Dimensionality transcending: a method for merging BCI datasets with different dimensionalities

Pedro L. C. Rodrigues, *Member, IEEE,* Marco Congedo, *Member, IEEE,* and Christian Jutten, *Fellow, IEEE*

***Abstract*—Objective: We present a transfer learning method for datasets with different dimensionalities, coming from different experimental setups but representing the same physical phenomena. We focus on the case where the data points are symmetric positive definite (SPD) matrices describing the statistical behavior of EEG-based brain computer interfaces (BCI). Method: Our proposal uses a two-step procedure that transforms the data points so that they become matched in terms of dimensionality and statistical distribution. In the dimensionality matching step, we use isometric transformations to map each dataset into a common space without changing their geometric structures. The statistical matching is done using a domain adaptation technique adapted for the intrinsic geometry of the space where the datasets are defined. Results: We illustrate our proposal on time series obtained from BCI systems with different experimental setups (e.g., different number of electrodes, different placement of electrodes). The results show that the proposed method can be used to transfer discriminative information between BCI recordings that, in principle, would be incompatible. Conclusion and significance: Such findings pave the way to a new generation of BCI systems capable of reusing information and learning from several sources of data despite differences in their electrodes positioning.**

***Index Terms*—Brain-computer interfaces, EEG, Heterogeneous domain adaptation, Symmetric positive definite matrices, Riemannian geometry**

I. Introduction

When setting up an experiment for measuring some physical phenomenon, an experimenter is faced with several practical choices, such as the kind and number of sensors to adopt, where to place them, which sampling frequency to use, etc. In general, it is reasonable to expect that the physical phenomenon under study is independent to such choices and that small changes in the experimental setup do not impact dramatically its ability to describe the system. For example, using 19 or 20 electrodes in an electroencephalography (EEG) experiment does not change significantly the information that we can observe from a subject's brain. Similarly, if the signal at one electrode is compromised during an EEG recording, it should be possible to use the information from the other sensors without having to discard the whole epoch. In this paper, we tackle this very practical yet seldom discussed problem. We focus our presentation on problems related to the classification of EEG signals coming from brain-computer interfaces (BCI), however, our theoretical contributions are more general than this.

P.L.C.Rodrigues, C.Jutten, and M.Congedo are with Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, 38000 Grenoble, France

Manuscript received XXX; revised XXX.

In general, it is imperative to pre-process data points from two datasets before pooling them together for joint analysis. This is necessary because the statistical distribution of datasets gathered under different experimental conditions are often different from each other [1]. For instance, the statistics of the EEG signals obtained from a subject performing a set of BCI tasks (e.g., left-hand/right-hand motor imagery) on a Monday morning are different from those obtained from the same subject on a Friday afternoon. Similarly, the EEG data of two subjects performing the same BCI experiment have different statistical distributions.

Our proposal. In this paper, we propose a new method for merging datasets with different dimensionalities (e.g. different number and/or position of recording sites) and different statistical distributions (e.g. coming from different recording sessions), allowing for the joint analysis of datasets that would otherwise be incompatible. Suppose we have two datasets, $\mathcal{A}$ and $\mathcal{B}$, containing data points with dimensionalities $d_{\mathcal{A}}$ and $d_{\mathcal{B}}$. We interpret such datasets as point clouds defined in high-dimensional metric spaces and use concepts from computational geometry [2] to study their geometrical properties and investigate commonalities between them. The procedure that we present consists of the following two steps:

(1) *Dimensionality matching.* We transform the data points from $\mathcal{A}$ and $\mathcal{B}$ into a common space with dimensionality $d \geq \max\{d_{\mathcal{A}}, d_{\mathcal{B}}\}$. The transformation is isometry preserving, which ensures that the statistical distributions of the original data points in their respective spaces remain the same in the new space.

(2) *Statistical matching.* We transform the elements of the dimensionality-matched datasets so that their statistical distributions become as similar as possible.

After these transformations, we have two datasets that are defined in the same space and that have compatible statistical distributions. Since our method surpasses the intrinsic limitations due to dimensionality mismatch, we name it *dimensionality transcending* (DT).

We use a Riemannian geometric framework to manipulate the data points from BCI datasets [3] [4]. Such approach parametrizes the second-order statistics of multivariate EEG time series via symmetric positive definite (SPD) matrices, e.g., their spatial covariance matrices, and allows for the comparison of time series in terms of their parametrizations. The set of SPD matrices define a Riemannian manifold whose intrinsic geometry is well known [5] and we take its properties into account when carrying out the steps involved in the DT procedure.

Related work. The branch of machine learning concerned with problems related to statistical mismatch between datasets is called *domain adaptation* (DA) and has been discussed in several works (see [6] for a survey). In its traditional form, DA deals with datasets having the same dimensionality, but it may be extended to cases where they differ; this is called *heterogeneous domain adaptation* (h-DA). Most proposals in the h-DA literature are based on procedures that learn the best projection of the datasets into a common latent space in which the differences between the two statistical distributions are minimized. An example is transfer component analysis (TCL) [7], a method that learns a projection into a reproducing kernel Hilbert space and then searches for a transformation that matches the statistics of the projected data points by minimizing their maximum mean discrepancy [8]. Although TCL appears to work rather well in practice, specially with databases containing texts and images, it is not crafted for taking into account the intrinsic geometry of the space where the data points are defined, discarding, therefore, valuable information for the matching of the datasets. Furthermore, TCL relies on an optimization procedure that solves a semi-definite programming problem that can be computationally costly when considering high-dimensional data points. In the EEG-BCI literature, there has been much research on the homogenous case for DA (see [9] for a review) but, to the best of our knowledge, no work has tackled the heterogenous setting. The method that we propose here builds mainly on the Riemannian Procrustes Analysis [10], which is a geometry-aware procedure for matching the statistics of datasets defined on the same SPD manifold.

Novelties. The three main novelties of our approach are:

- The fact of taking datasets into a higher-dimensional space ensures that we don't discard any valuable information and that we may define an isometric transformation. This is different from what is usually done in machine learning, where projections onto lower-dimensional space are preferred [11] [12].
- Our approach is guided by geometric considerations and intuitions. This makes DT easy to understand and implement. Furthermore, its two-step modularity allows for a better sense of the transformations carried out on the data points, allowing for improvements and adaptations according to the characteristics of the data space being considered.
- The transformations that we propose are not data-driven but defined in terms of the geometry of the space where the data points are defined. This leads to a simple, fast, and robust method.

Structure of the paper. The paper is organized as follows. In Section II, we present the dimensionality transcending method by first formalizing it mathematically and demonstrating some important properties associated to it. We also present the methodology used to validate the DT procedure. In Section III, we present our results on different publicly available datasets and in Section IV we discuss them. Section V concludes the paper.

TABLE I
TABLE OF SYMBOLS USED IN THE PAPER

$\mathcal{P}(d)$	manifold of d-dimensional SPD matrices
δ_R	geodesic distance defined in the space $\mathcal{P}(d)$
$\mathcal{A}$	dataset consisting of SPD matrices defined in $\mathcal{P}(d_{\mathcal{A}})$ and labels from $\{1,\dots,L\}$; see Eq. (8)
$\Theta_{\mathcal{A}}$	statistical distribution of the SPD matrices in $\mathcal{A}$
$\boldsymbol{M}^{\mathcal{A}}$	geometric mean of the SPD matrices in $\mathcal{A}$
$\sigma^{\mathcal{A}}$	dispersion of the points in $\mathcal{A}$ around $\boldsymbol{M}^{\mathcal{A}}$
$\boldsymbol{C}^{\uparrow}$	augmented version of matrix $\boldsymbol{C}$ using the isometric transformation defined in Eq. (14)
$\mathcal{A}^{\uparrow}$	dataset identical to $\mathcal{A}$ but with the SPD matrices augmented via the isometric transformation defined in Eq. (14)
$\mathcal{A}^{(\text{RPA})}$	dataset identical to $\mathcal{A}$ but with the SPD matrices transformed via RPA to match the statistical distribution of another dataset
$\mathcal{D}_{\text{train}}$	training set used to fit the parameters of a statistical classifier
$\mathcal{D}_{\text{test}}$	set of data points used to asssess the performance of a statistical classifier

II. METHODS

In this section, we present some properties of the SPD manifold and give a mathematical formulation of the DT procedure. Then, we present the datasets and pipelines used to validate our proposal. Table I gives a brief description of the mathematical symbols defined here and used in the rest of the paper.

A. Geometry of the SPD manifold

The set of SPD matrices is defined as

$$\mathcal{P}(d) = \left\{ \boldsymbol{C} \in \mathcal{S}(d) \;\middle|\; \boldsymbol{x}^T \boldsymbol{C} \boldsymbol{x} > 0,\ \forall \boldsymbol{x} \in \mathbb{R}^d,\ \boldsymbol{x} \neq \boldsymbol{0} \right\} , \quad (1)$$

where $\mathcal{S}(d)$ is the set of d-dimensional real symmetric matrices. Matrices in $\mathcal{P}(d)$ lie in a manifold [5], a set of points with the property that the neighborhood of each $\boldsymbol{C} \in \mathcal{P}(d)$ can be bijectively mapped onto an Euclidean space, also known as its tangent space $T_{\boldsymbol{C}}\mathcal{P}(d)$. Intuitively, we say that the neighbourhood of every point in the manifold is flat, but the whole manifold has a non-positive curvature [13]. If we endow every tangent space of a manifold with a metric that changes smoothly along its elements, we say that we have a Riemannian manifold. In this case, fundamental geometric notions are naturally defined, such as geodesic (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. In this work, we use the Riemannian metric defined for tangent vectors $\boldsymbol{\eta}, \boldsymbol{\xi} \in T_{\boldsymbol{C}}\mathcal{P}(d)$ as

$$\langle \boldsymbol{\eta}, \boldsymbol{\xi} \rangle_{\boldsymbol{C}} = \text{tr}\big(\boldsymbol{C}^{-1} \boldsymbol{\eta} \boldsymbol{C}^{-1} \boldsymbol{\xi} \big) , \quad (2)$$

where $\boldsymbol{C} \in \mathcal{P}(d)$ is the reference point for the inner product and $\text{tr}(\cdot)$ denotes the trace operator. It is possible to show that, for any invertible matrix $\boldsymbol{W} \in \mathbb{R}^{d \times d}$, it holds

