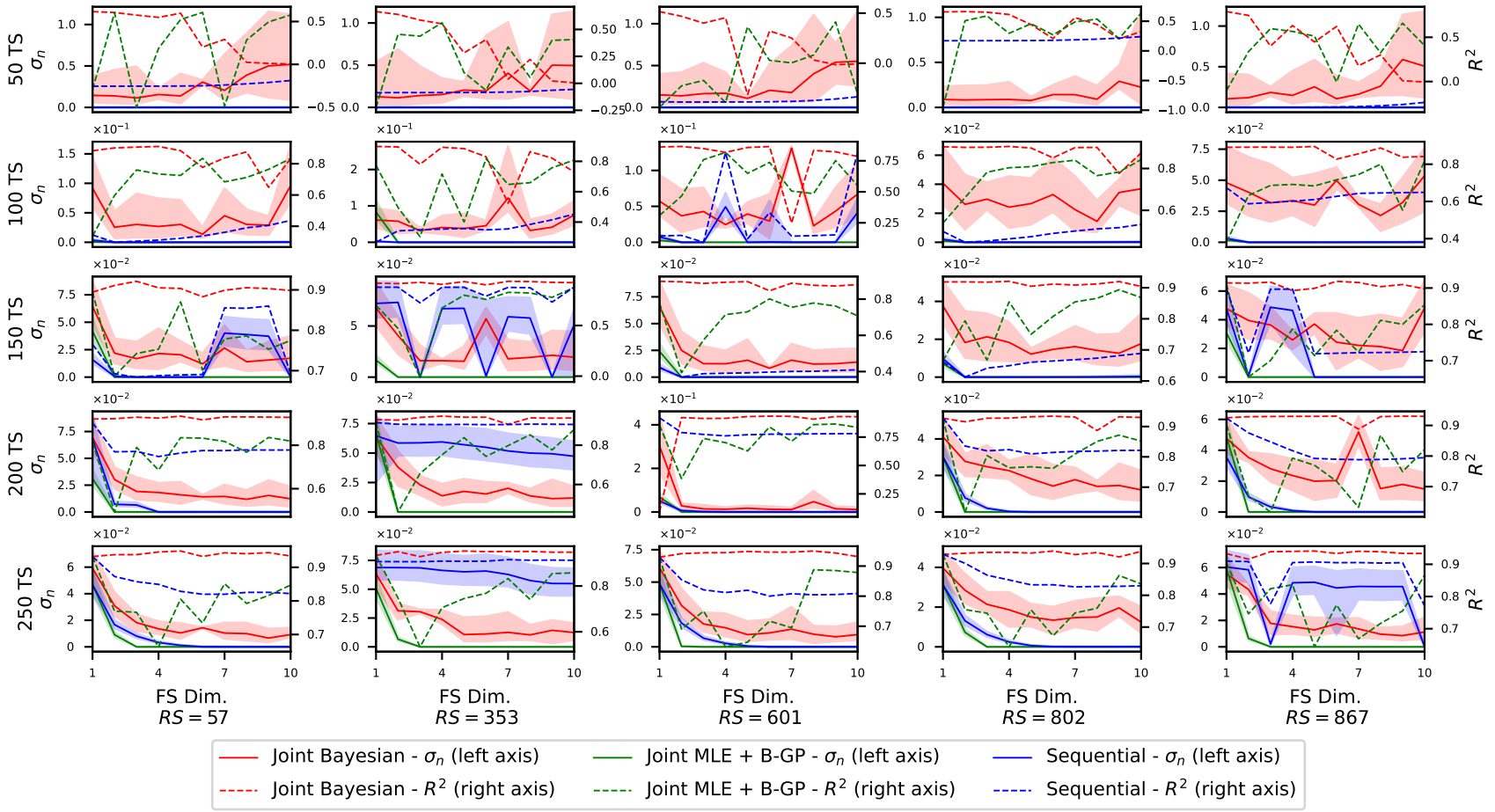


Figure B.31: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM drag dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

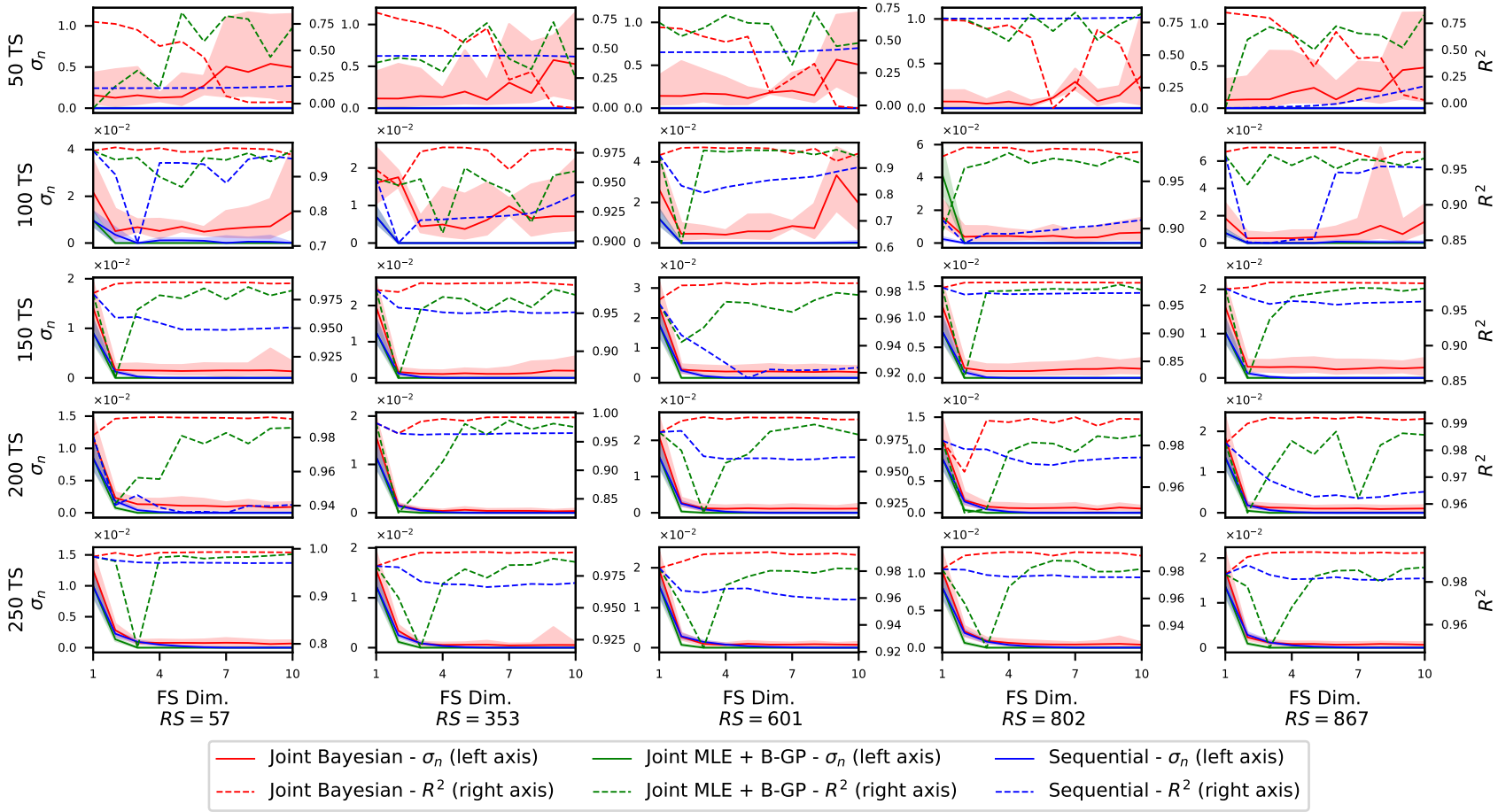


Figure B.32: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM lift dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

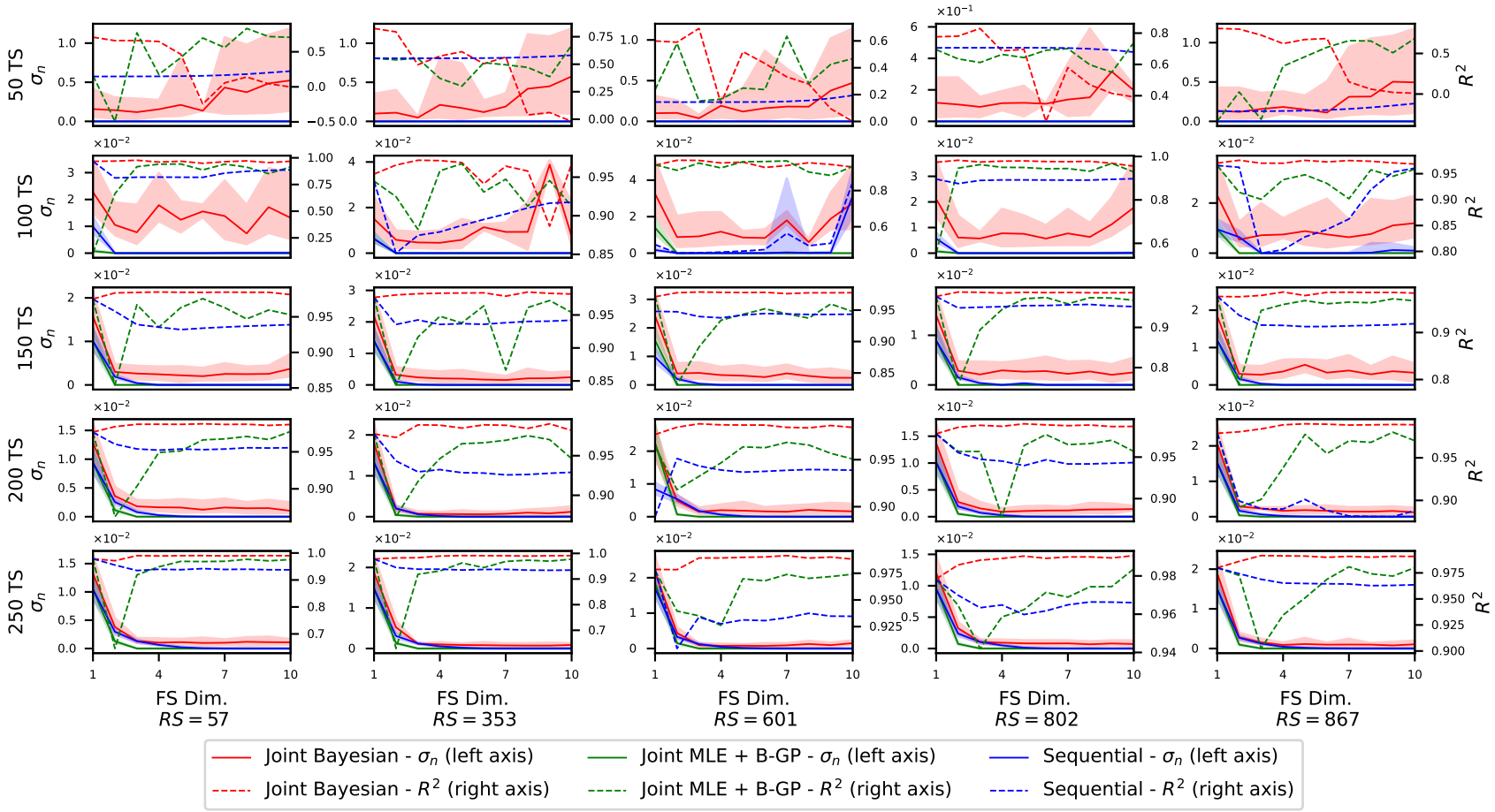


Figure B.33: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM x-moment dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

