Sliced-Wasserstein on Symmetric Positive Definite Matrices for M/EEG Signals

Clément Bonet [*1] **Benoît Malézieux** [*2] **Alain Rakotomamonjy** [3 4] **Lucas Drumetz** [5] **Thomas Moreau** [2]
Matthieu Kowalski [6] **Nicolas Courty** [7]

Abstract

When dealing with electro or magnetoencephalography records, many supervised prediction tasks are solved by working with covariance matrices to summarize the signals. Learning with these matrices requires using Riemanian geometry to account for their structure. In this paper, we propose a new method to deal with distributions of covariance matrices and demonstrate its computational efficiency on M/EEG multivariate time series. More specifically, we define a Sliced-Wasserstein distance between measures of symmetric positive definite matrices that comes with strong theoretical guarantees. Then, we take advantage of its properties and kernel methods to apply this distance to brain-age prediction from MEG data and compare it to state-of-the-art algorithms based on Riemannian geometry. Finally, we show that it is an efficient surrogate to the Wasserstein distance in domain adaptation for Brain Computer Interface applications.

1. Introduction

Magnetoencephalography and electroencephalography (M/EEG) are non-invasive techniques for recording the electrical activity of the brain (Hämäläinen et al., 1993). The data consist of multivariate time series output by sensors placed around the head, which capture the intensity of the magnetic or electric field with high temporal resolution. Those measurements provide information on cognitive processes as well as the biological state of a subject.

Successful machine learning (ML) techniques that deal with M/EEG data often rely on covariance matrices estimated from band-passed filtered signals in several frequency bands (Blankertz et al., 2007). The main difficulty that arises when processing such covariance matrices is that the set of symmetric positive definite (SPD) matrices is not a linear space, but a Riemannian manifold (Bhatia, 2009; Bridson & Haefliger, 2013). Therefore, specific algorithms have to be designed to take into account the non Euclidean structure of the data. The usage of Riemannian geometry on SPD matrices has become increasingly popular in the ML community (Huang & Van Gool, 2017; Chevallier et al., 2017; Ilea et al., 2018; Brooks et al., 2019). In particular, these tools have proven to be very effective on prediction tasks with M/EEG data in Brain Computer Interface (BCI) applications (Barachant et al., 2011; 2013; Gaur et al., 2018) or more recently in brain-age prediction (Sabbagh et al., 2019; 2020; Engemann et al., 2022). As covariance matrices sets from M/EEG data are often modeled as samples from a probability distribution – for instance in domain adaptation for BCI (Yair et al., 2019) – it is of great interest to develop efficient tools that work directly on those distributions.

Optimal transport (OT) (Villani, 2009; Peyré et al., 2019) provides a powerful theoretical framework and computational toolbox to compare probability distributions while respecting the geometry of the underlying space. It is well defined on Riemannian manifolds (McCann, 2001; Cui et al., 2019; Alvarez-Melis et al., 2020) and in particular on the space of SPD matrices that is considered in M/EEG learning tasks (Brigant & Puechmorel, 2018; Yair et al., 2019; Ju & Guan, 2022). The original OT problem defines the Wasserstein distance which has a super cubic complexity *w.r.t* samples. To alleviate the computational burden, different alternatives were proposed such as adding an entropic regularization (Cuturi, 2013) or computing the distance between mini-batches (Fatras et al., 2020). Another popular alternative is the Sliced-Wasserstein distance (SW) (Rabin et al., 2011) which computes the average of the Wasserstein distance between one-dimensional projections. SW has recently received a lot of attention as it significantly reduces the computational burden while preserving topological properties of Wasserstein (Bonnotte, 2013; Nadjahi et al., 2020; Bayraktar & Guo, 2021). Moreover, Kolouri et al. (2016); Meunier et al. (2022) have shown that, as opposed to Wasserstein, SW allows to properly extend kernel methods

[*]Equal contribution [1]Université Bretagne Sud, LMBA [2]Université Paris-Saclay, Inria, CEA [3]Criteo AI Lab [4]Université de Rouen, LITIS [5]IMT Atlantique, Lab-STICC [6]Université Paris-Saclay, CNRS, LISN [7]Université Bretagne Sud, IRISA. Correspondence to: Clément Bonet <clement.bonet@univ-ubs.fr>, Benoît Malézieux <benoit.malezieux@inria.fr>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

to data-sets of distributions with very efficient computation of the kernel matrix. This opens the way to new regression and classification methods. However, the initial construction of SW is restricted to Euclidean spaces. Thus, a new line of work focuses on its extension to specific manifolds (Rustamov & Majumdar, 2020; Bonet et al., 2022; 2023).

Contributions. In order to benefit from the advantages of SW in the context of M/EEG, we propose an SW distance on the manifold of SPD matrices and evaluate its efficiency on two prediction tasks.

- We introduce an SW discrepancy between measures of symmetric positive definite matrices (SPDSW), and provide a well-founded numerical approximation.
- We derive theoretical results, including topological, statistical, and computational properties. In particular, we prove that SPDSW is a distance topologically equivalent to the Wasserstein distance in this context.
- We extend the distribution regression with SW kernels to the case of SPD matrices, apply it to brain-age regression with MEG data, and show that it performs better than other methods based on Riemannian geometry.
- We show that SPDSW is an efficient surrogate to the Wasserstein distance in domain adaptation for BCI.

2. Sliced-Wasserstein on SPD matrices

In this section, we introduce an SW discrepancy on SPD matrices and provide a theoretical analysis of its properties and behavior. The proofs are deferred to Appendix C.

2.1. Euclidean Sliced-Wasserstein distance

For $\mu, \nu \in \mathcal{P}_p(\mathbb{R}^d)$ two measures with finite moments of order $p \geq 1$, the Wasserstein distance is defined as

$$W_p^p(\mu,\nu) = \inf_{\gamma \in \Pi(\mu,\nu)} \int \|x-y\|_2^p \,\mathrm{d}\gamma(x,y) \ , \quad (1)$$

where $\Pi(\mu,\nu) = \{\gamma \in \mathcal{P}(\mathbb{R}^d \times \mathbb{R}^d),\ \pi^1_\#\gamma = \mu,\ \pi^2_\#\gamma = \nu\}$ denotes the set of couplings between μ and ν, $\pi^1 : (x,y) \mapsto x$ and $\pi^2 : (x,y) \mapsto y$ the projections on the first and second coordinate and $\#$ is the push-forward operator, defined as a mapping on all borelian $A \subset \mathbb{R}^d$, such that $T_\#\mu(A) = \mu(T^{-1}(A))$. For practical ML applications, this distance is computed between two empirical distributions with n samples and the main bottleneck consists in solving the linear program (1). Its computational complexity is $O(n^3 \log n)$ (Pele & Werman, 2009) which is expensive for large scale applications.

While computing (1) is costly in general, it can be computed efficiently for problems where $d = 1$, as it admits the following closed-form (Peyré et al., 2019, Remark 2.30)

$$W_p^p(\mu,\nu) = \int_0^1 |F_\mu^{-1}(u) - F_\nu^{-1}(u)|^p \,\mathrm{d}u \ , \quad (2)$$

where F_μ^{-1} and F_ν^{-1} are the quantile functions of μ and ν. By computing order statistics, this can be approximated from samples in $O(n \log n)$.

This observation motivated the construction of the SW distance (Rabin et al., 2011; Bonneel et al., 2015) which is defined as the average of the Wasserstein distance between one dimensional projections of the measures in all directions, *i.e.* for $\mu, \nu \in \mathcal{P}_p(\mathbb{R}^d)$,

$$\mathrm{SW}_p^p(\mu,\nu) = \int_{S^{d-1}} W_p^p(t^\theta_\#\mu, t^\theta_\#\nu) \,\mathrm{d}\lambda(\theta) \ , \quad (3)$$

where λ is the uniform distribution on the sphere $S^{d-1} = \{\theta \in \mathbb{R}^d,\ \|\theta\|_2 = 1\}$ and t^θ is the coordinate of the projection on the line $\mathrm{span}(\theta)$, *i.e.* $t^\theta(x) = \langle x, \theta\rangle$ for $x \in \mathbb{R}^d$, $\theta \in S^{d-1}$. This distance has many advantages, motivating its use in place of the Wasserstein distance. First, it can be approximated in $O\big(Ln(d + \log n)\big)$ with L projections and a Monte-Carlo method. Moreover, it is topologically equivalent to the Wasserstein distance as it also metrizes the weak convergence (Nadjahi et al., 2019), and its sample complexity is independent of the dimension (Nadjahi et al., 2020) as opposed to Wasserstein. Finally, it is a Hilbertian metric and it can be used to define kernels over probability distributions (Kolouri et al., 2016; Carriere et al., 2017; Meunier et al., 2022). This is particularly interesting for regression or classification over data-sets of distributions, as we will see in Section 3.1 for brain-age prediction.

2.2. Background on SPD matrices

Let $S_d(\mathbb{R})$ be the set of symmetric matrices of $\mathbb{R}^{d\times d}$, and $S_d^{++}(\mathbb{R})$ be the set of SPD matrices of $\mathbb{R}^{d\times d}$, *i.e.* matrices $M \in S_d(\mathbb{R})$ satisfying

$$\forall x \in \mathbb{R}^d \setminus \{0\},\ x^T M x > 0 \ . \quad (4)$$

$S_d^{++}(\mathbb{R})$ is a Riemannian manifold (Bhatia, 2009), meaning that it behaves locally as a linear space, called a tangent space. Each point $M \in S_d^{++}(\mathbb{R})$ defines a tangent space $\mathcal{T}_M$, which can be given an inner product $\langle\cdot,\cdot\rangle_M : \mathcal{T}_M \times \mathcal{T}_M \to \mathbb{R}$, and thus a norm. The choice of this inner-product induces different geometry on the manifold. One example is the geometric and Affine-Invariant metric (Pennec et al., 2006), where the inner product is defined as

$$\begin{aligned} &\forall M \in S_d^{++}(\mathbb{R}),\ A, B \in \mathcal{T}_M, \\ &\langle A, B\rangle_M = \mathrm{Tr}(M^{-1}AM^{-1}B) \ . \end{aligned} \quad (5)$$

Denoting by Tr the Trace operator, the corresponding geodesic distance $d_{AI}(\cdot,\cdot)$ is given by

$$\forall X, Y \in S_d^{++}(\mathbb{R}), d_{AI}(X,Y) = \sqrt{\mathrm{Tr}(\log(X^{-1}Y)^2)} \ . \tag{6}$$

Another example is the Log-Euclidean metric (Arsigny et al., 2005; 2006) for which,

$$\begin{aligned} &\forall M \in S_d^{++}(\mathbb{R}),\ A, B \in \mathcal{T}_M, \\ &\langle A, B\rangle_M = \langle D_M \log A, D_M \log B\rangle \ , \end{aligned} \tag{7}$$

with log the matrix logarithm and $D_M \log A$ the directional derivative of the log at M along A (Huang et al., 2015). This definition provides another geodesic distance (Arsigny et al., 2006)

$$\forall X, Y \in S_d^{++}(\mathbb{R}),\ d_{LE}(X,Y) = \|\log X - \log Y\|_F \ , \tag{8}$$

which is simply an Euclidean distance in $S_d(\mathbb{R})$ in this case. We will use the Log-Euclidean metric in the following, as it is simpler and faster to compute while being a good first order approximation of the Affine-Invariant metric (Arsigny et al., 2005; Pennec, 2020). In this case, the geodesic between $X, Y \in S_d^{++}(\mathbb{R})$ is $t \in \mathbb{R} \mapsto \exp((1-t)\log X + t \log Y)$. log is a diffeomorphism from $S_d^{++}(\mathbb{R})$ to $S_d(\mathbb{R})$, whose inverse is exp. Thus, the geodesic line going through $A \in S_d(\mathbb{R})$ and the origin of $S_d(\mathbb{R})$ is $\mathcal{G}_A = \{\exp(tA),\ t \in \mathbb{R}\}$. To span all such geodesics, we can restrict to A with unit Frobenius norm, i.e. $\|A\|_F = 1$.

2.3. Construction of SPDSW

On a Euclidean space, the SW distance is defined by averaging the Wasserstein distance between the distributions projected over all possible straight lines passing through the origin. As $S_d^{++}(\mathbb{R})$ with Log-Euclidean metric is a geodesically complete Riemannian manifold, *i.e.* there exists a geodesic curve between each couple of points and each geodesic curve can be extended to $\mathbb{R}$, a natural generalization of SW on this space can be obtained by averaging the Wasserstein distance between distributions projected over all geodesics passing through the origin I_d.

To construct SPDSW, we need several ingredients. First, it is required to find the projection onto a geodesic $\mathcal{G}_A$ passing through I_d where $A \in S_d(\mathbb{R})$. Such projection $P^{\mathcal{G}_A}$ can be obtained as follows

$$\forall M \in S_d^{++}(\mathbb{R}),\ P^{\mathcal{G}_A}(M) = \underset{X \in \mathcal{G}_A}{\arg\min}\ d_{LE}(X, M) \ , \tag{9}$$

and we provide the closed-form in Proposition 2.1.

Proposition 2.1. *Let $A \in S_d(\mathbb{R})$ with $\|A\|_F = 1$, and let $\mathcal{G}_A$ be the associated geodesic line. Then, for any $M \in S_d^{++}(\mathbb{R})$, the geodesic projection on $\mathcal{G}_A$ is*

$$P^{\mathcal{G}_A}(M) = \exp\left(\mathrm{Tr}(A \log M) A\right) \ . \tag{10}$$

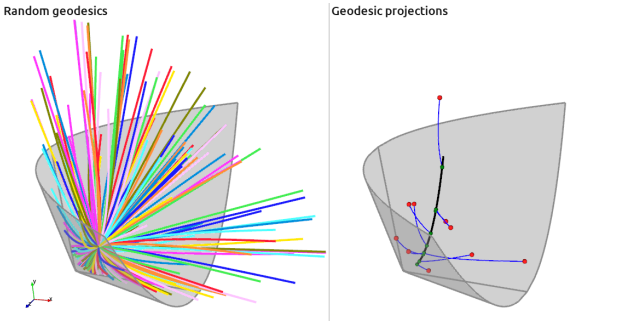


Figure 1: (**Left**) Random geodesics drawn in $S_2^{++}(\mathbb{R})$. (**Right**) Projections (green points) of covariance matrices (depicted as red points) over one geodesic (in black) passing through I_2 along the Log-Euclidean geodesics (blue lines).

Then, the coordinate of the projection on $\mathcal{G}_A$ can be obtained by giving an orientation to $\mathcal{G}_A$ and computing the distance between $P^{\mathcal{G}_A}(M)$ and the origin I_d, as follows

$$t^A(M) = \mathrm{sign}(\langle \log M, A\rangle_F) d_{LE}(P^{\mathcal{G}}(M), I_d) \ . \tag{11}$$

The closed-form expression is given by Proposition 2.2.

Proposition 2.2. *Let $A \in S_d(\mathbb{R})$ with $\|A\|_F = 1$, and let $\mathcal{G}_A$ be the associated geodesic line. Then, for any $M \in S_d^{++}(\mathbb{R})$, the geodesic coordinate on $\mathcal{G}_A$ is*

$$t^A(M) = \langle A, \log M\rangle_F = \mathrm{Tr}(A \log M) \ . \tag{12}$$

These two properties give a closed-form expression for the Riemannian equivalent of one-dimensional projection in a Euclidean space. Note that coordinates on the geodesic might also be found using Busemann coordinates, similarly to the construction proposed by Bonet et al. (2022), and that they actually coincide here. We add more details in Appendix C. In Figure 1, we illustrate the projections of matrices $M \in S_2^{++}(\mathbb{R})$ embedded as vectors $(m_{11}, m_{22}, m_{12}) \in \mathbb{R}^3$. $S_2^{++}(\mathbb{R})$ is an open cone and we plot the projections of random SPD matrices on geodesics passing through I_2.

We are now ready to define an SW discrepancy on measures in $\mathcal{P}_p(S_d^{++}(\mathbb{R})) = \{\mu \in \mathcal{P}(S_d^{++}(\mathbb{R})),\ \int d_{LE}(X, M_0)^p \ \mathrm{d}\mu(X) < \infty,\ M_0 \in S_d^{++}(\mathbb{R})\}$.

Definition 2.3. Let λ_S be the uniform distribution on $\{A \in S_d(\mathbb{R}),\ \|A\|_F = 1\}$. Let $p \geq 1$ and $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$, then the SPDSW discrepancy is defined as

$$\mathrm{SPDSW}_p^p(\mu, \nu) = \int_{S_d(\mathbb{R})} W_p^p(t^A_\# \mu, t^A_\# \nu) \ \mathrm{d}\lambda_S(A) \ . \tag{13}$$

As shown by the definition, being able to sample from λ_S is the cornerstone of the computation of SPDSW. In Lemma 2.4, we propose a practical way of uniformly sampling a symmetric matrix A. More specifically, we sample

an orthogonal matrix P and a diagonal matrix D of unit norm and compute $A = PDP^T$ which is a symmetric matrix of unit norm. This is equivalent to sampling from λ_S as the measures are equal up to a normalization factor $d!$ which represents the number of possible permutations of the columns of P and D for which $PDP^T = A$.

Lemma 2.4. *Let λ_O be the uniform distribution on $\mathcal{O}_d = \{P \in \mathbb{R}^{d\times d}, P^TP = PP^T = I\}$ (Haar distribution), and λ be the uniform distribution on $S^{d-1} = \{\theta \in \mathbb{R}^d, \|\theta\|_2 = 1\}$. Then $\lambda_S \in \mathcal{P}(S_d(\mathbb{R}))$, defined such that $\forall\ A = P\mathrm{diag}(\theta)P^T \in S_d(\mathbb{R})$, $\mathrm{d}\lambda_S(A) = d!\ \mathrm{d}\lambda_O(P)\mathrm{d}\lambda(\theta)$, is the uniform distribution on $\{A \in S_d(\mathbb{R}),\ \|A\|_F = 1\}$.*

Then, the coordinate of the projection on the geodesic $\mathcal{G}_A$ is provided by $t^A(\cdot) = \mathrm{Tr}(A \log \cdot)$ defined in Proposition 2.2. The Wasserstein distance is easily computed using order statistics, and this leads to a natural extension of the SW distance in $S_d^{++}(\mathbb{R})$. There exists a strong link between SW on distributions in $\mathbb{R}^{d\times d}$ and SPDSW. Indeed, Proposition 2.5 shows that SPDSW is equal to a variant of SW where projection parameters are sampled from unit norm matrices in $S_d(\mathbb{R})$ instead of the unit sphere, and where the distributions are pushed forward by the log operator.

