Riemannian classification for SSVEP based BCI: offline versus online implementations

Sylvain Chevallier, Emmanuel Kalunga, Quentin Barthélemy, Florian Yger

Riemannian classification for SSVEP based BCI: offline versus online implementations

Chapter 19 in Brain Computer Interfaces Handbook: Technological and Theoretical Advances

Sylvain Chevallier, Emmanuel Kalunga, Quentin Barthélemy, Florian Yger
sylvain.chevallier@uvsq.fr

Junuary 2018

Abstract

This chapter focus on the different implementations of brain-computer interface (BCI) based on Steady-State Visually Evoked Potentials (SSVEPs). In offline BCI, feature extraction and the classification are performed at the end of the session, when all trials are available. Whereas, in online settings, they are performed several times during each trial, usually for each available epoch recorded by the electroencephalogram (EEG) device, enabling real-time and asynchronous BCI. A recent successful approach in feature extraction and signal processing for BCI is Riemannian geometry, which deals with covariance matrices. They capture the degree of correlation between several random variables, that is how the brain signals change relatively to each other. These techniques have demonstrated their benefit on several occasions, leading to winning algorithms in international competitions and to state-of-the-art results on renowned BCI benchmarks. After reviewing some of the most robust approaches in feature extraction for SSVEP, this chapter will introduce newer tools based on Riemannian geometry. With an application to SSVEP, this article shows through a comparison how Riemannian geometry allows one to easily define offline and online implementations that have better accuracies than state of the art.

1 Introduction

This chapter presents the feature extraction and classification techniques applied for signals of a particular paradigm of brain-computer interface (BCI). The most successful approaches, that is those providing the highest transfer rates and the most robust representations, share one common ground: they are all estimating the covariance from the signal to build or to derive their feature. The covariance captures the degree of linear dependence between several random variables, that is how the brain signals change relatively to each other. If two signals show the same variations, increasing and decreasing at the same time, they are dependent. The notion of covariance is central in several fields of science, such as mathematical finance, meteorology, oceanography and, of course, signal processing. A correct covariance estimation requires to take into account a noticeable part of the time signal history. This fact has a strong impact when setting up a BCI system, as it introduces a delay to allow a correct processing and interpretation of the brain signal. Special attention is thus needed when dealing with covariance-based approaches in online systems, to aim for a robust system without impeding the interactions with a high latency.

After reviewing some of the most robust approaches in features extraction for Steady-State Visually Evoked Potentials (SSVEP), this chapter will present recently introduced tools for signal processing based on a non-Euclidean geometry, namely the Riemannian geometry. These techniques have demonstrated their benefit on several occasions, leading to winning algorithms in international competitions and to state-of-the-art results on renowned BCI benchmarks. Most of these achievements are built on the theoretical advances of a very active community working on Information Geometry and its applications to signal processing, for example in radar imagery, computer vision, or finance. A thorough review of the existing Riemannian approaches for BCI is proposed, with its application to SSVEP.

Regarding the implementation in offline BCI, feature extraction and classification are performed at the end of the session, when all trials are available. Based on such a setting, it is difficult to design algorithms for real-time BCI. On the contrary, in online implementation, feature extraction and classification are performed several times during each trial, usually for each available epoch recorded by the electroencephalogram (EEG) device. This setting being more realistic, it enables the development of real-time and asynchronous BCI. Applied to SSVEP, this chapter will compare on real data how Riemannian geometry allows one to easily design offline and online implementations for BCI with a better accuracy than state of the art. Moreover, using covariance matrices as feature allows one to define a resting-state class, which is a crucial point in BCI design.

Section 2 gives an overall view of the SSVEP-based BCI and the state-of-the-art approaches. The implication of designing online algorithms are explained and highlighted. Section 3 provides a detailed description of the mathematical and theoretical principles involved in the feature extraction step. The importance of the covariance matrices are shown and the link with Riemannian geometry is explained. In Section 4, an overview of the Riemannian tools and their application to covariance matrices are proposed. The existing approaches to estimate covariance matrices are explained before this section covers the known distances, the mean estimation and the classification in the space of covariance matrices. Offline and online implementations are described. Section 5 explains how these Riemannian approaches are applied to EEG signal. The existing algorithms are evaluated on an SSVEP data set and the results are analyzed in terms of quality of prediction and computing load. Section 6 concludes this chapter and opens on questions on the future of Riemannian-based BCI.

2 A review of SSVEP-based BCI

This section quickly reviews some points about SSVEP and then details its application in BCI. As the notion of offline analysis and online algorithms is at the center of this chapter, this section explains the differences between those approaches.

2.1 Steady-state visually evoked potentials

Dealing with sensory evoked potentials, it is a common view to oppose event-related potential and Steady-State Response (SSR) (Regan, 1982), including their visual counterpart SSVEP. This distinction originates from the idea that the SSR may be generated by neural oscillations elicited by the repeated stimulations (Takahashi, 2004, p. 4) whereas the ERP is the transient response to an event occurring at a sufficiently long time interval to allow the system to return to its initial state (Niedermeyer and Lopes da Silva, 2004).

The SSVEP-based BCI is often employed as a dependent BCI (Wolpaw et al., 2002), that is, some residual muscular capabilities are required to move the eye toward the blinking stimulus as opposed to independent BCI, such as Motor Imagery (MI), where the communication does not rely on any motor capability. It has been shown that SSVEP could be used as an independent BCI (Morgan et al., 1996; Müller et al., 2006) as the neural oscillations are strongly related to the focus of attention. Using covert attention, that is shifting the focus of attention without moving the eyes, subjects can generate different SSVEP responses.

Visual stimulus plays a crucial role, affecting the SSVEP-based BCI performance, and should be designed carefully. An in-depth review of the literature (Zhu et al., 2010) shows that LED stimuli provide better results than those obtained on a computer screen. Any stimulation between 2 and 50 Hz induces visible oscillations in the visual cortex (Herrmann, 2001), with a peak in signal-to-noise ratio visible around 15 Hz (Pastor et al., 2003). Common values employed in SSVEP studies are between 12 and 25 Hz, as they induce oscillations with higher amplitudes (Zhu et al., 2010). One should note that safety considerations should be taken into account as some frequency ranges of the stimulation train may trigger epileptic seizure (Fisher et al., 2005).

Review articles (Vialatte et al., 2010; Zhu et al., 2010; Liu et al., 2014; Stawicki et al., 2017) provide more details on SSVEP: properties of the visual stimuli (influence of the number of visual stimulation frequencies to accommodate an increasing number of BCI commands), stimulation paradigm (multiple frequencies sequential coding and frequency shift keying), and electrophysiological response. For more details about SSVEP, one should refer to (Stawicki et al., 2017) in the Part I of this book, Chapter 8 "Design and Development of User Friendly SSVEP-based BCI Applications for Elderly People".

2.2 Online and offline implementations for BCI

It is important to define some of the terminologies used in the chapter and in the literature. A first important notion is to differentiate between synchronous and asynchronous BCI. In synchronous setting, the system provides temporal cues for the subject, asking to perform specific action at a specific time. This is not the case in asynchronous BCI, where the subject could act or elicit an action from the system at any time. In SSVEP both settings could be found, even if the synchronous setting is very common in the literature, as it is simpler to obtain meaningful and reproducible results.

Implementations of feature extraction and classification could be online or offline. Distinguishing between those two implementations requires one to define some necessary notions. We call samples the values acquired by the system at each time step. A group of successive samples of a predetermined size is called an epoch; it is used to simplify the feature extraction. We define a trial as the sequence in the experimental plan where the user is asked to perform a specific task. A session is defined here as a group of trial performed successively by the subject. The classification results are generally obtained by averaging performances across all trials.

We can thus distinguish the different implementations (Barthélemy et al., 2017):

- In offline settings, feature extraction and the classification are performed at the end of the session; thus, all trials are available. It is not possible to design a real-time BCI system with such implementation.
- In block-online settings, classification is performed at the end of each trial; the trials are available one after the other. This is the common setting for synchronous interactions.
- In online settings, feature extraction and the classification are performed several times during each trial, usually for each available epoch recorded by the EEG device, sometimes for each sample if the computation time is compatible with the real-time constraints. With this implementation, it is possible to set up an asynchronous BCI.

Note that using or not using a preliminary calibration on a training set does not change these distinctions.

One of the most crucial problems in BCI is the difference between these offline and online implementations. Offline analyses represent the major part of the scientific publications, but they cannot serve, as they are, in practical implementations of a BCI, which are necessarily online or block-online. Offline implementations often use non-causal processing (e.g. bilateral filtering), normalizations or whitening that requires one to know the whole data distribution of the session. But these computations are not possible in online settings, hence converting an offline implementation to an online one is not trivial and can generate a loss of performances, in terms of classification accuracy.

3 Classifying SSVEP signals

This section presents the common techniques to classify SSVEP signals, that is finding the frequencies of brain waves in the visual cortex that correspond to the target stimulus.

3.1 Notations

In the following, C denotes the number of electrodes recorded by the EEG system. Depending on the sampling rate of the EEG system, a certain number of samples are recorded each second. Each

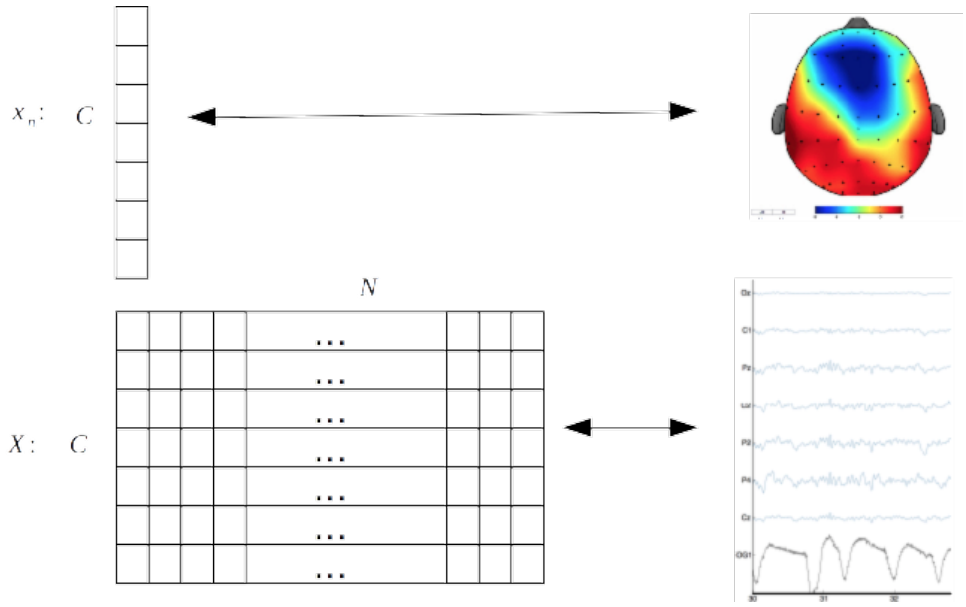


Figure 1: Potential differences measured on the scalp are gathered in vector x_n (top). A complete recording is represented as a matrix X of C lines, corresponding to the number of electrodes, and N columns, corresponding to the number of samples (bottom).

sample contains C values indicating the potential difference, usually expressed in microvolts, measured at each electrode site. The potential differences are represented in a vector of C lines, denoted x_n where $n = 1, \ldots, N$. This is illustrated in the upper part of the Fig. 1. A recording is represented as a matrix $X = [x_1, \ldots, x_N], X \in \mathbb{R}^{C \times N}$, which is the concatenation of all samples, as shown in the bottom of Fig. 1. We will make the hypothesis that all N samples x_n are randomly drawn from a distribution. It follows that $\boldsymbol{x}$ is a variable of random vectors and its expected vector is $\omega = E\{\boldsymbol{x}\}$ (Fukunaga, 1990). The covariance matrix of the random vector $\boldsymbol{x}$ is defined by $\Sigma = E\{(\boldsymbol{x} - \omega)(\boldsymbol{x} - \omega)^\intercal\}$.

