Data augmentation in Riemannian space for Brain-Computer Interfaces

Emmanuel Kalunga, Sylvain Chevallier, Quentin Barthélemy

Data augmentation in Riemannian space for Brain-Computer Interfaces

Emmanuel K. Kalunga EMMANUELKALUNGA.K@GMAIL.COM
Department of Electrical Engineering/FSATI, Tshwane University of Technology, Pretoria 0001, South Africa

Sylvain Chevallier SYLVAIN.CHEVALLIER@UVSQ.FR
Laboratoire d'Ingénierie des Systèmes de Versailles, Université de Versailles Saint-Quentin, 78140 Velizy, France

Quentin Bathélemy QUENTIN.BARTHELEMY@MENSIATECH.COM
Mensia Technologies, ICM, Hôpital de la Pitié-Salpêtrière 75013 Paris, France

Abstract

Brain-Computer Interfaces (BCI) try to interpret brain signals, such as EEG, to issue some command or to characterize the cognitive states of the subjects. A strong limitation is that BCI tasks require a high concentration of the user, de facto limiting the length of experiment and the size of the dataset. Furthermore, several BCI paradigms depend on rare events, as for event-related potentials, also reducing the number of training examples available. A common strategy in machine learning when dealing with scarce data is called data augmentation; new samples are generated by applying chosen transformations on the original dataset. In this contribution, we propose a scheme to adapt data augmentation in EEG-based BCI with a Riemannian standpoint: geometrical properties of EEG covariance matrix are taken into account to generate new training samples. Neural network are good candidates to benefit from such training scheme and a simple multi-layer perceptron offers good results. Experimental validation is conducted on two datasets: an SSVEP experiment with few training samples in each class and an error potential experiment with unbalanced classes (NER Kaggle competition).

1. Introduction

Brain Computer Interfaces (BCI) allow human-machine communication using brain signals, i.e. without involving the neuromuscular pathways. In rehabilitation and assistive technology, it constitutes a promising solution to compensate for motor disabilities. It can also be used to reinforce conventional human-machine interactions by providing information not explicitly expressed by the subjects (Mühl et al., 2014). This contribution focuses on BCI based on electroencephalography (EEG) to record brain signals. Some of the neurological phenomenon that are decoded are Steady State Visually Evoked Potentials (SSVEP) (Capilla et al., 2011), and event related potential (ERP) (Wolpaw et al., 2002).

The current state of the art in BCI performances is still facing challenges that limit their usage. To name but a few, these challenges are due to (1) the curse of dimensionality resulting from the high dimensional feature space (*i.e.* feature space too large compared to the number of trials available), (2) low signal-to-noise ratio and poor spatial resolution of EEG, (3) the inter-session and the inter-subject variabilities (non-stationarity in the data) , (4) the difficulty to acquire long and reliable recording from subjects. The latter is imputable to the high cognitive engagement required from the subjects to perform correctly the task. In the case of ERP, the signal of interest (i.e. neurological phenomenon) is a rare event. The curse of dimensionality is usually handled by mapping the feature space to a lower dimensional space. Several dimensionality reduction techniques, such as Principal Component Analysis (PCA), have been applied to this end (Blankertz et al., 2006). State-of-the-art methods address the noise corruption and the poor spatial resolution problems in EEG signals with spatial filters such as Common Spatial Pattern (CSP) (Muller-Gerking et al., 1999), xDAWN (Rivet et al., 2009), Canonical Correlation Analysis (CCA) (Kalunga et al., 2013). Most of these filtering methods depend on the estimation of the signal covariance matrices. Approaches drawing from non-Euclidean geometry have demonstrated successful re-

ICML Workshop on Statistics, Machine Learning and Neuroscience

sults, substituting the need of spatial filters and providing a simple yet effective framework (Congedo et al., 2013). The main idea is to consider covariance matrices, which are symmetric positive definite (SPD), in their original space.