Proposition 2.5. *Let $\tilde{\mu}, \tilde{\nu} \in \mathcal{P}_p(S_d(\mathbb{R}))$, and $\tilde{t}^A(B) = \mathrm{Tr}(A^TB)$ for $A, B \in S_d(\mathbb{R})$. We define*

$$\mathrm{SymSW}_p^p(\tilde{\mu}, \tilde{\nu}) = \int_{S_d(\mathbb{R})} W_p^p(\tilde{t}^A_\#\tilde{\mu}, \tilde{t}^A_\#\tilde{\nu})\ \mathrm{d}\lambda_S(A)\ . \quad (14)$$

Then, for $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$,

$$\mathrm{SPDSW}_p^p(\mu, \nu) = \mathrm{SymSW}_p^p(\log_\#\mu, \log_\#\nu)\ . \quad (15)$$

Thus, it seems natural to compare the results obtained with SPDSW to the Euclidean counterpart $\log\mathrm{SW} = \mathrm{SW}(\log_\#\cdot, \log_\#\cdot)$ where the distributions are made of projections in the log space and where the sampling is done with the uniform distribution on the sphere. The Wasserstein distance is also well defined on Riemannian manifolds, and in particular on the space of SPD matrices. Denoting d a geodesic distance on $S_d^{++}(\mathbb{R})$, we can define the corresponding Wasserstein distance between $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$ as

$$W_p^p(\mu, \nu) = \inf_{\gamma\in\Pi(\mu,\nu)} \int d(X, Y)^p\ \mathrm{d}\gamma(X, Y)\ . \quad (16)$$

In the following, we study properties of SPDSW and in particular, we show that it is a computationally efficient alternative to Wasserstein on $\mathcal{P}(S_d^{++}(\mathbb{R}))$ as it is topologically equivalent while having a better computational complexity and being better conditioned for regression of distributions.

2.4. Properties of SPDSW

We now derive theoretical properties of SPDSW.

Topology. Following usual arguments which are valid for any sliced divergence with any projection, we can show that SPDSW is a pseudo-distance. Here, $S_d^{++}(\mathbb{R})$ with the Log-Euclidean metric is of null sectional curvature (Arsigny et al., 2005; Xu, 2022) and we have access to a diffeomorphism to a Euclidean space – the log operator. This allows us to show that SPDSW is a distance in Theorem 2.6.

Theorem 2.6. *Let $p \geq 1$, then SPDSW_p is a finite distance on $\mathcal{P}_p(S_d^{++}(\mathbb{R}))$.*

In the case of the Affine-Invariant metric, the Riemannian manifold endowed with this metric has a non-positive and non-constant sectional curvature, and closed-forms of geodesics projections are not known to the best of our knowledge. We can however derive Busemann coordinates, which involve a costly additional projection. Moreover, whether or not it satisfies the indiscernible property remains an open question. Hence, we focus on SPDSW with Log-Euclidean metric and discuss the use of the Affine-Invariant metric in Appendix D.

An important property which justifies the use of the SW distance in place of the Wasserstein distance in the Euclidean case is that they both metrize the weak convergence (Bonnotte, 2013). We show in Theorem 2.7 that this is also the case with SPDSW in $\mathcal{P}_p(S_d^{++}(\mathbb{R}))$.

Theorem 2.7. *For $p \geq 1$, SPDSW_p metrizes the weak convergence,* i.e. *for $\mu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$ and a sequence $(\mu_k)_k$ in $\mathcal{P}_p(S_d^{++}(\mathbb{R}))$, $\lim_{k\to\infty}\mathrm{SPDSW}_p(\mu_k, \mu) = 0$ if and only if $(\mu_k)_k$ converges weakly to μ.*

Moreover, SPDSW_p and W_p – the p-Wasserstein distance with Log-Euclidean ground cost – are also weakly equivalent on compactly supported measures on $\mathcal{P}_p(S_d^{++}(\mathbb{R}))$, as demonstrated in Theorem 2.8.

Theorem 2.8. *Let $p \geq 1$, let $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$. Then*

$$\mathrm{SPDSW}_p^p(\mu, \nu) \leq c_{d,p}^p W_p^p(\mu, \nu)\ , \quad (17)$$

where $c_{d,p}^p = \frac{1}{d}\int \|\theta\|_p^p\ \mathrm{d}\lambda(\theta)$. Let $R > 0$ and $B(I, R) = \{A \in S_d^{++}(\mathbb{R}),\ d_{LE}(A, I_d) = \|\log A\|_F \leq R\}$ be a closed ball. Then there exists a constant $C_{d,p,R}$ such that for all $\mu, \nu \in \mathcal{P}_p(B(I, R))$,

$$W_p^p(\mu, \nu) \leq C_{d,p,R}\mathrm{SPDSW}_p(\mu, \nu)^{\frac{2}{d(d+1)+2}}\ . \quad (18)$$

The theorems above highlight that SPDSW_p behaves similarly to W_p on $\mathcal{P}_p(S_d^{++}(\mathbb{R}))$. Thus, it is justified to use SPDSW_p as a surrogate of Wasserstein and take advantage of the statistical and computational benefits that we present now.

Statistical properties. In practice, we approximate SPDSW using the plug-in estimator (Niles-Weed & Rigollet, 2022; Manole et al., 2022), *i.e.* for $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$,

we approximate $\text{SPDSW}_p^p(\mu,\nu)$ by $\text{SPDSW}_p^p(\hat{\mu}_n,\hat{\nu}_n)$ where $\hat{\mu}_n$ and $\hat{\nu}_n$ denote empirical distributions of μ and ν. Hence, we are interested in the speed of convergence towards $\text{SPDSW}_p^p(\mu,\nu)$, which we call the sample complexity. We derive the convergence rate for SPDSW in Proposition 2.9, relying on the proof of Nadjahi et al. (2020) and on the sample complexity of the Wasserstein distance (Fournier & Guillin, 2015). The sample complexity we find does not depend on the dimension, which is an important property of sliced divergences (Nadjahi et al., 2020).

Proposition 2.9. *Let $q > p \geq 1$, $\mu,\nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$, and $\hat{\mu}_n,\hat{\nu}_n$ the associated empirical measures. We define the moment of order q by $M_q(\mu) = \int \|X\|_F^q \, \mathrm{d}\mu(X)$, and $M_q(\mu,\nu) = M_q(\log_{\#}\mu)^{1/q} + M_q(\log_{\#}\nu)^{1/q}$. Then, there exists a constant $C_{p,q}$ depending only on p and q such that*

$$\mathbb{E}\big[|\text{SPDSW}_p(\hat{\mu}_n,\hat{\nu}_n) - \text{SPDSW}_p(\mu,\nu)|\big] \leq \alpha_{n,p,q} C_{p,q}^{1/p} M_q(\mu,\nu) \ , \tag{19}$$

$$\textit{where} \quad \alpha_{n,p,q} = \begin{cases} n^{-1/(2p)} & \textit{if } q > 2p \\ n^{-1/(2p)}\log(n)^{1/p} & \textit{if } q = 2p \\ n^{-(q-p)/(pq)} & \textit{if } q \in (p,2p) \end{cases} .$$

Proposition 2.9 assumes we can exactly compute the outer integral, which is not the case in practice, as it requires a Monte-Carlo approximation. In Proposition 2.10, we show that, L being the number of projections, the Monte-Carlo error is $O(\frac{1}{\sqrt{L}})$ for a fixed dimension d. This time, the dimension intervenes in $\text{Var}_{A\sim\lambda_S}\left[W_p^p(t_{\#}^A\mu, t_{\#}^A\nu)\right]$.

Proposition 2.10. *Let $p \geq 1$, $\mu,\nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$. Then, the error made by the Monte Carlo estimate of SPDSW_p with L projections can be bounded as follows*

$$\mathbb{E}_A\left[|\widehat{\text{SPDSW}}_{p,L}^p(\mu,\nu) - \text{SPDSW}_p^p(\mu,\nu)|\right]^2 \leq \frac{1}{L}\text{Var}_{A\sim\lambda_S}\left[W_p^p(t_{\#}^A\mu, t_{\#}^A\nu)\right] \ , \tag{20}$$

where $\widehat{\text{SPDSW}}_{p,L}^p(\mu,\nu) = \frac{1}{L}\sum_{i=1}^L W_p^p(t_{\#}^{A_i}\mu, t_{\#}^{A_i}\nu)$ with $(A_i)_{i=1}^L$ independent samples from λ_S.

Computational complexity and implementation. Let $\mu,\nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$ and $(X_i)_{i=1}^n$ (resp. $(Y_j)_{j=1}^m$) samples from μ (resp. from ν). We approximate $\text{SPDSW}_p^p(\mu,\nu)$ by $\widehat{\text{SPDSW}}_{p,L}^p(\hat{\mu}_n,\hat{\nu}_m)$ where $\hat{\mu}_n = \frac{1}{n}\sum_{i=1}^n \delta_{X_i}$ and $\hat{\nu}_m = \frac{1}{m}\sum_{j=1}^m \delta_{Y_j}$. Sampling from λ_O requires drawing a matrix $Z \in \mathbb{R}^{d\times d}$ with i.i.d normally distributed coefficients, and then taking the QR factorization with positive entries on the diagonal of R (Mezzadri, 2006), which needs $O(d^3)$ operations (Golub & Van Loan, 2013, Section 5.2). Then, computing n matrix logarithms takes $O(nd^3)$ operations. Given L projections, the inner-products require $O(Lnd^2)$ operations, and the computation of the one-dimensional Wasserstein distances is done in $O(Ln\log n)$ operations. Therefore, the complexity of SPDSW is $O(Ln(\log n + d^2) + (L+n)d^3)$. The procedure is detailed in Algorithm 1. In practice, when it is required to call SPDSW several times in optimization procedures, the computational complexity can be reduced by drawing projections only once at the beginning.

Algorithm 1 Computation of SPDSW

Input: $(X_i)_{i=1}^n \sim \mu$, $(Y_j)_{j=1}^m \sim \nu$, L the number of projections, p the order
for $\ell = 1$ **to** L **do**
  Draw $\theta \sim \text{Unif}(S^{d-1}) = \lambda$
  Draw $P \sim \text{Unif}(O_d(\mathbb{R})) = \lambda_O$
  $A = P\text{diag}(\theta)P^T$
  $\forall i,j,\ \hat{X}_i^\ell = t^A(X_i), \hat{Y}_j^\ell = t^A(Y_j)$
  Compute $W_p^p(\frac{1}{n}\sum_{i=1}^n \delta_{\hat{X}_i^\ell}, \frac{1}{m}\sum_{j=1}^m \delta_{\hat{Y}_j^\ell})$
end for
Return $\frac{1}{L}\sum_{\ell=1}^L W_p^p(\frac{1}{n}\sum_{i=1}^n \delta_{\hat{X}_i^\ell}, \frac{1}{m}\sum_{j=1}^m \delta_{\hat{Y}_j^\ell})$

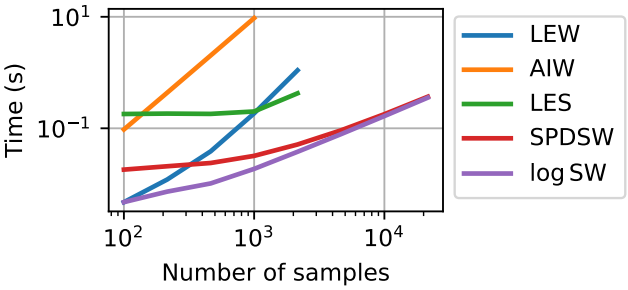


Figure 2: Runtime in log-log scale of SPDSW and log SW (200 proj., d=20) compared to alternatives based on Wasserstein between Wishart samples. Sliced discrepancies can scale to larger distributions in $S_d^{++}(\mathbb{R})$.

Note that it is possible to draw symmetric matrices with complexity $O(d^2)$ by taking $A = \frac{Z+Z^T}{\|Z+Z^T\|_F}$. Although this is a great advantage from the point of view of computation time, we leave it as an open question to know whether this breaks the bounds in Theorem 2.8.

We illustrate the computational complexity *w.r.t* samples in Figure 2. The computations have been performed on a GPU NVIDIA Tesla V100-DGXS 32GB using `PyTorch` (Paszke et al., 2017)[1]. We compare the runtime to the Wasserstein distance with Affine-Invariant (AIW) and Log-Euclidean (LEW) metrics, and to Sinkhorn algorithm (LES) which is a classical alternative to Wasserstein to reduce the computational cost. When enough samples are available, then computing the Wasserstein distance takes more time than computing the cost matrix, and SPDSW is fast to compute.

[1]Code is available at https://github.com/clbonet/SPDSW.

3. From Brain Data to Distributions in $S_d^{++}(\mathbb{R})$

M/EEG data consists of multivariate time series $X \in \mathbb{R}^{N_C \times T}$, with N_C channels, and T time samples. A widely adopted model assumes that the measurements X are linear combinations of N_S sources $S \in \mathbb{R}^{N_S \times T}$ degraded by noise $N \in \mathbb{R}^{N_C \times T}$. This leads to $X = AS + N$, where $A \in \mathbb{R}^{N_C \times N_S}$ is the forward linear operator (Hämäläinen et al., 1993). A common practice in statistical learning on M/EEG data is to consider that the target is a function of the power of the sources, *i.e.* $\mathbb{E}[SS^T]$ (Blankertz et al., 2007; Dähne et al., 2014; Sabbagh et al., 2019). In particular, a broad range of methods rely on second-order statistics of the measurements, *i.e.* covariance matrices of the form $C = \frac{XX^T}{T}$, which are less costly and uncertain than solving the inverse problem to recover S before training the model. After proper rank reduction to turn the covariance estimates into SPD matrices (Harandi et al., 2017), and appropriate band-pass filtering to stick to specific physiological patterns (Blankertz et al., 2007), Riemannian geometry becomes an appropriate tool to deal with such data.

In this section, we propose two applications of SPDSW to prediction tasks from M/EEG data. More specifically, we introduce a new method to perform brain-age regression, building on the work of Sabbagh et al. (2019) and Meunier et al. (2022), and another for domain adaptation in BCI.

3.1. Distributions Regression for Brain-age Prediction

Learning to predict brain age from population-level neuroimaging data-sets can help characterize biological aging and disease severity (Spiegelhalter, 2016; Cole & Franke, 2017; Cole et al., 2018). Thus, this task has encountered more and more interest in the neuroscience community in recent years (Xifra-Porxas et al., 2021; Peng et al., 2021; Engemann et al., 2022). In particular, Sabbagh et al. (2019) take advantage of Riemannian geometry for feature engineering and prediction with the following steps. First, one covariance estimate is computed per frequency band from each subject recording. Then these covariance matrices are projected onto a lower dimensional space to make them full rank, for instance with a PCA. Each newly obtained SPD matrix is projected onto the log space to obtain a feature after vectorization and aggregation among frequency bands. Finally, a Ridge regression model predicts brain age. This white-box method achieves state-of-the-art brain age prediction scores on MEG datasets like Cam-CAN (Taylor et al., 2017).

MEG recordings as distributions of covariance matrices. Instead of modeling each frequency band by a unique covariance matrix, we propose to use a distribution of covariance matrices estimated from small time frames. Concretely, given a time series $X \in \mathbb{R}^{N_C \times T}$ and a time-frame length $t < T$, a covariance matrix is estimated from each one of the $n = \lfloor \frac{T}{t} \rfloor$ chunks of signal available. This process models each subject by as many empirical distributions of covariance estimates $(C_i)_{i=1}^n$ as there are frequency bands. Then, all samples are projected on a lower dimensional space with a PCA, as done in Sabbagh et al. (2019). Here, we study whether modeling a subject by such distributions provides additional information compared to feature engineering based on a unique covariance matrix. In order to perform brain age prediction from these distributions, we extend recent results on distribution regression with SW kernels (Kolouri et al., 2016; Meunier et al., 2022) to SPD matrices, and show that SPDSW performs well on this prediction task while being easy to implement.

SPDSW **kernels for distributions regression.** As shown in Section 2.4, SPDSW is a well-defined distance on distributions in $S_d^{++}(\mathbb{R})$. The most straightforward way to build a kernel from this distance is to resort to well-known Gaussian kernels, *i.e.* $K(\mu, \nu) = e^{-\frac{1}{2\sigma^2}\mathrm{SPDSW}_2^2(\mu,\nu)}$.

However, this is not sufficient to make it a proper positive kernel. Indeed, we need SPDSW to be a Hilbertian distance (Hein & Bousquet, 2005). A pseudo-distance d on $\mathcal{X}$ is Hilbertian if there exists a Hilbert space $\mathcal{H}$ and a feature map $\Phi : \mathcal{X} \to \mathcal{H}$ such that $\forall x, y \in \mathcal{X}, d(x, y) = \|\Phi(x) - \Phi(y)\|_{\mathcal{H}}$. We now extend Meunier et al. (2022, Proposition 5) to the case of SPDSW in Proposition 3.1.

