STATISTICAL LEARNING MODELS FOR MANIFOLD-VALUED MEASUREMENTS WITH APPLICATIONS TO COMPUTER VISION AND NEUROIMAGING

by

Hyunwoo J. Kim

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

(Computer Sciences)

at the

UNIVERSITY OF WISCONSIN–MADISON

2017

Date of final oral examination: 5/04/2017

The dissertation is approved by the following members of the Final Oral Committee:

Vikas Singh, Associate Professor, Department of Biostatistics
Sterling C. Johnson, Professor, Department of Medicine-Geriatrics
Baba C. Vemuri, Professor, Dept. of CISE, University of Florida
Mohit Gupta, Assistant Professor, Department of Computer Sciences
Charles R. Dyer, Professor, Department of Computer Sciences
Xiaojin Zhu, Professor, Department of Computer Sciences

To my parents and lovely wife.

ACKNOWLEDGMENTS

My Ph.D. program including this thesis was a long journey. Like most long journeys, this journey was a combination of lots of help and support from many people. I use this space to thank them. During writing acknowledgments, I realized that words are powerless to express all my gratitude. It may not be enough but I want to share how lucky I was to have a great privilege and an honor to work on something that I am passionate about with great mentors, collaborators, and colleagues. Especially, my path was not linear and sometimes I felt like I was exploring too much and even lost. Thanks to my visionary mentors I could turn my detours into my asset and build my own research theme.

When I came to UW-Madison, I found that this is a great place to study numerical optimization that is a fundamental theory to estimate learning models and find solutions. For the first two years, I totally focused on numerical optimization. I really enjoyed it but I was still craving data science. So I was looking for the balance between theoretical research and practical data science. At that moment, I met my advisor Vikas Singh. He gave me a new topic 'manifold statistics', which is amazingly interesting. I thank him for giving me such an opportunity. Indeed, when we started working on this topic, I was doing my internship at Amazon, in Seattle. I spent the whole summer of 2013 reading relevant materials. Every morning and evening of that summer, I read relevant papers and books. That was the start of this thesis. Further, during my Ph.D. program, Vikas put tremendous time and efforts in me. Because of this, how I could finish my Ph.D. in a timely manner even though I joined his group quite late. I've learned a lot from him such as how to write a paper, do a presentation, communicate with collaborators from other fields, and so on. I used to start writing a paper without consulting other papers in terms of style and jargon. He taught me how to read papers in order to write a paper. It

improved my writing significantly. I believe that without his support my Ph.D. program would not be as successful and productive as it is now.

I also thank my committee members of my final defense and preliminary exam. Baba C. Vemuri has been helping me and filling me in with great ideas. Further, he is my mentor in terms of differential geometry for data science. From my second paper at UW-Madison to my most recent work, he has contributed to most of my projects. We often spent time together at vision conferences. Especially, at the ICCV 2015, he chose great restaurants and great local dishes. They were amazing and I thank him for not letting me go wrong restaurants. Sterling C. Johnson helped me to start research in neuroimaging. I've learned a lot from meetings with him. He is very kind and sharp. He helped me to understand the WRAP study and data at ADRC, interpret the results and identify important questions from the neuroscience perspective. From the imaging group meeting with Barbara B. Bendlin and Ozioma Okonkwo, I got valuable feedback in a neuroimaging perspective. Charles R. Dyer gave me great support and I was lucky to do a project with him. We developed Abundant Inverse Regression (AIR) and he gave me ideas on its potential applications such as air pollution monitoring. Mohit Gupta is a great mentor and reinitiated Computer Vision Reading Group (CVRG). I enjoyed the talks there on diverse topics in vision and I also got valuable feedback on my job talk presentations from him. I believe that the way he interacts with his students and inspires them is a great example for me to pursue my career later on. The first professor that I met at UW-Madison was Xiaojin (Jerry) Zhu. He is gifted at discussion. Whenever he joins a seminar, he always points out critical issues and gives good suggestions including my final oral defense. His course Advanced Machine Learning was amazing and I enjoyed all his classes. I've learned a lot from him how to facilitate discussion and make students participate in the class. Also, this class offered me a good starting point to do my manifold-valued Dirichlet Process project. I took

"Statistical Methods for Medical Image Analysis" taught by Moo C. Chung. Through the course and collaboration with him, I had great opportunities to discuss with him and check whether frameworks are statistically valid. Also, I learned how to analyze functions defined on graphs. Jude W. Shavlik served my prelim committee members gave me valuable comments. Further through GAC (Graduate Advising Committee) meeting, he encouraged me when I was having a hard time with course works at the beginning of my Ph.D. program. My Ph.D. minor is statistics. Thanks to that, I took great statistics courses from amazing faculty. Michael A. Newton covered foundations for computational statistics and through multiple discussions with him, I could double-check my frameworks. I also appreciate Grace Wahba. She came to my preliminary exam with her students. We had valuable discussions later on. I learned a lot from her class including kernel machines and 'R'.

All of my projects included in the thesis are not possible without the help from my lab mates. First of all, Nagesh Adluru has been a great friend. He helped my very first project. Since then, we are working together closely. I realized that so far we wrote seven papers together. He gave me a lot of feedback as a potential customer of my machinery. We were a great team with complementary skill sets. Beyond research, he gave me rides to my home uncountably many times. Brandon M Smith is a really sweet friend of mine. We from time to time grabbed a coffee together. We went to computer vision conferences together and we explored random pubs and tried local beers. When we wrote ECCV '16 paper, he saved our project. I was about to give up the submission but he pushed it hard as a co-leader of that project and we could finish it timely. I studied with Sathya Ravi in many optimization classes and we joined the Vikas' lab at a similar time. He always amused me with fancy math and great papers. He introduced me great blogs of top scholars. Maxwell Collins had his desk right next to mine. He is very smart and very kind.

He is extremely calm and peaceful. In some sense, he has a quite rare combination of good traits. Whenever I asked questions, he was willing to give his time and find the answer together. He has proficiency in math, programming, and Linux. I was so lucky to have such a lab mate. If I become a professor, I hope to have such a versatile and mature student in my future lab. My savvy and passionate lab mate, Jia Xu, always knows what he wants and think through how to achieve it. I enjoyed working with him. I appreciated it that he came to my wedding from China with a warm wedding card. Won Hwa Kim was very persistent and productive. It was a good example for Vikas students. I really enjoyed the project with him about graphical model estimations. It's not published yet but I also worked on graphical models with Ronak. Whenever I had a question on the American culture, he gave me a clear answer. Seong Jae Hwang organized our tennis sessions with lab members. I and Yunyang Xiong did a project on gaze estimation. We conducted experiments with a variety of data and deep learning frameworks. I also thank other lab mates Vamsi K. Ithapu, Deepti Pachauri, Mona Jalal, and Ligang Zheng.

Indeed, my research interests are quite wide. Thanks to my remote collaborators, I could work on independent projects from my main research theme. Kyogu Lee at SNU in Korea is my former supervisor at Music and Audio Research Group. Thanks to the collaboration with him and Hoon Heo, I finally published at the best music application venue, ISMIR. I met him once a year when I visit Korea and every time he gave me many good pieces of advice on how to pursue my Ph.D. program in the US and he shared his story. I appreciate his mental/financial support. I and Jinseok Nam at TU Darmstadt in Germany worked on novel deep learning frameworks for natural language processing. I thank him for our fun projects.

When I started my Ph.D. at UW-Madison, I was really into music applications and machine learning such as music search engine and cover song

detection. At this school, it was hard to find someone who is interested in such a topic. I discussed with Jerry Zhu (Xiojin Zhu) and he suggested me to take numerical optimization courses. I knew that kernel machines and many machine learning algorithms use numerical optimization theories to estimate models. Luckily, Benjamin Recht let me study with him and we worked on stochastic gradient algorithms for large-scale machine learning. It was my first time to implement a numerical optimization algorithm from scratch and apply it to my formulation. I've learned how to work efficiently and what is good research. Our algorithm improved the performance of cover song search algorithms. After that, I took literally all of the numerical optimization courses at UW-Madison. These courses gave me a foundation for numerical optimization and mathematical reasoning. I enjoyed classes taught by all optimization faculty including Jeffrey T. Linderoth and Michael C. Ferris. I did a small project with Steve J. Wright and learned recent works about optimization for machine learning from him. I love Integer Programming by James Luedtke and we worked on an interesting relaxation for my summer research project. I could learn a foundation of combinatorial optimization from him, which is useful in many machine learning and vision problems. Thanks to Convex Analysis taught by Stephen M. Robinson, I gained a deeper understanding of convexity. He inspired me and made me dream about being a great and active senior scholar.

I thank my optimization study group including Namsuk Cho, Yu Sun, Merve Bodur, Gizem Cavuslar, Youngdae Kim, and Andy Chang. I could learn numerical optimization much faster with them. It was really great memories to work on the same problems and help each other.

Outside of my academic sphere, I want to thank few more people. I thank Elizabeth D. Butler. She spent a great amount time with Hojin. Especially, I could not go to any Halloween party for six years of my Ph.D. program. She brought my wife Hojin to State Street Halloween Party! I

really thank her to be a good friend of Hojin. I thank Lisa Marvel Johnson for helping me to outline some chapters in my thesis and proofreading.

Lastly, I want to thank my family for unconditioned and constant support. Without my parents' support, I could not be able to even start this journey. Hojin, my lovely wife, sacrificed her entire career and came to the US solely to be with me. She was supportive and understand the expectation of Ph.D. program. She believes in me and whenever I was frustrated, she raised me up. Also, I appreciate her for eating my amateurish food on Friday. Without her patience, we could not have our Friday tradition.

The projects in the thesis are supported in part by AG040396, AG021155, BRAIN Initiative R01-EB022883, UW ICTR 1UL1RR025011, UW ADRC AG033514, Waisman IDDRC U54-HD090256, UW CPCP AI117924, and NSF grants. Also, joint work with Dr. Vemuri included in this thesis was partly funded by the NSF grant IIS 1525431 to Dr. Vemuri.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

In modern data analysis, we frequently need to analyze objects such as directional data, special types of matrices, probability distributions, and so on. Such *structured data* are becoming increasingly common in various disciplines. It turns out that many of these data lie on manifolds, which are a natural generalization of Euclidean spaces. The geometry of such a data space (and resulting model space) is crucial to develop more accurate and effective learning models especially when the data space does not exhibit Euclidean geometry. The key focus of this dissertation is to develop statistical machine learning algorithms for the structured data motivated by applications in vision and neuroimaging. The thesis is motivated by some distinct demands of structured data analysis applications covering several scientific domains:

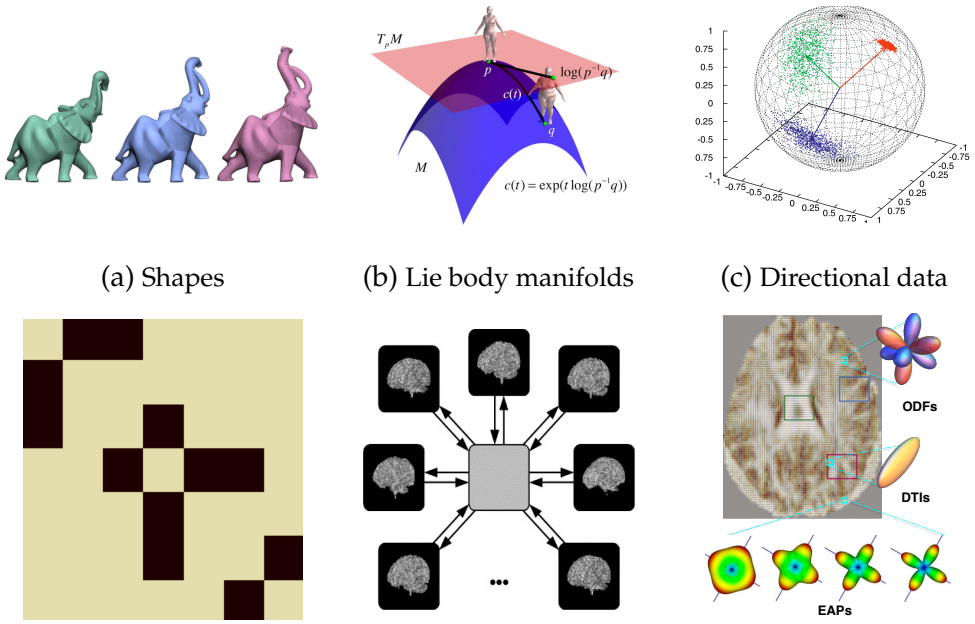
1. How can we model "structured" data in a way that respects the underlying geometry of the data spaces?
2. How can we estimate such models with structured parameters efficiently without leaving the structured data/model spaces?
3. How can we improve statistical power of statistical machine learning models in cross-sectional and longitudinal analysis that involve structured data spaces?

Using geometrical reasoning, this thesis provides effective statistical learning models for structured data in the context of interpolation, dimensionality reduction and parametric/nonparametric regression for cross-sectional and longitudinal analysis and demonstrates their effectiveness on a broad range of problems motivated from neuroimaging.

1 INTRODUCTION

In modern data analysis, we frequently must operate on objects such as graphs, trees, special types of matrices, probability distributions, the unit sphere and so on. Such "structured data" are becoming increasingly common in various disciplines including physics, psychology, health and social sciences. For example, directional data is common in applications that analyze measurements from antennas (Mammasis and Stewart, 2010), whereas time series data (i.e., curves) are widely used in finance (Tsay, 2005) and health sciences (Dominici et al., 2002). Surface normal vectors on the unit sphere (for computer vision or graphics) (Straub et al., 2015), probability density functions (in functional data analysis) (Srivastava et al., 2007), covariance matrices (for use in conditional independences, image texture descriptors) (Tuzel et al., 2006a), rigid motions (registration) (Park and Ravani, 1995), shape representations (shape space analysis) (Kendall, 1984), tree-based data (parse tree in natural language processing) (Quirk et al., 2005), subspaces (videos, segmentation) (Xu et al., 2013; Elhamifar and Vidal, 2009), low-rank matrices (Candes and Recht, 2012; Vandereycken, 2013), and kernel matrices (Schölkopf and Smola, 2002) are structured data, see Fig. 1.1. In neuroimaging, a brain image has a structured measurement at each voxel to describe water diffusion (Basser et al., 1994; Leow et al., 2009; Özarslan and Mareci, 2003; Aganj et al., 2009; Cheng, 2012) or local structural change (Hua et al., 2008; Zacur et al., 2014).

It turns out that the data in many of these example problems do not exhibit Euclidean geometry. In other words, the data spaces are curved and the standard arithmetic operations (e.g., addition, subtraction, and multiplication) may not be available. For example, directional data are on the unit sphere and the addition of two unit vectors is not on the unit sphere anymore. The addition cannot be defined in the same way



(a) Shapes (b) Lie body manifolds (c) Directional data

(d) Covariance matrix (e) Brain atlas (f) Diffusion Measurements

Figure 1.1: Examples of structured data. (a) A shape (a set of landmarks) in the Kendall's shape space is invariant to translation, scale, and rotation (Kilian et al., 2007; Kendall, 1984) (b) Three-dimensional (human body) shapes are studied in the context of Riemannian structures (Freifeld, 2013) (c) Directional data can be viewed as a point on the unit sphere S^n(Hamelryck et al., 2006) (d) Covariance matrices can be be viewed as a point on a SPD(n) manifold (Kim et al., 2016c) (e) Estimation of atlases in the large deformation diffeomorphic setting (Joshi et al., 2004) and (f) Diffusion process of water molecules in brain images is represented by structured measurements such as diffusion tensor (DTI), orientation density function (ODF), and ensemble average propagator (EAP) (Goh et al., 2011).

without leaving the space of interest. Many traditional machine learning and statistical models are defined with these operations assuming that data lie on a vector space. Therefore, even basic/fundamental statistical models (e.g., linear models defined by additions as $y = ax + b$) may not be directly applicable to data in a non-Euclidean space.

Forcibly enforcing Euclidean structure on such data is problematic. This may yield poor goodness of fit and/or weak statistical power. One

reason is that Euclidean distance is often less accurate distance metric for structured data. In the example of directional data, Euclidean models measure the prediction errors using the Euclidean distance between two unit vectors whereas the quantity of interest is the distance measured along the surface of the unit sphere. Further, Euclidean models may not yield valid predictions (or syntheses). It is easy to see that predictions from Euclidean linear models are not on the unit sphere. So, artificial post processing is required to get a valid prediction. This leads to a gap between the desired predictions and predictions attainable by the chosen learning model.

Driven by these motivations, there is a rapidly developing body of theoretical and applied work which generalizes classical tools from Euclidean spaces to manifolds. To this end, this thesis provides new statistical machine learning algorithms for manifold-valued data which commonly occur in various problems in computer vision and neuroimaging. We discuss shortly why the geometry of data space is a key ingredient to develop more effective statistical learning models for structured data and generate valid predictions in the structured data space without additional preprocessing and post processing.

1.1 Why data spaces matter for inference?

An increasingly large number of problems in data analysis today rely on the use of statistical inference methods to drive one or more stages of the overall workflow. Consider a statistical inference algorithm $\mathcal{L}$ within an image analysis application, it seeks an appropriate model $\hat{\theta}$ from a model space Θ based on observations e.g., images or pixel measurements, $\{x_i\}_{i=1}^N$ that live in a data space $\mathcal{X}$, i.e., $\hat{\theta} = \mathcal{L}(\Theta, \{x_i\}_{i=1}^N)$. To do so, the algorithm invariably makes some explicit or implicit assumptions on the geometry of models space Θ such as smoothness, convexity, sparsity and so on.

In some sense, such an inductive bias is crucial to achieve efficiency and computational tractability of the estimation procedures. A similar motivation applies to the other input to the learning algorithm, i.e., the data space. Here, however, we typically transform the data $\{x_i\}_{i=1}^N$ to make them "easier" to work with — this yields efficiency and often significantly improves the performance/accuracy of the image analysis method. These transformations may include the Fourier transform, nonlinear embedding, dimensionality reduction using subspaces and feature extraction methods. Interestingly, in some specific situations, we find that the data space $\mathcal{X}$ is known a priori to have a nice mathematical structure with well-studied properties. It makes sense that if algorithms were to make use of this additional information, even more efficient inference procedures can be developed. Motivated by this intuition, the analogous expression for the inference task may be expressed as

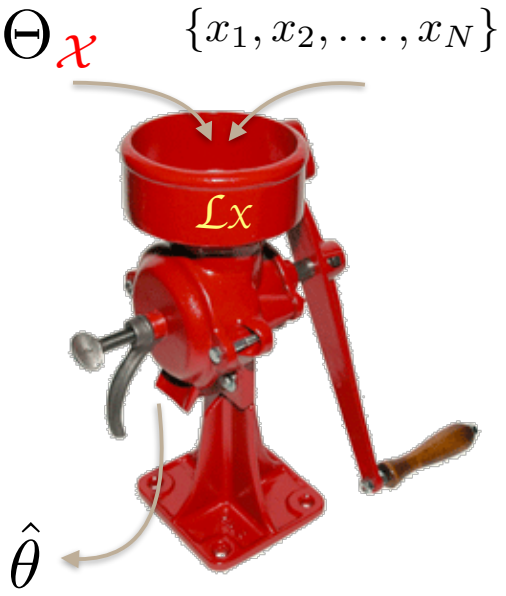


Figure 1.2: Geometrically inspired learning models

$$\hat{\theta} = \mathcal{L}_{\mathcal{X}}(\Theta_{\mathcal{X}}, \{x_i\}_{i=1}^N) \tag{1.1}$$

where $\mathcal{L}_{\mathcal{X}}$ refers an inference algorithm which can adapt based on a well characterized (possibly, analytical) structure of the data space $\mathcal{X}$.

Note that not only is the inference algorithm exploiting the structure of $\mathcal{X}$ but also $\Theta_{\mathcal{X}}$ corresponds to the model space determined by the geometry of the data space $\mathcal{X}$. This strategy is particularly appropriate

when data $\{x_i\}_{i=1}^N$ lie on an analytic manifold. Further, this framework allows systematically generalizing classical image analysis methods to manifolds using geometrical reasoning and defining an even abstract model for a group of different data spaces. For example, one may define an abstract model $\mathcal{L}_{\mathcal{X}}$ minimizing the squared geodesic errors on a *class* of spaces (e.g., symmetric space) with some known properties (e.g., the existence of the minimizing geodesic and closed form solutions to Jacobi field equations) and develop its inference algorithm with suitable operations (e.g. exponential maps, logarithm maps, interpolation etc.) determined by a concrete space $\mathcal{X}$ (e.g., Euclidean space, sphere, Grassmannian, and Stiefel manifolds). We point out that utilizing specific knowledge of the data space (e.g., manifold structure) to inform the choice of objection function and/or hypothesis space is not novel to our research and goes back at least several decades in computer vision, machine learning, and optimization (Karcher, 1977; Chikuse, 2003; Pennec, 2006; Grenander and Szegö, 2001; Mumford, 1994; Smith, 1994; Absil et al., 2009; Mardia and Jupp, 1999). For instance, shape spaces have been heavily used in medical imaging and Grassmannian have been studied for video analysis and machine learning (Turaga et al., 2011; Hamm and Lee, 2008). Nonetheless, there are many standard statistical formulations that are unavailable for specific manifolds arising frequently in practice, and central in a variety of image analysis tasks in applications. Enabling significantly improved accuracy in these image analysis applications is the key motivation of this dissertation.

1.2 Structured data

Data spaces in various recent scientific disciplines routinely correspond to non-Euclidean spaces, while classical models commonly assume that data live in Euclidean spaces. For example, in computer vision, one uses

region covariance descriptors for texture analysis (Tuzel et al., 2006b, 2008), rigid motions (including reflections, rotations, and translations) (Park and Ravani, 1995) and surface normal vectors on a unit sphere (Straub et al., 2015). In machine learning, we deal with subspaces, low-rank matrices (Boumal and Absil, 2011; Vandereycken, 2013), kernel matrices (Jayasumana et al., 2013; Feragen et al., 2015), normalized feature vectors with cosine similarity, probability density functions (PDFs) (Srivastava et al., 2007), and probability mass functions (PMFs) such as Dirichlet distribution and multinomial distribution (Lebanon et al., 2005), and so on. It turns out that many of these examples lie on manifolds which are a natural generalization of Euclidean spaces. Even when performing basic analysis on such datasets, in general, we cannot apply vector-space operations directly.

Beyond the applications in computer vision and machine learning described above, neuroimaging studies routinely acquire manifold-valued data that are becoming increasingly important across a spectrum of ongoing research studies. For example, diffusion tensor magnetic resonance images (DTI) (Basser et al., 1994; Le Bihan et al., 2001) allow one to infer the "diffusion tensor" characterizing the anisotropy of water diffusion at each voxel in an image volume. This tensorial feature can be visualized as an ellipsoid and represented by a 3×3 symmetric positive definite (SPD) matrix at each voxel in the acquired image volume (Lenglet et al., 2006). Neither the individual SPD matrices nor the field of these SPD matrices lie in a vector space but instead are elements of a negatively curved manifold where standard vector space operations are not valid. Classical Euclidean models are not applicable in this setting. Separate from this application, for T1-weighted Magnetic resonance images (MRIs) that are commonly used in brain imaging studies, we are frequently interested in analyzing not just the 3D intensity image on its own, but rather a quantity that captures the deformation field between each image and a *population*

template obtained via spatial registration methods. A registration between the image and the template yields the deformation field required to align the specific image with respect to a template. Quantities such as the Cauchy deformation tensor (CDT) defined as $\sqrt{J^T J}$ have been reported in literature for use in morphometric analysis (Hua et al., 2008). The input to the statistical analysis is a 3D image of voxels, where each voxel corresponds to a CDT matrix. Another important example is the diffusion weighted images: here, a manifold-valued field is derived from high angular resolution diffusion images (HARDI) (Tuch et al., 1999; Frank, 2002). These measurements can be used to compute the ensemble average propagator (EAP) at each voxel of the given HARDI data. The EAP is a probability density function that is related to the diffusion sensitized MR signal via the Fourier transform (Callaghan, 1991). Since the EAP is a probability density function (by using a square root parameterization of this density function), it is possible to identify it with a point on the unit Hilbert Sphere. Once again, to perform any statistical analysis of these data derived features, it is inappropriate to apply standard vector-space operations since the unit Hilbert sphere is a positively curved manifold.

1.3 Riemannian Manifolds for generalizing Euclidean models

Fortunately, many manifold-valued data happen to be in one nice subclass of manifolds, so called Riemannian manifolds, which are smooth manifolds (allowing calculus) equipped with a smoothly varying local metric (allowing distance). Riemannian geometry offers elegant tools to build models for manifold-valued data and generalize Euclidean models to nonlinear spaces. For example, a Riemannian metric on a manifold $\mathcal{M}$, an inner product $\langle \cdot, \cdot \rangle_x$, $x \in \mathcal{M}$ in the tangent space at each point, enables defining notions of distance, surface area, angle and curvature on

manifolds. To be specific, the length of a smooth curve on the manifold $\mathcal{M}$ parameterized by an interval $[a, b]$, $\gamma : [a, b] \to \mathcal{M}$ can be defined as

$$L(\gamma) = \int_a^b \sqrt{\langle \dot{\gamma}(t), \dot{\gamma}(t) \rangle_{\gamma(t)}} dt.$$

Then, the distance between two points $x, y \in \mathcal{M}$ is defined by the infimum of the length of all smooth curves between two points given as

$$d(x, y) = \inf_{\gamma \in \Gamma} \ell(\gamma), \tag{1.2}$$

where Γ is a set of all smooth curves between two points. Then, the notion of distance is extendable to nonlinear spaces. The so-called *geodesic distance* is known to be more accurate than Euclidean distance in the literature. This naturally enables more sensitive group difference analysis for manifold-valued data via statistical tests such as Cramer's method (Cramér, 1928; Baringhaus and Franz, 2004; Zacur et al., 2014).

Further, within a Riemannian framework, a quotient space with an equivalence relation and a metric with some invariances are useful to define a specialized space for specific problems. For example, in shape analysis, a common approach is to use a *shape manifold* which is a Riemannian manifold (Klassen et al., 2004). It is the quotient space of a finite number of landmarks by the equivalence classes defined by shape invariant transformations: rigid rotations, translations, and scaling. In other words, the same shapes transformed by rigid motions are mapped to the same point on the shape manifold. Also difference of shapes can be measured via a Riemannian metric (e.g., the Procrustean metric) on the shape manifold. Hence, we can draw a considerable body of research from differential geometry to develop more powerful and accurate models. Using (or inspired by) tools from differential geometry, a variety of statistical data analysis have been studied in literature. We now briefly

introduce a body of work describing ideas related to this thesis.

1.4 Examples of structured data analysis

This dissertation addresses structured data analysis motivated by applications in computer vision and neuroimaging. We study new generalizations/extensions of standard statistical learning models from the following perspectives: interpolation (of structured probability density functions), regression that involves structured response variables (i.e., $f : \mathbf{R}^d \to \mathcal{M}$) including both parametric and nonparametric models, dimensionality reduction on Riemannian manifolds, and longitudinal analysis for structured measurements or structural changes of objects (e.g., morphometric changes of brains). We will provide an overview of each topic one by one with related works in this section.

1.4.1 Interpolation of structured data

Interpolation is a fundamental operation in a variety of statistical inference procedure has been studied for structured data as well. On Riemannian manifolds, the center of mass was studied by (Karcher, 1977) and the main idea are widely used directly/indirectly in a variety of applications. Let $\{y_1, \ldots, y_N\}$ be structured measurements. The interpolation is the same as finding a minimizer to the following problem in a structured data space $\mathcal{M}$.

$$y = \arg\min_{y \in \mathcal{M}} \sum_{i=1}^{N} w_i \mathrm{d}(y, y_i)^2. \tag{1.3}$$

where $\mathrm{d}(\cdot, \cdot)$ is a distance metric and $\{w_1, \ldots, w_N\}$ are weights. The interpolant on a manifold is often referred as an intrinsic mean. Since the optimization problem in (1.3) may have multiple solutions on manifolds, numerical procedures may find a local minimum (a Karcher mean) or a

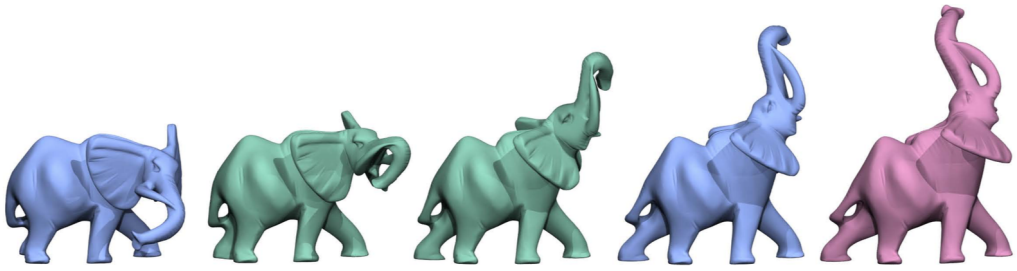


Figure 1.3: Interpolation of shapes (Kilian et al., 2007)

global minimum (Fréchet mean).

Interpolation of shapes has been studied in computer vision, graphics, and medical imaging. Shapes can be treated as points on a Riemannian manifold, called shape spaces pioneered by (Kendall, 1984). Multiple Riemannian frameworks have been proposed to interpolate shapes along geodesics with respect to useful Riemannian metrics Klassen et al. (2004); Kilian et al. (2007) as in 1.3. These can be viewed as finding a weighted mean with respect to (w.r.t) Riemannian metrics. Besides shapes, the means of structured matrices (Lie groups) have been studied (Manton, 2004; Moakher, 2006). Further, inductive/recursive intrinsic mean estimators have been studied on the hypersphere (Salehian et al., 2015), the Grassmannian manifold (Chakraborty and Vemuri, 2015), non-positively curved Riemannian manifolds (Cheng et al., 2016), and the SPD manifold (Cheng et al., 2012; Ho et al., 2013) for more computationally efficient estimation.

In neuroimaging, brain atlas estimation has been studied in the Riemannian setting by (Joshi et al., 2004; Fletcher et al., 2009). Also, voxel measurements in a brain image are often structured. So, transformation and manipulation of such brain images involve interpolations of structured measurements. To this end, multiple frameworks have been proposed, for instance, estimation and smoothing of diffusion tensor fields (Wang et al., 2004), splines for interpolation of diffusion tensor

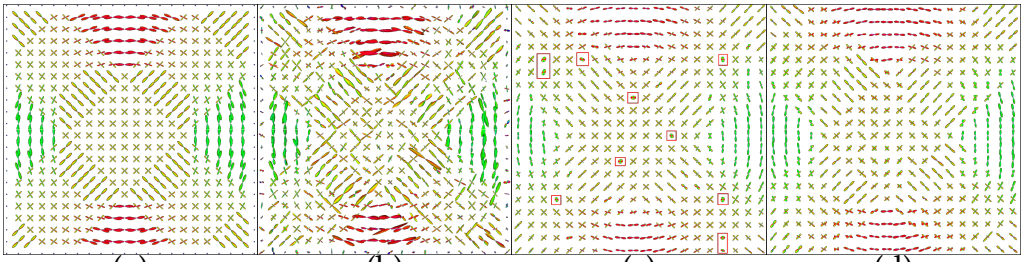


Figure 1.4: (a) Simulated EAP profiles. (b) EAP profiles with added Wishart noise. (c) Gaussian filtering. (d) Anisotropic filtering.

fields (Barmpoutis et al., 2007), orientation density funcions (ODFs) for high angular resolution diffusion images (Goh, 2010), ensemble average propagators (EAP) reconstruction via spherical polar Fourier diffusion MRI (Cheng et al., 2010), and a variety of EAP processing operiations (Cheng et al., 2011). A EAP profile is a random density function in the three dimensional space. EAPs are often represented by Gaussian mixture models (GMMs). A naive interpolation of GMMs in an L^2-space increases the number of components of GMMs resulting in more complex EAP profiles. Registration of brain images requires a non-trivial number of iterations involved with transformations of images. So, it easily reaches a huge number of model parameters after few iterations. Fig. 7.4 demonstrates that Gaussian filtering and anisotropic filtering without increasing the complexity of EAP profiles. We will study this problem in Chapter 7.

1.4.2 Regression for manifold-valued measurements

The notion of interpolation (or intrinsic mean) on manifolds naturally allows defining nonparametric regression on manifolds, e.g., kernel regression (Nadaraya-Watson kernel estimator) (Davis et al., 2007), and spline (Jupp and Kent, 1987). These models use interpolation as a module to perform regression.

Such regression models for structured data are useful in neuroimaging

analysis since brain images often comprises manifold-valued voxel measurements. For example, the DTI image in Fig. 1.5 has a 3×3 symmetric positive definite (SPD) matrix at each voxel. The most common analysis goal in neuroimaging is to associate a set of covariates such as age, gender and pathology with voxel-wise measurements. The relationship can be described as a function, i.e., $y = f(X_{age}, X_{gender}, X_{pathology})$. Since the voxel-wise measurement in DTI images is a SPD matrix, it cannot be directly handled by classical regression models. In the neuroimaging community, a simple solution (and still a widely used approach) is analysis based on a scalar summary of the manifold-valued data, e.g., univariate statistics from eigen values of diffusion tensors such as normalized standard deviation, mean, the first eigen value, and so on (Alexander et al., 2007).

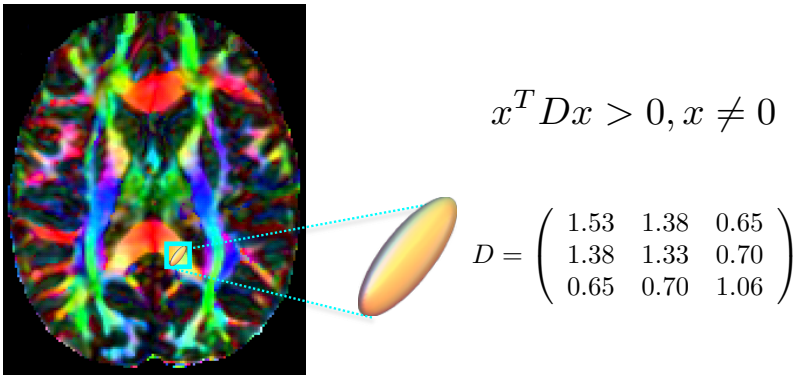


Figure 1.5: Diffusion tensor image with a structured measurement at each voxel.