For efficient learning in EEG based BCI, as in most machine learning applications, an important amount of training data is needed. However the amount of data available within the BCI community is little (Delorme A., 2015). Another particularity with BCI is that the inter-subject variability requires that data used for training come from the same subject that the testing ones. Because of the above mentioned difficulties in acquiring long signals from users and the need to keep the calibration time short, such training data are usually not available. Moreover, in some BCI applications the number of trials per class cannot be determined by the experimental paradigm, resulting in a class imbalance that disturbs the learning process.

A possible way of solving these problems related to data scarsity is *data augmentation*. In this approach, artificial data are generated by applying a transformation to the recorded data (Van Dyk & Meng, 2001; Grandvalet, 2000). This technique has been successfully applied on image classification, when the number of samples in each class is small. The common practice is to identify a set of possible transformations that could affect input images, e.g. rotation, translation, scaling, flipping, brightness adjustment, and to randomly applied those transformation to each training example (Dieleman et al., 2015). In the context of handwritten character recognition, an elastic distortion emulating uncontrolled oscillation of hand muscles is applied (Simard et al., 2003). Data augmentation works well when combined with artificial neural network (Duda et al., 2001; Ciresan et al., 2012; Krizhevsky et al., 2012). In BCI applications, a similar approach has been used to reduce calibration time in a motor imagery based BCI system (Lotte, 2015). Each recorded trial is segmented and segments from the original set are randomly selected and concatenated to form new artificial trials.

In this work a novel data augmentation method based on non-Euclidean geometry is proposed. Unlike those mentioned above, data are not generated in the input space. Each training trial is represented in the space of SPD matrices by its covariance matrix. The space of SPD matrices, with the proper structure and inner product, defines a Riemannian manifold. The augmented data lives on the manifold and within the convex hull defined by their class set. As a result, the convex hull of the class is densified with transformed versions of the original data. The augmented data are fed to a classifier, here we consider a multi-layer perceptron. This method is evaluated on two experimental datasets. The first one is a SSVEP-based BCI where only a limited number of training example are available. The second one is an error detection application of ERP-based BCI to generate artificial trials to balance the number of positive and negative trials. In the error related potential (ErrP) application paradigm, the number of trials with and without ErrP variable and not controlled. In the following, the proposed approach for training set augmentation is presented in Section 2. Data and results are presented in Sections 3 and 4, and a conclusion is drawn in Section 5.

2. Data augmentation on covariance matrices

This section presents the proposed approach of augmenting training data examples from their covariance matrices using Riemannian geometry. It is divided into three parts. First, it introduces the basics of Riemannian geometry as applied in the field of brain computer interface and considering only notions relevant to the current work. Then, a second part details the specific construction of covariance matrices used in this work. Following those definitions, a third part presents the proposed method to generate artificial data.

2.1. Riemannian geometry tools

Let $x_n \in \mathbb{R}^C$, $n = 1, \ldots, N$, denotes a sample of a multichannel EEG trial recorded on C electrodes. N is the trial length. Let $X \in \mathbb{R}^{C \times N}$ be the EEG trial such as $X = [x_1, \ldots, x_N]$. Under the hypothesis that all N samples x_n are randomly drawn from a distribution, it follows that X is a variable of random vectors and its expected vector is $\omega = \mathbb{E}\{X\}$ (Fukunaga, 1990). The covariance matrix of the random vector X is defined by $S = \mathbb{E}\{(X-\omega)(X-\omega)^\intercal\}$. Let Σ be an estimate of covariance matrix S. Σ belongs to the set $\mathcal{M}_C$ of the $C \times C$ symmetric positive definite matrices, which is defined as:

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

A *geodesic* γ is a smooth curve between two points, Σ_1 and Σ_2 on the manifold. The tangent space $T_\Sigma\mathcal{M}$ at point Σ is the vector space spanned by the tangent vectors of all geodesics on $\mathcal{M}$ passing through Σ. This additional structure defines a manifold over the set of SPD matrices. A *Riemannian* manifold is a manifold endowed with an inner product defined on every tangent space, which varies smoothly from point to point. The tangent space $T_\Sigma\mathcal{M}_C$ at point Σ is identified to be the set of symmetric matrices:

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

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

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

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

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

Details on the computation of the mappings can be found in (Bhatia, 2009).

