Classification of BCI-EEG based on augmented covariance matrix

Igor Carrara, Théodore Papadopoulo

CLASSIFICATION OF BCI-EEG BASED ON THE AUGMENTED COVARIANCE MATRIX

Igor Carrara[1] Théodore Papadopoulo[1]
[1] Université Côte d'Azur (UCA), INRIA, Cronos Team, France
igor.carrara@inria.fr and theodore.papadopoulo@inria.fr

Abstract

Objective: Electroencephalography signals are recorded as multidimensional datasets. We propose a new framework based on the augmented covariance that stems from an autoregressive model to improve motor imagery classification. *Methods*: From the autoregressive model can be derived the Yule-Walker equations, which show the emergence of a symmetric positive definite matrix: the augmented covariance matrix. The state-of the art for classifying covariance matrices is based on Riemannian Geometry. A fairly natural idea is therefore to apply this Riemannian Geometry based approach to these augmented covariance matrices. The methodology for creating the augmented covariance matrix shows a natural connection with the delay embedding theorem proposed by Takens for dynamical systems. Such an embedding method is based on the knowledge of two parameters: the delay and the embedding dimension, respectively related to the lag and the order of the autoregressive model. This approach provides new methods to compute the hyper-parameters in addition to standard grid search. *Results*: The augmented covariance matrix performed ACMs better than any state-of-the-art methods. We will test our approach on several datasets and several subjects using the MOABB framework, using both within-session and cross-session evaluation. *Conclusion*: The improvement in results is due to the fact that the augmented covariance matrix incorporates not only spatial but also temporal information. As such, it contains information on the nonlinear components of the signal through the embedding procedure, which allows the leveraging of dynamical systems algorithms. *Significance*: These results extend the concepts and the results of the Riemannian distance based classification algorithm.

Keywords Augmented Covariance, Autoregressive Models, Brain-computer interface, Riemannian geometry, Takens's Theorem.

1 Introduction

Electroencephalography (EEG) is a modality that allows the passive measurement outside of the head of the electrical potential arising mainly from the post-synaptic electrical activity of the brain. The measurements obtained at sensors at time t are linear combinations of the activities at this same time t of a set of electrical "sources" located at the level of the cerebral cortex. These sources are time dependent on each other according to a complex biophysical model that is poorly understood (because it depends on many parameters) and which is therefore not easy to establish. There are, however, simple models which, even if they are not completely justified by biology, are general enough to be used with EEG measurements. An example of those are autoregressive models (AR). These models are based on the hypothesis that the signal can be explained by a combination of its past values plus some random factor called innovation. The AR model has been used for EEG on several occasions and in different ways. For example, they allow to model the interactions between the different zones of the brain during cognitive tasks [1], to extract information from signals [2] or detect state changes [3]. Finally, autoregressive models

are sometimes used in conjunction with machine learning methods for performing classification tasks [4].

One promising application of EEG signal analysis is Brain Computer Interfaces (BCI). BCI can be defined as a technology that measures brain activity and translates that signal into instructions or commands for a digital system. We focus on electroencephalography-based BCI (BCI-EEG), which is non-invasive, has a high temporal resolution and is relatively inexpensive. Because of these characteristics, BCI-EEG interfaces are candidates to becoming the main BCI technology in a mass deployment perspective for everyday use. In particular, we focus on motor imagery BCI paradigms, where the user changes his brain activity by imagining the movement of a body part.

Even though BCI is often considered for medical purposes, the use of this technology has expanded, and found its way into non-medical applications ranging from drone control [5], rehabilitation in neurological diseases [6] to applications in the context of Virtual and Augmented reality [7].

Over the past 10 years, the Riemann distance-based classification paradigm has shown interesting performances for BCI-EEG classification [8]. The key idea behind this method is to map covariance matrices into an appropriate geometrical space: the space of Symmetric Positive Definite (SPD) matrices, which is actually a differentiable manifold with a natural Riemann structure [9]. It is therefore natural to try to extend this Riemannian approach with ideas from auto-regressive models in order to optimize the performance even for the most difficult subjects.

The goal of this work is to improve the performance of the classification algorithms, exploiting the nonlinear components of the EEG signal and combining this approach with Riemann distance-based classifiers. The idea is thus to push further the logic of the Riemannian approach by defining an appropriate SPD matrix that maximizes the amount of information related to a particular task, contained in the EEG signal.

The computation of the AR model is achieved by solving the Yule-Walker equations, which involve a SPD matrix that we call the augmented covariance matrix with lags. Thus, a fairly natural idea is to extend the standard Riemannian approach by using these augmented covariance matrices with lags.

This approach can also be seen from the point of view of dynamical systems: the AR Yule-Walker matrix can be seen as a standard covariance matrix computed on a particular embedding of the original system in a high dimensional space. The delay embedding theorem proposed by Takens [10] in the context of non-linear dynamical systems establishes that such a matrix incorporates the nonlinear properties of the EEG signal provided the embedding dimension is high enough. Most EEG classification algorithms are based on the assumption that the signal can be described through a linear model. However, studies have shown that the EEG signal follows a nonlinear model [11] and because of this nonlinear component, the use of signal analysis techniques based on linearity assumptions could lose important characteristics. Only recently some classification algorithms have been proposed based on the use of quantities defined in the context of nonlinear systems theory such as correlation dimension, Lyapunov exponents, mutual information and the minimum embedding dimension [12, 13]. Most of these non-linear features, however, are extremely computationally demanding.

The basic idea of our approach is to embed the EEG data in a high dimensional space using the theory of non-linear dynamics, and then to classify the EEG signal using the Riemannian approach. In this sense, we are thus combining the two strengths of both approaches, the use of the nonlinear component of the signal and the performance of the Riemannian distance-based algorithm.

This idea of using an augmented covariance matrix with delays was also recently introduced [14] where it was used in combination with a Linear Discriminant Analysis algorithm and applied on Event Related Potential datasets. Another use of the augmented covariance matrix methodology is outlined in a recently published paper [15]. Interestingly, this approach

is specifically tailored for the domain of EEG sleep stage classification.

In order to validate the methods, we apply our approach both with a classification on the Riemann surface using the Minimum Distance to the Mean and another classification on the Tangent Space using a Support Vector Machine (SVM). We will test our approach against several pipelines of the state-of-the-art (both Machine Learning (ML) and Deep Learning (DL)), on several datasets and several subjects using the MOABB framework [16], with both within-session and cross-session evaluation.

The article is structured as follows: in section 2, we first describe the theoretical approach that underlies the model, the datasets considered and the performed statistical analysis. In section 3, we list the results obtained using within-session and cross-session evaluation. Finally, in section 4, we analyze the implications of the method and its current limitations. Section 5 summarizes the results of our study.

2 Materials and Methods

2.1 BCI classification

The EEG signal can be represented as a multivariate time-series $\mathbf{X} \in \mathbb{R}^{d \times T}$ where T is the number of sampled data points and d is the number of electrodes. $\mathbf{X}_t \in \mathbb{R}^{d \times 1}$ is the EEG signal at time t.

This paper tackles the problem of predicting a task achieved by a subject using its EEG signal, which is central in the field of BCI. To do so, the EEG signal is segmented into different short-time windows (epochs) corresponding to a single task. The goal is to predict the actual task from the corresponding EEG measurements.

As an example, this work is validated on motor imagery classification tasks, for which we are interested in predicting whether a subject is thinking about moving e.g. the left or the right hand (see Fig. 1). In this example, we will get some dataset consisting of epochs associated respectively to the left and right hand tasks. This dataset is split to create disjoint training and testing sub-datasets. The training sub-dataset is then used to train a classification algorithm, which is used to predict the outcome on the testing sub-dataset.

2.2 Augmented Covariance

Over the past 10 years, Riemannian approaches have shown interesting performances for SPD (Symmetric Positive Definite) matrix classification [8]. The key idea behind this method is to map the covariance matrix into an appropriate geometrical space: the space of SPD matrices, which is a differentiable manifold with a natural Riemann structure [9]. For more details on Riemannian geometry approach please refer to Appendix A.1.

Riemann distance based classification algorithms are traditionally using the sample spatial covariance matrix for each epoch, which is naturally SPD:

$$\mathrm{Cov}(\mathbf{X}) = \frac{1}{Time - 1} \sum_{i=1}^{Time} \mathbf{X}_i \mathbf{X}_i^T \,, \tag{1}$$

where T denotes the transpose operator. This is not the only option for estimating the covariance, other estimators are possible, but this particular one has the advantage of being unbiased when the number of electrodes is much smaller than the temporal instants contained in an epoch. As the covariance matrix belongs to the SPD space, it is possible to classify the different mental states using the Riemannian framework. Note that the classification can be performed either on the Riemann surface, using algorithms specifically defined on the Riemann manifold such as the Minimum Distance to the Mean (MDM), or by moving manifold points to a tangent space

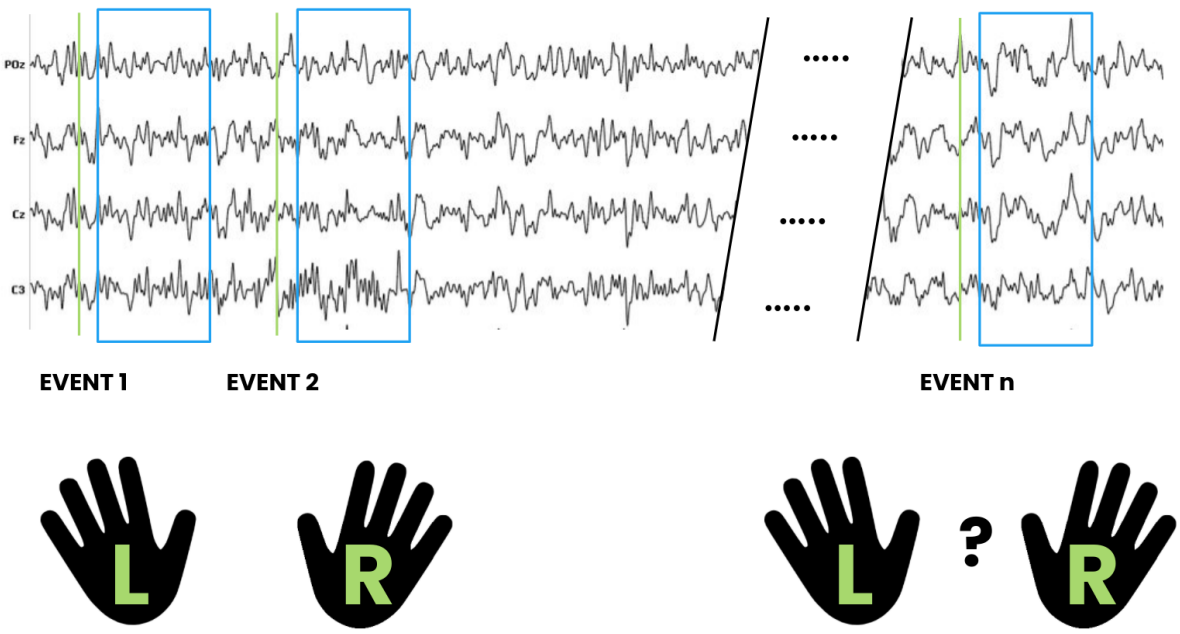


Figure 1: Example of an EEG dataset composed of 4 electrodes. For a motor imagery task for the classification between right and left hand. The experimental paradigm consists in asking the subject to imagine the movement of a body part (left or right hand in this case), at specific times instants marked with green lines. For each green line, an epoch of data (blue boxes) is collected and associated with its left/right hand label. A subset of these epochs forms the training dataset.

and using standard classification algorithms such as Support Vector Machine (SVM) or Logistic Regression [17].

The Augmented Covariance Method (ACM) is an extension of this framework with the introduction of a new SPD matrix. This new SPD matrix stems from the theory of autoregressive models. This approach consists in considering the recorded EEG signal as a realization of a multivariate random process whose law is unknown.

EEG time series are, in general, not stationary: the signal properties change over time according to the underlying mental processes. But for sufficiently short time windows where the mental task does not vary, we can consider the signal as weakly stationary, i.e. its first and second momentum are time invariant.

Under this assumption of weak stationarity, we can model the signal as a random process that follows an autoregressive model: this means that future values are expressed as a function of past ones plus a random component called innovation. Thus, the multivariate EEG signal can be modeled using an autoregressive model of order p [18]

$$\mathbf{X}_t = \sum_{i=1}^{p} \mathbf{A}_i \mathbf{X}_{t-i\tau} + \boldsymbol{\varepsilon}_\mathbf{t} \tag{2}$$

where $\mathbf{A}_i \in \mathbb{R}^{d\times d}$ are the autoregressive coefficients , τ is a parameter called lag and $\boldsymbol{\varepsilon}_t \in \mathbb{R}^{d\times 1}$ is the innovation component.

