# Optimal Session-to-Session Transport for BCI

**Steffen Schneider**
Technical University of Munich, Munich, Germany
`steffen.schneider@tum.de`

## Abstract

In this report, we present and discuss a pipeline to classifier error-related potentials (ErrPs) in electroencephalography (EEG) recordings during a cursor movement task. We compare several approaches for preprocessing, feature selection and classification. Maybe surprisingly, good performance is obtained even with simple feature selection methods apply directly to the time traces. We obtain an ACC of XX %, F1-score %, AUC % using a final ensemble of a Random Forest and SVM classifier. [1]

## 1  Introduction

[4, 5, 6, 15, 2, 1, 7, 11, 10, 13, 3, 14, 12, 9, 16, 8]

## 2  Methods

### 2.1  Feature Selection

To determine the relevance of different time points, the t-statistic between the two classes is computed. From this, the channel and timesteps of highest significance, i.e., with lowest $p$-value, is selected. [2]

For the baseline method, the identified timepoints were directly used for the further analysis.

- Peak-Picking
- Signal Averaging
- Spectogram analysis

### 2.2  Classification

For robust classification, an ensemble classifier is constructed in order to obtain a measure of uncertainty over the predictions. Candidates for the ensemble are Random Forest Classifiers (RFOs) and Support Vector Machines (SVMs) with variable hyperparameters.

The models are evaluated

1.

For the final evaluation, a 10-fold cross validation is used.

---

[1] The code for the approach is accessible at `github.com/stes/bci`.

[2] Already at this point, we shall note that the t-test is not designed to compare p-values. This is why the p-values here are of no statistical meaning of interest and should be considered as a mean to perform feature selection.
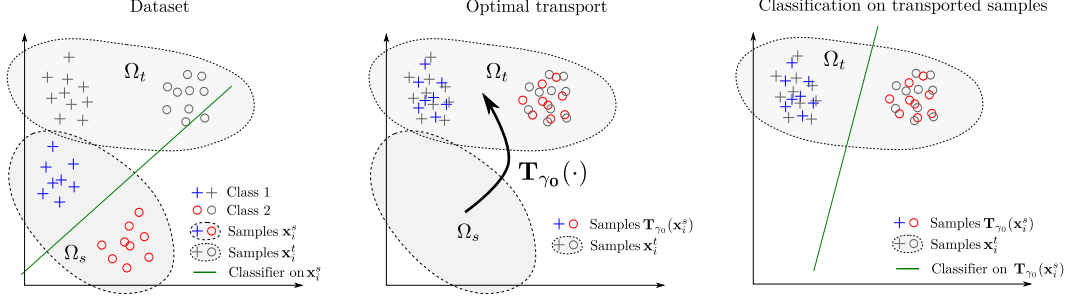
Figure 1: Schematic overview of optimal transport. Given the source and target domains $\Omega_s$ and $\Omega_t$, optimal transport is used to compute a mapping $T$ between source and target domain, transforming the training dataset to match the target distribution. Finally, the classifier is fitted to $T(\Omega_s)$, making it more suitable for classifying the data from $\Omega_t$. In the context of BCI research, we propose the use of this technique between features from different sessions or subjects. Reproduced from [3].

## 2.3 Visualization

As labels are not available for the validation set, we offer a qualitative evaluation of the classifier performance.

## 2.4 Domain Adaptation by Optimal Transport

In this work, we adapt the notion of domain adaptation also given in the review by [13]. A domain is denoted as a tuple $(\mathcal{X}, \mathbb{P}_{\mathcal{X}})$ of a space $\mathcal{X}$ and a distribution $\mathbb{P}_{\mathcal{X}}$. In our setting, we do not only deal with a single source and a single target domain, but rather with a set of domains $\{\mathcal{X}^k\}_{k=1}^N$ which are part of a common signal space $\mathcal{U} \supset \mathcal{X}^k \,\forall k \in [N]$. We consider each domain here to be fully defined as a set of samples directly given by $\mathcal{X}^k := \{\mathbf{x}^{(j)}\}_{j=1}^{N_k}$ drawn *i.i.d.* from $\mathbb{P}_{\mathcal{X}}$.