Tensor-based morphometry (TBM) (Freeborough and Fox, 1998; Chung et al., 2001; Riddle et al., 2004; Frackowiak, 2004; Hua et al., 2008) is another example in neuroimaging. TBM is a deformation-based image analysis technique for measuring brain structural differences over different populations (cross-sectional study) or over time (longitudinal study). In cross-sectional studies, TBM is calculated with the spatial derivatives of

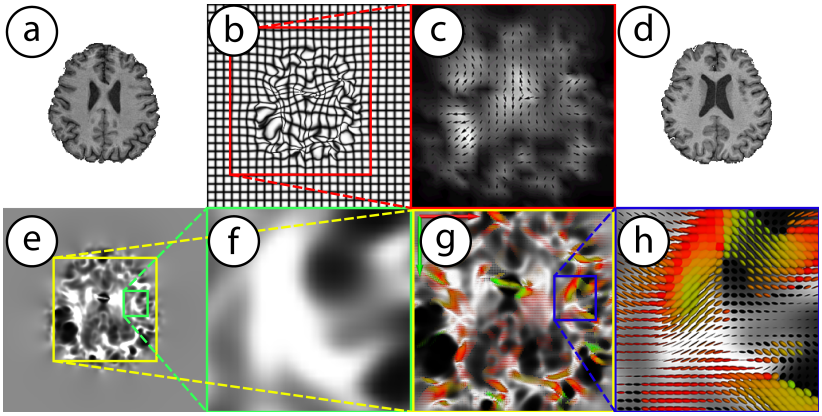


Figure 1.6: An example panel of data generated in morphometric studies. (a, d) The moving and fixed brain image respectively. (b) Warped spatial grid to move (a) to (d). (c) Vector field of local deformations. (e, f) A map of scalar features; det(J) of the deformation field. (g, h) The map of manifold-valued features known as Cauchy deformation tensor (CDTs) ($\sqrt{J^T J}$).

the transformations that align a set of subject images to a common anatomical template. In longitudinal studies, TBM is computed via nonlinear registration of multiple scans at different time points from the same individual. The classical TBM analysis uses the determinant of the Jacobian J (e,f) in Fig. 1.6, which is the spatial derivative matrix of the transformation. Obviously, the disadvantage of the classical analyses using scalar summaries of diffusion tensors and deformation tensors is that a significant amount of information is lost relative to the full manifold-valued measurements. So, any inference based on the summarized information about manifold-valued measurements is likely to suffer from poor statistical power.

To avoid the disadvantage of simple features derived from structured data, statistical models have been extended to manifolds. One of the simplest regression models in the Euclidean space is linear regression and it is also preferred since the number of samples is often small in neuroimaging study. But due to the technical difficulty in estimation

of parameters, parametric regression models are relatively less studied. Recently, generalizations of linear models on manifolds are studied using geodesics for linearity of models (Fletcher, 2013). Also, parametric nonlinear models are extended: polynomial regression (Hinkle et al., 2012) and geodesic regression with a parametric time-warping function (Hong et al., 2014). These models learn a regression function $f : \mathbf{R} \to \mathcal{M}$ so they allow only one covariate, .e.g., association between age and voxel measurement (Du et al., 2014). But these models are limited since the brain (and voxel measurements) may change as a function of multiple covariates (e.g., age, gender, and cognitive scores). To capture the association, linear regression models must be extended for $f : \mathbf{R}^d \to \mathcal{M}$. We study such an extension in Chapter 3.

1.4.3 Multimodal analysis and dimensionality reduction

Independent of the regression analysis described above, one common problem in image analysis is high-dimensionality of data. Specifically, in neuroimaging, each voxel value is not high-dimensional data but the number of voxels in a brain is large ($\approx 1M+$). So most voxel-wise analyses perform a large number of hypothesis tests simultaneously: as many as the number of voxels. Therefore, as the number of voxels increases, it is likely that we will see more tests with a significant p-value purely by chance, which is known as the multiple comparison problem (Benjamini and Hochberg, 1995; Hsu, 1996; Nichols and Hayasaka, 2003). To counteract the problem, in voxel-wise analysis of brain images, we need a multiple testing correction such as Bonferroni correction (Bonferroni, 1936; Perneger, 1998). This procedure is usually too conservative, especially when the number of voxels is large. It may remove the true positives as well. Also, each voxel is not truly independent for multiple reasons: spatial homogeneity of images, and preprocessing (smoothing). So based on random field theory, which takes spatial correlation into account, less

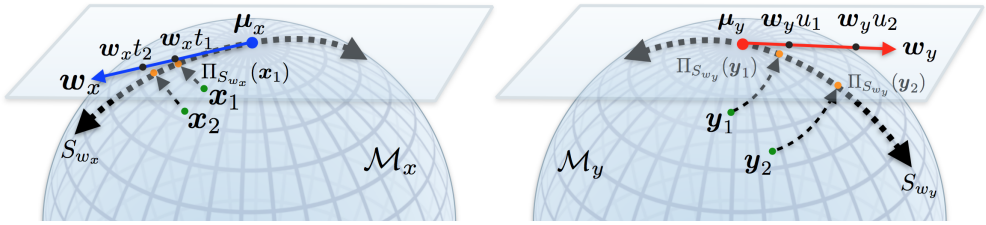


Figure 1.7: Canonical Correlation Analysis on manifolds.

stringent multiple comparison corrections have been proposed. Another potential solution is cluster-level analysis. since the number of hypothesis tests is much fewer than voxel-wise analysis. More systematically, a unified method using a conjugate Dirichlet process mixture model has been proposed in the literature and shown to be effective in the analysis of gene expression measured by microarrays (Dahl and Newton, 2007).

One may also address this issue by filtering out some of the voxels to reduce the number of hypothesis tests. However, filtering based on ROIs selected by human may introduce a bias in the final discovery. So data-driven filtering has been investigated by various authors. For example, a few years back, (Bourgon et al., 2010) showed that the statistical power of voxel-wise analysis (equivalently variable-by-variable statistical testing) can be improved by a two-stage approach; first filter voxels (or variables) by independent criterion of the test statistics and test hypothesis only over voxels (or variables) which pass the filter (Zheng et al., 2017).

Alternatively, dimensionality reduction algorithms may also play a filtering role. In various neuroimaging studies, for each participant, we may acquire different types of images such as computed tomography (CT), functional MRI (fMRI) and positron emission tomography (PET). Each imaging modality may capture a unique aspect of the disease pathology. Also, we may argue that the brain regions that are strongly correlated between different types of images may be important and can be used in downstream statistical analysis. For example, in a study of a large number

of subjects, rather than performing a hypothesis test on all brain voxels independently for each imaging modality, restricting the number of tests only to the set of 'relevant' voxels can improve statistical power.

So canonical correlation analysis (CCA) is a natural solution for voxel selections using multiple modalities of images. Since multivariate/manifold approaches outperform the Euclidean methods with a single type of images, multi-modal inference with manifold-valued measurements is expected to be more effective. We extend the CCA on manifolds for multi-modal neuroimaging data in Chapter 4, see in Fig. 1.7.

The related work for our construction, i.e., multi-modal inference for manifold-valued data, is limited. Except the structured data analysis aspect, multi-modal analysis has been studied by classical CCA and its non-linear variants. These include various interesting results based on kernelization (Akaho, 2006; Bach and Jordan, 2002; Hardoon et al., 2007), neural networks (Lai and Fyfe, 1999; Hsieh, 2000), and deep architectures (Andrew et al., 2013). The second line of work incorporates the specific geometry of the data directly within the estimation problem. Among projective dimensionality reduction methods, Principal Components Analysis (PCA) has been mainly generalized to Riemannian manifolds: PCA for spherical data (Jung et al., 2010, 2012), the generalization of PCA to Riemannian manifolds via the so-called Principal Geodesic Analysis (PGA) (Fletcher et al., 2004), Geodesic PCA (Huckemann et al., 2010a), Exact PGA (Sommer et al., 2014a), Horizontal Dimension Reduction (Sommer, 2013) with frame bundles, and an extension of PGA to the product space of Riemannian manifolds, namely, tensor fields (Xie et al., 2010). Separately, PCA has been generalized for tress (Aydın et al., 2012) as well.

1.4.4 Longitudinal analysis of structured data

The goal of longitudinal analysis is to model the subject-specific change over time as well as the population-level trajectories. This is a fundamental

tool to study time dependent image data (e.g., medical images as in Fig. 1.8). A longitudinal analysis naturally involves multiple samples from the same subject at multiple time points whereas cross-sectional analysis assumes that data are collected at a time point (i.e., one sample from one subject).

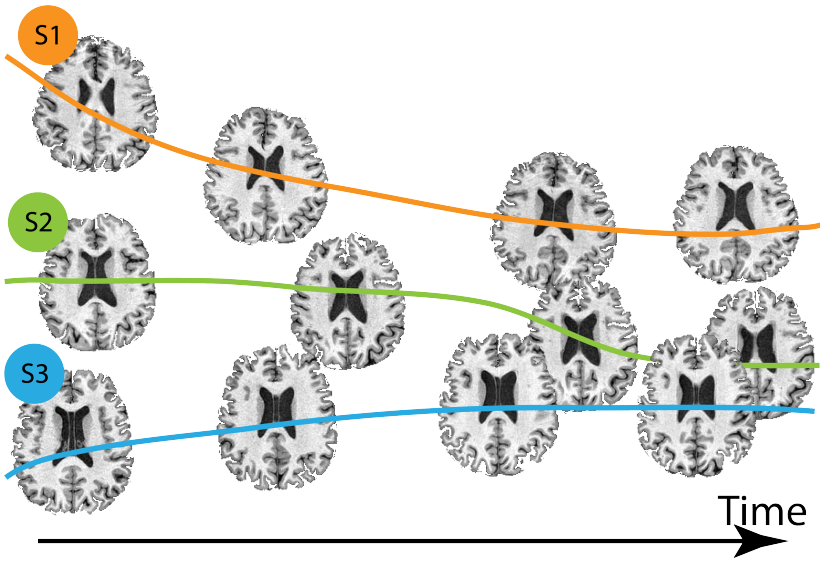


Figure 1.8: Trajectories of brain structures. Each subject (*S*1-*S*3) is observed at multiple time points. To analyze a population level trend, the structural changes need to be transported in a common coordinate system.

Samples from a particular subject are inherently not independent. Also, the trajectories (e.g., speed of change and start of change) may vary depending on subjects. This variability is called *random effects*, whereas the population level pattern captured by classical regression models is called *fixed effects* (Laird and Ware, 1982). Each subject has its own random effects and the samples from the subject are assumed to be affected by the random effects. So, a more accurate longitudinal analysis can be achieved by proper handling the subject-specific random effects (dependency between samples). To this end, in the literature, mixed effects models, capturing fixed effects and random effects, have been extensively

studied, for instance, linear mixed effects models (Laird and Ware, 1982), generalized linear mixed effect models (Wolfinger and O'connell, 1993; McCulloch and Neuhaus, 2001) and varying coefficient models (Hastie and Tibshirani, 1993). However, the generalization of these models for structured measurement is very limited. Recently, nonlinear mixed effects models on a unit interval with a specific Riemannian metric was proposed (Schiratti et al., 2015). But this model is not flexible enough to handle general manifold-valued data.

Separate from the statistical literature, longitudinal image analysis is an important topic in medical imaging as well. Modeling anatomical changes over time is crucial to study brain development, aging and disease progression. A representation of structural changes is registration maps. In neuroimaging, the registration maps are assumed to be diffeomorphisms (i.e, smooth, invertible and topology-preserving). For time-varying structures, geodesic regression (Niethammer et al., 2011) and age-specific brain atlas estimation (Yoon et al., 2009). But these methods are agnostic to subject-specific trajectories (dependency of samples from a particular subject). Most recent attempts estimate the population level trajectory simply by averaging subject-specific trajectories (Durrleman et al., 2009; Fishbaugh et al., 2012; Lorenzi et al., 2011a). These methods use the sample dependency within a subject but it is not flexible enough to consider general *random effects*. We will study more flexible mixed effects models in the context of capturing anatomical changes over time to take the dependency of samples (subject-specific random effects) into account in Chapter 6. Additionally, the extended mixed effects models provide interpretability, e.g., aging/disease progression with subject-specific time shift, acceleration, and intercepts for structured measurements.

1.5 Structure of the thesis

The rest of the thesis organized as follows:

Chapter 2 starts with the brief introduction of some concepts from differential geometry that are relevant to most of the dissertation. Other concepts that are used in a specific chapter will be introduced in that chapter as needed.

Chapter 3 based on our work in (Kim et al., 2014b) studies Manifold-valued Multivariate General Linear Models (MMGLMs), which are a generalization of linear regression models on Riemannian manifolds. We demonstrate that MMGLMs improves statistical power in statistical analysis of diffusion weighted images.

Chapter 4 based on our work in (Kim et al., 2014a, 2016b) extends Canonical Correlation Analysis (CCA) on manifolds and identifies meaningful correlations across diffusion tensor images (DTI) and Cauchy deformation tensor (CDT) fields. We develop efficient estimation schemes with computationally efficient operations on $\mathrm{SPD}(n)$.

Chapter 5 based on our work in (Kim et al., 2015b) establishes a nonparametric nonlinear regression model on manifolds. Using nonparametric Bayesian priors and MMGLMs, we develop the Dirichlet mixtures of multivariate general linear models on $\mathrm{SPD}(n)$ and show that it captures more complex patterns than MMGLMs. Also, we derive an efficient sampling methods for structured parameters.

Chapter 6 based on our work in (Kim et al., 2017a, 2016a) discusses tensor-based longitudinal analysis with manifold-valued data that captures local deformation. It provides a new Nonlinear Mixed Effects Models on Riemannian manifolds. The estimated random effects can be used for downstream analysis and offers interpretability of models at the level of individual subjects.

Chapter 7 based on our work in (Kim et al., 2015a) studies the interpolation on the manifold of K component Gaussian mixture models

(GMMs) without increasing the complexity of resulting interpolant. We also discuss the relationship of the proposed framework with functional clustering of probability density functions.

Chapter 8 summarizes the contributions of the thesis and discusses future directions.

2 PRELIMINARY

The models in the thesis are mainly developed based on tools from Riemannian geometry. We briefly summarize basic concepts and notations that the remainder of the thesis will utilize. Riemannian manifolds are very useful non-linear spaces to analyze data with nice mathematical properties. While we introduce relevant concepts that are used later in the thesis, additional details can be found in several excellent textbooks (Do Carmo, 1992; Amari, 1985; Amari and Nagaoka, 2000; Spivak, 1981).

Riemannian manifolds consist of three structural concepts: topological structure, differentiable structure and Riemannian metric. The topological structure allows defining topological notions (e.g., continuity and convergence). The differentiable structure (*smooth* manifolds) enables generalizing calculus since the charts of the smooth manifolds are suitably compatible (the transition between charts is differentiable). Lastly, Riemannian metric defines geometric quantities, e.g., length of curves, distance, angles, and curvature.

2.1 Topological manifolds

A Riemannian manifold is a topological manifold, which is a topological space which locally resembles a n-dimensional Euclidean space. The topological structure allows defining topological notions such as open sets, continuity and convergence. We start with the basic definition of a topological space.

Definition 2.1. *Let X a set and $\mathcal{T}$ be a collection of subsets of X. A* ***topological space*** *$(X, \mathcal{T})$ satisfies the following properties:*

1. *The empty set $\phi \in \mathcal{T}$ and $X \in \mathcal{T}$*
2. *Any union of elements of $\mathcal{T}$ is in $\mathcal{T}$*

3. *The intersection of any finite number of elements of $\mathcal{T}$ is in $\mathcal{T}$.*

Definition 2.2. *A topological space X is said to be **Hausdorff** if for any two points $x, y \in X$ and $x \neq y$, there exist two open sets U and V such that $U \cap V = \phi$ and $x \in U, y \in V$.*

Definition 2.3. *A **homeomorphism** between two topological spaces X and Y is a bijective function $f : X \to Y$ that both f and f^{-1} are continuous. If there exist a homeomorphism between X and Y, we then say that X is homeomorphic to Y.*

Now using the notion of homeomorphism, we are able to locally map the topological space (manifold) to the Euclidean space and generalize well-defined concepts in Euclidean space to manifolds.

Definition 2.4. *A n-dimensional **topological manifold** X is a **Hausdorff** space where every point $x \in X$ has a neighborhood $U \in X$ homeomorphic to an open set $\varphi(U) \in \boldsymbol{R}^d$. The local homeomorphism $\varphi : U \subset X \to \boldsymbol{R}^d$ is called a **coordinate chart** on U, which is often denoted by (U, φ), see Fig. 2.1.*

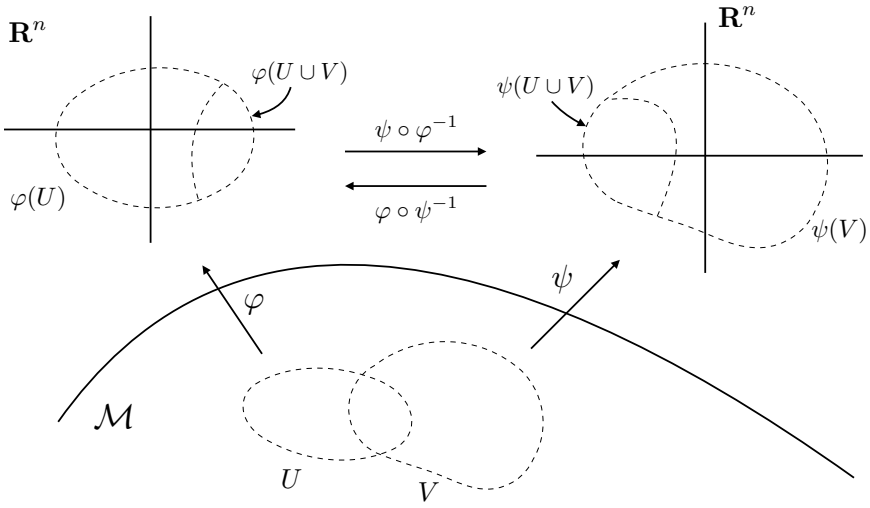


Figure 2.1: Coordinate charts.

The Hausdorff space is important to prevent unusual and counterintuitive convergence behavior in analysis since a non-Hausdorff topological space may have several distinct limit points. In addition to the Hausdorff property, manifolds are assumed to be *second-countable*, i.e., it has a countable topological basis. Second-countability is relevant to *partitions of unity* which is a theoretical tool to answer questions such as the existence of a Riemannian metric and an affine connection. This abstract definition of manifolds by Hausdorff and second countability is equivalent to the definition of manifolds embedded in the ambient Euclidean space by Whitney's embedding theorem (Lee, 2012).

A manifold is *connected* if there exist no disjoint union of two nonempty open sets. Equivalently, any two points in a connected (or path-connected) manifold can be joined by a piecewise smooth curve segment (*not* geodesic curves).

2.2 Differentiable manifolds

We now are ready to introduce the differentiable structure of manifolds. Given two coordinate charts (ϕ, U) and (ψ, V), the *transition function* is defined as

$$\phi \circ \psi^{-1} : \psi(U \cup V) \subset \mathbf{R}^n \to \phi(U \cup V) \subset \mathbf{R}^n. \tag{2.1}$$

where $U \cap V \neq \emptyset$.

Two charts (U, ϕ) and (V, ψ) are called *compatible* if the transition maps are *smooth* (differentiable), see Fig. 2.1. If every pair of charts of a manifold $\mathcal{M}$ is compatible, then the manifold $\mathcal{M}$ is called a *smooth (or differentiable) manifold*. Using charts, we can also generalize the differentiability of real valued functions on manifolds $f : \mathcal{M} \to R$ and functions from one manifold to another $f : \mathcal{M} \to \mathcal{N}$ where $\mathcal{N}$ is a different manifold. A continuous function $f : \mathcal{M} \to \mathbf{R}$ is said to be differentiable if for every

chart ϕ_U the function $f \circ \phi_U^{-1}$ is differentiable. Further, a map between differentiable manifolds, i.e., $f : \mathcal{M} \to \mathcal{N}$ is said to be differentiable if $r \circ f$ is differentiable for any differentiable $r : \mathcal{N} \to \mathbf{R}$. The map f between two differentiable manifolds is called *diffeomorphism* if it is a bijection and both f and f^{-1} are differentiable. Basically, a diffeomorphism is a differentiable homeomorphism. This differentiable structure allows calculus with functions defined with differentiable manifolds.

Note that a diffeomorphism between two manifolds implies that two manifolds have the same dimension due to the *inverse function theorem*. While homeomorphism is used to show the topological equivalence between two topological spaces, diffeomorphism is used for equivalence between two differentiable manifolds.

We have now defined the differentiable structure of manifolds. It naturally leads to the definitions of tangent vectors and tangent spaces. A tangent space $T_p\mathcal{M}$ at point $p \in \mathcal{M}$ is a vector space, which an n-dimensional vector space $T_p\mathcal{M}$ and isomorphic to $\mathbf{R}^n$. The elements of the tangent space are tangent vectors of smooth curves on $\mathcal{M}$ passing through $p \in \mathcal{M}$.

Definition 2.5. *Let $\gamma : [-\epsilon, \epsilon] \to \mathcal{M}, \gamma(0) = p \in \mathcal{M}$ be a smooth curve passing through p on a differentiable manifold $\mathcal{M}$. Let f be any differentiable function defined in a neighborhood of p. Then the **tangent vector** to the curve γ at $t = 0$ is defined as the operator $\gamma'(0)$ that maps function f to its directional derivative, i.e.*

$$\gamma'(0)f = \left.\frac{df \circ \gamma}{dt}\right|_{t=0}. \tag{2.2}$$

*The collection of all tangent vectors at $p \in \mathcal{M}$ is the **tangent space** denoted by $T_p\mathcal{M}$.*

The **tangent bundle** of $\mathcal{M}$, i.e., $T\mathcal{M}$, is the disjoint union of tangent spaces at all points of $\mathcal{M}$, $T\mathcal{M} = \coprod_{p\in\mathcal{M}} T_p\mathcal{M}$. There exists a natural *projection mapping* $\pi : T\mathcal{M} \to \mathcal{M}$ defined by $\phi(p, v) = p$ (Lee, 2012).

2.3 Riemannian manifold

We finally define a Riemannian manifold by adding a notion of distance, i.e., Riemannian metric g, to a smooth manifold. A Riemannian manifold $(\mathcal{M}, g)$ is a differentiable manifold $\mathcal{M}$ equipped with a smoothly varying inner product (Riemannian metric g).

Definition 2.6. *A **Riemannian metric** g on a differentiable manifold $\mathcal{M}$ is a smoothly varying inner product $g_p : T_p\mathcal{M} \times T_p\mathcal{M} \to \boldsymbol{R}$ on each of the tangent. In other words, for each $p \in \mathcal{M}$, g_p satisfies the follows:*

1. $g_p(u, v) = g(v, u)$ *for all* $u, v \in T_p\mathcal{M}$*; symmetric*
2. $g_p\left(\sum_{i=1}^n u_i, \sum_{i=1}^n v_i\right) = \sum_{i=1}^n \sum_{j=1}^n g_p(u_i, v_j)$*; bi-linear*
3. $g_p(u, u) \geq 0$ *for all* $u \in T_p\mathcal{M}$*; positive definite*
4. $g_p(u, u) = 0 \Leftrightarrow u = 0$*; positive definite*

Also g is smooth in the sense that for any two smooth vector fields X and Y , the function $p \in \mathcal{M} \to g_p(X_p, Y_p)$ is smooth (Pflaum, 2001; Lee, 2012).

The Riemannian metric allows measuring the length of a smooth curve $\gamma : [a, b] \to \mathcal{M}$ on the manifold by

$$L(\gamma) = \int_a^b \sqrt{g_x(\dot{\gamma}(t), \dot{\gamma}(t))}dt \tag{2.3}$$

where $\dot{\gamma(t)}$ is the tangent vector of γ at t. With this definition of the length of a curve, a *geodesic curve* on a Riemannian manifold is a locally length-minimizing path. The geodesic curve is a generalization of a straight line to a curved space. Later in the thesis, we use geodesic curves to generalize linear models, e.g., manifold-valued multivariate general models (MMGLMs) in Chapter 3 and Riemannian canonical correlation analysis (RCCA) in Chapter 4.

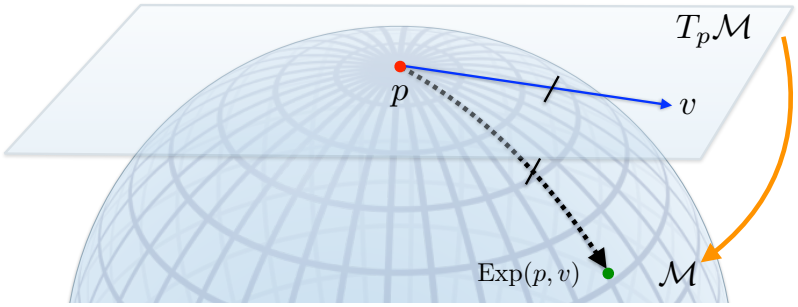


Figure 2.2: The exponential map $\mathrm{Exp}(p, v)$ maps v to a point on manifold $\mathcal{M}$ preserving the length of tangent vector v, i.e., the length of geodesic curve parameterized by v from p is the same as the norm of v in $T_p\mathcal{M}$.

The notion of the length of a curve naturally defines the distance between two points on manifolds given as

$$\mathrm{d}(p, q) = \inf_{\gamma \in \Gamma(p,q)} \int_0^1 \sqrt{g_{\gamma(t)}(\dot{\gamma}(t), \dot{\gamma}(t))} dt, \tag{2.4}$$

where $\Gamma(p, q)$ is a set of all piecewise smooth curves joining $p = \gamma(0)$ and $q = \gamma(1)$. This distance is called the *geodesic distance*. If the infimum is achieved by a smooth curve, it is a geodesic curve. But a geodesic curve between two points may not achieve the minimum length in general since the geodesic curve between two points may not be unique.

In a sufficiently small neighborhood where there exist a unique solution to *geodesic equation* (an ordinary partial equation) with initial point $\gamma(a) = p$ and initial velocity $\dot{\gamma(a)} = v$, the solution defines a map from a tangent vector $v \in T_p\mathcal{M}$ to a point $q \in \mathcal{M}$. This mapping is called the **exponential map** $\mathrm{Exp}(y_i, \cdot) : T_{y_i}\mathcal{M} \to \mathcal{M}$. It is defined formally as

Definition 2.7. *(Spivak, 1981) If $v \in T_p\mathcal{M}$ is a vector for which there is a geodesic*

$$\gamma : [0, 1] \to \mathcal{M} \tag{2.5}$$

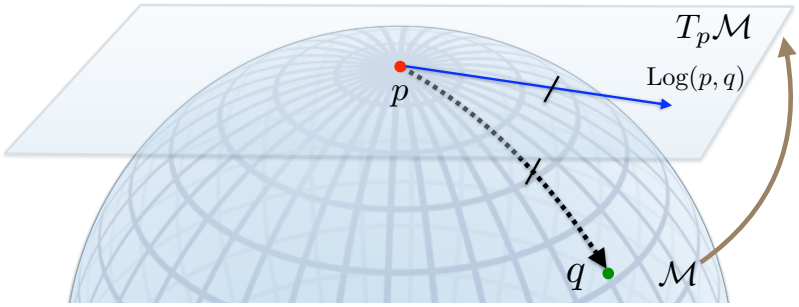


Figure 2.3: The logarithm map $\mathrm{Log}(p,q)$ maps $q \in \mathcal{M}$ to a tangent vector $v \in T_p\mathcal{M}$ preserving the length of the geodesic curve between p and q.

satisfying

$$\gamma(0) = p, \frac{d\gamma}{dt}(0) = v, \tag{2.6}$$

then we define the ***exponential*** *of* v *to be*

$$Exp(p,v) = \gamma(1) \tag{2.7}$$

In this thesis, we will often use multiple nested exponential maps. For better readability, we use a slightly different notation, i.e., $\mathrm{Exp}(p,v) := \mathrm{Exp}_p(v) := \mathrm{Exp}_p v$.

The exponential map is a local diffeomorphism (i.e., differentiable, bijective, and continuous). So, the inverse map is well defined within a small neighborhood. It is called the **logarithm map**.

Definition 2.8. *Given two points* $p, q \in \mathcal{M}$*, if there exists* $v \in T_p\mathcal{M}$ *such that* $Exp(p,v) = q$*, then the* ***logarithm map*** $Log(p,\cdot) : \mathcal{M} \to T_p\mathcal{M}$ *is defined as*

$$Log(p,q) = v. \tag{2.8}$$

Since the logarithm map is the inverse of exponential map, we have trivial identities, e.g., $\mathrm{Log}(p, \mathrm{Exp}(p,v)) = v$ and $\mathrm{Exp}(p, \mathrm{Log}(p,q)) = q$.

The geodesic curve from y_i to y_j can be parameterized by a tangent

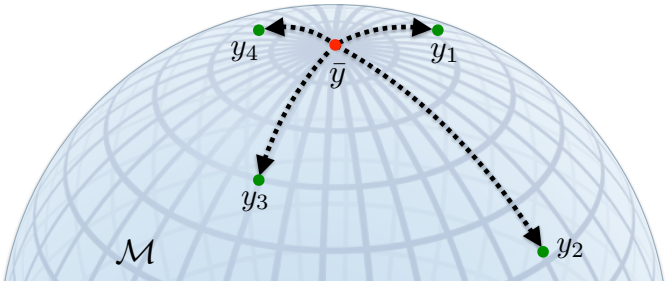


Figure 2.4: Intrinsic mean (or Fréchet mean) $\bar{y}$ is minimizing the sum of squared geodesic distance to data points $y_1, \ldots, y_4$ on manifold $\mathcal{M}$.

vector in the tangent space at y_i with an exponential map $\mathrm{Exp}(y_i, \cdot) : T_{y_i}\mathcal{M} \to \mathcal{M}$. The inverse of the exponential map is the logarithm map, $\mathrm{Log}(y_i, \cdot) : \mathcal{M} \to T_{y_i}\mathcal{M}$. For completeness, Table 2.1 shows corresponding operations in the Euclidean space and Riemannian manifolds. Separate from the above notation, matrix exponential (and logarithm) are simply given as $\exp(\cdot)$ (and $\log(\cdot)$).

Intrinsic mean. Let $d(\cdot, \cdot)$ define the distance between two points. The intrinsic (or Karcher) mean is the minimizer to

$$\bar{y} = \arg\min_{y \in \mathcal{M}} \sum_{i=1}^{N} w_i d(y, y_i)^2, \tag{2.9}$$

which may be an arithmetic, geometric or harmonic mean depending on $d(\cdot, \cdot)$, see Fig. 2.4. On manifolds, the Karcher mean with distance $d(y_i, y_j) = \|\mathrm{Log}_{y_i} y_j\|$ satisfies $\sum_{i=1}^{N} \mathrm{Log}_{\bar{y}} y_i = 0$. This identity means that $\bar{y}$ is a local minimum which has a zero norm gradient (Karcher, 1977), i.e., the sum of all tangent vectors corresponding to geodesic curves from mean $\bar{y}$ to all points y_i is zero in the tangent space $T_{\bar{y}}\mathcal{M}$. On manifolds, the existence and uniqueness of the Karcher mean is not guaranteed unless we assume, for uniqueness, that the data is in a small neighborhood.

The Karcher mean is obtained by Algorithm 1, where α denotes the

step size ($\alpha = 1$ was used in our experiments). Karcher mean can be

Algorithm 1 Karcher mean

Input: $y_1, \ldots, y_N \in \mathcal{M}$, α
Output: $\bar{y} \in \mathcal{M}$
$\bar{y}_0 = y_1$
while $\| \sum_{i=1}^{N} \mathrm{Log}(\bar{y}_k, y_i) \| > \epsilon$ **do**
  $\Delta\bar{y} = \frac{\alpha}{N} \sum_{i=1}^{N} \mathrm{Log}(\bar{y}_k, y_i)$
  $\bar{y}_{k+1} = \mathrm{Exp}(\bar{y}_k, \Delta\bar{y})$
end while

estimated by second-order methods as well. To do so, we need a notion of a Hessian on manifolds. This naturally leads to the so-called *affine connection* (Boumal, 2014)

Definition 2.9. *(affine connection). Let $\mathfrak{X}(\mathcal{M})$ and $\mathfrak{F}(\mathcal{M})$ be the a set of smooth vector fields on $\mathcal{M}$ and the set of smooth functions on $\mathcal{M}$. An affine connection ∇ on $\mathcal{M}$ is a mapping*

$$\nabla : \mathfrak{X}(\mathcal{M}) \times \mathfrak{X}(\mathcal{M}) \to \mathfrak{X}(\mathcal{M}) : (X, Y) \mapsto \nabla_X Y \tag{2.10}$$

which satisfies the following:

1. *$\mathfrak{F}(\mathcal{M})$-linearity in X: $\nabla_{fX+gY} Z = f\nabla_X Z + g\nabla_Y Z$*
2. *$\boldsymbol{R}$-linearity in Y: $\nabla_X(aY + bZ) = a\nabla_X Y + b\nabla_X Z$*
3. *Product rule (Leibniz' law): $\nabla_X(fX)Y + f\nabla_X Y$,*

where $X, Y, Z \in \mathfrak{X}(\mathcal{M})$, $f, g \in \mathfrak{F}(\mathcal{M})$ and $a, b \in \boldsymbol{R}$.

The affine connection is a foundation to develop a notion of the *Riemannian Hessian* (Boumal, 2014) for the second order optimization scheme. For more details, we refer to (Absil et al., 2009; Boumal, 2014). In this thesis, we mostly use the first order methods.

