

---

# A SYSTEMATIC EVALUATION OF EUCLIDEAN ALIGNMENT WITH DEEP LEARNING FOR EEG DECODING

---

Bruna Junqueira<sup>1,2</sup> Bruno Aristimunha<sup>2,3</sup> Sylvain Chevallier<sup>2</sup> Raphael Y. de Camargo<sup>3</sup>

<sup>1</sup>University of São Paulo, Sao Paulo, Brazil

<sup>2</sup>Université Paris-Saclay, Inria TAU team, LISN-CNRS, France

<sup>3</sup>Federal University of ABC, Santo Andre, Brazil

## ABSTRACT

Electroencephalography (EEG) signals are frequently used for various Brain-Computer Interface (BCI) tasks. While Deep Learning (DL) techniques have shown promising results, they are hindered by the substantial data requirements. By leveraging data from multiple subjects, transfer learning enables more effective training of DL models. A technique that is gaining popularity is Euclidean Alignment (EA) due to its ease of use, low computational complexity, and compatibility with Deep Learning models. However, few studies evaluate its impact on the training performance of shared and individual DL models. In this work, we systematically evaluate the effect of EA combined with DL for decoding BCI signals. We used EA to train shared models with data from multiple subjects and evaluated its transferability to new subjects. Our experimental results show that it improves decoding in the target subject by 4.33% and decreases convergence time by more than 70%. We also trained individual models for each subject to use as a majority-voting ensemble classifier. In this scenario, using EA improved the 3-model ensemble accuracy by 3.71%. However, when compared to the shared model with EA, the ensemble accuracy was 3.62% lower.

**Keywords** Brain-Computer Interfaces · Neural Network · Euclidean Alignment

## 1 Introduction

Electroencephalography (EEG) signals are the most common type of neurophysiological data used for decoding tasks in Brain-Computer interfaces (BCI). It is non-invasive, portable, and has a low acquisition cost, unlike other technologies, such as magnetoencephalography (MEG) and functional magnetic resonance imaging (fMRI), which are expensive and require large equipment, and electrocorticography (ECoG), which require surgical procedures. Despite those advantages, EEG data also presents some drawbacks. It has a low signal-to-noise ratio (SNR), is vulnerable to the position of the electrodes in the head, and has a poor spatial resolution.

Deep Learning (DL) algorithms have shown promising results in the BCI field, in some cases outperforming traditional machine learning methods for EEG classification [10, 20, 32, 33]. Despite the promising results of DL, it's important to acknowledge that DL models are data-hungry [21]. Existing efforts predominantly focus on either training a single model for a group of subjects (*Cross-Subject*, or *Inter-Model*) or training an individual model for each subject (*Within-Subject*, or *Intra-Model*) [32, 33, 46].

A way to improve the performance of DL models would be to use data from multiple subjects. Along with the expected increase in accuracy, it could reduce the calibration time to avoid unnecessary cognitive fatigue from the user [17]. But brain signals depend on the morphology and psychological state of the individual at the instant of the recording [24]. Therefore, variabilities of a person's signals between different moments in the same or between sessions make the decoding process more challenging. When considering different subjects, using models trained in one subject for decoding another can even worsen the predictions, a phenomenon called *negative transfer* [29]. These signal variabilities are one of the main factors causing the generalization problem in the BCI field, which is exacerbated by the relatively small size of EEG-based datasets.

This is the problem targeted by the field of Transfer Learning (TL) [35], which aims to improve the transferability of models between domains, where each domain can be an individual in a given task. There are two main approaches for TL: task and domain adaptation [29]. In task adaptation, one trains the model in a given domain and, when applying it to the target domains, fine-tune the model parameters to improve the performance on the target domain. In domain adaptation, we use the same model in both domains but change the input data so that both domains have similar representations.

When considered in the BCI application, a task adaptation would be to train the model on another subject and *fine-tune* the model using labeled data from the target [15, 45]. When using domain adaptation, one would apply pre-processing transformations to make the source and target EEG signals more similar [36, 48]. One well-known technique is the Riemannian Alignments (RA) [2, 23, 30, 47], which aims to reduce the distance between the subjects' covariance matrices in the Riemannian manifold [44]. For instance, Rodrigues et al. [30] propose a method based on the Procrustes analysis, applying geometrical transformations (re-centering, re-scale, rotation) on the Riemannian manifold. On the other hand, Zanini et al. [47] insert paradigm information trying to re-centering based on the resting state trials. However, RA transformations are done on covariance matrices, which limits the use of standard state-of-the-art DL models that work on raw EEG data. Even though there are efforts to apply SPDNet [14] to EEG decoding [39], these networks have not yet been well studied for transfer learning tasks or with alignments.

Another domain adaptation method is the Euclidean Alignment (EA) [13], which works directly on the EEG signal. This form of data alignment reduces the distance between data distributions by centering their Euclidean mean covariance matrix to the identity. As a result of its simplicity and computational efficiency, there has been an increased interest in using EA for EEG decoding, usually as a pre-processing step combined with state-of-the-art models [7, 19, 37], and has been showing some good results on both traditional machine learning [9, 22] and deep learning pipelines [41] and in the context of merging datasets [27]. Despite its growing popularity, there are limited studies on how EA can affect deep learning models' training and transfer performance for shared models trained with data from multiple subjects and individual models trained with data from single subjects.

Deep learning models are the main beneficiaries of transfer learning approaches since they benefit from large amounts of data. However, most existing DL approaches that use data from multiple subjects tend to ignore or overlook the issue of inter-subject variability [3–5, 11, 16, 42, 43], probably with the expectation that the DL model could accommodate for these differences. For this reason, it is essential to evaluate if applying data alignment approaches as a preprocessing step can improve the learning and inference processes. Moreover, it is crucial to closely examine the alignment mechanism's intricacies, its impact on the training set, and its influence on transferability across subjects when employing the EA. Other researchers had already compared the use of the Euclidean Alignment with deep learning models [19, 25, 28, 41], but in a very concise manner. Wu et al. [41], in their review, have compared the use of the EA in different Machine Learning and some DL models. In the DL case, however, they just considered the offline cross-subject models.

In this work, we make a systematic evaluation of the effect of EA combined with deep learning models for decoding BCI signals. We used EA to train shared models with data from multiple subjects and evaluated its transferability to new subjects. Our experimental results show that it improves decoding in the target subject by 4.33% and decreases convergence time by more than 70%. We also applied EA to models trained from individual source subjects and evaluated the improvements in transferability to target subjects, both for a single model and for majority-voting classifiers. In this scenario, using EA improved the 3-model classifier accuracy by 3.71%. However, when compared to the shared model with EA, the majority-voting classifier accuracy was 3.62% lower.

## 2 Methodology

This section provides a sound and complete explanation of the Euclidean Alignment method and describes the datasets used and the different training schemes proposed.

### 2.1 EEG Decoding

EEG decoding is the problem of classifying EEG signals, for instance, to retrieve mental states from the recorded signals [18]. We can define dataset  $E$  as the sample of  $N$  pairs of trials and labels,  $E = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ . The trial recordings  $\mathbf{x}$  are matrices having  $c$  rows and  $t$  columns, where  $c$  is the number of channels used for the sampling process and  $t$  is the number of time steps. Therefore, the feature space is  $\mathcal{X} \in \mathbb{R}^{c \times t}$ . In our experiments, we deal solely with the left-right imagery paradigm, and the decoding labels space is  $\mathcal{Y} = \{\text{'left\_hand'}, \text{'right\_hand'}\}$ , encode as  $\{0, 1\}$ , respectively. We work with data from more than one subject, and we consider each subject ( $k$ ) to have their own data  $E_k = \{(\mathbf{x}_i^k, y_i^k)\}_{i=1}^n$ , with  $\{\mathcal{X}_k, \mathcal{Y}_k\} \in \mathbb{R}$ .

Because of brain signal variability, we consider that the marginal distributions  $P_k(\cdot)$  of EEG data  $E$  are different from subject to subject. The classification hypothesis assumes that there exists some unknown decision function  $f_\theta(\cdot)$  with  $\theta$  parameters that result in  $f_\theta(\mathbf{x}) = y$ . We are interested in optimizing the  $\theta$  parameters to allow us to decode non-labeled brain signals. This is done by feeding a neural network with some labeled data (training  $t$  data formed by  $N_t < N$  examples) and minimizing the average loss  $\ell$ :

$$\min_{\theta} \frac{1}{N_t} \sum_{i=1}^{N_t} \ell(f_\theta(\mathbf{x}_i), y_i) \quad (1)$$

In all our experiments, we used the negative log-likelihood loss ( $\ell$ ) and AdamW optimizer for the minimization step.