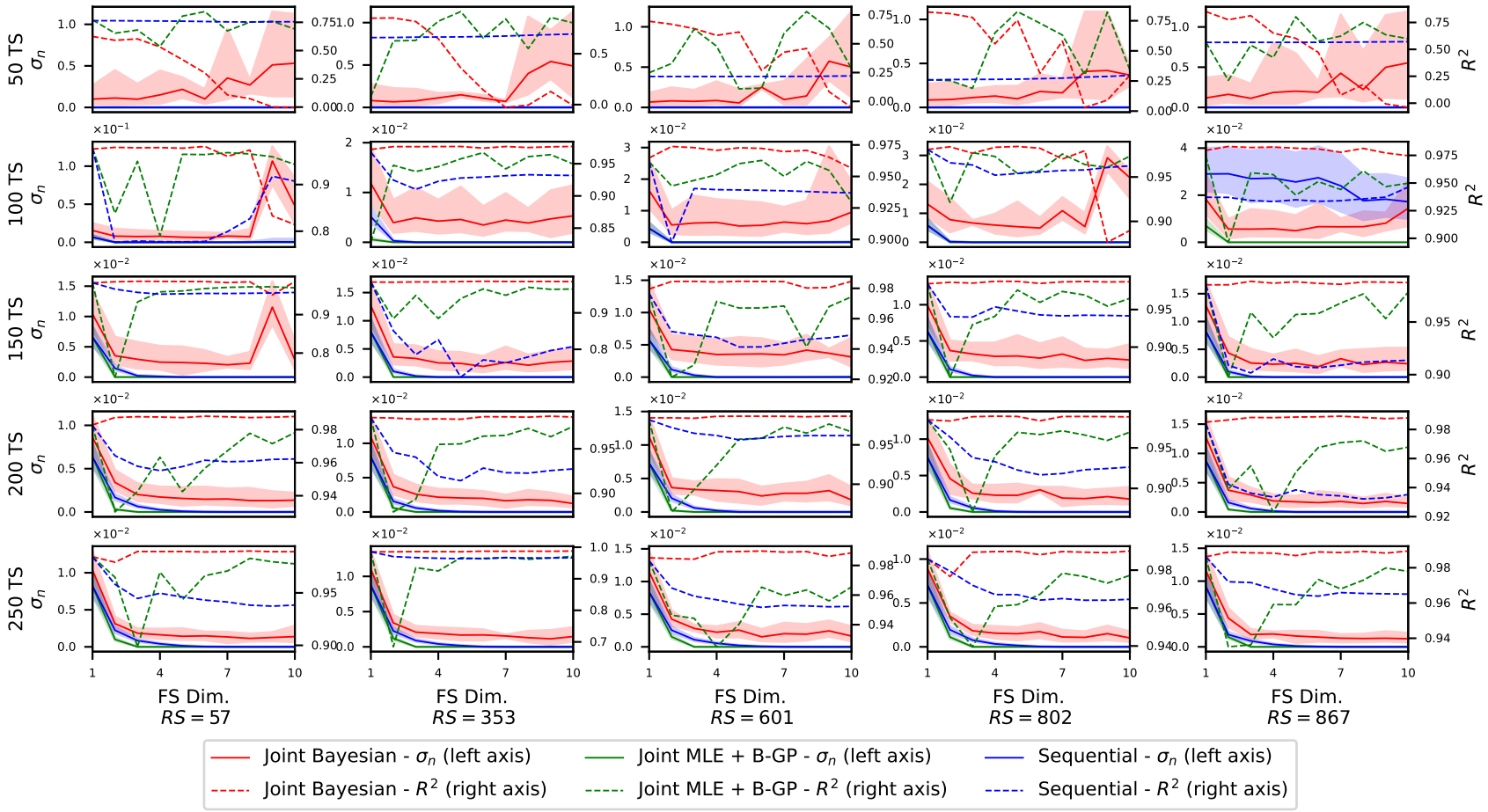


Figure B.34: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM y-moment dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

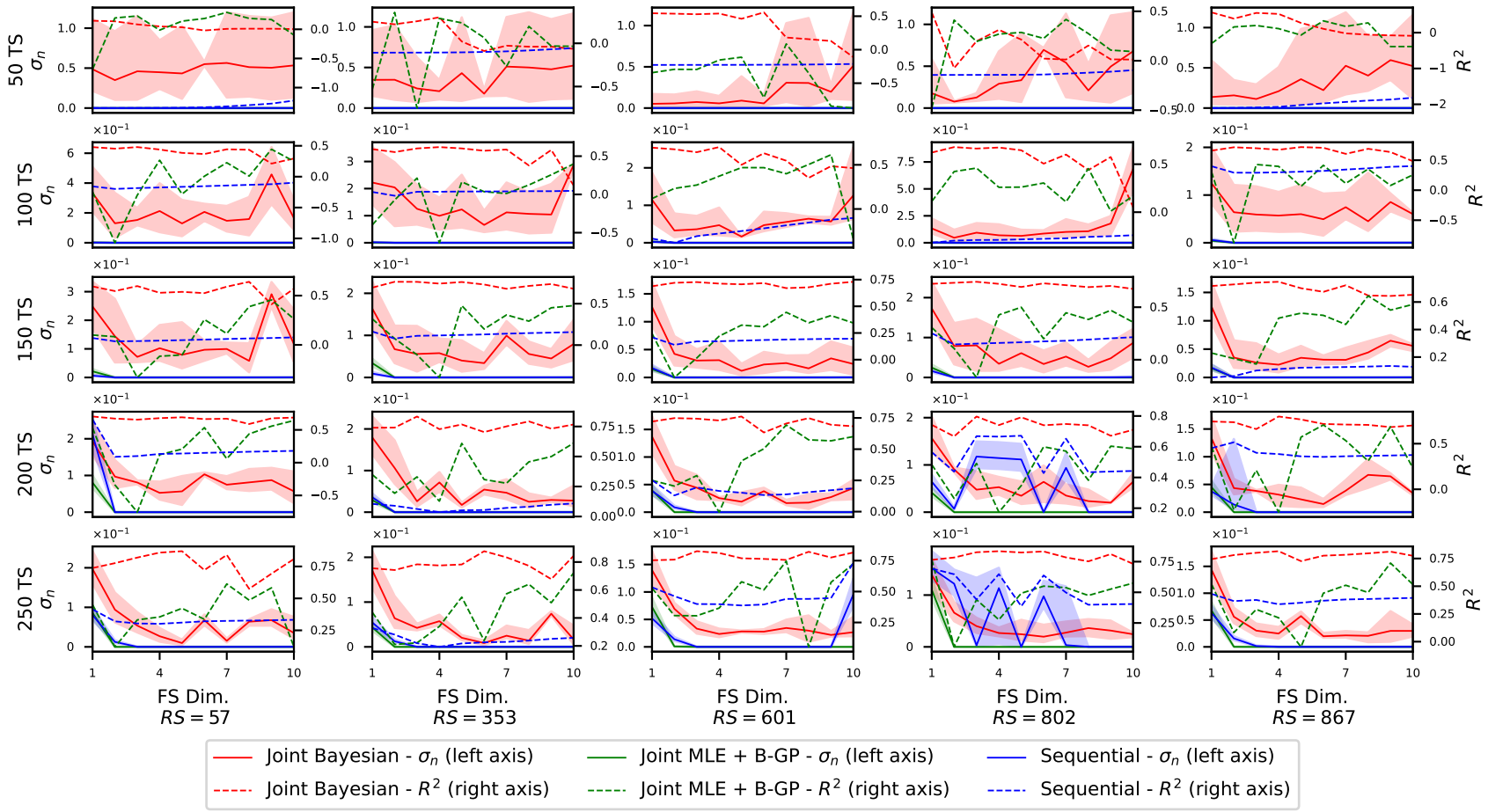


Figure B.35: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM z-moment dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

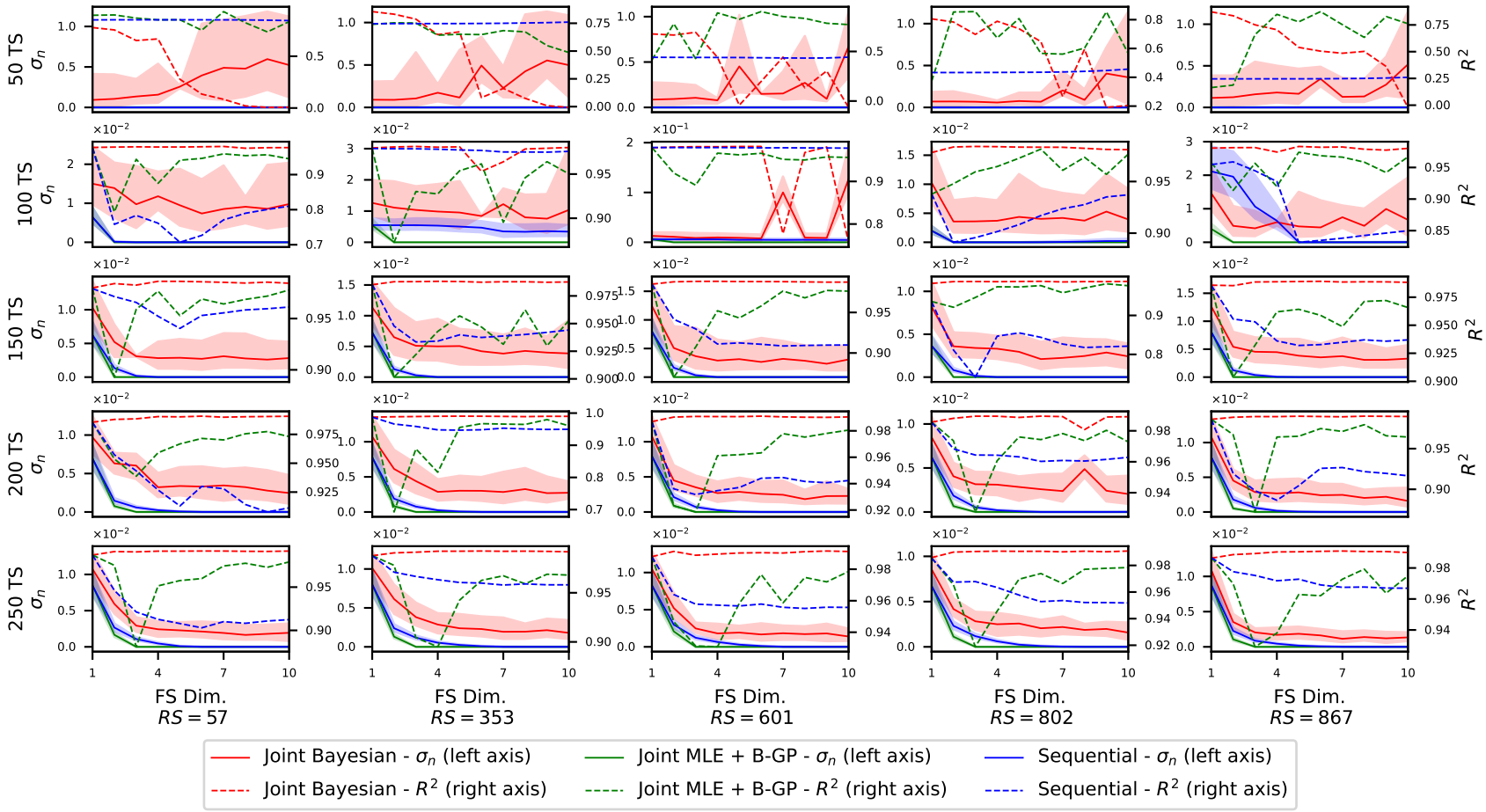


Figure B.36: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **transonic CRM sideforce dataset with 50 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