Parallel transport. Let $\mathcal{M}$ be a differentiable manifold with an affine connection ∇ and I be an open interval. Let $c : I \to \mathcal{M}$ be a differentiable curve in $\mathcal{M}$ and let V_0 be a tangent vector in $T_{c(t_0)}\mathcal{M}$, where $t_0 \in I$. Then, there exists a unique parallel vector field V along c, such that $V(t_0) = V_0$. Here, $V(t)$ is called the *parallel transport* of $V(t_0)$ along c.

Convexity of sets and functions are useful tools for analysis and numerical optimization. We briefly introduce generalized convexity on manifolds. We will use the following concepts to reformulate optimization problems.

Geodesically convex set. A subset C of $\mathcal{M}$ is said to be a *geodesically convex set* if there is a minimizing geodesic curve in C between *any* two points in C. This condition is commonly used for analysis (Corcuera and Kendall, 1999; Papadopoulos, 2005) and essential to ensure that the Riemannian operations such as the exponential and logarithm maps are well-defined.

Geodesically convex function. Let $A \subset \mathcal{M}$ be a geodesically convex set. Then, a function $f : A \to R$ is geodesically convex if its restrictions to all geodesic arcs belonging to A are convex in the arc length parameter, i.e., if $t \mapsto f(t) \equiv f(\mathrm{Exp}(x, tu))$ is convex over its domain for all $x \in \mathcal{M}$ and $u \in T_x\mathcal{M}$, where $\mathrm{Exp}(x, \cdot)$ is the exponential map at x (Moakher, 2005).

Operation	Euclidean	Riemannian
Subtraction	$x_i\overrightarrow{x_j} = x_j - x_i$	$x_i\overrightarrow{x_j} = \mathrm{Log}(x_i, x_j)$
Addition	$x_i + x_j\overrightarrow{x_k}$	$\mathrm{Exp}(x_i, x_j\overrightarrow{x_k})$
Distance	$\|x_i\overrightarrow{x_j}\|$	$\|\mathrm{Log}(x_i, x_j)\|_{x_i}$
Mean	$\sum_{i=1}^{n} \bar{x}\overrightarrow{x_i} = 0$	$\sum_{i=1}^{n} \mathrm{Log}(\bar{x}, x_i) = 0$
Covariance	$\mathbb{E}\left[(x_i - \bar{x})(x_i - \bar{x})^T\right]$	$\mathbb{E}\left[\mathrm{Log}(\bar{x}, x)\mathrm{Log}(\bar{x}, x)^T\right]$

Table 2.1: Basic operations in Euclidean space and Riemannian manifolds.

Generalized normal distributions Let $\overline{X} \in \mathcal{M}$ and $\sigma \in \mathbf{R}_+$. One generalization of the Gaussian distribution on Riemannian manifolds is given by

$$\begin{aligned} f(X; \overline{X}, \sigma) &= \frac{1}{Z(\overline{X}, \sigma)} \exp\left(-\frac{\mathrm{d}(X, \overline{X})^2}{2\sigma^2}\right), \\ \text{where } Z(\overline{X}, \sigma) &= \int_{\mathcal{M}} \exp\left(\frac{d(X, \overline{X})^2}{2\sigma^2}\right) dX. \end{aligned} \tag{2.11}$$

$d(\cdot, \cdot)$ denotes the geodesic distance between two manifold-valued data points. On SPD(n), we use the affine-invariant Riemannian metric for $d(\cdot, \cdot)$. $\overline{X} \in \mathcal{M}$ and $\sigma \in \mathbf{R}_+$ corresponds to the mean and variance. The notation σ denotes dispersion of manifold-valued variables. Multiple generalizations of Gaussian distributions are discussed in (Pennec, 2006; Cheng and Vemuri, 2013; Fletcher, 2013). $Z(\mu, \sigma)$ is the normalization factor to make the integration of the PDF in the space of SPD(n) work. It is known that $Z(\overline{X}, \sigma)$ is independent from μ on Riemannian symmetric spaces (Fletcher, 2013). However, it is difficult to calculate the normalization factor in practice (Said et al., 2017). This may result in a challenging maximum likelihood estimation of the dispersion (σ).

2.4 Manifolds for diffusion weighted imaging

In this thesis, one of main application topics is the analysis of diffusion weighted imaging (DWI). We now discuss some example manifolds, which are used for diffusion weighted imaging analysis.

Note that an inner product is the first step in defining Riemannian manifolds. To manipulate data, we also need the exponential map $\mathrm{Exp}(p, \cdot)$, logarithm map $\mathrm{Log}(p, \cdot)$, and parallel transport $\Gamma_{p \to q}(\cdot)$. Here, we provide details specific to each manifold.

2.4.1 Unit sphere

The unit sphere is a common manifold, and the first example demonstrated here. Let $S^n = \{p \in \mathbf{R}^{n+1} | p^T p = 1\}$ be the unit sphere in $\mathbf{R}^{n+1}$. Its tangent space is then,

$$T_p S^n = \{v \in \mathbf{R}^{n+1} | p^T v = 0\}$$

The inner product of two tangent vectors $u, v \in T_p\mathcal{M}$ is given by

$$\langle u, v \rangle_p = u^T v, \tag{2.12}$$

and the norm at p is

$$\|v\|_p = \sqrt{\langle u, v \rangle_p} \tag{2.13}$$

Since the inner product and the norm of tangent vectors are independent of p, we omit p in the following expressions. The main operators we need in this thesis on the unit sphere are,

$$\begin{aligned}
\mathrm{Exp}(p, v) &= p\cos(\|v\|) + \frac{v}{\|v\|}\sin(\|v\|) \\
\mathrm{Log}(p, q) &= \frac{(I - pp^T)q}{\sqrt{1 - (p^T q)^2}}\,\mathrm{acos}(p^T q) \\
\Gamma_{p \to q}(w) &= -p\sin(\|v\|)v^T w + \frac{v}{\|v\|}\cos(\|v\|)v^T w \\
&\quad + (I - \frac{vv^T}{\|v\|^2})w, \text{where } v = \mathrm{Log}(p, q)
\end{aligned} \tag{2.14}$$

2.4.2 Hilbert unit sphere

Probability density functions (PDFs) typically represent diffusion of water molecules. Using the square root parameterization, PDFs form a unit

Hilbert sphere (Cheng et al., 2009),

$$\Psi = \{\psi : \mathbb{S}^2 \to \mathbb{R}^+ | \forall s \in \mathbb{S}^2, \psi(s) \geq 0; \int_{s \in \mathbb{S}^2} \psi^2(s) ds = 1\} \tag{2.15}$$

The inner product is given by

$$\langle \xi_i, \xi_j \rangle_{\psi_i} = \int_{s \in \mathbb{S}^2} \xi_i(s) \xi_j(s) ds, \tag{2.16}$$

and coincides with the Fisher-Rao metric (Srivastava et al., 2007). All other operations needed are the same as those shown for the unit sphere above.

2.4.3 Symmetric positive definite matrices

Symmetric positive definite (SPD) matrices are used for the so-called "diffusion tensors" are at each voxel in the image (we will describe details of such image data). Also, we will use SPD manifolds for the representations of image-to-image warps in Chapter 6 and covariance descriptor (for texture) in Chapter 5. Let $\text{SPD}(n)$ be a manifold for symmetric positive definite matrices of size $n \times n$. This forms a *quotient space* $GL(n)/O(n)$, where $GL(n)$ denotes the general linear group (the group of $(n \times n)$ non-singular matrices) and $O(n)$ is the orthogonal group (the group of $(n \times n)$ orthogonal matrices). The quotient space is, intuitively speaking, a space equipped with an *equivalence relation* $\sim$, i.e., a relation that is

- reflexive: $x \sim x$ for all $x \in \mathcal{M}$
- symmetric: $x \sim y \Leftrightarrow y \sim x$ for all $x, y \in \mathcal{M}$
- transitive: if $x \sim y$ and $y \sim z$, then $x \sim z$ for all $x, y, z \in \mathcal{M}$.

The isomorphism $\text{SPD}(n) \cong GL(n)/O(n)$ can be explained with the following characterization. A matrix $P \in SPD(n)$ can be factorized as

$$P = ZZ^T = (ZO)(ZO)^T, \tag{2.17}$$

where $Z \in GL(n)$ and $O \in O(n)$. It is easy to see that the P is not affected by orthogonal transformation $Z \mapsto ZO$ and this defines the equivalence relation and the quotient geometry (Bonnabel and Sepulchre, 2009).

The inner product of two tangent vectors $u, v \in T_p\mathcal{M}$ is given by

$$\langle u, v \rangle_p = \text{tr}(p^{-1/2}up^{-1}vp^{-1/2}), \tag{2.18}$$

where $\text{tr}(\cdot)$ denotes the trace of a square matrix. This plays the role of the Fisher-Rao metric in the statistical model of multivariate distributions (Petz, 2005). Here, $T_p\mathcal{M}$ is a tangent space at p (which is a vector space) is the space of symmetric matrices of dimension $(n+1)n/2$. The geodesic distance is,

$$d(p, q)^2 = \text{tr}(\log^2(p^{-1/2}qp^{-1/2}))$$

The exponential map and logarithm map are,

$$\begin{aligned} \text{Exp}(p, v) &= p^{1/2} \exp(p^{-1/2}vp^{-1/2})p^{1/2} \\ \text{Log}(p, q) &= p^{1/2} \log(p^{-1/2}qp^{-1/2})p^{1/2} \end{aligned} \tag{2.19}$$

Let p, q be in $P(n)$ and a tangent vector $w \in T_p\mathcal{M}$, the tangent vector in $T_q\mathcal{M}$ which is the parallel transport of w along the shortest geodesic from p to q is given by (Ferreira et al., 2006)

$$\begin{aligned} &\Gamma_{p \to q}(w) = p^{1/2}rp^{-1/2}wp^{-1/2}rp^{1/2} \\ &\text{where } r = \exp\left(p^{-1/2}\frac{v}{2}p^{-1/2}\right) \text{ and } v = \text{Log}(p, q). \end{aligned} \tag{2.20}$$

2.5 Optimization on manifolds

Optimization on (Riemannian) manifolds is important to estimate statistical machine learning models for structured data, especially when the data belong to Riemannian manifolds. Even if data are in a vector space, we encounter optimization on manifolds due to certain types of constraints on parameters (e.g., Orthogonal Procrustes problem (Gower and Dijksterhuis, 2004) and generalized eigenvalue problem (Jae Hwang et al., 2015)). This is a fast growing research topic (Absil et al., 2009; Boumal et al., 2014; Boumal, 2014).

The main goal of optimization on manifolds is to provide efficient/stable numerical algorithms to find local/global minimum to

$$\min_{x \in \mathcal{M}} f(x) \tag{2.21}$$

without leaving the feasible region $\mathcal{M}$, which is a manifold.

Early attempts to address optimization on manifolds (Gabay, 1982; Udriste, 1994; Yang, 2007) introduced a gradient descent, Newton methods, and quasi-Newton methods with convergence analysis. Also, trust Region algorithms (Absil et al., 2007; Boumal and Absil, 2011), and Nedler-Mead method (Dreisigmeyer, 2006) have been studied.

In this section, we briefly introduce a simple first order method on manifolds. Let us first revisit a gradient descent method in Euclidean space. The main update rule at the k_{th} step is given as

$$x_{k+1} = x_k + \alpha_k \eta_k, \tag{2.22}$$

where α_k is a step size, η_k is a gradient or descent direction.

We can naturally extend the update in (2.22) to manifolds. The update indeed is to find a solution along a straight line parameterized by η_k. It can be generalized by substituting the straight line with a geodesic

curve parameterized by a tangent vector i.e., $\eta_k \in T_{x_k}\mathcal{M}$. *Exponential map*, $\text{Exp}(x, \cdot) : \mathcal{M} \to T_x\mathcal{M}$, maps the geodesic curve given a tangent vector and allows staying within a manifold after the update.

Another approach is to search solutions on manifolds along more general paths using *retraction* R. The retraction R at x is denoted by R_x and it is also a mapping from $T_x\mathcal{M}$ to $\mathcal{M}$. These mappings (exponential map and retraction) can be viewed as projections to get the feasible solutions. Then the update on manifolds is given by

$$x_{k+1} = \text{Exp}(x_k, \alpha_k \eta_k), \tag{2.23}$$

or with a retraction

$$x_{k+1} = R_{x_k}(\alpha_k \eta_k). \tag{2.24}$$

Note that with the canonical notations, (2.23) is written as $x_{k+1} = \text{Exp}_{x_k}(\alpha_k \eta_k)$. So (2.24) is very similar to (2.23) except the fact that it uses a general path. The *retraction* R_x is formally defined with a local rigidity and it preserves gradients at x.

Definition 2.10. *(Absil et al., 2009) A retraction on a manifold $\mathcal{M}$ is a smooth mapping R from the tangent bundle $T\mathcal{M}$ onto $\mathcal{M}$ with the following properties. Let R_x denote the restriction of R to $T_x\mathcal{M}$.*

- *$R_x(0_x) = x$, where 0_x denotes the zero element of $T_x\mathcal{M}$.*
- *With the canonical identification $T_{0_x}T_x\mathcal{M} \simeq T_x\mathcal{M}$, R_x satisfies*

$$DR_x(0_x) = id_{T_x\mathcal{M}},$$

*where $id_{T_x\mathcal{M}}$ denotes the identify mapping on $T_x\mathcal{M}$ and this is called **local rigidity condition**.*

Here is an example to compare a retraction and exponential map. The unit sphere S^{n-1} can be considered a Riemmanian manifold embedded in

$\mathbf{R}^n$. A retraction is given by

$$R_x(v) = \frac{x+v}{\|x+v\|} \tag{2.25}$$

where $x \in S^{n-1} \subset \mathbf{R}^n$ and $v \in T_x\mathcal{M}$. This addition is performed in the ambient space ($\mathbf{R}^n$). The exponential map in S^{n-1} is given as

$$\mathrm{Exp}(x,v) = x\cos(\|v\|) + \frac{v}{\|v\|}\sin(\|v\|). \tag{2.26}$$

Retractions are often preferred since they have a lower computational cost.

We discussed the mappings to search along a path on manifolds. Now, we discuss how to choose the direction η_k in (2.24) and (2.23) on manifolds. The search direction can be a gradient or a descent direction. We define such directions on manifolds. Let us revisit the directional derivative in the Euclidean space. It is given by

$$Df(x)[\eta] = \lim_{t\to 0} \frac{f(x+t\eta) - f(x)}{t} \tag{2.27}$$

where f is a real function. On manifolds, the directional derivative can be defined as the following.

Definition 2.11. *(Boumal, 2014) (directional derivative) The directional derivative of a scalar field f on $\mathcal{M}$ at $x \in \mathcal{M}$ in the direction $v \in T_x\mathcal{M}$ is the scalar:*

$$Df(x)[v] := \frac{d}{dt} f(c(t))\Big|_{t=0} = (f\circ c)'(0), \tag{2.28}$$

where $c(t)$ is a smooth curve on $\mathcal{M}$ and $\circ$ denotes the function composition.

Now, with the definition of directional derivative on manifolds, we can define the gradient on manifolds as the following.

Definition 2.12. *(Boumal, 2014) (gradient) Let f be a scalar function defined on a Riemannian manifold $\mathcal{M}$. The gradient of f at $x \in \mathcal{M}$ denoted by $grad f(x)$, is defined as the unique element of $T_x\mathcal{M}$ satisfying:*

$$Df(x)[v] = \langle grad f(x), v \rangle_x, \forall v \in T_x\mathcal{M}.$$

Since f maps a scalar value at each $x \in \mathcal{M}$, f is a *scalar field* defined on $\mathcal{M}$. Similarly, grad$f : \mathcal{M} \to T\mathcal{M}$ is a *vector field* on $\mathcal{M}$. Note that the grad$f$ depends on the Riemannian metric $\langle , \rangle_x$.

Algorithm 2 Line search minimization on manifolds.

1: Input f, R_x, $k = 0, x_0 \in \mathcal{M}$,
2: **while** until convergence **do**
3: choose a descent direction or gradient $v_k \in T_{x_k}\mathcal{M}$
4: choose a step length $a_k \in \mathbf{R}_+$
5: $x_{k+1} = R_{x_k}(a_k v_k)$
6: $k \leftarrow k + 1$
7: **end while**

In Alg. 2, the update step with a retraction in line 5 can be replaced with the update step with the exponential map as $x_{k+1} = \text{Exp}(x_k, a_k v_k)$. For models in the thesis, we will use a variant of Alg. 2 for accurate iterative methods. For faster estimation, we adopted log-Euclidean framework as well for approximation in the tangent space.

What if we have decision variables in both a manifold and its tangent space?

$$\min_{x \in \mathcal{M}, v \in T_x\mathcal{M}} f(x, v)$$

We will see this case in the thesis. In this case, to have tangent vector v in the tangent space at x, we need an extra step to ensure that the tangent vector is in the right tangent space during updates. The step is obtained by *parallel transport*, which is a generalization of parallel translation. Detailed

algorithms will be introduced in Chapter 3.

3 MANIFOLD-VALUED MULTIVARIATE GENERAL LINEAR MODELS (MMGLMS)

A general linear model (GLM) is widely used in many scientific disciplines since it is simple, robust to noise and easy to estimate. Despite the simplicity of the model, it may not be universally applicable. In modern image analysis, the response variables in many applications live on Riemannian manifolds where standard GLM is not directly applicable because of the absence of an additive structure. Some simple solutions (and still widely used approaches) are to perform standard GLM with scalar summaries of manifold-valued data or to run multivariate regressors imposing the Euclidean topology forcibly. However, sometimes these schemes can lead to poor estimation due to the coarse description of measurements and inaccurate distance metrics. To address this problem, we study a Manifold-valued Multivariate General Linear Models (MMGLMs) regressing a manifold-valued dependent variable against multiple independent variables, i.e., $f : \mathbf{R}^n \rightarrow \mathcal{M}$. In our neuroimaging experiments, our proposed methods improved statistical power in the statistical analysis of diffusion weighted images, based on both diffusion tensors for DTI and Orientation Distribution Functions (ODFs) from High Angular Resolution Diffusion-weighted Imaging (HARDI).

3.1 General linear model in Euclidean spaces

Regression is essential in scientific analysis to identify how a dependent variable, $y \in \mathcal{Y}$ relates to an independent variable, $x \in \mathcal{X}$. Here, we are provided training data in the form, $(x_i \in \mathcal{X}, y_i \in \mathcal{Y})_{i=1}^N \subset \mathcal{X} \times \mathcal{Y}$, and seek to find the best model that explains these observations, given an appropriate loss function. The classical setting typically makes an

assumption on the geometric structure of the data by capturing the notion of distance between points a and b by the expression, $\left(\sum_{j=1\ldots n}(a^j - b^j)^2\right)^{\frac{1}{2}}$, which holds whenever the data are real vectors. In the Euclidean setting, a simple parametric model, $y_i = \alpha + \beta x_i + \epsilon_i$, suffices to identify the linear relationship between scalar-valued $x_i \in \mathcal{X}$ and the dependent (i.e., response) variable $y_i \in \mathcal{Y}$ with error ϵ_i. The least squares estimate is,

$$(\hat{\alpha}, \hat{\beta}) = \arg\min_{(\alpha,\beta)} \sum_{i=1}^{N} \|y_i - \alpha - x_i\beta\|^2, \tag{3.1}$$

and the closed form solution to (3.1) is obtained as,

$$\hat{\beta} = \frac{\mathrm{Cov}[x, y]}{\mathrm{Var}[x]} = \frac{\mathbb{E}\left[(x - \bar{x})(y - \bar{y})\right]}{\mathbb{E}\left[(x - \bar{x})^2\right]}, \quad \hat{\alpha} = \bar{y} - \hat{\beta}\bar{x}. \tag{3.2}$$

If x, and y are multivariate variables, one can easily replace the multiplication and division operations with an outer product of vectors and matrix inversion respectively, and obtain an analytical solution.

General linear models (GLMs) have been used as a core module in a standard analysis of structural/functional MRI analysis since it is a simple model and requires a small number of samples. GLM is available in popular neuroimaging analysis softwares such as FSL (Jenkinson et al., 2012), and SPM (Friston et al., 2007). A GLM is given by

$$y = \alpha + \beta^1 x^1 + \beta^2 x^2 + \ldots + \beta^n x^n + \epsilon \tag{3.3}$$

where α, x^i, β^i and the error ϵ are in $\mathbf{R}$ and the superscript i of x^i denotes the coordinate (or dimension) not the power of x. One common goal of neuroimaging analysis is to identify some regions of a brain that are associated with a disease or risk factors (e.g., age, gender, and phenotypes). To do so, since all brains have different sizes and shapes, we first register brains in a standard space. This is also called spatial normalization. Now,

we can safely map the anatomically same location across all subjects. After the registration, we perform a regression (or GLM) at each voxel with covariates (e.g., group information, risk factors) and a response (e.g., a voxel value in the image). This is called voxel-wise analysis.

We now identify some brain regions that are significantly correlated with covariates by evaluating the quality of the estimated regression coefficients. This is called analysis-of-variance (ANOVA). For example, let us assume that we learned $y = \beta_0 + \beta_1 x$. Then, we can identify the brain regions by testing the following hypothesis

$$H_0 : \beta_1 = 0 \text{ vs. } H_1 : \beta_1 \neq 0, \tag{3.4}$$

which means that the null hypothesis says the response variable y is independent from covariates x. To test the hypothesis above, we compute

$$F = \frac{SSR/1}{SSE/(n-2)}, \tag{3.5}$$

$$\text{where } SSR := \sum_{i=1}^{N} (\hat{y}_i - \bar{y}_i)^2 \tag{3.6}$$

$$SSE := \sum_{i=1}^{N} (y_i - \hat{y}_i)^2. \tag{3.7}$$

Under the conditions of the null hypothesis, regression sum of squares (SSR) and error sum of squares (SSE) of (3.4) have 1 and $N-2$ degree of freedom[1]. So, we reject H_0 at the α-level of significance when $F > f_\alpha(1, n-2)$ and we conclude that there is a significant amount of variation in the response accounted for by the model. The particular voxel of brains is correlated with the covariate. This is called F-test, or F-ratio test. For

[1]The degree of freedom (df) in statistics is the number of values that vary freely in the final calculation of a statistic, e.g., the sample variance has $N-1$ df, since it is calculated from N random scores minus the one parameter (the sample mean) estimated in an intermediate step.

more details and other hypothesis tests, we refer to (Larsen et al., 1986; Wackerly et al., 2007; Montgomery et al., 2015; Walpole et al., 2016).

To perform the same analysis with manifold-valued response variables, we will extend the linear model to Riemannian manifolds and provide the analogous hypothesis test with multiple nuisance variables.

3.2 Related work: geodesic regression for a univariate covariate

The basic geodesic regression model in (Fletcher, 2013) extends a special case of linear regressions to the Riemannian manifold setting. It is given as

$$y = \mathrm{Exp}(\mathrm{Exp}(p, xv), \epsilon), \tag{3.8}$$

where ϵ is the error (a tangent vector), $x \in \mathbf{R}$ and $y \in \mathcal{M}$ are the independent and dependent variables respectively, $p \in \mathcal{M}$ is a 'base' point on the manifold, and $v \in T_p\mathcal{M}$ is a tangent vector at p. For consistency with Euclidean space, we use m for the dimensionality of $T_p\mathcal{M}$. Comparing (3.8) and Table 2.1, observe that p and v correspond to the intercept α and the slope β in (3.1). Given N pairs of the form (x_i, y_i), (Fletcher, 2013) solves for $(p, v) \in T\mathcal{M}$ to fit *one* geodesic curve to the data,

$$E(p, v) := \frac{1}{2} \sum_{i=1}^{N} d(\mathrm{Exp}(p, x_i v), y_i)^2, \tag{3.9}$$

providing the estimate $\hat{y}_i = \mathrm{Exp}(p, x_i v)$. Here, errors are measured by the geodesic distance on $\mathcal{M}$, i.e., $d(a, b) = \sqrt{\langle \mathrm{Log}(a, b), \mathrm{Log}(a, b) \rangle_a}$. Rewriting (3.9) in the form of a minimization problem using the definition of

$d(\cdot,\cdot)$, we obtain

$$\min_{(p,v)\in T\mathcal{M}} \quad \frac{1}{2}\sum_i \langle \mathrm{Log}(\hat{y}_i, y_i), \mathrm{Log}(\hat{y}_i, y_i)\rangle_{\hat{y}_i} \tag{3.10}$$

To minimize E, one first needs to specify $\nabla_p E(p,v)$ and $\nabla_v E(p,v)$. This requires computing derivatives of the exponential map with respect to p and v. The gradients are derived in terms of *Jacobi fields* (which are solutions to a second order equation subject to certain initial conditions (Fletcher, 2013)) or via introducing small perturbations (Du et al., 2013). To express this in a computable form, we need to find the so-called *adjoint derivative*. The adjoint of an operator is defined as the operator $T^\dagger$ such that $\langle Tu, v\rangle = \langle u, T^\dagger v\rangle$. The adjoint derivative is the adjoint operator of a derivative. In other words, let $d_p\mathrm{Exp}(p,v)$ be the derivative of the exponential map with respect to p. Then, its *adjoint derivative* is the operator $d_p\mathrm{Exp}(p,v)^\dagger$ such that $\langle d_p\mathrm{Exp}(p,v)u, w\rangle = \langle u, d_p\mathrm{Exp}(p,v)^\dagger w\rangle$, where w is a tangent vector. Putting these pieces together, the gradient descent scheme (Fletcher, 2013) can optimize (3.10) in a numerically stable manner and obtain the estimates of p and v.

Can we extend this idea to multiple linear regression? Given the precise form of the scheme above, it is natural to ask if a similar idea will work for the MMGLM. It turns out that there are certain conceptual and technical difficulties in attempting such an extension. Observe that geodesic regression in (3.9) works on a scalar independent variable in $\mathbf{R}$ (and thereby, a single geodesic). For multiple linear regression, one must invariably make use of a subspace instead. It is easy to see that a multiple linear model $y = B\boldsymbol{x}$ seeks a subspace since all $(\boldsymbol{x}, y)$ form a subspace. On manifolds, a subspace can be generalized as a geodesic submanifold (or geodesic subspace) used in dimensionality reduction methods on manifolds (Fletcher et al., 2004; Sommer et al., 2014b; Huckemann et al., 2010a) and Riemannian CCA in Chapter 4.2. It is defined as $S = \mathrm{Exp}(p, \mathrm{span}(\{v_i\}) \cup U)$,

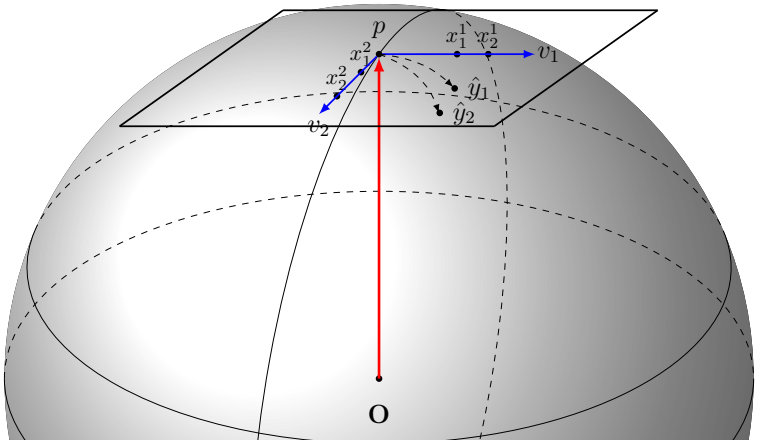


Figure 3.1: Manifold-valued multivariate general linear model (MMGLM). v^1, v^2 are tangent vectors. Each entry of independent variables $(x^1, x^2) \in \mathbf{R}^2$, is multiplied by v_1 and v_2 respectively in $T_p\mathcal{M}$. Here, x_i^j denotes j-th entry of the i-th instance.

where $U \subset T_p\mathcal{M}$, and $v_i \in T_p\mathcal{M}$. So, a linear regression from $\mathbf{R}^n$ to $\mathcal{M}$ (MMGLM) can be learned by searching a geodesic subspace in the product space $\mathbf{R}^n \times \mathcal{M}$. Alternatively, a MMGLM can be estimated by identifying multiple geodesic curves which correspond to 'basis' vectors in Euclidean space, see Fig. 3.1.

Let us now look at some of the technical difficulties by writing down the form of the gradients in geodesic regression.

$$\underbrace{-\sum_{i=1}^{N} d_p\mathrm{Exp}(p, x_i v)^\dagger \varpi_{y_i}}_{\nabla_p E(p,v)}, \text{ and } \underbrace{-\sum_{i=1}^{N} x_i d_v\mathrm{Exp}(p, x_i v)^\dagger \varpi_{y_i}}_{\nabla_v E(p,v)}$$

where $\varpi_{y_i} = \mathrm{Log}(\mathrm{Exp}(p, x_i v), y_i)$ and $d_p\mathrm{Exp}(p, x_i v)^\dagger$ is the adjoint derivative (Fletcher, 2013). The derivative of the exponential map, $d\mathrm{Exp}(p, v)$, is obtained by Jacobi fields *along* a geodesic curve parameterized by a tangent vector v. Here, this idea works because the prediction is a single geodesic curve. In multiple linear regression, predictions do not corre-

spond to one geodesic curve; so, expressing the corresponding tangent vector is problematic. Next, a key property of the adjoint derivative above is that the result of applying the operator, $d(p,v)^\dagger$, on w should lie in a tangent space. But for manifolds like $GL(n)/O(n)$, the tangent space corresponds to symmetric matrices (Cheng and Vemuri, 2013). This requires designing a special adjoint operator which guarantees this property, which is not trivial.

3.3 Manifold-valued Multivariate General linear models (MMGLMs)

Let x and y be vectors in $\mathbf{R}^n$ and $\mathbf{R}^m$ respectively. The multivariate multilinear model in Euclidean space is.

$$y = \alpha + \beta^1 x^1 + \beta^2 x^2 + \ldots + \beta^n x^n + \epsilon \tag{3.11}$$

where α, β^i and the error ϵ are in $\mathbf{R}^m$ and $x = [x^1 \ldots x^n]^T$ are the independent variables. Just as (3.11) uses a vector for each independent variable, MMGLMs use a geodesic basis, which corresponds to multiple tangent vectors, one for each independent random variable. Our formulation with multiple geodesic bases is

$$\min_{p\in\mathcal{M},\forall j, v^j\in T_p\mathcal{M}} \quad \frac{1}{2}\sum_{i=1}^{N} d(\mathrm{Exp}(p, \sum_{j=1}^{n} v^j x_i^j), y_i)^2, \tag{3.12}$$

where $Vx_i := \sum_{j=1}^{n} v^j x_i^j$.

Variational gradient descent method for MMGLMs. To address the technical issues pertaining to the adjoint derivatives which are needed for geodesic regression with a univariate covariate (Fletcher, 2013), we will attempt to obtain a similar effect to that operator, but via different

means. First, observe that in the geodesic regression, to enable summing up the individual terms $d(p, x_i v)^\dagger \text{Log}(\text{Exp}(p, x_i v), y_i)$'s which give the gradient, ∇E, a necessary condition is that these terms should lie in $T_p\mathcal{M}$. Here, $\text{Exp}(p, x_i v)$ gives the predicted $\hat{y}_i$ for y_i, and so $\text{Log}(\text{Exp}(p, x_i v), y_i)$ is the error and must lie in $T_{\text{Exp}(p,x_i v)}\mathcal{M}$, i.e., $T_{\hat{y}_i}\mathcal{M}$. By this argument, $d(p, x_i v)^\dagger$ actually plays a role of parallel transport to bring each error $\text{Log}(\text{Exp}(p, x_i v), y_i)$ from $T_{\hat{y}_i}\mathcal{M}$ to $T_p\mathcal{M}$. Since we hope to avoid constructing a special adjoint operator, we will instead perform parallel transport explicitly and derive approximate gradient terms, as outlined below.

Let us consider a slight variation of the objective function in (3.12). Let $\Gamma_{p\to q}(w)$ be a parallel transport of w from $T_p\mathcal{M}$ to $T_q\mathcal{M}$. Recall that parallel transport does *not* change the norm of tangent vectors, so the measurement of an error vector remains accurate. This ensures that the following modification preserves equivalence

$$\begin{aligned} E(p, v) &= \sum_i \langle \text{Log}(\hat{y}_i, y_i), \text{Log}(\hat{y}_i, y_i) \rangle_{\hat{y}_i} \\ &= \sum_i \langle \Gamma_{\hat{y}_i \to p} \text{Log}(\hat{y}_i, y_i), \Gamma_{\hat{y}_i \to p} \text{Log}(\hat{y}_i, y_i) \rangle_p, \end{aligned} \tag{3.13}$$

where $\hat{y}_i = \text{Exp}(p, \sum_j x_i^j v^j)$. Comparing two objective functions in (3.13), we see that in the first objective function, the inner product occurs at each tangent space $T_{\hat{y}_i}\mathcal{M}$, whereas the second objective function in (3.13) expresses all inner products in a tangent space $T_p\mathcal{M}$, *after* applying a parallel transport. For an object u on a manifold, let $u^\wr$ denote the corresponding object in tangent space of u at $T_p\mathcal{M}$. To derive our gradient expression, we will use the model in the tangent space as a temporary placeholder, to keep notations simple. Let us first define a few useful terms. Below, E is the error from (3.9) and $E^\wr$ gives the Log-Euclidean error in $T_p\mathcal{M}$. Let $p^\wr := \text{Log}(p, p)$ and $y_i^\wr := \text{Log}(p, y_i)$. We are searching

for not only tangent vectors but also the tangent space itself. To do so, by introducing a zero-norm tangent vector $p^\wr$ which corresponds to the origin of the tangent space $T_p\mathcal{M}$, the direction to move the origin is obtained. So, the estimate, $\hat{y}_i^\wr$ is $\text{Log}(p, \hat{y}_i) = \text{Log}(p, \text{Exp}(p, Vx_i)) = Vx_i + p^\wr$, where $V = [v_1 \dots v_n]$ is a m-by-n matrix, v^j is the m-dimensional tangent vector. The model in tangent space $T_p\mathcal{M}$ is given as

$$E^\wr(p^\wr, v) = \min_{p,v} \sum_i \|(\sum_j v^j x_i^j + p^\wr) - y_i^\wr\|^2 \tag{3.14}$$

Its gradient is expressed as

$$\nabla_{p^\wr} E^\wr = \sum_i (\hat{y}_i^\wr - y_i^\wr), \quad \nabla_{v^j} E^\wr = \sum_i x_i^j (\hat{y}_i^\wr - y_i^\wr), \tag{3.15}$$

where $\hat{y}_i^\wr = \sum_j v^j x_i^j + p^\wr$. Note that this gradient is the 'approximate' gradient in the linearized (tangent) space. Of course, we are actually interested in minimizing the parallel transported error on the manifold. So, we will substitute the parallel transported form for the linearized expression, $(\hat{y}_i^\wr - y_i^\wr)$ in (3.15) above and obtain,

$$\nabla_p E \approx -\sum_i \Gamma_{\hat{y}_i \to p} \omega_{y_i}, \quad \nabla_{v^j} E \approx -\sum_i x_i^j \Gamma_{\hat{y}_i \to p} \omega_{y_i}, \tag{3.16}$$

where $\omega_{y_i} = \text{Log}(\hat{y}_i, y_i)$ and $\hat{y}_i = \text{Exp}(p, Vx_i)$. Using this gradient and a line search algorithm on manifolds (Absil et al., 2009), the variational gradient descent scheme can optimize (3.12).