The order p of the process, the lag as well as the autoregressive coefficients $\mathbf{A}_i$ are unknowns and must be estimated from the signal. The classic approach to estimate the autoregressive coefficients is using the Yule-Walker equation. In order to derive it, we simply post multiply Equation (2) by $\mathbf{X}_{t-k}$ and compute its expectation value. Denoting by $\boldsymbol{\Gamma}(k) = \mathbb{E}(\mathbf{X}_t, \mathbf{X}_{t-k}^T) \in \mathbb{R}^{d\times d}$ the matrix of auto-covariance with lag k and by $\mathbf{U}$ the auto-covariance matrix of the

innovation considered as independent from past values, we get the following equations for $i \in [1, p]$

$$\begin{cases} \mathbf{\Gamma}(\mathbf{0}) = \sum_{k=1}^{p} \mathbf{A}_k \mathbf{\Gamma}(-k\tau) + \mathbf{U} & \text{for } i = 0 \\ \mathbf{\Gamma}(\mathbf{i}) = \sum_{k=1}^{p} \mathbf{A}_k \mathbf{\Gamma}(i - k\tau) & \text{for } i \neq 0 \end{cases} \tag{3}$$

The second Equation of (3) in matrix notation yields

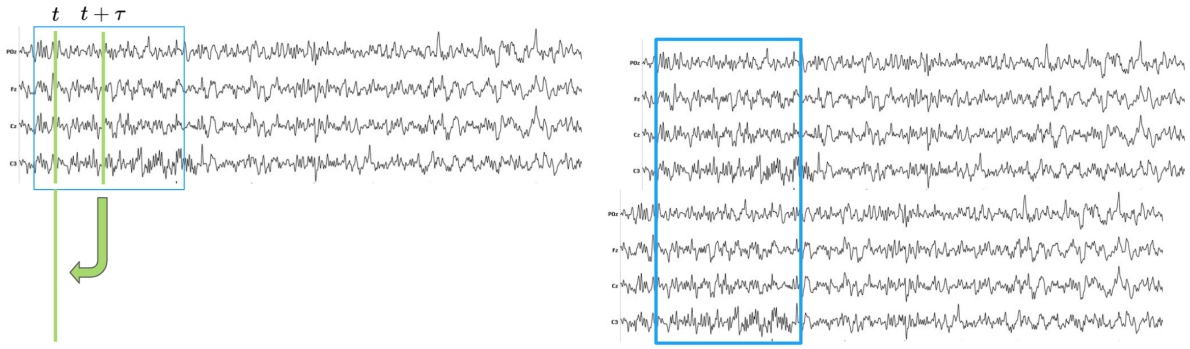
$$\begin{bmatrix} \mathbf{\Gamma}_0 & \mathbf{\Gamma}_{-1} & \mathbf{\Gamma}_{-2} & \cdots \\ \mathbf{\Gamma}_1 & \mathbf{\Gamma}_0 & \mathbf{\Gamma}_{-1} & \cdots \\ \mathbf{\Gamma}_2 & \mathbf{\Gamma}_1 & \mathbf{\Gamma}_0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ \mathbf{\Gamma}_{p-1} & \mathbf{\Gamma}_{p-2} & \mathbf{\Gamma}_{p-3} & \cdots \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \mathbf{A}_3 \\ \vdots \\ \mathbf{A}_p \end{bmatrix} = \begin{bmatrix} \mathbf{\Gamma}_1 \\ \mathbf{\Gamma}_2 \\ \mathbf{\Gamma}_3 \\ \vdots \\ \mathbf{\Gamma}_p \end{bmatrix} \tag{4}$$

The autoregressive coefficients $\mathbf{A}_i$ are thus solutions of a linear system. We call the matrix of this system $\mathbf{\Gamma}_{aug}$ the Augmented Covariance Matrix (ACM):

$$\mathbf{\Gamma}_{Aug} = \begin{bmatrix} \mathbf{\Gamma}_0 & \mathbf{\Gamma}_{-1} & \mathbf{\Gamma}_{-2} & \cdots \\ \mathbf{\Gamma}_1 & \mathbf{\Gamma}_0 & \mathbf{\Gamma}_{-1} & \cdots \\ \mathbf{\Gamma}_2 & \mathbf{\Gamma}_1 & \mathbf{\Gamma}_0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ \mathbf{\Gamma}_{p-1} & \mathbf{\Gamma}_{p-2} & \mathbf{\Gamma}_{p-3} & \cdots \end{bmatrix} \tag{5}$$

$\mathbf{\Gamma}_{aug}$ is symmetric by construction since $\mathbf{\Gamma}_i = \mathbf{\Gamma}_{-i}$. A fairly natural idea is therefore to use these $\mathbf{\Gamma}_{aug}$ matrices for classification algorithms using the Riemannian framework.

$\mathbf{\Gamma}_{aug}$ combines spatial covariance with some temporal information on the signal. $\mathbf{\Gamma}_{aug}$ is exactly the spatial covariance matrix obtained considering an augmented dataset obtained at each time t by considering p consecutive samples (separated by the fixed lag τ). This is an embedding of our original EEG dataset in a dimension equal to $d \times p$, with a fixed delay τ (Fig. 2). Segmenting this new augmented dataset in epochs and computing the epoch covariance matrices results in exactly the same $\mathbf{\Gamma}_{aug}$ matrix.



(a) (b)

Figure 2: (a) Consider the EEG signal with 4 channels and an epoch (blue box), and create a new dataset by concatenating its values at time t with those at time $t + \tau$. (b) Result of the previous procedure at every point of the dataset. This is a higher dimensional dataset for a fixed delay τ and with dimension $d \times p$ (here $d = 4$ and $p = 2$). The augmented covariance matrix of the original signal $\mathbf{\Gamma}_{aug}$ is exactly the standard covariance matrix of this concatenated dataset.

The sample covariance matrix is an efficient and unbiased estimator of the real covariance matrix [19]. In the case of BCI applications, we are in a Finite Observation Large-Dimensional Limit condition (FOLDL) [20]. In the case of $\mathbf{\Gamma}_{aug}$, the number of time sample T is kept fixed while the dimensionality increases since it is multiplied by the embedding dimension. However, in the parameter regime where we use the ACM, the FOLDL limit continues to apply.

2.3 Non Linear Dynamical approach

The formulation of the problem using the augmented dataset shows a natural connection with the Takens delay-embedding theorem [10] in the context of non-linear dynamical systems.

While most EEG classification algorithms are based on the assumption that the signal is describable through a linear theory, recent studies have proven that the EEG signal follows a nonlinear model [11]. Because of this nonlinear component, the use of signal analysis techniques based on linearity assumptions could lose important characteristics, which stresses the importance of considering the theory of dynamical systems.

Let us try to understand the idea behind phase space reconstruction based on Takens's theorem. Usually what we observe in an experiment is not a phase space itself but a time series, most likely only a sequence of scalar measurements, so that not all the dynamic variables involved in the dynamics of the system under study are made available through the measurement process. So, in most cases, a time series can be seen as the values taken by the observed variables of a partially observable dynamical system. It is obvious that, even with a precise knowledge of the measurement process, it may be impossible to reconstruct the phase space of the original system from the data. Fortunately, a reconstruction of the original phase space is not really necessary for data analysis and sometimes not even desirable. It is sufficient to construct a new space that is dynamically equivalent to the original one. The process that we have just enunciated is very well summarized by Fig. 3.

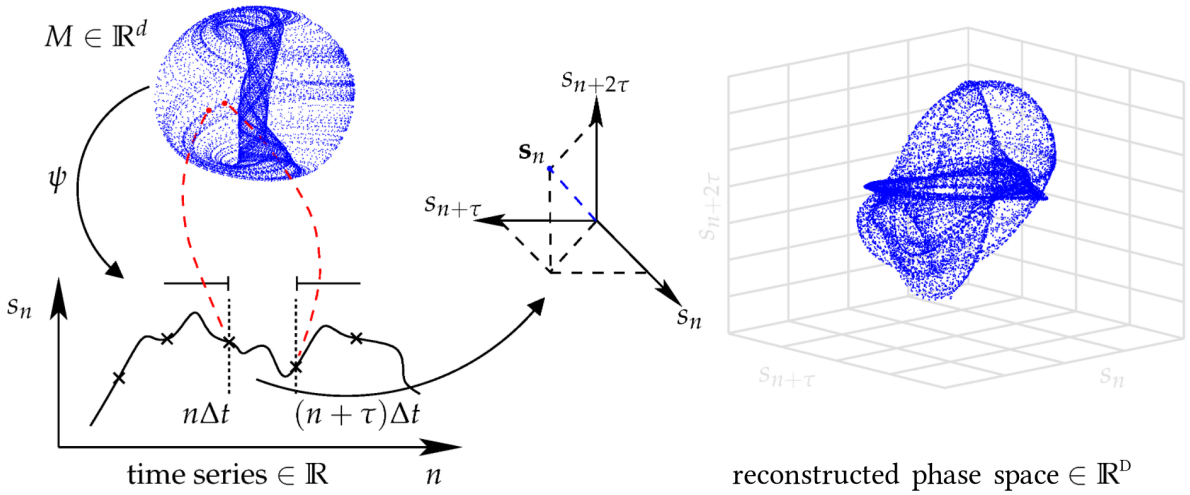


Figure 3: Illustration of the concept of the Delay-Embedding theorem. Reproduced with permission from Ana González-Marcos [21]

.

Considering that a process of measurement create a time series $s(n)$, according to Takens's theorem, the geometric structure of the multi-variable dynamics of the system can be unfolded from the observable $s(n)$ in a D-dimensional space constructed from the delay vector

$$\mathbf{s}_E(n) = [s(n), s(n-\tau), ..., s(n-(D-1)\tau)]^T \tag{6}$$

where τ is a positive integer called the embedding delay and D is the embedding dimension. $\mathbf{s}_E(n) \in \mathbb{R}^D$ is an uniform embedding of the original phase space. As seen previously, the augmented covariance matrix is equivalent to using a standard covariance matrix on an embedding the EEG data in a high dimensional space. The approach is thus related to the theory of nonlinear dynamics but, instead of trying to identify the dynamics's parameters, we classify the mental state using the Riemann framework. In this sense, the two strengths of both approaches are combined: the use of the nonlinear component of the signal and the performance of the Riemann distance-based algorithms. Our approach turns out to be a natural extension of the

Riemann distance-based classification that considers new SPD matrices containing a greater amount of information extracted from the signal based on dynamical systems theory. It is expected that the classification method based on the Riemann framework becomes even more effective by accounting for some non-linearities of the signal measured through the embedding of the data. For the sake of simplicity, we have opted for a uniform embedding approach to minimize the number of hyper-parameters. However, it's important to note that the current framework is highly flexible and can readily accommodate extensions to support non-uniform embeddings.

2.4 Considered Datasets

To test the performance of our method, we used open-access datasets available from the MOABB framework. We consider five motor imagery BCI datasets consisting of several subjects for each dataset and several sessions for each subject. Table 1 contains all the details about those datasets.

Table 1: Dataset considered during this study

Dataset	subjects	channels	sampling rate	sessions	tasks	trials/class	Epoch (s)
BNCI2014001 [22]	9	22	250 Hz	2	4	144	[2, 6]
BNCI2014002 [23]	14	15	512 Hz	1	2	80	[3, 8]
BNCI2014004 [24]	9	3	250 Hz	5	2	360	[3, 7.5]
BNCI2015001 [25]	12	13	512 Hz	3	2	200	[0, 5]
Zhou2016 [26]	3	14	250 Hz	3	3	160	[0, 5]

On all datasets, we apply a standard band-pass filter with range [8; 35] Hz, which is classical for motor imagery tasks [27]. We consider epochs with lengths equal to the duration of task conditions which varies with the datasets under consideration (see last column of Table 1). In these datasets, all classes are balanced.

2.5 Evaluation Procedure

We consider two evaluation paradigms: Within-Session (WS) and Cross-Session (CS). The WS evaluation procedure involves evaluating performance directly within the same session of a certain subject. The classical evaluation method employed in MOABB utilized a 5-fold Cross Validation. When used with grid search hyper-parameter tuning, this was leading to data leakage. To solve this problem, we implemented a Nested Cross Validation [28] in MOABB. In our implementation, the inner loop uses a 3-fold Cross Validation strategy. The CS evaluation procedure, on the other hand, focuses on a single subject. In MOABB, this is done with a Leave One Out Cross Validation, wherein each iteration involves using a single session for testing while utilizing all other sessions for training. Figure 4 provides a schematic explanation of the experimental procedure used in this research for the grid search based algorithms. Note that the MDOP strategy, as it is not based on classification score, does not suffer of data leakage. The hyper-parameter selection is run over each training set of the outer loop and thus does not require an inner loop.

Our model depends on two hyper-parameters, the order of the autoregressive model (equivalently the embedding dimension D_E) and the delay τ. In order to compute these two parameters for every dataset and every subject, we developed two different strategies:

- **Grid Search:**
  In order to find the hyper-parameters, we allow the grid search on two parameters to run on the domain defined by: $D \in [1, 10]$ and $\tau \in [1, 10]$. As this is an exhaustive search, it will always provide the best possible optimum in the defined domain.