Proposition 3.1. *Let m be the Lebesgue measure and let $\mathcal{H} = L^2([0,1] \times S_d(\mathbb{R}), m \otimes \lambda_S)$. We define Φ as*

$$\begin{aligned} \Phi : \mathcal{P}_2(S_d^{++}(\mathbb{R})) &\to \mathcal{H} \\ \mu &\mapsto \left((q, A) \mapsto F^{-1}_{t^A_\# \mu}(q)\right) \ , \end{aligned} \tag{21}$$

where $F^{-1}_{t^A_\# \mu}$ is the quantile function of $t^A_\# \mu$. Then, SPDSW_2 *is Hilbertian and for all $\mu, \nu \in \mathcal{P}_2(S_d^{++}(\mathbb{R}))$,*

$$\mathrm{SPDSW}_2^2(\mu, \nu) = \|\Phi(\mu) - \Phi(\nu)\|_{\mathcal{H}}^2 \ . \tag{22}$$

The proof is similar to the one of Meunier et al. (2022) for SW in Euclidean spaces and highlights two key results. The first one is that SPDSW extensions of Gaussian kernels are valid positive definite kernels, as opposed to what we would get with the Wasserstein distance (Meunier et al., 2022). The second one is that we have access to an explicit and easy-to-compute feature map that preserves SPDSW, making it possible to avoid inefficient quadratic algorithms on empirical distributions from very large data. In practice, we rely on the finite-dimensional approximation of projected distributions quantile functions proposed in Meunier et al. (2022) to compute the kernels more efficiently with the ℓ_2-norm. Then, we leverage Kernel Ridge regression for prediction (Murphy, 2012). Let $0 < q_1 < \cdots < q_M < 1$, and $(A_1, \ldots, A_L) \in S_d(\mathbb{R})^L$. The approximate feature

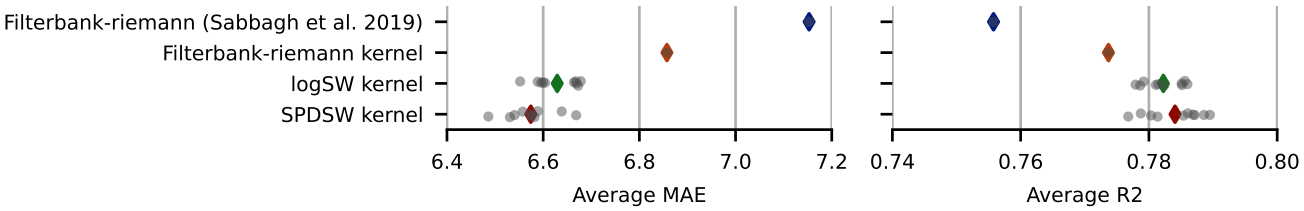


Figure 3: Average MAE and R^2 score for 10 random seeds on the Cam-CAN data-set with time-frames of 2s and 1000 projections. Kernel Ridge regression based on SW kernels performs best. SPDSW and log SW are close to each other. Sampling from symmetric matrices offers a slight advantage but does not play a key role on performance. For information, Euclidean SW led to poor results on the task (MAE 9.7).

map has a closed-form expression in the case of empirical distributions and is defined as

$$\hat{\Phi}(\mu) = \left(\frac{1}{\sqrt{ML}} F^{-1}_{t^{A_i}_{\#}\mu}(q_j) \right)_{1 \leq j \leq M, 1 \leq i \leq L} . \tag{23}$$

Regarding brain-age prediction, we model each couple of subject s and frequency band f as an empirical distribution $\mu_n^{s,f}$ of covariance estimates $(C_i)_{i=1}^n$. Hence, our data-set consists of the set of distributions in $S_d^{++}(\mathbb{R})$

$$\left(\mu_n^{s,f} = \frac{1}{n} \sum_{i=1}^{n} \delta_{C_i} \right)_{s,f} . \tag{24}$$

First, we compute the associated features $(\hat{\Phi}(\mu_n^{s,f}))_{s,f}$ by loading the data and band-pass filtering the signal once per subject. Then, as we are interested in comparing each subject in specific frequency bands, we compute one approximate kernel matrix per frequency f, as follows

$$K^f_{i,j} = e^{-\frac{1}{2\sigma^2} \|\hat{\Phi}(\mu_n^{i,f}) - \hat{\Phi}(\mu_n^{j,f})\|_2^2} . \tag{25}$$

Finally, the kernel matrix obtained as a sum over frequency bands, *i.e.* $K = \sum_f K^f$, is plugged into the Kernel Ridge regression of `scikit-learn` (Pedregosa et al., 2011).

Numerical results We demonstrate the ability of our algorithm to perform well on brain-age prediction on the largest publicly available MEG data-set Cam-CAN (Taylor et al., 2017), which contains recordings from 646 subjects at rest. We take advantage of the benchmark provided by Engemann et al. (2022) – available online[2] and described in Appendix B.2 – to replicate the same pre-processing and prediction steps from the data, and thus produce a meaningful and fair comparison.

For each one of the seven frequency bands, we divide every subject time series into frames of fixed length. We estimate covariance matrices from each timeframe with OAS (Chen et al., 2010) and apply PCA for rank-reduction, as in Sabbagh et al. (2019), to obtain SPD matrices of size 53×53.

[2] https://github.com/meeg-ml-benchmarks/brain-age-benchmark-paper

This leads to distributions of 275 points per subject and per frequency band. In Sabbagh et al. (2019), the authors rely on Ridge regression on vectorized projections of SPD matrices on the tangent space. We also provide a comparison to Kernel Ridge regression based on a kernel with the Log-Euclidean metric, *i.e.* $K^{\log}_{i,j} = e^{-\frac{1}{2\sigma^2} \| \log C_i - \log C_j \|_F^2}$.

Figure 3 shows that SPDSW and log SW (1000 projections, time-frames of 2s) perform best in average on 10-folds cross-validation for 10 random seeds, compared to the baseline with Ridge regression (Sabbagh et al., 2019) and to Kernel Ridge regression based on the Log-Euclidean metric, with identical pre-processing. We provide more details on scores for each fold on a single random seed in Figure A. In particular, it seems that evaluating the distance between distributions of covariance estimates instead of just the average covariance brings more information to the model in this brain-age prediction task, and allows to improve the score. Moreover, while SPDSW gives the best results, logSW actually performs well compared to baseline methods. Thus, both methods seem to be usable in practice, even though sampling symmetric matrices and taking into account the Riemannian geometry improves the performances compared to logSW. Also note that Log-Euclidean Kernel Ridge regression works better than the baseline method based on Ridge regression (Sabbagh et al., 2019). Then, Figure B in the appendix shows that SPDSW does not suffer from variance with more than 500 projections in this use case with matrices of size 53×53. Finally, Figure C shows that there is a trade-off to find between smaller time-frames for more samples per distribution and larger time-frames for less noise in the covariance estimates and that this is an important hyper-parameter of the model.

3.2. Domain Adaptation for BCI

BCI consists of establishing a communication interface between the brain and an external device, in order to assist or repair sensory-motor functions (Daly & Wolpaw, 2008; Nicolas-Alonso & Gomez-Gil, 2012; Wolpaw, 2013). The interface should be able to correctly interpret M/EEG signals and link them to actions that the subject would like to perform. One challenge of BCI is that ML methods are

Table 1: Accuracy and Runtime for Cross Session.

Subjects	Source	AISOTDA (Yair et al., 2019)	SPDSW	LogSW	LEW	LES	SPDSW	LogSW	LEW	LES
			Transformations in $S_d^{++}(\mathbb{R})$				Descent over particles			
1	82.21	80.90	84.70	84.48	84.34	84.70	85.20	85.20	77.94	82.92
3	79.85	87.86	85.57	84.10	85.71	86.08	87.11	86.37	82.42	81.47
7	72.20	82.29	81.01	76.32	81.23	81.23	81.81	81.73	79.06	73.29
8	79.34	83.25	83.54	81.03	82.29	83.03	84.13	83.32	80.07	85.02
9	75.76	80.25	77.35	77.88	77.65	77.65	80.30	79.02	76.14	70.45
Avg. acc.	77.87	82.93	82.43	80.76	82.24	82.54	83.71	83.12	79.13	78.63
Avg. time (s)	-	-	**4.34**	**4.32**	11.41	12.04	**3.68**	**3.67**	8.50	11.43

generally not robust to the change of data domain, which means that an algorithm trained on a particular subject will not be able to generalize to other subjects. Domain adaptation (DA) (Ben-David et al., 2006) offers a solution to this problem by taking into account the distributional shift between source and target domains. Classical DA techniques employed in BCI involve projecting target data on source data or vice versa, or learning a common embedding that erases the shift, sometimes with the help of optimal transport (Courty et al., 2016). As Riemannian geometry works well on BCI (Barachant et al., 2013; Yger et al., 2016), DA tools have been developed for SPD matrices (Yair et al., 2019; Ju & Guan, 2022).

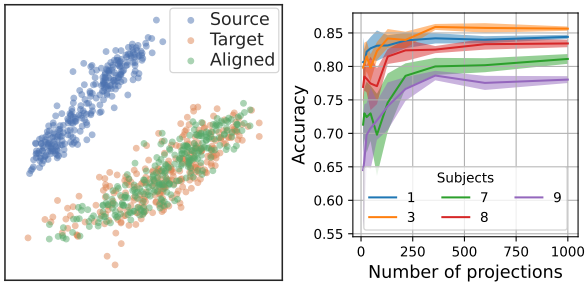


Figure 4: (**Left**) PCA on BCI data before and after alignment. Minimizing SPDSW with enough projections allows aligning sources on targets. (**Right**) Accuracy *w.r.t* num. of projections for the cross-session task with transformations. Here, there is no need for too many projections to converge.

SPDSW **for domain adaptation on SPD matrices.** We study two training frameworks on data from $\mathcal{P}(S_d^{++}(\mathbb{R}))$. In the first case, a push forward operator f_θ is trained to change a distribution μ_S in the source domain into a distribution μ_T in the target domain by minimizing a loss of the form $L(\theta) = \mathcal{L}\big((f_\theta)_\# \mu_S, \mu_T\big)$, where $\mathcal{L}$ is a transport cost like Wasserstein on $\mathcal{P}(S_d^{++}(\mathbb{R}))$ or SPDSW. The model f_θ is a sequence of simple transformations in $S_d^{++}(\mathbb{R})$ (Rodrigues et al., 2018), *i.e.* $T_W(C) = W^T C W$ for $W \in S_d^{++}(\mathbb{R})$ (translations) or $W \in \mathrm{SO}_d$ (rotations), potentially combined to specific non-linearities (Huang & Van Gool, 2017). The advantage of such models is that they provide a high level of structure with a small number of parameters.

In the second case, we directly align the source on the target by minimizing $\mathcal{L}$ with a Riemannian gradient descent directly over the particles (Boumal, 2020), *i.e.* by denoting $\mu_S((x_i)_{i=1}^{|X_S|}) = \frac{1}{|X_S|}\sum_{i=1}^{|X_S|} \delta_{x_i}$ with $X_S = \{x_i^S\}_i$ the samples of the source, we initialize at $(x_i^S)_{i=1}^{|X_S|}$ and minimize $L((x_i)_{i=1}^{|X_S|}) = \mathcal{L}\left(\mu_S((x_i)_{i=1}^{|X_S|}), \mu_T\right)$.

We use `Geoopt` (Kochurov et al., 2020) and `Pytorch` (Paszke et al., 2017) to optimize on manifolds. Then, an SVM is trained on the vectorized projections of X_S in the log space, *i.e.* from couples $(\mathrm{vect}(\log x_i^S), y_i)_{i=1}^{|X_S|}$, and we evaluate the model on the target distribution.

Numerical results. In Table 1, we focus on cross-session classification for the BCI IV 2.a Competition dataset (Brunner et al., 2008) with 4 target classes and about 270 samples per subject and session. We compare accuracies and runtimes for several methods run on a GPU Tesla V100-DGXS-32GB. The distributions are aligned by minimizing different discrepancies, namely SPDSW, logSW, Log-Euclidean Wasserstein (LEW) and Sinkhorn (LES), computed with `POT` (Flamary et al., 2021). Note that we did not tune hyperparameters on each particular subject and discrepancy, but only used a grid search to train the SVM on the source dataset, and optimized each loss until convergence, *i.e.* without early stopping. We compare this approach to the naive one without DA, and to the barycentric OTDA (Courty et al., 2016) with Affine-Invariant metric reported from Yair et al. (2019). We provide further comparisons on cross-subject in Appendix A.2. Our results show that all discrepancies give equivalent accuracies. As expected, SPDSW has an advantage in terms of computation time compared to other transport losses. Moreover, transformations in $S_d^{++}(\mathbb{R})$ and descent over the particles work almost equally well in the case of SPDSW. We illustrate the alignment we obtain by minimizing SPDSW in Figure 4, with a PCA for visualization purposes. Additionally, Figure 4 shows that SPDSW does not need too many projections to reach optimal performance. We provide more experimental details in Appendix B.

4. Conclusion

We proposed SPDSW, a discrepancy between distributions of SPD matrices with appealing properties such as being a distance and metrizing the weak convergence. Being a Hilbertian metric, it can be plugged as is into Kernel methods, as we demonstrate for brain age prediction from MEG data. Moreover, it is usable in loss functions dealing with distributions of SPD matrices, for instance in domain adaptation for BCI, with less computational complexity than its counterparts. Beyond M/EEG data, our discrepancy is of interest for any learning problem that involves distributions of SPD matrices, and we expect to see other applications of SPDSW in the future. One might also be interested in using other metrics on positive definite or semi-definite matrices such as the Bures-Wasserstein metric, with the additional challenges that this space is positively curved and not geodesically complete (Thanwerdas & Pennec, 2023).

Acknowledgements

Clément Bonet, Nicolas Courty and Lucas Drumetz contributions were supported by project DynaLearn from Labex CominLabs and Région Bretagne ARED DLearnMe. Nicolas Courty is partially funded by the project OTTOPIA ANR-20-CHIA-0030 of the French National Research Agency (ANR). Benoît Malézieux contributions were supported by grants from Digiteo France.

References

Alvarez-Melis, D., Mroueh, Y., and Jaakkola, T. Unsupervised hierarchy matching with optimal transport over hyperbolic spaces. In *International Conference on Artificial Intelligence and Statistics*, pp. 1606–1617. PMLR, 2020.

Arsigny, V., Fillard, P., Pennec, X., and Ayache, N. *Fast and Simple Computations on Tensors with Log-Euclidean Metrics.* PhD thesis, INRIA, 2005.

Arsigny, V., Fillard, P., Pennec, X., and Ayache, N. Log-euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 56(2):411–421, 2006.

Barachant, A., Bonnet, S., Congedo, M., and Jutten, C. Multiclass brain–computer interface classification by riemannian geometry. *IEEE Transactions on Biomedical Engineering*, 59(4):920–928, 2011.

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

Bayraktar, E. and Guo, G. Strong equivalence between metrics of wasserstein type. *Electronic Communications in Probability*, 26:1–13, 2021.

Ben-David, S., Blitzer, J., Crammer, K., and Pereira, F. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006.

Bhatia, R. Positive definite matrices. In *Positive Definite Matrices*. Princeton university press, 2009.

Blankertz, B., Tomioka, R., Lemm, S., Kawanabe, M., and Muller, K.-R. Optimizing spatial filters for robust eeg single-trial analysis. *IEEE Signal processing magazine*, 25(1):41–56, 2007.

Bogachev, V. I. and Ruas, M. A. S. *Measure theory*, volume 1. Springer, 2007.

Bonet, C., Chapel, L., Drumetz, L., and Courty, N. Hyperbolic sliced-wasserstein via geodesic and horospherical projections. *arXiv preprint arXiv:2211.10066*, 2022.

Bonet, C., Berg, P., Courty, N., Septier, F., Drumetz, L., and Pham, M.-T. Spherical sliced-wasserstein. In *International Conference on Learning Representations*, 2023.

Bonneel, N., Rabin, J., Peyré, G., and Pfister, H. Sliced and radon wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51(1):22–45, 2015.

Bonnotte, N. *Unidimensional and evolution methods for optimal transportation*. PhD thesis, Paris 11, 2013.

Boumal, N. An introduction to optimization on smooth manifolds. *Available online, May*, 3, 2020.

Bridson, M. R. and Haefliger, A. *Metric spaces of non-positive curvature*, volume 319. Springer Science & Business Media, 2013.

Brigant, A. L. and Puechmorel, S. Optimal riemannian quantization with an application to air traffic analysis. *arXiv preprint arXiv:1806.07605*, 2018.

Brooks, D., Schwander, O., Barbaresco, F., Schneider, J.-Y., and Cord, M. Riemannian batch normalization for spd neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Brunner, C., Leeb, R., Müller-Putz, G., Schlögl, A., and Pfurtscheller, G. Bci competition 2008–graz data set a. *Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology*, 16: 1–6, 2008.

Carriere, M., Cuturi, M., and Oudot, S. Sliced wasserstein kernel for persistence diagrams. In *International conference on machine learning*, pp. 664–673. PMLR, 2017.

Chami, I., Gu, A., Nguyen, D. P., and Ré, C. Horopca: Hyperbolic dimensionality reduction via horospherical projections. In *International Conference on Machine Learning*, pp. 1419–1429. PMLR, 2021.

Chen, Y., Wiesel, A., Eldar, Y. C., and Hero, A. O. Shrinkage algorithms for mmse covariance estimation. *IEEE Transactions on Signal Processing*, 58(10):5016–5029, 2010.

Chevallier, E., Kalunga, E., and Angulo, J. Kernel density estimation on spaces of gaussian distributions and symmetric positive definite matrices. *SIAM Journal on Imaging Sciences*, 10(1):191–215, 2017.

Cole, J. H. and Franke, K. Predicting age using neuroimaging: innovative brain ageing biomarkers. *Trends in neurosciences*, 40(12):681–690, 2017.

Cole, J. H., Ritchie, S. J., Bastin, M. E., Hernández, V., Muñoz Maniega, S., Royle, N., Corley, J., Pattie, A., Harris, S. E., Zhang, Q., et al. Brain age predicts mortality. *Molecular psychiatry*, 23(5):1385–1392, 2018.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9): 1853–1865, 2016.

Cui, L., Qi, X., Wen, C., Lei, N., Li, X., Zhang, M., and Gu, X. Spherical optimal transportation. *Computer-Aided Design*, 115:181–193, 2019.

Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

Dähne, S., Meinecke, F. C., Haufe, S., Höhne, J., Tangermann, M., Müller, K.-R., and Nikulin, V. V. Spoc: a novel framework for relating the amplitude of neuronal oscillations to behaviorally relevant parameters. *NeuroImage*, 86:111–122, 2014.

Daly, J. J. and Wolpaw, J. R. Brain–computer interfaces in neurological rehabilitation. *The Lancet Neurology*, 7(11): 1032–1043, 2008.

Engemann, D. A., Mellot, A., Höchenberger, R., Banville, H., Sabbagh, D., Gemein, L., Ball, T., and Gramfort, A. A reusable benchmark of brain-age prediction from m/eeg resting-state signals. *Neuroimage*, 262:119521, 2022.

