Proceedings of the
7th Graz Brain-Computer Interface Conference 2017 DOI: 10.3217/978-3-85125-533-1-16

DIMENSIONALITY REDUCTION FOR BCI CLASSIFICATION USING RIEMANNIAN GEOMETRY

P. L. C. Rodrigues[1], F. Bouchard[1], M. Congedo[1], C. Jutten[1]

[1]GIPSA-lab, CNRS, University Grenoble Alpes, Grenoble Institute of Technology, Grenoble, France

E-mail: pedro-luiz.coelho-rodrigues@gipsa-lab.fr

ABSTRACT: In the past few years, there has been an increasing interest among the Brain-Computer Interface research community in classification algorithms that respect the intrinsic geometry of covariance matrices. These methods are based on concepts of Riemannian geometry and, despite demonstrating good performances on several occasions, do not scale well when the number of electrodes increases. In this paper, we evaluate two methods for reducing the dimension of the covariance matrices in a geometry-aware fashion. Our results on three different datasets show that it is possible to considerably reduce the dimension of covariance matrices without losing classification power.

INTRODUCTION

In recent years a new trend of algorithms using concepts from Riemannian geometry have demonstrated remarkable performance on classification of BCI signals, often superior to the current state of the art. As shown in a recent literature survey [1], such results gave rise to a new generation of Brain-Computer Interface (BCI) systems that is becoming each year more popular among the research community.

In BCI classification we are given a dataset containing short-time recordings of EEG, each associated to a condition (or class). The goal is to train an algorithm on an ensemble of trials with known labels and use it to correctly classify a set with unknown labels. The usual approach is to select certain features describing the trials and use statistical models to classify them [3]. A useful feature one may consider when working with EEG signals is their spatial covariance matrix, since different classes are expected to have different patterns of correlation between electrodes. The core idea behind algorithms using Riemannian geometry is to manipulate covariance matrices in the manifold of symmetric positive-definite (SPD) matrices and use them directly as features in a classifier that respects their intrinsic geometry.

The computational complexity of algorithms based on this premise is of concern for high-density EEG data. This happens because Riemannian algorithms rely on eigendecompositions, whose number of operations is on the order of n^3, where n is the number of electrodes. Also, due to very low eigenvalues in the spectrum of high-dimensional covariance matrices (mainly associated to noise), logarithmic maps used by Riemannian algorithms may encounter numerical difficulties. Furthermore, classifiers using high-dimensional covariance matrices as features are prone to overfitting because of the curse of dimensionality and the limited number of trials usually available in BCI datasets [3].

Fortunately, the very nature of EEG recordings allows us to consider only a subspace of the data without losing much information. This is possible because of the strong statistical correlation between signals recorded from close positions and the small number of independent sources that are active during brain activity. By exploring this redundancy, we can reduce the dimensions of spatial covariance matrices and use Riemannian geometric algorithms more efficiently.

The literature of dimensionality reduction (DR) is very rich and many methods already exist. Some are general-purpose algorithms, like principal component analysis (PCA) and multi-dimensional scaling (MDS), others are specific to the analysis of EEG signals, such as common spatial patterns (CSP). However, none of these alternatives take into account the intrinsic geometry of the covariance matrices to reduce their dimensions in a principled manner.

Recently, in the computer vision literature, Ref. [4] presented two geometry-aware methods for reducing the dimensions of SPD matrices, a supervised and an unsupervised approach. Both algorithms are based on the theory of optimization on manifolds [7] and demonstrated good results on image and video databases. Shortly after, Ref. [5] applied the unsupervised dimensionality reduction described in [4] to datasets of Motor Imagery (MI) BCI and obtained encouraging results.