$$\langle \boldsymbol{W} \boldsymbol{\eta} \boldsymbol{W}^T, \boldsymbol{W} \boldsymbol{\xi} \boldsymbol{W}^T \rangle_{\boldsymbol{W}\boldsymbol{C}\boldsymbol{W}^T} = \langle \boldsymbol{\eta}, \boldsymbol{\xi} \rangle_{\boldsymbol{C}} , \quad (3)$$

meaning that the inner product between two tangent vectors is congruence-invariant. Because of such property, metric (2) is named the affine-invariant Riemannian metric (AIRM) and is known as the "natural" Riemannian metric for the SPD

manifold [5], [13], [14]. The AIRM metric induces a geodesic distance between two SPD matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ in $\mathcal{P}(d)$ whose explicit expression is

$$\begin{aligned} \delta_R^2(\boldsymbol{A},\boldsymbol{B}) &= \left\| \log\left(\boldsymbol{A}^{-1/2}\boldsymbol{B}\boldsymbol{A}^{-1/2}\right)\right\|_F^2 \\ &= \sum_{k=1}^{d} \log^2(\lambda_k) \ , \end{aligned} \tag{4}$$

where $\{\lambda_1,\ldots,\lambda_d\}$ is the set of eigenvalues of $\boldsymbol{A}^{-1/2}\boldsymbol{B}\boldsymbol{A}^{-1/2}$ (which are the same as for $\boldsymbol{A}^{-1}\boldsymbol{B}$). Because of the affine-invariance of the metric from which (4) is induced, it is easy to show that for any invertible $\boldsymbol{W}\in\mathbb{R}^{d\times d}$ it holds

$$\delta_R^2(\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^T,\boldsymbol{W}\boldsymbol{B}\boldsymbol{W}^T) = \delta_R^2(\boldsymbol{A},\boldsymbol{B}) \ . \tag{5}$$

Once we have the expression for the distance between any two points in $\mathcal{P}(d)$, it is natural to define a notion of mean, or center of mass, of a set of SPD matrices,

$$\mathcal{A} = \left\{\boldsymbol{C}_1,\ldots,\boldsymbol{C}_K\right\} \subset \mathcal{P}(d) \ .$$

We define such mean as the matrix that minimizes the dispersion in $\mathcal{A}$, as per

$$\boldsymbol{M}^{\mathcal{A}} = \underset{\boldsymbol{M}\in\mathcal{P}(d)}{\operatorname{argmin}} \ \mathcal{L}_{\mathcal{A}}(\boldsymbol{M}) \ , \tag{6}$$

with

$$\mathcal{L}_{\mathcal{A}}(\boldsymbol{M}) = \sum_{i=1}^{K} \delta_R^2(\boldsymbol{M},\boldsymbol{C}_i) \tag{7}$$

and $\sigma^{\mathcal{A}} = \mathcal{L}_{\mathcal{A}}(\boldsymbol{M}^{\mathcal{A}})$. Note, also, that when the elements of $\mathcal{A}$ are strictly positive scalars, $\boldsymbol{M}^{\mathcal{A}}$ is simply their geometric mean. This explains why many researchers [5], [15]–[17] adopt the term "geometric mean" to refer to the center of mass of a set of SPD matrices. When $K \geq 3$, there is no closed form solution for Eq. (6) in general, however, due to the non-positive curvature of the SPD manifold, it is possible to show that there always exists a solution for its optimization problem [18]. Many researchers have proposed procedures for calculating the center of mass of a set of SPD matrices iteratively, as in [19] and [20].

B. Problem statement

Consider two datasets,

$$\begin{aligned} \mathcal{A} &= \left\{(\boldsymbol{C}_i^{\mathcal{A}},\ \ell_i^{\mathcal{A}}) \text{ for } i=1,\ldots,K_{\mathcal{A}}\right\} \ , \\ \mathcal{B} &= \left\{(\boldsymbol{C}_i^{\mathcal{B}},\ \ell_i^{\mathcal{B}}) \text{ for } i=1,\ldots,K_{\mathcal{B}}\right\} \ , \end{aligned} \tag{8}$$

with data points $\boldsymbol{C}_i^{\mathcal{A}} \in \mathcal{P}(d_{\mathcal{A}})$ and $\boldsymbol{C}_i^{\mathcal{B}} \in \mathcal{P}(d_{\mathcal{B}})$, and class labels $\ell_i^{\mathcal{A}}, \ell_i^{\mathcal{B}} \in \{1,\ldots,L\}$, where L is the number of classes. We assume that these SPD matrices describe the second order statistics of feature vectors, so each of their dimensions has a physical meaning.

We denote $\boldsymbol{M}^{\mathcal{A}}$ and $\boldsymbol{M}^{\mathcal{B}}$ the geometric means of the matrices of each dataset, and $\sigma^{\mathcal{A}}$ and $\sigma^{\mathcal{B}}$ the dispersions around the geometric mean. The class means for each dataset are denoted $\boldsymbol{M}_\ell^{\mathcal{A}}$ and $\boldsymbol{M}_\ell^{\mathcal{B}}$, with $\ell \in \{1,\ldots,L\}$, and are geometric means defined as in Eq. (6) but calculated only on data points from each class. Note that, in the SPD manifold, the geometric mean of the class means is not equal to the geometric mean of all the data points [5]. We parametrize the statistical distributions of the data points in $\mathcal{A}$ and $\mathcal{B}$ as

$$\begin{aligned} \mathcal{A} \sim \Theta_{\mathcal{A}} &= \left\{\boldsymbol{M}^{\mathcal{A}}, \boldsymbol{M}_1^{\mathcal{A}},\ldots,\boldsymbol{M}_L^{\mathcal{A}},\sigma^{\mathcal{A}}\right\} \ , \\ \mathcal{B} \sim \Theta_{\mathcal{B}} &= \left\{\boldsymbol{M}^{\mathcal{B}}, \boldsymbol{M}_1^{\mathcal{B}},\ldots,\boldsymbol{M}_L^{\mathcal{B}},\sigma^{\mathcal{B}}\right\} \ . \end{aligned} \tag{9}$$

This parametrization is analogous to the description of a mixture of Gaussian distributions in Euclidean space, where each mixture corresponds to a class and the class dispersions are supposed equal. Our goal is to define a procedure for transforming the elements of both datasets so that they are defined in the same space and for which the parametrization of the statistical distributions of the transformed data points are as similar as possible. Note that if $d_{\mathcal{A}} = d_{\mathcal{B}}$, the problem reduces to domain adaptation in the SPD manifold as investigated in [21].

C. Dimensionality transcending on $\mathcal{P}(d)$

The two steps of dimensionality transcending applied to SPD data are defined as follows:

- *Dimensionality matching.* Let $\mathcal{E}_{\mathcal{A}}$ and $\mathcal{E}_{\mathcal{B}}$ be two ordered sets describing the physical meaning of each dimension on the data points. For instance, if they are related to EEG recordings, the elements of $\mathcal{E}_{\mathcal{A}}$ and $\mathcal{E}_{\mathcal{B}}$ are the locations of the electrodes used in $\mathcal{A}$ and $\mathcal{B}$, respectively. To match the datasets, first we define a new set $\mathcal{E} = \mathcal{E}_{\mathcal{A}} \cup \mathcal{E}_{\mathcal{B}}$ and augment the data points in $\mathcal{A}$ and $\mathcal{B}$ so that they become d-dimensional SPD matrices, $d = |\mathcal{E}|$, the number of elements in $\mathcal{E}$. Note that it may happen that the ordering of the dimensions of the expanded data points from $\mathcal{A}$ and $\mathcal{B}$ are not the same. In this case, we may apply permutation matrices to the augmented data points to rearrange their rows and columns so that they correspond to the same physical quantities. We give more details about the augmentation step in Section II-D.

- *Statistical matching.* Once the data points are defined in the same space, we match their statistical distributions. For this, we use the Riemannian Procrustes analysis (RPA) that we have proposed in [21]. RPA is an extension of the classical Procrustes analysis [22] to a setting where the data points are defined in the SPD manifold. It applies rigid transformations to data points (i.e., translation, stretching and rotation) from two datasets in order to match their statistical distributions. By the end of the procedure, both datasets are expected to follow the same statistical distribution Θ. We describe this step in Section II-E.

Figure 1 summarizes the steps described above and indicates the features of each dataset that change after the DT procedure.

Note that the data points transformed via DT are defined in higher-dimensional spaces than their original versions, requiring more space to be stored. A natural question to ask is whether it would be preferable to reduce the dimensionality

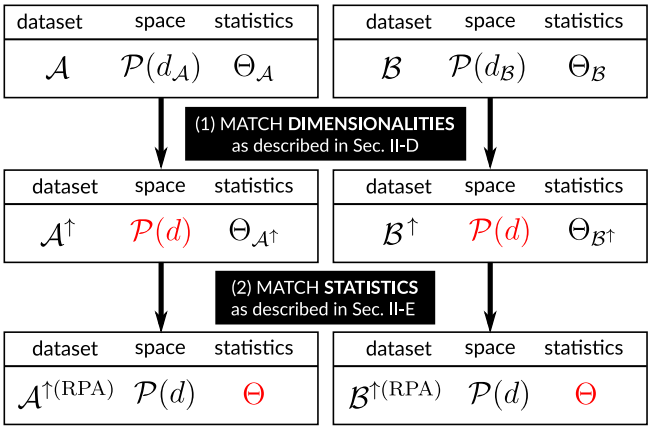

Fig. 1. Summary of the steps in the dimensionality transcending procedure applied to two datasets, $\mathcal{A}$ and $\mathcal{B}$, initially defined in SPD manifolds of different dimensionalities. The upward arrow (↑) on the names of the datasets indicates a dimensionality-augmentation step and a (RPA) superscript means that the Riemannian Procrustes Analysis [24] was used to match the statistics of the datasets. Quantities colored in **red** are those that change after each step.

of the data points into a common space (using, for example, the methods presented in [23]) and then apply a procedure for statistical matching on the new data points (using, for instance, RPA). This would avoid increasing the dimensionality of the data points, however, it would also have the risk of losing important discriminative information from the datasets.