Fatras, K., Zine, Y., Flamary, R., Gribonval, R., and Courty, N. Learning with minibatch wasserstein : asymptotic and gradient properties. In Chiappa, S. and Calandra, R. (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2131–2141. PMLR, 26–28 Aug 2020.

Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., et al. Pot: Python optimal transport. *J. Mach. Learn. Res.*, 22(78):1–8, 2021.

Fletcher, P. T., Moeller, J., Phillips, J. M., and Venkatasubramanian, S. Computing hulls and centerpoints in positive definite space. *arXiv preprint arXiv:0912.1580*, 2009.

Fletcher, P. T., Moeller, J., Phillips, J. M., and Venkatasubramanian, S. Horoball hulls and extents in positive definite space. In *Workshop on Algorithms and Data Structures*, pp. 386–398. Springer, 2011.

Fournier, N. and Guillin, A. On the rate of convergence in wasserstein distance of the empirical measure. *Probability Theory and Related Fields*, 162(3):707–738, 2015.

Gaur, P., Pachori, R. B., Wang, H., and Prasad, G. A multiclass eeg-based bci classification using multivariate empirical mode decomposition based filtering and riemannian geometry. *Expert Systems with Applications*, 95: 201–211, 2018.

Golub, G. H. and Van Loan, C. F. *Matrix computations*. JHU press, 2013.

Hämäläinen, M., Hari, R., Ilmoniemi, R. J., Knuutila, J., and Lounasmaa, O. V. Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain. *Reviews of modern Physics*, 65(2):413, 1993.

Harandi, M., Salzmann, M., and Hartley, R. Dimensionality reduction on spd manifolds: The emergence of geometry-aware methods. *IEEE transactions on pattern analysis and machine intelligence*, 40(1):48–62, 2017.

Hein, M. and Bousquet, O. Hilbertian metrics and positive definite kernels on probability measures. In *International Workshop on Artificial Intelligence and Statistics*, pp. 136–143. PMLR, 2005.

Hersche, M., Rellstab, T., Schiavone, P. D., Cavigelli, L., Benini, L., and Rahimi, A. Fast and accurate multiclass inference for mi-bcis using large multiscale temporal and spectral features. In *2018 26th European Signal Processing Conference (EUSIPCO)*, pp. 1690–1694. IEEE, 2018.

Huang, Z. and Van Gool, L. A riemannian network for spd matrix learning. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

Huang, Z., Wang, R., Shan, S., Li, X., and Chen, X. Log-euclidean metric learning on symmetric positive definite manifold with application to image set classification. In *International conference on machine learning*, pp. 720–729. PMLR, 2015.

Ilea, I., Bombrun, L., Said, S., and Berthoumieu, Y. Covariance matrices encoding based on the log-euclidean and affine invariant riemannian metrics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 393–402, 2018.

Ju, C. and Guan, C. Deep optimal transport on spd manifolds for domain adaptation. *arXiv preprint arXiv:2201.05745*, 2022.

Kochurov, M., Karimov, R., and Kozlukov, S. Geoopt: Riemannian optimization in pytorch. *arXiv preprint arXiv:2005.02819*, 2020.

Kolouri, S., Zou, Y., and Rohde, G. K. Sliced wasserstein kernels for probability distributions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5258–5267, 2016.

Manole, T., Balakrishnan, S., and Wasserman, L. Minimax confidence intervals for the sliced wasserstein distance. *Electronic Journal of Statistics*, 16(1):2252–2345, 2022.

McCann, R. J. Polar factorization of maps on riemannian manifolds. *Geometric & Functional Analysis GAFA*, 11 (3):589–608, 2001.

Meunier, D., Pontil, M., and Ciliberto, C. Distribution regression with sliced Wasserstein kernels. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 15501–15523. PMLR, 17–23 Jul 2022.

Mezzadri, F. How to generate random matrices from the classical compact groups. *arXiv preprint math-ph/0609050*, 2006.

Murphy, K. P. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Nadjahi, K., Durmus, A., Simsekli, U., and Badeau, R. Asymptotic guarantees for learning generative models with the sliced-wasserstein distance. *Advances in Neural Information Processing Systems*, 32, 2019.

Nadjahi, K., Durmus, A., Chizat, L., Kolouri, S., Shahrampour, S., and Simsekli, U. Statistical and topological properties of sliced probability divergences. *Advances in Neural Information Processing Systems*, 33:20802–20812, 2020.

Nicolas-Alonso, L. F. and Gomez-Gil, J. Brain computer interfaces, a review. *sensors*, 12(2):1211–1279, 2012.

Niles-Weed, J. and Rigollet, P. Estimation of wasserstein distances in the spiked transport model. *Bernoulli*, 28(4): 2663–2688, 2022.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. 2017.

Paty, F.-P. and Cuturi, M. Subspace robust wasserstein distances. In *International conference on machine learning*, pp. 5072–5081. PMLR, 2019.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Pele, O. and Werman, M. Fast and robust earth mover's distances. In *2009 IEEE 12th international conference on computer vision*, pp. 460–467. IEEE, 2009.

Peng, H., Gong, W., Beckmann, C. F., Vedaldi, A., and Smith, S. M. Accurate brain age prediction with lightweight deep neural networks. *Medical image analysis*, 68:101871, 2021.

Pennec, X. Manifold-valued image processing with spd matrices. In *Riemannian geometric statistics in medical image analysis*, pp. 75–134. Elsevier, 2020.

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

Peyré, G., Cuturi, M., et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

Rabin, J., Peyré, G., Delon, J., and Bernot, M. Wasserstein barycenter and its application to texture mixing. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pp. 435–446. Springer, 2011.

Rakotomamonjy, A., Alaya, M. Z., Berar, M., and Gasso, G. Statistical and topological properties of gaussian smoothed sliced probability divergences. *arXiv preprint arXiv:2110.10524*, 2021.

Rivin, I. Surface area and other measures of ellipsoids. *Advances in Applied Mathematics*, 39(4):409–427, 2007.

Rodrigues, P. L. C., Jutten, C., and Congedo, M. Riemannian procrustes analysis: transfer learning for brain–computer interfaces. *IEEE Transactions on Biomedical Engineering*, 66(8):2390–2401, 2018.

Rustamov, R. M. and Majumdar, S. Intrinsic sliced wasserstein distances for comparing collections of probability distributions on manifolds and graphs. *arXiv preprint arXiv:2010.15285*, 2020.

Sabbagh, D., Ablin, P., Varoquaux, G., Gramfort, A., and Engemann, D. A. Manifold-regression to predict from meg/eeg brain signals without source modeling. *Advances in Neural Information Processing Systems*, 32, 2019.

Sabbagh, D., Ablin, P., Varoquaux, G., Gramfort, A., and Engemann, D. A. Predictive regression modeling with meg/eeg: from source power to signals and cognitive states. *NeuroImage*, 222:116893, 2020.

Spiegelhalter, D. How old are you, really? communicating chronic risk through 'effective age'of your body and organs. *BMC medical informatics and decision making*, 16 (1):1–6, 2016.

Taylor, J. R., Williams, N., Cusack, R., Auer, T., Shafto, M. A., Dixon, M., Tyler, L. K., Henson, R. N., et al. The cambridge centre for ageing and neuroscience (cam-can) data repository: Structural and functional mri, meg, and cognitive data from a cross-sectional adult lifespan sample. *neuroimage*, 144:262–269, 2017.

Thanwerdas, Y. and Pennec, X. O (n)-invariant riemannian metrics on spd matrices. *Linear Algebra and its Applications*, 661:163–201, 2023.

Villani, C. *Optimal transport: old and new*, volume 338. Springer, 2009.

Wolpaw, J. R. Brain–computer interfaces. In *Handbook of Clinical Neurology*, volume 110, pp. 67–74. Elsevier, 2013.

Xifra-Porxas, A., Ghosh, A., Mitsis, G. D., and Boudrias, M.-H. Estimating brain age from structural mri and meg data: Insights from dimensionality reduction techniques. *NeuroImage*, 231:117822, 2021.

Xu, H. Unsupervised manifold learning with polynomial mapping on symmetric positive definite matrices. *Information Sciences*, 609:215–227, 2022.

Yair, O., Dietrich, F., Talmon, R., and Kevrekidis, I. G. Domain adaptation with optimal transport on the manifold of spd matrices. *arXiv preprint arXiv:1906.00616*, 2019.

Yger, F., Berar, M., and Lotte, F. Riemannian approaches in brain-computer interfaces: a review. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25 (10):1753–1762, 2016.

A. Complementary experiments

A.1. Brain Age Prediction

Performance of SPDSW-based brain age regression on 10-folds cross validation for one random seed. In Figure A, we display the Mean Absolute Error (MAE) and the R^2 coefficient on 10-folds cross validation with one random seed. SPDSW is run with time-frames of 2s and 1000 projections.

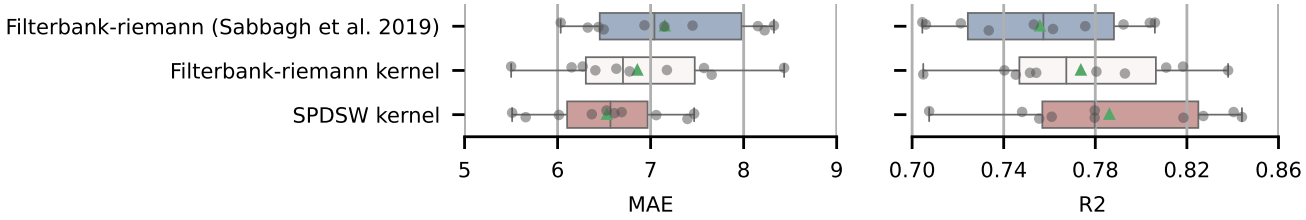


Figure A: Results of 10-folds cross validation on the Cam-CAN data-set for one random seed. We display the Mean Absolute Error (MAE) and the R^2 coefficient. SPDSW, with time-frames of 2s and 1000 projections, performs best. Note that Kernel Ridge regression based on the Log-Euclidean distance performs better than Ridge regression.

Performance of SPDSW-based brain age regression depending on number of projections. In Figure B, we display the MAE and R^2 score on brain age regression with different number of projections for 10 random seeds. In this example, the variance and scores are acceptable for 500 projections and more.

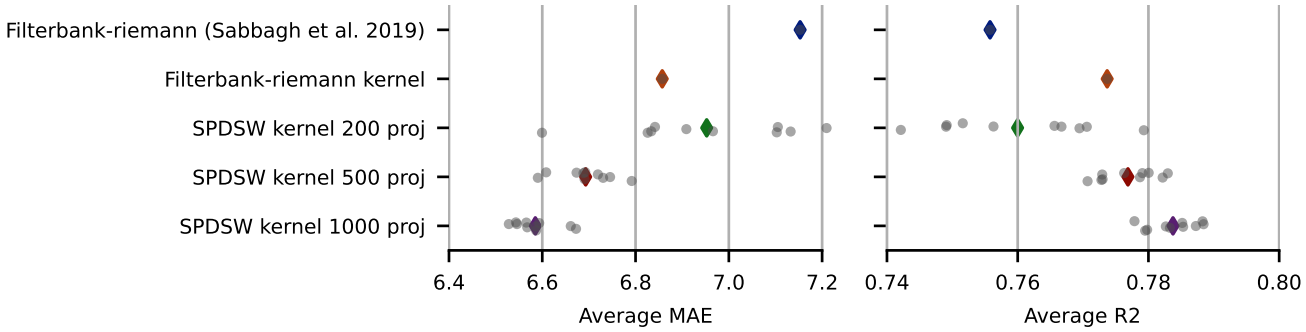


Figure B: Average results for 10 random seeds with 200, 500 and 1000 projections for SPDSW compared to average MAE and R^2 obtained with Ridge and Kernel Ridge regression on features from covariance estimates (Sabbagh et al., 2019). With enough projections, SPDSW kernel does not suffer from variance and performs best.

Performance of SPDSW-based brain age regression depending on timeframe length. In Figure C, we display the MAE and R^2 score on brain age regression with different time-frame lengths for 10 random seeds. The performance of $SPDSW$-kernel Ridge regression depends on a trade-off between the number of samples in each distribution (smaller time-frames for more samples), and the level of noise in the covariance estimate (larger time-frame for less noise). In this example, time-frames of 400 samples seems to be a good choice.

A.2. Domain Adaptation for BCI

Alignement. We plot on Figure D the classes of the target session (circles) and of the source session after alignment (crosses) on each subject. We observe that the classes seem to be well aligned, which explains why simple transformations work on this data-set. Hence, minimizing a discrepancy allows to align the classes even without taking them into account in the loss. More complicated data-sets might require to take into account the classes for the alignment.

Cross Subject Task. In Table 2, we add the results obtained on the cross subject task. On the column "subjects", we denote the source subject, and we report in the table the mean of the accuracies obtained over all other subjects as targets.

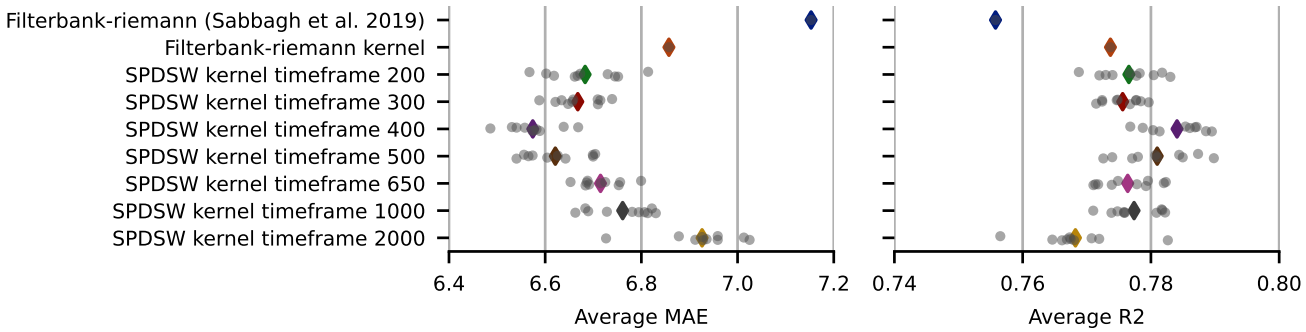


Figure C: Average MAE and R^2 score on brain age regression with different time-frame lengths for 10 random seeds The performance depends on the time-frame length, and there is a trade-off to find between number of samples and noise in the samples.

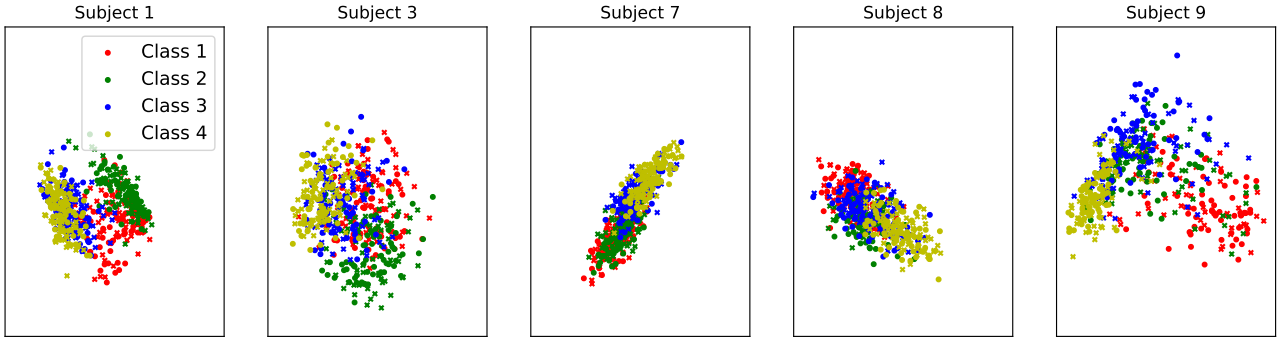


Figure D: PCA representation on BCI data. Circles represent points from the target session and crosses points from the source after alignment.

The results for AISTODA are taken from Yair et al. (2019, Table 1.b, Alg.1 (u)). The preprocessing and hyperparameters might not be the same as in our setting.

We add on Table 3 the detailed accuracies between subjects (with on the rows the Table, and on the columns the targets) for SPDSW, LEW, and when applying the classifier on the source.

Table 3: Accuracy between subjects. The row denote the source and the columns the targets.

Table 4: Source.

	1	3	7	8	9
1	-	52.22	50.55	39.02	26.58
3	34.43	-	30.10	49.62	27.43
7	52.01	53.33	-	26.14	26.58
8	49.82	57.78	24.35	0	39.66
9	26.74	28.52	24.72	39.39	-

Table 5: Particles + SPDSW.

	1	3	7	8	9
1	-	69.04	60.89	68.18	52.15
3	66.23	-	70.18	70.83	55.70
7	58.02	71.04	-	61.82	53.00
8	57.73	70.44	58.16	-	57.47
9	55.24	61.85	52.10	65.68	-

Table 6: Particles + LEW.

	1	3	7	8	9
1	-	72.59	55.42	69.32	54.01
3	63.37	-	61.99	62.12	53.59
7	50.18	62.96	-	48.11	51.48
8	61.54	74.07	53.87	-	57.22
9	48.35	63.33	57.20	64.02	-

Table 7: Transf. + SPDSW.

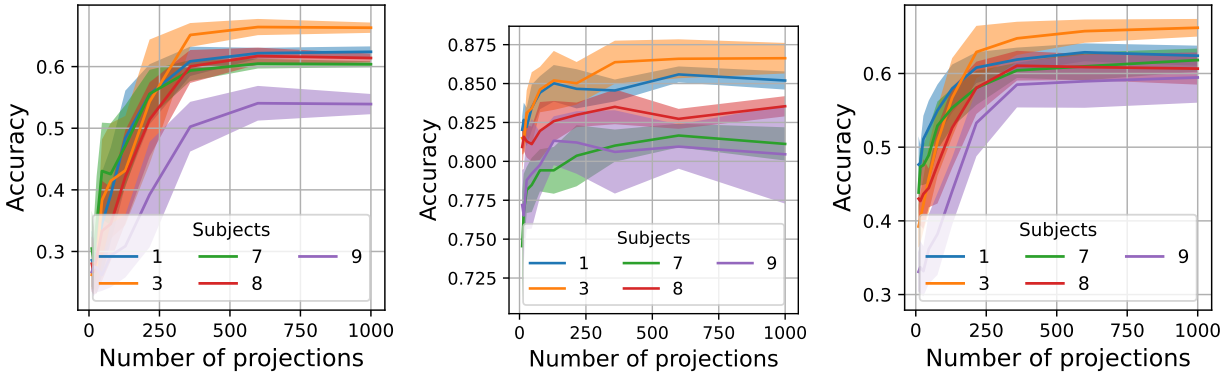
	1	3	7	8	9
1	-	68.00	59.04	68.79	51.81
3	68.42	-	71.07	69.24	56.88
7	57.66	69.78	-	60.83	53.42
8	62.71	72.07	53.87	-	55.70
9	53.92	59.04	40.15	60.15	-

Table 8: Transf. + LEW.

