EXPLORING STRENGTH IN WEAKNESS: MULTIFACETED WEAKLY SUPERVISED LEARNING APPROACHES FOR TELEHEALTH AND INDUSTRIAL QUALITY PREDICTIONS

A Dissertation
Presented to
The Academic Faculty

by

Dhari Alenezi

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Industrial and Systems Engineering

Georgia Institute of Technology
December 2024

EXPLORING STRENGTH IN WEAKNESS: MULTIFACETED WEAKLY SUPERVISED LEARNING APPROACHES FOR TELEHEALTH AND INDUSTRIAL QUALITY PREDICTIONS

Approved by:

Dr. Jianjun Shi, Advisor
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Jing Li, Advisor
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Kamran Paynabar
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Nagi Gebraeel
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Andi Wang
School of Industrial and Systems Engineering
University of Wisconsin-Madison

Date Approved: November 15th, 2024

To My wife and Parents.

ACKNOWLEDGMENTS

First and foremost, I would like to express my deepest gratitude to my advisors, Dr. Jianjun Shi and Dr. Jing Li, for their unwavering support, guidance, and mentorship throughout my Ph.D. journey. Their insightful advice, constructive feedback, and constant encouragement have been invaluable in shaping my research and helping me grow both academically and personally. Their passion, dedication, and kindness have left a lasting impact on me, and I feel incredibly fortunate to have worked under their supervision.

I am also immensely grateful to my dissertation committee members, Dr. Kamran Paynabar, Dr. Nagi Gebraeel, and Dr. Andi Wang, for their time, thoughtful feedback, and valuable contributions to this work. Their expertise and guidance have helped me refine my research and achieve my academic goals.

A special thank you goes to my lab mates: Dr. Andi Wang, Dr. Xinran Shi, Dr. Feng Wang, Dr. Shancong Mou, Mr. Michael Biehler, Ms. Alina Gorbunova, Dr. Huihui Miao, and Ms. Zihan Zhang. I am truly fortunate to have shared this journey with such talented and supportive individuals. Your camaraderie, collaboration, and friendship have made this experience not only productive but also enjoyable.

Lastly, and most importantly, I would like to extend my heartfelt appreciation to my parents and my wife for their unconditional love, support, and encouragement. To my parents, thank you for believing in me and always pushing me to pursue my dreams. To my wife, your unwavering patience, love, and care have been my greatest source of strength

throughout this process. This work would not have been possible without you. This thesis is dedicated to you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The rapid advancement in data collection technologies has significantly impacted the fields of telehealth and industrial manufacturing. However, the challenge of obtaining comprehensive and precise labels for this data poses significant difficulties for traditional machine learning models. This thesis addresses these challenges by proposing three novel weakly supervised learning (WSL) approaches, specifically targeting the domains of telehealth and manufacturing quality prediction.

The first study introduces a Ranking-Based Weakly Supervised Learning (RWSL) model for assessing disease severity in telemonitoring, with a focus on Parkinson's disease. Using data from the mPower app, this model integrates both labeled and ranked samples to improve predictive accuracy, overcoming the challenge of limited labeled data. By leveraging weak supervision, the RWSL model provides a more accurate and timely assessment of disease progression.

The second study presents a Physics-Informed Weakly Supervised Learning (PWL) framework designed for quality prediction in industrial manufacturing processes. This approach integrates the physics-based understanding of manufacturing processes with machine learning models, improving prediction accuracy despite the scarcity of labeled samples. The PWL model bridges the gap between theoretical physics-based models and practical machine learning applications, enabling more effective quality control in manufacturing.

The third study proposes a Multi-source Multi-task Weakly Supervised Transfer Learning (M2WeST) approach for telehealth, which addresses the heterogeneity of disease manifestations across patients. This model combines data from multiple patients, incorporating both strong and weak labels to provide personalized disease severity predictions. The M2WeST framework significantly improves the robustness and accuracy of predictions, even in scenarios with limited labeled data.

These three studies contribute to the advancement of weakly supervised learning in both telehealth and industrial applications, offering innovative solutions for extracting meaningful insights from weakly labeled data.

CHAPTER 1. INTRODUCTION

In the modern era of telehealth and manufacturing, the rapid growth of data collection technologies is undeniable. Contemporary systems generate data that is complex and abundant but often lacks the comprehensive labeling necessary for traditional supervised learning techniques. Whether from remote health monitoring devices tracking patient activity or sensors embedded across manufacturing lines, this data contains valuable insights. However, this potential is often underutilized due to the prohibitive costs and practical challenges associated with obtaining precise labels. To address this, the research landscape has begun shifting toward methods that learn from limited or imprecise labels, such as Weakly Supervised Learning (WSL). WSL offers a valuable approach to address labeled data insufficiency, providing tools to extract meaningful predictions even in scenarios with incomplete information.

This thesis delves into the innovative domain of WSL, presenting a series of multifaceted approaches that leverage weak labels to derive significant predictions in telehealth monitoring and industrial quality assurance. By focusing on models that operate effectively under weak supervision constraints, this research pushes the boundaries of what can be achieved with limited labeled data. Through three pivotal papers, this thesis establishes new methodologies in (i) Ranking-Based Models for Disease Severity Assessment in Telemonitoring (F. Alenezi et al., 2022), (ii) Physics-Informed Learning for Quality Prediction in Manufacturing (Alenezi et al., 2024), and (iii) Multi-Source Multi-Task Modeling for Robust Disease Severity Assessment in Telehealth (Alenezi et al., in preparation). The following chapters outline the development, implementation, and evaluation of these frameworks, emphasizing their contributions to advancing WSL and their real-world applicability.

In Chapter 2, we explore healthcare analytics in the context of telemonitoring for Parkinson's disease (PD) by introducing a novel Ranking-Based Weakly Supervised Learning (RWSL) model. This model is designed to handle data from mobile health applications, such as the mPower app (Bot et al., 2016), which captures patient activities relevant to assessing PD symptoms.

Telemonitoring for PD has traditionally focused on data collection and patient classification relative to healthy controls. This has resulted in significant research around feature extraction from telemonitoring data for clinical diagnosis (Far et al., 2021; Lenain et al., 2020). However, a critical gap remains in using telemonitoring data to predict disease severity, which is essential for timely intervention and disease progression monitoring. While platforms like mPower enable comprehensive data collection, they lack predictive analytics capabilities necessary to provide feedback on disease severity (Wang et al., 2020).

The challenge of building predictive models for PD severity lies primarily in the scarcity of labeled samples, where the disease severity (Y) corresponding to collected activity data (X) is not readily available. This is due to the infrequent clinical assessments that contrast with the higher frequency of data collection in telemonitoring apps like mPower. Additionally, self-administered data collection introduces issues like non-adherence and data entry errors, which further diminish the number of usable samples (Sica et al., 2021). Moreover, although apps like mPower collect additional patient data, such as medication usage, integrating this information into predictive models poses its own challenges.

To address these challenges, the RWSL model utilizes domain knowledge, particularly regarding medication intake, to infer the relative ranking of unlabeled data samples (F. Alenezi et al., 2022). Leveraging both labeled and unlabeled data aligns with semi-supervised learning (SSL), but RWSL differs by utilizing domain knowledge to provide weak labeling information in the form of ranked samples. For instance, in PD, the effect of medication on reducing disease severity can be used to rank two samples of a patient's activity data that are collected before and after medication as $y_i \geq y_j$, where y_i and y_j are the severity levels corresponding to the two samples, even though the exact values for y_i and y_j are unknown.

This innovative approach not only mitigates the challenge of insufficient labeled data but also allows for the integration of ranked samples to build more robust and accurate predictive models. While RWSL leverages techniques akin to learning-to-rank algorithms in Information Retrieval (Liu, 2011), its goal is to predict disease severity rather than document relevance. Thus, the RWSL model represents a significant advancement in telemonitoring analytics, enabling more effective management of PD through improved disease monitoring.

In Chapter 3, we introduce the Physics-Informed Weakly-Supervised Learning (PWL) framework, where weak labels are derived from partial physical knowledge embedded in manufacturing systems. Manufacturing processes generate substantial amounts of data through sensors that monitor key process variables, but labeled samples indicating product quality are often scarce due to the high cost and impracticality of end-of-line quality testing. These weak labels help improve quality predictions by utilizing physics-based models and constraints to augment limited labeled data from manufacturing processes. This innovative use of physics-based weak supervision offers a novel way to integrate theoretical models with machine learning in environments where labeled data is costly or sparse.

To address the scarcity of labeled data, there has been growing interest in combining physics-based models with data-driven machine learning approaches. Physics-based models, grounded in physical laws and principles, offer valuable insights into manufacturing processes without requiring large amounts of labeled data. However, recent research integrating physics-based models with ML models (Karniadakis et al., 2021; Leonesio and Fagiano, 2022; Montes et al., 2021; Nabian and Meidani, 2020; Chen et al., 2021) has revealed gaps. These include ineffective handling of data scarcity, calibration challenges, and a lack of optimization within the integrated models. Moreover, traditional weakly-supervised learning models have not adequately explored the integration of physics-based weak supervision, where the physics-based knowledge acts as an indirect source of weak labels (Zhou, 2018).

The integration of physics-based models with data-driven ML presents several challenges. Physics-based models often contain uncertainties, and their outputs may be correlated with product quality but not directly measure it. Additionally, calibration is often ad-hoc and not optimized. The proposed PWL model (Alenezi et al., 2024) optimizes both the ML and physics-based models, addressing calibration issues and providing a comprehensive framework that bridges the gap between theoretical physics and practical data-driven predictions. By leveraging partial physical knowledge as a source of weak labels, the PWL model enhances the predictive capability of quality assessment models in real-world manufacturing environments with scarce labeled data.

In Chapter 4, we address the challenge of managing the substantial variability in chronic disease manifestations across different patients, each exhibiting unique disease progression patterns. This variability complicates the development of generalized models, especially in telemonitoring settings where each patient's data reflects individual-specific disease progression. While transfer learning techniques have been widely employed to transfer knowledge between domains, these approaches are typically designed for single-source scenarios and often struggle to handle the heterogeneity and patient-specific variability present in multi-source data (Pan & Yang, 2010).

To tackle this issue, Chapter 4 introduces the Multi-source Multi-task Weakly Supervised Transfer Learning (M2WeST) model. The M2WeST model leverages data from multiple patients, integrating both strongly labeled data (disease severity assessments) and weakly labeled data (ranked comparisons based on relative severity) to build a more robust predictive model for disease severity (Alenezi et al., in preparation). For each of the target and source patients, weak labels are generated by utilizing domain-specific knowledge, such as the effects of medication, to rank data points from the same patient, allowing the model to capture relative severity differences even when exact labels are unavailable.

By incorporating strongly labeled data and weakly labeled data from multiple patients, M2WeST is able to account for the inherent variability in disease progression across different individuals. This approach allows the model to leverage both patient-specific details and shared patterns across patients. By handling data from multiple sources and learning from both types of labels, M2WeST is able to adjust its predictions based on individual disease trajectories while utilizing commonalities to enhance model robustness. This results in more personalized and accurate predictions that are better suited to telehealth applications, where patient data is often incomplete and heterogeneous.

Finally, Chapter 5 concludes the thesis, summarizing the key contributions and offering directions for future research in weakly supervised learning for healthcare and manufacturing.

CHAPTER 2. A RANKING-BASED WEAKLY SUPERVISED LEARNING MODEL FOR TELEMONITORING OF PARKINSON'S DISEASE

2.1 Introduction

In today's world, mobile phones have moved from being a simple communication tool to being an essential part of people's daily lives. According to App Annie, an application analytics firm, Americans spent an average of four hours per day on their phones in 2020 (Kristianto, 2021). Combining this high screen time and the increasing sensing capabilities of mobile phones, remote and continuous monitoring of an individual's health has become easier than ever, and with negligible costs. Modern smartphones are equipped with various sensors and custom-built applications that can collect user-specific health data. Monitoring such health data is commonly referred to as telemonitoring. Telemonitoring is defined as the use of technological devices to remotely monitor and transmit information related to a person's health status (Dansky et al., 2008).

In this chapter we focus on the monitoring and modeling of mobile phone-collected data in patients with Parkinson's disease (PD). PD is a brain disorder that leads to aberrations in movement, including tremors, slowness of movement, rigidity, and difficulty with walking, balance, and coordination. Globally, PD affects seven to ten million people worldwide (Goetz et al., 2009). PD costs the United States $52 billion every year (Michael J. Fox Foundation [MJFF], 2019). There is currently no cure for PD, but treatments are available to control and reduce symptoms. However, to effectively control symptoms, frequent monitoring of the patient's condition is required to titrate mediations, engage in PD-specific physical therapies, and adjust devices such as deep brain stimulators.

The conventional approach to assess the patient's PD condition is through clinical visits where medical experts inquire about symptoms and perform physical examination. These include questions about the patient's medical history and observations of physical signs or symptoms of PD (Rizek et al., 2016). During the physician's assessment, standardized clinical instruments/questionnaires are typically used to provide some guidance and help reduce subjectivity and misdiagnosis errors. For example, Movement Disorder Society-Unified Parkinson's Disease Rating Scale (MDS-UPDRS) is one of the commonly used clinical instruments. UPDRS was developed in 1987 as a gold standard by neurologists for monitoring signs and symptoms of PD (Fahn & Elton, 1987). UPDRS remained the most widely used PD scale until an updated version, MDS-UPDRS, was commissioned by the MDS in 2007. MDS-UPDRS improves upon the original UPDRS by enhancing scale properties and including more non-motor items so that the breadth of PD manifestations is adequately captured.

In conventional clinical practice, clinical visits with a physician happen around every 4–6 months and much less frequently in resource-limited countries and regions (e.g., in years) (Dotchin et al., 2011). The lack of frequent assessment of a patient's PD condition leads to a delay in effective intervention. Telemonitoring technologies offer one possible solution to addressing this issue by making it possible to remotely and frequently assess the patient' clinical condition based on individually collected mobile health data.

To enable mobile-based telemonitoring, an application is needed to become the platform for data collection and transmission. Generally, telemonitoring systems can be divided into self-administered telemonitoring systems such as mPower (Bot et al., 2016) and wearable inertial sensors-based telemonitoring systems such as those introduced in (Sica et al., 2021). mPower is an app installed on the user's mobile phone. Once launched, the app would guide the user to perform several pre-designed activities that measure PD symptoms, such as tapping, speaking, and walking. The data of these activities would be recorded by the mobile's built-in sensors such as accelerometers, gyroscopes, and microphones. Alternatively, wearable inertial sensors-based telemonitoring systems collect free-living movements for a prolonged period of time (Sica et al., 2021). Such an approach has been shown to reduce the well-known "Hawthorne observation effect" (Maetzler et al., 2013) since patients tend to pay less attention when performing their daily free-living activities.

Both types of aforementioned telemonitoring systems open the opportunity for more frequent and convenient assessment of the conditions for PD patients compared to the conventional clinical setting. This chapter focuses on the former type of systems represented by mPower with several promising features: Firstly, mPower collects data specific to activities that a patient commonly performs in the clinic; such activities have been shown to capture symptoms relevant to assessing PD severity. On the other hand, the data collected by inertial sensors systems are less focused and not specific to activities that are commonly used to assess PD severity. Secondly, in many inertial sensors-based systems, the assessment of multiple symptoms necessitates employing a high number of wearable devices, compromising patients' comfort. In contrast, mPower does not have such issues since the telemonitoring is conducted purely through a mobile app interface. Thirdly, the ease of using a mobile phone as the data collection platform by mPower makes it possible to collect other important information for PD patients in addition to their activities, such as medication use. Such information helps provide a better assessment of the PD condition and progression.

Despite the promising features, telemonitoring systems like mPower have some limitations that need to be addressed to fully utilize the system's capabilities. First and foremost, mPower provides a data collection platform with no feedback to users (i.e., patients and physicians). Providing feedback in terms of the severity and condition of the patient's disease is critical as it would help with timely intervention. To provide such feedback, data analytics and machine learning algorithms are needed to integrate the collected activity data into predicted disease severity scores. However, current telemonitoring systems do not have adequate predictive analytics capability beyond providing a data collection platform. This chapter aims to mitigate this gap.

The predictive analytics task posed above is not trivial due to some uniquely challenging properties of the datasets collected via mPower. First, there is a lack of sufficient labeled samples to train the machine learning model. Specifically, while the activity data (X) can be conveniently obtained via the telemonitoring system, it is not easy to obtain the matched observation for PD severity (Y, a.k.a. label) as this requires expert assessment. For example, using the mPower app, a patient can perform app-guided activities such as tapping, walking, and speaking anytime and anywhere; thus, the activity data can be collected as frequently as multiple times a day. However, disease severity assessment typically only happens at most every 4-6 months. As a result, the sample size with matched (X,Y) datasets from each patient is quite limited. Furthermore, because the data collection process is patient self-administered, there is missing data due to non-adherence and mistakes (Son et al. 2020), which further reduces the qualified samples that can be used to train a robust model. Lastly, as mentioned previously as a promising feature, mPower collects other crucial information such as medication use beyond just the activity data. However, how to effectively leverage this information to augment the capability of the predictive model training remains a challenging issue.

To overcome the aforementioned challenges and provide advanced analytics capability to integrate with the mPower platform, we propose a novel machine learning based predictive model called the Ranking-based Weakly Supervised Learning (RWSL) model. RWSL aims to overcome the challenge of insufficient labeled samples (X,Y) by effectively leveraging and integrating the knowledge in the abundant activity data without a matching Y, known as unlabeled data. Integrating unlabeled and labeled data to train a machine learning model is known as semi-supervised learning (SSL), belonging to the broader field of weakly-supervised learning. However, different from the existing SSL that assumes the Y for each unlabeled sample is completely unknown, we note that domain knowledge can be utilized to provide some weak labelling information of Y. Specifically, domain knowledge may allow for ranking of unlabeled samples in terms of their Y values. As mentioned previously, mPower collects other crucial patient information beyond the activity data, such as medication use. In PD, medication can lessen the severity of the disease; thus, two samples of a patient's activity data that are collected before and after medication can be ranked as $y_i \geq y_j$, where y_i and y_j are the severity levels corresponding to the two samples, even though the exact values for y_i and y_j are unknown. Our model aims to integrate insufficient labeled samples and domain knowledge based ranked samples to build a predictive model for a response variable Y (e.g., disease severity) using features X (e.g., mobile-collected activity data). Since RWSL utilizes domain knowledge to create ranked samples, the model is also related to the machine learning subfield of Information Retrieval (IR), in which learning-to-rank algorithms have been developed to determine the relevance of documents concerning a given user query and order them accordingly. However, although RWSL can incorporate ranked samples in training, the objective of RWSL is different from existing learning-to-rank algorithms in IR, as our model aims to integrate ranked samples with labeled samples to build a superior predictive model.

The remainder of this chapter is organized as follows: Section 2.2 reviews the related work and points out gaps. Section 2.3 presents the proposed model, RWSL. Section 2.4 presents simulation experiments. Section 2.5 presents a real data application in PD. Section 2.6 concludes the chapter.

2.2 Literature Review

Our work exists at the intersection of three lines of research pursued by different research communities. Section 2.2.1 will be dedicated to reviewing research related to healthcare analytics for PD based on telemonitoring data. Section 2.2.2 will focus on reviewing research in two subfields of machine learning that are related to our model.

2.2.1 Healthcare analytics for PD based on telemonitoring data

Telemonitoring has enabled the generation of a tremendous amount of patient-specific data that is yet to be fully explored. Such data has made it possible for monitoring, predictive analytics, and understanding of various diseases. The importance of telemonitoring of PD lies in providing data about a patient's daily life not observed by a physician, which may help gain understanding of the complex nature of PD and develop better interventions. Generally, the healthcare analytics literature for PD can be divided based on the end goal and type of PD data utilized. For example, some work has focused on processing of the raw telemonitoring data and feature engineering to obtain proper features related to a clinician's diagnosis (Far et al., 2021; Lenain et al., 2020; Wang et al., 2020). Other research has focused on applying classification models to the extracted features related to PD to classify subjects as PD patients or healthy controls, which may help with PD screening (Abujrida et al., 2017; Arora et al., 2014; Zhang et al., 2020). Another area of research focuses on specific symptoms related to PD (Pastorino et al., 2011; Rigas et al., 2012; Yoneyama et al., 2013); for example, (Rigas et al., 2012) built a hidden Markov model for assessing tremor in PD. Lastly, one major area of research is related to the ability of using telemonitoring apps to collect samples with respect to the timing of medication; such samples have provided researchers the opportunity to study patients' responses to medication (Matarazzo et al., 2019; Zhan et al., 2016).

In summary, the existing telemonitoring data analytics research for PD primarily focuses on feature engineering, classification of PD patients from healthy controls, modeling and understanding of the dynamics of some specific symptoms, and tracking of response to medication. Little research has been done to use telemonitoring activity data to predict disease severity, which is an important task to allow for timely monitoring of disease progression and effective intervention. The present chapter aims to mitigate this gap. On the other hand, it is a challenging task to train a robust machine learning model to predict disease severity using telemonitoring activity datasets. While the activity data (X) can be conveniently obtained via the telemonitoring system, it is not easy to obtain the matched observation for PD severity (Y, a.k.a. label). This results in a small labeled sample size. Furthermore, because the telemonitoring activity is patient self-administered, there is missing data due to non-adherence and mistake (Son et al. 2020), which further reduces the qualified samples that can be used to train a robust model. In this chapter, we address these challenges by proposing a new machine learning model, RWSL, which integrates ranked samples and labeled samples for training a robust model to predict PD severity (Y) using activity data (X).

2.2.2 Weakly-supervised learning and information retrieval algorithms in machine learning

Supervised learning is a type of machine learning models that uses features x to predict or classify a response variable y. To train a supervised learning model, a training dataset is needed, which typically contains samples with both x and y available. Such samples are called labeled samples because their response variables are precisely measured. However, it may be difficult to obtain precisely measured response variable for each sample in many application domains, i.e., the label is “weak”. This creates the need for weakly-supervised learning algorithms. Weakly-supervised learning can be divided into several major subfields: incomplete-supervision, inexact-supervision, and inaccurate-supervision (Zhou, 2018). For incomplete supervision, only a subset of the training samples has labels while the other samples are unlabeled, which is the category our proposed method falls into.

Incomplete supervision can be further divided into two sub-fields: active learning (Settles, 2009) and semi-supervised learning (Chapelle et al., 2006; Zhu, 2008; Zhou & Li, 2010). Active learning assumes that there is an 'oracle', such as a human expert, that can be queried to get labels for selected unlabeled samples (Settles, 2009). The goal of active learning algorithms is to select the most valuable unlabeled samples to query.

On the other hand, semi-supervised learning (SSL) assumes no 'oracle' intervention; instead, it aims to integrate labeled and unlabeled samples to train a model to predict or classify y using x. There are different types of SSL algorithms. Some methods assume that samples have inherent cluster structure, and therefore samples falling into the same cluster should have a similar class label (Chapelle et al., 2006). Some methods treat the labels of unlabeled samples as missing values based on the assumption that labeled and unlabeled data samples are both generated from the same model (Nigam et al., 2000). Graph-based methods are popular in SSL, in which a graph is constructed with nodes corresponding to training samples and edges representing relationships between the nodes (Fujino et al., 2005; Blum & Chawla, 2001; Zhu et al., 2003; Zhou et al., 2004). Another type of popular methods assumes that low-dense regions separate labels and aims to identify a classification boundary that goes across the less-dense region while keeping the labeled data correctly classified (Joachims, 1999; Chapelle & Zien, 2005; Li et al., 2013). Despite the fact that SSL is a popular field in machine learning, to our best knowledge, there is little work on leveraging the rank information of unlabeled samples like our method.

On the other hand, we found that ranking algorithms have been mainly investigated in another machine learning field called information retrieval (IR). IR is the process of finding documents of an unstructured nature that satisfies an information/query need from extensive collections of documents (Sanderson et al., 2010). IR relies heavily on learning-to-rank algorithms since returned results may not match the search query and need to be ranked by relevance. Learning-to-rank algorithms can be divided into three categories: pointwise approaches, pairwise approaches, and listwise approaches (Liu, 2011). Pointwise approaches are the earliest in this field. The basic idea is to map the documents' feature vector from ordinal scales to numeric values, and then solve the ranking as regression, classification, and ordinal regression, respectively (Li et al., 2007; Crammer & Singer, 2001). A limitation of pointwise approaches is that the input is a single document without considering the inter-dependency between documents, and thus the position of a document in the final ranked list is invisible to the pointwise loss function (Liu, 2011). Alternatively, listwise approaches directly compare the relevance of a list of documents to a query considering the inter-dependency between documents (Cao et al., 2007; Kondor et al., 2007). Although the performance of listwise approaches has been shown to be better than pointwise and pairwise approaches in many cases, the computational complexity is very high due to the permutation-based loss function evaluation (Liu, 2011). Finally, pairwise methods compare every two documents' relevance, and then rank all the documents based on all these comparison results (Burges et al., 2006; Burges et al., 2005; Joachims, 2002).

Although our method can incorporate ranked samples in training, the objective of our method is different from existing learning-to-rank algorithms in IR. First, learning-to-rank algorithms are mainly aimed to rank a set of documents related to a search query. This means that the objective of the problem is to find the best set of coefficients that meets an unknown latent ranking variable. Hence, a solution of the problem may meet the ordinal requirement but not necessarily produce a good prediction performance. In contrast, our method aims to integrate ranked samples with labeled samples to build an accurate predictive model. Second, learning-to-rank algorithms start by a set of documents whose relevance to a search query is judged using a feature extraction or retrieval function. For example, decisions about ranking a certain pair of documents relative to a query may be based on how frequently a matching term appears within a specific document. This is not how features are intended to be used in our problem, while our objective is to find how features are relevant to response variable of interest.

In summary, even though our method has some connection with the fields of weakly-supervised learning and IR in machine learning, these existing fields do not address the same problem as ours, which is to integrate labeled and ranked samples to build a predictive model for a response variable y using features x.

2.3 Ranking-based Weakly Supervised Learning (RWSL) model

2.3.1 Model Formulation

Let $\boldsymbol{x}$ denote the feature vector, e.g., features extracted from the tapping signal recorded on a PD patient's smartphone. Let y denote the severity level of the disease, e.g., the MDS-UPDRS score of the patient. Our goal is to learn a model $f(\boldsymbol{x})$ to predict y. The uniqueness of RWSL is that it can integrate two types of data in training the predictive model: labeled samples and ranked samples. A labeled sample is one for which both the feature vector and the corresponding disease severity level are available. Suppose there are L labeled samples, $\{\boldsymbol{x}_l, y_l\}$, l=1, …, L. Ranked samples are those for which only the feature vector of each sample is available, and there is a criterion d that allows for ranking of each pair of samples in terms of their disease severity levels. Suppose there are $|\Omega_d|$ pairs of ranked samples, $\{\boldsymbol{x}_i \succcurlyeq \boldsymbol{x}_j\}$, $(i, j) \in \Omega_d$, and the notation "$\succcurlyeq$" means that $y_i \geq y_j$, while the exact values for y_i and y_j are unknown. The criterion d is typically known by domain knowledge, which will be discussed in further detail in section 2.3.3.

To incorporate ranked samples in model training, the problem boils down to learning the most robust function $f(\boldsymbol{x})$ that correctly predicts the ordering of any given ranked samples. Hence, the best function f is the one that will give us a positive margin such that

$$f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j) \geq 0, \qquad (i, j) \in \Omega_d. \tag{1}$$

There are several algorithms in the literature that attempt to solve this problem. One of the popular approaches is the SVMRank algorithm (Jankovic, 2008). SVMRank makes it possible to design an efficient algorithm for finding the function $f \in \mathcal{F}$ that maximizes the ordering constraint in (1) and generalizes well beyond the training data. The algorithm uses margin-maximization which leads to an ordering that is more robust with respect to noise in $\boldsymbol{x}$. However, SVMRank has a different objective from ours: it aims to train a model to rank samples whereas we want to train a model to predict y.

We propose the RWSL model that adopts the margin-maximization concept of SVMRank for incorporating ranked samples, while at the same time adding labeled samples through supervised learning. RWSL has the following model formulation:

$$\min_{\boldsymbol{\beta}} \frac{1}{2} \|\boldsymbol{\beta}\|_2^2 + \frac{\lambda_L}{2|L|} \sum_{l=1}^{L} \left\| y_l - f_{\boldsymbol{\beta}}(\boldsymbol{x}_l) \right\|_2^2 + \frac{\lambda_R}{|\Omega_d|} \sum_{(i,j)\in\Omega_d} \xi_{ij}\,, \tag{2}$$

$s.t$

$$f_{\boldsymbol{\beta}}(\boldsymbol{x}_i) - f_{\boldsymbol{\beta}}(\boldsymbol{x}_j) \geq 1 - \xi_{ij}\,, \forall\,(i,j) \in \Omega_d,$$

$\xi_{ij} \geq 0\,, \forall\,(i,j) \in \Omega_d$, where $\boldsymbol{\beta}$ contains the model coefficients, ξ_{ij} is a non-negative slack variable, and $\| \quad \|_2^2$ is the squared L_2-norm. The objective function consists of three terms: the first term aims to achieve a good generalization on the ranked samples by maximizing the closest distance between two ranked samples defined as $\frac{1}{\|\beta\|_2}$, where it can be shown that maximizing $\frac{1}{\|\beta\|_2}$ is equivalent to minimizing $\|\beta\|_2^2$; the second term uses a quadratic loss to encourage the predicted responses to be close to the true responses for labeled samples; the third term upper-bounds the slack variables to preserve the ordering of ranked samples, which will serve as a "soft" ordering constraint that may allow for some samples to be mis-ordered. Depending on the application, the choice for the mapping function $f_{\boldsymbol{\beta}}$ may vary. In our application, a linear function showed satisfactory results. The coefficients λ_L and λ_R control the relative degrees of emphasis on the labeled and ranked samples, respectively. The degree of emphasis on the ranked samples will depend on the noise in those samples, which is dependent on the choice of ranking threshold. In section 2.4.3, we try to provide some guidance on the effect of ranking threshold selection in reducing noise in ranked samples.

2.3.2 Optimization algorithm

The constrained optimization problem in (2) can be converted to an unconstrained optimization with a hinge loss $L_{hinge}(t) = max(0, 1 - t)$, i.e.,

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + \frac{\lambda_L}{2|L|}\sum_{l=1}^{L}\left\|y_l - f_{\boldsymbol{\beta}}(\boldsymbol{x}_l)\right\|_2^2 + \frac{\lambda_R}{|\Omega_d|}\sum_{(i,j)\in\Omega_d} L_{hinge}\left(1 - \left(f_{\boldsymbol{\beta}}(\boldsymbol{x}_i) - f_{\boldsymbol{\beta}}(\boldsymbol{x}_j)\right)\right). \quad (3)$$

(3) is convex but not differentiable. To make it differentiable, the hinge loss can be replaced by another loss function that has comparable performance (Chapelle & Keerthi, 2010), a Huber loss:

$$L_{huber}(t) = \max(0, 1 - t)^2 - \max(0, -t)^2 \quad (4)$$

The Huber loss combines the robustness of L_1-norm with the stability of L_2-norm. For huge errors, it is linear; for small errors, it is quadratic. The Huber loss also gives fast methods for computing gradient and performing Hessian times vector operations. The problem is now convex, unconstrained, and differentiable, and can be formulated as follows:

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + \frac{\lambda_L}{2|L|}\sum_{l=1}^{L}\left\|y_l - f_{\boldsymbol{\beta}}(\boldsymbol{x}_l)\right\|_2^2 + \frac{\lambda_R}{|\Omega_d|}\sum_{(i,j)\in\Omega_d} L_{huber}\left(1 - \left(f_{\boldsymbol{\beta}}(\boldsymbol{x}_i) - f_{\boldsymbol{\beta}}(\boldsymbol{x}_j)\right)\right). \quad (5)$$

By choosing a linear mapping for the f function, the problem can be further simplified to:

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + \frac{\lambda_L}{2|L|}\sum_{l=1}^{L}\|y_l - \boldsymbol{\beta}^T\boldsymbol{x}_l\|_2^2 + \frac{\lambda_R}{|\Omega_d|}\sum_{(i,j)\in\Omega_d} L_{huber}(1 - \boldsymbol{\beta}^T(\boldsymbol{x}_i - \boldsymbol{x}_j)), \quad (6)$$

which can be solved using the Newton's method (Dennis & Schnabel, 1996).

Parameter tuning: There are two tuning parameters in RWSL, λ_L and λ_R, which control the relative degrees of emphasis on the labeled and ranked samples in training, respectively. Due to the computational efficiency of RWSL, a grid search of the optimal tuning parameters is feasible. Specifically, the dataset is divided into a training set with labeled and ranked samples, and a validation set with labeled samples. Given fixed values of the two tuning parameters, a model is trained, which is then used to predict the response variables of the samples in the validation set. The mean square error (MSE) between the predicted and true responses for the validation samples is computed. In this way, we can compute the MSEs for all possible combinations of the tuning parameter values and choose the optimal tuning parameter combination, $(\lambda_L^*, \lambda_R^*)$, as the one that yields the smallest MSE. When the dataset has limited labeled samples, a cross-validation scheme can be used to replace the training-validation split of the data. Let $\widehat{\boldsymbol{\beta}}_{\lambda_L^*,\lambda_R^*}$ denote the parameter estimates by solving the optimization in (6) under $(\lambda_L^*, \lambda_R^*)$. Then, for any new sample, $\boldsymbol{x}_{new}$, the predicted response variable can be obtained by $\hat{y}_{new} = \widehat{\boldsymbol{\beta}}_{\lambda_L^*,\lambda_R^*}{}^T \boldsymbol{x}_{new}$.

2.3.3 Criteria for ranking samples

The proposed RWSL model assumes the availability of ranked samples according to a criterion d. In the medical field, such criteria typically exist through domain knowledge. For example, medication can lessen the severity of a disease. Thus, two samples of a patient,

x_i and x_j, that are collected before and after medication respectively, can be ranked as $x_i \succcurlyeq x_j$.

Specifically related to PD, the movement disorder of PD occurs largely due to the selective loss of neurons that results in depletion of dopamine in the striatum (Jankovic, 2008; Samii et al., 2004; Sveinbjornsdottir, 2016). Dopaminergic drugs designed to replace the action of dopamine in the deplete striatum. Generally, the clinical effect of dopaminergic drugs is noticed quickly, and may last for several hours, particularly in the early stages of the disease (Zahoor et al., 2018). As a result, it is reasonable to use medication as the criterion d to create pairs of ranked samples for each patient. It can be assumed that the disease is more severe before medication than relatively immediately after medication. Also, it is common for telemonitoring apps to collect samples with respect to the timing of medication, which naturally creates ranked samples. For example, the mPower app requests the users to perform their activities three times a day with at least one time before and one time after medication. Finally, our model is designed to address the scenario of limited labeled samples by compensating for that shortage with ranked samples; however, the model is expected to perform as good as a supervised learning model for patients without ranked samples.

2.4 Simulation Study

In this section, we use simulated data to test the performance and capabilities of our model in comparison with competing methods. Our simulation experiments in section 2.4.1- 2.4.4 aim to answer the following questions: How is the performance of RWSL compared to competing methods under different sample sizes for labeled and ranked data (section

2.4.1)? How much labeled data can be saved by including ranked data in training (section 2.4.2)? What type of ranked data should be included to achieve the best performance of RWSL (section 2.4.3)? How robust is RWSL with respect to label noise (section 2.4.4)?

Data generation: In all the experiments, the data generation process is similar, whereas the parameters values may vary. The data generation process starts by sampling the feature vector from a multivariate normal distribution, i.e., $\boldsymbol{x} \sim MN(\boldsymbol{0}, \boldsymbol{\Sigma}_x)$, where the covariance between the i-th and j-th samples is created by having $\boldsymbol{\Sigma}_{x,i,j} = \alpha^{|i-j|}$, where $\alpha \in (0,1]$. Then, we generate the coefficients corresponding the features by $\boldsymbol{\beta} \sim N(\boldsymbol{\mu}_\beta, \sigma_\beta^2 \mathbf{I})$. All the coefficients are held fixed after they are generated and considered as the ground truth values to be compared with the estimated coefficients by a model. Furthermore, the response variable is created using an additive model as follows: $y = \boldsymbol{\beta}^{\boldsymbol{T}}\boldsymbol{x} + \varepsilon$, where ε is sampled from $N(0, \sigma_\varepsilon^2)$. σ_ε^2 is decided based on the desired signal-to-noise ratio $SNR = \frac{Var(\boldsymbol{\beta}^{\boldsymbol{T}}\boldsymbol{x})}{\sigma_\varepsilon^2}$. In our simulations, the SNR is fixed at five unless stated otherwise. Also, the number of real features is set to be 15, while we add 15 noise features to test the robustness of the model.

Models under comparison: The proposed RWSL integrates labeled and ranked samples in training a predictive model. To our best knowledge, there is no existing model that provides this capability. Because RWSL adopts the margin-maximization concept in its formulation, which is also the basis of RankSVM, we compare RWSL with RankSVM, which can only use ranked samples in training. Additionally, we compare RWSL with supervised learning, which can only use labeled samples in training. Ridge regression is a supervised learning model that adopts a quadratic loss for labeled samples and a squared

L_2 penalty for model coefficients, which are similar to the part of RWSL that accounts for labeled samples. Therefore, we choose ridge regression to compare with RWSL.

Performance metrics: Several performance metrics are adopted to compare the models: 1) Mean Squared Error (MSE) measures the average squared difference between predicted and true responses. The MSE is aimed to evaluate the model's ability to quantify the disease severity accurately. 2) Predictive correlation, i.e., the Pearson correlation between predicted and true responses, which complements MSE as it is bounded between -1 and 1. The predictive correlation is aimed to evaluate if the prediction index is reasonably correlated with the true disease severity, to identify if the patient's condition is getting better or worse. 3) Feature selection (FS) accuracy. As our simulation data purposely includes noise features, we define the following metric to evaluate the capability of a model for identifying the true features:

$$FS\ accuracy = \frac{\sum_{i \in K} |\hat{\beta}_i|}{\sum_{i \in K} |\hat{\beta}_i| + \sum_{j \in N} |\hat{\beta}_j|},$$

where K and N are index sets of true and noise features, respectively.

2.4.1 Model performance under different sample sizes

This experiment aims to evaluate and compare the performances of models under different sample sizes. We create 20 different scenarios, starting from a small-sample setting with 10 labeled samples and 10 ranked pairs (i.e., scenario 1), and gradually increasing the numbers of labeled samples and ranked pairs in a fixed step size in each scenario until reaching a large-sample setting of 200 labeled samples and 1000 ranked pairs (i.e., scenario

20). In each scenario, 70% of the labeled samples and ranked pairs are used for training and the rest for validation. Additionally, we generate a blind test set of 300 labeled samples to evaluate and compare the prediction accuracy of different model.