An example. Suppose we have two datasets, $\mathcal{A}$ and $\mathcal{B}$, consisting of SPD matrices that describe the statistics of EEG epochs recorded with electrode sets $\mathcal{E}_{\mathcal{A}} = \{\mathtt{Fz}, \mathtt{C3}, \mathtt{C4}, \mathtt{Pz}\}$ and $\mathcal{E}_{\mathcal{B}} = \{\mathtt{C4}, \mathtt{C3}, \mathtt{Cz}\}$. The DT procedure begins with a dimensionality matching step in which we define a new set $\mathcal{E} = \mathcal{E}_{\mathcal{A}} \cup \mathcal{E}_{\mathcal{B}} = \{\mathtt{C3}, \mathtt{C4}, \mathtt{Fz}, \mathtt{Cz}, \mathtt{Pz}\}$ whose order is considered as fixed. Then, we augment the dimensionality of the data points in both $\mathcal{A}$ and $\mathcal{B}$, so as to have new SPD matrices defined in $\mathcal{P}(d)$, with $d = |\mathcal{E}| = 5$. If necessary, we may change the ordering of the electrodes of the augmented matrices so that they correspond to the order imposed by $\mathcal{E}$: this ensures that the dimensions from the expanded versions of $\mathcal{A}$ and $\mathcal{B}$ are comparable. Finally, we use RPA to match the statistics of the dimensionality-augmented datasets.

D. Expanding the dimensions of a SPD matrix

In what follows, we present the general problem of transforming a d'-dimensional SPD matrix into a d-dimensional SPD matrix ($d > d'$). We show how such transformation has to be defined in order to guarantee the positive definiteness of the d-dimensional matrices and how certain geometric constraints can be imposed.

Choosing how to expand. Without loss of generality, we will first assume that $d = d' + 1$, so that expanding a matrix $\boldsymbol{C} \in \mathcal{P}(d')$ amounts to defining two parameters $\boldsymbol{v} \in \mathbb{R}^{d'}$ and $\alpha \in \mathbb{R}$ to obtain

$$\boldsymbol{C}^{\uparrow} = \begin{bmatrix} \boldsymbol{C} & \boldsymbol{v} \\ \boldsymbol{v}^T & \alpha \end{bmatrix} \in \mathbb{R}^{(d'+1)\times(d'+1)} . \tag{10}$$

To guarantee that $\boldsymbol{C}^{\uparrow}$ is an element of $\mathcal{P}(d'+1)$, one can use the fact that a matrix is SPD if, and only if, all of its principal minors have positive determinants. Since $\boldsymbol{C}$ is SPD, the determinant of all of its principal minors are positive, so we can conclude that $\boldsymbol{C}^{\uparrow}$ will be SPD if, and only if, its determinant is positive. From matrix analysis, we know that [25]

$$\det\left(\begin{bmatrix} \boldsymbol{C} & \boldsymbol{v} \\ \boldsymbol{v}^T & \alpha \end{bmatrix}\right) = \det(\boldsymbol{C})\,(\alpha - \boldsymbol{v}^T\boldsymbol{C}^{-1}\boldsymbol{v}) , \tag{11}$$

thus, a necessary and sufficient condition for the expansion of $\boldsymbol{C}$, denoted by $\boldsymbol{C}^{\uparrow}$, to be SPD is

$$\boldsymbol{v}^T\boldsymbol{C}^{-1}\boldsymbol{v} < \alpha . \tag{12}$$

Geometry of expanded points. Once we know the conditions for α and $\boldsymbol{v}$, the next natural question is how the geometry of a set of data points $\mathcal{A} \subset \mathcal{P}(d')$ changes when its elements are expanded via Eq. (10) and forms a new set $\mathcal{A}^{\uparrow} \subset \mathcal{P}(d)$. For this, we need to understand how the distance between two expanded data points in $\mathcal{P}(d')$ relates to their distance in $\mathcal{P}(d)$.

Consider we expand two SPD matrices $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$ by Eq. (10). Using a $\boldsymbol{v}$ respecting condition (12) for both $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$, and, without loss of generality, fixing $\alpha = 1$, the Riemannian geodesic distance between the expanded matrices is given by

$$\delta_R^2(\boldsymbol{C}_i^{\uparrow}, \boldsymbol{C}_j^{\uparrow}) = \sum_{k=1}^{d} \log^2(\lambda_k^{\uparrow}) ,$$

where $\lambda((\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow}) = \{\lambda_1^{\uparrow}, \ldots, \lambda_d^{\uparrow}\}$ is the set of eigenvalues of $(\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow}$. Similarly, the distance between $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$ is given by

$$\delta_R^2(\boldsymbol{C}_i, \boldsymbol{C}_j) = \sum_{k=1}^{d'} \log^2(\lambda_k) ,$$

where $\lambda(\boldsymbol{C}_i^{-1}\boldsymbol{C}_j) = \{\lambda_1, \ldots, \lambda_{d'}\}$. Our goal is to be able to write $\delta_R^2(\boldsymbol{C}_i^{\uparrow}, \boldsymbol{C}_j^{\uparrow})$ in terms of $\delta_R^2(\boldsymbol{C}_i, \boldsymbol{C}_j)$. For this, we write explicitly the expression for the expanded matrix

$$(\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow} = \begin{bmatrix} \boldsymbol{C}_i^{-1}\boldsymbol{C}_j + \boldsymbol{C}_i^{-1}\boldsymbol{v}\boldsymbol{v}^T & \dfrac{\boldsymbol{C}_i^{-1}\boldsymbol{C}_j - I_{d'}}{1 - \boldsymbol{v}^T\boldsymbol{C}_i^{-1}\boldsymbol{v}} & \boldsymbol{0}_{d'\times 1} \\ \boldsymbol{v}^T(\boldsymbol{I}_{d'} - \boldsymbol{C}_i^{-1}\boldsymbol{C}_j) & & 1 \end{bmatrix} ,$$

where $\boldsymbol{0}_{r\times s}$ is a $r \times s$ matrix filled with zeros and $\boldsymbol{I}_{d'}$ is a d'-dimensional Identity matrix. Because of the block structure of $(\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow}$, it is easy to show that

$$\lambda\Big((\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow}\Big) = \{1\} \cup \lambda\Big(\big((\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow}\big)_{\mathrm{UL}}\Big) ,$$

where $\big((\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow}\big)_{\mathrm{UL}}$ is the upper-left block of $(\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow}$.

An isometric transformation. Different choices of $\boldsymbol{v}$ will lead to different expressions for $\lambda((\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow})$ and, consequently, different relations between $\delta_R^2(\boldsymbol{C}_i, \boldsymbol{C}_j)$ and $\delta_R^2(\boldsymbol{C}_i^{\uparrow}, \boldsymbol{C}_j^{\uparrow})$. Among such relations, one that is particularly interesting is when

$$\delta_R^2(\boldsymbol{C}_i, \boldsymbol{C}_j) = \delta_R^2(\boldsymbol{C}_i^{\uparrow}, \boldsymbol{C}_j^{\uparrow}) , \tag{13}$$

meaning that the expansion preserves the pairwise distance between the data points $\boldsymbol{C}_i, \boldsymbol{C}_j \in \mathcal{P}(d')$ in the new space

$\mathcal{P}(d)$. An interesting consequence is that classification algorithms that use distances between data points as features (e.g., the MDM classifier [26]) have exactly the same performance when applied to the data points in $\mathcal{P}(d')$ or to their transformed version in $\mathcal{P}(d)$. Therefore, we ensure that the dimensionality augmentation does not affect (either negatively or positively) the discriminatory power of classifiers over the transformed datasets. A simple algebraic solution that preserves the pairwise distances is to choose $\boldsymbol{v} = \mathbf{0}_{d'\times 1}$, so that

$$(\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow} = \begin{bmatrix} \boldsymbol{C}_i^{-1}\boldsymbol{C}_j & \mathbf{0}_{d'\times 1} \\ \mathbf{0}_{1\times d'} & 1 \end{bmatrix} ,$$

with

$$\lambda\Big((\boldsymbol{C}_i^{\uparrow})^{-1}\boldsymbol{C}_j^{\uparrow}\Big) = \{1\} \ \cup \ \lambda(\boldsymbol{C}_i^{-1}\boldsymbol{C}_j) ,$$

and, consequently,

$$\begin{aligned} \delta_R^2(\boldsymbol{C}_i^{\uparrow}, \boldsymbol{C}_j^{\uparrow}) &= \sum_{k=1}^{d} \log^2(\lambda_k^{\uparrow}) , \\ &= \sum_{k=1}^{d'} \log^2(\lambda_k) + \log^2(1) , \\ &= \delta_R^2\left(\boldsymbol{C}_i, \boldsymbol{C}_j\right) . \end{aligned}$$

Note that this choice also ensures that Eq. (12) is verified for any positive α and any pair of matrices $\boldsymbol{C}_i, \boldsymbol{C}_j \in \mathcal{P}(d')$. By induction, one can easily show that the same reasoning holds for any $d' > d$ and an expansion given by

$$\boldsymbol{C}^{\uparrow} = \begin{bmatrix} \boldsymbol{C} & \mathbf{0}_{d'\times p} \\ \mathbf{0}_{p\times d'} & \boldsymbol{I}_p \end{bmatrix} ,$$

where $p = d' - d$.

Based on the results above, we may define the transformation

$$\begin{aligned} E_{d'\to d} \ : \ \mathcal{P}(d') &\to \mathcal{P}(d) \\ \boldsymbol{C} &\mapsto \begin{bmatrix} \boldsymbol{C} & \mathbf{0}_{d'\times p} \\ \mathbf{0}_{p\times d'} & \boldsymbol{I}_p \end{bmatrix} , \end{aligned} \tag{14}$$

with $p = d - d'$, which is an isometric transformation between manifolds $\mathcal{P}(d')$ and $\mathcal{P}(d)$ in terms of the AIRM distance between SPD matrices, that is,

$$\delta_R^2\Big(E_{d'\to d}(\boldsymbol{C}_i), E_{d'\to d}(\boldsymbol{C}_j)\Big) = \delta_R^2\left(\boldsymbol{C}_i, \boldsymbol{C}_j\right) . \tag{15}$$

Occam's razor. Instead of fixing $\boldsymbol{v} = \mathbf{0}_{d'\times 1}$ in Eq. (10), we could have chosen a data-driven approach for determining an appropriate vector $\boldsymbol{v}$ for a set of data points in $\mathcal{A} = \{\boldsymbol{C}_i\}_{i=1}^{K} \subset \mathcal{P}(d')$. The output of such procedure would have to satisfy condition (12) for each element of $\mathcal{A}$. Moreover, if the distances between each pair of matrices $\boldsymbol{C}_i, \boldsymbol{C}_j \in \mathcal{A}$ were to be preserved after the expansion via Eq. (10), the data-driven approach would have to take into account $(K-1)K/2$ additional constraints, which could lead to a very challenging algorithmic problem. Furthermore, the vector $\boldsymbol{v}$ would have to be recalculated for every new data point added to $\mathcal{A}$. In light of all such constraints, we here prefer to retain the simple algebraic solution $\boldsymbol{v} = \mathbf{0}_{d'\times 1}$, which ensures the isometric property of the dimensionality augmentation step for any pair of data points defined in $\mathcal{P}(d')$.