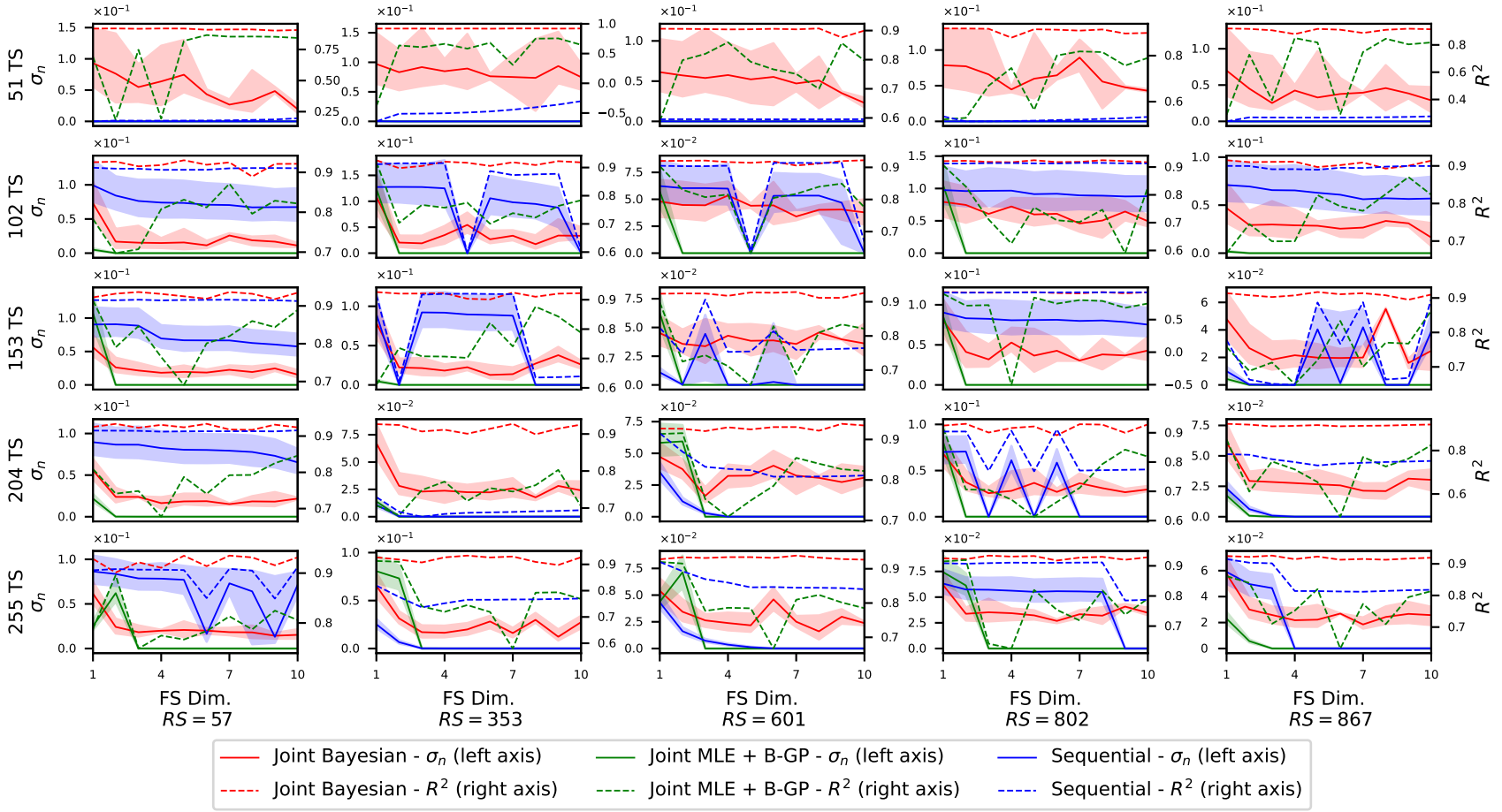


Figure B.37: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **RAE2822 lift dataset with 51 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

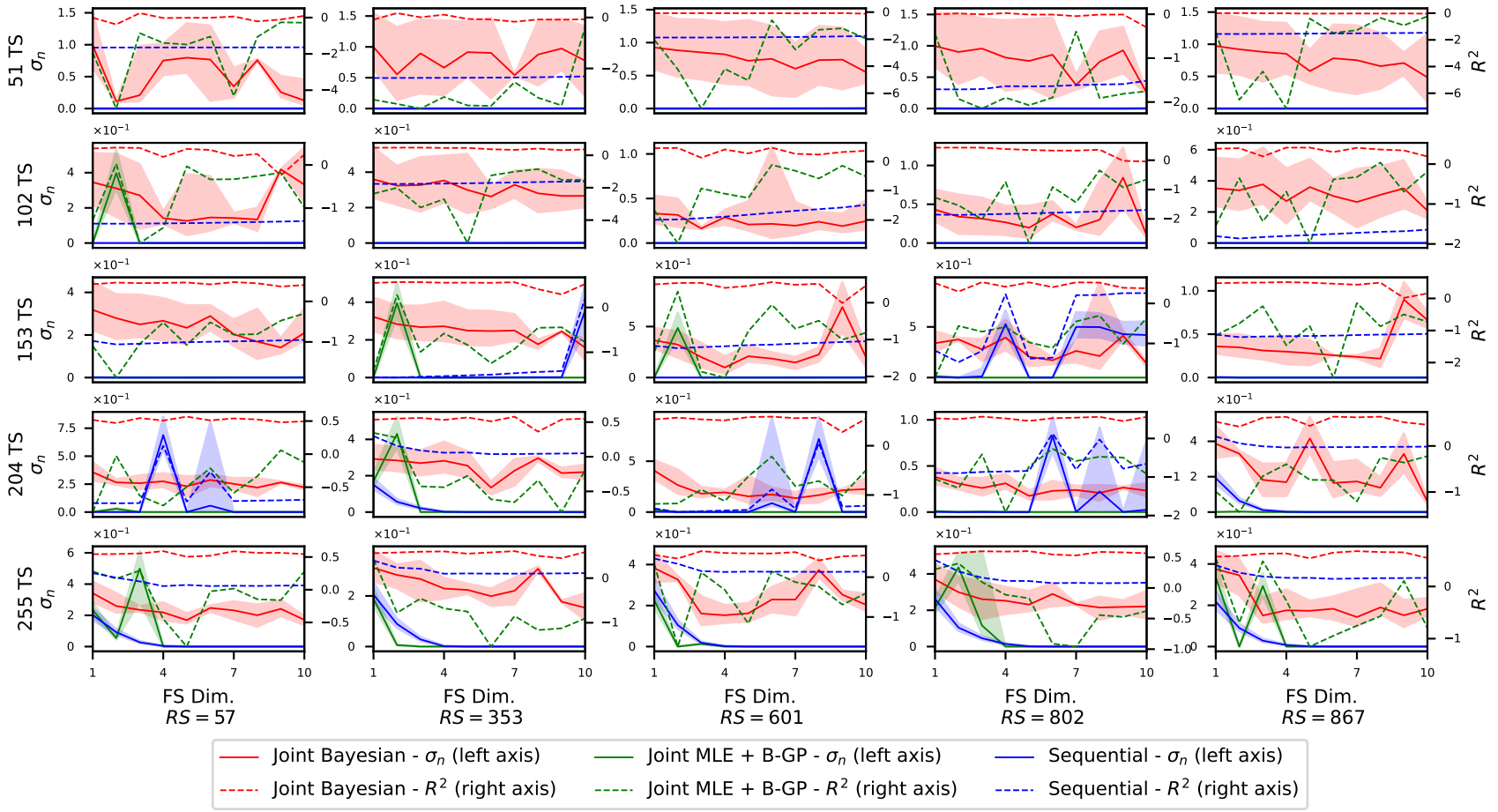


Figure B.38: Evolution of the noise variance (σ_n^2, blue, left) and the coefficient of determination (R^2, red, right) as a function of the number of feature space dimensions (FS Dim.) for the **RAE2822 z-moment dataset with 51 inputs**. The number of training samples (TS) increases by row from top to bottom. Each column represents a repetition where a different random seed (RS) is used to split training and validation points.

B.3.2 Comparison of the Training Durations

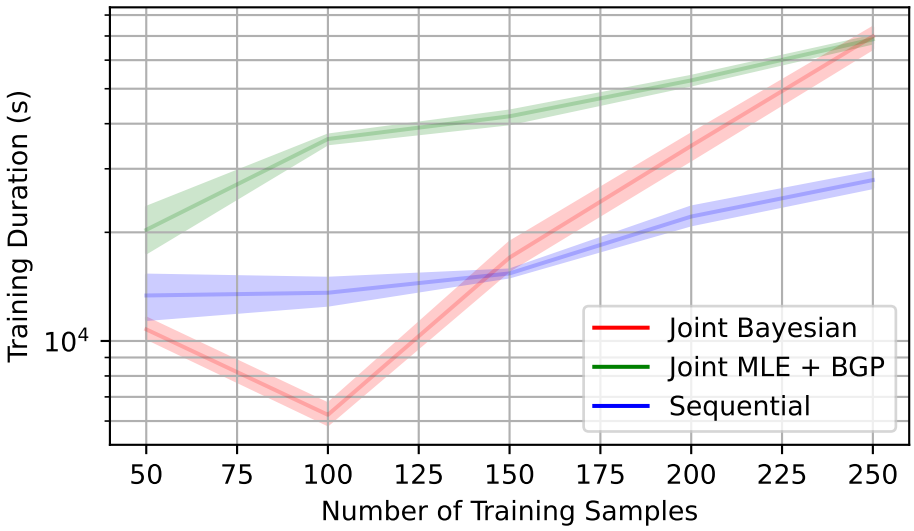


Figure B.39: Evolution of the training duration as a function of the number of training samples for the subsonic CRM lift dataset with 50 inputs.

