Transfer learning: a Riemannian geometry framework with applications to Brain-Computer Interfaces

Paolo Zanini, Marco Congedo, Christian Jutten, Salem Said, Yannick Berthoumieu

Transfer learning: a Riemannian geometry framework with applications to Brain-Computer Interfaces

Paolo Zanini, Marco Congedo, Christian Jutten, Salem Said, and Yannick Berthoumieu

***Abstract*—Objective: This paper tackles the problem of transfer learning in the context of EEG-based Brain Computer Interface (BCI) classification. In particular the problems of cross-session and cross-subject classification are considered. These problems concern the ability to use data from previous sessions or from a database of past users to calibrate and initialize the classifier, allowing a calibration-less BCI mode of operation. Methods: Data are represented using spatial covariance matrices of the EEG signals, exploiting the recent successful techniques based on the Riemannian geometry of the manifold of Symmetric Positive Definite (SPD) matrices. Cross-session and cross-subject classification can be difficult, due to the many changes intervening between sessions and between subjects, including physiological, environmental, as well as instrumental changes. Here we propose to affine transform the covariance matrices of every session/subject in order to center them with respect to a reference covariance matrix, making data from different sessions/subjects comparable. Then, classification is performed both using a standard Minimum Distance to Mean (MDM) classifier, and through a probabilistic classifier recently developed in the literature, based on a density function (mixture of Riemannian Gaussian distributions) defined on the SPD manifold. Results: The improvements in terms of classification performances achieved by introducing the affine transformation are documented with the analysis of two BCI data sets. Conclusion and significance: Hence, we make, through the affine transformation proposed, data from different sessions and subject comparable, providing a significant improvement in the BCI transfer learning problem.**

***Index Terms*—Brain Computer Interface, electroencephalography, covariance matrices, Riemannian geometry, mixtures of Gaussian.**

I. Introduction

A Brain Computer Interface (BCI) is a system capable of predicting or classifying cognitive states and intentions of the user through the analysis of neurophysiological signals [24], [32]. Historically, BCIs have been developed to allow severely paralyzed people to communicate or interact with their environment without relying on the normal muscular or peripheral nerve outputs [8]. More recently, BCIs have been proposed also for healthy people, for instance in driving, forensics, or gaming applications [11], [20], [29]. Several neurophysiological signals can be used for a BCI, either invasive or semi-invasive, like electrodes implanted into the grey matter or sub-durally. Most BCIs however make use of non-invasive neuroimaging modalities, such as near-infrared spectroscopy and, especially, electroencephalography (EEG), which suit both clinical and healthy populations. In this paper we focus on EEG-based BCIs.

The standard classification technique consists of two operational stages [9], [18]. First, EEG signals of a training set are transformed through frequency and/or spatial filters in order to extract discriminant features [8], [16]. A very popular filter in this stage is the Common Spatial Pattern (CSP) [18], [19]. Second, the features enter a machine learning algorithm in order to compute a decision function for performing classification on the test set. This is done by supervised techniques like, for instance, Linear Discriminant Analysis (LDA) [9].

A different approach was presented in [2], where classification is performed using the signal covariance matrices as feature of interest. Covariance matrices do not belong to an Euclidean space, instead they belong to the smooth Riemannian manifold of Symmetric Positive Definite (SPD) matrices [5]. Hence, in [2], the properties of SPD manifold are used to perform BCI classification directly on the manifold, as illustrated in subsection II-D. In this paper we consider two separate improvements with respect to the method described in [2]. The first improvement relates to the classification techniques. In [2] the authors used a basic classifier, named Minimum Distance to Mean (MDM), which takes into account distances on the manifold between the observations and some reference points of the classes, known as centers of mass, means, or barycenters. Here we introduce a probabilistic classifier, modeling the class probability distributions, exploiting Riemannian Gaussian and mixture of Gaussian distributions introduced in [34], and applied to EEG classification in [37]. The second improvement relates to the problem of transfer learning [30]. In the machine learning field, transfer learning is defined as the ability to use previous knowledge as features in a new task/domain related to the previous one. Some examples of transfer learning applied to BCI problem can be found in [15], [21], [27] and [36]. In this paper we focus specifically on the problem of *cross-session* and *cross-subject* BCI *learning*. A classical BCI requires a calibration stage at each run, even for a known user. The calibration stage, however short, is inconvenient both for patients, because it wastes part of their limited attention, and for the general public, which is

P. Zanini is at Gipsa-Lab, Université Grenoble Alpes, France, and IMS, Université de Bordeaux, France (e-mail: paolo.zanini@gipsa-lab.fr).

M. Congedo and C. Jutten are at Gipsa-Lab, Université Grenoble Alpes, France (e-mail: marco.congedo@gipsa-lab.fr; christian.jutten@gipsa-lab.fr).

S. Said and Y. Berthoumieu are at Univ. Bordeaux, Bordeaux INP, CNRS, IMS, UMR 5218, 33400 Talence, France (e-mail: salem.said@ims-bordeaux.fr; yannick.berthoumieu@ims-bordeaux.fr).

usually unwilling to undergo repeated calibration sessions. As proposed in [12] a BCI should be able to calibrate online while it is being used. The problem is then to provide a workable initialization, that is, one that allows the operation of the BCI at the very beginning of the session, even if suboptimal. For a new user, a database of past users can be considered to initialize the classifier. This form of learning is referred to as cross-subject learning. From the second usage on, past data from previous sessions of the user can be employed. This is referred to as cross-session learning. Cross-session learning is known to be a difficult task due to several changes intervening in between the sessions, including physiological, environmental, as well as instrumental changes (e.g., electrode positioning and impedance). Even more difficult is the cross-subject learning, because the spatial and temporal configuration of brain dipolar sources is subject to substantial individual variability. In the Riemannian framework the cross-session and cross-subject changes can be understood as geometric transformations of the covariance matrices. In this work we will refer to this geometric transformation as a "shift", although we should keep in mind that a transformation may entail more than a simple displacement on the manifold.

A first attempt to solve the shift problem is described in [33], however this work does not consider the structure of the covariance matrix manifold. In [3], instead, the authors introduce a way to solve the shift problem in a Riemannian framework, for the cross-session situation, however this approach depends on the order of the tasks performed during an experiment and on the (unknown) structure of the classes in the classification problem. In this paper we develop an idea similar to the one presented in [33], but in a Riemannian framework. Our approach does not depend on the (unknown) label sequence of the observations obtained during the experiment. We assume that different source configurations and electrode positions induce shifts of covariance matrices with respect to a reference (resting) state, but that when the brain is engaged in a specific task, covariance matrices move over the SPD manifold in the same direction. This assumption allows a workable model and a simple solution thanks to the congruence invariance property of SPD matrices (that we will describe in subsection II-A). We will center the covariance matrices of every session/subject with respect to a reference covariance matrix so that what we observe is only the displacement with respect to the reference state due to the task. We estimate a reference matrix for every session, but different between sessions and between subjects. Then, we perform a congruent transformation of our data using this reference matrix. In this way observations belonging to the same session and subject do not change their relative distances and geometric structure. However, since the reference matrix varies among sessions and among subjects, these data are moved in the manifold in different directions and, if the reference matrix is chosen accurately, data from different sessions/subjects become comparable. As we will show with the analysis of two BCI data sets, this procedure provides an efficient initialization for cross-session and cross-subject classification problems.

In EEG-based BCI literature, different kinds of tasks can be used to design a BCI (see [12] for an exhaustive description). In this work we analyze two different paradigms in order to widen the scope of our analysis. The first one relates to a Motor Imagery (MI) paradigm and the second one to an Event-Related Potential (ERP) paradigm. For the first dataset we analyze nine subjects, each one performing two sessions, and we evaluate the accuracy for cross-session and cross-subject classification. We obtain significant improvements by using the proposed procedure, especially for the cross-subject classification, where we can increase the performance by 30% in some cases. For the second dataset we analyze 17 subjects and we evaluate the precision for cross-subject classification. Also in this case we obtain substantial improvements by introducing our procedure. Furthermore, for both datasets, we discuss the situations where the introduction of a probabilistic classifier can result in further improvements.