A time series interpretation. As mentioned in the Introduction, we are particularly interested in the case where our SPD matrices are spatial covariance matrices describing the second-order statistics of zero-mean multivariate time series. For a T-sample realization of a zero-mean d'-dimensional time series, $\boldsymbol{X} \in \mathbb{R}^{d'\times T}$, the spatial covariance matrix, $\boldsymbol{C} \in \mathcal{P}(d')$, is estimated as

$$\boldsymbol{C} = \frac{1}{T}\boldsymbol{X}\boldsymbol{X}^T . \tag{16}$$

We can interpret, then, that the dimensionality augmentation transformation $E_{d'\to d}$ applied to $\boldsymbol{C}$ adds $p = d - d'$ new dimensions to the multivariate time series $\boldsymbol{X}$ and fill them with a T-sample realization of a p-dimensional uncorrelated white noise, $\boldsymbol{x}_p$, with

$$\boldsymbol{X}^{\uparrow} = \begin{bmatrix} \boldsymbol{X} \\ \mathbf{x}_p \end{bmatrix} . \tag{17}$$

It is worth noting that by adding uncorrelated white noise to the new dimensions of Eq. (17), we follow a maximum entropy approach, that is, we use no *a priori* information to fill the new samples of the time series. This serves as further justification to our choice of expanding SPD matrices via zero-padding[1] instead of using a data-driven approach.

Comparison with interpolation. Our dimensionality augmentation step may be interpreted as a way to fill p dimensions of a multivariate time series $\boldsymbol{X}^{\uparrow}$ with samples whose second order statistics have some desired structure. Note, however, that there exists other methods to solve this problem. For instance, for magnetoencephalographic (MEG) and EEG signals the method of reference in the literature is the *spherical spline interpolation* [27], which fills the signals on problematic channels by taking linear combinations of the time series on electrodes which are spatially close to them. Unfortunately, although such expanded times series are d-dimensional, their row-rank is only d', so their spatial covariance matrices are rank-defficient. Therefore, we cannot use the Riemannian geometric framework presented in Section II-A to classify them.

Statistics of the expanded data points. Consider a set of SPD data points

$$\mathcal{C} = \Big\{\boldsymbol{C}_1, \ldots, \boldsymbol{C}_{K_{\mathcal{A}}}\Big\} \subset \mathcal{P}(d') , \tag{18}$$

with geometric mean $\boldsymbol{M}^{\mathcal{C}}$ and dispersion $\sigma^{\mathcal{C}}$. Expanding each element of $\mathcal{C}$, we obtain a new set of SPD matrices

$$\mathcal{C}^{\uparrow} = \Big\{\boldsymbol{C}_1^{\uparrow}, \ldots, \boldsymbol{C}_{Kc}^{\uparrow}\Big\} \subset \mathcal{P}(d) , \tag{19}$$

[1]Note that the origin of the SPD manifold is the identity matrix, so a zero-padding in this space includes a diagonal of ones

where $\boldsymbol{C}_k^{\uparrow} = E_{d'\to d}(\boldsymbol{C}_k)$. The geometric mean of $\mathcal{C}^{\uparrow}$ is

$$\begin{aligned} \boldsymbol{M}^{\mathcal{C}^{\uparrow}} &= \underset{\boldsymbol{M}^{\uparrow}\in\mathcal{P}(d)}{\operatorname{argmin}} \sum_{k=1}^{K_{\mathcal{S}}} \delta_R^2(\boldsymbol{M}^{\uparrow}, \boldsymbol{C}_k^{\uparrow}) \ , \\ &= \begin{bmatrix} \boldsymbol{M}^{\mathcal{C}} & \boldsymbol{0}_{d'\times p} \\ \boldsymbol{0}_{p\times d'} & \boldsymbol{I}_p \end{bmatrix} \ , \\ &= E_{d'\to d}(\boldsymbol{M}^{\mathcal{C}}) \ . \end{aligned} \tag{20}$$

with $p = d - d'$. Moreover, because of the isometric property of transformation (14), we have that $\sigma^{\mathcal{C}^{\uparrow}} = \sigma^{\mathcal{C}}$. Finally, note that if each element of $\mathcal{C}$ had a class label associated to it, the class means of their expanded counterparts would be determined as in Eq. (20).

E. Matching the statistics of two datasets

Expanding the $d_{\mathcal{A}}$-dimensional data points from $\mathcal{A}$ and the $d_{\mathcal{B}}$-dimensional data points from $\mathcal{B}$ yields two new datasets,

$$\begin{aligned} \mathcal{A}^{\uparrow} &= \left\{ (\boldsymbol{C}_i^{\mathcal{A}^{\uparrow}},\ \ell_i^{\mathcal{A}}) \text{ for } i = 1, \ldots, K_{\mathcal{A}} \right\} \ , \\ \mathcal{B}^{\uparrow} &= \left\{ (\boldsymbol{C}_i^{\mathcal{B}^{\uparrow}},\ \ell_i^{\mathcal{B}}) \text{ for } i = 1, \ldots, K_{\mathcal{B}} \right\} \ , \end{aligned} \tag{21}$$

where the $\boldsymbol{C}_i^{\mathcal{A}^{\uparrow}}$ and $\boldsymbol{C}_i^{\mathcal{B}^{\uparrow}}$ are all d-dimensional SPD matrices. The next step is to transform the elements of each dataset so that their statistical distributions, $\Theta_{\mathcal{A}^{\uparrow}}$ and $\Theta_{\mathcal{B}^{\uparrow}}$, become as close as possible. To do so, we use the Riemannian Procrustes analysis (RPA), which is a generalization of the classical Procrustes Analysis to a non-Euclidean setting [10]. The method considers the distributions of points in two datasets (the *source* and *target* datasets) as shapes in a high-dimensional space and performs rigid geometric operations to make their shapes as similar as possible (see Figure 2 for a visual representation of these operations). The steps involved in the procedure are summarized as follows:

- *Re-center* the data points in $\mathcal{A}^{\uparrow}$ and $\mathcal{B}^{\uparrow}$ such as

$$\begin{aligned} \boldsymbol{C}_i^{\mathcal{A}^{\uparrow(\text{rct})}} &= \left(\boldsymbol{M}^{\mathcal{A}^{\uparrow}}\right)^{-1/2} \boldsymbol{C}_i^{\mathcal{A}^{\uparrow}} \left(\boldsymbol{M}^{\mathcal{A}^{\uparrow}}\right)^{-1/2} \ , \\ \boldsymbol{C}_i^{\mathcal{B}^{\uparrow(\text{rct})}} &= \left(\boldsymbol{M}^{\mathcal{B}^{\uparrow}}\right)^{-1/2} \boldsymbol{C}_i^{\mathcal{B}^{\uparrow}} \left(\boldsymbol{M}^{\mathcal{B}^{\uparrow}}\right)^{-1/2} \ . \end{aligned} \tag{22}$$

  This forms two new datasets, $\mathcal{A}^{\uparrow(\text{rct})}$ and $\mathcal{B}^{\uparrow(\text{rct})}$, whose statistical distributions are parametrized by

$$\begin{aligned} \Theta_{\mathcal{A}^{\uparrow(\text{rct})}} &= \left\{ \boldsymbol{I}_d, \boldsymbol{M}_1^{\mathcal{A}^{\uparrow(\text{rct})}}, \ldots, \boldsymbol{M}_L^{\mathcal{A}^{\uparrow(\text{rct})}}, \sigma^{\mathcal{A}} \right\} \ , \\ \Theta_{\mathcal{B}^{\uparrow(\text{rct})}} &= \left\{ \boldsymbol{I}_d, \boldsymbol{M}_1^{\mathcal{B}^{\uparrow(\text{rct})}}, \ldots, \boldsymbol{M}_L^{\mathcal{B}^{\uparrow(\text{rct})}}, \sigma^{\mathcal{B}} \right\} \ . \end{aligned} \tag{23}$$

  The reader with a signal processing background will recognize Eq. (22) as a whitening step applied to the multivariate time series associated to each of the SPD matrices. In differential geometry, these operations are also known as the parallel transport on the SPD manifold [29]. Intuitively, the re-centering step may be seen as a translation of the center of mass of the data points from each dataset to a common reference.