3.2 Classification using canonical correlation

To classify SSVEP signal, one should find the frequencies of brain waves in the visual cortex corresponding to the target stimulus. Several methods have been proposed, using power spectral density or other decomposition methods (Liu et al., 2014). The most robust approaches are based either on canonical correlation or on Riemannian geometry.

Until recently, methods relying on Canonical Correlation Analysis (CCA) originating from Hotelling (1936), were achieving the highest classification performances, but implementation is offline (Lin et al., 2006). Given two set of signals, CCA aims at finding the projection space that maximizes their cross-covariance while jointly minimizing their covariance.

For SSVEP, the CCA aims to find a new subspace such that the two sets of variables have maximal correlation when projected on this subspace. The main idea is to use $X \in \mathbb{R}^{C \times N}$ as a first set of variables and $2N_h$ reference signals $Y_f \in \mathbb{R}^{2N_h \times N}$ as the second set. A common way of generating the representation of the simulation signal at frequency f is:

$$Y_f = \begin{bmatrix} \sin(2\pi f n) \\ \cos(2\pi f n) \\ \vdots \\ \sin(2\pi N_h f n) \\ \cos(2\pi N_h f n) \end{bmatrix}, \quad n = \frac{1}{T_s}, \frac{2}{T_s}, \ldots, \frac{N}{T_s} \tag{1}$$

where T_s is the EEG sampling frequency, N_h is the number of harmonics, and N is the number of samples. N_h is a parameter that can be defined by cross validation.

The CCA seeks two projection directions w_X and w_{Y_f} such that $w_X^T X$ and $w_{Y_f} Y_f$ have maximal correlation. w_X and w_{Y_f} maximize the correlation function $\rho(w_X, w_{Y_f})$:

$$\begin{aligned} \rho(w_X, w_{Y_f}) &= \mathrm{corr}(w_X^T X, w_{Y_f} Y_f) \\ &= \frac{w_X^T \Sigma_{XY_f} w_{Y_f}}{\sqrt{w_X^T \Sigma_X w_X w_{Y_f}^T \Sigma_{Y_f} w_{Y_f}}} , \end{aligned} \tag{2}$$

where Σ_{XY_f} is the between-set covariance matrix; Σ_X and Σ_{Y_f} are the within-set covariance matrices. CCA can be solved as in Hardoon et al. (2004):

$$\begin{aligned} \underset{w_X, w_{Y_f}}{\text{maximize}} \quad & w_X^T \Sigma_{XY_f} w_{Y_f} \\ \text{subject to} \quad & w_X^T \Sigma_X w_X = 1, \\ & w_{Y_f}^T \Sigma_{Y_f} w_{Y_f} = 1. \end{aligned}$$

The CCA objective is to maximize the correlation ρ between $w_X^T X$ and $Y_f^T w_{Y_f}$ as follows:

$$\rho_f = \max_{w_X, w_{Y_f}} \frac{w_X^T X Y_f^T w_{Y_f}}{\sqrt{w_X^T X X^T w_X w_{Y_f}^T Y_f Y_f^T w_{Y_f}}} . \tag{3}$$

Note that when more than one couple of directions w_X and w_{Y_f} are needed, a deflation step can be applied and then another problem of CCA is solved. To determinate the target frequency $\hat{f}$, the values of ρ_f are used for classification as (Lin et al., 2006):

$$\hat{f} = \arg\max_f \{\rho_f\} . \tag{4}$$

A different approach considers CCA only to obtain a spatial filter, using a support vector machine (SVM) to process the spectral features of the filtered signal Kalunga et al. (2013). This approach is referred to as CCA+SVM in this work.

Several aspects regarding the CCA are still debated and not clearly established, especially concerning the harmonics. The number of harmonics to choose and its impact on the classification accuracy are unclear. Chen et al. (2015a) introduce the Filter Bank Canonical Correlation Analysis (FBCCA), which is an online algorithm. The EEG signal is filtered into $n = 1 \dots N_b$ sub-bands covering multiple harmonic frequency bands. As described above, CCA allows one to obtain a correlation coefficient ρ_f^n for each sub-band n. The correlation coefficients are combined as

$$\tilde{\rho}_f = \sum_{n=1}^{N_b} w_n (\rho_f^n)^2 , \tag{5}$$

where N_b is the number of sub-bands, n_b and w_n respectively the index and the weight of a sub-band. The weights are defined as:

$$w_n = n^{-a} + b, \quad n \in [1 \dots N_b] , \tag{6}$$

where a and b are constants that maximize the classification performance. Hyper-parameters a, b, and N_b are determined in practice with a grid search conducted on an offline analysis. Once all $\tilde{\rho}_f$ are determined, the classification is done with Eq. (4). FBCCA significantly improves the performance of standard CCA (Chen et al., 2015a).

Recent studies conducted by Nakanishi et al. (2014) and by Chen et al. (2015b) made a breakthrough in high-speed SSVEP-based speller, used in online settings. The objective of these experiments is to successively select letters for a spelling task; SSVEP achieves the highest information transfer rate (ITR) reported in EEG-based BCI. Nakanishi *et al.* brought the ITR to an average of 166.91 bits/min

(Nakanishi et al., 2014) and later, Chen *et al.* raised it to an average of 270 bits/min (Chen et al., 2015b). Nakanishi et al. (2014) achieved high classification performances, mainly relying on methodological improvement of the stimulation protocol (phase modulation and combination of frequency) and stimulus presentation (the proposed letters follow a similar scheme to SMS writing). However, there are two strong limitations to these works. First, these studies do not consider a resting-state class (or a reject class), where the user does not gaze at any specific stimulus, meaning that the system continuously output a selected letter without any pause. Allowing users to act at their own pace is utterly important in Human-Machine Interface (HMI), and hence in BCI. Moreover, taking into account the case where the user does not gaze at any stimulus is a challenging issue as traces of stimulation frequencies are still observable in the EEG. Second, the stimulation protocol is complex: a successful classification requires a synchronization between the stimulation and EEG acquisition devices to ensure a proper time reference for the phase measurements. This is not guaranteed in many experimental settings and it is a limitation for the potential applications.

3.3 Classification using Riemannian geometry

The covariance is central in state-of-the-art CCA approaches as it allows one to estimate spatial filters that are projections of the original sensor space into a surrogate sensor space that enhances the signal of interest. Spatial filters are user-specific signal processing techniques yielding a robust representation, with strong discrimination properties, allowing good accuracy in classification. Spatial filters are very efficient on clean datasets obtained from strongly constrained environment but they are sensitive to artifacts and outliers (Lotte and Guan, 2011; Tomioka et al., 2007). But working directly on covariance matrices is advantageous: it simplifies the whole BCI system (Yger, 2013), avoiding the alignment of two learning steps (spatial filters and classifiers) that might lead to overfitting. The covariance matrices, being Symmetric and Positive-Definite (SPD), are best handled by tools provided by Riemannian geometry (Bhatia, 2009). Classification in the space of SPD matrices eliminates the need of spatial filters and improves the system robustness (Barachant et al., 2012; Congedo et al., 2013; Yger, 2013). Riemannian-based approaches have demonstrated their efficiency by outranking all other existing techniques on real competitive problems: DecMeg 2014 challenge[1], BCI Challenge - NER 2015[2] and Grasp-and-Lift EEG Detection 2015[3].

To detail how Riemannian geometry can improve machine learning algorithms, one can argue that it takes explicitly into consideration the underlying structure of the data. Different approaches in the literature use the geometry of data in machine learning (for complete review see (Yger et al., 2017; Congedo et al., 2017)). A first possibility is to rely on the mapping of the Riemannian manifold onto a Euclidean vector space, where common algorithm could be applied. One such mapping, called logarithmic mapping, exists between the manifold and its tangent space, which is a Euclidean space, and has been used in classification task for BCI (Barachant et al., 2013b). Some kernels have been applied successfully to this end: Stein kernel, Log-Euclidean kernels as well as their normalized versions (Yger, 2013). The family of kernels defined on the Riemannian manifold allows the implementation of extensions of all kernel-based methods, such as SVM, kernel-PCA or kernel k-means (Jayasumana et al., 2013). Apart from the kernel approaches, once the data are mapped onto a vector space, any machine learning algorithm working in Euclidean space, such as LDA, could be applied (Barachant et al., 2012).

Another possibility is to develop new algorithms directly for Riemannian manifolds. The Minimum Distance to Riemannian Mean (MDRM) relies on a Riemannian metric to implement a multiclass classifier and have been applied on EEG. New EEG trials are assigned to the class whose average covariance matrix is the closest to the trial covariance matrix (Barachant et al., 2012). The MDRM classification can be preceded by a spatial filtering of covariance matrices, like in (Barachant et al., 2010) where covariance matrices are filtered with LDA components in the tangent space, and then brought back to the

[1] https://www.kaggle.com/c/decoding-the-human-brain
[2] https://www.kaggle.com/c/inria-bci-challenge
[3] https://www.kaggle.com/c/grasp-and-lift-eeg-detection

Riemannian space for classification with MDRM. Another example is the Riemannian Potato (Barachant et al., 2013a), an unsupervised and adaptive artifact detection method, providing an online EEG outlier removal. Incoming epochs are rejected if their covariance matrix lies beyond a predefined z-score.

To apply Riemannian geometry to SSVEP, the sample covariance matrices can be defined from a rearrangement of the recorded data. The rearrangement is done such that the temporal or frequency information is captured (Congedo et al., 2013; Congedo, 2013). With similar motivations, Li et al. (2012) and Li and Wong (2013) defined a new Riemannian distance between SPD matrices filtered in different frequency bands: an optimized spatial weighting matrix is commonly applied on covariance matrices of each frequency band. They use this new distance as a dissimilarity between weighted matrices of power spectral density to classify EEG into different sleep states by k-nearest neighbors.

4 Riemannian geometry

This section starts with a presentation of the tools and characteristics of the geometry of SPD matrices. A simple classification algorithm is then described, which works directly in the space of covariance matrices. This simple yet efficient classifier allows one to achieve near state-of-the-art results. It relies on two key aspects, the estimation of covariance matrices, which should be as accurate as possible, and the choice of a metric to estimate the center of mass of a set of covariance matrices. These aspects, covariance estimation and existing metrics, are presented in this section. Finally, MDRM classifier is presented for offline and online implementations.

