Regression Uncertainty on the Grassmannian

Yi Hong
University of Georgia

Xiao Yang
University of North Carolina at Chapel Hill

Roland Kwitt
University of Salzburg

Martin Styner
University of North Carolina at Chapel Hill

Marc Niethammer
University of North Carolina at Chapel Hill

Abstract

Trends in longitudinal or cross-sectional studies over time are often captured through regression models. In their simplest manifestation, these regression models are formulated in $\mathbb{R}^n$. However, in the context of imaging studies, the objects of interest which are to be regressed are frequently best modeled as elements of a Riemannian manifold. Regression on such spaces can be accomplished through *geodesic regression*. This paper develops an approach to compute confidence intervals for geodesic regression models. The approach is general, but illustrated and specifically developed for the Grassmann manifold, which allows us, e.g., to regress shapes or linear dynamical systems. Extensions to other manifolds can be obtained in a similar manner. We demonstrate our approach for regression with 2D/3D shapes using synthetic and real data.

1 Introduction

Linear regression models in Euclidean space are classical and frequently-used statistical methods [21]. However, imaging studies are often faced with objects such as shapes, diffeomorphisms, or linear dynamical systems, which are best represented as elements of a Riemannian manifold. Hence, extensions of linear regression models to such spaces, so-called *geodesic regression models*, are required and have recently been proposed [18, 14, 8]. Geodesic regression models are compact representations of trends. However, response

variables in these models are often high-dimensional, which makes it difficult to assess (1) model fit as well as (2) confidence in the estimated model parameters. In this paper, we propose an approach to quantify geodesic regression *confidence*. While, Gaussian processes [16, 3] do naturally possess the property to quantify uncertainty, the uncertainty is in the model predictions and response variables are typically scalar-valued. In contrast, we are specifically interested in response variables on Riemannian manifolds and uncertainty in the model parameters. For illustration, our approach is developed in the context of the Grassmannian, i.e., the manifold of p-dimensional linear subspaces of $\mathbb{R}^n$, but it could readily be extended to other Riemannian manifolds. This will be computationally easiest in a finite-dimensional setting, but extensions to infinite-dimensional manifolds are conceivable; see for example [22] for a related approach in image registration on the space of diffeomorphisms.

Our specific *contributions* are as follows:

- An approach to efficiently compute the Hessian of the energy of the geodesic regression problem on the Grassmannian. This approach is based on the second variation of the energy, which can be computed via the *tangent linear model (TLM)* and the *tangent linear adjoint model (TLAM)* to the best-fitting geodesic. These tangent models allow to efficiently compute Hessian-vector-products and, as a result, columns of the Hessian can be obtained via a simple forward-backward sweep [7, 13]. The inverse of this Hessian corresponds to the covariance matrix of the model parameters (the Laplace approximation to the posterior distribution).

- An approach to propagate the estimated covariance matrix of the model parameters along a fitted geodesic, inspired by a simple special case of the Kalman filter [9].

- An estimation method for Mahalanobis-like geodesic distance functions, making use of block-

diagonal covariance matrices. This estimation method properly deals with geodesic-to-measurement residuals, which are defined in separate tangent spaces. The block-diagonal structure of the covariance matrices is essential to be able to capture *localized* confidence. Commonly used homoscedastic models for geodesic regression are not appropriate in this context.

- A simple visualization of the point position marginals (as ellipses/ellipsoids) of the shapes described by the regression geodesic.

Organization. Sec. 2 motivates our approach from the perspective of linear least-squares regression in Euclidean space and details how to compute the Hessian using an adjoint method. Sec. 3 explains the related computations on the Grassmannian. Sec. 4 then presents experimental results and Sec. 5 concludes the paper with a discussion and an outlook on future work.

2 Uncertainty for Linear Regression

Given a collection of N measurements in $\mathbb{R}$ and their associated independent variables, i.e., $\{(y_i, t_i)\}_{i=1}^N$, we assume a linear regression model of the form $y_i = a + bt_i + \epsilon_i$, where a is the y-intercept and b the slope. We assume the measurement noise ϵ_i at each point is independent and identically distributed, in particular, $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$. Under this assumption, the maximum likelihood estimates (MLE) of (a, b) coincide with the least-squares solution obtained by minimizing the energy

$$E(a, b) = \frac{1}{2\sigma^2} \sum_{i=1}^N (y_i - a - bt_i)^2 = \frac{1}{2\sigma^2} \sum_{i=1}^N r_i^2 \ , \quad (1)$$

where r_i denotes the i-th measurement residual. In practice, the MLE is obtained by solving the linear system $\boldsymbol{y} = \mathbf{X}\boldsymbol{\beta}$ with $\boldsymbol{y} = [y_1, \cdots, y_N]^\top$, $\mathbf{X} = [\mathbf{1}, \mathbf{t}]$ ($\mathbf{1} = [1, \cdots, 1]^\top$, $\mathbf{t} = [t_1, \cdots, t_N]^\top$) and $\boldsymbol{\beta} = [a \ b]^\top$. We also know that the MLE $\hat{\boldsymbol{\beta}}$ follows a Gaussian $\mathcal{N}(\boldsymbol{\beta}, \mathbf{C})$, where $\mathbf{C} = [\mathbf{I}(\boldsymbol{\beta})]^{-1}$ denotes the inverse Fisher information matrix. An estimate of the covariance for the optimal parameter $\hat{\boldsymbol{\beta}}$ can be obtained from the observed Fisher information or, equivalently, by computing the Hessian of the energy in Eq. (1) which is proportional to the negative log-likelihood $-\log p(\mathbf{y}|\mathbf{X}; \boldsymbol{\beta}, \sigma^2)$.

The Hessian, $\mathbf{H}$, of the energy $E(a, b)$ in Eq. (1) is:

$$\mathbf{H} = \frac{1}{2\sigma^2} \begin{pmatrix} \frac{\partial^2 E}{\partial a^2} & \frac{\partial^2 E}{\partial a \partial b} \\ \frac{\partial^2 E}{\partial a \partial b} & \frac{\partial^2 E}{\partial b^2} \end{pmatrix} = \frac{1}{\sigma^2} \begin{pmatrix} N & \sum_i t_i \\ \sum_i t_i & \sum_i t_i^2 \end{pmatrix} \ . \quad (2)$$

By inversion of $\mathbf{H}$, we obtain an estimate $\hat{\mathbf{C}}$ of the covariance of $\hat{\boldsymbol{\beta}}$, i.e.,

$$\mathbf{H}^{-1} = \hat{\mathbf{C}} = \frac{\sigma^2}{N \sum_i (t_i - \bar{t})^2} \begin{pmatrix} \sum_i t_i^2 & -\sum_i t_i \\ -\sum_i t_i & N \end{pmatrix} \quad (3)$$

with $\bar{t} = N^{-1} \sum_i t_i$. The standard errors of the intercept (s_a) and slope (s_b) can then be read-off as the diagonal elements, i.e.,

$$s_a = \sigma \sqrt{\frac{\sum_i t_i^2}{N \sum_i (t_i - \bar{t})^2}}, \quad s_b = \frac{\sigma}{\sqrt{\sum_i (t_i - \bar{t})^2}} \ . \quad (4)$$

Note that as N increases, the uncertainty in the estimated parameters decreases.