The tangent vector of the geodesic γ between Σ_1 and Σ_2 is defined as:

$$v = \overrightarrow{\Sigma_1\Sigma_2} = \mathrm{Log}_{\Sigma_1}(\Sigma_2) \ . \tag{3}$$

A Riemannian distance between Σ_1 and Σ_2 can thus be defined as (Bhatia, 2009):

$$\delta(\Sigma_1, \Sigma_2) = \| \mathrm{Log}(\Sigma_1^{-1}\Sigma_2)\|_F = \left[\sum_{c=1}^{C} \log^2 \lambda_c\right]^{1/2} , \tag{4}$$

where λ_c, $c = 1, \ldots, C$, are the eigenvalues of $\Sigma_1^{-1}\Sigma_2$. From Eq. (4), the mean of I points Σ_i on the manifold, $i = 1, \ldots, I$, can be defined as the point that minimizes the sum of squared distances to all Σ_i:

$$\bar{\Sigma} = \arg\min_{\Sigma \in \mathcal{M}_C} \sum_{i=1}^{I} \delta^2(\Sigma_i, \Sigma) \ . \tag{5}$$

Contrary to the arithmetic mean, this geometric mean has no closed form for $I > 2$ and can be computed iteratively with a gradient descent (Fletcher et al., 2004).

In the following, features $w \in \mathbb{R}^{C(C+1)/2}$ are obtained projecting matrices on the tangent space at their mean $\bar{\Sigma}$ (Barachant et al., 2013b):

$$\Theta_i = \bar{\Sigma}^{-\frac{1}{2}} \mathrm{Log}_{\bar{\Sigma}}(\Sigma_i)\bar{\Sigma}^{-\frac{1}{2}} = \mathrm{Log}(\bar{\Sigma}^{-\frac{1}{2}}\Sigma_i\bar{\Sigma}^{-\frac{1}{2}}) \ , \tag{6}$$

and then extracting the upper triangular part of a symmetric matrix Θ_i and vectorizing it (applying $\sqrt{2}$ weight for out-of-diagonal elements).

2.2. Covariance matrices for SSVEP and ERP signals

Covariance matrices are constructed such that they contain discriminative information for either SSVEP or ERP. The approach used was introduced in (Congedo et al., 2013). The covariance matrices are estimated from a modified version of the input signal X.

For SSVEP, with F stimuli frequencies, the input signal is modified as:

$$X \in \mathbb{R}^{C\times N} \rightarrow \begin{bmatrix} X_{\mathrm{freq}_1} \\ \vdots \\ X_{\mathrm{freq}_F} \end{bmatrix} \in \mathbb{R}^{FC\times N} \ , \tag{7}$$

The resulting signal is a concatenation of signals X_{freq_f} obtained by filtering X around various stimuli frequencies freq_f, $f = 1, \ldots, F$. The covariance matrix $\hat{\Sigma}$ estimated from such modified signal is of size $(F \times C)^2$. It is a multiclass classification with $K = F + 1$ classes: one class per stimulus and one resting state class.

For ERP paradigm with a number E of different ERPs, the modified signal is the concatenation of the original signal and the grand averages of trials containing the target ERPs $\bar{X}_e$, $e = 1, \ldots, E$:

$$X \in \mathbb{R}^{C\times N} \rightarrow \begin{bmatrix} \bar{X}_1 \\ \vdots \\ \bar{X}_E \\ X \end{bmatrix} \in \mathbb{R}^{(E+1)C\times N} \ , \tag{8}$$

The resulting covariance matrix will be of size $((E + 1) \times C)^2$. Adding a non-target class, it is a multiclass classification with $K = E + 1$ classes.

The covariance matrices are estimated from the modified input signals using Schäfer shrinkage covariance matrix estimator (Schäfer & Strimmer, 2005).