4.1 Geometry of covariance matrices

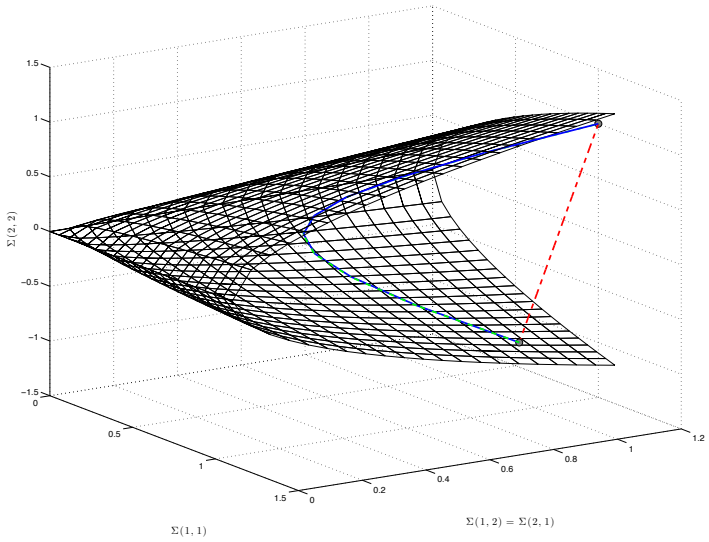


Figure 2: Visualization of the covariance matrices $\Sigma \in \mathbb{R}^{2\times 2}$. The Euclidean distance (dashed red line) does not consider the curvature of the space and define chordal distance between elements. Riemannian distances (AIR in plain blue and Log-Euclidean in dashed-dotted green) follows a geodesic and are thus taking into account the shape of the space where covariance matrices lie.

Covariance matrices are SPD and are thus constrained to lie strictly inside a convex cone, as shown in Fig. 2. This special topological space is a Riemannian manifold. Manifold could be described as a collection of small flat structures, "glued" together. For each of these small structure, the neighborhood of an element could be considered as flat, which is similar to a tangent space. The bijection between the neighborhood of a manifold element and $\mathbb{R}^m$ is called a chart. A smooth differentiable atlas is a

collection of charts verifying that the elements overlap smoothly (Absil et al., 2009). When the manifold is equipped with a complete, smooth differentiable atlas and with an inner product defined on the tangent space of each element, it is called a Riemannian manifold.

As illustrated by Yger et al. (2017), we can draw the parallel with the situation that occurs on Earth. Indeed, the surface of Earth is a smooth lower-dimensional subspace. At every point of the surface, we can approximate the surface as a map (a locally accurate flat approximation). Then, the shortest path between two points on the surface is a curved called a geodesic.

The covariance matrices are elements of $\mathcal{M}_C$, a manifold of $C \times C$ symmetric positive-definite matrices,

$$\mathcal{M}_C = \left\{ \Sigma \in \mathbb{R}^{C \times C} \mid \Sigma = \Sigma^\intercal \text{ and } x^\intercal \Sigma x > 0, \forall x \in \mathbb{R}^C \backslash 0 \right\} .$$

SPD matrices verify the properties listed in Table 1.

Symmetry	$\Sigma = \Sigma^\intercal$
Positive definiteness	$x^\intercal \Sigma x > 0, \forall x \in \mathbb{R}^C \backslash 0$
Strict positivity of diagonal elements	$\Sigma(i,j) > 0 \mid i = j, \forall i,j \in \{1,\dots,C\}$ i.e. positive variance
Cauchy-Schwarz inequalities	$\lvert\Sigma(i,j)\rvert \leq (\Sigma(i,i)\Sigma(j,j))^{1/2}, \forall i,j \in \{1,\dots,C\}$

Table 1: Properties of the SPD matrices.

The tangent space $T_\Sigma \mathcal{M}_C$, which is a local linear approximation of the manifold at the point Σ, is identified to the Euclidean space of symmetric matrices:

$$\mathcal{S}_C = \left\{ \Theta \in \mathbb{R}^{C \times C} : \Theta = \Theta^\intercal \right\} .$$

The dimension of the manifold $\mathcal{M}_C$, and its tangent space $T_\Sigma \mathcal{M}_C$, is $m = C(C+1)/2$.

The mapping from a point Θ_i of the tangent space to the manifold is called the exponential mapping $\mathrm{Exp}_\Sigma(\Theta_i)$: $T_\Sigma \mathcal{M}_C \to \mathcal{M}_C$ and is defined as:

$$\mathrm{Exp}_\Sigma(\Theta_i) = \Sigma^{\frac{1}{2}} \, \mathrm{Exp}(\Sigma^{-\frac{1}{2}} \Theta_i \Sigma^{-\frac{1}{2}}) \Sigma^{\frac{1}{2}} . \tag{7}$$

Its inverse mapping, from the manifold to the tangent space is called the logarithmic mapping $\mathrm{Log}_\Sigma(\Sigma_i)$: $\mathcal{M}_C \to T_\Sigma \mathcal{M}_C$ and is defined as:

$$\mathrm{Log}_\Sigma(\Sigma_i) = \Sigma^{\frac{1}{2}} \, \mathrm{Log}(\Sigma^{-\frac{1}{2}} \Sigma_i \Sigma^{-\frac{1}{2}}) \Sigma^{\frac{1}{2}} . \tag{8}$$

$\mathrm{Exp}(\cdot)$ and $\mathrm{Log}(\cdot)$ are the matrix exponential and matrix logarithm respectively. These two mappings are illustrated in Fig. 3. The computation of these operators is straightforward for SPD matrices of $\mathcal{M}_C$. They are obtained from their eigenvalue decomposition:

$$\begin{aligned}
\Sigma &= U \, \mathrm{diag}(\lambda_1, \dots, \lambda_C) \, U^\intercal , \\
\mathrm{Exp}(\Sigma) &= U \, \mathrm{diag}(\log(\lambda_1), \dots, \log(\lambda_C)) \, U^\intercal , \\
\mathrm{Log}(\Sigma) &= U \, \mathrm{diag}(\exp(\lambda_1), \dots, \exp(\lambda_C)) \, U^\intercal ,
\end{aligned}$$

where $\lambda_1, \dots, \lambda_C$ are the eigenvalues and U is the matrix of eigenvectors of Σ. As any SPD matrix can be diagonalized with strictly positive eigenvalues, $\mathrm{Log}(\cdot)$ is always defined. Similarly the square root $\Sigma^{\frac{1}{2}}$ is obtained as:

$$\Sigma^{\frac{1}{2}} = U \, \mathrm{diag}(\lambda_1^{\frac{1}{2}}, \dots, \lambda_C^{\frac{1}{2}}) \, U^\intercal ,$$

and is unique. The same goes for $\Sigma^{-\frac{1}{2}}$.

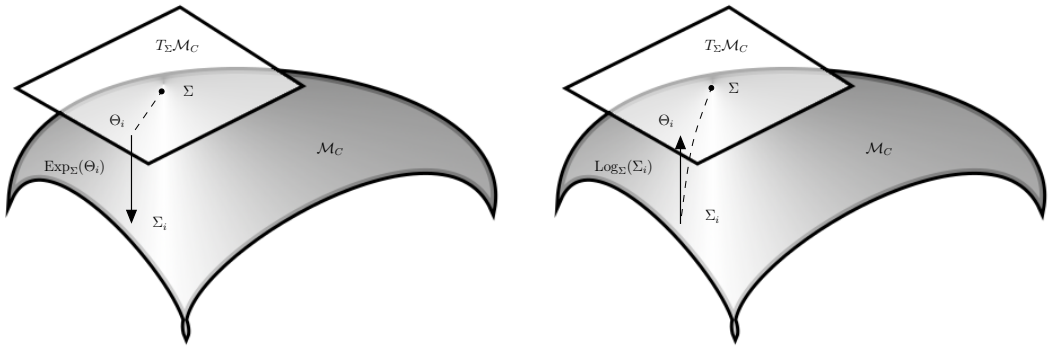


Figure 3: Left: the exponential mapping of Θ_i, an element of the tangent space $T_\Sigma\mathcal{M}_C$ at point Σ, on the manifold $\mathcal{M}_C$ is Σ_i. Right: the logarithmic mapping of Σ_i at point Σ is Θ_i on the tangent space $T_\Sigma\mathcal{M}_C$.

4.2 Estimators of covariance matrices

Estimation of covariance matrices is a critical step required to design a working and efficient BCI system. The covariance matrix is defined as $\Sigma = E\{(\boldsymbol{x} - \omega)(\boldsymbol{x} - \omega)^\intercal\}$ and is unknown. Only an estimate $\hat{\Sigma}$ could be computed from the observations. The first step is to choose an appropriate estimator. It is crucial to verify that the obtained covariance matrices fulfill the following properties: they should be accurate, symmetric, positive-definite, and well conditioned. The second step is to remove the outliers properly, that is the samples that are contaminated either by exogenous or endogenous noise, to avoid bias when estimating the mean of a class or when processing newly recorded EEG signals.

Regarding the first step, that is, the covariance estimation, an important property is the matrix conditioning. It requires that the ratio between the maximum and minimum eigenvalue is not too large. Moreover, to ensure the computational stability of the algorithm, the estimator should provide full-rank matrices, and its inversion should not amplify estimation errors.

This section describes three different classes of estimators: the sample, the shrinkage, and the fixed-point estimators. These classes of estimators are evaluated on a real EEG data set to assess their accuracy in terms of classification and the condition of the obtained matrices. The most common estimator is the Maximum Likelihood Estimator (MLE) under a multivariate Gaussian assumption. It is called the empirical *sample covariance matrix* (SCM), defined as:

$$\begin{aligned}\hat{\Sigma}_{\text{scm}} &= \frac{1}{N-1}\sum_{n=1}^{N}(x_n - \bar{x})(x_n - \bar{x})^\intercal \\ &= \frac{1}{N-1}X\left(\mathbf{I}_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\intercal\right)X^\intercal\,,\end{aligned} \tag{9}$$

where $\bar{x} \in \mathbb{R}^C$ is the sample mean vector $\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n$. In the matrix notation, $\mathbf{I}_N$ is the $N \times N$ identity matrix and $\mathbf{1}_N$ is the vector $[1, \ldots, 1]$. The SCM is often normalized (Fukunaga, 1990) as:

$$\hat{\Sigma}_{\text{nscm}} = \frac{C}{N}\sum_{n=1}^{N}\frac{(x_n - \bar{x})(x_n - \bar{x})^\intercal}{\sigma_{x_n}^2}\,, \tag{10}$$

with the inter-channel variance at time n defined as $\sigma_{x_n}^2 = (x_n - \bar{x})^\intercal(x_n - \bar{x})$. Other normalization techniques could be used. This estimation is fast and computationally simple. However when $C \approx N$, the SCM is not a good estimator of the true covariance. In the case $C > N$, the SCM is not even full rank.

To overcome the shortcomings of SCM, the shrinkage estimators have been developed as a weighted combination of the SCM and a target covariance matrix, which is often chosen to be close to the identity

matrix, that is, resulting from almost independent variables of unit variance:

$$\hat{\Sigma}_{\text{shrink}} = \kappa\Gamma + (1-\kappa)\hat{\Sigma}_{\text{scm}} \ , \tag{11}$$

where $0 \leqslant \kappa < 1$. This estimator provides a regularized covariance that outperforms the empirical $\hat{\Sigma}_{\text{scm}}$ for small sample size, that is $C \approx N$. The shrinkage estimator has the same eigenvectors as the SCM, but the extreme eigenvalues are modified, that is, the estimator is shrunk or elongated toward the average. The different shrinkage estimators differ in their definition of the target covariance matrix Γ. Ledoit and Wolf (2004) ($\hat{\Sigma}_{\text{shrink\_ledoit}}$ on Fig. 9) have proposed $\Gamma = v\mathbf{I}_C$, with $v = \text{Tr}(\hat{\Sigma}_{\text{scm}})$. Blankertz et al. (2011) ($\hat{\Sigma}_{\text{shrink\_blank}}$) defines Γ also as $v\mathbf{I}_C$ but with $v = \frac{\text{Tr}(\hat{\Sigma}_{\text{scm}})}{C}$. Schäfer ($\hat{\Sigma}_{\text{shrink\_schaf}}$) proposes several ways of defining Γ depending on the observed $\hat{\Sigma}_{\text{scm}}$ (Schäfer and Strimmer, 2005).