As σ is constant, it does not change the optimization solution for $\boldsymbol{\beta}$ in Eq. (1). However, it does scale the Hessian and consequently the covariance; for that reason, it needs to be jointly estimated. The (biased) MLE of σ is given by the mean of the squared residuals r_i^2 and can be made unbiased for the linear regression problem with two degrees of freedom by multiplying the mean of the residuals by $N/(N-2)$ [4].

In practice, closed-form solutions of the Hessian cannot be easily computed for more complex energy functions, in particular, when they involve dynamic constraints. Such dynamic constraints appear naturally when dealing with geodesic equations on manifolds which may not be written in explicit form. We will therefore discuss an *alternative* solution which will allow us to extend Hessian computations to Riemannian manifolds, the Grassmannian in particular.

2.1 Hessian Estimation

In geodesic regression formulations, the energy of Eq. (1) is replaced by the sum-of-squared-residuals between measurements and the corresponding values on the geodesic. The geodesic itself is expressed in the form of a differential equation; see, e.g., [12] for a formulation on the Grassmannian. In this setting, the linear regression problem can be reformulated as minimizing the energy[1]

$$E(\mathbf{x}_0) = \frac{1}{2\sigma^2} \sum_{i=1}^N (y_i - x_1(t_i))^2, \quad (5)$$

$$\text{s.t.} \quad \dot{x}_1 = x_2, \quad \text{and} \quad \dot{x}_2 = 0,$$

over $\mathbf{x}_0 = [x_1(0), x_2(0)]^\top$, i.e., the initial conditions, including the intercept and the slope. For simplicity we assume $\{t_i\}_{i=1}^N$ are normalized to $[0, 1]$. We are now no longer optimizing over the two parameters (a, b),

---

[1]We omit the dependency of x_1 and x_2 on t, i.e., $x_1(t)$, $x_2(t)$, for readability and define $\dot{x}_i := d/dt \ x_i(t)$.

but over the initial conditions of a system of differential equations. Hence, we require variational calculus to compute the analog of a directional first and second derivative (i.e., the first and second variations). While the first variation allows us to compute optimality conditions (the functional equivalent of the Karush-Kuhn-Tucker conditions [15]), the second variation allows us to derive equations to compute Hessian-vector-products. Specifically, the first variation results in a set of adjoint equations: $\dot{\lambda}_1 = 0, \lambda_1(1) = 0$, with jumps at each measurement $\lambda_1(t_i-) = \lambda_1(t_i+) - \frac{1}{\sigma^2}(x_1(t_i) - y_i)$ and $\dot{\lambda}_2 = -\lambda_1, \lambda_2(1) = 0$, where λ_1 and λ_2 are the adjoint variables for x_1 and x_2. Integrating the state equations ($\dot{x}_1 = x_2$ and $\dot{x}_2 = 0$ with initializations $x_1(0) = x_{1,0}$ and $x_2(0) = x_{2,0}$) forward in time, followed by a backward-in-time integration of the adjoint equations allows us to compute the gradients of the energy with respect to the initial conditions, i.e., $\nabla_{x_1(0)}E = -\lambda_1(0)$ and $\nabla_{x_2(0)}E = -\lambda_2(0)$. These gradients then facilitate the numerical minimization of the regression energy in Eq. (5). Importantly, if the state equations are replaced by other geodesic equations[2] and the squared residual terms by squared distance measures on the manifold, one obtains the optimization strategy for geodesic regression. The adjoint equation system can then be considered the geodesic regression equivalent of error-backpropagation.

A similar forward-backward approach can be used to compute Hessian-vector-products which now involves the second variation of the energy [19, 20]. The second variation of the least-squares energy in the direction $\delta\mathbf{x}_0$ is defined as

$$\delta^2 E(\mathbf{x}_0; \delta\mathbf{x}_0) := \frac{\partial^2}{\partial\epsilon^2} E(\mathbf{x}_0 + \epsilon\delta\mathbf{x}_0)|_{\epsilon=0} \ . \tag{6}$$

Our goal is to derive an equation system from the second variation such that the Hessian-vector-product can be directly read-off, i.e.,

$$\delta^2 E(\mathbf{x}_0; \delta\mathbf{x}_0) = \langle \delta\mathbf{x}_0, \nabla^2 E \delta\mathbf{x}_0 \rangle \ . \tag{7}$$

Here, $\nabla^2 E$ is the Hessian of $E(\mathbf{x}_0)$, and $\nabla^2 E \delta\mathbf{x}_0$ denotes the Hessian-vector-product from the second variation. It turns out that this goal can be achieved through a forward-sweep of the tangent linear model (TLM) (i.e., the geodesic equations linearized around the optimal solution), followed by a backward sweep of the tangent linear adjoint model (TLAM) (i.e., the adjoint equations linearized around the optimal solution) with appropriate boundary conditions. Specifically, given perturbation $\delta\mathbf{x} = [\delta x_1, \delta x_2]^\top$ at the optimal solution, the equations for the *forward TLM* can

be derived[3] as:

$$\begin{aligned} \delta\dot{x}_1 - \delta x_2 &= 0, \\ \delta x_1(0) &= \delta x_{1,0}, \end{aligned} \quad \text{and} \quad \begin{aligned} \delta\dot{x}_2 &= 0, \\ \delta x_2(0) &= \delta x_{2,0} \ . \end{aligned} \tag{8}$$

The equations for the *backward TLAM* are

$$\begin{aligned} \delta\dot{\lambda}_1 &= 0, \\ \delta\lambda_1(1) &= 0, \end{aligned} \quad \text{and} \quad \begin{aligned} \delta\dot{\lambda}_2 &= -\delta\lambda_1, \\ \delta\lambda_2(1) &= 0 \ , \end{aligned} \tag{9}$$

with jumps $\delta\lambda_1(t_i-) = \delta\lambda_1(t_i+) - \frac{1}{\sigma^2}\delta x_1(t_i)$ for $i = 1, \ldots, N$. This allows computing the Hessian-vector-product as $\nabla^2 E \delta\mathbf{x}_0 = [-\delta\lambda_1(0), -\delta\lambda_2(0)]^\top$. Choosing $\delta\mathbf{x}_0 = [1,0]^\top$ and $\delta\mathbf{x}_0 = [0,1]^\top$ as the initial values for the forward TLM, we can easily compute the first and the second column of the Hessian $\nabla^2 E$. This is equivalent to the exact solution of the Hessian in Eq. (2). More importantly, it allows us to *generalize* the Hessian computation to Riemannian manifolds, as we can use it to deal with differential-equation constraints required to express the geodesic equations.