	1	3	7	8	9
1	-	70.00	59.78	68.18	53.59
3	69.60	-	71.59	69.32	54.85
7	57.88	73.37	-	61.74	53.59
8	63.00	72.22	54.24	-	55.70
9	55.31	60.00	39.48	64.02	-

Table 2: Accuracy and Runtime for Cross Subject.

Subjects	Source	AISOTDA (Yair et al., 2019)	SPDSW	LogSW	LEW	LES	SPDSW	LogSW	LEW	LES
			Transformations in $S_d^{++}(\mathbb{R})$				Descent over particles			
1	42.09	62.94	61.91	60.50	62.89	63.64	62.56	61.91	62.84	63.25
3	35.62	71.01	66.40	66.53	66.34	66.30	65.74	64.96	60.27	62.29
7	39.52	63.98	60.42	57.29	60.89	60.43	60.97	58.49	53.18	59.52
8	42.90	66.06	61.09	60.19	61.29	62.14	60.95	60.00	61.68	61.77
9	29.94	59.18	53.31	50.63	54.79	54.89	58.72	54.91	58.22	64.90
Avg. acc.	38.01	64.43	60.63	59.03	61.24	61.48	61.79	60.05	59.24	62.55
Avg. time	-	-	**4.34**	**4.31**	11.76	11.21	**3.67**	**3.64**	9.54	10.32



(a) Transformations on cross-subjects. (b) Particles on cross-session. (c) Particles on cross-subject.

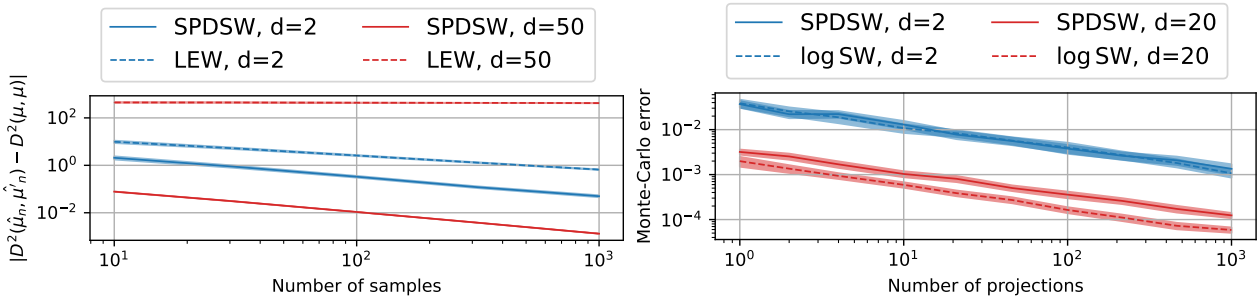
Figure E: Accuracy *w.r.t* the number of projections when optimizing over particles or transformations, and for the cross-session task and cross subject task. In all cases, the accuracy converge for 500 projections.

Evolution of the accuracy *w.r.t* the number of projections. On Figure 4, we plot the evolution of the accuracy obtained by learning transformations on $S_d^{++}(\mathbb{R})$ on the cross session task. We report on Figure E the plot for the other cases. We compared the results for $L \in \{10, 16, 27, 46, 77, 129, 215, 359, 599, 1000\}$ projections, which are evenly spaced in log scale. Other parameters are the same as in Table 1 and are detailed in Appendix B.3. The results were averaged over 10 runs, and we report the standard deviation.

A.3. Illustrations

Sample Complexity. We illustrate Proposition 2.9 in Figure Fa by plotting SPDSW and the Wasserstein distance with Log-Euclidean ground cost (LEW) between samples drawn from the same Wishart distribution, for $d = 2$ and $d = 50$. SPDSW is computed with $L = 1000$ projections. We observe that SPDSW converges with the same speed in both dimensions while LEW converges slower in dimension 50.

Projection Complexity. We illustrate Proposition 2.10 on Figure Fb by plotting the absolute error between $\widehat{\text{SPDSW}}^2_{2,L}$ and $\widehat{\text{SPDSW}}^2_{2,L^*}$. We fix L^* at 10000 which gives a good idea of the true value of SPDSW and we vary L between 1 and 10^3 evenly in log scale. We average the results over 100 runs and plot 95% confidence intervals. We observe that the Monte-Carlo error converges to 0 with a convergence rate of $O(\frac{1}{\sqrt{L}})$.



(a) Sample complexity of $D = \text{SPDSW}$ and $D = \text{LEW}$ for $d = 2$ and $d = 50$.

(b) Projection complexity of SPDSW and the logSW for $d = 2$ and $d = 20$.

Figure F: Sample and projection complexity. Experiments are replicated 100 times and we report the 95% confidence intervals. We note $\hat{\mu}_n$ and $\hat{\mu}'_n$ two different empirical distributions of μ. The sample complexity of SPDSW does not depend on the dimension contrary to Wasserstein. The projections complexity has a slope which decreases in $O(\frac{1}{\sqrt{L}})$.

B. Experimental details

B.1. Runtime

In Figure 2, we plot the runtime *w.r.t* the number of samples for different OT discrepancies. Namely, we compare SPDSW, log SW, the Wasserstein distance with Affine-Invariant ground cost, the Wasserstein distance with Log-Euclidean ground cost, and the Sinkhorn algorithm used to compute the entropic regularized OT problem with Log-Euclidean ground cost. The distance ground costs are computed with `geoopt` (Kochurov et al., 2020) while Wasserstein and Sinkhorn are computed with `POT` (Flamary et al., 2021). All computations are done on a A6000 GPU. We average the results over 20 runs and for $n \in \{100, 215, 464, 1000, 2154, 4641, 10000, 21544, 46415, 100000\}$ samples, which are evenly spaced in log scale, from a Wishart distribution in dimension $d = 20$. For the sliced methods, we fix $L = 200$ projections. For the Sinkhorn algorithm, we use a stopping threshold of 10^{-10} with maximum 10^5 iterations and a regularization parameter of $\epsilon = 1$.

B.2. Brain Age Prediction

We reuse the code for preprocessing steps and benchmarking procedure described in Engemann et al. (2022) for the CamCAN data-set, and available at `https://github.com/meeg-ml-benchmarks/brain-age-benchmark-paper`, which we recall here.

The data consist of measurements from 102 magnetometers and 204 gradiometers. First, we apply a band-pass filtering between 0.1Hz and 49Hz. Then, the signal is subsampled with a decimation factor of 5, leading to a sample frequency of 200Hz. Then, we apply the temporal signal-space-separation (tSSS). Default settings were applied for the harmonic decomposition (8 components of the internal sources, 3 for the external sources) on a 10-s sliding window. To discard segments for which inner and outer signal components were poorly distinguishable, we applied a correlation threshold of 98%.

For analysis, the band frequencies used are the following: (0.1Hz, 1Hz), (1Hz, 4Hz), (4Hz, 8Hz), (8Hz, 15Hz), (15Hz, 26Hz), (26Hz, 35Hz), (35Hz, 49Hz). The rank of the covariance matrices obtained after OAS is reduced to 53 with a PCA, which leads to the best score on this problem as mentioned in Sabbagh et al. (2020).

The code for the MEG experiments is essentially based on the work by Engemann et al. (2022), the class `SPDSW` available in the supplementary material, and the Kernel Ridge Regression of `scikit-learn`. The full version will be added later in order to respect anonymity.

B.3. Domain Adaptation for BCI

For both the optimization over particles and over transformations, we use `geoopt` (Kochurov et al., 2020) with the Riemannian gradient descent. We now detail the hyperparameters and the procedure.

First, the data from the BCI Competition IV 2a are preprocessed using the code from Hersche et al. (2018) available at

https://github.com/MultiScale-BCI/IV-2a. We applied a band-pass filter between 8 and 30 Hz. With these hyper-parameters, we get one regularized covariance matrix per subject.

For all experiments, we report the results averaged over 5 runs. For the sliced discrepancies, we always use $L = 500$ projections which we draw only once at the beginning. When optimizing over particles, we used a learning rate of 1000 for the sliced methods and of 10 for Wasserstein and Sinkhorn. The number of epochs was fixed at 500 for the cross-session task and for the cross-subject tasks. For the basic transformations, we always use 500 epochs and we choose a learning rate of $1e^{-1}$ on cross session and $5e^{-1}$ on cross subject for sliced methods, and of $1e^{-2}$ for Wasserstein and Sinkhorn. For the Sinkhorn algorithm, we use $\epsilon = 10$ with the default hyperparameters from the `POT` implementation. Moreover, we only use one translation and rotation for the transformation.

Furthermore, the results reported for AISOTDA in Table 1 and Table 2 are taken from Yair et al. (2019) (Table 1.a, column Alg.1 (u)). We note however that they may not have used the same preprocessing and hyperparameters to load the covariance matrices.

C. Proofs

Proposition 2.1. *Let $A \in S_d(\mathbb{R})$ with $\|A\|_F = 1$, and let $\mathcal{G}_A$ be the associated geodesic line. Then, for any $M \in S_d^{++}(\mathbb{R})$, the geodesic projection on $\mathcal{G}_A$ is*

$$P^{\mathcal{G}_A}(M) = \exp\big(\mathrm{Tr}(A \log M)A\big) \ . \tag{10}$$

Proof. Let $M \in S_d^{++}(\mathbb{R})$. We want to solve

$$P^{\mathcal{G}_A}(M) = \underset{X \in \mathcal{G}_A}{\arg\min}\ d_{LE}(X, M)^2 \ . \tag{26}$$

In the case of the Log-Euclidean metric, $\mathcal{G}_A = \{\exp(tA),\ t \in \mathbb{R}\}$. We have

$$\begin{aligned} d_{LE}(\exp(tA), M)^2 &= \|\log\exp(tA) - \log M\|_F^2 \\ &= \|tA - \log M\|_F^2 \\ &= t^2\mathrm{Tr}(A^2) + \mathrm{Tr}(\log(M)^2) - 2t\mathrm{Tr}(A\log M) \\ &= g(t) \ . \end{aligned} \tag{27}$$

Hence

$$g'(t) = 0 \iff t = \frac{\mathrm{Tr}(A\log M)}{\mathrm{Tr}(A^2)} \ . \tag{28}$$

Therefore

$$P^{\mathcal{G}_A}(M) = \exp\left(\frac{\mathrm{Tr}(A\log M)}{\mathrm{Tr}(A^2)}A\right) = \exp\left(\mathrm{Tr}(A\log M)A\right) \ , \tag{29}$$

since $\|A\|_F^2 = \mathrm{Tr}(A^2) = 1$. □

Proposition 2.2. *Let $A \in S_d(\mathbb{R})$ with $\|A\|_F = 1$, and let $\mathcal{G}_A$ be the associated geodesic line. Then, for any $M \in S_d^{++}(\mathbb{R})$, the geodesic coordinate on $\mathcal{G}_A$ is*

$$t^A(M) = \langle A, \log M\rangle_F = \mathrm{Tr}(A\log M) \ . \tag{12}$$

Proof. First, we give an orientation to the geodesic. This can be done by taking the sign of the inner product between $\log(P^{\mathcal{G}_A}(M))$ and A.

$$\begin{aligned} t^A(M) &= \mathrm{sign}(\langle A, \log(P^{\mathcal{G}_A}(M))\rangle_F) d\big(P^A(M), I\big) \\ &= \mathrm{sign}(\langle A, \log(P^{\mathcal{G}_A}(M))\rangle_F) d\left(\exp\left(\mathrm{Tr}(A\log M)A\right), I\right) \\ &= \mathrm{sign}(\langle A, \langle A, \log M\rangle_F A\rangle_F)\|\langle A\log M\rangle_F A - \log I\|_F \\ &= \mathrm{sign}(\langle A, \log M\rangle_F)|\langle A, \log M\rangle_F| \\ &= \langle A, \log M\rangle_F \\ &= \mathrm{Tr}(A\log M) \ . \end{aligned} \tag{30}$$

□

There are actually two possible ways to find coordinates on geodesically complete Riemannian manifolds (Bonet et al., 2022). The first one is to take the geodesic projection as previously done. A second solution is to use Busemann coordinates (Bridson & Haefliger, 2013; Chami et al., 2021).

Definition C.1. Let γ be a geodesic ray on a geodesically complete Riemannian manifold $\mathcal{M}$, *i.e.* for all $s, t \geq 0$, $d(\gamma(s), \gamma(t)) = |t - s|$. Then, the Busemann function B_γ associated to γ is defined as, for all $x \in \mathcal{M}$,

$$B_\gamma(x) = \lim_{t \to \infty} \left(d(x, \gamma(t)) - t \right) \ . \tag{31}$$

This function allows to derive coordinates on geodesically complete geodesics. For example, on $\mathbb{R}^d$, it actually coincides with the geodesic projection as it can be shown that, for $\theta \in S^{d-1}$,

$$\forall x \in \mathbb{R}^d, \ B_{\mathrm{span}(\theta)}(x) = -\langle x, \theta \rangle \ . \tag{32}$$

In Proposition C.2, we derive a closed-form for the Busemann function associated to a geodesic ray on $S_d^{++}(\mathbb{R})$ passing through the identity.

Proposition C.2 (Busemann coordinates)**.** *Let $A \in S_d(\mathbb{R})$ such that $\|A\|_F = 1$, and let $\mathcal{G}_A$ be the associated geodesic line. Then, the Busemann function associated to $\mathcal{G}_A$ is defined as*

$$\forall M \in S_d^{++}(\mathbb{R}), \ B^A(M) = -\mathrm{Tr}(A \log M) \ . \tag{33}$$

Proof. First, following (Bridson & Haefliger, 2013), we have for all $M \in S_d^{++}(\mathbb{R})$,

$$B^A(M) = \lim_{t \to \infty} \left(d_{LE}(\gamma_A(t), M) - t \right) = \lim_{t \to \infty} \frac{d_{LE}(\gamma_A(t), M)^2 - t^2}{2t} \ , \tag{34}$$

denoting $\gamma_A : t \mapsto \exp(tA)$ is the geodesic line associated to $\mathcal{G}_A$. Then, we get

$$\begin{aligned} \frac{d_{LE}(\gamma_A(t), M)^2 - t^2}{2t} &= \frac{1}{2t} \left(\| \log \gamma_A(t) - \log M \|_F^2 - t^2 \right) \\ &= \frac{1}{2t} \left(\| tA - \log M \|_F^2 - t^2 \right) \\ &= \frac{1}{2t} \left(t^2 \|A\|_F^2 + \| \log M \|_F^2 - 2t \langle A, \log M \rangle_F - t^2 \right) \\ &= -\langle A, \log M \rangle_F + \frac{1}{2t} \| \log M \|_F^2 \ , \end{aligned} \tag{35}$$

using that $\|A\|_F = 1$. Then, by passing to the limit $t \to \infty$, we find

$$B^A(t) = -\langle A, \log M \rangle_F = -\mathrm{Tr}(A \log M) \ . \tag{36}$$

□

We actually find that the Busemann coordinates are equal to the geodesic coordinates obtained in Proposition 2.2 up to the direction of the geodesic. We also show in Proposition C.3 that both projections on the geodesic coincide.

Proposition C.3 (Busemann projections)**.** *Let $A \in S_d(\mathbb{R})$ with $\|A\|_F = 1$ and let $\mathcal{G}_A$ the geodesic line associated. Then, for any $M \in S_d^{++}(\mathbb{R})$, the Busemann projection on $\mathcal{G}_A$ is*

$$P^A(M) = \exp\left(\mathrm{Tr}(A \log M) A \right) \ . \tag{37}$$

Proof. The geodesic line is of the form

$$\forall t \in \mathbb{R}, \ \gamma_A(t) = \exp(tA) \ . \tag{38}$$

We want to find a positive definite matrix on this geodesic with the same Busemann coordinate of M. Hence, we want to find t such that

$$\begin{aligned} B^A(M) = B^A(\gamma_A(t)) &\iff \mathrm{Tr}\big(A \log M\big) = \mathrm{Tr}\big(A \log(\exp(tA))\big) \\ &\iff \mathrm{Tr}(A \log M) = t\mathrm{Tr}(A^2) \\ &\iff t = \frac{\mathrm{Tr}(A \log M)}{\mathrm{Tr}(A^2)} = \mathrm{Tr}(A \log M) \ , \end{aligned} \tag{39}$$

since $\|A\|_F^2 = \mathrm{Tr}(A^2) = 1$.