## 2.2 Transfer Learning

As defined in [50], a *domain*  $\mathcal{D}$  is the set formed by the feature space  $\mathcal{X}$  and a marginal distribution  $P(\mathbf{X})$ , with  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$ . The *task*  $\mathcal{T}$  is composed of the label set  $\mathcal{Y}$  and the decision function  $f$ , which we want to discover. Mathematically,  $\mathcal{D} = \{\mathcal{X}, P(\mathbf{X})\}$  and  $\mathcal{T} = \{\mathcal{Y}, f(\mathbf{X})\}$ . We consider each person a *domain*, and we call the source domain the domain with known knowledge, and the target is the unknown one. Transfer learning is the framework by which, given a source domain ( $\mathcal{D}_s$ ), source task ( $\mathcal{T}_s$ ), and target domain ( $\mathcal{D}_t$ ), target task ( $\mathcal{T}_t$ ), with  $\mathcal{D}_s \neq \mathcal{D}_t$  or  $\mathcal{T}_s \neq \mathcal{T}_t$ , the knowledge from the source ( $s$ ) is used to learn the target's ( $t$ ) (knowledge).

Since we deal with data from different subjects, we can define a domain as  $\mathcal{D}_k = \{\mathcal{X}_k, P_k(\mathbf{X}^k)\}$  and a task  $\mathcal{T}_k = \{\mathcal{Y}_k, f_\theta^k(\mathbf{X}^k)\}$  for each individual  $k$ . Given of the nature of the data, it is reasonable to consider that feature space  $\mathcal{X}_k$ , label space  $\mathcal{Y}_k$ , and decision function  $f_\theta^k(\cdot)$  share a same attribute space for each subject  $k$ , but, the marginal distributions  $P_k(\cdot)$  of EEG data are different from subject to subject. We use a *leave-one-subject-out* approach, with multiple sources ( $k - 1$  subjects) with different marginal distributions  $P(\cdot)$ , where we approximate the marginal distributions using a data alignment technique called *Euclidean Alignment*.

We also evaluated the transferability between every pair of subjects, using a model trained in a source subject  $f_\theta^s(\cdot)$  to decode the signal on all other subjects  $t$ . We define the mean accuracy of the  $f_\theta^s(\cdot)$  to classify the EEG signals of other subjects as the *transferability* of the model. Similarly, we denote as the *receivability* of the model as the mean accuracy when applying the individual models  $f_\theta^s(\cdot)$  from all source subjects  $s$  to the EEG data from the target subject.

## 2.3 Euclidean Alignment

The inherent variability in brain signals poses a significant challenge in developing a model that can effectively generalize across diverse individuals. To address this challenge, we analyze the effectiveness of Euclidean Alignment (EA) [13], which matches each subject's mean covariance matrix of EEG trials with the identity matrix, reducing their dissimilarities [12].

Suppose that, for each of the  $N$  subjects, we have a set of  $n$  trials,  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ . For each subject  $j$ , we first calculate the arithmetic mean  $\bar{R}^j$  of the covariance matrices of all trials.

$$\bar{R}^j = \frac{1}{n} \sum_{i=1}^n X_i^j X_i^{j\top}, \quad (2)$$

where  $X_i^j$  represents the covariance matrix for the  $i$ th EEG trial, of the  $j$ th subject. Afterward, we define the transformation matrix as the square root of the inverse of  $\bar{R}^j$  and apply it to each trial of the subject:

$$\tilde{X}_i^j = (\bar{R}^j)^{-1/2} X_i^j. \quad (3)$$

As a result, the arithmetic mean of the covariance matrices of each subject becomes equal to the identity matrix, making the distributions more similar, as illustrated in Figure 1.

## 2.4 Riemannian Alignment

Riemannian Alignment (RA) [8] is another set of transformations to align EEG signals, but applied to the covariance matrices of EEG signals. Some of these transformations, such as the complete Riemannian Procrustes (RPA) Rodrigues

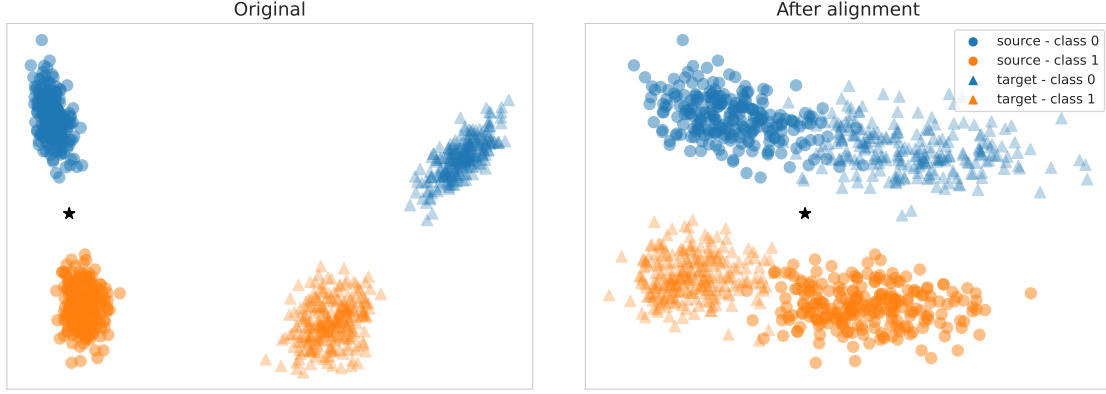


Figure 1: Euclidean Alignment steps illustrated. The star represents the origin of the space.

et al. [30], are difficult to adapt for use on raw EEG data. Consequently, we decided to use the first step of the RPA, which is the recentering to the geometric mean. This method takes advantage of the congruence invariance property [8] of the Affine-Invariant metric to recenter the mean of a distribution to the Identity matrix. Let  $M_R^j$  be the geometric mean of the covariance matrices  $\Sigma_1, \Sigma_2, \dots, \Sigma_n$  of subject  $j$

$$M_R^j = \arg \min_{C \in \mathcal{P}} \sum_{i=1}^n \delta^2(C, \Sigma_i^j), \quad (4)$$

with  $\mathcal{P}$  the space of  $c \times c$  SPD matrices and  $\delta$  the Affine-Invariant distance. Since  $(M_R^j)^{-1/2}$  is also SPD, by the invariance property,

$$\delta(\Sigma_i, M_R^j) = \delta((M_R^j)^{-1/2} \Sigma_i (M_R^j)^{-1/2}, (M_R^j)^{-1/2} M_R^j (M_R^j)^{-1/2}) = \delta((M_R^j)^{-1/2} \Sigma_i (M_R^j)^{-1/2}, I). \quad (5)$$

Therefore, the recentered matrix is obtained by  $(M_R^j)^{-1/2} \Sigma_i (M_R^j)^{-1/2}$ . To maintain the data in the time domain, we applied the geometric mean  $\bar{R}_R$  directly to the EEG signal:  $\bar{R}_R^{-1/2} X$ .

The Riemannian Procrustes has other steps, but those need to be applied directly to the covariance matrices. He and Wu [13] used the recentering to the resting state [47] instead of the geometric mean of trials to compare to EA. However, we consider that the geometric mean is more similar to the steps performed by the Euclidean Alignment.

## 2.5 Fine-Tuning

We used linear probing, a sub-type of fine-tuning. We define a neural network  $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$  as a composition of different functions  $f_\theta = f_{\theta_1} \circ \dots \circ f_{\theta_l}$ , where each function  $i$  have  $\theta_i$  corresponding to parameters from a subpart of the model. In the fine-tuning, we replace the last layer by  $f_\phi$ , resulting in  $f'_{\theta'} = f_{\theta_1} \circ \dots \circ f_{\theta_{l-1}} \circ f_\phi$ , and update only the weights of this last layer, freezing the others.

## 2.6 Datasets

We used two datasets available at the MOABB [1] repository. The first is the dataset IIa from BCI Competition 4 [34] (BNCI2014). It consists of data from 9 healthy individuals, where each individual can perform four different motor imagery tasks: left hand (class 1), right hand (class 2), both feet (class 3), and tongue (class 4). It also has two sessions, recorded on different days, named train session (*session\_T*) and evaluation session (*session\_E*). Each session contains six runs, separated by a *short break*, and each run contains 12 trials per class. We used both sessions, but only the first two classes, following other BCI studies. The EEG signals contain 22 channels sampled with a frequency of 250 Hz.