In this work, we apply both algorithms given in [4] to the context of BCI signals. We extend the results from [5] by considering datasets with several subjects and test the algorithms not only on MI but also on the P300 paradigm. We examine the sensitivity of the classification algorithms to the choice of the reduced dimension and investigate the conditions in which a DR would be advisable or not. This paper continues with a section on Materials and Methods, where we give a brief presentation of concepts of Riemannian geometry and an overview of methods for geometry-aware dimensionality reduction. We also present the datasets and the classification pipelines used for assessing the quality of each dimensionality reduction proposal. We continue with a section of Results and Discussion and leave final comments to the Conclusions section.

MATERIALS AND METHODS

This section begins with a brief introduction to concepts of Riemannian geometry on SPD matrices. Then, we cast dimensionality reduction as an optimization problem and consider two cost functions encoding different criteria. Finally, we describe the datasets in which we applied our classification pipelines.

We denote by $X_k \in \mathbb{R}^{n \times T}$ the recording of T samples on n electrodes of the k^{th} trial in an ensemble of K trials and y_k the class associated to X_k. The spatial covariance matrix C_k of X_k is a $n \times n$ matrix estimated using

$$C_k = \frac{1}{T-1} X_k X_k^T. \tag{1}$$

Riemannian geometry of SPD matrices: Given enough samples, a covariance matrix estimated with (1) is symmetric positive definite (SPD), which means that all of its eigenvalues are strictly positive. Matrices with such property form a manifold $\mathcal{M}$, a set of points with the property that the neighborhood of each $\boldsymbol{x} \in \mathcal{M}$ can be mapped to an Euclidean space, also known as its tangent space $T_{\boldsymbol{x}}\mathcal{M}$. When associated to a metric, $\mathcal{M}$ becomes a Riemannian manifold and fundamental geometric notions are naturally defined, such as geodesics (shortest curve joining two points), distance between two points (length of the geodesic connecting them), the center of mass of a set of points, etc.

We denote the manifold of SPD matrices by $\mathcal{S}_n^{++}$ and endow it with the affine-invariant Riemannian metric. This metric induces a distance between any two matrices, as [6]

$$\delta(C_i, C_j) = \| \log(C_i^{-1/2} C_j C_i^{-1/2}) \|_F, \tag{2}$$

offering a more appropriate distance in the SPD space as compared to the Euclidean distance. In fact, it is possible to show that $\mathcal{S}_n^{++}$ is a manifold with nonpositive curvature [6], so concepts from Euclidean geometry do not necessarily apply. For instance, the sum of angles in a triangle is different than 180 degrees (see Figure 1).

The center of mass M according to distance (2) of a set of covariance matrices $\{C_1, \ldots, C_K\}$ is defined as [1]

$$M = \underset{M \in \mathcal{S}_n^{++}}{\text{argmin}} \sum_{k=1}^{K} \delta^2(M, C_k). \tag{3}$$

Note that M is the point in the manifold minimizing the dispersion (variance) of the set of matrices. When $n = 1$ (C_k is a strictly positive scalar), M corresponds to the geometric mean of the C_k's.

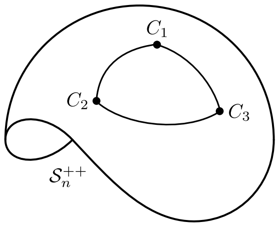

Figure 1: The manifold $\mathcal{S}_n^{++}$ is portrayed as a surface with nonpositive curvature. The distance between any two elements is the length of the geodesic.

This explains why many researchers adopt the term "geometric mean" to refer to the center of mass of a set of covariance matrices. The geometric mean of two SPD matrices is the half-way point of the geodesic that connects them. For $K > 2$, there is no closed form solution for M, so one has to resort to iterative algorithms [2]. The above definitions suffice for the intents of this paper. The interested reader will find a thorough treatment of the subject in the book of R. Bhatia [6].