From J labelled training trials $\{X_i\}_{j=1}^{J}$ recorded per subject, $k = 1, \ldots, dK$ centers of classes $\bar{\Sigma}^{(k)}$ are estimated using Algorithm 1 of (Kalunga et al., 2015). Finally, feature w_i is extracted using (6) with the mean $\bar{\Sigma}$ computed on all available data. Using a Riemannian kernel, projecting matrices on the global mean of all data is equivalent to first whitening the covariance matrices of the dataset and then using the identity matrix as reference for projecting on the tangent space (Barachant et al., 2013b; Yger & Sugiyama, 2015).

2.3. Generating artificial points on Riemannian manifold

Each trial's covariance matrix being represented as a point on the manifold, artificial trials are generated by interpolating new points between original trials' covariance matrices belonging to one class. This interpolation is done on the geodesic connecting each pair of original trials such that the generated point remains on the manifold and within the convex hull of the set of the class original data. This approach is similar to tensor linear interpolation introduced in (Pennec et al., 2006). Given the definition of the tangent vector $\overrightarrow{\Sigma_1\Sigma_2}$ between Σ_1 and Σ_2 in (3), the geodesic γ on the manifold can be obtained by the exponential mapping of $\overrightarrow{\Sigma_1\Sigma_2}$ defined in (1) as: $\gamma = \mathrm{Exp}_{\Sigma_1}(\mathrm{Log}_{\Sigma_1}(\Sigma_2))$. Defining $t \in [0; 1]$, points lying on the geodesic are defined by:

$$\begin{aligned} \Sigma(t) &= \mathrm{Exp}_{\Sigma_1}(t \ \mathrm{Log}_{\Sigma_1}(\Sigma_2)) \\ &= \Sigma_1^{\frac{1}{2}}(\Sigma_1^{-\frac{1}{2}}\Sigma_2\Sigma_1^{-\frac{1}{2}})^t\Sigma_1^{\frac{1}{2}} \end{aligned} \tag{9}$$

with $\Sigma_1 = \Sigma(0)$ and $\Sigma_2 = \Sigma(1)$. Remark that the interpolation (9) is equivalent to $(1-t)\Sigma_1 + t\Sigma_2$ in Euclidean space. One can note that this interpolation avoid the "swelling effect" afflicting Euclidean interpolation, *i.e.* the determinant of $\Sigma(t)$ can be significantly larger than determinants of Σ_1 or Σ_2. Artificial points for data augmentation are obtained between original points by setting t in (9) to any value other than 0 and 1. In our experiments, interpolated matrices between each pair Σ_1, Σ_2 are linearly spaced on the geodesic between 0 and 1, and all possible pairs are considered.

Outliers in the pool of original data covariance matrices can distort the convex hull of classes, resulting in misclassification of new trials. To alleviate these effects, outliers are rejected from the original data before the generation of artificial data using an offline Riemannian potato (Barachant et al., 2013a). The Riemannian mean of matrices belonging to one class is used as the center of the Riemannian potato for that class. For each class, all matrices beyond the z-score of 1 from the class center are rejected. This value has been chosen after careful cross-validation.

2.4. Classification

To evaluate the benefit of applying the proposed data augmentation method, three classifiers are considered: a multilayer perceptron (MLP) neural network (Duda et al., 2001) which is used on original data and then on augmented data, a tangent space linear discriminant analysis (TSLDA) (Barachant et al., 2012) and a Riemannian-kernel support vector machine (RK-SVM) (Yger, 2013). The choice for a MLP is motivated by the fact that neural networks are known to be sensitive to the amount and diversity of examples of data they are presented with (Ciresan et al., 2012; Krizhevsky et al., 2012). On the other hand, RK-SVM and TSLDA are versions of SVM and LDA adapted to data lying on a Riemannian space. They are arguably the state-of-the-art concerning EEG covariance classification in tangent space (Barachant et al., 2012; 2013b). These 3 classification methods are offline since the feature extraction (6) requires the projection on the global mean. However, online extensions are possible (Barachant et al., 2013b; Kalunga et al., 2015).