Hence,

$$P^A(M) = \exp\left(\mathrm{Tr}(A \log M) A\right) \ . \tag{40}$$

□

Lemma 2.4. *Let λ_O be the uniform distribution on $\mathcal{O}_d = \{P \in \mathbb{R}^{d\times d}, P^TP = PP^T = I\}$ (Haar distribution), and λ be the uniform distribution on $S^{d-1} = \{\theta \in \mathbb{R}^d, \|\theta\|_2 = 1\}$. Then $\lambda_S \in \mathcal{P}(S_d(\mathbb{R}))$, defined such that $\forall\ A = P\mathrm{diag}(\theta)P^T \in S_d(\mathbb{R})$, $\mathrm{d}\lambda_S(A) = d!\ \mathrm{d}\lambda_O(P)\mathrm{d}\lambda(\theta)$, is the uniform distribution on $\{A \in S_d(\mathbb{R}),\ \|A\|_F = 1\}$.*

Proof. A matrix in $S_d(\mathbb{R})$ has a unique decomposition $P\mathrm{diag}(\theta)P^T$ up to permutations of the columns of $P \in \mathcal{O}_d$ and coefficients of $\theta \in S^{d-1}$. Thus, there is a bijection between $\{A \in S_d(\mathbb{R}),\ \|A\|_F = 1\}$ and the set $S_{(O),S^{d-1}}$ of $d!$-tuple $\{(P_1, \theta_1), \dots, (P_{d!}, \theta_{d!}) \in (\mathcal{O}_d \times S^{d-1})^{d!}\}$ such that (P_i, θ_i) is a permutation of (P_j, θ_j). Therefore, the uniform distribution $\lambda_{S_{(O),S^{d-1}}}$ on $S_{(O),S^{d-1}}$, defined as $\mathrm{d}\lambda_{S_{(O),S^{d-1}}}((P_1, \theta_1), \dots, (P_{d!}, \theta_{d!})) = \sum_{i=1}^{n!} \mathrm{d}(\lambda_O \otimes \lambda)(P_i, \theta_i) = d! \cdot \mathrm{d}(\lambda_O \otimes \lambda)(P_1, \theta_1)$, allows to define a uniform distribution λ_S on $\{A \in S_d(\mathbb{R}),\ \|A\|_F = 1\}$. Let $A = P\mathrm{diag}\theta P^T$ with $(P, \theta) \in \mathcal{O}_d \times S^{d-1}$, then

$$\mathrm{d}\lambda_S(A) = d!\ \mathrm{d}(\lambda_O \otimes \lambda)(P, \theta) \ . \tag{41}$$

Proposition 2.5. *Let $\tilde{\mu}, \tilde{\nu} \in \mathcal{P}_p(S_d(\mathbb{R}))$, and $\tilde{t}^A(B) = \mathrm{Tr}(A^TB)$ for $A, B \in S_d(\mathbb{R})$. We define*

$$\mathrm{SymSW}_p^p(\tilde{\mu}, \tilde{\nu}) = \int_{S_d(\mathbb{R})} W_p^p(\tilde{t}^A_\#\tilde{\mu}, \tilde{t}^A_\#\tilde{\nu})\ \mathrm{d}\lambda_S(A) \ . \tag{14}$$

Then, for $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$,

$$\mathrm{SPDSW}_p^p(\mu, \nu) = \mathrm{SymSW}_p^p(\log_\#\mu, \log_\#\nu) \ . \tag{15}$$

Proof. Denoting $\tilde{t}^A(B) = \langle B, A\rangle_F$ for all $B \in S_d(\mathbb{R})$, we obtain using (Paty & Cuturi, 2019, Lemma 6)

$$\begin{aligned} W_p^p(\tilde{t}^A_\#\log_\#\mu, \tilde{t}^A_\#\log_\#\nu) &= \inf_{\gamma\in\Pi(\mu,\nu)} \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} |\tilde{t}^A(\log(X)) - \tilde{t}^A(\log(Y))|^p\ \mathrm{d}\gamma(X, Y) \\ &= \inf_{\gamma\in\Pi(\mu,\nu)} \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} |t^A(X) - t^A(Y)|^p\ \mathrm{d}\gamma(X, Y) \\ &= W_p^p(t^A_\#\mu, t^A_\#\nu) \ , \end{aligned} \tag{42}$$

since $\tilde{t}^A(\log X) = \langle A, \log X\rangle_F = t^A(X)$. Hence,

$$\mathrm{SymSW}_p^p(\log_\#\mu, \log_\#\nu) = \mathrm{SPDSW}_p^p(\mu, \nu) \ . \tag{43}$$

Theorem 2.6. *Let $p \geq 1$, then SPDSW_p is a finite distance on $\mathcal{P}_p(S_d^{++}(\mathbb{R}))$.*

Proof. Let $p \geq 1$, and $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$. First, let's check that $\mathrm{SPDSW}_p^p(\mu, \nu) < \infty$.

To see that, we will use on one hand Villani (2009, Definition 6.4) which states that on a Riemannian manifold $\mathcal{M}$, for any $x_0 \in \mathcal{M}$,

$$\forall x, y \in \mathcal{M},\ d(x, y)^p \leq 2^{p-1}\big(d(x, x^0)^p + d(x^0, y)^p\big) \ . \tag{44}$$

Moreover, we will use that the projection t^A is equal (up to a sign) to the Busemann function which is 1-Lipschitz (Bridson & Haefliger, 2013, II. Proposition 8.22) and hence for any $A \in S_d(\mathbb{R})$ such that $\|A\|_F = 1$ and $X, Y \in S_d^{++}(\mathbb{R})$,

$|t^A(X) - t^A(Y)| \leq d_{LE}(X,Y)$. Then, using Paty & Cuturi (2019, Lemma 6), we have, for any $\pi \in \Pi(\mu,\nu)$ and $X_0 \in S_d^{++}(\mathbb{R})$,

$$\begin{aligned} W_p^p(t^A_\#\mu, t^A_\#\nu) &= \inf_{\gamma\in\Pi(\mu,\nu)} \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} |t^A(X) - t^A(Y)|^p \,\mathrm{d}\gamma(X,Y) \\ &\leq \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} |t^A(X) - t^A(Y)|^p \,\mathrm{d}\pi(X,Y) \\ &\leq 2^{p-1}\left(\int_{S_d^{++}(\mathbb{R})} |t^A(X) - t^A(X_0)|^p \,\mathrm{d}\mu(X) + \int_{S_d^{++}(\mathbb{R})} |t^A(X_0) - t^A(Y)|^p \,\mathrm{d}\nu(Y)\right) \\ &\leq 2^{p-1}\left(\int_{S_d^{++}(\mathbb{R})} d_{LE}(X,X_0)^p \,\mathrm{d}\mu(X) + \int_{S_d^{++}(\mathbb{R})} d_{LE}(Y,X_0)^p \,\mathrm{d}\nu(Y)\right) \\ &< \infty \ . \end{aligned} \tag{45}$$

Let $p \geq 1$, then for all $\mu,\nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$, it is straightforward to see that $\mathrm{SPDSW}_p(\mu,\nu) \geq 0$, $\mathrm{SPDSW}_p(\mu,\nu) = \mathrm{SPDSW}_p(\nu,\mu)$. It is also easy to see that $\mu = \nu \implies \mathrm{SPDSW}_p(\mu,\nu) = 0$ using that W_p is a distance.

Now, we can also derive the triangular inequality using the triangular inequality for W_p and the Minkowski inequality:

$$\begin{aligned} \forall \mu,\nu,\alpha \in \mathcal{P}_p(S_d^{++}(\mathbb{R})),\ \mathrm{SPDSW}_p(\mu,\nu) &= \Big(\int_{S_d(\mathbb{R})} W_p^p(t^A\#\mu, t^A\#\nu)\,\mathrm{d}\lambda_S(A)\Big)^{\frac{1}{p}} \\ &\leq \Big(\int_{S_d(\mathbb{R})} \big(W_p(t^A_\#\mu, t^A_\#\alpha) + W_p(t^A_\#\alpha, t^A_\#\nu)\big)^p \,\mathrm{d}\lambda_S(A)\Big)^{\frac{1}{p}} \\ &\leq \Big(\int_{S_d(\mathbb{R})} W_p^p(t^A\#\mu, t^A_\#\alpha)\,\mathrm{d}\lambda_S(A)\Big)^{\frac{1}{p}} \\ &+ \Big(\int_{S_d(\mathbb{R})} W_p^p(t^A_\#\alpha, t^A_\#\nu)\,\mathrm{d}\lambda_S(A)\Big)^{\frac{1}{p}} \\ &= \mathrm{SPDSW}_p(\mu,\alpha) + \mathrm{SPDSW}_p(\alpha,\nu) \ . \end{aligned} \tag{46}$$

Lastly, we can derive the indiscernible property. Let $\mu,\nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$ such that $\mathrm{SPDSW}_p(\mu,\nu) = 0$. Then, as for all $A \in S_d(\mathbb{R})$, $W_p^p(t^A_\#\mu, t^A_\#\nu) \geq 0$, it implies that for λ_S-almost every A, $W_p^p(t^A_\#\mu, t^A_\#\nu) = 0$ which implies $t^A_\#\mu = t^A_\#\nu$ for λ_S-almost every A since W_p is a distance. By taking the Fourier transform, this implies that for all $s \in \mathbb{R}$, $\widehat{t^A_\#\mu}(s) = \widehat{t^A_\#\nu}(s)$. But, we have

$$\begin{aligned} \widehat{t^A_\#\mu}(s) &= \int_{\mathbb{R}} e^{-2i\pi ts}\,\mathrm{d}(t^A_\#\mu)(s) \\ &= \int_{S_d^{++}(\mathbb{R})} e^{-2i\pi t^A(M)s}\,\mathrm{d}\mu(M) \\ &= \int_{S_d^{++}(\mathbb{R})} e^{-2i\pi\langle sA, \log M\rangle_F}\,\mathrm{d}\mu(M) \\ &= \int_{S_d(\mathbb{R})} e^{-2i\pi\langle sA, S\rangle_F}\,\mathrm{d}(\log_\#\mu)(S) \\ &= \widehat{\log_\#\mu}(sA) \ . \end{aligned} \tag{47}$$

Hence, we get that $\mathrm{SPDSW}_p(\mu,\nu) = 0$ implies that for λ_S-almost every A,

$$\forall s \in \mathbb{R},\ \widehat{\log_\#\mu}(sA) = \widehat{t^A_\#\mu}(s) = \widehat{t^A_\#\nu}(s) = \widehat{\log_\#\nu}(sA) \ . \tag{48}$$

By injectivity of the Fourier transform on $S_d(\mathbb{R})$, we get $\log_\#\mu = \log_\#\nu$. Then, as $\log$ is a bijection from $S_d^{++}(\mathbb{R})$ to

$S_d(\mathbb{R})$, we have for all Borelian $M \subset S_d^{++}(\mathbb{R})$,

$$\begin{aligned}\mu(M) &= \int_{S_d^{++}(\mathbb{R})} \mathbb{1}_M(X)\, \mathrm{d}\mu(X) \\ &= \int_{S_d(\mathbb{R})} \mathbb{1}_M(\exp(S))\, \mathrm{d}(\log_{\#}\mu)(S) \\ &= \int_{S_d(\mathbb{R})} \mathbb{1}_M(\exp(S))\, \mathrm{d}(\log_{\#}\nu)(S) \\ &= \int_{S_d^{++}(\mathbb{R})} \mathbb{1}_M(Y)\, \mathrm{d}\nu(Y) \\ &= \nu(M)\ .\end{aligned} \tag{49}$$

Hence, we conclude that $\mu = \nu$ and that SPDSW_p is a distance. □

To prove Theorem 2.7, we will adapt the proof of Nadjahi et al. (2020) to our projection. First, we start to adapt Nadjahi et al. (2020, Lemma S1):

Lemma C.4 (Lemma S1 in Nadjahi et al. (2020))**.** *Let* $(\mu_k)_k \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$ *and* $\mu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$ *such that* $\lim_{k\to\infty} \mathrm{SPDSW}_1(\mu_k, \mu) = 0$*. Then, there exists* $\varphi : \mathbb{N} \to \mathbb{N}$ *non decreasing such that* $\mu_{\varphi(k)} \xrightarrow[k\to\infty]{\mathcal{L}} \mu$*.*

Proof. By Bogachev & Ruas (2007, Theorem 2.2.5),

$$\lim_{k\to\infty} \int_{S^d(\mathbb{R})} W_1(t^A_{\#}\mu_k, t^A_{\#}\mu)\, \mathrm{d}\lambda_S(A) = 0 \tag{50}$$

implies that there exits a subsequence $(\mu_{\varphi(k)})_k$ such that for λ_S-almost every $A \in S_d(\mathbb{R})$,

$$W_1(t^A_{\#}\mu_{\varphi(k)}, t^A_{\#}\mu) \xrightarrow[k\to\infty]{} 0\ . \tag{51}$$

As the Wasserstein distance metrizes the weak convergence, this is equivalent to $t^A_{\#}\mu_{\varphi(k)} \xrightarrow[k\to\infty]{\mathcal{L}} t^A_{\#}\mu$.

Then, by Levy's characterization theorem, this is equivalent with the pointwise convergence of the characterization function, *i.e.* for all $t \in \mathbb{R}$, $\phi_{t^A_{\#}\mu_{\varphi(k)}}(t) \xrightarrow[k\to\infty]{} \phi_{t^A_{\#}\mu}(t)$. Moreover, we have for all $s \in \mathbb{R}$,

$$\begin{aligned}\phi_{t^A_{\#}\mu_{\varphi(k)}}(s) &= \int_{\mathbb{R}} e^{-its} \mathrm{d}(t^A_{\#}\mu_{\varphi(k)})(t) \\ &= \int_{S_d^{++}(\mathbb{R})} e^{-it^A(M)s}\, \mathrm{d}\mu_{\varphi(k)}(M) \\ &= \int_{S_d^{++}(\mathbb{R})} e^{-i\langle sA, \log M\rangle_F}\, \mathrm{d}\mu_{\varphi(k)}(M) \\ &= \int_{S_d(\mathbb{R})} e^{-i\langle sA, S\rangle_F}\, \mathrm{d}(\log_{\#}\mu_{\varphi(k)})(S) \\ &= \phi_{\log_{\#}\mu_{\varphi(k)}}(sA) \\ &\xrightarrow[k\to\infty]{} \phi_{\log_{\#}\mu}(sA)\ .\end{aligned} \tag{52}$$

Then, working in $S_d(\mathbb{R})$ with the Frobenius norm, we can use the same proof of Nadjahi et al. (2020) by using a convolution with a gaussian kernel and show that it implies that $\log_{\#}\mu_{\varphi(k)} \xrightarrow[k\to\infty]{\mathcal{L}} \log_{\#}\mu$.

Finally, let's show that it implies the weak convergence of $(\mu_{\varphi(k)})_k$ towards μ. Let $f \in C_b(S_d^{++}(\mathbb{R}))$, then

$$\begin{aligned}\int_{S_d^{++}(\mathbb{R})} f \, \mathrm{d}\mu_{\varphi(k)} &= \int_{S_d(\mathbb{R})} f \circ \exp \, \mathrm{d}(\log_{\#} \mu_{\varphi(k)}) \\ &\xrightarrow[k\to\infty]{} \int_{S_d(\mathbb{R})} f \circ \exp \, \mathrm{d}(\log_{\#} \mu) \\ &= \int_{S_d^{++}(\mathbb{R})} f \, \mathrm{d}\mu \ .\end{aligned} \tag{53}$$

Hence, we an conclude that $\mu_{\varphi(k)} \xrightarrow[k\to\infty]{\mathcal{L}} \mu$. □

Theorem 2.7. *For $p \geq 1$, SPDSW_p metrizes the weak convergence,* i.e. *for $\mu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$ and a sequence $(\mu_k)_k$ in $\mathcal{P}_p(S_d^{++}(\mathbb{R}))$, $\lim_{k\to\infty} \mathrm{SPDSW}_p(\mu_k, \mu) = 0$ if and only if $(\mu_k)_k$ converges weakly to μ.*

Proof. First, we suppose that $\mu_k \xrightarrow[k\to\infty]{\mathcal{L}} \mu$ in $\mathcal{P}_p(S_d^{++}(\mathbb{R}))$. Then, by continuity, we have that for λ_S almost every $A \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$, $t^A_{\#}\mu_k \xrightarrow[k\to\infty]{} t^A_{\#}\mu$. Moreover, as the Wasserstein distance on $\mathbb{R}$ metrizes the weak convergence, $W_p(t^A_{\#}\mu_k, t^A_{\#}\mu) \xrightarrow[k\to\infty]{} 0$. Finally, as W_p is bounded and it converges for λ_S-almost every A, we have by the Lebesgue convergence dominated theorem that $\mathrm{SPDSW}_p^p(\mu_k, \mu) \xrightarrow[k\to\infty]{} 0$.

On the other hand, suppose that $\mathrm{SPDSW}_p(\mu_k, \mu) \xrightarrow[k\to\infty]{} 0$. We first adapt Lemma S1 of (Nadjahi et al., 2020) in Lemma C.4 and observe that by the Hölder inequality,

$$\mathrm{SPDSW}_1(\mu, \nu) \leq \mathrm{SPDSW}_p(\mu, \nu) \ , \tag{54}$$

and hence $\mathrm{SPDSW}_1(\mu_k, \mu) \xrightarrow[k\to\infty]{} 0$.

By the same contradiction argument as in Nadjahi et al. (2020), let's suppose that $(\mu_k)_k$ does not converge to μ. Then, by denoting d_P the Lévy-Prokhorov metric, $\lim_{k\to\infty} d_P(\mu_k, \mu) \neq 0$.

Then, we have first that $\lim_{k\to\infty} \mathrm{SPDSW}_1(\mu_{\varphi(k)}, \mu) = 0$. Thus, by Lemma C.4, there exists a subsequence $(\mu_{\psi(\varphi(k))})_k$ such that $\mu_{\psi(\varphi(k))} \xrightarrow[k\to\infty]{\mathcal{L}} \mu$ which is equivalent to $\lim_{k\to\infty} d_P(\mu_{\psi(\varphi(k))}, \mu) = 0$ which contradicts the hypothesis.

We conclude that $(\mu_k)_k$ converges weakly to μ. □

For the proof of Theorem 2.8, we will first recall the following Theorem:

Theorem C.5 ((Rivin, 2007), Theorem 3)**.** *Let $f : \mathbb{R}^d \mapsto \mathbb{R}$ a homogeneous function of degree p (*i.e. *$\forall \alpha \in \mathbb{R}, \ f(\alpha x) = \alpha^p f(x)$). Then,*

$$\Gamma\Big(\frac{d+p}{2}\Big) \int_{S^{d-1}} f(x) \, \lambda(\mathrm{d}x) = \Gamma\Big(\frac{d}{2}\Big) \mathbb{E}[f(X)] \ , \tag{55}$$

where $\forall i \in \{1, ..., d\}$, $X_i \sim \mathcal{N}(0, \frac{1}{2})$ and $(X_i)_i$ are independent.