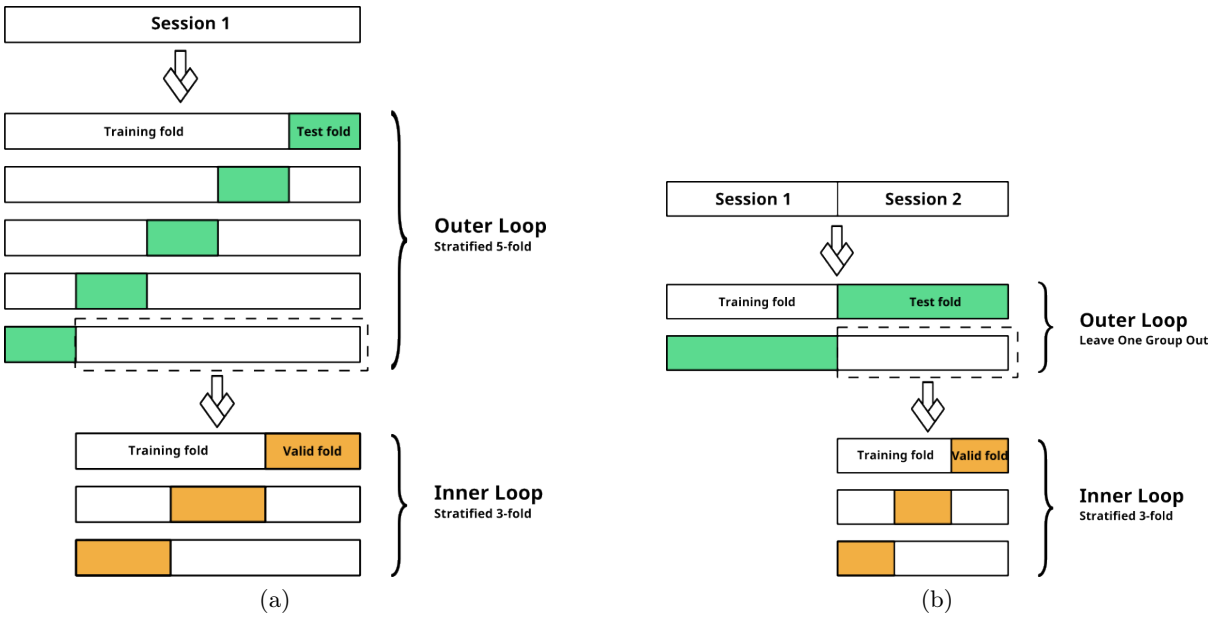


Figure 4: Grid search evaluation procedures. In order to avoid data leakage, a Nested Cross Validation procedure is adopted. Plot (a) shows the within-session evaluation procedure: the outer loop uses a stratified 5-fold CV, while the inner loop, in charge of tuning the model's hyper-parameters, uses a stratified 3-fold CV. Plot (b) shows the cross-session evaluation procedure, which closely resembles the within-session one, with the key distinction being that the outer loop adopts a leave-one-session-out CV strategy for testing.

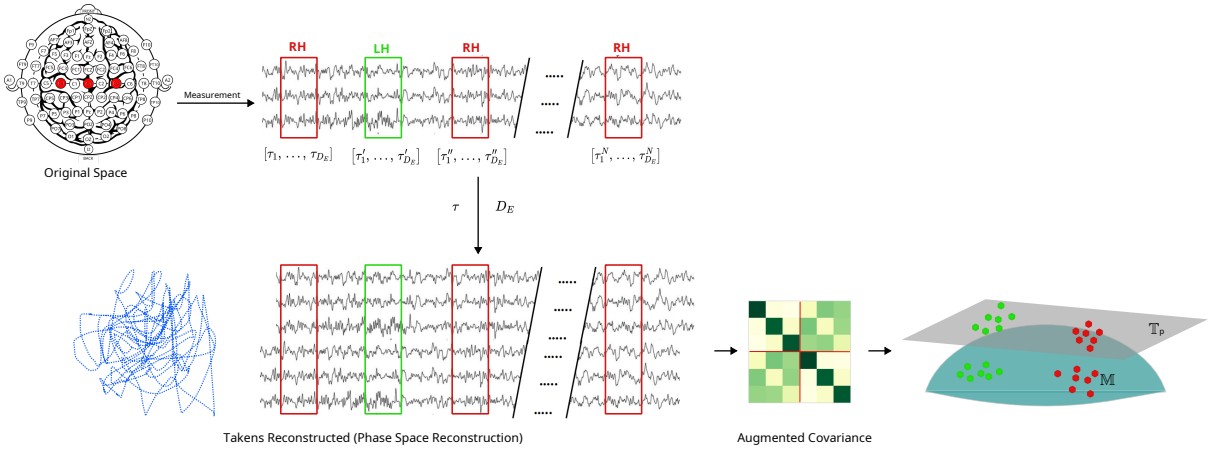


Figure 5: The figure schematically illustrates the ACM methodology. The presented example uses the BNCI2014004 dataset that has only 3 electrodes (in red on the top left plot). The measurement process of the original dynamic system is this a 3-dimensional time series. The process then begins with the extraction of epoch signal representing left and right hand tasks. The selection of hyper-parameters can be made via grid search using the nested approach or employing MDOP algorithm. Specifically, the MDOP methodology selects different sets of delays for each embedding and for each epoch and then obtains a unique value for delay and embedding dimension using an average across delays and epochs. Once these parameters are selected, we use the phase space reconstruction process to obtain a dynamic system equivalent to the original one (in this figure, we see an embedding corresponding to $D_E = 2$ and $\tau = 10$). From this high-dimensional space, the covariance matrix is extracted and classified using algorithms on the Riemannian manifold (MDM) or on the Tangent space (SVM) through projection.

- **Non Linear Dynamical approach:**
  The parameters D and τ can also be estimated using the concepts of non-linear dynamical systems. There are several algorithms for estimating optimal parameters in noisy conditions that can be grouped in two categories: traditional and unified approaches. In this research, we focused on unified approaches that were implemented using the DynamicalSystem.jl library [29]. Specifically, we used the MDOP algorithm [30], which works iteratively by finding simultaneously the best delay and embedding dimensions. The algorithm starts with the current dataset and adds a lagged version of that dataset to it by finding the correct value of τ such that the redundancy between coordinates is minimized, i.e. by maximizing the beta statistics [30]. After each embedding cycle the FNN-statistic [31] is being checked and as soon as this statistic drops below a threshold, the algorithm terminates. It should be noted that this algorithm produces different values of τ for each embedding cycle. We decided to select the optimal value of τ as an average of these values, while we select D_E as the number of cycles. This decision is rooted in our methodology's reliance on uniform embedding.

2.6 Studied Pipelines

We intend to evaluate how the augmented covariance method (ACM) as a general technique can increase the performance in different settings. We used the ACM with several Riemannian classification methods on the surface or on the tangent space as shown in Table 2. These pipelines will be denoted by the blue color in the result tables: a lighter blue is used to distinguish those working directly on the Riemannian manifold from those working in the tangent space. A visual representation of the methodology employed in this paper is available in Fig. 5.

Table 2: Considered pipelines with classifications on the Riemann surface or the Tangent Space.

Surface	**Standard algorithm**	**Augmented algorithm**	
		Grid Search	**Unified Approach**
Riemann Surface	MDM	ACM+MDM (Grid)	ACM+MDM (MDOP)
Tangent Space	TGSP+SVM	ACM+TGSP+SVM (Grid)	ACM+TGSP+SVM (MDOP)

We compared our pipelines against eight state-of-the-art algorithms of the BCI literature:

- A reference method (colored in red in the result tables):

  1. "CSP+LDA", a combination of the Common spatial pattern (CSP) algorithm followed by a classification performed on a shrinkage Linear Discriminant Analysis (LDA) [32].

- Four Riemannian methods (colored in yellow in result tables):

  2. "FgMDM", a classification method by Minimum Distance to the Mean after having applied a geodesic filtering [33].
  3. "MDM", a classification method by Minimum Distance to the Mean [8].
  4. "COV+EN", a classification method in the tangent space using an Elastic Network as classifier [34].
  5. "TGSP+SVM", a classification method in the tangent space using a Support Vector Machine [8].

- Three Deep Learning (DL) methods (colored in green in result tables):

  6. "EEGNet", a DL architecture based on CNN, with standardized and re-sampled EEG signal at 128Hz [35].

7. "DeepConvNet", a DL architecture based on CNN, with standardized and re-sampled EEG signal at 250Hz [36].
8. "ShallowConvNet", a DL architecture based on convolutional layer (CNN), with standardized and re-sampled EEG signal at 250Hz [36].

The pipeline 1 is the most basic pipeline and is given for reference. For pipelines 2–5, which are based on Riemannian geometry, all the hyper-parameters of the classifiers used in this study are optimized with grid search. Table A1 summarizes the domains over which these grid searches are performed.

The DL pipelines 6–8 were implemented using the TensorFlow [37] and SciKeras libraries. To enhance result stability, each DL pipeline starts with a standardization step, which normalizes every channel to have a zero mean and unit standard deviation. Additionally, we employed a re-sampling procedure to ensure that each architecture integrates a temporal filter aligned with the state-of-the-art techniques' implementations. For specific details on the DL hyper-parameters, please refer to Table A2.

2.7 Statistical Analysis

In order to validate the best pipeline from a statistical point of view, we use the instrument provided by the MOABB framework. To remain fast, it uses a one-tailed permutation-based paired t-test [38] for datasets with less than 20 subjects, or a Wilcoxon non-parametric signed-rank test [39] otherwise. This statistical test is used to generate a p-value that compares two pipelines for each pair of pipelines. Then, the p-values are combined using Stouffer's method [40] in order to get a final p-value for each hypothesis. In this way, we can validate the results using several subjects and several datasets.

The score are obtained using the Area Under Curve (AUC) of the Receiver Operating Characteristic (ROC) curve. AUC ranges in value from 0 to 1. A higher AUC value characterizes a better classifier. In case of a multi-class classification, the metric used is the accuracy.

3 Results

In this section, we report the results of the augmented covariance method over the classification on both the Riemann surface and the Tangent Space. To validate the robustness and validity of our approach, we test the algorithm on different datasets, subjects and tasks using the MOABB framework.

3.1 Right hand vs Left hand

We consider 3 different datasets – BNCI2014001, BNCI2014004 and Zhou2016 – to classify a right vs left hand task classification using within-session and cross-session evaluation procedures.

The various approaches of this study are compared in Table 3. A detailed picture of the statistical significance of these results is provided in Fig. 6 and A2. Out of the three datasets, Zhou2016 is generally giving good results with all methods. For this dataset, the best result is obtained with ShallowConvNet but is followed very closely by our best approach – ACM+TGSP+SVM (Grid) – with a non-significant decrease of 1% in the AUC score. With the other datasets, ACM+TGSP+SVM (Grid) brings a notable improvement over all other non-ACM methods with an average improvement of AUC by 4-16%. The improvements are very similar with both the WS and CS validation modes. Except in one case, all ACM methods improve on their non-ACM versions. The exceptional case is ACM+TGSP+SVM (MDOP) vs TGSP+SVM for the BNCI2014001 dataset with WS evaluation. The small regression (non significant) can be attributed to a failure of the MDOP algorithm to find optimal hyper-parameter values as the ACM+TGSP+SVM (Grid) clearly performs well in the same benchmark.

Table 3: Performance (AUC) of Right hand vs Left hand classification. The color code refers to the type of method (see section 2.6).

Pipeline	Eval	BNCI2014001	BNCI2014004	Zhou2016
ACM+MDM (Grid)	WS	0.89 ± 0.13	$\mathbf{0.82 \pm 0.16}$	0.91 ± 0.07
ACM+MDM (MDOP)	WS	0.86 ± 0.14	0.81 ± 0.16	0.91 ± 0.08
ACM+TGSP+SVM (Grid)	WS	$\mathbf{0.92 \pm 0.11}$	$\mathbf{0.82 \pm 0.16}$	0.94 ± 0.06
ACM+TGSP+SVM (MDOP)	WS	0.87 ± 0.13	$\mathbf{0.82 \pm 0.16}$	0.93 ± 0.06
CSP+LDA	WS	0.84 ± 0.15	0.79 ± 0.16	0.91 ± 0.10
FgMDM	WS	0.88 ± 0.11	0.79 ± 0.15	0.91 ± 0.08
MDM	WS	0.84 ± 0.14	0.78 ± 0.16	0.89 ± 0.08
Cov+EN	WS	0.88 ± 0.12	0.80 ± 0.16	0.93 ± 0.08
TGSP+SVM	WS	0.88 ± 0.12	0.79 ± 0.15	0.92 ± 0.08
EEGNet	WS	0.76 ± 0.21	0.69 ± 0.20	0.94 ± 0.02
DeepConvNet	WS	0.82 ± 0.17	0.72 ± 0.19	0.93 ± 0.07
ShallowConvNet	WS	0.85 ± 0.15	0.72 ± 0.18	$\mathbf{0.95 \pm 0.06}$
ACM+MDM (Grid)	CS	0.89 ± 0.13	0.83 ± 0.15	0.94 ± 0.04
ACM+MDM (MDOP)	CS	0.86 ± 0.13	0.81 ± 0.15	0.93 ± 0.04
ACM+TGSP+SVM (Grid)	CS	$\mathbf{0.91 \pm 0.10}$	$\mathbf{0.83 \pm 0.15}$	0.93 ± 0.06
ACM+TGSP+SVM (MDOP)	CS	0.87 ± 0.12	0.81 ± 0.15	0.94 ± 0.05
CSP+LDA	CS	0.84 ± 0.15	0.81 ± 0.14	0.92 ± 0.08
FgMDM	CS	0.85 ± 0.14	0.80 ± 0.14	0.90 ± 0.09
MDM	CS	0.83 ± 0.15	0.79 ± 0.15	0.92 ± 0.05
Cov+EN	CS	0.86 ± 0.13	0.81 ± 0.14	0.93 ± 0.07
TGSP+SVM	CS	0.86 ± 0.14	0.80 ± 0.14	0.91 ± 0.07
EEGNet	CS	0.75 ± 0.21	0.75 ± 0.17	$\mathbf{0.95 \pm 0.05}$
DeepConvNet	CS	0.82 ± 0.17	0.75 ± 0.17	0.94 ± 0.07
ShallowConvNet	CS	0.87 ± 0.12	0.73 ± 0.14	0.94 ± 0.06

3.2 Right hand vs Feet

This classification task was performed on 4 different datasets - BNCI2014001, BNCI2014002, BNCI2015001 and Zhou2016 - to perform a right hand vs feet task classification using a within-session and cross-session evaluation procedure. For sake of simplicity, we do not report the results for left hand vs feet task as they are very similar. There are no CS results for BNCI2014002 because this dataset is composed by only one session.