The paper is organized as it follows. In Section II basic concepts of Riemannian geometry are introduced. In Section III the two BCI paradigms are described in details, focusing in particular on how to build SPD matrices in the two cases to be used in a Riemannian framework. Then, in Section IV we describe the proposed Riemannian transfer learning methods. In Section V we present the results obtained with the two datasets analyzed. Finally, we conclude our work in Section VI.

II. Elements of Riemannian Geometry

In this section we present some basic properties of the space of SPD matrices, introducing a probabilistic distribution on this space and defining some classification rules to classify SPD matrices.

A. Manifold of SPD matrices: basic concepts

We start by introducing $M(n)$ and $S(n)$ as the vector space of $n \times n$ square matrices, and the vector space in $M(n)$ of symmetric $n \times n$ square matrices, respectively. Specifically, $M(n) = \{M \in \mathbb{R}^{n\times n}\}$, while $S(n) = \{S \in M(n), S = S^T\}$. The set of SPD matrices $P(n) = \{P \in S(n), \mathbf{u}^T P\mathbf{u} > 0 \ \forall \mathbf{u} \in \mathbb{R}^n, \mathbf{u} \neq 0\}$ is an open subset of $S(n)$, in particular it is an open convex cone of dimension $\frac{n(n+1)}{2}$. $P(n)$ is the space of covariance matrices and it is our space of interest. If endowed with the Fisher-Rao metrics [5], $P(n)$ turns out to be a smooth Riemannian manifold with non positive curvature. This means that for every point $P \in P(n)$, in the tangent space T_P (that in this case can be identified with $S(n)$), we define a scalar product which varies smoothly with P. The local inner product and, as a consequence, the local norm, are defined as

$$\begin{aligned} \langle U, V\rangle_P &= \text{tr}(P^{-1}UP^{-1}V), \\ \|U\|_P^2 &= \langle U, U\rangle_P, \end{aligned} \tag{1}$$

respectively, where $U, V \in S(n)$. Through the natural metrics (1), a distance between two points $P_1, P_2 \in P(n)$ can be

defined as the length of the unique shortest curve (called geodesic) connecting P_1 and P_2 [5]

$$\delta(P_1,P_2)=\|\log(P_1^{-1/2}P_2P_1^{-1/2})\|_F=\left(\sum_{i=1}^{n}\log^2\lambda_i\right)^{1/2}, \tag{2}$$

with $\|\cdot\|_F$ the Frobenius norm, and $\lambda_1,...,\lambda_n$ the eigenvalues of $P_1^{-1/2}P_2P_1^{-1/2}$ (or $P_1^{-1}P_2$, with the indices 1 and 2 that can be permuted since $\delta(\cdot,\cdot)$ is symmetric). The Riemannian distance $\delta(\cdot,\cdot)$ has two important invariances:

i. $\delta(P_1^{-1},P_2^{-1})=\delta(P_1,P_2)$;

ii. $\delta(C^TP_1C,C^TP_2C)=\delta(P_1,P_2)\ \forall C\in GL(n)$,

with $GL(n)=\{C\in M(n),\ C \text{ invertible}\}$ the set of invertible matrices. Property ii, called congruence invariance, means that the distance between two SPD matrices is invariant with respect to a change of reference, i.e., to any linear invertible transformation in the data (recordings) space. This property will be particularly important in the following.

B. Center of mass of a set of SPD matrices

The simplest statistical descriptor of a set of objects is the concept of mean value, which is meant to provide a suitable representative of the set. The most famous mean is the arithmetic mean. It has an important variational characterization: given a set $P_1,...,P_N$ of SPD matrices, the arithmetic mean $\mathcal{A}(P_1,...,P_N)$ is the point P which minimizes the sum of squared Euclidian distances $d_e(\cdot,\cdot)$

$$\mathcal{A}(P_1,...,P_N)=\arg\min_{P\in P(n)}\sum_{i=1}^{N}d_e^2(P_i,P), \tag{3}$$

Similarly, it has been shown that we can use the Riemannian distance to define a geometric mean, or center of mass, of a set of SPD matrices, through a variational approach [6]. The center of mass $\mathcal{G}(P_1,...,P_N)$ is defined as the point of the manifold satisfying

$$\mathcal{G}(P_1,...,P_N)=\arg\min_{P\in P(n)}\sum_{i=1}^{N}\delta^2(P_i,P). \tag{4}$$

with $\delta(\cdot,\cdot)$ defined in (2). In the literature, (4) is often called Cartan/Fréchet/Karcher mean [5], [6], [22]. Since $P(n)$ is a Riemannian manifold of non-positive curvature, existence and unicity of the Riemannian mean can be proved [1], [28]. However, an explicit solution exists only for $N=2$, where it coincides with the middle point of the geodesic connecting the two SPD matrices of the set. For $N>2$ a solution can be found iteratively and several algorithms following different approaches have been developed in the literature [22]. Some of them try to find the right value through numerical procedure like deterministic line search [17], [26], simple or stochastic gradient descent [10], [31]. Other faster and computational lighter approaches look for some suitable approximation of the center of mass, see for instance [6], [13], [14].

An important invariance property for the center of mass is:

$$\mathcal{G}(C^TP_1C,...,C^TP_NC)=C^T\mathcal{G}(P_1,...,P_N)C\ \ \forall C\in GL(n),$$

inherited from the congruance invariance of the Riemannian distance mentioned above. This result means that the center of gravity is shifted through the same affine transformation as the matrices of the set.

C. Mixtures of Gaussian distributions on the manifold of SPD matrices

Distance and center of mass are geometric concepts concerning the properties of the manifold of SPD matrices, but they do not concern any probabilistic assumptions on a sample of SPD matrices. To consider a probabilistic model we introduce a class of probability distributions on the space $P(n)$, called *Riemannian Gaussian distributions* and defined in [34]. It will be denoted $G(\overline{P},\sigma)$ and depends on two parameters, $\overline{P}\in P(n)$ and $\sigma>0$. It is defined by its probability density function

$$f(P|\overline{P},\sigma)=\frac{1}{\zeta(\sigma)}\exp\left(-\frac{\delta^2(P,\overline{P})}{\sigma^2}\right) \tag{5}$$

where $\zeta(\sigma)$ is a normalization function. In [34] it has been shown that, given $P_1,...,P_N$ i.i.d. from (5), the Maximum Likelihood Estimator (MLE) of $\overline{P}$ coincides with the center of mass (4). For the MLE of σ, instead, an efficient procedure is presented in [37]. If we consider only Gaussian distribution, we are not able to describe a wide range of real problems. In general in the classical Euclidean framework, in order to include several distribution shapes, mixtures of Gaussians have been considered [34]. In the Riemannian framework this is also possible in a straightforward way. A mixture of Riemannian Gaussian distributions is a distribution on $P(n)$ whose density function can be written as

$$f(P)=\sum_{m=1}^{M}w_mf(P|\overline{P}_m,\sigma_m), \tag{6}$$

with $w_1,...,w_M$ non-negative weights summing up to 1. The parameters of (6) can be found, for instance, through an Expectation-Maximization (EM) algorithm, as described in [34]. This class of distributions will be used to build a probabilistic classifier for data in $P(n)$, as described in the next subsection.

