Inductive means and sequences applied to online classification of EEG

Estelle M Massart, Sylvain Chevallier

Inductive means and sequences applied to online classification of EEG

Estelle M. Massart[1], Sylvain Chevallier[2]

[1] ICTEAM Institute, Université catholique de Louvain, Belgium
[2] LISV, Université de Versailles Saint-Quentin, France

Abstract. The translation of brain activity into user command, through Brain-Computer Interfaces (BCI), is a very active topic in machine learning and signal processing. As commercial applications and out-of-the-lab solutions are proposed, there is an increased pressure to provide online algorithms and real-time implementations. Electroencephalography (EEG) systems offer lightweight and wearable solutions, at the expense of signal quality. Approaches based on covariance matrices have demonstrated good robustness to noise and provide a suitable representation for classification tasks, relying on advances in Riemannian geometry. We propose to equip the minimum distance to mean (MDM) classifier with a new family of means, based on the inductive mean, for block-online classification tasks and to embed the inductive mean in an incremental learning algorithm for online classification of EEG.

1 Introduction

Real-time recording and decoding of brain signals allow to control a large variety of systems, such as wheelchairs, exoskeletons, robotic arms or other types of Brain-Computer Interface (BCI) devices [3]. With electroencephalography (EEG), the brain signal is recorded at the surface of the head (on the scalp), offering a simple setup that does not require surgery as it is the case for invasive recording methods. The signal quality of EEG is lower than with invasive methods and the recording is very sensitive to noise, nonetheless possible applications offer promising results [11]. As technologies and signal processing techniques are more and more mature, out-of-the-lab applications and commercial systems are the focus of growing interests [3]. These applications and systems rely on a small number of electrodes for recording and low-cost hardware for signal processing. Thus the denoising and classification algorithms should work online and with a reasonable computational load. One of the most challenging issues with EEG-based BCI is to harness the individual variability of brain signals, which could change from hour-to-hour for a user and are highly variable from one user to the other.

Among all the methods considered in the literature for EEG signal processing, the ones relying on covariance matrices were shown numerically to achieve good performances [12]. In this approach, a portion of the EEG signal is represented by a covariance matrix, whose elements correspond to the covariance of the signals recorded with different electrodes, possibly filtered around different frequencies. The fact that covariance matrices belong to a non-Euclidean space – the manifold of symmetric positive definite (SPD) matrices – calls for efficient classifiers adapted to that geometry.

In this paper, we work with the Minimum Distance to Mean (MDM) classifier, initially proposed in [2]. This classifier assigns covariance matrices to the class with the closest mean. The classification results were shown to depend heavily on the mean and distance definition used, and many possibilities were compared in [5]. In the following we will distinguish the offline setting, where the classifier's parameters are selected and evaluated using all available data, the block-online setting, where the classifier is parametrized on a first batch of data (usually the beginning of a session) and evaluated on another batch of data (the rest of the session), and the online setting, where there is no data available beforehand from the user and the classifier is assessed directly on new data. We equip here this classifier with a new family of means based on the so-called inductive mean, which has the main advantage of being computed incrementally, a key property when working in an online setting. This property was already used in [4] for k-means clustering. We show numerically that the use of these new means achieves a classification accuracy in a block-online framework comparable to the most accurate nonparametric mean: the Riemannian barycenter with respect to the affine-invariant metric (less than 1% of difference on average), while their computation cost is lower. We also propose a variant of the online classification algorithm proposed in [6]. In our algorithm, the means of the classes are adapted online, following an incremental learning scheme. Starting from classes learned with other users, the goal is to enable the algorithm to progressively fit with the observed data of a new user.

The paper is organized as follows. Section 2 is devoted to block-online classification: we define the MDM classifier and the family of means we use, and compare numerically the classification results with other state-of-the-art methods. In Section 3, we present our incremental learning algorithm for online classification.

2 Offline and block-online classification of EEG

The proposed approaches are applied on steady-state visual evoked potentials (SSVEP), that is brain responses to visual stimuli, but are valid on other kinds of BCI stimuli. In a SSVEP experiment, blinking LEDs are placed at different locations in the visual field of a user. The LEDs are blinking at F different frequencies ($\mathrm{freq}_1, \ldots, \mathrm{freq}_F$). The subject is either asked to focus on one specific blinking LED (with a known frequency) or to focus on a location without LED (resting state). The blinking LED elicit induced oscillations in the brain, which are visible in the EEG. The goal is to determine based on the EEG if the user is focusing on a blinking LED and if so, on which one.

