DISCRETE REGRESSION METHODS ON THE CONE OF POSITIVE-DEFINITE MATRICES

Nicolas Boumal[⋆†] *P.-A. Absil*[⋆]

[⋆]Center for Systems Engineering and Applied Mechanics (CESAME) and ICTEAM Institute,
Université catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium.
[†]Tel: (+32) 10 47 80 10; e-mail: nicolas.boumal@uclouvain.be.

ABSTRACT

We consider the problem of fitting a discrete curve to time-labeled data points on the set $\mathbb{P}_n$ of all n-by-n symmetric positive-definite matrices. The quality of a curve is measured by a weighted sum of a term that penalizes its lack of fit to the data and a regularization term that penalizes speed and acceleration. The corresponding objective function depends on the choice of a Riemannian metric on $\mathbb{P}_n$. We consider the Euclidean metric, the Log-Euclidean metric and the affine-invariant metric. For each, we derive a numerical algorithm to minimize the objective function. We compare these in terms of reliability and speed, and we assess the visual appearance of the solutions on examples for $n = 2$. Notably, we find that the Log-Euclidean and the affine-invariant metrics tend to yield similar—and sometimes identical—results, while the former allows for much faster and more reliable algorithms than the latter.

***Index Terms*—** Positive-definite matrices, non-parametric regression, Riemannian metrics, finite differences.

1. INTRODUCTION

We address the problem of fitting curves to data on the set $\mathbb{P}_n$ of n-by-n symmetric positive-definite (s.p.d.) matrices. Specifically, the data of our problem consists in N matrices $p_i \in \mathbb{P}_n$ with time labels t_i such that $t_1 \leq \cdots \leq t_N$. The goal is to find a curve on $\mathbb{P}_n$ that simultaneously (i) fits the data and (ii) is sufficiently "smooth". The measures of fitting and smoothness are conveniently defined in terms of a Riemannian metric on the set $\mathbb{P}_n$. We are thus faced with a curve fitting problem on a Riemannian manifold.

There is a general interest in numerical algorithms capable of fitting "smooth" curves to time-labeled data points. While the problem is well described for data belonging to a Euclidean space, the case where the data belongs to a manifold usually proves more challenging. General frameworks and algorithms for regression on Riemannian manifolds have recently been proposed [1, 2, 3]. The case of $\mathbb{P}_n$ is of particular interest as it appears in several applications, notably in Diffusion Tensor Imaging [4].

In this work, we consider discrete curves $\gamma = (\gamma_1, \ldots, \gamma_{N_\mathrm{d}})$ in the curve space $\Gamma = \mathbb{P}_n \times \cdots \times \mathbb{P}_n$, i.e., sequences of N_d s.p.d. matrices. Each matrix γ_i corresponds to a fixed time τ_i such that $t_1 = \tau_1 < \cdots < \tau_{N_\mathrm{d}} = t_N$. For ease of notation, we will only consider evenly spaced discretization times τ_i with spacing $\Delta\tau$. No significant difficulty arises with unevenly spaced τ_i's.

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors. The first author is a F.R.S.-FNRS Research Fellow.

Endowed with a Riemannian metric—whose choice will be discussed below—$\mathbb{P}_n$ becomes a Riemannian manifold. Building upon prior work [1], we define the optimal regression curve across data on a Riemannian manifold $\mathcal{M}$ as the minimizer of an objective function E over $\mathcal{M}^{N_\mathrm{d}}$, where E is defined as:

$$E(\gamma) = \frac{1}{2}\sum_{i=1}^{N} \left\|\mathrm{Log}_{p_i}\left(\gamma_{s_i}\right)\right\|_{p_i}^2 + \frac{\lambda}{2}\sum_{i=1}^{N_\mathrm{d}-1} \Delta\tau \left\|\frac{\mathrm{Log}_{\gamma_i}\left(\gamma_{i+1}\right)}{\Delta\tau}\right\|_{\gamma_i}^2 + \frac{\mu}{2}\sum_{i=2}^{N_\mathrm{d}-1} \Delta\tau \left\|\frac{\mathrm{Log}_{\gamma_i}\left(\gamma_{i+1}\right) + \mathrm{Log}_{\gamma_i}\left(\gamma_{i-1}\right)}{\Delta\tau^2}\right\|_{\gamma_i}^2. \quad (1)$$