The fixed- covariance matrix (Pascal et al., 2005) is based on an M-estimator, which generalizes the MLE, and is a solution to the following equation:

$$\hat{\Sigma}_{\text{fp}} = \hat{\ell} = \frac{C}{N}\sum_{n=1}^{N}\left(\frac{(x_n-\bar{x})(x_n-\bar{x})^\intercal}{(x_n-\bar{x})^\intercal\hat{\ell}^{-1}(x_n-\bar{x})}\right) \ . \tag{12}$$

As there is no closed form expression to Eq. (12), it can be written as a function of $\hat{\ell}$: $g(\hat{\ell}) = \hat{\Sigma}_{\text{fp}}$. g admits a single *fixed point* $\hat{\ell}^*$, where $g(\hat{\ell}^*) = \hat{\ell}^*$, which is a solution to Eq. (12). Using $\hat{\ell}_0 := \hat{\Sigma}_{\text{nscm}}$ as the initial value of $\hat{\ell}$, it is solved recursively as $\hat{\ell}_t \underset{t\to\infty}{\longrightarrow} \hat{\ell}^*$.

4.3 Distances and means

From the definition of Riemannian manifolds, we have seen that they are defined by their atlas and by the choice of an inner product defined on the tangent space. In the case of covariance matrices, there exist several candidate inner products that lead to different distances and , thus to different geometries of Riemannian manifolds. A distance function $d : \mathcal{M}_C \times \mathcal{M}_C \to \mathbb{R}^+$ has the properties listed in Table 2 for all $\Sigma_1, \Sigma_2, \Sigma_3 \in \mathcal{M}_C$.

(i) Non-negativity	$d(\Sigma_1, \Sigma_2) \geq 0$
(ii) Identity	$d(\Sigma_1, \Sigma_2) = 0$ iff $\Sigma_1 = \Sigma_2$
(iii) Symmetry	$d(\Sigma_1, \Sigma_2) = d(\Sigma_2, \Sigma_1)$
(iv) Triangular inequality	$d(\Sigma_1, \Sigma_3) \leq d(\Sigma_1, \Sigma_2) + d(\Sigma_2, \Sigma_3)$

Table 2: Properties of a distance.

Divergences are very similar to distances, with the difference that properties (iii) and (iv) do not have to be satisfied. In the context of SPD matrices, divergences and distances are carefully chosen to induce a Riemannian metric. Divergences offer interesting properties but do not lead to qualitatively different results. For the sake of clarity, this chapter focuses only on distances; an in-depth review of the existing divergence and the analysis of their results are available in Kalunga et al. (2015).

As we will see in Section 4.4, the notion of mean, or center of mass, of a set of SPD matrices is tightly linked with the task of classification in machine learning approaches. Given a set of covariance matrices $\{\Sigma_i\}_{i=1,\dots,I}$, the mean $\bar{\Sigma}$ of those SPD matrices is a covariance matrix that minimizes the dispersion of matrices Σ_i:

$$\bar{\Sigma} = \mu(\{\Sigma_1, \dots, \Sigma_I\}) = \arg\min_{\Sigma\in\mathcal{M}_C} \sum_{i=1}^{I} d^2(\Sigma_i, \Sigma) \ , \tag{13}$$

where $d(\cdot,\cdot)$ is a distance between two matrices. In the literature, $\bar{\Sigma}$ could be referred to as the geometric mean, the *Cartan mean*, the *Frechet mean*, or the *Karcher mean* [4] (Ando et al., 2004; Lim and Pálfia,

[4] This appellation has been recently criticized by Karcher himself (Karcher, 2014).

2012). Depending on the distance used, several means can be defined from (13). Hereafter, some of the existing distances are briefly presented, along with their associated mean, and they are summarized in Table 3.

A trivial choice is the Euclidean distance, which is not a Riemannian distance, and is derived from the Frobenius inner product:

$$d_{\mathrm{E}}(\Sigma_1, \Sigma_2) = \|\Sigma_1 - \Sigma_2\|_F \,. \tag{14}$$

Euclidean distance yields the arithmetic mean:

$$\bar{\Sigma}_{\mathrm{E}} = \frac{1}{I} \sum_{i=1}^{I} \Sigma_i \,. \tag{15}$$

Averaging covariance matrices with this arithmetic mean is not adequate in the space of SPD matrices for two main reasons: first, the Euclidean distance and averaging do not guarantee invariance under inversion: a matrix and its inverse are supposed to be at the same distance from the identity matrix. Second, the Euclidean averaging of covariance SPD leads to a *swelling effect*: the determinant of the arithmetic mean of SPD matrices can be larger than the determinant of its individual components, as illustrated in Fig. 4. Since the determinant of a covariance matrix is a direct measure of the dispersion of the multivariate variable, the swelling effect introduces a large distortion on the dispersion of data (Arsigny et al., 2007). For these reasons, other means that adapt to the convex cone of SPD matrices are more adequate than this arithmetic mean.

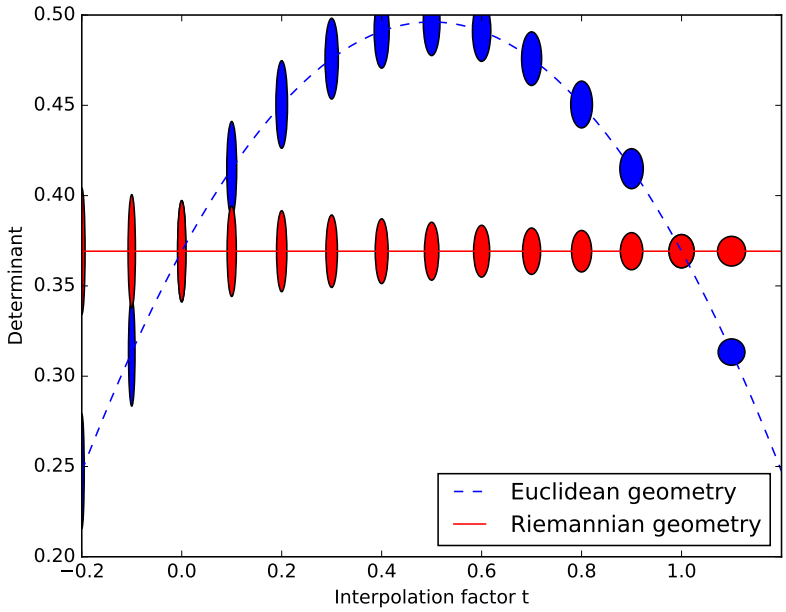


Figure 4: Evolution of the determinant of a matrix along a geodesic. Both matrices being interpolated have the same determinant (for $t = 0$ and $t = 1$); and along a Riemannian geodesic (in red), the determinant remains the same, but along a Euclidean geodesic (in blue), the determinant rises. This latter phenomenon is called the swelling effect.

To avoid these problems, a natural approach is to consider a distance built on curves from the manifold, called geodesics; Pennec and Ayache (1998) give a method to generate invariant distance on the manifold). The tangent vector of the geodesic $\gamma(t)$ between Σ_1 and Σ_2, where $\gamma(0) = \Sigma_1$ and $\gamma(1) = \Sigma_2$ is defined as $v = \overrightarrow{\Sigma_1 \Sigma_2} = \mathrm{Log}_{\Sigma_1}(\Sigma_2)$. A natural distance between Σ_1 and Σ_2 can thus be

defined as Moakher (2005):

$$d_{\text{AIR}}(\Sigma_1, \Sigma_2) = \| \operatorname{Log}(\Sigma_1^{-1/2}\Sigma_2\Sigma_1^{-1/2})\|_F = \left(\sum_{c=1}^{C} \log^2 \lambda_c \right)^{1/2} , \tag{16}$$

where λ_c, $c = 1, \ldots, C$, are the eigenvalues of $\Sigma_1^{-1/2}\Sigma_2\Sigma_1^{-1/2}$. This distance is known as the *affine-invariant Riemannian* (AIR) distance. Inserting (16) in (13) yields the mean $\bar{\Sigma}_{\text{AIR}}$ associated to the affine-invariant Riemannian metric:

$$\sum_{i=1}^{I} \operatorname{Log}(\bar{\Sigma}_{\text{AIR}}^{-1/2}\Sigma_i\bar{\Sigma}_{\text{AIR}}^{-1/2}) = 0 . \tag{17}$$

It has no closed form solution and can be solved iteratively through a gradient descent algorithm (Fletcher et al., 2004). This AIR distance has many properties (Bhatia, 2009, chap 6), and the most interesting one is the invariance under congruent transformation:

$$d_{\text{AIR}}(\Sigma_1, \Sigma_2) = d_{\text{AIR}}(W^\intercal\Sigma_1 W, W^\intercal\Sigma_2 W) , \tag{18}$$

for any invertible matrix $W \in \mathbb{R}^{C\times C}$. This property is crucial, since it shows that a full-rank spatial filter has no influence on Riemannian distance. Using Riemannian geometry, calculus is already performed on the optimal feature space.

Another distance is the Log-Euclidean, which was introduced to alleviate the complexity involved in the computation of the affine-invariant Riemannian distance and its related mean (Arsigny et al., 2007). The mean associated to the Log-Euclidean distance corresponds to an arithmetic mean in the domain of matrix algorithm. The distance between two SPD matrices is expressed as:

$$d_{\text{LE}}(\Sigma_1, \Sigma_2) = \|\operatorname{Log}(\Sigma_1) - \operatorname{Log}(\Sigma_2)\|_F , \tag{19}$$

and its associated mean is defined explicitly:

$$\bar{\Sigma}_{\text{LE}} = \operatorname{Exp}\left(\frac{1}{I} \sum_{i=1}^{I} \operatorname{Log}(\Sigma_i) \right) . \tag{20}$$

Note that d_{LE} can be interpreted at the Euclidean distance computed on the tangent plan at the identity matrix. Hence, this distance can be generalized as in (Yger and Sugiyama, 2015) to other reference points.

The distances presented above are summed up the following Table 3.

	Distance	Mean	Reference
Euclidean	$d_{\text{E}} = \|\Sigma_1 - \Sigma_2\|_F$	$\bar{\Sigma}_{\text{E}} = \frac{1}{I}\sum_{i=1}^{I}\Sigma_i$	
Log-Euclidean	$d_{\text{LE}} = \|\operatorname{Log}(\Sigma_1) - \operatorname{Log}(\Sigma_2)\|_F$	$\bar{\Sigma}_{\text{LE}} = \operatorname{Exp}\left(\frac{1}{I}\sum_{i=1}^{I}\operatorname{Log}(\Sigma_i)\right)$	Arsigny et al. (2007)
Affine-invariant	$d_{\text{AIR}} = \|\operatorname{Log}(\Sigma_1^{-1}\Sigma_2)\|_F$	Algorithm 3 in Fletcher and Joshi (2004)	Moakher (2005)

Table 3: Some of the distances and means considered in this chapter.