We summarize in Algorithm 1 the block-online classification method proposed in [5]. Each time that the user is asked to focus on a stimulus, the portion of the EEG recording following the cue onset (the time at which the user was instructed to focus on the blinking LED) is first transformed into a covariance matrix and then classified using the MDM classifier. The means of the classes are estimated beforehand, based on a collection of labelled data, according to the offline training scheme detailed in Algorithm 2.

Algorithm 1 Block-online classification - MDM algorithm

Inputs: $\bar{\Sigma}^{(k)}$, the mean of the class k, for $k = 1, \ldots, K$ (obtained using Algorithm 2) and an unlabelled EEG trial $X \in \mathbb{R}^{C \times N}$ (with C the number of electrodes and N the number of time samples).
Output: $\hat{k}$, the predicted label of X.
1: Compute $\hat{\Sigma}$, an estimate of the covariance matrix of X (see Section 2.1).
2: Define the class label associated to trial X as $\hat{k} = \mathrm{argmin}_{k=1,\ldots,K}\, \delta(\hat{\Sigma}, \bar{\Sigma}^{(k)})$, where $\delta(\Sigma_1, \Sigma_2) = ||\Sigma_1^{-1/2}\Sigma_2\Sigma_1^{-1/2}||_{\mathrm{F}}$ is the Riemannian distance between Σ_1 and Σ_2.

Algorithm 2 Offline training

Inputs : $X_i \in \mathbb{R}^{C \times N}$, for $i = 1, \ldots, l$, a set of labelled EEG trials, and $\mathcal{I}(k)$, $k = 1, \ldots, K$, the set of indices of trials belonging to class k.
Output: $\bar{\Sigma}^{(k)}$, the mean of the class k, for $k = 1, \ldots, K$.
1: Compute $\hat{\Sigma}_i$, an estimate of the covariance matrix of X_i, for $i = 1, \ldots, l$ (see Section 2.1).
2: **For** $k = 1 : K$ **do**
3: Compute the center of class $\bar{\Sigma}^{(k)} = \mu(\{\hat{\Sigma}_i | i \in \mathcal{I}(k)\})$ (see Section 2.2).

2.1 Estimation of covariance matrices

Algorithms 1 and 2 require to estimate the covariance matrix of an EEG trial $X \in \mathbb{R}^{C \times N}$, where N is the number of time samples and C the number of electrodes. The signal X is first band-pass filtered around the F frequencies used in the experiment, to yield an extended signal as follows:

$$X \in \mathbb{R}^{C \times N} \rightarrow X_{\mathrm{Ext}} = \left[X_{\mathrm{freq}_1}^T, \ldots, X_{\mathrm{freq}_F}^T\right]^T \in \mathbb{R}^{FC \times N}.$$

The covariance matrix $\Sigma \in \mathbb{P}_{CF}$ of the signal X_{Ext}, with $\mathbb{P}_{CF}$ the set of SPD matrices of size $CF \times CF$, is then estimated using the Schäfer estimator [10]. We refer the reader to [6] for more information regarding the choice of the estimator.

2.2 Inductive means and sequences

The training of the classifier also relies on the definition of a mean μ on the set of SPD matrices. Several means were already considered in [5]. Among the non-parametric means, the Riemannian barycenter with respect to the affine-invariant metric was shown numerically to provide the most accurate classification results (we will use the shortcut “Riemannian barycenter” in the rest of the paper, implying that we work here with the affine-invariant metric). However, its computation is rather costly. To remedy this problem, another family of means was proposed in [8]. These means are based on the inductive mean (see [9]).