As a last requirement, we assume that a feature space $\mathcal{V}$ and *measurement functions* $\Phi^k : \mathcal{V} \mapsto X^k$ exists such that for each $\mathbf{x} \in \mathcal{X}^k$, there exists $v \in \mathcal{V}$ with $\Phi^k(v) = \mathbf{x}^k$. This view can also be extended in a probabilistic way by adding noise to the measurement process. For a subset of domains with $k \in \mathcal{I}$, labels $\mathcal{Y}^k = \{y^{(j)}\}_{j=1}^{N_k}$ are available. Goal of the adaptation is to be able to apply an algorithm fitted to $(\mathcal{X}^{\mathcal{I}}, \mathcal{Y}^{\mathcal{I}})$ on all data domains by transforming $\mathbf{x} \in \mathcal{X}^k$ to $\mathbf{x}' \in \mathcal{X}^{\mathcal{I}}$ such that the latent representation $v \in \mathcal{V}$ (the content) of both samples is preserved.

Recently [3] presented a methods to apply mechanisms from optimal transport [4, 6] to this domain adaptation problem. We evaluate Optimal Transport for session-to-session transfer in the detection of ErrPs. The general scheme of transferring a source to the target domain using label information is depicted in figure 1.

# 3 Experiments

Two datasets were considered for the evaluation procedure. In addition to the original dataset, we evaluated classifier performance on the BCI dataset by [**?**]. We evaluate the approach on the ICS ERP Dataset, which is comprised of 300 training and 300 validation epochs. However, in this report, we will only report results for the ICS ERP dataset.

## 3.1 Feature Selection

Results on the training set in figure **??** A yield timepoints of 208.9 ms for the N200 and 291.1 ms for the P300 responses. Especially during the P300 response, highest differences are observed in the Cz electrode. During the N200 response and the later response denoted as P400 (which is probably a delated P300), we found the Pz electrode to offer the best discriminative power.
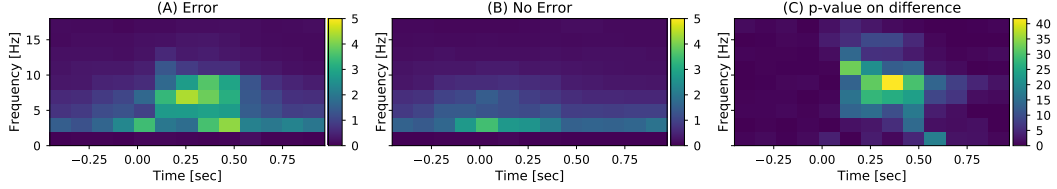
2

Train
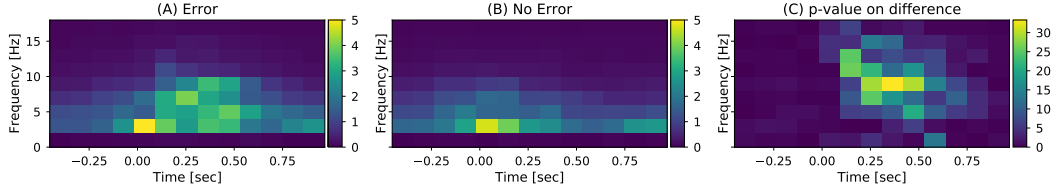


Test



Figure 2: Spectrogram analysis of the dataset. (A) and (B) depict the mean spectograms for the Cz channel during Error and NoError events, respectively. To evaluate discriminative power, a p-test is performed between the samples, yielding the distribution depicted in (C) for the Cz channel.

| Fold | LDA | LinSVM | RFO |
|------|--------|--------|--------|
| 0 | 87.2 % | 85.7 % | 85.7 % |
| 1 | 86.8 % | 86.2 % | 84.8 % |
| 2 | 88.3 % | 88.3 % | 88.0 % |
| 3 | 85.7 % | 85.8 % | 84.7 % |
| 4 | 87.5 % | 87.8 % | 85.7 % |

Table 1: Results

As one qualitative measure of test performance, we compute the same statistics for the test set and compare these timesteps. While the N200 and P400 responses are very consistent over datasets, a significant shift could be observed for the exact P300 timepoints.

### 3.2 Classifiers

Given the limited amount of data samples, we evaluated the use of SVMs and Random Forests as the classifiers. Data was preprocessed using an Independent Component Analysis (ICA).