The various approaches of this study are compared in Table 4. A detailed picture of the statistical significance of these results is provided in Fig. A3 and A4. Generally speaking, the results are better than with the Left hand - Right hand case, showing that the Right hand - Feet task is generally the easiest of the two. Our best approach – ACM+TANG+SVM (Grid) – brings 0-12% of improvements over all other methods. The only method that gives some competitive results is again ShallowConvNet on the Zhou2016 dataset, which, as in the previous case, generally gives good results with all methods. Except in three cases, all ACM methods improve on their non-ACM counterparts. The exceptional cases again arise for ACM+TGSP+SVM (MDOP) vs TGSP+SVM. The MDOP algorithm seems to have more difficulty to find optimal hyper-parameter values with this specific method.

Table 4: Performance (AUC) of Right hand vs Feet classification. The color code refers to the type of method (see section 2.6).

Pipeline	Eval	BNCI2014001	BNCI2015001	Zhou2016	BNCI2014002
ACM+MDM (Grid)	WS	0.96 ± 0.05	0.90 ± 0.11	0.94 ± 0.04	0.82 ± 0.15
ACM+MDM (MDOP)	WS	0.93 ± 0.08	0.89 ± 0.12	0.94 ± 0.05	0.78 ± 0.17
ACM+TGSP+SVM (Grid)	WS	$\mathbf{0.97 \pm 0.03}$	$\mathbf{0.93 \pm 0.08}$	$\mathbf{0.98 \pm 0.02}$	$\mathbf{0.89 \pm 0.11}$
ACM+TGSP+SVM (MDOP)	WS	0.94 ± 0.08	0.92 ± 0.09	0.97 ± 0.03	0.83 ± 0.16
CSP+LDA	WS	0.92 ± 0.10	0.89 ± 0.11	0.94 ± 0.04	0.85 ± 0.13
FgMDM	WS	0.94 ± 0.08	0.90 ± 0.10	0.96 ± 0.03	0.85 ± 0.12
MDM	WS	0.91 ± 0.10	0.86 ± 0.13	0.92 ± 0.05	0.77 ± 0.15
Cov+EN	WS	0.95 ± 0.07	0.91 ± 0.09	0.97 ± 0.03	0.86 ± 0.12
TGSP+SVM	WS	0.95 ± 0.07	0.91 ± 0.09	0.96 ± 0.03	0.86 ± 0.12
EEGNet	WS	0.89 ± 0.15	0.90 ± 0.12	0.95 ± 0.03	0.84 ± 0.17
DeepConvNet	WS	0.88 ± 0.13	0.88 ± 0.14	0.96 ± 0.04	0.88 ± 0.11
ShallowConvNet	WS	0.92 ± 0.10	0.91 ± 0.11	$\mathbf{0.98 \pm 0.02}$	0.88 ± 0.12
ACM+MDM (Grid)	CS	0.94 ± 0.06	0.91 ± 0.09	0.95 ± 0.04	
ACM+MDM (MDOP)	CS	0.92 ± 0.09	0.90 ± 0.10	0.95 ± 0.05	
ACM+TGSP+SVM (Grid)	CS	$\mathbf{0.96 \pm 0.05}$	$\mathbf{0.93 \pm 0.07}$	$\mathbf{0.99 \pm 0.01}$	
ACM+TGSP+SVM (MDOP)	CS	0.93 ± 0.07	0.92 ± 0.09	0.97 ± 0.04	
CSP+LDA	CS	0.89 ± 0.14	0.88 ± 0.11	0.94 ± 0.05	
FgMDM	CS	0.93 ± 0.08	0.90 ± 0.10	0.95 ± 0.06	
MDM	CS	0.90 ± 0.11	0.87 ± 0.12	0.93 ± 0.06	
Cov+EN	CS	0.94 ± 0.07	0.90 ± 0.10	0.96 ± 0.04	
TGSP+SVM	CS	0.94 ± 0.07	0.88 ± 0.12	0.96 ± 0.05	
EEGNet	CS	0.87 ± 0.16	0.90 ± 0.12	0.98 ± 0.02	
DeepConvNet	CS	0.86 ± 0.16	0.90 ± 0.12	0.97 ± 0.03	
ShallowConvNet	CS	0.90 ± 0.12	0.92 ± 0.10	$\mathbf{0.99 \pm 0.01}$	

3.3 Right hand vs Left hand vs Feet

In this last case, we use 2 different datasets (BNCI2014001 and Zhou2016) to perform a three tasks classification – right hand vs left hand vs feet – using the within-session and cross-session evaluation procedures.

The various approaches considered in this study are compared in Table 5. A detailed picture of the statistical significance of these results is provided in Fig. A5 and A6. With the BNCI2014001 dataset, ACM+TGSP+SVM (Grid) brings a notable performance improvement with an average improvement of Accuracy of 6-12% over all non DL methods and of 0-37% with respect to DL methods. DeepConvNet seems to perform particularly bad on this benchmark. Removing it gives improvements of 0-17% over DL methods. ShallowConvNet is competitive for the CS evaluation method. For the Zhou2016 dataset, the best method is ShallowConvNet. The ACM+TGSP+SVM (Grid) method is a close second with a performance loss of 2-8%. This dataset appears to be noticeably more challenging for 3 tasks classification than for binary classification. As for the other tasks, ACM based methods improve on their non-ACM counterparts, with exceptions arising only with the ACM+TGSP+SVM (MDOP) algorithm.

4 Discussion

The numerous tests on different classification tasks, both binary and multi class, have shown some notable observations that we summarize here.

Table 5: Performance (Accuracy) of Right hand vs Left hand vs Feet classification. The color code refers to the type of method (see section 2.6).

Pipeline	Eval	BNCI2014001	Zhou2016
ACM+MDM (Grid)	WS	0.78 ± 0.12	0.79 ± 0.04
ACM+MDM (MDOP)	WS	0.76 ± 0.14	0.79 ± 0.06
ACM+TGSP+SVM (Grid)	WS	$\mathbf{0.84 \pm 0.11}$	0.84 ± 0.04
ACM+TGSP+SVM (MDOP)	WS	0.78 ± 0.15	0.82 ± 0.04
CSP+LDA	WS	0.74 ± 0.16	0.81 ± 0.06
FgMDM	WS	0.77 ± 0.14	0.81 ± 0.06
MDM	WS	0.72 ± 0.15	0.75 ± 0.07
Cov+EN	WS	0.79 ± 0.14	0.83 ± 0.06
TGSP+SVM	WS	0.78 ± 0.14	0.83 ± 0.05
EEGNet	WS	0.67 ± 0.20	0.84 ± 0.04
DeepConvNet	WS	0.47 ± 0.08	0.56 ± 0.06
ShallowConvNet	WS	0.76 ± 0.16	$\mathbf{0.86 \pm 0.04}$
ACM+MDM (Grid)	CS	0.73 ± 0.12	0.72 ± 0.08
ACM+MDM (MDOP)	CS	0.70 ± 0.13	0.72 ± 0.09
ACM+TGSP+SVM (Grid)	CS	$\mathbf{0.76 \pm 0.11}$	0.78 ± 0.06
ACM+TGSP+SVM (MDOP)	CS	0.71 ± 0.14	0.76 ± 0.10
CSP+LDA	CS	0.66 ± 0.16	0.73 ± 0.13
FgMDM	CS	0.69 ± 0.13	0.72 ± 0.10
MDM	CS	0.66 ± 0.13	0.70 ± 0.09
Cov+EN	CS	0.70 ± 0.12	0.74 ± 0.10
TGSP+SVM	CS	0.67 ± 0.15	0.73 ± 0.10
EEGNet	CS	0.66 ± 0.21	0.84 ± 0.06
DeepConvNet	CS	0.41 ± 0.10	0.68 ± 0.12
ShallowConvNet	CS	$\mathbf{0.76 \pm 0.16}$	$\mathbf{0.86 \pm 0.05}$

4.1 Tangent Space classification outperforms the manifold method, ACM enhances performance in both scenarios.

In general, with either ACM or non ACM approaches, the Riemannian method based on tangent space classification is more effective than Riemann surface classification approaches (as shown in Tables 3, 4 and 5). ACM-based methods consistently demonstrate enhancements over their standard covariance matrix counterparts, excelling in both Riemannian manifold and tangent space classification. This trend is more marked with the grid search hyper-parameters estimation than with that based on MDOP, which sometimes seems to have trouble to find a good set of hyper-parameters. This is counterbalanced by the fact that MDOP is generally faster than grid search, while often still providing improved results.

4.2 ACM improves performance in both WS and CS evaluation

The utilization of the ACM methodology enhances performance in both the WS and CS cases, demonstrating its stability and applicability in the more complex CS situations, which involve more variability. This is important for clinical or real-world scenarios, for which CS is a more realistic condition. This finding shows that the ACM methodology brings for some new and

robust information to the classifiers, which helps then in handling the variability of EEG signals.

4.3 ACM+TGSP+SVM (Grid) outperforms all DL pipelines. ShallowConvNet is sometimes competitive

Globally, deep learning methods perform less favorably than the non DL ACM ones. One exception is ShallowConvNet that sometimes is better than the best ACM method ACM+TGSP+SVM (Grid). However, the improvements it provides are never statistically significant. This success can be attributed to the specific tailored feature created using the ACM compared to the automatic feature extraction achieved by DL models.

4.4 hyper-parameter stability

The performance improvements are naturally less important when the baseline methods are performing better initially, but this definitely shows that the augmented covariance approach brings some valuable information into the classification problem for these datasets. The selection of hyper-parameters that produces the best performance is through the use of grid search, however, the modality based on non-linear dynamics concepts (MDOP) is also shown to produce in general better performance than standard algorithms (with a few exceptions).

However, the selection of the hyper-parameters through grid search is time consuming in the training phase. Generally speaking, hyper-parameters values are not stable across subjects and sessions. Fig. 7 (a) shows one example of the spread of the order and lag hyper-parameters over subjects and sessions for BNCI2014001. Comparatively, it can be seen that the MDOP unified approach gives more condensed point sets. Yet, this method based on nonlinear systems theory places an important theoretical foundation for finding efficiently proper hyper-parameters which generally achieve performance improvements for both WS and CS evaluations, even if far from optimal. Fig. 7 (b) shows the set of parameters that provide an accuracy within 99% of the best one. On this example, we see that there are numerous almost equivalent sets of hyper-parameters. This shows that focusing on the actual optimal value is maybe too strict. Fig. 7 (c) is a voting map which counts for each couple of hyper-parameters when they are found in this 99% region over subjects $\times$ sessions. The distribution of optimal hyper-parameters is not unimodal demonstrating that finding fixed values for these hyper-parameters that work in every situation is difficult. Other classification tasks and datasets exhibit the same behavior.

4.5 Computational time

In this section, we report the computational times of the proposed algorithms with respect to their scores. To allow a fair comparison, we have run all the algorithms on the same hardware, a Dell C6420 dual-Xeon Cascade Lake SP Gold 6240 @ 2.60GHz. The results for the BNCI2014001 dataset for the right hand vs left hand classification with WS evaluation procedure are shown in Fig. 8. The MDOP algorithms (ACM+MDM (MDOP), ACM+TGSP+SVM (MDOP)) exhibit a similar speed as ACM+MDM (Grid) while delivering slightly lower performance. On the other hand, the ACM+TANG+SVM (Grid) algorithm outshines in terms of performance but comes at the cost of using substantial computational resources. This increased computational load primarily arises from the necessity to explore a larger parameter space during grid search optimization, particularly due to the SVM component. Overall, the best trade off between algorithmic performance and computational efficiency is provided by ACM+MDM (Grid). Alternatively, opting for hyper-parameter selection through MDOP can be advantageous, especially when the classification algorithm itself requires extensive computational resources.