In this formula, Log is the logarithmic map on $\mathcal{M}$, i.e., Log_x is the reciprocal of the Riemannian exponential Exp_x [5]. The indices s_i, $i = 1, \ldots, N$, are chosen such that τ_{s_i} is closest (ideally equal) to t_i. $\|\cdot\|_x$ is the norm induced by the Riemannian metric at $x \in \mathcal{M}$. The first term penalizes misfit between γ and the data points p_i. The second term, weighted by $\lambda \geq 0$, penalizes speed along the curve. The third term, weighted by $\mu \geq 0$, penalizes acceleration along γ.

Given a set of data points, one can choose to favor interpolation over smoothness (or the other way around) by tuning the parameters λ and μ appearing in the objective function E. The influence of these parameters is best described in terms of their role in the continuous regression problem from which E originates [1, §1]. In particular, (i) for $\lambda = 0,\ \mu > 0$, the solutions are approximating cubic splines, and (ii) for $\lambda > 0,\ \mu = 0$, the solutions are piecewise geodesic, with breaking points at the data time labels [3]. In the former case, as μ goes to infinity, the solution goes to a regression geodesic. In the latter case, as λ goes to infinity, the solution goes to a single point: the Riemannian mean [6] of the data points. In both cases, as the nonnegative parameter goes to zero, the solution nears an interpolatory curve. As we will see in the results section, our discrete formulation exhibits similar behavior.

Note that when $\mathrm{Log}_a(b) = b - a$, which is the case for Euclidean spaces, E simplifies to:

$$E(\gamma) = \frac{1}{2}\sum_{i=1}^{N} \|p_i - \gamma_{s_i}\|^2 + \frac{\lambda}{2}\sum_{i=1}^{N_\mathrm{d}-1} \Delta\tau \left\|\frac{\gamma_{i+1} - \gamma_i}{\Delta\tau}\right\|^2 + \frac{\mu}{2}\sum_{i=2}^{N_\mathrm{d}-1} \Delta\tau \left\|\frac{\gamma_{i+1} - 2\gamma_i + \gamma_{i-1}}{\Delta\tau^2}\right\|^2. \quad (2)$$

In this formula, one can recognize the usual first order finite differences for velocity and acceleration.

We let $\mathbb{S}_n$ denote the set of real, symmetric matrices of size n and $\mathbb{P}_n = \{A \in \mathbb{S}_n : x^\top A x > 0\ \forall x \in \mathbb{R}^n, x \neq 0\}$ the set of s.p.d. matrices of size n. The embedding space $\mathbb{S}_n$ is endowed with the usual Euclidean metric

$$\langle H_1, H_2 \rangle = \mathrm{trace}\left(H_2^\top H_1\right). \tag{3}$$

The associated norm is the Frobenius norm, $\|H\| = \sqrt{\langle H, H \rangle}$.

The set of s.p.d. matrices is an open convex subset of $\mathbb{S}_n$ [7], hence giving $\mathbb{P}_n$ a Riemannian manifold structure by restricting the above metric to $\mathbb{P}_n$ would yield a non-complete manifold. The set $\mathbb{P}_n$ may be endowed with a different Riemannian metric in order to make it complete. Arsigny et al. study two such metrics, namely the *affine-invariant metric* and the *Log-Euclidean metric* [8]. Intuitively, they both stretch the limits of $\mathbb{P}_n$ to infinity. Since these metrics are not the restriction of the metric on $\mathbb{S}_n$ to $\mathbb{P}_n$, endowed with either of these metrics, $\mathbb{P}_n$ is not a Riemannian submanifold of $\mathbb{S}_n$. Regardless of the metric though, $\mathbb{P}_n$ being an open subset of $\mathbb{S}_n$, the tangent space to $\mathbb{P}_n$ at any matrix in $\mathbb{P}_n$ can be identified with the set $\mathbb{S}_n$.

