TRANSFER LEARNING FOR BRAIN SIGNALS USING OPTIMAL TRANSPORT

A Dissertation
Presented to
The Academic Faculty

By

Ekansh Gupta

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in the
Department of Electrical and Computer Engineering

Georgia Institute of Technology

May 2024

© Ekansh Gupta 2024
TRANSFER LEARNING FOR BRAIN SIGNALS USING OPTIMAL TRANSPORT

Thesis committee:

Dr. Raghupathy Sivakumar  
Department of Electrical and Computer Engineering  
*Georgia Institute of Technology*

Dr. Faramarz Fekri  
Department of Electrical and Computer Engineering  
*Georgia Institute of Technology*

Dr. Karthikeyan Sundaresan  
Department of Electrical and Computer Engineering  
*Georgia Institute of Technology*

Date approved: April 25, 2024
ACKNOWLEDGMENTS

I express profound gratitude to Prof. Raghupathy Sivakumar, my supervisor, for his invaluable guidance and unwavering support throughout the completion of my dissertation. His direction and encouragement have been instrumental, and without them, this dissertation would not have come to fruition. I am honored to have worked under his excellent mentorship.

I thank all my labmates, Cheng-Yeh Chen, Shyam Krishnan, and Ching-Lun Tai for their helpful discussions and advice. I also thank my friends for being the wonderful people they are, and helping me out whenever I needed. I would also like to thank the members of my thesis committee for their help in the preparation of this work. Lastly and most importantly, I extend my heartfelt appreciation to my parents and family, who have stood by me at every juncture of my life, providing unwavering support. Their unmatched encouragement has been the primary driving force in my life and undoubtedly will continue to be so.
TABLE OF CONTENTS

Acknowledgments ................................................................. iii

List of Tables ................................................................. vi

List of Figures ................................................................. vii

Summary ................................................................. viii

Chapter 1: Introduction ......................................................... 1

Chapter 2: Background ......................................................... 5
  2.1 Types of BCIs .......................................................... 5
  2.2 BCI Research and Types of Signals .................................. 5
  2.3 The Error Potential Signal ........................................... 6
  2.4 Related Work .......................................................... 7
  2.5 Dataset ................................................................. 8

Chapter 3: Problem Definition and solution direction ................. 10
  3.1 Factors limiting generalization ...................................... 10
  3.2 Unsupervised Approach to Address Covariate Shift ............. 15
    3.2.1 Gaussian Mixture Models (GMM) ............................. 15
**Chapter 4: Partial Target-Aware Optimal Transport**

4.1 Optimal Transport

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1.1</td>
<td>20</td>
</tr>
<tr>
<td>4.1.2</td>
<td>21</td>
</tr>
<tr>
<td>4.1.3</td>
<td>23</td>
</tr>
<tr>
<td>4.1.4</td>
<td>26</td>
</tr>
</tbody>
</table>

**Chapter 5: Discussion and Conclusion**

5.1 Challenges and Future Work

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>27</td>
</tr>
</tbody>
</table>

5.2 Conclusion

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2</td>
<td>28</td>
</tr>
</tbody>
</table>

**References**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>References</td>
<td>30</td>
</tr>
</tbody>
</table>
LIST OF TABLES

3.1 Balanced accuracy for label-assisted/label-free stage 1 and 2 for xRG algorithm and the two hypotheses classes (MDM and ElasticNet).................. 15

4.1 Subject-wise cross-user transfer learning accuracy for label-free xRG vs our algorithm................................................................. 25
# LIST OF FIGURES

1.1 Some important applications of BCI .................................................. 2

1.2 covariate shift in signals from different users leading to poor generalization.  
Figure taken from [5] ................................................................. 2

2.1 Experimental framework ............................................................... 8

3.1 Different scenarios for when a model does not generalize ................. 11

3.2 Schematic of the xRG model with supervised blocks highlighted (orange). 11

3.3 Silhouette score as a measure of separability of two clusters. For each point  
$d_i$, $a(d_i)$ refers to the average distance from $d_i$ to all other points in its own  
cluster and $b(d_i)$ refers to the average distance from $d_i$ to all other points in  
another cluster. $C_1$ and $C_2$ are the two cluster centroids. Figure taken from  
[35]. ................................................................. 12

3.4 Per user Silhouette scores for label-free vs label-assisted embeddings and  
class discrimination ............................................................... 14

3.5 Predicting class-discrimination using unsupervised Gaussian Mixture Model 16

4.1 Different kinds of transports, namely positive (a $\rightarrow$ b), neutral (c $\rightarrow$ d), and  
negative (e $\rightarrow$ f) transports .................................................. 22

4.2 Partial target-aware optimal transport visualizations and examples of transport maps  .............................................. 24
SUMMARY

Brain-computer interfaces (BCIs) have surfaced as a powerful modality in human-machine interaction and wearable technology with powered futuristic applications like virtual reality, robot control, gaming, etc. Using BCIs, the brain’s intent can be harnessed without explicit communication. Despite the vast promise, systems designed for BCIs generalize poorly to new or unseen individuals due to high variability in brain signals among different subjects, resulting in long retraining/calibration sessions. This lack of generalization is typically attributed to a covariate shift of signals in the probability space, which manifests itself as disparate marginal and class conditional distributions. In this thesis, we overview the factors contributing to poor generalization on a more granular level by analyzing a specific brain signal called the Error Potential (ErrP), a signal well-known for its noisy characteristics and high variability, explore unsupervised and semi-supervised methods to improve its generalization performance, and propose a novel algorithm to mitigate the associated covariate shift using partial target-aware optimal transport. We demonstrate our method on an ErrP dataset collected in our lab. Our method outperforms state-of-the-art models for cross-user generalization which translates to a reduction in calibration time by an order of magnitude.
Brain-Computer Interfaces, or BCIs, are a direct communication pathway between the activity inside the brain and an external device or an AI system. By tapping directly into the brain, BCIs bypass the physical limitations of the body, like pressing buttons on a keyboard or giving commands using speech. Current research and availability of user-grade BCI headsets have unlocked significant possibilities for commercial BCI usage in entertainment [1], wellness [2], security [3], and other interactive wearable applications (refer Figure 1.1). The different types of BCIs range from highly intrusive, very high resolution to non-invasive over-the-scalp wearables. BCIs work by sensing the electrical/magnetic activity in the brain, decoding the user's intent by signal processing and machine learning algorithms. The final step involves a system that is responsible for the action taken in accordance with the interpreted activity, like a software system in a BCI-wearable for a virtual reality application, etc.