Finally, note that the augmented methods increase computational time because of the increased dimensionality of the augmented covariance compared to the standard covariance [41], as shown in Fig. 9 for the BNCI2014001 dataset for the right hand vs feet classification with WS

evaluation procedure. In this graph we show the computational performance of ACM+MDM without the grid search hyper-parameter selection but setting specifically the order. Yet, even in the higher orders, the total classification time of a new sample with the augmented approaches is compatible with real time computations.

4.6 Limitations

While our approach has been shown to achieve improvement in performance in many BCI applications in the Motor Imagery task, it has at least one limitation : as other methods, it relies on the assumption of the validity of the sample covariance estimator. When the span of the used time windows are way greater than the dimension of covariance matrix, this assumption remains valid. But this dimension increases quickly with higher AR orders. In that case, the matrix is still a symmetric positive matrix but is not definite and Riemannian metrics can no longer be used. To solve this issue, it is possible to adopt shrinkage or other methods to estimate covariance [42, 43, 44].

5 Conclusion

In this research, we explored the use of the augmented covariance matrix for the classification of Motor Imagery task in EEG-BCI. This methodology amounts to integrate temporal information into the normal spatial covariance, and improves quite dramatically classification results over the state-of-the-art when used with Riemann distance based classification algorithms. There remain some issues with the definiteness of the augmented covariance matrix, which stems from the augmentation of its dimensionality when the order increases. Another issue is the choice of the hyper-parameters which is costly when done exhaustively (grid search). Yet, we show that methods inspired by the Takens' theorem can provide some reasonable parameter values at the cost of a decrease of the improvements (but with improvements still in most cases).

The current study focused on offline classification. Our next direction will be to extend the approach for online classification. To do this, it will be necessary to test the algorithms on narrower epoch sliding windows. Another important development of the augmented approach would be to check its efficiency with smaller training dataset i.e. with either a low number of electrodes or with fewer training samples. This would provide an important usability improvement for BCIs, potentially reducing the setup and/or the calibration times. We have already seen that performance over the state-of-the-art is improved in the case of BNCI2014004, which possesses only 3 electrodes, but this needs to be generalized to other datasets and to a smaller set of training samples.

Another work direction is to analyze more in depth the distribution of hyper-parameters with the objective of identifying a methodology to make the classifier parameter free (e.g. by using ensemble classifiers methodology).

Acknowledgment

This work has been partially financed by a EUR DS4H/Neuromod fellowship. The authors are grateful to the OPAL infrastructure from Université Côte d'Azur for providing resources and support.

Data and Code Availability

The codes used to produce the results of this study are publicly available in this Github repository: `https://github.com/carraraig/Augmented_Covariance_BCI`. In the repository, we

have made available the code along with the results table that includes additional scoring metrics not explicitly reported in the paper. Moreover we release the function `AugmentedDataset` in the MOABB library in order to facilitate the use of the ACM.

References

[1] Brahim Belaoucha and Théodore Papadopoulo. "Structural connectivity to reconstruct brain activation and effective connectivity between brain regions". In: *Journal of Neural Engineering* 17.3 (June 2020), p. 035006. DOI: 10.1088/1741-2552/ab8b2b. URL: https://hal.inria.fr/hal-02945585.

[2] Mingzhou Ding et al. "Short-window spectral analysis of cortical event-related potentials by adaptive multivariate autoregressive modeling: data preprocessing, model validation, and variability assessment". In: *Biological cybernetics* 83.1 (2000), pp. 35–45.

[3] Claudia Kirch, Birte Muhsal, and Hernando Ombao. "Detection of changes in multivariate time series with application to EEG data". In: *Journal of the American Statistical Association* 110.511 (2015), pp. 1197–1216.

[4] Farhad Faradji, Rabab K Ward, and Gary E Birch. "A self-paced two-state mental task-based brain-computer interface with few EEG channels". In: *New Frontiers in Brain-Computer Interfaces*. IntechOpen London, UK, 2019.

[5] Xu Duan et al. "Quadcopter flight control using a non-invasive multi-modal brain computer interface". In: *Frontiers in neurorobotics* 13 (2019), p. 23.

[6] Henri Lorach et al. "Walking naturally after spinal cord injury using a brain–spine interface". In: *Nature* (2023), pp. 1–8.

[7] Dong Wen et al. "Combining brain–computer interface and virtual reality for rehabilitation in neurological diseases: A narrative review". In: *Annals of physical and rehabilitation medicine* 64.1 (2021), p. 101404.

[8] Alexandre Barachant et al. "Riemannian geometry applied to BCI classification". In: *International conference on latent variable analysis and signal separation*. Springer. 2010, pp. 629–636.

[9] Wolfgang Förstner and Boudewijn Moonen. "A metric for covariance matrices". In: *Geodesy-the Challenge of the 3rd Millennium*. Springer, 2003, pp. 299–309.

[10] Floris Takens. "Detecting strange attractors in turbulence". In: *Dynamical systems and turbulence, Warwick 1980*. Springer, 1981, pp. 366–381.

[11] Martin C Casdagli et al. "Non-linearity in invasive EEG recordings from patients with temporal lobe epilepsy". In: *Electroencephalography and clinical Neurophysiology* 102.2 (1997), pp. 98–105.

[12] Yonghui Fang, Minyou Chen, and Xufei Zheng. "Extracting features from phase space of EEG signals in brain–computer interfaces". In: *Neurocomputing* 151 (2015), pp. 1477–1485.

[13] Behshad Hosseinifard, Mohammad Hassan Moradi, and Reza Rostami. "Classifying depression patients and normal subjects using machine learning techniques and nonlinear features from EEG signal". In: *Computer methods and programs in biomedicine* 109.3 (2013), pp. 339–345.

[14] Jan Sosulski and Michael Tangermann. "Introducing Block-Toeplitz Covariance Matrices to Remaster Linear Discriminant Analysis for Event-related Potential Brain-computer Interfaces". In: *arXiv preprint arXiv:2202.02001* (2022).

[15] Xueling Zhou et al. "Multivariate phase space reconstruction and Riemannian manifold for sleep stage classification". In: *Biomedical Signal Processing and Control* 88 (2024), p. 105572.

[16] Bruno Aristimunha et al. "Mother of all BCI Benchmarks". In: *Zenodo* (2023).

[17] Alexandre Barachant et al. "A Brain-Switch using Riemannian Geometry". In: *Proceedings of the 5th International BCI Conference 2011*. Graz, Autriche, Sept. 2011, pp. 64–67. URL: http://hal.archives-ouvertes.fr/hal-00629110.

[18] Helmut Lütkepohl. *New introduction to multiple time series analysis*. Springer Science & Business Media, 2005.

[19] Trevor Hastie et al. *The elements of statistical learning: data mining, inference, and prediction*. Vol. 2. Springer, 2009.

[20] Daniel Bartz. *Advances in high-dimensional covariance matrix estimation*. Technische Universitaet Berlin (Germany), 2016.

[21] Javier de Pedro-Carracedo et al. "Phase space reconstruction from a biological time series: A photoplethysmographic signal case study". In: *Applied Sciences* 10.4 (2020), p. 1430.

[22] Michael Tangermann et al. "Review of the BCI competition IV". In: *Frontiers in neuroscience* (2012), p. 55.

[23] David Steyrl et al. "Random forests in non-invasive sensorimotor rhythm brain-computer interfaces: a practical and convenient non-linear classifier". In: *Biomedical Engineering/Biomedizinische Technik* 61.1 (2016), pp. 77–86.

[24] R. Leeb et al. "Brain–computer communication: motivation, aim, and impact of exploring a virtual apartment". In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 15.4 (2007), pp. 473–482.

[25] Josef Faller et al. "Autocalibration and recurrent adaptation: Towards a plug and play online ERD-BCI". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 20.3 (2012), pp. 313–319.

[26] Bangyan Zhou et al. "A fully automated trial selection method for optimization of motor imagery based brain-computer interface". In: *PloS one* 11.9 (2016), e0162657.

[27] Chang S Nam, Anton Nijholt, and Fabien Lotte. *Brain–computer interfaces handbook: technological and theoretical advances*. CRC Press, 2018.

[28] Gavin C Cawley and Nicola LC Talbot. "On over-fitting in model selection and subsequent selection bias in performance evaluation". In: *The Journal of Machine Learning Research* 11 (2010), pp. 2079–2107.

[29] George Datseris. "DynamicalSystems.jl: A Julia software library for chaos and nonlinear dynamics". In: *Journal of Open Source Software* 3.23 (2018), p. 598. DOI: 10.21105/joss.00598. URL: https://doi.org/10.21105/joss.00598.

[30] Chetan Nichkawde. "Optimal state-space reconstruction using derivatives on projected manifold". In: *Physical Review E* 87.2 (2013), p. 022905.

[31] Matthew B Kennel, Reggie Brown, and Henry DI Abarbanel. "Determining embedding dimension for phase-space reconstruction using a geometrical construction". In: *Physical review A* 45.6 (1992), p. 3403.

[32] Fabien Lotte and Cuntai Guan. "Learning from other Subjects Helps Reducing Brain-Computer Interface Calibration Time". In: *International Conference on Audio Speech and Signal Processing (ICASSP)*. Dallas, United States, Mar. 2010, pp. 614–617. URL: http://hal.inria.fr/inria-00441670.

[33] Florian Yger, Maxime Berar, and Fabien Lotte. "Riemannian approaches in brain-computer interfaces: a review". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 25.10 (2016), pp. 1753–1762.

[34] Marie-Constance Corsi et al. "Functional connectivity ensemble method to enhance BCI performance (FUCONE)". In: *IEEE Transactions on Biomedical Engineering* (2022).

[35] Vernon J Lawhern et al. "EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces". In: *Journal of neural engineering* 15.5 (2018), p. 056013.

[36] Robin Tibor Schirrmeister et al. "Deep learning with convolutional neural networks for EEG decoding and visualization". In: *Human brain mapping* 38.11 (2017), pp. 5391–5420.

[37] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: `https://www.tensorflow.org/`.

[38] Student. "The probable error of a mean". In: *Biometrika* (1908), pp. 1–25.

[39] Frank Wilcoxon. "Individual comparisons by ranking methods". In: *Breakthroughs in statistics*. Springer, 1992, pp. 196–202.

[40] Samuel A Stouffer et al. *The american soldier: Adjustment during army life.(studies in social psychology in world war ii), vol. 1*. Princeton Univ. Press, 1949.

[41] Kisung You and Hae-Jeong Park. "Geometric learning of functional brain network on the correlation manifold". In: *Scientific reports* 12.1 (2022), pp. 1–13.

[42] Olivier Ledoit and Michael Wolf. "A well-conditioned estimator for large-dimensional covariance matrices". In: *Journal of multivariate analysis* 88.2 (2004), pp. 365–411.

[43] Daniel Bartz and Klaus-Robert Müller. "Covariance shrinkage for autocorrelated data". In: *Advances in neural information processing systems* 27 (2014).

[44] David Sabbagh et al. "Manifold-regression to predict from MEG/EEG brain signals without source modeling". In: *Advances in Neural Information Processing Systems* 32 (2019).

[45] M. Moakher. "A Differential Geometric Approach to the Geometric Mean of Symmetric Positive-Definite Matrices". In: *SIAM J. Matrix Anal. Appl.* 26.3 (Apr. 2005), pp. 735–747.

[46] Marco Congedo, Alexandre Barachant, and Rajendra Bhatia. "Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review". In: *Brain-Computer Interfaces* 4.3 (2017), pp. 155–174.

[47] X. Pennec, P. Fillard, and N. Ayache. *A Riemannian Framework for Tensor Computing*. Tech. rep. 5255. INRIA, Sophia Antipolis, July 2004.

