Riemannian optimization for non-centered mixture of scaled Gaussian distributions

Antoine Collas, Arnaud Breloy, Chengfang Ren, Guillaume Ginolhac, Jean-Philippe Ovarlez

Abstract—This paper studies the statistical model of the non-centered mixture of scaled Gaussian distributions (NC-MSG). Using the Fisher-Rao information geometry associated with this distribution, we derive a Riemannian gradient descent algorithm. This algorithm is leveraged for two minimization problems. The first is the minimization of a regularized negative log-likelihood (NLL). The latter makes the trade-off between a white Gaussian distribution and the NC-MSG. Conditions on the regularization are given so that the existence of a minimum to this problem is guaranteed without assumptions on the samples. Then, the Kullback-Leibler (KL) divergence between two NC-MSG is derived. This divergence enables us to define a second minimization problem. The latter is the computation of centers of mass of several NC-MSGs. Numerical experiments show the good performance and the speed of the Riemannian gradient descent on the two problems. Finally, a *Nearest centroïd classifier* is implemented leveraging the KL divergence and its associated center of mass. Applied on the large-scale dataset *Breizhcrops*, this classifier shows good accuracies and robustness to rigid transformations of the test set.

Index Terms—Non-centered mixture of scaled Gaussian distributions, Robust location and scatter estimation, Riemannian optimization, Fisher Information Metric, Classification, Kullback-Leibler divergence, Center of mass

I. INTRODUCTION

The first and second-order statistical moments of the sample set $\{x_i\}_{i=1}^n \in (\mathbb{R}^p)^n$ are ubiquitous features in signal processing and machine learning algorithms. Classically, these parameters are estimated using the empirical mean and the sample covariance matrix (SCM), which correspond to the maximum likelihood estimators (MLE) of the multivariate Gaussian model. However, these estimates tend to perform poorly in the context of heavy-tailed distributions, or when the sample set contains outliers. In such setups, one can obtain a better fit to empirical distributions by considering more general statistical models, such as the elliptical distributions [1]. Within this broad family of distributions, M-estimators of the location and scatter [2] appear as generalized MLEs and have been leveraged for their robustness properties in many fields (cf. [3] for an extensive review).

An important subfamily of elliptical distributions is the compound Gaussian distributions, which models samples as $x \overset{d}{=} \mu + \sqrt{\tau} u_i$, where $\mu \in \mathbb{R}^p$ is the center (also referred to as location) of the distribution, $u_i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ is

the *speckle* (centered Gaussian distribution with covariance matrix $\mathbf{\Sigma}$), and $\tau \in \mathbb{R}_*^+$ is an independent random scaling factor called the *texture*. The flexibility regarding the choice of the probability density function for τ results in various models for x. Compound Gaussian distributions encompass the t-distribution (including the Cauchy distribution), and the K-distribution. In practice, the underlying distribution is generally unknown, which is why the textures have often been modeled as unknown and deterministic in the centered case, *i.e.*, $x_i \sim \mathcal{N}(\mathbf{0}, \tau_i \mathbf{\Sigma})$. Such model will be referred to as *mixture of scaled Gaussian distributions* (MSG) [4]. The MLE of the scatter matrix $\mathbf{\Sigma}$ of this model coincides with Tyler's M-estimator of the scatter [5], which attracted considerable activity due to its robustness and distribution-free properties over the elliptical distributions family [6]–[9]. However, its transposition to the non-centered case from the model $x_i \sim \mathcal{N}(\mu, \tau_i \mathbf{\Sigma})$ received less interest. This might be because the usual fixed-point iterations to evaluate its maximum likelihood may diverge in practice, which motivated the present work.

In this paper, we tackle optimization problems related to parameter estimation and classification for a non-centered mixture of scaled Gaussian distributions (NC-MSG). The contribution is threefold:

First, we derive a new Riemannian gradient descent algorithm based on the Fisher-Rao information geometry of the considered statistical model. Indeed the parameter space (location, scatter, textures) is a product manifold that can be endowed with a Riemannian metric to leverage the Riemannian optimization framework [10], [11]. The Fisher-Rao information geometry corresponds to the one induced by the Fisher information metric. It is of particular interest since it is inherently well suited to the natural geometry of the data [12]. In this scope, we derive the Riemannian gradient (also referred to as the natural gradient) and a second-order retraction of this geometry. These tools are enough to cast a gradient descent applicable to any function of the parameters. We focus on two prominent examples that are regularized maximum likelihood estimation and center of mass computation. Simulations evidence that the proposed approach allows for fast computation of critical points, as it can converge with up to one order of magnitude less of iterations compared to other Riemannian descent approaches.

The second line of contributions concerns the problem of maximum likelihood estimation, for which we propose a new class of regularization penalties. A main issue with NC-MSGs is that the existence of the maximum likelihood is not guaranteed. This is due to attraction points where the likelihood

function diverges. This also explains why standard fixed-point algorithms to evaluate the solution may diverge in practice. Related issues are well known in the context of M-estimators because their existence is subject to strict conditions that are not always met in practice [2], [3], [5], for example when there is insufficient sample support ($n < p$). In such setups, it is common to rely on regularization penalties to ensure the existence of a solution, and the stability of corresponding iterative algorithms. In the centered case of elliptical distributions, several works considered shrinkage of M-estimators to a target scatter matrix [13]–[15], and regularizing both the mean and the scatter for the non-centered t-distribution was studied in [16]. Other regularizations formulated on the spectrum of the scatter matrix were proposed in [4], [17], [18] for the centered case. For NC-MSGs, we propose here a family of penalties that can be interpreted as a divergence between the initial model and a white Gaussian one (*i.e.*, that shrinks both the textures and eigenvalues of the scatter matrix to a pre-defined $\kappa \in \mathbb{R}_*^+$). We derive the general conditions for these penalties to ensure the existence of a solution for the regularized MLE. Interestingly, we show that this existence is only conditioned to the design of the penalty, and does not depend on the size of the sample support. We also study the invariance properties of the resulting estimators.

Finally, we apply the proposed algorithm to perform Riemannian classification. We consider the framework where statistical features of sample batches are used to discriminate between classes [19]–[22]. The Riemannian approach then consists in generalizing usual classification algorithms (*e.g.*, the *Nearest centroïd classifier*) by replacing the Euclidean distance and arithmetic mean by divergence and its corresponding center of mass [23]–[25]. In this setup, the information geometry can help design meaningful distances between the features, and improve the output performance [19], [22]. Unfortunately, the geodesic distance associated with the Fisher information metric of the NC-MSG remains unobtainable in closed-form (it is still unknown for the non-centered multivariate Gaussian model [26]–[28]). Instead, we propose to rely on the Kullback-Leibler (KL) divergence and its associated center of mass (computed using the proposed Riemannian optimization algorithm). We apply such Riemannian classification framework to the *Breizhcrops* dataset [29]. Our experiments evidence that regularizing the estimation greatly improves the accuracy. Thanks to the invariance properties of the proposed estimators, we also show that this process exhibits good robustness to rigid transformations of the samples during the inference.

The rest of the paper is organized as follows. Section II presents NC-MSGs and casts their parameter space as a manifold. Section III presents elements of Riemannian geometry, and studies the Fisher-Rao information geometry for this model. Section IV derives a Riemannian gradient descent algorithm following this geometry. Section V discusses parameter estimation in the considered model, presents a new class of regularized estimators, and studies some of their properties (existence, invariances). Section VI derives the KL divergence of the model and its associated center of mass. Section VII concludes with validation simulations and an application to Riemannian classification of the *Breizhcrops* dataset. For con-

ciseness, some technical proofs are in appendices that are provided as supplementary materials.

II. NON-CENTERED MIXTURE OF SCALED GAUSSIAN DISTRIBUTIONS AND ITS PARAMETER SPACE $\mathcal{M}_{p,n}$

A. Data model

Let a set of n data points $\{\boldsymbol{x}_i\}_{i=1}^n$ belonging to $\mathbb{R}^p$ and distributed according to the following statistical model

$$\boldsymbol{x}_i \overset{d}{=} \boldsymbol{\mu} + \sqrt{\tau_i}\,\boldsymbol{\Sigma}^{\frac{1}{2}}\,\boldsymbol{u}_i\,, \tag{1}$$

where $\boldsymbol{u}_i$ follows a white circular Gaussian distribution *i.e.* $\boldsymbol{u}_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_p)$. The variables $\boldsymbol{\mu} \in \mathbb{R}^p$ and $\boldsymbol{\Sigma} \in \mathcal{S}_p^{++}$ (set of $p \times p$ symmetric positive definite matrices) are respectively named the location and scatter parameters. Then, the unknown texture parameters $\{\tau_i\}_{i=1}^n$ are stacked into the vector $\boldsymbol{\tau} \in (\mathbb{R}_*^+)^n$ (set of strictly positive vectors). If these textures admit a probability density function (p.d.f.), then the random variables (r.v.) $\boldsymbol{x}_i$ follow a Compound Gaussian distribution [3], [30]. However, in general, this p.d.f. is unknown. In order not to rely on additional pdf assumptions on the textures, these are often assumed to be unkown and deterministic [7], [31]. In this case, the r.v. $\boldsymbol{x}_i$ follow a NC-MSG, *i.e.*

$$\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{\mu}, \tau_i \boldsymbol{\Sigma})\,. \tag{2}$$

Thus $\boldsymbol{x}_i$ admits a p.d.f. f defined from the Gaussian one f_G

$$f\left(\boldsymbol{x}_i | \left(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \tau_i\right)\right) = f_G\left(\boldsymbol{x}_i | \left(\boldsymbol{\mu}, \tau_i \boldsymbol{\Sigma}\right)\right)\,, \tag{3}$$

with $\forall \boldsymbol{x} \in \mathbb{R}^p$

$$f_G\left(\boldsymbol{x} | (\boldsymbol{\mu}, \boldsymbol{\Sigma})\right) =$$
$$(2\pi)^{-\frac{p}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu})\right]. \tag{4}$$

The negative log-likelihood (NLL) of the sample set $\{\boldsymbol{x}_i\}_{i=1}^n$ is then defined on the set of parameters $\theta = (\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\tau}) \in \mathbb{R}^p \times \mathcal{S}_p^{++} \times (\mathbb{R}_*^+)^n$ as (neglecting terms not depending on θ)

$$\mathcal{L}\left(\theta | \{\boldsymbol{x}_i\}_{i=1}^n\right) =$$
$$\frac{1}{2}\sum_{i=1}^n \left[\log|\tau_i \boldsymbol{\Sigma}| + \frac{(\boldsymbol{x}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu})}{\tau_i}\right]. \tag{5}$$

One can observe an ambiguity between the textures $\boldsymbol{\tau}$ and the scatter matrix $\boldsymbol{\Sigma}$. Indeed, $\forall \alpha > 0$, we have

$$\mathcal{L}\left(\left(\boldsymbol{\mu}, \alpha \boldsymbol{\Sigma}, \alpha^{-1} \boldsymbol{\tau}\right) | \{\boldsymbol{x}_i\}_{i=1}^n\right) =$$
$$\mathcal{L}\left(\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\tau}\right) | \{\boldsymbol{x}_i\}_{i=1}^n\right). \tag{6}$$

Thus, to identify the textures and scatter matrix parameters, a constraint on $\boldsymbol{\tau}$ or $\boldsymbol{\Sigma}$ can be added. Here the choice is made to constrain the textures by fixing their product to be equal to one, *i.e.* $\prod_{i=1}^n \tau_i = 1$. We point out that most of the results of the paper could be obtained by constraining the scatter matrix instead of the textures, with a unit determinant constraint, *i.e.* $|\boldsymbol{\Sigma}| = 1$ [32], [33]. Then, the parameter space of interest is

$$\mathcal{M}_{p,n} = \mathbb{R}^p \times \mathcal{S}_p^{++} \times \mathcal{S}(\mathbb{R}_*^+)^n\,, \tag{7}$$

where $\mathcal{S}(\mathbb{R}_*^+)^n$ is the set of textures with the unit product,

$$\mathcal{S}(\mathbb{R}_*^+)^n = \left\{ \boldsymbol{\tau} \in (\mathbb{R}_*^+)^n : \prod_{i=1}^n \tau_i = 1 \right\}. \qquad (8)$$

The choice of adding a constraint is motivated by two results additional to the identifiability: *(i)* it reduces the dimension of the parameter space by removing the indeterminacy (6), *(ii)* the associated FIM (see Proposition 1) admits a simpler expression, which will be instrumental in the rest of the paper as it turns $\mathcal{M}_{p,n}$ into a Riemannian manifold. Its simple formula could not have been obtained without adding this constraint (either on $\boldsymbol{\tau}$ or its counterpart on $\boldsymbol{\Sigma}$).

B. Related works

When $\{\boldsymbol{x}_i\}_{i=1}^n$ is sampled from an underlying heavy-tailed Compound Gaussian distribution, the empirical mean and SCM do not provide robust and accurate estimates of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. In this setup, M-estimators [2], raised increasing interest in the past decades (see e.g. [3]). These estimators are expressed through the two joint fixed-point equations

$$\boldsymbol{\mu} = \left(\sum_{i=1}^n u_1(t_i) \right)^{-1} \sum_{i=1}^n u_1(t_i) \boldsymbol{x} \triangleq \mathcal{H}_{\boldsymbol{\mu}}(\boldsymbol{\mu}, \boldsymbol{\Sigma}),$$
$$\boldsymbol{\Sigma} = \frac{1}{n} \sum_{i=1}^n u_2(t_i)(\boldsymbol{x}-\boldsymbol{\mu})(\boldsymbol{x}-\boldsymbol{\mu})^T \triangleq \mathcal{H}_{\boldsymbol{\Sigma}}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \qquad (9)$$

where $t_i \triangleq (\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})$, u_1 and u_2 are functions that respect Maronna's conditions[1] [2]. Under certain conditions [2], these estimators can be computed with fixed-point iterations

$$\boldsymbol{\mu}_{k+1} = \mathcal{H}_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k),$$
$$\boldsymbol{\Sigma}_{k+1} = \mathcal{H}_{\boldsymbol{\Sigma}}(\boldsymbol{\mu}_{k+1}, \boldsymbol{\Sigma}_k), \qquad (10)$$

that converge towards a unique solution satisfying (9). Interestingly, some M-estimators also appear as MLE when $u_1(t) = u_2(t)$ is linked to the p.d.f. of an elliptical distribution [3]. Expressing these estimators as the solution of an optimization problem drove a more recent line of work leveraging optimization theory allowing, e.g., for generalizations to structured scatter matrix matrices [34]–[36] or regularized location and scatter matrix estimation [16].

In the context of scatter matrix estimation, Tyler's M-estimator [5] is especially interesting thanks to its robustness and "distribution-free" properties over the elliptical distributions family. Tyler's M-estimator is obtained for $\boldsymbol{\mu} = \boldsymbol{0}$ and $u_2(t) = p/t$, and also coincide with the MLE of the centered MSG [6], [7]. However, this estimator cannot trivially be transposed to the case of joint mean-scatter matrix estimation. Indeed, the MLE solution associated with NC-MSG is obtained with $u_1(t) = u_2(t) = p/t$, which does not satisfy Maronna's conditions [2], and for which the fixed-point iterations (10) generally diverge. Thus, Tyler's M-estimator of the scatter matrix is usually applied on demeaned data, where

the mean is estimated in a prior step[2]. It was yet experienced that the MLE of NC-MSG could be evaluated in practice with Riemannian optimization rather than potentially unstable fixed-point iterations in [37] (still, without any theoretical guarantees). The following of this paper builds upon this finding in several directions: optimization in sections III-B and IV, regularized estimation with theoretical guarantees in section V, and classification in sections VI and VII.

III. RIEMANNIAN GEOMETRY OF $\mathcal{M}_{p,n}$

The objective of this section is to present the information geometry of the NC-MSG (2); *i.e.* the Riemannian geometry of $\mathcal{M}_{p,n}$ with the FIM as a Riemannian metric. This Riemannian geometry is leveraged to optimize several cost functions $h : \mathcal{M}_{p,n} \to \mathbb{R}$. Notably, two cost functions will be studied: a regularized NLL in Section V, and a cost function to compute centers of mass of sets of points $\{\theta_i\} \subset \mathcal{M}_{p,n}$ in Section VI. Before turning $\mathcal{M}_{p,n}$ into a Riemannian manifold, a brief introduction to Riemannian geometry is made. For a complete introduction to the topic, see [10], [11].

A. Riemannian geometry

Let $\mathcal{E}$ be a *linear space* of dimension d. Informally, a *smooth embedded manifold* $\mathcal{M} \subset \mathcal{E}$ of dimension $l \leq d$ is a nonempty set that locally resembles a l-dimensional linear space. Indeed, $\mathcal{M}$ is a smooth embedded manifold of $\mathcal{E}$ if and only if it is locally diffeomorphic[3] with open sets of a l-dimensional linear subspace in $\mathbb{R}^d$. Then, *smooth curves* c are smooth functions from open intervals I of $\mathbb{R}$ to $\mathcal{M}$; *i.e.* $c : I \to \mathcal{M}$. Collecting velocities of the curves passing through $x \in \mathcal{M}$, we get the *tangent space* at x:

$$T_x\mathcal{M} = \{c'(0) \,|\, c : I \to \mathcal{M} \text{ is smooth and } c(0) = x\}. \quad (11)$$

This tangent space corresponds to a linearization of $\mathcal{M}$ at x. The tangent bundle of $\mathcal{M}$ is then the disjoint union of all the tangent spaces of $\mathcal{M}$, i.e., $T\mathcal{M} = \{(x, \xi) : x \in \mathcal{M} \text{ and } \xi \in T_x\mathcal{M}\}$.

So far, we have defined the notion of the smooth embedded manifold of a linear space. To turn $\mathcal{M}$ into a *Riemannian manifold*, its tangent spaces $T_x\mathcal{M}$ are equipped with a *Riemannian metric* which is an *inner product*[4] $\langle .,. \rangle_x^{\mathcal{M}} : T_x\mathcal{M} \times T_x\mathcal{M} \to \mathbb{R}$ that varies smoothly with respect to x.[5]

Then, to move on $\mathcal{M}$, a *geodesic* is a smooth curve on $\mathcal{M}$ with zero acceleration along its path. In a *Euclidean space* $\mathcal{E}$ the acceleration is classically defined as the second derivative. Thus, a geodesic $c : I \to \mathcal{E}$ is such that $\ddot{\gamma}(t) = 0 \; \forall t \in I$. If $\gamma(0) = x$ and $\dot{\gamma}(0) = \xi$, then, by integrating, we recover the

---

[1]Notice that [2] rather uses a formulation of (9) involving "$u_1(t_i)$" and "$u_2(t_i^2)$", with $t_i = \sqrt{(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}$. Without loss of generality, this paper uses the present notation to simplify some discussions.

[2]We point out that a closely related estimator proposed in [5] uses $u_1(t) = \sqrt{p/t}$ and $u_2(t) = p/t$, which yields converging fixed-point iterations in practice despite being a limit case of Maronna's conditions. This estimate, however, is not obtained as the solution of an underlying optimization problem, i.e., has no MLE interpretation.

[3]A *diffeomorphism* is a bijective map $f : U \to V$ where U, V are open sets and such that both f and f^{-1} are smooth (or infinitely differentiable).

[4]An *inner product* is a bilinear, symmetric, positive definite function on a $\mathbb{R}$-vector space.

[5]For all smooth vector fields ξ, η on $\mathcal{M}$ the function $x \mapsto \langle \xi, \eta \rangle_x^{\mathcal{M}}$ is smooth.

classical straight line $\gamma(t) = x + t\,\xi$. This notion of acceleration is generalized to manifolds using the *Levi-Civita connection* denoted by ∇. This notion requires first defining smooth vector fields, which are smooth mappings that associate a vector in $T\mathcal{M}$ for each point of the manifold $\mathcal{M}$, i.e.:

$$\xi: \begin{array}{ccc} \mathcal{M} & \rightarrow & T\mathcal{M} \\ x & \mapsto & \xi(x). \end{array} \qquad (12)$$

Notice that given this definition, $\xi(x) \in T_x\mathcal{M} \ \forall \ x \in \mathcal{M}$, so we also use the symbol ξ (respectively η) to denote a tangent vector when there is no ambiguity. Now, the Levi-Civita connection itself is defined as an operator that generalizes the directional derivative of vectors fields to Riemannian manifolds, and associates to every couple of smooth vector fields (ξ, η) on $\mathcal{M}$ a new vector field $\nabla_\xi \eta$ on $\mathcal{M}$. Given a Riemannian manifold $\mathcal{M}$, the Levi-Civita connection is unique and defined by the *Koszul* formula. It should be noted that the Levi-Civita connection depends on the chosen Riemannian metric. Using this object, a geodesic $\gamma : I \rightarrow \mathcal{M}$ with initial conditions $\gamma(0) = x$ and $\dot{\gamma}(0) = \xi$ is defined as a smooth curve having zero acceleration as defined by the Levi-Civita connection

$$\nabla_{\dot{\gamma}(t)}\dot{\gamma}(t) = 0_{\gamma(t)}, \quad \forall t \in I \qquad (13)$$

where $\dot{\gamma}(t) = \frac{d}{dt}\gamma(t)$ and $0_{\gamma(t)}$ is the zero element of $T_{\gamma(t)}\mathcal{M}$. Let γ be a geodesic defined on $[0,1]$ with initial conditions $\gamma(0) = x$ and $\dot{\gamma}(0) = \xi$. Then, the *Riemannian exponential mapping* $\exp_x^{\mathcal{M}} : T_x\mathcal{M} \rightarrow \mathcal{M}$ at $x \in \mathcal{M}$ is defined as $\exp_x^{\mathcal{M}}(\xi) = \gamma(1)$. For $x, y \in \mathcal{M}$, its inverse function, the *Riemannian logarithm mapping*, is defined as $\log_x^{\mathcal{M}}(y) = \arg\min_{\xi \in T_x\mathcal{M}} \|\xi\|_x^2$ subject to $\exp_x^{\mathcal{M}}(\xi) = y$ with $\|\xi\|_x^2 = \langle \xi, \xi \rangle_x^{\mathcal{M}}$. Finally, the *Riemannian distance* between two points $x, y \in \mathcal{M}$ is computed as $d_{\mathcal{M}}(x,y) = \|\log_x^{\mathcal{M}}(y)\|_x$.