Most BCI applications use non-invasive techniques, like EEG, MEG, NIRS, etc., to record brain signals. While each of these techniques has its own utility, EEG remains by far the most popular option, partially because it lies in a unique sweet spot of cost-effectiveness, portability, and user-friendliness [4]. Despite the huge promise of BCIs and EEG in general as a crucial enabling technology for smart wearables, it faces some characteristic disadvantages. Systems designed for EEG suffer from low signal-to-noise ratio and high variance of user-specific brain signals, making its detection challenging. EEG data, while being very noisy, also exhibits a lot of variation and difference among different subjects, tasks, environments, and even different sessions for the same subject. A useful analogy for brain signals is that they are like fingerprints, in the sense that they are universal to all humans but they also have abundant individual differences. This hampers the
(a) BCI gaming is an emerging application of BCI

(b) Robot control through intrinsic neural feedback of BCI

Figure 1.1: Some important applications of BCI

Figure 1.2: covariate shift in signals from different users leading to poor generalization. Figure taken from [5]

generalization capabilities of algorithms that process these signals on new or unseen data. A common solution is calibrating a BCI device for every new user. As a real-world example, a mood detection wearable may work for one user’s brain signals but require hours of calibration on a new user to adapt to the latter’s brain signals.

Typically, covariate shift [6] is the contributing factor for poor generalization (ref Fig Figure 1.2). It is described as a shift in the probability distribution of a target/test dataset relative to the source/training dataset. Numerous solutions have been proposed to mitigate the effects of covariate shift which range from spatial filtering and domain adaptation
to deep learning methods. While these approaches attempt to estimate cleaner versions of these signals and marginally improve generalization, they do not fundamentally align the two distributions in the probability space. In this thesis, we look at different factors that contribute to poor generalization and demonstrate that while poor generalization is attributed to the covariate shift of the data, there are two distinct phenomena in the data that contribute to this covariate shift. We show that poor class separability and marginal distribution shift both contribute to a degradation in the generalization performance for new and unseen data. While the former can be addressed by the usage of better spatial filters, the latter can be mitigated by a combination of unsupervised and semi-supervised methods. We dissect the different factors contributing to poor generalization across users and propose an algorithm that not only aligns a high dimensional source and target probability distribution but also matches the positive and negative class labels with the target dataset, achieving seamless generalization that approaches the performance of a regular classification model, thereby significantly accelerating model adaptation and reducing calibration time by an order of magnitude. We demonstrate our algorithm on a real-world dataset of the error potential signal (ErrP), a brain signal that is well known for its poor generalization accuracy [7]. Our research contributions are as follows:

1. We closely examine the specific factors behind poor generalization and estimate an upper bound for two hypotheses classes for the generalization accuracy for our data, given supervised and unsupervised methods.

2. We propose a novel algorithm that uses partially estimated class centroids to adapt to a target domain and demonstrate our results on an ErrP dataset. The proposed algorithm comes within 95.6% of the accuracy of a label-assisted classifier while only using 5% of the labeled samples, thereby accelerating the process of adapting to a new user by an order of magnitude.

The rest of the thesis is organized as follows. Chapter 2 provides a background on the
ErrP signal, its poor generalization, and the related work done to address it. Chapter 3 provides an overview of poor generalization as a function of covariate shift and class separation. It also outlines an unsupervised approach (Gaussian Mixture Models) to improve the generalization accuracy of ErrP. Chapter 4 introduces Optimal Transport and outlines Partial Target-Aware Optimal Transport, a semi-supervised method for improving cross-user detection accuracy, and provides promising results on a target dataset, and then finally, section Chapter 5 talks about future work and concludes the thesis. Part of this work has been published in ACM SmartWear’23 [8].
CHAPTER 2
BACKGROUND

2.1 Types of BCIs

Broadly speaking, BCIs can be divided into two major categories, namely, invasive BCI, and non-invasive BCI (some publications refer to a third category called semi-invasive BCI as well). Invasive BCI refers to the systems where sensors and electrodes are surgically placed inside a subject’s scalp or grey matter, whereas non-invasive BCI refers to systems where the electro-chemical activity in the brain is externally recorded from the scalp. Invasive BCI methods have some advantages, like higher signal resolution and superior signal quality, but they also require risky procedures, which makes them not as attractive as non-invasive methods to the general population and thus, they have been limited to mostly medical applications.

Non-Invasive BCIs are divided into several categories. The most popular non-invasive BCI technology is Electroencephalography (EEG) which uses electrodes placed directly over a subject’s scalp to record brain activity. EEG has excellent temporal resolution but suffers from poor spatial resolution. Other forms of non-invasive BCI systems are Magnetoencephalography (MEG), functional Magnetic Resonance Imaging (fMRI), near-infrared spectroscopy (NIRS), etc. Most of these systems require bulky and expensive equipment to record signals and are therefore impractical for everyday usage, unlike EEG. EEG remains the most popular BCI system out of all invasive and non-invasive paradigms [9].