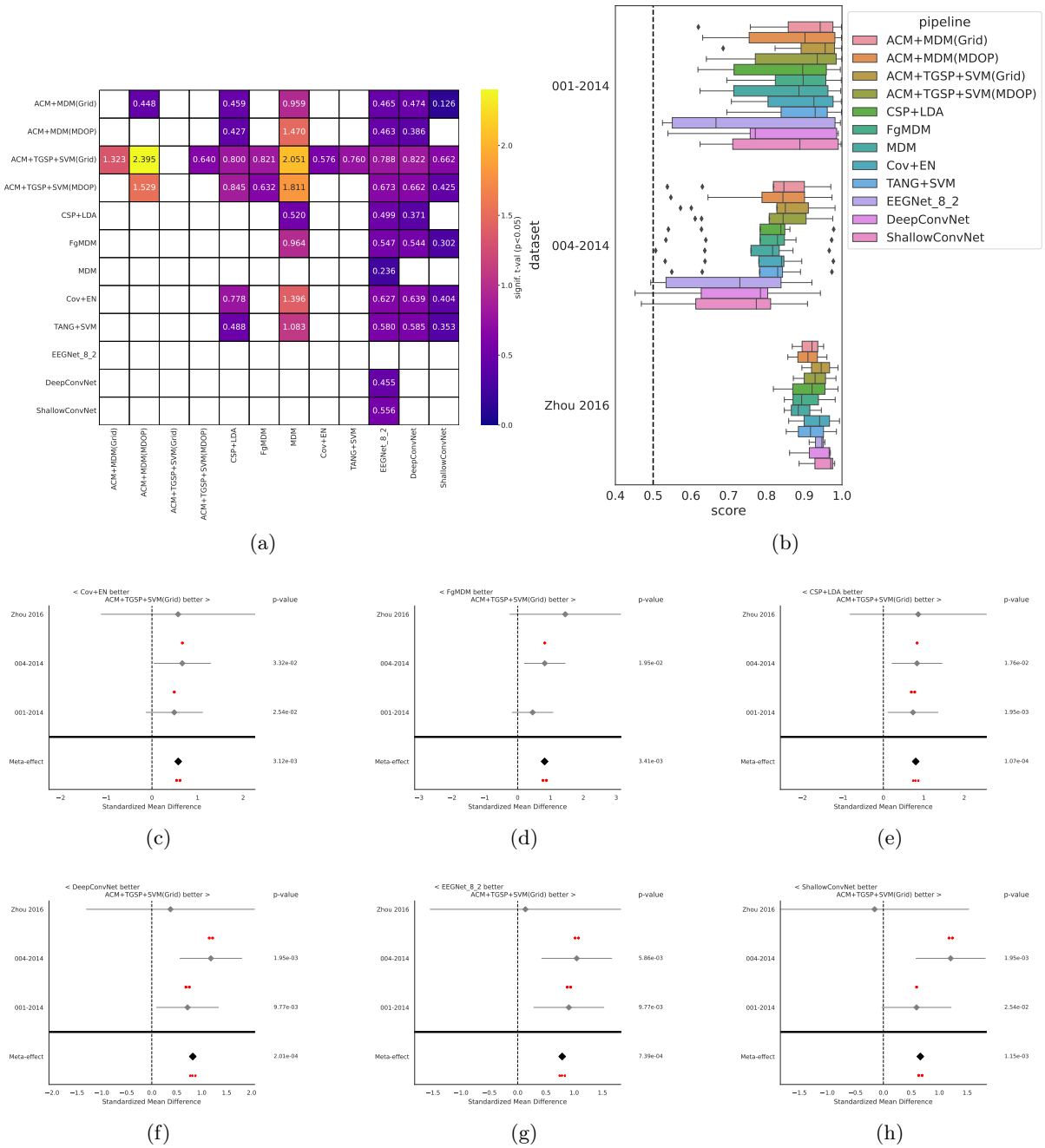
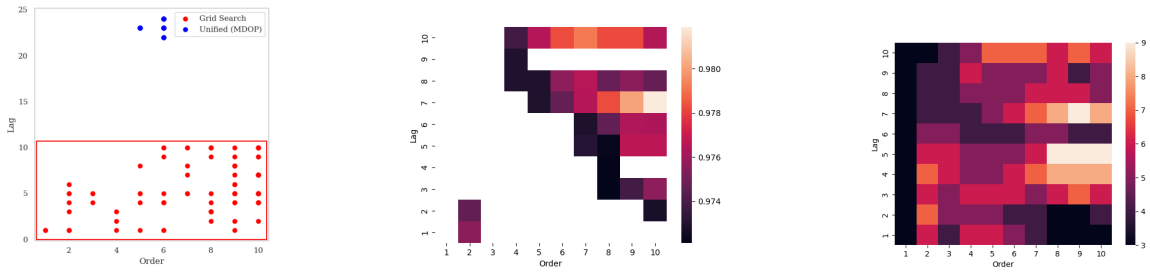


Figure 6: Results for Right vs Left hand classification, using WS evaluation. Plot (a) provides a combined meta analysis (over all datasets) of the different pipelines. It shows the significance that the algorithm on the y-axis is better than the one on the x-axis. The color represents the significance level of the difference of accuracy, in terms of t-values. We only show significant interactions ($p < 0.05$). Plot (b) is a box plot of the score with the error of the different pipelines for every dataset considered. Plots (c), (d), (e), (f), (g) and (h) show the meta analysis of ACM+TGSP+SVM (Grid) against respectively Cov+EN, FgMDM, CSP+LDA, DeepConvNet, EEGNet and ShallowConvNet. We show the standardized mean differences of p-values computed as one-tailed Wilcoxon signed-rank test for the hypothesis given as title of the plot. The gray bar denotes the 95% interval. * stands for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.



(a) (b) (c)

Figure 7: Plot (a) shows the best hyper-parameters for the ACM+MDM algorithm for each session and subject in the left hand right hand task, using the BNCI2014001 dataset. The red box defines the grid search domain. The heat-map in plot (b) shows the area that contains the hyper-parameter that provide a classification score of 99% of the maximum score obtained. These results are computed for Subject 1 in BNCI2014001 Session E in a right hand vs left hand task. The heat-map of plot (c) shows the maximal occurrence of best area of hyper-parameters across subjects and sessions of the BNCI2014001 dataset. It is obtained by counting the number of non empty boxes of plots similar to plot (b) over all sessions and subjects.

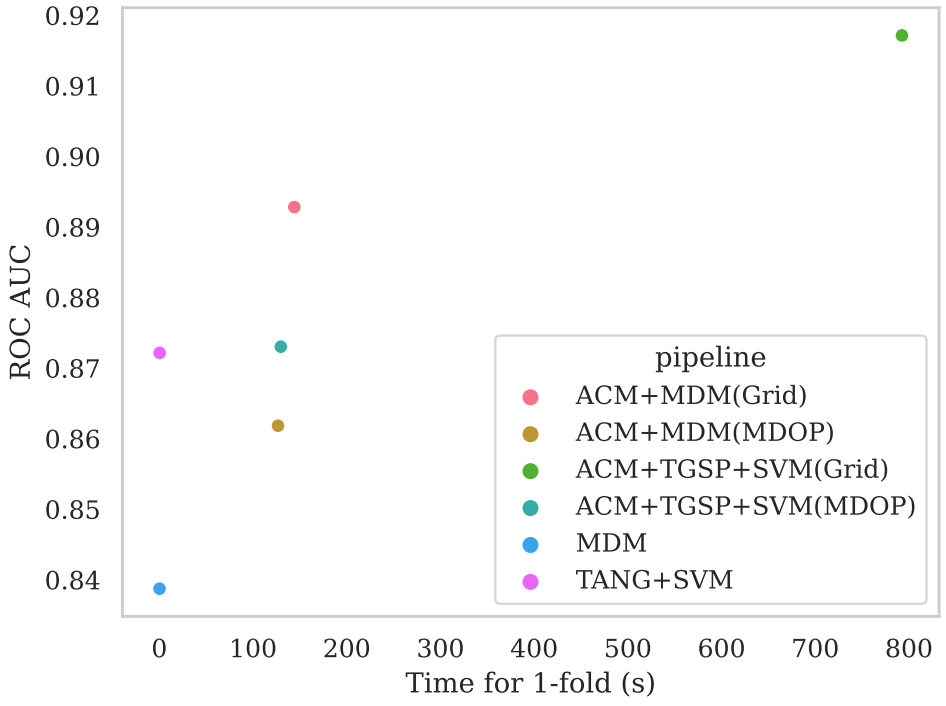


Figure 8: Total computational time (training+testing) vs ROC AUC criterion for the MOABB within-session evaluation of BNCI2014001 with the task right vs left hand classification. Similar behaviors occur in the other evaluation procedures, tasks and datasets.

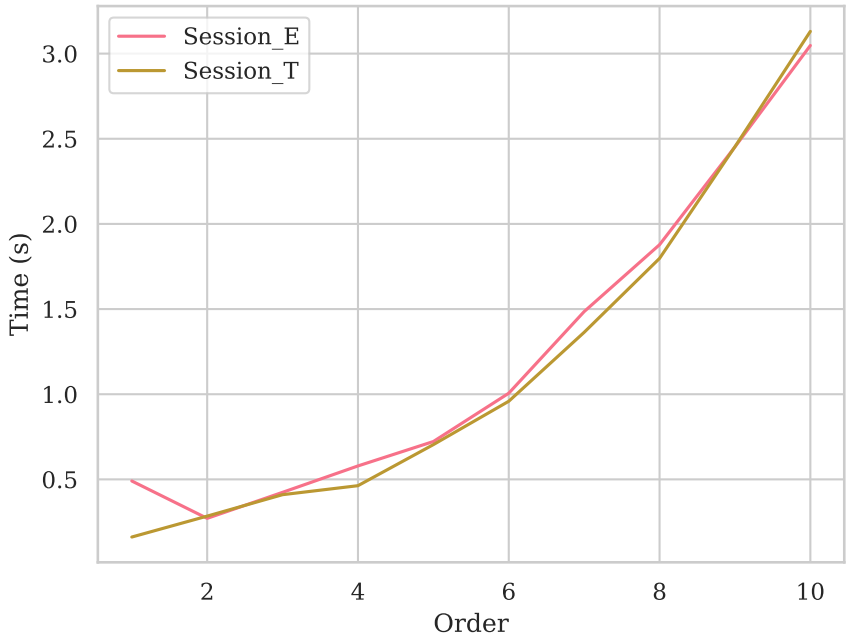


Figure 9: Total computational time (training+testing) for the MOABB within-session evaluation for subject 1 of BNCI2014001 with the task right hand vs feet using ACM+MDM without the grid search hyper-parameter selection but setting specifically the order. Computational times at order 1 correspond to standard covariance. Similar behaviors occur in the other evaluation procedures, subjects, tasks and datasets.

A Supplementary Material

A.1 Riemann geometry

Over the past 10 years, Riemannian approaches have shown interesting performances for SPD matrix classification [8]. The key idea behind this method is to map the covariance matrix into an appropriate geometrical space. The space of SPD matrix is a differentiable manifold with a natural Riemann structure [9].

The spaces of $n \times n$ real square and real symmetric matrices are denoted by $M(n)$ and $S(n)$, respectively: $S(n) = \left\{\mathbf{S} \in M(n),\, \mathbf{S}^T = \mathbf{S}\right\}$. The space of SPD matrices is defined as $P(n) = \{\mathbf{P} \in S(n), \mathbf{P} > 0\}$. In this way, we can represent a matrices $\mathbf{P} \in S(n)$ as a point on a Riemann manifold of dimension $n(n+1)/2$. Since the space of SPD matrix is a manifold of non zero curvature, we cannot expect that concepts of Euclidean geometry preserve the manifold structure. It is possible to define a distance between two SPD matrix $\mathbf{P}_1$ and $\mathbf{P}_2$ in $P(n)$ as the length of the geodesic that connects $\mathbf{P}_1$ and $\mathbf{P}_2$ on the Riemann manifold. In this paper, we use the affine-invariant metric [45]

$$\delta_R\left(\mathbf{P}_1, \mathbf{P}_2\right) = \left\| \log\left(\mathbf{P}_1^{-\frac{1}{2}} \mathbf{P}_2 \mathbf{P}_1^{-\frac{1}{2}}\right)\right\|_F = \left[\sum_{i=1}^{n} \log^2 \lambda_i\right]^{1/2} \tag{7}$$

where $\|\cdot\|_F$ is the Frobenius norm, $Log()$ is the logarithm of a matrix and λ_i are the eigenvalues of $\mathbf{P}_1^{-\frac{1}{2}} \mathbf{P}_2 \mathbf{P}_1^{-\frac{1}{2}}$. The choice of the metric is not unique but we choose this metric because it possesses relevant properties for BCI applications: invariance under reordering, invariance under congruent transformation and invariance under inversion [46]. To define the mean of m SPD matrices $(\mathbf{P}_1, ..., \mathbf{P}_m)$, we use the concept of Fréchet mean [45]

$$\overline{\mathbf{P}} = \underset{\mathbf{P} \in P(n)}{\operatorname{argmin}} \sum_{i=1}^{m} \delta_R^2\left(\mathbf{P}, \mathbf{P}_i\right) \tag{8}$$

$\overline{\mathbf{P}}$ is called the geometric mean in the Riemannian sense or the center of mass of $(\mathbf{P}_1, ..., \mathbf{P}_m)$. Note that, for $m > 2$, no closed-form expression is know, so we need to employ an iterative algorithm [47]. Due to the properties of Eq. (7), the geometric mean inherits the properties of congruence invariance [46].

For every point $\mathbf{P}_i \in P(n)$, in a complete Riemann space, we can define a point $\mathbf{S}_i \in S(n)$ in the tangent space $\mathbf{T}_P$ to the Riemann surface at point $\mathbf{P} \in P(n)$. The tangent space has zero curvature by definition and allows us to use standard ML algorithms and other classical tools. Mappings from the Riemann surface to the tangent space and from the tangent space to the Riemann surface, are provided respectively by the Exp and Log maps [45] (see Fig. A1):

$$\operatorname{Exp}_{\mathbf{P}}\left(\mathbf{S}_i\right) = \mathbf{P}^{1/2} \operatorname{Exp}\left(\mathbf{P}^{-1/2} \mathbf{S}_i \mathbf{P}^{-1/2}\right) \mathbf{P}^{1/2} \operatorname{Log}_{\mathbf{P}}\left(\mathbf{P}_i\right) = \mathbf{P}^{1/2} \operatorname{Log}\left(\mathbf{P}^{-1/2} \mathbf{P}_i \mathbf{P}^{-1/2}\right) \mathbf{P}^{1/2} \tag{9}$$

In the context of a Riemann geometry based classification algorithm, we estimate a spatial covariance matrix for each epoch with the standard approach of the spatial sample covariance estimator

$$\operatorname{Cov}(\mathbf{X}) = \frac{1}{Time - 1} \sum_{i=1}^{Time} \mathbf{X}_i \mathbf{X}_i^T\,, \tag{10}$$

where T is the transpose operator. This is not the only option for estimating the covariance. We could have used other estimators, but this one has the advantage of being unbiased when the number of electrodes is much smaller than the temporal instants contained in an epoch.