3   Regression Uncertainty on $\mathcal{G}(p, n)$

This section develops the approach for computing the Hessian for the geodesic regression energy on the Grassmannian $G(p, n)$ by following the adjoint strategy of Sec. 2.1. In particular, Sec. 3.1 reviews geodesic regression on $\mathcal{G}(p, n)$ and Sec. 3.2 develops the method to compute Hessian-vector-products. The measurement noise model is a bit more complex as (i) a homoscedastic model is in general no longer sufficient, (ii) residuals of the Euclidean model need to be computed via the Riemannian Log-map, and (iii) the tangent vectors obtained by the Log-Map (corresponding to measurement residuals) all live in their own tangent spaces. Covariance estimation and propagation is described in Sec. 3.3.

3.1   Review of Geodesic Regression on $\mathcal{G}(p, n)$

In [12] a numerical solution to perform geodesic regression on the Grassmann manifold is proposed, based on a generalization of linear least-squares regression using the previously discussed adjoint method. In particular, geodesic regression allows to capture the relationship between data points on the Grassmannian $\{\mathcal{Y}_i\}_{i=1}^N$ and their associated independent variables $\{t_i\}_{i=1}^N$. On the Grassmannian, $\mathcal{G}(p, n)$, a data point (i.e., a subspace) $\mathcal{Y}_i$ is represented by a $n \times p$ orthonormal matrix $\mathbf{Y}_i$ such that $\mathcal{Y}_i = \text{span}(\mathbf{Y}_i)$. Note that, capital N denotes the number of measurements, while lowercase n refers to the number of rows of the

---

[2]The state equations for linear least-squares can equivalently be written as $\ddot{x} = 0$ (which corresponds to a straight line). This is of course a geodesic in Euclidean space.

[3]Forward TLM and backward TLAM equations arise from taking the second variation (cf. Eq. (6)) of the energy in Eq. (5) and integration by parts.

$n \times p$ matrix $\mathbf{Y}_i$ representing $\mathcal{Y}_i$. The independent variable t_i is a scalar value, normalized to $[0,1]$ to simplify notation. This allows ordering the $(t_i, \mathbf{Y}_i)$ pairs based on the ordering of the t_i's. Specifically, the optimization problem in this regression setting is

$$\min_{\{\mathbf{X}_i(0)\}} E(\{\mathbf{X}_i(0)\}) := \frac{1}{2\sigma^2} \sum_{i=1}^{N} d_g(\mathbf{Y}_i, \mathbf{X}_1(t_i))^2$$

$$\text{s.t.} \quad \left. \begin{array}{l} \dot{\mathbf{X}}_1 = \mathbf{X}_2 \\ \dot{\mathbf{X}}_2 = -\mathbf{X}_1(\mathbf{X}_2^\top \mathbf{X}_2) \end{array} \right\} \textit{forward equations}$$

$$\text{and} \quad \left. \begin{array}{l} \mathbf{X}_1(0)^\top \mathbf{X}_1(0) = \mathbf{I} \\ \mathbf{X}_1(0)^\top \mathbf{X}_2(0) = \mathbf{0} \end{array} \right\} \textit{initial conditions .} \quad (10)$$

Here, $d_g : \mathcal{G}(p,n) \times \mathcal{G}(p,n) \to \mathbb{R}_{\geq 0}$ denotes the *geodesic distance* on the Grassmannian [5, 12] (based on the canonical metric) and $\mathbf{X}_1(0)$ and $\mathbf{X}_2(0)$ are the initial conditions to be estimated. These initial conditions correspond to intercept and slope in linear regression. In this paper, we effectively use a uniform prior, but priors on the parameters (the initial conditions) can be added if desired. Just as in the Euclidean case of Sec. 2, the optimization problem can be solved using the adjoint method. The equations for the adjoint variables λ_1 and λ_2 are

$$\begin{aligned} \dot{\lambda}_1 &= \lambda_2 \mathbf{X}_2^\top \mathbf{X}_2, \quad \text{and} \\ \dot{\lambda}_2 &= -\lambda_1 + \mathbf{X}_2(\lambda_2^\top \mathbf{X}_1 + \mathbf{X}_1^\top \lambda_2) \end{aligned} \quad (11)$$

with $\lambda_1(1) = 0, \lambda_2(1) = 0$. Also, we have jumps for λ_1 at each data point ($i = 1, \ldots, N$)

$$\begin{aligned} \lambda_1(t_i-) &= \lambda_1(t_i+) - \frac{1}{2\sigma^2} \nabla_{\mathbf{X}_1(t_i)} d_g(\mathbf{X}_1(t_i), \mathbf{Y}_i)^2 \\ &= \lambda_1(t_i+) + \frac{1}{\sigma^2} \mathrm{Log}_{\mathbf{X}_1(t_i)} \mathbf{Y}_i , \end{aligned}$$
$$(12)$$

where we used (see [12]) that

$$\nabla_{\mathbf{X}_1(t_i)} d_g(\mathbf{X}_1(t_i), \mathbf{Y}_i)^2 = -2\mathrm{Log}_{\mathbf{X}_1(t_i)} \mathbf{Y}_i . \quad (13)$$

In more detail, this computes the Riemannian Log-Map[4] which allows us to *shoot* $\mathbf{X}_1(t_i)$ to $\mathbf{Y}_i$ in unit time along the geodesic connecting $\mathbf{X}_1(t_i)$ and $\mathbf{Y}_i$. The gradients of the energy from Eq. (10) w.r.t. $\mathbf{X}_1(0)$ and $\mathbf{X}_2(0)$ are given by

$$\begin{aligned} \nabla_{\mathbf{X}_1(0)} E = &-(\mathbf{I}_n - \mathbf{X}_1(0)\mathbf{X}_1(0)^\top)\lambda_1(0) \\ &+ \mathbf{X}_2(0)\lambda_2(0)^\top \mathbf{X}_1(0) \end{aligned} \quad (14)$$

and

$$\nabla_{\mathbf{X}_2(0)} E = -(\mathbf{I}_n - \mathbf{X}_1(0)\mathbf{X}_1(0)^\top)\lambda_2(0) . \quad (15)$$

Based on these gradients, the optimization problem can be solved numerically [12].