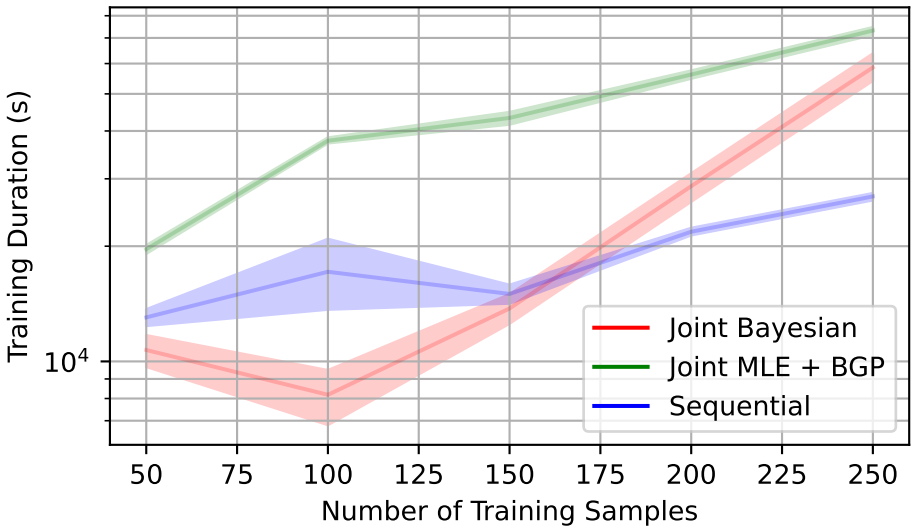


Figure B.40: Evolution of the training duration as a function of the number of training samples for the subsonic CRM x-moment dataset with 50 inputs.

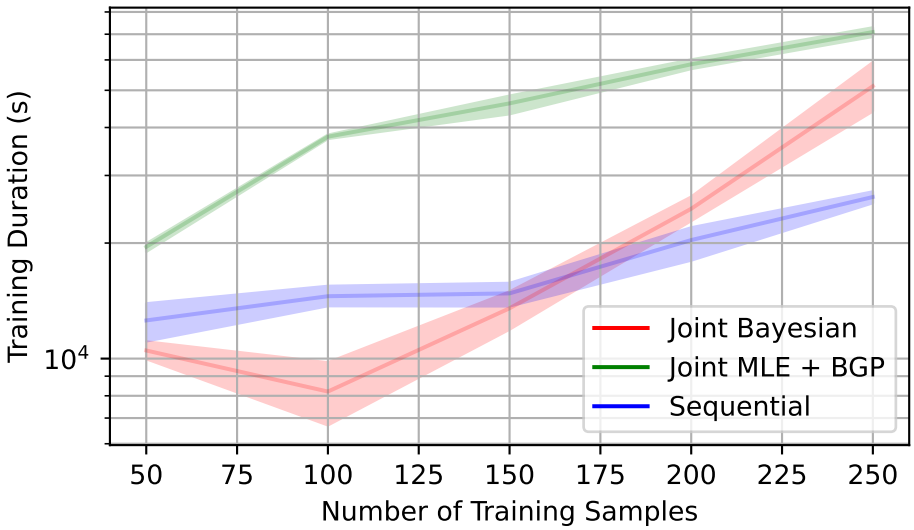


Figure B.41: Evolution of the training duration as a function of the number of training samples for the subsonic CRM y-moment dataset with 50 inputs.

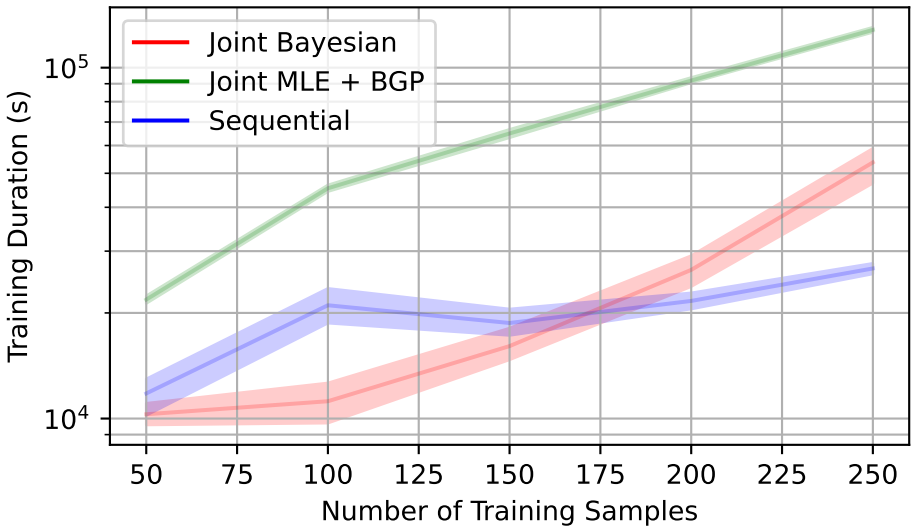


Figure B.42: Evolution of the training duration as a function of the number of training samples for the subsonic CRM sideforce dataset with 50 inputs.

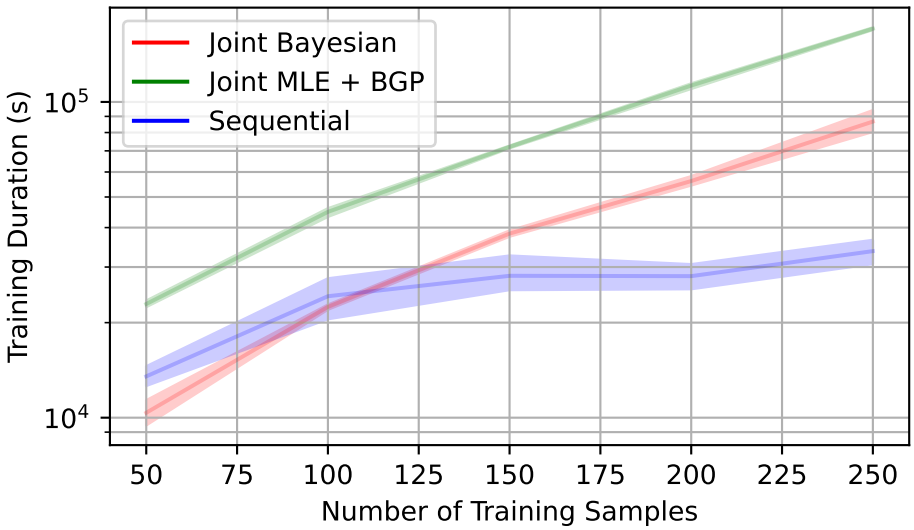


Figure B.43: Evolution of the training duration as a function of the number of training samples for the transonic CRM drag dataset with 50 inputs.

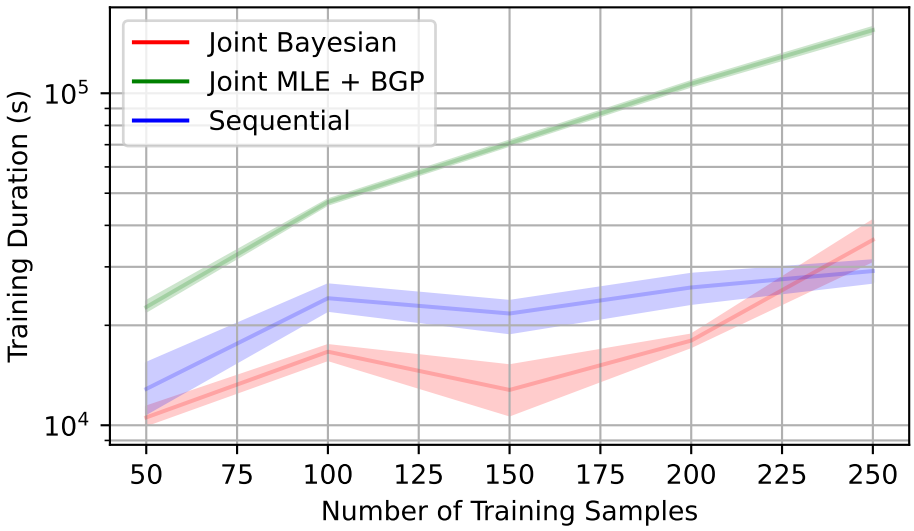


Figure B.44: Evolution of the training duration as a function of the number of training samples for the transonic CRM lift dataset with 50 inputs.

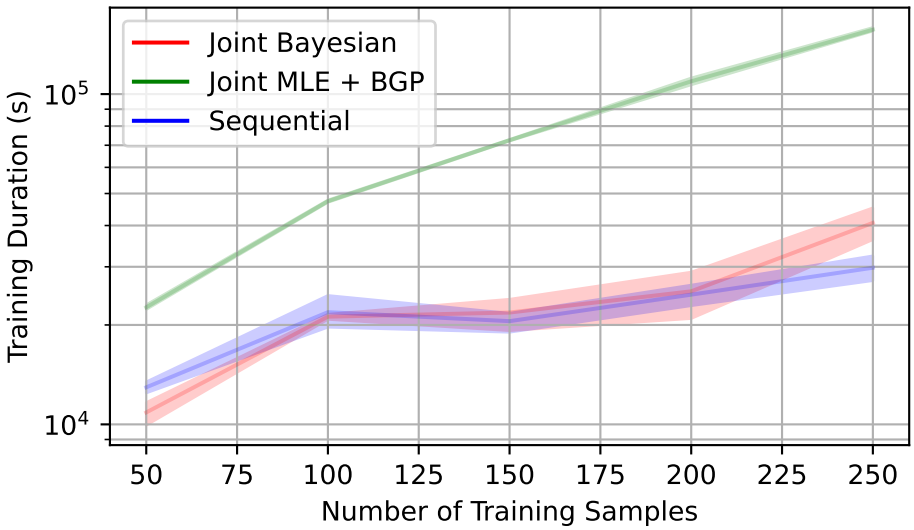


Figure B.45: Evolution of the training duration as a function of the number of training samples for the transonic CRM x-moment dataset with 50 inputs.

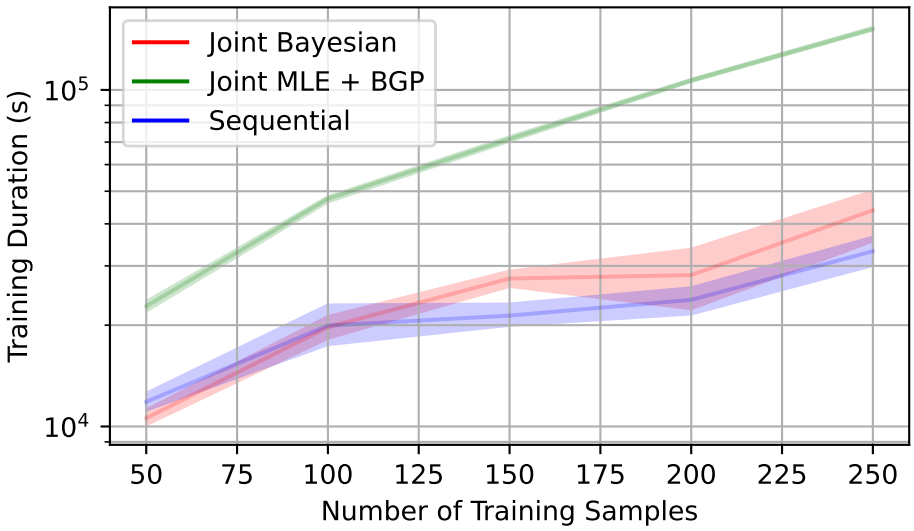


Figure B.46: Evolution of the training duration as a function of the number of training samples for the transonic CRM y-moment dataset with 50 inputs.