In each scenario, RWSL is trained based on both labeled samples and ranked pairs. The two tuning parameters of RWSL are selected by minimizing the MSE on the validation data. Ridge regression can only include labeled samples in training and parameter tuning of ridge regression is done similarly to RWSL by minimizing the validation MSE. RankSVM can only include ranked pairs in training and the model is to rank samples not to predict a response variable y. Thus, parameter tuning of RankSVM cannot use validation MSE. Instead, we tune RankSVM by maximizing the percentage of pairs ordered correctly (POC) among the total number of ranked pairs included in the validation set. In the experiments, we note that the POC criterion suffers from the fact that there could be several tunning parameters' values that give the same maximum POC. Therefore, a selection rule has to be in place to deal with this situation. We follow a rule that chooses the smallest parameter value, which showed the most robust performance in our experiments.

The training/validation/testing process is repeated ten times for each model and the average MSE on the test set is plotted in Figure 1. We can observe that the average MSE achieved by RWSL is consistently lower than ridge regression and RankSVM. To assess the statistical significance of the performance difference, we conduct one-sided hypothesis testing to compare RWSL and each of the two competing models. The result shows that the mean MSE of RWSL is significantly lower than the competing models for all scenarios except scenario 1 that is the most challenging small-sample setting. Furthermore, comparing RWSL with ridge regression, we can see that the gap between the two models is most significant in scenarios 2-6, which are the scenarios with limited labeled samples. This shows the advantage of RWSL due to its capability of incorporating ranked pairs to augment labeled samples in training. Moreover, comparing RWSL with RankSVM, we can observe that the gap between the two models gets bigger in later scenarios. This indicates that RankSVM has a performance limit even with a large number of ranked samples used in training, whereas RWSL does not suffer from this issue due to the inclusion of labeled samples.

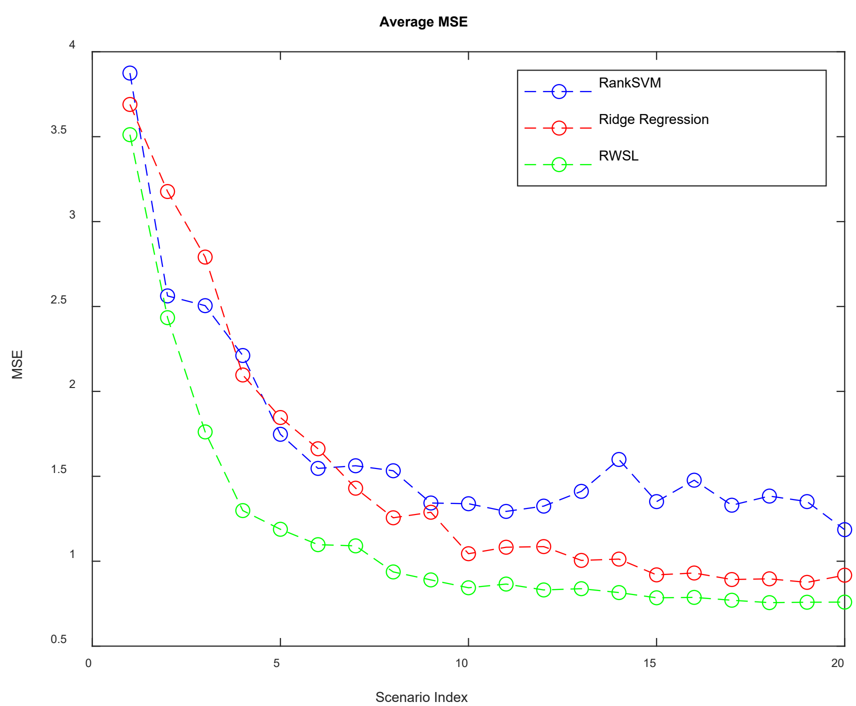

Figure 1. Average MSE comparison of three models on test data.

Furthermore, in order to compare not only the average but also the variability of MSE over different runs, we show the boxplot of MSE at each scenario for the three models in Figure 2. We can observe a faster drop in both the mean and variance of MSE for RWSL. We can also notice that RankSVM experiences higher variance and mean fluctuations. This is because RankSVM, by design, is a model to rank samples, instead of predicting a response variable y. When being used as a predictive model, the performance of RankSVM is less satisfying and is unstable.

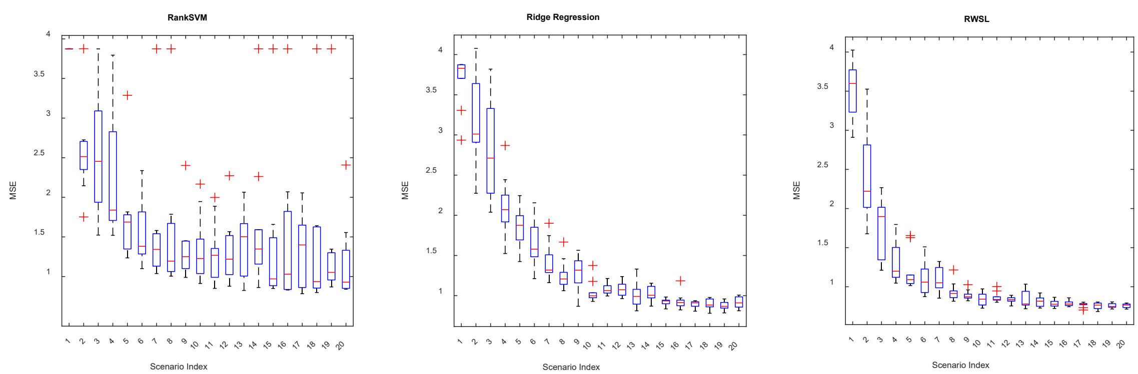

Figure 2. Boxplots of MSE on test data for three models.

Additionally, we compare the three models on other performance metrics such as predictive correlation and feature selection accuracy. Specifically, Figures 3 and 4 show average curve and boxplots of the correlation between true and predicted responses on the test set. Figures 5 and 6 show the average curve and boxplots of the feature selection accuracy. Similar observations to MSE can be drawn.

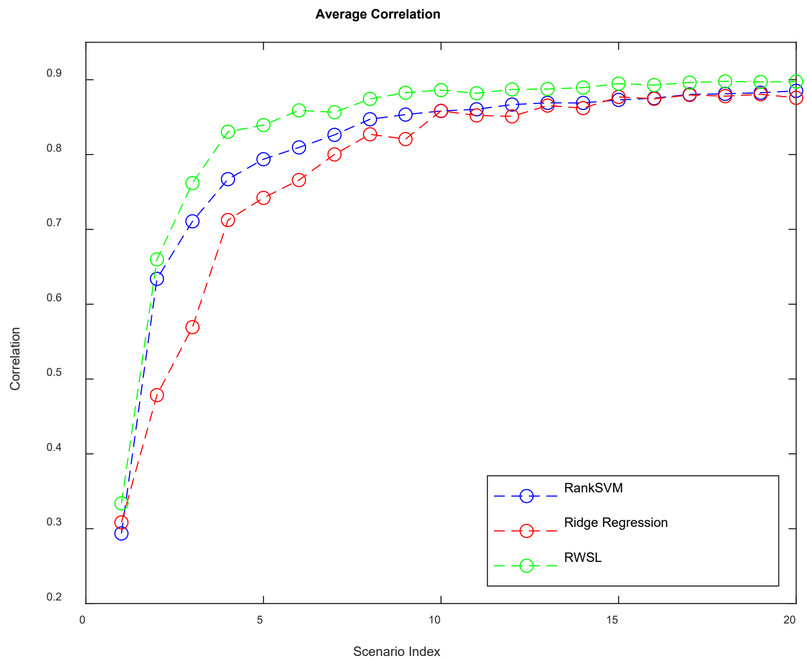

Figure 3. Average predictive correlation comparison of three models on test data.

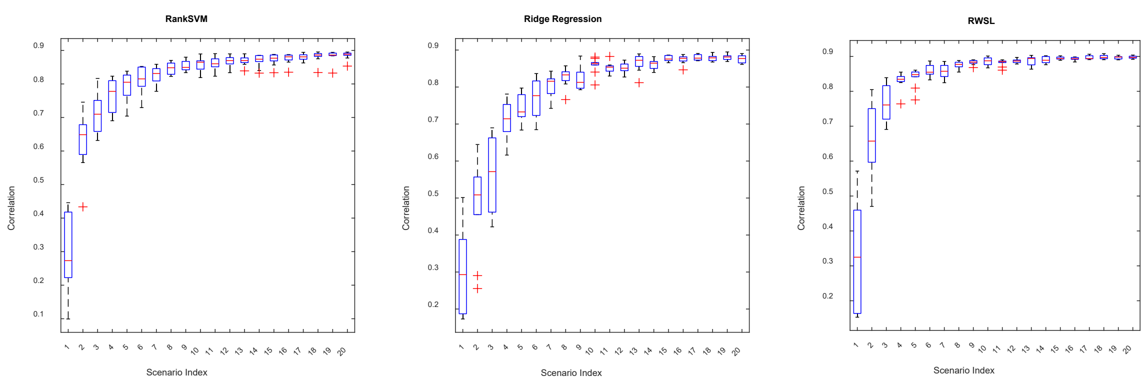

Figure 4. Boxplots of predictive correlation on test data for three models.

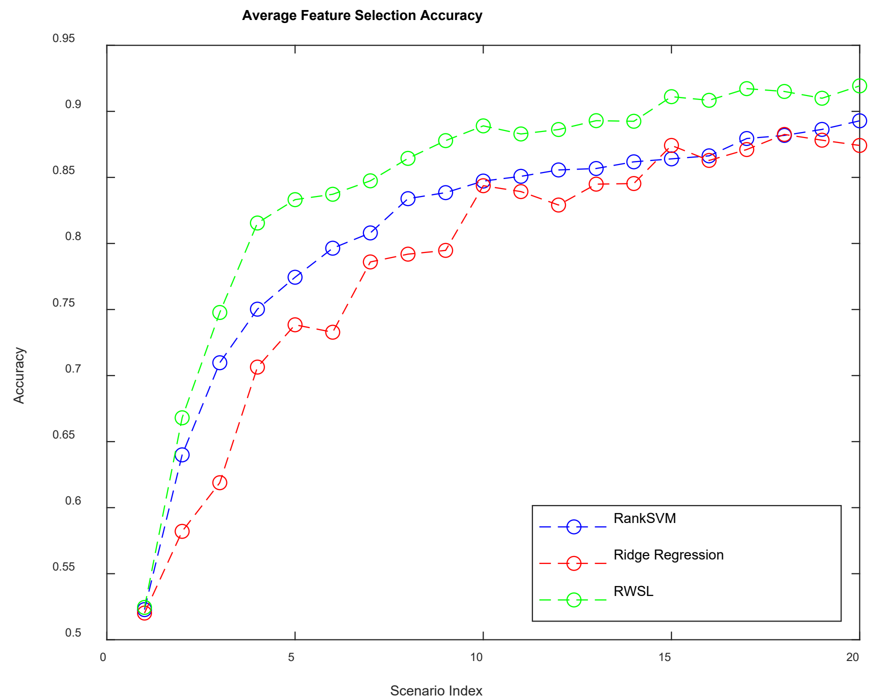

Figure 5. Average feature selection accuracy comparison of three models on test data.

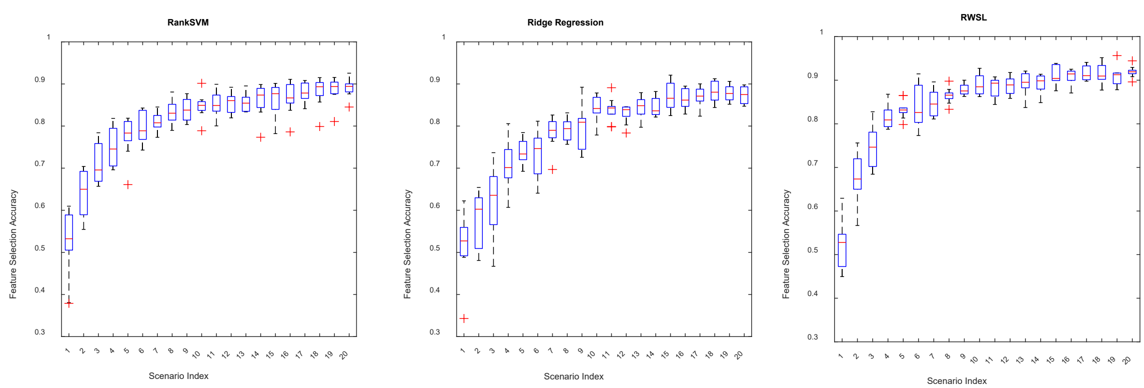

Figure 6. Boxplots of feature selection accuracy on test data for three models.

2.4.2 Saving of labeled samples by RWSL

The The findings in the previous section showed that RWSL can help us improve performance under a limited number of labeled samples compared with using a supervised learning model such as ridge regression. The improvement is achieved by incorporating ranked samples in training, which are typically easier to obtain than labeled samples. This result gives rise to an essential question of whether RWSL can achieve a desired performance by just increasing the number of ranked samples and fixing the labeled samples to a low level. In other words, it is desirable to learn how many labeled samples can be saved by adopting RWSL. To answer this question, we focus on scenario 8 in section 2.4.1, in which RWSL achieved an average MSE of 0.937 with a training set of 80 labeled samples and 375 ranked pairs. Ridge regression trained only based on 80 labeled samples achieved an average MSE of 1.256. To identify the number of additional labeled samples needed for ridge regression to reach the same MSE as RWSL, we vary the labeled samples from 80 to 150 with a step size of 10 and train ridge regression under each sample size. Figure 7 shows the MSE of ridge regression (red curve) in comparison with RWSL (blue curve).

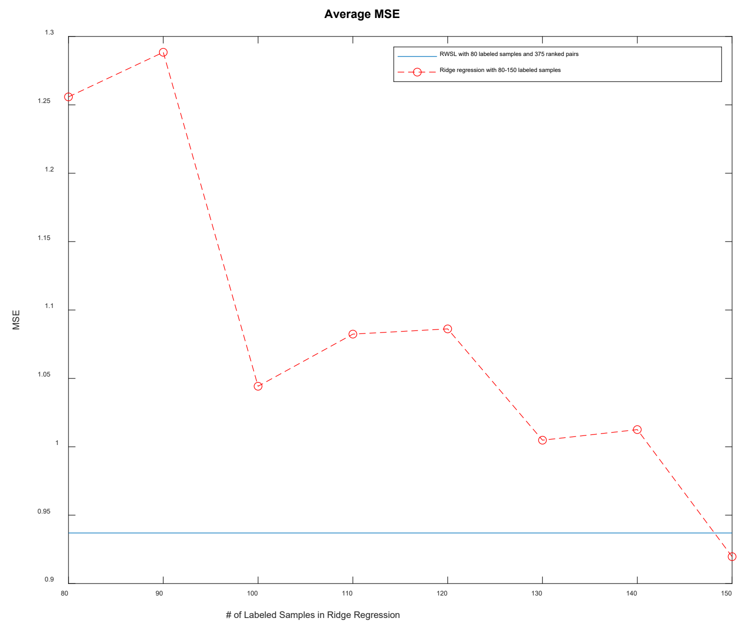

Figure 7. MSE performance of ridge regression with an increasing labeled sample size.

We can observe that the MSE of ridge regression drops to be lower than that of RWSL with 150 labeled samples in training. To account for statistical significance in comparing models, we conduct one-sided hypothesis testing to see when ridge regression has statistically equivalent MSE to RWSL. We find that the first time when the two models have statistically equivalent performance is when ridge regression includes 130 samples in training. This indicates that an increment of 130-80=50 labeled samples in ridge regression is equivalent to 375 ranked pairs in RWSL. If labeled samples are expensive to collect while ranked pairs cost substantially less, the saving of the cost in data collection by RWSL is a clear benefit.

2.4.3 Performance sensitivity with respect to ranked data difference

Ranked samples play an important role in training RWSL. However, two samples satisfying $y_i \geq y_j$ could have slightly or drastically different response variables. This experiment aims to evaluate how the difference between ranked samples, i.e., $|y_i - y_j|$, affects the performance of RWSL. Let $\Delta y = |y_i - y_j|$ denote a pre-selected threshold. Only ranked samples exceeding Δy are included in model training. We compare three settings with small, medium, and large Δy. The three settings are chosen relative to the value of the noise level in the data. The small Δy is smaller than 1σ of the noise, the medium Δy is around 3σ of the noise, and the large Δy is just above 6σ of the noise. Under each setting, ranked samples of a size between 150 and 900 are simulated, creating 15 scenarios, while the labeled samples are kept at 50 for all scenarios. Figure 8 shows the average MSE on test data under the three settings of Δy. We can see that the MSE performance with medium Δy is the best. The value of the medium Δy is around 3σ of the noise in the response variable. The practical implication of this finding is to provide some guidance on the effect of threshold selection. Underestimating the threshold will result in incorrectly ranked samples due to noise. On the other hand, overestimating the threshold will result in ignoring many valid ranked samples. Hence, the choice of the threshold will affect the type of ranked samples included in training RWSL and eventually affect the performance of the model.

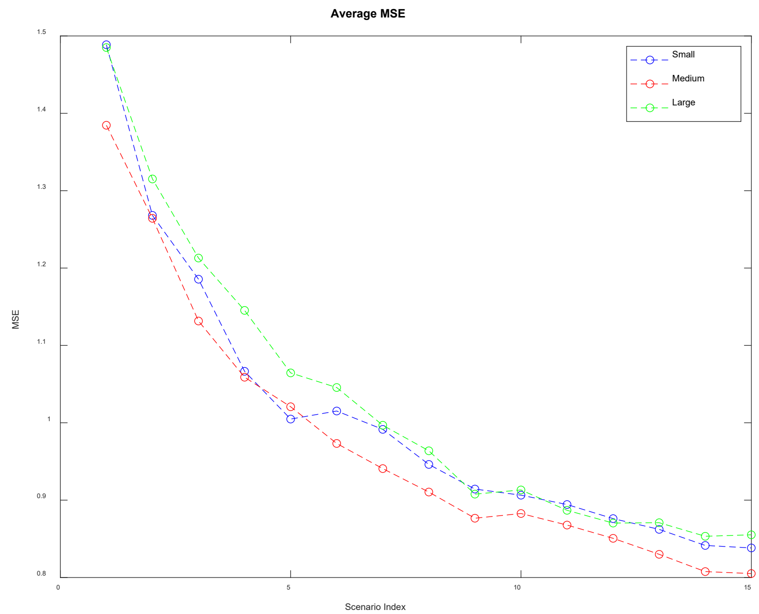

Figure 8. Average MSE comparison of RWSL under small, medium, and large ranked data difference.

2.4.4 Performance sensitivity with respect to labeled data noise

In many applications, it may be difficult to measure the response variable precisely. Thus, measurement errors or label noise in the training data are inevitable. Supervised learning models like ridge regression completely rely on labeled samples in training. As a result, they are susceptible to label noise. On the other hand, RWSL may be more robust as it can additionally include ranked data in training that is less susceptible to label noise. To verify this, we add label noise as a percentage of the variance of the response variable when simulating the training data. The test set is kept without noise for performance evaluation. We set the label noise percentage at 0%, 20%, 40%, and 60%. Under each setting, we compare the performance of RWTL with ridge regression. The results are shown in Figure 9. As the noise percentage increases, we can observe that the performance gap between

RWSL and ridge regression increases. Such a behaviour can be explained by the sole reliance of ridge regression on labeled samples in training. On the other hand, RWSL draws its performance from both labeled and ranked data, which results in high robustness against noise. Hence, under high noise scenarios, RWSL offers the ability to tolerate perturbations that might affect prediction performance.

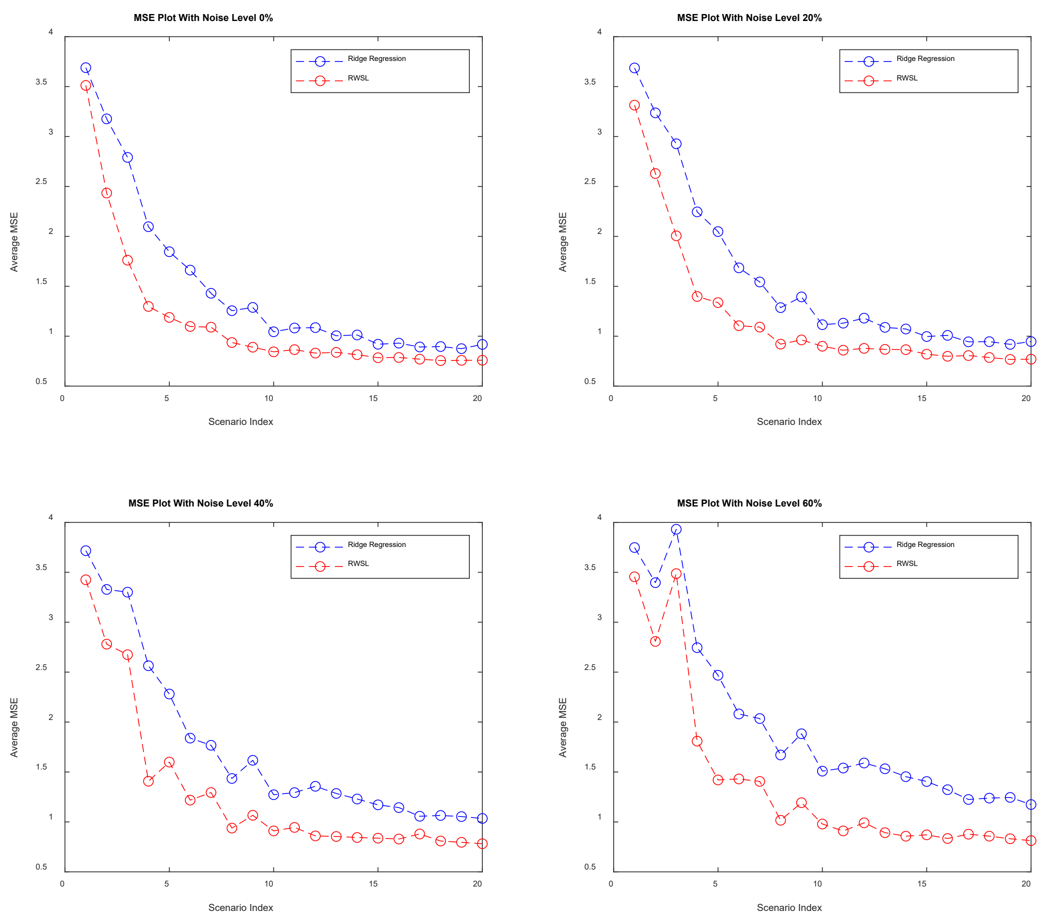

Figure 9. Average MSE comparison between RWSL and ridge regression under different levels of label noise.

2.5 Real data application

In this section, we will present an application of RWSL for predicting the MDS-UPDRS of PD patients using the tapping data collected by their mobile phones.

2.5.1 Data description

The data was collected by the mPower study (Bot et al., 2016). For each PD patient enrolled in the study, the patient was asked to use the mPower app installed on their mobile phone to perform a tapping activity three times a day: before medication, after medication, and another time. The tapping activity is pre-designed, which requires the patient to use two fingers from the same hand to alternatively tap two stationary points on the screen for 20 seconds. Figure 10 shows the interface on a mobile phone taken from the mPower app (Bot et al., 2016). The tapping activity aims to measure dexterity as assessed by speed, precision, and steadiness (Bot et al., 2016). Raw sensor data collected during a single session of performing the tapping activity is in the form of time series of the screen x-y coordinates on each tap. From the raw time series data, 43 features were extracted, such as the number of taps and the mean inter-tapping interval based on findings from previous studies (Taylor et al., 2005; Arora et al., 2015; Kassavetis et al., 2016). The 43 features are included in the RWSL model and histograms of some of those features can be seen in Figure 11.

In addition to providing tapping data, each participant of the mPower study was also requested to fill out the MDS-UPDRS questionnaire, which was collected on a less frequent basis (usually monthly). The MDS-UPDRS total score is used as the response variable for representing PD severity. MDS-UPDRS score ranges between 0 and 64, with 0 denoting a healthy response, 64 indicating complete disability.

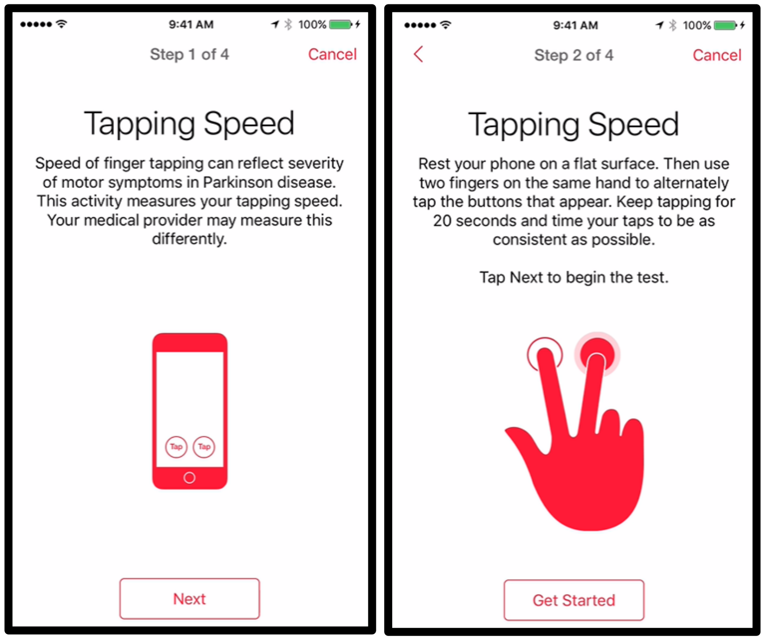

Figure 10. The mPower app instructs a patient to perform a tapping activity (Bot et al., 2016).

Among the PD participants in the mPower study, males comprise a higher proportion than females accounting for 65.8% of cases. Although an imbalance exists between sexes, the participants are well age-matched with respect to the ages at which males report disease onset (56.6 ± 9.6 years), showing no significant difference to that of females (56.2 ± 9.0 years) (Prince et al., 2018). Moreover, regarding the correlation between gender and tapping activity performance, no significant correlation was found in baseline performance between males (135.4 ± 61.3 taps) and females (133.9 ± 58.0 taps). Regarding the correlation between years since diagnosis and tapping performances, no significant correlations were found regarding baseline and steady-state performance (Prince et al., 2018). Lastly, the correlation between the age of the patient and the clinical severity assessed by the UPDRS scale was found to be weak and statistically insignificant ($p > 0.05$) (Taravari et al., 2014).

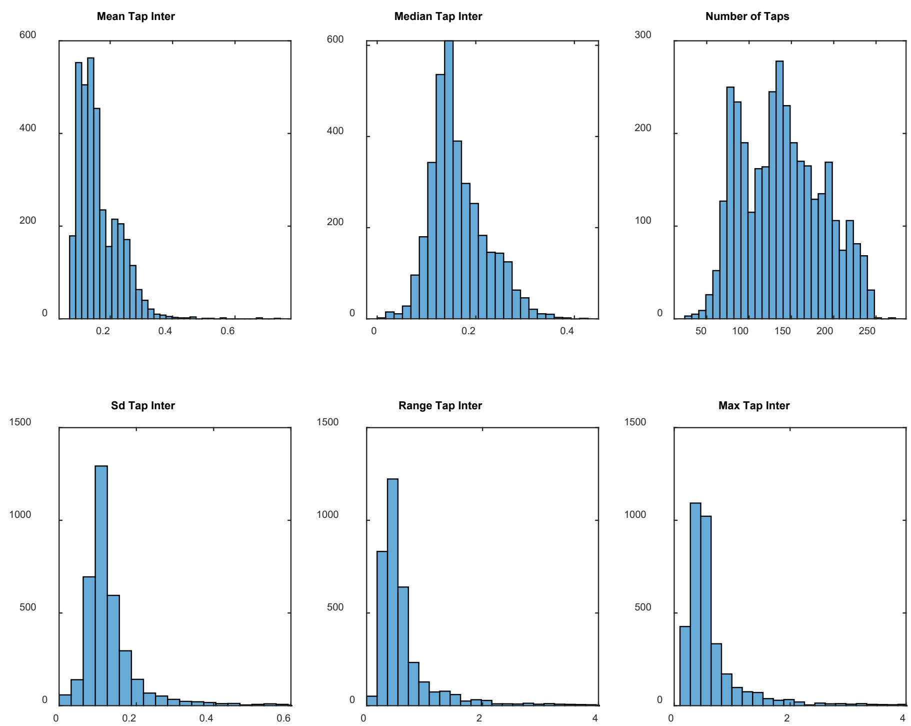

Figure 11. Histograms of some of the tapping features

2.5.2 Composition of ranked and labeled datasets

Since mPower is a mobile-based study, the number of participants is quite large. However, because the data is self-collected instead of being collected in a controlled environment, quality of the data collected from each patient varies significantly from one patient to another due to patient commitment and compliance. Therefore, to ensure a more reliable dataset, patient selection needs to be conducted. Our analysis is restricted to patients who performed at least 30 tapping tasks before medication, and 30 or more tasks after medication, and we have 57 such patients. Among those patients, we chose 16 patients who showed strong statistical evidence for medication response based on hypothesis testing suggested by the literature (Chaibub Neto et al., 2016). From these patients, there are a total of 3711 tapping activity records before and after medication. Finally, the ranked dataset is composed of before- and after- medication information within a 3-day window, producing 1284 valid ranked pairs.

Labeled samples are those with MDS-UPDRS measurements available. There is substantial missing data of MDS-UPDRS due to lack of commitment from participants. A possible solution to this problem is to interpolate the data with missing values. According to the PD literature (Tsanas et al., 2009), a linear trend of MDS-UPDRS as PD progresses is the most plausible trend. Hence, we linearly interpolate the MDS-UPDRS to create weekly measurements, ultimately obtaining 168 labeled samples

2.5.3 Model training and performance evaluation

We train RWSL, RankSVM, and ridge regression using 5-fold cross-validation (CV) and the CV accuracy of each model is reported in Table 1 using two metrics: MSE and predictive correlation. We can see that RWSL produces the smallest MSE and the highest predictive correlation, and the smallest standard deviation in both metrics demonstrating good stability of the algorithm.

Table 1. CV accuracy of MDS-UPDRS prediction based on tapping features.

Model	**RankSVM**	**Ridge regression**	**RWSL**
Predictive correlation: Mean (std)	0.7335 (0.141)	0.6913 (0.158)	**0.7579 (0.083)**
MSE: Mean (std)	0.0384 (0.016)	0.0342 (0.016)	**0.0276 (0.013)**

Among the 43 features included in training the RWSL, the coefficients of two features, median tapping interval and mean tapping interval, rank the top in magnitude. The absolute values of model coefficients are commonly interpreted in the regularized regression literature as crude feature importance scores (Tibshirani, 1996). The sum of absolute coefficients of these two features takes 37% of the sum of absolute total coefficients where we can observe their estimates in Table 2. This is consistent with the literature in which PD patients have been found to have a shorter inter-tap interval of finger tapping due to a lack of control in fine motor capabilities (Taylor Tavares et al., 2005; Roalf et al., 2018).

Table 2. Top coefficient estimates of RWSL.

Feature	Mean	Std
Median Tap Inter	0.6791	0.1490
Mean Tap Inter	0.7494	0.1572
Button Freq	-0.1244	0.0149
Sd Tap Inter	0.1539	0.0263
Skew Tap Inter	-0.1497	0.0219
Sd Drift Right	0.2667	0.0618

2.5.4 Practical implications

Telemonitoring is an emerging health care platform enabled by smartphones and wearables, which produces a tremendous amount of patient-specific health data. This chapter addresses the data science challenges in leveraging the telemonitoring platform to benefit patient care. Specifically, the proposed RWSL allows for both labeled and ranked samples to be integrated in training a robust model to predict the disease severity of patients with PD based upon mobile-collected tapping activity data. The model provides a step toward continuous disease monitoring of PD patients, thereby providing a tool for physicians to get insight into patients' real-life functioning, deliver timely interventions, reduce clinic visits, and facilitate improving of patients' life quality. Those benefits are of extreme interest, especially to health care practitioners and patients within less privileged healthcare systems under limited resources.

2.6 Conclusion

We proposed a new RWSL model that allows for both labeled and ranked samples to be integrated in training a predictive model. Simulation experiments showed that RWSL was superior to ridge regression and RankSVM, especially in scenarios of scarce labeled samples. RWSL was applied to the tapping activity data of patients with PD collected by their mobile phones and demonstrated good performance in predicting their disease severity.

Despite the promising features of our model, some limitations need to be addressed and open the doors for future research. First, PD condition/severity is a latent construct, it cannot be perfectly known no matter which clinical instrument is used or even by more comprehensive clinical assessment. In this chapter, we chose to use the MDS-UPDRS score as a surrogate measure for PD condition/severity because it has a long history of being a gold standard by neurologists for monitoring signs and symptoms of PD, and it has shown excellent internal consistency across multiple studies and across stages of disease severity as measured by the Hoehn and Yahr staging system (Goetz et al., 2008; Martinez-Martin et al., 1994; Louis et al., 1996). On the other hand, we acknowledge that MDS-UPDRS may not perfectly capture all aspects of the disease. It is expected that a more comprehensive PD severity scale will be developed by the medical society in near future driven by the advances of diagnostic instrument and medical sciences. By that time, RWSL can be re-trained to predict the new severity scale using mobile activity data to allow for more accurate and reliable prediction of PD severity for each patient.

Furthermore, the telemonitoring data collection process is patient self-administered. Hence, there is missing data due to non-adherence and mistakes. This raises questions that are worth exploring in the future about the best data imputation and pre-processing techniques that can mitigate the effect of these occurrences in the data. Also, related to the specific structure of our model, RWSL relies only on medication as the criteria to create ranked samples, and it is worth studying for including other PD-related criteria to create ranked samples.

Finally, it is worth mentioning that there are practical issues that need to be addressed before our model can be implemented in clinical practice. For example, the mPower dataset we used in this chapter was created by a research study, whose quality is relatively higher than the data from the general patient population. It remains a challenge about how to improve user compliance for using the app and generate quality-assured tapping activity data for each patient. Furthermore, RWSL predicts disease severity of each patient based on their mobile-collected datasets. While the information may be useful to support clinical decision, this tool needs to gain physicians' trust before they are willing to adopt it. To this end, extensive clinical validations are needed. Also, since the data is collected by patients' mobile phones, substantial efforts would be needed to ensure cyber security so that the data is not maliciously altered, misplaced, or misused.

CHAPTER 3. PHYSICS-INFORMED WEAKLY-SUPERVISED LEARNING FOR QUALITY PREDICTION OF MANUFACTURING PROCESSES

3.1 Introduction

In manufacturing processes, a multitude of sensors are deployed to collect data of process parameters. This provides an opportunity to better predict and control the quality of the final product. However, the relationship between the process variables and the desired final quality is often not well understood. To establish this relationship, machine learning (ML) models can be built based on historical datasets which contain the final quality of each product (Y) and the process variables involved in making the product (X), known as labeled datasets. However, it can be challenging to obtain the labels of products, which often involve comprehensive end-of-line quality testing. This can be costly and often infeasible for all products in the case of destructive testing techniques. As a result, the labeled samples used to train ML models are limited. This challenging situation is intensified by the fact that industrial processes are inherently noisy. Without sufficient data, it would be difficult to build robust models and consequently model-based quality prediction would have poor performance.

Recently, there has been some research interest in integrating physics-based models with data-driven ML models to improve the predictive performance (Chakravarti et al., 2021; Chen et al., 2021; Cheng & Fu, 2022; Erichson et al., 2019; Guo et al., 2022; Herrero-Garcia et al., 2022; Kapusuzoglu & Mahadevan, 2020; Karniadakis et al., 2021; Leonesio

& Fagiano, 2022; Montes et al., 2021; Nabian & Meidani, 2020; Raissi et al., 2019; Sacks & Welch, 1989; Wang et al., 2022). Physics-based models exploit known information from physical knowledge to understand the behavior of a manufacturing process. Such models are commonly implemented using either mathematical equations or computer simulation, both based on physical laws and principles. These models are often more cost-effective than obtaining labeled samples via end-of-line quality testing.

Here we provide a motivating example for a spot-welding process shown in Fig. 12. Destructive testing is still the most common method to test spot weld properties with the two most common testing methods being peel and chisel tests (International Organization for Standardization, 2006). However, these methods are time-consuming, costly, and infeasible in continuous production. To avoid destroying samples, physics-based models have been used to predict final quality features related to the spot-welding process. For instance, consider the case that the final quality we are interested in predicting is temperature difference of the weld $[\Delta T]$. Process variables in the welding process include force [kN], current [kA], time of weld [s], and mass [g]. A physics-based model will utilize mathematical formula from physics to produce a prediction for the final quality, temperature difference. The input of the mathematical formula includes some of the process variables such as current, time of weld, and mass, as well as parameters related to fundamental mechanical principles such as elastic-plastic part deformation and thermal expansion (Zhou et al., 2018). The output from the mathematical formula is an estimate of the true temperature difference of the weld $[\Delta T]$.

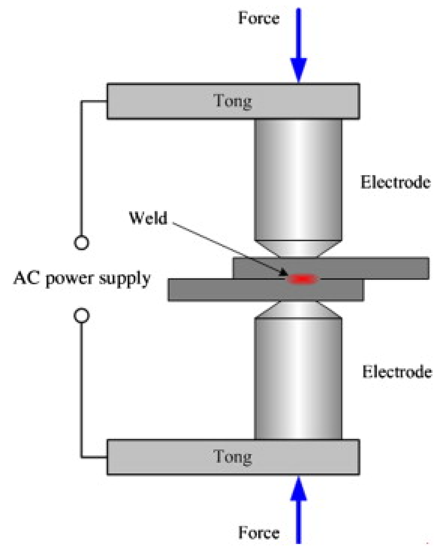

Figure 12. Spot welding process.

Physics-based models can be utilized to provide an estimate of the final quality of the product, which will mitigate the shortage of labeled samples for building data-driven ML models. However, physics-based models have limitations. First, it is typical that physics-based models can only take a small number of process variables as input. Thus, there may be a discrepancy between the output of physics-based models and the product quality. Also, physics-based models need to include some parameters related to the physical principles that are not observable. To estimate the unobservable parameters, a technique referred to as model calibration is needed (Kennedy & O'Hagan, 2001). However, the calibration process may be inadequate because it often relies on oversimplified models and flawed assumptions, which can lead to inaccurate predictions (Oliver & Alfonzo, 2018). Hence, a physics-based model can be inexact in characterizing the process-quality relationship, and thus may not be used as a stand-alone model.