For each of the aforementioned three metrics, we provide an algorithm to solve the associated regression problem. As we will see, the choice of a metric has a strong impact on the objective function, and hence also on the optimal regression curve.

2. THE EUCLIDEAN METRIC

We first consider the case where $\mathbb{P}_n$ is a Riemannian submanifold of $\mathbb{S}_n$, i.e., it is endowed with the metric (3) of $\mathbb{S}_n$. Since the Frobenius norm is convex, the problem of minimizing E (2) under the constraints $\gamma_i \in \mathbb{P}_n$ is convex. But the admissible set Γ—i.e., the curve space—is not complete. Some problems may thus not have a solution. A pragmatic approach consists in minimizing E under the softened constraint that the γ_i's are semipositive definite. This way, the admissible set is complete and the whole problem can be solved easily, efficiently and reliably using solvers such as SeDuMi [9]. If the solution to the relaxed problem belongs to Γ, we have found a legitimate solution of our original problem. However, there is no guarantee that will be the case.

3. THE LOG-EUCLIDEAN METRIC

Arsigny et al. [8] propose a different Riemannian metric on $\mathbb{P}_n$, based on the observation that the restricted matrix exponential

$$\exp : \mathbb{S}_n \to \mathbb{P}_n : H \mapsto \exp(H) = \sum_{k=0}^{\infty} \frac{H^k}{k!}$$

is a smooth diffeomorphism whose inverse mapping is the (principal) matrix logarithm $\log$. They introduce new summation and scaling operators (for $A, B \in \mathbb{P}_n$ and $\alpha \in \mathbb{R}$):

$$A \oplus B = \exp(\log(A) + \log(B)),$$
$$\alpha \otimes A = \exp(\alpha \log(A)) = A^\alpha.$$

By construction, $\exp : (\mathbb{S}_n, +, .) \to (\mathbb{P}_n, \oplus, \otimes)$ is a vector space isomorphism. Arsigny et al. use this to endow $\mathbb{P}_n$ with the metric

$$\langle H_1, H_2 \rangle_A = \langle \mathrm{D}\log(A)[H_1], \mathrm{D}\log(A)[H_2] \rangle, \tag{4}$$

termed the Log-Euclidean metric. $\mathrm{D}\log(A)[H]$ denotes the directional derivative of the matrix logarithm at A along H and the inner product on the right hand side is the Euclidean metric on $\mathbb{S}_n$. Endowed with (4), $\mathbb{P}_n$ is a complete Riemannian manifold [8, Prop. 3.4]. Although the metric (4) takes a complicated form, the actual computations we need are made very easy thanks to the vector space structure. It can indeed be proven that minimizing E (1) with the Log-Euclidean metric comes down to minimizing E (2) in the log-domain, which is the vector space $\mathbb{S}_n$. More precisely, to solve the smooth regression problem on $\mathbb{P}_n$ endowed with the Log-Euclidean metric, simply:

1. Compute the new data points $\tilde{p}_i = \log(p_i) \in \mathbb{S}_n$ for each original data point $p_i \in \mathbb{P}_n$;
2. Compute $\tilde{\gamma} \in \mathbb{S}_n \times \cdots \times \mathbb{S}_n$, the solution of the regression problem in $\mathbb{S}_n$ with the new data points;
3. Compute γ, the solution of the original problem, as $\gamma_i = \exp(\tilde{\gamma}_i)$.