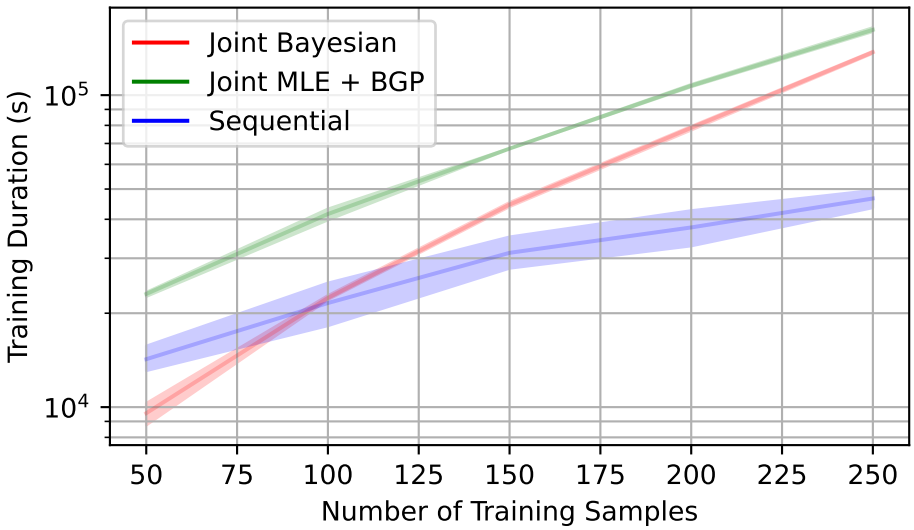


Figure B.47: Evolution of the training duration as a function of the number of training samples for the transonic CRM z-moment dataset with 50 inputs.

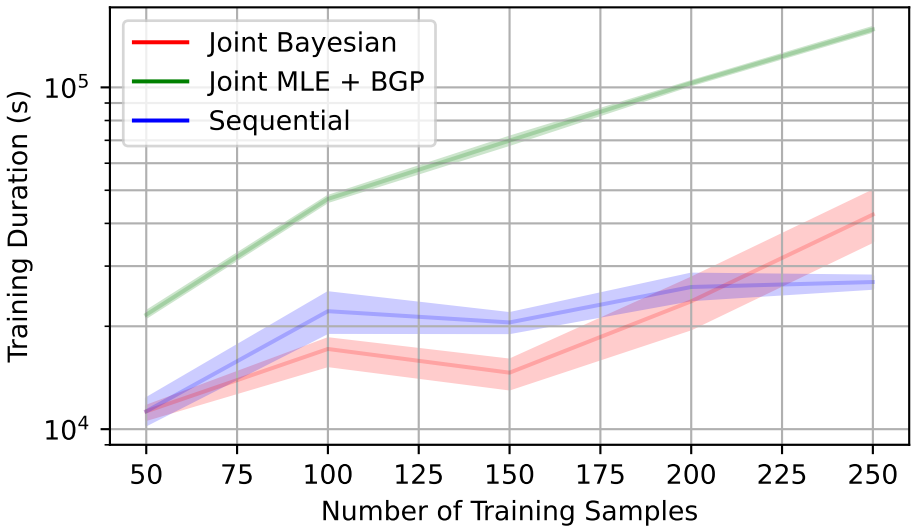


Figure B.48: Evolution of the training duration as a function of the number of training samples for the transonic CRM sideforce dataset with 50 inputs.

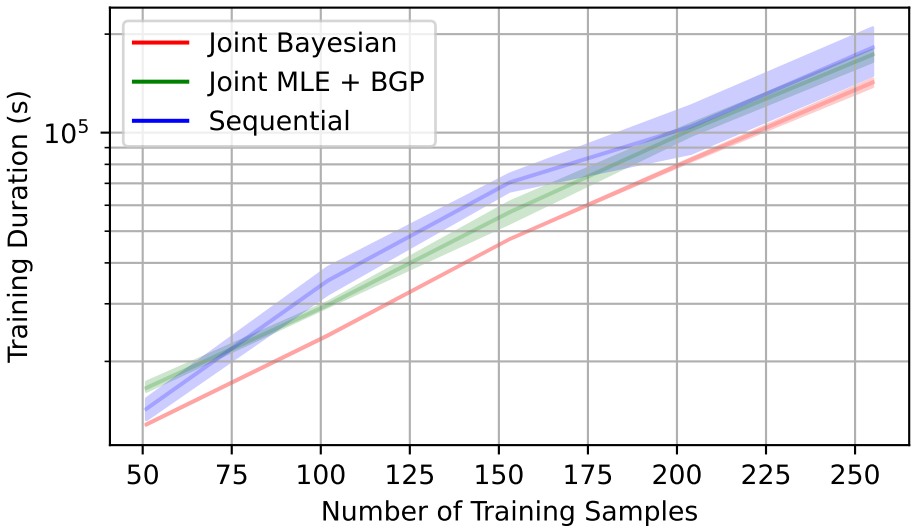


Figure B.49: Evolution of the training duration as a function of the number of training samples for the RAE2822 lift dataset with 51 inputs.

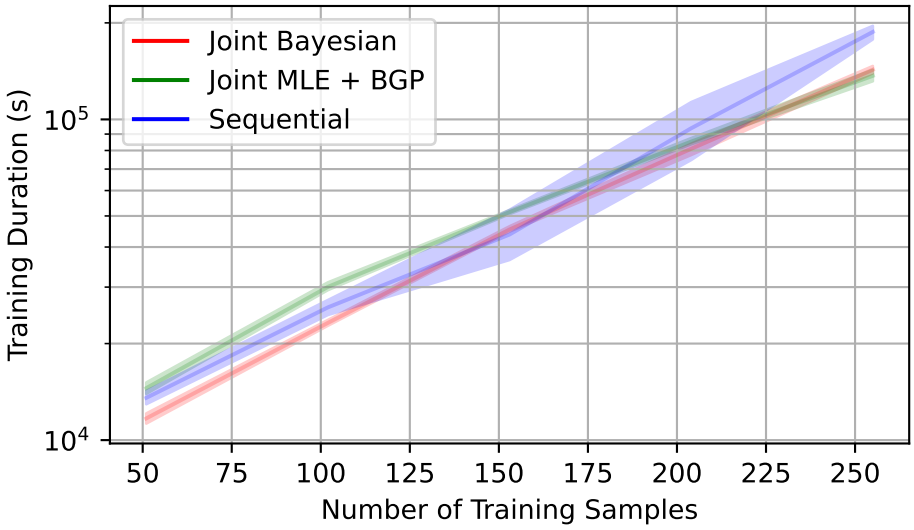


Figure B.50: Evolution of the training duration as a function of the number of training samples for the RAE2822 z-moment dataset with 51 inputs.

REFERENCES

[1] G. E. Dieter and L. C. Schmidt, *Engineering Design*, 5th ed. New York: McGraw-Hill, 2013, 1 p., ISBN: 978-0-07-339814-3.

[2] D. Raymer, *Aircraft Design: A Conceptual Approach 5e and RDSWin STUDENT*. Washington, DC: American Institute of Aeronautics and Astronautics, Inc., 2012, ISBN: 978-1-60086-921-1.

[3] J. Roskam, *Part I: Preliminary Sizing of Airplanes* (Airplane Design), 8 vols. DAR-corporation, 1997, vol. 1.

[4] J. D. Anderson, *Aircraft Performance and Design*. McGraw-Hill Science/Engineering/Math, 1999.

[5] D. H. Hodges and G. A. Pierce, *Introduction to Structural Dynamics and Aeroelasticity*. Cambridge: Cambridge University Press, 2011, ISBN: 978-0-511-99711-2.

[6] J. R. Martins and A. B. Lambe, "Multidisciplinary design optimization: A survey of architectures," *AIAA journal*, vol. 51, no. 9, pp. 2049–2075, 9 2013.

[7] R. T. Haftka, "Simultaneous analysis and design," *AIAA journal*, vol. 23, no. 7, pp. 1099–1103, 7 1985.

[8] J. Sobieszczanski-Sobieski, "Optimization by decomposition: A step from hierarchic to non-hierarchic systems," 1988.

[9] J. Sobieszczanski-Sobieski, J. S. Agte, and R. R. Sandusky, "Bilevel integrated system synthesis," *AIAA journal*, vol. 38, no. 1, pp. 164–172, 1 2000.

[10] E. J. Cramer, J. E. Dennis, P. D. Frank, R. M. Lewis, and G. R. Shubin, "Problem formulation for multidisciplinary optimization," *SIAM Journal on Optimization*, vol. 4, no. 4, pp. 754–776, 4 1994.

[11] R. D. Braun, "Collaborative optimization: An architecture for large-scale distributed design.," 1997.

[12] H. M. Kim, "Target cascading in optimal system design," Ph.D. dissertation, University of Michigan Ann Arbor, MI, 2001.

[13] D. N. Mavris and D. A. DeLaurentis, "Methodology for examining the simultaneous impact of requirements, vehicle characteristics, and technologies on military aircraft design," 2000.

[14] R. Ghanem, *Handbook of Uncertainty Quantification*. New York, NY: Springer Berlin Heidelberg, 2017, ISBN: 978-3-319-12384-4.

[15] T. Sullivan, *Introduction to Uncertainty Quantification*. New York, NY: Springer Science+Business Media, 2015, ISBN: 978-3-319-23394-9.

[16] R. C. Smith, *Uncertainty Quantification: Theory, Implementation, and Applications*. Siam, 2013, vol. 12.

[17] T. J. Santner, B. J. Williams, and W. I. Notz, *The Design and Analysis of Computer Experiments*. New York, NY: Springer Science+Business Media, LLC, 2018, ISBN: 978-1-4939-8845-7.

[18] A. Keane and P. Nair, *Computational Approaches for Aerospace Design: The Pursuit of Excellence*. John Wiley & Sons, 2005.

[19] A. Forrester, A. Sobester, and A. Keane, *Engineering Design via Surrogate Modelling: A Practical Guide*. John Wiley & Sons, 2008.

[20] W. Johnson and C. Silva, "Observations from exploration of vtol urban air mobility designs," 2018.

[21] M. Eldred and D. Dunlavy, "Formulations for surrogate-based optimization with data fit, multifidelity, and reduced-order models," in *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2006, p. 7117.