D. Classification techniques in the manifold of SPD matrices

In [2] the authors proposed a classification procedure based on Minimum Distance to Mean (MDM) classifier, which is defined as it follows: given K classes and a training phase where the centers of mass $\widehat{C}(k)$ of the classes ($k=1,...,K$) are estimated, a new observation C_i is assigned to the $\widehat{k}$ class according to the classification rule

$$\widehat{k}=\arg\min_{k\in\{1,...,K\}}\{d_R(C_i,\widehat{C}(k))\}. \tag{7}$$

This rule takes into consideration the Riemannian distance of the new observation to the centers of mass, ignoring information on the dispersion of the groups, encoded by the parameter σ in the Riemannian Gaussian distribution (5). The principle of Bayesian classification can be used exploiting such

a distribution. In this case, the classification rule based on the a posteriori distribution reads

$$\widehat{k} = \arg \min_{k \in \{1,\dots,K\}} \left\{ \log \zeta(\widehat{\sigma}(k)) + \frac{d_R^2(C_i, \widehat{C}(k))}{2\widehat{\sigma}^2(k)} \right\}, \quad (8)$$

where $\widehat{\sigma}(k)$ is the MLE estimate of the dispersion parameter of the k-th class [37]. Of course, if the $\widehat{\sigma}(k)$ coincide for all classes, (8) reduces to (7). In order not to be limited to a simple class of distributions, we can consider mixtures of Gaussian (6), updating Bayesian classification rule accordingly. In this paper we consider a number of mixture components M varying from 2 to 4.

III. Data

We analyze two different EEG-based BCI datasets, related to MI and ERP frameworks. The way to build SPD matrices is different between the two cases and it is described in subsection III-A and III-B, respectively. Then, in subsection III-C, we will show how cross-session and cross-subject classifications can be problematic, exploiting a visualization technique for high-dimensional data named t-Stochastic Neighbor Embedding (t-SNE) [35].

A. Motor Imagery: data construction

The analyzed dataset is the one from BCI competition [25], already analyzed in [2], [18]. It contains EEG data from nine subjects performing four kinds of motor imagery (right hand, left hand, foot, and tongue imagined movements). A total of 576 trials per subject are available, each trial corresponding to a movement (balanced experiment, i.e., 144 trials per class). Half of the trials (288) are obtained during the first session, and the other half during a second session. For each trial l we register the centered EEG signal $X_l \in \mathbb{R}^{n \times T}$, where n is the number of electrodes and T the number of sample points of the time window considered to evaluate sample covariance, in this case from 0.5 to 2.5 seconds after the stimulus. Then we use for the analysis the empirical covariance matrix defined as

$$C_{X_l} = \frac{1}{T-1} X_l X_l^T.$$

In this experiment signals are recorded using 22 electrodes ($n = 22$), hence covariance matrices here belong to $P(22)$. As usual with motor imagery data, before computing covariance matrices, EEG signals are bandpass filtered by a 5-th order Butterworth filter in the frequency band of 8 – 30 Hz.

B. ERP: data construction

This dataset cames from a *Brain Invaders* experiment carried out at GIPSA-lab in Grenoble, France [11]. Subjects watch a screen with 36 aliens flashing alternatively. They are requested to mentally count the number of specific (known) target alien flashes. This experiment generates in the EEG signals an Event-Related Potential (ERP) named P300 whenever the target alien flashes [11]. The main goal is to detect the target trials from the EEG signals. Thus, we have two classes in this experiment, P300 signals (target class) and normal signals (non target class). In this framework we cannot simply consider the covariance matrices C_{X_l}. Indeed, if we randomly shuffle the time instants for a specific trial, the estimate of its covariance matrix does not change, and thus the classification result. Since temporal information are essential to detect ERP, we augmented the vector by integrating a component related to the temporal profile of the ERP event considered, following the procedure described in [4] and [23]. Specifically, we considered the average ERP response

$$E = \frac{1}{|K^+|} \sum_{l \in K^+} X_l \in \mathbb{R}^{n \times T},$$

where K^+ is the group of target trials (ERP in this case). Then we built an augmented trial signal matrix $\widetilde{X}_l$, defined as

$$\widetilde{X}_l = \begin{bmatrix} E \\ X_l \end{bmatrix} \in \mathbb{R}^{2n \times T},$$

and then we considered an augmented covariance matrix $\widetilde{C}_{\widetilde{X}_l}$ of dimension $2n \times 2n$:

$$\widetilde{C}_{\widetilde{X}_l} = \begin{bmatrix} C_E & C_{EX_l} \\ C_{X_l E} & C_{X_l} \end{bmatrix}.$$

Relevant information for distinguishing a target from a non-target trial is embedded in the block C_{EX_l} (and in its transpose $C_{X_l E}$). In these blocks, entries will be far from zero only for target trials, since only the time series of target trials are correlated to the average ERP E. Thus, on the SPD manifold augmented covariance matrices for target trials will be far from the augmented covariance matrices for non-target trials. Notice that if we randomly shuffle the time instants for a specific trials, the augmented covariance matrix does change, which means that we have effectively embedded the temporal information into these matrices. A training-phase is needed to build the average ERP response. In this experiment we consider 17 subjects, with a number of trials different from one subject to another, ranging from 500 to 750. EEG signals are recorded at a frequency of 512 Hz using 13 electrodes (i.e. $n = 13$), hence covariance matrices here belong to $P(26)$. Every trial is registered for a period of time of one second after the stimulus (the flash). Thus, augmented covariance matrices are estimated using 512 observations.

C. Data visualization using t-SNE

The visualization technique called t-SNE [35], visualizes high-dimensional data by mapping each point to a location in a 2- or 3-dimensional space, while optimizing the pairwise distances in the reduced space with respect to the distances in the original manifold. In our case we aim to represent each covariance matrix as a point in a 2 dimensional space in order to appreciate the effect of cross-session and cross-subject shift.

In Figure 1 and 5 the data from the MI experiment are shown. In each plot of Figure 1, data for the two sessions are depicted (circles for session 1 and crosses for session 2), with colors identifying the classes. In Figure 5 a more detailed representation of subject 9 is depicted, with plots divided by class. We can observe that data relative to session 2 are shifted with respect to session 1, for every subject. This means that, in

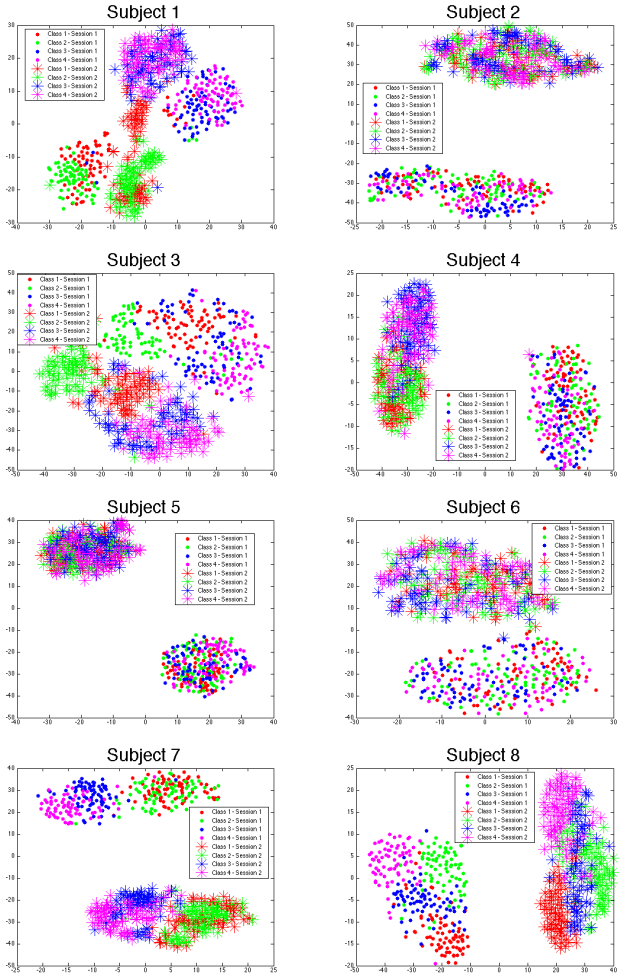

Fig. 1. Motor Imagery dataset: for every subject the original covariance matrices for session 1 (circles) and 2 (crosses) are depicted. Data from different sessions are well separated. Data are also colored by classes (right hand, left hand, foot, and tongue imagined movements). Visualization obtained through t-SNE method using the Riemannian distance (2).