3. Experimental data description

The assessment of the proposed data augmentation method is conducted on two datasets. The first one is a SSVEP-based experiment, with a few number of sample for each class. The second dataset is an error-related potential detection, where the number of positive examples (the error potential) is smaller than the number of negative examples, that is a problem with unbalanced classes.

3.1. SSVEP dataset

This dataset was recorded during an SSVEP-based BCI experiment (Kalunga et al., 2014). EEG are recorded on $C = 8$ channels from 12 subjects. The subjects are presented with $F = 3$ visual target stimuli blinking respectively at 13, 17 and 21Hz. When a subject set his gaze at specific stimulus, the neuronal activity in the primary visual cortex are expected to be synchronize at the stimulus frequency. It is this synchronization that should be identified by classification algorithms. It is a $K = 4$ classes setup combining $F = 3$ stimulus classes and one resting class (no-SSVEP). In a session, 32 trials are recorded: 8 for each visual stimulus and 8 for the resting class. The number of sessions recorded for training varies from 1 to 4 i.e. the size of training set for a subject varies between 32 and 128 trials equally distributed across 4 classes. A session is recorded within 5 min. Calibration time is short, and the number of training trials is limited, making this dataset a good candidate for training set augmentation.

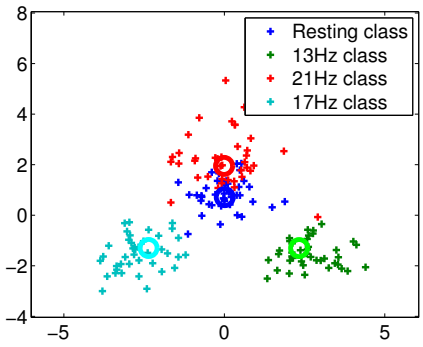
3.2. ERP dataset

The dataset, available for the NER Kaggle competition, was recorded during an online P300 speller experiment for error detection in the speller (Perrin et al., 2012). 16 healthy subjects participated in the experiment, the brain activity was recorded on $C = 56$ channels. Subjects have to spell a series a letter in under two spelling conditions: a fast, more error-prone condition (each item is flashed 4 times), and a slower, less error-prone (each item is flashed 8 times). The subjects had to go through five spelling sessions. Each session consisted of twelve 5-letter words, except the fifth which consisted of twenty 5-letter words making up for a total of 340 letters. For each spelled letter, the feedback of the result of the speller is displayed on a screen. The time of feedback is recorded and the labelled of feedback (correct or incorrect) is also recorded. In case of error in the spelling, an error evoked potential occurs in the EEG. In the current work we focus on the detection of the error in spelling based on this *a priori*. The task of learning algorithms is to detect errors, *i.e.* to classify trials as incorrect or correct ($K = 2$, positive or negative). In such experiments, the number of positive and negative trials is not balanced. In case of a good speller, the number of positive trials are very limited. In this dataset the number of positive trials is largely inferior to the number of negative trials creating a problem of class unbalance in training set. To balance training set from this experiment, artificial data can be generated in the class with less number of trials.

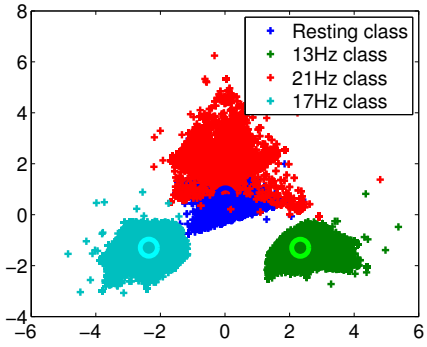
4. Results and discussion

4.1. SSVEP dataset

SSVEP training set is augmented with different number of artificial samples for each class. One to five samples are interpolated between each pair of original samples belonging to a single class. Figure 1 shows the densification effect resulting from the augmentation process. Original covariance matrices of each class are projected on the tangent space computed at the mean of all the matrices, and the two principal components (obtained by applying PCA) are shown on Fig. 1(a). Similarly, Fig. 1(b) shows the augmented covariance matrices after interpolation of 5 points between each pair of covariance matrices within each class. The augmented data are within the convex hull of the original data.