Dimensionality reduction: Our approach for dimensionality reduction determines a map that takes a set of matrices $\{C_k\}$ in $\mathcal{S}_n^{++}$ to a new set $\{C_k^{\downarrow}\}$ in $\mathcal{S}_p^{++}$ ($p < n$) and keeps a maximum amount of information (under some criterium) from the original matrices. To do so, we search for a p-dimensional subspace of $\mathbb{R}^n$ containing the most relevant features spanned by the columns of the original C_k's. This subspace is represented by a matrix $W \in \mathbb{R}^{n \times p}$ whose columns form a basis for the subspace. We use W to select linear combinations of electrodes in X_k via

$$X_k^{\downarrow} = W^T X_k,$$

which is the same as calculating

$$C_k^{\downarrow} = W^T C_k W \in \mathbb{R}^{p \times p}. \tag{4}$$

Without loss of generality, we impose W to be an orthonormal matrix. Note that because W is full rank the dimension-reduced matrices are guaranteed to be positive definite.

The procedure for choosing W is cast as an optimization problem,

$$\begin{aligned} &\text{minimize} && \mathcal{L}(W), \\ &\text{subject to} && W^T W = \mathbf{I}_p, \end{aligned} \tag{5}$$

where $\mathcal{L}$ is a loss function that encodes the criteria for reducing the dimension of the covariance matrices. One possible criterium is that of making sure that the distances of points C_k to a given "landmark" L do not change very much for the dimension-reduced matrices in $\mathcal{S}_p^{++}$. This can be written formally as

$$\mathcal{L}_u(W) = \sum_{k=1}^{K} \Big(\delta^2(C_k, L) - \delta^2(W^T C_k W, W^T L W) \Big).$$

If we choose L to be the geometric mean of the set of matrices $\mathbb{C}$, the loss function $\mathcal{L}_u$ is the one proposed in [4].

Note that $\mathcal{L}_u$ is based on an unsupervised criterium, since it does not assume knowledge of the labels y_k of each covariance matrix. In the supervised case, W can be chosen to enforce the separability of classes in the reduced-dimension manifold, as in the function

$$\mathcal{L}_s(W) = \sum_{i=1}^{K}\sum_{j=1}^{K} A_{ij}\delta^2(W^T C_i W, W^T C_j W),$$

where the A_{ij}'s encode a measure of affinity between matrices C_i and C_j, so that

$$A_{ij} = g_w(C_i, C_j) - g_b(C_i, C_j),$$

with

$$g_w(C_i, C_j) = \begin{cases} 1, & \text{if } C_i \in \mathcal{N}_w(C_j) \text{ or } C_j \in \mathcal{N}_w(C_i) \\ 0, & \text{otherwise} \end{cases},$$

and

$$g_b(C_i, C_j) = \begin{cases} 1, & \text{if } C_i \in \mathcal{N}_b(C_j) \text{ or } C_j \in \mathcal{N}_b(C_i) \\ 0, & \text{otherwise} \end{cases},$$

where $\mathcal{N}_w(C_i)$ is the set of n_w nearest neighbours of C_i with the same label as y_i and $\mathcal{N}_b(C_i)$ contains the n_b nearest neighbours whose labels are different from y_i. With this definition, $\mathcal{L}_s$ tries to preserve the distances between each pair of matrices in the dimension-reduced space while at the same time enhancing the class separability: for large positive values of A_{ij} (within class) the dimension-reduced matrices are encouraged to come closer to one another, while for small negative values (between classes) their distances tend to increase. Figure 2 illustrates the two aforementioned criteria.

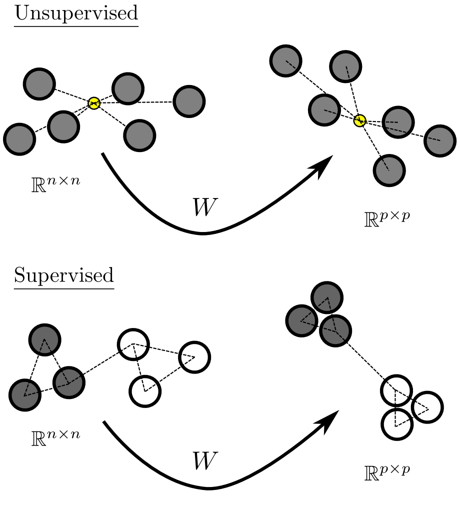

Figure 2: Illustration of the priorities for each type of dimensionality reduction. In the unsupervised case, the distances to a landmark point are preserved, while for the supervised approach the intra-class distances decrease and the inter-class distances tend to augment.