Integrating physics-based models with data-driven ML models presents several challenges:

(1) Uncertainty in final quality estimates: Physics-based models are often subject to various sources of uncertainty, requiring model discrepancy to be addressed before using the model.

(2) Indirect measurement of final quality: The output of a physics-based model may be highly correlated with the final quality but may not directly measure it in some cases.

(3) Calibration issues: Physics-based models need to be calibrated, but calibration is often performed using ad-hoc procedures rather than optimization.

Several studies in the literature have attempted to incorporate physics knowledge into their frameworks, yet none have fully tackled the challenges mentioned earlier. Some involve integrating physics knowledge into deep learning models (Chen et al., 2021; Cheng & Fu, 2022; Erichson et al., 2019; Guo et al., 2022; Kapusuzoglu & Mahadevan, 2020; Wang et al., 2022), many of which find applications in additive manufacturing (Guo et al., 2022; Kapusuzoglu & Mahadevan, 2020; Wang et al., 2022). For instance, one method leverages dimensionality augmentation to incorporate information derived from the selective laser melting process into the model (Wang et al., 2022). Other strategies for incorporating physics knowledge into models include the use of physically-informed priors (Erichson et al., 2019), incorporating physics constraints within the loss function (Kapusuzoglu & Mahadevan, 2020), utilizing outputs from physics models as additional inputs (Kapusuzoglu & Mahadevan, 2020), and pre-training with input-output data from physics models (Kapusuzoglu & Mahadevan, 2020). Despite the innovative approaches, a pervasive issue remains: these strategies do not directly address data scarcity, a significant concern given the data-intensive nature of deep learning models (Chen et al., 2021; Cheng

& Fu, 2022; Erichson et al., 2019; Guo et al., 2022; Kapusuzoglu & Mahadevan, 2020; Wang et al., 2022). Furthermore, they overlook the calibration challenges associated with physics models. While some research has attempted to tackle the calibration issue through a two-stage approach (Chen et al., 2021), it remains isolated and not fully integrated with the overall prediction process.

Another area of research involves integrating physics knowledge through semi-supervised learning techniques. One common approach is utilizing large databases of unlabeled physics events and focusing on uncovering data patterns and correlations to facilitate label spreading (Montes et al., 2021). Another approach focuses on utilizing simulations based on established physical theories and principles to create a dataset that improves training (Chakravarti et al., 2021; Herrero-Garcia et al., 2022; Leonesio & Fagiano, 2022). For example, Herrero-Garcia et al. (Herrero-Garcia et al., 2022) pre-trained their model using simulations from established physics theories to represent known physical phenomena in dark matter, then trained on real experimental data. Moreover, Chakravarti et al. (Chakravarti et al., 2021) presented a method that leverages a known physics model to generate a background dataset and then uses a classifier to differentiate between this background data and the experimental data. Semi-supervised learning methodology shows promising results in addressing data scarcity issues; however, these approaches heavily rely on the quality and representativeness of the simulated data from the physical model. Moreover, such approaches do not address the challenges central to physics-based models, such as uncertainty in final estimates and calibration issues.

To address the challenges, we draw upon ideas from the field of weakly-supervised learning in ML (Zhou, 2018). Weakly-supervised learning deals with scenarios where it is difficult to obtain a large number of samples with precise labels to train an ML model, but it is relatively easy to obtain unlabeled samples to provide weak supervision for the ML model training. We consider the output from physics-based models as providing weak labels for samples. Driven by this idea, we propose a novel framework for Physics-informed Weakly-supervised Learning (PWL) in this chapter. The novelties of PWL are summarized as follows:

(1) PWL utilizes a data-driven approach to handle the uncertainty and inadequacy of physics-based models.
(2) PWL utilizes the outputs of physics-based model as weak labels, and integrates them with limited labeled data collected from the manufacturing process.
(3) PWL simultaneously optimizes the data-driven ML model as well as the discrepancy and calibration parameters of the physics-based model, thus achieving a better performance than each type of model used alone.

The rest of this chapter is structured as follows: Section 3.2 discusses related research and highlights the gaps in existing work, providing motivation for the development of PWL. Section 3.3 presents the model, algorithm, and theoretical property of PWL. Section 3.4 presents simulation experiments and comparison of PWL with existing methods. Section 3.5 provides two real-data case studies of manufacturing processes to demonstrate the real-world applications of PWL. Finally, Section 3.6 provides the conclusion.

3.2 Literature Review

The model we are proposing has significant connections with two established research fields: physics-informed ML and weakly-supervised learning. This section will present a summary of both research fields and highlight the unique aspects of our work.

3.2.1 Physics-informed ML

This research field aims to integrate physical laws, principles, and models with data-driven ML (Shi et al., 2016). Due to the variety of forms in which physics is represented in various application domains, a diverse collection of methods is needed to develop physics-informed ML models. These methods can be generally divided into two categories: i) physics-informed architecture ii) physics-informed regularization (Meng et al., 2022).

Physics-informed architecture is a recent research direction that can make use of specific characteristics of the problem being solved to implicitly encode physical consistency or other desired physical properties in designing the ML model. This area is focused largely on neural network architecture, due to their modular and flexible nature that makes them a prime candidate for architecture modification. For example, in lake temperature modeling, Daw et al. (Daw et al., 2020) incorporated a physical intermediate variable as part of a monotonicity-preserving structure. Similarly, in the paper by Muralidlar et al. (Muralidhar et al., 2020), physics-constrained variables were inserted as the intermediate variables in the convolutional neural network (CNN) architecture, which achieved significant improvement in predicting drag force on particle suspensions in moving fluids. Other research focuses on designing computational graphs that mimic the behavior of physics-based methods in preserving energy conservation laws (Cranmer et al., 2020; Greydanus et al., 2019; Lutter et al., 2019). Despite the promising features of physics-informed architecture, formulating a trainable architecture that represents the problem at hand is a challenging task (Guo et al., 2022). Moreover, even if an architecture is successfully designed, it is often highly application-specific. Hence, a new architecture needs to be formulated and trained for any change in the initial application conditions.

Physics-informed regularization could integrate physical understanding into the loss function of an ML model, which helps generate results that are consistent with established physical laws (Jia et al., 2019; Karpatne et al., 2017; Read et al., 2019). Pure data-driven ML can fail to produce such results directly from data, especially when provided with only limited samples. For instance, in the context of lake temperature level modelling, one approach was to include an extra physics-based penalty that guaranteed predictions of denser water were at reduced depths than those of much less thick water, which was a known monotonic relationship (Daw et al., 2022). Moreover, other approaches may attempt to create a regularization term that accounted for the underlying physics by penalizing divergence form the governing equations, such as Burgers' equation (Raissi et al., 2019). However, physics-informed regularization may not be feasible due to several challenges (Karniadakis et al., 2021):

(1) Mismatched input variables: Physics-informed regularization may not be feasible if the physics-based model and the ML model do not share the same input variables.
(2) Limited generalizability: Regularization with limited samples may restrict the model's generalizability, leading to performance deterioration.

(3) Incomplete or imprecise knowledge: It is essential to recognize that the knowledge from physics-based models may be incomplete or imprecise.

(4) Complexity of large-scale neural networks: Physics-informed ML models often involve training large-scale neural networks with complicated loss functions, which can be highly non-convex optimization problems.

(5) Competing loss function terms: The terms in the loss function may compete with each other during training, making the training process less robust and stable, and potentially preventing convergence to the global minimum.

An integration framework that is capable of integrating incomplete/imprecise physical knowledge with ML is highly desirable, which is the goal of this research.

3.2.2 Weakly-Supervised Learning

In ML, the models trained using only labelled samples are known as supervised learning models. In many applications, there may be samples that are not precisely labelled but can still be useful to provide some "weak" supervision for the ML model. This field is known as weakly-supervised learning. Research in weakly-supervised learning can be categorized based on the quality and characteristics of weakly labeled data to three main fields: inexact supervision, inaccurate supervision, and incomplete supervision (Zhou, 2018). Inexact supervision refers to situations where the labels or annotations provided are incomplete or noisy, meaning that they do not fully capture the desired concept or class (Dietterich et al., 1997; Foulds & Frank, 2010). For example, in image classification, an image might have multiple objects, but only one object is labelled (Chen & Wang, 2004). Inaccurate supervision, on the other hand, refers to situations where the provided labels or annotations are wrong. For example, an image of a dog might be labeled as a cat (Frénay & Verleysen, 2013). In the case of incomplete supervision, a portion of the training samples is labeled, with the remainder being unlabeled. This presents unique challenges as the algorithm must learn from both labeled and unlabeled data, often requiring innovative approaches to effectively leverage the unlabeled portion. Techniques like semi-supervised learning (Chapelle et al., 2006; Zhou & Li, 2010) and active learning (Settles, 2009) are commonly employed under incomplete supervision, aiming to maximize learning efficiency from limited labeled data. Active learning operates under the assumption of having an 'oracle', like a human expert, who can provide labels for chosen unlabeled samples. The objective of active learning methods is to identify the most informative unlabeled samples to send them to the 'oracle' for querying.

In the semi-supervised learning (SSL) approaches, the dependency on an external 'oracle' for guidance is eliminated. This approach combines the use of both labeled and unlabeled data to enhance model training in areas such as classification and prediction. Several SSL algorithms exist, with some based on the concept that data naturally forms clusters, suggesting that samples in the same cluster share similar class labels (Chapelle et al., 2006). Graph-based learning, a key technique in SSL, involves creating graphs where nodes are training samples and edges reflect their interrelations (Blum & Chawla, 2001; Fujino et al., 2005). Other strategies consider the labels for unlabeled data as simply missing, operating under the premise that all data, labeled or not, are derived from a unified model (Nigam et al., 2000). Furthermore, certain SSL methods focus on pinpointing boundaries in less densely populated data areas while ensuring the accurate classification of labeled data (Joachims, 1999; Chapelle & Zien, 2005; Li et al., 2013). However, the field of SSL, while extensive, has yet to fully explore incorporating physics-related information from unlabeled samples into model building.

Moreover, recent work in weak supervision capitalizes on domain knowledge to facilitate ranking unlabeled data samples, a strategy that has found considerable applicability in healthcare settings (F. Alenezi et al., 2022). Particularly, this methodology (F. Alenezi et al., 2022) hinges on utilizing expert insights to generate relative rankings based on anticipated reactions to treatments, such as medication effects in Parkinson's disease management, thus providing a binary label for a pair of unlabeled samples. However, this strategy finds limited applicability in the manufacturing sector because it lacks a parallel to the healthcare knowledge-driven ranked samples. Also, the manufacturing domain, governed rigidly by physical laws, demands a more structured and quantifiable approach to data labeling derived from established physical principles.

3.2.3 Gaps

The proposed PWL model integrates the output of the physics-based model with data-driven ML, while at the same time optimizing the physics-based model by estimating the discrepancy and calibration parameters of the physics-based model. In this way, the integration can be more effective. In comparison, the existing works in physics-informed ML integrate physical laws, principles, and models as they are, without optimizing them. In the field of weakly-supervised learning, none of the existing works consider weak supervision from the physics-based model. In this sense, PWL is also substantially different from the existing weakly-supervised models in ML.

3.3 Physics Informed Weak Learning (PWL)

In this section, we will present the development of PWL. Section 3.3.1 defines the problem setting and notations. Section 3.3.2 proposes the model formulation. Section 3.3.3 proposes an optimization problem for model estimation. Section 3.3.4 presents the algorithm for solving the optimization and the associated theoretical properties.

3.3.1 Notations and Problem Statement

Let $\mathcal{y}$ denote the output variable (i.e., product quality) of a manufacturing process. $\mathcal{y}$ is typically affected by a set of variables and parameters that are responsible for making the final product. In practice, we are interested in predicting the product quality $\mathcal{y}$ either empirically using a physics-based model or a data-driven ML model.

3.3.1.1 Physics-based Models and Limitations

Physics-based models depend on physical understanding of the system's input and output, and characterize the relationship between the input variables and the output variable as follows:

$$\mathcal{y}_i = \varphi[\boldsymbol{\eta}(\boldsymbol{x}_i^{\boldsymbol{ph}}, \boldsymbol{\theta})] + \varepsilon_i^{disc} + \varepsilon_i, \tag{7}$$

where $\boldsymbol{\eta}(\cdot,\ \cdot)\colon \mathbb{R}^{(q_1+q_3)} \to\ \mathbb{R}^z$ is the deterministic output from a physics-based model, φ is a mapping function that maps from the physics-based model output to the product quality, $\varepsilon_i^{disc}\ \in \mathbb{R}$ is an unknown model discrepancy term between the mapped physics-based model output and the product quality, $\boldsymbol{x}_i^{\boldsymbol{ph}} \in \mathbb{R}^{q_1}$ are input variables to the physics-based model, $\boldsymbol{\theta} \in \mathbb{R}^{q_3}$ are calibration parameters which cannot be directly controlled or measured in a real manufacturing process, such as material properties, boundary conditions, etc., $\varepsilon_i \in \mathbb{R}$ is an random error, and i is the index for samples such as products made by the manufacturing process.

Generally, physics-based models are a complex mathematical representation of the true system that are implemented through a computer model or a physical equation. The mathematical representation may not accurately emulate the process observations mainly due to two reasons:

First, application of physics-based models is accompanied by various sources of uncertainty. In our framework, we address two sources of uncertainty that significantly affect the performance of a physics-based model. The first source is commonly referred to as model discrepancy. Model discrepancy can be defined as the difference between the output of the real-world manufacturing process such as product quality and the output of the physics-based model at the same inputs' values. Hence, even if we know the true values of all the inputs required to generate an output from the physics-based model, this output may not be equal to the true output of the manufacturing process, thus creating a discrepancy denoted by ε_i^{disc} in Eq. 7. This discrepancy exists because of insufficient inputs. Physics-based models are built only on input variables which have a well-understood relationship with the model's output based on physical knowledge. Hence, it is likely that many variables in the manufacturing process cannot be included in the physics-based models.

Second, physics-based models have unknown parameters that need to be calibrated, denoted by $\boldsymbol{\theta}$ in Eq. 7. Calibration can be defined as the activity of adjusting $\boldsymbol{\theta}$ until the output of the physics-based model fits the observed data from the real manufacturing process (Kennedy & O'Hagan, 2001). In practice, calibration is typically used as an ad-hoc fitting in a pre-step. The fitted calibration parameter is substituted into the physics-based model to generate an output. This sequential approach may be sub-optimal due to inefficiencies in the two-step process, potential overfitting during ad-hoc fitting, incomplete parameter optimization, and a lack of joint optimization between the physics-based and ML models. An alternative approach that integrates both models more effectively can lead to better predictive performance and improved understanding of the manufacturing process (Raissi et al., 2019).

3.3.1.2 Data-driven ML Models and Limitations

Due to the advanced sensing capability of modern manufacturing systems, $\boldsymbol{x^{ph}}$ are not the only measurable variables in the system. Hence, we define another set of variables $\boldsymbol{x^{pr}} \in \mathbb{R}^{q_2}$ which represent measurable variables in a real system that are not included in the physics-based model. This motivates another modeling approach, data-driven models, to model the relationship between all measured variables in the system and $\mathcal{y}$ as follows:

$$\mathcal{y}_i = f\left(\boldsymbol{x_i^{ph}}, \boldsymbol{x_i^{pr}}\right) + \varepsilon_i \,, \tag{8}$$

where $f \in \mathbb{R}^{(q_1+q_2)} \rightarrow \mathbb{R}$ is an ML model to be trained based on observational data.

In practice, the performance of data-driven models is hindered by two main challenges. Firstly, the number of labeled samples can be very limited, making the model training less robust. Secondly, these models follow a pure data-driven approach without taking physical knowledge into consideration. In fact, we may consider physical knowledge to provide a form of "virtual/synthetic data" to help mitigate the shortage of labeled samples.

3.3.1.3 A Joint Model Perspective

We seek to integrate physics-based models and data-driven models in a unified framework to leverage their complementary power. To further highlight the significance of this approach, let us revisit our motivating example, the spot-welding process: The product quality variable y is weld temperature difference. The physics-based model $\boldsymbol{\eta}(\cdot,\ \cdot)$ is a mathematical equation that utilizes principles of thermal resistivity and conductivity. The inputs to the physics-based model, $\boldsymbol{x^{ph}}$, include the current magnitude and time of the weld. The unknown calibration parameters $\boldsymbol{\theta}$ in the physics-based model include thermal conductivity and resistivity coefficients. On the other hand, there are other variables in the process that are not included in $\boldsymbol{x^{ph}}$, i.e., $\boldsymbol{x^{pr}}$ which includes the force magnitude. A data-driven model can be built based on $\boldsymbol{x^{ph}}$ and $\boldsymbol{x^{pr}}$. It can be seen that neither model is sufficient by itself to produce an adequate predictive performance, because they neglect the fact that y is jointly influenced by the three sets of variables as follows:

$$y_i = u\left(\boldsymbol{x_i^{ph}}, \boldsymbol{x_i^{pr}}, \boldsymbol{\theta}\right) + \varepsilon_i\ , \tag{9}$$

where $u(\cdot,\ \cdot,\ \cdot): \mathbb{R}^{(q_1+q_2+q_3)} \rightarrow \mathbb{R}$ is the underlying unknown true process-quality relationship.

Models that are developed without taking Eq. 9 into consideration will lead to suboptimal solutions. PWL focuses mainly on the improvement and integration of the two modeling approaches, physics-based models and data-driven models, while respecting the assumption in Eq. 9.

3.3.2 Joint Model Formulation

It is difficult to directly formulate a joint model as in Eq. 9. We propose the following formulation:

$$y_i = \boldsymbol{H}(\boldsymbol{x}_i^{\boldsymbol{ph}}, \boldsymbol{\theta})\boldsymbol{g} + \boldsymbol{B}(\boldsymbol{x}_i^{\boldsymbol{pr}}, \boldsymbol{x}_i^{\boldsymbol{ph}})\boldsymbol{d} + \varepsilon_i\,, \tag{10}$$

where $\boldsymbol{H}: \mathbb{R}^{q_1+q_3} \to \mathbb{R}^p$ are basis functions for creating patterns associated with $\boldsymbol{x}_i^{\boldsymbol{ph}}$ and $\boldsymbol{\theta}$, $\boldsymbol{g} \in \mathbb{R}^p$ are coefficients vectors associated with the patterns created by $\boldsymbol{H}$; $\boldsymbol{B}: \mathbb{R}^{q_1+q_2} \to \mathbb{R}^b$ are basis functions for creating patterns associated with $\boldsymbol{x}_i^{\boldsymbol{pr}}$ and $\boldsymbol{x}_i^{\boldsymbol{ph}}$, $\boldsymbol{d} \in \mathbb{R}^b$ are coefficients vectors associated with the patterns created by $\boldsymbol{B}$. The basis matrix formed by $\boldsymbol{H}$ and $\boldsymbol{B}$ are used to represent patterns in the original feature matrix, and the coefficients $\boldsymbol{g}$ and $\boldsymbol{d}$ are estimated based on the patterns in $\boldsymbol{H}$ and $\boldsymbol{B}$, respectively. By using basis, the objective function aims to identify the most important patterns in the data for predicting the response variable such as product quality.

The structure of the model proposed in Eq. 10 offers three significant advantages. First, the model is structured in a way that it allows the first term to have the same input as the physics-based model to allow knowledge distillation. Knowledge distillation (Hinton et al., 2015) is a technique used in ML to transfer knowledge from a larger, more complex model (known as the teacher model) to a smaller, simpler model (known as the student model). This will help address labeled data shortage by allowing the model to additionally learn from the abundant weakly labeled samples produced by the physics-based model $\boldsymbol{\eta}(\cdot,$

$\cdot$). Second, the second term helps mitigate the discrepancy of using the physics-based model alone. Since this term is built upon both $\boldsymbol{x^{ph}}$ and $\boldsymbol{x^{pr}}$, it can leverage the data-driven model. Overall, the model in Eq. 10 combines physics-based and data-driven models together. Third, this is a highly flexible framework that can accommodate various types of basis functions. By choosing different basis functions, the model can be tailored to specific problem settings, allowing it to capture diverse patterns and relationships in the data. This adaptability makes the proposed framework suitable for a wide range of applications in manufacturing and other domains.

Notably, in the model in Eq. 10, the calibration parameters $\boldsymbol{\theta}$ are integrated with the model formulation, which makes it possible to optimize $\boldsymbol{\theta}$ together with other model parameters, instead of making calibration a pre-step.

3.3.3 Optimization

In order to estimate the model parameters and calibration parameters for the model in Eq. 10, we propose the following optimization:

$$\min_{\boldsymbol{g},\boldsymbol{\theta},\,\boldsymbol{d}} \quad \begin{gathered} \sum_{i=1}^{n} \left\| \mathcal{Y}_i - \boldsymbol{H}\left(\boldsymbol{x_i^{ph}},\, \boldsymbol{\theta}\right)\boldsymbol{g} - \boldsymbol{B}\left(\boldsymbol{x_i^{pr}}, \boldsymbol{x_i^{ph}}\right)\boldsymbol{d} \right\|_{\boldsymbol{F}}^{\boldsymbol{2}} + \\ \lambda_1 \sum_{j=1}^{N} \left\| \boldsymbol{H}\left(\boldsymbol{x_j^{ph}},\, \boldsymbol{\theta}\right)\boldsymbol{g} - \varphi[\boldsymbol{\eta}\left(\boldsymbol{x_j^{ph}},\, \boldsymbol{\theta}\right)] \right\|_{\boldsymbol{F}}^{\boldsymbol{2}} \\ + \lambda_2 \|\boldsymbol{g}\|_{\boldsymbol{1}} + \lambda_3 \sum_{\boldsymbol{q}=\boldsymbol{1}}^{(q_1+q_3)} \left\|\boldsymbol{g_q}\right\|_{\boldsymbol{2}} \end{gathered} \tag{11}$$

s.t.

$$\boldsymbol{\omega_l} \leq \boldsymbol{\theta} \leq \boldsymbol{\omega_u}\,,$$

where $\boldsymbol{\omega_l} \in \mathbb{R}^{q_3}$ and $\boldsymbol{\omega_u} \in \mathbb{R}^{q_3}$ are lower and upper bounds of the calibration parameters $\boldsymbol{\theta}$. N is the number of samples generated by the physics-based model, while n is the number of labeled samples collected from the manufacturing system. N is generally much larger than n. Next, we explain the purpose of each term in Eq. 11:

(1) The first term comes from Eq. 10, which aims to minimize the prediction error on the product quality variable, which is the overall goal of the framework.

(2) The second term has two purposes. The first purpose is to regularize how much the $\boldsymbol{H}(\boldsymbol{x}_i^{ph}, \boldsymbol{\theta})\boldsymbol{g}$ learns from the physics-based model $\boldsymbol{\eta}(\cdot, \cdot)$. The amount of regularization is dependent on the accuracy of the physics-based model. If the physics-based model is inaccurate, regularization should be relatively weak, which can be realized by a smaller λ_1. The second purpose is to handle the mismatch between the quality variable y_i and the physics-based model output $\boldsymbol{\eta}(\cdot, \cdot)$. That is, they can be highly correlated but not necessarily measure the same thing. Hence, we introduced the φ function.

(3) The third term is the L_1 penalty, which encourages sparsity at the individual coefficient level by applying an L_1 norm penalty to all the coefficients in $\boldsymbol{g}$. This penalty term helps to identify the most important individual variables for predicting the outcome.

(4) The fourth term is a group lasso penalty, which encourages group sparsity by applying an L_2 norm penalty to each group of coefficients $\boldsymbol{g_q}$ separately and then summing over all the groups. This penalty term helps to identify the most important groups of variables for predicting the outcome.

(5) Together, the third and fourth terms form a combination of group selection and variable selection methods, known as the sparse group lasso. The parameters λ_2 and λ_3 control the trade-off between group sparsity and individual sparsity, respectively. This is required since we are generating patterns related to the $\boldsymbol{x}_i^{ph}$ and $\boldsymbol{\theta}$ using the basis function $\boldsymbol{H}$. Some of those patterns are related to the same phenomena, and hence we desire their effects to be chosen or unchosen together. Moreover, the patterns generated by $\boldsymbol{H}$ are not necessarily independent, and hence this penalty can help with identifiability issues in our model which will be discussed in more details in Section 3.3.4.3.

The optimization model in Eq. 11 jointly optimizes the physics-based model and data-driven model to achieve better predictive performance. Also, the model optimizes the calibration parameters $\boldsymbol{\theta}$ in the physics-based model in the integrated framework. The proposed model in Eq. 11 is clearly non-convex and highly complicated to optimize. The algorithms developed to optimize the model will be discussed in detail in Section 3.3.4.

3.3.4 Algorithm

The problem in Eq. 11 is not convex. However, it can be solved through a Blockwise Coordinate Descent (BCD) algorithm, where we iteratively update the $\boldsymbol{g}, \boldsymbol{\theta}$, and $\boldsymbol{d}$ as shown in Algorithm 1.

Algorithm 1. The PWL algorithm

Input: $\boldsymbol{H}(.,.), \boldsymbol{B}(.,.), \varphi(.), \varepsilon_0$

Initialize: $\boldsymbol{d} = \boldsymbol{d}^{\boldsymbol{0}}, \boldsymbol{\theta} = \boldsymbol{\theta}^{\boldsymbol{0}}$

While $\left(\left[y_i - \left(\boldsymbol{H}\left(\boldsymbol{x}_i^{ph},\, \boldsymbol{\theta}\right)\boldsymbol{g} - \boldsymbol{B}\left(\boldsymbol{x}_i^{pr}, \boldsymbol{x}_i^{ph}\right)\boldsymbol{d}\right)\right]^2 \geq \varepsilon_0\right)$

1) Update $\boldsymbol{g}$ according to ADMM algorithm (Section 3.3.4.1.)
2) Update $\boldsymbol{d}$ as:

$$\min_{\boldsymbol{d}} \sum_{i=1}^{n} \left\| y_i - \boldsymbol{H}\left(\boldsymbol{x}_i^{ph},\, \boldsymbol{\theta}\right)\boldsymbol{g} - \boldsymbol{B}\left(\boldsymbol{x}_i^{pr}, \boldsymbol{x}_i^{ph}\right)\boldsymbol{d} \right\|_{\boldsymbol{F}}^{\boldsymbol{2}}$$

3) Update $\boldsymbol{\theta}$ according to ADMM algorithm (Section 3.3.4.2.)

End

Even though the problem in Eq. 11 is not convex, the three updates in Algorithm 1 are convex. In the PWL algorithm, there are two intuitive considerations about the initialization. First, we want to initialize $\boldsymbol{d}$ using small values to allow $\boldsymbol{g}$ to learn from the physics-based model in the first update. Second, we want to initialize $\boldsymbol{\theta}$ to give the best possible performance of the physics-based model $\boldsymbol{\eta}(\cdot,\ \cdot)$; hence it will be initialized according to our physics-based model calibration approach that to be discussed in Section 3.4. The $\boldsymbol{d}$ update has a close-form solution since we can cast it as a least square problem. On the other hand, the $\boldsymbol{g}$ and $\boldsymbol{\theta}$ updates are not as simple and will be discussed in detail in Sections 3.3.4.1. and 3.3.4.2., respectively. Lastly, we will discuss properties of our algorithm related to identifiability and uniqueness in Section 3.3.4.3.

3.3.4.1 *The* $\boldsymbol{g}$ *Update*

In the $\boldsymbol{g}$ update, the problem in Eq. 11 reduces to solving the following problem:

$$\min_{\boldsymbol{g}} \begin{array}{c} \sum_{i=1}^{n} \left\| y_i - \boldsymbol{H}\left(\boldsymbol{x}_i^{ph},\, \boldsymbol{\theta}\right)\boldsymbol{g} - \boldsymbol{B}\left(\boldsymbol{x}_i^{pr}, \boldsymbol{x}_i^{ph}\right)\boldsymbol{d} \right\|_{\boldsymbol{F}}^{\boldsymbol{2}} + \\ \lambda_1 \sum_{j=1}^{N} \left\| \boldsymbol{H}\left(\boldsymbol{x}_j^{ph},\, \boldsymbol{\theta}\right)\boldsymbol{g} - \varphi\left[\boldsymbol{\eta}\left(\boldsymbol{x}_j^{ph},\, \boldsymbol{\theta}\right)\right] \right\|_{\boldsymbol{F}}^{\boldsymbol{2}} + \\ \lambda_2 \|\boldsymbol{g}\|_{\boldsymbol{1}} + \lambda_3 \sum_{q=1}^{(q_1+q_3)} \left\|\boldsymbol{g}_q\right\|_{\boldsymbol{2}}\ . \end{array} \tag{12}$$

The problem in Eq. 12 is a sum of convex functions and hence convex. Assuming the basis functions $\boldsymbol{H}\left(\boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{ph}},\ \boldsymbol{\theta}\right), \boldsymbol{B}\left(\boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{pr}}, \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{ph}}\right)$ and $\boldsymbol{H}\left(\boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{ph}},\ \boldsymbol{\theta}\right)$ were selected, to facilitate the estimation procedure, we represent them in the matrix form as $\boldsymbol{H}_{\boldsymbol{1}} \in \mathbb{R}^{n\times p}$, $\boldsymbol{B}_{\boldsymbol{1}} \in \mathbb{R}^{n\times b}$ and $\boldsymbol{H}_{\boldsymbol{2}} \in \mathbb{R}^{N\times p}$, respectively. The problem in Eq. 12 is reformulated as follows:

$$\underset{\boldsymbol{z}}{\boldsymbol{min}} \sum_{m=1}^{M} f_m(\boldsymbol{z}), \tag{13}$$

where $M = 4,\ f_1(\boldsymbol{g}) = \|\mathbf{Y} - \boldsymbol{H}_{\boldsymbol{1}}\boldsymbol{g} - \boldsymbol{B}_{\boldsymbol{1}}\boldsymbol{d}\|_{\boldsymbol{F}}^{\boldsymbol{2}},$ $f_2(\boldsymbol{g}) = \lambda_1\|\boldsymbol{H}_{\boldsymbol{2}}\boldsymbol{g} - \widetilde{\boldsymbol{\eta}}\|_{\boldsymbol{F}}^{\boldsymbol{2}},$

$$f_3(\boldsymbol{g}) = \lambda_2\|\boldsymbol{g}\|_{\boldsymbol{1}},\quad f_4(\boldsymbol{g}) = \lambda_3 \textstyle\sum_{q=1}^{(q_1+q_3)} \left\|\boldsymbol{g}_{\boldsymbol{q}}\right\|_{\boldsymbol{2}}, \widetilde{\boldsymbol{\eta}} = \begin{bmatrix} \varphi[\boldsymbol{\eta}\left(\boldsymbol{x}_{\boldsymbol{1}}^{\boldsymbol{ph}},\ \boldsymbol{\theta}\right)] \\ \vdots \\ \varphi[\boldsymbol{\eta}\left(\boldsymbol{x}_{\boldsymbol{N}}^{\boldsymbol{ph}},\ \boldsymbol{\theta}\right)] \end{bmatrix},\ \mathbf{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}.$$

An effective and efficient algorithm to solve the problem in Eq. 13 can be developed using the Alternating Direction Method of Multipliers (ADMM) consensus algorithm (Parikh & Boyd, 2014), summarized in Algorithm 2 below. In order to apply the ADMM consensus algorithm to this problem, we first derive the proximal operator for the objective function, which involves computing the optimal solution for each variable while keeping the other variables fixed. The proximal operator is a key component in the ADMM framework and is crucial for the convergence and stability of the algorithm. By incorporating the proximal operator into the ADMM consensus algorithm, we can efficiently solve the optimization problem in Eq. 13 while ensuring a robust and stable solution.

Algorithm 2. The ADMM consensus algorithm

Input: step size Δ, ϵ

Initialize: replicates $\boldsymbol{z}_{\boldsymbol{m}}^{(0)} = \mathbf{0}, \boldsymbol{u}_{\boldsymbol{m}}^{(0)} = \mathbf{0}, \bar{\mathbf{z}} = \mathbf{0}$, for $m = 1, \dots, M,$ of the same shape as z.

For $k = 1,2, \dots$ **do**

Update $\boldsymbol{z}_{\boldsymbol{m}}^{(k)} = \operatorname{prox}_{\Delta f_m}\left[\bar{\mathbf{z}}^{(k-1)} - \boldsymbol{u}_{\boldsymbol{m}}^{(k-1)}\right]$, for $m = 1, \dots, M,$ in parallel.

$$\bar{\mathbf{z}}^{(k)} = \sum_{m=1}^{M} \frac{\boldsymbol{z}_m^{(k)}}{M}$$

$$\boldsymbol{u}_{\boldsymbol{m}}^{(k)} = \boldsymbol{u}_{\boldsymbol{m}}^{(k-1)} + \boldsymbol{z}_{\boldsymbol{m}}^{(k)} - \bar{\mathbf{z}}^{(k)}$$

If $\left|\bar{\mathbf{z}}^{(k)} - \bar{\mathbf{z}}^{(k-1)}\right| < \epsilon$

Stop

End

End

Even To implement Algorithm 2 in solving Eq. 13, we need to evaluate the proximal operators of $\Delta f_1, \dots \Delta f_M$**.** The results are in Proposition 1, and the derivation is in Section A.1 in the Appendix A.

Proposition 1. *Let* $f_1(\boldsymbol{g}) = \|\boldsymbol{Y} - \boldsymbol{H_1 g} - \boldsymbol{B_1 d}\|_{\boldsymbol{F}}^{\boldsymbol{2}}$, $f_2(\boldsymbol{g}) = \lambda_1 \|\boldsymbol{H_2 g} - \tilde{\boldsymbol{\eta}}]\|_{\boldsymbol{F}}^{\boldsymbol{2}}$, $f_3(\boldsymbol{g}) = \lambda_2 \|\boldsymbol{g}\|_{\mathbf{1}}$, *and* $f_4(\boldsymbol{g}) = \lambda_3 \sum_{q=1}^{(q_1+q_3)} \left\|\boldsymbol{g}_{\boldsymbol{q}}\right\|_{\mathbf{2}}$. *The proximal operators of* $\Delta f_1, \dots \Delta f_4$ *are given as follows: Let* $\mathbf{z}$ *and* $\boldsymbol{a}$ *have the same size as* $\boldsymbol{g}$.

1) *If* $\boldsymbol{z} = \operatorname{prox}_{\Delta f_1}[\boldsymbol{a}]$, *we have*

$\operatorname{prox}_{\Delta f_1}[\boldsymbol{a}] = \left(\mathbf{I} + \Delta \boldsymbol{H_1}^{\top} \boldsymbol{H_1}\right)^{-\mathbf{1}} \left(\boldsymbol{a} + \Delta \boldsymbol{H_1}^{\top} (\mathbf{Y} - \boldsymbol{B_1 d})\right)$, *where* $\mathbf{I}$ *is the identity matrix of order p.*

2) *If* $\boldsymbol{z} = \text{prox}_{\Delta f_2}[\boldsymbol{a}]$, *we have*

$\text{prox}_{\Delta f_2}[\boldsymbol{a}] = \left(\mathbf{I} + \lambda_1 \Delta \boldsymbol{H_2}^{\top} \boldsymbol{H_2}\right)^{-\mathbf{1}} \left(\boldsymbol{a} + \lambda_1 \Delta \boldsymbol{H_2}^{\top} \widetilde{\boldsymbol{\eta}}\right)$, *where* **I** *is the identity matrix of order p.*

3) *If* $\boldsymbol{z} = \text{prox}_{\Delta f_3}[\boldsymbol{a}]$, *we have*

$$\text{prox}_{\Delta f_3}[\boldsymbol{a}] = S_{\lambda_2 \Delta}(\boldsymbol{a}) = \begin{cases} \boldsymbol{a} - \boldsymbol{\lambda}_2 \Delta, \boldsymbol{a} > \boldsymbol{\lambda}_2 \Delta \\ \mathbf{0}, -\boldsymbol{\lambda}_2 \Delta \leq \boldsymbol{a} \leq \boldsymbol{\lambda}_2 \Delta \\ \boldsymbol{a} + \boldsymbol{\lambda}_2 \Delta, \boldsymbol{a} \leq \boldsymbol{\lambda}_2 \Delta\,. \end{cases}$$

4) *If* $\boldsymbol{z} = \text{prox}_{\Delta f_4}[\boldsymbol{a}]$, *we have*

$$\text{prox}_{\Delta f_4}[\boldsymbol{a}] = \left[\mathbf{1} - \frac{\boldsymbol{\lambda}_3 \Delta}{\|\boldsymbol{a}\|_2}\right]_+ \boldsymbol{a}\,.$$

3.3.4.2 The $\boldsymbol{\theta}$ Update

In the $\boldsymbol{\theta}$ update, the problem in Eq. 11 reduces to solving the following problem:

$$\min_{\boldsymbol{\theta}} \begin{array}{c} \sum_{i=1}^{n} \left\| \mathcal{y}_i - \boldsymbol{H}(\boldsymbol{x}_i^{\boldsymbol{ph}}, \boldsymbol{\theta}) \boldsymbol{g} - \boldsymbol{B}(\boldsymbol{x}_i^{\boldsymbol{pr}}, \boldsymbol{x}_i^{\boldsymbol{ph}}) \boldsymbol{d} \right\|_{\boldsymbol{F}}^{\mathbf{2}} + \\ \lambda_1 \sum_{j=1}^{N} \left\| \boldsymbol{H}\left(\boldsymbol{x}_j^{\boldsymbol{ph}}, \boldsymbol{\theta}\right) \boldsymbol{g} - \varphi[\boldsymbol{\eta}\left(\boldsymbol{x}_j^{\boldsymbol{ph}}, \boldsymbol{\theta}\right)] \right\|_{\boldsymbol{F}}^{\mathbf{2}} \end{array} \tag{14}$$

s.t.

$$\boldsymbol{\omega_L} \leq \boldsymbol{\theta} \leq \boldsymbol{\omega_U}\,.$$

The problem in Eq. 14 is a sum of convex functions under a convex constraint and hence convex. We can utilize Algorithm 2 to solve it, with $M = 3$ and $f_1(\boldsymbol{g}) = \|\mathbf{Y} - \boldsymbol{H_1 g} - \boldsymbol{B_1 d}\|_F^2$, $f_2(\boldsymbol{g}) = \lambda_1 \|\boldsymbol{H_2 g} - \widetilde{\boldsymbol{\eta}}\|_F^2$, $f_3(\boldsymbol{g}) = \sum_{k=1}^{q_3} \boldsymbol{I}_{\omega_{L_k} \leq \theta_k \leq \omega_{L_k}}$. The proximal operators of the functions that are given in Proposition 2. The derivation is given in Section A.2 in the Appendix A.