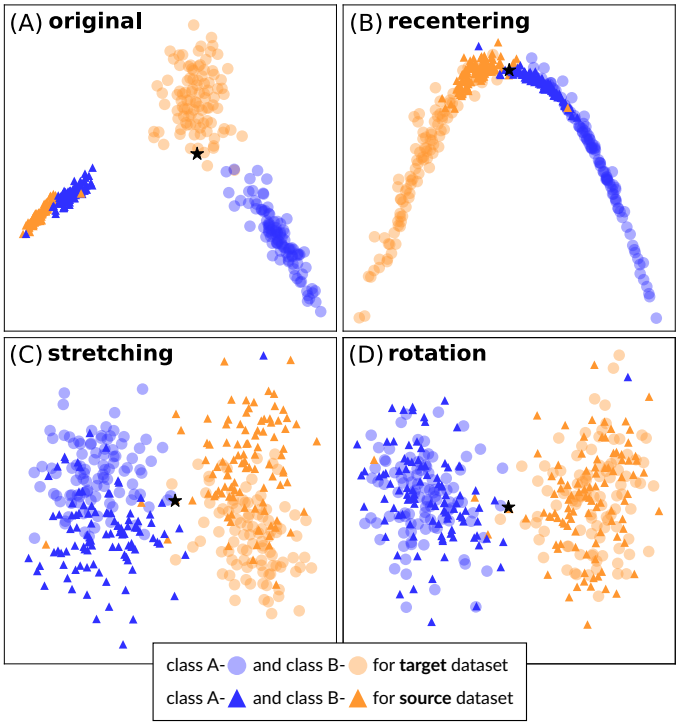

Fig. 2. Representation of the sequence of operations of RPA applied to a simulated *source* and *target* datasets (see [24] for more details on the simulation). Each point on the scatter plot represents a SPD matrix and the axes for the figures were obtained using diffusion maps [28]. The filled dots (degree of transparency set to $\alpha = 0.30$) represent the *target* dataset whereas the triangles are the *source* dataset. Each color represents a class and the black star is the Identity matrix. (A) Distribution of the SPD matrices in the *source* and *target* datasets as they are originally available and (B) after re-centering their geometric means to the Identity. In (C) the distribution after the stretching operation and (D) after the rotation. See the animation in `https://youtu.be/QzyikCLNAWI` linking the operations of each figure. (Figure reused from [24]. Copyright © 2019, IEEE)

- *Stretch* the dispersion around the mean for the points in $\mathcal{A}^{\uparrow(\text{rct})}$ and $\mathcal{B}^{\uparrow(\text{rct})}$ so that they are equal to one, as

$$\begin{aligned} \boldsymbol{C}_i^{\mathcal{A}^{\uparrow(\text{rct+str})}} &= \left(\boldsymbol{C}_i^{\mathcal{A}^{\uparrow(\text{rct})}}\right)^{1/\sigma^2_{\mathcal{A}}} \ , \\ \boldsymbol{C}_i^{\mathcal{B}^{\uparrow(\text{rct+str})}} &= \left(\boldsymbol{C}_i^{\mathcal{B}^{\uparrow(\text{rct})}}\right)^{1/\sigma^2_{\mathcal{B}}} \ . \end{aligned} \tag{24}$$

  This yields two new datasets $\mathcal{A}^{\uparrow(\text{rct+str})}$ and $\mathcal{B}^{\uparrow(\text{rct+str})}$ with equal dispersions and distributions parametrized as

$$\begin{aligned} \Theta_{\mathcal{A}^{\uparrow(\text{rct+str})}} &= \left\{ \boldsymbol{I}_d, \boldsymbol{M}_1^{\mathcal{A}^{\uparrow(\text{rct+str})}}, \ldots, \boldsymbol{M}_L^{\mathcal{A}^{\uparrow(\text{rct+str})}}, 1 \right\} \ , \\ \Theta_{\mathcal{B}^{\uparrow(\text{rct+str})}} &= \left\{ \boldsymbol{I}_d, \boldsymbol{M}_1^{\mathcal{B}^{\uparrow(\text{rct+str})}}, \ldots, \boldsymbol{M}_L^{\mathcal{B}^{\uparrow(\text{rct+str})}}, 1 \right\} \ . \end{aligned}$$

  The stretching operation is analogous to the variance normalization usually done in statistical data analysis. We may interpret its combination with the re-centering as a standardization procedure.
- *Rotate* the data points from $\mathcal{B}^{\uparrow(\text{rct+str})}$ to make its class means as close as possible to the class means of $\mathcal{A}^{\uparrow(\text{rct+str})}$. We have then

$$\begin{aligned} \boldsymbol{C}_i^{\mathcal{A}^{\uparrow(\text{rct+str+rot})}} &= \boldsymbol{C}_i^{\mathcal{A}^{\uparrow(\text{rct+str})}} \ , \\ \boldsymbol{C}_i^{\mathcal{B}^{\uparrow(\text{rct+str+rot})}} &= \boldsymbol{U}^T \boldsymbol{C}_i^{\mathcal{B}^{\uparrow(\text{rct+str})}} \boldsymbol{U} \ , \end{aligned}$$

with $\boldsymbol{U}$ obtained from the optimization problem

$$\underset{\boldsymbol{U}^T\boldsymbol{U}=\boldsymbol{I}_d}{\text{minimize}} \quad \sum_{c=1}^{L} \delta_R^2 \left(\boldsymbol{U}^T \boldsymbol{M}_c^{\mathcal{B}^{\uparrow(\text{rct+str})}} \boldsymbol{U},\ \boldsymbol{M}_c^{\mathcal{A}^{\uparrow(\text{rct+str})}} \right) . \quad (25)$$

The rotation step acts to match the class means of the datasets via the application of an orthogonal matrix. The precise interpretation of this transformation in terms of changes in the time series associated to each SPD matrix remains an open question, but we know that it is closely related to the differences in the electrode positioning of each dataset [24].

- *Form* two new datasets

$$\mathcal{A}^{\uparrow(\text{RPA})} = \left\{ (\boldsymbol{C}_i^{\mathcal{A}^{\uparrow(\text{rct+str+rot})}},\ \ell_i^{\mathcal{A}}) \text{ for } i = 1, \ldots, K_{\mathcal{A}} \right\} ,$$

$$\mathcal{B}^{\uparrow(\text{RPA})} = \left\{ (\boldsymbol{C}_i^{\mathcal{B}^{\uparrow(\text{rct+str+rot})}},\ \ell_i^{\mathcal{B}}) \text{ for } i = 1, \ldots, K_{\mathcal{B}} \right\} .$$

By the end of the RPA procedure, the statistical distributions of the datasets become as close as possible. One way to measure such proximity is in terms of the maximum-mean discrepancy, as shown in [30].

F. Pre-processing BCI data

A typical BCI experiment consists of several trials during which a subject performs a task and the goal is to be able to infer which task the subject was performing based on the EEG signals. Put in mathematical terms, if the recordings are obtained on d electrodes, and each one of the K trials is composed of T time samples, the typical BCI dataset is composed of a set of coupled pairs

$$\mathcal{X} = \{(\boldsymbol{X}_i, \ell_i)\}_{i=1}^{K} \subset \mathbb{R}^{d \times T} \times \{1, \ldots, L\} ,$$

where $\boldsymbol{X}_i$ is the i-th EEG trial recorded in the experiment and ℓ_i its associated label (out of L possible classes). Different BCI paradigms consist of different cognitive tasks and the recorded signals are filtered differently. For instance, in motor imagery (MI), the epochs in $\mathcal{X}$ are usually bandpass filtered between 8 Hz and 35 Hz, whereas for experiments based on the P300 component of event-related potentials, the samples are filtered between 1 Hz and 24 Hz. All our EEG processing is performed using MNE for Python [31].

We use the RG framework described in Section II-A to manipulate the trials of the BCI experiment. For this purpose, for each trial $\boldsymbol{X}_i \in \mathcal{X}$ we estimate a SPD matrix $\boldsymbol{C}_i \in \mathcal{P}(d)$ which describes its second-order statistics. For data following the MI paradigm, this matrix is simply the spatial covariance matrix of the EEG signals in the trial, as given by Eq. (16) (note that because of the bandpass filtering, the signals in $\boldsymbol{X}_i$ are zero-mean), whereas for the P300 paradigm it is a special augmented covariance matrix defined in $\mathcal{P}(2d)$, as described in [32]. We classify the BCI data using the minimum distance to mean clasifier (MDM), which is a geometry-aware classifier that generalizes the nearest-centroid classifier to the case where the data points are defined in the SPD manifold [26]. In the *training* phase, the MDM calculates the geometric means for each class of a *training* dataset ($\mathcal{D}_{\text{train}}$). In the *testing* phase, each SPD matrix from the *testing* dataset ($\mathcal{D}_{\text{test}}$) is associated to the label of the closest class mean. Note that we could have used other more flexible and powerful classifiers for SPD data, such as the probabilistic classifier proposed in [33]. However, since this contribution is mainly concerned with the statistical and dimensionality matching of datasets, we have preferred to illustrate this proposal using the simplest classifier available yet well validated for BCI classification [3].

G. Datasets

We present examples with BCI data from both MI and P300 paradigms. For MI, we use the `Zhou2016` [34], `BNCI2015001` [35], and `AlexMI` [36] datasets. The `Zhou2016` dataset consists of recordings on 14 electrodes from 4 subjects executing either a left-hand/right-hand or a feet/right-hand motor imagery task; we denote these sub-datasets by `Zhou2016-LR` (`LR` for left-hand/right-hand) and `Zhou2016-FR` (`FR` for feet/right-hand). Dataset `BNCI2015001` is composed of EEG signals from 13 electrodes and 12 subjects (from which we have selected 7 with the best self-scores, e.g., the score of a classifier trained and tested on the same dataset), all executing a feet/right-hand motor imagery task. Dataset `AlexMI` contains recordings from 8 subjects performing right hand/feet motor imagination (from which we selected the 4 with best self-scores). The classes on all datasets are balanced.

The examples on the P300 paradigm use the `BNCI2014009` [37], `BI.2012` [38], and `BI.2013` [39] datasets. All three datasets were recorded with 16 electrodes, but each one used a different positioning. For `BNCI2014009` we used the 5 subjects with the best self-scores (out of 10 available subjects), for `BI.2012` we selected the best 8 subjects out of 24 available subjects, and for `BI.2013` we selected the 8 subjects with the best self-scores out of 25 available subjects. All P300 datasets are from recordings on experiments with a 6-by-6 grid with flashing cues, but the subjects' cognitive tasks are slightly different : in `BNCI2014009` they must concentrate on letters to spell words, whereas in `BI.2012` and `BI.2013` the subjects are asked to fix their attention on target cues representing "aliens" to be destroyed. The classes of the trials are unbalanced, with one "target" trial for every five "non-target" trials.

All datasets mentioned above are publicly available on the `MOABB` framework [40].