Remarks. Consider the Euclidean setting where x_i and x_j are large. The optimal intercept, p^*, will be far from y_i and y_j which is not a problem since we can explicitly solve for any value for the intercept. However, parametric models for Riemannian manifolds are based on the assumption that data are distributed in a sufficiently small neighborhood where exponential and logarithm maps are well-defined. In addition, x should

not have "large" entries (relative to the variance of x) otherwise p^* might be too far from the data and there is no well-defined exponential map to represent $y = \text{Exp}(p^*, Vx)$. Thus, we may explicitly solve for a parameter to translate the x variables, $y = \text{Exp}(p, V(x-b))$ where $b \in \mathbf{R}^n$. However, it may lead to many local minima. A simple fix to this problem is to first "center" the x variables which makes the optimization scheme stable (see pseudocode in Alg. 3).

Log-Euclidean framework for MMGLMs. We here outline an approximate algorithm that is simpler and offers more flexibility in analysis at the cost of a few empirically derived assumptions. To motivate the formulation, let us take a manifold perspective of (3.2): we see that analytical solutions can be obtained using the difference of each point from its mean both in $\mathcal{X}$ and $\mathcal{Y}$ space — that is, the quantities $\bar{x}\overrightarrow{x_i}$ and $\bar{y}\overrightarrow{y_i}$ calculated in the tangent space, $T_{p^*}\mathcal{M}$. Note that in (3.2), β corresponds to tangent vectors and α corresponds to p^*. Our scheme in (3.12) explicitly searches for p^*, but in experiments, we found that p^* frequently turns out to be quite close to $\bar{y}$. This observation yields a heuristic where rather than solve for p, we operate entirely in $T_{\bar{y}}\mathcal{M}$. With this assumption, using the Karcher mean as $\bar{y}$ in (3.2) and the Log-Euclidean distance as a substitute for $\bar{x}\overrightarrow{x_i}$

Algorithm 3 Iterative method for MMGLM

Input: $x_1, \ldots, x_N \in \mathbf{R}^n$, $y_1, \ldots, y_N \in \mathcal{M}$
Output: $p \in \mathcal{M}$, $v^1, \ldots, v^n \in T_p\mathcal{M}$,

Initialize $p, v, \alpha, \alpha_{max}$ and center x
while termination condition **do**
  $p_{new} = \text{Exp}(p, -\alpha\nabla_p E)$
  $V_{new} = \Gamma_{p \to p_{new}}(V - \alpha\nabla_V E)$
  if $E(p_{new}, V_{new}) < E(p, V)$ **then**
    $V \leftarrow V_{new}$ and $P \leftarrow P_{new}$
    $\alpha = \min(2\alpha, \alpha_{max})$
  else
    $\alpha = \alpha/2$
  end if
end while

and $\bar{y}\overrightarrow{y_i}$, we can derive a faster approximate procedure. This scheme has the additional benefit that it allows analyzing multiple manifold-valued *independent* variables, i.e., $f : \mathcal{M} \to \mathcal{M}'$, if desired.

The Log-Euclidean MMGLM estimates a linear relationship between centered variables $\{x_i^\wr\}_{i=1}^N$ and $\{y_i^\wr\}_{i=1}^N$ where $x_i^\wr = x_i - \bar{x}$ and $y_i^\wr = \text{Log}(\bar{y}, y_i)$, where the number of tangent vectors we estimate is exactly equal to the number of independent variables, x. Our basic procedure estimates the set of vectors $V = [v_1 \dots v_n]$ in tangent space $T_{\bar{y}}\mathcal{M}$ and p a point on $\mathcal{M}$ using the relation $Y^\wr = VX^\wr$. $Y^\wr \equiv [y_1^\wr \dots y_N^\wr]$ and $X^\wr \equiv [x_1^\wr \dots x_N^\wr]$ are respectively the mean centered data. Here, p^* is given by the Karcher mean $\bar{y}$. The target V^* is given by the least squares estimation with respect to the Log-Euclidean metric and can be computed using the closed form solution, $Y^\wr X^{\wr T} \left(X^\wr X^{\wr T}\right)^{-1}$.

The following analysis shows that under some conditions, heuristically substituting $\bar{y}$ for p^* is justifiable beyond empirical arguments alone. In particular, if the y observations come from *some* geodesic curve, then all of the data can be parameterized by one tangent vector in the tangent space at $\bar{y}$. More specifically, we show that a Karcher mean exists on a geodesic curve. Therefore, if the Karcher mean is unique, then the Karcher mean must lie on the geodesic curve. By the definition of the exponential map and since the data are in sufficiently small neighborhood, it becomes possible to parameterize the observations by $\text{Exp}(\bar{y}, vx)$.

Prop. 1 shows the existence of the Karcher mean on a geodesic curve when the data lies on the unique geodesic curve, Ω, between two points.

Proposition 1. *Let $Y = \{y_1, \dots, y_N\}$ be a subset of a manifold $\mathcal{M}$. Suppose that Y is in a sufficiently small open cover $\mathcal{B}$ such that the exponential and logarithm maps are bijections. Suppose that all $y \in Y$ are on a curve Ω that is the unique geodesic curve between some y_i and y_j in Y. Then there exists $\bar{y}$ in Ω such that $\sum_{y \in Y} Log_{\bar{y}} y = 0$ (the first order condition for Karcher mean).*

Proof. Let $v \in T_{y_i}\mathcal{M}$ be the tangent vector $v = \text{Log}_{y_i} y_j$. Since all points

of Y are a subset of a geodesic curve Ω between y_i, and y_j, for each $y_k \in Y$, there exists an $x_k \in [0,1]$ such that $y_k = \text{Exp}(y_i, vx_k)$. So, let $\bar{x} = \frac{1}{N}\sum_{k=1}^{N} x_k$ and $\bar{y} = \text{Exp}(y_i, v\bar{x})$. Then, $\bar{y}$ satisfies $\sum_k \text{Log}_{\bar{y}} y_k = 0$ and it is in Ω since the arithmetic mean $\bar{x}$ is in $[0,1]$. □

With this result in hand, we next show that the data can be parameterized by V.

Proposition 2. *If $\bar{y}$ is the unique Karcher mean of $Y \subset \Omega$, and it is obtained in $\mathcal{B}$, then $\bar{y} \in \Omega$. Further, for some $v \in T_{\bar{y}}\mathcal{M}$ and each y, there exists $x \in \boldsymbol{R}$ such that $y = Exp(\bar{y}, vx)$.*

Proof. If $\bar{y}$ is a unique Karcher mean of Y on $\mathcal{M}$ and it is obtained in a sufficiently small neighborhood $\mathcal{B} \subset \mathcal{M}$ of data, then $\sum_{k=1}^{N} \text{Log}_{\bar{y}} y_k = 0$ holds by the first order optimality condition of (2.9). The uniqueness of $\bar{y}$ and Prop. 1 imply that $\bar{y}$ is in Ω. In a small neighborhood, by definition of exponential map, there must exist an appropriate x. □

3.4 Experimental results

Here, we show the application of our models for statistical analysis of diffusion weighted imaging (DWI) data. Most neuroimaging studies acquire DWI data to perform statistical analysis. Assume that the images are already registered to a common template. The scientific question may be to identify which regions of the brain *vary* across two groups of subjects: diseased and healthy. This can be answered by performing a hypothesis test at each voxel over the entire brain, and reporting the statistically significant ones as different across groups. Separately, one may want to identify regions which have a strong relationship with disease status. Independent of the specific setting, the classical analysis makes use a scalar-valued summary measure at each voxel: fractional anisotropy FA for DTI or generalized fractional anisotropy GFA for HARDI (Tuch, 2004).

But this simplification, which makes the differential signal harder to detect, can be avoided. To do so, we used manifold-valued measurements that are richer representations and introduced in Chapter 2.4. In DTI, the diffusion tensors are represented as a point on a SPD(3) manifold (i.e., the quotient space $GL(3)/O(3)$). In HARDI, the square root parameterization of the ODF is represented as a point on a unit Hilbert sphere (S) (Cheng et al., 2009), which in practice, is expressed as a l-th order spherical harmonics: we use $l = 4$ implying the S^{14} setting. With the appropriate statistical models in hand, we may regress the manifold data directly against one or more independent variables. On synthetic simulations and real data analysis from two distinct neuroimaging studies, the experiments evaluate whether (and to what extent) general linear model (GLM) analysis on diffusion weighted images in neuroimaging can benefit from **(a)** the ability to deal with manifold-valued data and **(b)** allowing multiple explanatory (including nuisance) variables.

3.4.1 Synthetic setting

We first artificially generate ODF and DTI data via a generative multiple linear model. We then estimate using our MMGLM framework and the model in (Du et al., 2013) (certain adjustments to (Du et al., 2013) were needed for the SPD(3) manifold). The results in Fig. 3.2 give strong evidence that when the characteristics of the data depend on multiple independent variables (e.g., disease *and* age), MMGLM significantly outperforms (single) linear geodesic regression (SLGR) which regresses $y \in \mathcal{Y}$ against $x \in \mathbf{R}$. In Fig. 3.2, GR1 and GR2 refer to the estimates from SLGR using variables x^1 and x^2 individually. MMGLM is able to estimate the true signal far more accurately compared to both GR1 and GR2. Fig. 3.3 shows the quantitative results of regression using four independent variables as a function of sample sizes. As expected, we see that the fit improves significantly with MMGLM.

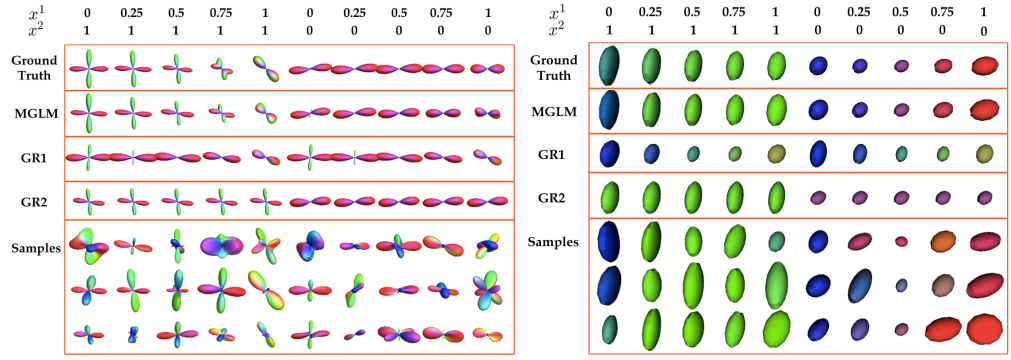


Figure 3.2: MMGLM and SLGR results using synthesized ODF and DTI. First two rows give the values for x^1 and x^2 of the generative model.

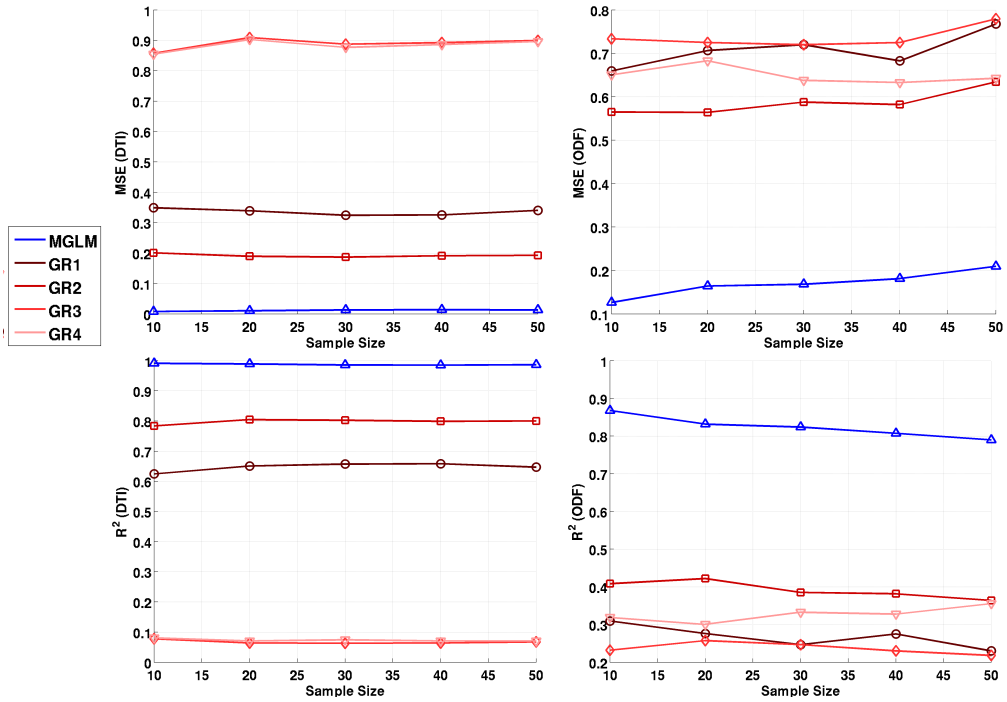


Figure 3.3: Plots showing the effect of sample size on mean squared error (MSE) and R^2 for the MMGLM as well as SLGR using the individual variables. GR$\{1, \cdots, 4\}$ refer to the estimates from SLGR using the individual variables $x^{\{1,\cdots,4\}}$ individually.

3.4.2 Neuroimaging data evaluations

We now present experiments using DWI data from two real neuroimaging studies. The first study investigates the neuroplasticity effects of meditation practice (e.g., for emotional well-being) on white matter. Meditators were trained in Buddhist meditation techniques, which lead to emotion regulation and attention control. An example scientific question here may be: what is the relationship between the number of years of meditation training and white matter when conditioned on age? Here, diffusion images in 48 non-collinear diffusion encoding directions were acquired, which after a sequence of pre-processing steps, provide the ODF representations for 23 long-term meditators (LTM) and 26 control (WLC) subjects. In the second study, we investigate the effect of a genotype risk factor (i.e., APOE4 positive or negative) in Alzheimer's disease (AD) on white matter integrity in the brain. A representative scientific question here may be: what is the effect of age on white matter when we control for genotype and gender? Here, 40 encoding directions were acquired and diffusion tensor images were obtained after pre-processing. The dataset covers 343 subjects (123 with APOE4+ and 220 with APOE4-).

DWI acquisition and processing. The data was acquired using a diffusion-weighted, spin-echo, single-shot, echo planar imaging radio-frequency (RF) pulse sequence. For the meditation study, diffusion data in 48 non-collinear diffusion encoding directions with diffusion weighting factor of $b = 1000s/mm^2$ and eight non-diffusion weighted ($b = 0$) images was acquired. For each of the 23 long-term meditators (LTM) and 26 control (WLC) subjects, ODFs were estimated and the square-root parameterization was obtained via linear spherical harmonic transform (Goh et al., 2011). For the AD-risk study, images with 40 encoding directions at $b = 1300s/mm^2$ and eight $b = 0$ images were acquired for each of the 343 subjects (123 with APOE4+ and 220 with APOE4-, where APOE denotes the Apolipoprotein E genotype). The diffusion tensors were estimated

from DWI data from both the studies, using non-linear estimation using the Camino library (Cook et al., 2006). The images for both studies were normalized spatially before statistical analysis using DTI-TK (Zhang et al., 2006a).

GLM results. We estimate the following model at each voxel for both studies,

$$\begin{aligned} \text{GLM}_{\text{Full}}: &\quad y = \text{Exp}(p, v^1\text{Group} + v^2\text{Gender} + v^3\text{Age}), \\ \text{GLM}_{\text{Age}}: &\quad y = \text{Exp}(p, v^2\text{Gender} + v^3\text{Age}), \\ \text{GLM}_{\text{Group}}: &\quad y = \text{Exp}(p, v^1\text{Group} + v^2\text{Gender}), \end{aligned} \tag{3.17}$$

where $y \in S^{14}$, Group $\in$ {LTM, WLC} for meditation study, and $y \in$ SPD(3), Group $\in$ {APOE4+, APOE4-} for AD-risk study.

As a baseline, we present regression results using FA as the measure of interest. We note that regressing y against *one* independent variable as in Fig. 3.3 is a possible baseline but because it is restricted, it cannot fit the full model in (3.17). Therefore, FA is a better baseline for comparisons. The null hypothesis, H_0 here is that the linear combination of 'group', 'gender' and 'age' has no effect on the y measurement. Therefore, (3.17) serves as the "full" model and the intercept alone serves as the nested model. Then, an F-statistic can yield voxel-wise p-value maps when we regress on FA. However, for manifold-valued variables, a parametric null distribution of F-statistics is not available. So, to obtain p-values, we use $20,000$ permutations to characterize the Null distribution of the R^2-fit. Then, the unpermuted R^2 is used to calculate the p-values. This is called the *permutation test*. Comparing the two p-values maps (FA vs. ODF) shows which procedure is successfully picking up more differential signal in a statistically sound manner. Fig. 3.4 shows the p-value maps, for FA and ODF based regression. We can observe the improved statistical sensitivity using the MMGLM framework.

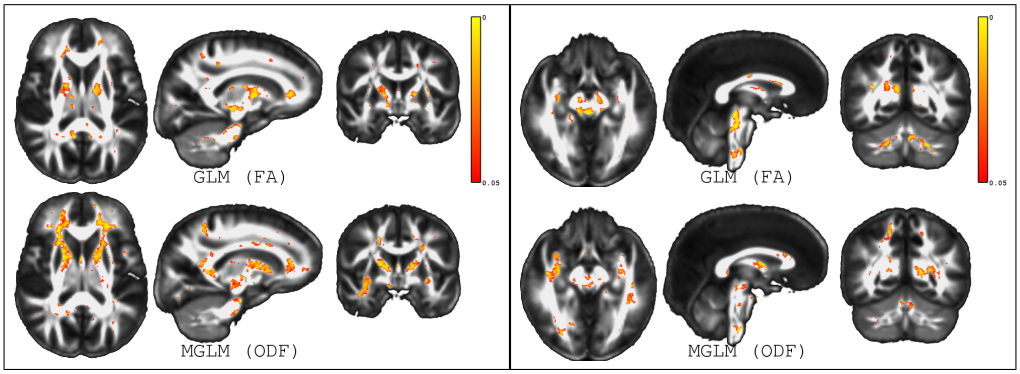


Figure 3.4: (Left panel) Uncorrected p-value maps obtained using FA (smoothed) based GLM as well as ODF based MGLM frameworks. The voxels that pass the threshold of $p \leq 0.05$ are spatially more contiguous when performing MGLM. (Right panel) The thresholded p-value maps showing that the spatial extents in the brain stem (axial-left and sagittal-middle) and cerebellar (coronal-right) white matter are reduced when using MGLM.

Fig. 3.5 shows p-value maps and histograms for GLM_{Full} for the AD-risk study. Figs. 3.6 and 3.8 show p-value maps and histograms for specific effect of Age and Group. For MMGLMs, we compute p-values by simulating the null distribution of the F ratio statistic again using 20,000 permutations. The F ratio statistic is defined for a pair of nested GLMs as,

$$F = \frac{\frac{\text{RSS}_1 - \text{RSS}_2}{p_2 - p_1}}{\frac{\text{RSS}_2}{N - p_2}}, \tag{3.18}$$

where $\text{RSS}_j = \sum_{i=1}^{N} \text{d}(\hat{y}_{ij}, y_i)^2$, $\hat{y}_{ij}$s are estimated using GLM_j and p_j is the number of independent parameters in GLM_j. For obtaining the effect of Age and Group, RSS_1 is obtained using GLM_{Age} and $\text{GLM}_{\text{Group}}$, respectively. RSS_2 is obtained using GLM_{Full} in both the cases. The corresponding maps and histograms are shown in respectively.

Main observations. In Fig. 3.5, we can observe that the thresholded regions are spatially more contiguous when using MMGLM in the sagittal

(middle) and coronal (right) slices compared to those obtained by performing exactly the same model on the univariate (smoothed) FA images (the unsmoothed FA results are much worse). Note that a smoothing procedure on the DTI data (i.e., the tensors) further improves the results but the purpose here is to show that even the unsmoothed DTI (with MMGLM) yields comparable (or better) results, as can be noticed in the histogram in the right panel in Fig. 3.5. We see the same behavior in Fig. 3.6. The left panel shows p-values obtained using the effect of age while the right panel shows the effect of group variable i.e., APOE4+ vs. APOE4-. Note that existing literature provides little guidance on algorithms for performing such GLM models on the DTI data directly, as we show here. In the case of age effect, MMGLM provides more spatially contiguous regions but in the analysis of APOE4 vs. APOE4-,FA smoothed and MMGLM are comparable, as can also be noted in Fig. 3.8. In Fig. 3.7, axial (left-column) and sagittal (middle-column) views in both panels show that our MMGLM using ODF provides the improved statistical power in the age effect (left panel) and meditation effect (right panel). Fig. 3.9 quantitatively shows that our MMGLM using ODF rejected the null hypothesis $(p < 0.05)$ at more voxels rather than GLM using GFA and GLM using (smoothed) GFA both when trying to detect the effect of age as well as the group. Our experiments support that MMGLM based analysis provides comparable or improved statistical power compared to the GLM based analysis.

3.5 Summary

This chapter extends multivariate general linear model (MGLM) to the manifold setting. Such an extension allows regressing a manifold valued dependent variable $y \in \mathcal{M}$ against multiple independent variables, $x \in \mathcal{X}$. This extends the range of applicability of existing methods and will allow practitioners to easily regress voxel measurements in diffusion

weighted imaging against clinical variables, while controlling for nuisance parameters, thereby obtaining results which better reflect hypotheses under study. The experiments give strong evidence of the improvements we may expect over traditional alternatives. The chapter is accompanied by an open source codebase [2], which will enable easy deployment in practice. For large scale analysis on Amazon Web Service[3] or HTCondor[4], our extended code is available as well.

[2] https://github.com/MLman/riem-mglm-cvpr2014
[3] https://github.com/MLman/MMGLMAWS
[4] https://github.com/MLman/MMGLM_HTCONDOR

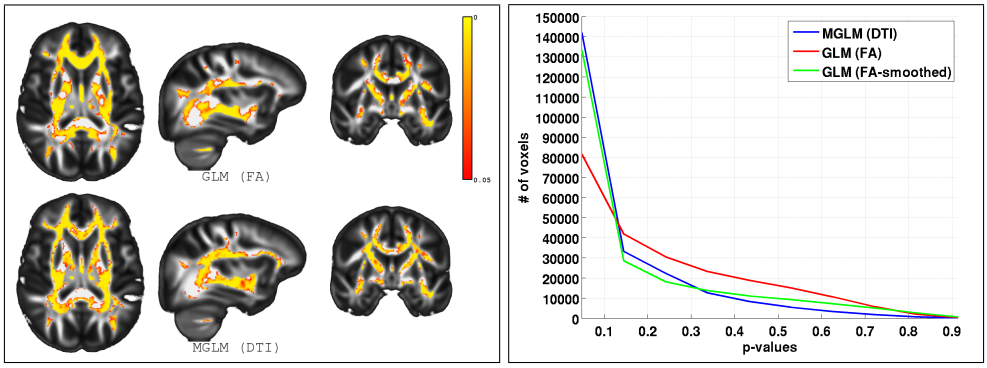


Figure 3.5: p-value maps obtained using FA (smoothed) based GLM as well as DTI based MMGLM frameworks. Left panel shows thresholded ($p \leq 0.05$) p-values obtained using the full model. We can observe that the thresholded regions are spatially more contiguous when using MMGLM. We can notice that more clearly in the sagittal (middle) and coronal (right) slices. Right panel shows distribution of p-values obtained using MMGLM using DTI and GLM using both smoothed and unsmoothed FA images.

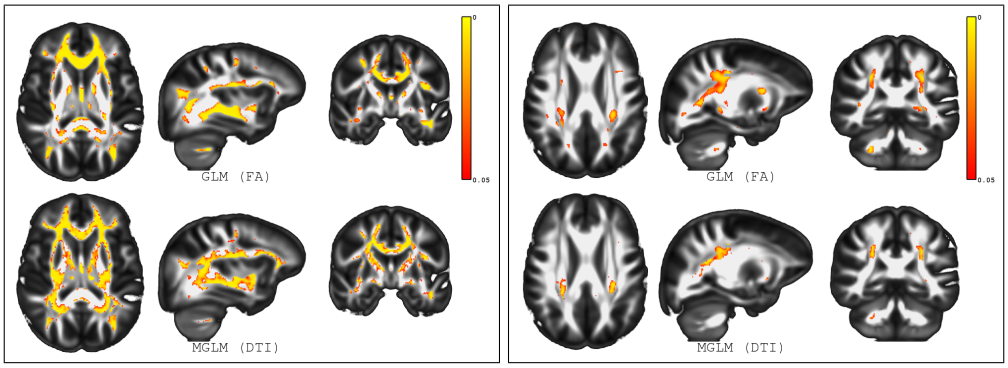


Figure 3.6: p-value maps obtained using FA (smoothed) based GLM as well as DTI based MMGLM frameworks. Left panel shows p-values obtained using the effect of age while the right panel shows the effect of group variable i.e. APOE4+ vs. APOE4-.

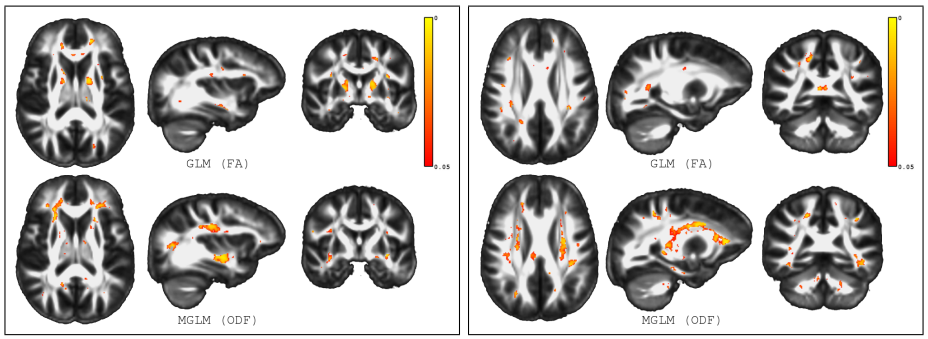


Figure 3.7: *p*-value maps obtained using FA (smoothed) based GLM as well as ODF based MMGLM frameworks. Left panel shows the effect of age while the right panel shows the effect of group variable i.e. LTM vs. WLC.

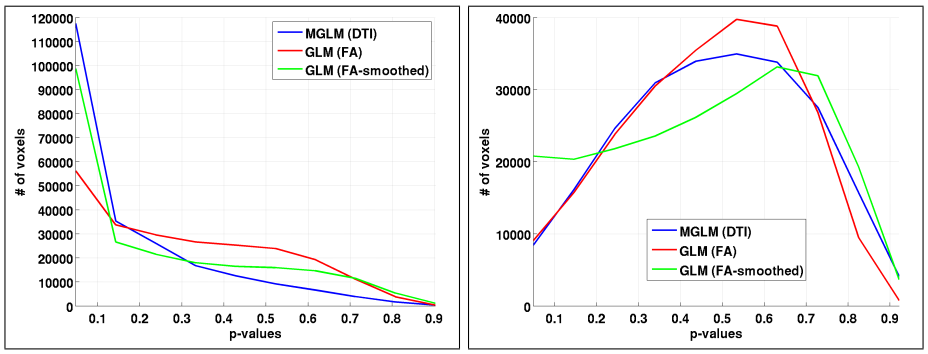


Figure 3.8: Distribution of *p*-values obtained using MMGLM using DTI and GLM using both smoothed and unsmoothed FA images. Left: Age effect. Right: APOE4 effect. As discussed in Fig. 3.6, in case of APOE4 effect the smoothed FA based GLM and DTI based MMGLM perform comparably.

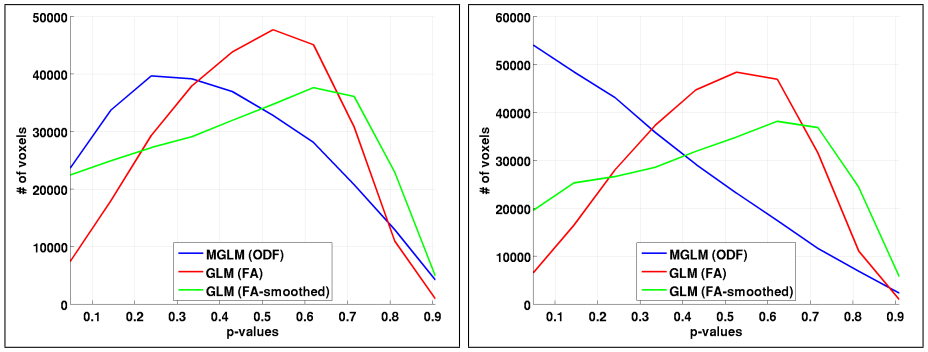


Figure 3.9: Distribution of *p*-values obtained using MMGLM using ODF and GLM using both smoothed and unsmoothed GFA images. Left: Age effect. Right: Group effect (LTM vs. WLC).

4 RIEMANNIAN CANONICAL CORRELATION ANALYSIS (RCCA)

The aim of this chapter is to generalize Canonical Correlation Analysis (CCA) to a manifold or the product of manifolds to identify meaningful correlations across two different modalities. Our formulation results in a multi-level optimization problem. We derive solutions using the first order condition of projection and an augmented Lagrangian method. In addition, we also develop an efficient algorithm with approximate projections. On the experimental side, we presented neuroimaging findings using our proposed CCA on Diffusion tensor images (DTI) and T1-weighted magnetic resonance images (T1W) on an Alzheimer's disease (AD) dataset focused on risk factors for this disease. SPD manifolds are used for diffusion tensors in DTI and Cauchy deformation tensor (CDT) introduced in Chapter 1.4.2. The CDTs are derived from T1W. Here, the proposed methods perform well and yield scientifically meaningful results.

4.1 Canonical Correlation in Euclidean Space

First, we will briefly review the classical CCA in the Euclidean space (Hotelling, 1936; Hardoon et al., 2004). Recall that Pearson's product-moment correlation coefficient is a quantity to measure the relationship of two random variables, $x \in \mathbf{R}$ and $y \in \mathbf{R}$. For one dimensional random variables,

$$\rho_{x,y} = \frac{\mathrm{COV}(x,y)}{\sigma_x \sigma_y} = \frac{\mathbb{E}[(x-\mu_x)(y-\mu_y)]}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{N}(x_i-\mu_x)(y_i-\mu_y)}{\sqrt{\sum_{i=1}^{N}(x_i-\mu_x)^2}\sqrt{\sum_{i=1}^{N}(y_i-\mu_y)^2}} \tag{4.1}$$

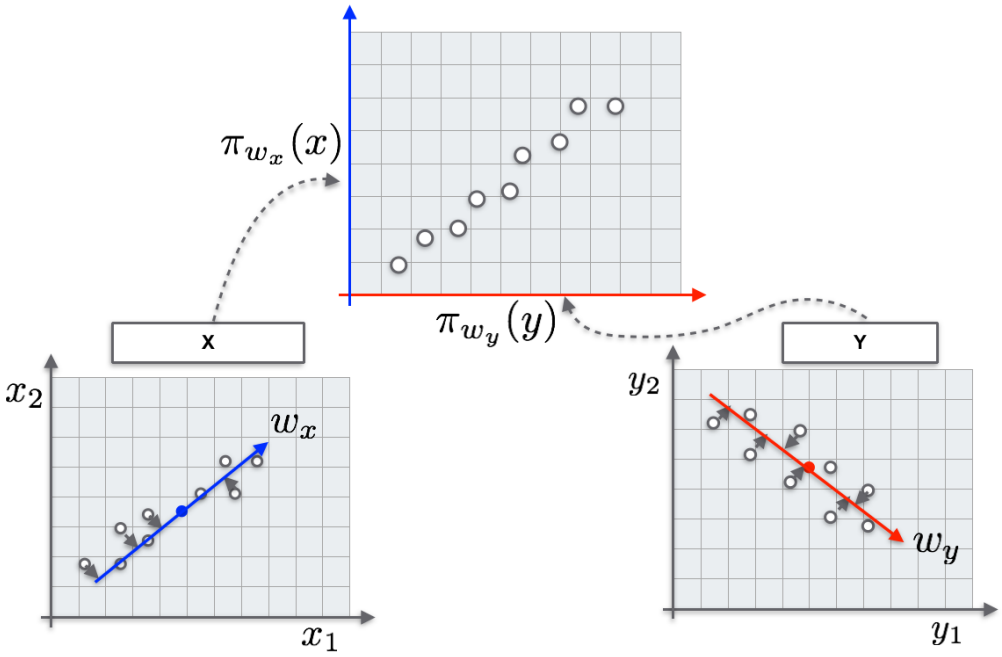


Figure 4.1: Canonical Correlation Analysis (CCA) in Euclidean space. CCA seeks the best subspaces to project data in each space to maximize the correlation between $\pi_{w_x}(X)$ and $\pi_{w_y}(Y)$.

where μ_x and μ_y are the means of $\{x_i\}_{i=1}^N$ and $\{y_i\}_{i=1}^N$. For high dimensional data, $\boldsymbol{x} \in \mathbf{R}^m$ and $\boldsymbol{y} \in \mathbf{R}^n$, we cannot however perform a direct calculation as above. So, we need to project each set of variables on to a special axis in each space $\mathcal{X}$ and $\mathcal{Y}$. CCA generalizes the concept of correlation to random vectors (potentially of different dimensions). It is convenient to think of CCA as a measure of correlation between two multivariate data based on the *best* projection which maximizes their mutual correlation.