B. Description of the Riemannian manifold $\mathcal{M}_{p,n}$

This subsection gives the Riemannian structure, induced by the FIM, of the parameter set $\mathcal{M}_{p,n}$. To specify the latter, we begin by defining the ambient space

$$\mathcal{E}_{p,n} = \mathbb{R}^p \times \mathbb{R}^{p \times p} \times \mathbb{R}^n. \qquad (14)$$

Therefore, the tangent space of $\mathcal{M}_{p,n}$ at θ is a subspace of the ambient space $\mathcal{E}_{p,n}$

$$T_\theta \mathcal{M}_{p,n} = \left\{ \xi = (\boldsymbol{\xi_\mu}, \boldsymbol{\xi_\Sigma}, \boldsymbol{\xi_\tau}) \in \mathbb{R}^p \times \mathcal{S}_p \times \mathbb{R}^n : \right.$$
$$\left. \boldsymbol{\xi_\tau}^T \boldsymbol{\tau}^{\odot -1} = 0 \right\}, \quad (15)$$

where $\mathcal{S}_p$ is the set of symmetric matrices and $.^{\odot -1}$ is the elementwise inverse operator. To turn $\mathcal{M}_{p,n}$ into a Riemannian manifold, we must equip $\mathcal{M}_{p,n}$ with a Riemannian metric. Many possibilities are available to us, however, a preferable one is the FIM [38] derived in Proposition 1. Indeed, it is calculated using the NLL (5) and thus is associated with the statistical model (1).

Proposition 1 (Fisher Information Metric). *Let $\theta \in \mathcal{M}_{p,n}$ and $\xi, \eta \in T_\theta\mathcal{M}_{p,n}$, the Fisher Information Metric at θ associated with the NLL (5) is*

$$\langle \xi, \eta \rangle_\theta^{\mathcal{M}_{p,n}} = \sum_{i=1}^n \left(\frac{1}{\tau_i}\right) \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta_\mu} + \frac{n}{2} \operatorname{Tr}\left(\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma}\right)$$
$$+ \frac{p}{2}(\boldsymbol{\xi_\tau} \odot \boldsymbol{\tau}^{\odot -1})^T (\boldsymbol{\eta_\tau} \odot \boldsymbol{\tau}^{\odot -1}),$$

where $\odot$ is the elementwise product operator.

Proof. See supplementary material A. $\qquad \square$

Then, the orthogonal projection according to the FIM from $\mathcal{E}_{p,n}$ onto $T_\theta\mathcal{M}_{p,n}$ is given in Proposition 2.

Proposition 2 (Orthogonal projection). *The orthogonal projection associated with the FIM of Proposition 1 from $\mathcal{E}_{p,n}$ onto $T_\theta\mathcal{M}_{p,n}$ is*

$$P_\theta^{\mathcal{M}_{p,n}}(\xi) = \left(\boldsymbol{\xi_\mu}, \operatorname{sym}(\boldsymbol{\xi_\Sigma}), \boldsymbol{\xi_\tau} - \frac{\boldsymbol{\xi_\tau}^T \boldsymbol{\tau}^{\odot -1}}{n} \boldsymbol{\tau} \right),$$

where $\operatorname{sym}(\boldsymbol{\xi}) = \frac{1}{2}\left(\boldsymbol{\xi} + \boldsymbol{\xi}^T\right)$.

Proof. See supplementary material B. $\qquad \square$

The orthogonal projection proves helpful to derive elements in tangent spaces such as the Riemannian gradient or the Levi-Civita connection. The latter is given for the manifold $\mathcal{M}_{p,n}$ in Proposition 3.

Proposition 3 (Levi-Civita connection). *Let $\theta \in \mathcal{M}_{p,n}$ and ξ, η be smooth vector fields of $\mathcal{M}_{p,n}$, the Levi-Civita connection of $\mathcal{M}_{p,n}$ evaluated at θ is*

$$\nabla_\xi \eta = P_\theta^{\mathcal{M}_{p,n}}\left(\overline{\nabla}_\xi \eta\right),$$

where

$$\overline{\nabla}_\xi \eta = \operatorname{D}\eta[\xi] + \left(-\frac{1}{2}\left[\left(\frac{\boldsymbol{\xi_\tau}^T \boldsymbol{\tau}^{\odot -2}}{\sum_{i=1}^n \frac{1}{\tau_i}} \boldsymbol{I}_p + \boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\right) \boldsymbol{\eta_\mu} \right. \right.$$
$$\left. + \left(\frac{\boldsymbol{\eta_\tau}^T \boldsymbol{\tau}^{\odot -2}}{\sum_{i=1}^n \frac{1}{\tau_i}} \boldsymbol{I}_p + \boldsymbol{\eta_\Sigma}\boldsymbol{\Sigma}^{-1}\right) \boldsymbol{\xi_\mu} \right],$$
$$\frac{1}{n}\sum_{i=1}^n \left(\frac{1}{\tau_i}\right) \boldsymbol{\eta_\mu}\boldsymbol{\xi_\mu}^T - \boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma},$$
$$\left. \frac{1}{p}\boldsymbol{\xi_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\mu}\mathbb{1}_n - \boldsymbol{\xi_\tau} \odot \boldsymbol{\eta_\tau} \odot \boldsymbol{\tau}^{\odot -1}\right).$$

Proof. See supplementary material C. $\qquad \square$

As detailed in Subsection III-A the Levi-Civita connection defines geodesics on a Riemannian manifold. Indeed, for I an open interval of $\mathbb{R}$, a geodesic $\gamma : I \rightarrow \mathcal{M}_{p,n}$ with initial position $\gamma(0) = \theta \in \mathcal{M}_{p,n}$ and initial velocity $\dot{\gamma}(0) = \xi \in T_\theta\mathcal{M}_{p,n}$ must respect

$$\nabla_{\dot{\gamma}(t)}\dot{\gamma}(t) = 0_{\gamma(t)}, \quad \forall t \in I. \qquad (16)$$

However, an analytical solution of (16) remains unknown. A retraction (approximation of the geodesic) can still be obtained (see Proposition 5) which allows us to optimize functions on

---

Algorithm 1: Riemannian gradient descent on $\mathcal{M}_{p,n}$

Input: Initialization $\theta^{(0)} \in \mathcal{M}_{p,n}$
Output: Iterates $\theta^{(k)} \in \mathcal{M}_{p,n}$
for $k = 0$ *to convergence* **do**
  Compute a step-size α using Algorithm 2
  Set $\theta^{(k+1)} \leftarrow R_{\theta^{(k)}}^{\mathcal{M}_{p,n}}\left(-\alpha \operatorname{grad}_{\mathcal{M}_{p,n}} h(\theta^{(k)})\right)$

Algorithm 2: Riemannian backtracking on $\mathcal{M}_{p,n}$

Input: Current iterate $\theta^{(k)} \in \mathcal{M}_{p,n}$, and constants
  $\alpha \in]0, t_{\max}[$, $c \in]0,1[$ and $\varepsilon \in \mathbb{R}_*^+$
Output: Step-size α
Set $\theta(\alpha) \leftarrow R_{\theta^{(k)}}^{\mathcal{M}_{p,n}}\left(-\alpha \operatorname{grad}_{\mathcal{M}_{p,n}} h(\theta^{(k)})\right)$
while $h(\theta^{(k)}) - h(\theta(\alpha)) < \varepsilon\alpha||\operatorname{grad}_{\mathcal{M}_{p,n}} h(\theta^{(k)})||_{\theta^{(k)}}^2$ **do**
  Set $\alpha \leftarrow c\,\alpha$
  Set $\theta(\alpha) \leftarrow R_{\theta^{(k)}}^{\mathcal{M}_{p,n}}\left(-\alpha \operatorname{grad}_{\mathcal{M}_{p,n}} h(\theta^{(k)})\right)$

$\mathcal{M}_{p,n}$. Moreover, the geodesic between two points θ_1 and θ_2 is unknown. This implies that the geodesic distance is also unknown. This is not surprising since the geodesic and the Riemannian distance between two Gaussian distributions with different locations are unknown [26]–[28], [39], [40]. To alleviate this problem, a divergence associated with the NC-MSG (2) is proposed in Section VI.

IV. RIEMANNIAN OPTIMIZATION ON $\mathcal{M}_{p,n}$

The objective of this subsection is to propose tools to perform optimization on the Riemannian manifold $\mathcal{M}_{p,n}$. Indeed, we aim to minimize smooth functions $h : \mathcal{M}_{p,n} \to \mathbb{R}$,

$$\underset{\theta \in \mathcal{M}_{p,n}}{\text{minimize}} \ h(\theta). \tag{17}$$

An example of such a function is the NLL (5). As mentioned in Section III, two additional cost functions are studied in Sections V and VI. To realize (17), we consider a Riemannian steepest gradient descent on $\mathcal{M}_{p,n}$. Only the tools required for this algorithm are derived here. For a detailed introduction to optimization on Riemannian manifolds, see [10], [11]. Two optimization tools are needed: *(i)* the Riemannian gradient of h, *(ii)* a retraction that maps tangent vectors from $T_\theta \mathcal{M}_{p,n}$ $\forall \theta \in \mathcal{M}_{p,n}$ onto $\mathcal{M}_{p,n}$. Once these are defined, the Riemannian steepest gradient descent retracts iteratively minus the gradient of h times a step size onto the manifold.

We begin with the Riemannian gradient of h at θ. For every $\theta \in \mathcal{M}_{p,n}$, it is defined through the Riemannian metric as the unique tangent vector in $T_\theta \mathcal{M}_{p,n}$ such that, $\forall \xi \in T_\theta \mathcal{M}_{p,n}$,

$$\mathrm{D}\, h(\theta)[\xi] = \langle \operatorname{grad}_{\mathcal{M}_{p,n}} h(\theta), \xi \rangle_\theta^{\mathcal{M}_{p,n}}, \tag{18}$$

where $\mathrm{D}\, h(\theta)[\xi]$ is the directional derivative of h at θ in the direction ξ. In the case where for every $\theta \in \mathcal{M}_{p,n}$, there exists an open U of $\mathcal{E}_{p,n}$, with $\theta \in U$, and a differentiable function $\bar{h} : U \to \mathbb{R}$ such that $\bar{h}$ restricted to $\mathcal{M}_{p,n}$ is equal to h, this Riemannian gradient can be computed from the Euclidean gradient of h at θ. In particular, this assumption is

met by the different cost functions considered in the rest of the manuscript and the transformation of the Euclidean gradient into the Riemannian one is given in Proposition 4. The latter is very convenient since this Euclidean gradient can be computed using automatic differentiation libraries such as Autograd [41] or JAX [42].

Proposition 4 (Riemannian gradient). *Let $\theta \in \mathcal{M}_{p,n}$ and h be a real-valued function defined on $\mathcal{M}_{p,n}$. The Riemannian gradient of h at θ is*

$$\operatorname{grad}_{\mathcal{M}_{p,n}} h(\theta) =$$
$$P_\theta^{\mathcal{M}_{p,n}}\left(\left(\sum_{i=1}^n \frac{1}{\tau_i}\right)^{-1} \boldsymbol{\Sigma} \boldsymbol{G}_{\boldsymbol{\mu}}, \frac{2}{n} \boldsymbol{\Sigma} \boldsymbol{G}_{\boldsymbol{\Sigma}} \boldsymbol{\Sigma}, \frac{2}{p} \boldsymbol{\tau}^{\odot 2} \odot \boldsymbol{G}_{\boldsymbol{\tau}} \right),$$

where $\operatorname{grad} h(\theta) = (\boldsymbol{G}_{\boldsymbol{\mu}}, \boldsymbol{G}_{\boldsymbol{\Sigma}}, \boldsymbol{G}_{\boldsymbol{\tau}})$ is the Euclidean gradient of h in $\mathbb{R}^p \times \mathbb{R}^{p \times p} \times \mathbb{R}^n$.

Proof. See supplementary material D. $\qquad\square$

Then, it remains to define a retraction for every θ on $\mathcal{M}_{p,n}$. A retraction $R_\theta^{\mathcal{M}_{p,n}}$ maps every $\xi \in T_\theta \mathcal{M}_{p,n}$ to a point $R_\theta^{\mathcal{M}_{p,n}}(\xi) \in \mathcal{M}_{p,n}$ and is such that $R_\theta^{\mathcal{M}_{p,n}}(\xi) = \theta + \xi + o(\|\xi\|)$. Several retractions could be obtained from this definition. Furthermore, it should be noted that a map respecting this definition is not necessarily related to the Riemannian metric of $\mathcal{M}_{p,n}$. Thus, we choose to enforce an additional property: the desired retraction must have a zero initial acceleration, *i.e.*

$$\nabla_{\dot{r}(t)} \dot{r}(t)\Big|_{t=0} = 0, \tag{19}$$

where $\dot{r}(t) = \frac{d}{dt} R_\theta^{\mathcal{M}_{p,n}}(t\xi)$ and ∇ is the Levi-Civita connection from Proposition 3. Such a retraction is called a second-order retraction. Furthermore, the property of zero initial acceleration is linked to the definition of the geodesic. Indeed, a geodesic has a zero acceleration $\forall t$ along its path (see (13)) whereas here this condition is only needed at $t = 0$. By respecting this property, the retraction is associated with the Riemannian metric of Proposition 1 since the Levi-Civita connection is derived from this Riemannian metric. Such a retraction is presented in Proposition 5.

Proposition 5 (Second order retraction). *Let $\theta \in \mathcal{M}_{p,n}$ and $\xi \in T_\theta \mathcal{M}_{p,n}$. There exists $t_{max} > 0$ (specified in the Supplementary material) such that $\forall t \in [0, t_{max}[$, a second order retraction on $\mathcal{M}_{p,n}$ at θ is*

$$R_\theta^{\mathcal{M}_{p,n}}(t\xi) = \left(\boldsymbol{\mu} + t\boldsymbol{\xi}_{\boldsymbol{\mu}} + \frac{t^2}{2} \left[\frac{\boldsymbol{\xi}_{\boldsymbol{\tau}}^T \boldsymbol{\tau}^{\odot -2}}{\sum_{i=1}^n \frac{1}{\tau_i}} \boldsymbol{I}_p + \boldsymbol{\xi}_{\boldsymbol{\Sigma}} \boldsymbol{\Sigma}^{-1} \right] \boldsymbol{\xi}_{\boldsymbol{\mu}}, \right.$$

$$\boldsymbol{\Sigma} + t\boldsymbol{\xi}_{\boldsymbol{\Sigma}} + \frac{t^2}{2}\left(\boldsymbol{\xi}_{\boldsymbol{\Sigma}} \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi}_{\boldsymbol{\Sigma}} - \frac{1}{n}\sum_{i=1}^n \left(\frac{1}{\tau_i}\right) \boldsymbol{\xi}_{\boldsymbol{\mu}} \boldsymbol{\xi}_{\boldsymbol{\mu}}^T \right),$$

$$N\left(\boldsymbol{\tau} + t\boldsymbol{\xi}_{\boldsymbol{\tau}} + \frac{t^2}{2}\left(\boldsymbol{\xi}_{\boldsymbol{\tau}}^{\odot 2} \odot \boldsymbol{\tau}^{\odot -1} - \frac{1}{p} \boldsymbol{\xi}_{\boldsymbol{\mu}}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi}_{\boldsymbol{\mu}} \mathbb{1}_n \right) \right) \bigg),$$

where $\forall \boldsymbol{x} = (x_i)_{1 \leq i \leq n} \in (\mathbb{R}_*^+)^n$, $N(\boldsymbol{x}) = \left(\prod_{i=1}^n x_i\right)^{-\frac{1}{n}} \boldsymbol{x}$.

Proof. See supplementary material E. $\qquad\square$

With this retraction and the Riemannian gradient from Proposition 4, we have all the tools required to derive a Riemannian steepest descent. The latter is presented in Algorithm 1. It should be noted that, in practice, the step size is chosen using a backtracking algorithm [11, Ch. 4]. Given an initial step-size $\alpha \in]0, t_{\max}[$ with $t_{\max}$ defined in Proposition 5, the algorithm reduces α by a factor $c \in]0, 1[$ until the Armijo–Goldstein condition is satisfied. Given $\varepsilon \in \mathbb{R}_*^+$ (generally fixed at 10^{-4}) and the tentative next iterate

$$\theta(\alpha) = R_{\theta^{(k)}}^{\mathcal{M}_{p,n}} \left(-\alpha \operatorname{grad}_{\mathcal{M}_{p,n}} h(\theta^{(k)}) \right), \qquad (20)$$

the Armijo–Goldstein condition writes

$$h(\theta^{(k)}) - h(\theta(\alpha)) \geq \varepsilon \alpha \left\| \operatorname{grad}_{\mathcal{M}_{p,n}} h(\theta^{(k)}) \right\|_{\theta^{(k)}}^2. \qquad (21)$$

This procedure is presented in Algorithm 2.

V. PARAMETER ESTIMATION OF THE NON-CENTERED MIXTURE OF SCALED GAUSSIAN DISTRIBUTIONS

In the previous subsection, tools to perform optimization on $\mathcal{M}_{p,n}$ have been developed. In this subsection, the objective is to leverage these tools to estimate the parameters of an NC-MSG (2). In the following, we assume having $n \geq 1$ data points $\{\boldsymbol{x}_i\}_{i=1}^n \subset \mathbb{R}^p$. The estimation of the parameters of the statistical model (2) is performed by maximizing the associated likelihood on $\mathcal{M}_{p,n}$:

$$\underset{\theta \in \mathcal{M}_{p,n}}{\operatorname{minimize}} \mathcal{L}\left(\theta | \{\boldsymbol{x}_i\}_{i=1}^n\right), \qquad (22)$$

where $\mathcal{L}$ is the NLL (5). However, the existence of a solution to this problem is not guaranteed. To build intuition, we present a short example of a problematical case where $\boldsymbol{\mu}$ gets attracted by one data point $\boldsymbol{x}_j$. Let k be the current iteration of a given optimizer of (22). For $k \to +\infty$, if $\boldsymbol{\mu}^{(k)} \to \boldsymbol{x}_j$ faster than $\tau_j^{(k)} \to 0$ and $\forall i \neq j$, $\tau_i^{(k)} \to +\infty$, then the quadratic form in $\mathcal{L}$ (5) tends to zero, which is its minimum,

$$\sum_{i=1}^n \frac{(\boldsymbol{x}_i - \boldsymbol{\mu}^{(k)})^T \left(\boldsymbol{\Sigma}^{(k)}\right)^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}^{(k)})}{\tau_i^{(k)}} \xrightarrow[k \to +\infty]{} 0. \qquad (23)$$

Then, if an eigenvalue $\lambda^{(k)}$ of $\boldsymbol{\Sigma}^{(k)}$ tends 0 slower than the respective limits of $\boldsymbol{\mu}^{(k)}$, $\tau_i^{(k)}$ and $\tau_j^{(k)}$ and since $\sum_{i=1}^n \log |\tau_i \boldsymbol{\Sigma}| = n \log |\boldsymbol{\Sigma}|$, we obtain

$$\mathcal{L}(\theta^{(k)} | \{\boldsymbol{x}_i\}_{i=1}^n) \xrightarrow[k \to +\infty]{} -\infty. \qquad (24)$$

Hence, depending on the data points $\{\boldsymbol{x}_i\}_{i=1}^n$, a solution of the problem (22) does not necessarily exist.

To overcome this issue, we propose a regularization approach to the NLL. Firstly, we prove that this allows the existence of a solution depending on some assumptions on the regularization term in V-A. Some interpretations on the chosen regularization are next given in V-B, and finally, we study the robustness of the solution to rigid transformations in V-C.

A. Existence of solution with a regularized version of the NLL

We present a regularized version of the NLL (5):

$$\mathcal{L}_{\mathcal{R}_\kappa}\left(\theta | \{\boldsymbol{x}_i\}_{i=1}^n\right) = \mathcal{L}\left(\theta | \{\boldsymbol{x}_i\}_{i=1}^n\right) + \beta \mathcal{R}_\kappa(\theta), \qquad (25)$$

where $\beta \in \mathbb{R}_*^+$ and $\mathcal{R}_\kappa : \mathcal{M}_{p,n} \to \mathbb{R}$ is a regularization. Thus, the minimization problem (22) becomes

$$\underset{\theta \in \mathcal{M}_{p,n}}{\operatorname{minimize}} \mathcal{L}_{\mathcal{R}_\kappa}\left(\theta | \{\boldsymbol{x}_i\}_{i=1}^n\right). \qquad (26)$$

Though (26) is a generic formulation, we will focus on several proposals that ensure the existence of a solution. The proposed approach is to rewrite $\mathcal{R}_\kappa$ as a sum of regularizations r_κ on the eigenvalues of $\tau_i \boldsymbol{\Sigma}$. This rewriting is formalized in Assumption 1.