We should mention that the computational cost for calculating $\mathcal{L}_u$ and $\mathcal{L}_s$ is not comparable. In the unsupervised case the number of operations increases linearly with K since all distances are calculated with respect to a single landmark. In the supervised algorithm the number of operations scales quadratically with K, a rather problematic aspect when working with large datasets.

Problem (5) has a special structure and can be solved as an optimization problem on manifolds, a branch of applied mathematics with a considerable amount of recent research [7] and excellent computational tools available online, such as the Python package `pymanopt` [9] used in this work. In particular, we use a version of the conjugate gradient algorithm adapted for manifold optimization and solved (5) considering the W matrices as elements of a Grassmann manifold. We will not delve into more the details of these procedures, but the interested reader will find more information in [4] and [7].

Classification pipeline: We classify each trial X_k via the minimum distance to mean (MDM) algorithm. It determines the geometric mean of the covariance matrices in each class of the training set and then assigns to each matrix in the test set the class to which the distance to the mean is the smallest [8].

We compare three different pipelines for classification:

MDM: No dimensionality reduction (DR) and classification using the MDM algorithm.

unsDR + MDM: Unsupervised DR with $\mathcal{L}_u$ as cost function and landmark L fixed to the geometric mean of the dataset. Classification using MDM.

supDR + MDM: Supervised DR with $\mathcal{L}_s$ as cost function, n_w always fixed to the minimum number of elements in each class and n_b chosen via cross-validation. Classification using MDM.

The performance of each pipeline is assessed via a 10-fold cross-validation procedure and compared by their AUC (area under the receiver operating characteristic curve).

Datasets: We carried out our analysis on three datasets, two from MI experiments and one using the P300 paradigm. The first MI database comes from the BCI Competition III – Dataset IV [10] and contains recordings from 5 subjects with 118 electrodes. We applied our classification pipelines on 140 trials corresponding to tasks of left and right imagined hand movements (70 for each class). The second MI database is available at the Physionet website [11] and comprises recordings on 64 electrodes from 109 subjects. We only used the data from tasks of imagined hands and feet movement, which corresponds to approximately 44 trials per subject (22 for each class). The P300 dataset comes from experiments performed in our laboratory on the P300-based game Brain Invaders [12]. We used data from 32 electrodes on 38 subjects with 720 trials each (120 target and 600 non-target).

The data from each BCI paradigm were processed differently. For MI we filtered the EEG signals in the 8-30 Hz band and considered each trial as a segment from 0.5 to 2.5s after each trial onset. We estimated the spatial covariance matrices using (1). For the P300 data we used filters from 1 to 20 Hz and considered each epoch with a duration of one second and starting just after a flash. We used the approach described in [13] to estimate a special form of covariance matrices capturing signals of interest in event-related potentials.

RESULTS AND DISCUSSION

This section describes the analysis on each dataset and discuss the obtained results.