Proposition 2. *Let* $f_1(\boldsymbol{g}) = \|\mathbf{Y} - \boldsymbol{H_1 g} - \boldsymbol{B_1 d}\|_F^2$, $f_2(\boldsymbol{g}) = \lambda_1\|\boldsymbol{H_2 g} - \widetilde{\boldsymbol{\eta}}\|_F^2$ *and* $f_3(\boldsymbol{g}) = \sum_{k=1}^{q_3} \boldsymbol{I}_{\omega_{L_k} \leq \theta_k \leq \omega_{U_k}}$. *The proximal operators of* $\Delta f_1, \dots \Delta f_3$ *are given as follows: Let* $\boldsymbol{z}$ *and* $\boldsymbol{a}$ *have the same size as* θ.

1) *If* $\boldsymbol{z} = \mathrm{prox}_{\Delta f_1}[\boldsymbol{a}]$, *we have*

   $\mathrm{prox}_{\Delta f_1}[\boldsymbol{a}] = (\mathbf{I} + \Delta \boldsymbol{g}\boldsymbol{g}^\top)^{-1}(\boldsymbol{a} + \Delta \boldsymbol{g}(\mathbf{Y} - \boldsymbol{B_1 d})^\top)$, *where* $\mathbf{I}$ *is the identity matrix of order p.*

2) *If* $\boldsymbol{z} = \mathrm{prox}_{\Delta f_2}[\boldsymbol{a}]$, *we have*

   $\mathrm{prox}_{\Delta f_2}[\boldsymbol{a}] = (\mathbf{I} + \lambda_1 \Delta \boldsymbol{g}\boldsymbol{g}^\top)^{-1}(\boldsymbol{a} + \lambda_1 \Delta \boldsymbol{g}\widetilde{\boldsymbol{\eta}}^\top)$, *where* $\mathbf{I}$ *is the identity matrix of order p.*

3) *If* $\boldsymbol{z} = \mathrm{prox}_{\Delta f_3}[\boldsymbol{a}]$, *we have*

$$z_k = \mathrm{prox}_{\Delta f_3}[a_k] = max\left\{\omega_{L_k}, min\left\{a_k, \omega_{U_k}\right\}\right\}.$$

3.3.4.3 Identifiability and Uniqueness

Another aspect of our algorithm that needs to be discussed is identifiability issues that may arise in evaluating terms on the form $\boldsymbol{H_1 g}$ appearing in Eq. 13 for instance. This problem can be casted as a dictionary learning problem since we are in some sense trying to learn a basis matrix, $\boldsymbol{H_1}$, and a sparse coefficints vector, $\boldsymbol{g}$, The dictionary learning method does not assume that the patterns in $\boldsymbol{H_1}$ to be orthogonal like principal component analysis, and the reconstructed signal is often evaluated accurately under mild conditions. However, this does not guarantee that dictionary learning can uniquely recover sparse data from samples.

In other words, the uniqueness of the learned basis and sparse coefficients vector is not always guaranteed, which may create an identifiability issue. The detailed statistical assumptions that guarantee the identifiability is stated in proposition 3 and further discussed in Section A.3 in the Appendix A.

Proposition 3. *Suppose* $\boldsymbol{g}$ *is a vector with at most* $k \ll m$ *nonzero entries, called* k*-sparse. Assume the basis* $\boldsymbol{H}$ *is spark on the vectors* $\boldsymbol{g}$ *such that:*

$$\boldsymbol{H}\boldsymbol{g}_{\mathbf{1}} = \boldsymbol{H}\boldsymbol{g}_{\mathbf{2}} \; for \; \boldsymbol{g}_{\mathbf{1}}, \boldsymbol{g}_{\mathbf{2}} \in \mathbb{R}^m \Rightarrow \boldsymbol{g}_{\mathbf{1}} = \boldsymbol{g}_{\mathbf{2}}$$

Assuming $\boldsymbol{H}$ *is given, and our signal is constructed such that*

$$\widetilde{\mathbf{Y}} = \boldsymbol{H}\boldsymbol{g} \, ,$$

where $\widetilde{\boldsymbol{Y}} = \boldsymbol{Y} - \boldsymbol{B}_{\mathbf{1}}\boldsymbol{d}$*, a lower bound can be set on the number of samples,* n*, of* $\widetilde{\boldsymbol{Y}}$ *that guarantees unique recovery of sparse vector* $\boldsymbol{g}$ *from a signal* $\widetilde{\boldsymbol{Y}}$ *with a given basis matrix* $\boldsymbol{A}$ *such that:*

$$n \geq Ck \log\left(\frac{m}{k}\right),$$

where C *is a constant independent of* $m, n,$ *and* k*.*

In the case **H** *is unknown, assuming the above holds, we assume we can subsample N similar samples of* $\widetilde{\boldsymbol{Y}}$ *from our system such that we have* $\breve{\boldsymbol{Y}} = \{\widetilde{\boldsymbol{Y}}_1 , \ldots , \widetilde{\boldsymbol{Y}}_N\}$*. Then, the uniqueness of both* $\boldsymbol{g}$ *and* $\boldsymbol{H}$ *can be ensured if we establish a lower bound on the required number of subsamples, N, for the uniqueness of* $\boldsymbol{A}$ *with a probability of* $(1 - \beta)$ *as follows:*

$$N = \frac{k+1}{\beta}\binom{m}{k}.$$

The formula above implies that if the number of subsamples is sufficiently large, as specified in Proposition 3, the sparse matrix factorization of the basis matrix will be unique.

For further discussion and proofs on the identifiability, readers are referred to (Baraniuk et al., 2008).

3.4 Simulation Studies

In This section of this chapter utilizes simulated data to explore and measure the capabilities and efficiency of our model. We have structured our simulation experiments, presented from sections 3.4.1 to 3.4.3, with the aim of resolving the questions below:

(1) How well does PWL perform under different sample sizes of the labeled data (Section 3.4.1)?

(2) How does the accuracy of the physics-based model affect PWL performance (Section 3.4.2)?

(3) How many labeled samples can be conserved through integrating samples generated from the physics-based model (Section 3.4.3)?

Data generation: The process for generating data is uniform across all experiments, but the specific values of the parameters can vary. The process of generating simulated data begins with sampling the input variables $\left(\boldsymbol{x}_i^{pr}, \boldsymbol{x}_i^{ph}\right)$ from a multivariate normal distribution, i.e., $\left(\boldsymbol{x}_i^{pr}, \boldsymbol{x}_i^{ph}\right) \sim MN(\boldsymbol{0}, \boldsymbol{\Sigma}_x)$, where $\boldsymbol{\Sigma}_x$ is the covariance matrix. The calibration parameters are sampled from a uniform distribution, $\boldsymbol{\theta} \in (0,1]$. Then, we adopt the synthetic toy example from Bastos and O'Hagan (Bastos & O'Hagan, 2009) with some modification to simulate the output variable as follows: $y_i = \boldsymbol{\eta}_{\boldsymbol{true}}\left(\boldsymbol{x}_i^{ph}, \boldsymbol{\theta}\right) + h\left(\boldsymbol{x}_i^{pr}\right) + \varepsilon_i$, where $\boldsymbol{\eta}_{\boldsymbol{true}}(.,.)$ is the physics-based model, $h(.)$ is a function of the process variables not included in the physics-based model, and ε_i is sampled from $N(0, \sigma_\varepsilon^2)$. σ_ε^2 is the noise variance based on the selected signal-to-noise ratio $SNR = \frac{Var(y)}{\sigma_\varepsilon^2}$. We assume in our simulation that we cannot observe $\boldsymbol{\eta_{true}}(.,.)$ but a less accurate version of it as follows: $\boldsymbol{\eta}(\boldsymbol{x_i^{ph}}, \boldsymbol{\theta}) = \boldsymbol{\eta_{true}}(\boldsymbol{x_i^{ph}}, \boldsymbol{\theta}) + \delta(\boldsymbol{x_i^{ph}}) + \varepsilon_\eta$, where, $\delta(\,.\,)$ is a function that simulates the discrepancy of the physics-based model, and ε_η is uncertainty in the physics-based model sampled from $N(0, \sigma_{\varepsilon_\eta}^2)$. In our simulations, the SNR is fixed at five, q_1 to three, q_2 to two, and q_3 to two. The functional forms for our models in the simulation are chosen as follows:

$$\boldsymbol{\eta_{true}}(\boldsymbol{x_i^{ph}}, \boldsymbol{\theta}) = \left(1 - exp\left(-\frac{1}{2x_3}\right)\right) \times \frac{10\theta_1 x_1^3 + 19x_1 + 60}{9\theta_2 x_2^3 + 4x_2 + 20} \tag{15}$$

$$\delta(\boldsymbol{x_i^{ph}}) = \frac{10x_1^2 + 4x_2^2 + 2x_3^3}{50x_1x_2 + 10} \tag{16}$$

$$h(\boldsymbol{x_i^{pr}}) = \frac{4x_4^2 + 8x_5^2}{5x_4x_5} \tag{17}$$

Models under comparison: We compare PWL with a variety of supervised learning models in ML, which can only use labeled samples in training and cannot incorporate the physics-based model as PWL. The following models are compared against in the simulations:

- **Ridge regression (RR)** as representative of linear regression (**LR**) models, it uses quadratic loss and a squared L_2 regularization for the model's coefficients.
- **Support vector regression (SVR)** as representative of kernel learning methods. The kernel type selected is a polynomial kernel function which produced best results on our simulated dataset.

- **Decision tree (DT)** as representative of is a tree-based nonparametric supervised learning method. The maximum depth was selected based on prior research findings (Blum & Chawla, 2001; Chen & Wang, 2004) and best performance was selected in every experiment.
- **Random forests (RF)** as representative of ensemble methods and gradient boosted decision trees (**GBDT**) are typical choice in this case. For the hyperparameter settings, the number of boosting stages was selected from [30, 100, 160], learning rate was selected from [0.1, 0.15, 0.2], and the maximum depth was selected based on prior research findings (Blum & Chawla, 2001; Cang et al., 2018; Chapelle et al., 2006), and the best performing parameters were selected in every experiment.
- **Gaussian process regression (GP)** as representative of nonparametric, Bayesian approaches. A squared exponential kernel showed best performance on our dataset.
- **Physics-based model (Physics)** as representative of physics-based models. The model is calibrated using a two-step procedure. First, calibration parameters were obtained through least-squares estimation as follows:

$$\widehat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta} \in [\boldsymbol{\omega_L}, \boldsymbol{\omega_U}]} \sum_{i=1}^{n} \left[y_i - \boldsymbol{f}\left[\boldsymbol{\eta}\left(\boldsymbol{x}_i^{ph}, \boldsymbol{\theta}\right)\right]\right]^2 \tag{18}$$

Then, the model discrepancy was estimated using a gaussian process model similar to the approach described in Kennedy and O’Hagan (Kennedy & O'Hagan, 2001) to achieve best performance.

Performance metrics: To verify the effectiveness of the proposed method, three commonly used metrics in regression task are employed in this work, i.e., MSE, RMSE, and MAE. Besides, the mean values and the standard deviations are obtained by independent experiments with different random seeds to ensure the experiments reliability.

3.4.1 Model Performance with Varying Sample Sizes

In The objective of this experiment is to assess and contrast the effectiveness of models using varying samples size. Twelve unique scenarios are constructed, beginning with a scenario of only 10 labeled samples, and progressively adding 10 more labeled samples in each subsequent scenario. This increment continues until the final scenario, which encompasses a total of 120 labeled samples. On the other hand, we assume in each scenario that there are 100 samples generated from the physics-based model $\boldsymbol{\eta}(\cdot, \cdot)$. In each of these scenarios, we allocate 70% of the labeled samples for the training process and use the remaining 30% for validation. We also produce a separate test set with 200 labeled samples, which is employed to assess and compare the predictive performance across different models.

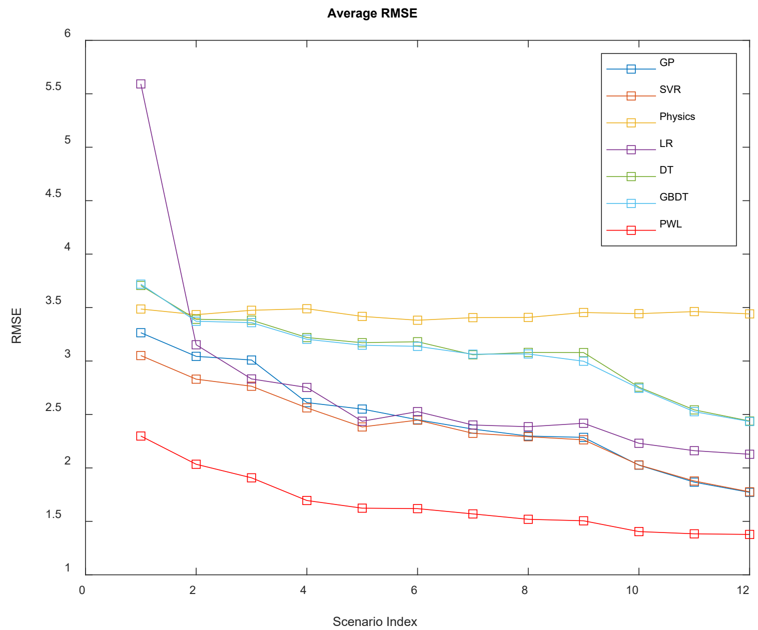

Figure 1. Average test RMSE across all models

In each scenario, PWL is trained based on both the labeled samples and the samples generated from the physics-based model. For PWL, the three tuning parameters are determined by achieving the lowest MSE on the validation set. A similar approach is applied for tuning the parameters of other models, focusing on minimizing the MSE during validation. This training, validation, and testing process is repeated 20 times randomly for each model, with recoding both the average and standard deviation of their performance metrics. In Figure 13, it is evident that PWL consistently outperforms its counterparts in terms of average RMSE. To evaluate the statistical significance of this performance disparity, one-sided hypothesis tests are performed comparing PWL against each competing model. The results indicate that PWL's performance metrics are significantly better across all scenarios compared to the metrics of competing models.

Table 3. All models' performance metrics on test data for three selected scenarios.

	RMSE (Mean ± Std)								
	Labeled samples in training								
Model	10			60			120		
LR	5.592	±	3.751	2.527	±	0.251	2.128	±	0.143
SVR	3.051	±	0.478	2.447	±	0.227	1.776	±	0.127
DT	3.707	±	0.329	3.182	±	0.310	2.439	±	0.205
GP	3.265	±	0.497	2.451	±	0.271	1.771	±	0.129
GBDT	3.713	±	0.352	3.128	±	0.323	2.423	±	0.220
Physics	3.242	±	0.155	3.143	±	0.161	3.201	±	0.216
PWL	**2.298**	**±**	**0.507**	**1.619**	**±**	**0.200**	**1.378**	**±**	**0.113**
	MSE (Mean ± Std)								
	Labeled samples in training								
Model	10			60			120		
LR	44.879	±	79.772	6.445	±	1.362	4.547	±	0.613
SVR	9.531	±	3.034	6.037	±	1.186	3.171	±	0.453
DT	13.844	±	2.420	10.221	±	2.033	5.990	±	0.999
GP	10.898	±	3.254	6.078	±	1.412	3.153	±	0.457
GBDT	13.904	±	2.617	9.882	±	2.089	5.919	±	1.072
Physics	10.536	±	1.001	9.905	±	1.017	10.293	±	1.373
PWL	**5.530**	**±**	**2.696**	**2.660**	**±**	**0.710**	**1.910**	**±**	**0.314**
	MAE (Mean ± Std)								
	Labeled samples in training								
Model	10			60			120		
LR	4.492	±	3.087	1.958	±	0.193	1.659	±	0.100
SVR	2.413	±	0.372	1.924	±	0.175	1.420	±	0.109
DT	2.959	±	0.250	2.496	±	0.252	1.901	±	0.149
GP	2.572	±	0.380	1.912	±	0.204	1.420	±	0.110
GBDT	2.963	±	0.262	2.463	±	0.251	1.898	±	0.167
Physics	2.750	±	0.156	2.650	±	0.150	2.719	±	0.202
PWL	**1.821**	**±**	**0.405**	**1.284**	**±**	**0.159**	**1.088**	**±**	**0.094**

Table 3 shows the detailed performance of three of our scenarios representing low, medium, and high labeled data availability in training. Moreover, the comparison between PWL and its competing models reveals that the largest performance gap is observed in cases with a limited number of labeled samples. This shows the advantage of PWL due to its capability of incorporating knowledge from the physics-based model to compensate for the shortage of labeled samples in training. Moreover, by comparing PWL with the physics-based model, we can observe that, regardless of increasing the number of training samples in training, the physics-based model exhibits a limit in its performance, due to the less flexible structure of the physics-based model.

Moreover, for a more comprehensive assessment that includes performance stability, Figure 14. features a boxplot analysis of the MSE in each scenario for PWL versus the best performance of the competing models. This analysis highlights a more rapid decrease in both the mean and the spread of MSE values for PWL. Additionally, the comparison reveals that the competing model exhibits greater fluctuations in both variance and mean, suggesting that its performance is less robust compared to PWL.

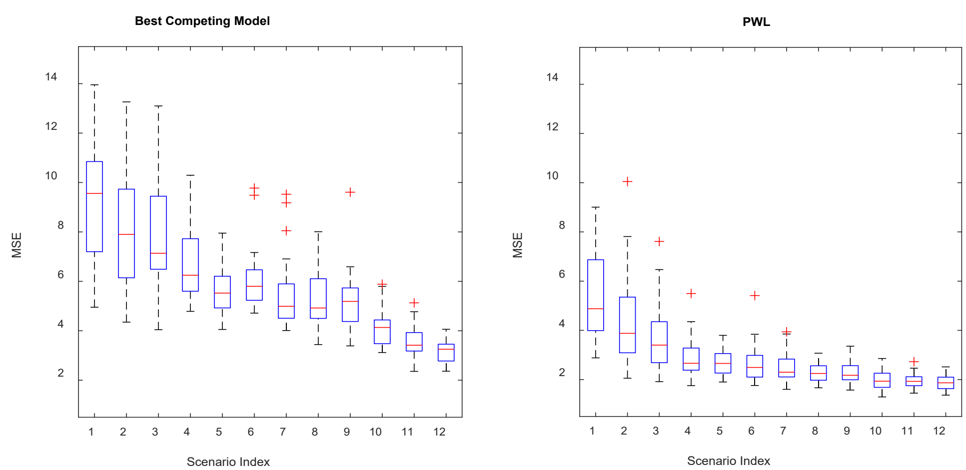

Figure 2. Boxplots of MSE for PWL vs. best competing model.

3.4.2 Model Performance Under Different Physics-based model Accuracies

The findings in the previous section showed that PWL can help improve performance under a limited number of labeled samples compared with competing models. The improvement is achieved by incorporating physical knowledge in training, which can help mitigate the shortage of labeled samples. This result leads to the question regarding how much the performance will be affected by the inaccuracy of the physics-based model.

To answer this question, we focus in this experiment on the physics-based model $\boldsymbol{\eta}(\cdot,\ \cdot)$ used in PWL. We use Pearson correlation between the label y_i and the output of the physics-based model $\boldsymbol{\eta}(\cdot,\ \cdot)$ as a metric for the accuracy of the physics-based model. The correlation will be controlled through varying the magnitude of the discrepancy function $\delta\left(\boldsymbol{x}_i^{\boldsymbol{ph}}\right)$. We create three different scenarios over which the physics-based model accuracy will vary, including a highly accurate model (i.e., scenario 1) with a correlation of 0.85, a mildly accurate model (i.e., scenario 2) with a correlation of 0.7, and an insufficiently accurate model (i.e., scenario 3) with a correlation of 0.5. All scenarios are based on 40 labeled samples and 100 samples generated from the physics-based model $\boldsymbol{\eta}(\cdot,\ \cdot)$. Table 4 shows the performance metrics of the experiments. We can see that the accuracy of the physics-based model significantly affects PWL performance with the best performance achieved with the highly accurate physics-based model (i.e., PWL-H). However, one important observation is that the performance of PWL with insufficiently accurate physics-based model (i.e., PWL-L) is statistically undisguisable from the best performance achieved by competing methods. This indicates that even when the physics-based model is not sufficiently accurate, we can expect that the performance of PWL is still comparable to competing methods that only include labeled samples in training. This result is due to the fact that PWL utilizes a tunning λ_1 to control how much the physics-based model should be emphasized in the training process.

Table 4. PWL performance under different physics-based model accuracies vs. competing models.

Model	RMSE (Mean ± Std)			MSE (Mean ± Std)			MAE (Mean ± Std)		
LR	2.409	±	0.319	5.900	±	1.626	1.882	±	0.260
SVR	2.311	±	0.296	5.422	±	1.511	1.831	±	0.257
DT	3.042	±	0.315	9.347	±	2.044	2.414	±	0.265
GP	2.336	±	0.214	5.501	±	1.043	1.829	±	0.184
GBDT	3.068	±	0.384	9.555	±	2.480	2.443	±	0.314
PWL-H	**1.392**	**±**	**0.161**	**1.962**	**±**	**0.462**	**1.106**	**±**	**0.129**
PWL-M	**2.054**	**±**	**0.251**	**4.279**	**±**	**1.056**	**1.614**	**±**	**0.198**
PWL-L	**2.404**	**±**	**0.279**	**5.854**	**±**	**1.343**	**1.890**	**±**	**0.233**

3.4.3 Saving of Labeled Samples Using PWL

The previous experiments showed that PWL can be very helpful when the labeled samples are limited, compared with models that rely only on the labeled samples. Hence, it is desirable to learn how many labeled samples can be saved by adopting PWL in order to achieve the same performance as other models. To answer this question, we focus on scenario 3 in Section 3.4.1, in which PWL achieved an average MSE of 3.745 with a training set of 30 labeled samples and 100 samples from the physics-based model. To determine how many extra labeled samples are required for the competing models to achieve the same MSE as PWL, the number of labeled samples is varied incrementally from 30 to 110 with 10 sample increments. Under each sample size, all competing models are trained and only the top-performing model's results are record. Figure 15 shows the MSE of best competing model (red curve) in comparison with PWL (blue curve).

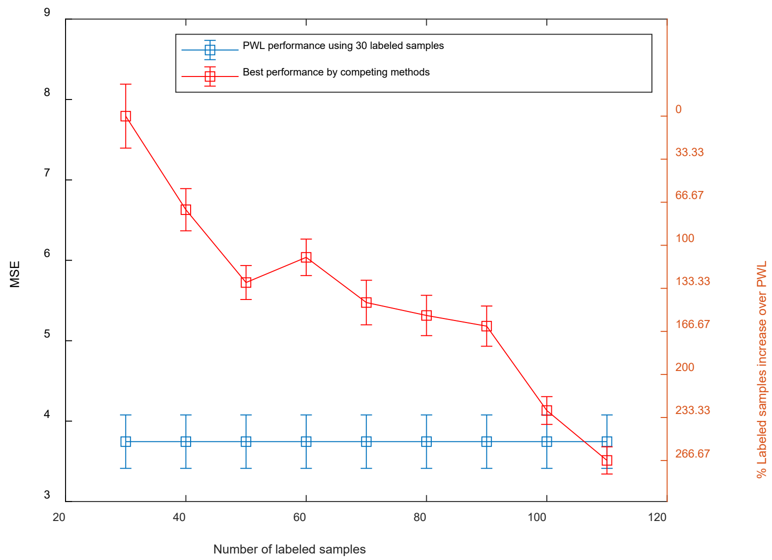

Figure 15. MSE performance of competing models with increasing labeled sample size.

We can observe that the MSE of competing models drops to be lower than that of PWL with 110 labeled samples in training. We conduct one-sided hypothesis testing to see when the performance of competing models has statistically equivalent MSE to PWL. The result shows that statistical equivalence is achieved when 100 samples are included in training. This indicates that an increment of 233.3% in labeled samples is required by the competing models to achieve the same performance of PWL. Hence, the benefit of PWL in situations of costly data collection is substantial.

3.5 Case Studies

In this section, we will present the application of PWL in two real datasets.

3.5.1 Temperature Difference Prediction in Spot Welding

In spot welding, high temperature difference usually results in a poor-quality weld. The objective of this study is to accurately predict temperature increase using process variables.

To validate our proposed PWL framework, we conducted extensive spot-welding experiments using an Electric X-type Welding Gun with a 75-kVA capacity and 50Hz frequency. During welding, in-line measurements of voltage, current, and pressure variations were recorded. The nugget formation and temperature of the welding zone were monitored using an infrared camera (TACHYON 1024 μCAMERA). Spherical tip electrodes made of CuCrZr alloy conforming to DIN 44750, Part 1, were used, and the electrode tips were replaced every 5 welds to avoid alloying and eliminate potential deformations. The electrodes had a ball radius of 75mm and a diameter of 20mm. For the workpieces, we used uncoated steel plates commonly used in the automotive industry, specifically low carbon EED quality steel (St1203 according to DIN1623), cut into 20 x 40-mm pieces with a thickness varying between 1 and 3mm. To prepare the specimens, the front face of each welded pair was ground flat with a file and finished with a 600x abrasive.

There are four controllable variables in the welding process: force [kN] (x_1), current [kA] (x_2), time of weld [s] (x_3), mass [g] (x_4), and the objective is to predict the temperature difference on a part $[\Delta T]$ (y) given input setting. The time of weld represents the length of time during which the welding process is actively conducted on the workpiece. Process data are obtained from 35 welding experiments we conducted using a Latin Hypercube design over the range of controllable parameters. The physics-based model $\boldsymbol{\eta}(.,.)$ is represented by a mathematical equation that has a set of input variables that are different from the experiment as follows:

$$\Delta T = \frac{I^2 \cdot \rho \cdot L \cdot t}{A \cdot m \cdot C_p} , \tag{19}$$

where

I: current [kA],

t: time of weld [s],

m: mass [g],

ρ: Resistivity [ohm/m],

C_p: thermal conductivity [W/mK],

L: Length of path [m],

A: Cross sectional area [m].

We can observe that the physics-based model includes three of the controllable variables from our process setting (i.e., x_2, x_3, x_4), which we refer to in PWL as $\boldsymbol{x^{ph}}$, and the variables that are only included in the process setting (i.e., x_1) are referred to as $\boldsymbol{x^{pr}}$. On the other hand, there are some parameters that are only included in the physics-based model, which PWL refers to as $\boldsymbol{\theta}$. We calibrate the physics-based model to our process samples according to the two-step procedure we described in Eq. 18. The physics-based model shows a good correlation with our process samples of 0.884, however a poor predictive performance with RMSE of 72.8. This type of physics-based model represents a desirable candidate for our PWL framework. In the first experiment, we will compare PWL performance with the benchmarking methods described in Section 3.4. We will use the 35 available samples and 35 samples from the physics equation using the same variable setting in process, and report the 5-fold cross validation mean and standard deviation performance of RMSE, MSE, and MAE from 5 random runs. Table 5 shows the performance recorded in our experiment. PWL performance in all metrics is superior to competing models. PWL not only has the smallest mean in all metrics, but also shows good robustness with small standard deviation compared to competing models.

Table 5. PWL performance vs. competing models on case study A.

Model	RMSE (Mean ± Std)			MSE (Mean ± Std)			MAE (Mean ± Std)		
LR	32.68	±	10.84	1177.49	±	839.79	26.62	±	9.78
SVR	34.04	±	14.43	1352.99	±	1141.79	27.69	±	12.92
DT	68.11	±	29.41	5359.43	±	5260.32	56.46	±	22.79
GP	24.88	±	8.93	693.46	±	466.24	21.37	±	7.82
GBDT	62.80	±	23.22	4393.36	±	3545.8	50.56	±	17.59
Physics	73.62	±	3.74	5432.69	±	561.04	61.95	±	4.21
PWL	**20.75**	**±**	**8.34**	**495.49**	**±**	**331.18**	**16.99**	**±**	**7.76**

To assess the impact of excluding variables such as force, we can compare the performance of the standalone physics-based model against that of the comprehensive PWL model, as shown in Table 5. This comparison not only underscores the limitations of the physics-based model when used in isolation but also demonstrates the improved accuracy and reliability achieved through its seamless integration with machine learning. Such an analysis provides valuable insights into the roles of missing variables in the physics-based model and the overall effectiveness of our PWL approach, thereby offering a more holistic understanding of temperature differences in spot welding.

Another aspect of PWL that we want to explore particularly in this case study is the performance under limited labeled data availability. The number of labeled samples provided in this case study is 35, which is considered a small number to fully understand predictors-response relationship. This phenomenon is very common especially in experiments that require the sample to be destroyed in some process setting to get a response reading. Hence, a model such as PWL that requires only a few labeled samples and is able to exploit unlabeled samples is highly desirable in this setting. Therefore, we will compare PWL performance with models that can utilize labeled and unlabeled samples in training such as semi-supervised learning models. The models to be compared with are CoReg (Zhou & Li, 2005), ICT (Verma et al., 2022), Π Model (Laine & Aila, 2016), and Mean Teacher (Tarvainen & Valpola, 2017). Among the 35 labeled samples, we will consider only 15 to be labeled and the rest as unlabeled. We conducted 5 random runs and the performance on the unlabeled samples is reported in Table 6.

Table 6. PWL performance vs. Semi-supervised learning models on case study A.

Model	RMSE (Mean ± Std)			MSE (Mean ± Std)			MAE (Mean ± Std)		
COREG	40.6	±	8.0	1695.6	±	667.0	31.7	±	5.2
ICT	34.4	±	13.3	1280.7	±	1086.1	35.0	±	7.7
Π Model	32.5	±	12.8	1093.8	±	1011.2	31.6	±	8.6
Mean Teacher	32.9	±	12.3	1118.1	±	960.8	32.1	±	7.4
PWL	**26.6**	±	**5.2**	**730.7**	±	**278.0**	**21.8**	±	**5.3**

We can observe in Table 6 that PWL performance is superior in mean RMSE, MSE and MAE over all competing models. PWL also shows a smaller standard deviation in all of the three metrics in Table 6, which indicates a more stable performance. Based on the observations in Tables 5 and 6, semi-supervised models demonstrated statistically significant superior performance compared to certain fully supervised models, particularly GBDT and DT. This outcome is often observed in scenarios with limited data availability. Semi-supervised models excel by utilizing both labeled and unlabeled data, thereby providing a more holistic understanding of the dataset than fully supervised models, which rely exclusively on labeled data. In contrast, fully supervised models such as GBDT and DT are more prone to overfitting when faced with a data-limited situations.

3.5.2 Nugget Diameter Prediction in Spot Welding

In this section, we consider the spot-welding experiment given in (Bayarri et al., 2007; Higdon et al., 2004). There are three variables in this process: load (x_1), current (x_2), and gage (x_3), and the objective is to optimize the nugget diameter of the weld (y). The physics-based model is quite complex and contains one calibration parameter ($\boldsymbol{\theta}$) additional to the three variables in the process. The physics-based model uses coupled partial differential equations to represent thermal, electrical, and mechanical interactions, implemented via the ANSYS commercial code, as detailed in (Bayarri et al., 2007).

The first step is to approximate the physics-based model. Because there is no random error in the physics-based computer simulation model, an interpolating model such as kriging is suitable for the model approximation (Sacks & Welch, 1989). We fit an ordinary kriging model with Gaussian correlation function using the 35 simulations in Table 5 of (Bayarri et al., 2007). Afterwards, the produced physics-based model is calibrated to the process samples described in Table 6 of (Bayarri et al., 2007) according to the two-step procedure we described in Eq. 18. The process samples in (Bayarri et al., 2007) represent 12 experimental settings with 10 replicates per setting. In this case study, we obtained more process samples, 120 in total, than the previous case study. Thus, we are able to test PWL performance under low and high data availability scenarios. We will compare two scenarios, one with only 10 labeled process samples in training and the other with 100 process samples in training. The remaining samples are used for testing, and we will conduct five random splits to report the mean and the standard deviation of RMSE, MSE, and MAE.

Table 7. PWL performance vs. Competing models on case study B.

			RMSE			
			Labeled samples in training			
Model		10			100	
LR	0.5835	±	0.0759	0.4629	±	0.0469
SVR	0.6075	±	0.0526	0.4590	±	0.0518
DT	1.0664	±	0.0371	0.4823	±	0.0904
GP	0.5981	±	0.0784	0.4708	±	0.0758
GBDT	1.1420	±	0.0654	0.5872	±	0.0576
Physics	0.9021	±	0.0395	0.8279	±	0.1138
PWL	**0.5566**	**±**	**0.0328**	**0.4519**	**±**	**0.0497**
			MSE			
			Labeled samples in training			
Model		10			100	
LR	0.3443	±	0.0907	0.2158	±	0.0444
SVR	0.3709	±	0.0639	0.2124	±	0.0486
DT	1.1382	±	0.0795	0.2380	±	0.0844
GP	0.3618	±	0.0913	0.2255	±	0.0681
GBDT	1.3069	±	0.1510	0.3470	±	0.0671
Physics	0.8149	±	0.0711	0.6940	±	0.1842
PWL	**0.3105**	**±**	**0.0361**	**0.2060**	**±**	**0.0458**
			MAE			
			Labeled samples in training			
Model		10			100	
LR	0.4769	±	0.0612	0.3696	±	0.0311
SVR	0.4960	±	0.0472	0.3643	±	0.0354
DT	0.8842	±	0.0190	0.3863	±	0.0615
GP	0.4852	±	0.0612	0.3746	±	0.0410
GBDT	0.9354	±	0.0461	0.4821	±	0.0571
Physics	0.7582	±	0.0485	0.7055	±	0.0989
PWL	**0.4578**	**±**	**0.0293**	**0.3534**	**±**	**0.0194**

Table 7 shows PWL performance compared to benchmarks under the two data availability scenarios. Generally, the best MSE value should be the closest to the variance of noise in data which was estimated to be 0.2 from the replicates. PWL achieved this desired MSE performance with a value of 0.206 under the scenario with 100 labeled samples in training. We can also observe that PWL outperforms the benchmarking methods in all performance metrics under the scenario of limited data availability.

Table 8. PWL performance vs. Semi-supervised learning models on case study B.

Labeled Samples	10		
Model	RMSE (Mean ± Std)	MSE (Mean ± Std)	MAE (Mean ± Std)
COREG	0.5843 ± 0.0713	0.3464 ± 0.0852	0.4671 ± 0.0595
ICT	0.5892 ± 0.0672	0.3517 ± 0.0804	0.4740 ± 0.0559
Π Model	0.5794 ± 0.0679	0.3404 ± 0.0787	0.4681 ± 0.0616
Mean Teacher	0.5893 ± 0.0721	0.3525 ± 0.0859	0.4691 ± 0.0568
PWL	**0.5566 ± 0.0328**	**0.3105 ± 0.0361**	**0.4578 ± 0.0293**
Labeled Samples	100		
Model	RMSE (Mean ± Std)	MSE (Mean ± Std)	MAE (Mean ± Std)
COREG	0.5024 ± 0.0319	0.2533 ± 0.0317	0.4147 ± 0.0452
ICT	0.5089 ± 0.0469	0.2612 ± 0.0482	0.4138 ± 0.0556
Π Model	0.5126 ± 0.0353	0.2640 ± 0.0361	0.4202 ± 0.0299
Mean Teacher	0.5491 ± 0.0395	0.3031 ± 0.0443	0.4592 ± 0.0455
PWL	**0.4519 ± 0.0497**	**0.2060 ± 0.0458**	**0.3534 ± 0.0194**

In the results in Table 8, the PWL model consistently outperforms other models for both 10 and 100 labeled samples, showcasing superior average performance and greater consistency across different runs as indicated by its low standard deviations. Among the semi-supervised models, their performances are relatively close, especially with a smaller labeled dataset which suggests comparable benefits in their strategies of leveraging unlabeled data. Furthermore, the performance of PWL improves more significantly with the increase in labeled samples compared to the other methods. This indicates that PWL is effective at utilizing unlabeled data and scales well with more labeled data.

3.6 Conclusion

In this chapter, we introduce a novel methodology to enhance ML model performance in manufacturing processes utilizing the weak supervision paradigm. Traditional methods relying on expensive, time-consuming labeled datasets have limitations, which we overcome by integrating partial physical knowledge, significantly improving model performance. Our approach is validated by comprehensive simulations and two case studies, demonstrating robustness across various noise levels and numbers of labeled samples, while also highlighting the immense potential of our proposed method in diverse manufacturing processes.

Methodologically, our approach offers several advantages, including substantial improvements in predictive performance and efficient optimization via the ADMM algorithm. This chapter establishes that weak supervision is a promising alternative to conventional approaches for enhancing the accuracy of ML models in manufacturing processes, especially when labeled datasets are either unavailable or prohibitively expensive to acquire.

In summary, our methodology holds the potential to reduce costs, increase efficiency, and improve in-process quality in manufacturing processes. The findings of this study open up exciting new directions for future research. A direct extension could involve adapting our flexible PWL model to manage more complex or higher-dimensional outputs, while addressing the new challenges that such outputs may present. Furthermore, our framework could be expanded to tackle intrinsic uncertainties in some physics-based models. For instances where the output variable is within a known range at a given confidence, the PWL framework might be adapted by integrating techniques like weighted loss functions, soft constraints on predictions, and probabilistic outputs, though these enhancements may add complexity to optimization.

Researchers can also explore our approach's applicability in other fields with scarce labeled data, such as non-destructive testing or predictive maintenance. There is also potential for integrating our framework with advanced machine learning techniques such as deep learning and reinforcement learning to improve predictive performance; or with weak supervision methods like crowdsourcing or semi-supervised learning to build robust models with limited labeled data. Lastly, examining the impact of incorporating domain-specific knowledge, such as expert heuristics or insights from materials science, could further improve the method's effectiveness in the manufacturing domain.

CHAPTER 4. MULTI-SOURCE MULTI-TASK WEAKLY SUPERVISED TRANSFER LEARNING FOR TELEMONITORING OF PARKINSON'S DISEASE

4.1 Introduction

The advent of mobile technology has revolutionized healthcare, particularly in the domain of chronic disease management. Mobile phones, now equipped with advanced sensors and high computing power, have transitioned from mere communication devices to pivotal tools in health monitoring. This transition has enabled continuous and remote monitoring of patients' health, significantly enhancing the ability to manage diseases like Parkinson's disease (PD) effectively (WHO, 2011).