(a)



(b)

Figure 1. Mapping of covariance matrices of trials from each class on the tangent space (2). Matrices on the tangent space are vectorized and the 2 most significant components from PCA are used to obtain the 2-D representation. Covariance matrices of original data (a) and augmented data (b)..

The performance of the augmentation approach is evaluated in terms of classification accuracy obtained with an MLP classifier and the the results are compared with those obtained with TSLDA and RK-SVM classifiers. The inputs to the MLP are trials covariance matrices mapped on the tangent space. The MLP has 108 input units, one hidden layer with 50 neurons, and 4 output units. The classification obtained with each number of interpolated points are compared to the performance without training set augmentation. Figure 2 shows the classification performances from zero interpolated point (no training set augmentation) to 5 points interpolated. Due to the non-convexity of MLP optimization, results averaged over subjects, are then averaged over 10 repetitions. Significant p-values show that average classification across all subjects is improved by the data augmentation. The effect of data augmentation varies depending on the quality of training examples from individual subjects. In Figure 3, the effect of augmenting training data in the subject with the lowest BCI performance and the subject with highest BCI performances are put side by side. In Table 1, the classification accuracies (in %) of the MLP preceded with data augmentation are compared with RK-SVM and TSLDA.

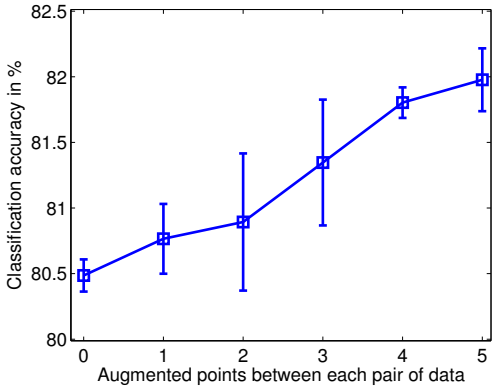


Figure 2. Mean classification accuracy in % across all 12 subjects for different level of data augmentation. At 0, there is no augmented data. At 1, one artificial data is interpolated between each pair of original data within each class, and so forth

4.2. ERP dataset

On the ERP dataset the data augmentation is done to balance the number of positive trials (incorrect P300 feedback where ErrP is present) and negative trials (feedback with no error) in the training set. Each subject has 240 or 280 trials in the training set. The number of positive trials can be as low as 2% of the training set. The number of generated artificial trials g is determined by the gap between the number of positive trials and negative trials in the training set. To generate g trials, a covariance matrix is interpolated between g pairs of randomly selected original matrices. The effect of balancing classes with artificial trials is evaluated

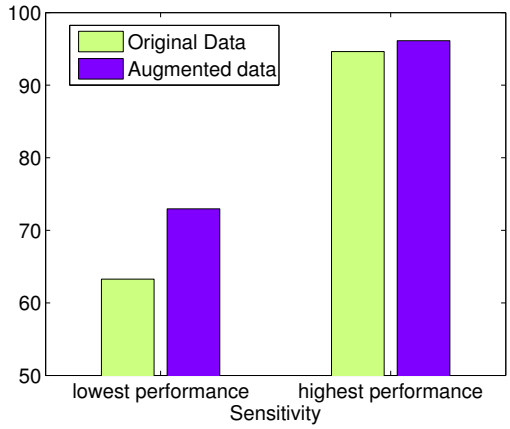


Figure 3. Classification accuracy of subject with lowest BCI performance versus subject with highest BCI performance, using original training set and using augmented training set with 5 interpolated points between each pair of original data within each class.