---

[4]See [1] for a detailed explanation of the Log-Map of the Grassmannian.

3.2 Hessian estimation

As discussed in Sec. 2, although the noise variance σ^2 is unknown before estimation, it is a constant that will not affect the optimal solution of the regression problem. However, the noise variance is important for Hessian estimation to obtain a proper covariance matrix through the Hessian inverse. In standard geodesic regression, measurement noise is assumed to be homoscedastic. In Eq. (10), this manifests in a simple sum-of-squared geodesic distances. However, this may not always be a realistic noise assumption. In particular, it is generally not well suited for *shape regression* which is our target example application (see Sec. 4). Hence, we generalize the squared geodesic distance to a weighted-squared geodesic distance, reminiscent of a Gaussian noise model with a more general covariance structure. We note that $d_g(\mathbf{Y}_i, \mathbf{X}_1(t_i))^2 = \|\mathrm{Log}_{\mathbf{X}_1(t_i)} \mathbf{Y}_i\|_F^2$, and the Riemannian Log-map on $\mathcal{G}(p,n)$ returns a $n \times p$ matrix. Hence, we reformulate the noise model in the tangent spaces of points on the Grassmannian and define a *weighted squared geodesic distance* as

$$d_{gw}(\mathbf{Y}_i, \mathbf{X}_1(t_i); \boldsymbol{\Sigma}_i)^2 := \|\boldsymbol{\Sigma}_i^{-\frac{1}{2}} \mathrm{vec}(\mathrm{Log}_{\mathbf{X}_1(t_i)} \mathbf{Y}_i)\|^2 . \quad (16)$$

Here, $\mathrm{vec}(\cdot)$ vectorizes a matrix and produces a column vector. Vectorization depends on the context and needs to be consistent with the structure of $\boldsymbol{\Sigma}_i$. In fact, this weighted squared geodesic distance is a generalization of the geodesic distance d_g on the Grassmannian; by setting $\boldsymbol{\Sigma}_i = \mathrm{diag}(\sigma^2, \cdots, \sigma^2)$, Eq. (16) reduces to the standard geodesic distance. The energy term for geodesic regression consequently changes to

$$E(\{\mathbf{X}_i(0)\}) := \frac{1}{2} \sum_{i=1}^{N} d_{gw}(\mathbf{Y}_i, \mathbf{X}_1(t_i); \boldsymbol{\Sigma}_i)^2 , \quad (17)$$

subject to the geodesic constraints and the constraints on the initial condition from Eq. (10). Notably, this modification only changes the regression formulation of Sec. 3.1 slightly: jumps for λ_1 change to $\lambda_1(t_i-) = \lambda_1(t_i+) + \mathrm{mat}(\Sigma_i^{-1} \mathrm{vec}(\mathrm{Log}_{\mathbf{X}_1(t_i)} \mathbf{Y}_i))$, where $\mathrm{mat}(\cdot)$ reshapes the matrix from its vectorized representation.

The newly introduced weighting requires us to estimate the $\boldsymbol{\Sigma}_i$'s for computing the weighted geodesic distance. With a sufficient amount of data, the full $\boldsymbol{\Sigma}_i$ can be estimated reliably. If only a few measurements are available, we need to assume a simpler structure of $\boldsymbol{\Sigma}_i$. For our shape regression problems, we will use a block-diagonal form, with sub-covariance matrices $\{\boldsymbol{\Sigma}_i^k\}_{k=1}^n$ of size $p \times p$, because of the limited number of data samples. Hence, for each entry of $\mathbf{X}_1$ we consider row-wise variances. For instance, in the shape analysis of Sec. 4, $\mathbf{X}_1$ is a $n \times p$ matrix with columns representing the coordinates of points (i.e., $p = 2, 3$ for 2D or

3D shapes, resp.) and n being the number of boundary points (of the shape). We assume the coordinates of each point[5] are correlated, which is estimated in its corresponding sub-covariance matrix $\boldsymbol{\Sigma}_i^k$. In this way, for each point of a shape, we have $p(p+1)/2$ unknowns in the symmetric matrix $\boldsymbol{\Sigma}_i^k$, which can be estimated from at least $p(p+1)/2$ samples. While, in principle, all the measurement time-points could share one matrix $\boldsymbol{\Sigma}$, this is not appropriate in curved spaces, as each estimated point $\mathbf{X}_1(t_i)$ has its own tangent space. Hence, we estimate *tangent-space specific* $\boldsymbol{\Sigma}_i$'s, i.e., at each time point t_i there is an associated covariance matrix $\boldsymbol{\Sigma}_i$.

In more detail, we first initialize all $\boldsymbol{\Sigma}_i$'s with identity matrices and then alternatingly (i) fit the geodesic and (ii) estimate the covariance matrices $\boldsymbol{\Sigma}_i$ of the measurement noise model. For each $\boldsymbol{\Sigma}_i$, we first compute the analogs of the residuals $\mathbf{V}_i^* = \mathrm{Log}_{\mathbf{X}_1(t_i)} \mathbf{Y}_i$ for each measurement $\mathbf{Y}_i$ and its corresponding point $\mathbf{X}_1(t_i)$ on the regression geodesic. We then parallel-transport each of the residual tangent vectors to all other measurement time points. Finally, $\boldsymbol{\Sigma}_i$ is estimated from the residual tangent vectors at this time-point and *all* the parallel-transported residual tangent vectors of the other measurement time points, i.e., $\{\mathbf{V}_i\}_{i=1}^N$ obtained from $\{\mathbf{V}_i^*\}_{i=1}^N$. For our block-diagonal case, each block-diagonal sub-covariance matrix $\{\boldsymbol{\Sigma}_i^k\}_{k=1}^n$ is computed as $\frac{1}{N} \sum_{i=1}^N (\mathbf{V}_i^k)^\top \mathbf{V}_i^k$, where $\mathbf{V}_i^k$ is the k-th row of the residual matrix $\mathbf{V}_i$.