The second dataset is the High-Gamma [33], denoted here as *Schirrneister2017*, containing data from 14 healthy individuals performing motor execution tasks: left hand, right hand, both feet, and rest. Just as it was done with BNCI2014, we used solely left- and right-hand tasks to maintain results comparable. Data is divided into two sessions: training, with approximately 880 trials, and testing, with approximately 160 trials. The EEG trials contain 128 channels, sampled at 512 Hz.

We employed standard pre-processing steps designed for the *Motor Imagery* (MI) based in [1, 26]. We applied band-pass filtering with the overlap-add method between  $[8 - 32]$  Hz and resampled the signals to 250Hz. Additionally, we selected the 22 channels based on Tangermann et al. [34] to standardize with the BNCI2014 dataset.

## 2.7 Models and Data Splitting

We analyzed the use of EA in two different scenarios:

- i. Using Cross-Subject models (*shared*) trained with data from all source subjects (Figure 2a). We relied on a Leave-One-Subject-Out (LOSO) validation strategy for data split: at each time, we picked one subject as the target (test set) while using the remaining subjects as the training data;
- ii. Using *individual* models, one per source subject, trained with data only from that subject. We evaluated each model separately (Figure 2b) or combined in majority-voting classifiers (Figure 2c), using the other subjects as test subjects.

We then use these models to perform two-class motor imagery classification in the target subjects. These methodologies were carefully chosen to ensure robust evaluations and unbiased results by preventing data leakage between test, validation, and training sets [38].

For the analysis, we evaluated each scenario using *offline* and *pseudo-online* evaluation approaches [6, 40]. For the offline one, we used all data from the target subject to determine the reference matrix for the Euclidean alignment, while in the pseudo-online, we used only the first group of 24 trials from the target to compute the reference matrix. For the experiments with fine-tuning, we also used the first 24 trials in the fine-tuning process. We selected 24 trials to match the length of a single run in the *BNCI2014* dataset and by following the study conducted by [40], which recommends using between 16 and 32 trials for alignment.

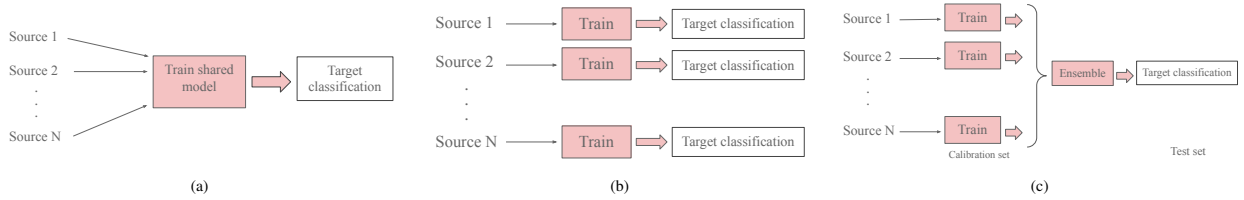


Figure 2: Shared, individual, and majority-voting classifier models. (a) We use data from all source subjects to train a single shared model, with and without EA. (b) We use data from each source subject to train a separate individual model, with and without EA. (c) We use calibration data from the target subject to select the best source models and combine them in a majority-voting classifier.

We used EEGNet [20], ShallowNet, and DeepNet [33] for the evaluations since they are the standard benchmarks in BCI tasks while showing good accuracy in BCI cross-subject classification tasks [41]. We used the model’s implementations from BRAINDECODE [33].

## 2.8 Data alignment

For the EEG alignment step, we applied different methods for the *offline* and *pseudo-online* experiments. For the offline approach, we had access to all trials for each subject, so we aligned each subject in groups of 24 trials to reduce noise in the reference matrices for transformations [47]. Regarding the simulated online context, we aligned the target data using the first group of 24 trials from the target subject to compute the reference matrix and then used this matrix to align the test data. In the BNCI2014001 dataset [34], there are 12 trials per class per run, and as we use two events (left and right hand), we simply align each run.

However, the High-gamma dataset [33] presented an additional challenge as it does not have a consistent number of trials per subject, and it may not always be a multiple of 24. To address this, we performed an extra pre-alignment step where we randomly removed some trials while maintaining the class proportion, ensuring that the remaining trials were divisible by 24.

## 2.9 Model Training

We trained the *shared models* using the leave-one-subject-out paradigm. We used all data from the source subjects since the models were evaluated in different target subjects. Inside the source subjects set, we initially randomly separated 20% for validation and 80% for training. For the shared models, we used a batch of 64 trials with 1000 epochs, an early stopping of 250 iterations, and a dropout rate of 25%. For each architecture, we executed a Grid search using

leave-one-group-out cross-validation to optimize the learning rate and weight decay, with the search space shown in Table 1. We ended up with  $w_d = 1e^{-5}$  weight decay for all models and a learning rate of  $\eta = 1e^{-3}$  for both EEGNet and DeepNet and  $\eta = 8.25e^{-4}$  for Shallow ConvNet. When using fine-tuning, we retrained them for an additional 600 epochs, with a patience of 150. During fine-tuning, we used the first 24 trials from each training subject and the first 24 trials from the target.

For the *individual models*, we also used all data for training, with 20% for validation, since the models were evaluated on target subjects distinct from the source ones. We trained for 1000 epochs, with a patient of 250 and a dropout rate of 35%. We executed a hyper-parameter tuning to select the best learning rate and ended up with  $\eta = 8.25e^{-4}$  for both EEGNet and ShallowNet and  $\eta = 7.25e^{-4}$  for DeepNet, as shown in Table 1. For the individual models, we didn’t execute fine-tuning, as it would be impractical in a real scenario to fine-tune each individual model to each target.

Table 1: Search space of hyperparameters for the grid search procedure.

Hyperparameters ( $1e^{-4}$ )	Models	
	Shared	Individual
Learning rate	[12.5, 10.0, 8.25, 6.25]	[12.5, 10.0, 9.25, 8.25, 7.25]
Weight decay	[0, 0.10, 1]	[0, 0.10, 1]

### 3 Results

We evaluated the effectiveness of Euclidean Alignment for the offline and pseudo-online scenarios, using the *shared* and *individual* models. For the shared models, we trained a separate model for each target subject using the leave-one-subject-out paradigm. For the individual models, we trained each network only once, as the majority-voting classifiers use different combinations of the individual models. We performed multiple experiments to answer the following research questions:

1. **EA impact:** Does EA improve a shared model transfer performance in the offline and pseudo-online contexts?
2. **Fine tuning:** Does fine-tuning improve the performance of shared models?
3. **Transferability:** How transferable are models learned on source subjects to a target subject?
4. **EA transferability:** Does EA improve the transferability of individual models across subjects?
5. **Individual vs shared models:** How do majority-voting classifiers compare to shared models when using EA?

For individual models, we organized three different scenarios to evaluate the transfer learning performance and to choose the best classifier for the target:

1. **best-model(-EA):** we use the first run of the target subject as a calibration set to select the individual model that provided the best accuracy and, in the EA case, to align the EEG data from the target subject;
2. **k-Models(-EA):** we use the first run of the target subject as a calibration set to select the  $k$  best models to be used as a majority voting classifier and, in the EA case, to align the EEG data from the target subject.

We tested the significance of the results using a one-tailed permutation-based t-test using MOABB’s [1] functions *compute\_dataset\_statistics* and *find\_significant\_differences* to compute the p-values for each pair of pipelines. The null hypothesis is that the performance of the pipelines is equal.

#### 3.1 Transfer-learning using Euclidean aligned shared models

Euclidean Alignment improved the average decoding performance on all evaluated datasets and architectures, as shown in line No-EA, offline-EA, and online-EA in Table 2. The offline-EA had an accuracy of 1.26% better than the online-EA, which occurs because all the target data is used to estimate the covariance matrices, resulting in a better alignment. However, the online-EA had an overall mean accuracy of 4.33% better than the non-aligned case, showing the benefit of using EA in a more realistic scenario. Finally, using EA and RA resulted in very similar results, with slightly better accuracies for one of the approaches depending on the model or dataset.