Step 2 can be carried out by minimizing E (2) in the vector space $(\mathbb{S}_n)^{N_\mathrm{d}}$. E is a linear least-squares objective, hence it is convex and quadratic. The optimization problem is unconstrained. Furthermore, a little algebra shows that E is decoupled along the dimensions of $\mathbb{S}_n$, of which there are $\dim \mathbb{S}_n = n(n+1)/2$, and that the Hessian of the objective along each dimension is pentadiagonal and does not depend on the data. Consequently, we need only solve a pentadiagonal system of size N_d for $n(n+1)/2$ right hand sides, which is fast and reliable. Using modern quadratic programming (QP) solvers, it is also easy to add constraints to the original problem in the form of linear equalities and inequalities.

4. THE AFFINE-INVARIANT METRIC

Before the Log-Euclidean metric was introduced, Pennec et al. described another Riemannian metric on $\mathbb{P}_n$ and named it the affine-invariant metric [10]:

$$\langle H_1, H_2 \rangle_A = \left\langle A^{-1/2} H_1 A^{-1/2}, A^{-1/2} H_2 A^{-1/2} \right\rangle. \tag{5}$$

Pennec et al. give the Riemannian exponential and logarithmic maps [5] for this metric [10, §3.4]:

$$\mathrm{Exp}_A(H) = A^{1/2} \exp\left(A^{-1/2} H A^{-1/2}\right) A^{1/2},$$
$$\mathrm{Log}_A(B) = A^{1/2} \log\left(A^{-1/2} B A^{-1/2}\right) A^{1/2}.$$

To solve the regression problem with this metric, we need to minimize E (1) under the constraints $\gamma_i \in \mathbb{P}_n$ for $i = 1, \ldots, N_\mathrm{d}$. The objective is no longer convex, hence convex programming cannot help like it did in Section 2. Instead, we use a Riemannian conjugate gradient (RCG) descent method [5] outlined in Algorithm 2. Any classic step size choosing algorithm can be used. The RCG method executes a descent by stepping along geodesics on the manifold $\Gamma = \mathbb{P}_n \times \cdots \times \mathbb{P}_n$. By definition, those geodesics never leave Γ since it is complete when endowed with the (componentwise-extended) affine-invariant metric. Consequently, provided the algorithm converges, it will do so toward an admissible curve.

Algorithm 1 Directional derivative of $\log$

Input: $A \in \mathbb{P}_n$, $H \in \mathbb{S}_n$.
Output: $\mathrm{D}\log(A)[H] \in \mathbb{S}_n$.
Diagonalize: $A = UDU^\top$, $D = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, $U^\top = U^{-1}$.
Compute $\tilde{H} = U^\top H U$.
Compute $\tilde{Z}$ with $\tilde{Z}_{ij} = \begin{cases} \frac{\log(\lambda_i) - \log(\lambda_j)}{\lambda_i - \lambda_j} & \text{if } \lambda_i \neq \lambda_j, \\ \frac{1}{\lambda_i} & \text{if } \lambda_i = \lambda_j. \end{cases}$
return $U(\tilde{H} \odot \tilde{Z})U^\top$, where $\odot$ is the Hadamard product.

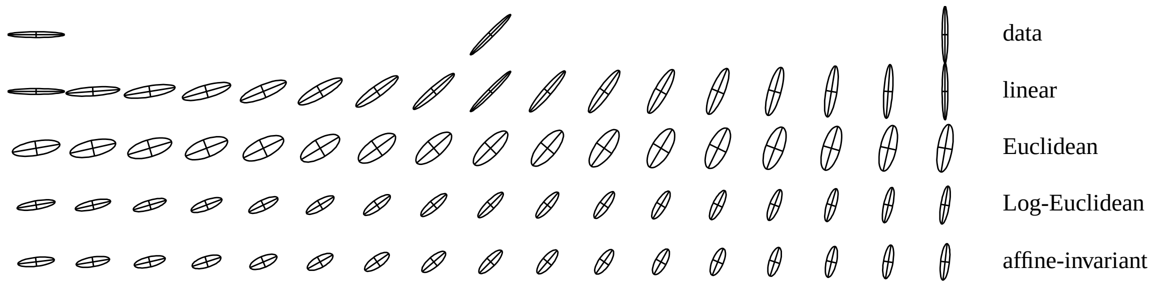

Fig. 1. $\lambda = 10^{-1}, \mu = 0$. An example of piecewise geodesic regression. The first line shows the data points. The second line shows linear interpolation between the data. The third line shows regression for the Euclidean metric and is obtained by solving a convex program. The fourth line is the solution for the Log-Euclidean metric and is computed by solving an unconstrained quadratic program. The fifth line is the result for the affine-invariant metric obtained by running the RCG algorithm with the Log-Euclidean solution as initial guess.