We select most significant points for the different events:

- N200: $[180\text{ms}, 210\text{ms}]$
- P300: $[250\text{ms}, 350\text{ms}]$
- delayed P300: $[350\text{ms}, 450\text{ms}]$

Quite consistently, we found that linear SVMs outperform RFOs and RBF SVMS in classification performance. Below, we report the cross-validation results for the approaches:

- RFO: The number of tree ensembles was varied between 2 and 20. After cross-validation, the value was fixed to 18
- Linear SVM: The regularization parameter $C$ was varied from $10^{-6}$ to $10^2$. Best performance was obtained for $10^{-1} \ldots 1.5$.
- RBF SVM: The regulariziation parameter was varied in the same range as for the linear SVM. However, no satisfying performance was obtained for the simple feature set, which is why this method was dismissed after cross validation

### 3.3 Session-to-session Transfer

Using the Tübingen Dataset, we estimated performance in session-to-session transfer settings.
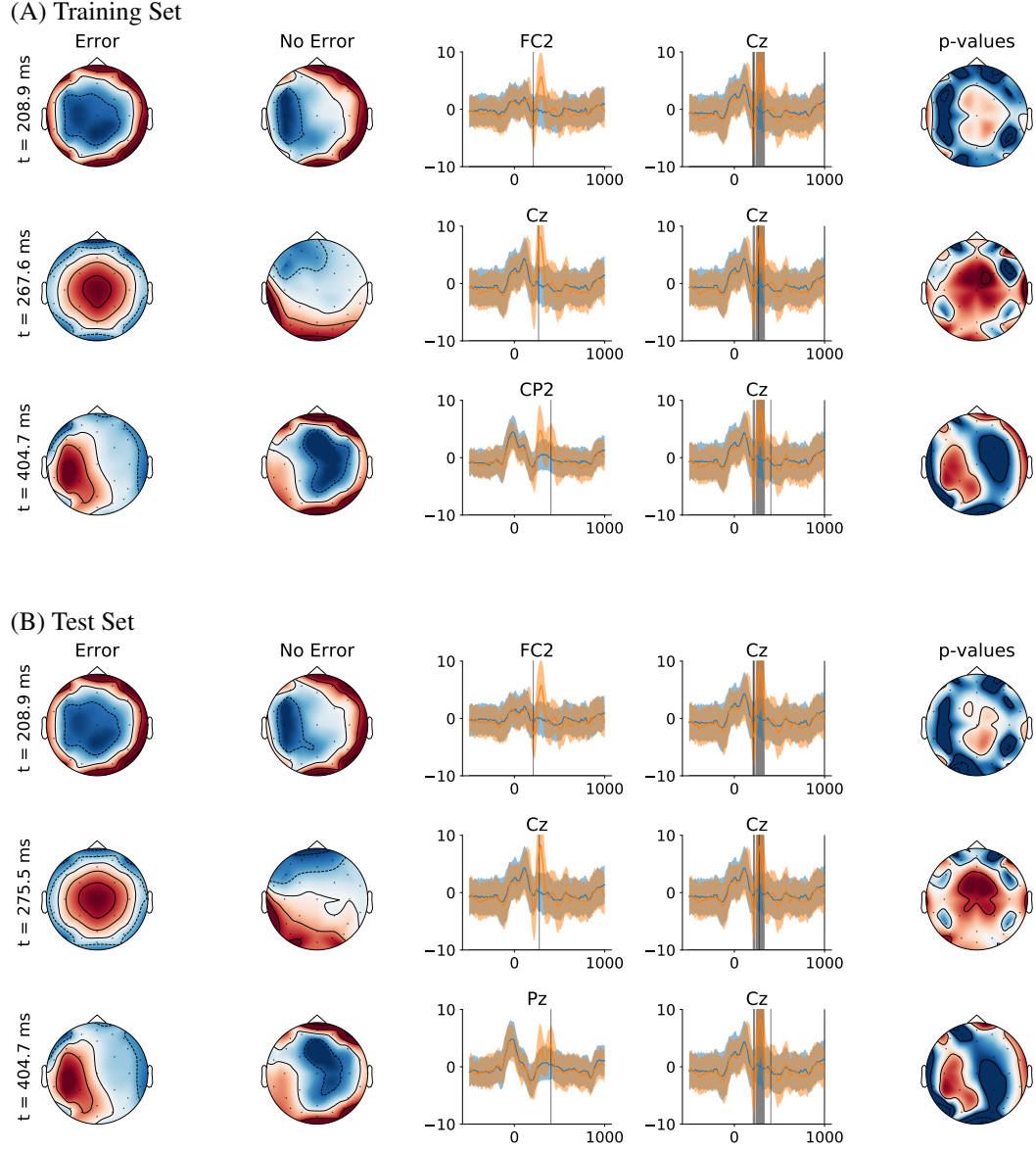
Figure 3: Spectrogram analysis of the test dataset. (A) and (B) depict the mean spectograms for the Cz channel during Error and NoError events, respectively. To evaluate discriminative power, a p-test is performed between the samples, yielding the distribution depicted in (C) for the Cz channel.