Parkinson's disease is a neurodegenerative disorder characterized by motor symptoms such as tremors, bradykinesia, rigidity, and postural instability (Bloem et al., 2021). The global burden of PD is substantial, affecting millions of individuals and imposing significant economic costs. In the United States alone, the economic impact of Parkinson's disease is estimated to be $52 billion annually (Yang et al., 2020). Effective management of PD requires regular monitoring and precise adjustments in treatment regimens, which are traditionally achieved through clinical visits. However, these visits provide only sporadic snapshots of a patient's condition and often rely heavily on subjective reports from patients and caregivers (Dorsey et al., 2018).

Telemonitoring, defined as the use of technology to remotely monitor and transmit health-related information, offers a promising solution to these challenges (Dansky et al., 2008). Modern smartphones, equipped with a variety of built-in sensors, facilitate the continuous monitoring of PD symptoms (Patel et al., 2012). A key example of this is the mPower app, launched in 2015 as part of a clinical observational study aimed at Parkinson's disease telemonitoring (Bot et al., 2016). The mPower app guides users through several predefined activities and surveys designed to assess PD symptoms, including memory tasks, tapping exercises, voice recordings, and walking tests. This extensive data collection provides invaluable insights into the condition of PD patients, enabling the development of predictive models for disease severity. Such models have the potential to significantly facilitate comprehensive telemonitoring, contributing to better disease management and personalized treatment plans.

The data collected through the mPower app can be utilized to build a predictive model for PD severity, specifically, the model requires activity data (X), which consists of measurements from activities performed by the patient. Alongside this, a matched PD severity assessment (Y), often through patient self-reports using the Movement Disorder Society-Unified Parkinson's Disease Rating Scale (MDS-UPDRS). Ideally, each patient would have sufficient N observations of (X, Y) to build their own supervised learning model. However, several significant challenges arise in this task. While activity data can be conveniently collected multiple times a day, obtaining matched PD severity assessments is much more difficult as it requires expert evaluation. In the mPower study, patients typically self-reported PD severity using the MDS-UPDRS, which is only available monthly. This disparity results in a limited sample size (N) for model training. Furthermore, PD severity is inherently a condition that is not directly measurable, and the MDS-UPDRS provides only a partial view of the patient's condition. Comprehensive clinical assessments demand additional examinations based on medical history, symptoms, and physical and neurological examinations. Moreover, patient self-assessments can lead to inaccuracies due to subjective interpretation of questions and recall bias. These challenges indicate that relying solely on paired activity (X) and severity (Y) data is insufficient for training an accurate supervised learning model. Therefore, additional data sources and advanced modeling techniques must be leveraged.

The abundance of unlabeled activity data presents an opportunity to address these challenges using recent advancements in weak supervision techniques. Weak supervision leverages both labeled (X, Y) and unlabeled data, along with incomplete knowledge to provide weak labeling information for Y. Recent advancements in weak supervision leverage such domain knowledge to rank unlabeled data samples (F. Alenezi et al., 2022), using expert insights to generate binary labels for pairs of unlabeled samples. For example, medication can lessen the severity of a disease, allowing two samples collected before and after medication to be ranked as $y_i \geq y_j$, even if the exact values for y_i and y_j are unknown. While promising, these strategies are limited to patients with available medication information and those who exhibit significant responses to medication. Additionally, this approach is restricted to single-patient data and cannot utilize data from other patients, making it difficult to develop robust predictive models in this context. Transfer learning, on the other hand, allows the model to leverage knowledge from related tasks or datasets, improving performance when labeled data is scarce (Pan & Yang, 2010). However, Transfer learning techniques relies on the assumption that there is sufficient labeled data within each source domain to effectively transfer knowledge. In the context of PD severity prediction, individual patients often do not have enough labeled data to form a robust model, limiting the effectiveness of transfer learning. Additionally, transfer learning strategies are typically designed to handle data from a single domain and may not effectively integrate multi-domain data from multiple patients. Consequently, despite their potential; more robust and comprehensive approaches that combine the strengths of both weak supervision and transfer learning are required to effectively model and predict disease severity.

In the problem setting, we consider multiple source patients, each contributing their own sets of activity data and corresponding severity assessments. For the sth source patient, we denote the labeled data as (X_s, Y_s) and the unlabeled data as (X_u^s). The labeled data (X_s, Y_s) is typically sparse due to the infrequent and labor-intensive nature of obtaining accurate severity assessments. Conversely, the unlabeled data (X_u^s) is abundant, as activity data can be collected continuously and passively via the smartphone sensors.

The target patient's data is represented similarly, with labeled observations (X_t, Y_t) and unlabeled observations X_u^t. The primary goal is to predict the severity Y_t for the target patient using a predictive model trained on both the target and source patients' data. This involves leveraging the labeled and unlabeled data from both the target and multiple source patients to improve predictive accuracy and robustness.

To address these challenges, we propose a novel Multi-source Multi-task Weakly Supervised Transfer Learning (M2WeST) method. This approach leverages both labeled and weakly labeled data from multiple sources and incorporates multi-task learning to improve predictive performance. The M2WeST method uses a weak learning task to utilize the unlabeled data and a strong learning task for the labeled data. It is designed to handle the heterogeneity of PD across patients and enhance the accuracy of disease severity predictions by combining data from multiple patients. Our contributions in this chapter are fourfold:

(1) **Multiple Data Types**: We present a detailed formulation of the M2WeST model, highlighting its integration of multiple data sources and the use of both strong and weak learning tasks to effectively handle labeled and unlabeled data, respectively.

(2) **Multiple Heterogeneous Data Sources**: We design the M2WeST method to manage the heterogeneity of PD across different patients, making it possible to combine data from multiple patients for enhanced predictive accuracy.

(3) **Algorithm Development**: We propose a novel solution algorithm to address the nonconvex nature of the model formulation, ensuring efficient and effective optimization.

(4) **Empirical Evaluation**: We validate the performance of the M2WeST model through simulations and two case studies using real-world data from the mPower app, demonstrating its superiority over existing methods in predicting PD severity.

In summary, the proposed M2WeST method offers a robust and scalable solution for telemonitoring Parkinson's disease, addressing the posed challenges and ultimately contributing to improved patient management and care. The remainder of this chapter is organized as follows: Section 4.2 covers literature review to our model and points out the limitations. Section 4.3 presents the proposed methodology for M2WeST. Section 4.4 covers the conducted simulation experiments. Section 4.5 presents two real data application in PD. Section 4.6 discusses our conclusions.

4.2 Literature Review

Our Mobile technology has revolutionized chronic disease management, particularly through telemonitoring, enabling the collection of vast amounts of patient-specific data. This data is pivotal for understanding diseases like Parkinson's Disease (PD), as it captures daily patient behavior that cannot be observed during clinical visits (Far et al., 2021; Lenain et al., 2020). Telemonitoring has been used to develop predictive analytics for PD, but most research has focused on feature engineering and classification tasks, such as distinguishing PD patients from healthy controls (Abujrida et al., 2017; Zhang et al., 2020). Despite this progress, there remains a significant gap in using telemonitoring data to predict disease severity—an essential task for timely monitoring of disease progression and improving interventions (Matarazzo et al., 2019; Zhan et al., 2016).

The challenge of using telemonitoring data lies in the scarcity of labeled data, as obtaining expert-labeled disease severity scores is labor-intensive and time-consuming. Weakly supervised learning (WSL) offers a way to address this by leveraging large amounts of unlabeled or weakly labeled data. WSL is divided into three categories: incomplete, inexact, and inaccurate supervision (Zhou, 2018). Incomplete supervision occurs when only a subset of data is labeled, making methods like semi-supervised learning (SSL) and active learning particularly relevant. Semi-supervised learning integrates labeled and unlabeled data to improve model performance, assuming the data shares a common structure (Chapelle et al., 2006; Zhou & Li, 2010). In healthcare, SSL has been applied to problems such PD progression prediction (Gaw et al., 2022). Inexact supervision uses coarse labels, such as in multiple instance learning (MIL) frameworks, where symptoms like tremors are detected without precise onset labels (Das et al., 2012). Self-supervised learning, a subfield of WSL, has shown promise in predicting PD progression, especially when combined with anomaly detection to identify deviations in patient health trends (Jiang et al., 2021). Another example is the work of Zhang et al. (2017), who applied WSL algorithms to detect PD tremors using stratified weak labels, significantly improving model accuracy despite imprecise data. Furthermore, Alenezi et al. (2022) extend WSL by leveraging domain knowledge to uniquely integrate ranked samples with labeled data using learning-to-rank methods from Information Retrieval (IR). This approach improves predictive accuracy by combining weakly supervised ranking with labeled data, making it particularly effective in telemonitoring scenarios where precise labels are scarce. However, these approaches typically focus on single-domain data and fail to handle the variability across multiple patients. The M2WeST model extends WSL to multi-source datasets, allowing it to improve predictions using data from multiple patients and addressing both label scarcity and data heterogeneity.

Transfer Learning (TL) is a promising technique in machine learning that allows models to transfer knowledge from one domain to another (Pan & Yang, 2010). In the context of PD, multi-source transfer learning (MSTL) is particularly useful, as it combines data from multiple patients to enhance model performance (Prince et al., 2019). MSTL techniques are typically classified into instance transfer, feature transfer, and parameter transfer (Pan & Yang, 2010). Instance transfer reuses data from source domains to augment target domain data (Chattopadhyay et al., 2012), while feature transfer focuses on finding a shared feature representation across source and target domains (Sun et al., 2021).

Parameter transfer assumes related domains should share model parameters, allowing for joint parameter estimation (Tommasi et al., 2010).

These techniques enable models to transfer knowledge between patients, addressing the challenge of limited labeled data while enhancing predictive accuracy. For instance, Yoon and Li (2019) focus on building personalized, patient-specific models, ensuring that knowledge is only transferred when the source patient's data closely matches the target patient’s characteristics, thereby minimizing the risk of negative transfer. Similarly, Prince et al. (2019) aggregates data from multiple PD patients to develop a more generalized model for detecting disease symptoms and predicting severity across different patients. MSTL has also been applied in glucose prediction for diabetic patients (De Bois et al., 2020) and in medical image analysis for endoscopic lesion segmentation (Dong et al., 2020).

MSTL has demonstrated significant potential in healthcare, but it also faces several key limitations. First, is source heterogeneity, where patient symptoms, such as those seen in Parkinson’s Disease (PD), vary widely across individuals, leading to negative transfer. This occurs when knowledge transferred from one patient degrades model performance for another, a problem worsened by most MSTL methods assuming homogeneity across sources (Chandra & Kapoor, 2020; Sun et al., 2021). Second, is the assumption of sufficient labeled data in source domains. In telemonitoring, sensor data is collected continuously, but expert-labeled data—such as disease severity scores—are often limited due to the labor-intensive nature of clinical assessments (Dinov et al., 2016). This scarcity of labeled data hampers the effectiveness of traditional MSTL models, which require a large amount of labeled data at least in source domains to function optimally. Third, the combination of data from multiple patients can introduce conflicting signals, resulting in poor model performance when data from different patients compete (Karaman et al., 2021).

To overcome these limitations, we propose the M2WeST model, which integrates weak supervision, multi-source learning, and a robust optimization framework. First, M2WeST addresses the challenge of source heterogeneity by assigning a unique parameter set to each source patient, capturing their individual symptom profiles and progression patterns. This approach allows the model to leverage diverse patient data while preserving the distinct characteristics of each patient, rather than forcing homogeneity across sources, reducing the risk of negative transfer. Second, the model tackles the scarcity of labeled data in target and source domains by utilizing weak supervision through ranked pairs derived from unlabeled data. This allows the model to exploit relative severity relationships when exact labels are unavailable, improving predictive accuracy with limited labeled data. Lastly, M2WeST mitigates the issue of conflicting signals from multiple patients by applying a feature augmentation function that combines insights from all sources in a balanced way, ensuring that the data enhances the target patient's prediction without introducing harmful noise. Through careful optimization, the model balances contributions from each source, preventing conflicting data from degrading overall performance.

In summary, M2WeST's novel integration of weak supervision with multi-source learning, its ability to address source heterogeneity, and its optimized feature augmentation approach make it a powerful tool for improving PD severity predictions in telemonitoring, overcoming the major limitations present in existing approaches.

4.3 Multi-source Multi-task Weakly Supervised Transfer Learning (M2WeST) Model

This section provides a comprehensive overview of the proposed Multi-source Multi-task Weakly Supervised Transfer Learning (M2WeST) model. The methodology is structured into several subsections to systematically address the various components involved in the model's design and implementation. First, in 4.3.1 we outline the notations and problem statement. Next, in 4.3.2 we cover model overview and formulation. Then, in 4.3.3 we outline the optimization formulation to solve our problem. Finally, in 4.3.4 we provide the proposed algorithm to solve the optimization problem and discuss theoretical properties.

4.3.1 Notations and Problem Statement

To effectively describe the Multi-source Multi-task Weakly Supervised Transfer Learning (M2WeST) model, we need to establish the related notations. Let (X_l^t, Y_l^t) represent the labeled data for the target patient, where $X_l^t = \{x_i^t\}_{i=1}^{N_l^t}$ is the set of feature vectors $x_i^t \in \mathbb{R}^d$ for the target patient t and $Y_l^t = \{y_i^t\}_{i=1}^{N_l^t}$ is the corresponding set of severity scores $y_i^t \in \mathbb{R}$, with N_l^t representing the total number of labeled samples. Let $X_u^t = \{x_i^t\}_{i=N_l^t+1}^{N_u^t}$ denote the unlabeled data for the target patient, where N_u^t is the number of unlabeled samples. To generate weak labels, we construct ranked pairs from the unlabeled data using domain knowledge criteria. Such an approach is well-supported by recent advances in weakly supervised learning, where domain knowledge can provide a partial ordering of samples based on their expected severity levels, thus enriching the model's training process with valuable insights (F. Alenezi et al., 2022). For the target patient, we define a set of ranked pairs Ω_w^{t}, where each pair $\left(x_p^t, x_q^t\right) \in \Omega_w^t$ satisfies the constraint:

$$y_p^t \geq y_q^t \tag{26}$$

This constraint indicates that sample x_p^t corresponds to a more severe condition than sample x_q^t, even though the exact severity values y_p^t and y_q^t are unknown. The ranking is derived from medical insights, such as the expected effect of medication or disease progression over time. This process allows the model to capture valuable relative severity information from the unlabeled data, even in the absence of exact labels.

The M2WeST model also integrates data from multiple source patients to improve the predictions for the target patient. For each source patient $s \in \{1, \ldots, S\}$, the labeled data is denoted by (X_l^s, Y_l^s) represents the labeled data for the source patient, where $X_l^s = \left\{x_j^s\right\}_{j=1}^{N_l^s}$ is the set of feature vectors $x_j^s \in \mathbb{R}^d$ for the source patient s and $Y_l^s = \left\{y_j^s\right\}_{j=1}^{N_l^s}$ is the corresponding set of severity scores $y_j^s \in \mathbb{R}$, with N_l^s representing the total number of labeled samples for the source patient. Similarly to the target patient, let $X_u^s = \left\{x_j^s\right\}_{j=N_l^s+1}^{N_u^s}$ denote the unlabeled data for each source patient s, where N_u^s represents the number of unlabeled samples for the source patient. For each source patient s, we define ranked pairs Ω_w^s in the same way we did for the target.

Given the inherent challenges in monitoring Parkinson's Disease (PD) progression, such as the scarcity of labeled data and the variability in symptoms across different patients, traditional models often fail to generalize effectively. To address these limitations, we need a model that can effectively utilize both labeled and weakly labeled data from multiple patients while overcoming the risk of negative transfer due to patient heterogeneity. The M2WeST model focuses on aligning potentially conflicting goals from different data sources and tasks to optimize the prediction accuracy for the target patient, making use of every available piece of data to refine its predictions under the constraints of limited direct labels and patient heterogeneity.

4.3.2 Model Formulation

Having established the foundational data structures and problem setting, we proceed to detail the formulation of the M2WeST model, which integrates these elements into a cohesive predictive framework.

To address the issue of limited labeled data, we incorporate weak supervision by utilizing ranked pairs derived from unlabeled data. This approach allows the model to exploit additional information by learning relative relationships between data points, even when exact severity labels are unavailable. The ranking loss for the target patient is formulated as:

$$\mathcal{L}_{weak}^{t} = \frac{\lambda_w^t}{\Omega_w^t|} \sum_{(p,q)\in \Omega_w^t} \max\left(0, 1 - \left(f_\beta\left(x_p^t\right) - f_\beta\left(x_q^t\right)\right)\right) + \frac{1}{2}|\beta|_2^2 \qquad (27)$$

where $f_\beta(.)$ represents the model's prediction for the target patient's severity score, parameterized by β, and λ_w^t is a tuning parameter to control the importance of correctly ordering ranked pairs. This ranking loss is based on the hinge loss, inspired by the SVMRank algorithm introduced for pairwise ranking by Herbrich et al. (2000) and Joachims (2002), and has been effectively utilized in medical data contexts in Alenezi et al. (2022). The hinge loss penalizes the model if the margin between two predictions for a ranked pair does not meet a minimum threshold. The goal is to ensure that the predictions respect the relative ordering derived from the ranking constraint (e.g., severity before and after medication). Specifically, for each ranked pair (x_p^t, x_q^t), where it is known that the severity of condition x_p^t is greater than or equal to x_q^t (i.e., $y_p^t \geq y_q^t$), the model penalizes predictions where $f_\beta(x_p^t) - f_\beta(x_q^t)$ does not reflect this relationship. The squared L_2 norm aims to achieve a good generalization on the ranked samples by maximizing the closest distance between two ranked samples defined as $\frac{1}{|\beta|}$, which is equivalent to minimizing $|\beta|_2^2$. This weak supervision component enables the model to leverage the abundant unlabeled data to learn relative severity rankings, thereby improving its ability to generalize in scenarios where labeled data is limited.

In addition to utilizing weakly supervised learning from Eq. 27, the M2WeST model integrates data from multiple source patients. We define for each source patient s supervised loss similar to the target patient as follows:

$$\mathcal{L}_{supervised}^{s} = \frac{1}{N_l^s} \sum_{j=1}^{N_l^s} \| y_j^s - g_{\theta_s}(x_j^s) \|^2 \tag{28}$$

where $g_{\theta_s}(x_j^s)$ represents the model's prediction for the s^{th} source patient's severity score, parameterized by θ_s. By minimizing these loss functions, the model learns to accurately predict severity scores for source patients based on their labeled data. Similarly to the target patient, the ranking loss for each source patient s is defined as:

$$\mathcal{L}_{weak}^{s} = \frac{\lambda_w^s}{\Omega_w^s|} \sum_{(k,m)\in\,\Omega_w^s} \max\left(0, 1 - \left(g_{\theta_s}(x_k^s) - g_{\theta_s}(x_m^s)\right)\right) + \frac{1}{2}|\theta_s|_2^2 \tag{29}$$

where λ_w^t is a tuning parameter to control the importance of correctly ordering ranked pairs.

Despite the strengths of weak supervision, and multi-source learning, each approach presents significant limitations when applied separately to predicting Parkinson's disease (PD) severity. Weak supervision using ranked pairs from unlabeled data offers a way to leverage additional data by learning relative relationships, but it struggles to capture the full complexity of PD progression, which requires both absolute and relative severity assessments for more accurate predictions. On the other hand, multi-source learning adds another layer of complexity. While incorporating data from multiple source patients can enhance learning by providing diverse information, the heterogeneity of PD symptoms across different patients presents a significant challenge. Symptoms can manifest differently across patients, meaning that transferring knowledge from one source patient to another can sometimes degrade model performance for the target patient—a phenomenon known as negative transfer (Pan & Yang, 2010). Each source patient might have distinct symptom progression patterns or responses to treatment, making it difficult to effectively combine information across sources without careful optimization.

To fully leverage information from the source patients, the augmentation function $h(x) = [x, g_{\theta_1}(x), \dots, g_{\theta_S}(x)]$ is introduced, which enriches the feature space of the target patient by incorporating predictions from all source models. This function combines the target patient's original features with predictions from source models, allowing the target model to benefit from multi-source learning and extract diverse patterns of disease progression. For the target patient, the supervised loss is then defined using the augmented feature space:

$$\mathcal{L}_{supervised}^{t} = \frac{1}{N_l^t} \sum_{i=1}^{N_l^t} \| y_i^t - f_\beta(h(x_i^t)) \|^2 \tag{30}$$

With both the target and source patient losses defined, the total objective of the M2WeST model is to minimize a combination of these losses. Given the outlined challenges and the disparate nature of the data sources, the M2WeST model seeks to synergize these components through a comprehensive optimization framework. This integration not only addresses the individual limitations of each component but also aligns them towards a common goal: enhancing the predictive accuracy of Parkinson's disease severity for the target patient. The subsequent section will detail the optimization strategy employed to navigate these challenges to ensure robust convergence and meaningful learning outcomes.

4.3.3 Optimization

In the M2WeST model, the optimization problem is crucial for effectively integrating the various learning components. We propose the M2WeST cohesive formulation that integrates supervised learning, weak supervision, and multi-source learning in a unified optimization formulation as follows:

$$\min_{\beta,\theta_1,\ldots,\theta_S} \begin{gathered} \frac{1}{2N_l^t}\sum_{i=1}^{N_l^t} \| y_i^t - f_\beta\left(h(x_i^t)\right) \|^2 + \\ \frac{\lambda_w^t}{|\Omega_w^t|}\sum_{(p,q)\in\,\Omega_w^t} \max\left(0, 1-\left(f_\beta\left(h(x_p^t)\right) - f_\beta\left(h(x_q^t)\right)\right)\right) + \\ \lambda_1^t\,|\beta|_1 + \frac{\lambda_2^t}{2}\,|\beta|_2^2 + \sum_{s=1}^{S}\frac{\lambda_l^s}{2N_l^s}\sum_{j=1}^{N_l^s} \| y_j^s - g_{\theta_s}(x_j^s) \|^2 + \\ \frac{\lambda_w^s}{|\Omega_w^s|}\sum_{(k,m)\in\,\Omega_w^s} \max\left(0, 1-\left(g_{\theta_s}(x_k^s) - g_{\theta_s}(x_m^s)\right)\right) + \frac{\lambda_2^s}{2}\,|\theta_s|_2^2 \end{gathered} \tag{31}$$

The components of Eq. 31 can be explained as follows:

(1) The first term represents the mean squared error between the observed severity scores and the predictions for the target patient, providing a direct measurement of prediction accuracy using strongly labeled data. The fifth term analogously aims to minimize prediction errors for each source patient on their labeled data.

(2) The second term applies hinge loss to enforce the target model's adherence to the ranked order constraints derived from the target patient's weak labels. The sixth term accomplishes the same task for each source model utilizing their weak labels.

(3) The L_1 regularization on f_β promotes sparsity in the model parameters, reducing reliance on less informative features and improving model generalization. The L_2 regularization terms are part of the ranking loss functions of the target and source patients.

(4) The augmentation function $h(x) = [x, g_{\theta_1}(x), \ldots, g_{\theta_S}(x)]$ enhances the feature space with derived features from source-specific models. This enables f_β to utilize

not only the target patient's features but also contextual insights from multiple sources, enriching the predictive process. The tuning hyperparameters $(\lambda_w^t, \lambda_1^t, \lambda_2^t, \lambda_l^s, \lambda_w^s, \lambda_2^s)$ are employed to identify optimal values that balance the influence of each model component, ensuring robustness against the model's overfitting and underfitting tendencies.

This optimization framework effectively integrates the core components of the M2WeST model. The first and fifth terms support learning from two tasks by unifying strongly labeled data for precise predictions and weakly labeled data for relative severity rankings, all within the same parameter space. By doing so, the model efficiently leverages both exact and comparative information to predict a single severity score for the target patient. Multi-source learning is achieved through the inclusion of source-specific losses and the augmentation function $h(x)$, which enriches the target patient's feature space with predictions from diverse source patients. This allows the model to incorporate valuable insights from various symptom profiles without forcing homogeneity, ensuring that each patient's unique progression pattern is preserved. Transfer learning is facilitated by the regularization terms and the careful integration of source patient predictions, which mitigate the risk of negative transfer by balancing contributions from multiple sources. Altogether, this unified framework enables the M2WeST model to effectively handle multi-source, multi-task, and weakly supervised data, addressing key challenges such as limited labeled data and patient heterogeneity. The algorithms developed to optimize the model will be discussed in the subsequent section.

4.3.4 Algorithm

This section elaborates on the strategy employed to solve the optimization problem in Eq. 31, including parameter tuning and techniques used to ensure the model's convergence and generalization. The proposed model in Eq. 31 is clearly non-convex and highly complicated to optimize. To solve the M2WeST optimization problem, we use a Block Coordinate Descent (BCD) framework combined with Alternating Direction Method of Multipliers (ADMM), as outlined in Algorithm 3. This approach breaks the complex non-convex objective into smaller subproblems, allowing efficient updates of the target patient parameters β and source patient parameters θ_s. Each step handles regularization and ranking constraints while balancing the contributions of strong and weak supervision. In our model, we choose a linear mapping for the f_β and g_{θ_s} functions. This approach simplifies the optimization process and, in our application, has shown satisfactory results, balancing model complexity with predictive accuracy. Moreover, we enhance the model's robustness and optimization smoothness by replacing the hinge loss with a modified Huber loss, defined as $L_{huber}(t) = \max(0, 1-t)^2 - \max(0, -t)^2$, which offers quadratic behavior for small errors and linear treatment for larger deviations, thereby improving numerical stability and managing ranking violations effectively (Chapelle & Keerthi, 2010).

Algorithm 3. The M2WeST algorithm

Input: $\boldsymbol{h}(.), \lambda_w^t, \lambda_1^t, \lambda_2^t, \lambda_l^s, \lambda_w^s, \lambda_2^s, \varepsilon_0$

Initialize: $\boldsymbol{\beta} = \boldsymbol{\beta}^{\mathbf{0}}, \boldsymbol{\theta}_{\mathbf{1}} = \boldsymbol{\theta}_{\mathbf{1}}^{\mathbf{0}}, \dots, \boldsymbol{\theta}_{\boldsymbol{K}} = \boldsymbol{\theta}_{\boldsymbol{K}}^{\mathbf{0}}$

While ($\left[y_i^t - f_\beta\left(h(x_i^t)\right)\right]^2 \geq \varepsilon_0$**)**

1) Update $\boldsymbol{\beta}$ according to ADMM algorithm (Section 4.3.4.1.)

2) Update $\boldsymbol{\theta_s}$ according to ADMM algorithm (Section 4.3.4.2.)

End

Even though the overall optimization problem in Eq. 31 is non-convex, the two major updates in Algorithm 3— $\boldsymbol{\beta}$ (target patient) and $\boldsymbol{\theta_s}$ (source patient)—are individually convex when viewed as separate subproblems. In the proposed BCD-ADMM framework, there are several key considerations regarding the initialization. The $\boldsymbol{\beta}$ should be initialized with values that ensure the model can effectively learn from both the supervised target data and weak supervision constraints. We typically start with $\boldsymbol{\beta}$ from a purely supervised model trained on the target patient data. Similarly, the $\boldsymbol{\theta_s}$ initialization should aim to capture useful information from the source patients to enhance the transfer learning process; thus, we initialize θ_s from a purely supervised model trained on the source patient data.

4.3.4.1 The $\boldsymbol{\beta}$ Update

In the β update for the M2WeST model, we address the optimization problem as detailed in Eq. 31. This problem involves optimizing a combination of prediction accuracy, adherence to ranking constraints, and regularization terms. The problem in Eq. 31 reduces to solving the following problem:

$$\min_{\boldsymbol{\beta}} \begin{array}{c} \frac{1}{2N_l^t} \parallel \boldsymbol{Y} - \boldsymbol{H_l^t \beta} \parallel^2 + \frac{\lambda_w^t}{|\Omega_w^t|} l_{huber}(1 - \boldsymbol{H_w^t \beta}) + \\ \lambda_1^t \, \|\boldsymbol{\beta}\|_1 + \frac{\lambda_2^t}{2} \, \|\boldsymbol{\beta}\|_2^2 \end{array} \tag{32}$$

where $\boldsymbol{Y}$: Vector of observed outputs for the target patient's labeled data, $\boldsymbol{Y} \in \mathbb{R}^{N_l^t}$, $\boldsymbol{H}_l^t$: Matrix of transformed features for the target patient's labeled data, $\boldsymbol{H}_l^t \in \mathbb{R}^{N_l^t \times d_h}$, $\boldsymbol{H}_w^t$: Matrix representing differences between features for weak supervision ranked pairs, $\boldsymbol{H}_w^t \in \mathbb{R}^{|\Omega_w^t| \times d_h}$, and d_h is the dimension of the transformed features.

The problem in Eq. 32 is a sum of convex functions and hence convex. The problem in Eq. 32 is reformulated as follows:

$$\min_{\boldsymbol{z}} \sum_{m=1}^{M} f_m(\boldsymbol{z}), \tag{33}$$

where $M = 3, f_1(\boldsymbol{\beta}) = \frac{1}{2N_l^t} \| \boldsymbol{Y} - \boldsymbol{H}_l^t \boldsymbol{\beta} \|^2$, $f_2(\boldsymbol{\beta}) = \frac{\lambda_w^t}{|\Omega_w^t|} l_{huber}(1 - \boldsymbol{H}_w^t \boldsymbol{\beta})$, $f_3(\boldsymbol{\beta}) = \lambda_1^t \|\boldsymbol{\beta}\|_1 + \frac{\lambda_2^t}{2} \|\boldsymbol{\beta}\|_2^2$.

An efficient algorithm to address the challenges presented in Eq. 33 can be formulated using the Alternating Direction Method of Multipliers (ADMM) consensus algorithm (Muralidhar et al., 2020), detailed in Algorithm 4 below. The first step in applying the ADMM to our problem involves deriving the proximal operators for the objective function. These operators are instrumental in computing the optimal solution for each variable independently while keeping the rest fixed. Proximal operators play a critical role in the ADMM framework, as they are essential for ensuring the algorithm's convergence and maintaining overall stability. By integrating these proximal operators within the ADMM consensus algorithm, we can solve the optimization problem described in Eq. 33 both effectively and efficiently, ensuring the solution is both robust and stable.

Algorithm 4. The ADMM Consensus Algorithm

Input: Step size Δ, convergence threshold ε

Initialize: For m = 1 to M, set $\boldsymbol{z}_{\boldsymbol{m}}^{(0)} = \mathbf{0},\ \boldsymbol{u}_{\boldsymbol{m}}^{(0)} = \mathbf{0},\ \bar{\boldsymbol{z}} = \mathbf{0}.$

Repeat For $k = 1,2,\ldots$ until convergence**:**

Parallel Update:

For m = 1 to M, update:

$$\boldsymbol{z}_{\boldsymbol{m}}^{(k)} = \mathrm{prox}_{\Delta f_m}\left[\bar{\boldsymbol{z}}^{(k-1)} - \boldsymbol{u}_{\boldsymbol{m}}^{(k-1)}\right], \text{ for } m = 1, \ldots, M.$$

Aggregate Update:

$$\bar{\boldsymbol{z}}^{(k)} = \sum_{m=1}^{M} \frac{\boldsymbol{z}_m^{(k)}}{M}$$

Dual Update:

For m = 1 to M, update:

$$\boldsymbol{u}_{\boldsymbol{m}}^{(k)} = \boldsymbol{u}_{\boldsymbol{m}}^{(k-1)} + \boldsymbol{z}_{\boldsymbol{m}}^{(k)} - \bar{\boldsymbol{z}}^{(k)}$$

Convergence Check:

If $\left|\bar{\boldsymbol{z}}^{(k)} - \bar{\boldsymbol{z}}^{(k-1)}\right| < \epsilon$

Stop

End

Even To implement Algorithm 4 in solving Eq. 33, we need to evaluate the proximal operators of $\Delta f_1, \ldots \Delta f_M$**.** The results are in Proposition 4, and the derivation is in Section B.1 in Appendix B.

Proposition 4. *Let* $f_1(\boldsymbol{\beta}) = \frac{1}{2N_l^t} \parallel \boldsymbol{Y} - \boldsymbol{H}_{\boldsymbol{l}}^{\boldsymbol{t}}\boldsymbol{\beta} \parallel^2$, $f_2(\boldsymbol{\beta}) = \frac{\lambda_w^t}{|\Omega_w^t|} l_{huber}(1 - \boldsymbol{H}_{\boldsymbol{w}}^{\boldsymbol{t}}\boldsymbol{\beta})$, $f_3(\boldsymbol{\beta}) = \lambda_1^t \|\boldsymbol{\beta}\|_1 + \frac{\lambda_2^t}{2} \|\boldsymbol{\beta}\|_2^2$. *The proximal operators of* $\Delta f_1, \ldots \Delta f_3$ *are given as* *follows: Let $\boldsymbol{v}$ have the same size as $\boldsymbol{\beta}$.*

1) *For* $\mathrm{prox}_{\Delta f_1}[\boldsymbol{v}]$, *we have*

$\mathrm{prox}_{\Delta f_1}[\boldsymbol{v}] = \left(\mathbf{I} + \frac{\Delta}{N_l^t}(\boldsymbol{H}_l^t)^\top \boldsymbol{H}_l^t\right)^{-\mathbf{1}} \left(\boldsymbol{v} + \frac{\Delta}{N_l^t}(\boldsymbol{H}_l^t)^\top \boldsymbol{Y}\right)$, *where* $\mathbf{I}$ *is the identity matrix of order* d_h.

2) *For* $\mathrm{prox}_{\Delta f_2}[\boldsymbol{v}]$, *we have*

$$\mathrm{prox}_{\Delta f_2}[\boldsymbol{v}] = \begin{cases} \boldsymbol{v} & \text{if } \boldsymbol{v} \geq 1 \\ \dfrac{2\Delta \frac{\lambda_w^t}{|\Omega_w^t|} + \boldsymbol{v}}{2\Delta \frac{\lambda_w^t}{|\Omega_w^t|} + 1} & \text{if } 0 \leq \boldsymbol{v} < 1 \\ \boldsymbol{v} - 2\Delta \dfrac{\lambda_w^t}{|\Omega_w^t|} & \text{if } \boldsymbol{v} < 0 \end{cases}$$

3) *For* $\mathrm{prox}_{\Delta f_3}[\boldsymbol{v}]$, *we have*

$$\mathrm{prox}_{\Delta f_3}[\boldsymbol{v}] = \frac{1}{1 + \Delta\lambda_2^t} S_{\Delta\lambda_1^t}(\boldsymbol{v}), where\ S_{\Delta\lambda_1^t} = \begin{cases} \boldsymbol{v} - \Delta\lambda_1^t, & \boldsymbol{v} > \Delta\lambda_1^t \\ \mathbf{0}, & -\Delta\lambda_1^t \leq \boldsymbol{v} \leq \Delta\lambda_1^t \\ \boldsymbol{v} + \Delta\lambda_1^t, & \boldsymbol{v} \leq \Delta\lambda_1^t\,. \end{cases}$$

4.3.4.2 The $\boldsymbol{\theta_s}$ Update

In the θ_s update for the M2WeST model, we address the optimization problem as detailed in Eq. 31. The problem in Eq. 31 reduces to solving the following problem:

$$\min_{\theta_s} \frac{\lambda_l^s}{2N_l^s} \parallel \boldsymbol{Y^s} - \boldsymbol{H_l^s}\boldsymbol{\theta_s} \parallel^2 + \frac{\lambda_w^s}{\Omega_w^s|} l_{huber}(1 - \boldsymbol{H_w^s}\boldsymbol{\theta_s}) + \frac{\lambda_2^s}{2} \|\boldsymbol{\theta_s}\|_2^2 \tag{34}$$

where $\boldsymbol{Y^s}$**:** Vector of observed outputs for the source patient s labeled data, $\boldsymbol{Y^s} \in \mathbb{R}^{N_l^S}$, $\boldsymbol{H}_l^s$: Matrix of transformed features for the source patient s labeled data, $\boldsymbol{H}_l^s \in \mathbb{R}^{N_l^S \times d_s}$, $\boldsymbol{H}_w^s$: Matrix representing differences between features for weak supervision ranked pairs, $\boldsymbol{H}_w^s \in \mathbb{R}^{|\Omega_w^s| \times d_s}$, and d_s is the dimension of the source patient features.

The problem in Eq. 34 is a sum of convex functions and hence convex. The problem in Eq. 34 is reformulated as follows:

$$\boldsymbol{\min_{z}} \sum_{m=1}^{M} f_m(\boldsymbol{z}), \tag{35}$$

where $M = 3, f_1(\boldsymbol{\theta_s}) = \frac{\lambda_l^s}{2N_l^s} \parallel \boldsymbol{Y^s} - \boldsymbol{H_l^s \theta_s} \parallel^2,\ f_2(\boldsymbol{\theta_s}) = \frac{\lambda_w^s}{|\Omega_w^s|} l_{huber}(1 - \boldsymbol{H_w^s \theta_s}),\ f_3(\boldsymbol{\theta_s}) = \frac{\lambda_2^s}{2} \|\boldsymbol{\theta_s}\|_2^2$. Similar to the β-update, we can implement Algorithm 4 in solving Eq. 10, which requires evaluating the proximal operators of $\Delta f_1, \ldots \Delta f_M$**.** The results are in Proposition 5, and the derivation is in Section B.2 in Appendix B.

Proposition 5. *Let* $f_1(\boldsymbol{\theta_s}) = \frac{\lambda_l^s}{2N_l^s} \parallel \boldsymbol{Y^s} - \boldsymbol{H_l^s \theta_s} \parallel^2,\quad f_2(\boldsymbol{\theta_s}) = \frac{\lambda_w^s}{|\Omega_w^s|} l_{huber}(1 - \boldsymbol{H_w^s \theta_s}),\ f_3(\boldsymbol{\theta_s}) = \frac{\lambda_2^s}{2} \|\boldsymbol{\theta_s}\|_2^2$. *The proximal operators of* $\Delta f_1, \ldots \Delta f_3$ *are given as follows: Let* $\boldsymbol{v}$ *have the same size as* $\boldsymbol{\theta_s}$.

1) *For* $\text{prox}_{\Delta f_1}[\boldsymbol{v}]$, *we have*

$\text{prox}_{\Delta f_1}[\boldsymbol{v}] = \left(\mathbf{I} + \frac{\Delta}{N_l^s}(\boldsymbol{H_l^s})^\top \boldsymbol{H_l^s}\right)^{\mathbf{-1}} \left(\boldsymbol{v} + \frac{\Delta}{N_l^s}(\boldsymbol{H_l^s})^\top \boldsymbol{Y^s}\right)$, *where* $\mathbf{I}$ *is the identity matrix of order* d_s.