4.4 Minimum distance to mean classifier

The distances seen in the previous section allow one to take into account the curvature of the space of SPD matrices. Nonetheless, to be able to exploit those Riemannian metric in the context of BCI, one

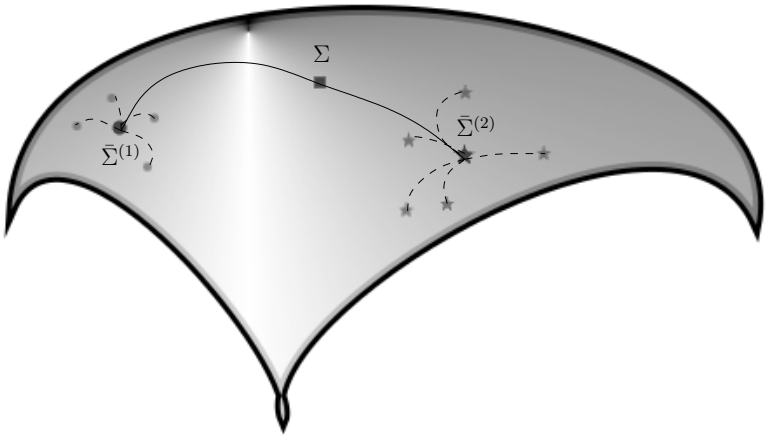


Figure 5: Illustration of a two-classes classification in the space of SPD matrices with the MDM algorithm.

needs to design an algorithm to discriminate between the different patterns in brain signals. One of the most simple classification algorithms consists in assigning a previously unseen signal to the class with the closest mean. This implies a computation of means of classes and a measure of distances from the means.

The classifier *Minimum Distance to Mean* (MDM), introduced in Barachant et al. (2010, 2012), is presented for multi-class classification[5]. It is a simple Bayesian classifier, under the hypotheses that classes have identical dispersions and that it is operating on a manageable space. The covariance matrices of EEG trials are classified based on their distance to the $k = 1 \dots K$ centers of the classes $\bar{\Sigma}^{(k)}$, that is, means, medians or centroids, as illustrated in Fig. 5. The predicted class k^* of the current matrix Σ is defined as:

$$k^* = \arg\min_k \; d(\Sigma, \bar{\Sigma}^{(k)}) \; . \tag{21}$$

As described in Algorithm 1, from I labeled training trials $\{X_i\}_{i=1}^{I}$ recorded per subject, K centers of classes $\bar{\Sigma}^{(k)}$ are estimated (step 3, detailled in Algorithm 1) . A new unlabeled test trial Z is predicted to belong to the class whose mean $\bar{\Sigma}^{(k)}$ is the closest to the trial covariance matrix, with respect to one of the distances from Table 3 (step 2 of Algorithm 2).

This very simple classifier has outperformed classical approaches based on spatial filtering and machine learning classifiers (Barachant et al., 2013a,b). Note that MDM is a classifier that can be applied offline or block-online (thus, in the experiments, these two cases will be grouped), but not online.

Algorithm 1 Offline Estimation of Riemannian Centers of Classes

Inputs: $X_i \in \mathbb{R}^{C \times N}$, for $i = 1, \dots, I$, a set of labeled trials.
Inputs: $\mathcal{I}(k)$, a set of indices of trials belonging to class k.
Output: $\bar{\Sigma}^{(k)}$, $k = 1, \dots, K$, centers of classes.

1: Compute covariance matrices Σ_i of X_i
2: **for** k = 1 **to** K **do**
3: $\quad \bar{\Sigma}^{(k)} = \mu(\{\Sigma_i \mid i \in \mathcal{I}(k)\})$, Eq. (13)
4: **end**
5: **return** $\bar{\Sigma}^{(k)}$

[5] A sample code is available at https://git.io/vDVRB.

Algorithm 2 Minimum Distance to Mean

Inputs: $\bar{\Sigma}^{(k)}$, K centers of classes from Algorithm 1.
Input: $Z \in \mathbb{R}^{C \times N}$, an unlabeled test trial.
Output: k^*, the predicted label of Z.

1: Compute covariance matrix Σ of Z
2: $k^* = \operatorname{argmin}_k d(\Sigma, \bar{\Sigma}^{(k)})$
3: **return** k^*

4.5 MDM for SSVEP

The MDM has been extended in Congedo et al. (2013); Congedo (2013) for possible offline applications on SSVEP signals. For SSVEP classification, $K = F + 1$ classes are considered: one class for each target frequency, and one for the resting state.

To embed frequency information of SSVEP in the covariance matrices, we use a construction of matrices proposed in Congedo et al. (2013). Let $X \in \mathbb{R}^{C \times N}$ be an EEG trial measured on C channels and N samples in an SSVEP experiment with F stimulus blinking at different frequencies. The covariance matrices are estimated[6] from a modified version of the input signal X:

$$X \in \mathbb{R}^{C \times N} \rightarrow \begin{bmatrix} X_{\text{freq}_1} \\ \vdots \\ X_{\text{freq}_F} \end{bmatrix} \in \mathbb{R}^{FC \times N} , \tag{22}$$

where X_{freq_f} is the input signal X band-pass filtered around frequency freq_f, $f = 1, \ldots, F$. Henceforth, all EEG signals will be considered as filtered and modified by Eq. (22), as shown in Fig. 6. The associated covariance matrix $\Sigma \in \mathcal{M}_{FC}$ (i.e., a manifold of dimension $m = FC(FC+1)/2$) is estimated using the Schäfer shrinkage estimator (Schäfer and Strimmer, 2005) which was experimentally found to be the most adequate for the set of data used (Kalunga et al., 2016). These matrices are shown on Fig. 7.

4.6 Online MDM

The MDM algorithm as described above is suitable for offline and block-online BCI settings. Covariance matrices were computed from a reference time given by the cue onsets used to locate SSVEP occurrences. However, in an online and asynchronous setup, there is no cue onset, and EEG epochs are thus classified on the fly. In this section we present an MDM online algorithm suitable for asynchronous BCI Kalunga et al. (2016).

The algorithm identifies a period (i.e., time interval) in the online EEG $\chi \in \mathbb{R}^{FC \times \mathcal{N}}$, where $\mathcal{N}$ is the number of recorded samples, associated with a high probability (above threshold) of observing an SSVEP at a specific frequency, as illustrated in Algorithm 3.

To locate this interval, we focus on the last D recorded EEG overlapping epochs $\left\{X_j \in \mathbb{R}^{FC \times w}\right\}_{j \in \mathcal{J}(d)}$, with the set of indices $\mathcal{J}(d) = d - D + 1, \ldots, d-1, d$; where d is the index of the current epoch X_d in the online recording $\chi(n)$. Epochs have size w, and the interval between two consecutive epochs is Δn, with $w > \Delta n$:

$$X_d = \chi(n - w, \ldots, n) . \tag{23}$$

To obtain the first D epochs $X_{j \in \mathcal{J}(d)}$, at least $w + (D-1)\,\Delta n$ samples of χ should be recorded (step 4).

The classification outputs $k^*_{j \in \mathcal{J}(d)}$ obtained in step 3 by applying Algorithm 2 on $X_{j \in \mathcal{J}(d)}$ are stored in a vector $\mathcal{K}$, which always contains the latest D classification outputs. The class that occurs the most in $\mathcal{K}$ (step 5), with an occurrence probability $\rho(k)$ above a defined threshold ϑ, is considered to be the class, denoted $\bar{k}$, of the ongoing EEG recording $\chi(n)$. The vector ρ is defined as

$$\rho(k) = \frac{\#\{k^*_{j \in \mathcal{J}(d)} = k\}}{D}, \text{ for } k = 1, \ldots, K, \tag{24}$$

[6] A code example is available at https://git.io/vDVRI

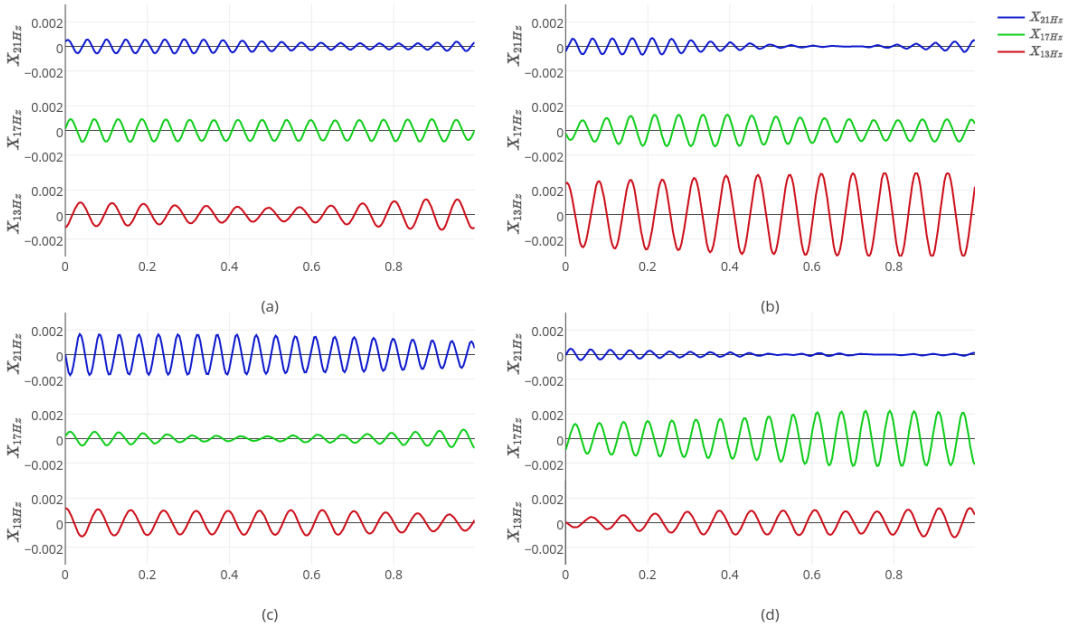


Figure 6: Samples of EEG trials from Eq. (22). For each sub-band X_{freq_f} only channel O_z is shown. Each subplot shows the first seconds of a trial from classes: (a) resting-state, (b) 13 Hz, (c) 21 Hz, (d) 17 Hz.