[22] P. Benner, S. Gugercin, and K. Willcox, "A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems," *SIAM Review*, vol. 57, no. 4, pp. 483–531, 4 Jan. 2015.

[23] R. Yondo, E. Andrés, and E. Valero, "A review on design of experiments and surrogate models in aircraft real-time and many-query aerodynamic analyses," *Progress in Aerospace Sciences*, vol. 96, pp. 23–61, Jan. 2018.

[24] J. Friedman, T. Hastie, and R. Tibshirani, *The Elements of Statistical Learning*. Springer series in statistics New York, NY, USA: 2001, vol. 1.

[25] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT press, 2016.

[26] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 7553 2015.

[27] S. Shan and G. G. Wang, "Metamodeling for High Dimensional Simulation-Based Design Problems," *Journal of Mechanical Design*, vol. 132, no. 5, pp. 051009-051009–11, 5 May 17, 2010.

[28] T. Bui-Thanh, M. Damodaran, and K. E. Willcox, "Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition," *AIAA journal*, vol. 42, no. 8, pp. 1505–1516, 8 2004.

[29] P. Benner and H. Faßbender, "Model Order Reduction: Techniques and Tools," in *Encyclopedia of Systems and Control*, J. Baillieul and T. Samad, Eds., London: Springer London, 2013, pp. 1–10, ISBN: 978-1-4471-5102-9.

[30] M. Frangos, Y. Marzouk, K. Willcox, and B. van Bloemen Waanders, "Surrogate and reduced-order modeling: A comparison of approaches for large-scale statistical inverse problems [Chapter 7]," 2010.

[31] C. Audouze, F. De Vuyst, and P. B. Nair, "Reduced-order modeling of parameterized PDEs using time–space-parameter principal component analysis," *International journal for numerical methods in engineering*, vol. 80, no. 8, pp. 1025–1057, 8 2009.

[32] D. Xiao, F. Fang, C. C. Pain, I. M. Navon, P. Salinas, and A. Muggeridge, "Non-intrusive model reduction for a 3D unstructured mesh control volume finite element reservoir model and its application to fluvial channels," p. 25, 2016.

[33] B. Peherstorfer, T. Cui, Y. Marzouk, and K. Willcox, "Multifidelity Importance Sampling," *Computer Methods in Applied Mechanics and Engineering*, vol. 300, pp. 490–509, 2016.

[34] B. Peherstorfer, K. Willcox, and M. Gunzburger, "Survey of multifidelity methods in uncertainty propagation, inference, and optimization," *SIAM Review*, vol. 60, no. 3, pp. 550–591, 3 2018.

[35] A. I. Forrester, A. Sóbester, and A. J. Keane, "Multi-fidelity optimization via surrogate modelling," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 463, no. 2088, pp. 3251–3269, 2088 Dec. 8, 2007.

[36] M. G. Fernández-Godino, C. Park, N.-H. Kim, and R. T. Haftka. "Review of multi-fidelity models." arXiv: 1609.07196. (2016).

[37] M. Kennedy, "Predicting the output from a complex computer code when fast approximations are available," *Biometrika*, vol. 87, no. 1, pp. 1–13, 1 Mar. 1, 2000.

[38] H. Liu, Y.-S. Ong, and J. Cai, "A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design," *Structural and Multidisciplinary Optimization*, vol. 57, no. 1, pp. 393–416, 1 2018.

[39] V. V. Fedorov, *Theory of Optimal Experiments*. Elsevier, 2013.

[40] A. Atkinson, A. Donev, and R. Tobias, *Optimum Experimental Designs, with SAS*. Oxford University Press, 2007, vol. 34.

[41] G. E. Box and N. R. Draper, *Empirical Model-Building and Response Surfaces*. John Wiley & Sons, 1987.

[42] R. A. Fisher, "The design of experiments.," *The design of experiments.*, no. 7th Ed, 7th Ed 1960.

[43] A. D. Kiureghian and O. Ditlevsen, "Aleatory or epistemic? Does it matter?" *Structural Safety*, vol. 31, no. 2, pp. 105–112, 2 2009.

[44] A. Sauer, R. B. Gramacy, and D. Higdon. "Active Learning for Deep Gaussian Process Surrogates." arXiv: 2012.08015 [stat]. (Aug. 26, 2021), (visited on 01/23/2022).

[45] A. Der Kiureghian and O. Ditlevsen, "Aleatory or epistemic? Does it matter?" *Structural Safety*, vol. 31, no. 2, pp. 105–112, 2 2009.

[46] R. E. Bellman, *Dynamic Programming*. 1957.

[47] D. L. Donoho, "High-Dimensional Data Analysis: The Curses and Blessings of Dimensionality," p. 33, 2000.

[48] C. M. Bishop and N. M. Nasrabadi, *Pattern Recognition and Machine Learning*. Springer, 2006, vol. 4.

[49] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, UNITED STATES: MIT Press, 2012, ISBN: 978-0-262-30524-2.

[50] P. Perdikaris, D. Venturi, and G. E. Karniadakis, "Multifidelity information fusion algorithms for high-dimensional systems and massive data sets," *SIAM Journal on Scientific Computing*, vol. 38, no. 4, B521–B538, 4 2016.

[51] M. C. Kennedy and A. O'Hagan, "Bayesian calibration of computer models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 3, pp. 425–464, 3 2001.

[52] I. T. Jolliffe, "Principal component analysis. Encyclopedia of statistics in behavioral science," in *Encyclopedia of Statistics in Behavioral Science*, John Wiley & Sons, Ltd, 2005, p. 518, ISBN: 0-387-95442-2.

[53] Y. Zhu and N. Zabaras, "Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification," *Journal of Computational Physics*, vol. 366, pp. 415–447, 2018.

[54] F. Dietrich, F. Künzner, T. Neckel, G. Köster, and H. J. Bungartz, "Fast and flexible uncertainty quantification through a data-driven surrogate model," *International Journal for Uncertainty Quantification*, vol. 8, no. 2, pp. 175–192, 2 2018.

[55] D. J. Lucia, P. S. Beran, and W. A. Silva, "Reduced-order modeling: New approaches for computational physics," *Progress in aerospace sciences*, vol. 40, no. 1-2, pp. 51–117, 1-2 2004.

[56] E. Dowell, K. Hall, J. Thomas, R. Florea, B. Epureanu, and J. Heeg, "Reduced order models in unsteady aerodynamics," in *40th Structures, Structural Dynamics, and Materials Conference and Exhibit*, 1999, p. 1261.

[57] K. Hall, J. Thomas, and E. Dowell, "Reduced-order modelling of unsteady small-disturbance flows using a frequency-domain proper orthogonal decomposition technique," in *37th Aerospace Sciences Meeting and Exhibit*, 1999, p. 655.

[58] M. Romanowski, "Reduced order unsteady aerodynamic and aeroelastic models using Karhunen-Loeve eigenmodes," in *6th Symposium on Multidisciplinary Analysis and Optimization*, 1996, p. 3981.

[59] SAS Institute Inc., "JMP ® 15 Predictive and Specialized Modeling," in Cary, NC: SAS Institute Inc., 2019.

[60] F. Theurich and K. Becker, "Introduction to Airbus Use-Case "FlexCraft"," in *Symposium on AeroStructures*, Springer, 2015, pp. 79–84.

[61] D. W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization*. Somerset, UNITED STATES: John Wiley & Sons, Incorporated, 2015, ISBN: 978-1-118-57548-2.

[62] C. Lataniotis, S. Marelli, and B. Sudret, "Extending classical surrogate modeling to high dimensions through supervised dimensionality reduction: A data-driven approach," *International Journal for Uncertainty Quantification*, vol. 10, no. 1, pp. 55–82, 1 2020.

[63] S. Shan and G. G. Wang, "Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions," *Structural and Multidisciplinary Optimization*, vol. 41, no. 2, pp. 219–241, 2 2010.

[64] M. Kubicek, E. Minisci, and M. Cisternino, "High dimensional sensitivity analysis using surrogate modeling and high dimensional model representation," *International Journal for Uncertainty Quantification*, vol. 5, no. 5, pp. 393–414, 5 2015.

[65] S. Keiper, "Approximation of generalized ridge functions in high dimensions," *Journal of Approximation Theory*, vol. 245, pp. 101–129, 2019.

[66] B. Doerr and S. Mayer. "The recovery of ridge functions on the hypercube suffers from the curse of dimensionality." arXiv: 1903.10223. (Mar. 2019).

[67] D. Oglic, "Constructive approximation and learning by greedy algorithms," Universit{\"a}ts- und Landesbibliothek Bonn, 2018.

[68] A. Glaws and P. G. Constantine. "A Lanczos-Stieltjes method for one-dimensional ridge function approximation and integration." arXiv: 1808.02095. (2018).

[69] A. Kolleck, "On some aspects of recovery of sparse signals in high dimensions from nonlinear measurements using compressed sensing," Technischen Universität Berlin, 2017.

[70] A. Pinkus, "Approximating by ridge functions," *Surface fitting and multiresolution methods*, pp. 279–292, 1997.

[71] R. DeVore and G. Lorentz, *Constructive Approximation*. Springer Science \& Business Media, 1993, p. 303.

[72] H. Tyagi and V. Cevher, "Learning ridge functions with randomized sampling in high dimensions," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, IEEE, 2012, pp. 2025–2028, ISBN: 978-1-4673-0046-9.

[73] M. Fornasier, K. Schnass, and J. Vybiral, "Learning functions of few arbitrary linear parameters in high dimensions," *Foundations of Computational Mathematics*, vol. 12, no. 2, pp. 229–262, 2 Apr. 2012.

[74] A. Cohen, I. Daubechies, R. DeVore, G. Kerkyacharian, and D. Picard, "Capturing Ridge Functions in High Dimensions from Point Queries," *Constructive Approximation*, vol. 35, no. 2, pp. 225–243, 2 Apr. 2012.

[75] K. Schnass and J. Vybíral, "Compressed learning of high-dimensional sparse functions," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, IEEE, 2011, pp. 3924–3927, ISBN: 978-1-4577-0539-7.

[76] B. Li, *Sufficient Dimension Reduction: Methods and Applications with r*. Chapman and Hall/CRC, 2018, ISBN: 978-1-315-11942-7.

[77] C. J. Burges, "Dimension reduction: A guided tour," *Foundations and Trends in Machine Learning*, vol. 2, no. 4, pp. 275–365, 4 Aug. 2009.

[78] K. Fukumizu, F. R. Bach, and M. I. Jordan, "Kernel dimension reduction in regression," *Annals of Statistics*, vol. 37, no. 4, pp. 1871–1905, 4 Aug. 2009.

[79] K. P. Adragni and R. D. Cook, "Sufficient dimension reduction and prediction in regression," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 367, no. 1906, pp. 4385–4405, 1906 Nov. 2009.

[80] B. Li and S. Wang, "On directional regression for dimension reduction," *Journal of the American Statistical Association*, vol. 102, no. 479, pp. 997–1008, 479 Sep. 2007.

[81] M. Borga, T. Landelius, and H. Knutsson, "A unified approach to pca, pls, mlr and cca," *Technical Report*, LiTH-ISY-R–1992, 1992.