Hessian-vector products. To compute the Hessian-vector-products, we follow Sec. 2.1 and take the second variation of the energy from Eq. (17) in the direction $\delta\mathbf{X}_0 = [\delta\mathbf{X}_{1,0}^\top, \delta\mathbf{X}_{2,0}^\top]^\top$. As a result, around the optimal solution, we can obtain the linearized TLM equations for shooting the perturbation $\delta\mathbf{X}_0$ *forward* as

$$\begin{aligned} \delta\dot{\mathbf{X}}_1 =\ & \delta\mathbf{X}_2 \\ \delta\dot{\mathbf{X}}_2 =\ & -\delta\mathbf{X}_1(\mathbf{X}_2^\top \mathbf{X}_2) - \mathbf{X}_1(\delta\mathbf{X}_2^\top \mathbf{X}_2) \\ & - \mathbf{X}_1(\mathbf{X}_2^\top \delta\mathbf{X}_2) \end{aligned} \quad (18)$$

with $\delta\mathbf{X}_1(0) = \delta\mathbf{X}_{1,0}$ and $\delta\mathbf{X}_2(0) = \delta\mathbf{X}_{2,0}$. The associated TLAM equations for shooting the adjoint variables $\delta\lambda_1$ and $\delta\lambda_2$ *backward* are

$$\begin{aligned} \delta\dot{\lambda}_1 =\ & \delta\lambda_2(\mathbf{X}_2^\top \mathbf{X}_2) + \lambda_2(\delta\mathbf{X}_2^\top \mathbf{X}_2) + \lambda_2(\mathbf{X}_2^\top \delta\mathbf{X}_2), \\ \delta\dot{\lambda}_2 =\ & \mathbf{X}_2(\delta\lambda_2^\top \mathbf{X}_1 + \lambda_2^\top \delta\mathbf{X}_1 + \delta\mathbf{X}_1^\top \lambda_2 + \mathbf{X}_1^\top \delta\lambda_2) \\ & + \delta\mathbf{X}_2(\lambda_2^\top \mathbf{X}_1 + \mathbf{X}_1^\top \lambda_2) - \delta\lambda_1, \end{aligned} \quad (19)$$

subject to $\delta\lambda_1(1) = \delta\lambda_2(1) = 0$. Again, we also have

jumps for $\delta\lambda_1$ at each data point, i.e.,

$$\delta\lambda_1(t_i-) = \delta\lambda_1(t_i+) - \frac{1}{2}\mathrm{mat}(\boldsymbol{\Sigma}^{-1}\mathrm{vec}(\mathbf{Z})) \quad (20)$$

with

$$\mathbf{Z} = (\nabla^2_{\mathbf{X}_1(t_i)} d_g(\mathbf{X}_1(t_i), \mathbf{Y}_i)^2)\delta\mathbf{X}_1(t_i) \ . \quad (21)$$

Computing $\mathbf{Z}$ is non-trivial as $\mathbf{Z}$ essentially is the product of the Hessian of the Grassmannian squared distance function and the direction $\delta\mathbf{X}_1(t_i)$. We numerically approximate this term by using the derivative of the gradient, i.e., $(\partial\nabla_{\mathbf{X}_1(t_i)+\epsilon\delta\mathbf{X}_1(t_i)} d_g(\mathbf{X}_1(t_i)+\epsilon\delta\mathbf{X}_1(t_i), \mathbf{Y}_i)^2/\partial\epsilon)|_{\epsilon=0}$, since we already know the solution to $\nabla_{\mathbf{X}_1(t_i)} d_g(\mathbf{X}_1(t_i), \mathbf{Y}_i)^2$ from Eq. (13). As an alternative to this numerical approximation, the Hessian of the squared geodesic distance could be computed analytically following the algorithm in [6].

Solving the forward TLM and backward TLAM equations for a given perturbation $\delta\mathbf{X}_0$ allows us to compute the Hessian-vector-product with respect to $\mathbf{X}_0 = [\mathbf{X}_1(0)^\top, \mathbf{X}_2(0)^\top]^\top$ as

$$\nabla^2 E \begin{pmatrix} \delta\mathbf{X}_1(0) \\ \delta\mathbf{X}_2(0) \end{pmatrix} = \begin{pmatrix} h_1 \\ h_2 \end{pmatrix} \quad (22)$$

with h_1, h_2 given by

$$\begin{aligned} h_1 =\ & \nabla^2_{\mathbf{X}_1\mathbf{X}_1} E\delta\mathbf{X}_1(0) + \nabla^2_{\mathbf{X}_1\mathbf{X}_2} E\delta\mathbf{X}_2(0) \\ =\ & -(\mathbf{I}_n - \mathbf{X}_1(0)\mathbf{X}_1(0)^\top)\delta\lambda_1(0) \\ & + \delta\mathbf{X}_1(0)\mathbf{X}_1(0)^\top\lambda_1(0) + \mathbf{X}_1(0)\delta\mathbf{X}_1(0)^\top\lambda_1(0) \\ & + \delta\mathbf{X}_2(0)\lambda_2(0)^\top\mathbf{X}_1(0) + \mathbf{X}_2(0)\delta\lambda_2(0)^\top\mathbf{X}_1(0) \\ & + \mathbf{X}_2(0)\lambda_2(0)^\top\delta\mathbf{X}_1(0) \end{aligned}$$

$$(23)$$

and

$$\begin{aligned} h_2 =\ & \nabla^2_{\mathbf{X}_2\mathbf{X}_1} E\delta\mathbf{X}_1(0) + \nabla^2_{\mathbf{X}_2\mathbf{X}_2} E\delta\mathbf{X}_2(0) \\ =\ & -(\mathbf{I}_n - \mathbf{X}_1(0)\mathbf{X}_1(0)^\top)\delta\lambda_2(0) \\ & + \delta\mathbf{X}_1(0)\mathbf{X}_1(0)^\top\lambda_2(0) \\ & + \mathbf{X}_1(0)\delta\mathbf{X}_1(0)^\top\lambda_2(0) \ . \end{aligned} \quad (24)$$

Note that the Hessian $\nabla^2 E$ is of size $2np \times 2np$, involving *all* the parameters in the initial conditions $\mathbf{X}_0$. At each iteration, by initializing the perturbation $\delta\mathbf{X}_0$ as $\mathrm{mat}([0,\cdots,0,1,0,\cdots,0]_{2np}^\top)$, we obtain the equivalent of one column of the desired Hessian $\nabla^2 E$.