H. Experiments

The goal of our experiments is to assess whether dimensionality transcending allows a classifier to leverage from discriminative information in EEG data recorded from other subjects, even if they were obtained under different experimental setups. We consider the semi-supervised cross-subject transfer learning paradigm for BCI, where one wants to determine the unknown labels from a *target* dataset ($\mathcal{T}_u$) using information from a few labeled trials in the target dataset ($\mathcal{T}_\ell$), as well as the full information available from a *source* dataset ($\mathcal{S}$) containing recordings from another subject. We compare three classification pipelines assuming that there are different numbers of labeled covariance matrices from each class on the $\mathcal{T}_\ell$ dataset:

- **calibration**: the data points in $\mathcal{T}_u$ are classified using a MDM classifier trained with only the labeled data points available in the $\mathcal{T}_\ell$ dataset, such as

$$\mathcal{D}_{\text{train}} = \mathcal{T}_\ell \quad \text{and} \quad \mathcal{D}_{\text{test}} = \mathcal{T}_u \ .$$

- **DT-uns**: only the *uns*upervised steps (hence DT-*uns*) of the RPA are used, re-centering and stretching, for matching the statistics of two dimension-matched datasets. A MDM classifier is trained on a set containing the labeled covariance matrices from the *target* dataset as well as the dimension-matched and RPA-transformed data points from a source subject (for which we know all the labels), such as

$$\mathcal{D}_{\text{train}} = \mathcal{T}_\ell^{\uparrow(\text{rct+str})} \cup \mathcal{S}^{\uparrow(\text{rct+str})} \quad \text{and} \quad \mathcal{D}_{\text{test}} = \mathcal{T}_u^{\uparrow(\text{rct+str})} \ .$$

  This variant is relevant because it does not require knowledge of the labels of the trials on the *target* dataset.
- **DT**: full RPA (re-centering, stretching, rotation) is used to match the statistics of two dimension-matched datasets. A MDM classifier is trained on a set containing the labeled covariance matrices from the *target* dataset as well as the dimension-matched and RPA-transformed data points from a source subject (for which we know all the labels), such as

$$\mathcal{D}_{\text{train}} = \mathcal{T}_\ell^{\uparrow(\text{RPA})} \cup \mathcal{S}^{\uparrow(\text{RPA})} \quad \text{and} \quad \mathcal{D}_{\text{test}} = \mathcal{T}_u^{\uparrow(\text{RPA})} \ .$$

Note that we could have considered other pipelines where the statistical matching step would be carried out differently. However, in [10] we have demonstrated the superiority of RPA as compared to other statistical matching methods in the SPD manifold. Furthermore, other procedures from the literature on heterogeneous domain adaptation are not suitable for data points defined on a SPD space, which is why we have not included them in our comparisons.

We use the area under the ROC curve (AUC score) for quantifying the classification performance of the MDM classifier in all analyses. We randomly split the *target* dataset into labeled and unlabeled subsets five times and average the classification scores obtained in each realization. We may say that dimensionality transcending is useful for cross-subject transfer learning when the score of the **DT** pipeline is superior to that of the **calibration** pipeline, since it means that information from a *source* subject improved the classification score on a *target* dataset.

III. Results

Figures 3 and 4 provide a qualitative summary of the results for the cross-subject classification scores. The scores of the classification pipelines on each *target* subject are displayed on different rectangular regions in which the vertical lines indicate the scores for the **calibration** pipeline. The scatter points in each rectangular box represent the cross-subject scores for each *source* subject and each classification pipeline. The black dots represent the results with **DT** and the gray dots the results with **DT-uns**. The rectangular boxes are ordered according to the score of the **calibration** pipeline.

We also provide a quantitative comparison of the scores with the **DT** pipeline against those with **calibration** based on statistical hypothesis testing. We did our statistical analysis using paired t-tests with p-values obtained via permutation methods [41]:

(1) For each *target* subject i, we perform a signed paired t-test comparing the scores of method **DT** to **calibration** along all *source* subjects. Each of these tests yields a statistic T_i and a p-value p_i is obtained via permutations tests.
(2) We combine the p-values of all the *target* subjects using Stouffer's Z-score method [42]. This yields a single p-value for the comparison between methods as well as the direction to which the null hypothesis has been rejected (i.e., whether method **DT** is better than **calibration** or vice-versa).
(3) We adjust the p-values of each pairwise comparison using Holm's step-down procedure [43] to account for the multiple comparison problem.

The results are displayed in Table II, where the average values of the classification pipelines are taken over all the cross-subject classification scores for all pairs of *source-target* subjects.

IV. Discussion

As mentioned in Sec. II-H, the goal of our experiments was to investigate whether pipelines using dimensionality transcending (unsupervised and supervised) have better scores than just doing **calibration**. To assess this, we first examine the positions of the scatter points in Figure 3 and Figure 4 and compare them to the vertical lines corresponding to the **calibration** scores. We observe that the scores tend to be higher for *target* subjects for whom the **calibration** score is higher; this goes in line with observations from [21], where the *target* subjects with the best self-scores were also the best "receivers" of data from *source* subjects. We also observe that, in general, the results with **DT-uns** are inferior to that of **calibration**, whereas those for **DT** are, in most cases, superior to **calibration**, implying that the rotation step of RPA is indeed essential for the statistical matching of the datasets. This can be explained by the fact that the electrode positioning of the databases are different and the rotation matrix in RPA acts to mitigate such differences. Because of the poor performance of **DT-uns**, we limit our quantitative analysis to the results with **DT**. Table II shows that this pipeline is better (or at least equivalent) than **calibration** on most situations, confirming that it is indeed a good approach for leveraging discriminative information from other datasets.

It is interesting to observe how dimensionality transcending performs when the cognitive tasks of the subjects of each database are different. We first consider motor imagery data: `BNCI2015001` and `AlexMI` have trials for right-hand/feet MI tasks, whereas `Zhou2016-LR` has classes left-hand/right-hand. The results in Table II (two last rows on the left column) show that **DT** yields poorer results as compared to when the cognitive tasks are the same; they are in fact worse than **calibration** most of the time. This is related

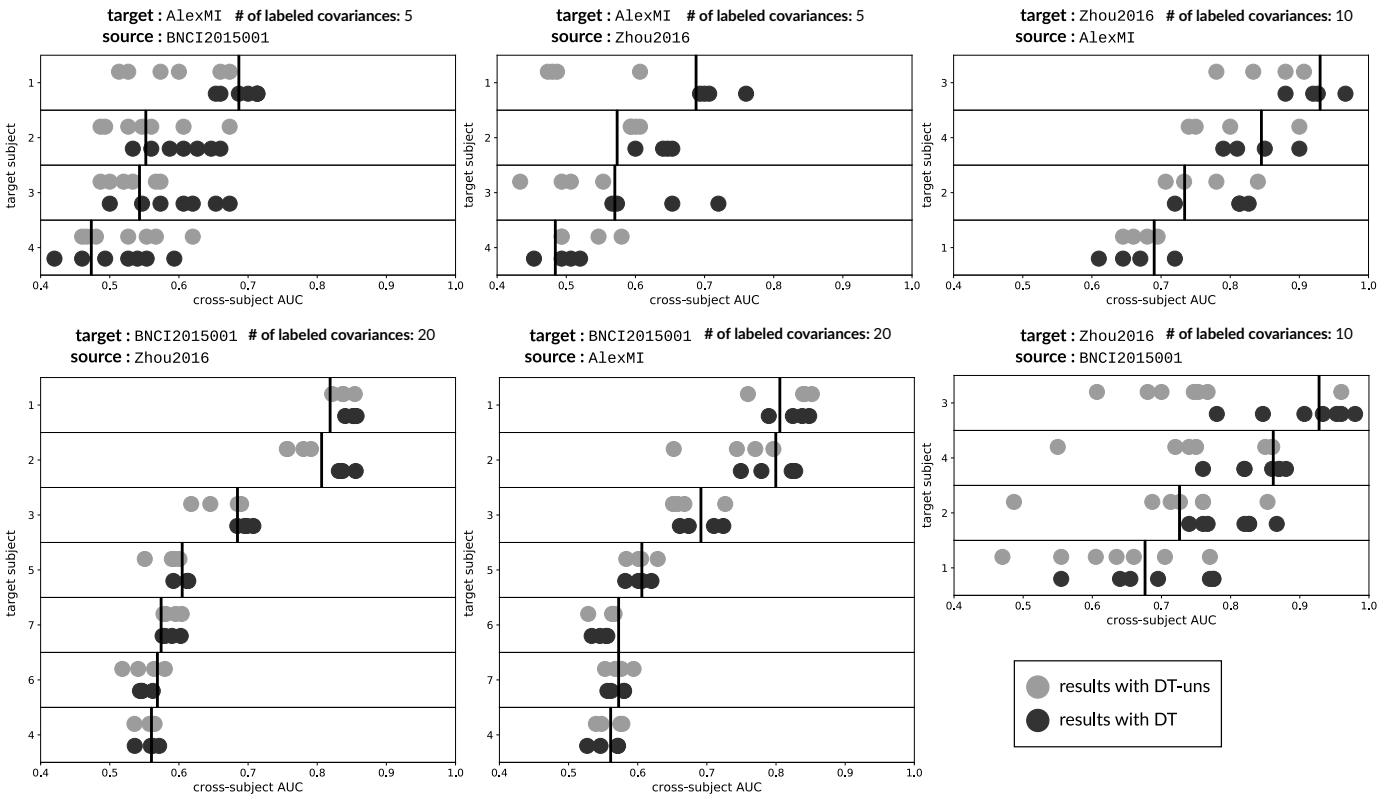

Fig. 3. Results for the MI paradigm. We represent the AUC scores for cross-subject classification considering different pairwise combinations of *source* and *target* databases. The scores of each *target* subject are displayed in different rectangular boxes, with the different scatter points indicating the scores obtained for each *source* subject. Different color markers indicate different classification pipelines (**DT-uns** is in **gray** and **DT** in **black**; see text for a description of each pipeline). The vertical line inside each rectangular box indicates the **calibration** score for the corresponding *target* subject for different number of labeled covariance matrices (indicated in the figure) in the *target* dataset.

to the fact that datasets containing signals which are not physiologically comparable are not expected to share the same discriminative information and, therefore, are incompatible for transfer learning. In our second example, the experimental protocols behind the generation of the P300 data were not always the same (subjects were asked to focus on a letter to spell in `BNCI2014009` and a 'target alien' in `BI.2012` and `BI.2013`). However, the cognitive tasks on both datasets were all based on concentrating on a given target cue. Consequently, the discriminative features used to do classification on these datasets were all related to the P300 component of the event-related potentials in the EEG signals. This explains why **DT** works well for all pairs of P300 databases, despite the differences in cognitive tasks. It is also worth noting that dimensionality transcending does not provide any new discriminative information: if the electrodes originally chosen for a certain dataset do not have any discriminatory power for a given BCI task, expanding the dimensionality of the data points will not improve the performance of classifiers trained on them. It remains an open question, however, how the difference in the number of electrodes on the *source* and *target* datasets quantitatively impacts the performance of the DT procedure.