[82] T. D. Bie, N. Cristianini, and R. Rosipal, "Eigenproblems in pattern recognition," in *Handbook of Geometric Computing*, Springer, 2005, pp. 129–167.

[83] R. Rosipal and N. Krämer, "Overview and Recent Advances in Partial Least Squares," in *Subspace, Latent Structure and Feature Selection, Saunders, C., et al. (Eds.) (Heidelberg: Springer-Verlag, 2006)*, vol. 3940, 2006, pp. 34–51, ISBN: 9783540341376. pmid: 238094700002.

[84] R. Rosipal, "Nonlinear partial least squares: An overview," *Chemoinformatics and Advanced Machine Learning Perspectives: Complex Computational Methods and Collaborative Techniques*, pp. 169–189, 2011.

[85] R. Rosipal, "Kernel partial least squares for nonlinear regression and discrimination," *Neural Network World*, vol. 13, no. 4, p. 449, 4 2003.

[86] H. Abdi, "Partial Least Squares (PLS) Regression," *Encyclopedia for research methods for the social sciences*, pp. 792–795, 2003. pmid: 20539106.

[87] M. Vivien, "Approches PLS linéaires et non linéaires pour la modélisation de multitableaux. Théorie et applications," Université Montpellier 1, 2002, p. 313.

[88] K. S. Ng, "A simple explanation of partial least squares," *The Australian National University, Canberra*, pp. 1–10, 2013.

[89] S. Yu, K. Yu, V. Tresp, H. P. Kriegel, and M. Wu, "Supervised probabilistic principal component analysis," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '06, vol. 2006, New York, NY, USA: ACM, 2006, pp. 464–473, ISBN: 1-59593-339-5.

[90] G. Chao, Y. Luo, and W. Ding, "Recent Advances in Supervised Dimension Reduction: A Survey," *Machine Learning and Knowledge Extraction*, vol. 1, no. 1, pp. 341–358, 1 Jan. 7, 2019.

[91] R. Tripathy, I. Bilionis, and M. Gonzalez, "Gaussian processes with built-in dimensionality reduction: Applications to high-dimensional uncertainty propagation," *Journal of Computational Physics*, vol. 321, pp. 191–223, Sep. 2016.

[92] P. Tsilifis and R. G. Ghanem, "Bayesian adaptation of chaos representations using variational inference and sampling on geodesics," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 474, no. 2217, 2217 Sep. 2018.

[93] I. Guyon and A. Elisseefl, "An introduction to feature extraction," *Studies in Fuzziness and Soft Computing*, vol. 207, pp. 1–25, 2006.

[94] R. DeVore, G. Petrova, and P. Wojtaszczyk, "Approximation of functions of few variables in high dimensions," *Constructive Approximation*, vol. 33, no. 1, pp. 125–143, 1 2011.

[95] Y. Ma and L. Zhu, "A review on dimension reduction," *International Statistical Review*, vol. 81, no. 1, pp. 134–150, 1 2013.

[96] P. Chen and O. Ghattas, "Hessian-based sampling for high-dimensional model reduction," *International Journal for Uncertainty Quantification*, vol. 9, no. 2, pp. 103–121, 2 2019.

[97] P. Constantine, Q. Wang, A. Doostan, and G. Iaccarino, "A surrogate accelerated Bayesian inverse analysis of the HyShot II flight data," in *52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 19th AIAA/ASME/AHS Adaptive Structures Conference 13t*, 2011, p. 2037.

[98] S. H. Berguin and D. N. Mavris, "Dimensionality reduction in aerodynamic design using principal component analysis with gradient information," in *10th AIAA Multidisciplinary Design Optimization Conference*, 2014, p. 0112.

[99] S. H. Berguin, D. Rancourt, and D. N. Mavris, "Method to facilitate high-dimensional design space exploration using computationally expensive analyses," *AIAA Journal*, vol. 53, no. 12, pp. 3752–3765, 12 Dec. 2015.

[100] S. H. Berguin, "A method for reducing dimensionality in large design problems with computationally expensive analyses," Georgia Institute of Technology, 2015.

[101] P. G. Constantine, A. Eftekhari, J. Hokanson, and R. A. Ward, "A near-stationary subspace for ridge approximation," *Computer Methods in Applied Mechanics and Engineering*, vol. 326, pp. 402–421, Nov. 2017.

[102] M. Stoyanov and C. G. Webster, "A gradient-based sampling approach for dimension reduction of partial differential equations with stochasticcoefficients," *International Journal for Uncertainty Quantification*, vol. 5, no. 1, pp. 49–72, 1 2015.

[103] P. G. Constantine, M. Emory, J. Larsson, and G. Iaccarino, "Exploiting active subspaces to quantify uncertainty in the numerical simulation of the HyShot II scramjet," *Journal of Computational Physics*, vol. 302, pp. 1–20, Dec. 2015.

[104] P. G. Constantine, C. Kent, and T. Bui-Thanh, "Accelerating markov chain monte carlo with active subspaces," *SIAM Journal on Scientific Computing*, vol. 38, no. 5, A2779–A2805, 5 2016.

[105] M. Tezzele, F. Salmoiraghi, A. Mola, and G. Rozza, "Dimension reduction in heterogeneous parametric spaces with application to naval engineering shape design problems," *Advanced Modeling and Simulation in Engineering Sciences*, vol. 5, no. 1, 1 2018.

[106] J. C. Gross, P. Seshadri, and G. Parks, "Optimisation with intrinsic dimension reduction: A ridge informed trust-region method," in *AIAA Scitech 2020 Forum*, 2020, pp. 1–21.

[107] N. Demo, M. Tezzele, and G. Rozza. "A supervised learning approach involving active subspaces for an efficient genetic algorithm in high-dimensional optimization problems." arXiv: 2006.07282. (2020).

[108] O. Zahm, P. G. Constantine, C. Prieur, and Y. M. Marzouk, "Gradient-based dimension reduction of multivariate vector-valued functions," *SIAM Journal on Scientific Computing*, vol. 42, no. 1, A534–A558, 1 2020.

[109] R. R. Lam, O. Zahm, Y. M. Marzouk, and K. E. Willcox, "Multifidelity dimension reduction via active subspaces," *SIAM Journal on Scientific Computing*, vol. 42, no. 2, A929–A956, 2 2020.

[110] P. G. Constantine, *Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies*. SIAM, 2015, vol. 2.

[111] P. G. Constantine, E. Dow, and Q. Wang, "Active subspace methods in theory and practice: Applications to kriging surfaces," *SIAM Journal on Scientific Computing*, vol. 36, no. 4, A1500–A1524, 4 2014.

[112] D. Rajaram, R. H. Gautier, C. Perron, O. J. Pinon-Fischer, and D. Mavris, "Non-Intrusive Parametric Reduced Order Models with High-Dimensional Inputs via Gradient-Free Active Subspace," in *AIAA AVIATION 2020 FORUM*, VIRTUAL EVENT: American Institute of Aeronautics and Astronautics, Jun. 15, 2020, ISBN: 978-1-62410-598-2.

[113] T. M. Russi, "Uncertainty quantification with experimental data and complex system models," UC Berkeley, 2010, p. 176.

[114] P. Seshadri, S. Yuchi, and G. T. Parks, "Dimension reduction via Gaussian ridge functions," *SIAM-ASA Journal on Uncertainty Quantification*, vol. 7, no. 4, pp. 1301–1322, 4 Feb. 2019.

[115] D. Rajaram, C. Perron, T. G. Puranik, and D. N. Mavris, "Randomized Algorithms for Non-Intrusive Parametric Reduced Order Modeling," *AIAA Journal*, vol. 58, no. 12, pp. 5389–5407, 12 Dec. 2020.

[116] P. A. Absil, R. Mahony, and R. Sepulchre, "Optimization algorithms on matrix manifolds," in *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, 2009, ISBN: 978-0-691-13298-3.

[117] J. Townsend, N. Koep, and S. Weichwald, "Pymanopt: A python toolbox for optimization on manifolds using automatic differentiation," *Journal of Machine Learning Research*, vol. 17, no. 137, pp. 1–5, 137 2016.

[118] R. Gautier, P. Pandita, S. Ghosh, and D. Mavris, "A fully Bayesian gradient-free supervised dimension reduction method using Gaussian processes," *International Journal for Uncertainty Quantification*, vol. 12, no. 2, 2022.

[119] A. Gelman, *Bayesian Data Analysis* (Chapman & Hall/CRC Texts in Statistical Science), Third edition. Boca Raton: CRC Press, 2014, 661 pp., ISBN: 978-1-4398-4095-5.

[120] R. McElreath, *Statistical Rethinking: A Bayesian Course with Examples in R and Stan* (Chapman & Hall/CRC Texts in Statistical Science Series 122). Boca Raton: CRC Press/Taylor & Francis Group, 2016, 469 pp., ISBN: 978-1-4822-5344-3.

[121] S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, *Handbook of Markov Chain Monte Carlo*. CRC press, 2011.

[122] R. Tipireddy and R. Ghanem, "Basis adaptation in homogeneous chaos spaces," *Journal of Computational Physics*, vol. 259, pp. 304–317, 2014.

[123] M. Girolami, B. Calderhead, and S. A. Chin. "Riemannian manifold hamiltonian monte carlo." arXiv: 0907.1100. (2009).

[124] S. Byrne and M. Girolami, "Geodesic monte carlo on embedded manifolds," *Scandinavian Journal of Statistics*, vol. 40, no. 4, pp. 825–845, 4 2013.

[125] R. Shepard, G. Gidofalvi, and S. R. Brozell, "The multifacet graphically contracted function method. II. A general procedure for the parameterization of orthogonal matrices and its application to arc factors," *Journal of Chemical Physics*, vol. 141, no. 6, 6 2014.

[126] R. S. Nirwan and N. Bertschinger, "Rotation invariant householder parameterization for Bayesian PCA," *36th International Conference on Machine Learning, ICML 2019*, vol. 2019-June, pp. 8466–8474, 2019.

[127] M. Jauch, P. D. Hoff, and D. B. Dunson. "Monte Carlo simulation on the Stiefel manifold via polar expansion." arXiv: 1906.07684. (2019).

[128] A. A. Pourzanjani, R. M. Jiang, B. Mitchell, P. J. Atzberger, and L. R. Petzold. "Bayesian inference over the stiefel manifold via the givens representation." arXiv: 1710.09443. (2017).

[129] R. K. Tripathy and I. Bilionis, "Deep UQ: Learning deep neural network surrogate models for high dimensional uncertainty quantification," *Journal of Computational Physics*, vol. 375, pp. 565–588, 2018.

[130] M. J. Betancourt. "Generalizing the no-u-turn sampler to riemannian manifolds." arXiv: 1304.1920. (2013).

[131] P. Tsilifis, P. Pandita, S. Ghosh, V. Andreoli, T. Vandeputte, and L. Wang. "Bayesian learning of orthogonal embeddings for multi-fidelity Gaussian Processes." arXiv: 2008.02386. (2020).