2) *For* $\text{prox}_{\Delta f_2}[\boldsymbol{v}]$, *we have*

$$\mathrm{prox}_{\Delta f_2}[\boldsymbol{v}] = \begin{cases} \boldsymbol{v} & \text{if } \boldsymbol{v} \geq 1 \\ \dfrac{2\Delta \dfrac{\lambda_w^s}{|\Omega_w^s|} + \boldsymbol{v}}{2\Delta \dfrac{\lambda_w^s}{|\Omega_w^s|} + 1} & \text{if } 0 \leq \boldsymbol{v} < 1 \\ \boldsymbol{v} - 2\Delta \dfrac{\lambda_w^s}{|\Omega_w^s|} & \text{if } \boldsymbol{v} < 0 \end{cases}$$

3) *For* $\mathrm{prox}_{\Delta f_3}[\boldsymbol{v}]$, *we have*

$$\mathrm{prox}_{\Delta f_3}[\boldsymbol{v}] = \frac{\boldsymbol{v}}{1 + \Delta\lambda_2^s}.$$

4.3.4.3 Practical considerations and theoretical properties

In this section, we discuss the practical aspects of the **M2WeST** model for optimal performance, along with a theoretical analysis of its generalization properties. The M2WeST model's performance depends on careful selection of hyperparameters, hence we used Optuna, an efficient hyperparameter optimization framework, to tune the hyperparameters of the M2WeST model. Optuna utilizes a novel Tree-structured Parzen Estimator (TPE) for adaptive sampling of the hyperparameter space, allowing it to efficiently navigate complex, high-dimensional optimization problems while reducing computational cost (Akiba et al., 2019). Optuna is particularly useful in our optimization context due to its ability to automate the search for the best hyperparameters based on validation performance, significantly reducing manual tuning efforts. It also supports pruning of unpromising trials, which improves computational efficiency. Optuna iteratively adjusted hyperparameters, based on the MSE values from previous trials. Each trial ran the M2WeST model with different hyperparameter settings, and Optuna tracked the results to find an optimal configuration.

Furthermore, in practical applications of the **M2WeST** model, particularly when there are a large number of source patients, it becomes essential to filter out irrelevant source patients that may negatively affect the transfer learning process. Including sources that do not align well with the target patient may introduce negative transfer, leading to suboptimal performance. Therefore, it is necessary to employ a pre-selection step to identify the most relevant sources before applying the transfer learning model. Here, we propose in Algorithm 5 a source selection algorithm that evaluates each source based on two key criteria: 1) Accuracy potential: How well the source predicts the target patient's outcomes. 2) Correlation potential: This measures the redundancy between sources, reflecting how similar or different the predictions of the sources are. If multiple sources are highly correlated, they provide similar information, which can be redundant and may not contribute additional value to the transfer learning process. By selecting sources with the highest combined potential—balancing both accuracy and correlation—we reduce the risk of including redundant sources that do not offer diverse insights. This helps to avoid negative transfer and improves the overall performance of the M2WeST model by ensuring that the selected sources provide complementary, rather than redundant, information for the target patient. This approach draws on concepts from recent work on identifying useful sources (Shin et al., 2022).

Algorithm 5. Pre-selection of Source Patients for M2WeST

Input: $X_l^t, Y_l^t, g_\theta = \{g_{\theta_1}, g_{\theta_2}, \dots, g_{\theta_S}\}, \gamma_1, \gamma_2, K$

Initialize: Calculate $\mathrm{MSE_s} = \frac{1}{N_l^t} \sum_{i=1}^{N_l^t} \parallel y_i^t - g_s(x_i^t) \parallel^2$ for each source model g_s, $\forall\, s \in S$

1. Normalize $\mathrm{MSE_s}$ for each source model using min-max scaling

2. Compute correlation $Corr\left(g_a(x_i^t), g_b(x_i^t)\right)$ between each pair of source models
3. Calculate combined potential P_a for each source:

$$P_a = -\gamma_1 \, . \, \mathrm{MSE_s} - \frac{\gamma_2}{S-1} \sum_{a \neq b} Corr\left(g_a(x_i^t), g_b(x_i^t)\right)$$

4. Rank sources by combined potential P_a
5. Select the top K sources based on the highest P_a

Output: Selected_Sources: Top K source models for the target patient

This algorithm serves as a pre-selection step in cases where there is a large pool of potential source patients. Including all sources could increase computational costs and potentially harm the model's performance due to negative transfer. By applying this selection process, we focus on the most relevant and diverse sources, enhancing the effectiveness and efficiency of the M2WeST model.

The function $h(.)$ is used as an augmentation function to transfer knowledge from the source to the target task. The hypothesis is that $h(.)$ encodes useful information from the source task that helps improve generalization performance on the target task, especially when the target task has limited labeled data. The role of function is to combine features from both the target and source tasks to create an augmented feature representation that is more informative for learning the target task. The augmented representation for each input x_i^t in the target task carries both target-specific and source-specific knowledge, which can help the target predictor $f_\beta\left(h(x_i^t)\right)$ generalize better, particularly when the source and target tasks are related. We can derive a proposition to show how using $h(.)$ can reduce the error in a simple case with a single source.

Proposition 6. *Let* $\mathcal{D}_{target} = \{(x_i^t, y_i^t)\}_{i=1}^{N_l^t}$ *be the labeled dataset for the target task, and* $\mathcal{D}_{source} = \left\{\left(x_j^s, y_j^s\right)\right\}_{j=1}^{N_l^s}$ *be the labeled dataset for the source task be the dataset for the source task. Assume that the source task* $g_{\theta_s}(x)$ *generalizes with a risk bound of* $\mathcal{O}\left(\frac{1}{(N_l^s)^{\gamma_s}}\right)$. *Assume that* $\ell(f(x), y)$ *is a* L'*-Lipschitz continuous loss function with respect to its parameters.*

Then, the excess risk for the target task is bounded by:

$$\mathbb{E}[\ell(f_\beta(h(x)), y^t) - \ell(f^*(h(x)), y^t)] = \mathcal{O}\left(\frac{d\gamma_s\rho\log(RL'N_l^t) + \log\left(\frac{1}{p}\right)}{N_l^t} + \frac{L'}{(N_l^t)^{\gamma_s\rho}}\right)$$

The bound holds with probability at least $(1-p)$, *provided that* $N_l^s = \Omega((N_l^t)^{2\rho})$.

where γ_s *is the learning rate of the source task,* d *is the number of parameters in* f_β, *and* ρ *is the relatedness factor between the source and target tasks.*

The proof is provided in in Section B.3 in the Appendix B. Readers interested in a more detailed derivation are encouraged to refer to (Robinson et al., 2020) for a thorough discussion on the theoretical foundations. In this proposition, we have demonstrated the potential benefit of using the augmentation function $h(\cdot)$to transfer knowledge from a single source task to a target task. This is particularly relevant in cases where the target task has limited labeled data, and the source task can provide complementary information that improves generalization performance. While this proposition focuses on the case of a single source, it is possible to extend this derivation to the multiple source case, where multiple sources of knowledge are integrated into the target task via the augmentation function. Such an extension would involve combining feature representations from several source tasks while managing the potential interactions between them. We leave the formal derivation of the multiple sources case for future work.

4.4 Simulation Studies

In this section, we use simulated data to evaluate the performance and capabilities of our M2WeST model in comparison with several benchmark methods. Our simulation experiments in sections 4.4.1 to 4.4.4 aim to answer the following key questions: How does M2WeST perform relative to other methods in terms of predictive accuracy (section 4.4.1)? How does the relationship between source and target patients affect the model's transfer learning performance (section 4.4.2)? How does M2WeST handle uncertainty in source data (section 4.4.3)? Finally, how robust is M2WeST to variations in weak label uncertainty (section 4.4.4)? These experiments are designed to showcase the flexibility and strength of M2WeST under a variety of conditions.

Data Generation: In all the experiments, the data generation process follows a consistent structure, though specific parameter values may vary across experiments. We generated synthetic patient data for both the target and source domains to simulate the variability and complexity typically observed in real-world patient data for Parkinson's Disease. For the target patient, $\boldsymbol{\beta}$ coefficients are sampled from a multivariate normal distribution, $\boldsymbol{\beta} \sim N(\boldsymbol{\mu}_\beta, \boldsymbol{\Sigma}_\beta)$, where $\boldsymbol{\mu}_\beta$ is a vector of ones and $\boldsymbol{\Sigma}_\beta$ is a symmetric positive semi-definite matrix. For each source patient s, the $\boldsymbol{\theta}_s$ coefficients are generated as $\boldsymbol{\theta}_s = \boldsymbol{\beta} + \boldsymbol{\eta}_s$, where $\boldsymbol{\eta}_s \sim N(0, \sigma_{\delta_1}^2 \mathbf{I})$, and δ_1 controls the degree of patient-specific variability in the coefficients. This setup simulates patient-specific variability in disease progression and treatment response, drawing on concepts from the probabilistic frameworks for transfer learning and personalized modeling (Zou, 2015; Alaa & Van Der Schaar, 2017). The generated coefficients are fixed and treated as the ground truth for comparison against model estimates. Two sets of features were generated: relevant features ($\mathbf{Z}_1$) and noise features ($\mathbf{Z}_2$), both drawn from multivariate normal distributions $N(\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$. These features capture relationships between patient covariates and the underlying disease severity, modeling real-world variability in patient data (Goncalves et al., 2020). The response variable for target $\mathbf{Y}^t$ is then generated as $\mathbf{Y}^t = \mathbf{Z}_1^t\boldsymbol{\beta} + \boldsymbol{\epsilon}_{noise}$, where $\boldsymbol{\epsilon}_{noise} \sim N(0, \sigma_{noise}^2)$ captures the inherent noise. The noise variance σ_{noise}^2 is adjusted based on the desired signal-to-noise ratio (SNR), ensuring realistic variance in the generated data, where $SNR = \frac{Var(\mathbf{Z}_1\boldsymbol{\beta})}{\sigma_{noise}^2}$. The response variable for source $\mathbf{Y}^s$ is generated similarly as $\mathbf{Y}^s = \mathbf{Z}_1^s\boldsymbol{\theta}_s + \boldsymbol{\epsilon}_{noise}$. This process was repeated for 100 source patients, each with independently generated feature data. For all experiments, the number of relevant features is set to 20, and noise features to 5 to test model robustness. The number of target and source patient samples and the amount of weakly labeled data are also varied across experiments to simulate different conditions.

To incorporate weakly labeled data, we generate ranked pairs from the unlabeled data for both the target and source patients. For each patient, the pairwise comparisons are constructed based on the differences in the response values, providing partial orderings of the data points which ensures meaningful weak supervision (Joachims, 2002). For the target patient, we define a set of ranked pairs $\Omega_{\mathrm{w}}^{\mathrm{t}}$, where each pair $(\mathbf{z}_p^t, \mathbf{z}_q^t) \in \Omega_{\mathrm{w}}^{\mathrm{t}}$ satisfies: $y_p^t \geq y_q^t$. The difference in feature vectors is given by $\Delta_{\mathbf{z}_{pq}^t} = \mathbf{z}_p^t - \mathbf{z}_q^t$, and a weak label y_w^t is assigned as: $y_w^t = sign(y_p^t - y_q^t + \epsilon_{\delta_3})$. A threshold is applied to include the pair only if $\left|y_p^t - y_q^t + \epsilon_{\delta_3}\right| \geq threshold$, where $\epsilon_{\delta_3} \sim N(0, \sigma_{\delta_3}^2)$ captures the uncertainty in weak labels. The threshold for generating pairwise comparisons is typically chosen based on the noise variance. A reasonable approach would be to set the threshold to a value larger than the standard deviation σ_{noise} of the noise term to ensure that only comparisons between significantly different samples are included. For source patients, pairwise comparisons are generated similarly to the target.

In this study, we compare the performance of our proposed M2WeST method, which uniquely integrates multi-source learning and weakly supervised learning, with several benchmark models. These benchmarks include Single Learning (SL), which uses only the labeled data from the target patient in a ridge regression model; TrAdaBoost, a transfer learning model that uses fully labeled data from source patients (Dai et al., 2007); Correlation Alignment (CORAL), which aligns the distributions between source and target domains focusing on reducing domain shift (Sun et al., 2016); and Mean Teacher, a semi-supervised model that combines labeled and unlabeled data from the target patient (Tarvainen & Valpola, 2017). Additionally, we include a ranking-based weak supervision (RWSL) model that enhances predictions using weak supervision within the target domain (F. Alenezi et al., 2022), and Adapted Tree Boosting for Transfer Learning (ATBTL), a multi-source transfer learning model that improves predictions through knowledge transfer across multiple source patients (Fang et al., 2019).

To assess these models, we use several performance metrics, including Mean Squared Error (MSE), which measures the average squared difference between predicted and true responses; Root Mean Squared Error (RMSE), which provides a more interpretable measure of prediction error variability; and Predictive Correlation (Pearson correlation), which evaluates the correlation between predicted and true responses, providing complementary insights to MSE. This comprehensive evaluation ensures that each model's ability to capture the complexities of Parkinson's disease progression is thoroughly assessed, particularly in its use of both labeled and weakly labeled data.

4.4.1 Predictive Performance Evaluation

In this experiment, we aim to evaluate the predictive performance of the M2WeST model in comparison with several benchmark methods under controlled simulation settings. The synthetic data generation follows the setup described earlier. both target and source patients are assigned 50 labeled training samples and 1000 ranked pairs generated from pairwise comparisons. An 80:20 split was applied to the training samples and a blind test set of 200 labeled samples was used for the final evaluation of model performance. Each model was tuned using the validation MSE to optimize its hyperparameters during training. The performance results are summarized in Figure 16 (a)-(c), which shows the distribution of MSE, RMSE, and Pearson Correlation for each model.

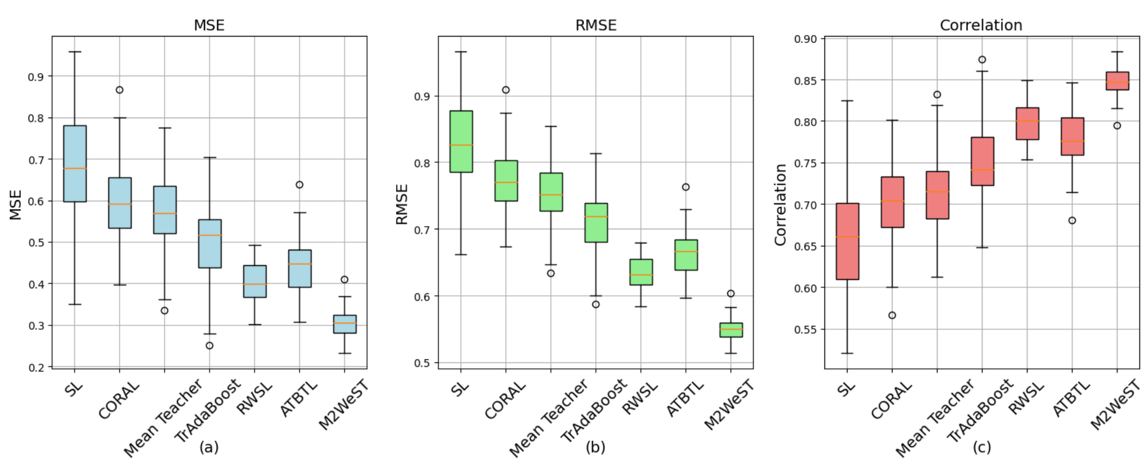

Figure 16. Boxplots showing the performance of each model.

As demonstrated, the M2WeST model consistently outperformed all other methods, achieving the lowest MSE and RMSE, and the highest Correlation. This indicates that M2WeST not only provided more accurate predictions but also exhibited less variability across the different random seeds compared to the other models. While RWSL and ATBTL demonstrated competitive performance, they exhibited higher variability compared to M2WeST. The SL and CORAL models, in particular, had the highest error rates and lowest correlation, indicating their inability to fully leverage the available data for prediction tasks. The statistical significance of these results was confirmed through one-sided hypothesis testing, showing that the mean performance of M2WeST was significantly better than the competing models across all metrics.

4.4.2 Performance under different source to target relatedness

In this experiment, we explore how models perform under varying levels of source-target relatedness, specifically examining their ability to handle patient heterogeneity. We assess the performance of four models—TrAdaBoost, ATBTL, CORAL, and M2WeST—by varying the degree of similarity between the source and target domains. The degree of relatedness between the source and target is incrementally adjusted using δ_1. The relatedness levels, are ranged from 1 (least related) to 5 (most related) to simulate different scenarios allowing us to observe how each model adapts as the source and target become more similar. The primary evaluation metric used is MSE.

The plot in Figure 17 illustrates the MSE of each model across the five relatedness levels. As the relatedness increases, all models show a reduction in MSE, reflecting improved performance as the heterogeneity between patients decreases. M2WeST demonstrates the lowest MSE across all levels of relatedness. This is because M2WeST places greater weight on weakly supervised data from the target's own domain at high heterogeneity scenarios, allowing it to mitigate the negative transfer effects that occur when the source patients are very different from the target. As the relatedness between the source and target increases, M2WeST continues to improve, leveraging both the weak supervision and increased similarity to further reduce its error. In contrast, models like ATBTL and TrAdaBoost show higher MSE at lower relatedness levels due to their reliance on closer source-target similarities to transfer knowledge effectively. They perform better as the relatedness improves but struggle with high patient heterogeneity.

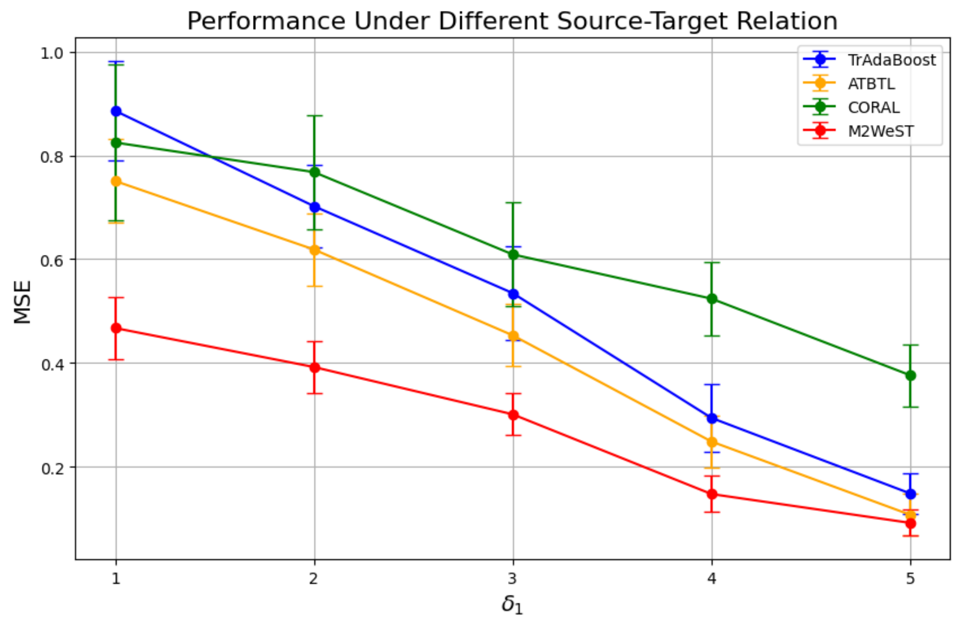

Figure 17. Average MSE performance on test data across different relatedness levels.

This experiment underscores M2WeST's ability to handle both weak supervision and patient heterogeneity, making it the most robust model in scenarios where the source and target patients are dissimilar. The other models benefit more from reduced heterogeneity but are less adaptable when patient differences are more pronounced.

4.4.3 Performance under different source uncertainty

In this experiment, we investigate how models—TrAdaBoost, ATBTL, CORAL, and M2WeST—perform under varying levels of source uncertainty, represented by δ_2 . The goal is to assess each model's ability to handle situations where the data from the source patients becomes less reliable or more uncertain. Source uncertainty is introduced by varying numbers of labeled samples available in the source domains, simulating conditions where the model parameter of the source might be noisy or less accurate due to insufficient samples.

We simulate five different scenarios where the number of labeled samples in the source domains ranges from 100 (scenario1) to 20 (scenario 5), corresponding to increasing levels of source uncertainty (δ_2). The target domain is kept constant with 50 labeled samples, and additionally there are 1,000 weakly labeled ranked samples from both the source and target domains. Figure 18 shows the MSE results across the five scenarios for each model. As expected, all models exhibit improved performance when more labeled data is available in the source domains. However, the magnitude of the improvement and the ability to cope with reduced labeled data varies significantly between models.

M2WeST consistently outperforms the other models in all scenarios. This is particularly evident in scenarios where the number of labeled samples in the source domains is low (e.g., scenarios 4 and 5). M2WeST's ability to leverage weak supervision from both the source and target domains allows it to compensate for the lack of labeled data, which is reflected in its lower MSE and more stable performance as the sample size decreases. The gap between M2WeST and the other models grows larger in the later scenarios as the number of labeled samples becomes more limited.

TrAdaBoost and ATBTL demonstrate decent performance when there is a moderate amount of labeled data (e.g., scenarios 1 and 2) but struggle as the number of labeled samples decreases. Their reliance on labeled source data leads to higher MSE in the later scenarios, where fewer labeled samples are available for effective transfer learning. The variability of their performance also increases, indicating less stability in their predictions as the labeled data diminishes. CORAL performs well when there are sufficient labeled samples in the source domains but suffers the most in later scenarios where only a small number of labeled samples are available. CORAL's domain alignment mechanism is less effective when there is insufficient labeled data to accurately align the source and target domains, leading to a sharp increase in MSE in scenarios with low sample sizes.

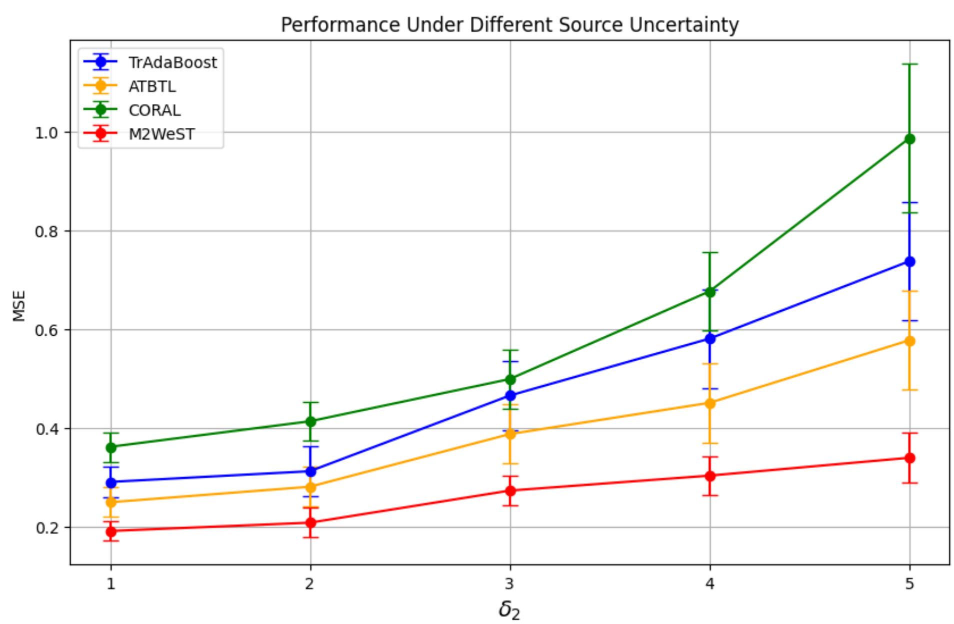

Figure 18. Average MSE performance on test data across different source uncertainty.

4.4.4 Performance under different weak labels uncertainty

In this experiment, we evaluate how models perform under varying levels of uncertainty in weak labels. The purpose is to assess the robustness of the models, particularly M2WeST and RWSL, to inaccuracies and noise introduced in the weakly labeled ranking data. The uncertainty is progressively increased across five scenarios by increasing $\sigma^2_{\delta_3}$, simulating different levels of uncertainty in the weakly labeled ranked pairs generated from the target domain.

As shown in Figure 19, the M2WeST model maintains a consistently low MSE across all levels of weak label uncertainty. This result highlights M2WeST's robustness to inaccuracies or noise in the weak labels. Even as the level of uncertainty increases, M2WeST's performance remains relatively stable, with only a slight increase in MSE. This behavior suggests that M2WeST effectively leverages both weak and strong data, allowing it to mitigate the negative impacts of noisy weak labels and maintain high predictive accuracy by relying more on strong data from both the target and source domains.

In contrast, the RWSL model shows a drastic increase in MSE as the weak label uncertainty grows. While RWSL starts with a relatively low MSE at δ_3=1, its error increases sharply at higher levels of δ_3. This indicates that RWSL is highly sensitive to the quality of the weak labels, with its performance deteriorating significantly as the uncertainty in the labels rises. The results suggest that RWSL's reliance on weak label quality limits its robustness in noisy scenarios, making it less adaptable to uncertain environments compared to M2WeST.

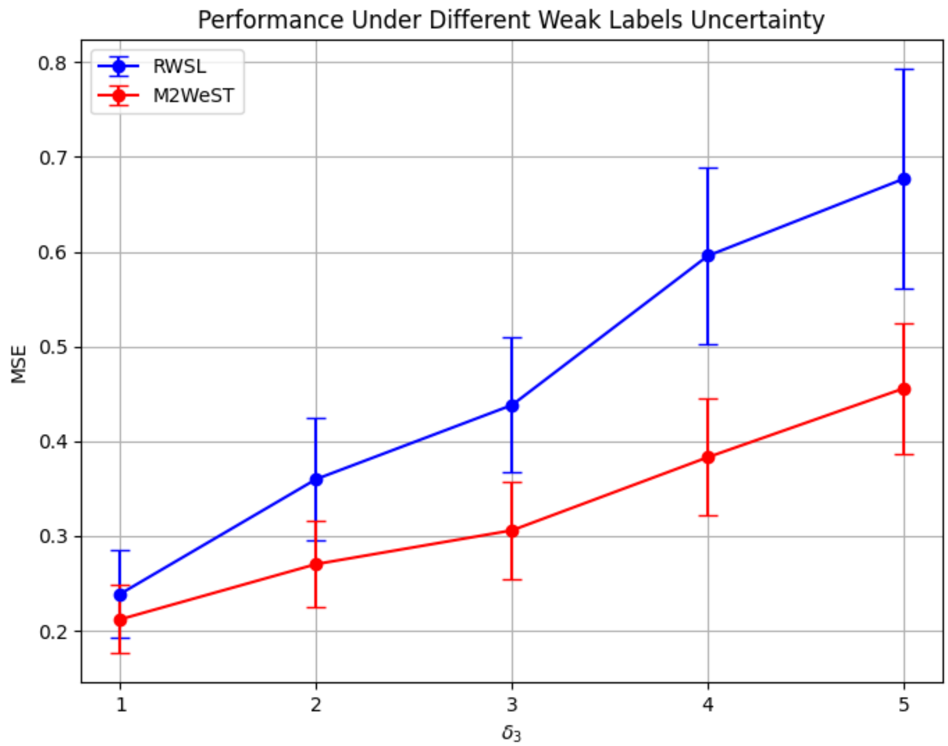

Figure 19. Average MSE performance on test data across different weak labels uncertainty.

4.5 Real data application

In this section, we apply the proposed M2WeST model to two specific case studies for predicting the MDS-UPDRS of PD patients: 1) Using the tapping data collected by their mobile phones 2) Using voice recordings recorded on their mobile phones. These activities capture key aspects of PD progression and serve as valuable indicators of disease severity.

4.5.1 Description of mPower Data

The data utilized in this study was collected through the mPower project (Bot et al., 2016), a large-scale mobile health initiative designed for the telemonitoring of Parkinson's disease

(PD) patients. Each patient enrolled in the study was asked to use the mPower app installed on their smartphone to perform specific tasks, including tapping and voice activities.

For the tapping activity, patients performed the task three times per day—before medication, after medication, and at another designated time. The task required patients to alternately tap two fixed points on the screen with two fingers from the same hand for a period of 20 seconds as shown in Figure Tap. This activity was designed to measure motor functions such as dexterity, speed, precision, and steadiness (Bot et al., 2016). The raw data from each session was captured as time series x-y coordinates for each tap. From this data, 43 features were extracted, including the total number of taps and the mean inter-tap interval, based on prior studies (Taylor et al., 2005; Arora et al., 2015; Kassavetis et al., 2016). These features were used as input to the M2WeST model to predict PD severity.

For the voice activity, patients were instructed to sustain the vowel sound "aaaah" into their phone’s microphone for a period of up to 10 seconds. This activity was designed to capture vocal characteristics affected by PD, such as changes in amplitude, frequency, and voice quality. The raw voice data was recorded as a time series signal, from which 339 features were extracted, including amplitude variation (shimmer), frequency variation (jitter), and signal-to-noise ratio (Tsanas et al., 2011; Yoon and Li, 2019). These features are crucial for assessing vocal impairments, which are common in PD patients, and were also incorporated into the M2WeST model to improve predictions.

In addition to the activity data, participants were asked to complete the Movement Disorder Society-Unified Parkinson's Disease Rating Scale (MDS-UPDRS), a clinical assessment for measuring PD severity. These scores were collected on a monthly basis and ranged from 0 (no disability) to 64 (complete disability). The MDS-UPDRS scores served as the response variable for the supervised learning tasks in the M2WeST model, providing a basis for evaluating disease severity.

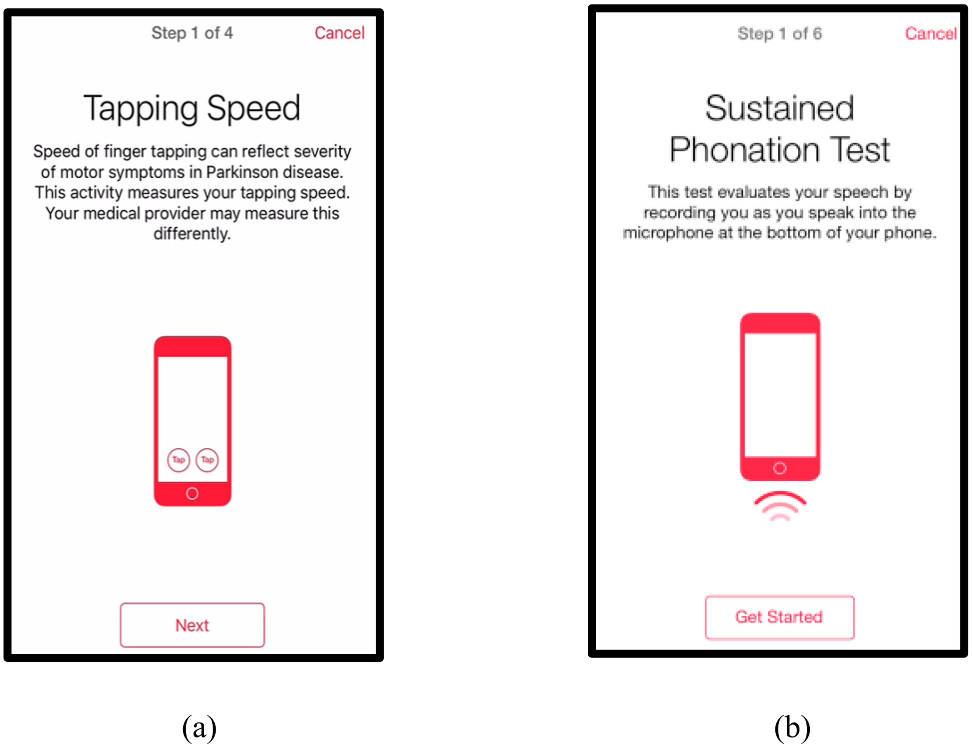

Figure 20. The mPower app instructs a patient to activities (Bot et al., 2016).

4.5.2 Patient Selection and Data Preprocessing

To ensure the reliability of the dataset used in the M2WeST model, a patient selection process was implemented. Since the mPower data is self-collected, variability in data quality is expected due to differences in patient adherence and engagement with the study. Following a patient selection process similar to previous work (F. Alenezi et al., 2022), we included only those patients who completed a sufficient number of tapping and voice tasks. Additionally, we selected patients who demonstrated significant medication response based on hypothesis testing, as supported by Chaibub Neto et al. (2016).

For the tapping activity, patients who performed at least 30 sessions before medication and 30 sessions after medication were selected. This resulted in a dataset containing 558 labeled samples, with 2784 samples recorded before medication and 2299 samples after medication. A total of 31,458 valid ranking pairs were generated from this data, representing relative changes in symptom severity due to the effects of medication.

Similarly, for the voice activity, patients were required to meet the same minimum task completion criteria. This provided 496 labeled samples, with 2695 samples recorded before medication and 2137 samples after medication. From this dataset, 29,992 ranking pairs were created to support the weakly supervised learning process in the M2WeST model.

These ranking pairs represent relative changes in PD symptom severity and play a crucial role in enhancing the model's ability to predict disease progression by leveraging weakly supervised learning techniques. In combination with labeled MDS-UPDRS data, the ranking-based weak supervision helps M2WeST overcome the challenge of limited labeled data, which is often sparse and difficult to obtain.

To address the scarcity of MDS-UPDRS labels, linear interpolation was applied to estimate missing values, following the approach supported by Parkinson's disease literature (Tsanas et al., 2009). This method allowed for the generation of weekly estimates of the MDS-UPDRS scores, ensuring that sufficient labeled data was available to complement the weakly supervised learning process.

By utilizing both tapping and voice data, along with weakly supervised learning and transfer learning techniques, the M2WeST model effectively integrates multi-source data, allowing for robust and accurate predictions of PD severity, even in the presence of label scarcity.

4.5.3 Model Performance Evaluation

We evaluated the M2WeST model against several benchmark models, including Single Learning (SL), TrAdaBoost, CORAL, Mean Teacher, RWSL, and ATBTL. The models were assessed for their ability to predict MDS-UPDRS scores using both tapping and voice data. Performance metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Pearson Correlation (Corr) were used. The results were obtained using 5-fold cross-validation, with the mean and standard deviation for each metric reported.

For tapping data, M2WeST exhibited the best overall performance as seen in Table. 9, delivering the lowest MSE, the lowest RMSE, and the highest correlation. TrAdaBoost, Mean Teacher and CORAL all performed worse than SL. These methods were unable to generalize effectively in the tapping data context due to their lack of adaptability to the variability across patient-specific symptoms. Also, they do not leverage target and source data weak labels, limiting its ability to improve predictions. RWSL outperformed SL by benefiting from weak supervision. ATBTL also outperformed SL by using multi-source learning, though the absence of weak supervision hindered its ability to maximize the value of unlabeled data.

Table 9. CV metrics of MDS-UPDRS on tapping features.

	MSE		**RMSE**		**Corr**	
Method	Mean	Std	Mean	Std	Mean	Std
SL	1.621	0.333	1.106	0.111	0.477	0.107
TrAdaboost	2.006	0.347	1.212	0.107	0.417	0.151
CORAL	1.734	0.343	1.136	0.120	0.407	0.153
Mean Teacher	1.778	0.896	1.111	0.230	0.539	0.115
RWSL	1.409	0.265	1.025	0.114	0.544	0.125
ATBTL	1.530	0.473	1.056	0.167	0.546	0.130
M2WeST	**1.229**	**0.242**	**0.936**	**0.106**	**0.641**	**0.096**

On voice data, M2WeST again outperformed all benchmarks as seen in Table. 10, with the lowest MSE and highest correlation. TrAdaBoost and CORAL again struggled due to negative transfer and poor adaptation to variability in vocal features. Mean Teacher performed slightly better than SL but it's limited since it uses the target patient's data without leveraging weak labels. RWSL and ATBTL again outperformed SL but did not reach M2WeST's level of performance.

Table 10. CV metrics of MDS-UPDRS on voice features.

	MSE		**RMSE**		**Corr**	
Method	Mean	Std	Mean	Std	Mean	Std
SL	2.257	0.624	1.238	0.150	0.439	0.113
TrAdaboost	3.546	0.369	1.526	**0.096**	0.202	0.159
CORAL	2.918	0.497	1.389	0.128	0.161	0.161
Mean Teacher	2.195	0.440	1.209	0.131	0.466	0.138
RWSL	1.956	**0.403**	1.160	0.129	0.517	0.146
ATBTL	2.023	0.532	1.163	0.165	0.544	0.116
M2WeST	**1.708**	0.413	**1.064**	0.132	**0.626**	**0.091**

The main reasons for the failure of benchmark models stem from how they handle data integration. TrAdaBoost and CORAL suffered from negative transfer by using data from source patients without addressing the variability between them, leading to poor generalization. Mean Teacher, while using unlabeled data from the target patient, failed to improve predictions because it did not incorporate data from other patients or use weak supervision to improve the quality of those unlabeled samples. SL avoided negative transfer but missed out on the advantages of multi-source learning and weak supervision, limiting its generalization capabilities. RWSL performed better by leveraging weak supervision, but its failure to integrate source data reduced its potential to only patients with significant weak data. In contrast, ATBTL leveraged multi-source learning but struggled to fully benefit from patients with very limited labeled data and abundant unlabeled data due to the lack of a weak supervision mechanism.

M2WeST's success lies in its balanced integration of multi-source learning and weak supervision, which allows it to avoid harmful information from heterogeneous patients while utilizing patient with very limited labeled samples by integrating both labeled and weakly labeled data. This makes M2WeST the most robust and effective model for predicting Parkinson's disease severity across both tapping and voice data.

4.6 Conclusion

In this chapter, we introduced the Multi-source Multi-task Weakly Supervised Transfer Learning (M2WeST) method, a novel approach designed to address the challenges of integrating labeled and weakly labeled data from multiple heterogeneous sources. Our contributions focused on advancing telemonitoring applications, particularly in the context of predicting disease severity for Parkinson's Disease (PD) patients.

M2WeST demonstrated significant improvements in predictive performance by effectively leveraging weak supervision and handling patient heterogeneity across different data sources. Through a combination of strong and weak learning tasks, our model was able to make use of both labeled and ranked pairs data, enhancing its robustness in scenarios with limited labeled data while avoiding the pitfalls of negative transfer.

The proposed algorithm efficiently optimized the nonconvex formulation of the M2WeST model, ensuring scalability across multiple sources, making it applicable to real-world data scenarios. In simulations and case studies using the mPower dataset, M2WeST consistently outperformed other benchmark models, demonstrating its superiority in scenarios with high heterogeneity, limited labeled data, and high uncertainty in weak labels.