The inductive mean of a set of SPD matrices $\Sigma_1, \dots, \Sigma_l \in \mathbb{P}_{CF}$ is defined as:

$$M^{\mathrm{Ind}}(\Sigma_1, \dots, \Sigma_l) = \left(\left(\left(\Sigma_1 \#_{\frac{1}{2}} \Sigma_2 \right) \#_{\frac{1}{3}} \Sigma_3 \right) \dots \#_{\frac{1}{l}} \Sigma_l \right) , \tag{1}$$

where $A \#_s B = A^{\frac{1}{2}} (A^{-\frac{1}{2}} B A^{-\frac{1}{2}})^s A^{\frac{1}{2}}$, with $s \in [0, 1]$, is the (unique) point located on the geodesic from A to B, at a distance $s\delta(A, B)$ of A.

If all the matrices pairwise commute, then the Riemannian barycenter and the inductive mean coincide. Otherwise, the inductive mean looses the property of invariance under permutation: in general, $M^{\mathrm{Ind}}(\Sigma_1, \dots, \Sigma_l) \neq M^{\mathrm{Ind}}(\Sigma_{\pi(1)}, \dots, \Sigma_{\pi(l)})$, where π is a permutation of $(1, \dots, l)$. Moreover, in [8], the authors illustrate numerically that the inductive mean $M^{\mathrm{Ind}}(\Sigma_1, \dots, \Sigma_l)$ tends to overemphasize the last data points (i.e., Σ_l, Σ_{l-1}, ...). To remedy this, they developed an inductive sequence $(X_j^{\mathrm{Ind}})_{j=1,2,\dots}$, i.e., an extension of the inductive mean in which each element X_{jl}^{Ind}, with $j = 1, 2, \dots$ and l the total number of matrices, is defined as:

$$X_{jl}^{\mathrm{Ind}} = M^{\mathrm{Ind}} \left(\pi \left(\underbrace{\Sigma_1, \Sigma_2, \dots, \Sigma_l, \dots, \Sigma_1, \Sigma_2, \dots, \Sigma_l}_{j \times l \text{ elements}} \right) \right) \tag{2}$$

where π is a shuffling operator. The sequence $(X_j^{\mathrm{Ind}})_{j=1,2,\dots}$ converges to the Riemannian barycenter, and the shuffling improves the convergence rate by reducing the bias mentioned above.

2.3 Experimental results for block-online classification

Table 1 compares block-online classification accuracy and computation times for several mean definitions. Our validation is performed on the same datasets as in [5]. These datasets were obtained in a SSVEP experiment with three frequencies (13, 17, or 21 Hz). This is thus a classification task with four classes (one for each frequency and one for the resting class). For each subject, the recorded session is made of several batches (from 2 to 5), one batch consisting in 32 trials (i.e., the responses to 32 stimuli, 8 for each class). As in [5], we used, for each subject, the last batch as validation set and all other batches as training set. We refer the reader to [5] for more detail regarding the experimental protocol.

We compare inductive means with the Euclidean mean, the Log-Euclidean mean and the Riemannian barycenter (estimated using a steepest descent algorithm). For comparison, we also provide results obtained with a state-of-the-art method not based on covariance matrices: the SVM algorithm with CCA filtering used in [5]. The last row of Table 1 presents the average performances obtained with the different means. It indicates that the inductive mean is a nice trade-off between the Log-Euclidean mean, which is cheaper to compute but also less accurate, and the Riemannian barycenter, which is more accurate but considerably more costly. Inductive sequences improve further the accuracy, but become also more costly.

Observe finally that Algorithms 1 and 2 are only suitable for block-online classification, and require to know the cue onsets, i.e., the time at which the stimuli are applied to the subject. Those requirements will be relaxed in the online classification approach presented in the next section.