3.3 Covariance Matrix Estimation

If the Hessian is symmetric and positive definite, its inverse (i.e., the covariance matrix estimate) is symmetric and positive definite. However, the Hessian on the

Grassmannian is semi-positive definite, because the dimensionality of $\mathcal{G}(p, n)$ is $p(n - p)$ instead of np [5]. Hence, among the np principal directions of the Hessian for $\mathbf{X}_1$, p^2 principal directions will have eigenvalues of zero. The same situation holds for $\mathbf{X}_2$. Consequently, the first $2p^2$ eigenvalues of the Hessian are zero. In practice, because of numerical errors, these eigenvalues may not be exactly zero but some small negative or positive values. We therefore set the first $2p^2$ smallest eigenvalues of the Hessian to zero. The Moore-Penrose pseudoinverse of the Hessian then gives the covariance matrix estimate for $\mathbf{X}_0$. Importantly, if we partition $\mathbf{C}(0)$ into 4 $np \times np$ blocks, the diagonal sub-matrices are the variances for $\mathbf{X}_1(0)$ and $\mathbf{X}_2(0)$ and the off-diagonal sub-matrices are the covariance between them.

3.3.1 Covariance Matrix Propagation

Now that we have the covariance matrix $\mathbf{C}(0)$ for parameters $\mathbf{X}_1(0)$ and $\mathbf{X}_2(0)$, we need to appropriately propagate $\mathbf{C}(0)$ along the regression geodesic. To achieve this goal, we rewrite the linearized forward equations, i.e., Eq. (18), around the optimal solution as $\delta\dot{\mathbf{X}} = \mathbf{A}(t)\delta\mathbf{X}$, where $\delta\mathbf{X} = [\text{vec}(\delta\mathbf{X}_1)^\top, \text{vec}(\delta\mathbf{X}_2)^\top]^\top$ is a column vector of size $2np$, resulting in

$$\mathbf{A}(t) = \begin{bmatrix} \mathbf{0}_{np} & \mathbf{I}_{np} \\ \mathbf{R} & \mathbf{S} \end{bmatrix} \qquad (25)$$

with

$$\begin{aligned} \mathbf{R} = &- \left(\mathbf{X}_2(t)^\top \mathbf{X}_2(t)\right) \otimes \mathbf{I}_n \\ \mathbf{S} = &- \left(\mathbf{X}_2(t)^\top \otimes \mathbf{X}_1(t)\right)\mathbf{T}_{np} \\ &- \mathbf{I}_p \otimes \left(\mathbf{X}_1(t)\mathbf{X}_2(t)^\top\right) \end{aligned} \qquad (26)$$

and $\mathbf{T}_{np}$ being an orthogonal permutation matrix of size $np \times np$ that satisfies $\text{vec}(\delta\mathbf{X}_2^\top) = \mathbf{T}_{np} \cdot \text{vec}(\delta\mathbf{X}_2)$. The time-dependent matrix $\mathbf{A}(t)$ is determined by the regression geodesic, i.e., at each time point t it is computed using the optimal $\mathbf{X}_1(t)$ and $\mathbf{X}_2(t)$. The covariance then propagates as $\dot{\mathbf{C}} = \mathbf{CA}^\top + \mathbf{AC}$. These are the same equations as for the covariance propagation in a continuous-time Kalman filter *without noise* [9].

4 Experiments

We demonstrate our approach for estimating confidence intervals for geodesic regression on $\mathcal{G}(p, n)$ on synthetic and real data[6]. In the experiments, we primarily consider *shape data*. Each shape is represented on the Grassmannian using singular value decomposition (SVD) on the coordinate matrix [12]. For example, for a 2D shape with n points its coordinate matrix

of size $n \times 2$ is $\mathbf{L} = [(x_1, y_1); (x_2, y_2); ...; (x_n, y_n)]$. For a 3D shape the coordinate matrix is $n \times 3$. By applying SVD on this matrix, i.e., $\mathbf{L} = \mathbf{U\Sigma V}^\top$, we obtain an affine-invariant shape representation (cf. [1]) using the left-singular vectors $\mathbf{U}$, which is a representative for an element (i.e., a subspace) on $\mathcal{G}(p, n)$.

For each independent value t_i, there is an estimated shape $\mathbf{X}_1$, its corresponding tangent vector, i.e., the velocity $\mathbf{X}_2$, and an estimated covariance matrix (propagated to t_i) for $\mathbf{X}_1$ and $\mathbf{X}_2$. While there is no straightforward way to jointly visualize the covariance matrix for both starting point *and* velocity, we can visualize them separately. E.g., given 2D shapes, the covariance matrix is $2np \times 2np$. We can easily extract the sub-matrix corresponding to the initial condition (or velocity) and then consider each 2×2 block along the diagonal. These blocks correspond to sub-covariances at each point on the shape (i.e., the marginals of a point). A confidence interval (CI) is not straightforwardly defined on manifolds; hence, we visualize the confidence interval for each point on the shape via its marginals. By fixing a 95% CI, we can visualize the sub-covariances using an error ellipse (or ellipsoid in 3D) and color each ellipse/ellipsoid by the corresponding matrix determinant.

Synthetic data. We generate two sets of synthetic data to validate our approach. The first one is a sequence of 11 "T-shapes" with the horizontal part gradually thinning. Each shape has an associated scalar value, i.e., the time point t which is uniformly sampled within [0, 1]. To add noise to these T-shapes, we shift the vertical part slightly to the left or to the right at each time point. The generated shapes and their corresponding shapes on the regression geodesic are shown in Fig. 1(a). In this experiment, the regression geodesic captures the thinning trend of the horizontal part of the T-shape. However, the added noise due to *shifting* the vertical parts adds to the local uncertainty, as shown by the CIs in Fig. 1(a).