BCI III-IV: We began our investigations on a dataset where dimensionality reduction is of major concern, because of its 118×118 covariance matrices. We compared the classification pipelines with different values of p, the dimension of the reduced covariance matrices, and n_b, the number of neighbors considered in $\mathcal{N}_b(C_i)$ for the supervised DR. The three values of p were chosen in the following way: obtain the geometric mean M of the covariances of the dataset (all classes together) and compute its eigenvalue decomposition. Sort the eigenvalues in decreasing order and select the values of p for which their cumulative sum equals to at least 80%, 95% and 99% of their total sum. For the BCI III-IV dataset this corresponds to p = 4, 12, and 32, respectively.
The results in Figure 3 show that for $p = 32$ the AUC of pipelines with dimensionality reduction were at least equivalent to those using all 118 available electrodes. This can be explained by the low-dimensional structure of the subspace spanned by the columns of the spatial covariance matrices. Consequently, most of the variance of these matrices is associated to their first few principal vectors. In contrast, reducing the dimensions to $p = 4$ degrades the classification performance on most subjects, a consequence of the loss of discriminatory features in the reduced matrices. Figure 3 also indicates that the parameter n_b of supervised DR does not seem to have much influence over the scores of the pipelines.

Physionet: In this second dataset we tested the performance of classification pipelines on a wide range of individuals. Having data from so many subjects allows us to observe certain patterns and make general conclusions that would be difficult otherwise. Figure 4 displays the results on three subjects for multiple values of n_b and fixed $p = 24$. For certain choices of n_b the score with supervised DR was higher than the other pipelines, but in general we did not observe any considerable improvement. In fact, one could include a grid-search step to the pipeline with supervised DR for choosing the best value of n_b for each subject. However, this would lead to a considerable increase in processing time, since the quadratic scaling of supervised DR makes it a quite expensive operation by itself. With this in mind, we fixed $n_b = 10$ in all of the following analysis, accepting the compromise that it might not be the optimal value for all subjects.

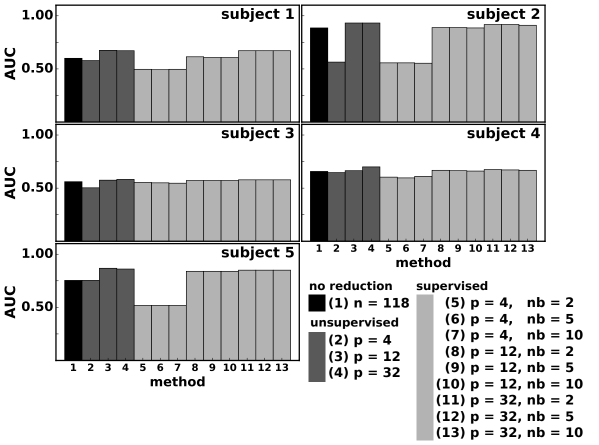

Figure 3: AUC of the classification pipelines on five subjects from dataset BCI III-IV. We considered pipelines with $p \in \{4, 12, 32\}$. For the supervised DR we fixed $n_w = 70$ and varied n_b in $\{2, 5, 10\}$.

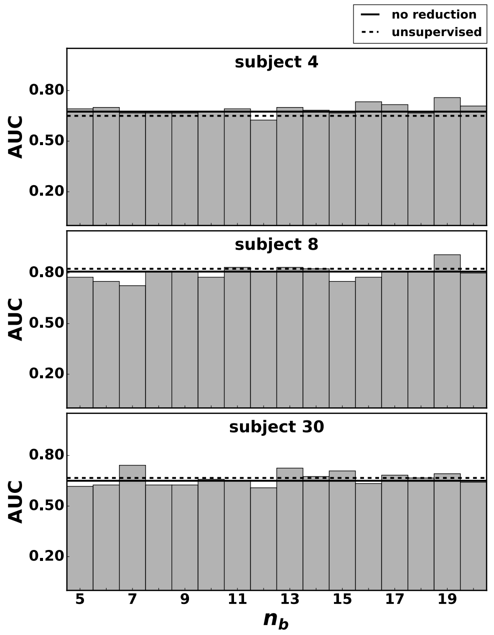

Figure 4: AUC of the classification pipelines with supervised DR on three subjects from the Physionet database. We considered multiple values of n_b and fixed $n_w = 22$ and $p = 24$. Horizontal lines correspond to AUCs of pipelines with no dimensionality reduction and unsupervised DR.