Canonical correlation for $\boldsymbol{x} \in \mathbf{R}^m$ and $\boldsymbol{y} \in \mathbf{R}^n$ is given by

$$\begin{aligned} &\max_{\boldsymbol{w}_x, \boldsymbol{w}_y} \operatorname{corr}(\pi_{\boldsymbol{w}_x}(\boldsymbol{x}), \pi_{\boldsymbol{w}_y}(\boldsymbol{y})) \\ &= \max_{\boldsymbol{w}_x, \boldsymbol{w}_y} \frac{\sum_{i=1}^N \boldsymbol{w}_x^T(\boldsymbol{x}_i - \boldsymbol{\mu}_x)\boldsymbol{w}_y^T(\boldsymbol{y}_i - \boldsymbol{\mu}_y)}{\sqrt{\sum_{i=1}^N (\boldsymbol{w}_x^T(\boldsymbol{x}_i - \boldsymbol{\mu}_x))^2}\sqrt{\sum_{i=1}^N \left(\boldsymbol{w}_y^T(\boldsymbol{y}_i - \boldsymbol{\mu}_y)\right)^2}}. \end{aligned} \tag{4.2}$$

where $\pi_{\boldsymbol{w}_x}(\boldsymbol{x}) := \arg\min_{t \in \mathbb{R}} \mathrm{d}(t\boldsymbol{w}_x, \boldsymbol{x})^2$. We will call $\pi_{\boldsymbol{w}_x}(\boldsymbol{x})$ the *projection coefficient* for $\boldsymbol{x}$ (similarly for $\boldsymbol{y}$). Define $S_{\boldsymbol{w}_x}$ as the subspace which is the span of $\boldsymbol{w}_x$. The projection of $\boldsymbol{x}$ onto $S_{\boldsymbol{w}_x}$ is given by $\Pi_{S_{\boldsymbol{w}_x}}(\boldsymbol{x}) := \arg\min_{x' \in S_{\boldsymbol{w}_x}} \mathrm{d}(\boldsymbol{x}, \boldsymbol{x}')^2 = \frac{\boldsymbol{w}_x^T \boldsymbol{x}}{\|\boldsymbol{w}_x\|^2}\boldsymbol{w}_x = \pi_{\boldsymbol{w}_x}(\boldsymbol{x})\boldsymbol{w}_x$.

In the Euclidean space, $\Pi_{S_{\boldsymbol{w}_x}}(\boldsymbol{x})$ has a closed form solution. In fact, it is obtained by an inner product, $\boldsymbol{w}_x^T\boldsymbol{x}$. Hence, by replacing the projection coefficient $\pi_{\boldsymbol{w}_x}(\boldsymbol{x})$ with $\boldsymbol{w}_x^T\boldsymbol{x}/\|\boldsymbol{w}_x\|^2$ and after a simple calculation, one obtains the form in (4.2).

4.2 A Model for CCA on Riemannian Manifolds

We now present a step-by-step derivation of our Riemannian CCA model. Classical CCA finds the mean of each data modality. Then, it maximizes correlation between projected data on each subspace at the mean. Similarly, CCA on manifolds must first compute the intrinsic mean (i.e., Karcher mean) of each data set. It must then identify a 'generalized' version of a subspace at each Karcher mean to maximize the correlation of projected data. The generalized form of a subspace on Riemannian manifolds has been studied in the literature (Sommer et al., 2014b; Lebanon et al., 2005; Huckemann et al., 2010b; Fletcher et al., 2004). The so-called *geodesic submanifold* (Fletcher et al., 2004; Xie et al., 2010; Kim et al., 2014c) which has been used for geodesic regression serves our purpose well

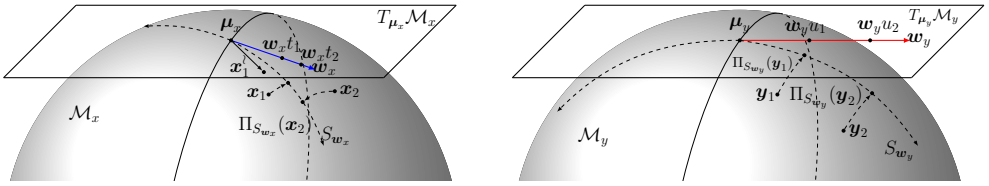


Figure 4.2: CCA on Riemannian manifolds. CCA searches geodesic submanifolds (subspaces), S_{w_x} and S_{w_y} at the Karcher mean of data on each manifold. Correlation between projected points $\{\Pi_{S_{w_x}}(\boldsymbol{x}_i)\}_{i=1}^N$ and $\{\Pi_{S_{w_y}}(\boldsymbol{y}_i)\}_{i=1}^N$ is equivalent to the correlation between *projection coefficients* $\{t_i\}_{i=1}^N$ and $\{u_i\}_{i=1}^N$. Although $\boldsymbol{x}$ and $\boldsymbol{y}$ belong to the same manifold we show them in different plots for ease of explanation.

and is defined as $S = \text{Exp}(\boldsymbol{\mu}, \text{span}(\{\boldsymbol{v}_i\}) \cap U)$, where $U \subset T_{\boldsymbol{\mu}}\mathcal{M}$, and $\boldsymbol{v}_i \in T_{\boldsymbol{\mu}}\mathcal{M}$ (Fletcher et al., 2004). When S has only one tangent vector $\boldsymbol{v}$, then the geodesic submanifold is simply a geodesic curve, e.g., S_{w_x} and S_{w_y} in Figure 4.2.

We can now proceed to formulate the precise form of projection on to a geodesic submanifold. Recall that when given a point, its projection on a set is the closest point in the set. So, the projection on to a geodesic submanifold (S) must be a function satisfying this behavior. This is given by,

$$\Pi_S(\boldsymbol{x}) = \arg\min_{\boldsymbol{x}' \in S} \text{d}(\boldsymbol{x}, \boldsymbol{x}')^2 \tag{4.3}$$

In Euclidean space, the projection on a convex set (e.g., subspace) is unique. It is also unique on some manifolds under special conditions, e.g., quaternion sphere (Said et al., 2007). However, the uniqueness of the projection on geodesic submanifolds in general conditions cannot be ensured. Like other methods (Fletcher et al., 2004; Huckemann et al., 2010a; Sommer et al., 2014a), we assume that given the specific manifold and the data, the projection is well-posed.

Finally, the correlation of points (*after* projection) can be measured by the distance from the mean to the projected points. To be specific, the

projection on a geodesic submanifold corresponding to $\boldsymbol{w}_x$ in classical CCA is given by

$$\Pi_{S_{w_x}}(\boldsymbol{x}) \;\; := \arg\min_{\boldsymbol{x}' \in S_{w_x}} \|\mathrm{Log}(\boldsymbol{x}, \boldsymbol{x}')\|_{\boldsymbol{x}}^2 \tag{4.4}$$

$S_{w_x} := \mathrm{Exp}(\boldsymbol{\mu}_x, \mathrm{span}\{\boldsymbol{w}_x\} \cap U)$ where $\boldsymbol{w}_x$ is a basis tangent vector and $U \subset T_{\boldsymbol{\mu}_x}\mathcal{M}_x$ is a small neighborhood of $\boldsymbol{\mu}_x$. The expression for *projection coefficients* can now be given as

$$t_i = \pi_{\boldsymbol{w}_x}(\boldsymbol{x}_i) \;\; := \arg\min_{t_i' \in (-\epsilon, \epsilon)} \|\mathrm{Log}(\mathrm{Exp}(\boldsymbol{\mu}_x, t_i' \boldsymbol{w}_x), \boldsymbol{x}_i)\|_{\boldsymbol{\mu}_x}^2 \tag{4.5}$$

where $\boldsymbol{x}_i, \boldsymbol{\mu}_x \in \mathcal{M}_x$, $\boldsymbol{w}_x \in T_{\boldsymbol{\mu}_x}\mathcal{M}_x, t_i \in \mathbf{R}$. The term, $u_i = \pi_{\boldsymbol{w}_y}(\boldsymbol{y})$ is defined analogously. t_i is a real value to obtain the point $\Pi_{S_{w_x}}(\boldsymbol{x}) = \mathrm{Exp}(\boldsymbol{\mu}_x, t_i \boldsymbol{w}_x)$.

Notice that the projection coefficient is proportional to the length of the geodesic curve from the base point $\boldsymbol{\mu}_x$ to the projection of $\boldsymbol{x}$, i.e., $\mathrm{d}(\boldsymbol{\mu}_x, \Pi_{S_{w_x}}(\boldsymbol{x}_i)) = \|\mathrm{Log}(\boldsymbol{\mu}_x, \mathrm{Exp}(\boldsymbol{\mu}_x, \boldsymbol{w}_x t_i))\|_{\boldsymbol{\mu}_x} = t_i \|\boldsymbol{w}_x\|_{\boldsymbol{\mu}_x}$. Correlation is scale invariant, as expected. Therefore, the correlation between projected points $\{\Pi_{S_{w_x}}(\boldsymbol{x}_i)\}_{i=1}^N$ and $\{\Pi_{S_{w_y}}(\boldsymbol{y}_i)\}_{i=1}^N$ reduces to the correlation between the quantities that serve as projection coefficients here, $\{t_i\}_{i=1}^N$ and $\{u_i\}_{i=1}^N$.

Putting these pieces together, we obtain our generalized formulation for CCA,

$$\rho_{\boldsymbol{x},\boldsymbol{y}} = \mathrm{corr}(\pi_{\boldsymbol{w}_x}(\boldsymbol{x}), \pi_{\boldsymbol{w}_y}(\boldsymbol{y})) = \max_{\boldsymbol{w}_x, \boldsymbol{w}_y, \boldsymbol{t}, \boldsymbol{u}} \frac{\sum_{i=1}^N (t_i - \bar{t})(u_i - \bar{u})}{\sqrt{\sum_{i=1}^N (t_i - \bar{t})^2} \sqrt{\sum_{i=1}^N (u_i - \bar{u})^2}} \tag{4.6}$$

where $t_i = \pi_{\boldsymbol{w}_x}(\boldsymbol{x}_i)$, $\boldsymbol{t} := \{t_i\}$, $u_i = \pi_{\boldsymbol{w}_y}(\boldsymbol{y}_i)$, $\boldsymbol{u} := \{u_i\}$, $\bar{t} = \frac{1}{N}\sum_{i=1}^N t_i$ and $\bar{u} = \frac{1}{N}\sum_{i=1}^N u_i$. Expanding out components in (4.6) further, it takes the

form,

$$\begin{aligned}\rho_{x,y} = \max_{\boldsymbol{w}_x,\boldsymbol{w}_y,\boldsymbol{t},\boldsymbol{u}} & \frac{\sum_{i=1}^{N}(t_i-\bar{t})(u_i-\bar{u})}{\sqrt{\sum_{i=1}^{N}(t_i-\bar{t})^2}\sqrt{\sum_{i=1}^{N}(u_i-\bar{u})^2}} \\ s.t. \quad t_i &= \arg\min_{t_i\in(-\epsilon,\epsilon)} \|\mathrm{Log}(\mathrm{Exp}(\boldsymbol{\mu}_x, t_i\boldsymbol{w}_x), \boldsymbol{x}_i)\|^2, \forall i \in \{1,\dots,N\} \\ u_i &= \arg\min_{u_i\in(-\epsilon,\epsilon)} \|\mathrm{Log}(\mathrm{Exp}(\boldsymbol{\mu}_y, u_i\boldsymbol{w}_y), \boldsymbol{y}_i)\|^2, \forall i \in \{1,\dots,N\}\end{aligned} \tag{4.7}$$

Directly, we see that (4.3.2) is a multilevel optimization and solutions from nested sub-optimization problems (argmin in constraints) may be needed to solve the higher level problem. It turns out that deriving the first order optimality conditions suggests a cleaner formulation as

$$\begin{aligned}\rho(\boldsymbol{w}_x,\boldsymbol{w}_y) = \max_{\boldsymbol{w}_x,\boldsymbol{w}_y,\boldsymbol{t},\boldsymbol{u}} & f(\boldsymbol{t},\boldsymbol{u}) \\ s.t. \ \ & \nabla_{t_i} g(t_i,\boldsymbol{w}_x) = 0, \ \forall i \in \{1,\dots,N\} \\ & \nabla_{u_i} g(u_i,\boldsymbol{w}_y) = 0, \ \forall i \in \{1,\dots,N\}\end{aligned} \tag{4.8}$$

where $f(\boldsymbol{t},\boldsymbol{u}) := \frac{\sum_{i=1}^{N}(t_i-\bar{t})(u_i-\bar{u})}{\sqrt{\sum_{i=1}^{N}(t_i-\bar{t})^2}\sqrt{\sum_{i=1}^{N}(u_i-\bar{u})^2}}$, $g(t_i,\boldsymbol{w}_x) := \|\mathrm{Log}(\mathrm{Exp}(\boldsymbol{\mu}_x, t_i\boldsymbol{w}_x), \boldsymbol{x}_i)\|^2$, and $g(u_i,\boldsymbol{w}_y) := \|\mathrm{Log}(\mathrm{Exp}(\boldsymbol{\mu}_y, u_i\boldsymbol{w}_y), \boldsymbol{y}_i)\|^2$.

Note. We use the first order optimality condition in (4.8). In general, the first order optimality condition is necessary but not a sufficient condition. So, is (4.8) a relaxed version of (4.3.2)? Interestingly, on SPD manifolds, the first order condition is sufficient for the optimality of projection. To see this, we need the concept of *geodesic convexity* (Rapcsák, 1991). The following definitions are also introduced in Section 2.3.

Definition 4.1. *A set $A \subset \mathcal{M}$ is geodesically convex (g-convex) if any two points of A are joined by a geodesic belonging to A.*

Definition 4.2. *Let $A \subset \mathcal{M}$ be a g-convex set. Then, a function $f : A \to R$ is*

g-convex if its restrictions to all geodesic arcs belonging to A are convex in the arc length parameter, i.e., if $t\ f(t) \equiv f(Exp(x, tu))$ is convex over its domain for all $x \in \mathcal{M}$ and $u \in T_x\mathcal{M}$, where $Exp(x, \cdot)$ is the exponential map at x (Moakher, 2005).

Lemma 4.3. *The function $d(Exp(\mu, tu), S)$ on SPD manifolds is convex with respect to t where $\mu, S \in \mathcal{M}$ and $u \in T_\mu\mathcal{M}$.*

Proof. This can be shown by the definition of the geodesic convexity of the function and the fact that the real-valued function defined on $SPD(n)$ by $P \mapsto d(P, S)$ is geodesically convex, where $S \in SPD(n)$ is fixed and $d(\cdot, \cdot)$ is the geodesic distance (Mostow, 1973; Moakher, 2005). □

Lemma 4.3 shows that the projection to a geodesic curve on SPD manifolds is a convex problem and the first order condition for projection coefficients is sufficient.

4.3 Optimization Schemes

We present two different algorithms to solve the problem of computing CCA on Riemannian manifolds. The first algorithm is based on a numerical optimization for (4.8). Subsequently, we present the second approach which is based on an approximation for a more efficient algorithm.

4.3.1 An Augmented Lagrangian Method

Due to the nature of our formulation, especially the constraints, our options for numerical optimization scheme are limited. In particular, to avoid dealing with the second order derivatives of the constraints leads us to first order methods. One option here is a gradient projection method (Bertsekas, 1999). However, we will need to define distance metric over

the decision variables and projection on the feasible set accordingly. In this case, efficient projections on the feasible set may not be available.

The other option is a quadratic penalty algorithm. Given a constrained optimization problem $\max f(\boldsymbol{x})$ s.t. $c_i(\boldsymbol{x}) = 0, \forall i$, such an algorithm optimizes the quadratic penalty function, i.e., $\max f(x) - \nu^k \sum_i c_i(x)^2$. Classical penalty algorithms iteratively solve a sequence of models above while increasing ν^k to infinity. Here, we chose a well known variation of the penalty method called augmented Lagrangian technique (ALM) (Nocedal and Wright, 2006b). It is generally preferred to the classical quadratic penalty method since there is little extra computational cost. In particular, by avoiding $\mu \to \infty$, we reduce the possibility of ill conditioning by introducing explicit Lagrange multiplier estimates into the function to be minimized (Nocedal and Wright, 2006b). The augmented Lagrangian method solves a sequence of the following models while increasing ν_k.

$$\max f(\boldsymbol{x}) + \sum_i \lambda_i c_i(\boldsymbol{x}) - \nu^k \sum_i c_i(\boldsymbol{x})^2 \tag{4.9}$$

The augmented Lagrangian formulation for our CCA formulation is given by

$$\begin{aligned}\max_{\boldsymbol{w}_x, \boldsymbol{w}_y, \boldsymbol{t}, \boldsymbol{u}} \mathcal{L}_A(\boldsymbol{w}_x, \boldsymbol{w}_y, \boldsymbol{t}, \boldsymbol{u}, \boldsymbol{\lambda}^k; \nu^k) = \max_{\boldsymbol{w}_x, \boldsymbol{w}_y, \boldsymbol{t}, \boldsymbol{u}} f(\boldsymbol{t}, \boldsymbol{u}) + \sum_i^N \lambda_{t_i}^k \nabla_{t_i} g(t_i, \boldsymbol{w}_x) + \\ \sum_i^N \lambda_{u_i}^k \nabla_{u_i} g(u_i, \boldsymbol{w}_y) - \frac{\nu^k}{2} \left(\sum_{i=1}^N \nabla_{t_i} g(t_i, \boldsymbol{w}_x)^2 + \nabla_{u_i} g(u_i, \boldsymbol{w}_y)^2 \right)\end{aligned} \tag{4.10}$$

The pseudocode for our algorithm is summarized in Algorithm 4.

Remarks. Note that for Algorithm 4, we need the second derivative of g, in particular, for $\frac{d^2}{dwdt} g$ and $\frac{d^2}{dt^2} g$. The literature does not provide a great deal of guidance on second derivatives of functions involving $\mathrm{Log}(\cdot)$ and

Algorithm 4 Riemannian CCA based on the Augmented Lagarangian method

$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathcal{M}_x, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_N \in \mathcal{M}_y$
Given $\nu^0 > 0, \tau^0 > 0$, starting points $(\boldsymbol{w}_x^0, \boldsymbol{w}_y^0, \boldsymbol{t}^0, \boldsymbol{u}^0)$ and $\boldsymbol{\lambda}^0$
for $k = 0, 1, 2 \ldots$ **do**
    Start at $(\boldsymbol{w}_x^k, \boldsymbol{w}_y^k, \boldsymbol{t}^k, \boldsymbol{u}^k)$
    Find an approximate minimizer $(\boldsymbol{w}_x^k, \boldsymbol{w}_y^k, \boldsymbol{t}^k, \boldsymbol{u}^k)$ of $\mathcal{L}_A(\cdot, \boldsymbol{\lambda}^k; \nu^k)$, and terminate when $\|\nabla \mathcal{L}_A(\boldsymbol{w}_x^k, \boldsymbol{w}_y^k, \boldsymbol{t}^k, \boldsymbol{u}^k, \boldsymbol{\lambda}^k; \nu^k)\| \leq \tau^k$
    if a convergence test for (4.8) is satisfied **then**
        Stop with approximate feasible solution
    end if
    $\lambda_{t_i}^{k+1} = \lambda_{t_i}^k - \nu^k \nabla_{t_i} g(t_i, \boldsymbol{w}_x), \forall i$
    $\lambda_{u_i}^{k+1} = \lambda_{u_i}^k - \nu^k \nabla_{u_i} g(u_i, \boldsymbol{w}_y), \forall i$
    Choose new penalty parameter $\nu^{k+1} \geq \nu^k$
    Set starting point for the next iteration
    Select tolerance τ^{k+1}
end for

$\text{Exp}(\cdot)$ maps on general Riemannian manifolds. However, depending on the manifold, it can be obtained analytically or numerically (see Section A.2).

Approximate strategies. It is clear that the core difficulty in deriving the algorithm above was the lack of a closed form solution to projections on to geodesic submanifolds. If however, an approximate form of the projection can lead to significant gains in computational efficiency with little sacrifice in accuracy, it is worthy of consideration. The simplest approximation is to use a Log-Euclidean model. But it is well known that the Log-Euclidean is reasonable for data that are tightly clustered on the manifold and not otherwise. Further, the Log-Euclidean metric lacks the important property of affine invariance. We can obtain a more accurate projection using the submanifold expression given in (Xie et al., 2010). The form of projection is,

Algorithm 5 CCA with approximate projection

1: Input $X_1, \ldots, X_N \in \mathcal{M}_y$, $Y_1, \ldots, Y_N \in M_y$
2: Compute intrinsic mean $\boldsymbol{\mu}_x, \boldsymbol{\mu}_y$ of $\{X_i\}, \{Y_i\}$
3: Compute $X_i^{\backprime} = \mathrm{Log}(\boldsymbol{\mu}_x, X_i)$, $Y_i^{\backprime} = \mathrm{Log}(\boldsymbol{\mu}_y, Y_i)$
4: Transform (using group action) $\{X_i^{\backprime}\}, \{Y_i^{\backprime}\}$ to the $T_I\mathcal{M}_x, T_I\mathcal{M}_y$
5: Perform CCA between $T_I\mathcal{M}_x, T_I\mathcal{M}_y$ and get axes $W_a \in T_I\mathcal{M}_x$, $W_b \in T_I\mathcal{M}_y$
6: Transform (using group action) W_a, W_b to $T_{\boldsymbol{\mu}_x}\mathcal{M}_x$, $T_{\boldsymbol{\mu}_y}\mathcal{M}_y$

$$\Pi_S(\boldsymbol{x}) \approx \mathrm{Exp}(\boldsymbol{\mu}, \sum_{i=1}^{d} \boldsymbol{v}_i \langle \boldsymbol{v}_i, \mathrm{Log}(\boldsymbol{\mu}, \boldsymbol{x}) \rangle_{\boldsymbol{\mu}}) \tag{4.11}$$

where $\{\boldsymbol{v}_i\}$ are *orthonormal basis* at $T_{\boldsymbol{\mu}}\mathcal{M}$. The CCA algorithm with this approximation for the projection is summarized as Algorithm 5.

Algorithm 5 finds a globally optimal solution to the approximate problem, i.e., the classical version of CCA between two tangent spaces $T_I\mathcal{M}_x$ and $T_I\mathcal{M}_y$. It does not require any initialization. On the other hand, Algorithm 4 is a first order optimization scheme. It converges to a local minimum. Different initializations may lead to different local solutions. In our experiments, for Algorithm 4, we initialized $\mathbf{w}_x$ and $\mathbf{w}_y$ by Algorithm 5. Further, $\boldsymbol{t}$ and $\boldsymbol{u}$ are are initialized by the corresponding projection coefficients to $\mathbf{w}_x$ and $\mathbf{w}_y$ using the iterative method minimizing (4.5).

Finally, we provide a brief remark on one remaining issue. This relates to the question of why we use the group action rather than other transformations such as parallel transport. Observe that Algorithm 5 sends the data from the tangent space at the Karcher mean of the samples to the tangent space at Identity I. The purpose of the transformation is to put all samples at the Identity of the SPD manifold, to obtain a more accurate projection, which can be understood by inspecting (4.11). The projection and inner product depend on the anchor point μ. If μ is Identity, then

there is no discrepancy between the Euclidean and the Riemannian inner products. Of course, one may use a parallel transport. However, group action may be substantially more efficient than parallel transport since the former does not require computing a geodesic curve (which is needed for parallel transport). Interestingly, Theorem 4.4 says that on SPD manifolds with a GL-invariant metric, parallel transport from an arbitrary point p to Identity I is *equivalent* to the transform via a group action. So, one can parallel transport tangent vectors from p to I (or vice versa) using the group action more efficiently.

Theorem 4.4. *On SPD manifold, let $\Gamma_{p\to I}(w)$ denote the parallel transport of $w \in T_p\mathcal{M}$ along the geodesic from $p \in \mathcal{M}$ to $I \in \mathcal{M}$. The parallel transport is equivalent to group action by $p^{-1/2}wp^{-T/2}$, where the inner product $\langle u, v\rangle_p = tr(p^{-1/2}up^{-1}vp^{-1/2})$.*

Proof. Parallel transport Γ from p to q is given by (Ferreira et al., 2006)

$$\begin{aligned}
&\Gamma_{p\to q}(w) = p^{1/2}rp^{-1/2}wp^{-1/2}rp^{1/2},\\
&\text{where } r = \exp(p^{-1/2}\frac{v}{2}p^{-1/2})\\
&\quad \text{and} \quad v = \mathrm{Log}(p,q) = p^{1/2}\log(p^{-1/2}qp^{-1/2})p^{1/2}
\end{aligned}$$

Let us transform the tangent vector w at $T_p\mathcal{M}$ to I by setting $q = I$.

$$\Gamma_{p\to I}(w) = p^{1/2}rp^{-1/2}wp^{-1/2}rp^{1/2} \quad \text{where } r = \exp(p^{-1/2}\frac{v}{2}p^{-1/2}) \quad \text{and}$$
$$v = \mathrm{Log}(p,I) = p^{1/2}\log(p^{-1/2}Ip^{-1/2})p^{1/2} = p^{1/2}\log(p^{-1})p^{1/2} \tag{a}$$

Then r is given as,

$$\begin{aligned} r &= \exp(p^{-1/2}\frac{v}{2}p^{-1/2}) \\ &= \exp(p^{-1/2}p^{1/2}\log(p^{-1})p^{1/2}p^{-1/2}/2), \text{ by (a)} \\ &= \exp(\log(p^{-1})/2) \\ &= p^{-1/2} \end{aligned} \tag{b}$$

Also,

$$\begin{aligned} \Gamma_{p\to I}(w) &= p^{1/2}rp^{-1/2}wp^{-1/2}rp^{1/2} \\ &= p^{1/2}p^{-1/2}p^{-1/2}wp^{-1/2}p^{-1/2}p^{1/2}, \text{ by (b)} \\ &= p^{-1/2}wp^{-1/2} \\ &= p^{-1/2}wp^{-T/2} \text{ since } p^{-1/2} \text{ is SPD.} \end{aligned}$$

□

Theorem 4.5. *On SPD manifolds, let $\Gamma_{I\to q}(w)$ denote the parallel transport of $w \in T_I\mathcal{M}$ along the geodesic from $I \in \mathcal{M}$ to $q \in \mathcal{M}$. The parallel transport is equivalent to a group action by $q^{1/2}wq^{T/2}$.*

Proof. The proof is similar to that of Theorem 4.4. By substitution, the parallel transport is given by $\Gamma_{I\to q}(w) = rwr$, where $r = \exp(\frac{v}{2})$ and $v = \text{Log}(I, q) = \log(q)$. Then, r is $q^{1/2}$. Hence, $\Gamma_{I\to q}(w) = q^{1/2}wq^{1/2} = q^{1/2}wq^{T/2}$ since $q^{1/2}$ is SPD. □

Remarks. Theorem 4.4 and Theorem 4.5 show that the parallel transport from or to I is replaceable with group actions. However, in general, the parallel transport of $w \in T_p\mathcal{M}$ from p to q is *not* equivalent to the composition of group actions to transform from p to I and from I to q. This is consistent with the fact that parallel transport from a to c may not be same as two parallel transports: from a to b and then from b to c. The following is an example for SPD(2) manifold.

Example 4.3.1. *When p and q are given as*

$$p = \begin{pmatrix} 2 & 3 \\ 3 & 5 \end{pmatrix} \quad q = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix} \quad w = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

The parallel transport of w from p to q is

$$\Gamma_{p \to q}(w) \approx \begin{pmatrix} -8 & -1 \\ -1 & 0 \end{pmatrix}$$

Transform of w by two group actions $(p \to I \to q)$ is

$$q^{1/2} p^{-1/2} w p^{-T/2} q^{T/2} \approx \begin{pmatrix} -4 & 1 \\ 1 & 0 \end{pmatrix}$$

The transform by group actions above is identical to the composition of two parallel transports $\Gamma_{I \to q}(\Gamma_{p \to I}(w))$. However, it is different from $\Gamma_{p \to q}(w)$.

4.3.2 Extensions to the product Riemannian manifold

For applications of this algorithm, we study the problem of statistical analysis of an entire population of images (of multiple types). For such data, each image must be treated as a single entity, which necessitates extending the formulation above to a Riemannian product space.

In other words, our CCA will be performed on product manifolds given as

$$\boldsymbol{\mathcal{M}}_x := \mathcal{M}_x^1 \times \ldots \times \mathcal{M}_x^m, \text{ and } \boldsymbol{\mathcal{M}}_y := \mathcal{M}_y^1 \times \ldots \times \mathcal{M}_y^n. \tag{4.12}$$

We seek $Wb_x := (W_x^1, \ldots, W_x^m) \in T_{\boldsymbol{\mu}_x} \boldsymbol{\mathcal{M}}_x$, $Wb_y := (W_y^1, \ldots, W_y^m) \in T_{\boldsymbol{\mu}_y} \boldsymbol{\mathcal{M}}_y$, where $T_{\mu_x} \boldsymbol{\mathcal{M}}_x := T_{\mu_x^1} \boldsymbol{\mathcal{M}}_x^1 \times \ldots T_{\mu_x^m} \boldsymbol{\mathcal{M}}_x^m$, and $T_{\mu_y} \boldsymbol{\mathcal{M}}_y := T_{\mu_y^1} \boldsymbol{\mathcal{M}}_y^1 \times \ldots T_{\mu_y^n} \boldsymbol{\mathcal{M}}_y^n$. We will discuss a Riemannian metric on the product space and *projection coefficients*. Finally, we will offer the extended formulation

of our method.

First, let us define a Riemannian metric on the product space $\boldsymbol{\mathcal{M}} = \mathcal{M}^1 \times \ldots \times \mathcal{M}^m$. A natural choice is the following idea from (Xie et al., 2010).

$$\langle \boldsymbol{X}_1, \boldsymbol{X}_2 \rangle_{\boldsymbol{P}} = \sum_{j=1}^{m} \langle X_1^j, X_2^j \rangle_{Pj} \tag{4.13}$$

where $\boldsymbol{X}_1 = \left(X_1^1, \ldots, X_1^m\right) \in \boldsymbol{\mathcal{M}}$, and $\boldsymbol{X}_2 = \left(X_2^1, \ldots, X_2^m\right) \in \boldsymbol{\mathcal{M}}$ and $\boldsymbol{P} = \left(P^1, \ldots, P^m\right) \in \boldsymbol{\mathcal{M}}$. Once we have the exponential and logarithm maps, CCA on a Riemannian product space can be directly performed by Algorithm 5. The exponential map $\mathrm{Exp}(\boldsymbol{P}, \boldsymbol{V})$ and logarithm map $\mathrm{Log}(\boldsymbol{P}, \boldsymbol{X})$ are given by

$$(\mathrm{Exp}(P^1, V^1), \ldots, \mathrm{Exp}(P^m, V^m)) \text{ and } (\mathrm{Log}(P^1, X^1), \ldots, \mathrm{Log}(P^m, X^m)) \tag{4.14}$$

respectively, where $\boldsymbol{V} = (V^1, \ldots, V^m) \in T_{\boldsymbol{P}}\boldsymbol{\mathcal{M}}$. The length of tangent vector is $\|\boldsymbol{V}\| = \sqrt{\|V^1\|_{P1}^2 + \cdots + \|V^m\|_{Pm}^2}$, where $V^i \in T_{Pi}\mathcal{M}_i$. The geodesic distance between two points $\mathrm{d}(\boldsymbol{X}_1, \boldsymbol{X}_2)$ on Riemannian product space is also measured by the length of tangent vector from one point to the other. So, we have

$$\mathrm{d}(\boldsymbol{\mu}_x, \boldsymbol{X}) = \sqrt{\mathrm{d}(\mu_x^1, X^1)^2 + \cdots + \mathrm{d}(\mu_x^m, X^m)^2} \tag{4.15}$$

From our previous discussion of the relationship between *projection coefficients* and distance from the mean to points (after *projection*) in Section 4.2, we have $t_i = \mathrm{d}(\boldsymbol{\mu}_x, \Pi_{S_{\boldsymbol{W}x}}(\boldsymbol{X}_i))/\|\boldsymbol{W}_x\|_{\boldsymbol{\mu}_x}$ and $t_i^j = \mathrm{d}(\mu_x^j, \Pi_{S_{W_x^j}}(X_i^j))/\|W_x^j\|_{\mu_x^j}$. By substitution, the *projection coefficients* on Riemannian product space are

given by

$$t_i = \mathrm{d}(\boldsymbol{\mu}_x, \Pi_{S_{W_x}}(\boldsymbol{X}_i))/\|\boldsymbol{W}_x\|_{\boldsymbol{\mu}_x} = \sqrt{\sum_j^m \left(t_i^j \left\|W_x^j\right\|_{\mu_x^j}\right)^2 / \sum_{j=1}^m \left\|W_x^j\right\|_{\mu_x^j}^2} \tag{4.16}$$

We can now mechanically substitute these "product space" versions of the terms in (4.16) to derive a CCA on a Riemannian product space.