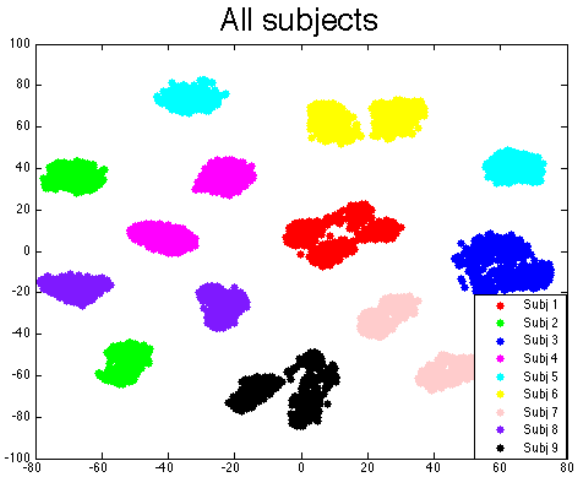

Fig. 2. Motor Imagery dataset: original covariance matrices of all subjects. There are two data groups for each subject, related to session 1 and 2. Visualization obtained through t-SNE method using the Riemannian distance (2).

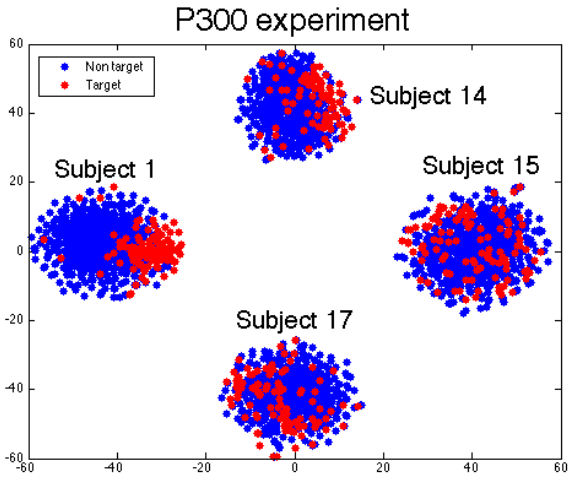

Fig. 3. P300 dataset: original covariance matrices of some subjects. Data are colored by classes (target and non-target). Visualization obtained through t-SNE method using the Riemannian distance (2).

the original space, the two groups (session 1 and session 2) are well separated, and that the cross-session classification is not a trivial problem. In Figure 2, instead, the data of all subjects are depicted together, showing an even worse separation among subjects.

Regarding the P300 experiment, we have one session per subject, thus we focus only on the cross-subject transfer learning. The augmented covariance matrices for four subjects are depicted in Figure 3. Even in this case it is clear that data related to different subjects are far away from each other, making cross-subject classification in the original data space hopeless.

IV. Methods

From the visualization analysis of section III-C, it is clear that a data transformation is needed in order to make cross-session and cross-subject classification efficient. If we consider again Figure 1 (and Figure 5 for more detailed pictures, separated by class, related to subject 9), apart from the shift, data coming from different sessions present a similar shape. We assume that from one session to another, what it is changing can be captured in a "reference state", whereas covariance matrices move in a consistent direction according to the task performed by the subject. This assumption leads to the idea introduced in [33], which we here translate in the Riemannian framework: let $\overline{R}^{(1)}$ and $\overline{R}^{(2)}$ be the centers of mass (unknown in principle) of the reference state for session 1 and 2, respectively. Let $\{C_1^{(1)}, ..., C_{N_1}^{(1)}\}$ and $\{C_1^{(2)}, ..., C_{N_2}^{(2)}\}$ be the covariance matrices observed in session 1 and 2, respectively. Let us align the two datasets from sessions 1 and 2 by transformation:

$$C_i^{(j)} \Rightarrow (\overline{R}^{(j)})^{-1/2} C_i^{(j)} (\overline{R}^{(j)})^{-1/2} \quad i = 1, ..., N_j \;\; j = 1, 2 \tag{9}$$

As a consequence, due to the congruence invariance property of the Riemannian distance, while the distances between points of the same session remain unchanged, the reference state of

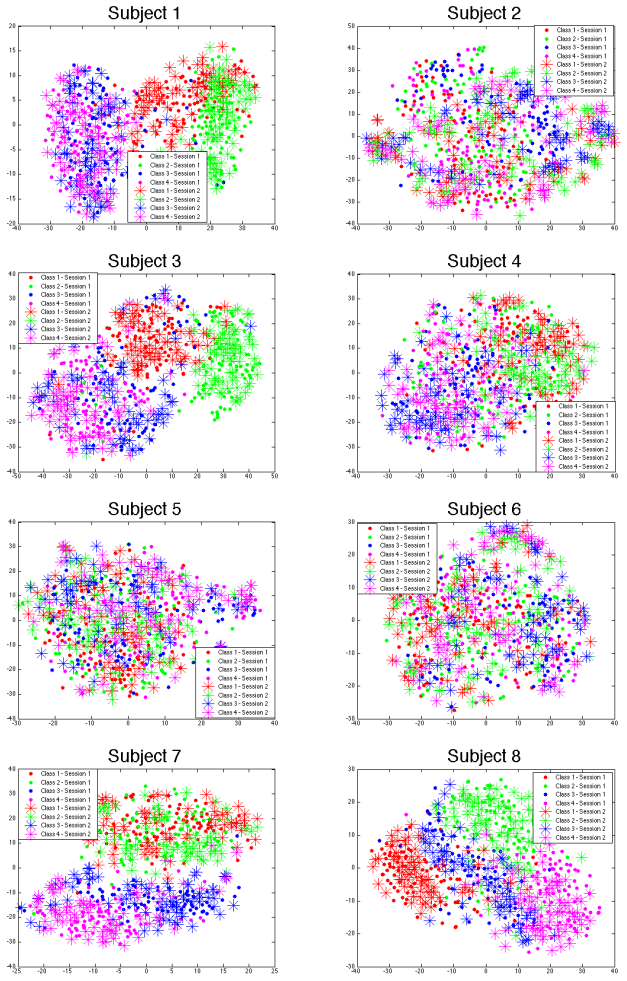

Fig. 4. Motor Imagery dataset: for every subject the affine-transformed covariance matrices for session 1 (circles) and 2 (crosses) are depicted. Data from different sessions in this case are grouped togheter. Data are also colored by classes (right hand, left hand, foot, and tongue imagined movements). Visualization obtained through t-SNE method using the Riemannian distance (2).

both sessions is centered at the identity matrix. Hence, if the assumption holds, the data points relative to a specific task move from the identity in the same way along the manifold for all session. This procedure can be followed also to make data relative to different subjects comparable, both for the MI and the P300 analysis. We simply need to define a reference matrix and a suitable procedure to implement transformation (9) online.

In the MI dataset, reference EEG signals are directly available. Indeed, between the different trials, there are time windows of 1.5 seconds where the subject is not performing any task. If we call these matrices $\{R_1^{(1)}, ..., R_{N_1}^{(1)}\}$ and $\{R_1^{(2)}, ..., R_{N_2}^{(2)}\}$ for the two sessions, we can use these samples to obtain the estimates of the center of mass of the reference state $\overline{R}^{(1)}$ and $\overline{R}^{(2)}$. Then, we can perform the affine transformation described above. We depict in Figure 4 and 5 the projection in the 2-dimensional space of covariance matrices after the affine transformation. It is quite apparent that the shift between two different sessions has been removed. The same procedure can be applied in the cross-subjects classification. However we have to take into account that in a real situation we have to implement this affine transformation online. If we consider, for instance, the cross-session classification using session 1 as training set and session 2 as test set, the reference state matrices of session 2 are not available at the beginning, but we observe them one at a time. Hence, we propose an online estimate of $\overline{R}^{(2)}$ (or $\overline{R}^{(1)}$ if we consider session 1 as test set). Specifically, at observation j, we evaluate a weighted center of mass modifying (4) such as

$$\overline{R}_j^{(2)} = \arg\min_R \sum_{t=1}^{j} \frac{t}{j} d^2(R_t^{(2)}, R) \tag{10}$$

Thus, we can use this online affine transformation strategy to perform cross-session and cross-subject classification.