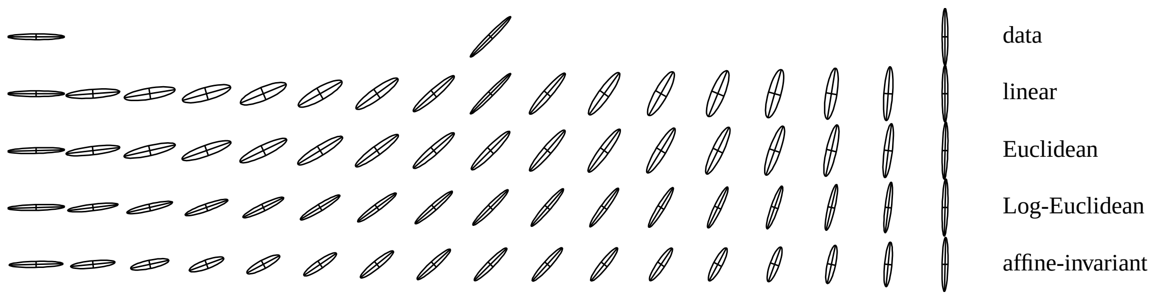

Fig. 2. $\lambda = 0, \mu = 10^{-3}$. The three methods of interest (bottom lines) give sensible results resembling approximating cubic splines.

It seems sensible to feed the (easy to compute) Log-Euclidean regression curve to the RCG algorithm as initial guess when computing the regression curve for the affine-invariant metric. Numerical evidence suggests that, when the data points p_i commute w.r.t. matrix multiplication, both the Log-Euclidean and the affine-invariant metric yield the same optimal regression curve. A partial hint to this is that the objective functions associated to both metrics are identically equal when restricted to the set of s.p.d. matrices which commute with the data points. An important fact intervening in this matter is that commuting symmetric matrices are diagonalized by a common orthogonal matrix [7, p. 23].

The descent directions in the RCG method depend on the gradient of the objective E. The latter is a linear combination of functions f and g of s.p.d. matrices evaluated at neighboring γ_i's, with:

$$f(A, B) = \| \operatorname{Log}_A(B) \|_A^2,$$
$$g(A, B, C) = \langle \operatorname{Log}_A(B), \operatorname{Log}_A(C) \rangle_A .$$

Hence, it is sufficient to derive formulas for the gradients of f and g in order to compute $\operatorname{grad} E$. A general result [1, §5.3] states that

$$\operatorname{grad}\left(X \mapsto f(X, B)\right)(A) = -2 \operatorname{Log}_A(B).$$

Since f is symmetric in its two arguments, it remains to compute the gradients of g in each of its variables. Since g makes use of the matrix logarithm, we will need a means of computing $\operatorname{Dlog}(A)[H]$, the derivative of $\log$ at $A \in \mathbb{P}_n$ along $H \in \mathbb{S}_n$. We cover that in Algorithm 1 [1]. A rather long development yields:

$$\begin{aligned}
&\operatorname{grad}\big(X \mapsto g(X, B, C)\big)(A) \\
&\quad = A\Big(\operatorname{Dlog}(A_B)\left[\operatorname{sym}\left(\log\left(C_{B^{-1}}^{-1} A_B\right)\right)\right]\Big)_B A \\
&\quad\quad + A\Big(\operatorname{Dlog}(A_C)\left[\operatorname{sym}\left(\log\left(A_C\, B_{C^{-1}}^{-1}\right)\right)\right]\Big)_C A,
\end{aligned}$$

$$\begin{aligned}
&\operatorname{grad}\big(X \mapsto g(A, X, C)\big)(B) \\
&\quad = \operatorname{grad}\big(X \mapsto g(A, C, X)\big)(B) \\
&\quad = B\Big(\operatorname{Dlog}(B_A)\left[\log(C_A)\right]\Big)_A B.
\end{aligned}$$