Our formulation is

$$\begin{aligned}
\rho_{X,Y} = \max_{\boldsymbol{W}_x, \boldsymbol{W}_y, \boldsymbol{t}, \boldsymbol{u}} & \frac{\sum_{i=1}^N (t_i - \bar{t})(u_i - \bar{u})}{\sqrt{\sum_{i=1}^N (t_i - \bar{t})^2}\sqrt{\sum_{i=1}^N (u_i - \bar{u})^2}} \\
s.t.\ \ t_i^j &= \arg\min_{t_i^j \in (-\epsilon, \epsilon)} \|\mathrm{Log}(\mathrm{Exp}(\mu_x^j, t_i^j W_x^j), X_i^j)\|^2, \forall i, \forall j \\
u_i^k &= \arg\min_{u_i^k \in (-\epsilon, \epsilon)} \|\mathrm{Log}(\mathrm{Exp}(\mu_y^k, u_i^k W_y^k), Y_i^k)\|^2, \forall i, \forall k \qquad (4.17) \\
t_i &= \frac{\sqrt{\sum_{j=1}^m \left(t_i^j \|W_x^j\|_{\mu_x^j}\right)^2}}{\sqrt{\sum_{j=1}^m \|W_x^j\|_{\mu_x^j}^2}}, u_i = \frac{\sqrt{\sum_{k=1}^n \left(u_i^k \|W_y^k\|_{\mu_y^k}\right)^2}}{\sqrt{\sum_{k=1}^n \|W_y^k\|_{\mu_y^k}^2}} \qquad (4.18) \\
\bar{t} &= \frac{1}{N}\sum_i^N t_i, \bar{u} = \frac{1}{N}\sum_i^N u_i \forall i
\end{aligned}$$

where $i \in \{1, \ldots, N\}$, $j \in \{1, \ldots, m\}$, and $\forall k \in \{1, \ldots, n\}$. This can be optimized by constrained optimization algorithms similar to those described in Section 4.3.1 with relatively minor changes.

4.4 Experimental results

Diffusion tensors are symmetric positive definite matrices $\mathrm{SPD}(n)$ at each voxel in a diffusion tensor image (DTI). We introduced $\mathrm{SPD}(n)$ and related operations on that space in Chapter 2.4.3 including related operations.

Recall that the exponential map and logarithm map in SPD(n) are defined as,

$$\mathrm{Exp}(p,v) = p^{1/2}\exp(p^{-1/2}vp^{-1/2})p^{1/2}, \;\; \mathrm{Log}(p,q) = p^{1/2}\log(p^{-1/2}qp^{-1/2})p^{1/2} \tag{4.19}$$

and the first derivative of g in (4.8) on SPD(n) is given by

$$\begin{aligned}\frac{d}{dt_i}g(t_i,\boldsymbol{w}_x) &= \frac{d}{dt_i}\|\mathrm{Log}(\mathrm{Exp}(\mu_x,t_iW_x),X_i)\|^2 = \frac{d}{dt_i}\mathrm{tr}[\log^2(X_i^{-1}S(t_i))] \\ &= 2\mathrm{tr}[\log(X_i^{-1}S(t_i))S(t_i)^{-1}\dot{S}(t_i)], \text{ by Prop. 2.1 in (Moakher, 2005)}\end{aligned} \tag{4.20}$$

where

$$\begin{aligned}S(t_i) &= \mathrm{Exp}(\mu_x,t_iW_x) = \mu_x^{1/2}\exp^{t_iA}\mu_x^{1/2} \\ \dot{S}(t_i) &= \mu_x^{1/2}A\exp^{t_iA}\mu_x^{1/2} \\ A &= \mu_x^{-1/2}W_x\mu_x^{-1/2}\end{aligned} \tag{4.21}$$

Note that the derivative of the equality constraints in (4.8), namely $\frac{d^2}{dWdt}g$, $\frac{d^2}{dt^2}g$, are calculated numerically. The numerical differentiation requires an orthonormal basis of the tangent space.

4.4.1 Synthetic experiments

In this section, we provide experimental results using a synthetic dataset to evaluate the performance of Riemannian CCA. To simplify presentation, we introduce two operations vec($\cdot$) and mat($\cdot$) that will be used in the following description. We use 'vec' to give the operation of embedding tangent vectors in $T_I\mathcal{M}$ into $\mathbf{R}^6$; 'mat' refers to the inversion of 'vec'. By construction, we have $\langle S_1,S_2\rangle_I = \langle v_1,v_2\rangle$, where $v_i = \mathrm{vec}(S_i)$. In other words, the distance from a base point/origin to each point is identical

in the two spaces by the construction. Using group actions and these subroutines, points can be mapped from an arbitrary tangent space $T_p\mathcal{M}$ to $\mathbf{R}^6$, or vice versa, where $p \in \mathcal{M}$. On SPD(3), the two operations are given by

$$\text{vec}(S) := [s_{11}, \sqrt{2}s_{12}, \sqrt{2}s_{13}, s_{22}, \sqrt{2}s_{23}, s_{33}]^T, \text{where } S = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix}$$
$$\text{and} \quad \text{mat}(v) := \begin{bmatrix} v_1 & \frac{1}{\sqrt{2}}v_2 & \frac{1}{\sqrt{2}}v_3 \\ \frac{1}{\sqrt{2}}v_2 & v_4 & \frac{1}{\sqrt{2}}v_5 \\ \frac{1}{\sqrt{2}}v_3 & \frac{1}{\sqrt{2}}v_5 & v_6 \end{bmatrix}, \text{where } v = \begin{bmatrix} v_1 \\ \vdots \\ v_6 \end{bmatrix}^T. \tag{4.22}$$

Our CCA algorithm with approximate projections, namely, Algorithm 5 can be implemented by these two subroutines with group actions.

We now discuss the synthetic data generation. The samples are generated to be spread far apart on the manifold $\mathcal{M}(\equiv \text{SPD}(3))$ — observe that if the data are closely clustered, a manifold algorithm will behave similar to its non-manifold counterpart. We generate data around two well separated means $\mu_{x_1}, \mu_{x_2} \in \mathcal{X}$, $\mu_{y_1}, \mu_{y_2} \in \mathcal{Y}$ by perturbing the data randomly in the corresponding tangent spaces, i.e., adding Gaussian-like noise in each tangent space at cluster mean μ_{x_j} and μ_{y_j}, where $j \in \{1, 2\}$ is the index for cluster. The procedure is described in Algorithm 6.

We plot data projected on to the CCA axes (P_X and P_Y) and compute the correlation coefficients. In our experiments, we see that the algorithm offers improvements. For example, by inspecting the correlation coefficients $\rho_{x,y}$ in Fig. 4.3, we see that manifold CCA yields significantly better correlation relative to other baselines.

Algorithm 6 The procedure simulates truncated-Gaussian-like noise. The second step (safeguard) in the pseudocode ensures that the data lives in a reasonably small neighborhood to avoid numerical issues. We define subroutines *mat* for mapping from $\mathbf{R}^{n(n+1)/2}$ to SPD(n) and *vec* for the inversion.

1: $\boldsymbol{\epsilon}' \in \mathbf{R}^{n(n+1)/2} \sim \mathcal{N}(0, \sigma I)$
2: $\boldsymbol{\epsilon}' \leftarrow \boldsymbol{\epsilon}' \min(\|\boldsymbol{\epsilon}'\|, c_1)/\|\boldsymbol{\epsilon}'\|$ ▷ c_1 is a parameter for a safeguard
3: $\epsilon_I \leftarrow \text{mat}(\boldsymbol{\epsilon}')$, ▷ tangent vector at I
4: Transform (using group action) ϵ_I to $T_\mu\mathcal{M}$
5: Perturb data $X \leftarrow \text{Exp}(\mu, \epsilon_\mu)$, where $\epsilon_\mu \in T_\mu\mathcal{M}$

4.4.2 CCA for multi-modal AD risk factor analysis

We collected multi-modal magnetic resonance imaging (MRI) data to investigate the effects of risk for Alzheimer's disease (AD) on the white and gray matter in the brain. One of the goals in analyzing this dataset is to find statistically significant relationships between AD risk and the brain structure. We can adopt many different ways of modeling these relationships but a potentially useful way is to analyze multimodality imaging data simultaneously, using CCA. CCA allows to identify important features (brain regions) based on the correlation between two modalities.

In the current experiments, we include a subset of 343 subjects and first investigate the effects of age and gender in a multimodal fashion since these variables are also important factors in healthy aging. Brain structure is characterized by diffusion weighted images (DWI) for white matter and T1-weighted (T1W) image data for the gray matter. DWI data provides us information about the microstructure of the white matter. We use diffusion tensor ($\mathcal{D} \in$ SPD(3)) model to represent the diffusivity in the microstructure. T1W data can be used to obtain volumetric properties of the gray-matter (Garrido et al., 2009). The volumetric information is obtained from Jacobian matrices (J) of the diffeomorphic mapping to a population specific template. These Jacobian matrices can be used

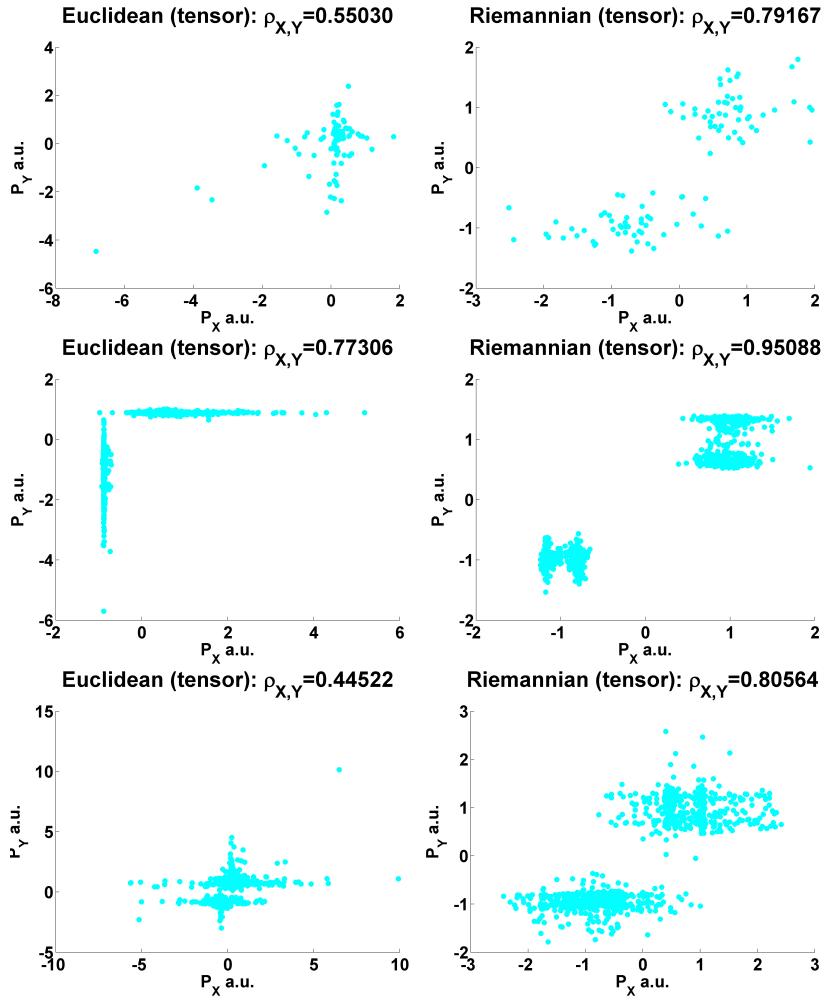


Figure 4.3: Synthetic experiments showing the benefits of Riemannian CCA. The left column shows the projected data using the Euclidean CCA and the right column is obtained using Riemannian CCA. P_X and P_Y denote the projected axes. Each row represents a synthetic experiment with a specific set of $\{\mu_{x_j}, \epsilon_{x_j}; \mu_{y_j}, \epsilon_{y_j}\}$. The first row has 100 samples while the last two rows have 1000 samples. The improvements in the correlation coefficients $\rho_{x,y}$ can be seen from the corresponding titles.

to obtain the Cauchy deformation tensors ($\sqrt{J^T J}$) which also belong to SPD(3).

We first focus our analysis using two important regions called hip-

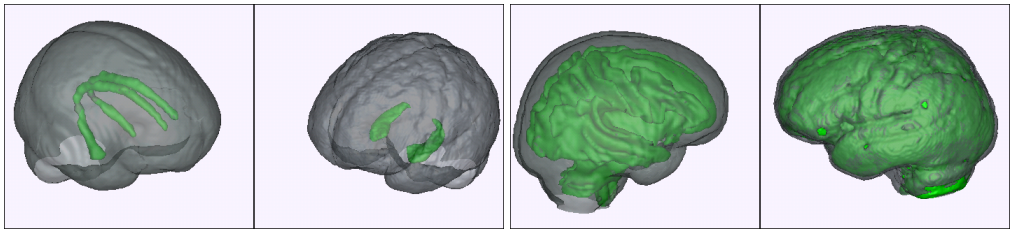


Figure 4.4: Shown on the left are the bilateral cingulum bundles (green) inside a brain surface obtained from a population DTI template. Similarly on the right are the bilateral hippocampi. The full gray matter and white matter are shown on the right.

pocampus and cingulum bundle (shown in Fig. 4.4) which are significantly affected by Alzheimer's disease. However, the statistical power of detecting changes in the brain structures *before* the memory/cognitive function is impaired is difficult due to several factors such as noise in the data, small sample and effect sizes. One approach to improving statistical power in such a setting is to perform only a few number of tests using average properties of the substructures. This procedure reduces both noise and the number of tests. However, taking averages will also dampen the signal of interest which is already weak in certain studies. CCA can take the multi-modal information from the imaging data and project the voxels into a space where the signal of interest is likely to be stronger.

Experimental design: The main idea is to detect age and gender effects on the gray and white matter interactions. Hence the key multimodal linear relations we examine for the purpose are

$$Y_{\text{DTI}} = \beta_0 + \beta_1 \text{Gender} + \beta_2 X_{\text{T1W}} + \beta_3 X_{\text{T1W}} \cdot \text{Gender} + \varepsilon,$$
$$Y_{\text{DTI}} = \beta_0' + \beta_1' \text{AgeGroup} + \beta_2' X_{\text{T1W}} + \beta_3' X_{\text{T1W}} \cdot \text{AgeGroup} + \varepsilon,$$

where the AgeGroup is defined as a categorical variable with 0 (middle aged) if the age of the subject ≤ 65 and 1 (old) otherwise. The sample under investigation is between 43 and 75 years of age (see Fig. 4.5 for the full distributions of age and gender). The statistical tests we ask are if

the Null hypotheses $\beta_3 = 0$ and $\beta_3' = 0$ can be rejected using our data at $\alpha = 0.05$. The gender and age distributions of the sample are shown in Fig. 4.5.

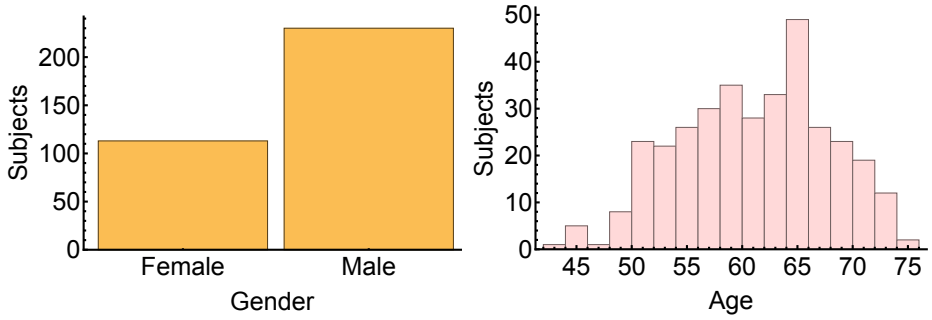


Figure 4.5: The sample characteristics in terms of gender and age distributions.

We report the results from the following four sets of analyses: **(i)** Classical ROI-average analysis: This is a standard type of setting where the brain measurements in an ROI are averaged. Here $Y_{\text{DTI}} = \overline{\text{MD}}$ i.e., the average mean diffusivity in the cingulum bundle. $X_{\text{T1W}} = \overline{\log|J|}$ i.e., the average volumetric change (relative to the population template) in the hippocampus. **(ii)** Euclidean CCA using scalar measures (MD and $\log|J|$) in the ROIs: Here, the voxel data is projected using the classical CCA approach (Witten et al., 2009) i.e., $Y_{\text{DTI}} = \mathbf{w}_{\text{MD}}^T \text{MD}$ and $X_{\text{T1W}} = \mathbf{w}_{\log|J|}^T \log|J|$. **(iii)** Euclidean CCA using $\mathcal{D}$ and $\mathcal{J}$ in the ROIs: This setting is an improvement to the setting in **(ii)** above in that the projections are performed using the full tensor data (Witten et al., 2009). Here $Y_{\text{DTI}} = \mathbf{w}_{\mathcal{D}}^T \mathcal{D}$ and $X_{\text{T1W}} = \mathbf{w}_{\mathcal{J}}^T \mathcal{J}$. **(iv)** Riemannian CCA using $\mathcal{D}$ and $\mathcal{J}$ in the ROIs: Here $Y_{\text{DTI}} = \langle \mathbf{w}_{\mathcal{D}}, \mathcal{D} \rangle_{\mu_{\mathcal{D}}}$ and $X_{\text{T1W}} = \langle \mathbf{w}_{\mathcal{J}}, \mathcal{J} \rangle_{\mu_{\mathcal{J}}}$.

We first show the statistical sensitivities of the four approaches in Fig. 4.6. We can see that the performance of CCA using the full tensor information improves the statistical significance for both Euclidean and Riemannian approaches. The weight vectors in the different settings for both Euclidean and Riemannian CCA are shown in Fig. 4.7, top row. We

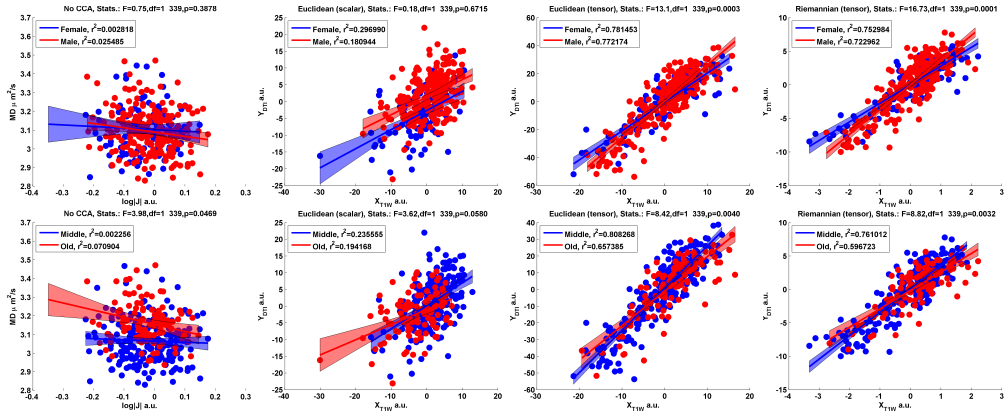


Figure 4.6: Experimental evidence showing the improvements in statistical significance of finding the multi-modal risk-brain interaction effects. Top row shows the gender, volume and diffusivity interactions. Second row shows the interaction effects of the middle/old age groups.

would like to note that there are several different approaches of using the data from CCA and we performed additional experiments with full gray matter and white matter regions in the brain. We only show and discuss briefly the representative weight vectors in the bottom row of Fig. 4.7 bottom row and refer the interested reader to Appendix 2 for more comprehensive details of the full brain experiments. Interestingly, the weight vectors are spatially cohesive even without enforcing any spatial constraints. What is even more interesting is that the regions picked between the DTI and T1W modalities are complimentary in a biological sense. Specifically, when performing our CCA on the ROIs, although the cingulum bundle extends into the superior mid-brain regions, the weights are non-zero in its hippocampal projections. In the case of entire white and gray matter regions, the volumetric difference (from the population template) in the inferior part of the corpus callosum seems to be highly cross-correlated to the diffusivity in the corpus callosum. Our CCA finds these projections without any a priori constraints in the optimization suggesting that performing CCA on the native data can reveal biologically

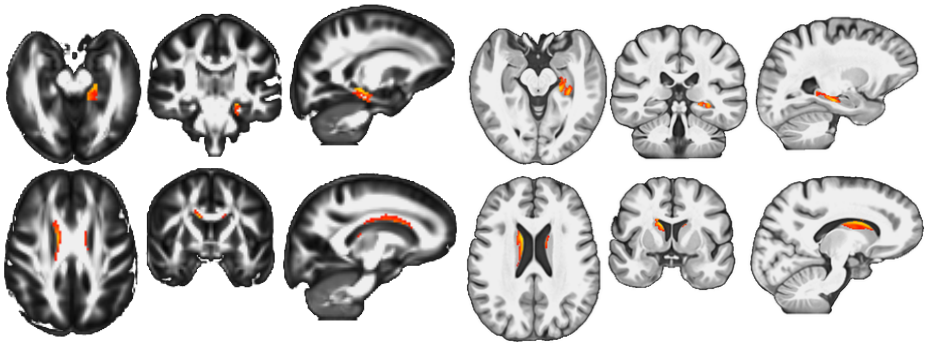


Figure 4.7: Weight vectors (in red-yellow color) obtained from our Riemannian CCA approach. The weights are in arbitrary units. The top row is from applying Riemannian CCA on data from the cingulum and hippocampus structures (Fig. 3) while the bottom row is obtained using data from the entire white and gray matter regions of the brain. On the left (three columns) block we show the results in orthogonal view for DTI and on the right for T1W. The corresponding underlays are the population averages of the fractional anisotropy and T1W contrast images respectively.

meaningful patterns.

We have presented in these experiments (including those in the appendix) evidence that CCA when performed using the intrinsic properties of the MRI data can reveal biologically meaningful patterns without any *a priori* biological input to the model. We showed that we can perform various types of multi-modal hypothesis testing of linear relationships using the projection vectors from the CCA, which can be easily extended to discriminant analysis (predicting gender and age group using the multi-modal brain data) using the CCA projection vectors. CCA can be applied to settings beyond multi-modal imaging data, where one can try to directly maximize the correlation between imaging and non-imaging data using a cross-validation technique (Avants et al., 2014). Our Riemannian CCA can provide a starting point for such studies.

4.5 Summary

The classical CCA assumes that data live in a pair of vector spaces. However, many modern scientific disciplines require the analysis of data which belong to *curved* spaces where classical CCA is no longer applicable. Motivated by the properties of imaging data from neuroimaging studies, we generalize CCA to Riemannian manifolds. We employ differential geometry tools to extend operations in CCA to the manifold setting. Such a formulation results in a multi-level optimization problem. We derive solutions using the first order condition of projection and an augmented Lagrangian method. In addition, we also develop an efficient single path algorithm with approximate projections. Finally, we propose a generalization to the product space of SPD(n), namely, tensor fields allowing us to treat a full brain image as a point on the product manifold. Experiments show the applicability of these ideas for the analysis of neuroimaging data. The code is publicly available [1].

[1] https://github.com/MLman/Riem-CCA

5 THE DIRICHLET MIXTURES OF MANIFOLD-VALUED MULTIVARIATE GENERAL LINEAR MODELS

The aim of this chapter is to develop a model to capture complex nonlinear correlations (beyond geodesic relationship) between Euclidean covariates and manifold-valued responses. To do so, we propose a new nonparametric model which is defined over multiple relevant tangent spaces, namely, Dirichlet process mixtures of manifold-valued multivariate general linear models (DP-MMGLMs), see Fig. 5.1. For efficient estimation of the model on manifolds, we propose a new Hamiltonian/Hybrid Monte Carlo (HMC) algorithm and a new distribution to obtain a set of parameters on the SPD manifold and its tangent space. Our experiments show that a DP-MMGLM captures more complicated nonlinear correlation rather than a MMGLM introduced in Chapter 3. Also, our model clusters samples based on nonlinear correlations between Euclidean covariates and manifold-valued response variables. We demonstrate the clustering effect by grouping voxels in a patch based on spatially-based covariates and the shape of 3D tensors $\mathrm{SPD}(3)$. For real-world data, we investigate how facial landmark appearances evolve with age using region covariance descriptors.

5.1 DP-GLM in the Euclidean space

Recall that we studied the general linear model (GLM) and its generalization on Riemannian manifolds (MMGLMs) in Chapter 3. We will extend these with Dirichlet Process. Let us revisit the well-known multivariate general linear model (MGLM in Euclidean space). Given pairs of covariates $\boldsymbol{x}_i \in \mathbf{R}^d$ and response variables $\boldsymbol{y}_i \in \mathbf{R}^{d'}$, we solve,

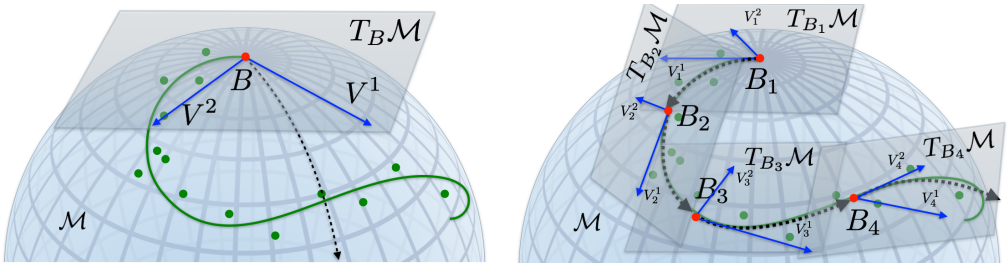


Figure 5.1: Comparison between MMGLM vs DP-MMGLM. MMGLM may not be capable to capture highly nonlinear patterns whereas DP-MMGLM learns the patterns with infinite mixtures of MMGLMs.

$\boldsymbol{y}_i = \boldsymbol{\beta}^0 + \boldsymbol{\beta}^1 x_i^1 + \ldots + \boldsymbol{\beta}^d x_i^d + \epsilon$, where $\{\boldsymbol{\beta}^j\}_{j=0}^d \subset \mathbf{R}^{d'}$ are regression coefficients. It is known that the MGLM model assumes that $\boldsymbol{x}_i$ the covariates relate to $\boldsymbol{y}_i$ the responses via a linear function. If desired, one may apply non-linearity to the output but this cannot be a direct function of the covariates. To address this limitation and allow the response to be non-linearly related to the covariates, we may write a modified version as,

$$\boldsymbol{y}_i = \boldsymbol{\beta}_i^0 + \boldsymbol{\beta}_i^1 x_i^1 + \boldsymbol{\beta}_i^2 x_i^2 + \ldots + \boldsymbol{\beta}_i^d x_i^d + \epsilon \tag{5.1}$$

where $\{\boldsymbol{\beta}_i^j\}_{j=0}^d \subset \mathbf{R}^{d'}$ are the unknown regression coefficients for each i. In this formulation, we allow each instance to have its own regression parameters, which offers advantages but creates an overfitting problem. The main flexibility offered by (5.1) is that the nonlinearity can be achieved by a mixture of an infinite number of linear models. On the other hand, fitting this model is ill-posed unless the regression parameters are constrained. Fortunately, the latter issue can be addressed by imposing a Dirichlet process (DP) prior as in (Hannah et al., 2011; Zhang et al., 2014). The DP mixture model is given by

$$(\boldsymbol{x}_i, \boldsymbol{y}_i)|\theta_i \sim F(\theta_i),\ \theta_i|G \sim G,\ G \sim DP(G_0, \nu). \tag{5.2}$$

where G_0 is a base distribution and ν is a concentration parameter. Using (5.2), a DP mixture of multivariate general linear models (DP-MGLM) is simply obtained by plugging in a d'-dimensional response Y into a DP mixture of generalized linear models (DP-GLM) studied in (Hannah et al., 2011; Mukhopadhyay and Gelfand, 1997). Specifically, we assume that the covariates X are modeled by a mixture of normal distributions, and that the responses Y are modeled by MGLMs conditioned on the covariates. The models are connected by associating a set of MGLM coefficients θ_y with each mixture component θ_x. Let $\theta = (\theta_x, \theta_y)$ be the set of parameters over X and $Y|X$, and let G_0 be a base distribution on θ. Then the DP-MGLM model, a special case of (Hannah et al., 2011), is given by,

$$\begin{aligned} \boldsymbol{y}_i|\boldsymbol{x}_i, \theta_{y_i} &\sim \mathcal{N}(\hat{\boldsymbol{y}}_i, \sigma_y^2), \text{ where } \hat{\boldsymbol{y}}_i = MGLM(\boldsymbol{x}_i, \theta_{y_i}) \\ \boldsymbol{x}_i|\theta_{x_i} &\sim \mathcal{N}(\boldsymbol{\mu}_{x_i}, \sigma_{x_i}^2), \text{ where } \theta_{x_i} = (\boldsymbol{\mu}_{x_i}, \sigma_{x_i}^2) \\ \theta_i|G &\sim G, \; G \sim DP(G_0, \nu), \text{ where } \theta_i = (\theta_{x_i}, \theta_{y_i}). \end{aligned} \tag{5.3}$$

What if Y is manifold-valued? Observe that the MGLM in (5.3) assumes that the response variable Y is in a *vector space*. It ignores the underlying intrinsic geometry of the manifold-valued data. As earlier chapters, we will provide intrinsic models and tailored schemes to estimate/sample parameters respecting the intrinsic geometry of structured data and parameter spaces.

5.2 DP-MMGLM on Riemannian manifolds

The basic component of DP-MMGLM is the manifold-valued multivariate general linear model (MMGLM) introduced in Chapter 3 which is given

by

$$\boldsymbol{y} = \text{Exp}(\text{Exp}(B, \sum_{j=1}^{d} V^j x^j), \epsilon), \tag{5.4}$$

where $B \in \mathcal{M}$ is an anchor (base) point and $\{V^j\}_{j=1}^d \subset T_B\mathcal{M}$ denote tangent vectors. They correspond to $\boldsymbol{\beta}^0$ and $\{\boldsymbol{\beta}^j\}_{j=1}^d$ resp. in (5.1). As described above for the Euclidean case, DP-MGLM on Riemannian manifolds also allows each example i to have its own regression parameters. That is, each example $(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathbf{R}^d \times \mathcal{M}$ has parameters $(B_i, \boldsymbol{V}_i)$. To reduce notational clutter, we will use the shorthand $\boldsymbol{V}\boldsymbol{x} := \sum_{j=1}^d \boldsymbol{V}^j x^j$, where $\boldsymbol{x} \in \mathbf{R}^d$.

In this section, we specify an end-to-end model for DP-MMGLM on the SPD manifold. To do this, we need a few key technical ingredients:

Step (a). First, we need to model the cluster of covariates, X which follows from an adaptation of existing work on DP-GLM (Hannah et al., 2011).

Step (b). Next, we need to characterize the conditional distribution $\mathbb{P}(y|x)$ specifically for the case where $y \in \text{SPD}(n)$. This requires two key steps. **i)** We need to specify the parameters for DP-MMGLM for the SPD manifold setting. In particular, we should identify which space (i.e., the manifold) each parameter corresponds to when $y \in \text{SPD}(n)$. **ii)** Then, we must make appropriate distributional assumptions for the respective spaces so that the follow-up inference scheme is both statistically sound and computationally feasible.

We first discuss Step (a). To model the relationship between x and y, we non-parametrically model the joint distribution $\mathbb{P}(x, y|\theta) = \mathbb{P}(y|x, \theta)\mathbb{P}(x|\theta)$, using a Dirichlet process mixture (θ is a cluster model parameter). Within each cluster, the relationship between y and x is expressed using an MMGLM. Note that the covariates X live in a Euclidean space $\mathbf{R}^d$. The parameters for X are $\theta_x = (\boldsymbol{\mu}_x, \boldsymbol{\sigma}_x^2)$, same as in (5.3). So, we can model a cluster of covariates X by a Gaussian distribution with parameters $(\boldsymbol{\mu}_x, \boldsymbol{\sigma}_x^2)$. The prior for these parameters is given by a DP-prior.

We now describe Step (b). For the Riemannian setting, we first give the corresponding expression for (5.3) for parameters of the MMGLM, i.e., $\theta_y = (B, \boldsymbol{V})$, where $B \in \text{SPD}(n)$ and $\boldsymbol{V} \in \text{Sym}(n)^d$. Here, $\text{Sym}(n)$ denotes the space of $n \times n$ symmetric matrices and we have d separate V's in $\boldsymbol{V}$. Recall that in a GLM, noise is modeled as a Normal distribution so that the Maximum Likelihood estimate (MLE) minimizes the least squares error. In the current setting, ideally, the MLE must minimize the geodesic distance-based error. So, we need an analogous form (for the Normal distribution) for manifold-valued y's. The solution to this is to use the "generalized Normal" distribution on the manifold (Cheng and Vemuri, 2013). Then, the maximum likelihood estimator of the MGLM turns out to be equivalent to the minimization of a least squares geodesic-distance error, given the covariance parameter σ_y^2. In the next section, we will discuss explicit forms of the density function of the generalized Normal distribution and the equivalence between the log likelihood function and squared geodesic error. So, the joint distribution in one cluster, i.e., $F(\theta_i)$ in (5.2), is given by,

$$\begin{aligned} Y_i|\boldsymbol{x}_i, \theta_{y_i} &\sim \mathcal{N}_{\text{SPD}}(\hat{Y}_i, \sigma_y^2), \text{where } \hat{Y}_i = \text{Exp}(B_i, \boldsymbol{V}_i \boldsymbol{x}_i) \\ \boldsymbol{x}_i|\theta_{x_i} &\sim \mathcal{N}(\boldsymbol{\mu}_{x_i}, \boldsymbol{\sigma}_{x_i}^2), \text{ where } \theta_{\boldsymbol{x}_i} = (\boldsymbol{\mu}_{\boldsymbol{x}_i}, \boldsymbol{\sigma}_{x_i}^2) \end{aligned} \tag{5.5}$$

where, $\mathcal{N}$ is a Normal distribution for $\boldsymbol{x} \in \mathbf{R}^d$, and $\mathcal{N}_{\text{SPD}}$ denotes the "generalized Normal" distribution for $Y \in \text{SPD}(n)$. The next step is to define the base distribution G_0 over $\theta = (\boldsymbol{\mu}_x, \boldsymbol{\sigma}_x^2, B, \boldsymbol{V})$ where σ_y is assumed to be given (or empirically estimated).

$$\begin{aligned} \boldsymbol{\mu}_x|\boldsymbol{\mu}_0, \sigma_0 &\sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\sigma}_0^2), \ \log(\boldsymbol{\sigma}_x^2)|M_\sigma, \Sigma_\sigma \sim \mathcal{N}(M_\sigma, \Sigma_\sigma^2) \\ B &\sim \mathcal{N}_{\text{SPD}}(\mu_B, \sigma_B^2), \ \boldsymbol{V} \sim \mathcal{N}_{\text{Sym}}(\mu_V, \sigma_V^2)^d, \end{aligned} \tag{5.6}$$

where $\mathcal{N}_{\text{Sym}}$ is a symmetric matrix-variate Normal distribution over $V \in \text{Sym}(n)$ defined later in (5.8). To make it analytically feasible, we use a

Normal (or log normal) distribution.