|   | Ensemble | Final | OT Ensemble | OT Final |
|---|----------|-------|-------------|----------|
| 0 | 86.7 % | 90.3 % | 100.0 % | 100.0 % |
| 1 | 89.0 % | 90.0 % | 100.0 % | 100.0 % |
| 2 | 81.7 % | 76.0 % | 100.0 % | 98.0 % |
| 3 | 85.7 % | 81.0 % | 100.0 % | 98.7 % |
| 4 | 88.0 % | 87.3 % | 100.0 % | 99.3 % |
| 5 | 89.7 % | 92.3 % | 100.0 % | 98.7 % |
| 6 | 81.7 % | 89.3 % | 100.0 % | 100.0 % |
| 7 | 91.0 % | 91.7 % | 100.0 % | 99.3 % |
| 8 | 85.3 % | 88.0 % | 100.0 % | 100.0 % |
| 9 | 95.7 % | 97.0 % | 100.0 % | 100.0 % |

# 4 Discussion

Given the small size of the dataset, interpretation of performance and generalization of the results presented here should be done with care.

**Adaptive Normalization Techniques**  We propose the use of techniques known from style transfer [**?**] to perform domain adaptation for medical data. While the original algorithm was based on an iterative optimization algorithm, recent work deals with training a deep neural network to perform feed-forward stylization. Interestingly, it was shown that it is sufficient to retrain the parameters of the network's Batch Normalization [**?**] layers. Inspired by this technique, Instance Normalization [**?**] and Adaptive Instance Normalization [**?**] were developed. In the context of color normalization for domain adaptation in digital pathology, Feature Aware Normalization (FAN) [1] was developed using similar underlying principles.

Building on the work of [1], we investigate FAN for artifact removal and normalization of EEG signals and other electrophysiological recordings. For training, the domains $\mathcal{X}^k$ can be chosen to match patients (subject-to-subject transfer), time (session-to-session transfer) or the signal acquisition equipment.

# References

[1] D. Bug, S. Schneider, A. Grote, E. Oswald, F. Feuerhake, J. Schüler, and D. Merhof. Context-based normalization of histological stains using deep convolutional features. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10553 LNCS:135–142, aug 2017.

[2] X. Chai, Q. Wang, Y. Zhao, Y. Li, D. Liu, X. Liu, and O. Bai. A Fast, Efficient Domain Adaptation Technique for Cross-Domain Electroencephalography(EEG)-Based Emotion Recognition. *Sensors*, 17(5):1014, may 2017.

[3] N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy. Optimal Transport for Domain Adaptation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, X(X):1–14, 2015.

[4] M. Cuturi. Sinkhorn Distances: Lightspeed Computation of Optimal Transportation Distances. 2013.

[5] S. Ehrlich and G. Cheng. A neuro-based method for detecting context-dependent erroneous robot action. *IEEE-RAS International Conference on Humanoid Robots*, pages 477–482, 2016.

[6] R. Flamary, M. Cuturi, N. Courty, and A. Rakotomamonjy. Wasserstein Discriminant Analysis. 2016.

[7] X. Huang and S. Belongie. Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization. *arXiv preprint*, 2017.

[8] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 448–456, 2015.

[9] M. A. Lebedev and M. A. L. Nicolelis. Brain-machine interfaces: past, present and future. *TRENDS in Neurosciences*, 29(9):536–546, 2006.

[10] Y. Li, N. Wang, J. Liu, and X. Hou. Demystifying Neural Style Transfer. 2017.

[11] F. Lotte. A Tutorial on EEG Signal Processing Techniques for Mental State Recognition in Brain-Computer Interfaces. *Eduardo Reck Miranda Julien Castet. Guide to Brain-Computer Music Interfacing*, 2014.

[12] J. d. R. Millán, R. Rupp, G. Mueller-Putz, R. Murray-Smith, C. Giugliemma, M. Tangermann, C. Vidaurre, F. Cincotti, A. Kubler, R. Leeb, and Others. Combining brain–computer interfaces and assistive technologies: state-of-the-art and challenges. *Frontiers in neuroscience*, 4:161, 2010.

[13] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[14] S. Silvoni, A. Ramos-Murguialday, M. Cavinato, C. Volpato, G. Cisotto, A. Turolla, F. Piccione, and N. Birbaumer. Brain-computer interface in stroke: a review of progress. *Clinical EEG and Neuroscience*, 42(4):245–252, 2011.

[15] M. Spüler and C. Niethammer. Error-related potentials during continuous feedback: using EEG to detect errors of different type and severity. *Frontiers in human neuroscience*, 9:155, 2015.

[16] S. Stober, A. Sternin, A. M. Owen, and J. A. Grahn. Deep Feature Learning for EEG Recordings. 2015.

# A  Analysis