2.2 BCI Research and Types of Signals

Enterprising research on BCIs started flourishing in the 1970s when many researchers worked on establishing direct brain-to-machine communication [10]. In 1973, a Belgian
researcher, Jacques C. Vidal talked about "evoked potentials", which were variations in brain activity as a response to a specific sensory stimulus or event. Research on evoked potentials, also synonymously used with "Event-Related Potentials" (ERP), gained momentum in the '80s. In 1988, Farwell demonstrated that subjects can communicate 12 bits per minute without talking, using the P300 ERP in his paper, "Talking off the top of your head" [11]. In 1991, Wolpaw presented a system to mentally control a cursor using the 8-12Hz µ-frequency band [12]. In the same year, Falkenstein showed the presence of the "Error Potential" or ErrP in humans when they detected that an error had been committed in an experimental trial [13]

2.3 The Error Potential Signal

ErrP is a measure of the brain detecting/processing an error (for instance, seeing a robot perform a task incorrectly). In 1991, Falkenstein showed the presence of the Error Potential signal in humans in a choice-reaction experimental setup [13]. ErrPs are extremely valuable for BCI applications as they provide a generalized notion of error detection in a diverse set of tasks [14] across a wide variety of input modalities (e.g., audio [13], visual [15], somatosensory [16], etc.). ErrPs have a lot of promise and have been used in applications for improving the performance and reliability of BCI spellers [17], correcting and adapting AI systems, as well as aiding in learning for AI agents like correcting a robot’s mistakes [18] and accelerating learning for a reinforcement learning agent [19]. ErrP has been used in applications for improving the performance and reliability of BCI spellers [17], correcting and adapting systems to accelerate reinforcement learning for AI agents [20], etc. However, their generalization accuracy is inadequate owing to their high variability among individuals. For this purpose, we use Error Potential signals as our signal of choice in this thesis to evaluate our algorithms.

As previously mentioned, poor generalization adversely impacts the usability of a wearable, which makes its mitigation crucial. We address the generalization problem by looking
at the probability space of a signal and formulating our problem as a method to reconcile disparate distributions. Mathematically, we aim to reconcile $P(L_i|X_S)$ and $P(L_i|X_T)$, where $X_S$ and $X_T$ represent source and target distributions. $P(L_i|X_S)$ is the class conditional probability for our source dataset (the dataset on which we train our classifier), and $P(L_i|X_T)$ is the class conditional probability for the target dataset (where we evaluate our model). $L_i$ stands for our $i^{th}$ class label. In this thesis, we evaluate our generalization algorithms on the ErrP signal (ErrP).

### 2.4 Related Work

Traditionally, spatial filtering techniques [21] have been used to mitigate the generalization problem in ErrP signals as well as other BCI signals in general. The state-of-the-art method for cross-user ErrP detection uses XDAWN spatial filtering [22] and Riemannian Geometry [23]. This method works equally well for other ERPs (Event-related potentials) such as P300 [24]. [5] improved upon this method by introducing affine transforms which make the data reference across different users identical which improves the accuracy of cross-user generalization. More recently, deep learning models have emerged which outperform traditional spatial filtering approaches for specific tasks. There have been RNN-based zero-shot learning methods [25] for classifying object classes using EEG. DeepConvNet and ShallowConvNet proposed by [26] are deep learning models for EEG decoding and visualization. With EEGNet [27], the authors created a shallow deep learning model with 1082 or 2290 parameters (depending on the configuration) and showed promising numbers for ErrP detection. [28] used ErrP detection from multiple observers and few-shot learning to improve the generalization accuracy of ErrP signals.

While these methods provide incremental performance improvements, they do not address the fundamental reason for poor generalization in ErrP signals, which is the incongruous probability distributions in the train and test domains. To address this challenge, several works have used domain adaptation techniques like optimal transport [29] to mod-
ify the probability distribution of the signal data in the feature space. Regularized optimal transport [30] is an effective technique for aligning disparate probability distributions by keeping similar labels close to each other. [31] used regularized optimal transport with class labels to improve the transfer learning of P300 signals. Similarly, [32] used regularized optimal transport in the domain of semi-positive definite (SPD) matrices and used the Riemannian distance metric to align ERP (Event-related potential) signal distributions. However, these methods often do not provide sufficient cross-user accuracy for ErrP signals (more details in section section 4.1) due to a lack of target class distribution information. In this work, we address this problem by estimating the target class centroids and using this information to minimize the disparity between a source and a target ErrP signal distribution, in terms of the marginal as well as the class-conditional distribution.

2.5 Dataset

For this purpose, we use an ErrP dataset collected in our lab. For the collection of this dataset, We use the BIOPAC CAP-100C electrode cap that continuously captures the electrical activity of a subject at 125Hz (electrode placement map shown in Fig Figure 2.1c). We simulate an environment where a computer agent plays an Atari-based maze game (ref Figure 2.1a and Figure 2.1b) while occasionally making wrong moves, which is being observed by a human wearing a BCI headset. The agent is free to move along the top/right/bottom/left directions and at some cells in the grid, it is possible to have multiple
right or wrong actions. We utilized ten human subjects (mean age 26.7 with a standard deviation of 1.6, 2 female) to perform this experiment. More details about this dataset can be found in [20]. This study was approved by the Institute Review Board (IRB) and included the EEG data of 10 human subjects recorded from 16 electrodes/channels over their scalp in a non-invasive manner. The data was sampled at 125Hz and each signal instance comprised a time window of 1.5 seconds. To remove high-frequency noise, we pass the signals through a 4-th order Butterworth filter with frequency ranging from 0.5Hz to 40Hz and select 10 channels (C3, C4, Cz, P3, P4, Pz, F3, F4, Fz and Fp2) located near the angulate cingulate cortex region of the brain as they are more relevant to the ErrP signal. The total number of samples for the 10 users is 4350. We use balanced accuracy for our evaluation since it penalizes models that are overly specific and sensitive to any class label and thus, favors a more robust model.
CHAPTER 3
PROBLEM DEFINITION AND SOLUTION DIRECTION