For the P300 analysis we need to apply further care. Indeed a separated resting-state signal is not available during the experiment, since flashes occur one after the other and the associated ERP overlap. However, the non-target trials can be considered as resting state, or reference events. Then, we build the reference matrix $\overline{R}$ using the elements $\widetilde{C}_{\widetilde{X}_l}$ belonging to the non-target group. We use the true labels, because this experiment can be done in a supervised setting, since classification is used in the game to destroy the aliens, but the true labels can be controlled online. More generally, random epochs bootstrapped from the incoming flow of EEG can be used to define the resting state. Then, for the online affine transformation strategy, we use equation (10).

V. Results

In this section we firstly present, in subsection V-A, the results obtained for cross-session and cross-subject classification of MI data. Then, in subsection V-B, we present cross-subject classification results related to the P300 problem. In both cases the classification methods considered are the Minimum Distance to Mean (MDM) and the Bayesian classifiers with Gaussian distribution (GM) and with mixtures of Gaussian distributions with M components (GM-M). The best results obtained using mixtures of Gaussians is also reported (GM-b).

A. Motor Imagery

The BCI competition dataset for Motor Imagery has already been analyzed using MDM in [2], and with a first attempt to introduce Riemannian mixtures of Gaussians in [37]. In [2], [12], [15] a comparison of Riemannian techniques to other standard methods, like Common Spatial Pattern (CSP) and Linear Discriminant Analysis (LDA), is presented. Here the focus is on the cross-session and cross-subject extensions, analyzing the strength of our proposal based on an affine transformation of the covariance matrix. This makes our procedure suitable to deal with Riemannian methods, but not directly relevant for other standard methods like LDA or CSP. In table I the accuracies, that is the proportion of correctly classified observations, for cross-session classification are shown. For every method and for every subject we report the mean accuracy using session 1 as training set and session 2 as test set, and vice versa, and we compare these means before

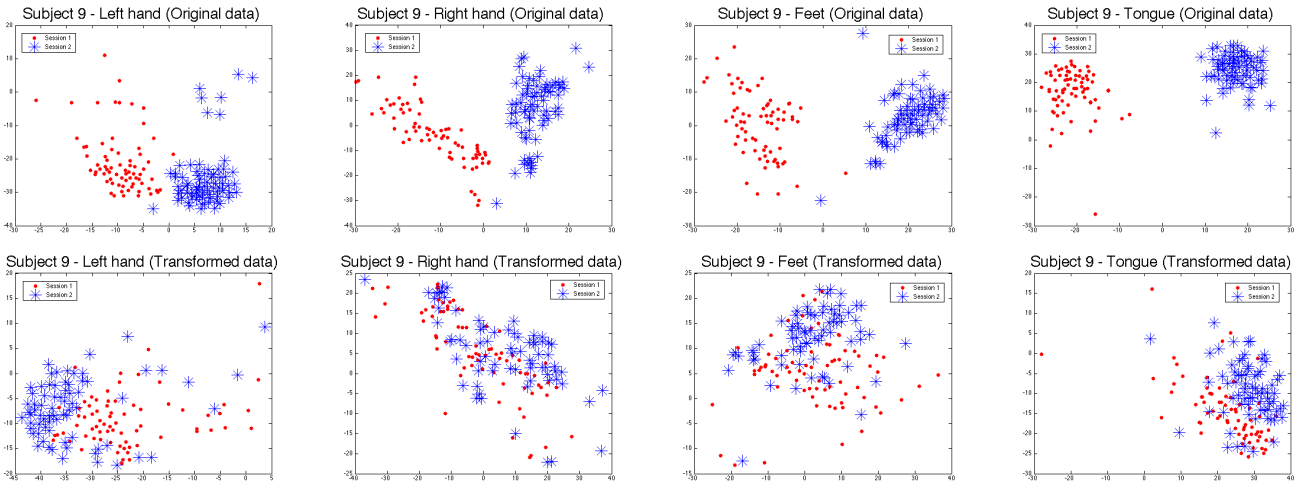

Fig. 5. Motor Imagery dataset: comparison class by class for subject 9. For every class original data (on the top) and affine-transformed data (on the bottom) are shown. Visualization obtained through t-SNE method using the Riemannian distance (2).

Subject	MDM		GM		GM-2		GM-3		GM-4		GM-b	
1	79.2	**79.4**	76.6	**79.2**	73.1	**77.3**	71.5	**79.6**	72.6	**73.6**	73.1	**79.6**
3	72.8	**76.7**	72.6	**76.9**	71.6	**80.1**	71.4	**81.5**	73.8	**78.2**	73.8	**81.5**
7	64.4	**76.1**	61.2	**75.9**	69.8	**75.2**	64.6	**72.6**	66.5	**73.6**	69.8	**75.2**
8	73.1	**79.3**	72.6	**79.2**	73.1	**81.6**	72.4	**81.1**	70.0	**82.1**	73.1	**82.1**
9	74.3	**74.7**	74.5	**75.2**	76.4	**81.8**	77.3	**78.6**	77.3	**78.3**	77.3	**81.8**
Mean	72.8	**77.2**	71.5	**77.3**	72.8	**79.2**	71.4	**78.7**	72.0	**77.2**	73.4	**80.0**
2	51.9	**53.7**	51.4	**52.1**	40.1	**49.7**	35.1	**48.9**	36.3	**50.6**	40.1	**50.6**
4	55.8	**53.8**	52.3	**53.1**	46.4	**51.6**	44.6	**49.7**	45.5	**48.8**	46.4	**51.6**
5	42.2	**46.0**	36.3	**45.2**	32.7	**43.2**	31.1	**42.2**	30.1	**41.0**	32.7	**43.2**
6	44.1	**45.3**	44.6	**45.3**	42.4	**43.4**	38.8	**40.5**	42.9	**44.6**	42.9	**44.6**
Mean	48.5	**49.7**	46.2	**48.9**	40.4	**46.9**	37.4	**45.3**	38.7	**46.3**	40.5	**47.5**

TABLE I

MOTOR IMAGERY DATASET: MEAN CLASSIFICATION ACCURACY USING SESSION 1 AS TRAINING SET AND SESSION 2 AS TEST SET, AND VICEVERSA. SUBJECTS ARE DIVIDED ACCORDING TO THEIR PERFORMANCES IN GOOD SUBJECTS (5 OF THEM, SHOWN IN THE TOP PART OF THE TABLE) AND BAD SUBJECTS (4 OF THEM, SHOWN IN THE BOTTOM PART OF THE TABLE). THE RESULTS ARE RELATIVE TO ORIGINAL COVARIANCE MATRICES (IN BLACK) | AFFINE-TRANSFORMED COVARIANCE MATRICES (IN BOLD GREEN).

Test Subj.	MDM	GM-1	GM-2	GM-3	GM-4
1	46.8 (14.6) | **60.4 (7.9)**	48.3 (15.7) | **61.0 (8.5)**	46.1 (14.0) | **61.0 (8.4)**	41.9 (13.4) | **59.2 (8.7)**	40.7 (14.6) | **61.9 (8.0)**
3	47.2 (15.6) | **69.4 (3.5)**	45.8 (14.6) | **69.0 (2.5)**	47.0 (15.5) | **71.5 (3.0)**	44.4 (12.4) | **71.7 (2.9)**	52.6 (5.1) | **70.2 (6.5)**
7	35.2 (7.9) | **57.0 (8.9)**	35.2 (9.2) | **56.1 (8.3)**	34.9 (9.1) | **55.9 (8.2)**	37.6 (12.9) | **55.8 (8.7)**	37.6 (13.9) | **56.1 (9.0)**
8	35.0 (9.5) | **63.2 (6.7)**	34.5 (9.5) | **63.2 (6.9)**	36.7 (6.0) | **65.3 (7.9)**	33.9 (7.8) | **63.5 (8.5)**	40.9 (13.6) | **66.0 (8.4)**
9	30.0 (5.8) | **68.8 (6.1)**	28.7 (3.1) | **68.9 (5.9)**	30.1 (7.8) | **67.8 (7.0)**	28.6 (6.9) | **67.9 (6.4)**	36.1 (8.3) | **66.7 (8.1)**