Assumption 1. *The regularization $\mathcal{R}_\kappa$ is a positive function that is a sum of regularizations on the eigenvalues of $\tau_i \boldsymbol{\Sigma}$*

$$\mathcal{R}_\kappa(\theta) = \sum_{i=1}^n \sum_{j=1}^p r_\kappa(\tau_i \lambda_j),$$

where $\lambda_j \in \mathbb{R}_*^+$ are the eigenvalues of $\boldsymbol{\Sigma}$ and $r_\kappa : \mathbb{R}_*^+ \to \mathbb{R}$ is a continuous function.

In the following, we assume that $\mathcal{R}_\kappa$ respects Assumption 1. To prevent the eigenvalues of $\tau_i \boldsymbol{\Sigma}$ from taking values that are too large or too small, a second Assumption is added. Indeed, Assumption 2 states that the regularization of the log function by the penalty function r_κ goes to infinite when its argument goes to 0^+ or $+\infty$. This assumption is made so that if an eigenvalue of $\tau_i \boldsymbol{\Sigma}$ tends to 0^+ or $+\infty$ then $\mathcal{L}_{\mathcal{R}_\kappa} \to +\infty$.

Assumption 2. *The following function admits the limit $\forall \beta \in \mathbb{R}_*^+$*

$$\lim_{x \to \partial \mathbb{R}_*^+} \log(x) + \beta r_\kappa(x) = +\infty, \qquad (27)$$

with $\partial \mathbb{R}_*^+$ is a border of $\mathbb{R}_*^+$, i.e. 0^+ or $+\infty$.

Assumptions 1 and 2 are sufficient to solve the problem of existence stated earlier. Indeed, when $\mathcal{R}_\kappa$ respects these assumptions, Proposition 6 states that the problem (26) has a solution, *i.e.* $\mathcal{L}_{\mathcal{R}_\kappa}$ admits a minimum in $\mathcal{M}_{p,n}$. Finally, Assumptions 1 and 2 are quite easy to meet in practice. Indeed, several regularizations respecting these assumptions are proposed in Table I.

Proposition 6 (Existence). *Under Assumptions 1 and 2, and $\forall \beta \in \mathbb{R}_*^+$, the regularized NLL*

$$\theta \mapsto \mathcal{L}_{\mathcal{R}_\kappa}\left(\theta | \{\boldsymbol{x}_i\}_{i=1}^n\right) = \mathcal{L}\left(\theta | \{\boldsymbol{x}_i\}_{i=1}^n\right) + \beta \mathcal{R}_\kappa(\theta),$$

with $\mathcal{L}$ being the NLL (5), admits a minimum in $\mathcal{M}_{p,n}$.

Proof. $\mathcal{L}_{\mathcal{R}_\kappa}$ is a continuous function on $\mathcal{M}_{p,n}$. Hence, to prove the existence of a solution to the minimization problem (26), it is enough to show that for sequences $\theta^{(k)} \to \partial\theta$, the boundary of $\mathcal{M}_{p,n}$, we have that

$$\lim_{k \to +\infty} \mathcal{L}_{\mathcal{R}_\kappa}(\theta^{(k)} | \{\boldsymbol{x}_i\}_{i=1}^n) = +\infty. \qquad (28)$$

First, we handle the cases where $\left\| \boldsymbol{\mu}^{(k)} \right\|_2 \nrightarrow +\infty$. Since $\theta^{(k)} \to \partial\theta$, this means that, at least, one $\lambda_j^{(k)} \to \partial \mathbb{R}_*^+$ and/or

Name	$\mathcal{R}_\kappa(\theta)$	$r_\kappa(x)$
L1 penalty	$\left\|(\mathrm{diag}(\boldsymbol{\tau}) \otimes \boldsymbol{\Sigma})^{-1} - \kappa^{-1}\boldsymbol{I}_{np}\right\|_1 = \sum_{i,j}\left\|(\tau_i\lambda_j)^{-1} - \kappa^{-1}\right\|$	$\|x^{-1} - \kappa^{-1}\|$
L2 penalty	$\left\|(\mathrm{diag}(\boldsymbol{\tau}) \otimes \boldsymbol{\Sigma})^{-1} - \kappa^{-1}\boldsymbol{I}_{np}\right\|_2^2 = \sum_{i,j}\left((\tau_i\lambda_j)^{-1} - \kappa^{-1}\right)^2$	$(x^{-1} - \kappa^{-1})^2$
Bures-Wasserstein squared distance	$d_{\mathrm{BW}}^2\left((\mathrm{diag}(\boldsymbol{\tau}) \otimes \boldsymbol{\Sigma})^{-1}, \kappa^{-1}\boldsymbol{I}_{np}\right) = \sum_{i,j}\left((\tau_i\lambda_j)^{-\frac{1}{2}} - \kappa^{-\frac{1}{2}}\right)^2$	$\left(x^{-\frac{1}{2}} - \kappa^{-\frac{1}{2}}\right)^2$
Gaussian KL divergence	$\delta_{\mathrm{KL}}(\kappa\boldsymbol{I}_{np}, \mathrm{diag}(\boldsymbol{\tau}) \otimes \boldsymbol{\Sigma}) = \frac{1}{2}\left[\sum_{i,j}\left(\kappa(\tau_i\lambda_j)^{-1} + \log(\tau_i\lambda_j)\right) - np(1 + \log(\kappa))\right]$	$\frac{1}{2}\left[\kappa x^{-1} + \log(x) - (1 + \log(\kappa))\right]$

TABLE I: Examples of regularizations $\mathcal{R}_\kappa$ respecting the Assumptions 1, 2 and 3. $\forall q \in \mathbb{N}^*$, $\|.\|_q$ is the Schatten norm, *i.e.* $\forall \boldsymbol{A} \in \mathcal{S}_p$ $\|\boldsymbol{A}\|_q^q = \sum_i |\lambda_i|^q$ where λ_i are the eigenvalues of $\boldsymbol{A}$. The diagonal matrix with elements of $\boldsymbol{\tau}$ is denoted $\mathrm{diag}(\boldsymbol{\tau})$. The Kronecker product between matrices is denoted $\otimes$.

one $\tau_i^{(k)} \to \partial\mathbb{R}_*^+$, with $\partial\mathbb{R}_*^+$ being the boundary of $\mathbb{R}_*^+$, *i.e.* $\partial\mathbb{R}_*^+ = \{0, +\infty\}$. Using the positivity of the quadratic form in the NLL (5), we get the following inequality

$$\mathcal{L}(\theta^{(k)}|\{\boldsymbol{x}_i\}_{i=1}^n) \geq \sum_{i=1}^n \log\left|\tau_i^{(k)}\boldsymbol{\Sigma}^{(k)}\right| . \tag{29}$$

Hence, we get the resulting inequality on the regularized cost function

$$\mathcal{L}_{\mathcal{R}_\kappa}(\theta^{(k)}|\{\boldsymbol{x}_i\}_{i=1}^n) \geq$$
$$\sum_{i=1}^n \sum_{j=1}^p \left[\log(\tau_i^{(k)}\lambda_j^{(k)}) + \beta r_\kappa(\tau_i^{(k)}\lambda_j^{(k)})\right] . \tag{30}$$

Then, we give a sufficient condition to prove (28) when $\boldsymbol{\Sigma}^{(k)} \to \partial\mathcal{S}_p^{++}$ and/or $\boldsymbol{\tau}^{(k)} \to \partial\mathcal{S}(\mathbb{R}_*^+)^n$, the boundaries of $\partial\mathcal{S}_p^{++}$ and $\partial\mathcal{S}(\mathbb{R}_*^+)^n$ respectively. To give this sufficient condition, we first recall Assumption 1, $\forall\beta \in \mathbb{R}_*^+$

$$\lim_{x \to \partial\mathbb{R}_*^+} \log(x) + \beta r(x) = +\infty .$$

Thus, to prove (28), a sufficient condition, when $\boldsymbol{\Sigma}^{(k)} \to \partial\mathcal{S}_p^{++}$ and/or $\boldsymbol{\tau}^{(k)} \to \partial\mathcal{S}(\mathbb{R}_*^+)^n$ is that there exists at least one term $\tau_i^{(k)}\lambda_j^{(k)}$ such that

$$\tau_i^{(k)}\lambda_j^{(k)} \to \partial\mathbb{R}_*^+ . \tag{31}$$

Since $\boldsymbol{\Sigma}^{(k)} \to \partial\mathcal{S}_p^{++}$ and/or $\boldsymbol{\tau}^{(k)} \to \partial\mathcal{S}(\mathbb{R}_*^+)^n$, there exists at least one $\lambda_j^{(k)} \to \partial\mathbb{R}_*^+$ and/or one $\tau_i^{(k)} \to \partial\mathbb{R}_*^+$.

The condition (31) is of course met in the four following cases

$$\lambda_j^{(k)} \to 0^+ \text{ and/or } \tau_i^{(k)} \to 0^+ ,$$

$$\lambda_j^{(k)} \to +\infty \text{ and/or } \tau_i^{(k)} \to +\infty ,$$

$$\lambda_j^{(k)} \to 0^+ \text{ and } \tau_i^{(k)} \to +\infty \text{ such that } \tau_i^{(k)}\lambda_j^{(k)} \to \partial\mathbb{R}_*^+ ,$$

$$\lambda_j^{(k)} \to +\infty \text{ and } \tau_i^{(k)} \to 0^+ \text{ such that } \tau_i^{(k)}\lambda_j^{(k)} \to \partial\mathbb{R}_*^+ .$$

Finally, we treat the case where $\forall l \in \{1, \cdots, n\}$, $\lambda_l^{(k)} \to \partial\mathbb{R}_*^+$ and $\tau_i^{(k)} \to \partial\mathbb{R}_*^+$ such that $\tau_i^{(k)}\lambda_l^{(k)} \not\to \partial\mathbb{R}_*^+$. Since $\prod_{m=1}^n \tau_m^{(k)} = 1$, there exists at least one τ_q, with $q \neq i$, such that

$$\tau_q^{(k)}\lambda_j^{(k)} \to \partial\mathbb{R}_*^+ . \tag{32}$$

Hence, the condition (28) is met.

Before going further, we define the two following functions:

$$g_{\mathcal{S}_p^{++}}(\boldsymbol{\Sigma}^{(k)}) = \left\|\log(\boldsymbol{\Sigma}^{(k)})\right\|_F^2 = \sum_{j=1}^p \log(\lambda_j^{(k)})^2 ,$$

and

$$g_{(\mathbb{R}_*^+)^n}(\boldsymbol{\tau}^{(k)}) = \left\|\log(\boldsymbol{\tau}^{(k)})\right\|_2^2 = \sum_{i=1}^n \log(\tau_i^{(k)})^2 .$$

It should be noted that $\sup(g_{\mathcal{S}_p^{++}}(\boldsymbol{\Sigma}^{(k)})_{k=0}^{+\infty}) = +\infty$ if and only if there exists j such that $(\lambda_j^{(k)})_{k=0}^{+\infty}$ has $\partial\mathbb{R}_*^+$ as accumulation point. Similarly, $\sup(g_{(\mathbb{R}_*^+)^n}(\boldsymbol{\tau}^{(k)})_{k=0}^{+\infty}) = +\infty$ if and only if there exists i such that $(\tau_i^{(k)})_{k=0}^{+\infty}$ has $\partial\mathbb{R}_*^+$ as accumulation point.

Second, we consider the cases where $\|\boldsymbol{\mu}^{(k)}\|_2 \to +\infty$, $\sup(g_{\mathcal{S}_p^{++}}(\boldsymbol{\Sigma}^{(k)})_{k=0}^{+\infty}) < +\infty$ and $\sup(g_{(\mathbb{R}_*^+)^n}(\boldsymbol{\tau}^{(k)})_{k=0}^{+\infty}) < +\infty$. In this case, there exists $\lambda_{\min}, \lambda_{\max} > 0$ and $\tau_{\min}, \tau_{\max} > 0$ such that for all k, $\lambda_{\min}\boldsymbol{I}_p \preceq \boldsymbol{\Sigma}^{(k)} \preceq \lambda_{\max}\boldsymbol{I}_p$ and $\tau_{\min}\mathbb{1}_n \leq \boldsymbol{\tau}^{(k)} \leq \tau_{\max}\mathbb{1}_n$. Indeed, otherwise there would exist j, i such that $\partial\mathbb{R}_*^+$ is an accumulation point of $(\lambda_j^{(k)})_{k=0}^{+\infty}$ and/or $(\tau_i^{(k)})_{k=0}^{+\infty}$. Using these inequalities and the positivity of the regularization $\mathcal{R}_\kappa$, we get that

$$\mathcal{L}_{\mathcal{R}_\kappa}(\theta^{(k)}|\{\boldsymbol{x}_i\}_{i=1}^n) \geq \sum_{i=1}^n \frac{\left\|\boldsymbol{x}_i - \boldsymbol{\mu}^{(k)}\right\|^2}{\lambda_{\max}\tau_{\max}} + \mathrm{const} ,$$

where the constant is independent from $\boldsymbol{\mu}^{(k)}$. Thus, we have

$$\lim_{k \to +\infty} \mathcal{L}_{\mathcal{R}_\kappa}(\theta^{(k)}|\{\boldsymbol{x}_i\}_{i=1}^n) = +\infty .$$

Third, it remains to check the cases where $\|\boldsymbol{\mu}^{(k)}\| \to +\infty$, $\sup(g_{\mathcal{S}_p^{++}}(\boldsymbol{\Sigma}^{(k)})_{k=0}^{+\infty}) = +\infty$ and/or $\sup(g_{(\mathbb{R}_*^+)^n}(\boldsymbol{\tau}^{(k)})_{k=0}^{+\infty}) = +\infty$. Thus, as said previously, there exists at least one j and/or one i such that $(\lambda_j^{(k)})_{k=0}^{+\infty}$ and/or $(\tau_i^{(k)})_{k=0}^{+\infty}$ has/have $\partial\mathbb{R}_*^+$ as accumulation point. For each each of those j, i, we extract sub-sequences from $(\theta^{(k)})_{k=0}^{+\infty}$ whose limits in λ_j and/or τ_i are these problematic accumulation points. Then, we construct a partition of $\mathbb{N}$ with the indices corresponding to the elements of these sub-sequences and the indices of the remaining elements of the initial sequence $(\theta^{(k)})_{k=0}^{+\infty}$. Let $\mathcal{I}$ be such partition of $\mathbb{N}$. If $\mathcal{I}$ has a finite number of elements and if for every $(k_\ell)_{\ell=0}^{+\infty} \in \mathcal{I}$ we have

$$\lim_{\ell \to +\infty} \mathcal{L}_{\mathcal{R}_\kappa}(\theta^{(k_\ell)}|\{\boldsymbol{x}_i\}_{i=1}^n) = +\infty , \tag{33}$$

then we get (28).

To do so, in the space $\overline{\mathbb{R}^+} = [0, +\infty]$ equipped with the metric $d(x, y) = |\arctan(x) - \arctan(y)|$, given a $(\lambda_j^{(k)})_{k=0}^{+\infty}$ that has $L \in \partial\mathbb{R}_*^+$ as accumulation point, one can extract a sub-sequence of indices $(k_\ell)_{\ell=0}^{+\infty}$ such that $\lambda_j^{(k_\ell)} \to L$ and for $(k_m)_{m=0}^{+\infty} = \mathbb{N} \setminus (k_\ell)_{\ell=0}^{+\infty}$, $(\lambda_j^{(k_m)})_{m=0}^{+\infty}$ does not have L as accumulation point. The same process is repeated iteratively from the remaining indices $(k_m)_{m=0}^{+\infty}$ for all j such that $(\lambda_j^{(k_m)})_{m=0}^{+\infty}$ still has an $L \in \partial\mathbb{R}_*^+$ as accumulation point. It finishes when the sequence associated with the remaining elements of the original sequence $(\lambda_j^{(k)})_{k=0}^{+\infty}$ has no accumulation points in $\partial\mathbb{R}_*^+$. Lets denotes $(k_q)_{q=0}^{+\infty}$ the remaining indices. Then, the same process is also performed on $(\tau_i^{(k_q)})_{q=0}^{+\infty}$ if $\sup(g_{(\mathbb{R}_*^+)^n}(\boldsymbol{\tau}^{(k_q)})_{q=0}^{+\infty}) = +\infty$. All the obtained sequences of indices $(k_\ell)_{\ell=0}^{+\infty}$ along with the remaining elements of the original indices form a partition of $\mathbb{N}$. Due to its construction, this partition has at most $\operatorname{card}(\partial\mathbb{R}_*^+)^{p+n} + 1 = 2^{p+n} + 1$ elements. Furthermore, we point out that, since $\|\boldsymbol{\mu}^{(k)}\| \to +\infty$, we have that for every sub-sequence $(\boldsymbol{\mu}^{(k_\ell)})_{\ell=0}^{+\infty}$, $\|\boldsymbol{\mu}^{(k_\ell)}\| \to +\infty$. Thus, for every $(k_\ell)_{\ell=0}^{+\infty} \in \mathcal{I}$, we have

- either $\|\boldsymbol{\mu}^{(k_\ell)}\| \to +\infty$, $\sup(g_{\mathcal{S}_p^{++}}(\boldsymbol{\Sigma}^{(k_\ell)})_{\ell=0}^{+\infty}) < +\infty$ and $\sup(g_{(\mathbb{R}_*^+)^n}(\boldsymbol{\tau}^{(k_\ell)})_{\ell=0}^{+\infty}) < +\infty$,
- or $\|\boldsymbol{\mu}^{(k_\ell)}\| \to +\infty$ and there exists i and/or j such that $\lambda_j^{(k_\ell)} \to \partial\mathbb{R}_*^+$ and/or $\tau_i^{(k_\ell)} \to \partial\mathbb{R}_*^+$.

The former case has already been treated earlier. For the latter case, one can reuse the arguments between (29) and (32) to prove (33). Indeed, (29) discards the quadratic form in $\boldsymbol{\mu}^{(k_\ell)}$ and hence the equations between (29) and (32) hold. Thus, the condition (28) is met.

$\square$

B. Interpretations of the regularization term

So far, the regularization penalty has been chosen to guarantee the existence of a solution to the problem (26) without having specific insights on its impact on the estimate. Therefore, this section thus discusses the interpretations of various classes of penalties and their related shrinkage effect.

A Bayesian interpretation of the considered penalties $\mathcal{R}_\kappa$ requires first discussing the case where it is decoupled in terms of $\{\tau_i\}_{i=1}^n$ and $\{\lambda_j\}_{j=1}^p$, i.e., when it can be expressed as

$$\mathcal{R}_\kappa(\theta) = p \sum_{i=1}^n r_\kappa^\tau(\tau_i) + n \sum_{j=1}^p r_\kappa^\lambda(\lambda_j). \qquad (34)$$

In such cases,

- r_κ^τ can be linked to a pdf on τ, denoted f_τ. Assuming that $r_\kappa^\lambda(t) = 0$ the optimization problem relates to the maximum a posteriori estimation of the Compound Gaussian model $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \tau\boldsymbol{\Sigma})$ with $\tau \sim f_\tau$ [2], [3]. Such a procedure is not often put into practice because it is generally possible (and preferable) to study the resulting pdf for the observations $\boldsymbol{x}$:

$$f_{CG}(\boldsymbol{x}) \propto \int f_G(\boldsymbol{x}|\boldsymbol{\mu}, \tau\boldsymbol{\Sigma}) f_\tau(\tau) \, \mathrm{d}\tau, \qquad (35)$$

whose MLE estimator appears as a special case of M-estimators of location and scatter, and is tractable with a fixed point algorithm [2], [3].

- The penalty r_κ^λ could also be interpreted as a pdf on the eigenvalues of $\boldsymbol{\Sigma}$. This approach is less often studied from the Bayesian point of view because it does not have a clear interpretation of the distribution of the resulting $\boldsymbol{\Sigma}$. Still, such penalties were leveraged to ensure existence of solutions of regularized M-estimators when $n < p$, e.g, in [17], [18], [43].

When additional prior information is available (power constraints that bound the eigenvalues, a rough estimate of the textures pdf, etc.) a Bayesian approach can be practically leveraged to select the form of the regularization penalty and the regularization parameters κ and β.

In the general case of Assumption 1, i.e., where $\mathcal{R}_\kappa$ is possibly not decoupled, a Bayesian interpretation of $\mathcal{R}_\kappa$ is not as apparent. Still, we can show that when the penalty can be interpreted as a divergence, it allows for explaining its effect on the estimate. First, we recall the definition of a divergence:

Definition 1 (Divergence). *A divergence on a set E is a function $\delta(.,.) : E \times E \to \mathbb{R}$ satisfying, $\forall x, y \in E$:*

1) $\delta(x, y) \geq 0$,
2) $\delta(x, y) = 0 \iff x = y$.

We can then state the following assumption, which is notably verified for all regularization examples given in Table 1:

Assumption 3. *The regularization $\mathcal{R}_\kappa$ can be written as*

$$\mathcal{R}_\kappa(\theta) = \delta_{\mathcal{S}_p^{++}}(\operatorname{diag}(\boldsymbol{\tau}) \otimes \boldsymbol{\Sigma}, \kappa\boldsymbol{I}_{np}),$$

where $\delta_{\mathcal{S}_p^{++}}$ is a divergence on the set $\mathcal{S}_p^{++}$ and $\kappa \in \mathbb{R}_*^+$.