3.1 Factors limiting generalization

In this section, we investigate the reasons for the poor generalization accuracy of ErrP detection algorithms. We start with the xDAWN + Riemannian Geometry (referred to as xRG from hereon) based supervised model that obtains state-of-the-art performance for ErrP generalization [23]. Figure 3.2 shows the general schematic of the xRG model. It contains two stages (denoted by dashed lines) that use supervised learning, namely the template generation stage (stage 1), and the classification stage (stage 2), and hence requires the ground-truth labels of a user’s data. The template generation stage generates signal embeddings from raw signal data using a template estimated for each class label in a supervised manner. These embeddings are in the form of covariance matrices and are classified as belonging to ErrP or non-ErrP classes by a classifier. The aim of generalization is to systematically replace these supervised stages with label-free stages so as to generate signal embeddings without using a target user’s labels as well as using a classifier that is completely blind to the target distribution. We iteratively turn these two stages from target-label-assisted to target-label-free. For any given target user, the embeddings are generated either with or without the labels of the target user, and the classifier is either trained with or without the target user’s distribution. We analyze the decline in accuracy in the 4 combinations as follows:

1. Label-assisted stages 1 and 2: for a user $U_i$, labels from $U_i$ are used for creating embeddings, which are used with labels from $U_i$ for training the classifier.

2. Label-free stage 1 + label-assisted stage 2: for $U_i$, labels from $U_{j\neq i}$ are used for creating embeddings which are used with labels from $U_i$ for training the classifier.
Figure 3.1: Different scenarios for when a model does not generalize

Figure 3.2: Schematic of the xRG model with supervised blocks highlighted (orange).
Figure 3.3: Silhouette score as a measure of separability of two clusters. For each point $d_i$, $a(d_i)$ refers to the average distance from $d_i$ to all other points in its own cluster and $b(d_i)$ refers to the average distance from $d_i$ to all other points in another cluster. $C_1$ and $C_2$ are the two cluster centroids. Figure taken from [35].

3. Label-assisted stage 1 + label-free stage 2: $U_i$’s embeddings are generated using $U_i$’s labels, which are classified by a classifier trained on $U_{j\neq i}$’s embeddings.

4. Label-free stage 1 + label-free stage 2: $U_i$’s embeddings are generated using $U_{j\neq i}$’s labels, which are classified by a classifier trained on $U_{j\neq i}$’s embeddings.

We apply two classifiers for evaluation: the minimum-distance-to-mean (MDM) classifier [33] and the ElasticNet classifier [34]. The mean per-user detection accuracies for these 4 scenarios are shown in Table 3.1. We used 5-fold cross-validation for this analysis, wherein we partition the training data into 5 equal folds and each fold is iteratively used for model validation and the remainder of the data for training. When a model is used in a completely label-assisted manner, it achieves an overall 5-fold cross-validation accuracy of 76% and 78.8% for MDM and ElasticNet, respectively. Such accuracy serves as an upper bound of the possible accuracy (for the respective classifiers) as all the steps are label-assisted.

Since we make the stages label-free one by one, we observe a drop in the overall detection accuracy. In the second case, the overall detection accuracy drops down to 65.4%
and 71.3% for MDM and ElasticNet, respectively. Covariate shift is not responsible for the drop in accuracy since the training and test data are sampled from the same distribution (the classifier is label-assisted). The decline in accuracy is attributed to diminished class discrimination in the label-free embeddings compared to the label-assisted case. This can also be empirically shown by measuring the extent of class discrimination in our embeddings using a supervised metric like the Silhouette score [36]. This score measures how closely grouped the points in the same class are compared to points from other classes (ref Fig Figure 3.3). For each data point in a distribution, the Silhouette score for that data point is defined as follows:

\[
s(i) = \frac{b(i) - a(i)}{\max\{b(i), a(i)\}},
\]

where \(a(i)\) is the average distance between the data point and all other points in its own class, and \(b(i)\) is the average distance between the data point and all the other points belonging to another class. The total Silhouette score is then a mean of all the \(s(i)\)’s calculated for each point in the distribution. The amount of class discrimination present in a dataset imposes a fundamental lower bound on the classifier error trained on that dataset. The Silhouette score, which is a measure of class discrimination in a dataset is a good predictor of the maximum classifier accuracy achievable on that dataset [37]. A higher Silhouette score correlates with higher classification accuracy for that user. In our experiments, we calculate this score for every user on their embeddings, for both instances, where the embeddings are generated in a label-assisted and a label-free manner, and show their graph in Figure Figure 3.4. Note that stage 2 being label-free or label-assisted has no bearing on the Silhouette score as it does not change the embeddings. Figure Figure 3.4 shows the per-user Silhouette scores for two cases, when the embeddings are generated with labels vs when they are generated label-free. The label-free embeddings have a lower Silhouette score and thus, lower class discrimination, resulting in poor classification performance even with supervised classifiers. Therefore, in this case, diminished class discrimination (not covariate shift) is the contributor to low accuracy.
Figure 3.4: Per user Silhouette scores for label-free vs label-assisted embeddings and class discrimination

(a) Silhouette score for label-assisted and label-free embeddings

(b) Label-free embeddings with lower class discrimination

(c) Label-assisted embeddings with higher class discrimination
Table 3.1: Balanced accuracy for label-assisted/label-free stage 1 and 2 for xRG algorithm and the two hypotheses classes (MDM and ElasticNet).