TABLE II

MOTOR IMAGERY DATASET: CLASSIFICATION ACCURACY FOR THE CROSS-SUBJECT GENERALIZATION. FOR EVERY TEST SUBJECT, THE OTHER GOOD SUBJECTS ARE USED ONE AT EACH TIME AS TRAINING SUBJECT. IN THE TABLE WE REPORT MEAN, WITH STANDARD DEVIATION IN BRACES. THE RESULTS ARE RELATIVE TO ORIGINAL COVARIANCE MATRICES (IN BLACK) | AFFINE-TRANSFORMED COVARIANCE MATRICES (IN BOLD GREEN).

and after the online affine transformation described in section IV. We separate in the table subjects with a higher accuracy (good subjects), shown in the top of Table I, with subject with lower accuracy, because the results appear to depend upon the level of the performances. In Figure 6 (top panel), a scatter plot for the MDM classifier accuracies is shown. The affine transformation provides significant improvements for all subjects: a t-test on the difference between before and after the affine transformation, considering the values relative to MDM method, provides a p-value of 0.03, despite the low power of the test having only 9 observations. The comparison between the methods confirms what was observed in [37], with Bayesian classifiers displaying better performances for good subjects.

We now consider cross-subject classification results. Since cross-subject generalization is even more tangled than cross-session, we focus only on the good subjects, in order to avoid that bad accuracies due to the subjects could affect the interpretation of the results. In Table II we can compare accuracies before and after the affine transformation of a cross-subject classification where each subject is alternatively used as test set, with the others, one at the time, used as training set. Mean and standard deviation accuracies are reported in the Table. Here the benefits due to affine transformation are even stronger. Specifically, in Table III, we can look in details at the confusion matrices relative to the third row of Table II,

	$k=1$	$k=2$	$k=3$	$k=4$
$\widehat{k}=1$	11	0	1	0
$\widehat{k}=2$	4	11	8	1
$\widehat{k}=3$	26	4	26	0
$\widehat{k}=4$	59	85	65	99
Tot. (%)	100	100	100	100

	$k=1$	$k=2$	$k=3$	$k=4$
$\widehat{k}=1$	70	9	18	1
$\widehat{k}=2$	0	56	1	0
$\widehat{k}=3$	19	4	45	3
$\widehat{k}=4$	11	31	36	96
Tot. (%)	100	100	100	100

TABLE III

MOTOR IMAGERY DATASET: CONFUSION MATRIX (WITH PERCENTAGES REPORTED) FOR MDM CLASSIFICATION METHOD WITH ORIGINAL (ON THE LEFT) AND AFFINE-TRANSFORMED (ON THE RIGHT) COVARIANCE MATRICES. SUBJECTS 3 IS USED AS TRAINING SET, WHILE SUBJECT 8 AS TEST SET.

Subject	**MDM**	**GM-1**	**GM-2**	**GM-3**	**GM-4**	**GM-b**
1	94.0	94.6	94.9	95.5	**96.6**	96.6
2	85.7	86.2	87.5	87.1	**88.0**	88.0
3	84.6	84.0	83.8	85.2	**88.1**	88.1
5	83.7	83.9	84.6	85.5	**86.9**	86.9
6	79.1	78.7	84.1	**86.5**	86.4	86.5
7	79.5	78.8	77.4	**80.6**	75.6	80.6
10	84.4	84.4	83.8	88.1	**90.5**	90.5
11	80.7	80.5	80.7	83.0	**85.8**	85.8
12	93.2	93.0	92.8	**95.2**	94.7	95.2
13	84.8	84.6	85.2	**86.3**	84.2	86.3
14	92.1	92.2	**93.6**	91.3	93.4	93.6
17	77.2	77.5	76.5	**79.7**	79.5	79.7
Mean	84.9	84.9	85.4	87.0	87.5	88.2
4	72.6	**72.7**	71.7	68.9	67.9	71.7
8	73.2	73.0	73.2	**74.8**	72.1	74.8
9	**42.8**	41.6	36.1	37.3	38.3	38.3
15	**66.7**	66.2	65.5	61.6	62.9	65.5
16	**49.3**	49.5	44.5	43.3	45.3	45.3
Mean	60.9	60.6	58.2	57.2	57.3	59.1

TABLE IV

P300 DATASET: MEAN CLASSIFICATION PRECISION FOR THE SUBJECTS USING 30% OF DATA AS TRAINING SET. TRAINING SET IS CHOSEN AT RANDOM, AND THE PROCEDURE IS REPEATED 25 TIMES. SUBJECTS ARE DIVIDED ACCORDING TO THEIR PERFORMANCES IN GOOD SUBJECTS (12 OF THEM, SHOWN IN THE TOP PART OF THE TABLE) AND BAD SUBJECTS (5 OF THEM, SHOWN IN THE BOTTOM PART OF THE TABLE).

when subject 3 is used as training set and subject 8 is used as test set. Table III refers to the MDM algorithm, and on the left we can observe that, without any transformation, a lot of trails are assigned to the class number 4, with an accuracy equal to 36.8%. After the affine transformation, instead, the classification is significantly better, with an accuracy equal to 66.5%. In particular, elements of the classes 1 and 4 are very well predicted, as shown on the right of Table III. In Figure 6 (bottom panel), a scatter plot for the MDM classifier accuracies is shown.

B. ERP

In this section we consider the results of the ERP P300 experiment. First of all we point out that since the two classes are strongly unbalanced (1/6th of target elements), accuracy is not a suitable index of classification performance (a classifier which assigns every element to the non-target class will have an accuracy of 0.83). For this reason we consider the precision index pr, defined as

$$pr = \frac{TP}{TP + FP} \tag{11}$$

where TP (True Positive) is the number of elements correctly classified as target, while FP (False Positive) is the number of elements wrongly classified as target.

First of all, we analyze each subject separately, in order to understand how the different classification methods work in this framework and how performances vary between the subjects. To do that, for each subject, we evaluate the precision using the 30% of the data (randomly chosen) as training set, and the other part as test set. We repeat this procedure 25 times, evaluating the mean value. Results are reported in Table IV, where once again we separated good subjects (those with a precision higher than 0.75) from subjects with lower performances.

Second, to make a comparison between algorithms we can observe that the introduction of Riemannian mixtures provides significant improvements in the group of the good subjects, while this is not true for the subjects with lower performances. If we consider the 12 good subjects, a paired t-test between MDM and GM-4 provides a p-value around 10^{-4}.

Next, we analyze the cross-subject classification problem. Also in this case we focus on the good subjects. In Table V we compare precisions before and after the affine transformation. Every good subject is alternatively used as test set, with the others, one at the time, used as training set. Mean and standard deviation accuracies are reported in the Table. The results obtained after the affine transformation are very good, similar to those obtained for the classical training/test cross-validation procedure, even if in this case there is not a large difference in the performance between the different methods.