This assumption allows us to state the following proposition:

Proposition 7 (Minima of $\mathcal{R}_\kappa$). *Under the Assumption 3, the set of minima in $\mathcal{M}_{p,n}$ of the regularization $\mathcal{R}_\kappa$ is*

$$\{\theta = (\boldsymbol{\mu}, \kappa\boldsymbol{I}_p, \mathbb{1}_n) : \boldsymbol{\mu} \in \mathbb{R}^p\}.$$

Proof. The objective of this proof is to solve

$$\underset{\theta \in \mathcal{M}_{p,n}}{\operatorname{minimize}} \mathcal{R}_\kappa(\theta).$$

Using Assumption 3, we know that $\mathcal{R}_\kappa(\theta) \geq 0$ and $\mathcal{R}_\kappa(\theta) = 0 \iff \operatorname{diag}(\boldsymbol{\tau}) \otimes \boldsymbol{\Sigma} = \kappa\boldsymbol{I}_{np}$. Thus, the minimum of $\mathcal{R}_\kappa$ is 0 and is reached at $\operatorname{diag}(\boldsymbol{\tau}) \otimes \boldsymbol{\Sigma} = \kappa\boldsymbol{I}_{np}, \forall \boldsymbol{\mu} \in \mathbb{R}^p$. This implies that the minimum satisfies the following system of equations

$$\tau_i \lambda_j = \kappa \quad \forall i, j.$$

Hence, we deduce that $\tau_1 = \cdots = \tau_n$. Using the constraint $\prod_{i=1}^n \tau_i = 1$, we get that $\tau_1 = \cdots = \tau_n = 1$. Thus, $\lambda_1 = \cdots = \lambda_p = \kappa$. This means that

$$\{(\boldsymbol{\mu}, \kappa\boldsymbol{I}_p, \mathbb{1}_n) : \boldsymbol{\mu} \in \mathbb{R}^p\} = \underset{\theta \in \mathcal{M}_{p,n}}{\arg\min} \mathcal{R}_\kappa(\theta),$$

which characterizes Proposition 7. $\square$

Thus, under Assumption 3, the minimum of (26) tends to $\left(\frac{1}{n} \sum_{i=1}^n \boldsymbol{x}_i, \kappa\boldsymbol{I}_p, \mathbb{1}_n\right)$ as $\beta \to +\infty$. This corresponds to

the MLE of a Gaussian distribution with a covariance matrix $\kappa \boldsymbol{I}$. Thus, the hyperparameter β makes the trade-off between an NC-MSG (2) and a white Gaussian distribution. Hence, one can set in practice the hyperparameter κ as $\kappa = \frac{1}{p} \text{Tr}(\frac{1}{n} \boldsymbol{X} \boldsymbol{X}^T) = \frac{1}{np} \sum_{i=1}^{n} \|\boldsymbol{x}_i\|_2^2$, meaning that the eigenvalues will be shrunk towards their empirical mean. The effect of the regularization then echoes to existing shrinkage of M-estimators that have the same action [13]–[15], [43], [44].

To conclude, Assumption 2 and Proposition 6 provide the conditions that ensure the existence of a solution of the regularized MLE for any $\kappa > 0$, whether $\mathcal{R}_\kappa$ is decoupled (with a Bayesian interpretation), interpretable as a divergence (following Assumption 3 and Proposition 7), or not. This class of regularization penalties thus allows going beyond the Bayesian estimation framework. In practice, we mostly consider minimizing the estimation bias induced by the penalty and set β close to 0. For other tasks such as estimates used in classification, we resort to cross-validation procedures to select β (see example in Figure 7).

C. Robustness to rigid transformations

We finish this section with a remark on estimating the parameter θ when data undergo a rigid transformation. First of all, we define the set of orthogonal matrices

$$\mathcal{O}_p = \left\{ \boldsymbol{Q} \in \mathbb{R}^{p \times p} : \boldsymbol{Q}^T \boldsymbol{Q} = \boldsymbol{I}_p \right\}. \tag{36}$$

Then, given $\boldsymbol{Q} \in \mathcal{O}_p$ and $\boldsymbol{\mu}_0 \in \mathbb{R}^p$, the rigid transformation ψ of a set of data $\{\boldsymbol{x}_i\}_{i=1}^n$ is defined as

$$\psi \left(\{\boldsymbol{x}_i\}_{i=1}^n \right) = \left\{ \boldsymbol{Q}^T \boldsymbol{x}_i + \boldsymbol{\mu}_0 \right\}_{i=1}^n. \tag{37}$$

These rigid transformations define isometries on $\mathbb{R}^p$ since

$$\| \psi (\boldsymbol{x}_i) - \psi (\boldsymbol{x}_j) \|_2 = \| \boldsymbol{x}_i - \boldsymbol{x}_j \|_2, \tag{38}$$

$\forall \boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^p$. These are important in machine learning problems since they transform data without changing distances. An important property of the regularized NLL (25) is that the estimated textures of the model are invariant under rigid transformations of the data; see Proposition 8. This is interesting since having parameters invariant to these transformations can improve performances when transformations happen between the training and the test sets for a given supervised problem. Numerical experiments in Section VII leverage this property and show robust performances when data undergo a rigid transformation during the testing phase.

Proposition 8 (Minima of $\mathcal{L}_{\mathcal{R}_\kappa}$ and rigid transformations). *Let $\mathcal{R}_\kappa$ be a regularization satisfying Assumption 1, and $\theta^\star = (\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\tau})$ be a minimum of the regularized NLL (26) computed on data $\{\boldsymbol{x}_i\}_{i=1}^n$, i.e.*

$$\theta^\star \in \arg\min_{\theta \in \mathcal{M}_{p,n}} \mathcal{L}_{\mathcal{R}_\kappa} \left(\theta | \{\boldsymbol{x}_i\}_{i=1}^n \right),$$

then, given $\boldsymbol{Q} \in \mathcal{O}_p$ and $\boldsymbol{\mu}_0 \in \mathbb{R}^p$, a minimum of the regularized NLL computed on the transformed data $\psi \left(\{\boldsymbol{x}_i\}_{i=1}^n \right) = \left\{ \boldsymbol{Q}^T \boldsymbol{x}_i + \boldsymbol{\mu}_0 \right\}_{i=1}^n$ is $\phi(\theta^\star) = \left(\boldsymbol{Q}^T \boldsymbol{\mu} + \boldsymbol{\mu}_0, \boldsymbol{Q}^T \boldsymbol{\Sigma} \boldsymbol{Q}, \boldsymbol{\tau} \right)$, i.e.

$$\phi(\theta^\star) \in \arg\min_{\theta \in \mathcal{M}_{p,n}} \mathcal{L}_{\mathcal{R}_\kappa} \left(\theta | \psi \left(\{\boldsymbol{x}_i\}_{i=1}^n \right) \right).$$

Proof. First of all, given $\boldsymbol{Q} \in \mathcal{O}_p$ and $\boldsymbol{\mu}_0 \in \mathbb{R}^p$, one can check that

$$\mathcal{L} \left(\phi(\theta) | \psi \left(\{\boldsymbol{x}_i\}_{i=1}^n \right) \right) = \mathcal{L} \left(\theta | \{\boldsymbol{x}_i\}_{i=1}^n \right),$$

where $\mathcal{L}$ is the NLL defined in (5), $\theta = (\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\tau})$, $\phi(\theta) = (\boldsymbol{Q}^T \boldsymbol{\mu} + \boldsymbol{\mu}_0, \boldsymbol{Q}^T \boldsymbol{\Sigma} \boldsymbol{Q}, \boldsymbol{\tau})$ and ψ is defined in equation (37). Then, $\mathcal{R}_\kappa$ satisfies Assumption 1 and thus only depends on the eigenvalues of the matrices $\tau_i \boldsymbol{\Sigma}$. This implies that $\mathcal{R}_\kappa(\phi(\theta)) = \mathcal{R}_\kappa(\theta)$ and hence we get that

$$\mathcal{L}_{\mathcal{R}_\kappa} \left(\phi(\theta) | \psi \left(\{\boldsymbol{x}_i\}_{i=1}^n \right) \right) = \mathcal{L}_{\mathcal{R}_\kappa} \left(\theta | \{\boldsymbol{x}_i\}_{i=1}^n \right).$$

This implies that if $\theta^\star \in \arg\min_{\theta \in \mathcal{M}_{p,n}} \mathcal{L}_{\mathcal{R}_\kappa} (\theta | \{\boldsymbol{x}_i\}_{i=1}^n)$, then $\phi(\theta^\star) \in \arg\min_{\theta \in \mathcal{M}_{p,n}} \mathcal{L}_{\mathcal{R}_\kappa} (\theta | \psi (\{\boldsymbol{x}_i\}_{i=1}^n))$, which concludes the proof. $\square$

VI. KL DIVERGENCE AND RIEMANNIAN CENTER OF MASS

In the previous section, we proposed to optimize the regularized NLL (26) of the NC-MSG (2). Once these parameters are estimated, they can be used as features for Riemannian classification/clustering algorithms [19]–[22]. To do this classification/clustering, two tools are presented in this section. Firstly, since no closed-form formula of the Riemannian distance on $\mathcal{M}_{p,n}$ is known, a divergence between pairs of parameters is defined. The proposed one is the KL divergence between two NC-MSGs (2). It benefits from a simple closed-form formula presented in Subsection VI-A. Secondly, simple classification algorithms, such as *K-means* or the *Nearest centroïd classifier*, rely on an algorithm to average parameters. Thus, an algorithm to compute centers of mass of estimated parameters θ must be defined. This center of mass is defined using the KL divergence and is presented in Subsection VI-B. Its computation is realized with Algorithm 1.

A. KL divergence

Classification/clustering algorithms, such as *K-means* or the *Nearest centroïd classifier*, rely on a divergence between points. Thus, it remains to define a divergence on $\mathcal{M}_{p,n}$. The latter must be related to the NC-MSG (2). Indeed, the objective is to classify its parameters θ. In the context of measuring proximities between distributions admitting probability density functions, a classical divergence is the KL. The latter measures the similarity between two probability density functions. Definition 2 gives the general formula of the KL divergence.

Definition 2 (KL divergence). *Given two probability density functions p and q defined on the sample space $\mathcal{X}$, the KL divergence is*

$$\delta_{\text{KL}}(p, q) = \int_{\mathcal{X}} p(x) \log \left(\frac{p(x)}{q(x)} \right) dx.$$

Applied to NC-MSGs, the KL divergence is derived from the Gaussian one and is presented in Proposition 9. It benefits from a simple closed-form formula and therefore is of practical interest.

Proposition 9 (KL divergence). *Given the r.v.* $x = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ *and two NC-MSGs of probability density functions* $p_{\theta_1}(x) = \prod_{i=1}^{n} f\left(\boldsymbol{x}_i | (\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1, \tau_{1,i})\right)$ *and* $p_{\theta_2}(x) = \prod_{i=1}^{n} f\left(\boldsymbol{x}_i | (\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2, \tau_{2,i})\right)$, *the KL divergence is*

$$\delta_{\text{KL}}(\theta_1, \theta_2) = \frac{1}{2}\left(\sum_{i=1}^{n} \frac{\tau_{1,i}}{\tau_{2,i}} \operatorname{Tr}\left(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1\right) + \right.$$
$$\left. \sum_{i=1}^{n} \frac{1}{\tau_{2,i}} \Delta\boldsymbol{\mu}^T \boldsymbol{\Sigma}_2^{-1} \Delta\boldsymbol{\mu} + n\log\left(\frac{|\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|}\right) - np\right),$$

with $\Delta\boldsymbol{\mu} = \boldsymbol{\mu}_2 - \boldsymbol{\mu}_1$.

Proof. The r.v. $x = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ can be vectorized into $\boldsymbol{x} = [\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_n^T]^T \in \mathbb{R}^{np}$ which follows a multivariate Gaussian distribution of location the concatenation of the locations of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ and of block-diagonal covariance matrix whose elements are the covariance matrices of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. Thus, the KL divergence between the probability density functions p_{θ_1} and p_{θ_2} is the KL divergence between two multivariate Gaussian distributions whose covariance matrices are block diagonal. Using the KL divergence between Gaussian distributions and the constraint $\prod_{i=1}^{n} \tau_{1,i} = \prod_{i=1}^{n} \tau_{2,i} = 1$, we get the desired formula. $\qquad\square$

Finally, this KL divergence is non-symmetrical. We rely on the classical symmetrization to define the proposed divergence $\delta_{\mathcal{M}_{p,n}} : \mathcal{M}_{p,n} \times \mathcal{M}_{p,n} \to \mathbb{R}$,

$$\delta_{\mathcal{M}_{p,n}}(\theta_1, \theta_2) = \frac{1}{2}\left(\delta_{\text{KL}}(\theta_1, \theta_2) + \delta_{\text{KL}}(\theta_2, \theta_1)\right). \quad (39)$$

B. Center of mass computation

To implement simple machine learning algorithms such as *K-means* or the *Nearest centroïd classifier* on $\mathcal{M}_{p,n}$, it remains to define an averaging algorithm. To do so, we leverage a classical definition of centers of mass which are minimizers of variances [23], [45]. Given a set of parameters $\{\theta_i\}_{i=1}^{M}$, its center of mass on $\mathcal{M}_{p,n}$ is defined as the solution of

$$\underset{\theta \in \mathcal{M}_{p,n}}{\text{minimize}} \frac{1}{M} \sum_{i=1}^{M} \delta_{\mathcal{M}_{p,n}}(\theta, \theta_i), \quad (40)$$

where $\delta_{\mathcal{M}_{p,n}}$ is the symmetrized KL divergence from Equation (39). To realize (40), Algorithm 1 can be employed.

VII. Numerical experiments

The objective of this section is to show the practical interests of the tools developed in the previous sections. More precisely, this section presents numerical experiments and is divided into two parts.

First, the subsection VII-A studies the performance of Algorithm 1, in terms of speed of convergence on the cost functions (26) and (40) and in terms of estimation error on the cost function (22). Both studies are done through simulations. Algorithm 1 is shown to be fast. Indeed, it requires from 5 to 30 times fewer iterations to minimize costs functions (26) and (40) compared to other sophisticated optimization algorithms. This demonstrates the interest in the choice of the FIM to develop Riemannian optimization algorithms. Also,

Algorithm 1 applied to the cost function (22) gives lower estimation errors than other classical estimators such as the Tyler joint mean-scatter one and the Gaussian ones.

Second, an application on the crop classification dataset *Breizhcrops* [29] is presented in Subsection VII-B. This dataset consists of 600 000 time series to be classified into 9 classes. The application implements a *Nearest centroïd classsifier* on $\mathcal{M}_{p,n}$ using the divergence (39) and the Riemannian center of mass (40). Three results ensue. First, the proposed algorithms can be used on large-scale datasets. Second, the proposed regularization in Section V plays an important role in classification. Third, considering an NC-MSG (2) is interesting for time series especially when data undergo a rigid transformation (37).

Python code implementing the different experiments can be found at https://github.com/antoinecollas/optim_compound.

A. Simulation

In this simulation setting, we set the parameters $\theta = (\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\tau}) \in \mathcal{M}_{p,n}$ as follows. First, each component of $\boldsymbol{\mu}$ is sampled from a univariate Gaussian distribution $\mathcal{N}(0,1)$. Second, $\boldsymbol{\Sigma}$ is generated using its eigendecomposition $\boldsymbol{\Sigma} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$. $\boldsymbol{U} \in \mathcal{O}_p$ is drawn from the uniform distribution on $\mathcal{O}_p$ [46] using the module "scipy.stats" from the Scipy library [47]. Then, the elements on the diagonal of the diagonal matrix $\boldsymbol{\Lambda}$ are drawn from a χ_1^2 distribution. Third, the τ_i are drawn from a $\Gamma(\nu, 1/\nu)$ distribution with ν a parameter to be chosen. The smaller the ν, the greater the variance. In order to respect the constraint $\prod_{i=1}^{n} \tau_i = 1$, the vector $\boldsymbol{\tau}$ is normalized.

The speed of convergence of Algorithm 1 is studied on two cost functions: the regularized NLL (26) and the cost function (40) to compute the center of mass associated to the KL divergence of Proposition 9.

We begin with the minimization of the regularized NLL (26). $n = 150$ data $\boldsymbol{x}_i \in \mathbb{R}^{10}$ are drawn from a NC-MSG, *i.e.* $\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{\mu}, \tau_i\boldsymbol{\Sigma})$. The parameter $\theta = (\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\tau})$ of this distribution is generated as explained in the introduction of this subsection with $\nu = 1$. Different parameters β in (26) are considered: $\beta \in \{0, 10^{-5}, 10^{-3}\}$. The chosen regularization is the L2 penalty from Table I. When $\beta = 0$ the NLL is the plain one, *i.e.* it is not regularized. We point out that, in this setup, the optimization goes well although the existence of a solution to this problem is not proven. When $\beta > 0$ a solution to the minimization problem exists from Proposition 6. The minimization is performed with three different algorithms.

- The plain conjugate gradient presented in [37]. It is a Riemannian conjugate gradient descent that uses a sum of three independent Riemannian metrics associated with the three parameters $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, and $\boldsymbol{\tau}$. Thus, the corresponding Riemannian geometry is easier to derive but is not linked to the NC-MSG.
- The plain steepest descent. It is similar to the plain conjugate gradient. Still, it only uses the gradient as a direction of descent (and not a linear combination with the direction of descent of the previous step).
- The Algorithm 1 that leverages the information geometry of the NC-MSG presented in Section III-B.

The results of this experiment are presented in Figures 1 and 3 in terms of iterations and computation time respectively.

We observe that Algorithm 1 is much faster than the two others regardless of the β parameter. Indeed, in the case $\beta \in \{0, 10^{-5}\}$, the Algorithm 1 is at least 100 times faster than the plain steepest descent and 10 times faster than the plain conjugate gradient. In the case of $\beta = 10^{-3}$, Algorithm 1 is at least 20 times faster than the plain steepest descent and 3 times faster than the plain conjugate gradient. Furthermore, we observe these results are valid either in the number of iterations or in computation time. Indeed, the three considered algorithms have iterations with similar computational costs in $\mathcal{O}(np^2 + p^3)$. Thus, a reduction in the number of iterations results in a reduction in computation time.

Then, a similar experiment is performed with the cost function (40) to compute the center of mass. $M \in \{2, 10, 100\}$ parameters θ are generated as described in the introduction of Subsection VII-A with $\nu = 1$. The minimization is performed with the same optimization algorithms as previously: the plain steepest descent, the plain conjugate gradient, and Algorithm 1. The results of this experiment are presented in Figures 2 and 4 in terms of iterations and computation time, respectively.

We observe that Algorithm 1 is much faster than the two others regardless of M. Indeed, when $M = 2$, Algorithm 1 converges in 40 iterations whereas the plain conjugate gradient requires 300 iterations and the plain steepest descent still has not converged after 1000 iterations. When $M \in \{10, 100\}$, Algorithm 1 converges in less than 60 iterations which is 4 times faster than the plain conjugate gradient. It should be noted that the plain steepest descent has not converged after 1000 iterations in the cases $M \in \{100, 1000\}$. Once again, these results are valid either in the number of iterations or in computation time since the three considered algorithms have iterations with similar computational costs in $\mathcal{O}\left(M\left(n + p^3\right)\right)$. Hence, reducing the number of iterations implies a reduction in computation time.

The estimation error made by Algorithm 1 applied on the NLL (5) is studied with numerical experiments on simulated data. $n \in \llbracket 20, 1000 \rrbracket$ data $\boldsymbol{x}_i$ are sampled from the NC-MSG (2). The parameter $\theta = (\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\tau})$ of this distribution is generated as presented in the introduction of Subsection VII-A with $\nu = 0.1$ to have heterogeneous textures τ_i. The considered estimators for this numerical experiment are the following:

- Gaussian estimators: the sample mean $\boldsymbol{\mu}^{\mathrm{G}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i$ and the SCM $\boldsymbol{\Sigma}^{\mathrm{G}} = \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{x}_i - \boldsymbol{\mu}^{\mathrm{G}}\right)\left(\boldsymbol{x}_i - \boldsymbol{\mu}^{\mathrm{G}}\right)^T$.
- Tyler's joint location-scatter matrix estimator [5] denoted $\boldsymbol{\mu}^{\mathrm{Ty}}$ and $\boldsymbol{\Sigma}^{\mathrm{Ty}}$.
- Tyler's M-estimator with location known [5]. The sampled data $\boldsymbol{x}_i$ are centered with the true location $\boldsymbol{\mu}$, and then $\boldsymbol{\Sigma}$ is estimated. This estimator is denoted $\boldsymbol{\Sigma}^{\mathrm{Ty}, \boldsymbol{\mu}}$.
- The proposed estimator denoted $\boldsymbol{\mu}^{\mathrm{IG}}$ and $\boldsymbol{\Sigma}^{\mathrm{IG}}$. Algorithm 1 minimizes the NLL (5). The initialization is the Gaussian maximum likelihood *i.e.* $\theta_{\mathrm{init}} = \left(\boldsymbol{\mu}^{\mathrm{G}}, \boldsymbol{\Sigma}^{\mathrm{G}}, \mathbf{1}_n\right)$, where $\boldsymbol{\mu}^{\mathrm{G}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i$, $\boldsymbol{\Sigma}^{\mathrm{G}} = \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{x}_i - \boldsymbol{\mu}^{\mathrm{G}}\right)\left(\boldsymbol{x}_i - \boldsymbol{\mu}^{\mathrm{G}}\right)^T$ and $\mathbf{1}_n = (1, \cdots, 1)^T$.