<table>
<thead>
<tr>
<th>MDM / ElasticNet</th>
<th>Label-assisted stage 1</th>
<th>Label-free stage 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label-assisted stage 2</td>
<td>76.0% / 78.8%</td>
<td>64.8% / 71.3%</td>
</tr>
<tr>
<td>Label-free stage 2</td>
<td>57.1% / 60.3%</td>
<td>55.8% / 59.1%</td>
</tr>
<tr>
<td>Silhouette score</td>
<td>0.0202</td>
<td>0.0116</td>
</tr>
</tbody>
</table>

In the third scenario, since the embeddings are generated in a label-assisted manner, the decline in performance is a result of using a label-free classifier, i.e., a classifier that was trained on data with a different distribution as compared to the test data. In order to overcome this, we need to adapt the source dataset (consisting of embeddings that the classifier is trained on) to the target dataset such that not only do the marginal distributions of both datasets achieve parity, but the class conditional distributions also equalize.

3.2 Unsupervised Approach to Address Covariate Shift

We first explore if we can obtain superior generalization performance using a completely label-free approach. As seen in Table 3.1, making the first stage label-free puts an upper bound on the final accuracy irrespective of whether the second stage is label-free or label-assisted. This reduction of the upper bound is caused by diminished class separability as discussed above. By making both stages label-free, we see that by using the state-of-the-art method, we obtain generalization accuracies 55.8% and 59.1% for the two hypotheses classes respectively. In this section, we investigate if we can recover the class-separabilities by using unsupervised Gaussian-Mixture Models (GMM).

3.2.1 Gaussian Mixture Models (GMM)

Gaussian Mixture Models [38] is a method for soft-clustering of data points in a distribution. It assumes that the distribution is composed of a mixture of Gaussian distributions. Formally, a Gaussian Mixture comprises $K$ Gaussian distributions which the means $\mu_i$, covariance matrix $\Sigma_i$, and mixing probability $\pi_i$ for $i \in 1, 2, .., K$. Let $f_i(x)$ denote the
conditional probability of point $x$ given that it belongs to cluster $i$, the probability of the point $x$ is given by:

$$p(x) = \pi_1 * f_1(x) + \pi_2 * f_2(x) + ... + \pi_K * f_K(x) \tag{3.2}$$

Where $\sum_{i=1}^{K} \pi_i = 1$. Given that the conditional probability functions are modeled as Gaussians, this can be re-written as:

$$p(x) = \pi_1 * N(x|\mu_1, \Sigma_1) + \pi_2 * N(x|\mu_2, \Sigma_2) + ... + \pi_K * N(x|\mu_K, \Sigma_K) \tag{3.3}$$

Where $N(x|\mu_K, \Sigma_K)$ represents the conditional probability w.r.t. the $K^{th}$ Gaussian. If we denote the $i^{th}$ Gaussian as $z_i$, the membership of a point $x$ is calculated as:

$$P(z_i|x_n) = \tau(z_{nk}) = \frac{p(x_n|z_i) * p(z_i)}{p(x_n)} \tag{3.4}$$

$$P(z_i|x) = \frac{N(x|\mu_i, \Sigma_i) * \pi_i}{\pi_1 * N(x|\mu_1, \Sigma_1) + \pi_2 * N(x|\mu_2, \Sigma_2) + ... + \pi_K * N(x|\mu_K, \Sigma_K)} \tag{3.5}$$

Where $\tau(z_{nk})$ is also known as the responsibility of a Gaussian $k$ given a data point $x_n$. Given a distribution assumed to be a mixture of Gaussians, Gaussian Mixture Models
compute the underlying parameters $\mu_i, \Sigma_i, \pi_i$ for all $K$ components by using a maximum likelihood estimator by maximizing the total probability of the distribution. This estimator is as follows:

$$\arg \max p(X|\theta = \mu, \Sigma, \pi) = \prod_{i=1}^{N} p(x_i)$$ (3.6)

$$\arg \max p(X|\mu, \Sigma, \pi) = \prod_{i=1}^{N} \sum_{k=1}^{K} \pi_k \ast N(x_i|\mu_k, \Sigma_k)$$ (3.7)

$$\ln(p(X|\mu, \Sigma, \pi)) = \sum_{i=1}^{N} \ln(\sum_{k=1}^{K} \pi_k \ast N(x_i|\mu_k, \Sigma_k))$$ (3.8)

By differentiating this MLE and setting the partials to zero, we obtain the following formulae for calculating $\mu_i, \Sigma_i, \pi_i$ for all $i \in 1, 2, \ldots, K$.

$$\mu_k = \frac{\sum_{n=1}^{N} \tau(nk)x_n}{\sum_{n=1}^{N} \tau(nk)}$$ (3.9)

$$\Sigma_k = \frac{\sum_{n=1}^{N} \tau(nk)(x_n - \mu_k)(x_n - \mu_k)^T}{\sum_{n=1}^{N} \tau(nk)}$$ (3.10)

$$\pi_k = \frac{\sum_{n=1}^{N} \tau(nk)}{N}$$ (3.11)

Since our dataset has each data point represented as a positive semi-definite matrix in the Riemannian space, we use the Riemannian distance metric to calculate our sample probabilities as well as cluster means. We denote the respective equations as follows:

$$p(x_n|z_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(\delta(x_n, \mu_k))^2}{2\sigma_k^2}}$$ (3.12)

$$\sigma_k^2 = \frac{\sum_{n=1}^{N} (\delta(x_n, \mu_k))^2}{N_k - 1}$$ (3.13)

$$\mu_k = \arg \min_{\mu_k} \sum_{i=1}^{N} \delta^2(x_i, \mu_k)$$ (3.14)

$$\delta(A, B) = ||\log(A^{-1/2}BA^{-1/2})||_F = (\sum_{i=1}^{n} \log^2 \lambda_i)^{1/2}$$ (3.15)

Where $\sigma_k$ is the cluster-specific variance and $\delta(x_n, u_k)$ is the Riemannian distance be-
Algorithm 1: Expectation Maximization for Gaussian Mixture Models

1: **Input:** Source set \( \{ S_i | i = 1, \cdots, n_s \} \) and target set \( \{ T_i | i = 1, \cdots, n_t \} \) with different class weights \( C_S \) and \( C_T \) respectively. Convergence threshold \( T \).