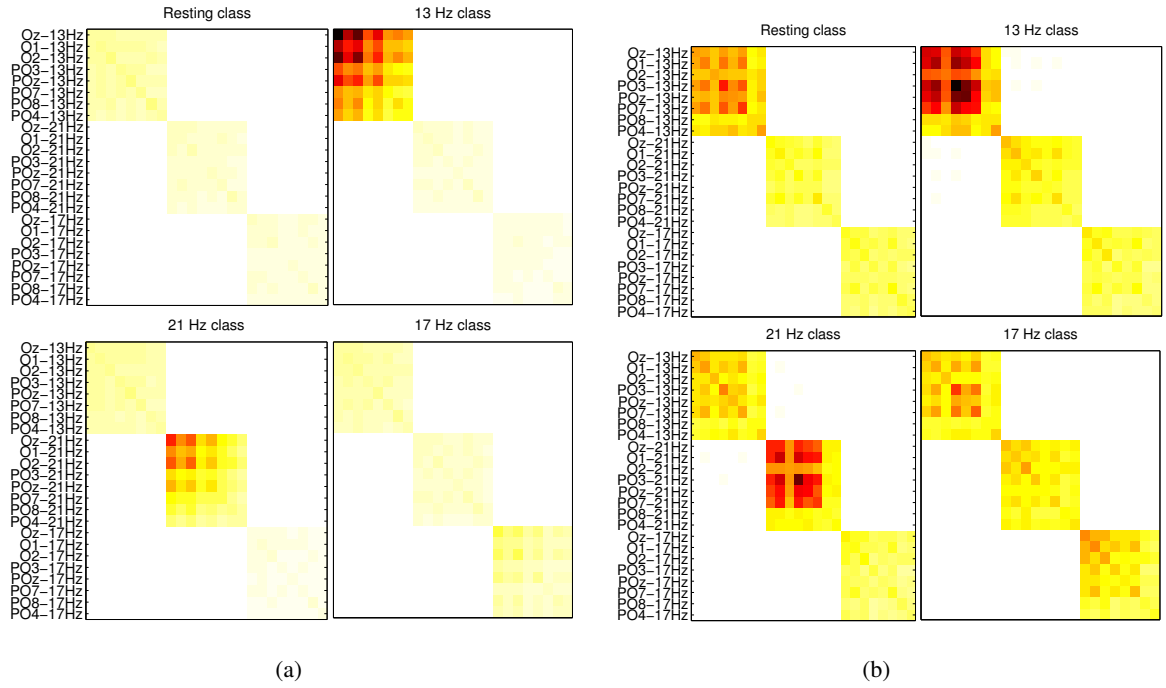


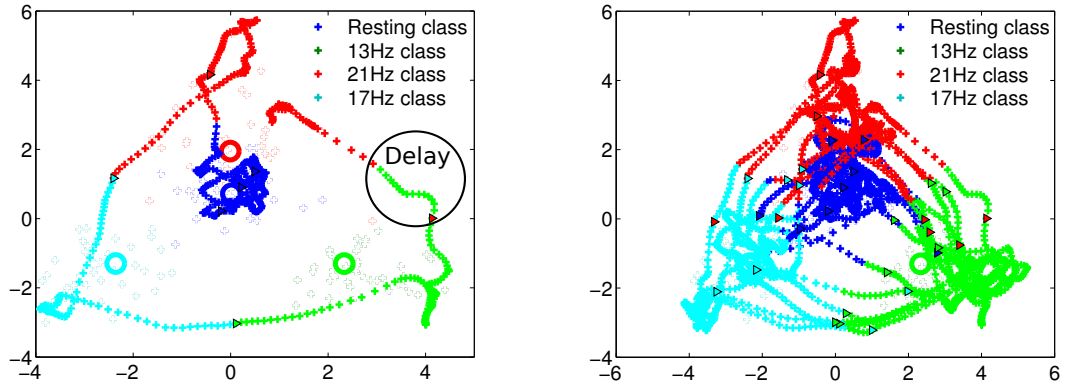
Figure 7: Representation of covariance matrices: each image is the covariance matrix mean $\bar{\Sigma}^{(k)}$ of the class k, for one session of the recording. The diagonal blocks show the covariance in different frequency bands, that is, 13 Hz in the upper-left block, 21 Hz in the middle, and 17 Hz in the bottom-right. Subjects with highest (a) and lowest (b) BCI performance.

with $\bar{k} = \arg\max_k \rho(k)$; then $\rho(\bar{k})$ is compared to the threshold ϑ. If ϑ is not reached within the last D epochs, the classification output is held back, and the sliding process continues until ϑ is reached. In the last D epochs, once a class $\bar{k}$ has been identified, a curve direction criterion is introduced to enforce the robustness of the result. For class $\bar{k}$ to be validated, this criterion requires that the direction taken by the displacement of covariance matrices $\Sigma_{j\in\mathcal{J}(d)}$ be toward the center of class $\bar{\Sigma}^{(\bar{k})}$. Hence, $\widetilde{\delta}_{\bar{k}}$, the sum of gradients (i.e., differentials) of the curve made by distances from $\Sigma_{j\in\mathcal{J}(d)}$ to $\bar{\Sigma}^{(\bar{k})}$ should be negative (step 8):

$$\widetilde{\delta}_{\bar{k}} = \sum_{j\in\mathcal{J}(d)} \frac{\Delta\delta_{\bar{k}}(j)}{\Delta j} = \sum_{j=d-D+2}^{d} \delta_{\bar{k}}(j) - \delta_{\bar{k}}(j-1) \; < 0$$
$$\text{with} \quad \delta_{\bar{k}}(j) = \frac{\delta(\Sigma_j, \bar{\Sigma}^{(\bar{k})})}{\sum_{k=1}^{K} \delta(\Sigma_j, \bar{\Sigma}^{(k)})} \; . \tag{25}$$

The occurrence criterion is inspired by the dynamic stopping of Verschore et al. (2012); there is no fixed trial length for classification. The occurrence criterion ensures that the detected user intention is unaffected by any short time disturbances attributed to noise or subject inattention, as presented in Algorithm 3. This approach offers a good trade-off to obtain robust results within a short and flexible time.

The curve direction criterion solves both the problems of latency in the EEG synchronization and the problem of the delays inserted by the EEG epochs processing. Indeed, some EEG epochs gather signals from different classes, that is, intermediary states, and might be wrongfully classified if the decision is solely based on the distance with the center of the class. This situation and the effect of the curve direction criterion are shown in Fig. 8. The colors in the figure show the classification result before the curve direction criteria. Ensuring that the covariance matrices are displaced toward the center of the detected class provides a guarantee that it matches the current EEG state. Inversely, if the direction of the curve is moving away from the center of the detected class, it might indicate that there has been a change in the EEG state that has not been detected.



(a) (b)

Figure 8: Covariance matrices trajectory during a 4-class SSVEP online recording. The circles represent class centers. The triangles mark the beginning of a new trial in the experiment, whose true class is indicated by the triangle's color. The colors of the crosses show the class identified by MDM before the curve direction criterion is applied. 8(a) shows the first 7 trials. The first 3 trials are from the resting class, the remaining are respectively class 13, 17, and 21 Hz. 8(b) shows the entire recording. Data are taken from the subject with the highest BCI performance.

Algorithm 3 has four hyperparameters: w, Δn, D, and ϑ. For the results of online classification presented in Table 5, they are set through cross validation to: D=5, ϑ=0.7, $w = 2.6$, $\Delta n = 0.2$.

Although a large window size w is expected to increase the classification accuracy, it increases the response time, thus reducing the time resolution, and extends the overlap between different EEG states. The step size Δn should be set to a minimum value to allow a maximum number of overlapping epochs (D) within a short time. However, it should be large enough to avoid too many calculations within a time interval with small or inexistent changes in EEG states. If the number of epoch D is too small, the classification will be sensitive to nonintentional and abrupt changes in the EEG. A very large D will increase the momentum and reinforce the influence of the past EEG signals. It should also be mentioned that both the occurrence and the curve direction criteria cannot have a significant impact if the value of D is too small. The probability threshold parameter ϑ acts like a rejection parameter: high ϑ values correspond to high rejection rate.

Algorithm 3 Online MDM

Inputs: hyper-parameters w, Δn, D, and ϑ.
Inputs: $\bar{\Sigma}^{(k)}$, $k = 1, \ldots, K$, centers of classes from Algorithm 1 (offline training).
Inputs: Online EEG recording $\chi(n)$.
Output: $\tilde{k}(n)$, online predicted class.

$d = 1$
for $n = w$ **to** $\mathcal{N}$ **step** Δn
  Epoch X_d, Eq. (23), and classify it with Algorithm 2
  if $d \geq D$
    Find the most recurrent class in $\mathcal{K} = k^*_{j \in \mathcal{J}(d)} \mid \bar{k} = \arg\max_k \rho(k)$, Eq. (24)
    if $\rho(\bar{k}) > \vartheta$
      Compute $\widetilde{\delta}_{\bar{k}}$, Eq. (25)
      if $\widetilde{\delta}_{\bar{k}} < 0$
        return $\tilde{k} = \bar{k}$
      end
    end
  end
  $d = d + 1$
end

5 Experimental evaluation on SSVEP dataset

5.1 SSVEP dataset description

To assess the covariance estimators, a benchmark is proposed on real SSVEP data set; this data set is freely available[7]. The signals are recorded from 12 subjects during an SSVEP experiment. EEG is measured on $C = 8$ channels: O_Z, O_1, O_2, PO_Z, PO_3, PO_4, PO_7, and PO_8. The ground and the reference electrodes were placed on F_Z, and the right hear mastoid respectively. The acquisition rate is $T_s = 256$ Hz on a gTec MobiLab Amp (gTec, Graz, Austria). The subjects are presented with $F = 3$ visual target stimuli blinking respectively at freq $= 13$, 17, and 21 Hz. It is a $K = 4$ classes BCI setup made of the $F = 3$ stimulus classes and one resting class (no-SSVEP). In a session, which lasts 5 min, 32 trials are recorded: 8 for each visual stimulus and 8 for the resting class. The number of sessions recorded per subject varies from 2 to 5. Thus the longest EEG recorded for a single subject is 25 min or 160 trials. The trial length is 6 s, that is $N = 6 \times T_s = 1536$ samples. For each subject, a test set is made of 32 trials, whereas the remaining trials (which might vary from 32 to 128) make up the training set.

[7] Data could be freely downloaded from https://github.com/sylvchev/dataset-ssvep-exoskeleton.

5.2 Evaluation of the covariance estimators

The effectiveness of covariance matrix estimators is evaluated for SSVEP signals. The evaluation is done in terms of classification accuracy, and the conditioning of covariance matrices is also investigated. A bootstrapping with 1000 replications is performed to assess the performances of each estimator. Estimators are compared on 10 trial lengths $t \in \{0.5, 1.0, \dots 5.0\}$ seconds, as these are known to affect the estimators performance. Here $N \in \{128, 256, \dots, 1280\}$ is computed as $N = t \times T_s$.

Fig. 9 shows the classification accuracy of each estimator computed across all subjects. Even if the error bars show an important intersubject variability, the increase in the accuracy can be attributed to the fact that the relevant patterns in EEG accumulate with the trial length, producing better estimation of the covariance matrices. This is known to be particularly true for the SCM estimator, and it could be seen in Fig. 9. It appears that shrinkage estimators (especially Ledoit and Schäfer) are less affected by the reduction of epoch sizes than the other estimators. This is a direct consequence of the regularization between the sample covariance matrices and the targeted (expected) covariance matrix of independent variables.

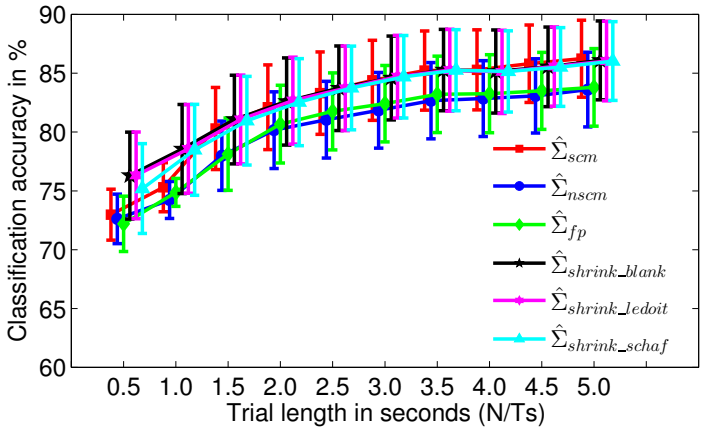


Figure 9: Comparison of covariance estimators in terms of classification accuracy obtained with offline MDRM with increasing EEG trial length. For each trial length, the accuracy mean+/-std across all subjects and across all replications is shown.