	MLP	aug+MLP	RK-SVM	TSLDA
Sub 1	70.63	70.63	68.75	**73.44**
Sub 2	71.25	78.28	**82.81**	76.56
Sub 3	94.22	**95.00**	93.75	93.75
Sub 4	84.06	86.72	92.19	**93.75**
Sub 5	**73.75**	67.50	**73.44**	71.88
Sub 6	84.84	**87.66**	82.81	84.38
Sub 7	90.73	**91.67**	89.58	90.63
Sub 8	89.22	**92.19**	89.06	90.63
Sub 9	**70.78**	68.28	62.50	67.19
Sub 10	78.44	76.72	**78.91**	78.13
Sub 11	63.28	**72.97**	71.88	70.31
Sub 12	94.62	**96.13**	**95.63**	93.13
Average	80.49	**81.98**	81.78	**81.98**

Table 1. Comparison of classification accuracies (in %) using the MLP on original dataset, MLP with data augmentation (aug+MLP), RK-SVM and TSLDA.

with the three classifiers (*i.e.* MLP, TSLDA and RK-SVM). The MLP has 10 input units, one hidden layer with 50 neurons and two output units. The number of MLP units is chosen after a cross-validation phase.

Since the class unbalance is still present in the evaluation set, the classification performances are evaluated in terms of sensitivity. Figure 4 shows the performance achieved when classes are balanced by augmenting data in the positive class. They are compared to the results achieved when using unbalanced training set. A t-test was performed and the p-values reveal significant improvement after data augmentation. Table 2 show details of classifiers performance per subject in terms of sensitivity with and without data augmentation.

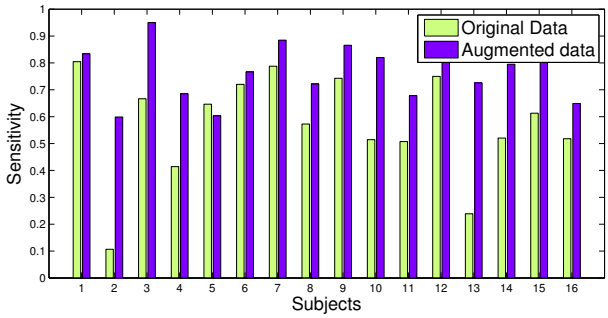


Figure 4. Classification performance in terms of sensitivity. For each of the 16 subjects these measures are given for classification based on training on original unbalance training set and training on augmented and balanced training set.

5. Conclusion

In BCI, datasets with reduced numbers of samples and unbalanced classes are frequent. This contribution introduces a data augmentation scheme based on the geometry of covariance matrices. From the geodesics passing through pairs of samples, new samples are drawn and fed to a neural classifier. The data augmentation allows to boost the classification accuracy when there is only a few number of samples per class. It is also possible to compensate for dataset with unbalanced classes as it is often the case in event-related potential paradigm. The choice of the classifier is important when dealing with this augmented data; neural networks yields the best results. Future works will focus on the optimization of the neural networks: determining the best architecture (in term of layers and neurons) for processing covariance matrices and the investigation of common deep learning methods to improve results (dropouts, ReLU units, etc).

Acknowledgments

The authors would like to thank Florian Yger for his valuable comments and suggestions.

References

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

Barachant, A., Andreev, A., and Congedo, M. The Rieman-

	Imbalanced classes			Balanced classes		
Sub.	MLP	RK-SVM	TSLDA	MLP	RK-SVM	TSLDA
1	0.85	0.76	0.79	0.83	0.77	**0.85**
2	0.11	0	0.32	**0.60**	0.07	0.57
3	0.67	0.60	0.72	**0.95**	0.63	**0.95**
4	0.41	0.42	0.63	0.69	0.32	**0.70**
5	0.65	0.51	0.61	0.60	0.49	**0.68**
6	0.72	0.71	0.74	**0.77**	0.70	0.76
7	0.79	0.70	0.78	0.88	0.70	**0.89**
8	0.57	0.33	0.63	**0.72**	0.25	0.70
9	0.74	0.59	0.77	0.87	0.58	**0.89**
10	0.51	0.34	0.59	0.82	0.34	**0.90**
11	0.51	0.27	0.57	**0.68**	0.27	0.61
12	0.75	0.65	0.82	0.97	0.65	**0.99**
13	0.24	0	0.57	0.73	0.08	**0.75**
14	0.52	0.47	0.62	**0.80**	0.43	0.75
15	0.61	0.51	0.65	0.81	0.60	**0.83**
16	0.52	0.46	0.54	**0.65**	0.42	0.53
Average	0.570	0.459	0.648	**0.773**	0.46	0.772

Table 2. Sensitivity analysis of performances obtained with 3 classifiers trained with imbalanced training set versus trained with balanced training set. The class imbalance of the ERP dataset is solved with data augmentation.

nian potato: an automatic and adaptive artifact detection method for online experiments using Riemannian geometry. In *Proc. of TOBI Workshop IV*, pp. 19–20, 2013a.