The second synthetic data, shown in Fig. 1(b), also includes 11 shapes, all uniformly sampled from a geodesic that connects two shapes, a flower and a bird.
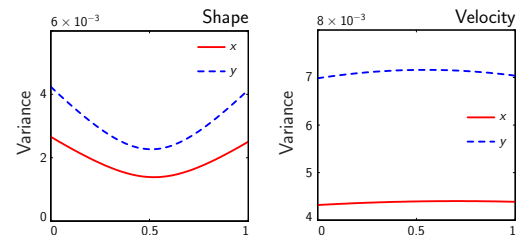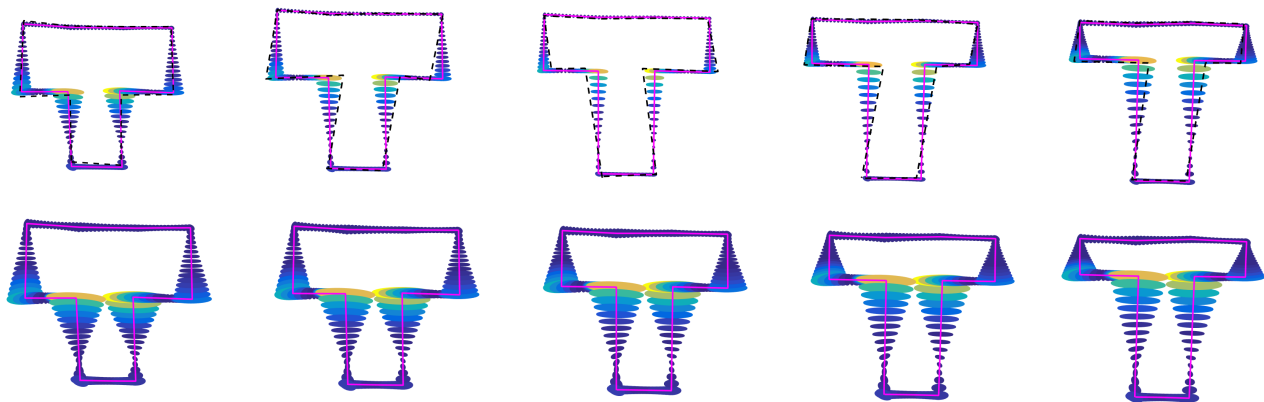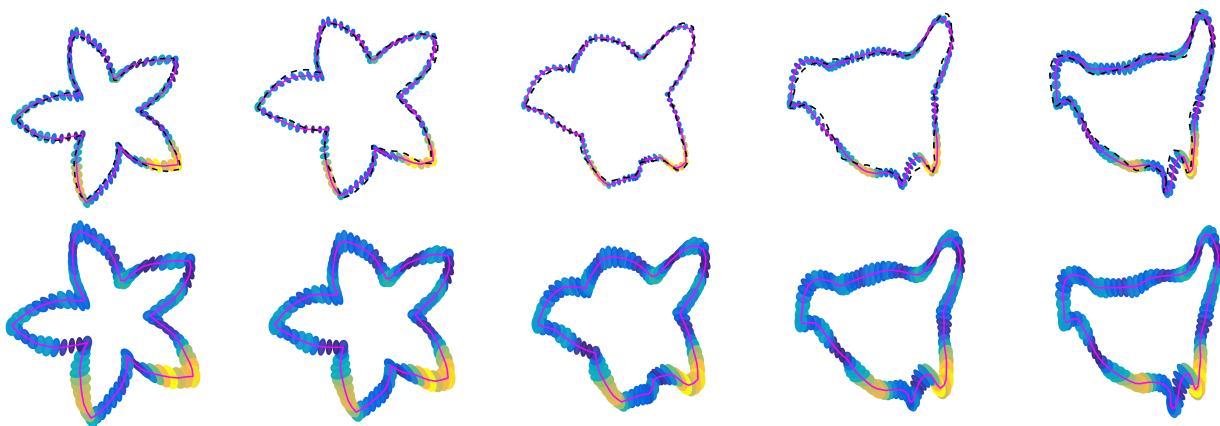

Figure 2: Trace of the variance at the 1st point (x, y) of the synthetic shape in the "flower" → "bird" experiment.

(a) Synthetic T-shapes

(b) Synthetic flower and bird shapes

Figure 1: Synthetic shapes with 95% CIs for the shape (*top* row in (a) and (b), resp.) and the velocity (*bottom* row in (a) and (b), resp.) at five time points (*left* to *right*). The black dashed curves are the generated shapes, and the magenta ones are the corresponding shapes on the geodesic. The color coding of the ellipses indicates the determinant of the sub-covariances, from small values (blue) to high values (yellow).

As before, each shape has an associated independent value t, uniformly distributed within $[0, 1]$. To "pull" the sampled shapes away from the geodesic, we use another set of 11 different shapes to build individual geodesics, along which we move our sampled shapes for time $t = 1/5$. The resulting shapes are *no longer* on one geodesic. Fig. 1(b) shows the local uncertainty introduced by moving the shapes away from the geodesic, visualized through the corresponding CIs.

Most notably, in both experiments, the CIs for the shapes first decrease and then increase when moving along the regression geodesic. Yet, the CI for the velocity remains almost unchanged. This can be more clearly seen from Fig. 2. In particular, we choose the first point of a shape and trace its variance in each coordinate (i.e., the marginals for x and y, resp.) along the geodesic. Around the mid-time point, the variance of the shape reaches its minimum as at this time point

we have the most measurements on both sides available for estimation. However, in case of the velocity CIs we do not have related measurement information. These observations are consistent with the CIs we observe for least-squares regression with points in $\mathbb{R}^2$ [17]. They also hold in the following experiments.

Real data. In our experiments with real data, we apply our method to study (1) *corpus callosum degeneration* and (2) *longitudinal caudate changes*. The corpus callosum shapes are 2D shapes collected from 32 subjects. Each shape is represented by 64 2D landmarks with an associated *subject age*, varying from 19 to 90 years. Fig. 3 (top) demonstrates the 95% CIs for estimated shapes at three ages. We only show one 95% CI for the velocity, because (as discussed before) the variance of the velocity remains almost unchanged along the regression geodesic. In this data set, the variances of corpus callosum shapes are quite small. Neverthe-
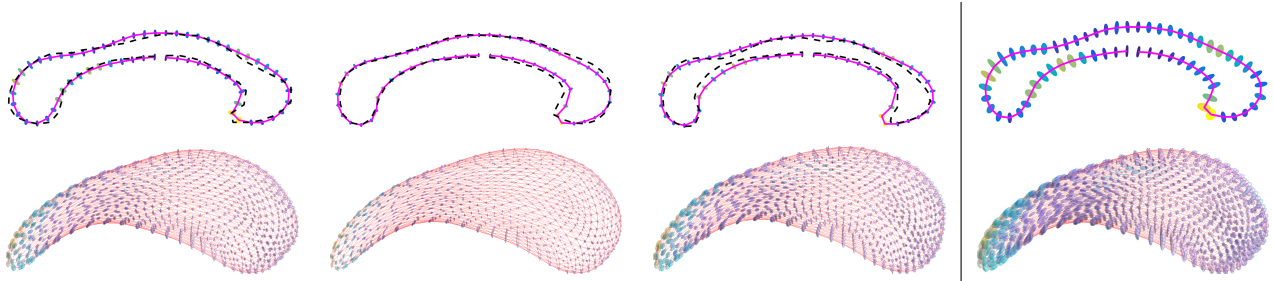
Figure 3: Real shapes with 95% CIs for the shape (first *three* columns) at three time points (*left* to *right*: 19, 55, 90 years for corpus callosum and 6, 16, 24 months for caudate) and 95% CIs for the velocity (*rightmost* column). For the corpus callosum shapes, the black dashed curves are real shapes, and the magenta ones are the corresponding shapes on the regression geodesic. For caudates, the magenta surface meshes are the estimated caudates on the regression geodesic. As in Fig. 1, the color coding of the ellipsoids indicates the determinant of the sub-covariances, from small values (blue) to high values (yellow).