Then, making extensive use of this theorem, we show the following lemma:

Lemma C.6.

$$\forall S \in S_d(\mathbb{R}), \ \int_{S^{d-1}} |\langle \mathrm{diag}(\theta), S \rangle_F|^p \, \lambda(\mathrm{d}\theta) = \frac{1}{d} \left(\sum_i S_{ii}^2 \right)^{\frac{p}{2}} \int_{S^{d-1}} \|\theta\|_p^p \, \lambda(\mathrm{d}\theta) \ . \tag{56}$$

Proof. Let $f : \theta \mapsto \|\theta\|_p^p = \sum_{i=1}^d \theta_i^p$, then we have $f(\alpha\theta) = \alpha^p f(\theta)$ and f is p-homogeneous. By applying Theorem C.5, we have:

$$\begin{aligned}\int_{S^{d-1}} \|\theta\|_p^p \, \lambda(\mathrm{d}\theta) &= \frac{\Gamma\left(\frac{d}{2}\right)}{\Gamma\left(\frac{d+p}{2}\right)} \mathbb{E}[\|X\|_p^p] \text{ with } X_i \overset{\text{iid}}{\sim} \mathcal{N}(0, \frac{1}{2}) \\ &= \frac{\Gamma\left(\frac{d}{2}\right)}{\Gamma\left(\frac{d+p}{2}\right)} d \, \mathbb{E}[|X_1|_p^p] \\ &= \frac{\Gamma\left(\frac{d}{2}\right)}{\Gamma\left(\frac{d+p}{2}\right)} d \int |t|^p \frac{1}{\sqrt{\pi}} e^{-t^2} \mathrm{d}t \ .\end{aligned} \tag{57}$$

On the other hand, let $\tilde{f} : \theta \mapsto |\langle \mathrm{diag}(\theta), S \rangle_F|^p$, then $\tilde{f}(\alpha\theta) = \alpha^p \tilde{f}(\theta)$ and $\tilde{f}$ is p-homogeneous. By applying Theorem C.5, we have:

$$\begin{aligned}\int_{S^{d-1}} |\langle \mathrm{diag}(\theta), S \rangle_F|^p \, \lambda(\mathrm{d}\theta) &= \frac{\Gamma\left(\frac{d}{2}\right)}{\Gamma\left(\frac{d+p}{2}\right)} \mathbb{E}[|\langle \mathrm{diag}(X), S \rangle_F|^p] \text{ with } X_i \overset{\text{iid}}{\sim} \mathcal{N}(0, \frac{1}{2}) \\ &= \frac{\Gamma\left(\frac{d}{2}\right)}{\Gamma\left(\frac{d+p}{2}\right)} \int |t|^p \frac{1}{\sqrt{\sum_i S_{ii}^2 \pi}} e^{-\frac{t^2}{\sum_i z_{ii}^2}} \, \mathrm{d}t \text{ as } \langle \mathrm{diag}(X), S \rangle_F = \sum_i S_{ii} X_i \sim \mathcal{N}\Big(0, \frac{\sum_i S_{ii}^2}{2}\Big) \\ &= \frac{\Gamma\left(\frac{d}{2}\right)}{\Gamma\left(\frac{d+p}{2}\right)} \left(\sum_i S_{ii}^2\right)^{\frac{p}{2}} \int |u|^p \frac{1}{\sqrt{\sum_i S_{ii}^2 \pi}} e^{-u^2} \sqrt{\sum_i S_{ii}^2} \mathrm{d}u \text{ by } u = \frac{t}{\sqrt{\sum_i S_{ii}^2}} \\ &= \frac{\Gamma\left(\frac{d}{2}\right)}{\Gamma\left(\frac{d+p}{2}\right)} \left(\sum_i S_{ii}^2\right)^{\frac{p}{2}} \int |u|^p \frac{1}{\sqrt{\pi}} e^{-u^2} \mathrm{d}u \ .\end{aligned} \tag{58}$$

Hence, we deduce that

$$\int_{S^{d-1}} |\langle \mathrm{diag}(\theta), S \rangle_F|^p \, \lambda(\mathrm{d}\theta) = \frac{1}{d} \left(\sum_i S_{ii}^2\right)^{\frac{p}{2}} \int_{S^{d-1}} \|\theta\|_p^p \, \mathrm{d}\lambda(\theta) \ . \tag{59}$$

□

Theorem 2.8. *Let $p \geq 1$, let $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$. Then*

$$\mathrm{SPDSW}_p^p(\mu, \nu) \leq c_{d,p}^p W_p^p(\mu, \nu) \ , \tag{17}$$

where $c_{d,p}^p = \frac{1}{d} \int \|\theta\|_p^p \, \mathrm{d}\lambda(\theta)$. Let $R > 0$ and $B(I, R) = \{A \in S_d^{++}(\mathbb{R}), \ d_{LE}(A, I_d) = \|\log A\|_F \leq R\}$ be a closed ball. Then there exists a constant $C_{d,p,R}$ such that for all $\mu, \nu \in \mathcal{P}_p(B(I, R))$,

$$W_p^p(\mu, \nu) \leq C_{d,p,R} \mathrm{SPDSW}_p(\mu, \nu)^{\frac{2}{d(d+1)+2}} \ . \tag{18}$$

Proof. First, we show the upper bound of SPDSW_p. Let $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R})$ and $\gamma \in \Pi(\mu, \nu)$ an optimal coupling. Then, following the proof of Bonnotte (2013, Proposition 5.1.3), and using Paty & Cuturi (2019, Lemma 6) combined with the

fact that $(t^A \otimes t^A)_\# \gamma \in \Pi(t^A_\# \mu, t^A_\# \nu)$ for any $A \in S_d(\mathbb{R})$ such that $\|A\|_F = 1$, we obtain

$$\begin{aligned}
\mathrm{SPDSW}_p^p(\mu,\nu) &= \int_{S_d(\mathbb{R})} W_p^p(t^A_\#\mu, t^A_\#\nu)\, \mathrm{d}\lambda_S(A) \\
&\leq \int_{S_d(\mathbb{R})} \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} |t^A(X) - t^A(Y)|^p \, \mathrm{d}\gamma(X,Y)\, \mathrm{d}\lambda_S(A) \\
&= \int_{S_d(\mathbb{R})} \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} |\langle A, \log X - \log Y\rangle_F|^p \, \mathrm{d}\gamma(X,Y)\, \mathrm{d}\lambda_S(A) \\
&= \int_{S^{d-1}} \int_{\mathcal{O}_d} \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} |\langle P\mathrm{diag}(\theta)P^T, \log X - \log Y\rangle_F|^p \, \mathrm{d}\gamma(X,Y)\, \mathrm{d}\lambda_O(P)\mathrm{d}\lambda(\theta) \\
&= \int_{S^{d-1}} \int_{\mathcal{O}_d} \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} |\langle \mathrm{diag}(\theta), P^T(\log X - \log Y)P\rangle_F|^p \, \mathrm{d}\gamma(X,Y)\, \mathrm{d}\lambda_O(P)\mathrm{d}\lambda(\theta) \ .
\end{aligned} \tag{60}$$

By Lemma C.6, noting $S = P^T(\log X - \log Y)P$, we have

$$\begin{aligned}
\int_{S^{d-1}} |\langle \mathrm{diag}(\theta), S\rangle_F|^p \, \mathrm{d}\lambda(\theta) &= \frac{1}{d}\left(\sum_i S_{ii}^2\right)^{\frac{p}{2}} \int_{S^{d-1}} \|\theta\|_p^p \, \mathrm{d}\lambda(\theta) \\
&\leq \frac{1}{d}\|S\|_F^p \int_{S^{d-1}} \|\theta\|_p^p \, \mathrm{d}\lambda(\theta) \ ,
\end{aligned} \tag{61}$$

since $\|S\|_F^2 = \sum_{i,j} S_{ij}^2 \geq \sum_i S_{ii}^2$. Moreover, $\|S\|_F = \|P^T(\log X - \log Y)P\|_F = \|\log X - \log Y\|_F$. Hence, coming back to (60), we find

$$\begin{aligned}
\mathrm{SPDSW}_p^p(\mu,\nu) &\leq \frac{1}{d}\int_{S^{d-1}} \|\theta\|_p^p \, \mathrm{d}\lambda(\theta) \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} \|\log X - \log Y\|_F^p \, \mathrm{d}\gamma(X,Y) \\
&= \frac{1}{d}\int_{S^{d-1}} \|\theta\|_p^p \, \mathrm{d}\lambda(\theta)\, W_p^p(\mu,\nu) \\
&= c_{d,p}^p W_p^p(\mu,\nu) \ .
\end{aligned} \tag{62}$$

since γ is an optimal coupling between μ and ν for the Wasserstein distance with Log-Euclidean cost.

For the lower bound, let us first observe that

$$\begin{aligned}
W_1(\mu,\nu) &= \inf_{\gamma\in\Pi(\mu,\nu)} \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} d_{LE}(X,Y)\, \mathrm{d}\gamma(X,Y) \\
&= \inf_{\gamma\in\Pi(\mu,\nu)} \int_{S_d^{++}(\mathbb{R})\times S_d^{++}(\mathbb{R})} \|\log X - \log Y\|_F \, \mathrm{d}\gamma(X,Y) \\
&= \inf_{\gamma\in\Pi(\mu,\nu)} \int_{S_d(\mathbb{R})\times S_d(\mathbb{R})} \|U - V\|_F \, \mathrm{d}(\log\otimes\log)_\#\gamma(U,V) \\
&= \inf_{\gamma\in\Pi(\log_\#\mu,\log_\#\nu)} \int_{S_d(\mathbb{R})\times S_d(\mathbb{R})} \|U - V\|_F \, \mathrm{d}\gamma(U,V) \\
&= W_1(\log_\#\mu, \log_\#\nu) \ ,
\end{aligned} \tag{63}$$

where we used Paty & Cuturi (2019, Lemma 6).

Using Proposition 2.5, we have

$$\mathrm{SymSW}_1(\log_\#\mu, \log_\#\nu) = \mathrm{SPDSW}_1(\mu,\nu) \ . \tag{64}$$

Therefore, as $S_d(\mathbb{R})$ is an Euclidean space of dimension $d(d+1)/2$, we can use (Bonnotte, 2013, Lemma 5.1.4) and we obtain that

$$W_1(\log_\#\mu, \log_\#\nu) \leq C_{d(d+1)/2} R^{d(d+1)/(d(d+1)+2)} \mathrm{SymSW}_1(\log_\#\mu, \log_\#\nu)^{2/(d(d+1)+2)} \ . \tag{65}$$

Then, using that $\mathrm{SymSW}_1(\log_{\#}\mu, \log_{\#}\nu) = \mathrm{SPDSW}_1(\mu,\nu)$ and $W_1(\log_{\#}\mu, \log_{\#}\nu) = W_1(\mu,\nu)$, we obtain

$$W_1(\mu,\nu) \leq C_{d(d+1)/2} R^{d(d+1)/(d(d+1)+2)} \mathrm{SPDSW}_1(\mu,\nu)^{2/(d(d+1)+2)} \ . \tag{66}$$

Now, following the proof of Bonnotte (2013, Theorem 5.1.5), we use that on one hand, $W_p^p(\mu,\nu) \leq (2R)^{p-1} W_1(\mu,\nu)$, and on the other hand, by Hölder, $\mathrm{SPDSW}_1(\mu,\nu) \leq \mathrm{SPDSW}_p(\mu,\nu)$. Hence, using inequalities (62) and (66), we get

$$\begin{aligned}
\mathrm{SPDSW}_p^p(\mu,\nu) &\leq c_{d,p}^p W_p^p(\mu,\nu) \\
&\leq (2R)^{p-1} W_1(\mu,\nu) \\
&\leq 2^{p-1} C_{d(d+1)/2} R^{p-1+d(d+1)/(d(d+1)+2)} \mathrm{SPDSW}_1(\mu,\nu)^{2/(d(d+1)/2)} \\
&= C_{d,p}^d R^{p-2/(d(d+1))} \mathrm{SPDSW}_1(\mu,\nu)^{2/(d(d+1)+2)} \ .
\end{aligned} \tag{67}$$

□

Proposition 2.9. *Let $q > p \geq 1$, $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$, and $\hat{\mu}_n, \hat{\nu}_n$ the associated empirical measures. We define the moment of order q by $M_q(\mu) = \int \|X\|_F^q \, \mathrm{d}\mu(X)$, and $M_q(\mu,\nu) = M_q(\log_{\#}\mu)^{1/q} + M_q(\log_{\#}\nu)^{1/q}$. Then, there exists a constant $C_{p,q}$ depending only on p and q such that*

$$\begin{aligned}
&\mathbb{E}\big[|\mathrm{SPDSW}_p(\hat{\mu}_n,\hat{\nu}_n) - \mathrm{SPDSW}_p(\mu,\nu)|\big] \\
&\qquad \leq \alpha_{n,p,q} C_{p,q}^{1/p} M_q(\mu,\nu) \ ,
\end{aligned} \tag{19}$$

$$\text{where} \quad \alpha_{n,p,q} = \begin{cases} n^{-1/(2p)} & \text{if } q > 2p \\ n^{-1/(2p)} \log(n)^{1/p} & \text{if } q = 2p \\ n^{-(q-p)/(pq)} & \text{if } q \in (p, 2p) \ . \end{cases}$$

Proof. In this proof, we will follow the derivations used in Nadjahi et al. (2020) and in (Rakotomamonjy et al., 2021). Notably, we will use the adaptation of Fournier & Guillin (2015, Theorem 2) reported in Rakotomamonjy et al. (2021, Lemma 1), which we recall now.

Lemma C.7 (Lemma 1 in Rakotomamonjy et al. (2021) and Theorem 2 in Fournier & Guillin (2015))**.** *Let $p \geq 1$ and $\eta \in \mathcal{P}_p(\mathbb{R})$. Denote $M_q(\eta) = \int |x|^q \, \mathrm{d}\eta(x)$ the moments of order q and assume that $M_q(\eta) < \infty$ for some $q > p$. Then, there exists a constant $C_{p,q}$ depending only on p, q such that for all $n \geq 1$,*

$$\mathbb{E}[W_p^p(\hat{\eta}_n, \eta)] \leq C_{p,q} M_q(\eta)^{p/q} \left(n^{-1/2} \mathbb{1}_{\{q>2p\}} + n^{-1/2} \log(n) \mathbb{1}_{\{q=2p\}} + n^{-(q-p)/q} \mathbb{1}_{\{q\in(p,2p)\}} \right) \ . \tag{68}$$

First, let us observe that by the triangular and reverse triangular inequalities, as well as Jensen for $x \mapsto x^{1/p}$ (which is concave since $p \geq 1$),

$$\begin{aligned}
\mathbb{E}\left[|\mathrm{SPDSW}_p(\hat{\mu}_n,\hat{\nu}_n) - \mathrm{SPDSW}_p(\mu,\nu)|\right] &= \mathbb{E}[|\mathrm{SPDSW}_p(\hat{\mu}_n,\hat{\nu}_n) - \mathrm{SPDSW}_p(\hat{\mu}_n,\nu) \\
&\quad + \mathrm{SPDSW}_p(\hat{\mu}_n,\nu) - \mathrm{SPDSW}_p(\mu,\nu)|] \\
&\leq \mathbb{E}[|\mathrm{SPDSW}_p(\hat{\mu}_n,\hat{\nu}_n) - \mathrm{SPDSW}_p(\hat{\mu}_n,\nu)|] \\
&\quad + \mathbb{E}[|\mathrm{SPDSW}_p(\hat{\mu}_n,\nu) - \mathrm{SPDSW}_p(\mu,\nu)|] \\
&\leq \mathbb{E}[\mathrm{SPDSW}_p(\hat{\nu}_n,\nu)] + \mathbb{E}[\mathrm{SPDSW}_p(\hat{\mu}_n,\mu)] \\
&\leq \mathbb{E}[\mathrm{SPDSW}_p^p(\hat{\nu}_n,\nu)]^{1/p} + \mathbb{E}[\mathrm{SPDSW}_p^p(\hat{\mu}_n,\mu)]^{1/p} \ .
\end{aligned} \tag{69}$$

Moreover, by Fubini-Tonelli,

$$\begin{aligned}
\mathbb{E}[\mathrm{SPDSW}_p^p(\hat{\mu}_n,\mu)] &= \mathbb{E}\left[\int_{S_d(\mathbb{R})} W_p^p(t^A_{\#}\hat{\mu}_n, t^A_{\#}\mu) \, \mathrm{d}\lambda_S(A) \right] \\
&= \int_{S_d(\mathbb{R})} \mathbb{E}[W_p^p(t^A_{\#}\hat{\mu}_n, t^A_{\#}\mu)] \, \mathrm{d}\lambda_S(A) \ .
\end{aligned} \tag{70}$$

By applying Lemma C.7, we get for $q > p$ that there exists a constant $C_{p,q}$ such that,

$$\mathbb{E}[W_p^p(t^A_\# \hat{\mu}_n, t^A_\# \mu)] \leq C_{p,q} M_q(t^A_\# \mu)^{p/q} \left(n^{-1/2} \mathbb{1}_{\{q>2\}} + n^{-1/2} \log(n) \mathbb{1}_{\{q=2p\}} + n^{-(q-p)/q} \mathbb{1}_{\{q\in(p,2p)\}} \right) \ . \tag{71}$$

Furthermore, using Cauchy-Schwartz and that $\|A\|_F = 1$,

$$\begin{aligned} M_q(t^A_\# \mu) &= \int_{\mathbb{R}} |x|^q \, \mathrm{d}(t^A_\# \mu)(x) \\ &= \int_{S_d^{++}(\mathbb{R})} |\langle A, \log X \rangle|^q \, \mathrm{d}\mu(X) \\ &\leq \int_{S_d^{++}(\mathbb{R})} \|\log X\|_F^q \, \mathrm{d}\mu(X) \\ &= M_q(\log_\# \mu) \ . \end{aligned} \tag{72}$$

Therefore, we have that

$$\mathbb{E}[\mathrm{SPDSW}_p^p(\hat{\mu}_n, \mu)] \leq C_{p,q} M_q(\log_\# \mu)^{p/q} \left(n^{-1/2} \mathbb{1}_{\{q>2p\}} + n^{-1/2} \log(n) \mathbb{1}_{\{q=2p\}} + n^{-(q-p)/q} \mathbb{1}_{\{q\in(p,2p)\}} \right) \ , \tag{73}$$

and similarly

$$\mathbb{E}[\mathrm{SPDSW}_p^p(\hat{\nu}_n, \nu)] \leq C_{p,q} M_q(\log_\# \nu)^{p/q} \left(n^{-1/2} \mathbb{1}_{\{q>2p\}} + n^{-1/2} \log(n) \mathbb{1}_{\{q=2p\}} + n^{-(q-p)/q} \mathbb{1}_{\{q\in(p,2p)\}} \right) \ . \tag{74}$$

Hence, we conclude that the sample complexity is

$$\mathbb{E}\left[|\mathrm{SPDSW}_p(\hat{\mu}_n, \hat{\nu}_n) - \mathrm{SPDSW}_p(\mu, \nu)| \right] \leq C_{p,q}^{1/p} \left(M_q(\log_\# \mu)^{1/q} + M_q(\log_\# \nu)^{1/q} \right) \begin{cases} n^{-1/(2p)} & \text{if } q > 2p \\ n^{-1/(2p)} \log(n)^{1/p} & \text{if } q = 2p \\ n^{-(q-p)/(pq)} & \text{if } q \in (p, 2p) \ . \end{cases} \tag{75}$$

□

Proposition 2.10. *Let $p \geq 1$, $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$. Then, the error made by the Monte Carlo estimate of SPDSW_p with L projections can be bounded as follows*

$$\begin{aligned} &\mathbb{E}_A \left[|\widehat{\mathrm{SPDSW}}_{p,L}^p(\mu, \nu) - \mathrm{SPDSW}_p^p(\mu, \nu)| \right]^2 \\ &\leq \frac{1}{L} \mathrm{Var}_{A\sim\lambda_S} \left[W_p^p(t^A_\# \mu, t^A_\# \nu) \right] \ , \end{aligned} \tag{20}$$

where $\widehat{\mathrm{SPDSW}}_{p,L}^p(\mu, \nu) = \frac{1}{L} \sum_{i=1}^L W_p^p(t^{A_i}_\# \mu, t^{A_i}_\# \nu)$ with $(A_i)_{i=1}^L$ independent samples from λ_S.