When considering the individual target subjects (Figure 3), there was no improvement in the median of subject accuracy in BNCI2014 for the EEGNet and ShallowNet, but the accuracy of the best and worst subjects improved in most cases,

Table 2: Mean accuracy of different training schemes for the shared model using different datasets and pipelines.

Pipelines	Shared Models			Datasets		Overall
	EEGNet [20]	DeepNet [33]	ShallowNet [33]	BNCI2014 [34]	Schirrmeister2017 [33]	
No-EA	72.18 $\pm$ 13.14	69.27 $\pm$ 12.44	69.99 $\pm$ 11.87	68.93 $\pm$ 12.61	72.46 $\pm$ 12.04	70.48 $\pm$ 12.43
Offline-EA	76.62 $\pm$ 12.38	76.86 $\pm$ 11.82	74.72 $\pm$ 10.97	73.98 $\pm$ 11.21	78.75 $\pm$ 11.80	76.07 $\pm$ 11.66
Offline-RA	76.42 $\pm$ 12.71	76.92 $\pm$ 11.93	74.23 $\pm$ 10.82	74.26 $\pm$ 11.11	77.91 $\pm$ 12.42	76.86 $\pm$ 11.78
Online-EA	74.55 $\pm$ 11.89	<b>75.92 <math>\pm</math> 11.80</b>	<b>73.95 <math>\pm</math> 10.60</b>	72.99 $\pm$ 11.53	<b>77.14 <math>\pm</math> 10.81</b>	<b>74.81 <math>\pm</math> 11.35</b>
Online-RA	75.25 $\pm$ 12.12	75.80 $\pm$ 12.52	72.12 $\pm$ 10.21	73.42 $\pm$ 11.59	75.63 $\pm$ 11.77	74.39 $\pm$ 11.66
No-EA fine-tuning	74.48 $\pm$ 12.35	70.74 $\pm$ 12.64	70.52 $\pm$ 11.13	70.70 $\pm$ 12.10	73.50 $\pm$ 11.99	71.91 $\pm$ 12.07
EA fine-tuning	74.85 $\pm$ 11.60	75.73 $\pm$ 11.40	73.79 $\pm$ 11.21	73.33 $\pm$ 11.35	76.67 $\pm$ 11.12	74.79 $\pm$ 11.31
RA fine-tuning	<b>75.65 <math>\pm</math> 12.29</b>	75.61 $\pm$ 12.08	72.39 $\pm$ 10.26	<b>73.75 <math>\pm</math> 11.41</b>	75.59 $\pm$ 11.80	74.55 $\pm$ 11.56

1. **Bold** for the best model between No-EA fine-tuning, Online-EA, Online-RA, EA fine-tuning, and RA fine-tuning.

resulting in a better average accuracy. Also, comparing EA and RA resulted in similar patterns to those shown in Table 2, with better results for one of the approaches depending on the model or dataset. In the Schirrmeister dataset, the improvements were more evident, both in the median and average accuracies. This better result could be due to the increased number of source subjects to train the model, but we would need further investigation to evaluate this hypothesis.

There was also a significantly faster convergence in the training set in aligned models compared to their non-aligned counterparts, as illustrated in Figure 4. Specifically, within a predefined time window of 100 iterations, the aligned model achieved the same mean accuracy in 70% fewer iterations as the non-aligned model and the same validation loss in just 88% of the iterations.

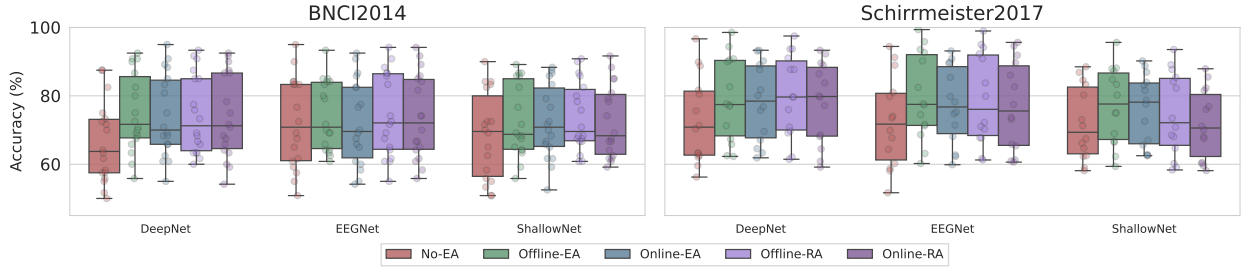


Figure 3: Prediction accuracy using ShallowNet, DeepNet, and EEGNet shared models without alignment, and with EA and RA, in the online and offline scenarios. Left: BNCI2014 dataset, right: Schirrmeister2017 dataset.

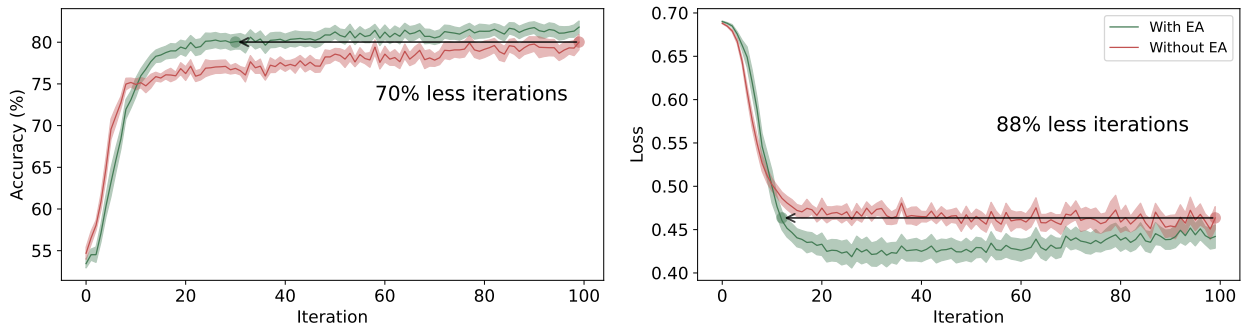


Figure 4: Validation accuracy and loss during training iterations using EEGNet and BCI2014 datasets, with shaded areas depicting the standard deviations over all trained models, one per target subject.

### 3.2 Fine-tuning the shared model

Using the same amount of calibration data for the non-aligned pipeline, performing fine-tuning in the aligned models did not improve the mean accuracy, with some slight increases in accuracies in some datasets or models and decreases in others, as shown in Table 2. This may indicate that applying EA adjusts the domains in a way that, for simple fine-tuning to improve performance, it would need more target data. For the non-aligned case, fine-tuning leads to improvements in accuracy for all datasets and models, with an increase of 1.43% in the general mean accuracy. When

considering the median, there were no changes in all cases. However, for the non-aligned scenarios, the first and third quartiles improved with fine-tuning in most cases, resulting in improved average accuracies.

Algorithm comparison

non-aligned					
fine-tuning-NA	0.80 p=4e-05				
offline-EA	1.25 p=1e-05	1.03 p=6e-05		0.56 p=8e-03	0.52 p=6e-03
online-EA	0.88 p=3e-04	0.64 p=4e-03			0.18 p=1e-01
fine-tuning-EA	0.84 p=5e-04	0.61 p=5e-03			
	non-aligned	fine-tuning-NA	offline-EA	online-EA	fine-tuning-EA

Figure 5: Significance matrix comparing all shared pipelines, with colored cells indicating a significant difference. The algorithms in the x-axis are the baselines, and the values shown correspond to the standardized mean difference between pipelines in the y-axis and the respective baselines and the p-values.

To test the significance of these shared model results, we executed a one-tailed permutation-based paired  $t$ -test (Figure 5), as explained before. Using EA improved the classification in all tested scenarios, with the offline scenario significantly better than all other scenarios. Also, online EA was significantly better than all non-aligned scenarios, while fine-tuning improved the results significantly only for the non-aligned case.