Remark. For a SPD matrix-valued variable B, other distributions such as log normal, Wishart or inverse Wishart distribution can also be used within G_0. However, these distributions do not necessarily yield a sample B around mean or mode of the distribution with respect to a GL-invariant metric. So, if one has knowledge of a highly probable B (e.g., the Fréchet mean) and its neighbors w.r.t. the geodesic distance, then a log Normal or the generalized Normal distribution in (5.9) is more suitable. Using a log Normal distribution is useful because it is easier to sample (compared to generalized Normal). However, the Jacobian of the matrix exponential varies as a function of the sample location, which makes it harder to deal with the derivative of its log likelihood. We provide candidate distributions for the base distribution over $\mathrm{Sym}(n)$ and $\mathrm{SPD}(n)$ and the corresponding density functions and their log likelihood in Chapter A.1 which are useful in deriving the final HMC algorithm.

5.3 Posterior Sampling

fletcher2013geodesicIn this section, we describe our proposed method for posterior inference. To place our contribution in context, we first summarize the conventional approach and then the key modifications needed.

If the base measure G_0 is *conjugate*, then it yields an efficient sampling procedure called the "collapsed Gibbs sampling" (Neal, 2000). Unfortunately, the distributions in (5.6) are not known to be conjugate. To address the above problem, we instead use Gibbs sampling with auxiliary parameters by adapting Algorithm 8 in (Neal, 2000). This requires sampling cluster parameters for each cluster such that the distribution remains invariant — in our setting, this is simpler for $\theta_x = (\boldsymbol{\mu}_x, \sigma_x^2)$ but more involved for $\theta_y = (B, \boldsymbol{V})$. For θ_x, we use a simple slice sampling for updat-

ing the parameters (Neal, 2000). Updating the regression parameters, θ_y is more challenging. This is because while slice sampling can be performed for each dimension independently, this is not true for the manifold-valued B. So, for a more effective sampling, we generalize the HMC method, used in Dirichlet process mixtures of multinomial logit model (dpMNL), a special case of DP-GLM (Hannah et al., 2011; Shahbaba and Neal, 2009).

The HMC algorithm needs to be generalized for the MGLM on Riemannian manifolds. Note that our formulation here is distinct from the Riemann manifold Langevin and Hamiltonian Monte Carlo (RMHMC) technique in (Girolami and Calderhead, 2011), which is Riemannian in the sense that it treats the joint probability space of the data as a Riemannian manifold. This is done by defining a Riemmanian metric (e.g., the Fisher-Rao metric) and the negative Hessian of the log-prior. However, the *data* itself are *not* assumed to lie on a manifold.

When the parameters lie in Euclidean space. Recall that conventional rejection sampling (such as Metropolis-Hastings) suffers from a low acceptance rate. However, HMC provides an ergodic Markov chain capable of achieving both large transitions and high acceptance rate. The underlying theory of HMC relies on Hamiltonian dynamics. Hamiltonian dynamics operates on a d-dimensional position vector q and a d-dimensional momentum vector p, so that the full state space has $2d$-dimensions. For HMC, we usually use Hamiltonian functions written as $H(q,p) = U(q) + K(p)$. Here, $U(q)$ is the *potential energy* and $K(p)$ is the *kinetic energy*. Generally, the posterior distribution for the model parameters is the usual object of interest and hence these parameters take the role of the position, q. The potential energy is $U(q) = \log[\pi(q)L(q|D)]$, where $\pi(q)$ is the prior density, and $L(q|D)$ is the likelihood function, given the data D. The kinetic energy is defined by $K(p) = p^T M^{-1} p/2$, where, p is the auxiliary variable which can be interpreted as momentum and M is the "mass matrix". HMC proposes transitions $\theta \rightarrow \theta^*$, which are then accepted with probability

based on Hamiltonian functions $\min\{1, \exp(H(q,p) - H(q^*,p^*)\}$, where q^* and p^* are proposed parameters and their momentum respectively (Neal, 2011).

Manifold setting. Defining the *potential energy* function for the HMC algorithm is simple – we can use the negative log of the joint probability. To define the *kinetic energy*, we must account for manifold-valued parameters; $B \in \mathcal{M}$ for the *intercept* and a set of tangent vectors $\boldsymbol{V}$ for the *slope*. To this end, the following description provides solutions to the main questions, **(a)** How to define the change of parameters B and $\boldsymbol{V}$? **(b)** How to update the parameters? **(c)** How to transport objects (such as momentum) to the appropriate tangent space? **(d)** How to sample the initial momentum?

First, we define the potential energy. To do so, we introduce the explicit form of probability density functions. The density function of the Normal distribution as a prior over $\text{Sym}(n)$ (definition 3.1.3) in (Schwartzman, 2006) is

$$f_{\text{Sym}}(V; \mu_V, B) = \quad \frac{1}{Z} \exp\left(-\frac{1}{2}\text{tr}[((V - \mu_V)B^{-1})^2]\right) \tag{5.7}$$

where $Z = (2\pi)^{q/2}|B|^{(n+1)/2}$, $|B|$ is the determinant of B and $q = n(n+1)/2$. Also, the simpler version (definition 3.1.4) in (Schwartzman, 2006) is

$$f_{\text{Sym}}(V; \mu_V, \sigma^2) = \quad \frac{1}{(2\pi)^{q/2}\sigma^q} \exp\left(-\frac{1}{2\sigma^2}\text{tr}[(V - \mu_V)^2]\right). \tag{5.8}$$

Next, to define the likelihood of $y \in \text{SPD}$, we introduce the generalized Normal distribution.

$$f_{\text{SPD}}(y; \mu_y, \sigma_y^2) = \frac{1}{Z(\mu_y, \sigma_y)} \exp\left(-\frac{d(y, \mu_y)^2}{2\sigma^2}\right) \tag{5.9}$$

where $Z(\mu_y, \sigma_y) = \int_{\mathcal{M}} \exp\left(-\frac{d(y,\mu_y)^2}{2\sigma_y^2}\right) dy$. Here, it turns out that $Z(\mu_y, \sigma_y)$ is constant w.r.t. μ when $\mathcal{M}$ is a symmetric space (Fletcher, 2013). So, the negative log-likelihood of each cluster c takes the form,

$$-\log \mathcal{L}(\theta_c^*|D_c) = n_c \log Z(\sigma_y) + \frac{1}{2\sigma_y^2} \sum_{i \in c} d(y_i, \hat{y}_i)^2 \tag{5.10}$$

where $\hat{y}_i = \mathrm{Exp}(B, \boldsymbol{V} x_i)$, c is a cluster, n_c is the number of its elements. Interestingly, because the normalization factor is constant, maximizing the log likelihood reduces to minimizing the least squares error. We can now define our potential function as

$$U(B, \boldsymbol{V}) := \frac{1}{\sigma^2} E(B, \boldsymbol{V}) - \log f_{\mathrm{SPD}}(B) - \log f_{\mathrm{Sym}}(\boldsymbol{V}) \tag{5.11}$$

where $E(B, \boldsymbol{V}) := \frac{1}{2} \sum_i d(y_i, \hat{y}_i)^2$. Our kinetic energy is given by

$$K(\dot{B}, \dot{\boldsymbol{V}}) := \frac{1}{2} \|\dot{B}\|_B + \frac{1}{2} \sum_{j=1}^{d} \|\dot{\boldsymbol{V}}^j\|_B \tag{5.12}$$

where the covariate is in $\mathbf{R}^d$.

We must now account for the change of parameters. Notice that the change of manifold valued $B \in \mathcal{M}$ is represented by a tangent vector $\dot{B} \in T_B\mathcal{M}$. However, the change of tangent vectors, $\dot{V}$, live in $T_V(T_B\mathcal{M})$ (a tangent space of a tangent space). Fortunately, the natural isomorphism $T_V(T_B\mathcal{M}) \cong T_B\mathcal{M}$ allows us to let $\dot{V}$ be in $T_B\mathcal{M}$. By construction, the priors for B and $\boldsymbol{V}$ are Gaussian and so the log of the prior density functions are quadratic forms whose derivatives can be obtained analytically. These

are given by,

$$\begin{aligned}\nabla_B U &\approx -\frac{1}{\sigma_y^2}\sum_{i=1}^{N}\Gamma_{\hat{y}_i\to B}\mathrm{Log}(\hat{y}_i, y_i) - \nabla_B \log f_{\mathrm{SPD}}(B)\\ \nabla_{V^j} U &\approx -\frac{1}{\sigma_y^2}\sum_{i=1}^{N} x_i^j\Gamma_{\hat{y}_i\to B}\mathrm{Log}(\hat{y}_i, y_i) - \nabla_{V^j} \log f_{\mathrm{Sym}}(V^j)\end{aligned} \tag{5.13}$$

where Γ is the parallel transport operation. More details on prior distributions and their derivatives are available in Chapter A.1.

Remarks. The least squares loss function is defined on a SPD manifold. If one uses the prior distribution over B which is defined in a Euclidean space instead of the generalized Normal distribution we use, then the gradient with respect to B needs to be separated into the derivative, $\nabla_B E$, along the curved surface (called covariant derivative) and the derivative along the ambient space $\nabla_B \log f_B$. Technically, these are not in the same space, which can verified by comparing their respective update schemes. For instance, the next iterate B via $\nabla_B E$ is $\mathrm{Exp}(B, \epsilon\nabla_B E)$ whereas the next iterate B suggested by $\nabla_B \log f_B$ is $B = B + \nabla_B \log f_B$. Fortunately, for V, the update schemes are identical. Both use the simple addition operation since $\nabla_{V^j} E$ and $\nabla_{V^j} \log f_{V^j}$ lie in vector spaces. A minor issue here is that their scales might be different since $\nabla_{V^j} E$ lies in $T_B\mathcal{M}$ with a locally defined inner product $\langle U, B\rangle_B = \mathrm{tr}(UB^{-1}VB^{-1})$ whereas $\nabla_{V^j} \log f_{V^j} \in \mathrm{Sym}(n)$ with the natural inner product $\langle U, V\rangle = \mathrm{tr}(UV)$ in Euclidean space where a symmetric matrix-variate normal distribution (5.8) is defined. In addition, there is no reason to expect that the samples drawn from this distribution in (5.8) are normally distributed in an arbitrary tangent space at B with respect to the GL-invariant metric. We provide a cleaner solution next.

5.3.1 Defining an alternative distribution for both the base point B and a set of tangent vectors V

As a solution, we propose a new distribution for $(B, V) \in \mathcal{M} \times T_B\mathcal{M}$ by conditionally combining two distributions.

$$B|\mu_B, \sigma_B^2 \sim \mathcal{N}_{\text{SPD}}(B|\mu_B, \sigma_B^2), V|\mu_V, B \sim \mathcal{N}_{\text{Sym}}(V|\mu_V, B) \tag{5.14}$$

Lemma 5.1 shows that the distribution in (5.14) is more of a "Normal like" distribution for both B and V w.r.t a GL-invariant metric.

Lemma 5.1. *Let $(B, V) \in SPD(n) \times Sym(n)$ be a sample drawn using (5.14), then V is Normally distributed w.r.t. a GL-invariant metric at the tangent space $T_B\mathcal{M}$ at B. For each B, the probability density function of V is proportional to $\exp(-\frac{1}{2}\|V\|_B^2))$ at $T_B\mathcal{M}$, when $\mu_V = 0$.*

Proof. We will derive an expression for the density. By inspection, we have

$$\int\int f(B; \mu_B, \sigma_B^2) f(V; \mu_V, B) dV dB = \int f(B; \mu_B, \sigma_B^2) \left[\int f(V; \mu_V, B) dV\right] dB = 1$$

Let $q = n(n+1)/2$. Given the density functions Eq. (7) and (8) in the main paper, the density of the proposed distribution $f_{\text{SPD,Sym}}((B, V)|\mu_B, \sigma_B^2, \mu_V)$ is the product of density functions given by

$$\begin{aligned}
&f((B, V)|\mu_B, \sigma_B^2, \mu_V) \\
&= \frac{1}{Z(\mu_B, \sigma_B^2)} \exp\left(-\frac{1}{2\sigma_B^2} \mathrm{d}(B, \mu_B)^2\right) \frac{1}{(2\pi)^{q/2}|B|^{(n+1)/2}} \exp\left(-\frac{1}{2}\mathrm{tr}[((V - \mu_V)B^{-1})^2]\right) \\
&= \frac{1}{Z(\mu_B, \sigma_B^2)} \exp\left(-\frac{1}{2\sigma_B^2} \mathrm{d}(B, \mu_B)^2\right) \frac{1}{(2\pi)^{q/2}|B|^{(n+1)/2}} \exp\left(-\frac{1}{2}\|V - \mu_V\|_B^2\right) \\
&= f(B; \mu_B, \sigma_B^2) \frac{1}{(2\pi)^{q/2}|B|^{(n+1)/2}} \exp\left(-\frac{1}{2}\|V\|_B^2\right), \text{ when } \mu_V = 0 \in \text{Sym}(n)
\end{aligned} \tag{5.15}$$

Algorithm 7 HMC algorithm for DP-MGLM on Riemannian manifolds

Input: $(B_{cur}, \boldsymbol{V}_{cur}) \in \mathcal{M} \times T_B\mathcal{M}^n$, Leapfrog (or step) size $\epsilon \in \mathbf{R}_{++}, L \in \mathbf{Z}_{++}$
Output: $(B_{next}, \boldsymbol{V}_{next}) \in \mathcal{M} \times T_B\mathcal{M}^n$
Sample $(\dot{B}_{cur}, \dot{\boldsymbol{V}}_{cur}) \in T_B\mathcal{M} \times T_B\mathcal{M}^n$ from independent normal distribution w.r.t. Riemannian metric.
Initialize $(B, \boldsymbol{V}, \dot{B}, \dot{\boldsymbol{V}}) \leftarrow (B_{cur}, \boldsymbol{V}_{cur}, \dot{B}_{cur}, \dot{\boldsymbol{V}}_{cur})$
$\dot{B} \leftarrow \dot{B} - \frac{\epsilon}{2}\nabla_B U(B, \boldsymbol{V})$ and $\dot{\boldsymbol{V}} \leftarrow \dot{\boldsymbol{V}} - \frac{\epsilon}{2}\nabla_{\boldsymbol{V}} U(B, \boldsymbol{V})$
for $i \in \{1, \cdots, L\}$ **do**
    $B' \leftarrow B$, $B \leftarrow \mathrm{Exp}(B, \epsilon\dot{B})$, $V \leftarrow V + \epsilon\dot{V}$
    $(\boldsymbol{V}, \dot{B}, \dot{\boldsymbol{V}}) \leftarrow (\Gamma_{B'\to B}\boldsymbol{V}, \Gamma_{B'\to B}\dot{B}, \Gamma_{B'\to B}\dot{\boldsymbol{V}})$ /* Parallel transport */
    if i !=L **then**
        $\dot{B} \leftarrow \dot{B} - \epsilon\nabla_B U(B, \boldsymbol{V})$ and $\dot{\boldsymbol{V}} \leftarrow \dot{\boldsymbol{V}} - \epsilon\nabla_{\boldsymbol{V}} U(B, \boldsymbol{V})$
    end if
end for
$\dot{B} \leftarrow \dot{B} - \frac{\epsilon}{2}\nabla_B U(B, \boldsymbol{V})$ and $\dot{\boldsymbol{V}} \leftarrow \dot{\boldsymbol{V}} - \frac{\epsilon}{2}\nabla_{\boldsymbol{V}} U(B, \boldsymbol{V})$
Accept $(B, \boldsymbol{V})$ with probability
$\min[1, \exp(H(\dot{B}_{cur}, \dot{\boldsymbol{V}}_{cur}, B_{cur}, \boldsymbol{V}_{cur}) - H(\dot{B}, \dot{\boldsymbol{V}}, B, \boldsymbol{V}))]$

where the inner product of $U, V \in T_B\mathcal{M}$ is $\langle U, V\rangle_B = \mathrm{tr}(B^{-1/2}UB^{-1}VB^{-1/2})$. □

Note that it is not exactly a Normal distribution because of the dependence on $|B|$. With these components, our final **HMC algorithm** is given by Algorithm 7.

Some additional details. We use the exponential map for parameter updates for $B \in \mathrm{SPD}(n)$. For all parameters in the vector space $(T_B\mathcal{M})$, the vector addition operation suffices. However, once the base point B_{old} changes to a new B, then the objects $\dot{B}, \dot{V}, V$ do not belong to the tangent space of B anymore. So, they need to be parallel transported from the old anchor point B_{old} to the new anchor point B. Then, the kinetic energy at each time point can be properly measured by the sum of squared norms of the tangent vectors in the new tangent space at B. Finally, we point out that the initial momentum is set by finding a random direction in the tangent space at B; its magnitude is given by the length w.r.t. the Riemannian inner product. Let D denote the measurements (or data). For the prediction of response Y, the conditional distribution

of $Y|X = x, D$ is $f(Y|X = x, D) \approx \frac{1}{S}\sum_{s=1}^{S} f(Y|X = x, \theta^{(s)})$. Thus, the prediction $\mathbb{E}[Y|X = x, D] = \mathbb{E}[\mathbb{E}[Y|X = x, \theta]|D]$ is approximated by the posterior samples $\{\theta^{(s)}\}_{s=1}^{S}$. Since Y is on $\mathcal{M}$, the expectation is the Fréchet mean. This can be updated in an online manner for the SPD manifold (Ho et al., 2013).

5.4 Experiments

To evaluate the proposed model, we conducted a set of experiments on synthetic and real-world data.

5.4.1 Experiments on synthetic data

Comparison between DP-MMGLM and MMGLM on synthetic data. We first evaluate whether our algorithm can simultaneously find a set of geodesic relationships between the covariates and the manifold-valued response variables. We follow the experimental protocol from (Hannah et al., 2011) which is broadly used in the literature, but with the distinction that now we have $Y \in \mathrm{SPD}(n)$. To do this, we simulate data from multiple geodesic curves which are parameterized by the covariates — this gives heteroscedasticity properties where DP-GLM approaches are known to be effective. The number of "local" models in this synthetic data varies between 2 to 5. Our sample size is 300. We perform a few hundred realizations where the number of MCMC samples in each realization is 1000. We set the burn-in period to 100 epochs. When the data is sampled from a single local model, one should expect both MMGLMs and DP-MMGLMs to perform well and estimate the parameters correctly. However, when the samples are drawn from a mixture of multiple local models, the flexibility offered by our framework must yield improvements. Since visualizing the model fit on the SPD manifold is not possible, we perform a Principal Geodesic analysis (PGA) to pick a prominent direction

of variance and project the original data onto this axis for evaluation. As shown in Fig. 5.2, in nearly all cases, the model provides a good fit and is able to identify a very good estimate of the real local relationships in the data, exactly as desired.

Since data is generated from multiple local models, DP-MMGLM yields significant improvements as we expected. As shown in Fig. 5.2 (multiple datasets), in nearly all cases, DP-MMGLM provides a good fit and is able to identify a very good estimate of the real local relationships in the data, exactly as desired. For visualizing the model fit on SPD manifolds, we project the data onto a prominent direction by Principal Geodesic analysis (PGA).

Estimating Models for Spatially-based Covariates. A number of applications motivating the need for statistics on manifold-valued responses come from image analysis. To evaluate our model in this setup, we synthesized an experiment where the responses form a distribution on SPD whereas the corresponding covariates are grid points on an image lattice. The ability to estimate such models faithfully offers numerous advantages including clustering and the ability to draw samples from the estimated model, e.g., for performing downstream hypothesis tests. We test these scenarios next in the context of estimating $\mathbb{E}(y|x)$.

Table 5.1: Mean squared errors and R-squared (R^2) statistic w.r.t the intrinsic metric on SPD(3) for eight synthetic datasets. MGLMc denotes MGLM with centered covariate $\boldsymbol{x}$.

	Mean Squared Error		R^2	
Model	Train	Test	Train	Test
DP-MGLM	$\mathbf{1}.18 \pm 0.99$	$\mathbf{1}.19 \pm 1.04$	$\mathbf{0}.80 \pm 0.06$	$\mathbf{0}.79 \pm 0.08$
MGLMc	3.40 ± 2.43	3.28 ± 2.14	0.39 ± 0.16	0.38 ± 0.16
MGLM	4.94 ± 3.40	4.80 ± 3.09	0.10 ± 0.04	0.10 ± 0.04

Our generating function is a mixture of models with spatially localized support. Each voxel is a manifold-valued measurement $Y \in \mathrm{SPD}(3)$ (such as in diffusion tensor imaging) whose grid locations are the covariates. For ease of visual assessment, each perceptual region in Fig. 5.3 (left column) is generated by a single function.

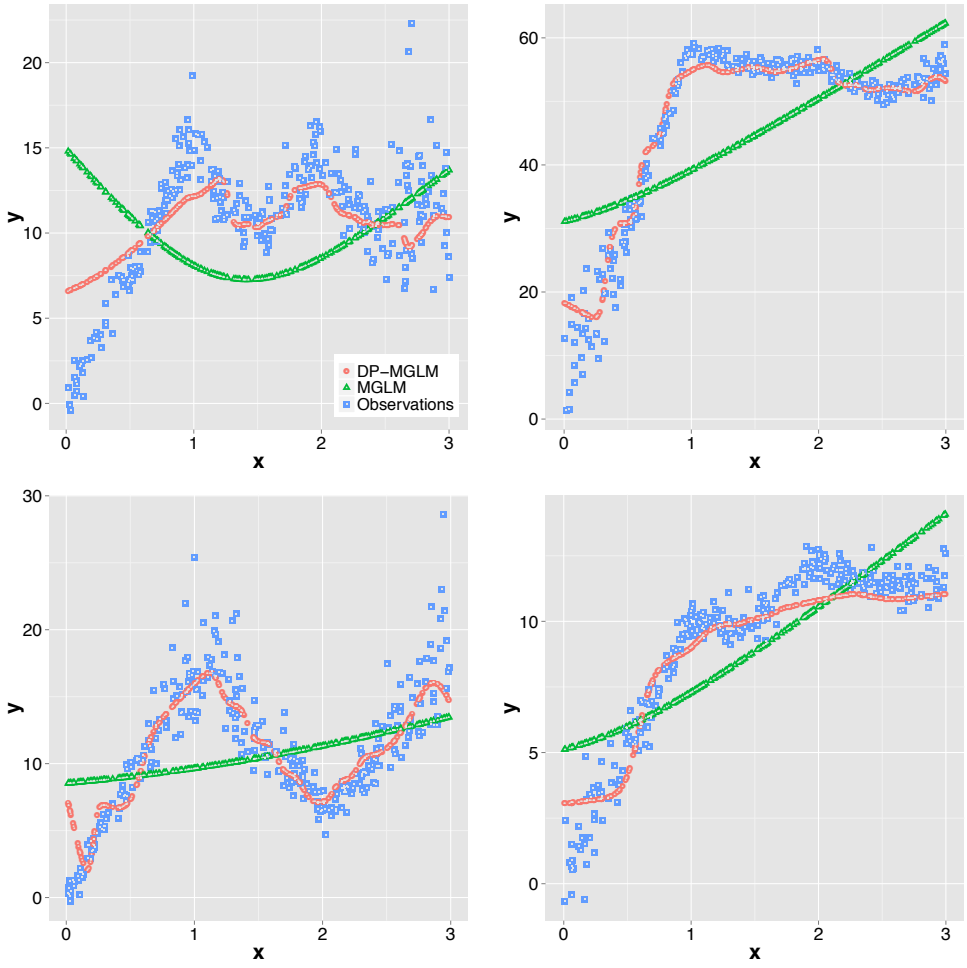


Figure 5.2: The figure shows the models fitted in the PGA axes space versus the covariates. The prediction of DP-MGLM (red) is shown using a single sample from the posterior, $\theta^{(i)}$. To visualize the response variable $Y \in \text{SPD}(3)$, we project the variables onto the axis obtained by PGA (y-axis). The x-axis is the covariate $x \in \mathbf{R}$. Green and blue correspond to our predictions of MGLM and the measurements respectively.

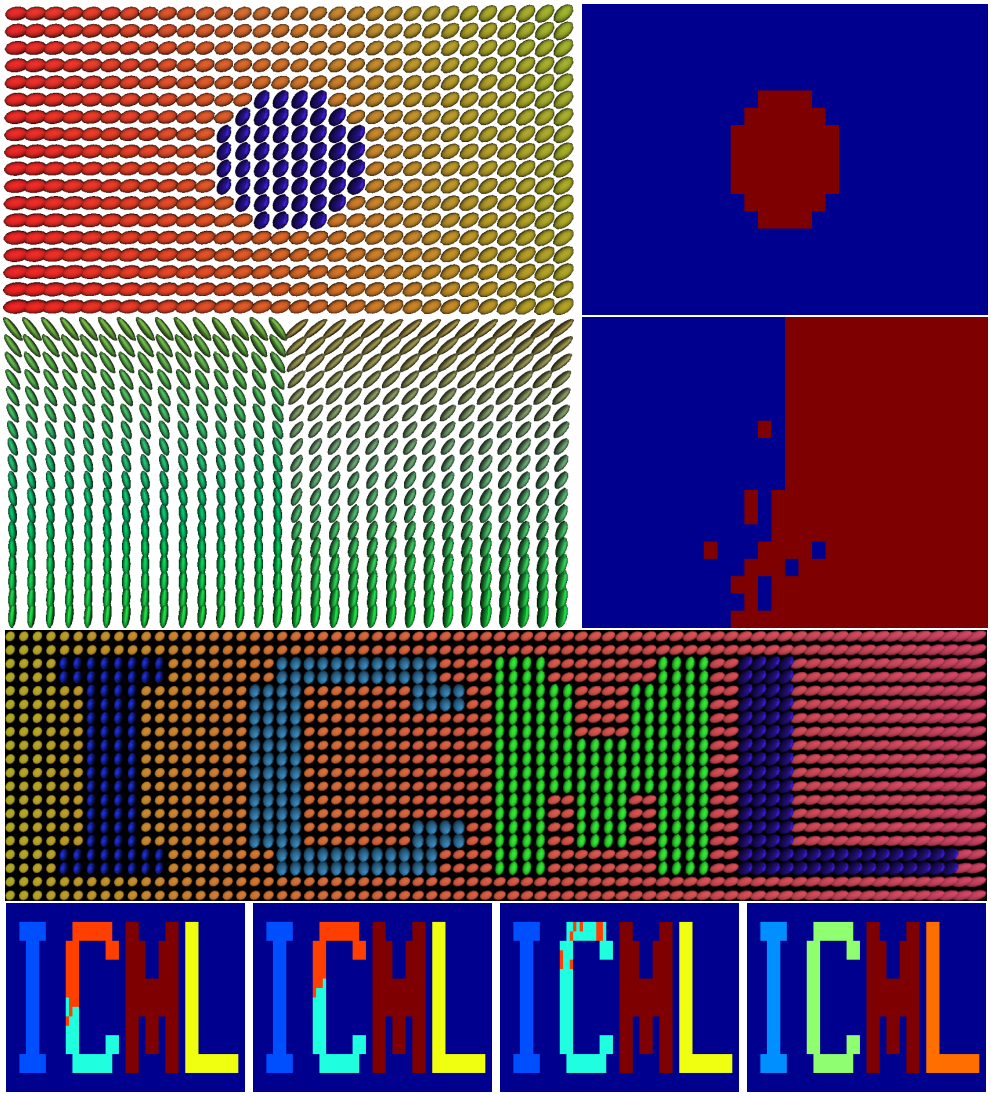


Figure 5.3: (Rows 1–2, col 1) Each voxel is a SPD(3) matrix; the covariates are the grid positions (horizontal, vertical coordinates). (Rows 1–2, col 2) shows a clustering result. (Row 3) is a glyph figure where the global mixture of local models is "ICML". (Row 4) A clustering based on the posterior samples $\theta^{(i)}$.

5.4.2 Experiments on real-world data

Next, we conduct an experiment on facial datasets which are derived from the important biometric task of face recognition and age estimation. In particular, we attempt to assess: how do facial landmark appearances evolve with age? Which age ranges/periods are most correlated with which face regions? This problem is important for facial age estimation (Guo et al., 2013). Since we expect that changes in different face regions will likely correspond to different age periods, it exhibits nice heteroscedasticity properties. We used the Lifespan database (Minear and Park, 2004), which contains 580 subjects with ages ranging from 18–93. To avoid the influence of facial expressions, we focus only on the "Neutral" subset which contains images without facial expressions and human labeled landmark points are provided (Guo et al., 2013). These include 40 points overall, see Fig. 5.4. We used the covariance descriptors common in image processing, computed from the feature vector $[r, c, R_{rc}, G_{rc}, B_{rc}, I_r, I_c]$, where r (and c) is row (and column) index, R, G, B are colors and I_r, I_c are intensity derivatives. The covariance matrix for an image patch (size 20×20) centered at each landmark is a 7×7 SPD, the response variable, $Y \in \mathcal{M}$. The age of the person associated with each image is the covariate, x.

We run Algorithm 7 on each landmark. The algorithm provides a set of local models for each landmark; here, these local models correspond to age ranges. In the manifold setting, each 'local' cluster (or model) can be interpreted as a geodesic explaining the relationship between the covariates (age range) and evolution in the covariance descriptor in that period. For each landmark, there are multiple clusters — we simply measure the length of the corresponding tangent vectors and pick the median as the representative. After normalization to $[0, 1]$, we show it as a color coded heat map in Fig. 5.4 shown in the bottom right of the figure. We see that our algorithm found that regions around the center of the eye (numbered as $2 \sim 5$, $7 \sim 10$) and nose ($27 \sim 29$) exhibit *no* meaningful

relationship with age (shown in blue). On the other hand, regions around the brow (12 ~ 18), cheeks (34 ~ 40) and forehead (21 ~ 23) exhibit a much *stronger* relationship (e.g., wrinkles) shown in red. This is consistent with prior findings (Montillo and Ling, 2009), which identified similar landmarks as the most distinguishing identifiers for age.

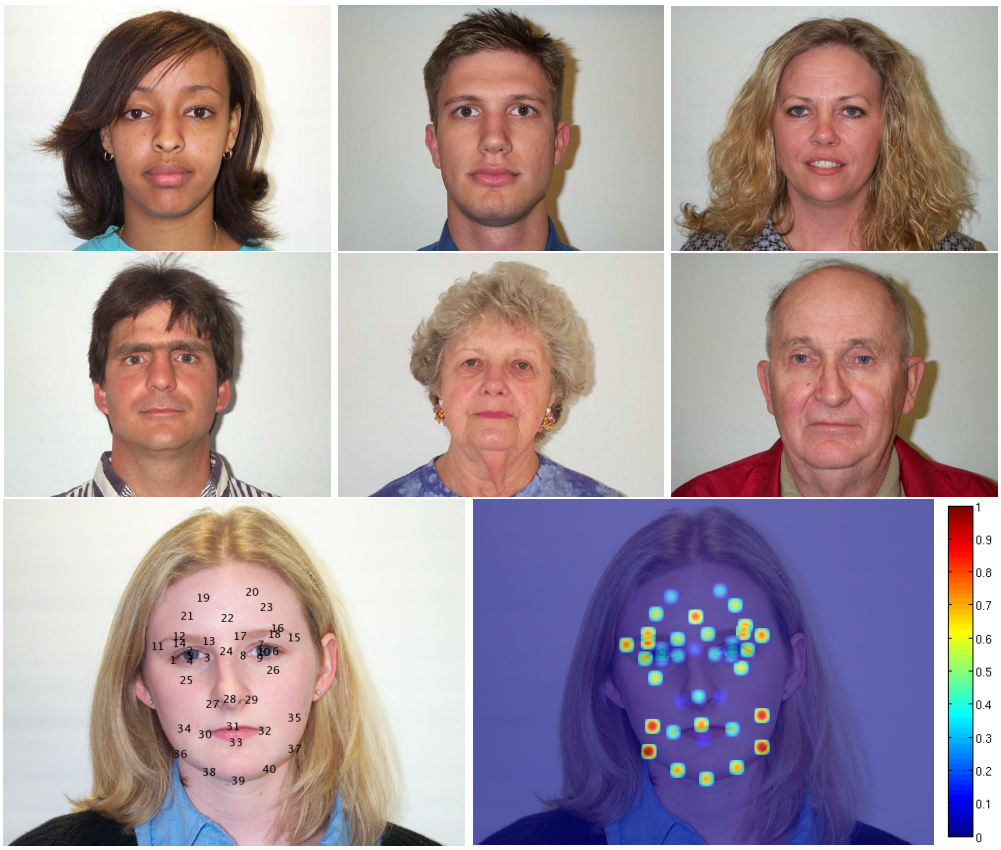


Figure 5.4: The top two rows show 6 sample faces with ages ranging from 20 ~ 80. The bottom row (left image) shows 40 landmarks (indexed by numbers) on an example image. The second image of the bottom row shows correlation magnitude of the landmark's variation with age as a heat map. **Best viewed in color.**

5.5 Summary

We have presented a novel algorithm for Dirichlet process mixtures of multivariate general linear models on Riemannian manifolds. The formulation globally extends the locally-defined parametric models on Riemannian manifolds using a mixture of local models, thereby solving the "locality" problem pervasive in various parametric formulations for a class of Riemannian manifolds. We derive specific sampling schemes for the SPD manifold but the ideas should apply to other manifolds with similar geometries (e.g., non-positively curved). We also studied and proposed a new distribution to get a pair of parameters for models on the SPD manifold and its tangent space. On the algorithm side, we derived a specialized HMC algorithm which efficiently estimates manifold-valued parameters, which may be of independent interest. While our development here is primarily on the theoretical side, we believe that the proposal will lead to practical sampling and inference schemes for various problems in medical imaging, machine learning and computer vision that involve statistical tasks on the SPD manifold. The code is publicly available [1].