Figure 5 compares the performances of the classification pipelines on all subjects for different values of p and fixed $n_b = 10$. The curves in each plot correspond to the AUC of each pipeline in decreasing order. We observe the same behavior as before: on most subjects, when the dimension of the reduced matrices (e.g. $p = 4$) is small, the AUC of the pipeline with full matrices (64×64) is higher as compared to both dimensionality reduction methods. The score of all pipelines become close to one another when p increases. Another important observation from Figure 5 is that the classification performance of the pipelines varies smoothly with the choices of the dimension p of the reduced covariances. This is of great practical value because it demonstrates that we do not need to choose a precise p for attaining good results; there exists a certain range where all choices are equivalent.

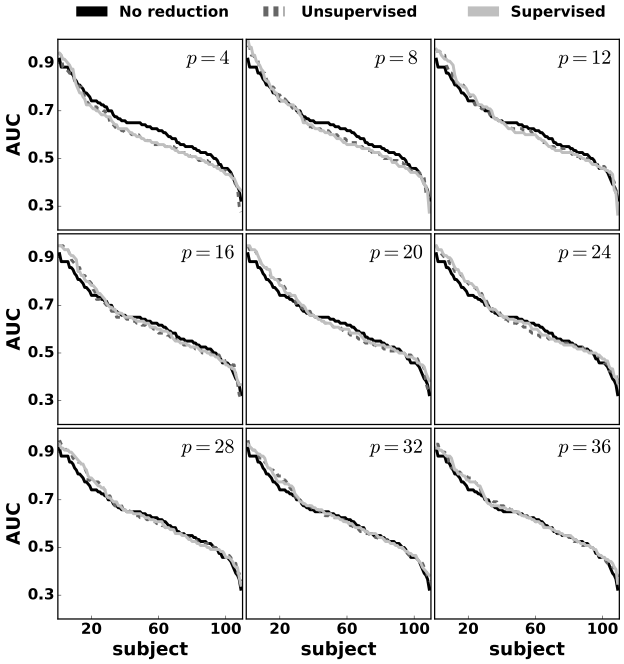

Figure 5: AUC scores in decreasing order for classifications on all subjects from the Physionet database. We fixed $n_b = 10$ and $n_w = 22$, and considered the values of p indicated in the figure.

P300: The results for our investigations of the P300 dataset are displayed in Figure 6. We compare once again the classification performance of a pipeline without dimensionality reduction (64×64 matrices) to classifiers using either an unsupervised or a supervised approach. We did all analysis with fixed $n_b = 10$ and considered multiple values of p. We observed the same behavior as before for the performance of dimensionality reduction algorithms: when p is too small the pipelines with DR are clearly inferior, as seen for $p = 8$, whereas for higher p the performances are all very similar.
The computing time for supervised DR in the P300 paradigm was excessively high, mainly because of the large number of trials in the dataset. We tried using a smaller set of trials, but in this case the classification performance of all pipelines were lower. In fact, usually P300 BCI systems are expected to improve their performance when more trials are available, so having a dimensionality reduction step that does not scale well with their number is problematic.

Comparing all pipelines: Besides investigating the conditions in which a dimensionality reduction would be useful or not, we tested whether any of the methods had a globally superior performance on the P300 and Physionet datasets. In theory, we expect the supervised approach to have better results because of the extra information it has concerning the labels of each covariance matrix. To test this hypothesis, we rearranged the results from Figures 5 and 6 into the plots in Figure 7, where each axis contains the AUC of a different pair of pipelines.