For computational purposes, it is important to look at the matrix conditioning. Fig. 10 shows the ratio $\mathcal{C}$ between the largest and smallest eigenvalues: in well-conditioned matrices, $\mathcal{C}$ is small. Shrinkage estimators offer better conditioned matrices whereas the SCM, NSCM, and fixed-point matrices are ill-conditioned below 2 s of trial length and may result in singular matrices.

It appears in Fig. 9 and 10 that the shrinkage estimator, especially the Ledoit-Wolfe and the Schäfer ones, are a good choice to obtain robust and accurate results, even if the covariances are estimated on a small number of samples. This is an important aspect, as the processing time induces a delay in the processing and should thus be reduced to provide responsive HMI.

5.3 Offline classification of SSVEP

Table 4 shows the offline classification accuracy for each subject obtained by the application of the MDM described in Section 4.4. The distances and mean listed in Table 3 are used: Euclidean, Log-Euclidean, and affine-invariant Riemannian. The performance of MDM approaches is compared to two CCA-based state-of-the-art methods proposed by (Lin et al., 2006) and (Nakanishi et al., 2014) respectively. In the

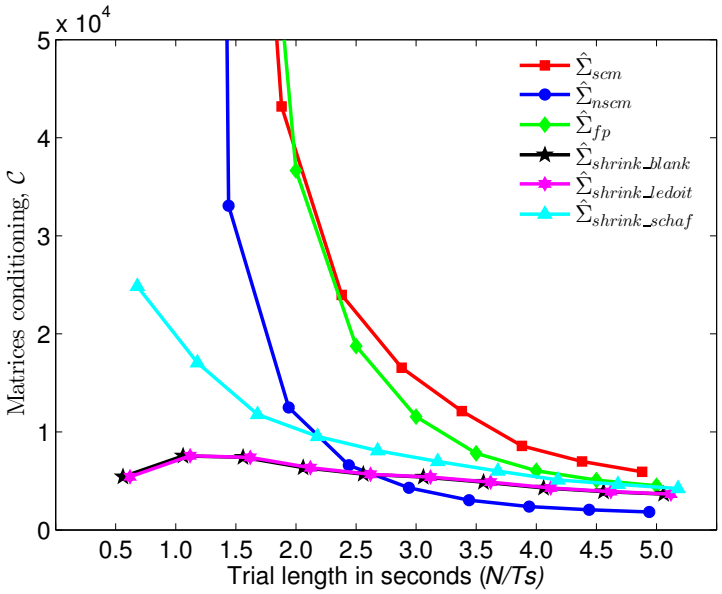


Figure 10: Covariance matrices condition expressed as the ratio $\mathcal{C}$ between largest and smallest eigenvalues for the different covariance estimators. The comparison is made with increasing EEG trial length.

implementation of these methods, the epochs are taken from $\tau_0 + 2$s, where τ_0 is the start time of the trial.

The MDM approach with Riemannian distances outperforms both CCA-based methods with an average classification accuracy of 89.2±4.9 % and an ITR of 15.52±5.2 bits/min for the Log-Euclidean, and 90.4±7.8 % and ITR of 16.3±5.3 bits/min for the AIR. The method by Lin et al. ranks third with 87.5% ±15.1 % and 15.5±6.8 bits/min. The method proposed by Nakanishi et al., which could be expected to achieve better results as reported in Nakanishi et al. (2014), only ranks fourth. This is mainly due to the fact that this method requires information on the phase of the stimuli. In fact, Nakanishi et al. use the average of all training trials belonging to a unique class as a reference signal in the CCA. When SSVEP trials belonging to a unique trial are not in-phase, which is the case in the current work, averaging them will cancel the signal. The MDM approach with Euclidean distance has the lowest performance. This shows how inappropriate it is to use Euclidean geometry on covariance matrices as the lie on a curved space.

	Offline algorithms for SSVEP classification									
	CCA				MDM					
	Lin *et al.*		Nakanishi *et al.*		Euclidean		Log-Euclidean		Affine-invariant	
	acc(%)	itr(bpm)	acc(%)	itr(bpm)	acc(%)	itr(bpm)	acc(%)	itr(bpm)	acc(%)	itr(bpm)
S1	91.7	16.3	67.6	3.5	71.6	6.6	86.2	13.1	84.7	12.22
S2	45.8	0.7	66.0	3.2	46.7	0.8	77.4	8.8	79.4	9.79
S3	100.0	23.8	90.2	10.3	83.6	11.6	99.2	22.7	99.3	22.7
S4	97.9	21.3	78.3	6.1	66.2	4.9	89.7	15.0	89.7	15.05
S5	83.3	11.5	76.0	5.5	46.5	0.8	85.7	12.7	89.5	14.94
S6	77.1	8.7	72.2	4.5	46.4	0.8	84.5	12.1	87.2	13.63
S7	98.6	22.0	90.0	10.2	78.4	9.1	98.8	22.1	99.8	23.5
S8	97.9	21.3	90.4	10.3	80.5	10.2	98.9	22.3	99.7	23.2
S9	91.7	16.3	64.0	2.8	68.2	5.5	83.3	11.5	85.8	12.8
S10	80.2	10.0	79.2	6.4	53.9	1.9	90.9	15.8	93.1	17.3
S11	89.6	15.0	54.8	1.4	56.2	2.4	77.4	8.8	78.2	9.2
S12	95.8	19.4	82.3	7.4	82.9	11.2	97.8	21.2	98.6	22.0
Mean	**87.5±15.1**	**15.5±6.8**	**81.2±14.1**	**11.8±6.0**	**65.1±9.5**	**5.5±4.2**	**89.2±4.9**	**15.52±5.2**	**90.4±7.8**	**16.3±5.3**

Table 4: Offline performance in terms of accuracy and ITR. Five methods are compared: (1) CCA approach introduced by Lin et al. (2006), (2) CCA approach introduced by Nakanishi et al. (2014), (3) MDM approach with Euclidean mean and distance, (4) MDM with Log-Euclidean mean and distance, and (5) MDM with affine-invariant distance and mean as described in Alg. (2).

	Online algorithms for SSVEP classification				
	CCA+SVM	FBCCA	Online MDM		
	Kalunga *et al.*	Chen *et al.*	Euclidean	Log-Euclidean	Affine-invariant
Sub.	acc (%)	acc (%)	acc (%)	acc (%)	acc (%)
1	54.68	75.00	53.12	71.88	73.44
2	37.50	41.67	43.75	78.13	79.69
3	89.06	85.42	67.19	85.94	85.93
4	79.69	97.92	54.68	84.38	87.50
5	50.00	64.58	37.50	62.50	68.75
6	87.50	75.00	34.37	84.38	85.94
7	77.08	80.56	60.42	87.50	88.54
8	73.44	72.92	67.19	90.63	92.19
9	60.94	66.67	57.81	70.31	70.31
10	67.97	65.62	38.28	75.00	80.47
11	71.88	64.58	48.44	60.94	65.63
12	95.63	80.83	71.25	96.25	96.69
Mean	**70.45±16.5**	**72.56±13.3**	**52.83±12.0**	**78.98±10.6**	**81.27±9.5**

Table 5: Subject classification accuracies (acc(%)) elapsed for the classification of a single trial. Classification is performed with MDM using either Euclidean or Riemannian means (see Table 3).

5.4 Online classification of SSVEP

For online implementation, the different algorithms are compared in terms of classification accuracy. The performances achieved using online MDM with the distances mentioned in Table 3 are compared to the two state-of-the-art methods, CCA+SVM and FBCCA, both described in Section 3.2. For these methods, the number of harmonics is set to $N_h = 6$. This value maximizes the classification accuracy of standard CCA in an offline analysis on training data. In CCA+SVM, the EEG is filtered between 13 and 126 Hz to accommodate all six harmonics. In FBCCA, the N_b sub-bands are constructed such that the nth sub-band starts from $n \times 8$ Hz and ends at 93 Hz, that is, 10 times the stimuli frequency range of 8 Hz (i.e., 13 to 21 Hz). Parameters a, b and N_b from Eq. (6) and (5) were set to 2, 0, and 3 respectively, through a grid search where their values were respectively limited to $a = 0.25 \times i_a$, with $i_a = 0, 1, \ldots, 40$, $b = 0.25 \times i_b$, with $i_b = 0, 1, \ldots, 4$, and $N_b = i_N$, with $i_N = 1, \ldots, 7$.

Table 5 summarizes results obtained for each subject and each method. Using AIR distance significantly improves classification performances (81.27%), in comparison with the state-of-the-art CCA-based methods that have an average performance across subjects of 70.45% and 72.56% for CCA+SVM and FBCCA respectively, and in comparison to the MDM with Euclidean distance that has an average performance across subjects of 52.83%. Relying on CCA coefficients for classification, as in the classic CCA method (Lin et al., 2006) or as in FBCCA (Chen et al., 2015a), has a strong limitation as it could not account for the reject class. As the reject class does not have a specific reference signal, it is not possible to determine the correlation coefficients associated with this class. This limitation disqualifies the FBCCA for any real implementation of BCI and avoids confronting the most challenging case (identifying when the user does not look at any stimulus) from a machine learning perspective. Nonetheless, to propose a thorough comparison with the existing approaches, we provide the results obtained with FBCCA without including the resting-state (no-SSVEP) class.

This experiment on real EEG data shows that it is crucial to process covariance matrices with dedicated Riemannian tools, affecting the efficiency of the classification. The obtained results show that the simple MDM classification scheme used within the Riemannian framework outperforms CCA-based state-of-the-art methods for SSVEP classification. With only four classes (three SSVEP stimulation frequencies and a reject class), this experiment does not aim at improving the ITR of the BCI system.

6 Conclusions

In this chapter, we have reviewed the different algorithms and implementations for feature extraction in SSVEP-based BCI. The spatial covariance, which describes the relative changes observed between the electrodes, plays a key role in the success of the state-of-the-art approaches. This observation also applies for several EEG-based BCIs, such as systems relying on ERP or MI. SSVEP offers promising results, as it is suggested by the fact that the highest ITR reported in the literature are using SSR. It is thus important to use the appropriate tools and algorithms when building up such a BCI system, to ensure that all the signal processing chain could work in online or block-online settings.

To estimate correctly the covariance from the data, a trade-off should be sought between accuracy and temporal precision: using some large time windows to estimate covariance allows to compute accurate and well-conditioned matrices at the expense of time resolution. Using an adequate estimator, it is possible to use smaller time windows (on the order of 1 or 2 s) for estimating covariance, while avoiding ill-conditioned matrices. One should keep in mind that this trade-off has a direct impact on the system reactivity and on the precision for online settings. The delay introduced by the covariance estimation lower the ITR: this has a strong incidence on the HMI aspect and it should be taken into account when designing experiments.

In the BCI literature, most of the studies propose offline analyses, which are of interest for a better understanding of the neurological phenomena. But offline implementations are not suitable to design a BCI as they need information from the whole session to extract features and to provide classification decisions. Converting offline algorithms to online ones is not an easy task and it systematically has a negative impact on the classification accuracy.

A common approach is to rely on spatial filters, that are carefully tuned during a calibration session. The system could then work online, applying the learned spatial filters on the acquired signal. The main disadvantage of these filters is that they are user specific and session specific. Moreover, these filters are best exploited when dealing with a clean signal, without artifacts, constraining the experiments to be conducted in a strongly controlled environment. Riemannian approaches offer different methods, that are efficient and more robust to noise. The algorithms could be relatively simple, such as the MDRM classifiers. Note that the online MDM introduced here for SSVEP data could be applied identically to MI-based BCI.