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

Bhatia, R. *Positive definite matrices*. Princeton University Press, 2009.

Blankertz, B., Muller, K. R., Krusienski, D. J., Schalk, G., Wolpaw, J. R., Schlogl, A., Pfurtscheller, G., Millan, Jd, Schroder, M., and Birbaumer, N. The BCI competition III: validating alternative approaches to actual BCI problems. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 14(2):153–159, 2006.

Capilla, A., Pazo-Alvarez, P., Darriba, A., Campo, P., and Gross, J. Steady-state visual evoked potentials can be explained by temporal superposition of transient event-related responses. *PLoS ONE*, 6(1):e14543, 2011.

Ciresan, D., Meier, U., and Schmidhuber, J. Multi-column deep neural networks for image classification. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 3642–3649, 2012.

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

Delorme A. EEG data available for public download, 2015. URL http://sccn.ucsd.edu/~arno/fam2data/publicly_available_EEG_data.html.

Dieleman, S., Willett, K. W, and Dambre, J. Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly Notices of the Royal Astronomical Society*, 450(2):1441–1459, 2015.

Duda, R., Hart, P., and Stork, D. *Pattern classification*. Wiley, 2 edition, 2001.

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

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

Grandvalet, Y. Anisotropic noise injection for input variables relevance determination. *Neural Networks, IEEE Transactions on*, pp. 463–468, 2000.

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

Kalunga, E., Chevallier, S., Rabreau, O., and Monacelli, E. Hybrid interface: Integrating BCI in multimodal human-machine interfaces. In *IEEE/ASME Int. Conf. on*

Advanced Intelligent Mechatronics (AIM), pp. 530–535, 2014.

Kalunga, E., Chevallier, S., and Barthélemy, Q. Using Riemannian geometry for SSVEP-based brain computer interface. *arXiv preprint arXiv:1501.03227*, 2015.

Krizhevsky, A., Sutskever, I., and Hinton, G. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. NIPS, 2012.

Lotte, F. Signal processing approaches to minimize or suppress calibration time in oscillatory Activity-Based BrainComputer interfaces. *Proceedings of the IEEE*, 103 (6):871–890, 2015.

Mühl, C., Jeunet, C., and Lotte, F. EEG-based workload estimation across affective contexts. *Frontiers in neuroscience*, 8, 2014.

Muller-Gerking, J., Pfurtscheller, G., and Flyvbjerg, H. Designing optimal spatial filters for single-trial EEG classification in a movement task. *Clinical Neurophysiology*, 110(5):787–798, 1999.

Pennec, X., Fillard, P., and Ayache, N. A Riemannian framework for tensor computing. *International Journal of Computer Vision*, 66(1):41–66, 2006.

Perrin, M., Maby, E., Daligault, S., Bertrand, O., and Mattout, J. Objective and subjective evaluation of online error correction during P300-based spelling. *Advances in Human-Computer Interaction*, 2012:4, 2012.

Rivet, B., Souloumiac, A., Attina, V., and Gibert, G. xDAWN algorithm to enhance evoked potentials: Application to brain-computer interface. *Biomedical Engineering, IEEE Transactions on*, 56(8):2035–2043, 2009.

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

Simard, P., Steinkraus, D., and Platt, J. Best practices for convolutional neural networks applied to visual document analysis. In *Int. Conf. on Document Analysis and Recognition*, volume 2, pp. 958–958. IEEE Computer Society, 2003.

Van Dyk, D. and Meng, X-L. The art of data augmentation. *Journal of Computational and Graphical Statistics*, 10 (1), 2001.

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

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

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