[132] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning* (Adaptive Computation and Machine Learning). Cambridge, Mass: MIT Press, 2006, 248 pp., ISBN: 978-0-262-18253-9.

[133] E. Bingham *et al.*, "Pyro: Deep universal probabilistic programming," *Journal of Machine Learning Research*, vol. 20, no. 1, pp. 973–978, 1 2019.

[134] N. Wycoff, M. Binois, and S. M. Wild. "Sequential learning of active subspaces." arXiv: 1907.11572. (2019).

[135] D. Phan, N. Pradhan, and M. Jankowiak. "Composable Effects for Flexible and Accelerated Probabilistic Programming in NumPyro." arXiv: 1912.11554 [cs, stat]. (Dec. 24, 2019), (visited on 02/03/2022).

[136] J. Bradbury *et al.*, *JAX: Composable transformations of Python+NumPy programs*, version 0.2.5, 2018.

[137] T. Loudon and S. Pankavich, "Mathematical analysis and dynamic active subspaces for a long term model of HIV," *Mathematical Biosciences and Engineering*, vol. 14, no. 3, pp. 709–733, 3 2017.

[138] T. W. Lukaczyk, P. Constantine, F. Palacios, and J. J. Alonso, "Active subspaces for shape optimization," in *10th AIAA Multidisciplinary Design Optimization Conference*, 2014, p. 1171.

[139] "As-data-sets/PDE_notebook.ipynb at master · paulcon/as-data-sets · GitHub." (), (visited on 04/13/2022).

[140] A. Knyazev and P. Zhu, "Principal angles between subspaces and their tangents," *Mitsubishi Electric Research Laboratories*, 2012.

[141] A. Gelman and D. B. Rubin, "Inference from Iterative Simulation Using Multiple Sequences," *Statistical Science*, vol. 7, no. 4, 4 Nov. 1, 1992.

[142] N. Wycoff, M. Binois, and S. M. Wild. "Sequential Learning of Active Subspaces." arXiv: 1907.11572 [cs, stat]. (Sep. 20, 2020), (visited on 02/06/2022).

[143] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (Springer Series in Statistics), 2nd ed. New York, NY: Springer, 2009, 745 pp., ISBN: 978-0-387-84857-0 978-0-387-84858-7.

[144] H. Akaike, "A new look at the statistical model identification," in *Selected Papers of Hirotugu Akaike*, Springer, 1974, pp. 215–222.

[145] S. Watanabe, "A widely applicable Bayesian information criterion," *Journal of Machine Learning Research*, vol. 14, pp. 867–897, Mar 2013.

[146] R. M. Neal, *Bayesian Learning for Neural Networks* (Lecture Notes in Statistics 118). New York: Springer, 1996, ISBN: 978-0-387-94724-2.

[147] I. Sobol', "Sensitivity estimates for nonlinear mathematical models," vol. 1, no. 4, pp. 407–414, 1993.

[148] C. Perron, D. Rajaram, and D. N. Mavris, "Multi-fidelity non-intrusive reduced-order modelling based on manifold alignment," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 477, no. 2253, p. 20 210 495, 2253 Sep. 2021.

[149] "Aerodynamic Design Optimization Discussion Group (ADODG) · MDO Lab." (), (visited on 04/13/2022).

[150] M. A. Bouhlel, N. Bartoli, A. Otsmane, and J. Morlier, "Improving kriging surrogates of high-dimensional design models by Partial Least Squares dimension reduction," *Structural and Multidisciplinary Optimization*, vol. 53, no. 5, pp. 935–952, 5 May 1, 2016.

[151] G. Geraci, A. Gorodetsky, M. Eldred, and J. Jakeman, "Leveraging active directions for efficient multifidelity UQ," in *7th European Conference on Computational Fluid Dynamics (ECFD 7)*, 2018, pp. 11–15.

[152] K. Panda, R. King, A. Glaws, and K. Potter, "Multi-fidelity Active Subspaces for Wind Farm Uncertainty Quantification," in *AIAA Scitech 2021 Forum*, VIRTUAL EVENT: American Institute of Aeronautics and Astronautics, Jan. 11, 2021, ISBN: 978-1-62410-609-5.

[153] B. Liu, *Multi-fidelity model with dimension reduction*, Dec. 2020.

[154] B. Liu and G. Lin, "High-Dimensional Nonlinear Multi-Fidelity Model with Gradient-Free Active Subspace Method," *Communications in Computational Physics*, vol. 28, no. 5, pp. 1937–1969, Jun. 2020.

[155] F. Romor, M. Tezzele, and G. Rozza, "Multi-fidelity data fusion for the approximation of scalar functions with low intrinsic dimensionality through active subspaces," *PAMM*, vol. 20, no. S1, Mar. 2021.

[156] F. Romor, M. Tezzele, M. Mrosek, C. Othmer, and G. Rozza. "Multi-fidelity data fusion through parameter space reduction with applications to automotive engineering." arXiv: 2110.14396 [cs, math]. (Oct. 27, 2021), (visited on 02/15/2022).

[157] K. Cutajar, M. Pullin, A. Damianou, N. Lawrence, and J. González. "Deep Gaussian Processes for Multi-fidelity Modeling." arXiv: 1903.07320 [cs, stat]. (Mar. 18, 2019), (visited on 02/17/2022).

[158] L. Le Gratiet, C. Cannamela, and B. Iooss, "A Bayesian approach for global sensitivity analysis of (multifidelity) computer codes," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 2, no. 1, pp. 336–363, 1 2014.

[159] M. S. Eldred *et al.*, "DAKOTA : A multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis. Version 5.0, user's manual.," SAND2010-2183, 991842, May 1, 2010, SAND2010–2183, 991 842.

[160] P. Perdikaris, M. Raissi, A. Damianou, N. D. Lawrence, and G. E. Karniadakis, "Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 473, no. 2198, p. 20 160 751, Feb. 2017.

[161] T. D. Economon, F. Palacios, S. R. Copeland, T. W. Lukaczyk, and J. J. Alonso, "SU2: An Open-Source Suite for Multiphysics Simulation and Design," *AIAA Journal*, vol. 54, no. 3, pp. 828–846, 3 Mar. 2016.

[162] PACE, *Partnership for an Advanced Computing Environment (PACE)*, manual, 2017.

[163] B. Letham, R. Calandra, A. Rai, and E. Bakshy, "Re-examining linear embeddings for high-dimensional bayesian optimization," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 1546–1558.

[164] M. Binois and N. Wycoff. "A survey on high-dimensional Gaussian process modeling with application to Bayesian optimization." arXiv: 2111.05040 [math]. (Nov. 9, 2021), (visited on 02/19/2022).

[165] A. Tran, "Scalable3-BO: Big Data Meets HPC - A Scalable Asynchronous Parallel High-Dimensional Bayesian Optimization Framework on Supercomputers," in *Volume 2: 41st Computers and Information in Engineering Conference (CIE)*, Virtual, Online: American Society of Mechanical Engineers, Aug. 17, 2021, V002T02A008, ISBN: 978-0-7918-8537-6.

[166] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, "A Limited Memory Algorithm for Bound Constrained Optimization," *SIAM Journal on Scientific Computing*, vol. 16, no. 5, pp. 1190–1208, Sep. 1995.

[167] Z. Wang, M. Zoghi, F. Hutter, D. Matheson, N. De Freitas, *et al.*, "Bayesian optimization in high dimensions via random embeddings.," in *IJCAI*, Citeseer, 2013, pp. 1778–1784.

[168] Z. Wang, F. Hutter, M. Zoghi, D. Matheson, and N. De Feitas, "Bayesian Optimization in a Billion Dimensions via Random Embeddings," *Journal of Artificial Intelligence Research*, vol. 55, pp. 361–387, Feb. 19, 2016.

[169] A. Nayebi, A. Munteanu, and M. Poloczek, "A framework for Bayesian optimization in embedded subspaces," in *Proceedings of the 36th International Conference on Machine Learning*, K. Chaudhuri and R. Salakhutdinov, Eds., ser. Proceedings of Machine Learning Research, vol. 97, PMLR, Jun. 9–15, 2019, pp. 4752–4761.

[170] R. Moriconi, M. P. Deisenroth, and K. S. Sesh Kumar, "High-dimensional Bayesian optimization using low-dimensional feature spaces," *Machine Learning*, vol. 109, no. 9-10, pp. 1925–1943, Sep. 2020.

[171] H. Tran-The, S. Gupta, S. Rana, and S. Venkatesh, "Trading Convergence Rate with Computational Budget in High Dimensional Bayesian Optimization," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 03, pp. 2425–2432, Apr. 3, 2020.

[172] D. Yenicelik. "Parameter Optimization using high-dimensional Bayesian Optimization." arXiv: 2010.03955 [cs, stat]. (Oct. 5, 2020), (visited on 02/19/2022).

[173] M. Malu, G. Dasarathy, and A. Spanias, "Bayesian Optimization in High-Dimensional Spaces: A Brief Survey," in *2021 12th International Conference on Information, Intelligence, Systems & Applications (IISA)*, Chania Crete, Greece: IEEE, Jul. 12, 2021, pp. 1–8, ISBN: 978-1-66540-032-9.

[174] T. Zhou and Y. Peng, "Active learning and active subspace enhancement for PDEM-based high-dimensional reliability analysis," *Structural Safety*, vol. 88, p. 102 026, Jan. 2021.

[175] D. Khatamsaz and D. L. Allaire, "Materials Design using an Active Subspace Batch Bayesian Optimization Approach," in *AIAA SCITECH 2022 Forum*, San Diego, CA & Virtual: American Institute of Aeronautics and Astronautics, Jan. 3, 2022, ISBN: 978-1-62410-631-6.

[176] N. Navaneeth and S. Chakraborty, "Surrogate assisted active subspace and active subspace assisted surrogate—A new paradigm for high dimensional structural reliability analysis," *Computer Methods in Applied Mechanics and Engineering*, vol. 389, p. 114 374, Feb. 2022.

[177] R. Jin, W. Chen, and A. Sudjianto, "On Sequential Sampling for Global Metamodeling in Engineering Design," in *Volume 2: 28th Design Automation Conference*, Montreal, Quebec, Canada: ASMEDC, Jan. 1, 2002, pp. 539–548, ISBN: 978-0-7918-3622-4.

[178] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, "Design and analysis of computer experiments," *Statistical science*, pp. 409–423, 1989.

[179] M. C. Shewry and H. P. Wynn, "Maximum entropy sampling," *Journal of Applied Statistics*, vol. 14, no. 2, pp. 165–170, Jan. 1987.

[180] M. D. Morris, T. J. Mitchell, and Donald Ylvisaker, "Bayesian design and analysis of computer experiments: Use of derivatives in surface prediction," *Technometrics : a journal of statistics for the physical, chemical, and engineering sciences*, vol. 35,

no. 3, pp. 243–255, 1993. eprint: https://www.tandfonline.com/doi/pdf/10.1080/00401706.1993.10485320.

[181] J. Beck and S. Guillas, "Sequential Design with Mutual Information for Computer Experiments (MICE): Emulation of a Tsunami Model," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 4, no. 1, pp. 739–766, Jan. 2016.