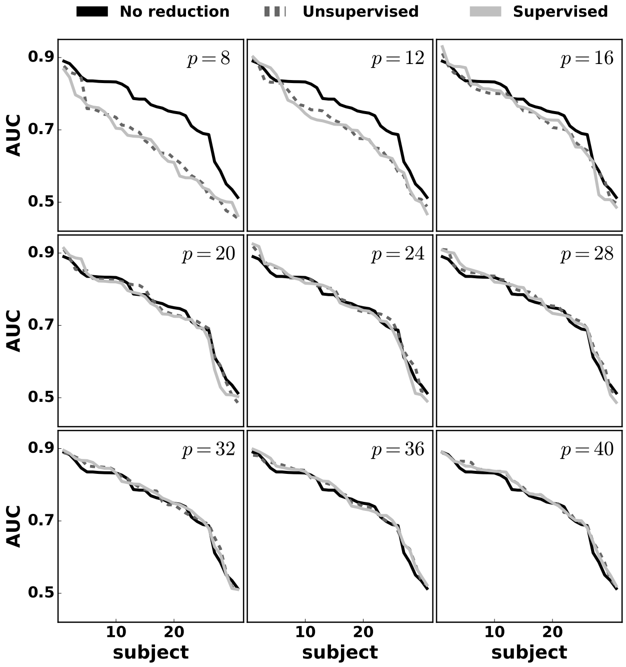

Figure 6: AUC values in decreasing order for the three pipelines applied to all subjects from the P300 database. We fixed $n_b = 10$ and $n_w = 120$, and considered the values of p indicated in the figure.

We estimated regression lines with intercept fixed to the origin for each plot and used a F-statistic to test if we could reject the hypothesis of its slope being equal to one. None of the statistical tests rejected the null hypothesis with type I error fixed to 5%, meaning that nothing can be said about one pipeline being consistently better than the others. This result indicates that the extra information used by the supervised DR is not enough for improving its classification power. It also means that adding a dimensionality reduction step to a classification pipeline does not harm its performance, a very useful fact that alleviates the computational burden of processing high-dimensional features using Riemannian geometry.

CONCLUSION

In this work, we evaluated two methods for reducing the dimension of positive-definite matrices and compared their scores in classification tasks on different BCI datasets. We observed that reducing too much the dimension discards important information from the original high-dimensional space and degrades the classification performance. Also, the choice of p showed a smooth influence over the scores of the classification pipelines, a very useful result in practice.

Our statistical tests did not reject the hypothesis of each pair of pipelines having equivalent performances, indicating that it is possible to reduce the dimensions of a spatial covariance matrix without losing classification performance. We should point out that we probably did not obtain better results for the supervised DR because we did not use a grid search for choosing the best n_b on each subject. However, if we had included this step the algorithm would have become impractical, because of the computational power that minimizing the loss function $\mathcal{L}_s$ demands.

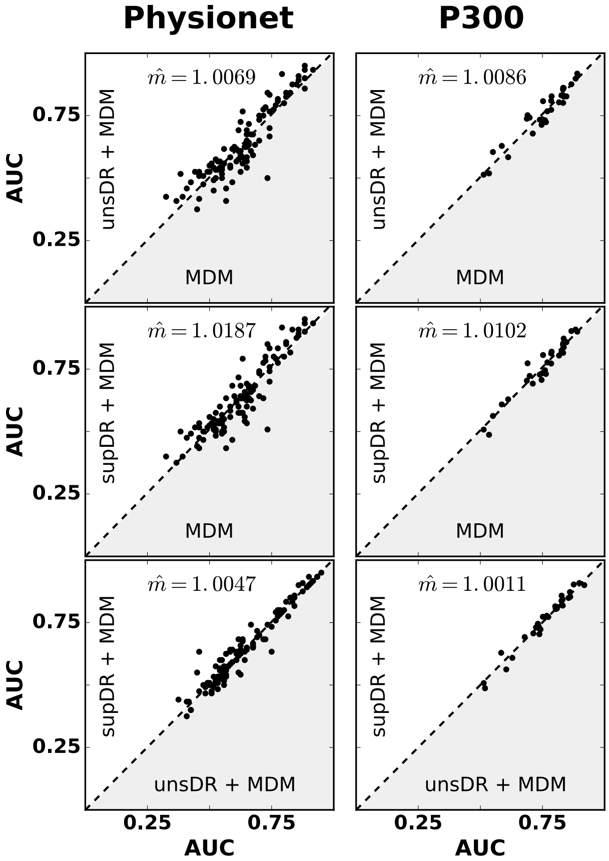

Figure 7: Scatter plots with the AUC scores of each pair of pipelines in the axis. We used only the results for $p = 24$ on both datasets. Coefficient $\hat{m}$ is the slope of a regression line with intercept fixed at the origin.