Proof. Let $(A_i)_{i=1}^L$ be iid samples of λ_S. Then, by first using Jensen inequality and then remembering that $\mathbb{E}_A[W_p^p(t^A_\# \mu, t^A_\# \nu)] = \mathrm{SPDSW}_p^p(\mu, \nu)$, we have

$$\begin{aligned} \mathbb{E}_A \left[|\widehat{\mathrm{SPDSW}}_{p,L}^p(\mu, \nu) - \mathrm{SPDSW}_p^p(\mu, \nu)| \right]^2 &\leq \mathbb{E}_A \left[\left| \widehat{\mathrm{SPDSW}}_{p,L}^p(\mu, \nu) - \mathrm{SPDSW}_p^p(\mu, \nu) \right|^2 \right] \\ &= \mathbb{E}_A \left[\left| \frac{1}{L} \sum_{i=1}^L \left(W_p^p(t^{A_i}_\# \mu, t^{A_i}_\# \nu) - \mathrm{SPDSW}_p^p(\mu, \nu) \right) \right|^2 \right] \\ &= \frac{1}{L^2} \mathrm{Var}_A \left[\sum_{i=1}^L W_p^p(t^{A_i}_\# \mu, t^{A_i}_\# \nu) \right] \\ &= \frac{1}{L} \mathrm{Var}_A \left[W_p^p(t^A_\# \mu, t^A_\# \nu) \right] \\ &= \frac{1}{L} \int_{S_d(\mathbb{R})} \left(W_p^p(t^A_\# \mu, t^A_\# \nu) - \mathrm{SPDSW}_p^p(\mu, \nu) \right)^2 \, \mathrm{d}\lambda_S(A) \ . \end{aligned} \tag{76}$$

□

Proposition 3.1. *Let m be the Lebesgue measure and let $\mathcal{H} = L^2([0,1] \times S_d(\mathbb{R}), m \otimes \lambda_S)$. We define Φ as*

$$\begin{aligned} \Phi : \mathcal{P}_2(S_d^{++}(\mathbb{R})) &\to \mathcal{H} \\ \mu &\mapsto \left((q, A) \mapsto F^{-1}_{t^A_\# \mu}(q)\right) \ , \end{aligned} \tag{21}$$

where $F^{-1}_{t^A_\# \mu}$ is the quantile function of $t^A_\# \mu$. Then, SPDSW_2 is Hilbertian and for all $\mu, \nu \in \mathcal{P}_2(S_d^{++}(\mathbb{R}))$,

$$\mathrm{SPDSW}_2^2(\mu, \nu) = \|\Phi(\mu) - \Phi(\nu)\|^2_{\mathcal{H}} \ . \tag{22}$$

Proof. Let μ, ν be probability distributions on $S_d^{++}(\mathbb{R})$ with moments of order $p = 2$. Then

$$\begin{aligned} \mathrm{SPDSW}_2^2(\mu, \nu) &= \int_{S_d} \|F^{-1}_{t^A_\# \mu} - F^{-1}_{t^A_\# \nu}\|^2 \, \mathrm{d}\lambda_S(A) \\ &= \int_{S_d} \int_0^1 \left(F^{-1}_{t^A_\# \mu}(q) - F^{-1}_{t^A_\# \nu}(q)\right)^2 \mathrm{d}q \mathrm{d}\lambda_S(A) \\ &= \|\Phi(\mu) - \Phi(\nu)\|^2_{\mathcal{H}} \ . \end{aligned}$$

Thus, SPDSW_2 is Hilbertian. □

D. SPDSW with Affine-Invariant Metric

In the main part of the paper, we focused on $S_d^{++}(\mathbb{R})$ endowed with the Log-Euclidean metric. With this metric, $S_d^{++}(\mathbb{R})$ is a Riemannian manifold of constant null curvature as classical Euclidean spaces. Another metric of interest, very related to the Log-Euclidean one, is the Affine-Invariant metric, which yields a Riemannian manifold of non-constant and non-positive curvature (Bhatia, 2009; Bridson & Haefliger, 2013). The Log-Euclidean distance is actually a lower bound of the Affine-Invariant distance, and they coincide when the matrices commute. Notably, they share the same geodesics passing through the identity (Pennec, 2020, Section 3.6.1). The Log-Euclidean metric can actually be seen as a good first order approximation of the Affine-Invariant metric (Arsigny et al., 2005; Pennec, 2020) which motivated the proposal of this metric. But it can lose some information when matrices are not commuting. Hence, we can wonder whether or not constructing a sliced discrepancy with projections obtained in $S_d^{++}(\mathbb{R})$ endowed with the Affine-Invariant metric could improve the results.

D.1. Busemann Function on Affine-Invariant Space

As recalled in Section 2.2, for the Affine-Invariant metric, the inner product in the tangent space is defined as

$$\forall M \in S_d^{++}(\mathbb{R}), \ A, B \in \mathcal{T}_M, \ \langle A, B \rangle_M = \mathrm{Tr}(M^{-1} A M^{-1} B) \ , \tag{77}$$

and the corresponding geodesic distance is

$$\forall X, Y \in S_d^{++}(\mathbb{R}), \ d_{AI}(X, Y) = \sqrt{\mathrm{Tr}(\log(X^{-1}Y)^2)} \ . \tag{78}$$

This distance notably satisfies the affine-invariant property, that is, for any $g \in GL_d(\mathbb{R})$, where $GL_d(\mathbb{R})$ denotes the set of invertibles matrices in $\mathbb{R}^{d \times d}$,

$$\forall X, Y \in S_d^{++}(\mathbb{R}), \ d_{AI}(g \cdot X, g \cdot Y) = d_{AI}(X, Y) \ , \tag{79}$$

where $g \cdot X = g X g^T$. Geodesics passing through the identity coincide with those of the Log-Euclidean metric, and are therefore of the form $\mathcal{G}_A = \{\exp(tA), \ t \in \mathbb{R}\}$ where $A \in S_d(\mathbb{R})$. Hence, we now need to find a projection of $M \in S_d^{++}(\mathbb{R})$ onto $\mathcal{G}_A$.

Unfortunately, to the best of our knowledge, there is no closed-form for the geodesic projection on geodesics. We will discuss here the horospherical projection which can be obtained with the Busemann function. For $A \in S_d(\mathbb{R})$ such that $\|A\|_F = 1$, denoting $\gamma_A : t \mapsto \exp(tA)$ the geodesic passing through I_d with direction A, the Busemann function B^A associated to γ_A writes as

$$\forall M \in S_d^{++}(\mathbb{R}), \ B^A(M) = \lim_{t \to \infty} \left(d_{AI}(\exp(tA), M) - t\right) \ . \tag{80}$$

Contrary to the Log-Euclidean case, we cannot directly compute this quantity by expanding the distance since $\exp(-tA)$ and M are not necessarily commuting. The main idea to solve this issue is to first find a group $G \subset GL_d(\mathbb{R})$ which will leave the Busemann function invariant. Then, we can find an element of this group which will project M on the space of matrices commuting with $\exp(A)$. This part of the space is of null curvature, *i.e.* it is isometric to an Euclidean space. In this case, we can compute the Busemann function as in Proposition C.2 as the matrices are commuting. Hence, the Busemann function is of the form

$$B^A(M) = -\langle A, \log(\pi_A(M))\rangle_F \ , \tag{81}$$

where π_A is a projection on the space of commuting matrices.

When A is diagonal with sorted values such that $A_{11} > \cdots > A_{dd}$, then the group leaving the Busemann function invariant is the set of upper triangular matrices with ones on the diagonal (Bridson & Haefliger, 2013, II. Proposition 10.66), *i.e.* for such matrix g, $B^A(M) = B^A(gMg^T)$. If the points are sorted in increasing order, then the group is the set of lower triangular matrices. Let's note G_U the set of upper triangular matrices with ones on the diagonal. For a general $A \in S_d(\mathbb{R})$, we can first find an appropriate diagonalization $A = P\tilde{A}P^T$, where $\tilde{A}$ is diagonal sorted, and apply the change of basis $\tilde{M} = P^T M P$ (Fletcher et al., 2009). Note that since we sample the eigenvalues from the uniform distribution on S^{d-1}, the values are all different almost surely. Therefore, we suppose that all the eigenvalues of A have an order of multiplicity of one. By the affine-invariance property, the distances do not change, *i.e.* $d_{AI}(\exp(tA), M) = d_{AI}(\exp(t\tilde{A}), \tilde{M})$ and hence, using the definition of the Busemann function, we have that $B^A(M) = B^{\tilde{A}}(\tilde{M})$. Then, we need to project $\tilde{M}$ on the space of matrices commuting with $e^{\tilde{A}}$ which we denote $F(A)$. By Bridson & Haefliger (2013, II. Proposition 10.67), this space corresponds to the diagonal matrices. Moreover, by Bridson & Haefliger (2013, II. Proposition 10.69), there is a unique pair $(g, D) \in G_U \times F(A)$ such that $\tilde{M} = gDg^T$, and therefore, we can note $\pi_A(\tilde{M}) = D$. This decomposition actually corresponds to a UDU decomposition. If the eigenvalues of A are sorted in increasing order, this would correspond to a LDL decomposition.

For more details about the Busemann function on the Affine-invariant space, we refer to Bridson & Haefliger (2013, Section II.10) and Fletcher et al. (2009; 2011).

D.2. Horospherical SPDSW

Now that we know how to compute the coordinates on geodesics passing through the identity, we can derive an associated sliced discrepancy, which we call horospherical SPDSW (HSPDSW) since the projection is made along level sets of the Busemann function, which are called horospheres (Fletcher et al., 2009).

Definition D.1. Let λ_O be the uniform distribution on orthogonal matrices $\mathcal{O}_d = \{P \in \mathbb{R}^{d\times d}, P^TP = PP^T = I\}$ (Haar distribution), λ be the uniform distribution on $S^{d-1} = \{\theta \in \mathbb{R}^d, \|\theta\|_2 = 1\}$, and λ_S be a probability distribution on $S_d(\mathbb{R})$ such that for all $V_S \in \sigma(S_d(\mathbb{R}))$, $\lambda_S(V_S) = (\lambda_O \otimes \lambda)(A_S)$ where $A_S = \{(P, \theta) \in \mathcal{O}_d \times S^{d-1},\ P\mathrm{diag}(\theta)P^T \in V_S\}$. Let $\mu, \nu \in \mathcal{P}_p(S_d^{++}(\mathbb{R}))$, the HSPDSW discrepancy is defined as

$$\mathrm{HSPDSW}_p^p(\mu, \nu) = \int_{S_d(\mathbb{R})} W_p^p(B^A_\#\mu, B^A_\#\nu)\, \mathrm{d}\lambda_S(A) \ , \tag{82}$$

where $B^A(M) = -\mathrm{Tr}\big(A\log(\pi_A(M))\big)$, with π^A the projection derived in Appendix D.1.

On the side of theoretical properties, this discrepancy is still a pseudo-distance. However, since the projection $\log \circ \pi_A$ is not a diffeomorphism, whether the indiscernible property holds or not remains an open question.

On the computational side, it requires an additional projection step with a UDU decomposition for each sample and projection. Hence the overall complexity becomes $O(Ln(\log n + d^3))$ where the $O(Lnd^3)$ comes from the UDU decomposition. In practice, it takes more time than SPDSW for results which are pretty similar. In the same setting detailed in Appendix B, we plot on Figure G the runtime *w.r.t* the number of samples and observe that it takes even more time than the Wasserstein distance. We detail the procedure to compute HSPDSW in Algorithm 2.

D.3. Experimental results on brain age prediction.

As for SPDSW, HSPDSW allows to define a kernel for distributions $(\mu_i)_{i=1}^n \in (\mathcal{P}_p(S_d^{++}(\mathbb{R})))^n$

$$K(\mu_i, \mu_j) = e^{-\frac{\mathrm{HSPDSW}(\mu_\mathrm{i}, \mu_\mathrm{j})}{2\sigma^2}} \ . \tag{83}$$

Algorithm 2 Computation of HSPDSW

Input: $(X_i)_{i=1}^n \sim \mu$, $(Y_j)_{j=1}^m \sim \nu$, L the number of projections, p the order
for $\ell = 1$ **to** L **do**
  Draw $\theta \sim \text{Unif}(S^{d-1}) = \lambda$
  Draw $P \sim \text{Unif}(O_d(\mathbb{R})) = \lambda_O$
  Get Q the permutation matrix such that $\tilde{\theta} = Q\theta$ is sorted in decreasing order
  Set $A = \text{diag}(\tilde{\theta})$, $\tilde{P} = PQ^T$
  $\forall i, j,\ \tilde{X}_i^\ell = \tilde{P}^T X_i \tilde{P}$, $\tilde{Y}_j^\ell = \tilde{P}^T Y_j \tilde{P}$
  $\forall i, j,\ D_i^\ell = UDU(\tilde{X}_i^\ell)$, $\Delta_j^\ell = UDU(\tilde{Y}_j^\ell)$
  $\forall i, j,\ \hat{X}_i^\ell = t^A(D_i^\ell)$, $\hat{Y}_j^\ell = t^A(\Delta_j^\ell)$
  Compute $W_p^p(\frac{1}{n}\sum_{i=1}^n \delta_{\hat{X}_i^\ell}, \frac{1}{m}\sum_{j=1}^m \delta_{\hat{Y}_j^\ell})$
end for
Return $\frac{1}{L}\sum_{\ell=1}^L W_p^p(\frac{1}{n}\sum_{i=1}^n \delta_{\hat{X}_i^\ell}, \frac{1}{m}\sum_{j=1}^m \delta_{\hat{Y}_j^\ell})$

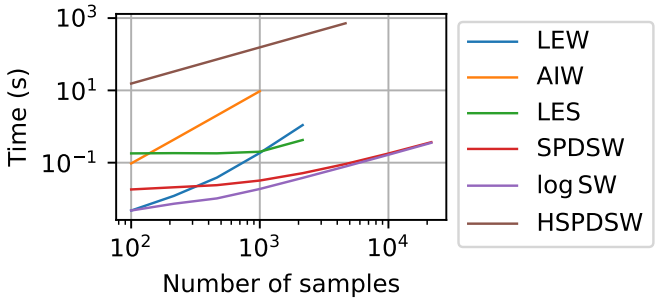


Figure G: Runtime of SPDSW, HPDSW and log SW (200 proj.) compared to alternatives based on Wasserstein between Wishart samples. Sliced discrepancies can scale to larger distributions in $S_d^{++}(\mathbb{R})$.

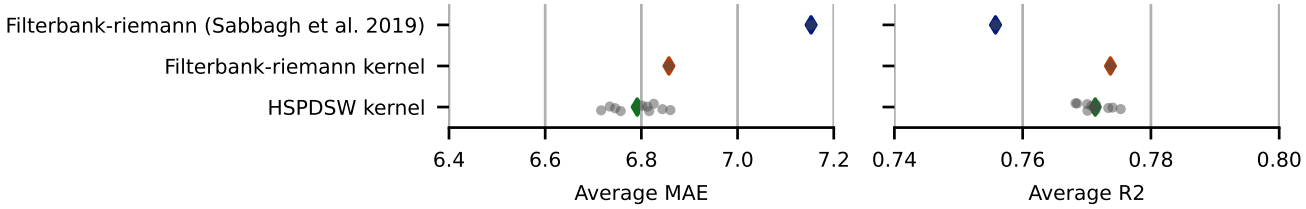


Figure H: Average MAE and R^2 score for 10 seeds on the Cam-CAN dataset with time-frames of 2s and 1000 projections. HSPDSW does not improve the performance of standard methods and the computation time is much higher than for SPDSW.

Moreover, it is also a Hilbertian pseudo-distance, thus the kernel is positive definite and well-defined for Kernel methods. Therefore, it can be easily adapted to brain age prediction, as done with SPDSW in Section 3.1. The Affine-Invariant metric is well-suited for problems involving source localization, as noted in Sabbagh et al. (2019). Even though we only have access to the Busemann coordinate, which might involve a loss of information due to the need of an additional projection, it is still of interest to compare to the Log-Euclidean metric. We report numerical results in the same setting as Figure 3 in Figure H. This time, the method does not beat Log-Euclidean Kernel Ridge regression based on the covariance matrices computed over all time samples. This suggests that the projection π_A derived in Appendix D.1 does not bring more information to the model in this scenario. Note that the computational cost suffers from the high complexity of the UDU decomposition needed for the calculation of each projection.