	CCA + SVM [7]	MDM											
		Euclidean		LogEuclid.		Riem. Baryc.		Ind. Mean		Ind. Seq. X_{2l}^{Ind}		Ind. Seq. X_{5l}^{Ind}	
	acc(%)	acc(%)	t(ms)	acc(%)	t(ms)	acc(%)	t(ms)	acc(%)	t(ms)	acc(%)	t(ms)	acc(%)	t(ms)
S1	54.68	53.12	0.6	71.88	15	**73.44**	74	70.31	15	**73.44**	20	**73.44**	54
S2	37.50	43.75	0.5	78.12	16	**79.69**	77	78.12	11	78.12	19	78.12	43
S3	**89.06**	67.19	0.7	85.94	18	85.94	72	85.94	16	85.94	23	85.94	60
S4	79.69	54.69	0.5	84.38	13	**87.50**	68	**87.50**	11	**87.50**	22	**87.50**	42
S5	50.00	37.50	0.6	62.50	16	**68.75**	63	67.19	11	**68.75**	21	**68.75**	44
S6	**87.50**	34.38	0.5	84.38	18	85.94	69	84.38	11	85.94	19	85.94	49
S7	77.08	60.42	0.9	87.50	29	88.54	131	**89.58**	27	**89.58**	39	**89.58**	99
S8	73.44	67.19	0.7	90.62	18	**92.19**	71	**92.19**	16	**92.19**	24	**92.19**	46
S9	60.94	57.81	0.6	**70.31**	13	**70.31**	69	**70.31**	15	**70.31**	23	**70.31**	56
S10	67.97	38.28	1.2	75.00	43	**80.47**	179	78.91	38	78.91	67	**80.47**	137
S11	**71.88**	48.44	0.8	60.94	18	65.62	82	64.06	16	64.06	22	65.62	47
S12	95.63	71.25	1.5	96.25	58	**96.88**	216	**96.88**	49	**96.88**	86	**96.88**	198
Avg	70.45	52.83	0.8	78.98	23	81.27	97	80.45	20	80.97	32	81.23	73

Table 1. Performances obtained for block-online classification. Each row of the table corresponds to one subject. The computation times recorded are the average times needed to compute the mean of the covariance matrices of a given class in the training set of the subject. They are larger for the subjects 7, 10 and 12 since the number of matrices to average were bigger for those subjects (their training sets were made of respectively 2, 3 and 4 batches instead of one for the other subjects).

3 Online classification using inductive means

In most cases, cue onsets are not available. The goal is then to detect parts of the EEG signal corresponding with a high probability to a given stimulus. Based on the incremental definition of the inductive mean, we propose a variant of the online classification algorithm detailed in [6]. Indeed, conversely to most other means, including the Riemannian barycenter, the inductive mean of $N+1$ data points can be easily computed from the inductive mean of N points:

$$M^{\text{Ind}}(\Sigma_1, \ldots, \Sigma_{N+1}) = M^{\text{Ind}}(\Sigma_1, \ldots, \Sigma_N) \#_{\frac{1}{N+1}} \Sigma_{N+1}.$$

It is then possible to update the means of the classes 'on-the-fly' in the classification algorithm. The complete classification scheme is presented in Algorithm 3.

It works as follows. The algorithm scans the EEG signal, considering successive frames of size w, the starting times of two successive frames being separated by Δn samples. The covariance matrix of the current frame is estimated and classified using the MDM classifier. The most recurrent class among the last D ones is considered to be the current class. If the confidence in this decision is high enough (see Section 3.1), the class is returned and the mean of the class is updated. Otherwise, the algorithm moves immediately to the next frame. It is of tremendous importance to avoid that possible misclassifications move the means of the classes in an erroneous direction. To this aim, we added in Algorithm 3 a filtering step, following similar ideas as in the Riemannian potato [1]. The mean of the class is updated at most once per trial (i.e., per different stimulus), in the direction of the 'best' covariance matrix scanned in the trial.

Algorithm 3 Online classification

Inputs : $\bar{\Sigma}^{(k)} \in \mathbb{P}_{FC}$, the mean of class k, for all class $k = 1, \dots, K$ (offline training, or default initialisation based on data available from other subjects), $\bar{d}_k$ the average distance between the training matrices belonging to class k and $\bar{\Sigma}^{(k)}$, a EEG recording $\mathcal{X}(n) \in \mathbb{R}^C$, $n = 0, \dots, N$, hyperparameters w, Δn, D, s.
Output: Classification decisions $\hat{k}(n)$.