2: **Initialization:** cluster centers \( \mu_1 \) and \( \mu_2 \).

3: **for Until Convergence do**

4: **E-Step:** Calculate responsibilities \( \tau(n,k) \).

5: **M-Step:** Calculate \( \mu_k, \sigma_k, \) and \( \pi_k \) based on \( \tau(n,k) \).

6: **Calculate:** Maximum likelihood \( \prod_{i=1}^{N} \sum_{k=1}^{K} \pi_k * p(x_n | z_k) \). Exit if increment < \( T \).

7: **end for**

8: **Return:** \( \tau(n,k) \).

For our case, we take \( K \), or the number of clusters to be two. The corresponding parameters are iteratively re-estimated until the objective function (the MLE function) is maximized. This method is called the Expectation Maximization algorithm (EM) and operates in two stages. In the first stage, point-specific responsibilities are calculated based on the current iteration of parameters for each point and each cluster. In the second stage, the maximum likelihood is calculated by plugging in the probabilities of each point based on the calculated responsibility. This is repeated until the MLE stops increasing or its successive increment reaches a specific threshold. The full algorithm is outlined in Algorithm 1.

Since our method is fully unsupervised, we use the unbalanced class weights of the two classes (ErrP and non-ErrP) to ascertain which cluster belongs to which class. For the case of balanced class-weights, this method can be extended by using a 1-shot approach where we query the ground-truth class of a random data point and assign the cluster that the data point belongs to, to the queried class and the remaining cluster to the other class.

Figure 3.5 shows an example where the unsupervised GMM predicts the class-specific embeddings for label-free stage-1. Upon obtaining the predicted class memberships of the signal embeddings, we use MDM and ElasticNet to train a classifier on the target data distribution.

The final generalization accuracy for the two classifiers by using label-free embed-
ding generation and Gaussian Mixture Models comes out to be 55.2% (MDM) and 57.1% (ElasticNet). We can see that unsupervised GMM not only underperforms, it also fails to outperform the baseline generalization accuracy as seen in Table 3.1 when both stages are label-free (55.8% and 59.1% respectively).

This can be attributed to diminished class separabilities (as evidenced by the reduced Silhouette score in Table 3.1) in the generated embeddings using unsupervised (label-free) methods. For our dataset, GMM fails to recover the ground-truth class memberships of these embeddings and thus yields sub-optimal performance.

In the next chapter, we address this problem by using a semi-supervised approach called Partial Target-Aware Optimal Transport.
4.1 Optimal Transport

4.1.1 Problem formulation

Optimal transport is the general problem of adapting one distribution to another as efficiently as possible. It requires a cost matrix that denotes the cost of moving a specific sample from the source distribution to another sample from the target distribution. Given two distributions, their associated cost matrix, and their marginal probabilities, the transport map for the source distribution is obtained by minimizing the objective function outlined below. Supposed that the densities of the source and target measures are sampled at $n_s$ and $n_t$ discrete points, we can denote the source and target probability densities by $\mathbf{a} \in \mathbb{R}^{n_s}$ and $\mathbf{b} \in \mathbb{R}^{n_t}$, respectively. Since they are sampled from a vector space, we set $[a_i = 1/n_s \forall 0 \leq i \leq n_s]$ and $[b_i = 1/n_t \forall 0 \leq i \leq n_t]$ Define $\langle \cdot, \cdot \rangle_F$ as the Frobenius inner product. The optimal transport problem is to find a transport plan $\gamma \in \mathbb{R}^{n_s \times n_t}$ from the source domain to the target domain that is the most efficient with respect to a cost matrix $\mathbf{M} \in \mathbb{R}^{n_s \times n_t}$. In this paper, we consider the entropy regularized optimal transport, which can be computed using Sinkhorn distances [39]. The optimal transport plan $\gamma$ for this problem is obtained by minimizing:

$$
\gamma = \arg \min_{\gamma} \langle \gamma, \mathbf{M} \rangle_F + \lambda \Omega_e(\gamma) + \eta \Omega_g(\gamma) 
$$ (4.1)

subject to $\gamma 1 = \mathbf{a}$, $\gamma^T 1 = \mathbf{b}$, $\gamma \succeq 0$ (4.2)

We set $M_{i,j}$, the transport cost between the $i$-th point $S_i$ in the source domain and the
$j$-th point $T_j$ in the target domain, as the square of the Riemannian distance between these two points: $M_{i,j} = \left\| \log(T_j^{-1/2}S_iT_j^{-1/2}) \right\|_2^2$ as defined in [33], since the source and target points are both covariance matrices lying in a Riemannian manifold in our scenario. There are two regularization terms in the OT problem: the entropic regularization term, $\Omega_e(\gamma) = \sum_{i,j} \gamma_{i,j} \log(\gamma_{i,j})$, and the group lasso regularization term, $\Omega_g(\gamma) = \sum_{i,c} \| \gamma_{i,c} \|_2^2$. $\Omega_e(\gamma)$ is used to ensure that the source data is transported smoothly to the target domain instead of abruptly in only a few locations. $\Omega_g(\gamma)$ is used to ensure that the source data maintains its class discrimination after transportation (labels of the same class are transported close together). We set the value of $\lambda$ to $0.02 \times \text{Median}(M)$ and set the value of $\eta$ to be 5.

Since these points are all positive semi-definite matrices, the distance in the case is calculated using the Riemannian distance metric.