Despite these successes, there are several areas where future research can further enhance the M2WeST model. First, while we demonstrated the model's robustness in handling weak supervision, there remain opportunities to explore alternative weak supervision mechanisms that incorporate more patient-specific features to improve the accuracy of weak label generation. Additionally, the inclusion of temporal data could extend M2WeST's utility, allowing for predictions that dynamically adapt to changes in a patient's health condition over time.

Moreover, the mPower dataset used in this study was sourced from a controlled research environment, and the challenge of generalizing M2WeST to more diverse and potentially noisier real-world settings remain. Issues such as missing data, non-adherence, and self-reported inaccuracies will need to be addressed through robust preprocessing techniques to ensure the reliability of model outputs in clinical practice.

Finally, to unlock the full potential of M2WeST in clinical settings, further clinical validation is required through well-designed trials with clinicians. This will help demonstrate how M2WeST can provide actionable insights that enhance clinical decision-making. The ability of the model to deliver insights that align with and support traditional clinical assessments will be critical to its adoption.

In conclusion, M2WeST represents a powerful and scalable solution to the problem of limited data and patient heterogeneity in telemonitoring applications for PD. Its ability to leverage multiple data sources and leverage knowledge from abundant unlabeled data makes it a valuable tool for improving the accuracy of disease severity predictions, ultimately contributing to better patient management and outcomes in healthcare.

CHAPTER 5. CONCLUSION

This thesis introduces three innovative contributions to weakly supervised learning (WSL) and transfer learning, addressing critical challenges in healthcare telemonitoring and manufacturing systems. The work focuses on enhancing model performance in the presence of limited labeled data, integrating domain knowledge, and improving multi-source learning. These contributions span a range of applications, from predicting Parkinson's disease (PD) severity to improving manufacturing quality, and establish new methodologies for leveraging weak supervision. This section summarizes the major contributions of the thesis and outlines future research directions.

5.1 Contributions Summary

Chapter 2 develops a Ranking-based Weakly Supervised Learning (RWSL) model designed to predict Parkinson's disease severity from mobile-collected tapping activity data. This model incorporates both labeled and ranked data, outperforming traditional methods like ridge regression and RankSVM, particularly in scenarios with limited labeled data. By using ranked pairs generated from medication effects, RWSL leverages domain knowledge to improve predictive performance. While RWSL shows strong potential, several challenges remain, including addressing missing data due to patient non-adherence, exploring other criteria for creating ranked pairs, and preparing the model for clinical implementation through extensive clinical validation. Future directions could involve improving data preprocessing techniques and incorporating new PD severity scales as they are developed.

Chapter 3 introduces a Physics-Informed Weakly Supervised Learning (PWL) framework, which combines weak supervision through partial physical knowledge with traditional machine learning models for quality prediction in manufacturing processes. This framework leverages weak labels derived from physical models and constraints, allowing the model to perform well even when labeled data is sparse or expensive to acquire. By integrating physics-based predictions with data-driven models, the PWL approach optimizes both the model and calibration parameters, demonstrating robustness across various noise levels and labeled sample sizes. The use of weak labels based on physical knowledge allows the model to make more accurate predictions in environments where labeled data is limited. Future research could explore applying this method to more complex or higher-dimensional outputs, or address intrinsic uncertainties in physics-based models using techniques like probabilistic outputs or weighted loss functions.

Chapter 4 introduces the Multi-source Multi-task Weakly Supervised Transfer Learning (M2WeST) method, which integrates data from multiple heterogeneous sources to improve PD severity prediction. M2WeST addresses the challenges of patient heterogeneity and limited labeled data by combining weak supervision with transfer learning across multiple patients. Weak labels are derived from domain-specific knowledge, such as the effects of medication, allowing the model to rank data points from the same patient based on relative severity. This method allows the model to leverage both strongly labeled and weakly labeled data across heterogeneous patients, improving its predictive performance while avoiding negative transfer. The integration of multiple sources enables M2WeST to scale effectively and generalize well to new patients. Future research could explore incorporating more patient-specific features into the weak supervision mechanism, integrating temporal data, and improving the generalizability of M2WeST to clinical setting.

5.2 Future Research Directions

A promising direction for future research is applying the RWSL and M2WeST models to other chronic diseases, such as Alzheimer's or diabetes, where weak supervision and transfer learning could address similar challenges of limited labeled data and patient heterogeneity. Additionally, integrating temporal data and multi-modal data (e.g., combining voice signal, movement videos, and medical sensor data) could further enhance the model's predictive accuracy and utility for disease progression monitoring.

In the manufacturing sector, extending the PWL model to other critical manufacturing processes, such as predictive maintenance for equipment health, where data sparsity and uncertainty are common challenges, would be a valuable next step. Additionally, integrating real-time sensor data from manufacturing processes could allow for adaptive quality predictions, enhancing the system's ability to respond to process variations in real-time. Exploring advanced optimization techniques, such as federated learning, to calibrate physics-based models across multiple production lines or plants with distributed data sources could further improve scalability and model accuracy.

Exploring alternative weak supervision mechanisms, such as incorporating more granular patient-specific information or integrating additional medical knowledge for ranking criteria, could further improve the performance of both RWSL and M2WeST. Investigating how these models can handle missing data, non-adherence, and self-reported inaccuracies, particularly in uncontrolled, real-world environments, will be key to ensuring their reliability in clinical practice.

On the theoretical side, formalizing the weak supervision frameworks for RWSL, M2WeST, and PWL would strengthen the foundation of these methods. This could involve developing rigorous guarantees for generalization, robustness, and convergence, especially in scenarios with limited labeled data and high heterogeneity. Investigating the theoretical bounds of these models under different levels of noise, missing data, and weak label quality would provide deeper insights into their performance and reliability.

On the practical side, while these models have already been tested with real-world data, the next step would be to focus on scaling up their deployment in clinical and industrial settings. For telemonitoring, ensuring data security and addressing healthcare compliance regulations will be key for broader clinical adoption and patient trust. In manufacturing, applying the PWL model in real-time production environments will test its ability to enhance adaptive quality predictions and improve overall process efficiency at scale.

The methodologies presented in this thesis address critical challenges such as limited labeled data in telemonitoring applications and the need for improved predictive models in manufacturing systems. By leveraging weak supervision, integrating physics-based knowledge, and applying multi-source learning, these models offer substantial improvements in accuracy and robustness, especially in complex, real-world settings. The contributions of this thesis pave the way for future advancements in weak supervision methodologies, transfer learning, and the incorporation of domain-specific knowledge, offering new opportunities for improving patient management and optimizing manufacturing processes. These approaches hold promise for broader application across healthcare and industrial domains, where data limitations and complexity remain significant barriers to accurate prediction and decision-making.

APPENDIX A. SUPPLEMENTARY MATERIALS OF CHAPTER 3

Thes supplementary materials sections are structured as follows: section. A.1 details proposition related to the $\boldsymbol{g}$ update, section. A.2 details proposition related the $\boldsymbol{\theta}$ update, and section. A.3 details proposition related to identifiability requirement.

A.1 $\boldsymbol{g}$ *Update*

Proposition 1. *Let* $f_1(\boldsymbol{g}) = \|\boldsymbol{Y} - \boldsymbol{H_1 g} - \boldsymbol{B_1 d}\|_F^2$, $f_2(\boldsymbol{g}) = \lambda_1 \|\boldsymbol{H_2 g} - \widetilde{\boldsymbol{\eta}}\boldsymbol{/}\|_F^2$, $f_3(\boldsymbol{g}) = \lambda_2 \|\boldsymbol{g}\|_{\mathbf{1}}$, *and* $f_4(\boldsymbol{g}) = \lambda_3 \sum_{\boldsymbol{q=1}}^{(\boldsymbol{q_1+q_3})} \|\boldsymbol{g_q}\|_{\mathbf{2}}$. *The proximal operators of proximal operators of* $\Delta \boldsymbol{f_1}, \ldots \Delta \boldsymbol{f_4}$ *are given as follows: Let* $\boldsymbol{z}$ *and* $\boldsymbol{a}$ *have the same size as* $\boldsymbol{g}$.

1) *If* $\boldsymbol{z} = \mathrm{prox}_{\Delta f_1}[\boldsymbol{a}]$, *we have*

   $\mathrm{prox}_{\Delta f_1}[\boldsymbol{a}] = \left(\mathbf{I} + \Delta \boldsymbol{H_1}^\top \boldsymbol{H_1}\right)^{\mathbf{-1}} \left(\boldsymbol{a} + \Delta \boldsymbol{H_1}^\top (\mathbf{Y} - \boldsymbol{B_1 d})\right)$, *where* $\mathbf{I}$ *is the identity matrix of order p.*

2) *If* $\boldsymbol{z} = \mathrm{prox}_{\Delta f_2}[\boldsymbol{a}]$, *we have*

   $\mathrm{prox}_{\Delta f_2}[\boldsymbol{a}] = \left(\mathbf{I} + \lambda_1 \Delta \boldsymbol{H_2}^\top \boldsymbol{H_2}\right)^{\mathbf{-1}} \left(\boldsymbol{a} + \lambda_1 \Delta \boldsymbol{H_2}^\top \widetilde{\boldsymbol{\eta}}\right)$, *where* $\mathbf{I}$ *is the identity matrix of order p.*

3) *If* $\boldsymbol{z} = \mathrm{prox}_{\Delta f_3}[\boldsymbol{a}]$, *we have*

$$\mathrm{prox}_{\Delta f_3}[\boldsymbol{a}] = S_{\lambda_2 \Delta}(\boldsymbol{a}) = \begin{cases} \boldsymbol{a} - \boldsymbol{\lambda_2}\Delta, \boldsymbol{a} > \boldsymbol{\lambda_2}\Delta \\ \mathbf{0}, -\boldsymbol{\lambda_2}\Delta \leq \boldsymbol{a} \leq \boldsymbol{\lambda_2}\Delta \\ \boldsymbol{a} + \boldsymbol{\lambda_2}\Delta, \boldsymbol{a} \leq \boldsymbol{\lambda_2}\Delta\,. \end{cases}$$

4) *If* $\boldsymbol{z} = \mathrm{prox}_{\Delta f_4}[\boldsymbol{a}]$, *we have*

$$\mathrm{prox}_{\Delta f_4}[\boldsymbol{a}] = \left[\mathbf{1} - \frac{\boldsymbol{\lambda_3}\Delta}{\|\boldsymbol{a}\|_{\mathbf{2}}}\right]_+ \boldsymbol{a}\,.$$

Proof

The proximal operator of Δf_1: The $f_1(\boldsymbol{g})$ can be written as a least square form as follows:

$$f_1(\boldsymbol{g}) = \|\mathbf{Y} - \boldsymbol{H_1 g} - \boldsymbol{B_1 d}\|_F^2 = \|(\mathbf{Y} - \boldsymbol{B_1 d}) - \boldsymbol{H_1 g}\|_F^2 .$$

According to (Parikh & Boyd, 2014), the proximal operator for $q(x) = \|c - \mathrm{Px}\|_2^2$ can be written as follows:

$$\mathrm{prox}_{\Delta q}[\mathbf{x}] = \operatorname*{argmin}_{\mathbf{t}} \left\{ \Delta q(\mathbf{t}) + \frac{1}{2}\|\mathbf{t} - \mathbf{x}\|_2^2 \right\} = (\mathbf{I} + \Delta \mathbf{P}^\top \mathbf{P})^{-1}(\mathbf{x} + \Delta \mathbf{P}^\top \mathbf{c})$$

In our setting, $\mathbf{P} = \boldsymbol{H_1}, \mathbf{c} = (\mathbf{Y} - \boldsymbol{B_1 d})$, hence we have,

$$\mathrm{prox}_{\Delta f_1}[\boldsymbol{a}] = \left(\mathbf{I} + \Delta \boldsymbol{H_1}^\top \boldsymbol{H_1}\right)^{-1}(\boldsymbol{a} + \Delta \boldsymbol{H_1}^\top (\mathbf{Y} - \boldsymbol{B_1 d}))$$

The proximal operator of Δf_2: The $\boldsymbol{f_2}(\boldsymbol{g})$ can be written in a similar format as $\boldsymbol{f_1}(\boldsymbol{g})$, hence, by setting $\mathbf{P} = \boldsymbol{H_2}, \mathbf{c} = \widetilde{\boldsymbol{\eta}}$, we have

$$\mathrm{prox}_{\Delta f_2}[\boldsymbol{a}] = \left(\mathbf{I} + \lambda_1 \Delta \boldsymbol{H_2}^\top \boldsymbol{H_2}\right)^{-1}(\boldsymbol{a} + \lambda_1 \Delta \boldsymbol{H_2}^\top \widetilde{\boldsymbol{\eta}})$$

The proximal operator of Δf_3: This can be obtained from the proximal operator of l_1 norm, and it is given in (Parikh & Boyd, 2014) and can be applied in our setting as follows:

$$\mathrm{prox}_{\Delta f_3}[\boldsymbol{a}] = \boldsymbol{S}_{\lambda_2 \Delta}(\boldsymbol{a}) = \begin{cases} \boldsymbol{a} - \lambda_2 \Delta, \boldsymbol{a} > \lambda_2 \Delta \\ \mathbf{0}, -\lambda_2 \Delta \leq \boldsymbol{a} \leq \lambda_2 \Delta \\ \boldsymbol{a} + \lambda_2 \Delta, \boldsymbol{a} \leq \lambda_2 \Delta \end{cases}$$

The proximal operator of Δf_4: This term is the sum of l_2 norms. By the separable property of the proximal operator, the result follows from the proximal operator of l_2 norms given in (Parikh & Boyd, 2014) as:

$$\mathrm{prox}_{\Delta f_4}[\boldsymbol{a}] = \left[1 - \frac{\lambda_3 \Delta}{\|\boldsymbol{a}\|_2}\right]_+ \boldsymbol{a}$$

A.2 $\boldsymbol{\theta}$ *Update*

Proposition 2. *Let* $f_1(\boldsymbol{g}) = \|\mathbf{Y} - \boldsymbol{H_1 g} - \boldsymbol{B_1 d}\|_F^2$, $f_2(\boldsymbol{g}) = \lambda_1 \|\boldsymbol{H_2 g} - \widetilde{\boldsymbol{\eta}}\|_F^2$ *and* $f_3(\boldsymbol{g}) = \sum_{\boldsymbol{k}}^{\boldsymbol{q_3}} \boldsymbol{I}_{\omega_{L_k} \leq \theta_k \leq \omega_{U_k}}$. *The proximal operators of proximal operators of* $\Delta f_1, \ldots \Delta f_3$ *are given as follows: Let* $\boldsymbol{z}$ *and* $\boldsymbol{a}$ *have the same size as* θ.

1) *If* $\boldsymbol{z} = \mathrm{prox}_{\Delta f_1}[\boldsymbol{a}]$, *we have*

   $\mathrm{prox}_{\Delta f_1}[\boldsymbol{a}] = (\mathbf{I} + \Delta \boldsymbol{g}\boldsymbol{g}^\top)^{-\mathbf{1}}(\boldsymbol{a} + \Delta \boldsymbol{g}(\mathbf{Y} - \boldsymbol{B_1 d})^\top)$, *where* $\mathbf{I}$ *is the identity matrix of order p.*

2) *If* $\boldsymbol{z} = \mathrm{prox}_{\Delta f_2}[\boldsymbol{a}]$, *we have*

   $\mathrm{prox}_{\Delta f_2}[\boldsymbol{a}] = (\mathbf{I} + \lambda_1 \Delta \boldsymbol{g}\boldsymbol{g}^\top)^{-\mathbf{1}}(\boldsymbol{a} + \lambda_1 \Delta \boldsymbol{g}\widetilde{\boldsymbol{\eta}}^\top)$, *where* $\mathbf{I}$ *is the identity matrix of order p.*

3) *If* $\boldsymbol{z} = \mathrm{prox}_{\Delta f_3}[\boldsymbol{a}]$, *we have*

$$z_k = \mathrm{prox}_{\Delta f_3}[a_k] = max\left\{\omega_{L_k}, min\left\{a_k, \omega_{U_k}\right\}\right\}.$$

Proof

The proximal operator of $\Delta \boldsymbol{f_1}$: The $f_1(\boldsymbol{\theta})$ can be written as a least square form and the proximal operator can be found similar to f_1 in proposition 4 as follows:

By setting, $\mathbf{P} = \boldsymbol{g}^{\top}, \mathbf{c} = (\mathbf{Y} - \boldsymbol{B_1 d})^{\top},\ \mathbf{x} = \ \boldsymbol{H}_{\mathbf{1}}^{\top}$, we have

$$\mathrm{prox}_{\Delta f_1}[\boldsymbol{a}] = (\mathbf{I} + \Delta \boldsymbol{g}\boldsymbol{g}^{\top})^{-1}(\boldsymbol{a} + \Delta \boldsymbol{g}(\mathbf{Y} - \boldsymbol{B_1 d})^{\top})$$

<u>The proximal operator of $\Delta \boldsymbol{f_2}$:</u> The $\boldsymbol{f_2}(\boldsymbol{\theta})$ can be written in a similar format as $\boldsymbol{f_1}(\boldsymbol{g})$, hence, by setting $\mathbf{P} = \boldsymbol{g}^{\top}, \mathbf{c} = \widetilde{\boldsymbol{\eta}}^{\top},\ \mathbf{x} = \ \boldsymbol{H}_{\mathbf{2}}^{\top}$, we have

$$\mathrm{prox}_{\Delta f_2}[\boldsymbol{a}] = (\mathbf{I} + \lambda_1 \Delta \boldsymbol{g}\boldsymbol{g}^{\top})^{-1}(\boldsymbol{a} + \lambda_1 \Delta \boldsymbol{g}\widetilde{\boldsymbol{\eta}}^{\top})$$

<u>The proximal operator of $\Delta \boldsymbol{f_3}$:</u> By the separable property of the proximal operator, the result follows from the proximal operator of box constraints given in (Parikh & Boyd, 2014) as follows:

$$z_k = \mathrm{prox}_{\Delta f_3}[a_k] = max\left\{\omega_{L_k}, min\left\{a_k, \omega_{U_k}\right\}\right\}$$

Another aspect of our algorithm that needs to be discussed is identifiability issues that may arise in evaluating terms on the form $\boldsymbol{H_1 g}$ appearing in (14) for instance. This problem can be cast as a dictionary learning problem since we are in some sense trying to learn a basis matrix, $\boldsymbol{H_1}$, and a sparse coefficints vector, $\boldsymbol{g}$, The success of the dictionary learning method does not require that the patterns in $\boldsymbol{H_1}$ to be orthogonal like PCA, and the reconstructed signal is often evaluated accurately under mild conditions. However, this does not guarantee that dictionary learning can uniquely recover sparse data from samples. Meaning, the uniqueness of the learned basis and sparse coefficients vector is not always guaranteed which may create an identifiability issue. The detailed statistical assumptions that guarantee the identifiability is stated in proposition 6 and further discussed in Appendix A.3.

A.3 *Identifiability Requirement*

Proposition 3. *Let assume $\boldsymbol{g}$ is a vector with at most $k \ll m$ nonzero entries, called k-sparse. Assume the basis $\boldsymbol{H}$ is Spark on the vectors $\boldsymbol{g}$ such that:*

$$\boldsymbol{H}\boldsymbol{g}_{\mathbf{1}} = \boldsymbol{H}\boldsymbol{g}_{\mathbf{2}} \ for \ \ \boldsymbol{g}_{\mathbf{1}}, \boldsymbol{g}_{\mathbf{2}} \in \mathbb{R}^{\boldsymbol{m}} \Rightarrow \boldsymbol{g}_{\mathbf{1}} = \boldsymbol{g}_{\mathbf{2}}$$

Assuming $\boldsymbol{H}$ is given, and our signal is constructed such that:

$$\widetilde{\mathbf{Y}} = \boldsymbol{H}\boldsymbol{g}$$

where $\widetilde{\boldsymbol{Y}} = \boldsymbol{Y} - \boldsymbol{B}_{\mathbf{1}}\boldsymbol{d}$

a lower bound can be set on the number of samples, n, of $\widetilde{\boldsymbol{Y}}$ that guarantees unique recovery of sparse vector $\boldsymbol{g}$ from a signal $\widetilde{\boldsymbol{Y}}$ with a given basis matrix $\boldsymbol{A}$ such that:

$$n \geq Ck \log\left(\frac{m}{k}\right)$$

where C is a constant independent of m, n, and k with more detailed discussions and proofs included in (Baraniuk et al., 2008).

In the case $\boldsymbol{H}$ is unknown, assuming the above holds, we assume we can subsample $\boldsymbol{N}$ similar samples of $\widetilde{\boldsymbol{Y}}$ from our system such that we have $\breve{\boldsymbol{Y}} = \{\widetilde{\boldsymbol{Y}}_{\mathbf{1}}, \dots, \widetilde{\boldsymbol{Y}}_{\boldsymbol{N}}\}$. Then, can guarantee the uniqueness of both $\boldsymbol{g}$ and $\boldsymbol{H}$ if we can guarantee a lower bound on the number of required subsamples $\boldsymbol{N}$ to guarantee uniqueness of $\boldsymbol{A}$ with probability $(\mathbf{1} - \boldsymbol{\beta})$ as follows:

$$\boldsymbol{N} = \frac{\boldsymbol{k} + \mathbf{1}}{\boldsymbol{\beta}} \binom{\boldsymbol{m}}{\boldsymbol{k}}$$

Proof:

Consider problem (13) and (14) in our formulation, since λ_1 control how much we emphasize on the physics-based model and can be considered as a regularization term, without loss of generality, we consider the case where $\lambda_1 \approx 0$, and reformulated the two problems as follows:

$$\min_{g,\theta} \left\| \|\mathbf{Y} - \boldsymbol{H_1 g} - \boldsymbol{B_1 d}\|_F^2 \right\|_F^2 + \boldsymbol{\lambda}_2 \|\boldsymbol{g}\|_1 + \boldsymbol{\lambda}_3 \sum_{q=1}^{(q_1+q_3)} \|\boldsymbol{g}_q\|_2$$

$$\Rightarrow \min_{g,\theta} \left\| \|\widetilde{\mathbf{Y}} - \boldsymbol{H_1 g}\|_F^2 \right\|_F^2 + \boldsymbol{\lambda}_2 \|\boldsymbol{g}\|_1 + \boldsymbol{\lambda}_3 \sum_{q=1}^{(q_1+q_3)} \|\boldsymbol{g}_q\|_2,$$

where $\widetilde{\mathbf{Y}} = \mathbf{Y} - \boldsymbol{B_1 d}$.

Our main concern here is under what assumptions when we evaluate our signal $\widetilde{\mathbf{Y}}$**,** we can assume guarantees about the uniqueness of the learned basis $\boldsymbol{H_1}$ and sparse coefficients vector $\boldsymbol{g}$. Assume that observed data $\widetilde{\mathbf{Y}}$ were generated as:

$$\widetilde{\mathbf{Y}} = \boldsymbol{Aa} \tag{20}$$

using a fixed $n \times m$ basis matrix $\boldsymbol{A}$ and sparse m dimensional latent representations $\boldsymbol{a}$. First, we address the accuracy of the recovered $\boldsymbol{a}$**,** which mainly concerns the $\boldsymbol{g}$ update in Algorithm 1 of our problem. This can be done using theory from compressed sensing (CS) (Donoho, 2006), where they have investigated conditions under which the sparse latent representations, $\boldsymbol{a}$, can be recovered from data (Baraniuk et al., 2008). Given an m-dimensional vector $\boldsymbol{a}$ with at most $k \ll m$ nonzero entries. Such vectors a are called k-*sparse*. And given a known basis matrix $\boldsymbol{A}$**,** a typical condition is:

$$\boldsymbol{A}\boldsymbol{a}_1 = \boldsymbol{A}\boldsymbol{a}_2 \text{ for k sparse } \boldsymbol{a}_1, \boldsymbol{a}_2 \in \mathbb{R}^{\boldsymbol{m}} \Rightarrow \boldsymbol{a}_1 = \boldsymbol{a}_2 \tag{21}$$

known as the *spark condition*. Another condition is related to the number of samples, n, of $\widetilde{\mathbf{Y}}$ needs to satisfy:

$$n \geq Ck \log\left(\frac{m}{k}\right), \tag{22}$$

where C is a constant independent of $m, n,$ and k with more detailed discussions and proofs included in (Baraniuk et al., 2008).

Conditions (23) and (24) describe when it is possible to recover a sparse vector $\boldsymbol{a}$ from a signal $\widetilde{\mathbf{Y}}$ with a given or known basis matrix $\boldsymbol{A}$.

Second, when the $\boldsymbol{A}$ is unknown, it is more difficult to guarantee the uniqueness of both $\boldsymbol{a}$ and $\boldsymbol{A}$ and requires more conditions. To show this, if $\boldsymbol{P}$ is a permutation matrix and $\boldsymbol{D}$ is an invertible diagonal matrix, then

$$\boldsymbol{A}\boldsymbol{a} = (\boldsymbol{A}\boldsymbol{D}^{-1}\boldsymbol{P}^{\top})(\boldsymbol{P}\boldsymbol{D}\boldsymbol{a}), \tag{23}$$

where $\boldsymbol{P}$ is a permutation matrix with binary entries and precisely one 1 in each row and column such that $\boldsymbol{P}\boldsymbol{P}^{\top} = \boldsymbol{P}^{\top}\boldsymbol{P} = \boldsymbol{I}$. Hence, if $\boldsymbol{A}$ is unknown, we cannot discriminate which of $\boldsymbol{a}$ or $\boldsymbol{P}\boldsymbol{D}\boldsymbol{a}$ (respectively, $\boldsymbol{A}$ or $\boldsymbol{A}\boldsymbol{D}^{-1}\boldsymbol{P}^{\top}$) was the original sparse vector or basis matrix, respectively. Suppose that the true signal $\widetilde{\mathbf{Y}}$ is created according to (20) and we are trying to evaluate it as follows:

$$\widehat{\mathbf{Y}} = \boldsymbol{B}\boldsymbol{b} \tag{24}$$

So, we can guarantee the uniqueness of both $\boldsymbol{a}$ and $\boldsymbol{A}$ if we can guarantee that $\boldsymbol{Aa} = \boldsymbol{Bb}$ under scenario such that in (23). Suppose in our setting that $\widetilde{\mathbf{Y}}$ is sampled from the system and we can subsample $\boldsymbol{N}$ similar samples such that we have $\breve{\mathbf{Y}} = \{\widetilde{\mathbf{Y}}_{\mathbf{1}}, \dots, \widetilde{\mathbf{Y}}_{\boldsymbol{N}}\}$. Now assuming each subsample maybe associated with different active patterns in the basis matrix $\boldsymbol{A}$ hence may generate different vectors $\boldsymbol{a}$ such that $\breve{\boldsymbol{a}} = \{\boldsymbol{a}_{\mathbf{1}}, \dots, \boldsymbol{a}_{\boldsymbol{N}}\}$. Assuming (21) and (22) hold, a lower bound on the number of required subsamples $\boldsymbol{N}$ can be found in (Hillar & Sommer, 2015) to guarantee uniqueness of $\boldsymbol{A}$ with probability $(\mathbf{1} - \boldsymbol{\beta})$ as follows:

$$N = \frac{k+1}{\beta} \binom{m}{k} \tag{25}$$

This indicates that sparse matrix factorization of basis matrix $\boldsymbol{A}$ is unique for a large enough number of subsamples.

APPENDIX B. SUPPLEMENTARY MATERIALS OF CHAPTER 4

B.1 *The β Update*

Proposition 4. *Let* $f_1(\boldsymbol{\beta}) = \frac{1}{2N_l^t} \parallel \boldsymbol{Y} - \boldsymbol{H_l^t}\boldsymbol{\beta} \parallel^2$, $f_2(\boldsymbol{\beta}) = \frac{\lambda_w^t}{|\Omega_w^t|} l_{huber}(1 - \boldsymbol{H_w^t}\boldsymbol{\beta})$, $f_3(\boldsymbol{\beta}) = \lambda_1^t \ \|\boldsymbol{\beta}\|_1 + \frac{\lambda_2^t}{2} \ \|\boldsymbol{\beta}\|_2^2$. *The proximal operators of* $\Delta f_1, \dots \Delta f_3$ *are given as follows: Let* $\boldsymbol{v}$ *have the same size as* $\boldsymbol{\beta}$.

1) *For* $\mathrm{prox}_{\Delta f_1}[\boldsymbol{v}]$, *we have*

$\mathrm{prox}_{\Delta f_1}[\boldsymbol{v}] = \left(\mathbf{I} + \frac{\Delta}{N_l^t}(\boldsymbol{H_l^t})^\top \boldsymbol{H_l^t}\right)^{\boldsymbol{-1}} \left(\boldsymbol{v} + \frac{\Delta}{N_l^t}(\boldsymbol{H_l^t})^\top \boldsymbol{Y}\right)$, *where* **I** *is the identity matrix of order* d_h.

2) *For* $\mathrm{prox}_{\Delta f_2}[\boldsymbol{v}]$, *we have*

$$\mathrm{prox}_{\Delta f_2}[\boldsymbol{v}] = \begin{cases} \boldsymbol{v} & \text{if } \boldsymbol{v} \geq 1 \\ \dfrac{2\Delta \dfrac{\lambda_w^t}{|\Omega_w^t|} + \boldsymbol{v}}{2\Delta \dfrac{\lambda_w^t}{|\Omega_w^t|} + 1} & \text{if } 0 \leq \boldsymbol{v} < 1 \\ \boldsymbol{v} - 2\Delta \dfrac{\lambda_w^t}{|\Omega_w^t|} & \text{if } \boldsymbol{v} < 0 \end{cases}$$

3) *For* $\mathrm{prox}_{\Delta f_3}[\boldsymbol{v}]$, *we have*

$$\mathrm{prox}_{\Delta f_3}[\boldsymbol{v}] = \frac{1}{1 + \Delta\lambda_2^t} S_{\Delta\lambda_1^t}(\boldsymbol{v}), \mathit{where}\ S_{\Delta\lambda_1^t} = \begin{cases} \boldsymbol{v} - \Delta\lambda_1^t, & \boldsymbol{v} > \Delta\lambda_1^t \\ \boldsymbol{0}, & -\Delta\lambda_1^t \leq \boldsymbol{v} \leq \Delta\lambda_1^t \\ \boldsymbol{v} + \Delta\lambda_1^t, & \boldsymbol{v} \leq \Delta\lambda_1^t\,. \end{cases}$$

Proof

The proximal operator of Δf_1: The $f_1(\boldsymbol{\beta}) = \frac{1}{2N_l^t} \parallel \boldsymbol{Y} - \boldsymbol{H_l^t}\boldsymbol{\beta} \parallel^2$ is on least square form. According to Parikh and Boyd (2014), the proximal operator for $q(\boldsymbol{v}) = \|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{v}\|_2^2$ can be written as follows:

$$\text{prox}_{\Delta q}[\boldsymbol{v}] = \underset{\boldsymbol{p}}{argmin}\left\{\Delta q(\boldsymbol{p}) + \frac{1}{2}\|\boldsymbol{p} - \boldsymbol{v}\|_2^2\right\} = (\boldsymbol{I} + \Delta \boldsymbol{A}^\top \boldsymbol{A})^{-1}(\boldsymbol{v} + \Delta \boldsymbol{A}^\top \boldsymbol{b})$$

In our setting, $\mathbf{A} = \boldsymbol{H_l^t}, \mathbf{c} = \boldsymbol{Y},$ hence we can write the proximal as follows,

$$\text{prox}_{\Delta f_1}[\boldsymbol{v}] = \left(\mathbf{I} + \frac{\Delta}{N_l^t}(\boldsymbol{H_l^t})^\top \boldsymbol{H_l^t}\right)^{\mathbf{-1}}\left(\boldsymbol{v} + \frac{\Delta}{N_l^t}(\boldsymbol{H_l^t})^\top \boldsymbol{Y}\right)$$

The proximal operator of Δf_2: The $f_2(\boldsymbol{\beta}) = \frac{\lambda_w^t}{|\Omega_w^t|} l_{huber}(1 - \boldsymbol{H_w^t}\boldsymbol{\beta})$ proximal can be found after finding the proximal operator of Huber loss.

$$L_{huber}(t) = \begin{cases} 0 & \text{if } \ t > 1 \\ (1-t)^2 & \text{if } \ 0 \le t < 1 \\ (1-2t) & \text{if } \ t < 0 \end{cases}$$

We can derive the proximal for each piece of the function as follows:

(1) $t \geq 1$:

The function is constantly zero, $L_{huber}(t) = 0$.

The minimization problem becomes: $\text{prox}_{\Delta q}[\boldsymbol{v}] = \underset{\boldsymbol{p}}{argmin}\left\{\Delta L_{huber}(\boldsymbol{p}) + \frac{1}{2}\|\boldsymbol{p} - \boldsymbol{v}\|_2^2\right\}$,

minimum $\boldsymbol{p} = \boldsymbol{v}$.

(2) $0 \leq t < 1$

The function is constantly zero, $L_{huber}(t) = (1-t)^2$.

The minimization problem becomes: $\text{prox}_{\Delta q}[\boldsymbol{v}] = \underset{\boldsymbol{p}}{argmin}\left\{\Delta L_{huber}(\boldsymbol{p}) + \frac{1}{2}\|\boldsymbol{p}-\boldsymbol{v}\|_2^2\right\}$,

$$\underset{\boldsymbol{p}}{argmin}\left\{\Delta(1-\boldsymbol{p})^2 + \frac{1}{2}\|\boldsymbol{p}-\boldsymbol{v}\|_2^2\right\}$$

Setting derivative w.r.t. $\boldsymbol{p}$ to zero:

$$2\Delta(1-\boldsymbol{p}) + (\boldsymbol{p}-\boldsymbol{v}) = 0$$

Minimum $\boldsymbol{p} = \frac{2\Delta+\boldsymbol{v}}{2\Delta+1}$.

(3) $t < 0$

The function is constantly zero, $L_{huber}(t) = (1-2t)$.

The minimization problem becomes: $\text{prox}_{\Delta q}[\boldsymbol{v}] = \underset{\boldsymbol{p}}{argmin}\left\{\Delta L_{huber}(\boldsymbol{p}) + \frac{1}{2}\|\boldsymbol{p}-\boldsymbol{v}\|_2^2\right\}$,

$$\underset{\boldsymbol{p}}{argmin}\left\{\Delta(1-2\boldsymbol{p}) + \frac{1}{2}\|\boldsymbol{p}-\boldsymbol{v}\|_2^2\right\}$$

Setting derivative w.r.t. $\boldsymbol{p}$ to zero:

$$-2\Delta + (\boldsymbol{p}-\boldsymbol{v}) = 0$$

Minimum $\boldsymbol{p} = \boldsymbol{v} - 2\Delta$.

By combining all cases and considering the scaling factor ($\frac{\lambda_w^t}{|\Omega_w^t|}$):

$$\text{prox}_{\Delta f_2}[\boldsymbol{v}] = \begin{cases} \boldsymbol{v} & \text{if } \boldsymbol{v} \geq 1 \\ \dfrac{2\Delta\frac{\lambda_w^t}{|\Omega_w^t|} + \boldsymbol{v}}{2\Delta\frac{\lambda_w^t}{|\Omega_w^t|} + 1} & \text{if } 0 \leq \boldsymbol{v} < 1 \\ \boldsymbol{v} - 2\Delta\dfrac{\lambda_w^t}{|\Omega_w^t|} & \text{if } \boldsymbol{v} < 0 \end{cases}$$

The proximal operator of Δf_3: The $f_3(\boldsymbol{\beta}) = \lambda_1^t \,\|\boldsymbol{\beta}\|_1 + \frac{\lambda_2^t}{2}\,\|\boldsymbol{\beta}\|_2^2$ can be obtained from the proximal operator of elastic net. According to Parikh and Boyd (2014), the proximal operator for $q(\boldsymbol{v}) = \|v\|_1 + \frac{\lambda}{2}\|\boldsymbol{v}\|_2^2$ can be written as follows:

$\mathrm{prox}_{\Delta q}[\boldsymbol{v}] = \frac{1}{1+\Delta\lambda} S_{\Delta\|.\|_1}(\boldsymbol{v}), where\ S_{\Delta\|.\|_1}$ is soft thresholding operator. Hence, we can write the proximal operator for f_3 as follows:

$$\mathrm{prox}_{\Delta f_3}[\boldsymbol{v}] = \frac{1}{1+\Delta\lambda_2^t} S_{\Delta\lambda_1^t}(\boldsymbol{v}), where\ S_{\Delta\lambda_1^t} = \begin{cases} \boldsymbol{v} - \Delta\lambda_1^t, & \boldsymbol{v} > \Delta\lambda_1^t \\ \mathbf{0}, & -\Delta\lambda_1^t \le \boldsymbol{v} \le \Delta\lambda_1^t \\ \boldsymbol{v} + \Delta\lambda_1^t, & \boldsymbol{v} \le \Delta\lambda_1^t \end{cases}$$

B.2 *The $\boldsymbol{\theta_s}$ Update*

Proposition 5. *Let $f_1(\boldsymbol{\theta_s}) = \frac{\lambda_l^s}{2N_l^s} \| \boldsymbol{Y^s} - \boldsymbol{H_l^s}\boldsymbol{\theta_s} \|^2$, $f_2(\boldsymbol{\theta_s}) = \frac{\lambda_w^s}{|\Omega_w^s|} l_{huber}(1 - \boldsymbol{H_w^s}\boldsymbol{\theta_s})$, $f_3(\boldsymbol{\theta_s}) = \frac{\lambda_2^s}{2}\,\|\boldsymbol{\theta_s}\|_2^2$. The proximal operators of $\Delta f_1, \dots \Delta f_3$ are given as follows: Let $\boldsymbol{v}$ have the same size as $\boldsymbol{\theta_s}$.*

1) *For* $\mathrm{prox}_{\Delta f_1}[\boldsymbol{v}]$, *we have*

   $\mathrm{prox}_{\Delta f_1}[\boldsymbol{v}] = \left(\mathbf{I} + \frac{\Delta}{N_l^s}(\boldsymbol{H_l^s})^\top \boldsymbol{H_l^s}\right)^{\mathbf{-1}} \left(\boldsymbol{v} + \frac{\Delta}{N_l^s}(\boldsymbol{H_l^s})^\top \boldsymbol{Y^s}\right)$, *where* $\mathbf{I}$ *is the identity matrix of order* d_s.