Initialisation: $\Sigma^{\text{best}} = \bar{\Sigma}^{(1)}$, $d^{\text{best}} = \infty$, $k_{\text{cur}} = -1$, $\hat{k}(n) = -1\ \forall n$ (default value, meaning no decision).
For $d = 0, \dots, \lfloor \frac{N-w}{\Delta n} \rfloor$ **do**
  $X_d := \mathcal{X}\left(d\Delta n, \dots, d\Delta n + w\right)$
  Compute $\hat{\Sigma}_d$, an estimate of the covariance matrix of X_d, and classify it:

$$k_d^* := \underset{k=1,\dots,K}{\operatorname{argmin}}\ \delta(\hat{\Sigma}_d, \bar{\Sigma}^{(k)}).$$

  If $d \geq D$ **then** find most recurrent class among D last classifications:

$$\bar{k} := \underset{k=1,\dots,K}{\operatorname{argmax}}\ \rho(k) \quad \text{with} \quad \rho(k) := \frac{\#\{k_j^* = k\}_{j=d,d-1,\dots,d-D+1}}{D}.$$

    Evaluate confidence criterion C (see Section 3.1)
    If $C =$ true **then**
      If $k_{\text{cur}} > 0$ and $\bar{k} \neq k_{\text{cur}}$ (we left previous class) and $d^{\text{best}} \leq \bar{d}_{k_{\text{cur}}}$ **then** update previous class:
        $\bar{\Sigma}^{(k_{\text{cur}})} := \bar{\Sigma}^{(k_{\text{cur}})} \#_{\frac{\alpha}{s+\alpha}} \hat{\Sigma}^{\text{best}}$ with $\alpha := 1 - \frac{\delta(\bar{\Sigma}^{(k_{\text{cur}})}, \hat{\Sigma}^{\text{best}})}{\bar{d}_{k_{\text{cur}}}}$
        $s := s + 1$, $\Sigma^{\text{best}} := \hat{\Sigma}_d$, $d^{\text{best}} := \delta(\bar{\Sigma}^{(\bar{k})}, \hat{\Sigma}_d)$
      elseif $\delta(\bar{\Sigma}^{(\bar{k})}, \hat{\Sigma}_d) \leq d^{\text{best}}$, improve current estimates:
        $\Sigma^{\text{best}} := \hat{\Sigma}_d$, $d^{\text{best}} := \delta(\bar{\Sigma}^{(\bar{k})}, \hat{\Sigma}_d)$
      $k_{\text{cur}} := \bar{k}$
      $\hat{k}(n) := \bar{k}$ for $n \in [d\Delta n, d\Delta n + w]$

3.1 Confidence criterion

Similarly as in [6], a confidence criterion is used in Algorithm 3 to discard unreliable classifications. Two conditions have to be encountered for this criterion to be satisfied. The first one verifies that the current classification decision is consistent with previous classifications: the class $\bar{k}$ should have been chosen among the D previous classes with a proportion larger than or equal to given threshold ϑ, i.e.,

$$\rho(\bar{k}) \geq \vartheta \; . \tag{3}$$

The second condition is related to the displacement of the covariance matrices: those should be in the direction of the mean of the class. Otherwise, we might expect that a new stimulus has been applied, and that the covariance matrices is moving away from the mean of the old class, to get closer to the mean of the new class. Hence, the relative distances to means should on average decrease on the last D frames:

$$\sum_{j=d-D+2}^{d} (\delta_{\bar{k}}^{\mathrm{rel}}(j) - \delta_{\bar{k}}^{\mathrm{rel}}(j-1)) \leq 0 \quad \text{with} \quad \delta_{\bar{k}}^{\mathrm{rel}}(d) = \frac{\delta(\hat{\Sigma}_d, \bar{\Sigma}^{(\bar{k})})}{\sum_{k=1}^{K} \delta(\hat{\Sigma}_d, \bar{\Sigma}^{(k)})} . \tag{4}$$

If conditions (3) and (4) are satisfied, the confidence criterion is satisfied, i.e., $C =$ true, otherwise $C =$ false.