Finally, we should comment on the relation between our choice of classifier and the performance of the **DT** pipeline. The RPA procedure used in DT's statistical matching step was derived in [10] for a setting where the datasets are sufficiently well described as a balanced mixture of Riemannian Gaussians in the SPD manifold (one mixture for each class, both with the same dispersion around the class mean). In such case, the geometric transformations in RPA are optimal and the MDM classifier is the most adequate classifier, since it is solely based on the difference between the means of the mixtures. However, if the hypothesis above is not satisfied, the RPA procedure should be adapted to match the statistics of datasets according to their characteristics and a different classifier might be preferable. In the context of BCI classification and, more particularly, for the datasets used in our numerical illustrations, the RPA+MDM tandem has already demonstrated good results [10], which motivates the choice of the MDM classifier.

V. Conclusion

In this work, we have tackled the problem of merging datasets that describe the same phenomenon but contain data points with different dimensionalities and/or different features. Our proposal, that we name "dimensionality transcending", consists of two steps: dimensionality matching followed by statistical distribution matching. We have illustrated our method using data from BCI experiments and investigated whether DT could be used for cross-subject classification when the data of the *source* and *target* subjects came from different databases. Our results show that a classification pipeline using

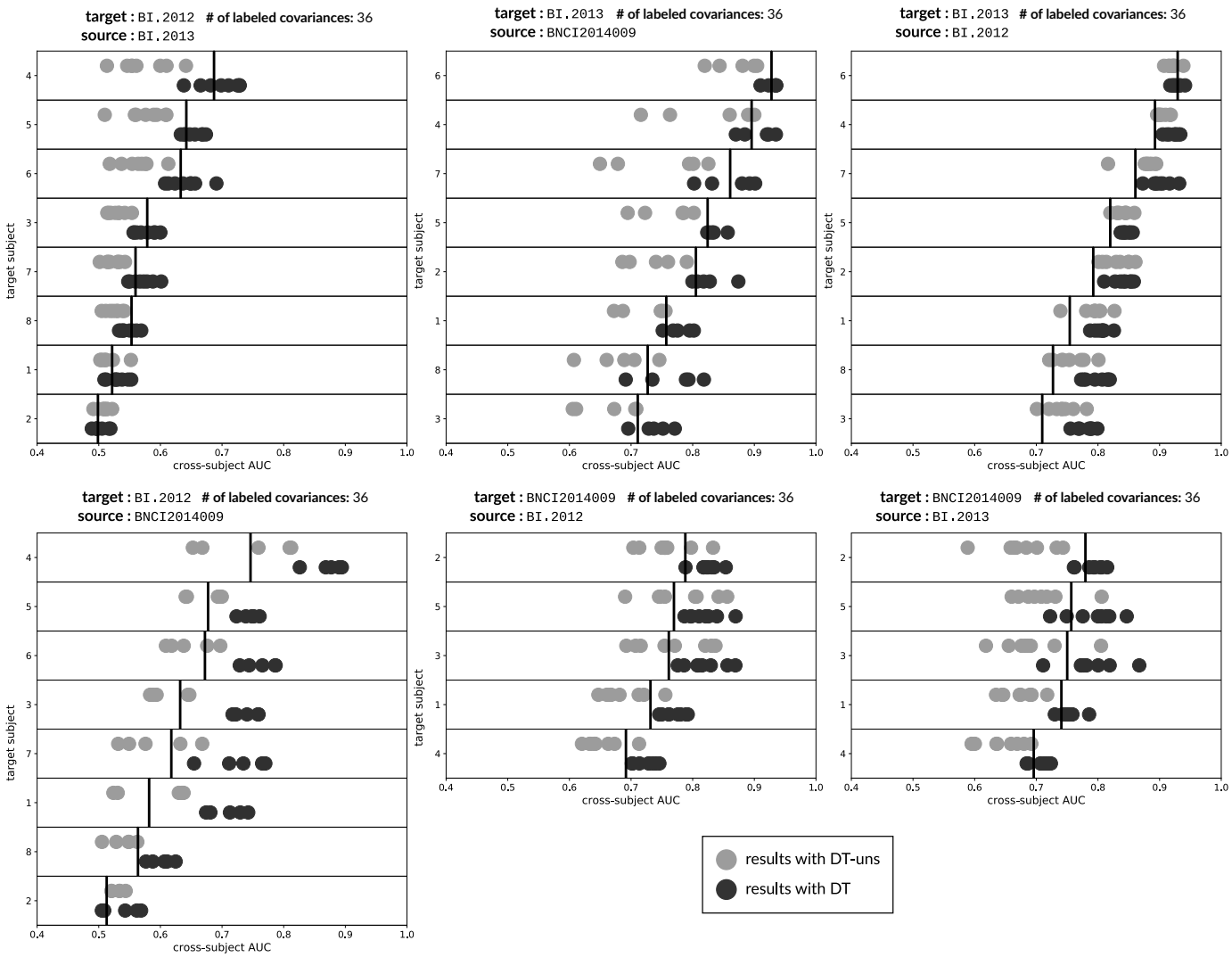

Fig. 4. Results for the P300 paradigm. We represent the AUC scores for cross-subject classification considering different pairwise combinations of *source* and *target* databases. The scores of each *target* subject are displayed in different rectangular boxes, with the different scatter points indicating the scores obtained for each *source* subject. Different color markers indicate different classification pipelines (**DT-uns** is in **gray** and **DT** in **black**; see text for a description of each pipeline). The vertical line inside each rectangular box indicates the **calibration** score for the corresponding *target* subject when a certain number of labeled covariance matrices (indicated in the figure) are available in the *target* dataset.

DT always attained superior (or at least equivalent) performance as compared to calibration, even with just a few labeled epochs, which is a remarkable result.

We have used the Riemannian geometry framework for working with SPD matrices describing the statistics of EEG recordings from BCI experiments. Because of this, we have presented a version of the DT technique tailored for the SPD manifold. It is easy, however, to extend the DT procedure to any other kind of data defined in a Riemannian manifold (e.g., the Euclidean space). To do so, we first need to determine an isometric transformation capable of taking data points with different dimensionalities into a common space (for instance, by padding zeros to an Euclidean feature vector) and, then, use some domain adaptation technique for matching the statistics of the datasets, such as the one proposed in [44] which uses optimal transport to match statistical distributions of data points defined in any metric space.

Our contribution is part of a larger effort in the machine learning research community with the goal of designing algorithms capable of extracting information shared between datasets with different dimensionalities, different statistical distributions, etc. The aim of such methods is to go against the current state of affairs of the "big data era", where large amounts of experimental data are gathered by different laboratories with total disregard to whether they can be jointly used for performing statistical tasks. On a societal point of view, such methods may be seen as "ecological", since they try to reuse and learn from information that already exists and for which some effort has already been put into its generation, the ultimate goal being to avoid the consumption of unnecessary energy for obtaining new data points as well as for storing them.

The topics considered in this paper open several important questions to be investigated in the future. For instance, the

TABLE II

MEAN VALUES OF THE AREA UNDER THE ROC CURVE (AUC) SCORE FOR CROSS-SUBJECT CLASSIFICATION USING PIPELINES **CALIBRATION** AND **DT**, ALL DESCRIBED IN THE TEXT. FOR EACH DATABASE BEING USED AS *target*, WE CONSIDER THREE DIFFERENT SIZES OF ITS LABELED PARTITION. THE FONTSTYLE OF THE AVERAGE SCORES REPRESENTED IN THE TABLE ARE DETERMINED FROM THE STATISTICAL TESTS THAT COMPARE THEIR VALUES WITH THAT OF **CALIBRATION**; SEE TEXT FOR AN EXPLANATION ON THE STATISTICAL PROCEDURE THAT WE USED. WHEN THE SCORE OF **DT** IS IN **BOLD** WITH A GRAY BACKGROUND, IT MEANS THAT IT IS BETTER THAN **CALIBRATION** IN AVERAGE, WHEREAS A SCORE THAT IS UNDERLINED INDICATES THAT THE PIPELINE'S PERFORMANCE IS INFERIOR TO **CALIBRATION** IN AVERAGE; A SCORE WITH NO FONTSTYLE IS ONE THAT IS NOT STATISTICALLY SIGNIFICANTLY DIFFERENT AS COMPARED TO **CALIBRATION**. THE LETTERS "T" AND "S" ON THE LEFT OF THE TABLE INDICATE WHICH DATABASE IS USED AS *target* AND *source* IN EACH COMPARISON.

Motor Imagery		DT			Calibration		
# of labeled covariances:		5	10	15	5	10	15
T: Zhou2016 / S: BNCI2015001		**0.77**	0.81	0.84	0.75	0.79	0.84
# of labeled covariances:		5	10	15	5	10	15
T: Zhou2016 / S: AlexMI		0.77	0.80	0.84	0.76	0.79	0.85
# of labeled covariances:		10	20	50	10	20	50
T: BNCI2015001 / S: Zhou2016		**0.65**	**0.67**	**0.70**	0.63	0.66	0.69
# of labeled covariances:		10	20	50	10	20	50
T: BNCI2015001 / S: AlexMI		0.64	0.65	0.69	0.64	0.65	0.69
# of labeled covariances:		1	5	10	1	5	10
T: AlexMI / S: BNCI2015001		**0.58**	**0.60**	**0.62**	0.53	0.56	0.59
# of labeled covariances:		1	5	10	1	5	10
T: AlexMI / S: Zhou2016		**0.55**	**0.62**	**0.64**	0.53	0.58	0.58
# of labeled covariances:		5	10	15	5	10	15
T: Zhou2016-LR / S: BNCI2015001		0.65	0.69	0.72	0.67	0.72	0.74
# of labeled covariances:		5	10	15	5	10	15
T: Zhou2016-LR / S: AlexMI		0.66	0.72	0.73	0.67	0.71	0.75

P300		DT			Calibration		
# of labeled covariances:		12	36	48	12	36	48
T: BI.2012 / S: BI.2013		**0.54**	**0.59**	**0.61**	0.53	0.58	0.6
# of labeled covariances:		12	36	48	12	36	48
T: BI.2012 / S: BNCI2014009		**0.62**	**0.71**	**0.74**	0.56	0.69	0.72
# of labeled covariances:		12	36	48	12	36	48
T: BI.2013 / S: BI.2012		**0.79**	**0.85**	**0.86**	0.66	0.80	0.83
# of labeled covariances:		12	36	48	12	36	48
T: BI.2013 / S: BNCI2014009		**0.77**	**0.84**	**0.85**	0.68	0.82	0.84
# of labeled covariances:		12	36	48	12	36	48
T: BNCI2014009 / S: BI.2013		**0.69**	**0.77**	**0.78**	0.62	0.74	0.76
# of labeled covariances:		12	36	48	12	36	48
T: BNCI2014009 / S: BI.2012		**0.72**	**0.79**	**0.80**	0.62	0.75	0.77

Same paradigm, but different cognitive tasks

T : target dataset
S : source dataset

parametrization of the statistical distributions of the SPD data points in the *source* and *target* datasets could be done so to take into account more complex distributions. Note, however, that in this case DT's distribution matching step would have to be carried out with a modified version of RPA. Another relevant question is how to use pooling and ensembling strategies for gathering information from several databases containing data points with different dimensionalities and combine them to form a single robust classifier as explored in [45].