### 4.1.2 Types of transport maps

Equation (Equation 4.1) can be solved in accordance with the matrix scaling algorithm outlined in [39] to get the transport map $\gamma$. However, directly solving (Equation 4.1) does not yield a good transport plan as the ground-truth relationship may not be necessarily captured by the minimization of transport cost in (Equation 4.1). During our experiments, we observed three distinct cases of transports, namely, positive, neutral, and negative transports. Visualizations for all these are shown in Figure Figure 4.1. For positive transport, the source domain maintains its class discrimination after transport, and its ErrP points are adjacent to the target ErrP points and vice versa. This is the desirable scenario as the classifier trained on the transported source data generalizes seamlessly to the target dataset and is characterized by high training accuracy on the source and high test accuracy on the target dataset. For a neutral transfer, the source ErrP and non-ErrP points do not necessarily show an affinity towards a specific class’ points in the target dataset. This case is characterized by high training accuracy on the source and close to random accuracy (50%) on the target dataset. Finally, for negative transport, the source ErrP points are adjacent to the non-ErrP
Figure 4.1: Different kinds of transports, namely positive (a → b), neutral (c → d), and negative (e → f) transports
Algorithm 2 Partial target-aware optimal transport

1: **Input:** Source set \( \{S_i | i = 1, \cdots, n_s\} \) with its density \( a \in \mathbb{R}^{n_s} \) and target set \( \{T_i | i = 1, \cdots, n_t\} \) with its density \( b \in \mathbb{R}^{n_t} \). Few-shot class labeled target sets \( \{L_0^i | i = 1, \cdots, m_0 \} \) with \( m_0 \ll n_t \) and \( \{L_1^i | i = 1, \cdots, m_1 \} \) with \( m_1 \ll n_t \).

2: **Initialization:**
   \[ M_{i,j} = \| \log \left( \frac{T_j^{-1/2} S_i T_j^{-1/2}}{2} \right) \|_2^2. \]

3: \[ C_0 = mean(L_0), C_1 = mean(L_1), \] the initial approximation of target class centroids.

4: **for** \( i = 1, \cdots, n_s \) **do**
   5:     **for** \( j = 1, \cdots, n_t \) **do**
   6:         \[ D_{j0} = dist(T_j, C_0), D_{j1} = dist(T_j, C_1) \]
   7:         **if** \( \text{class}(S_i) == 0 \) **then**
   8:             \[ \Delta = D_{j0}/D_{j1} \]
   9:         **else**
   10:             \[ \Delta = D_{j1}/D_{j0} \]
   11:         **end if**
   12:         \[ M'_{i,j} = M_{i,j} \times \Delta \]
   13: **end for**
   14: **end for**
15: \[ \gamma = \arg\min_{\gamma} \langle \gamma, M' \rangle_F + \lambda \Omega_e(\gamma) + \eta \Omega_g(\gamma) \] s.t. (Equation 4.2).
16: **Return:** \( \gamma \).

points in the target dataset and vice versa. Under this scenario, the respective labels of the source and target dataset are negatively matched. This case is characterized by high training accuracy on the source and very low (~50%) test accuracy on the target dataset. These three kinds of transfer occur uniformly in our dataset, which results in the overall performance of OT-based generalization being roughly equal to 50% (accuracy of a random classifier).

In the next subsection, we propose an algorithm that maximizes positive transport while suppressing neutral and negative transport.

4.1.3 Partial target-aware optimal transport

In order to mitigate negative transfer, we propose “partial target-aware optimal transport” by modifying the cost matrix \( M \) to establish the desired relationship between the source and target points. The full algorithm is detailed in Algorithm 2. In line 3, we first calculate the Riemannian mean of the centroids of the target data class by using only a few labeled samples from the target dataset (\( m_0 = 10 \) from class 0 and \( m_1 = 10 \) from class 1 in our
Figure 4.2: Partial target-aware optimal transport visualizations and examples of transport maps.

After obtaining these approximate centroids, we bias the transport map to avoid transporting source labels to an area that is close to the centroids of another class. From line 6 to line 12, we engineer the cost matrix $M$ by increasing the distance with a dynamically calculated factor between a source point and a target point which is closer to the centroid of another class than the centroid of the source point’s class. Similarly, we decrease the distance by a dynamically calculated factor between a source point and a target point that is closer to the centroid of the same class as the source point than the centroid of another class. Once we obtain $\gamma$, the transport map for the source distribution, the barycentric mapping for a source point $S_i$ is calculated as the weighted Riemannian...
Table 4.1: Subject-wise cross-user transfer learning accuracy for label-free xRG vs our algorithm.

<table>
<thead>
<tr>
<th></th>
<th>xRG MDM</th>
<th>PTA-OT MDM</th>
<th>xRG ElasticNet</th>
<th>PTA-OT ElasticNet</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>59.2%</td>
<td>61.2%</td>
<td>62.6%</td>
<td>66.6%</td>
</tr>
<tr>
<td>S2</td>
<td>53.8%</td>
<td>63.0%</td>
<td>56.2%</td>
<td>67.6%</td>
</tr>
<tr>
<td>S3</td>
<td>54.8%</td>
<td>61.4%</td>
<td>61.4%</td>
<td>63.2%</td>
</tr>
<tr>
<td>S4</td>
<td>60.4%</td>
<td>65.8%</td>
<td>61.4%</td>
<td>63.9%</td>
</tr>
<tr>
<td>S5</td>
<td>54.2%</td>
<td>59.9%</td>
<td>58.6%</td>
<td>68.5%</td>
</tr>
<tr>
<td>S6</td>
<td>56.7%</td>
<td>60.6%</td>
<td>59.4%</td>
<td>64.1%</td>
</tr>
<tr>
<td>S7</td>
<td>54.3%</td>
<td>59.0%</td>
<td>57.9%</td>
<td>60.1%</td>
</tr>
<tr>
<td>S8</td>
<td>55.2%</td>
<td>64.4%</td>
<td>53.8%</td>
<td>74.0%</td>
</tr>
<tr>
<td>S9</td>
<td>53.5%</td>
<td>60.9%</td>
<td>58.0%</td>
<td>62.2%</td>
</tr>
<tr>
<td>S10</td>
<td>55.5%</td>
<td>63.8%</td>
<td>62.0%</td>
<td>71.54%</td>
</tr>
<tr>
<td>Mean</td>
<td>55.8%</td>
<td><strong>62.0%</strong></td>
<td>59.1%</td>
<td><strong>66.2%</strong></td>
</tr>
</tbody>
</table>

mean of the $n_t$ target points, with the weight factor equal to $\gamma(i :,)$ which refers to the $i^{th}$ row of the transport map.