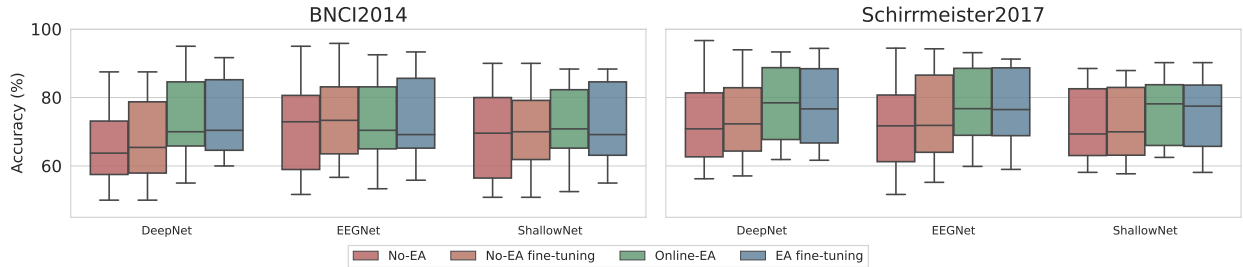


Figure 6: Accuracies of fine-tuned models using ShallowNet, DeepNet, and EEGNet shared models with and without EA.

### 3.3 Analyzing transferability between subjects

We evaluated how EA affects the transferability between subjects, evaluating the prediction accuracy per target and source subject. EA improved the transferability for most subjects, with an increase superior to 10% in the median accuracy in several cases, as shown in Figure 7.

The analysis also indicates that, in general, individuals who are bad donors (sources), i.e., whose models, on average, do not have a good performance on different target subjects, also tend to be bad receivers (targets), and vice versa, as shown in Figure 8. Applying Euclidean Alignment did not change the structure of good/bad donors and receivers but increased the correlation, as can be observed in Table 3. This behavior was expected since, in the aligned case, the data distribution becomes more similar between subjects. If the data from a source subject is useful for the target ones, we should expect the reverse also to occur. Without alignment, data from different subjects are most likely spread in different areas in the domain space.

The transferability information is useful when selecting source individual models for the majority-voting classifiers since it may be more advantageous to choose only the best individuals for a specific target [49].



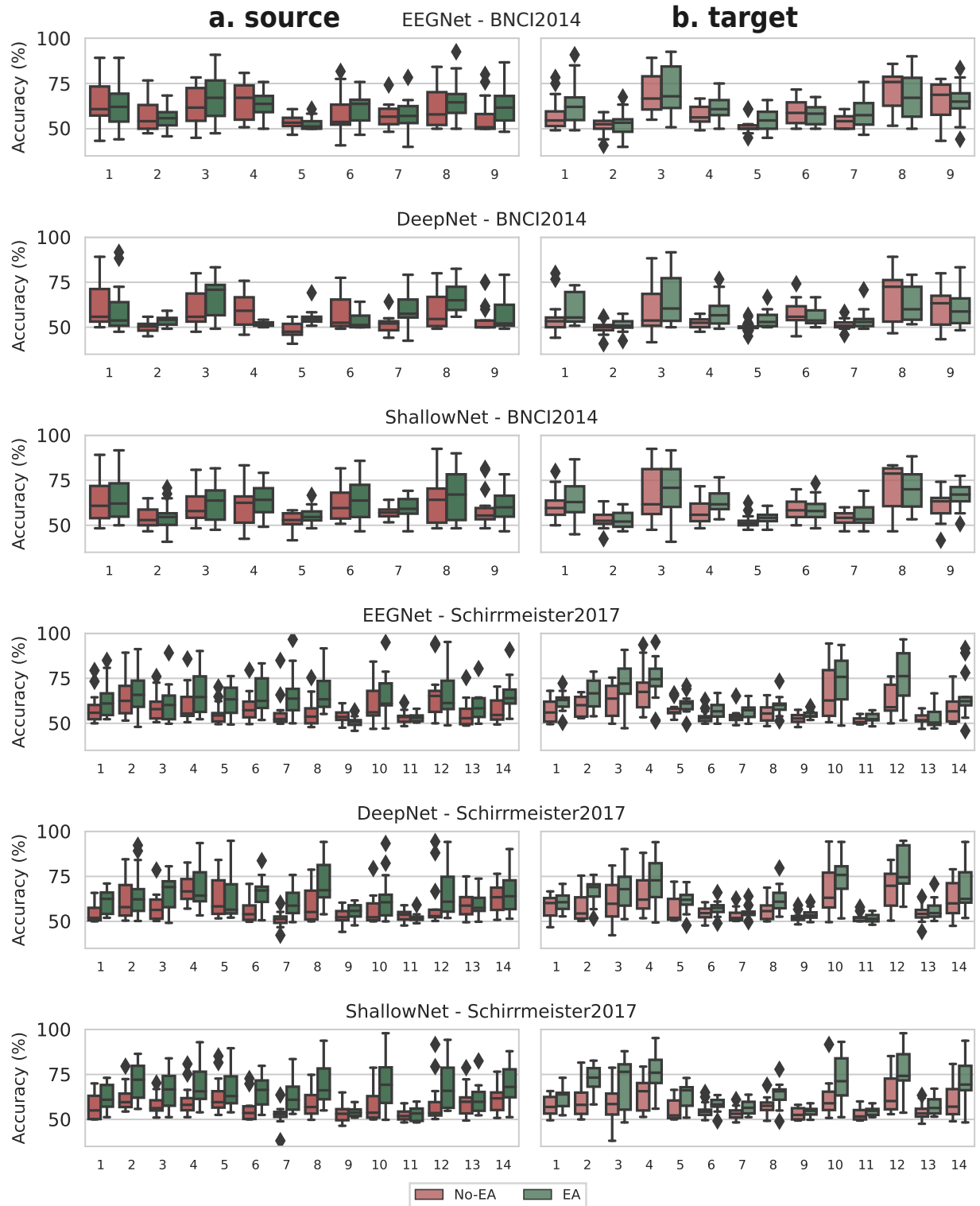


Figure 7: Transferability of trained individual models for different datasets and neural network models. Graphs on the left show the transferability from each source model (a), while graphs on the right (b) show the receivability of target individuals to source models.

Table 3: Mean Pearson correlation coefficient between the average accuracy of the subjects as sources and targets using individual models. The rows represent the models with and without alignment, and the columns indicate the different models and datasets.

Pipelines	Individual Models			Datasets		Overall
	EEGNet [20]	DeepNet [33]	ShallowNet [33]	BNCI2014 [34]	Schirrneister2017 [33]	
No-EA	0.41	0.50	0.49	0.52	0.41	0.47
Offline-EA	0.64	0.61	0.74	0.66	0.66	0.67