less, we can still see the changes in the variance of the shape along the regression geodesic. Furthermore, from the CIs for the shape and the velocity, we can see that the anterior and posterior ends of the corpus callosum exhibit larger uncertainty than the mid-caudate. This is consistent with previous studies.

The caudate data set is longitudinal, including 10 subjects with data collected at three time points, i.e., 6, 12, and 24 months. Each shape is represented by 1002 3D landmarks. In our experiment, we only use the left caudate. Fig. 3 (bottom) shows the 95% CIs for the shape and the velocity. As expected, the CIs for the shape at mid-age show highest confidence; again, the CIs for the velocity remains almost unchanged along the regression geodesic. Notably, the "tail" of the caudate exhibits higher variance, which is consistent with the data as the caudate tail is difficult to segment.

5   Discussion

We presented an approach to estimate confidence intervals for geodesic regression on the Grassmannian. In principle, this concept can be extended to other Riemannian manifolds by adjusting the appropriate terms (i.e., Log-Map, parallel transport, etc.). Our experimental results (1) show that estimated confidence intervals are consistent with the behavior of linear least-squares regression and (2) that we can provide *local uncertainties* of the regression results over time. Additionally, we have introduced block-diagonal matrices to approximate the general measurement covariance matrices to give different weights to the shape coordinates. In general, the full measurement noise model could be estimated, given a sufficiently large number of measurements. To ensure the numerical correctness of the covariance estimate of the regression parameters, we compared our results with an approximate analytical solution obtained from finite differences of the gra-

dient of the energy (and then taking the inverse). For the synthetic data, with a small perturbation at the optimal solution of the regression, e.g., 1e-6, the finite difference approximation results in a covariance close to ours, i.e., 1e-9 difference in the Frobenius-norm.

In addition, the covariance matrix estimation, i.e., computing the Hessian inverse, would be computationally demanding for a large and high-dimensional dataset. However, approximate solutions can be used in this case, e.g., by using a low-rank approximation of the Hessian [22].

Finally, we note that MCMC sampling could, in principle, be used to estimate model uncertainty. However, sampling manifold-valued parameters is nontrivial [10], especially for our case, where parameters reside in the tangent bundle of the Grassmannian. To compare our method with MCMC sampling, one possible strategy could be exploring Hybrid Monte Carlo sampling strategies on matrix manifolds [11] (or [2]) and further extending these strategies to the tangent bundle. This comparison, however, is not straightforward and is left for future work.

Acknowledgments

References

[1] E. Begelfor and W. Werman. Affine invariance revisited. In *CVPR*, 2006.

[2] M.A. Brubaker, M. Salzmann, and R. Urtasun. A family of MCMC methods on implicitly defined manifolds. In *AISTATS*, 2012.

[3] R. Calandra, J. Peters, C.E. Rasmussen, and M.P. Deisenroth. Manifold Gaussian process regression. In *IJCNN*, 2016.

[4] E.R. Cohen. An introduction to error analysis: The study of uncertainties in physical measurements, 1998.

[5] A. Edelman, T. Arias, and S.T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.*, 20(2):303–353, 1998.

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

[7] H. P. Flath, L. C. Wilcox, V. Akçelik, J. Hill, B. van Bloemen Waanders, and O. Ghattas. Fast algorithms for Bayesian uncertainty quantification in large-scale linear inverse problems based on low-rank partial Hessian approximations. *SIAM J. Sci. Comput.*, 33(1):407–432, 2011.

[8] T.P. Fletcher. Geodesic regression and the theory of least squares on Riemannian manifolds. *Int. J. Comput. Vision*, 105(2):171–185, 2013.

[9] A. Gelb. *Applied optimal estimation.* MIT press, 1974.

[10] M. Girolami, B. Calderhead, and S.A. Siu. Riemannian manifold Hamiltonian Monte Carlo. *arXiv preprint arXiv:0907.1100 [stat.CO]*, 2009.

[11] A. Holbrook, A. Vandenberg-Rodes, and B. Shahbaba. Bayesian inference on matrix manifolds for linear dimensionality reduction. *arXiv preprint arXiv:1606.04478 [stat.CO]*, 2016.

[12] Y. Hong, R. Kwitt, N. Singh, N. Vasconcelos, and M. Niethammer. Parametric regression on the Grassmannian. *IEEE TPAMI*, 2016.

[13] A.G. Kalmikov and P. Heimbach. A Hessian-based method for uncertainty quantification in global ocean state estimation. *SIAM J. Sci. Comput.*, 36(5):267–295, 2014.

[14] M. Niethammer, Y. Huang, and F.-X. Vialard. Geodesic regression for image time-series. In *MICCAI*, 2011.

[15] J. Nocedal and S. Wright. *Numerical optimization.* Springer Science & Business Media, 2006.

[16] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning.* The MIT Press, 2005.

[17] J.O. Rawlings, S.G. Pantula, and D.A. Dickey. *Applied regression analysis: a research tool.* Springer Science & Business Media, 1998.

[18] Q. Rentmeesters. A gradient method for geodesic data fitting on some symmetric Riemannian manifolds. In *CDC-ECC*, 2011.

[19] H. Sagan. *Introduction to the Calculus of Variations.* Courier Corporation, 2012.

[20] J.L. Troutman. *Variational calculus and optimal control: optimization with elementary convexity.* Springer Science & Business Media, 2012.

[21] R. E. Weiss. *Modeling Longitudinal Data.* Springer Science & Business Media, 2005.

[22] X. Yang and M. Niethammer. Uncertainty quantification for LDDMM using a low-rank Hessian approximation. In *MICCAI*, 2015.