In comparison to [5], our investigations on BCI signals were more thorough. We explored the effects of the choice for the reduced covariance matrices, used a dataset containing many more subjects and a BCI paradigm that had not been considered until now. In future work, we intend to explore new options for performing supervised DR. The approach proposed by [4] does not scale well for large datasets and we believe that there are better alternatives. Also, we would like to explore more deeply the effects of reducing the dimensions of covariance matrices, not only in terms of classification power but as a general problem in Riemannian geometry. Finally, we should extend our comparisons to other proposals available in the literature for reducing the dimension of EEG signals.

AKNOWLEDGEMENTS

This work was partly supported by the ERC grant CHESS (2012-ERC-AdG-320684).

REFERENCES

[1] Congedo, M., Barachant A., Bhatia R., Riemannian Geometry for EEG-based Brain Computer Interfaces; a primer and a review. Brain Computer Interfaces. 2017. In press.

[2] Congedo, M., Barachant, A., Koopaei, E. Fixed Point Algorithms for Estimating Power Means of Positive Definite Matrices, IEEE Transactions in Signal Processing. 2017. In press.

[3] Lotte, F., Congedo, M., Lecuyer, A., Lamarche, F., Arnaldi, B., A review of classification algorithms for EEG-based brain–computer interfaces, Journal of Neural Engineering. 2007;4(2):1741-2560.

[4] Harandi, M, Salzmann, M., Hartley, R., Dimensionality Reduction on SPD Manifolds: The Emergence of Geometry-Aware Methods, preprint arXiv:1605.06182.

[5] Horev, I. and Yger, F., Masashi S., Geometry-Aware Principal Component Analysis for Symmetric Positive Definite Matrices, in Proceedings of the 7th ACML, Hong Kong, 2015, 1–16.

[6] Bhatia, R. Positive Definite Matrices, Princeton University Press, Princeton, United States (2009).

[7] Absil, P.-A., Mahony, R., Sepulchre, R. Optimization Algorithms on Matrix Manifolds, Princeton University Press, Princeton, United States (2008).

[8] Barachant, A., Bonnet, S., Congedo, M., Jutten, C., Multiclass Brain-Computer Interface Classification by Riemannian Geometry, IEEE Transactions on Biomedical Engineering. 2012;59(4):920-928.

[9] Townsend, J., Koep, N., Weichwald, S., Pymanopt: A Python Toolbox for Manifold Optimization using Automatic Differentiation, preprint arXiv:1603.03236

[10] Schlogl, A., Lee, F., Bischof H., Pfurtscheller, G., Characterization of Four-Class Motor Imagery EEG Data for the BCI-Competition 2005, Journal of Neural Engineering. 2005;2(4):L14-L22.

[11] Schalk, G., McFarland, D. J., Hinterberger, T., Birbaumer N., Wolpaw, J. R., BCI2000: A general-purpose brain-computer interface (BCI) system, IEEE Transactions on Biomedical Engineering. 2004;51(6):1034-1043.

[12] Congedo, M., et al., “Brain Invaders”: a prototype of an open-source P300-based video game working with the OpenViBE platform, in Proc. 5th International BCI Conference, Graz, Austria, 2011, 280-283.

[13] Barachant, A. and Congedo, M., A plug & play P300 BCI using information geometry, preprint arXiv: 1409.0107