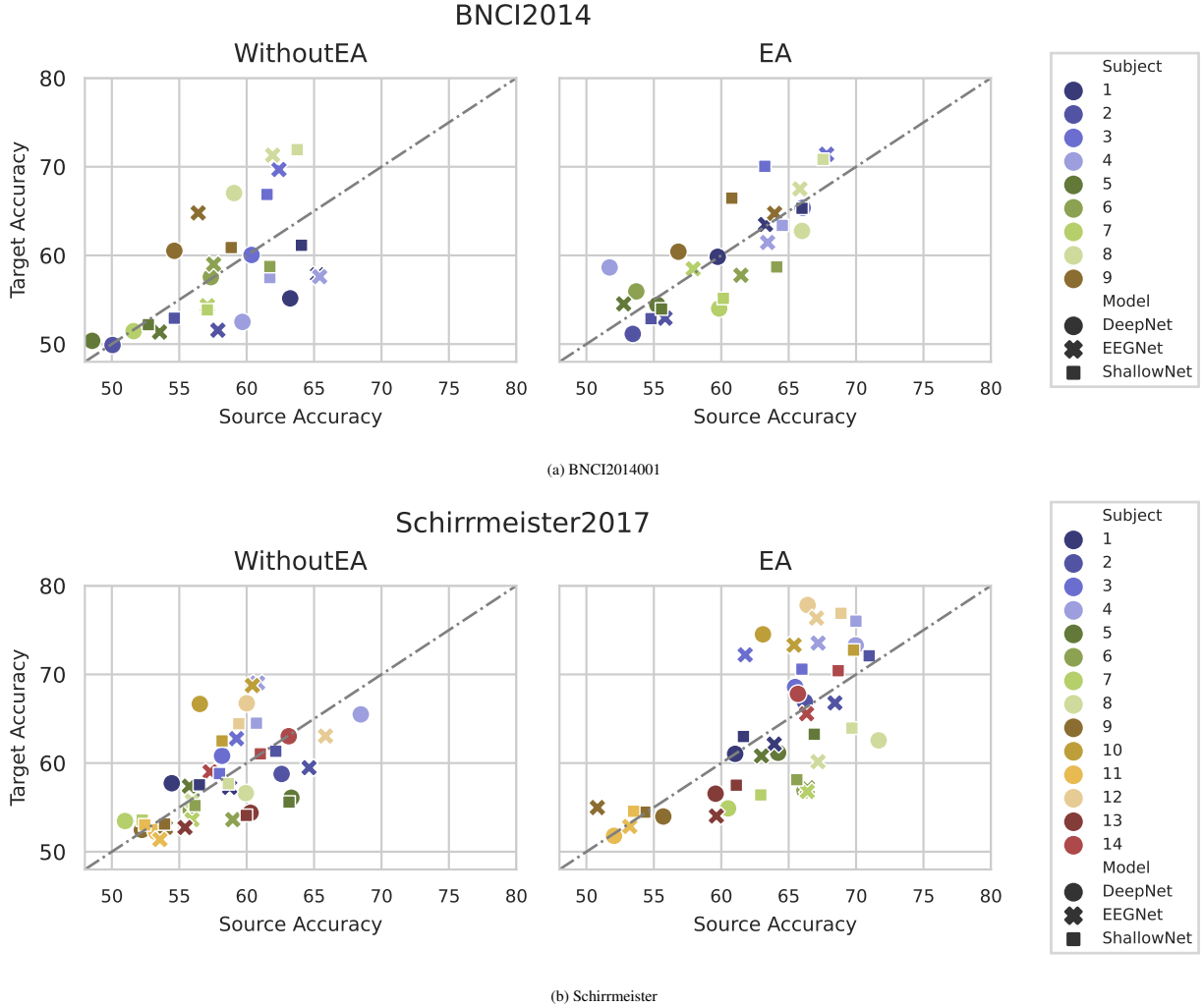


Figure 8: Transferability plots with the relation between average source and average target accuracies for each subject on each model.

### 3.4 Majority-voting classifiers for transfer learning

Given the improvements in transferability, we evaluated the feasibility of using the best-transferred model or a majority voting over the best models to classify signals from the target subjects. We constructed the majority voting classifiers using the same small labeled set of target data from the online pipelines, using the exponential of the accuracy to define the weights from each model.

It is possible to notice from Table 4 that, when training individual models, using EA also resulted in clear benefits, with a 3.71% increase in the overall mean accuracy when comparing the 3-model classifiers with and without EA. For all datasets and neural network models, the best mean accuracies occurred for the 3 and 5-model classifiers with EA.

Moreover, in the non-aligned scenario, increasing the number of voter models did not increase the accuracy significantly compared to using only the best model Figure 9a. When using EA, classifiers with 3 and 5 models significantly improved accuracy compared to using the best model. This is probably because, without the alignment, very few source

models have higher accuracies for each target, while with EA, more models transfer well. Although using the 3-model classifier produced a better mean accuracy than the 5-model classifier, the improvement was not statistically significant. We note that since we use a small number of trials to select the best models, our method will not always get the best models.

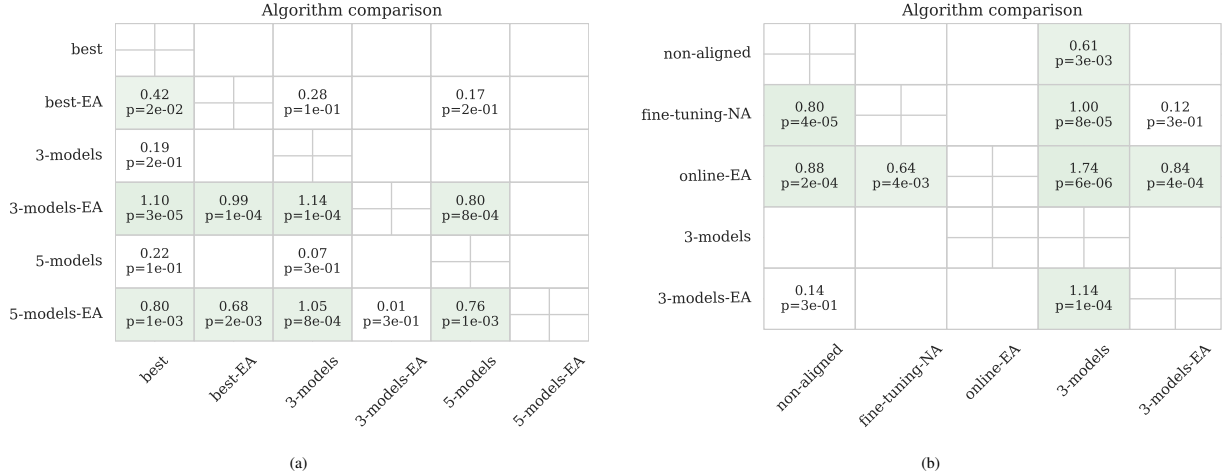


Figure 9: a) Significance matrix comparing all majority-voting pipelines. b) Significance matrix comparing the 3-model classifier and the shared model. Colored cells indicate a significant difference. The algorithms in the x-axis are the baselines, and the values shown correspond to the standardized mean difference between pipelines in the y-axis and the respective baselines and the p-values.

When comparing the majority-voting classifier with the shared model (Figure 10), we can see that the shared model median accuracy is superior in all scenarios, with the biggest improvements when considering the lower quartile. This shows the highest benefit of using the shared model is for the subjects with the worst accuracy.

Finally, the shared model produced significantly better accuracies (Figure 9b) when compared to majority-voting classifiers, with a difference of 3.62% in the mean accuracy to the best majority-voting model. This indicates that training a neural network model using data from all subjects to adjust the weights is advantageous over majority-voting classifiers. On the other hand, it is easier to include more subjects when constructing new majority-voting models than in the shared one since the latter requires retraining the whole model.

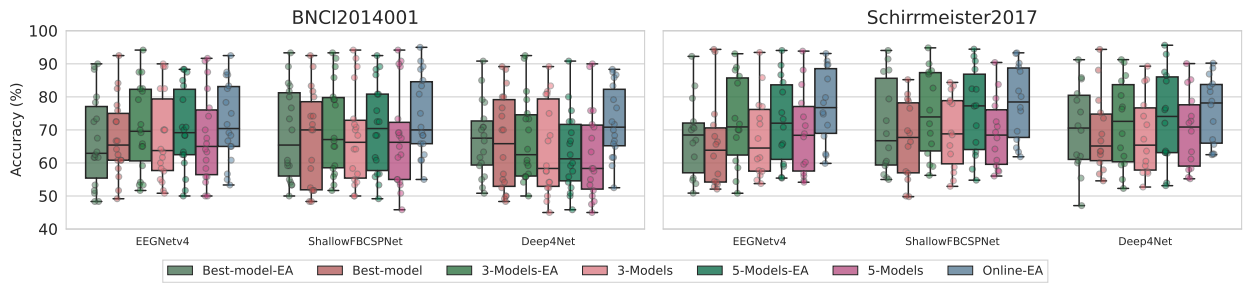


Figure 10: Comparison between aligned and non-aligned majority-voting models test accuracies for the best model, 3 models, and 5 models, compared to the shared model with online EA.

Table 4: Comparison between aligned and non-aligned majority-voting models test accuracies for the best model, 3 models, and 5 models.

Pipelines	Models			Datasets		Overall
	EEGNet [20]	DeepNet [33]	ShallowNet [33]	BNCI2014 [34]	Schirmer2017 [33]	
best-model	66.72 ± 12.60	67.32 ± 13.35	67.19 ± 13.32	66.94 ± 13.47	67.25 ± 12.42	67.08 ± 12.96
3-Models	67.79 ± 12.80	66.05 ± 13.25	68.60 ± 12.97	66.62 ± 13.85	68.59 ± 11.68	67.48 ± 12.92
5-Models	68.33 ± 13.13	65.87 ± 13.18	68.18 ± 13.06	65.99 ± 13.99	69.35 ± 11.57	67.46 ± 13.03
best-model-EA	67.22 ± 12.97	70.74 ± 11.95	69.70 ± 14.10	67.61 ± 12.97	69.74 ± 12.95	68.54 ± 12.94
3-Models-EA	<b>71.60 ± 13.33</b>	<b>75.73 ± 13.19</b>	72.65 ± 13.52	<b>69.58 ± 13.27</b>	73.25 ± 13.18	<b>71.19 ± 13.28</b>
5-Models-EA	71.29 ± 12.62	75.73 ± 13.68	<b>72.85 ± 13.56</b>	68.09 ± 12.91	<b>74.03 ± 13.23</b>	70.69 ± 13.32

1. **Bold** for the best model between aligned versus non-aligned pipelines.