VI. ACKNOWLEDGEMENT

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

REFERENCES

[1] M. Sugiyama, M. Krauledat, and K.-R. Müller, "Covariate shift adaptation by importance weighted cross validation," *J. Mach. Learn. Res.*, vol. 8, pp. 985–1005, Dec. 2007.

[2] F. Mémoli and G. Sapiro, "Comparing point clouds," in *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing - SGP 04*. ACM Press, 2004.

[3] M. Congedo, A. Barachant, and R. Bhatia, "Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review," *Brain-Computer Interfaces*, pp. 1–20, 2017.

[4] F. Yger, M. Berar, and F. Lotte, "Riemannian approaches in brain-computer interfaces: A review," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1753–1762, oct 2017.

[5] R. Bhatia, *Positive definite matrices*. Princeton university press, 2009.

[6] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, oct 2010.

[7] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Transactions on Neural Networks*, vol. 22, no. 2, pp. 199–210, Feb 2011.

[8] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *J. Mach. Learn. Res.*, vol. 13, pp. 723–773, Mar. 2012.

[9] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger, "A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update," *Journal of Neural Engineering*, vol. 15, no. 3, p. 031005, apr 2018.

[10] P. L. C. Rodrigues, C. Jutten, and M. Congedo, "Riemannian procrustes analysis: Transfer learning for brain-computer interfaces," *IEEE Transactions on Biomedical Engineering*, pp. 1–1, 2018.

[11] P. L. C. Rodrigues, F. Bouchard, M. Congedo, and C. Jutten, "Dimensionality reduction for BCI classification using Riemannian geometry," in *Graz BCI Conference 2017*, 2017.

[12] D. Sabbagh, P. Ablin, G. Varoquaux, A. Gramfort, and D. Engemann, "Manifold-regression to predict from MEG/EEG brain signals without source modeling," in *NeurIPS 2019 - 33th Annual Conference on Neural Information Processing Systems*, Vancouver, Canada, Dec. 2019. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02147708

[13] M. Moakher, "A differential geometric approach to the geometric mean of symmetric positive-definite matrices," *SIAM J. Matrix Anal. Appl.*, vol. 26, no. 3, pp. 735–747, Mar. 2005.

[14] X. Pennec, "Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements," *Journal of Mathematical Imaging and Vision*, vol. 25, no. 1, p. 127, 2006.

[15] M. Congedo, "EEG Source Analysis ," Habilitation à diriger des recherches, Université de Grenoble, Oct. 2013. [Online]. Available: https://tel.archives-ouvertes.fr/tel-00880483

[16] M. Arnaudon, F. Barbaresco, and L. Yang, "Riemannian medians and

means with applications to radar signal processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 4, pp. 595–604, Aug 2013.

[17] E. M. Massart, J. M. Hendrickx, and P.-A. Absil, "Matrix geometric means based on shuffled inductive sequences," *Linear Algebra and its Applications*, vol. 542, pp. 334 – 359, 2018, proceedings of the 20th ILAS Conference, Leuven, Belgium 2016.

[18] H. Karcher, "Riemannian center of mass and mollifier smoothing," *Communications on Pure and Applied Mathematics*, vol. 30, no. 5, pp. 509–541, Sep. 1977. [Online]. Available: https://doi.org/10.1002/cpa.3160300502

[19] F. Barbaresco, "Innovative tools for radar signal processing based on Cartan's geometry of SPD matrices information geometry," in *2008 IEEE Radar Conference*, May 2008, pp. 1–6.

[20] M. Congedo, A. Barachant, and E. K. Koopaei, "Fixed point algorithms for estimating power means of positive definite matrices," *IEEE Transactions on Signal Processing*, vol. 65, no. 9, pp. 2211–2220, May 2017.

[21] P. L. C. Rodrigues, M. Congedo, and C. Jutten, "Multivariate time-series analysis via manifold learning," in *2018 IEEE Statistical Signal Processing Workshop (SSP)*, June 2018, pp. 573–577.

[22] D. G. Kendall, "A survey of the statistical theory of shape," *Statistical Science*, vol. 4, no. 2, pp. 87–99, may 1989.

[23] M. Harandi, M. Salzmann, and R. Hartley, "Dimensionality reduction on SPD manifolds: The emergence of geometry-aware methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 1, pp. 48–62, jan 2018.

[24] P. L. C. Rodrigues, M. Congedo, and C. Jutten, "A data imputation method for matrices in the symmetric positive definite manifold," in *XXVIIème colloque GRETSI (GRETSI 2019)*, Lille, France, Aug. 2019.

[25] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. The Johns Hopkins University Press, 1996.

[26] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Multiclass brain–computer interface classification by Riemannian geometry," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 4, pp. 920–928, apr 2012.

[27] F. Perrin, J. Pernier, O. Bertrand, and J. Echallier, "Spherical splines for scalp potential and current density mapping," *Electroencephalography and Clinical Neurophysiology*, vol. 72, no. 2, pp. 184–187, Feb. 1989. [Online]. Available: https://doi.org/10.1016/0013-4694(89)90180-6

[28] S. Lafon and A. Lee, "Diffusion maps and coarse-graining: a unified framework for dimensionality reduction, graph partitioning, and data set parameterization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1393–1403, sep 2006.

[29] O. Yair, M. Ben-Chen, and R. Talmon, "Parallel transport on the cone manifold of SPD matrices for domain adaptation," *IEEE Transactions on Signal Processing*, vol. 67, no. 7, pp. 1797–1811, April 2019.

[30] P. Rodrigues, M. Congedo, and C. Jutten, "'When does it work ?' : An exploratory analysis of Transfer Learning for BCI," in *Graz BCI Conference 2019*, 2019.

[31] A. Gramfort, "MEG and EEG data analysis with MNE-python," *Frontiers in Neuroscience*, vol. 7, 2013. [Online]. Available: https://doi.org/10.3389/fnins.2013.00267

[32] M. Congedo, A. Barachant, and A. Andreev, "A New Generation of Brain-Computer Interface Based on Riemannian Geometry," *arXiv e-prints*, p. arXiv:1310.8115, Oct. 2013.

[33] S. Said, H. Hajri, L. Bombrun, and B. C. Vemuri, "Gaussian distributions on Riemannian symmetric spaces: Statistical learning with structured covariance matrices," *IEEE Transactions on Information Theory*, vol. 64, no. 2, pp. 752–772, Feb 2018.

[34] B. Zhou, X. Wu, Z. Lv, L. Zhang, and X. Guo, "A fully automated trial selection method for optimization of motor imagery based brain-computer interface," *PLOS ONE*, vol. 11, no. 9, p. e0162657, Sep. 2016.

[35] J. Faller, C. Vidaurre, T. Solis-Escalante, C. Neuper, and R. Scherer, "Autocalibration and recurrent adaptation: Towards a plug and play online ERD-BCI," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 20, no. 3, pp. 313–319, May 2012.

[36] A. Barachant, "Robust control of an actuator by EEG based asynchronous BCI," Ph.D Thesis, Université de Grenoble, Mar. 2012. [Online]. Available: https://tel.archives-ouvertes.fr/tel-01196752

[37] P. Aricò, F. Aloise, F. Schettini, S. Salinari, D. Mattia, and F. Cincotti, "Influence of P300 latency jitter on event related potential-based brain–computer interface performance," *Journal of Neural Engineering*, vol. 11, no. 3, p. 035008, May 2014. [Online]. Available: https://doi.org/10.1088/1741-2560/11/3/035008

[38] G. F. P. Van Veen, A. Barachant, A. Andreev, G. CATTAN, P. L. C. Rodrigues, and M. Congedo, "Building Brain Invaders: EEG data of an experimental validation," GIPSA-lab, Research Report 1, May 2019. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02126068

[39] E. Vaineau, A. Barachant, A. Andreev, P. L. C. Rodrigues, G. Cattan, and M. Congedo, "Brain invaders adaptive versus non-adaptive P300 brain-computer interface dataset," 2018.

[40] V. Jayaram and A. Barachant, "MOABB: trustworthy algorithm benchmarking for BCIs," *Journal of Neural Engineering*, vol. 15, no. 6, p. 066011, Sep. 2018.

[41] E. Edgington and P. Onghena, *Randomization Tests*. Chapman and Hall CRC, 2007.

[42] D. V. Zaykin, "Optimally weighted z-test is a powerful method for combining probabilities in meta-analysis," *Journal of Evolutionary Biology*, vol. 24, no. 8, pp. 1836–1841, may 2011.

[43] S. Holm, "A simple sequentially rejective multiple test procedure," *Scandinavian Journal of Statistics*, vol. 6, no. 2, pp. 65–70, 1979.

[44] N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy, "Optimal transport for domain adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 9, pp. 1853–1865, sep 2017.

[45] D. A. Engemann, F. Raimondo, J.-R. King, B. Rohaut, G. Louppe, F. Faugeras, J. Annen, H. Cassol, O. Gosseries, D. Fernandez-Slezak, S. Laureys, L. Naccache, S. Dehaene, and J. D. Sitt, "Robust EEG-based cross-site and cross-protocol classification of states of consciousness," *Brain*, vol. 141, no. 11, pp. 3179–3192, Oct. 2018. [Online]. Available: https://doi.org/10.1093/brain/awy251