2) *For* $\mathrm{prox}_{\Delta f_2}[\boldsymbol{v}]$, *we have*

$$\mathrm{prox}_{\Delta f_2}[\boldsymbol{v}] = \begin{cases} \boldsymbol{v} & \text{if } \boldsymbol{v} \ge 1 \\ \dfrac{2\Delta\frac{\lambda_w^s}{|\Omega_w^s|} + \boldsymbol{v}}{2\Delta\frac{\lambda_w^s}{|\Omega_w^s|} + 1} & \text{if } 0 \le \boldsymbol{v} < 1 \\ \boldsymbol{v} - 2\Delta\dfrac{\lambda_w^s}{|\Omega_w^s|} & \text{if } \boldsymbol{v} < 0 \end{cases}$$

3) *For* $\text{prox}_{\Delta f_3}[\boldsymbol{v}]$, *we have*

$$\text{prox}_{\Delta f_3}[\boldsymbol{v}] = \frac{\boldsymbol{v}}{1+\Delta\lambda_2^s}.$$

Proof

The proximal operator of Δf_1: The $f_1(\boldsymbol{\theta_s}) = \frac{\lambda_l^s}{2N_l^s} \parallel \boldsymbol{Y^s} - \boldsymbol{H_l^s}\boldsymbol{\theta_s} \parallel^2$ is on least square form. According to Parikh and Boyd (2014), the proximal operator for $q(\boldsymbol{v}) = \|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{v}\|_2^2$ can be written as follows:

$$\text{prox}_{\Delta q}[\boldsymbol{v}] = \underset{\boldsymbol{p}}{argmin}\left\{\Delta q(\boldsymbol{p}) + \frac{1}{2}\|\boldsymbol{p}-\boldsymbol{v}\|_2^2\right\} = (\boldsymbol{I} + \Delta\boldsymbol{A}^\top\boldsymbol{A})^{-1}(\boldsymbol{v} + \Delta\boldsymbol{A}^\top\boldsymbol{b})$$

In our setting, $\mathbf{A} = \boldsymbol{H_l^s}, \mathbf{c} = \boldsymbol{Y^s}$, hence we have,

$$\text{prox}_{\Delta f_1}[\boldsymbol{v}] = \left(\mathbf{I} + \frac{\Delta}{N_l^s}(\boldsymbol{H_l^s})^\top\boldsymbol{H_l^s}\right)^{\mathbf{-1}}\left(\boldsymbol{v} + \frac{\Delta}{N_l^s}(\boldsymbol{H_l^s})^\top\boldsymbol{Y^s}\right)$$

The proximal operator of Δf_2: The $f_2(\boldsymbol{\theta_s}) = \frac{\lambda_w^s}{|\Omega_w^s|} l_{huber}(1 - \boldsymbol{H_w^s}\boldsymbol{\theta_s})$, proximal can be found after finding the proximal operator of Huber loss which has been derived in Appendix B.1. Similarly, the proximal for the function can be found as follows:

$$\text{prox}_{\Delta f_2}[\boldsymbol{v}] = \begin{cases} \boldsymbol{v} & \text{if } \boldsymbol{v} \geq 1 \\ \dfrac{2\Delta\dfrac{\lambda_w^s}{|\Omega_w^s|} + \boldsymbol{v}}{2\Delta\dfrac{\lambda_w^s}{|\Omega_w^s|} + 1} & \text{if } 0 \leq \boldsymbol{v} < 1 \\ \boldsymbol{v} - 2\Delta\dfrac{\lambda_w^s}{|\Omega_w^s|} & \text{if } \boldsymbol{v} < 0 \end{cases}$$

The proximal operator of Δf_3: This can be obtained from the proximal operator of L_2 penalty. According to Parikh and Boyd (2014), the proximal operator for $q(\boldsymbol{v}) = \frac{\lambda}{2}\|\boldsymbol{v}\|_2^2$ can be written as follows:

$\text{prox}_{\Delta q}[\boldsymbol{v}] = \frac{\boldsymbol{v}}{1+\Delta\lambda}$. Hence, we can write the proximal operator for f_3 as follows:

$$\text{prox}_{\Delta f_3}[\boldsymbol{v}] = \frac{\boldsymbol{v}}{1+\Delta\lambda_2^s}.$$

B.3 *The β Update*

Proposition 6. *Let* $\mathcal{D}_{target} = \{(x_i^t, y_i^t)\}_{i=1}^{N_l^t}$ *be the labeled dataset for the target task, and* $\mathcal{D}_{source} = \left\{\left(x_j^s, y_j^s\right)\right\}_{j=1}^{N_l^s}$ *be the labeled dataset for the source task be the dataset for the* ***source task****. Assume that the source task* $g_{\theta_s}(x)$ *generalizes with a risk bound of* $\mathcal{O}\left(\frac{1}{(N_l^s)^{\gamma_s}}\right)$*. Assume that* $\ell(f(x), y)$ *is a* L'*-Lipschitz continuous loss function with respect to its parameters.*

Then, the excess risk for the target task is bounded by:

$$\mathbb{E}[\ell(f_\beta(h(x)), y^t) - \ell(f^*(h(x)), y^t)] = \mathcal{O}\left(\frac{d\gamma_s\rho\log(RL'N_l^t) + \log\left(\frac{1}{p}\right)}{N_l^t} + \frac{L'}{(N_l^t)^{\gamma_s\rho}}\right)$$

The bound holds with probability at least $(1-p)$*, provided that* $N_l^s = \Omega((N_l^t)^{2\rho})$.

where γ_s *is the learning rate of the source task,* d *is the number of parameters in* f_β*, and* ρ *is the relatedness factor between the source and target tasks.*

<u>Proof:</u>

In our setup, both the target and source tasks use a ranking loss as a regularizer to improve generalization by preserving relational structures within the data to respect the relative ordering between pairs of data points. While the ranking component may contribute to a tighter generalization bound, we do not explicitly include it in the theoretical proof presented here. Instead, we focus on the generalization bound derived from the weak supervision framework, treating the ranking loss as an additional regularization term that aids generalization in practice

The proof is derived by following the results and logic from (Robinson et al., 2020), specifically leveraging (Propositions 6, 8, 9, and Theorem 10), and applying them to our problem setting with continuous weak labels and feature augmentation to refer to transfer between the source and target patient.

Step 1: Excess Risk Decomposition

We begin by decomposing the excess risk of the target task into two parts:

$$\mathbb{E}[\ell(f_\beta(h(x)), y^t) - \ell(f^*(h(x)), y^t)] \leq 2L \cdot \mathrm{Rate}_{N_l^s}(\mathcal{G}, P_X, W) + \mathrm{Rate}_{N_l^t}(\mathcal{F}, \hat{P}),$$

where:

- $\mathrm{Rate}_{N_l^s}(\mathcal{G}, P_X, W)$ is the risk contribution from the source task with features $g_{\theta_s}(x)$,
- $\mathrm{Rate}_{N_l^t}(\mathcal{F}, \hat{P})$ is the risk from the target task with augmented features $h(x)$

This decomposition shows that the total excess risk is split into the source task and target task contributions. We will now control these components.

Step 2: Weak Central Condition

Since the source labels in our case are continuous, we apply the weak central condition which provides that, with high probability, the source task satisfies:

$(\ell, \hat{P}, \mathcal{F})$ satisfies the $\mathcal{O}\left(\frac{1}{(N_l^t)^{\gamma_s \rho}}\right)$-weak central condition with probability $1 - p$,

where:

- γ_s is the learning rate of the source task,
- ρ is the relatedness factor between the source and target tasks.

This shows that the source task provides useful information to the target task. The term $\mathcal{O}\left(\frac{1}{(N_l^t)^{\gamma_s \rho}}\right)$- represents the generalization error of the source task, which decreases as the number of source task samples increases. Thus, the source task's contribution to reducing the excess risk in the target task becomes significant as more source data is available.

Step 3: Generalization Bound for the Target Task

Now, using the weak central condition, according to (Robinson et al., 2020), we can obtain the following generalization bound for the target task:

$$\mathbb{E}[\ell(f_\beta(h(x)), y^t) - \ell(f^*(h(x)), y^t)] = \mathcal{O}\left(V \frac{d \log\left(\frac{RL'}{r}\right) + \log\left(\frac{1}{p}\right)}{N_l^t} + Vr\right)$$

with a probability of at least $(1 - p)$, *where* $r = \mathcal{O}\left(\frac{1}{N_l^s}\right)$ *,* $V = B + r$ *,* B *is a constant that bounds the maximum value of the loss function,* R *is the radius of the Euclidean ball in which the parameter* $\boldsymbol{\beta}$ *is constrained. and* d *is the number of parameters in* f_β*.*

Generalization Bound for with Source Supervision:

Now, we specialize this general bound to the specific case of empirical risk minimization (ERM), following the fact that the source task has a learning rate $\mathrm{Rate}_{N_l^s}(\mathcal{G}, P_X, W) = \mathcal{O}\left(\frac{1}{(N_l^s)^{\gamma_s}}\right)$, and applies this to the general bound. In the case of continuous weak labels, the number of source task samples must satisfy:

$$N_l^s = \Omega((N_l^t)^{2\rho})$$

This condition, gives the following generalization bound for the ERM algorithm:

$$\mathbb{E}[\ell(f_\beta(h(x)), y^t) - \ell(f^*(h(x)), y^t)] = \mathcal{O}\left(\frac{d\gamma_s\rho \log(RL'N_l^t) + \log\left(\frac{1}{p}\right)}{N_l^t} + \frac{L'}{(N_l^t)^{\gamma_s\rho}}\right)$$

The bound holds with probability at least $(1 - p)$.

REFERENCES

Abujrida, H., Agu, E., & Pahlavan, K. (2017, November). Smartphone-based gait assessment to infer Parkinson's disease severity using crowdsourced data. In *2017 IEEE Healthcare Innovations and Point of Care Technologies (HI-POCT)* (pp. 208-211). IEEE.

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019, July). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 2623-2631).

Alaa, A. M., & Van Der Schaar, M. (2017). Bayesian inference of individualized treatment effects using multi-task gaussian processes. *Advances in neural information processing systems*, 30.

Alenezi, D. F., Biehler, M., Shi, J., & Li, J. (2024). Physics-Informed Weakly-Supervised Learning for Quality Prediction of Manufacturing Processes. *IEEE Transactions on Automation Science and Engineering*.

Alenezi, D. F., Shi, J., & Li, J. (XXX). Multi-source Multi-task Weakly Supervised Transfer Learning. [In preparation XXX]

Arora, S., Venkataraman, V., Donohue, S., Biglan, K. M., Dorsey, E. R., & Little, M. A. (2014, May). High accuracy discrimination of Parkinson's disease participants from healthy controls using smartphones. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 3641-3644). IEEE.

Arora, S., Venkataraman, V., Zhan, A., Donohue, S., Biglan, K. M., Dorsey, E. R., & Little, M. A. (2015). Detecting and monitoring the symptoms of Parkinson's disease using smartphones: A pilot study. *Parkinsonism & related disorders*, *21*(6), 650-653.

Baraniuk, R., Davenport, M., DeVore, R., & Wakin, M. (2008). A simple proof of the restricted isometry property for random matrices. *Constructive approximation*, 28, 253-263.

Bastos, L. S., & O'hagan, A. (2009). Diagnostics for Gaussian process emulators. *Technometrics*, 51(4), 425-438.

Bayarri, M. J., Berger, J. O., Paulo, R., Sacks, J., Cafeo, J. A., Cavendish, J., Lin, C.-H., & Tu, J. (2007). A framework for validation of computer models. *Technometrics*, 49(2), 138-154.

Bloem, B. R., Okun, M. S., & Klein, C. (2021). Parkinson's disease. *The Lancet*, 397(10291), 2284-2303.

Blum, A., & Chawla, S. (2001). Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the Eighteenth International Conference on Machine Learning*, 19–26.

Bot, B. M., Suver, C., Neto, E. C., Kellen, M., Klein, A., Bare, C., ... & Trister, A. D. (2016). The mPower study, Parkinson disease mobile data collected using ResearchKit. *Scientific data*, *3*(1), 1-9.

Bruno, R. (2022, January 4). Tuffaloy Resistance Welding Products. Retrieved March 22, 2023, from https://www.tuffaloy.com/.

Burges, C., Ragno, R., & Le, Q. (2006). Learning to rank with nonsmooth cost functions. *Advances in neural information processing systems*, *19*, 193-200.

Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., & Hullender, G. (2005, August). Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, 89-96.

Cang, R., Li, H., Yao, H., Jiao, Y., & Ren, Y. (2018). Improving direct physical properties prediction of heterogeneous materials from imaging data via convolutional neural network and a morphology-aware generative model. *Computational Materials Science*, 150, 212-221.

Cao, Z., Qin, T., Liu, T. Y., Tsai, M. F., & Li, H. (2007, June). Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, 129-136.

Chaibub Neto, E. L. I. A. S., Bot, B. M., Perumal, T., Omberg, L., Guinney, J., Kellen, M., ... & Trister, A. D. (2016). Personalized hypothesis tests for detecting medication response in Parkinson disease patients using iPhone Sensor data. In *Biocomputing 2016: Proceedings of the Pacific Symposium*, 273-284.

Chakravarti, P., Kuusela, M., Lei, J., & Wasserman, L. (2021). Model-independent detection of new physics signals using interpretable semi-supervised classifier tests. arXiv preprint arXiv:2102.07679.

Chandra, R., & Kapoor, A. (2020). Bayesian multi-source transfer learning. *Neurocomputing*.

Chapelle, O., & Keerthi, S. S. (2010). Efficient algorithms for ranking with SVMs. *Information retrieval*, *13*(3), 201-215.

Chapelle, O., & Zien, A. (2005, January). Semi-supervised classification by low density separation. In *AISTATS* (Vol. 2005, 57-64).

Chapelle, O., Schölkopf, B., & Zien, A. (2006). A discussion of semi-supervised learning and transduction. In *Semi-supervised learning* (pp. 473-478). MIT Press.

Chattopadhyay, R., Sun, Q., Fan, W., Davidson, I., Panchanathan, S., & Ye, J. (2012). Multisource domain adaptation and its application to early detection of fatigue. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 6(4), 1-26.

Chen, Y., & Wang, J. Z. (2004). Image categorization by learning and reasoning with regions. *The Journal of Machine Learning Research*, 5, 913-939.

Chen, Z., Gao, J., Wang, W., & Yan, Z. (2021). Physics-informed generative neural network: an application to troposphere temperature prediction. *Environmental Research Letters*, 16(6), 065003.

Cheng, B., Liu, M., Zhang, D., & Shen, D. (2018). Robust multi-label transfer feature learning for early diagnosis of Alzheimer's disease. Brain Imaging and Behavior.

Cheng, Y., & Fu, L.-Y. (2022). Nonlinear seismic inversion by physics-informed Caianiello convolutional neural networks for overpressure prediction of source rocks in the offshore Xihu depression, East China. Journal of Petroleum Science and Engineering, 215, 110654.

Cheplygina, V., de Bruijne, M., & Pluim, J. (2018). Not-so-supervised: A survey of semi-supervised, multi-instance, and transfer learning in medical image analysis. Medical Image Analysis, 54, 280–296.

Crammer, K., & Singer, Y. (2001, December). Pranking with ranking. In *Nips* (Vol. 1, 641-647).

Cranmer, M., Sanchez Gonzalez, A., Battaglia, P., Xu, R., Cranmer, K., Spergel, D., & Ho, S. (2020). Discovering symbolic models from deep learning with inductive biases. Advances in Neural Information Processing Systems, 33, 17429-17442.

Dai, W., Yang, Q., Xue, G. R., & Yu, Y. (2007, June). Boosting for transfer learning. In Proceedings of the 24th international conference on Machine learning (pp. 193-200).

Dansky, K. H., Vasey, J., & Bowles, K. (2008). Impact of telehealth on clinical outcomes in patients with heart failure. *Clinical Nursing Research*, *17*(3), 182-199.

Daw, A., Karpatne, A., Watkins, W. D., Read, J. S., & Kumar, V. (2022). Physics-guided neural networks (pgnn): An application in lake temperature modeling. In Knowledge Guided Machine Learning (pp. 353-372). Chapman and Hall/CRC.

Daw, A., Thomas, R. Q., Carey, C. C., Read, J. S., Appling, A. P., & Karpatne, A. (2020). Physics-guided architecture (pga) of neural networks for quantifying uncertainty in lake temperature modeling. Proceedings of the 2020 siam international conference on data mining.

De Bois, M., El-Yacoubi, M., & Ammi, M. (2020). Adversarial multi-source transfer learning in healthcare: Application to glucose prediction for diabetic people. Computer Methods and Programs in Biomedicine.

Dennis Jr, J. E., & Schnabel, R. B. (1996). *Numerical methods for unconstrained optimization and nonlinear equations*. Society for Industrial and Applied Mathematics.

Dietterich, T. G., Lathrop, R. H., & Lozano-Pérez, T. (1997). Solving the multiple instance problem with axis-parallel rectangles. Artificial intelligence, 89(1-2), 31-71.

Dinov, I., Heavner, B., Tang, M., et al. (2016). Predictive big data analytics: A study of Parkinson’s disease using large, complex, heterogeneous, incongruent, multi-source and incomplete observations. PLoS ONE.

Dong, J., Cong, Y., Sun, G., & Ding, Z. (2020). Weakly-supervised cross-domain adaptation for endoscopic lesions segmentation. IEEE Transactions on Circuits and Systems for Video Technology.

Donoho, D. L. (2006). Compressed sensing. IEEE Transactions on information theory, 52(4), 1289-1306.

Dorsey, E., Sherer, T., Okun, M. S., & Bloem, B. R. (2018). The emerging evidence of the Parkinson pandemic. Journal of Parkinson's Disease, 8(s1), S3-S8.

Erichson, N. B., Muehlebach, M., & Mahoney, M. W. (2019). Physics-informed autoencoders for Lyapunov-stable fluid flow prediction. arXiv preprint arXiv:1905.10866.

F. Alenezi, D., Shi, H., & Li, J. (2022). A Ranking-based Weakly Supervised Learning model for telemonitoring of Parkinson’s disease. IISE Transactions on Healthcare Systems Engineering, 12(4), 322-336.

Fang, W., Chen, C., Song, B., Wang, L., Zhou, J., & Zhu, K. Q. (2019, December). Adapted tree boosting for transfer learning. In 2019 IEEE International Conference on Big Data (Big Data) (pp. 741-750). IEEE.

Far, M. S., Eickhoff, S. B., Goni, M., & Dukart, J. (2021). Exploring Test-Retest Reliability and Longitudinal Stability of Digital Biomarkers for Parkinson Disease in the m-Power Data Set: Cohort Study. Journal of Medical Internet Research, 23(9), e26608.

Farimani, A. B., Gomes, J., & Pande, V. S. (2017). Deep learning the physics of transport phenomena. arXiv preprint arXiv:1709.02432.

Foulds, J., & Frank, E. (2010). A review of multi-instance learning assumptions. The knowledge engineering review, 25(1), 1-25.

Frénay, B., & Verleysen, M. (2013). Classification in the presence of label noise: a survey. IEEE transactions on neural networks and learning systems, 25(5), 845-869.

Fujino, A., Ueda, N., & Saito, K. (2005, July). A hybrid generative/discriminative approach to semi-supervised classifier design. In *AAAI*, 764-769.

Gao, J., Ge, L., Li, K., & Ngo, H. (2014). On handling negative transfer and imbalanced distributions in multiple source transfer learning. Statistical Analysis and Data Mining: The ASA Data Science Journal.

Gaw, N., Li, J., & Yoon, H. (2022). A novel semi-supervised learning model for smartphone-based health telemonitoring. IEEE Transactions on Automation Science and Engineering, 21(1), 428-441.

Goetz, C. G., Stebbins, G. T., Wolff, D., DeLeeuw, W., Bronte-Stewart, H., Elble, R., ... & Taylor, C. B. (2009). Testing objective measures of motor impairment in early Parkinson's disease: Feasibility study of an at-home testing device. *Movement Disorders*, *24*(4), 551-556.

Goetz, C. G., Tilley, B. C., Shaftman, S. R., Stebbins, G. T., Fahn, S., Martinez-Martin, P., ... & LaPelle, N. (2008). Movement Disorder Society-sponsored revision of the Unified Parkinson's Disease Rating Scale (MDS-UPDRS): scale presentation and clinimetric testing results. Movement disorders: official journal of the Movement Disorder Society, 23(15), 2129-2170.

Goldfarb, D. (1976). Factorized variable metric methods for unconstrained optimization. Mathematics of Computation, 30(136), 796-811.

Goncalves, A., Ray, P., Soper, B., Stevens, J., Coyle, L., & Sales, A. P. (2020). Generation and evaluation of synthetic patient data. BMC medical research methodology, 20, 1-40.

Greydanus, S., Dzamba, M., & Yosinski, J. (2019). Hamiltonian neural networks. Advances in Neural Information Processing Systems, 32.

Gu, T., & Duan, R. (2022). SynTL: A synthetic-data-based transfer learning approach. eMERGE Network.

Guo, S., Agarwal, M., Cooper, C., Tian, Q., Gao, R. X., Grace, W. G., & Guo, Y. (2022). Machine learning for metal additive manufacturing: Towards a physics-informed data-driven paradigm. Journal of Manufacturing Systems, 62, 145-163.

Herbrich, R., Graepel, T., & Obermayer, K. (2000). Large margin rank boundaries for ordinal regression. In B. Smola & S. Schoelkopf (Eds.), Advances in large margin classifiers. Cambridge, MA: MIT Press.

Herrero-Garcia, J., Patrick, R., & Scaffidi, A. (2022). A semi-supervised approach to dark matter searches in direct detection data with machine learning. Journal of Cosmology and Astroparticle Physics, 2022(02), 039.

Higdon, D., Kennedy, M., Cavendish, J. C., Cafeo, J. A., & Ryne, R. D. (2004). Combining field data and computer simulations for calibration and prediction. SIAM Journal on Scientific Computing, 26(2), 448-466.

Hillar, C. J., & Sommer, F. T. (2015). When can dictionary learning uniquely recover sparse data from subsamples? IEEE Transactions on information theory, 61(11), 6290-6297.

Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.

International Organization for Standardization. (2006). ISO 10447:2006 resistance welding peel and chisel testing of resistance spot and projection welds.

Jankovic, J. (2008). Parkinson's disease: clinical features and diagnosis. Journal of neurology, neurosurgery & psychiatry, 79(4), 368-376.

Jia, X., Willard, J., Karpatne, A., Read, J., Zwart, J., Steinbach, M., & Kumar, V. (2019). Physics guided RNNs for modeling dynamical systems: A case study in simulating lake temperature profiles. Proceedings of the 2019 SIAM international conference on data mining,

Jiang, H., Lim, W. Y. B., Ng, J. S., Wang, Y., & Miao, C. (2021). Towards Parkinson's disease prognosis using self-supervised learning and anomaly detection. ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 3960-3964.

Joachims, T. (1999, June). Transductive inference for text classification using support vector machines. In Icml (Vol. 99, 200-209).

Joachims, T. (2002, July). Optimizing search engines using clickthrough data. In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 133-142).

Kapusuzoglu, B., & Mahadevan, S. (2020). Physics-informed and hybrid machine learning in additive manufacturing: application to fused filament fabrication. Jom, 72(12), 4695-4705.

Karaman, O., Çakın, H., Alhudhaif, A., & Polat, K. (2021). Robust automated Parkinson disease detection based on voice signals with transfer learning. Expert Systems with Applications.

Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., & Yang, L. (2021). Physics-informed machine learning. Nature Reviews Physics, 3(6), 422-440.

Karpatne, A., Atluri, G., Faghmous, J. H., Steinbach, M., Banerjee, A., Ganguly, A., Shekhar, S., Samatova, N., & Kumar, V. (2017). Theory-guided data science: A new paradigm for scientific discovery from data. IEEE Transactions on knowledge and data engineering, 29(10), 2318-2331.

Kassavetis, P., Saifee, T. A., Roussos, G., Drougkas, L., Kojovic, M., Rothwell, J. C., ... & Bhatia, K. P. (2016). Developing a tool for remote digital assessment of Parkinson's disease. *Movement disorders clinical practice*, *3*(1), 59-64.

Kennedy, M. C., & O'Hagan, A. (2001). Bayesian calibration of computer models. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 63(3), 425-464.

Kondor, R., Howard, A., & Jebara, T. (2007, March). Multi-object tracking with representations of the symmetric group. In *Artificial Intelligence and Statistics*, 211-218. PMLR.

Kovalenko, E., Shcherbak, A., Somov, A., & Bril, E. (2022). Detecting Parkinson's disease through simultaneous analysis of data from wearable sensors and video. IEEE Sensors Journal.

Kristianto, D. (2021, April). Winning the Attention War: Consumers in nine major markets now spend more than four hours a day in Apps. *App Annie.*

Laine, S., & Aila, T. (2016). Temporal ensembling for semi-supervised learning. arXiv preprint arXiv:1610.02242.

Lenain, R., Weston, J., Shivkumar, A., & Fristed, E. (2020). Surfboard: Audio feature extraction for modern machine learning. Interspeech 2020. doi:10.21437/interspeech.2020-2879.

Leonesio, M., & Fagiano, L. (2022). A semi-supervised physics-informed classifier for centerless grinding operations. 2022 IEEE Conference on Control Technology and Applications (CCTA).

Li, P., Wu, Q., & Burges, C. (2007). Mcrank: Learning to rank using multiple classification and gradient boosting. *Advances in neural information processing systems*, *20*, 897-904.

Li, Y. F., Tsang, I. W., Kwok, J. T., & Zhou, Z. H. (2013). Convex and scalable weakly labeled SVMs. *Journal of Machine Learning Research*, *14*(7).

Liu TY. (2011) Applications of Learning to Rank. In: Learning to Rank for Information Retrieval. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-14267-3_14.

Lutter, M., Ritter, C., & Peters, J. (2019). Deep lagrangian networks: Using physics as model prior for deep learning. arXiv preprint arXiv:1907.04490.

Maetzler, W., Domingos, J., Srulijes, K., Ferreira, J. J., & Bloem, B. R. (2013). Quantitative wearable sensors for objective assessment of Parkinson's disease. *Movement Disorders*, *28*(12), 1628-1637.

Matarazzo, M., Arroyo-Gallego, T., Montero, P., Puertas-Martín, V., Butterworth, I., Mendoza, C. S., ... & Sánchez-Ferro, Á. (2019). Remote monitoring of treatment response in Parkinson's disease: the habit of typing on a computer. *Movement Disorders*, *34*(10), 1488-1495.

Meng, C., Seo, S., Cao, D., Griesemer, S., & Liu, Y. (2022). When physics meets machine learning: A survey of physics-informed machine learning. arXiv preprint arXiv:2203.16797.

Michael J. Fox Foundation. (2019). Parkinson’s Disease Economic Burden on Patients, Families and the Federal Government Is $52 Billion, Doubling Previous Estimates.

Montes, K. J., Rea, C., Tinguely, R. A., Sweeney, R., Zhu, J., & Granetz, R. (2021). A semi-supervised machine learning detector for physics events in tokamak discharges. Nuclear Fusion, 61(2), 026022.

Muralidhar, N., Bu, J., Cao, Z., He, L., Ramakrishnan, N., Tafti, D., & Karpatne, A. (2020). Phynet: Physics guided neural networks for particle drag force prediction in assembly. Proceedings of the 2020 SIAM International Conference on Data Mining.

Nabian, M. A., & Meidani, H. (2020). Physics-driven regularization of deep neural networks for enhanced engineering design and analysis. Journal of Computing and Information Science in Engineering, 20(1), 011006.

Nigam, K., McCallum, A. K., Thrun, S., & Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. *Machine learning*, *39*(2), 103-134.

Oliver, D. S., & Alfonzo, M. (2018). Calibration of imperfect models to biased observations. Computational Geosciences, 22(1), 145-161.

Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, 22(10), 1345-1359.

Parikh, N., & Boyd, S. (2014). Proximal algorithms. Foundations and trends® in Optimization, 1(3), 127-239.

Pastorino, M., Cancela, J., Arredondo, M. T., Pansera, M., Pastor-Sanz, L., Villagra, F., ... & Martin, J. A. (2011, August). Assessment of bradykinesia in Parkinson's disease patients through a multi-parametric system. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (pp. 1810-1813). IEEE.

Patel, S., Park, H., Bonato, P., Chan, L., & Rodgers, M. (2012). A review of wearable sensors and systems with application in rehabilitation. Journal of Neuroengineering and Rehabilitation, 9, 1-17.

Prince, J., Andreotti, F., & de Vos, M. (2019). Multi-source ensemble learning for the remote prediction of Parkinson's disease. IEEE Transactions on Biomedical Engineering.

Prince, J., Arora, S., & de Vos, M. (2018). Big data in Parkinson's disease: using smartphones to remotely detect longitudinal disease phenotypes. *Physiological measurement*, *39*(4), 044005.

Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational physics, 378, 686-707.

Ratner, A. J., Hancock, B., Dunnmon, J. A., Sala, F., Pandey, S., & Ré, C. (2018). Training complex models with multi-task weak supervision. Proceedings of the AAAI Conference on Artificial Intelligence, 33, 4763-4771.

Read, J. S., Jia, X., Willard, J., Appling, A. P., Zwart, J. A., Oliver, S. K., Karpatne, A., Hansen, G. J., Hanson, P. C., & Watkins, W. (2019). Process-guided deep learning predictions of lake water temperature. Water Resources Research, 55(11), 9173-9190.

Richard, E., & Kelly, M. (2023). Detecting Parkinson's disease using machine learning from movement. Canadian Journal for the Academic Mind.

Rigas, G., Tzallas, A. T., Tsipouras, M. G., Bougia, P., Tripoliti, E. E., Baga, D., ... & Konitsiotis, S. (2012). Assessment of tremor activity in the Parkinson's disease using a set of wearable sensors. *IEEE Transactions on Information Technology in Biomedicine*, *16*(3), 478-487.

Rizek, P., Kumar, N., & Jog, M. S. (2016). An update on the diagnosis and treatment of Parkinson disease. *Cmaj*, *188*(16), 1157-1165

Roalf, D. R., Rupert, P., Mechanic-Hamilton, D., Brennan, L., Duda, J. E., Weintraub, D., ... & Moberg, P. J. (2018). Quantitative assessment of finger tapping characteristics in mild cognitive impairment, Alzheimer's disease, and Parkinson's disease. *Journal of neurology*, *265*(6), 1365-1375.

Robinson, J., Jegelka, S., & Sra, S. (2020, November). Strength from weakness: Fast learning using weak supervision. In *International Conference on Machine Learning* (pp. 8127-8136). PMLR.

Sacks, W., & Welch, T. (1989). Mitchell, and HP Wynn,“Design and analysis of computer experiments,”. Statistical Science, 4(4), 409-453.

Sajal, M. S. R., Ehsan, M. T., Vaidyanathan, R., Wang, S., Aziz, T., & Mamun, K. (2020). Telemonitoring Parkinson’s disease using machine learning by combining tremor and voice analysis. Brain Informatics.

Samii, A., Nutt, J. G., & Ransom, B. R. (2004). Parkinson's disease. *Lancet (London, England)*, *363*(9423), 1783–1793.

Sanderson, M., Christopher, D., & Manning, H. (2010). Introduction to information retrieval. *Natural Language Engineering*, *16*(1), 100.

Settles, B. (2009). Active learning literature survey. Technical Report 1648. Department of Computer Sciences, University of Wisconsin at Madison.

Shin, C., Li, W., Vishwakarma, H., Roberts, N. C., & Sala, F. (2022, January). Universalizing Weak Supervision. In *The Tenth International Conference on Learning Representations*.

Shi, Q., Gu, D., Xia, M., Cao, S., & Rong, T. (2016). Effects of laser processing parameters on thermal behavior and melting/solidification mechanism during selective laser melting of TiC/Inconel 718 composites. Optics & Laser Technology, 84, 9-22.

Sica, M., Tedesco, S., Crowe, C., Kenny, L., Moore, K., Timmons, S., ... & Komaris, D. S. (2021). Continuous home monitoring of Parkinson's disease using inertial sensors: A systematic review. *PloS one*, *16*(2), e0246528.

Sun Dansky, K. H., Vasey, J., & Bowles, K. (2008). Impact of telehealth on clinical outcomes in patients with heart failure. Clinical Nursing Research, 17(3), 182-199.

Sun, B., Feng, J., & Saenko, K. (2016, March). Return of frustratingly easy domain adaptation. In Proceedings of the AAAI conference on artificial intelligence (Vol. 30, No. 1).

Sun, D., Wang, N., Xu, J., & Gao, H. (2021, October). Improved transfer learning based on EMD for Parkinson’s diagnosis. In International Symposium on Artificial Intelligence and Robotics 2021 (Vol. 11884, pp. 278-288). SPIE.

Sveinbjornsdottir, S. (2016). The clinical symptoms of Parkinson's disease. *Journal of neurochemistry, 139*, 318-324.

Taravari, A., Medziti, F., Grunevska, B., Adili, F., Ademi, B., Miftari, V., & Haliti, G. (2014). Correlation of Age and Severity of Clinical Manifestation Assessed by UPDRS in Patients with Idiopathic Parkinson's Disease. *Medical Archives*, *68*(1), 44.

Tarvainen, A., & Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. Advances in Neural Information Processing Systems, 30.

Taylor Tavares, A. L., Jefferis, G. S., Koop, M., Hill, B. C., Hastie, T., Heit, G., & Bronte-Stewart, H. M. (2005). Quantitative measurements of alternating finger tapping in Parkinson's disease correlate with UPDRS motor disability and reveal the improvement in fine motor control from medication and deep brain stimulation. *Movement disorders: official journal of the Movement Disorder Society*, *20*(10), 1286-1298.

Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, *58*(1), 267–288. http://www.jstor.org/stable/2346178

Tommasi, T., Orabona, F., & Caputo, B. (2010, June). Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (pp. 3081-3088). IEEE.

Tsanas, A. (2012). Accurate telemonitoring of Parkinson’s disease symptom severity using nonlinear speech signal processing and statistical machine learning (Doctoral dissertation, Oxford University, UK).

Tsanas, A., Little, M. A., McSharry, P. E., & Ramig, L. O. (2010). Accurate telemonitoring of Parkinson’s disease progression by noninvasive speech tests. IEEE Transactions on Biomedical Engineering, 57(4), 884–893.

Tsanas, A., Little, M. A., McSharry, P. E., & Ramig, L. O. (2011). Nonlinear speech analysis algorithms mapped to a standard metric achieve clinically useful quantification of average Parkinson’s disease symptom severity. Journal of the Royal Society Interface, 8(59), 842–855.

Tsanas, A., Little, M., McSharry, P., & Ramig, L. (2009). Accurate telemonitoring of Parkinson’s disease progression by non-invasive speech tests. *Nature Precedings*, 1-1.

Verma, V., Kawaguchi, K., Lamb, A., Kannala, J., Solin, A., Bengio, Y., & Lopez-Paz, D. (2022). Interpolation consistency training for semi-supervised learning. Neural Networks, 145, 90-106.

Wang, H., Li, B., & Xuan, F.-Z. (2022). A dimensionally augmented and physics-informed machine learning for quality prediction of additively manufactured high-entropy alloy. Journal of Materials Processing Technology, 307, 117637.

Wang, M., Ge, W., Apthorp, D., & Suominen, H. (2020). Robust Feature Engineering for Parkinson Disease Diagnosis: New Machine Learning Techniques. *JMIR Biomedical Engineering*, *5*(1), e13611.

World Health Organization. (2011). mHealth: new horizons for health through mobile technologies. mHealth: new horizons for health through mobile technologies.

Yang, W., Hamilton, J. L., Kopil, C., Beck, J. C., Tanner, C. M., Albin, R. L., ... & Thompson, T. (2020). Current and projected future economic burden of Parkinson’s disease in the US. npj Parkinson's Disease, 6(1), 15.

Yoneyama, M., Kurihara, Y., Watanabe, K., & Mitoma, H. (2013). Accelerometry-based gait analysis and its application to Parkinson's disease assessment—part 1: detection of stride event. *IEEE Transactions on neural systems and rehabilitation engineering*, *22*(3), 613-622.

Yoon, H., & Gaw, N. (2021). A novel multi-task linear mixed model for smartphone-based telemonitoring. Expert Systems with Applications.

Yoon, H., & Li, J. (2018). A novel positive transfer learning approach for telemonitoring of Parkinson's disease. *IEEE Transactions on Automation Science and Engineering*, *16*(1), 180-191.

Yoon, H., & Li, J. (2019). A novel positive transfer learning approach for telemonitoring of Parkinson’s disease. IEEE Transactions on Automation Science and Engineering.

Zahoor, I., Shafi, A., & Haq, E. (2018). Pharmacological treatment of Parkinson’s disease. *Exon Publications*, 129-144.

Zhan, A., Little, M. A., Harris, D. A., Abiola, S. O., Dorsey, E., Saria, S., & Terzis, A. (2016). High frequency remote monitoring of Parkinson's disease via smartphone: Platform overview and medication response detection. arXiv preprint arXiv:1601.00960.

Zhang, A., Cebulla, A., Panev, S., Hodgins, J., & De la Torre, F. (2017). Weakly-supervised learning for Parkinson's disease tremor detection. 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 143-147.

Zhang, H., Deng, K., Li, H., Albin, R. L., & Guan, Y. (2020). Deep learning identifies digital biomarkers for self-reported Parkinson’s disease. Patterns (New York, N.Y.), 1(3), 100042.

Zhang, H., Zeng, Y., Lu, H., Zhang, L., Li, J., & Qi, J. (2021). Learning to detect salient object with multi-source weak supervision. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44, 3577-3589.

Zhou, B., Pychynski, T., Reischl, M., & Mikut, R. (2018). Comparison of machine learning approaches for time-series-based quality monitoring of resistance spot welding (RSW). Archives of Data Science, Series A (Online First), 5(1), 13.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., & Schölkopf, B. (2004). Learning with local and global consistency. In *Advances in neural information processing systems* 321-328.

Zhou, Z. H., & Li, M. (2010). Semi-supervised learning by disagreement. *Knowledge and Information Systems*, *24*(3), 415-439.

Zhou, Z.-H. (2018). A brief introduction to weakly supervised learning. National science review, 5(1), 44-53.

Zhou, Z.-H., & Li, M. (2005). Semi-supervised regression with co-training. IJCAI.

Zhou, Z.-H., & Li, M. (2010). Semi-supervised learning by disagreement. Knowledge and Information Systems, 24, 415-439.

Zhu, X. (2008). Semi-supervised learning literature survey contents. *Sci. York*, *10*(1530), 10.

Zhu, X., Ghahramani, Z., & Lafferty, J. D. (2003). Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International conference on Machine learning (ICML-03)*, 912-919.

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., & He, Q. (2019). A comprehensive survey on transfer learning. Proceedings of the IEEE.

Zou, N. (2015). A probabilistic framework of transfer learning: Theory and application. Arizona State University.