## 4 Discussion

Euclidean Alignment has emerged as a promising technique for domain adaptation in BCI systems, showing good results for transfer learning [13]. Despite its promising results, most works have focused on using classical BCI decoding methods [9, 22], and few works evaluated it in combination with Deep Learning methods [41].

When using single shared models with training data from all individuals, using EA allowed us to achieve the same level of accuracy as the models without EA using 70% fewer iterations and a final level of accuracy 4.33% higher when using the same number of iterations. This faster convergence was not explored in earlier works and probably occurs because we provide data with more similar distributions among individuals. This is a point that can be further explored in future studies.

Existing work using EA with Deep Learning focuses on the offline case, where all data from the target subject is used to determine the EA matrices. For instance, Wu et al. [41] evaluates the use of offline EA in EEGNet and ShallowNet. However, in real applications, calibration trials should be reduced as much as possible to avoid early cognitive fatigue before starting the task. He and Wu [13] evaluated the pseudo-online scenarios for the LDA algorithm and showed that 12% of the target data is sufficient for a good estimation, but using more data improves the accuracy by a few percent. We demonstrated that this data frugality also applies to Deep Learning models. Using data from a single run with 24 trials of each class resulted in an accuracy of only 1.26% below that of the offline case. This occurs because there is less data to determine the distribution of target classes in the latent space, and, unlike the offline case, the target data distribution differs from the data used to train the model. However, the online case is the only one that can be used in real scenarios.

Fine-tuning resulted in no improvements in accuracy for the shared model with EA and a 1.43% increase in accuracy without EA. The small increase without EA possibly occurred because there was only a very small amount of data to fine-tune the model. Using EA, since the models were aligned from the beginning, this small fine-tuning was not sufficient to improve the accuracy.

We also evaluated the scenario with multiple models trained with data from different subjects. One effect noted in other works [31] applying RA is that subjects with good self-scores tend to be good “receivers”. Using EA, we did not evaluate the self-scores, but we noted that subjects are good “donors” tend to be good “receivers” and bad “donors” tend to be bad “receivers.” We did not try to train the shared model using only the “good” donors, but this could be a way to improve the transfer accuracies.

When training individual models, using EA also resulted in clear benefits when selecting only the best source model and majority-voting classifiers of 3 and 5 models. Using weighted majority voting from 3 or 5 models worked better than just using the best model but had a 3.6% reduction in the average accuracy compared to the online shared model. One advantage of the majority-voting models is that adding more source subjects to the collection of source models is easy. We only have to add the new source models to the selection process for the best transferable models for each target subject. One disadvantage is that the inference takes a little longer. However, it should be fast enough for real applications.

Kostas and Rudzicz [19] discusses the concepts of domain generalization (DG), where we adapt the source domains (subjects) to be useful for multiple target domains (subjects), such as when using EA, and domain adaptation (DA), where the domains are adapted to a single target subject. An example of DA would be selecting different source subjects to transfer to each target subject, as done in the majority-voting models. The authors argue that one can apply DG and DA concurrently and can improve transfer learning, as we also verified in our experiments with our majority-voting classifiers.

### 4.1 Limitations

It is important to address some limitations of our study. Firstly, we didn’t explore the impact of different ranges of hyperparameters, such as learning rate, in models with and without EA. For instance, a model with aligned input may accept larger learning rates since there would be less difference between batches. We also used the same hyperparameters for all individual models, but it is well-known that subject-specific hyperparameters can improve accuracy. Other interesting contributions would be to explore the faster convergence of aligned models better and analyze for improvements in the variance of predictions, using, for example, the Monte-Carlo Dropout technique.

One of the approaches employed in this study was to use a weighted majority voting mechanism over the best individual models. However, combining models optimized for each individual is not the best way to generalize knowledge across diverse subjects. We selected this approach because the focus of the evaluation of individual models was to determine if Euclidean Alignment improves transferability between subjects.

## 5 Conclusions

We explored the impacts of the Euclidean Alignment, a domain adaptation technique commonly used on BCI pipelines for Transfer Learning, in the context of Deep Learning. We trained *shared models* using data from multiple subjects and observed an absolute improvement of 5.55% in the model accuracy for the offline case and 4.33% in the pseudo-online case. Moreover, Euclidean Alignment improved the mean accuracy for all cross-subject models and datasets evaluated and led to a 70% acceleration in convergence in the shared models. Consequently, we believe that Euclidean Alignment should be a standard pre-processing step when training cross-subject models.

## Acknowledgments

BJ and RYC are grateful to São Paulo Research Foundation (FAPESP) for the financial support (grants 22/08920-0 and 23/06407-7). The work of BA was supported in part by the CAPES under Grant 001/003 and Data IA mobilité international, Convergence Institute as part of the “Programme d’Investissement d’Avenir” (ANR-17-CONV-0003) operated by LISN.

## References

- [1] Bruno Aristimunha, Igor Carrara, Pierre Guetschel, Sara Sedlar, Pedro Rodrigues, Jan Sosulski, Divyesh Narayanan, Erik Bjareholt, Barthélemy Quentin, Robin Tibor Schirrmeister, Emmanuel Kalunga, Ludovic Darnet, Cattán Gregoire, Ali Abdul Hussain, Ramiro Gatti, Vladislav Goncharenko, Jordy Thielen, Thomas Moreau, Yannick Roy, Vinay Jayaram, Alexandre Barachant, and Sylvain Chevallier. Mother of all BCI Benchmarks, 2023. URL <https://github.com/NeuroTechX/moabb>. version 1.0.0, doi:10.5281/zenodo.10034223.
- [2] Alexandre Bleuzé, Jérémie Mattout, and Marco Congedo. Tangent space alignment: Transfer learning for Brain-Computer Interface. *Frontiers in Human Neuroscience*, pages 16–1049985, 2022.
- [3] Davide Borra and Elisa Magosso. Deep learning-based EEG analysis: investigating P3 ERP components. *Journal of Integrative Neuroscience*, 2021.
- [4] Davide Borra, Elisa Magosso, Miguel Castelo-Branco, and Marco Simões. A Bayesian-optimized design for an interpretable convolutional neural network to decode and analyze the P300 response in autism. *Journal of Neural Engineering*, 2022.
- [5] Davide Borra, Valeria Mondini, Elisa Magosso, and Gernot R. Müller-Putz. Decoding movement kinematics from EEG using an interpretable convolutional neural network. *Computers in Biology and Medicine*, 2023.
- [6] Igor Carrara and Theodore Papadopoulos. Pseudo-online framework for BCI evaluation: a MOABB perspective using various MI and SSVEP datasets. *Journal of Neural Engineering*, 2023.
- [7] Zhanyuan Chang, Congcong Zhang, and Chuanjiang Li. Motor imagery EEG classification based on transfer learning and multi-scale convolution network. *Micromachines*, 13(6), 2022.
- [8] Marco Congedo, Alexandre Barachant, and Rajendra Bhatia. Riemannian geometry for eeg-based brain-computer interfaces; a primer and a review. *Brain-Computer Interfaces*, 2017.
- [9] Orvin Demsy, David Achancaray, and Mitsuhiro Hayashibe. Inter-subject transfer learning using Euclidean alignment and transfer component analysis for motor imagery-based BCI. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 3176–3181, 2021.
- [10] Hauke Dose, Jakob S. Møller, Helle K. Iversen, and Sadasivan Puthusserypady. An end-to-end deep learning approach to MI-EEG signal classification for BCIs. *Expert Systems with Applications*, 114:532–542, 2018. ISSN 0957-4174.
- [11] Amr Farahat, Christoph Reichert, Catherine M. Sweeney-Reed, and Hermann & Hinrichs. Convolutional neural networks for decoding of covert attention focus and saliency maps for EEG feature visualization. *Journal of neural engineering*, 2019.
- [12] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola. A kernel method for the two-sample-problem. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006.
- [13] He He and Dongrui Wu. Transfer learning for brain-computer interfaces: A Euclidean space data alignment approach. *IEEE Transactions on Biomedical Engineering*, 67(2):399–410, 2020.
- [14] Zhiwu Huang and Luc Van Gool. A riemannian network for spd matrix learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