The estimation errors are measured with the Mean Squared Errors (MSE). These errors are computed as $\mathbb{E}[\|\hat{\boldsymbol{\mu}} - \boldsymbol{\mu}\|_2^2]$ and $\mathbb{E}[\|Q(\hat{\boldsymbol{\Sigma}}) - Q(\boldsymbol{\Sigma})\|_F^2]$, with $Q(\boldsymbol{\Sigma}) = |\boldsymbol{\Sigma}|^{-\frac{1}{p}}\boldsymbol{\Sigma}$, for the estimated location $\hat{\boldsymbol{\mu}}$ and the estimated scatter $\hat{\boldsymbol{\Sigma}}$ respectively, with 2000 Monte-Carlo. The MSE on the location and the scatter versus the number of samples $\boldsymbol{x}_i$ are plotted in Figure 5. First, we observe in both figures that the Gaussian estimators have a high MSE. This shows the interest in considering robust estimators such as Tyler's joint location-scatter matrix estimator or the proposed one when the textures τ_i are heterogeneous. Then, the proposed estimators realize a much lower MSE than Tyler's joint location-scatter estimator. We can note that when enough samples are provided, the MSE on the location realized by the proposed estimator reaches the machine precision and is therefore negligible. Finally, we compare the performance of the proposed estimator with Tyler's M-estimator for the scatter estimation. Indeed, when the location is known, Tyler's M-estimator is the MLE of the NC-MSG (2). We observe that when enough samples are provided, the proposed estimator matches the MSE of Tyler's M-estimator. Overall, this experimental subsection illustrates the good performance of the proposed estimator when data are sampled from a NC-MSG (2).

B. Application

In the previous subsection, the different theoretical results derived in Sections from III to VI showed several interests in synthetic data. We now focus on applying a *Nearest centroïd classifier* on $\mathcal{M}_{p,n}$ to real data using the estimation framework developed in Section V, the divergence and the Riemannian center of mass from Section VI as well as the optimization framework from Section IV. This classifier is compared with several other *Nearest centroïd classifiers* associated with different estimators and divergences.

To do so, we consider the dataset *Breizhcrops* [29]: a large-scale dataset of more than 600 000 crop time series from the Sentinel-2 satellite to classify. More specifically, for each crop $n = 45$ observations $\boldsymbol{x}_i \in \mathbb{R}^p$ are measured over time. Each $\boldsymbol{x}_i$ contains reflectance measurements of $p = 13$ spectral bands. Then, these measurements are concatenated into one batch $\boldsymbol{X}_j = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{p \times n}$. Hence, we get one matrix $\boldsymbol{X}_j$ per crop and each one belongs to an unknown class $y \in \llbracket 1, K \rrbracket$. These $K = 9$ classes represent crop types such as nuts, barley, or wheat and are heavily imbalanced, *i.e.* some classes are much more represented than others. An example of a time series of meadows is presented in Figure 6. We apply a single preprocessing step: all the data are centered using the global mean. For simplicity, the matrix $\boldsymbol{X}_j$ is noted $\boldsymbol{X}$ in the following.

To classify these crops, we apply a *Nearest centroïd classifier* on descriptors. Indeed, the use of statistical descriptors is a classical procedure in machine learning as they are often more discriminative than raw data (see *e.g.* [19], [20]). Hence, this classification algorithm works in three steps.

1) For each batch $\boldsymbol{X}$, a descriptor is computed, *e.g.* the parameter $\theta \in \mathcal{M}_{p,n}$ from the minimization of the regularized NLL (25).
2) Then, on the training set, the center of mass of the descriptors of each class is computed. This center of
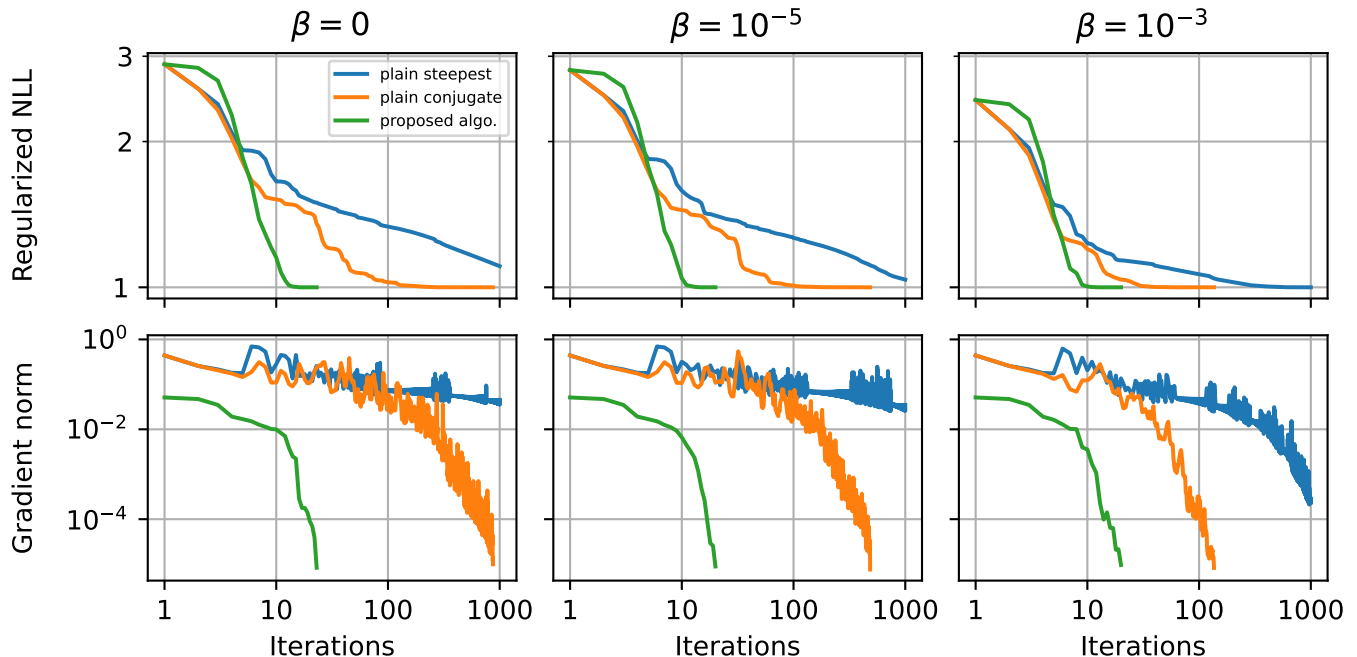
Fig. 1: Regularized NLL (26) and its gradient norm versus the iterations of three estimation algorithms. The chosen regularization is the L2 penalty (see Table I) and three different regularization intensities β are considered: 0 in the left column, 10^{-5} in the middle one, and 10^{-3} in the right one. Each estimation is performed on $n = 150$ samples in $\mathbb{R}^{10}$ sampled from an NC-MSG. The regularized NLL are normalized so that their minimum value is 1.

Fig. 2: Cost function (40) and its gradient norm versus the iterations of three estimation algorithms. The dimensions of the parameter space are $p = 10$ and $n = 150$. Three different numbers of points M are considered: 2 in the left column, 10 in the middle one, and 100 in the right one. The cost functions are normalized so that their minimum value is 1.
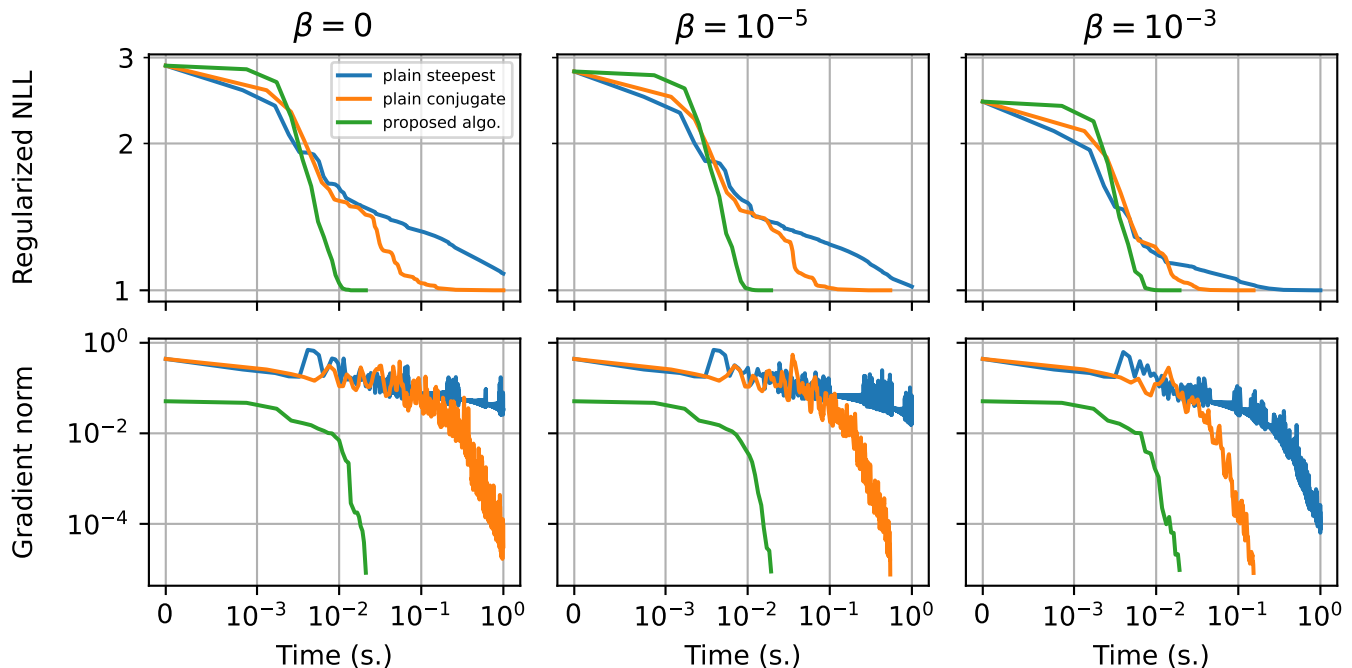
Fig. 3: Regularized NLL (26) and its gradient norm versus the computation time of three estimation algorithms. The chosen regularization is the L2 penalty (see Table I) and three different regularization intensities β are considered: 0 in the left column, 10^{-5} in the middle one, and 10^{-3} in the right one. Each estimation is performed on $n = 150$ samples in $\mathbb{R}^{10}$ sampled from an NC-MSG. The regularized NLL are normalized so that their minimum value is 1.

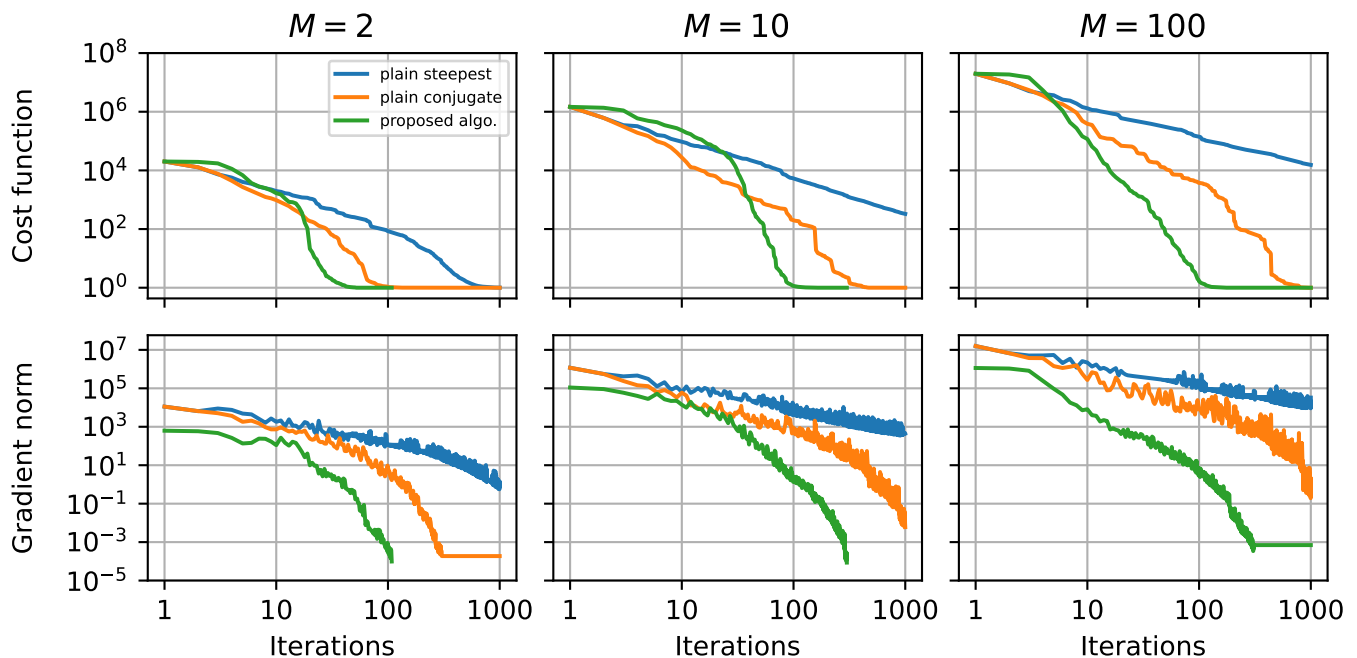
Fig. 4: Cost function (40) and its gradient norm versus the computation time of three estimation algorithms. The dimensions of the parameter space are $p = 10$ and $n = 150$. Three different numbers of points M are considered: 2 in the left column, 10 in the middle one, and 100 in the right one. The cost functions are normalized so that their minimum value is 1.
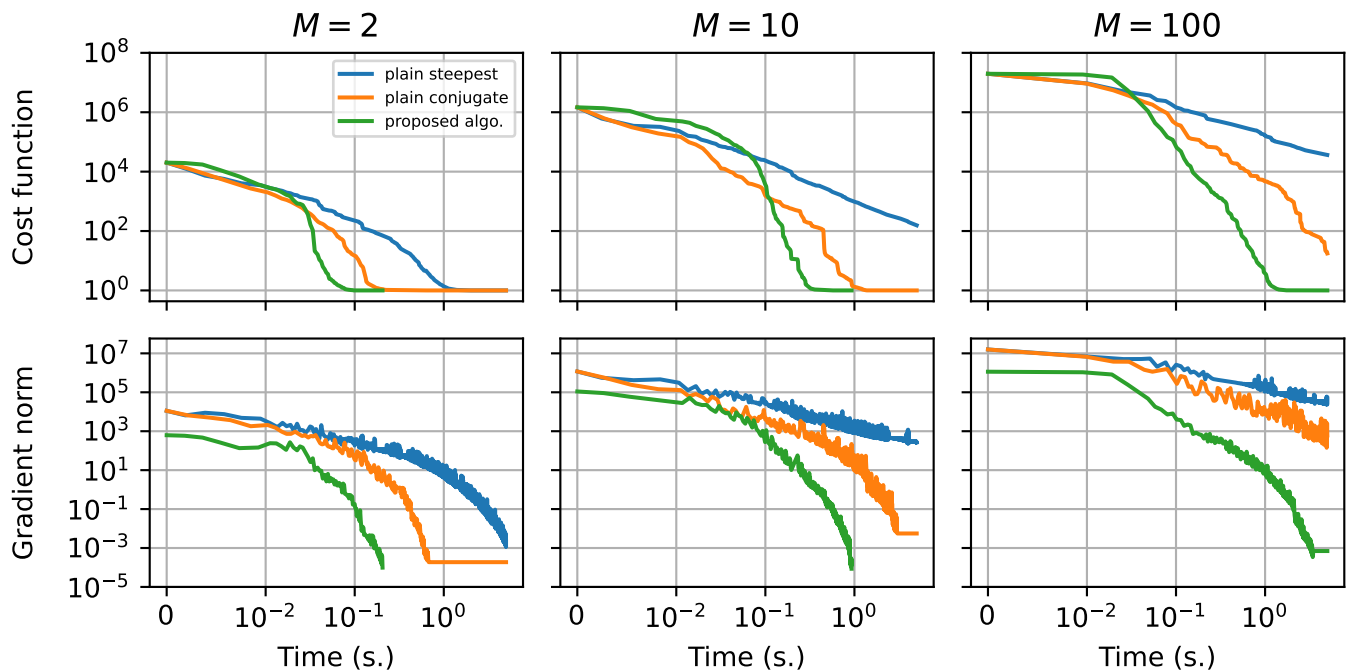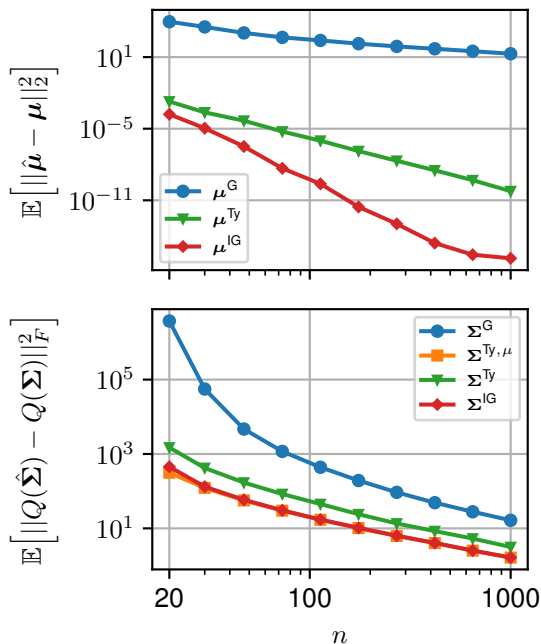
Fig. 5: MSE over 2000 simulated sets $\{\boldsymbol{x}_i\}_{i=1}^n \subset \mathbb{R}^{10}$ versus the number samples $\boldsymbol{x}_i$ for the considered estimators $\widehat{\boldsymbol{\mu}} \in \{\boldsymbol{\mu}^{\mathrm{G}}, \boldsymbol{\mu}^{\mathrm{Ty}}, \boldsymbol{\mu}^{\mathrm{IG}}\}$ and $\widehat{\boldsymbol{\Sigma}} \in \{\boldsymbol{\Sigma}^{\mathrm{G}}, \boldsymbol{\Sigma}^{\mathrm{Ty},\boldsymbol{\mu}}, \boldsymbol{\Sigma}^{\mathrm{Ty}}, \boldsymbol{\Sigma}^{\mathrm{IG}}\}$. The proposed estimators $\boldsymbol{\mu}^{\mathrm{IG}}$ and $\boldsymbol{\Sigma}^{\mathrm{IG}}$ are computed as in (22) using Algorithm 1 and the Q function normalizes scatter matrices, i.e. $\forall \boldsymbol{\Sigma} \in \mathcal{S}_p^{++}$, $Q(\boldsymbol{\Sigma}) = |\boldsymbol{\Sigma}|^{-\frac{1}{p}} \boldsymbol{\Sigma}$.
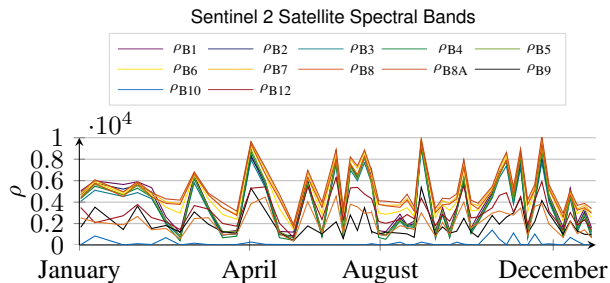

Fig. 6: Reflectances of a Sentinel-2 time series of meadows from the *Breizhcrops* dataset. Figure courtesy [29].

mass is always computed by minimizing the variance associated with a divergence between descriptors. For example, the center of mass on $\mathcal{M}_{p,n}$ is computed as in (40).

3) Finally, on the test set, each descriptor is labeled with the class of the nearest center of mass with respect to the chosen divergence.

Six *Nearest centroïd classifiers* are considered, and they are grouped according to the divergence they use: the Euclidean distance, the symmetrized KL divergence between Gaussian distributions, or the symmetrized KL divergence (39) between NC-MSGs. For each divergence, several *Nearest centroïd classifiers* are derived using several estimators. These estimators correspond to different assumptions on the data.