As the covariance matrix belongs to the SPD space, it is possible to classify the different mental states using the Riemannian framework. Note that the classification can be perform either

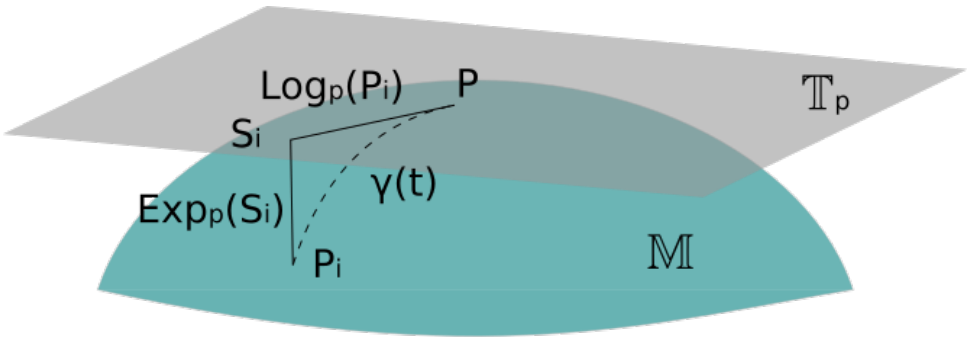


Figure A1: A Riemann manifold and the maps from/to the tangent space at point P..

on the Riemann surface, using algorithms specifically defined on the Riemann manifold as Minimum Distance to the Mean (MDM), or by moving points to tangent space and using standard classification algorithms such as Support Vector Machine (SVM) or Logistic Regression [17].

Table A1: Parameters for ML pipelines

Pipeline	**Parameter**	**Value**
"COV+EN"	l1 ratio	[0.15, 0.30, 0.45, 0.60, 0.75]
"TANG+SVM"	C	[0.5, 1, 1.5]
	Kernel	["linear", "rbf"]
"CSP+LDA"	nfilter	[1, 2, 3, 4, 5, 6, 7, 8]
"ACM+TANG+SVM"	C	[0.5, 1, 1.5]
	Kernel	["linear", "rbf"]
	Order	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
	Lag	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
"ACM+MDM"	Order	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
	Lag	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

Table A2: Common parameters for DL pipelines.

Parameter	**Value**
Epoch	300
Batch Size	64
Validation Split	0.2
Loss	Sparse Categorical Crossentropy
Optimizer	Adam Learning Rate = 0.001
Callbacks ES	Early Stopping Patience = 75 Monitor = Validation Loss
Callbacks LR	ReduceLROnPlateau Patience = 75 Monitor = Validation Loss Factor = 0.5

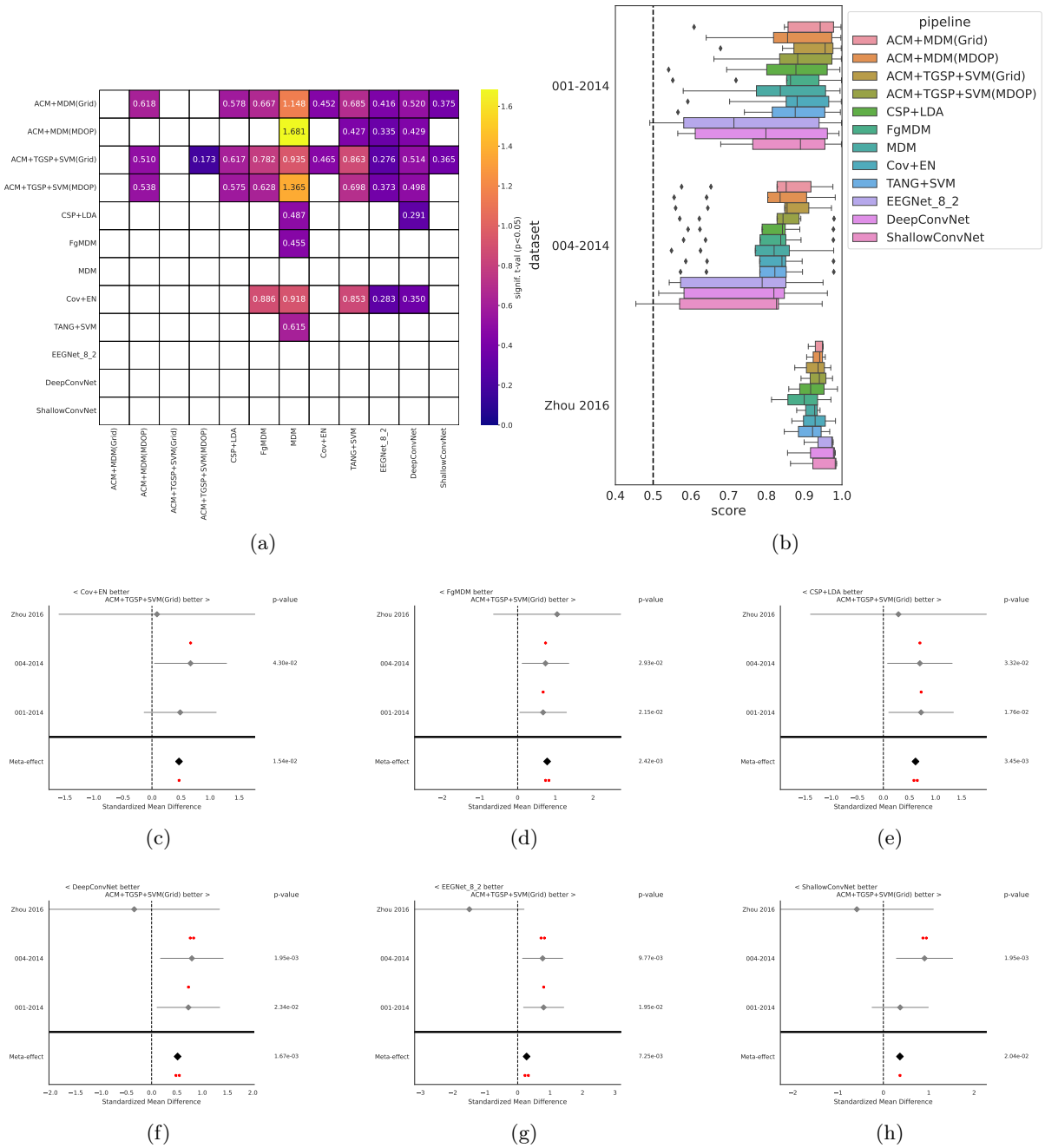


Figure A2: Results for Right vs Left hand classification, using CS evaluation. Plot (a) provides a combined meta analysis (over all datasets) of the different pipelines. It shows the significance that the algorithm on the y-axis is better than the one on the x-axis. The color represents the significance level of the difference of accuracy, in terms of t-values. We only show significant interactions ($p < 0.05$). Plot (b) is a box plot of the score with the error of the different pipelines for every dataset considered. Plots (c), (d), (e), (f), (g) and (h) show the meta analysis of ACM+TGSP+SVM (Grid) against respectively Cov+EN, FgMDM, CSP+LDA, DeepConvNet, EEGNet and ShallowConvNet. We show the standardized mean differences of p-values computed as one-tailed Wilcoxon signed-rank test for the hypothesis given as title of the plot. The gray bar denotes the 95% interval. * stands for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.

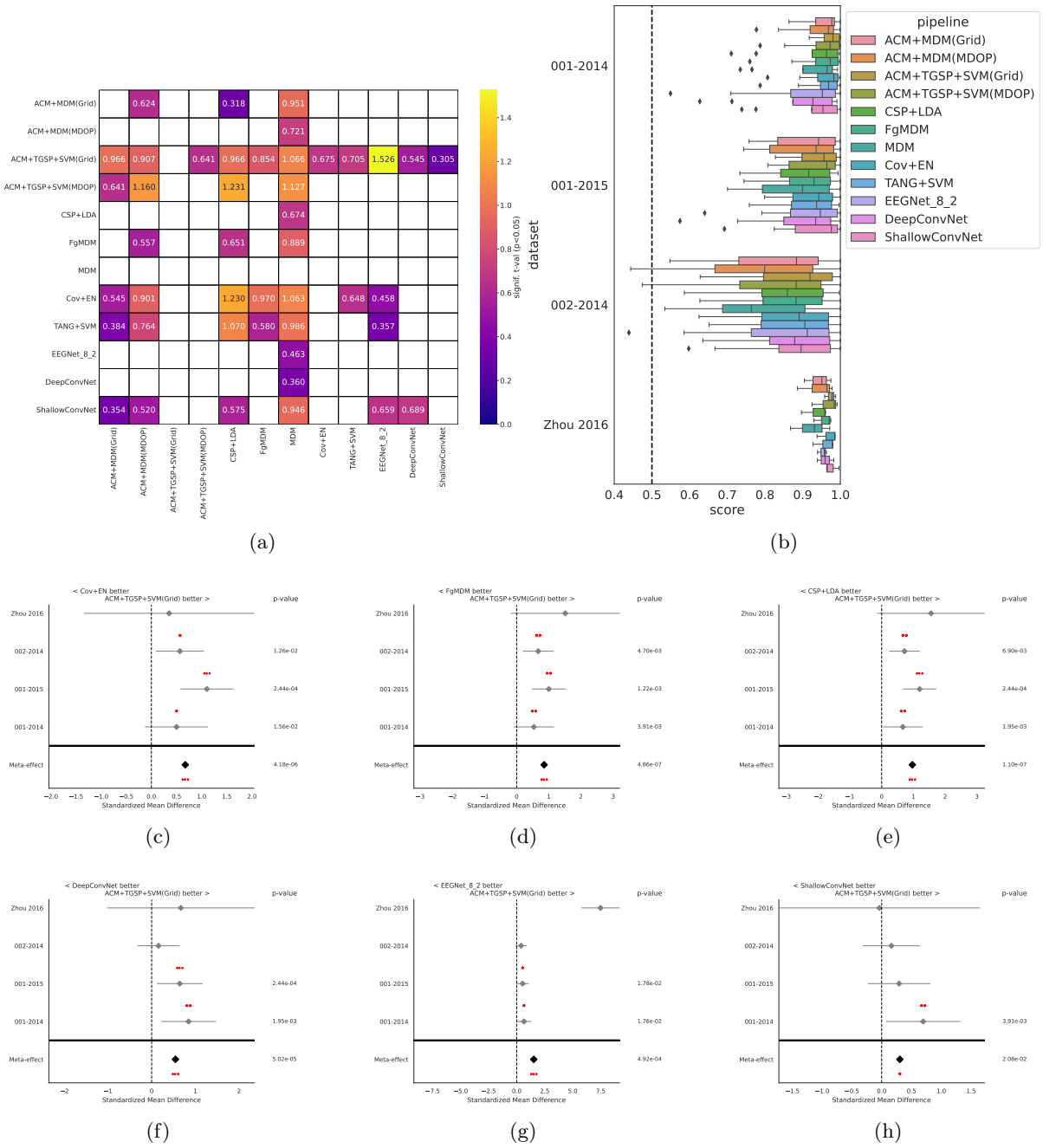


Figure A3: Results for Right hand vs Feet classification, using WS evaluation. Plot (a) provides a combined meta analysis (over all datasets) of the different pipelines. It shows the significance that the algorithm on the y-axis is better than the one on the x-axis. The color represents the significance level of the difference of accuracy, in terms of t-values. We only show significant interactions ($p < 0.05$). Plot (b) is a box plot of the score with the error of the different pipelines for every dataset considered. Plots (c), (d), (e), (f), (g) and (h) show the meta analysis of ACM+TGSP+SVM (Grid) against respectively Cov+EN, FgMDM, CSP+LDA, DeepConvNet, EEGNet and ShallowConvNet. We show the standardized mean differences of p-values computed as one-tailed Wilcoxon signed-rank test for the hypothesis given as title of the plot. The gray bar denotes the 95% interval. * stands for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.