For brevity, we used the notation $\operatorname{sym}(X) \triangleq (X + X^\top)/2$ (symmetric part of X) and $X_Y \triangleq Y^{-1/2} X Y^{-1/2}$. Note that if X and Y are s.p.d., so is X_Y [8, §3.1]. It is useful to notice that, for any s.p.d. matrix X, the operator $\operatorname{Dlog}(X)[\cdot]$ is self-adjoint in $\mathbb{S}_n$, i.e., for all symmetric matrices H_1, H_2, we have $\langle \operatorname{Dlog}(X)[H_1], H_2 \rangle = \langle H_1, \operatorname{Dlog}(X)[H_2] \rangle$.

The gradient formulas we gave in this section are sufficient to compute the gradient of the objective function E (1) for the affine-invariant metric. Together with the initial guess suggested earlier—namely, the optimal curve for the Log-Euclidean metric—this is enough material to run Algorithm 2 on our problem.

5. RESULTS

In Figures 1–3, a 2-by-2 s.p.d. matrix A is pictured as an ellipse whose axes are aligned with the eigenvectors of A and whose axes lengths are proportional to the corresponding eigenvalues. In all three figures, the data points are the same three non-commuting s.p.d. matrices. Each figure illustrates regression across those data points for a different choice of parameters λ and μ and juxtaposes the results for all three metrics we discussed. We chose $n = 2$ to ease the representation, but our algorithms work for general n.

Figure 1 shows piecewise geodesic regression. Only the breaking points need to be computed, which is fast for all three metrics.

Figure 2 exhibits cubic-spline–like regression.

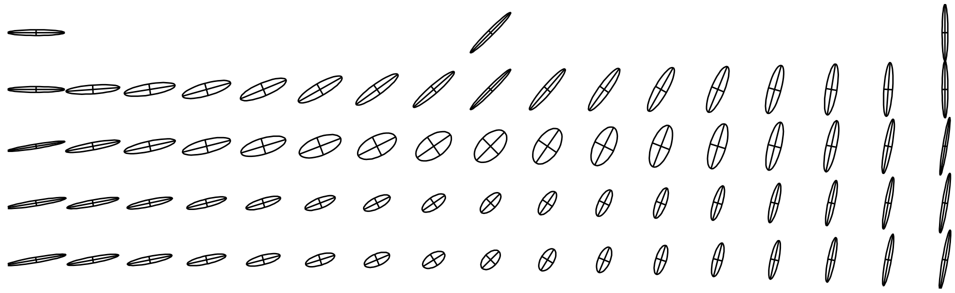

Fig. 3. $\lambda = 0, \mu = 10^3$. Geodesic-like regression. The Euclidean and Log-Euclidean metric solutions are accurate and quick to compute. The affine-invariant metric solution is noticeably slower to compute and its precision depends on the stopping criterion of Algorithm 2.

Algorithm 2 Riemannian conjugate gradient method (RCG) [5]

Input: A scalar field $E : \Gamma \to \mathbb{R}$, its gradient $\operatorname{grad} E$ and an initial guess $\gamma^{(0)} \in \Gamma$, where Γ is a Riemannian manifold.
Output: A sequence $\gamma^{(1)}, \gamma^{(2)}, \ldots$ in Γ.