3.2 Numerical results for online classification

The main interest of Algorithm 3 is that it allows to progressively update the user's means of the classes. To illustrate this, we used EEG batches from the three first subjects to initialize the centers of the classes and we run Algorithm 3 to perform classification on all the batches of the other users. In Figure 1, we compare the results obtained using Algorithm 3 with those obtained when the means of the classes are not updated, i.e. removing lines 10 to 12 in Algorithm 3, for the two subjects with the highest number of batches available, that is 5 for subject 12 and 4 for subject 10. Hyperparameters were set empirically to $w =$ 2.6s, $\Delta n = 0.2$s, $D = 5$, $s = 8$, $\vartheta = 0.7$. For subject 12, the classification accuracy improves with the batches, compared to the version with frozen means of the classes. However, this is not the case for subject 10: despite the use of the filtering step, some misclassification resulted in the displacement of the mean of one class in an erroneous direction, which alters subsequent classification decisions. Unfortunately, the low number of recordings per subject makes it difficult to obtain a reliable measure of the performance of our online algorithm. Further work should therefore aim at assessing the performance of the algorithm on larger datasets. Other filtering strategies can also be investigated for Algorithm 3, as well as the influence of the hyperparameters on the classification results.

References

1. A. Barachant, A. Andreev, and M. Congedo. The Riemannian Potato: an automatic and adaptive artifact detection method for online experiments using Riemannian geometry. In *TOBI Workshop lV*, pages 19–20, 2013.

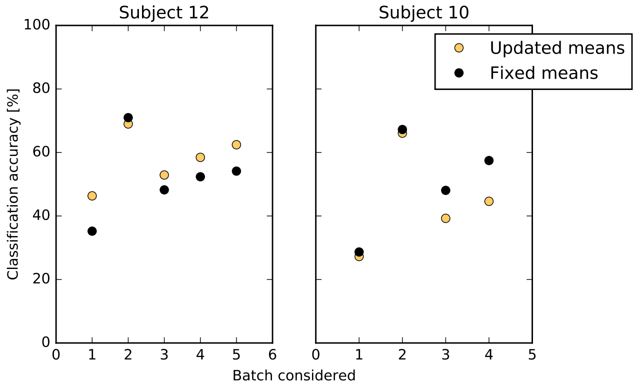

Fig. 1. Classification results of Algorithm 3 on the two subjects for which the largest number of recordings are available.

2. A. Barachant, S. Bonnet, M. Congedo, and C. Jutten. Riemannian Geometry Applied to BCI Classification. In *Latent Variable Analysis and Signal Separation*, pages 629–636. Springer, 2010.
3. M. Clerc, L. Bougrain, and F. Lotte. *Brain-Computer Interfaces 2: Technology and Applications*. Wiley, 2016.
4. J. Ho, G. Cheng, H. Salehian, and B. Vemuri. Recursive Karcher expectation estimators and geometric law of large numbers. In *Artificial Intelligence and Statistics*, pages 325–332, 2013.
5. E. K. Kalunga, S. Chevallier, Q. Barthélemy, K. Djouani, Y. Hamam, and E. Monacelli. From Euclidean to Riemannian Means: Information Geometry for SSVEP Classification. In *LNCS*, pages 595–604. Springer, 2015.
6. E. K. Kalunga, S. Chevallier, Q. Barthélemy, K. Djouani, E. Monacelli, and Y. Hamam. Online SSVEP-based BCI using Riemannian geometry. *Neurocomputing*, 191:55–68, 2016.
7. E. K. Kalunga, K. Djouani, Y. Hamam, S. Chevallier, and E. Monacelli. SSVEP enhancement based on Canonical Correlation Analysis to improve BCI performances. In *Africon*. IEEE, 2013.
8. E. M. Massart, J. M. Hendrickx, and P.-A. Absil. Matrix geometric means based on shuffled inductive sequences. *Linear Algebra and its Applications*, 2017, doi:10.1016/j.laa.2017.05.036.
9. M. Sagae and K. Tanabe. Upper and lower bounds for the arithmetic-geometric-harmonic means of positive definite matrices. *Linear and Multilinear Algebra*, 37(4):279–282, 1994.
10. J. Schäfer and K. Strimmer. A Shrinkage Approach to Large-Scale Covariance Matrix Estimation and Implications for Functional Genomics. *Statistical Applications in Genetics and Molecular Biology*, 4(1), jan 2005.
11. J. van Erp, F. Lotte, and M. Tangermann. Brain-computer interfaces: beyond medical applications. *Computer*, 45(4):26–34, 2012.
12. F. Yger, M. Berar, and F. Lotte. Riemannian approaches in Brain-Computer Interfaces: a review. *IEEE TNSRE*, 2016.