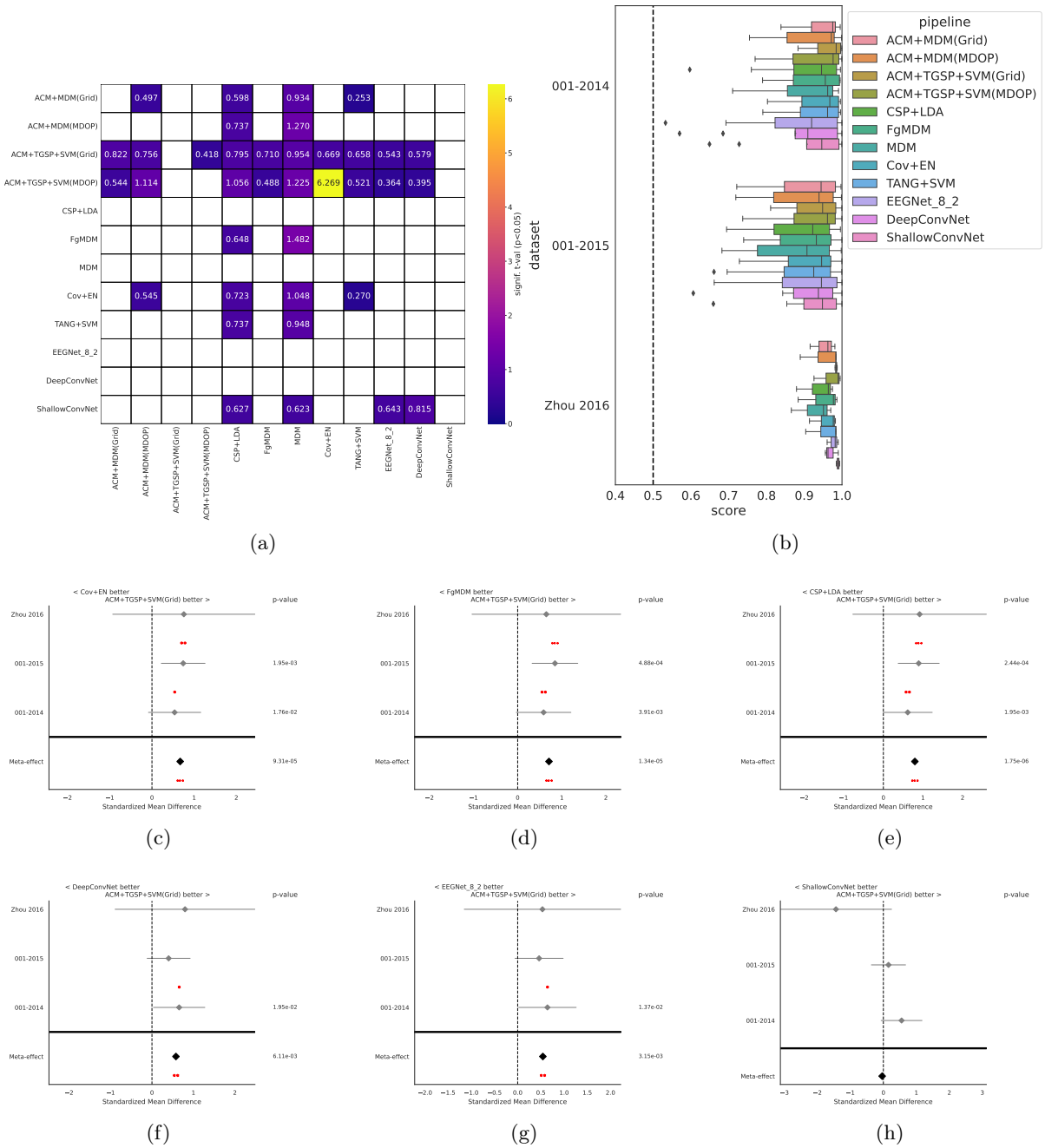


Figure A4: Results for Right hand vs Feet classification, using CS evaluation. Plot (a) provides a combined meta analysis (over all datasets) of the different pipelines. It shows the significance that the algorithm on the y-axis is better than the one on the x-axis. The color represents the significance level of the difference of accuracy, in terms of t-values. We only show significant interactions ($p < 0.05$). Plot (b) is a box plot of the score with the error of the different pipelines for every dataset considered. Plots (c), (d), (e), (f), (g) and (h) show the meta analysis of ACM+TGSP+SVM (Grid) against respectively Cov+EN, FgMDM, CSP+LDA, DeepConvNet, EEGNet and ShallowConvNet. We show the standardized mean differences of p-values computed as one-tailed Wilcoxon signed-rank test for the hypothesis given as title of the plot. The gray bar denotes the 95% interval. * stands for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.

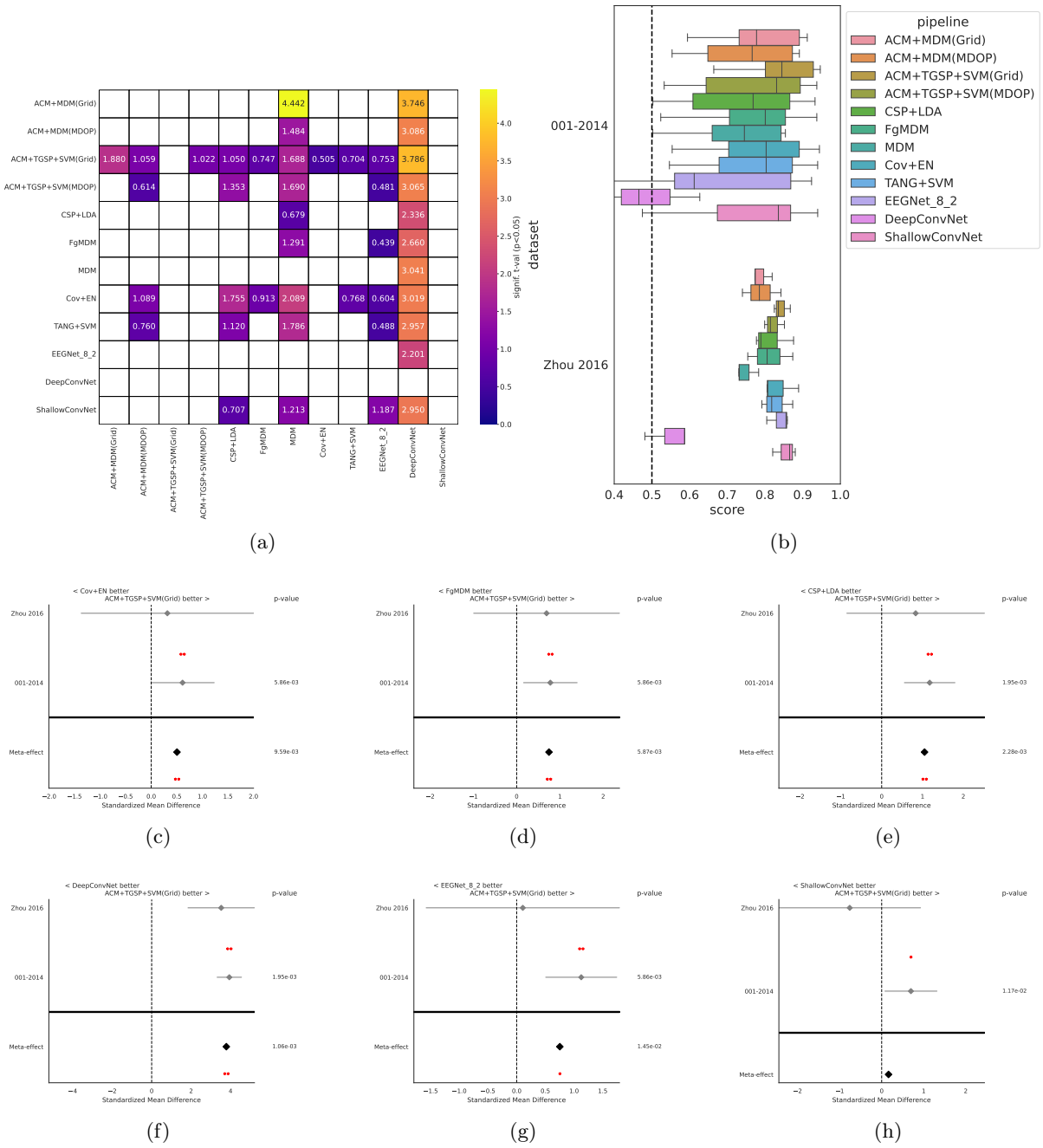


Figure A5: Results for Right hand vs Left hand vs Feet classification, using WS evaluation. Plot (a) provides a combined meta analysis (over all datasets) of the different pipelines. It shows the significance that the algorithm on the y-axis is better than the one on the x-axis. The color represents the significance level of the difference of accuracy, in terms of t-values. We only show significant interactions ($p < 0.05$). Plot (b) is a box plot of the score with the error of the different pipelines for every dataset considered. Plots (c), (d), (e), (f), (g) and (h) show the meta analysis of ACM+TGSP+SVM (Grid) against respectively Cov+EN, FgMDM, CSP+LDA, DeepConvNet, EEGNet and ShallowConvNet. We show the standardized mean differences of p-values computed as one-tailed Wilcoxon signed-rank test for the hypothesis given as title of the plot. The gray bar denotes the 95% interval. * stands for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.

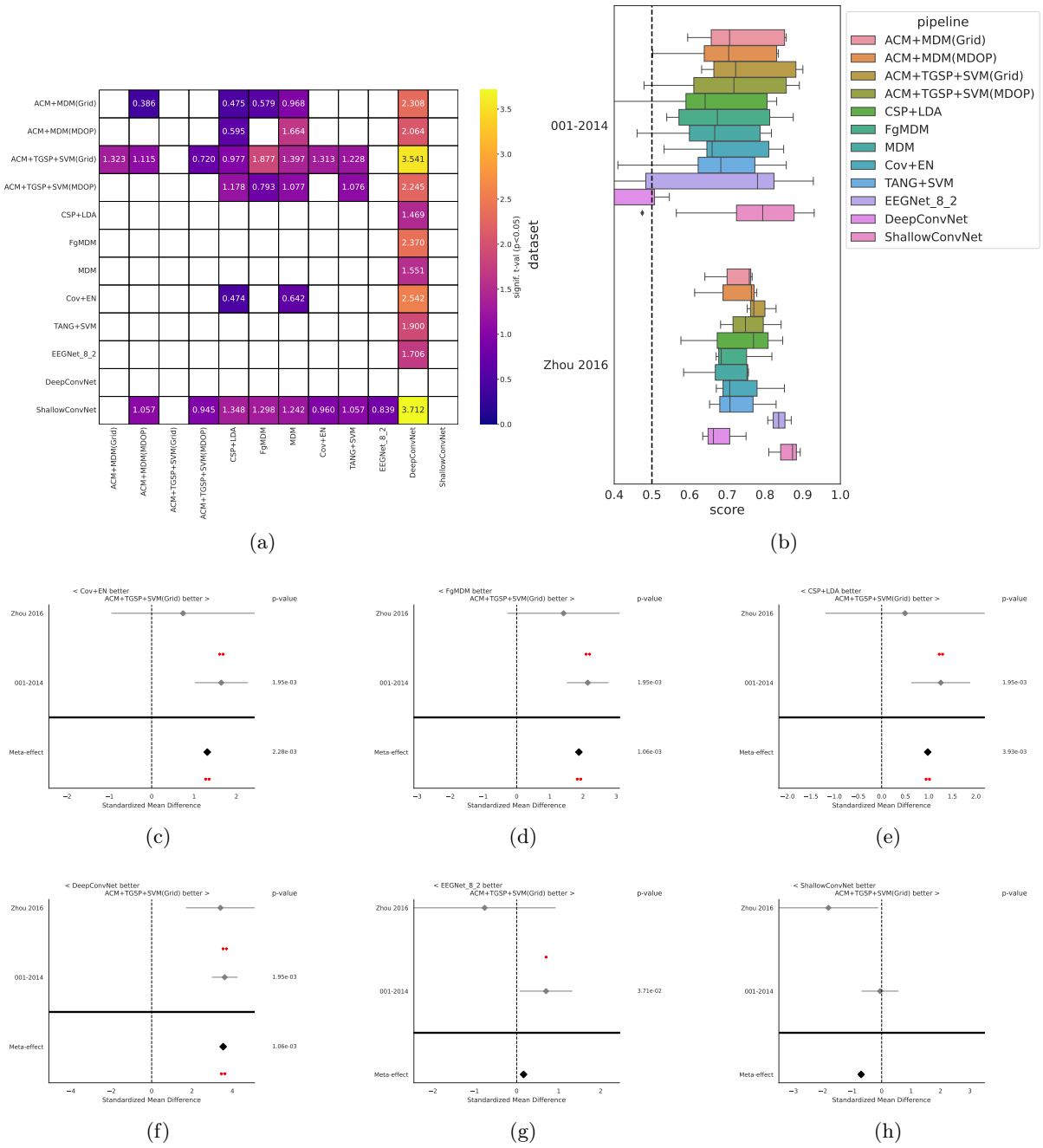


Figure A6: Results for Right hand vs Left hand vs Feet classification, using CS evaluation. Plot (a) provides a combined meta analysis (over all datasets) of the different pipelines. It shows the significance that the algorithm on the y-axis is better than the one on the x-axis. The color represents the significance level of the difference of accuracy, in terms of t-values. We only show significant interactions ($p < 0.05$). Plot (b) is a box plot of the score with the error of the different pipelines for every dataset considered. Plots (c), (d), (e), (f), (g) and (h) show the meta analysis of ACM+TGSP+SVM (Grid) against respectively Cov+EN, FgMDM, CSP+LDA, DeepConvNet, EEGNet and ShallowConvNet. We show the standardized mean differences of p-values computed as one-tailed Wilcoxon signed-rank test for the hypothesis given as title of the plot. The gray bar denotes the 95% interval. * stands for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.