Test Subj.	MDM		GM-1		GM-2		GM-3		GM-4	
1	64.4 (16.4)	**91.8 (3.5)**	64.4 (21.0)	**92.1 (3.2)**	59.1 (19.4)	**88.4 (4.7)**	59.9 (23.5)	**88.2 (6.3)**	59.1 (24.1)	**90.4 (4.2)**
2	39.4 (32.4)	**86.3 (3.8)**	41.8 (20.9)	**86.4 (3.9)**	46.7 (26.3)	**84.0 (6.0)**	29.9 (16.7)	**82.5 (4.6)**	39.3 (28.5)	**83.3 (6.9)**
3	30.1 (12.4)	**87.6 (6.1)**	29.8 (15.4)	**88.1 (5.9)**	25.5 (11.1)	**85.7 (7.2)**	31.3 (12.8)	**84.7 (6.4)**	22.9 (12.6)	**87.5 (4.3)**
5	69.1 (31.1)	**80.8 (5.1)**	71.1 (24.7)	**81.1 (5.5)**	68.5 (28.8)	**79.7 (6.1)**	67.5 (31.6)	**78.6 (7.6)**	45.2 (39.3)	**79.9 (8.6)**
6	49.2 (21.9)	**84.6 (6.2)**	45.8 (19.8)	**83.1 (8.0)**	50.6 (26.0)	**79.8 (9.5)**	39.0 (23.7)	**83.7 (11.8)**	41.3 (18.6)	**84.5 (8.4)**
7	24.1 (22.7)	**85.8 (5.4)**	27.7 (29.4)	**85.5 (6.6)**	33.6 (26.6)	**81.9 (8.3)**	27.2 (19.9)	**85.5 (6.2)**	17.8 (17.7)	**82.6 (7.7)**
10	55.1 (24.7)	**84.8 (3.8)**	52.7 (15.0)	**84.9 (4.4)**	51.0 (21.3)	**84.3 (4.5)**	47.7 (12.4)	**80.8 (7.5)**	53.4 (20.5)	**84.0 (6.5)**
11	63.2 (14.5)	**86.7 (4.3)**	61.2 (19.1)	**87.9 (4.0)**	54.6 (22.9)	**85.8 (6.4)**	59.7 (20.5)	**84.8 (5.9)**	60.0 (19.7)	**87.1 (5.8)**
12	59.3 (24.6)	**91.1 (3.6)**	59.9 (19.5)	**92.2 (3.3)**	61.9 (22.9)	**89.8 (4.8)**	67.1 (26.8)	**89.8 (4.9)**	60.5 (22.2)	**91.0 (3.8)**
13	46.2 (22.5)	**91.7 (4.3)**	48.2 (25.9)	**93.3 (4.3)**	36.7 (32.8)	**90.9 (4.3)**	35.4 (28.4)	**89.5 (5.1)**	46.1 (32.0)	**90.7 (5.1)**
14	77.2 (18.8)	**89.6 (3.7)**	76.2 (23.6)	**89.9 (3.9)**	62.6 (26.5)	**87.0 (6.1)**	71.4 (27.4)	**86.2 (5.2)**	69.5 (31.0)	**86.4 (6.1)**
17	39.9 (18.3)	**89.0 (2.8)**	39.6 (11.3)	**89.1 (2.9)**	43.8 (24.0)	**84.9 (8.0)**	31.7 (19.1)	**86.2 (4.3)**	36.8 (19.2)	**86.8 (3.6)**

TABLE V

P300 DATASET: CLASSIFICATION PRECISION FOR THE CROSS-SUBJECT GENERALIZATION. FOR EVERY TEST SUBJECT, THE OTHER SUBJECTS ARE USED ONE AT EACH TIME AS TRAINING SUBJECT. IN THE TABLE WE REPORT MEAN, WITH STANDARD DEVIATION IN BRACES. THE RESULTS ARE RELATIVE TO ORIGINAL COVARIANCE MATRICES (IN BLACK) | AFFINE-TRANSFORMED COVARIANCE MATRICES (IN BOLD GREEN).

Before affine transformation

	$k=0$	$k=1$
$\widehat{k}=0$	100	97
$\widehat{k}=1$	0	3
Tot. (%)	100	100

After affine transformation

	$k=0$	$k=1$
$\widehat{k}=0$	97	7
$\widehat{k}=1$	3	93
Tot. (%)	100	100

Before affine transformation

	$k=0$	$k=1$
$\widehat{k}=0$	100	100
$\widehat{k}=1$	0	0
Tot. (%)	100	100

After affine transformation

	$k=0$	$k=1$
$\widehat{k}=0$	98	37
$\widehat{k}=1$	2	63
Tot. (%)	100	100

TABLE VI

P300 DATASET: CONFUSION MATRICES (WITH PERCENTAGES REPORTED) FOR MDM CLASSIFICATION METHOD. ON THE LEFT, CONFUSION MATRICES TO COMPARE THE CROSS-SUBJECT CLASSIFICATION WITH SUBJECT 14 USED AS TEST SET AND SUBJECT 3 AS TRAINING SET, BEFORE AND AFTER AFFINE TRANSFORMATION. ON THE RIGHT CONFUSION MATRICES ARE RELATED TO THE CROSS-SUBJECT CLASSIFICATION WITH SUBJECT 7 USED AS TEST SET AND SUBJECT 5 AS TRAINING SET.

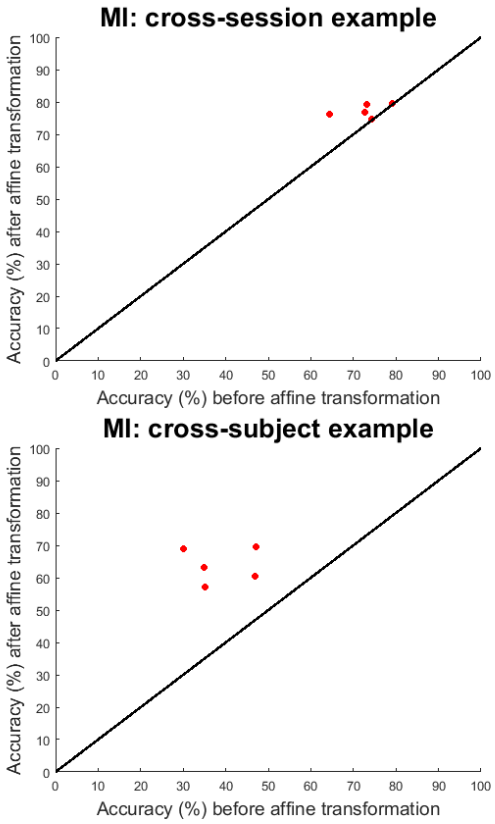

Fig. 6. MI example: scatter plot to compare classification results for cross-session (on the top) and cross-subject (on the bottom) for MDM classifier before (x-axis) and after (y-axis) affine transformation.

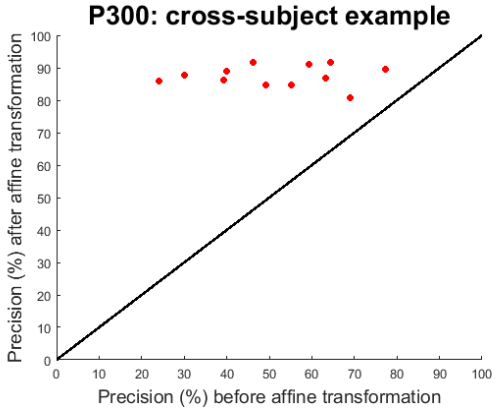

Fig. 7. P300 example: scatter plot to compare classification results for cross-subject for MDM classifier before (x-axis) and after (y-axis) affine transformation.

The results obtained without transforming data, as expected, are very poor. In Figure 7, a scatter plot for the MDM classifier averaged precisions shows the gain achieved by using the affine transformation. This result is even strenghtened by the fact that, for the evaluation of the means before the affine transformation, we did not consider the cases where all the observations are assigned to the non target class. Indeed this results in a NaN, since formula (11) corresponds to a $\frac{0}{0}$ if no observations are classified as target. Furthermore, the high standard deviation is caused by some situations with a precision of 100%, but that do not represent a good classification, since they correspond to situation where almost every observation is classified as non-target, and only few observations are (correctly) classified as target. This provides

a 100% precision, but the classification cannot be considered satisfactory. To clarify these two situations, in Table VI we compare confusion matrices related to MDM algorithm before and after the affine transformation. On the left, subject 14 is used as test set and subject 3 as training set and we can observe that, before the affine transformation, only 3 observations are classified as target. These observations are true target objects, resulting in a 100% precision evaluation, but the classification obtained after data are affine-transformed is clearly better. On the right, subject 7 is used as test set and subject 5 as training set. In this case, before applying the affine transformation, no observations are assigned to the target class.