Future work on this topic could range from very practical works and theoretical advances. For example, it is possible to improve the MDM classifier, taking into account various information about the subject. Another possibility is to investigate the distribution of covariance matrices on the manifold, as suggested in the work of Zanini et al. (2016) or of Gayraud et al. (2016). From a machine learning point of view, there are still many open questions on the possibilities to include transfer learning to cope with session-to-session or inter-subject variability (Waytowich et al., 2016).

References

Absil, P.-A., Mahony, R., and Sepulchre, R. (2009). *Optimization algorithms on matrix manifolds*. Princeton University Press.

Ando, T., Li, C.-K., and Mathias, R. (2004). Geometric means. *Linear algebra and its applications*, 385:305–334.

Arsigny, V., Fillard, P., Pennec, X., and Ayache, N. (2007). Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM Journal on Matrix Analysis and Applications*, 29(1):328–347.

Barachant, A., Andreev, A., and Congedo, M. (2013a). The Riemannian potato: an automatic and adaptive artifact detection method for online experiments using Riemannian geometry. In *Proceedings of TOBI Workshop IV*, pages 19–20.

Barachant, A., Bonnet, S., Congedo, M., and Jutten, C. (2010). Riemannian geometry applied to BCI classification. In *Latent Variable Analysis and Signal Separation*, pages 629–636. Springer.

Barachant, A., Bonnet, S., Congedo, M., and Jutten, C. (2012). Multiclass brain-computer interface classification by Riemannian geometry. *Biomedical Engineering, IEEE Transactions on*, 59(4):920–928.

Barachant, A., Bonnet, S., Congedo, M., and Jutten, C. (2013b). Classification of covariance matrices using a Riemannian-based kernel for BCI applications. *Neurocomputing*, 112:172–178.

Barthélemy, Q., Mayaud, L., Renard, Y., Kim, D., Kang, S., Gunkelman, J., and Congedo, M. (2017). Online denoising of eye-blinks in electroencephalography. *Neurophysiologie Clinique / Clinical Neurophysiology*, 47:371–391.

Bhatia, R. (2009). *Positive definite matrices*. Princeton university press.

Blankertz, B., Lemm, S., Treder, M., Haufe, S., and Müller, K.-R. (2011). Single-trial analysis and classification of ERP components: a tutorial. *NeuroImage*, 56(2):814–825.

Chen, X., Wang, Y., Gao, S., Jung, T.-P., and Gao, X. (2015a). Filter bank canonical correlation analysis for implementing a high-speed ssvep-based brain–computer interface. *Journal of Neural Engineering*, 12(4):046008.

Chen, X., Wang, Y., Nakanishi, M., Gao, X., Jung, T.-P., and Gao, S. (2015b). High-speed spelling with a noninvasive brain–computer interface. *Proceedings of the national academy of sciences*, 112(44):E6058–E6067.

Congedo, M. (2013). *EEG Source Analysis*. Habilitation à diriger des recherches, Université de Grenoble.

Congedo, M., Barachant, A., and Andreev, A. (2013). A new generation of brain-computer interface based on Riemannian geometry. *arXiv preprint arXiv:1310.8115*.

Congedo, M., Barachant, A., and Bathia, R. (2017). Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review. *Brain-Computer Interfaces*, pages 1–20.

Fisher, R. S., Harding, G., Erba, G., Barkley, G. L., and Wilkins, A. (2005). Photic- and pattern-induced seizures: A review for the epilepsy foundation of america working group. *Epilepsia*, 46(9):1426–1441.

Fletcher and Joshi, S. (2004). Principal Geodesic Analysis on Symmetric Spaces: Statistics of Diffusion Tensors. In *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis*, volume 3117 of *LNCS*, pages 87–98. Springer.

Fletcher, P. T., Lu, C., Pizer, S. M., and Joshi, S. (2004). Principal geodesic analysis for the study of nonlinear statistics of shape. *Medical Imaging, IEEE Transactions on*, 23(8):995–1005.

Fukunaga, K. (1990). *Introduction to statistical pattern recognition*. Academic press.

Gayraud, N., Foy, N., and Clerc, M. (2016). A separability marker based on high-dimensional statistics for classification confidence assessment. In *IEEE International Conference on Systems, Man, and Cybernetics*.

Hardoon, D. R., Szedmak, S. R., and Shawe-Taylor, J. R. (2004). Canonical Correlation Analysis: An Overview with Application to Learning Methods. *Neural Computation*, 16(12):2639–2664.

Herrmann, C. S. (2001). Human EEG responses to 1-100 hz flicker: resonance phenomena in visual cortex and their potential correlation to cognitive phenomena. *Experimental Brain Research*, 137:346–353.

Hotelling, H. (1936). Relations between two sets of variates. *Biometrika*, 28(3/4):321–377.

Jayasumana, S., Hartley, R., Salzmann, M., Li, H., and Harandi, M. (2013). Kernel methods on the Riemannian manifold of symmetric positive definite matrices. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 73–80. IEEE.

Kalunga, E., Djouani, K., Hamam, Y., Chevallier, S., and Monacelli, E. (2013). SSVEP enhancement based on Canonical Correlation Analysis to improve BCI performances. In *AFRICON, 2013*, pages 1–5.

Kalunga, E. K., Chevallier, S., Barthélemy, Q., Djouani, K., Hamam, Y., and Monacelli, E. (2015). From Euclidean to Riemannian Means: Information Geometry for SSVEP Classification. In Nielsen, F. and Barbaresco, F., editors, *Geometric Science of Information*, pages 595–604. Springer International Publishing.

Kalunga, E. K., Chevallier, S., Barthélemy, Q., Djouani, K., Monacelli, E., and Hamam, Y. (2016). Online SSVEP-based BCI using Riemannian geometry. *Neurocomputing*, 191:55–68.

Karcher, H. (2014). Riemannian center of mass and so called Karcher mean. *arXiv preprint arXiv:1407.2087*.

Ledoit, O. and Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of multivariate analysis*, 88(2):365–411.

Li, Y., Wong, K., and De Bruin, H. (2012). Electroencephalogram signals classification for sleepstate decision: A Riemannian geometry approach. *Signal Processing, IET*, 6(4):288–299.

Li, Y. and Wong, K. M. (2013). Riemannian Distances for Signal Classification by Power Spectral Density. *IEEE Journal of Selected Topics in Signal Processing*, 7(4):655–669.

Lim, Y. and Pálfia, M. (2012). Matrix power means and the Karcher mean. *Journal of Functional Analysis*, 262(4):1498–1514.

Lin, Z., Zhang, C., Wu, W., and Gao, X. (2006). Frequency Recognition Based on Canonical Correlation Analysis for SSVEP-Based BCIs. *Biomedical Engineering, IEEE Transactions on*, 53(12):2610–2614.

Liu, Q., Chen, K., Ai, Q., and Xie, S. (2014). Review: Recent development of signal processing algorithms for SSVEP-based brain computer interfaces. *Journal of Medical and Biological Engineering*, 34:299–309.

Lotte, F. and Guan, C. (2011). Regularizing Common Spatial Patterns to Improve BCI Designs: Unified Theory and New Algorithms. *Biomedical Engineering, IEEE Transactions on*, 58(2):355–362.

Moakher, M. (2005). A differential geometric approach to the geometric mean of symmetric positive-definite matrices. *SIAM Journal on Matrix Analysis and Applications*, 26(3):735–747.

Morgan, S. T., Hansen, J. C., and Hillyard, S. A. (1996). Selective attention to stimulus location modulates the steady-state visual evoked potential. *Proceedings of the National Academy of Sciences*, 93(10):4770–4774.

Müller, M. M., Andersen, S., Trujillo, N. J., Valdés-Sosa, P., Malinowski, P., and Hillyard, S. A. (2006). Feature-selective attention enhances color signals in early visual areas of the human brain. *Proceedings of the National Academy of Sciences*, 103(38):14250–14254.

Nakanishi, M., Wang, Y., Wang, Y.-T., Mitsukura, Y., and Jung, T.-P. (2014). A high-speed brain speller using steady-state visual evoked potentials. *International journal of neural systems*, 24(06):1450019.

Niedermeyer, E. and Lopes da Silva, F. (2004). *Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*. Lippincott Williams & Wilkins, 5th edition.

Pascal, F., Forster, P., Ovarlez, J. P., and Arzabal, P. (2005). Theoretical analysis of an improved covariance matrix estimator in non-gaussian noise. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP).*, volume 4.

Pastor, M. A., Artieda, J., Arbizu, J., Valencia, M., and Masdeu, J. C. (2003). Human cerebral activation during Steady-State Visual-Evoked responses. *The Journal of Neuroscience*, 23(37):11621–11627.

Pennec, X. and Ayache, N. (1998). Uniform distribution, distance and expectation problems for geometric features processing. *Journal of Mathematical Imaging and Vision*, 9(1):49–67.

Regan, D. (1982). Comparison of transient and steady-state methods. *Annals of the New York Academy of Sciences*, 388(1):45–71.

Schäfer, J. and Strimmer, K. (2005). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical applications in genetics and molecular biology*, 4(1).

Stawicki, P., Gembler, F., and Volosyak, I. (2017). *Brain Computer Interfaces Handbook: Technological and Theoretical Advances*, chapter Design and Development of User Friendly SSVEP-based BCI Applications for Elderly People. CRC Press.

Takahashi, T. (2004). *Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*, chapter Activation methods, pages 241–262. Lippincott Williams & Wilkins, 5th edition.

Tomioka, R., Aihara, K., and Müller, K.-R. (2007). Logistic regression for single trial EEG classification. In *NIPS*, volume 19, pages 1377–1384.

Verschore, H., Kindermans, P.-J., Verstraeten, D., and Schrauwen, B. (2012). Dynamic stopping improves the speed and accuracy of a P300 speller. In *Artificial Neural Networks and Machine Learning–ICANN 2012*, pages 661–668. Springer.

Vialatte, F.-B., Maurice, M., Dauwels, J., and Cichocki, A. (2010). Steady-state visually evoked potentials: Focus on essential paradigms and future perspectives. *Progress in Neurobiology*, 90(4):418–438.

Waytowich, N. R., Lawhern, V. J., Bohannon, A. W., Ball, K. R., and Lance, B. J. (2016). Spectral transfer learning using information geometry for a user-independent brain-computer interface. *Frontiers in Neuroscience*, 10.

Wolpaw, J., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., and Vaughan, T. M. (2002). Brain–computer interfaces for communication and control. *Clinical Neurophysiology*, 113(6):767–791.

Yger, F. (2013). A review of kernels on covariance matrices for BCI applications. In *Machine Learning for Signal Processing (MLSP), 2013 IEEE International Workshop on*, pages 1–6. IEEE.

Yger, F., Berar, M., and Lotte, F. (2017). Riemannian approaches in Brain-Computer Interfaces: a review. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, to appear.

Yger, F. and Sugiyama, M. (2015). Supervised LogEuclidean metric learning for symmetric positive definite matrices. *arXiv preprint arXiv:1502.03505*.

Zanini, P., Congedo, M., Jutten, C., Said, S., and Berthoumieu, Y. (2016). Parameters estimate of Riemannian Gaussian distribution in the manifold of covariance matrices. In *IEEE Sensor Array and Multichannel Signal Processing Workshop*.

Zhu, D., Bieger, J., Molina, G. G., and Aarts, R. M. (2010). A survey of stimulation methods used in SSVEP-based BCIs. *Computational Intelligence and Neuroscience*, 2010:1–12.