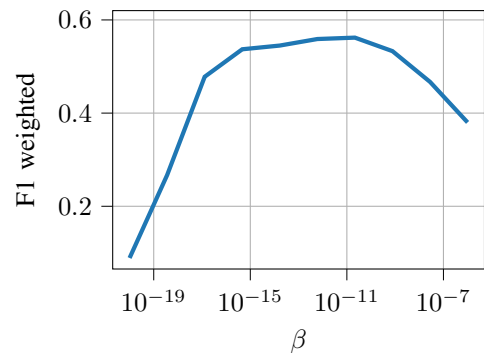Three *Nearest centroïd classifiers* rely on the Euclidean

Fig. 7: "F1 weighted" metric achieved by the proposed *Nearest centroïd classifier* on the *Breizhcrops* dataset versus the parameter of regularization β in (25). The chosen regularization is the L2 penalty from Table I.

distance between matrices. Given two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ of the same size, the Euclidean distance is $d(\boldsymbol{A}, \boldsymbol{B}) = \|\boldsymbol{A} - \boldsymbol{B}\|_F$. The center of mass of a given set $\{\boldsymbol{A}_i\}_{i=1}^M$ is the arithmetic mean $\frac{1}{M} \sum_{i=1}^M \boldsymbol{A}_i$ which is the solution of $\mathrm{minimize}_{\boldsymbol{Y}} \frac{1}{M} \sum_{i=1}^M \|\boldsymbol{Y} - \boldsymbol{A}_i\|_F^2$. From this geometry, three *Nearest centroïd classifiers* are derived using three estimators: the batch itself $\boldsymbol{X}$, the sample mean $\boldsymbol{\mu}^{\mathrm{G}}$ and $\boldsymbol{\Sigma}^{\mathrm{G},\boldsymbol{\mu}=\mathbf{0}} = \frac{1}{n} \sum_{i=1}^n \boldsymbol{x}_i \boldsymbol{x}_i^T$. The last two estimators correspond to the assumption that data follow a Gaussian distribution (either with the same scatter matrix for all batches or the same location).

Two *Nearest centroïd classifiers* rely on the symmetrized KL divergence between Gaussian distributions. Let $\mathcal{M}_p = \mathbb{R}^p \times \mathcal{S}_p^{++}$. Given two pairs of parameters $v_1 = (\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \in \mathcal{M}_p$ and $v_2 = (\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$, this divergence is given by $\delta_{\mathcal{M}_p}(v_1, v_2) = \frac{1}{2}(\delta_{\mathrm{KL}}(v_1, v_2) + \delta_{\mathrm{KL}}(v_2, v_1))$ where $\delta_{\mathrm{KL}}(v_1, v_2) = \frac{1}{2}\left(\mathrm{Tr}\left(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1\right) + \Delta\boldsymbol{\mu}^T \boldsymbol{\Sigma}_2^{-1} \Delta\boldsymbol{\mu} + \log\left(\frac{|\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|}\right) - p\right)$. The center of mass of $\{v_i\}_{i=1}^M$ is the solution of $\mathrm{minimize}_{v \in \mathcal{M}_p} \sum_{i=1}^M \delta_{\mathcal{M}_p}(v, v_i)$. Then, two *Nearest centroïd classifiers* are derived using two estimators: $\boldsymbol{\Sigma}^{\mathrm{G},\boldsymbol{\mu}=\mathbf{0}}$ and the MLE of the Gaussian distribution $(\boldsymbol{\mu}^{\mathrm{G}}, \boldsymbol{\Sigma}^{\mathrm{G}})$.

Finally, the proposed *Nearest centroïd classifier* on $\mathcal{M}_{p,n}$ relies on the symmetrized KL divergence (39) between NC-MSGs. The center of mass is computed as explained in the subsection VI-B and the estimation is described in Section V with the L2 penalty for the regularization. For initialization, we used the arithmetic mean, i.e. given a set of parameters $\{\theta_i \triangleq (\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, \boldsymbol{\tau}_i)\}_{i=1}^M$, with $\theta_{\mathrm{init}} = (\boldsymbol{\mu}_{\mathrm{mean}}, \boldsymbol{\Sigma}_{\mathrm{mean}}, N(\boldsymbol{\tau}_{\mathrm{mean}}))$, where $\boldsymbol{\mu}_{\mathrm{mean}} = \frac{1}{M} \sum_{i=1}^M \boldsymbol{\mu}_i$, $\boldsymbol{\Sigma}_{\mathrm{mean}} = \frac{1}{M} \sum_{i=1}^M \boldsymbol{\Sigma}_i$, $\boldsymbol{\tau}_{\mathrm{mean}} = \frac{1}{M} \sum_{i=1}^M \boldsymbol{\tau}_i$ and N is the normalization function: $\forall \boldsymbol{x} = (x_i)_{1 \le i \le n} \in (\mathbb{R}_*^+)^n$, $N(\boldsymbol{x}) = \left(\prod_{i=1}^n x_i\right)^{-\frac{1}{n}} \boldsymbol{x}$.

The data are divided into two sets: a training set and a test set with 485 649 and 122 614 batches respectively [29]. Among the six *Nearest centroïd classifiers*, only the one on $\mathcal{M}_{p,n}$ has a hyperparameter which the parameter β of the regularized NLL (25). Several values of β are tested on a training set and a validation set, and both are subsets of the original training set. The performance is measured with the
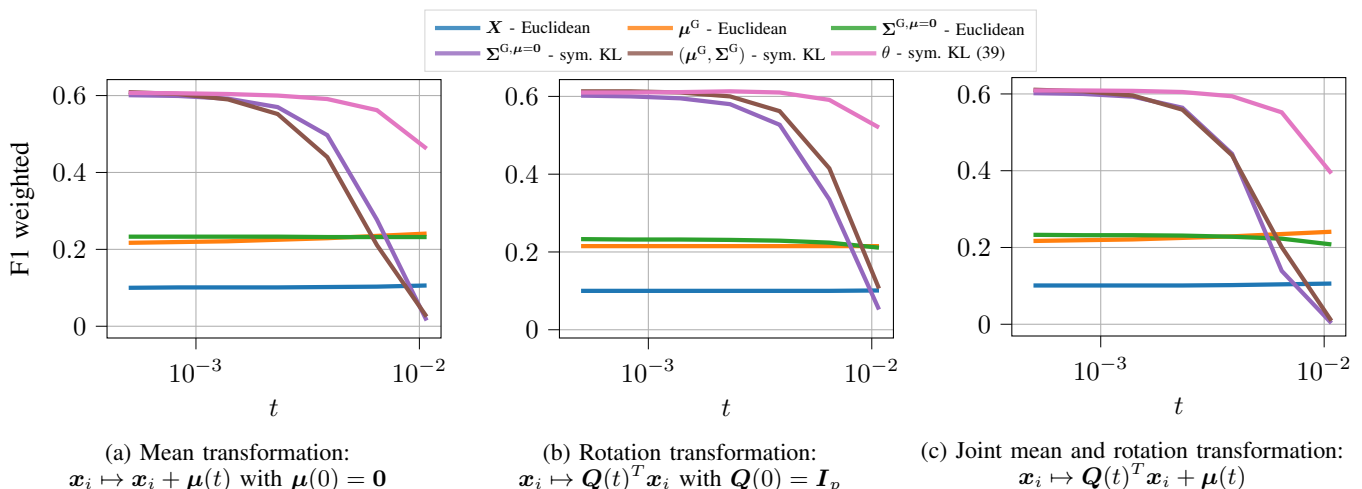
Fig. 8: "F1 weighted" metric versus the parameter t associated with three transformations applied to the test set of the *Breizhcrops* dataset. The different *Nearest centroïd classifiers* estimate the barycentres on the training data without transformations. Then, the classification is performed on the test set with three transformations. For $t = 0$, the test set is not transformed, and the larger t is, the more the test set is transformed. Six different *Nearest centroïd classifiers* are compared: each is a combination of an estimator, a divergence, and its associated center of mass computation. The proposed one is denoted "θ - sym. KL". The latter uses the Equations (26), (39) and (40) for the estimation, the divergence and the center of mass computation respectively. The regularization is the L2 penalty from Table I and β is fixed at 10^{-11}.

"F1 weighted" metric used in [29] and is plotted in Figure 7. The value of β with the highest "F1 weighted" metric is 10^{-11}. Hence, we use this value in the rest of the paper. Then, we propose an experiment to illustrate Proposition 8 on the invariance of the estimation of textures under rigid transformations. Indeed, we train the six *Nearest centroïd classifiers* on a subset of the original training set and apply them to the full test set with a rigid transformation. Thus, the more a *Nearest centroïd classifier* is robust to these rigid transformations, the better the "F1 weighted" metric. Given $t \in [0, 1]$, three different rigid transformations are performed: transformation of the mean $\boldsymbol{x}_i \mapsto \boldsymbol{x}_i + \boldsymbol{\mu}(t)$ with $\boldsymbol{\mu}(t) = t \, \boldsymbol{a}$ for a given $\boldsymbol{a} \in \mathbb{R}^p$, rotation transformation $\boldsymbol{x}_i \mapsto \boldsymbol{Q}(t)^T \boldsymbol{x}_i$ with $\boldsymbol{Q}(t) = \exp(t\boldsymbol{\xi})$ for a given skew-symmetric $\boldsymbol{\xi} \in \mathbb{R}^{p \times p}$ (hence $\boldsymbol{Q}(t) \in \mathcal{O}_p$), and the joint mean and rotation transformation $\boldsymbol{x}_i \mapsto \boldsymbol{Q}(t)^T \boldsymbol{x}_i + \boldsymbol{\mu}(t)$. It should be noted that at $t = 0$, the data are left unchanged. The results are presented in Figure 8.

The conclusions of these experiments are fourfold. First, the proposed *Nearest centroïd classifier* applies to large-scale datasets such as the *Breizhcrops* dataset. Second, the regularization proposed in Section V is important to get good classification performance. Indeed, we observe from Figure 7 that if β is too small, then the "F1 weighted" metric becomes very low. Also, if β is too large, then the "F1 weighted" metric also becomes very low. Third, using KL divergences and their associated centers of mass to classify estimators gives much better performance than the classical Euclidean distance. Indeed, even when data do not undergo rigid transformations, *Nearest centroïd classifiers* based on KL divergences outperform Euclidean *Nearest centroïd classifiers* in Figure 8. Fourth, considering NC-MSGs, as well as its KL divergence, instead of the Gaussian distribution, is interesting to classify time series especially when rigid transformations are applied to

the data. Indeed, in Figure 8, we observe a large performance improvement when data are considered distributed from a NC-MSG and undergo rigid transformations.

VIII. CONCLUSION

In this paper, we proposed a Riemannian gradient descent algorithm based on the Fisher-Rao information geometry of the NC-MSG. This algorithm is leveraged for two problems: parameter estimation and computation of centers of mass. The estimation problem of the NC-MSG is not straightforward. Indeed, a major issue is that the existence of a solution to the NLL minimization problem is not guaranteed. To overcome this issue, we proposed a class of regularized NLLs that make the trade-off between a white Gaussian distribution and the NC-MSG. These functions are guaranteed to have a minimum, and this result holds without conditions on the samples. Furthermore, we derived the KL divergence between NC-MSGs which enabled us to define the centers of mass of NC-MSGs as minimization problems. The latter is solved using the proposed Riemannian gradient descent. Simulations have shown that the proposed Riemannian gradient descent is fast on both minimization problems. Also, a *Nearest centroïd classifier* based on the KL divergence has been implemented. It has been applied on the large-scaled dataset *Breizhcrops* and showed robustness to transformations of the test set.

REFERENCES

[1] F. Kai-Tai and Z. Yao-Ting, *Generalized multivariate analysis*, vol. 19, Science Press Beijing and Springer-Verlag, Berlin, 1990.

[2] R. A. Maronna, "Robust M-estimators of multivariate location and scatter," *Ann. Statist.*, vol. 4, no. 1, pp. 51–67, 01 1976.

[3] E. Ollila, D. E. Tyler, V. Koivunen, and H. V. Poor, "Complex Elliptically Symmetric distributions: Survey, new results and applications," *IEEE Transactions on Signal Processing*, vol. 60, no. 11, pp. 5597–5625, 2012.

[4] A. Wiesel, "Unified framework to regularized covariance estimation in scaled gaussian models," *IEEE Transactions on Signal Processing*, vol. 60, no. 1, pp. 29–38, 2011.

[5] D. E. Tyler, "A distribution-free M-estimator of multivariate scatter," *Ann. Statist.*, vol. 15, no. 1, pp. 234–251, 03 1987.

[6] E. Conte, A. De Maio, and G. Ricci, "Recursive estimation of the covariance matrix of a compound-gaussian process and its application to adaptive cfar detection," *IEEE Transactions on signal processing*, vol. 50, no. 8, pp. 1908–1915, 2002.

[7] F. Pascal, P. Forster, J.-P. Ovarlez, and Larzabal P., "Performance analysis of covariance matrix estimates in impulsive noise," *IEEE Transactions on signal processing*, vol. 56, no. 6, pp. 2206–2217, 2008.

[8] G. Frahm and U. Jaekel, "Tyler's m-estimator, random matrix theory, and generalized elliptical distributions with applications to finance," *Random Matrix Theory, and Generalized Elliptical Distributions with Applications to Finance (October 21, 2008)*, 2008.

[9] T. Zhang, X. Cheng, and A. Singer, "Marčenko–Pastur law for Tyler's m-estimator," *Journal of Multivariate Analysis*, vol. 149, pp. 114–123, 2016.

[10] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, Princeton, NJ, USA, 2008.

[11] N. Boumal, "An introduction to optimization on smooth manifolds," in Cambridge University Press, March 2023.

[12] F. Nielsen, "The many faces of information geometry," *Not. Am. Math. Soc*, vol. 69, pp. 36–45, 2022.

[13] F. Pascal, Y. Chitour, and Y. Quek, "Generalized robust shrinkage estimator and its application to stap detection problem," *IEEE Transactions on Signal Processing*, vol. 62, no. 21, pp. 5640–5651, 2014.

[14] Y. Sun, P. Babu, and D. P. Palomar, "Regularized Tyler's scatter estimator: Existence, uniqueness, and algorithms," *IEEE Transactions on Signal Processing*, vol. 62, no. 19, pp. 5143–5156, 2014.

[15] E. Ollila and D. E Tyler, "Regularized m-estimators of scatter matrix," *IEEE Transactions on Signal Processing*, vol. 62, no. 22, pp. 6059–6070, 2014.

[16] Y. Sun, P. Babu, and D. P. Palomar, "Regularized robust estimation of mean and covariance matrix under heavy-tailed distributions," *IEEE Transactions on Signal Processing*, vol. 63, no. 12, pp. 3096–3109, 2015.

[17] A. Breloy, E. Ollila, and F. Pascal, "Spectral shrinkage of Tyler's m-estimator of covariance matrix," in *2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*. IEEE, 2019, pp. 535–538.

[18] M. Yi and D. E. Tyler, "Shrinking the covariance matrix using convex penalties on the matrix-log transformation," *Journal of Computational and Graphical Statistics*, vol. 30, no. 2, pp. 442–451, 2020.

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

[20] O. Tuzel, F. Porikli, and P. Meer, "Human detection via classification on riemannian manifolds," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[21] O. Tuzel, F. Porikli, and P. Meer, "Pedestrian detection via classification on riemannian manifolds," *IEEE transactions on pattern analysis and machine intelligence*, vol. 30, no. 10, pp. 1713–1727, 2008.

[22] P. Formont, J.-P. Ovarlez, and F. Pascal, "On the use of matrix information geometry for polarimetric sar image classification," in *Matrix Information Geometry*, pp. 257–276. Springer, 2013.

[23] H. Karcher, "Riemannian center of mass and mollifier smoothing," *Communications on Pure and Applied Mathematics*, vol. 30, no. 5, pp. 509–541, 1977.

[24] M. Arnaudon, F. Barbaresco, and L. Yang, *Medians and Means in Riemannian Geometry: Existence, Uniqueness and Computation*, pp. 169–197, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[25] S. Chevallier, E. K. Kalunga, Q. Barthélemy, and E. Monacelli, "Review of riemannian distances and divergences, applied to ssvep-based bci," *Neuroinformatics*, vol. 19, no. 1, pp. 93–106, 2021.

[26] M. Calvo and J. M. Oller, "An explicit solution of information geodesic equations for the multivariate normal model," *Statistics and Risk Modeling*, vol. 9, no. 1-2, pp. 119–138, 1991.

[27] M. Tang, Y. Rong, J. Zhou, and X. Li, "Information geometric approach to multisensor estimation fusion," *IEEE Transactions on Signal Processing*, vol. 67, no. 2, pp. 279–292, 2019.

[28] A. Collas, F. Bouchard, G. Ginolhac, A. Breloy, C. Ren, and J.-P. Ovarlez, "On the use of geodesic triangles between gaussian distributions for classification problems," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 5697–5701.

[29] M. Rußwurm, C. Pelletier, M. Zollner, S. Lefèvre, and M. Körner, "Breizhcrops: A time series dataset for crop type mapping," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences ISPRS (2020)*, 2020.

[30] E. Ollila, D. E. Tyler, V. Koivunen, and H. V. Poor, "Compound-gaussian clutter modeling with an inverse gaussian texture distribution," *IEEE Signal Processing Letters*, vol. 19, no. 12, pp. 876–879, 2012.

[31] F. Pascal, Y. Chitour, J.-P. Ovarlez, P. Forster, and Larzabal P., "Covariance structure maximum-likelihood estimates in compound gaussian noise: Existence and algorithm analysis," *IEEE Transactions on Signal Processing*, vol. 56, no. 1, pp. 34–48, 2007.

[32] A. Breloy, G. Ginolhac, A. Renaux, and F. Bouchard, "Intrinsic Cramér–Rao bounds for scatter and shape matrices estimation in CES distributions," *IEEE Signal Processing Letters*, vol. 26, no. 2, pp. 262–266, 2019.

[33] F. Bouchard, A. Mian, J. Zhou, S. Said, G. Ginolhac, and Y. Berthoumieu, "Riemannian geometry for compound Gaussian distributions: application to recursive change detection," *Signal Processing*, 2020.

[34] A. Wiesel and T. Zhang, *Structured robust covariance estimation*, Now Foundations and Trends, 2015.

[35] Y. Sun, P. Babu, and D. P. Palomar, "Robust estimation of structured covariance matrix for heavy-tailed elliptical distributions," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3576–3590, 2016.

[36] B. Meriaux, C. Ren, M. N. El Korso, A. Breloy, and P. Forster, "Robust estimation of structured scatter matrices in (mis) matched models," *Signal Processing*, vol. 165, pp. 163–174, 2019.

[37] A. Collas, F. Bouchard, A. Breloy, C. Ren, G. Ginolhac, and J.-P. Ovarlez, "A Tyler-type estimator of location and scatter leveraging riemannian optimization"," in *Acoustics, Speech and Signal Processing (ICASSP), 2021 IEEE International Conference on, Toronto, Canada. June 2021*, 2021.

[38] S. Amari, *Information Geometry and Its Applications*, Springer Publishing Company, Incorporated, 1st edition, 2016.

[39] L. T. Skovgaard, "A Riemannian geometry of the multivariate Normal model," *Scandinavian Journal of Statistics*, vol. 11, no. 4, pp. 211–223, 1984.

[40] M. Pilté and F. Barbaresco, "Tracking quality monitoring based on information geometry and geodesic shooting," in *2016 17th International Radar Symposium (IRS)*, 2016, pp. 1–6.

[41] D. Maclaurin, D. Duvenaud, and Adams R.P., "Autograd: Effortless gradients in pure numpy," *AutoML workshop ICML*, 2015.

[42] J. Bradbury, R. Frostig, P. Hawkins, M.J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang, "JAX: composable transformations of Python+NumPy programs," in ., 2018.

[43] A. Wiesel, "Geodesic convexity and covariance estimation," *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6182–6189, 2012.

[44] Esa Ollila, Daniel P. Palomar, and Frédéric Pascal, "Shrinking the eigenvalues of m-estimators of covariance matrix," *IEEE Transactions on Signal Processing*, vol. 69, pp. 256–269, 2021.

[45] M. Moakher, "A differential geometric approach to the geometric mean of symmetric positive-definite matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 26, no. 3, pp. 735–747, 2005.

[46] F. Mezzadri, "How to generate random matrices from the classical compact groups," *Notices of the AMS*, 2006.

[47] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, et al., and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[48] S. Smith, "Covariance, subspace, and intrinsic Cramér-Rao bounds," *Signal Processing, IEEE Transactions on*, vol. 53, pp. 1610–1630, 06 2005.

[49] F. Bouchard, A. Breloy, G. Ginolhac, A. Renaux, and F. Pascal, "A Riemannian framework for low-rank structured elliptical models," *IEEE Transactions on Signal Processing*, vol. 69, no. 3, pp. 1185–1199, 2021.