$p_0 := -\operatorname{grad} E(\gamma^{(0)})$
$k := 0$
while $\operatorname{grad} E(\gamma^{(k)}) \neq 0$ **do**
  $\alpha_k :=$ choose step size (any classic line-search, e.g., [5, §4.2])
  $\gamma^{(k+1)} := \operatorname{Exp}_{\gamma^{(k)}}(\alpha_k p_k)$
  $\beta_{k+1} := \|\operatorname{grad} E(\gamma^{(k+1)})\|^2 / \|\operatorname{grad} E(\gamma^{(k)})\|^2$
  $p_{k+1} := -\operatorname{grad} E(\gamma^{(k+1)}) + \beta_{k+1} p_k$
  $k := k + 1$
end while

Figure 3 demonstrates almost-geodesic regression. Note that for the Log-Euclidean and affine-invariant metrics those geodesics can be infinitely extended on both ends without ever leaving $\mathbb{P}_n$, whereas for the Euclidean metric the extended geodesic would eventually reach the border of the cone. The Euclidean metric also produces a visible swelling effect, that was already observed by Arsigny et al. for the two-point ($N = 2$) interpolation problem [8].

The Log-Euclidean and the affine-invariant metrics yield similar results. Considering the simplicity inherent to working with the Log-Euclidean metric, we would favor it over the affine-invariant metric in applications, unless there are application-related reasons not to do so. These could be linked to the respective properties of the two metrics. Arsigny et al. provide a study of those properties [8].

6. CONCLUSIONS

We studied the problem of fitting discrete curves to data on the set of symmetric positive-definite matrices $\mathbb{P}_n$. Building upon prior work, we formulated this problem as an optimization problem on manifolds. We gave $\mathbb{P}_n$ a Riemannian manifold structure using three different metrics. For each of these metrics, we provided a numerical algorithm to solve the corresponding regression problem. For the Euclidean metric, we saw that the problem does not always have a solution in $\mathbb{P}_n$. Semidefinite programming can sometimes provide a solution, if it exists. For the Log-Euclidean metric, we took advantage of the vector space structure to reduce the problem to an unconstrained quadratic program, leading to a simple, robust and fast algorithm. For the affine-invariant metric, a lot of algebra for the computation of the gradient of the objective permitted the use of the Riemannian conjugate gradient method. The latter proved effective but can be slow, especially for high μ. Numerical experiments show that the Log-Euclidean and the affine-invariant metrics tend to yield similar curves. We therefore conclude that the Log-Euclidean metric is the metric of choice for practical applications, unless said application imposes other constraints.

7. REFERENCES

[1] Nicolas Boumal and P.-A. Absil, "A discrete regression method on manifolds and its application to data on $\mathrm{SO}(n)$," *Technical Report UCL-INMA-2010.059 (submitted), Université catholique de Louvain*, 2010.

[2] Chafik Samir, P.-A. Absil, Anuj Srivastava, and Eric Klassen, "A gradient-descent method for curve fitting on Riemannian manifolds," *Technical Report UCL-INMA-2009.023 (submitted), Université catholique de Louvain*, 2009.

[3] Luís Machado and F. Silva Leite, "Fitting smooth paths on Riemannian manifolds," *Int. J. Appl. Math. Stat.*, vol. 4, no. J06, pp. 25–53, 2006.

[4] P. Fillard, X. Pennec, V. Arsigny, and N. Ayache, "Clinical DT-MRI estimation, smoothing, and fiber tracking with log-Euclidean metrics," *IEEE Transactions on Medical Imaging*, vol. 26, no. 11, pp. 1472–1482, 2007.

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

[6] R. Ferreira, J. Xavier, J.P. Costeira, and V. Barroso, "Newton method for Riemannian centroid computation in naturally reductive homogeneous spaces," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2006.

[7] R. Bhatia, *Positive definite matrices*, Princeton University Press, 2007.

[8] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Geometric means in a novel vector space structure on symmetric positive-definite matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 29, no. 1, pp. 328, 2008.

[9] Jos F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," 1998.

[10] X. Pennec, P. Fillard, and N. Ayache, "A Riemannian framework for tensor computing," *International Journal of Computer Vision*, vol. 66, no. 1, pp. 41–66, 2006.