- [15] Emmanuel K Kalunga, Sylvain Chevallier, and Quentin Barthélemy. Transfer learning for SSVEP-based BCI using Riemannian similarities between users. In *2018 26th European Signal Processing Conference (EUSIPCO)*, pages 1685–1689. IEEE, 2018.
- [16] Zahra Khademi, Farideh Ebrahimi, and Hussain Montazery Kordy. A transfer learning-based CNN and LSTM hybrid deep learning model to classify motor imagery EEG signals. *Computers in Biology and Medicine*, 2022.
- [17] Salim Khazem, Sylvain Chevallier, Quentin Barthélemy, Karim Haroun, and Camille Noûs. Minimizing subject-dependent calibration for BCI with Riemannian transfer learning. In *2021 10th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 523–526. IEEE, 2021.
- [18] Jean-Rémi King, Laura Gwilliams, Chris Holdgraf, Jona Sassenhagen, Alexandre Barachant, Denis Engemann, Eric Larson, and Alexandre Gramfort. *Encoding and Decoding Neuronal Dynamics: Methodological Framework to Uncover the Algorithms of Cognition*. 01 2018.
- [19] Demetres Kostas and Frank Rudzicz. Thinker invariance: enabling deep neural networks for BCI across more people. *Journal of Neural Engineering*, 17, 2020.
- [20] Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces. *Journal of Neural Engineering*, 15:056013, 10 2018.
- [21] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.
- [22] Yan Li, Qingguo Wei, Yuebin Chen, and Xichen Zhou. Transfer learning based on hybrid Riemannian and Euclidean space data alignment and subject selection in brain-computer interfaces. *IEEE Access*, 9:6201–6212, 2021.
- [23] Fabien Lotte. Signal Processing Approaches to Minimize or Suppress Calibration Time in Oscillatory Activity-Based Brain–Computer Interfaces. *Proceedings of the IEEE*, 103(6):871–890, 2015.
- [24] Fabien Lotte, Laurent Bougrain, Andrzej Cichocki, Maureen Clerc, Marco Congedo, Alain Rakotomamonjy, and Florian Yger. A review of classification algorithms for EEG-based brain-computer interfaces: A 10-year update. *Journal of Neural Engineering*, 15, 02 2018.
- [25] Zhengqing Miao, Meirong Zhao, Xin Zhang, and Dong Ming. LMDA-Net: A lightweight multi-dimensional attention network for general EEG-based brain-computer interfaces and interpretability. *NeuroImage*, 276:120209, 2023. ISSN 1053-8119.
- [26] Chang S Nam, Anton Nijholt, and Fabien Lotte. *Brain–computer interfaces handbook: technological and theoretical advances*. CRC Press, 2018.
- [27] Yassine El Ouahidi, Lucas Drumetz, Giulia Lioi, Nicolas Farrugia, Bastien Padeloup, and Vincent Gripon. Spatial Graph Signal Interpolation with an Application for Merging BCI Datasets with Various Dimensionalities. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.
- [28] Yassine El Ouahidi, Vincent Gripon, Bastien Padeloup, Ghaith Bouallegue, Nicolas Farrugia, and Giulia Lioi. A strong and simple deep learning baseline for bci mi decoding. *arXiv preprint arXiv:2309.07159*, 2023.
- [29] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.
- [30] Pedro Rodrigues, Christian Jutten, and Marco Congedo. Riemannian Procrustes Analysis: Transfer learning for brain-computer interfaces. *IEEE Transactions on Biomedical Engineering*, PP, 12 2018.
- [31] Pedro Rodrigues, Marco Congedo, and Christian Jutten. Dimensionality transcending: A method for merging BCI datasets with different dimensionalities. *IEEE Transactions on Biomedical Engineering*, 07 2020.
- [32] Yannick Roy, Hubert Banville, Isabela Albuquerque, Alexandre Gramfort, Tiago H Falk, and Jocelyn Faubert. Deep learning-based electroencephalography analysis: a systematic review. *Journal of Neural Engineering*, 16(5): 051001, aug 2019.
- [33] Robin Tibor Schirrmester, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggenberger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for EEG decoding and visualization. *Human Brain Mapping*, 38:5391–5420, 2017.
- [34] Michael Tangermann, Klaus-Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai Miller, Gernot Mueller-Putz, Guido Nolte, Gert Pfurtscheller, Hubert Preissl, Gerwin Schalk, Alois Schlögl, Carmen Vidaurre, Stephan Waldert, and Benjamin Blankertz. Review of the BCI Competition IV. *Frontiers in Neuroscience*, 6, 2012.

- [35] Zitong Wan, Rui Yang, Mengjie Huang, Nianyin Zeng, and Xiaohui Liu. A review on transfer learning in EEG signal analysis. *Neurocomputing*, 421:1–14, 01 2021.
- [36] Xiaoxi Wei, Pablo Ortega, and A. Aldo Faisal. Inter-subject Deep Transfer Learning for Motor Imagery EEG Decoding. In *2021 10th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 21–24, 2021.
- [37] Xiaoxi Wei, A Aldo Faisal, Moritz Grosse-Wentrup, Alexandre Gramfort, Sylvain Chevallier, Vinay Jayaram, Camille Jeunet, Stylianos Bakas, Siegfried Ludwig, Konstantinos Barmas, et al. 2021 BEETL Competition: Advancing Transfer Learning for Subject Independence Heterogenous EEG Data Sets. In *NeurIPS 2021 Competitions and Demonstrations Track*, pages 205–219. PMLR, 2022.
- [38] Jacob White and Sarah D Power. k-Fold Cross-Validation Can Significantly Over-Estimate True Classification Accuracy in Common EEG-Based Passive BCI Experimental Designs: An Empirical Investigation. *Sensors*, 23(13):6077, 2023.
- [39] Daniel Wilson, Robin Tibor Schirrmester, Lukas Alexander Wilhelm Gemein, and Tonio Ball. Deep riemannian networks for eeg decoding, 2023.
- [40] Martin Wimpff, Mario Döbler, and Bin Yang. Calibration-free online test-time adaptation for electroencephalography motor imagery decoding. In *2024 12th International Winter Conference on Brain-Computer Interface (BCI)*, pages 1–6, 2024.
- [41] Dongrui Wu, Xue Jiang, and Ruimin Peng. Transfer learning for motor imagery based brain–computer interfaces: A tutorial. *Neural Networks*, 153(C):235–253, sep 2022.
- [42] Fangzhou Xu, Yunjing Miao, Yanan Sun, Dongju Guo, Jiali Xu, Yuandong Wang, Jincheng Li, Han Li, Gege Dong, Fenqi Rong, Jiancai Leng, and Yang Zhang. A transfer learning framework based on motor imagery rehabilitation for stroke. *Scientific Reports*, 2021.
- [43] Hui Yen Yap, Yun-Huoy Choo, Zeratul Izzah Mohd Yusoh, and Wee How Khoh. An evaluation of transfer learning models in EEG-based authentication. *Brain Informatics*, 10(1):19, Aug 2023. ISSN 2198-4026. doi: 10.1186/s40708-023-00198-4. URL <https://doi.org/10.1186/s40708-023-00198-4>.
- [44] Florian Yger, Maxime Berar, and Fabien Lotte. Riemannian Approaches in Brain-Computer Interfaces: A Review. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1753–1762, 2017.
- [45] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.
- [46] Alberto Zancanaro, Giulia Cisotto, João Ruivo Paulo, Gabriel Pires, and Urbano J. Nunes. CNN-based Approaches For Cross-Subject Classification in Motor Imagery: From the State-of-The-Art to DynamicNet. In *2021 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–7, 2021.
- [47] Paolo Zanini, Marco Congedo, Christian Jutten, Salem Said, and Yannick Berthoumieu. Transfer Learning: A Riemannian Geometry Framework With Applications to Brain–Computer Interfaces. *IEEE Transactions on Biomedical Engineering*, 65(5):1107–1116, 2018.
- [48] Ruilong Zhang, Qun Zong, Liqian Dou, Xinyi Zhao, Yifan Tang, and Zhiyu Li. Hybrid deep neural network using transfer learning for EEG motor imagery decoding. *Biomedical Signal Processing and Control*, 63, 2021.
- [49] Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. Ensembling neural networks: Many could be better than all. *Artificial Intelligence*, 137(1):239–263, 2002.
- [50] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2021. doi: 10.1109/JPROC.2020.3004555.