SUPPLEMENTARY MATERIAL:
RIEMANNIAN OPTIMIZATION FOR A NON-CENTERED MIXTURE OF SCALED GAUSSIAN DISTRIBUTIONS

ANTOINE COLLAS, ARNAUD BRELOY, CHENGFANG REN, GUILLAUME GINOLHAC, JEAN-PHILIPPE OVARLEZ

APPENDIX

A. Proof of Proposition 1: Fisher Information Metric

First, we recall the definition of the FIM. See [48] for an in-depth presentation. Let $\{\boldsymbol{x}_i\}_{i=1}^n$ be data points. Assuming that the underlying distribution admits a p.d.f., the corresponding NLL, denoted $\mathcal{L}$, maps parameters θ belonging to the parameter space of the p.d.f., denoted $\mathcal{M}$, onto $\mathbb{R}$. By denoting $T_\theta\mathcal{M}$ the tangent space of $\mathcal{M}$ at $\theta \in \mathcal{M}$, and under conditions of regularity of $\mathcal{L}$, the FIM is defined $\forall \xi, \eta \in T_\theta\mathcal{M}$ as

$$\langle \xi, \eta \rangle_\theta^{\mathcal{M}} = \mathbb{E}[\mathrm{D}\,\mathcal{L}(\theta)[\xi]\,\mathrm{D}\,\mathcal{L}(\theta)[\eta]] = \mathbb{E}[\mathrm{D}^2\,\mathcal{L}(\theta)[\xi,\eta]].$$

To derive the FIM of the mixture of scaled Gaussians given in Proposition 1, we recall classical formulas for the Gaussian distribution. The NLL at $\theta = (\boldsymbol{\mu}, \boldsymbol{\Sigma}) \in \mathcal{M}_p = \mathbb{R}^p \times \mathcal{S}_p^{++}$ and associated to one data point $\boldsymbol{x}$ is (neglecting terms not depending on θ)

$$\mathcal{L}_{\boldsymbol{x}}^g(\theta) = \frac{1}{2}\left[\log|\boldsymbol{\Sigma}| + (\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu})\right] \qquad (41)$$

Since $\mathcal{M}_p$ is an open set in the vector space $\mathbb{R}^p \times \mathcal{S}_p$, the tangent space of $\mathcal{M}_p$ at θ is $T_\theta\mathcal{M}_p = \mathbb{R}^p \times \mathcal{S}_p$. Finally, $\forall \xi = (\boldsymbol{\xi_\mu}, \boldsymbol{\xi_\Sigma}), \eta = (\boldsymbol{\eta_\mu}, \boldsymbol{\eta_\Sigma}) \in T_\theta\mathcal{M}_p$, the FIM of the Gaussian distribution associated to the NLL (41) is [39]

$$\langle \xi, \eta \rangle_\theta^{\mathcal{M}_p} = \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta_\mu} + \frac{1}{2}\mathrm{Tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma}). \qquad (42)$$

Then, we derive the FIM associated with the NLL of the mixture of scaled Gaussian distributions (5). We begin by writing (5) as a sum of Gaussian NLL (41). Indeed, $\forall \theta \in \mathcal{M}_{p,n}$, we have

$$\mathcal{L}(\theta|\{\boldsymbol{x}_i\}_{i=1}^n) = \sum_{i=1}^n (\mathcal{L}_{\boldsymbol{x}_i}^g \circ \varphi_i)(\theta),$$

where $\varphi_i(\theta) = (\boldsymbol{\mu}, \tau_i\boldsymbol{\Sigma})$. Thus, $\forall \theta \in \mathcal{M}_{p,n}$, $\forall \xi, \eta \in T_\theta\mathcal{M}_{p,n}$, and following the reasoning of [49, Proposition 6] and [33, Proposition 3.1], the FIM of the mixture of scaled Gaussian is expressed as a sum of FIM of the Gaussian distribution (42)

$$\langle \xi, \eta \rangle_\theta^{\mathcal{M}_{p,n}} = \mathbb{E}\left[\mathrm{D}^2\,\mathcal{L}(\theta|\{\boldsymbol{x}_i\}_{i=1}^n)[\xi,\eta]\right]$$

$$= \sum_{i=1}^n \mathbb{E}\left[\mathrm{D}^2(\mathcal{L}_{\boldsymbol{x}_i}^g \circ \varphi_i)(\theta)[\xi,\eta]\right]$$

$$= \sum_{i=1}^n \mathbb{E}\left[\mathrm{D}(\mathcal{L}_{\boldsymbol{x}_i}^g \circ \varphi_i)(\theta)[\xi]\,\mathrm{D}(\mathcal{L}_{\boldsymbol{x}_i}^g \circ \varphi_i)(\theta)[\eta]\right]$$

$$= \sum_{i=1}^n \mathbb{E}\left[\mathrm{D}(\mathcal{L}_{\boldsymbol{x}_i}^g(\varphi_i(\theta)))[\mathrm{D}\,\varphi_i(\theta)[\xi]]\right.$$
$$\left. \mathrm{D}(\mathcal{L}_{\boldsymbol{x}_i}^g(\varphi_i(\theta)))[\mathrm{D}\,\varphi_i(\theta)[\eta]]\right]$$

$$= \sum_{i=1}^n \langle \mathrm{D}\,\varphi_i(\theta)[\xi], \mathrm{D}\,\varphi_i(\theta)[\eta]\rangle_{\varphi_i(\theta)}^{\mathcal{M}_p}.$$

In the following, the i-th components of $\boldsymbol{\xi_\tau}$ and $\boldsymbol{\eta_\tau}$ are denoted ξ_i and η_i respectively. Therefore, the directional derivative of the function φ_i is

$$\mathrm{D}\,\varphi_i(\theta)[\xi] = (\boldsymbol{\xi_\mu}, \xi_i\boldsymbol{\Sigma} + \tau_i\boldsymbol{\xi_\Sigma}).$$

Thus, we get

$$\langle \xi, \eta \rangle_\theta^{\mathcal{M}_{p,n}} = \sum_{i=1}^n \left[\boldsymbol{\xi_\mu}^T(\tau_i\boldsymbol{\Sigma})^{-1}\boldsymbol{\eta_\mu}\right.$$

$$+ \frac{1}{2}\mathrm{Tr}\left((\tau_i\boldsymbol{\Sigma})^{-1}(\xi_i\boldsymbol{\Sigma} + \tau_i\boldsymbol{\xi_\Sigma})(\tau_i\boldsymbol{\Sigma})^{-1}(\eta_i\boldsymbol{\Sigma} + \tau_i\boldsymbol{\eta_\Sigma}))\right]$$

$$= \sum_{i=1}^n \left[\frac{1}{\tau_i}\boldsymbol{\xi_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\mu} + \frac{1}{2}p\frac{\xi_i\eta_i}{\tau_i^2} + \frac{1}{2}\frac{\xi_i}{\tau_i}\mathrm{Tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma})\right.$$

$$\left. + \frac{1}{2}\frac{\eta_i}{\tau_i}\mathrm{Tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma}) + \frac{1}{2}\mathrm{Tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma})\right]$$

$$= \sum_{i=1}^n \left(\frac{1}{\tau_i}\right)\boldsymbol{\xi_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\mu} + \frac{n}{2}\mathrm{Tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma})$$

$$+ \frac{p}{2}\left(\boldsymbol{\xi_\tau} \odot \boldsymbol{\tau}^{-1}\right)^T\left(\boldsymbol{\eta_\tau} \odot \boldsymbol{\tau}^{-1}\right)$$

$$+ \frac{1}{2}\boldsymbol{\xi_\tau}^T\boldsymbol{\tau}^{\odot-1}\mathrm{Tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma}) + \frac{1}{2}\boldsymbol{\eta_\tau}^T\boldsymbol{\tau}^{\odot-1}\mathrm{Tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma})$$

Since $\boldsymbol{\xi_\tau}, \boldsymbol{\eta_\tau} \in T_\theta\mathcal{M}_{p,n}$, we have $\boldsymbol{\xi_\tau}^T\boldsymbol{\tau}^{\odot-1} = \boldsymbol{\eta_\tau}^T\boldsymbol{\tau}^{\odot-1} = 0$. Thus, the last two terms of the last equation cancel, and the expression of the FIM from Proposition 1 is obtained

$$\langle \xi, \eta \rangle_\theta^{\mathcal{M}_{p,n}} = \sum_{i=1}^n \left(\frac{1}{\tau_i}\right)\boldsymbol{\xi_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\mu} + \frac{n}{2}\mathrm{Tr}\left(\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma}\right)$$

$$+ \frac{p}{2}\left(\boldsymbol{\xi_\tau} \odot \boldsymbol{\tau}^{-1}\right)^T\left(\boldsymbol{\eta_\tau} \odot \boldsymbol{\tau}^{-1}\right).$$

It should be noted that this formula defines an inner product on $\mathcal{E}_{p,n}$ if a transpose is added to $\boldsymbol{\xi_\Sigma}$. Thus, $\langle ., . \rangle^{\mathcal{M}_{p,n}}$ is extended $\forall \xi, \eta \in \mathcal{E}_{p,n}$ as presented in Proposition 1.

B. Proof of Proposition 2: orthogonal projection on $\mathcal{M}_{p,n}$

First of all, $\forall \theta \in \mathcal{M}_{p,n}$ the ambient space $\mathcal{E}_{p,n}$ defined in (14) is decomposed into two complementary subspaces

$$\mathcal{E}_{p,n} = T_\theta\mathcal{M}_{p,n} + T_\theta^\perp\mathcal{M}_{p,n} \qquad (43)$$

where $T_\theta\mathcal{M}_{p,n}$ is the tangent space at θ defined in (15) and $T_\theta^\perp\mathcal{M}_{p,n}$ is the orthogonal complement

$$T_\theta^\perp\mathcal{M}_{p,n} = \left\{\xi \in \mathcal{E}_{p,n} : \langle \xi, \eta \rangle_\theta^{\mathcal{M}_{p,n}} = 0, \forall \eta \in T_\theta\mathcal{M}_{p,n}\right\}. \qquad (44)$$

It can be checked that this orthogonal complement is

$$T_\theta^\perp\mathcal{M}_{p,n} = \{\boldsymbol{0}\} \times \mathcal{A}_p \times \{\alpha\boldsymbol{\tau} : \alpha \in \mathbb{R}_*^+\} \qquad (45)$$

where $\mathcal{A}_p$ is the set of $p \times p$ skew-symmetric matrices. Indeed, the elements of (45) verify the definition (44) and $\dim(\mathcal{E}_{p,n}) = \dim(T_\theta \mathcal{M}_{p,n}) + \dim(T_\theta^\perp \mathcal{M}_{p,n})$. Using the equations (43) and (45), the orthogonal projection of $\xi = (\boldsymbol{\xi_\mu}, \boldsymbol{\xi_\Sigma}, \boldsymbol{\xi_\tau}) \in \mathcal{E}_{p,n}$ onto $T_\theta \mathcal{M}_{p,n}$ is

$$P_\theta^{\mathcal{M}_{p,n}}(\xi) = \left(\boldsymbol{\xi_\mu}, \boldsymbol{\xi_\Sigma} - \boldsymbol{A}, \boldsymbol{\xi_\tau} - \alpha \boldsymbol{\tau} \right)$$

where $\boldsymbol{A} \in \mathcal{A}_p$ and $\alpha \in \mathbb{R}_*^+$ have to be determined. Furthermore, $\forall \eta = (\boldsymbol{0}, \boldsymbol{\eta_\Sigma}, \beta \boldsymbol{\tau}) \in T_\theta^\perp \mathcal{M}_{p,n}$ with $\beta \in \mathbb{R}_*^+$, we must have

$$\langle P_\theta^{\mathcal{M}_{p,n}}(\xi), \eta \rangle_\theta^{\mathcal{M}_{p,n}} = 0.$$

This induces that

$$\begin{cases} \boldsymbol{\xi_\Sigma} - \boldsymbol{A} = \mathrm{sym}(\boldsymbol{\xi_\Sigma}) \\ \alpha = \frac{\boldsymbol{\xi_\tau}^T \boldsymbol{\tau}^{\odot -1}}{n} \end{cases}$$

where $\mathrm{sym}(\boldsymbol{\xi_\Sigma}) = \frac{1}{2}\left(\boldsymbol{\xi_\Sigma} + \boldsymbol{\xi_\Sigma}^T \right)$. Thus, we get the orthogonal projection from Proposition 2.

C. Proof of Proposition 3: Levi Civita connection on $\mathcal{M}_{p,n}$

First of all, the FIM defined in Proposition 1 is rewritten with a function g. Indeed, let $\theta \in \mathcal{M}_{p,n}$ and ξ, η be smooth vector fields on $\mathcal{M}_{p,n}$, the function g is defined as

$$g_\theta(\xi, \eta) = \langle \xi, \eta \rangle_\theta^{\mathcal{M}_{p,n}}. \tag{46}$$

This function g is of primary importance for the development of the Levi Civita connection.

We briefly introduce the Levi-Civita connection. The general theory of it can be found in [10, Ch. 5]. The Levi-Civita connection, simply denoted $\nabla : (\xi, \eta) \mapsto \nabla_\xi \eta$, is characterized by the Koszul formula. Let ν be a smooth vector field on $\mathcal{M}_{p,n}$, in our case the Koszul formula writes

$$\begin{aligned} g_\theta(\nabla_\xi \eta, \nu) - g_\theta(\mathrm{D}\,\eta[\xi], \nu) = \\ \frac{1}{2}\big(\mathrm{D}\,g_\theta[\xi](\eta, \nu) + \mathrm{D}\,g_\theta[\eta](\xi, \nu) - \mathrm{D}\,g_\theta[\nu](\xi, \eta) \big) \end{aligned} \tag{47}$$

where $\mathrm{D}\,g_\theta[\nu](\xi, \eta)$ is the directional derivative of the function $g_\cdot(\xi, \eta) : \theta \mapsto g_\theta(\xi, \eta)$. We begin by computing $\mathrm{D}\,g_\theta[\nu](\xi, \eta)$:

$$\begin{aligned} - \mathrm{D}\,g_\theta[\nu](\xi, \eta) &= \sum_{i=1}^n \left(\frac{\nu_i}{\tau_i^2} \right) \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta_\mu} \\ &+ \sum_{i=1}^n \left(\frac{1}{\tau_i} \right) \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\nu_\Sigma} \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta_\mu} \\ &+ n\,\mathrm{Tr}\left(\boldsymbol{\Sigma}^{-1} \mathrm{sym}(\boldsymbol{\xi_\Sigma} \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta_\Sigma}) \boldsymbol{\Sigma}^{-1} \boldsymbol{\nu_\Sigma} \right) \\ &+ p\left(\boldsymbol{\xi_\tau} \odot \boldsymbol{\eta_\tau} \odot \boldsymbol{\tau}^{\odot -2} \right)^T \left(\boldsymbol{\nu_\tau} \odot \boldsymbol{\tau}^{\odot -1} \right). \end{aligned}$$

Since the objective is to identify $\nabla_\xi \eta$ using (47) and the FIM from Proposition 1, the last equation needs to be rewritten. To do so, the following two terms are rewritten

$$\begin{aligned} \sum_{i=1}^n \left(\frac{\nu_i}{\tau_i^2} \right) \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta_\mu} = \\ p\left(\frac{1}{p} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta_\mu} \mathbb{1}_n \odot \boldsymbol{\tau}^{\odot -1} \right)^T \left(\boldsymbol{\nu} \odot \boldsymbol{\tau}^{\odot -1} \right), \end{aligned}$$

and, since $\boldsymbol{\nu_\Sigma} \in \mathcal{S}_p$

$$\begin{aligned} \sum_{i=1}^n \left(\frac{1}{\tau_i} \right) \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\nu_\Sigma} \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta_\mu} = \\ \sum_{i=1}^n \left(\frac{1}{\tau_i} \right) \mathrm{Tr}\left(\boldsymbol{\Sigma}^{-1} \mathrm{sym}(\boldsymbol{\eta_\mu} \boldsymbol{\xi_\mu}^T) \boldsymbol{\Sigma}^{-1} \boldsymbol{\nu_\Sigma} \right). \end{aligned}$$

Hence, we get the equation (48). We then compute $\mathrm{D}\,g_\theta[\xi](\eta, \nu)$ in equation (49). Using (48) and (49), we calculate the right-hand side of the Koszul formula (47) in equation (50). By identification of the Koszul formula (47) and orthogonal projection onto the tangent space, we get the Levi Civita connection from the Proposition 3.

D. Proof of Proposition 4: Riemannian gradient on $\mathcal{M}_{p,n}$

Let $f : \mathcal{M}_{p,n} \to \mathbb{R}$ be a smooth function and θ be a point in $\mathcal{M}_{p,n}$. We present the correspondence between the Euclidean gradient of f (which can be computed using automatic differentiation libraries such as Autograd [41] and JAX [42]) and the Riemannian gradient associated with the FIM defined in Proposition 1. The Euclidean gradient $\mathrm{grad}\,f(\theta) = (\boldsymbol{G_\mu}, \boldsymbol{G_\Sigma}, \boldsymbol{G_\tau})$ of f at $\theta \in \mathcal{M}_{p,n}$ is defined as the unique element in $\mathbb{R}^p \times \mathbb{R}^{p \times p} \times \mathbb{R}^n$ such that $\forall \xi \in \mathbb{R}^p \times \mathbb{R}^{p \times p} \times \mathbb{R}^n$

$$\mathrm{D}\,f(\theta)[\xi] = \langle \mathrm{grad}\,f(\theta), \xi \rangle_\theta = \boldsymbol{G_\mu}^T \boldsymbol{\xi_\mu} + \mathrm{Tr}\left(\boldsymbol{G_\Sigma}^T \boldsymbol{\xi_\Sigma} \right) + \boldsymbol{G_\tau}^T \boldsymbol{\xi_\tau}.$$

Then, the Riemannian gradient $\mathrm{grad}_{\mathcal{M}_{p,n}} f(\theta) = (\boldsymbol{G_\mu}^{\mathcal{M}_{p,n}}, \boldsymbol{G_\Sigma}^{\mathcal{M}_{p,n}}, \boldsymbol{G_\tau}^{\mathcal{M}_{p,n}})$ is defined as the unique element in $T_\theta \mathcal{M}_{p,n}$ such that $\forall \xi \in T_\theta \mathcal{M}_{p,n}$

$$\mathrm{D}\,f(\theta)[\xi] = \langle \mathrm{grad}_{\mathcal{M}_{p,n}} f(\theta), \xi \rangle_\theta^{\mathcal{M}_{p,n}}.$$

Hence, $\forall \xi \in T_\theta \mathcal{M}_{p,n}$, we get that

$$\begin{aligned} \mathrm{D}\,f(\theta)[\xi] &= \boldsymbol{G_\mu}^T \boldsymbol{\xi_\mu} + \mathrm{Tr}\left(\boldsymbol{G_\Sigma}^T \boldsymbol{\xi_\Sigma} \right) + \boldsymbol{G_\tau}^T \boldsymbol{\xi_\tau} \\ &= \left(\sum_{i=1}^n \frac{1}{\tau_i} \right) \left(\left(\sum_{i=1}^n \frac{1}{\tau_i} \right)^{-1} \boldsymbol{\Sigma} \boldsymbol{G_\mu} \right)^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \\ &\quad + \frac{n}{2} \mathrm{Tr}\left(\boldsymbol{\Sigma}^{-1} \left(\frac{2}{n} \boldsymbol{\Sigma} \boldsymbol{G_\Sigma} \boldsymbol{\Sigma} \right)^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\Sigma} \right) \\ &\quad + \frac{p}{2} \left(\boldsymbol{\tau}^{\odot -1} \odot \left(\frac{2}{p} \boldsymbol{\tau}^{\odot 2} \odot \boldsymbol{G_\tau} \right) \right)^T \left(\boldsymbol{\tau}^{\odot -1} \odot \boldsymbol{\xi_\tau} \right) \\ &= \left(\sum_{i=1}^n \frac{1}{\tau_i} \right) \boldsymbol{\eta_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} + \frac{n}{2} \mathrm{Tr}\left(\boldsymbol{\Sigma}^{-1} \boldsymbol{\eta_\Sigma}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\Sigma} \right) \\ &\quad + \frac{p}{2} \left(\boldsymbol{\tau}^{\odot -1} \odot \boldsymbol{\eta_\tau} \right)^T \left(\boldsymbol{\tau}^{\odot -1} \odot \boldsymbol{\xi_\tau} \right) \end{aligned}$$

where

$$\begin{aligned} \eta &= (\boldsymbol{\eta_\mu}, \boldsymbol{\eta_\Sigma}, \boldsymbol{\eta_\tau}) \\ &= \left(\left(\sum_{i=1}^n \frac{1}{\tau_i} \right)^{-1} \boldsymbol{\Sigma} \boldsymbol{G_\mu}, \frac{2}{n} \boldsymbol{\Sigma} \boldsymbol{G_\Sigma} \boldsymbol{\Sigma}, \frac{2}{p} \boldsymbol{\tau}^{\odot 2} \odot \boldsymbol{G_\tau} \right). \end{aligned}$$

To get the Riemannian gradient, it remains to project η into the tangent space $T_\theta \mathcal{M}_{p,n}$ using the orthogonal projection $P_\theta^{\mathcal{M}_{p,n}}$. Thus, we get the Riemannian gradient from the Proposition 4.