We want the following conditions to be true during and after the source domain is transported to the target domain.

- The target data should have some class discrimination (although we cannot control it as the target covariance matrices are generated in an unsupervised manner)
- Source data should have some class discrimination (usually the case as it is generated in a supervised manner unless extremely noisy data)
- The source data should be transported to approximately the same probability space as the target data. This can be controlled by the regularization parameter in the optimal transport equation $= reg.$
- The source data maintains its class discrimination after transportation. This can be controlled by the group lasso regularization term $\eta$.
- The source data is positively matched with the target data (positive source class labels are transported near positive target class labels and far away from negative target class labels and vice versa)
4.1.4 Performance evaluation

Table 4.1 details the per-user test accuracy when using partial target-aware optimal transport. Our preliminary results show that the total mean accuracy for our algorithm is equal to 62.0% for the MDM classifier (accuracy using fully label-assisted MDM classifier was 64.8%) and 66.2% for the ElasticNet classifier (accuracy using fully label-assisted ElasticNet was 71.3%). In the event a better classifier is used for ErrP detection, we expect its benefits to be distributed equally to the fully label-assisted as well as our method. If we express our accuracies as percentages with respect to the accuracy achieved by label-assisted classifiers, given the label-free embeddings, we are able to reach within 95.6% and 92.8% of the accuracy for MDM and ElasticNet, respectively. Figure 4.2 shows a few combinations of the source and target probability distributions before and after our algorithm. As we can see in the figure, the distribution after transport not only preserves the class discrimination in the source domain but also matches ErrP points from the source user (blue) to the target user (green) and non-ErrP points from the source user (orange) to the target user (red). Our results not only outperform the cross-user generalization performance of the state-of-the-art xRG model but also approach supervised classification performance for label-free embeddings while using only a small fraction of the target labels (20 as opposed to $\approx 400$), thereby accelerating model transfer by an order of magnitude. Please note that our algorithm is a general-purpose algorithm that works with all kinds of data distributions that suffer from covariate shifts and minimizes the disparity between marginal source and target distributions while also preserving the class conditional probabilities.
CHAPTER 5
DISCUSSION AND CONCLUSION

5.1 Challenges and Future Work

In this thesis, we attempted to address the poor generalization that is observed in brain signals, especially using non-invasive techniques like EEG. We chose Error Potential signals to be our target signal owing to their high variance among users and low generalization accuracy. There are several challenges associated with improving the generalization accuracy of these signals, which include but are not limited to:

- Noise: BCI signals are often very noisy and suffer from very low SNR. Typically, the voltage of the electrical neural activity inside the brain is of the order of \( \mu V \), which is further attenuated as this signal travels through the cranial structures with different compositions and conductivities to reach the scalp. Combined with interference from competing signals from nearby regions of the brain, this signal is contaminated with high levels of associated noise by the time it reaches the top of the scalp to be recorded. Given the high levels of noise, data-limited approaches like unsupervised and semi-supervised learning suffer due to an increased number of outliers as well as the availability of limited data to estimate better signal templates.

- Data Availability: Given the shortcomings of unsupervised approaches, supervised approaches also are limited by the lack of availability of BCI datasets. Brain signals are hard to obtain because collecting BCI data involves long and controlled sessions in a lab environment, which makes this process burdensome. As a result, data-hungry approaches like deep learning are unfit for the scenario where limited data is available.
• Representation Shift due to Changes: Brain signals are very susceptible to situations like slight movements, a change in electrode placement, or the neurophysiological state of the subject. All these changes introduce perturbations in the elicited signals which are manifested as representation shifts in the probability space. This contributes to the lack of generalization across tasks, environments, users, and physiological states.

In this context, we outline a few directions where future work can address the above-mentioned shortcomings.

• The improvement to the proposed algorithm can be two-staged. Firstly, better spatial filters and/or neural networks can be used to estimate the signal embeddings using unsupervised/semi-supervised methods such that the class discrimination between the samples is sufficiently close to the supervised case. Secondly, given the generated embeddings, different hypothesis classes can be explored for optimal performance.

• The proposed algorithm can also be refined to obtain the optimal barycentric mapping given the source and target distribution. The avenues for improvement lie in more accurate centroid estimation, cost-matrix generation based on other probabilistic measures (we used Riemannian distance in our approach), and finding optimal hyperparameters for the objective function.

5.2 Conclusion

In this thesis, we outlined an approach to improve transfer learning performance for a noisy and difficult-to-generalize brain signal. We demonstrated different scenarios where generalization accuracy is poor and modeled the contributors to it. We dissected the factors responsible for poor generalization into two distinct stages showing lower class discrimination as well as covariate shift due to unseen data. We used an unsupervised approach like Gaussian Mixture Models (GMM) to utilize the label-free embedding distribution to
generate labeled embeddings based on a mixture of Gaussians. We evaluated its generalization performance on our dataset and listed its shortcomings responsible for its poor performance. We then demonstrated an approach to mitigate the effects of covariate shift using partial target-aware optimal transport and obtained state-of-the-art performance on our dataset. Our preliminary results show significant potential in using partial target-aware optimal transport to mitigate the effects of covariate shifts in cases of transfer learning. As a next step, we aim to derive a mathematical model for optimal transport which maximizes positive transport. We also aim to experiment with more datasets as well as use different label-free methods to generate embeddings that preserve class discrimination.
REFERENCES