VI. Conclusion

In this paper we present an approach based on Riemannian geometry to deal with cross-session and cross-subject classification in BCI applications. These problems are part of a wider issue known as transfer learning, defined as the ability to use knowledge acquired previously in a new task related to the first. Here we propose to affine transform the spatial covariance matrices of the EEG signals of every session/subject to make data comparable. We assumed that, from one session (subject) to another, covariance matrices related to a specific task performed by the subject move with a similar relocation from a reference state, different between sessions or subjects. Hence, the idea is to center covariance matrices with respect to a reference matrix. Under our assumption, Riemannian geometry offers an optimal procedure for tackling the transfer learning problem due to the affine invariance property of the Riemannian distance and Riemannian mean. We considered two kinds of datasets, one related to a MI paradigm, and the second one to an ERP, P300 specifically, paradigm. We defined a suitable reference state proposing a way to estimate online the reference matrix, to make the procedure useful in a real-time application.

Then we have tested the proposed procedure in a classification problem, where data from different sessions (subjects) are used to estimate the class parameters needed to classify new observations.

We analyzed the improvements due to the affine transformation in the cross-session and cross-subject classification, observing that, while in the original data space often results are poor, in general the affine transformation allows much better classification accuracies and precisions. This illustrates the goodness of the affine transformation through the reference matrix, which we proposed here, to obtain satisfactory results in cross-session and cross-subject classification.

Acknowledgment

This work is partly supported by the ERC Grant CHESS, 2012-ERC-AdG-320684.

References

[1] Asfari, B., (2011). Riemannian L^p center of mass: existence, uniqueness and convexity. *Proc. Amer. Math. Soc.*, 139(2), 655–673.

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

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

[4] Barachant, A., Congedo, M., (2014). A Plug&Play P300 BCI Using Information Geometry. *arXiv:1409.0107*.

[5] Bhatia, R., (2007). *Positive definite matrices.* Princeton University Press.

[6] Bini, D.A., Iannazzo, B., (2013). Computing the Karcher mean of symmetric positive definite matrices. *Linear Algebra and its Applications*, 438(4), 1700–1710.

[7] Blankertz, B., Dornhege, G., Krauledat, M., Muller, K.R., Kunzmann, V., Losch, F. et al., (2006). The Berlin brain-computer interface: EEG-based communication without subject training. *IEEE Trans. Neural Syst. Rehabil. Eng.*, 14(2), 147–152.

[8] Blankertz, B., Tomioka,R., Lemm, S., Kawanabe, M., Muller, K., (2008). Optimizing spatial filters for robust EEG Single-Trial analysis. *IEEE Signal Processing Magazine*, 25(1), 41–56.

[9] Blankertz, B., Lemm, S., Treder, M., Haufe, S., Muller, K., (2011). Single-trial analysis and classification of ERP components - A tutorial. *NeuroImage*, 56(2), 814–825.

[10] Bonnabel, S., (2013). Stochastic gradient descent on Riemannian manifolds. *IEEE Transactions on Automatic Control*, 122(4), 2217–2229.

[11] Congedo, M., Goyat, M., Tarrin, N., Ionescu, G., Varnet, L., Rivet, B., Phlypo, R., Jrad, N., Acquadro, M., Jutten, C., (2011). "Brain Invaders": a prototype of an open-source P300- based video game working with the OpenViBE platform. *5th International Brain-Computer Interface Conference 2011 (BCI 2011)*.

[12] Congedo, M., (2013). EEG Source Analysis, *HDR Thesis*, Université de Grenoble. https://hal.archives-ouvertes.fr/tel-00880483/document

[13] Congedo, M., Afsari, B., Barachant, A., Moakher, M., (2015). Approximate Joint Diagonalization and Geometric Mean of Symmetric Positive Definite Matrices. *PLoS ONE*, 10(4).

[14] Congedo, M., Barachant, A., Kharati Koopaei, E., (2017). Fixed Point Algorithms for Estimating Power Means of Positive Definite Matrices. *IEEE Transactions on Signal Processing*, 65(9), 2211–2220.

[15] Congedo, M., Barachant, A., Bhatia, R., (2017). Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review. *Brain Computer Interfaces*.

[16] Farquhar, J., (2009). A linear feature space for simultaneous learning of spatio-spectral filters in BCI. *Neural Networks*, 22(9), 1278–1285.

[17] Ferreira, R., Xavier, J., Costeira, J.P., Barroso, V., (2013). Newton algorithms for Riemannian distance related to problems on connected locally symmetric manifolds. *IEEE J. Sel. Topics Signal Process.*, 7(4), 634–645.

[18] Gouy-Pailler, C., Congedo, M., Brunner, C., Jutten, C., Pfurtscheller, G., (2010). Nonstationary brain source separation for multiclass motor imagery. *IEEE Transactions on Biomedical Engineering*, 57(2), 469–478.

[19] Grosse-Wentrup, M., Buss, M., (2008). Multiclass common spatial patterns and information theoretic feature extractions. *IEEE Transactions on Biomedical Engineering*, 55(8), 1991–2000.

[20] Haufe, S., Treder, M.S., Gugler, M.F., Sagebaum, M., Curio, G., Blankertz, B., (2011). EEG potentials predict upcoming emergency brakings during simulated driving. *Journal of Neural Engineering*, 8.

[21] Jayaram, V., Alamgir, M., Altun, Y., Scholkopf, B., Grosse-Wentrup, M., (2016). Transfer Learning in Brain-Computer Interfaces *Computational Intelligence Magazine, IEEE*, 11(1), 20–31.

[22] Jeuris, B., Vandebril, R., Vandereycken, B., (2012). A survey and comparison of contemporary algortihms for computing the matrix geometric mean. *Electronic Transactions on Numerical Analysis*, 39, 379–402.

[23] Korczowski, L., Congedo, M., Jutten, C., (2015). Single-Trial Classification of Multi-User P300-based Brain-Computer Interface Using Riemannian Geometry. *37th Annual International Conference of IEEE Engineering in Medicine and Biology Society (EMBC)*, 1769 – 1772.

[24] Kothe, C.A., Mekeig, S., (2013). BCILAB: a platform for brain-computer interface development. *Journal of Neural Engineering*, 10(5).

[25] Leeb, R., Brunner, C., Muller-Putz, G.R., Schlogl, A., Pfurtscheller, G., (2008). BCI competition 2008 – Graz data set A. Graz University of Technology, Austria.

[26] Lenglet, C., Rousson, M., Deriche, R., Faugeras, O., (2006). Statistics on the manifold of multivariate normal distributions. *J. Math. Imaging Vis.*, 25(1), 127–154.

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

[28] Moakher, M., (2005). A differential geometric approach to the geometric mean of symmetric posiitve-definite matrices. *SIAM J. Matrix Anal. and Appl.*, 26(3), 735–747.

[29] Mohanchandra, K., (2015). Criminal forensic: An application to EEG. *Recent and Emerging trends in Computer and Computational Sciences (RETCOMP), 2015*, 18–21.

[30] Pan, S.J., Yang, Q., (2010). A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345–1359.

[31] Pennec, X., Fillard, P., Ayache, N., (2006). A Riemannian Framework for Tensor Computing. *International Journal of Computer Vision*, 66(1), 41–66.

[32] Rao, R.P.N., (2013). *Brain-Computer Interfacing - An Introduction.* Cambridge University Press.

[33] Reuderink, B., Farquhar, J., Poel, M., Nijholt, A., (2011). A subject-independent brain-computer interface based on smoothed, second-order baselining. *Proceedings of the 33st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2011).*

[34] Said, S., Bombrun, L., Berthoumieu, Y., Manton, J. H., (2015). Riemannian gaussian distributions on the space of covariance matrices. *arXiv:1507.01760v1*.

[35] van der Maaten, L., Hinton, G., (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9, 2579–2605.

[36] Wu, D., Lance, B.J., Parsons, T.D., (2013). Collaborative Filtering for Brain-Computer Interaction Using Transfer Learning and Active Class Selection. *PLoS ONE*, 8(2)

[37] Zanini, P., Congedo, M., Jutten, C., Said, S., Berthomieu, Y., (2016). Parameters estimate of Riemannian Gaussian distribution in the manifold of covariance matrices. *Proc. of the Ninth IEEE Sensor Array and Multichannel Signal Processing Workshop (IEEE SAM 2016)*, Rio de Janeiro, Brasil, July 2016.