$$
-\,\mathrm{D}\,g_\theta[\nu](\xi,\eta) = \sum_{i=1}^n \left(\frac{1}{\tau_i}\right)\mathrm{Tr}\left(\boldsymbol{\Sigma}^{-1}\mathrm{sym}(\boldsymbol{\eta_\mu}\boldsymbol{\xi_\mu}^T)\boldsymbol{\Sigma}^{-1}\boldsymbol{\nu_\Sigma}\right) + n\,\mathrm{Tr}\left(\boldsymbol{\Sigma}^{-1}\mathrm{sym}\left(\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma}\right)\boldsymbol{\Sigma}^{-1}\boldsymbol{\nu_\Sigma}\right)
$$

$$
+ p\left(\frac{1}{p}\boldsymbol{\xi_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\mu}\mathbb{1}_n \odot \boldsymbol{\tau}^{\odot-1}\right)^T\left(\boldsymbol{\nu_\tau}\odot\boldsymbol{\tau}^{\odot-1}\right) + p\left(\boldsymbol{\xi_\tau}\odot\boldsymbol{\eta_\tau}\odot\boldsymbol{\tau}^{\odot-2}\right)^T\left(\boldsymbol{\nu_\tau}\odot\boldsymbol{\tau}^{\odot-1}\right)
$$
(48)
$$
= n\,\mathrm{Tr}\left(\boldsymbol{\Sigma}^{-1}\mathrm{sym}\left[\frac{1}{n}\sum_{i=1}^n\frac{1}{\tau_i}\boldsymbol{\eta_\mu}\boldsymbol{\xi_\mu}^T + \boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma}\right]\boldsymbol{\Sigma}^{-1}\boldsymbol{\nu_\Sigma}\right)
$$

$$
+ p\left(\left[\frac{1}{p}\boldsymbol{\xi_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\mu}\mathbb{1}_n + \boldsymbol{\xi_\tau}\odot\boldsymbol{\eta_\tau}\odot\boldsymbol{\tau}^{\odot-1}\right]\odot\boldsymbol{\tau}^{\odot-1}\right)^T\left(\boldsymbol{\nu_\tau}\odot\boldsymbol{\tau}^{\odot-1}\right)
$$

$$
\mathrm{D}\,g_\theta[\xi](\eta,\nu) = -\sum_{i=1}^n\left(\frac{\xi_i}{\tau_i^2}\right)\boldsymbol{\eta_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\nu_\mu} - \sum_{i=1}^n\left(\frac{1}{\tau_i}\right)\boldsymbol{\eta_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\nu_\mu} - n\,\mathrm{Tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\nu_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma})
$$

$$
- p\left(\boldsymbol{\eta_\tau}\odot\boldsymbol{\nu_\tau}\odot\boldsymbol{\tau}^{\odot-2}\right)^T\left(\boldsymbol{\xi_\tau}\odot\boldsymbol{\tau}^{\odot-1}\right)
$$
(49)
$$
= -\sum_{i=1}^n\left(\frac{1}{\tau_i}\right)\boldsymbol{\eta_\mu}^T\left(\frac{\boldsymbol{\xi_\tau}^T\boldsymbol{\tau}^{\odot-2}}{\sum_{i=1}^n\frac{1}{\tau_i}}\boldsymbol{I}_p + \boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma}\right)\boldsymbol{\Sigma}^{-1}\boldsymbol{\nu_\mu} - n\,\mathrm{Tr}\left(\boldsymbol{\Sigma}^{-1}\mathrm{sym}(\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma})\boldsymbol{\Sigma}^{-1}\boldsymbol{\nu_\Sigma}\right)
$$

$$
- p\left(\boldsymbol{\xi_\tau}\odot\boldsymbol{\eta_\tau}\odot\boldsymbol{\tau}^{\odot-2}\right)^T\left(\boldsymbol{\nu_\tau}\odot\boldsymbol{\tau}^{\odot-1}\right)
$$

$$
\frac{1}{2}\left(\mathrm{D}\,g_\theta[\xi](\eta,\nu) + \mathrm{D}\,g_\theta[\eta](\xi,\nu) - \mathrm{D}\,g_\theta[\nu](\xi,\eta)\right) =
$$

$$
\sum_{i=1}^n\left(\frac{1}{\tau_i}\right)\left[-\frac{1}{2}\left[\boldsymbol{\eta_\mu}^T\left(\frac{\boldsymbol{\xi_\tau}^T\boldsymbol{\tau}^{\odot-2}}{\sum_{i=1}^n\frac{1}{\tau_i}}\boldsymbol{I}_p + \boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma}\right) + \boldsymbol{\xi_\mu}^T\left(\frac{\boldsymbol{\eta_\tau}^T\boldsymbol{\tau}^{\odot-2}}{\sum_{i=1}^n\frac{1}{\tau_i}}\boldsymbol{I}_p + \boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma}\right)\right]\right]\boldsymbol{\Sigma}^{-1}\boldsymbol{\nu_\mu}
$$
(50)
$$
+ \frac{n}{2}\,\mathrm{Tr}\left(\boldsymbol{\Sigma}^{-1}\left[\frac{1}{n}\sum_{i=1}^n\left(\frac{1}{\tau_i}\right)\mathrm{sym}(\boldsymbol{\eta_\mu}\boldsymbol{\xi_\mu}^T) - \mathrm{sym}(\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\Sigma})\right]\boldsymbol{\Sigma}^{-1}\boldsymbol{\nu_\Sigma}\right)
$$

$$
+ \frac{p}{2}\left(\left[\frac{1}{p}\boldsymbol{\xi_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta_\mu}\mathbb{1}_n - \boldsymbol{\xi_\tau}\odot\boldsymbol{\eta_\tau}\odot\boldsymbol{\tau}^{\odot-1}\right]\odot\boldsymbol{\tau}^{\odot-1}\right)^T\left(\boldsymbol{\nu_\tau}\odot\boldsymbol{\tau}^{\odot-1}\right)
$$

E. Proof of Proposition 5: a second order retraction on $\mathcal{M}_{p,n}$

Let $\theta\in\mathcal{M}_{p,n}$, $\xi\in T_\theta\mathcal{M}_{p,n}$ and $t\in[0,t_{max}[$ where t_{max} is to be defined. We denote $r(t)=R(t\xi)$ where R is defined in Proposition 5, *i.e.*

$$
r(t) = \left(\boldsymbol{\mu} + t\boldsymbol{\xi_\mu} + \frac{t^2}{2}\left[\frac{\boldsymbol{\xi_\tau}^T\boldsymbol{\tau}^{\odot-2}}{\sum_{i=1}^n\frac{1}{\tau_i}}\boldsymbol{I}_p + \boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\right]\boldsymbol{\xi_\mu},\right.
$$

$$
\boldsymbol{\Sigma} + t\boldsymbol{\xi_\Sigma} + \frac{t^2}{2}\left(\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma} - \frac{1}{n}\sum_{i=1}^n\left(\frac{1}{\tau_i}\right)\boldsymbol{\xi_\mu}\boldsymbol{\xi_\mu}^T\right),
$$

$$
\left.N\left(\boldsymbol{\tau} + t\boldsymbol{\xi_\tau} + \frac{t^2}{2}\left(\boldsymbol{\xi_\tau}^{\odot2}\odot\boldsymbol{\tau}^{\odot-1} - \frac{1}{p}\boldsymbol{\xi_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\mu}\mathbb{1}_n\right)\right)\right),
$$

where $\forall \boldsymbol{x}\in(\mathbb{R}_*^+)^n$, N is defined as $N(\boldsymbol{x}) = (\prod_{i=1}^n x_i)^{-1/n}\boldsymbol{x}$.

The objective is to prove that r is a second-order retraction on $\mathcal{M}_{p,n}$. The different properties of the definition of a second-order retraction are verified in the following; see [10, Ch. 4 and 5] for a complete definition.

First of all, we define t_{max} such that r is a valid retraction. Indeed, r must respect some constraints of positivity,

$$
\boldsymbol{\Sigma} + t\boldsymbol{\xi_\Sigma} + \frac{t^2}{2}\left[\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma} - \frac{1}{n}\sum_{i=1}^n\left(\frac{1}{\tau_i}\right)\boldsymbol{\xi_\mu}\boldsymbol{\xi_\mu}^T\right]\succ\boldsymbol{0}, \quad(51)
$$

$$
\boldsymbol{\tau} + t\boldsymbol{\xi_\tau} + \frac{t^2}{2}\left[\boldsymbol{\xi_\tau}^{\odot2}\odot\boldsymbol{\tau}^{\odot-1} - \frac{1}{p}\boldsymbol{\xi_\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\mu}\mathbb{1}_n\right]>\boldsymbol{0}, \quad(52)
$$

where for $\boldsymbol{A}\in\mathcal{S}_p$, $\boldsymbol{A}\succ\boldsymbol{0}$ means $\boldsymbol{A}$ is positive definite and for $\boldsymbol{x}\in\mathbb{R}^n$, $\boldsymbol{x}>\boldsymbol{0}$ means the components of $\boldsymbol{x}$ are strictly positive. Of course, (51) and (52) are not necessarily respected depending on the value of t. To define the value of t_{max} such that (51) and (52) are respected, we begin by studying the eigenvalues of the left side of (51). To do so, let $\lambda^-(\boldsymbol{A})$ be the smallest eigenvalue of $\boldsymbol{A}$ and $\boldsymbol{\Sigma}(t)$ be the left side of (51). Thus, we get that

$$
\lambda^-(\boldsymbol{\Sigma}(t)) \geq \lambda^-(\boldsymbol{\Sigma}) + t\lambda^-(\boldsymbol{\xi_\Sigma})
$$
$$
+ \frac{t^2}{2}\left[\lambda^-(\boldsymbol{\xi_\Sigma}\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi_\Sigma}) - \frac{1}{n}\sum_{i=1}^n\left(\frac{1}{\tau_i}\right)\|\boldsymbol{\xi_\mu}\|_2^2\right]
$$
$$
\geq \lambda^-(\boldsymbol{\Sigma}) + t\lambda^-(\boldsymbol{\xi_\Sigma}) - \frac{t^2}{2n}\sum_{i=1}^n\left(\frac{1}{\tau_i}\right)\|\boldsymbol{\xi_\mu}\|_2^2. \quad(53)
$$

A sufficient condition to satisfy (51) is that the right side of (53) is strictly positive. This is achieved whenever t is in $[0,t_1[$ where t_1 is defined as followed

- if $\boldsymbol{\xi_\mu}\neq\boldsymbol{0}$, $t_1 = n\frac{\lambda^-(\boldsymbol{\xi_\Sigma})+\sqrt{\Delta_1}}{\sum_{i=1}^n\left(\frac{1}{\tau_i}\right)\|\boldsymbol{\xi_\mu}\|_2^2} > 0$ and $\Delta_1 = \lambda^-(\boldsymbol{\xi_\Sigma})^2 + \frac{2}{n}\lambda^-(\boldsymbol{\Sigma})\sum_{i=1}^n\left(\frac{1}{\tau_i}\right)\|\boldsymbol{\xi_\mu}\|_2^2$,
- if $\boldsymbol{\xi_\mu}=\boldsymbol{0}$, $t_1 = \frac{\lambda^-(\boldsymbol{\Sigma})}{|\lambda^-(\boldsymbol{\xi_\Sigma})|} > 0$ for $\lambda^-(\boldsymbol{\xi_\Sigma})<0$, $t_1=+\infty$ otherwise.

Lets denote the minimum value coordinate of $\boldsymbol{x}\in\mathbb{R}^n$ by $(\boldsymbol{x})_{min}$. Using the same reasoning as before, one can show that

$$\frac{d^2}{dt^2}(N \circ \boldsymbol{x})(t) = -\frac{1}{n} \left(\prod_{i=1}^{n} x_i(t) \right) \left(\dot{\boldsymbol{x}}(t)^T \boldsymbol{x}(t)^{\odot -1} \right) \left(\prod_{i=1}^{n} x_i(t) \right)^{-\frac{1}{n}-1} \left[\dot{\boldsymbol{x}}(t) - \frac{1}{n} \left(\dot{\boldsymbol{x}}(t)^T \boldsymbol{x}(t)^{\odot -1} \right) \boldsymbol{x}(t) \right]$$

$$+ \left(\prod_{i=1}^{n} x_i(t) \right)^{-\frac{1}{n}} \left[\ddot{\boldsymbol{x}}(t) - \frac{1}{n} \left(\ddot{\boldsymbol{x}}(t)^T \boldsymbol{x}(t)^{\odot -1} \right) \boldsymbol{x}(t) + \frac{1}{n} \left(\left(\dot{\boldsymbol{x}}(t)^{\odot 2} \right)^T \boldsymbol{x}(t)^{\odot -2} \right) \boldsymbol{x}(t) - \frac{1}{n} \left(\dot{\boldsymbol{x}}(t)^T \boldsymbol{x}(t)^{\odot -1} \right) \dot{\boldsymbol{x}}(t) \right] \quad (46)$$

$$= \frac{1}{n} \left(\prod_{i=1}^{n} x_i(t) \right)^{-\frac{1}{n}} \left[n\ddot{\boldsymbol{x}}(t) + \left(\left(\dot{\boldsymbol{x}}(t)^{\odot 2} \right)^T \boldsymbol{x}(t)^{\odot -2} - \ddot{\boldsymbol{x}}(t)^T \boldsymbol{x}(t)^{\odot -1} \right) \boldsymbol{x}(t) - 2 \left(\dot{\boldsymbol{x}}(t)^T \boldsymbol{x}(t)^{\odot -1} \right) \dot{\boldsymbol{x}}(t) \right.$$

$$\left. + \frac{1}{n} \left(\dot{\boldsymbol{x}}(t)^T \boldsymbol{x}(t)^{\odot -1} \right)^2 \boldsymbol{x}(t) \right],$$

where $\ddot{\boldsymbol{x}}(t) = \frac{d^2}{dt^2} \boldsymbol{x}(t)$.

$$\ddot{\boldsymbol{\tau}}(0) = \frac{d^2}{dt^2} \left(N \circ \left(\boldsymbol{\tau} + t\boldsymbol{\xi_\tau} + \frac{t^2}{2} \left(\boldsymbol{\xi_\tau}^{\odot 2} \odot \boldsymbol{\tau}^{\odot -1} - \frac{1}{p} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \mathbb{1}_n \right) \right) \right) \Big|_{t=0}$$

$$= \boldsymbol{\xi_\tau}^{\odot 2} \odot \boldsymbol{\tau}^{\odot -1} - \frac{1}{p} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \mathbb{1}_n + \frac{1}{n} \left[(\boldsymbol{\xi_\tau}^{\odot 2})^T \boldsymbol{\tau}^{\odot -2} - \left(\boldsymbol{\xi_\tau}^{\odot 2} \odot \boldsymbol{\tau}^{\odot -1} - \frac{1}{p} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \mathbb{1}_n \right)^T \boldsymbol{\tau}^{\odot -1} \right] \boldsymbol{\tau} \quad (47)$$

$$= \boldsymbol{\xi_\tau}^{\odot 2} \odot \boldsymbol{\tau}^{\odot -1} - \frac{1}{p} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \mathbb{1}_n + \frac{1}{np} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \left(\mathbb{1}_n^T \boldsymbol{\tau}^{\odot -1} \right) \boldsymbol{\tau}.$$

---

whenever t is in $[0, t_2[$, where t_2 is defined in the following, (52) is satisfied.

- If $\boldsymbol{\xi_\mu} \neq \boldsymbol{0}$, $t_2 = p \frac{(\boldsymbol{\xi_\tau})_{min} + \sqrt{\Delta_2}}{\left\| \boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{\xi_\mu} \right\|_2^2} > 0$ and $\Delta_2 = (\boldsymbol{\xi_\tau})_{min}^2 + \frac{2}{p} (\boldsymbol{\tau})_{min} \left\| \boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{\xi_\mu} \right\|_2^2$.
- If $\boldsymbol{\xi_\mu} = \boldsymbol{0}$, $t_2 = \frac{(\boldsymbol{\tau})_{min}}{|(\boldsymbol{\xi_\tau})_{min}|} > 0$ for $(\boldsymbol{\xi_\tau})_{min} < 0$, $t_2 = +\infty$ otherwise.

Hence, we get $t_{max} = \min\{t_1, t_2\} > 0$ such that $\forall t \in [0, t_{max}[, r(t) \in \mathcal{M}_{p,n}$.

Then, to be a second-order retraction, it remains to check that the three following properties are respected,

$$r(0) = \theta, \quad \dot{r}(0) = \xi, \quad \nabla_{\dot{r}} \dot{r} \Big|_{t=0} = 0, \quad (54)$$

where $\dot{r}(t) = \frac{d}{dt} r(t)$ and ∇ is the Levi-Civita connection defined in Proposition 3. The first property is easily verified. In the rest of the proof, the following notations are used: $r(t) = (\boldsymbol{\mu}(t), \boldsymbol{\Sigma}(t), \boldsymbol{\tau}(t))$, $\dot{r}(t) = (\dot{\boldsymbol{\mu}}(t), \dot{\boldsymbol{\Sigma}}(t), \dot{\boldsymbol{\tau}}(t))$ and $\ddot{r}(t) = (\ddot{\boldsymbol{\mu}}(t), \ddot{\boldsymbol{\Sigma}}(t), \ddot{\boldsymbol{\tau}}(t))$.

We verify the second property of (54) which is $\dot{r}(0) = \xi$. It is readily check that $\dot{\boldsymbol{\mu}}(0) = \boldsymbol{\xi_\mu}$ and $\dot{\boldsymbol{\Sigma}}(0) = \boldsymbol{\xi_\Sigma}$. It remains to verify that $\dot{\boldsymbol{\tau}}(0) = \boldsymbol{\xi_\tau}$. Computing the derivative of N (defined in Proposition 3) at a point $\boldsymbol{x}(t) \in (\mathbb{R}_*^+)^n$, we get that

$$\frac{d}{dt}(N \circ \boldsymbol{x})(t) =$$

$$\left[\prod_{i=1}^{n} x_i(t) \right]^{-\frac{1}{n}} \left[\dot{\boldsymbol{x}}(t) - \frac{\dot{\boldsymbol{x}}(t)^T \boldsymbol{x}(t)^{\odot -1}}{n} \boldsymbol{x}(t) \right], \quad (55)$$

where $\dot{\boldsymbol{x}}(t) = \frac{d}{dt} \boldsymbol{x}(t)$. Using this derivative and the constraints $\prod_{i=1}^{n} \tau_i = 1$ and $\boldsymbol{\xi_\tau}^T \boldsymbol{\tau}^{\odot -1} = 0$, the desired property is derived

$$\dot{\boldsymbol{\tau}}(0) = \frac{d}{dt} \left(N \circ \left(\boldsymbol{\tau} + t\boldsymbol{\xi_\tau} \right. \right.$$

$$\left. \left. + \frac{t^2}{2} \left(\boldsymbol{\xi_\tau}^{\odot 2} \odot \boldsymbol{\tau}^{\odot -1} - \frac{1}{p} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \mathbb{1}_n \right) \right) \right) \Big|_{t=0} = \boldsymbol{\xi_\tau}.$$

It remains to check the third condition of (54). Using the first two conditions of (54), we get that $\nabla_{\dot{r}} \dot{r} \Big|_{t=0} = 0$ if and only if

$$\begin{cases} \ddot{\boldsymbol{\mu}}(0) = \left[\frac{\boldsymbol{\xi_\tau}^T \boldsymbol{\tau}^{\odot -2}}{\sum_{i=1}^{n} \frac{1}{\tau_i}} \boldsymbol{I}_p + \boldsymbol{\xi_\Sigma} \boldsymbol{\Sigma}^{-1} \right] \boldsymbol{\xi_\mu}, \\ \ddot{\boldsymbol{\Sigma}}(0) = \boldsymbol{\xi_\Sigma} \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\Sigma} - \frac{1}{n} \sum_{i=1}^{n} \left(\frac{1}{\tau_i} \right) \boldsymbol{\xi_\mu} \boldsymbol{\xi_\mu}^T, \\ P_{\boldsymbol{\tau}}^{\mathcal{S}(\mathbb{R}_*^+)^n} (\ddot{\boldsymbol{\tau}}(0)) = P_{\boldsymbol{\tau}}^{\mathcal{S}(\mathbb{R}_*^+)^n} \left(\boldsymbol{\xi_\tau}^{\odot 2} \odot \boldsymbol{\tau}^{\odot -1} - \frac{1}{p} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \mathbb{1}_n \right), \end{cases}$$

where, $\forall \boldsymbol{\xi} \in \mathbb{R}^n$, $P_{\boldsymbol{\tau}}^{\mathcal{S}(\mathbb{R}_*^+)^n}(\boldsymbol{\xi}) = \boldsymbol{\xi} - \frac{\boldsymbol{\xi}^T \boldsymbol{\tau}^{\odot -1}}{n} \boldsymbol{\tau}$. It is readily checked that the first two conditions are met. Thus, only the third condition remains to be verified. To do so, we differentiate (55) to get the second derivative of N in equation (46). Using this derivative and the constraints $\prod_{i=1}^{n} \tau_i = 1$ and $\boldsymbol{\xi_\tau}^T \boldsymbol{\tau}^{\odot -1} = 0$, the expression of $\ddot{\boldsymbol{\tau}}(0)$ is derived in the equation (47). Using the linearity of the projection $P_{\boldsymbol{\tau}}^{\mathcal{S}(\mathbb{R}_*^+)^n}$, (47) implies that

$$P_{\boldsymbol{\tau}}^{\mathcal{S}(\mathbb{R}_*^+)^n} (\ddot{\boldsymbol{\tau}}(0)) = P_{\boldsymbol{\tau}}^{\mathcal{S}(\mathbb{R}_*^+)^n} \left(\boldsymbol{\xi_\tau}^{\odot 2} \odot \boldsymbol{\tau}^{\odot -1} - \frac{1}{p} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \mathbb{1}_n \right)$$

$$+ P_{\boldsymbol{\tau}}^{\mathcal{S}(\mathbb{R}_*^+)^n} \left(\frac{1}{np} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \left(\mathbb{1}_n^T \boldsymbol{\tau}^{\odot -1} \right) \boldsymbol{\tau} \right).$$

Finally, one can check that $\forall \alpha \in \mathbb{R}$, $P_{\boldsymbol{\tau}}^{\mathcal{S}(\mathbb{R}_*^+)^n}(\alpha \boldsymbol{\tau}) = 0$. Hence, we get the desired expression

$$P_{\boldsymbol{\tau}}^{\mathcal{S}(\mathbb{R}_*^+)^n} (\ddot{\boldsymbol{\tau}}(0)) = P_{\boldsymbol{\tau}}^{\mathcal{S}(\mathbb{R}_*^+)^n} \left(\boldsymbol{\xi_\tau}^{\odot 2} \odot \boldsymbol{\tau}^{\odot -1} - \frac{1}{p} \boldsymbol{\xi_\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi_\mu} \mathbb{1}_n \right),$$

which completes the proof.