[1] https://github.com/MLman/DP-MMGLM

6 RIEMANNIAN NONLINEAR MIXED EFFECTS MODELS

The aim of this chapter is to develop a model for longitudinal analysis of manifold-valued measurements. So far, manifold-valued regression models including Manifold-valued Multivariate General Linear Models in Chapter 3 assume that the samples are independent. These models are called "fixed effects models" since the model parameters are fixed (not random quantities). But in most longitudinal analysis settings, multiple samples are obtained from each subject at multiple time points. In this case, samples from one subject may not be independent and can be affected by "random effects" specific to the subject. So, the application of fixed effects models in this situation is problematic. In an effort to address this need, "mixed effects models" have been studied in the literature to capture both *fixed effects* and *random effects*. In this chapter, we generalize mixed effects models to the regime where the response variable is manifold-valued. We derive the underlying model (including estimation schemes) and demonstrate the immediate benefits such a model can provide – both for group level and individual level analysis on longitudinal brain imaging data. The direct consequence of our results is that longitudinal analysis of manifold-valued measurements (especially, the symmetric positive definite manifold) can be conducted in a computationally tractable manner.

6.1 Longitudinal analysis and random effects

Longitudinal analysis has been extensively studied by a variety of statistical learning models in the Euclidean space. However, such models are relatively less studied for structured measurements. In this chapter, we develop a longitudinal analysis method for structured measurements. As

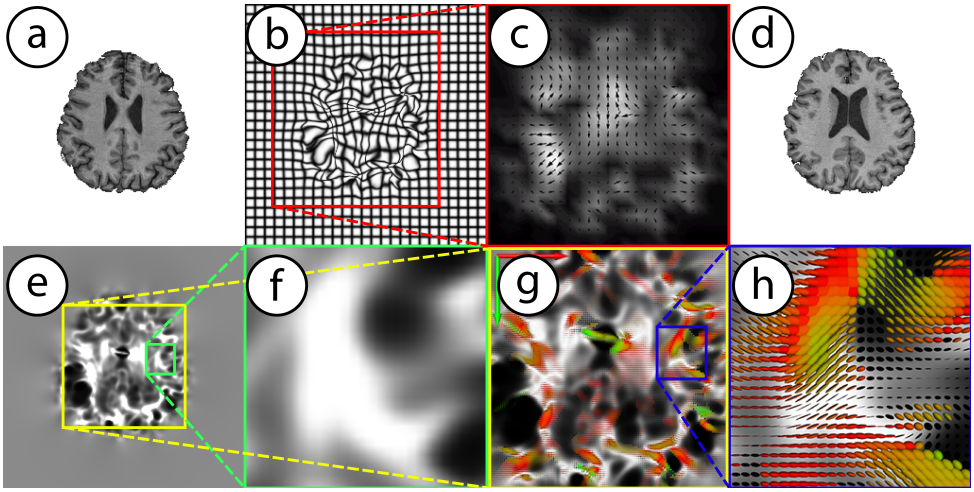


Figure 6.1: An example panel of data generated in morphometric studies. (a, d) The moving and fixed brain image respectively. (b) Warped spatial grid to move (a) to (d). (c) Vector field of local deformations. (e, f) A map of the $det(J)$ of the deformation field. (g, h) The Cauchy deformation tensor field (CDTs) ($\sqrt{J^T J}$). Among the different features of brain morphology that can be analyzed, CDTs are the focus of this chapter.

a motivating example, we discuss a longitudinal analysis task with structured measurements derived from a longitudinal neuroimaging study. But the proposed model here is applicable to any measurements on SPD(n).

In a longitudinal neuroimaging study of disease progression, the statistical models need to capture morphometric changes *over time* while controlling (or accounting) for the dependency of *repeated* measurements coming from the same subject. Consider the following analysis: we have two groups of subjects, "controls" and "disease" which correspond to healthy controls and individuals with a high risk of a disease. We want to identify group-differences with respect to time (or disease progression). It is known that anatomical changes at a voxel X can be captured by spatial derivatives, i.e., the Jacobian matrix $J(X)$, of deformation (or warping) maps of a subject (e.g., capturing changes between the first to the second time points, two years apart). The most widely used "deformation"

feature is the log determinant of the Jacobian matrices, $\log(\det(J(X)))$ — a scalar voxel-wise value which captures the volumetric/anatomical changes. The so-called Cauchy deformation tensor (CDT) (Lepore et al., 2008) represented as $\sqrt{J(X)^T J(X)}$ is a richer representation of $J(X)$, an object on the $\mathrm{SPD}(3)$ manifold, see Fig. 6.1. To understand how each voxel in the image is associated with a predictor variable (say, age or disease status), a regression between the predictor and the voxel-specific "response variable" $\sqrt{J(X)^T J(X)}$ will identify brain regions that are most affected by age or disease (via the calculated regression coefficients). This can be accomplished by generalizations of linear models on manifolds (Fletcher, 2013; Kim et al., 2014b; Cornea et al., 2016) introduced in Chapter 3, or kernel regression on manifolds (Banerjee et al., 2016).

Random effects in longitudinal analysis. Now, let us consider a slightly more involved setting where each subject provides data over multiple time points, a few years apart. In such a *longitudinal* setting, we obtain *one* CDT image (composed of CDTs at each voxel in the image) between each consecutive time point (i.e., pairs). A standard linear regression (or its manifold-valued analog) is *agnostic to dependency of temporal samples*. Since subjects are examined multiple times within the study, the repeated measurements from the same subject – commonly known as the subject specific "*random effect*". This dependency violates the i.i.d. assumptions of *fixed effects* models (e.g., generalized linear regression), including the manifold versions (Fletcher, 2013; Kim et al., 2014b; Cornea et al., 2016) discussed in Chapter 3. The fixed effects model assumes that all data are i.i.d. samples from the same underlying generating function with random noise on the response variable Y. As Fig. 6.2 shows, each subject may have a different trend. For example, subject A has an early disease onset (intercept). Subject B shows faster disease progression (slope). Also, based on the participants' age-range, there may be larger variability *between* subjects than the variability *within* a subject. So, a

fixed effects linear model for the data in Fig. 6.2, is *prone to fit population level variability (black) rather than the trajectory of each subject (red)*. Within statistical machine learning, such subject specific *random effects* can be modeled via the more general *mixed effects models*. The overarching **goal and contribution** of this chapter is to derive formulations/algorithms for the regime where the set of longitudinal responses Y is a manifold valued variable and the objective is to fit linear (or non-linear) mixed effects models. We note that regression models on manifolds were studied for the group of diffeomorphisms (Davis et al., 2007; Niethammer et al., 2011; Singh et al., 2013). This is relevant to how one models morphometric changes of brains. The closest work to the formulation proposed in this chapter is a recent independent result on mixed effects models in (Schiratti et al., 2015). This works deals with univariate manifolds $[0, 1]$, which is the unit interval in a real line $\mathbf{R}$ with a specifically designed metric to capture sigmoid function like patterns. The work is not directly applicable to multivariate manifold-valued variables (e.g., SPD); further it is computationally impractical for more than hundreds of voxels. In contrast, 3D CDT images we will analyze exceed 1M+ voxels.

6.2 Preliminary concepts and notations

We first briefly review *linear mixed effects models* and their estimation methods. Then, we introduce Cauchy deformation tensors and Jacobian matrices to capture longitudinal morphometric brain changes.

6.2.1 Euclidean Linear mixed effects model

In general, the estimation of regression models (such as linear/polynomial) assumes that the data come from an underlying model with i.i.d. noise; so the effects of the covariates/features are pertinent to the entire sample. These models are called *fixed effects*. For example, a linear

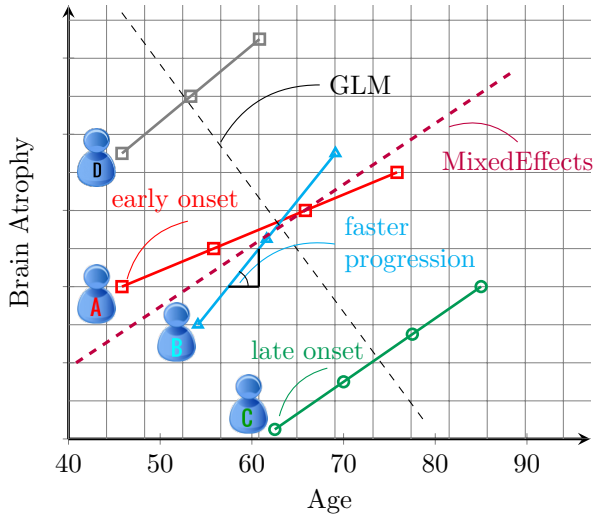


Figure 6.2: This figure demonstrates the key effects we are interested in capturing. Each subject has a different progression rate of the brain atrophy (acceleration effect) and has a different onset for the change (time shift). A regular general linear model (GLM) with fixed effects is insufficient to capture these effects in a regression framework while including random effects (subject-specific slope and intercept) in mixed effects models can capture these effects.

regression model is also a fixed effects model given as

$$y = \beta^0 + \beta^1 x^1 + \cdots + \beta^p x^p + \epsilon, \tag{6.1}$$

where $y \in \mathbf{R}$, $\boldsymbol{x} \in \mathbf{R}^p$, $\boldsymbol{\beta} = [\beta^0, \ldots, \beta^p]^T \in \mathbf{R}^{p+1}$. We see that the coefficients are 'fixed' and the same over the entire population. However, in longitudinal studies (see Fig. 6.2), the repeated measurements from the same subject are *no longer independent*. We need a more flexible specification – often covariates/features have different effects on individual subjects (or groups), which is called *random effects*. For example, the rate of brain atrophy and disease progression can vary over subjects given by

$$y_i = u_i^1 z^1 + \cdots + u_i^q z^q + \epsilon_i, \tag{6.2}$$

where $\mathbf{z}$ is a known vector specifying which subject (or group) a sample belongs to, and u_i^q is the q^{th} random effect for the i^{th} subject (or group) denoted by $\boldsymbol{u}_i$. This combination of *fixed* and *random* effects yields *mixed effects models* Laird and Ware (1982). When the model is linear, we get linear mixed effects models, which we introduce next. We then work with its *nonlinear* analog. The nonlinear mixed effects models are an intermediate (but necessary) step in deriving our final models for manifold-valued data, introduced in Sec. 6.3.2.

Specifying the model. Let $\boldsymbol{y}_i = \left[y_{[\![ij]\!]}\right]_{j=1}^{n_i}$ be a set of n_i repeated observations of a response or dependent variable for subject i. Here $\boldsymbol{y}_i$ is a n_i dimensional vector, vertically stacked with $y_{[\![ij]\!]}$ responses for subject i. The notation $[\![i, j]\!]$ simply recovers the specific observation j for subject i. Similarly, let the subject-specific matrix X_i of size $n_i \times p$ be setup as $\left[x_{[\![ij]\!]}^1\ x_{[\![ij]\!]}^2\ \cdots\ x_{[\![ij]\!]}^p\right]_{j=1}^{n_i}$ where we collect for subject i, all p measurements for all n_i visits as rows. The matrix Z_i will provide information on the number of longitudinal measurements for each subject (design matrix). Similar to X_i, we define Z_i by specifying rows as $Z_i = \left[z_{[\![ij]\!]}^1\ z_{[\![ij]\!]}^2\ \cdots\ z_{[\![ij]\!]}^q\right]_{j=1}^{n_i}$. These correspond to sets of p and q variables (features) for the i^{th} subject where one is interested in estimating fixed effects for the set X_i and random effects for the set Z_i on $\boldsymbol{y}_i$. In the classical setting, a linear mixed effects model (Laird and Ware (1982)) is given by

$$\begin{aligned} y_{[\![ij]\!]} =& \beta^0 + \beta^1 x_{[\![ij]\!]}^1 + \cdots + \beta^p x_{[\![ij]\!]}^p + \\ & u_i^1 z_{[\![ij]\!]}^1 + \cdots + u_i^q z_{[\![ij]\!]}^q + \epsilon_{[\![ij]\!]}, \end{aligned}$$

where $\beta^1, \ldots, \beta^p$ are the fixed effects shared over the entire population and $u_i^1, \ldots, u_i^q$ are the (subject-specific) random effects for the i^{th} subject. The random effects $\boldsymbol{u}_i = [u_i^1\ u_i^2\ \cdots\ u_i^q]^T$ are assumed to follow a multivariate normal distribution (zero mean and covariance matrix $\Sigma \in \mathbf{R}^{q \times q}$). The "unexplained" random error ϵ_i comes from a normal distribution

$\mathcal{N}(0, \Sigma^2_{\epsilon_i})$. We can compactly write the model using matrix notation as

$$\boldsymbol{y}_i = X_i\boldsymbol{\beta} + Z_i\boldsymbol{u}_i + \boldsymbol{\epsilon}_i. \tag{6.3}$$

Let 'vstack$(\cdot)$' be the vertical stack of parameters. By denoting $\boldsymbol{y} =$ vstack$(\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_N)$, and similarly $X, Z, \boldsymbol{u}$, the final model for all N subjects can be expressed as,

$$\boldsymbol{y} = X\boldsymbol{\beta} + Z\boldsymbol{u} + \sigma^2_\epsilon I,$$

where $\boldsymbol{u} \sim \mathcal{N}(0, \tilde{\Sigma})$ and $\tilde{\Sigma} = \text{diag}(\Sigma_1, \Sigma_2, \ldots, \Sigma_N) = \Sigma \otimes I$ (when $\Sigma_i = \Sigma \forall i$), and $Z = \text{diag}(Z_1, Z_2, \ldots, Z_N)$.

In general, estimation of linear mixed effects models does not have a closed form solution. If $\tilde{\Sigma}$ and σ^2_ϵ are known, analytical solutions can be obtained by the generalized least squares estimation. For more details, see Section A.3.1.

6.3 Longitudinal analysis of CDT images

Let $\mathcal{I}_{i,j}$ denote the image acquired from subject i at time point j. Given images $\mathcal{I}_{i,j}$ and $\mathcal{I}_{i,j+1}$ for successive visits $(j, j+1)$, we can compute a deformation (e.g., diffeomorphism) that aligns the two images (Avants et al., 2008; Klein et al., 2009). Let $\mathcal{I}_{i,1}$ (i.e., $j = 1$) give the subject-specific coordinate system denoted as Ω_i. This will provide the (intermediate) common coordinate system to represent the deformations undergone by subject i over time, $j = 1, 2, \cdots, n_i$. The global template (or coordinate system) where all $(n_i - 1)$ temporal deformations (i.e., CDT images) for each subject i will be represented is denoted as Ω. Then, a nonlinear deformation $\Phi(\texttt{vox})$ for voxels (spatial locations) $\texttt{vox} \in \Omega$ for each image

(rather, for each $(\mathcal{I}_{i,j+1}, \mathcal{I}_{i,j})$ pair) is given as

$$\begin{aligned} &\Phi : \mathcal{I}_{i,j+1} \rightarrow \mathcal{I}_{i,j} \\ &\Phi(\texttt{vox} + d\texttt{vox}) = \Phi(\texttt{vox}) + J(\texttt{vox})d\texttt{vox} + \mathcal{O}(d\texttt{vox}^2), \end{aligned} \tag{6.4}$$

where $J(\texttt{vox})$ denotes the Jacobian of the deformations at position $\texttt{vox}$. A nice property of CDTs is that it preserves the determinant of $J(\texttt{vox})$, since $\det(J(\texttt{vox})) > 0$. So, a CDT representation introduced in Sec. 6.1, nicely symmetrizes $J(\texttt{vox})$ without affecting the volumetric change information, i.e., $\det(J) = \det(\sqrt{J^T J})$. The CDT "image" comprised of voxel locations $\texttt{vox}$ is an object of the same size as $\mathcal{I}_{1,1}$ and derived from a black-box diffeomorphism solver given as a 3×3 SPD matrix $\sqrt{J^T J}$ at each voxel. It provides the deformation field between two longitudinal images of a subject. Various results have described the benefits of CDT images for analysis; an example from our experiments is given in Sec. 7.6.

6.3.1 A model with subject specific intercepts

We know that in any longitudinal/temporal dataset, the errors/noise of repeated measurements are dependent. To take this aspect of the data into account, a common approach is to express the random effects of subjects as nuisance parameters. If the set $\{i = 1, i = 2, \cdots, i = N\}$ indexes the columns, we may write the design matrix Z as $\text{diag}(1_{n_1}, 1_{n_2}, \cdots, 1_{n_N})$, where $1_{n_i} = [1 \ \cdots 1]^T \in R^{n_i}$. Then, the model in (6.3) becomes

$$y_{[\![ij]\!]} = \beta^0 + \boldsymbol{\beta}^T \boldsymbol{x}_{[\![ij]\!]} + u_i, \tag{6.5}$$

where $y_{[\![ij]\!]}, \beta^0, u_i \in \mathbf{R}$, and $\boldsymbol{\beta}, \boldsymbol{x}_{[\![ij]\!]} \in \mathbf{R}^p$. Note that $\mathbf{z}_{[\![ij]\!]} \in \mathbf{R}^N$ recovers a specific row corresponding to subject i's visit j from matrix Z taking dot product with $\boldsymbol{u}$ gives us the subject specific random effects, $u_i = \mathbf{z}_{[\![ij]\!]}\boldsymbol{u}$.

This model poses two problems. **1)** It has the same slope $\boldsymbol{\beta}$ for the

entire population, whereas subjects in the study may have different rate of disease progression; **2)** Another issue is the interpretation of u_i, which is viewed as subject-specific shift in the y space or x space, i.e., depending on whether we move it to the left or right of the equality in (6.5). In medical applications, readability of models is important to understand the disease. Our solution involves explicitly adding a subject-specific shift for x as well as a shift in y.

6.3.2 Nonlinear mixed effects models with $\psi_i(x)$

Based on the foregoing motivation, we can extend the linear mixed effects models with a subject-specific random function $\psi_i(\cdot)$ as

$$y_{[\![ij]\!]} = \beta_0 + \mathbf{z}_{[\![ij]\!]} u_i + \boldsymbol{\beta}.\psi_i(\boldsymbol{x}_{[\![ij]\!]}) \tag{6.6}$$

Depending on the form of $\psi_i(\cdot)$, (6.6) can be a nonlinear mixed effects model (NLMM). When ψ_i is the Identity, we simply get a linear mixed effects model. In our analysis, we use $\psi_i(\boldsymbol{x}) := \alpha_i(\boldsymbol{x} - \tau_i - t_0) + t_0$ motivated by Durrleman et al. (2013) where each subject can have their own speed of disease progression (α_i) and different onset time τ_i, but $(\beta_0, \boldsymbol{\beta}, t_0)$ are common for the population. Then, we have

$$y_{[\![ij]\!]} = \beta_0 + \mathbf{z}_{[\![ij]\!]} u_i + \boldsymbol{\beta}(\alpha_i(\boldsymbol{x}_{[\![ij]\!]} - \tau_i - t_0) + t_0). \tag{6.7}$$

Note that this extension is different from the *generalized linear mixed effects models* Lindstrom and Bates (1990), e.g., $y_{[\![ij]\!]} = h^{-1}(\boldsymbol{x}_{[\![ij]\!]}\boldsymbol{\beta} + \mathbf{z}_{[\![ij]\!]} u_i), u_i \sim \mathcal{N}(0, \Sigma_i)$. Next, we extend the mixed effects models in (6.5) and (6.7) to manifold-valued data.

6.4 Mixed effects models on manifolds

The Linear Mixed Effects Model (LMM) can be extended in many ways to the manifold setting depending on the order of addition and interpretation. For instance, recall that the associativity of addition, $(a+b)+c = a+(b+c)$, in the Euclidean space is not directly translated to manifolds, i.e., $\text{Exp}(\text{Exp}(a,b),c) \neq \text{Exp}(\text{Exp}(a,c'),b')$, where b' and c' are parallelly transported tangent vectors of b and c respectively, so that they are in the right tangent spaces. A natural extension of LMM in (6.3) can be written as

$$y_{[\![ij]\!]} = \text{Exp}(\text{Exp}(\text{Exp}(B, V\boldsymbol{x}_{[\![ij]\!]}), U_i\mathbf{z}_{[\![ij]\!]}), \epsilon_{[\![ij]\!]}), \tag{6.8}$$

where $y_{[\![ij]\!]}, B, B_i \in \mathcal{M}$, $V \in T_B\mathcal{M}^p$, $U_i \in T_{h_{[\![ij]\!]}}\mathcal{M}^q$, $h_{[\![ij]\!]} = \text{Exp}(B, V\boldsymbol{x}_{[\![ij]\!]})$, $\boldsymbol{x}_{[\![ij]\!]} \in \mathbf{R}^p$ and $\mathbf{z}_{[\![ij]\!]} \in \mathbf{R}^q$. Recall that the base point B on the manifold $\mathcal{M}$ is the analog to the intercept β^0 in (6.5) whereas V (and U_i) corresponds to the slope $\boldsymbol{\beta}$ (and the random effects $\boldsymbol{u}_i$) respectively. Unfortunately, the model above involves a subtle issue related to U_i. Note that U_i is used in different tangent spaces at $h_{[\![ij]\!]}$. Also, especially on $\text{SPD}(n)$ manifolds with the GL-invariant metric, the norm of the tangent vectors varies as a function of the base point B of the respective tangent spaces, i.e., $\|U\|_B^2 = \langle U, U\rangle_B = \text{tr}(UB^{-1}UB^{-1})$. So the corresponding scales might be different. As a result, the prior for U_i needs to be carefully designed (Kim et al., 2015b) so that it is consistent over all tangent spaces. To address this problem, we change the order of the exponential maps and propose a mixed effects model with subject specific intercepts (shift in y) on manifolds. Also, unlike the Euclidean space, in general, there is no equivalence between the shift in x and the shift in y. So, we can explicitly add in the shift in x, denoted as τ_i. Then, our formulation on manifolds is

given as

$$\boldsymbol{y}_{[\![ij]\!]} = \text{Exp}(\text{Exp}(B_i, \Gamma_{B\to B_i}(V)(\boldsymbol{x}_{[\![ij]\!]} - \boldsymbol{\tau}_i)), \boldsymbol{\epsilon}_{ij}), \tag{6.9}$$

$$B_i = \text{Exp}(B, U_i \mathbf{z}_{[\![ij]\!]}), \tag{6.10}$$

where $\boldsymbol{\tau}_i \in \mathbf{R}^p$, $B_i \in \mathcal{M}$, $V \in T_B\mathcal{M}^p$, $U_i \in T_B\mathcal{M}^q$, and the remaining variables are the same as before. As in the standard mixed effects models, U_i is assumed to follow a multivariate normal distribution. Recall that U_i is in the tangent space of $\text{SPD}(n)$, which we know is the space of symmetric matrices $\text{Sym}(n)$. So, we may specify a normal distributions for $\text{Sym}(n)$ (Schwartzman, 2006), see Chapter A.1. With this basic construction in hand, we may now include a subject-specific time shift in the onset time (similar to (6.7)) and assume that the progression of disease has the same overall pattern but only its speed/rate and onset time vary between subjects. This allows writing a model with fewer parameters given by

$$\begin{aligned} y_{[\![ij]\!]} &= \text{Exp}(\text{Exp}(B_i, \Gamma_{B\to B_i}(V)\alpha_i(x_{[\![ij]\!]} - \tau_i - t_0), \epsilon_{ij}))) \\ B_i &= \text{Exp}(B, U_i), \end{aligned} \tag{6.11}$$

where α_i is the subject-specific acceleration, $\alpha_i < 1$ (and $\alpha_i > 1$ resp.) means slower (and faster resp.) than the population. Further, τ_i is the subject-specific shift in onset time: $\tau_i > 0$ means a late onset time whereas t_0 is the global shift in onset time. Finally, U_i (or B_i) are the tangent vectors (or base points) that characterize the subject-specific shift in the response variable space (see the Euclidean case in Fig. 6.2). As in the classical setting, we may specify the following priors on the manifold valued parameters, $\Gamma_{B\to I}U_i \sim \mathcal{N}_{SYM}(0, \sigma_U^2)$, $\alpha_i \sim \mathcal{N}(1, \sigma_\alpha^2)$, $\tau_i \sim \mathcal{N}(0, \sigma_\tau^2)$.

6.5 Parameter estimation procedure

In general, accurate estimation of the (nonlinear) mixed effects models in even Euclidean space is computationally demanding. So the accurate estimation of the mixed effects models on manifolds in (6.10) that involves more complex nonlinear functions may be prohibitively computationally expensive to run over the entire brain image. Even for Euclidean response variables, efficient estimation methods for *nonlinear* mixed effects models are still being actively studied (in machine learning and statistics), e.g., Alternating algorithms (Lindstrom and Bates, 1990), Laplacian and adaptive Gaussian quadrature algorithms (Pinheiro and Chao, 2012), as well as generalized EM algorithms with MCMC (Meza et al., 2007). Unfortunately, this issue only gets worse in the manifold setting. Fitting a nonlinear mixed effects model exactly, even for *univariate manifolds on the real line* takes about a day (Schiratti et al., 2015) with a generalized EM algorithm. In our data set, the number of voxels is 1M+, it is impractical to perform exact analysis for the full brain. So, we present approximate algorithms based on a certain geometrical interpretation of the models.

6.5.1 Estimation of RNLMM

We observe that the main building block of our models, Riemannian nonlinear mixed effects models (RNLMMs), is a manifold-valued multivariate general linear model (MMGLM). This module has an efficient parameter estimation called the Log-Euclidean framework. In Chapter 3, we discussed that in practice the estimation can be well approximated in the tangent space at the Fréchet mean of the response variables Y with a centered X, i.e., $B \approx \bar{Y}, \tau \approx \bar{X}$. As in a global manifold-valued linear model, i.e, MMGLMs in Chapter 3, the parameter V will correspond to the full data set; however, we allow subject-specific variability for the base point B and τ via $B_i(r)$ and $\tau_i(r)$, where $r \in \mathbf{R}$ can be viewed as the

mixing rate between the local models that share a global V. This is given by

$$y_{[\![ij]\!]} = \text{Exp}(\text{Exp}(B_i(r), \Gamma_{B \to B_i(r)}(V)(\boldsymbol{x}_{[\![ij]\!]} - \tau_i(r))), \epsilon). \tag{6.12}$$

In other words, $r \in \mathbf{R}$ is a weight to globally average the population subject specific base points $B_i(r)$ and time shifts $\tau_i(r)$ — all subjects share the fixed effects V but each subject corresponds to its own shifts $\tau_i(r)$ and $B_i(r)$ in x and y spaces. When $r = 0$, the model reduces to the model in (Kim et al., 2014c; Cornea et al., 2016) with only global intercepts, see Section 6.6.1.

Our estimation for (6.29) is summarized in Alg. 8, where $y^{\wr}_{[\![ij]\!]}$ is a tangent vector obtained by: taking the response $y_{[\![ij]\!]}$ and mapping it to the tangent space at $B_i(r)$ and parallel transporting that mapping to $T_B\mathcal{M}$. We now briefly describe how we can perform the estimation efficiently. First, in Step 2, we solve for the linear interpolation of two SPD matrices w.r.t.

Algorithm 8 Riemannian mixed effects models

1: Calculate the mean for each subject, $\bar{y}_i$,

$$\bar{\boldsymbol{y}}_i = \underset{\boldsymbol{y} \in \mathcal{M}}{\text{argmin}} \sum_{j=1}^{n_i} \text{d}(\boldsymbol{y}, \boldsymbol{y}_{[\![ij]\!]})^2. \tag{6.13}$$

Similarly calculate $\bar{\boldsymbol{y}}$ for the entire population.

2: Given r, solve for $B_i(r)$ (interpolation of $\bar{\boldsymbol{y}}_i$ and $\bar{\boldsymbol{y}}$) by

$$B_i(r) = \bar{\boldsymbol{y}}(\bar{\boldsymbol{y}}^{-1}\bar{\boldsymbol{y}}_i)^r = \bar{\boldsymbol{y}}_i(\bar{\boldsymbol{y}}_i^{-1}\bar{\boldsymbol{y}})^{1-r}, 0 \leq r \leq 1.$$

3: $\boldsymbol{y}^{\wr}_{[\![ij]\!]} = \Gamma_{\bar{B}_i(r) \to B}\text{Log}(\bar{B}_i(r), \boldsymbol{y}_{[\![ij]\!]})$.

4: Transport $\boldsymbol{y}^{\wr}_{[\![ij]\!]}$ to I by group action.

5: Center $\boldsymbol{x}$ by $\boldsymbol{x}_{[\![ij]\!]}(r)$=$(1-r)(\boldsymbol{x}_{[\![ij]\!]} - \bar{\boldsymbol{x}}) + r(\boldsymbol{x}_{[\![ij]\!]} - \bar{\boldsymbol{x}}_i)$.

6: Calculate V^* using MMGLM on transported $\boldsymbol{y}^{\wr}_{[\![ij]\!]}$ and $\boldsymbol{x}_{[\![ij]\!]}(r)$.

7: Prediction is given by

$$\hat{\boldsymbol{y}}_{ij} = \text{Exp}(B_i(r), \Gamma_{B \to \bar{B}_i(r)}(V^*)\boldsymbol{x}_{ij}(r)). \tag{6.14}$$

the geodesic distance on the SPD manifold using the analytical form of the solution in (Moakher and Batchelor, 2006) (note that when the number of samples is large, recursive schemes exist (Ho et al., 2013)). In Step 4, we transport the tangent vectors from B to I and vice versa using group action, which is known to be more efficient than parallel transport but equivalent as discussed in Chapter 4.

6.5.2 Estimation of RNLMM with $\psi_i(x)$

The estimation of the model in (6.11) with the subject-specific random function $\psi_i(\cdot)$ involves few additional technical challenges. To reduce the problem complexity, we first find the main longitudinal change direction $\boldsymbol{\eta}$ controlling for the subject-specific random effects $\bar{Y}_i$ and $\bar{X}_i$ (since U_i and τ_i are random effects). This scheme is described in Alg. 9.

Algorithm 9 Calculate longitudinal change direction

1: Calculate the population Fréchet mean $\bar{\boldsymbol{y}}$ of response.
2: Calculate the Fréchet mean for each subject $\bar{\boldsymbol{y}}_i$.
3: Solve $\boldsymbol{y}^{\wr}_{[\![ij]\!]} = \Gamma_{\bar{\boldsymbol{y}}_i \to I} \mathrm{Log}(\bar{y}_i, \boldsymbol{y}_{[\![ij]\!]})$.
4: Solve $\boldsymbol{x}^{\wr}_{[\![ij]\!]} = \boldsymbol{x}_{[\![ij]\!]} - \bar{\boldsymbol{x}}_i$, where $\bar{\boldsymbol{x}}_i = \mathbb{E}_j[\boldsymbol{x}_{[\![ij]\!]}]$.
5: Collect $X^{\wr} = [\boldsymbol{x}^{\wr}_1; \ldots, \boldsymbol{x}^{\wr}_N]$, and $Y^{\wr} = [\boldsymbol{y}^{\wr}_1, \ldots, \boldsymbol{y}^{\wr}_N]$.
6: Calculate longitudinal change direction $\boldsymbol{\eta}$ by least squares estimation, $\boldsymbol{\eta} = ((X^{\wr})^T X^{\wr})^{-1}((X^{\wr})^T Y) \in T_I \mathcal{M}$.

Once the longitudinal change direction η (fixed effects for the entire population) is estimated, we solve for a subset of parameters at a time. This procedure is described in Alg. 10, where we solve for all parameters given the estimate of η. Note that for our downstream analysis, the bias induced by priors on parameters may reduce the statistical power. So, we simply used noninformative priors for all parameters. While Alg. 10 utilizes noninformative priors, with minor changes, we can easily incorporate normal distribution priors. It turns out that if the response y has a generalized normal distributed noise on manifolds, as in the

Euclidean space, the least-squares estimator (first term in (6.15)) is the same as the maximum likelihood estimator (Fletcher, 2013).

$$\min_{c,t_0,\{B_i,U_i,\alpha_i,\tau_i\}} \sum_{ij} \mathrm{d}(y_{[\![ij]\!]}, \mathrm{Exp}(B_i, \Gamma_{I\to B_i}(c\eta)(\psi_i(\boldsymbol{x}_{[\![ij]\!]}))))^2 + \lambda_u \sum_i \|U_i\|_B^2 + \lambda_\alpha \sum_i \|\alpha_i - 1\|^2 + \lambda_\tau \sum_i \|\tau_i\|^2, \tag{6.15}$$

where $\psi_i(\boldsymbol{x}_{[\![ij]\!]}) := \alpha_i(\boldsymbol{x}_{[\![ij]\!]} - \tau_i - t_0) + t_0$; $B_i = \mathrm{Exp}(B, U_i)$.

Remarks. Notice the regularizers in (6.15) comprise of the last three terms. This is based on the fact that MLE estimation of linear regression with a normal distributional prior for the coefficients is equivalent to the ridge regression estimate (Hoerl and Kennard, 1970). For inference with priors, Step 4 and 7-8 in Alg. 10 need to be substituted with ridge regression.

Alg. 10 contains many steps in common with Alg. 8. In Step 7, we estimate the fixed effects V and t_0 by fixing all other variables (c is a dummy variable). In Step 8, we estimate the subject-specific random effects α_i and τ_i by fixing V and t_0 (d_i are dummy variables).

Derivation of step 7. in Alg. 10, given $\boldsymbol{\eta}$

$$\begin{aligned} y_{ij} &= \mathrm{Exp}(B_i, \Gamma_{I\to B_i}(c\eta)(\alpha_i(x_{ij} - \tau_i - t_0) + t_0)) \\ \mathrm{Log}(B_i, y_{ij}) &= \Gamma_{I\to B_i}(c\eta)(\alpha_i(x_{ij} - \tau_i - t_0) + t_0) \\ y_{ij}^{\wr} &= c\eta(\alpha_i x_{ij} - \alpha_i\tau_i - \alpha_i t_0 + t_0), \text{ by } \Gamma_{B_i\to I} \\ &= \eta(\alpha_i x_{ij} - \alpha_i\tau_i)c + \eta(1-\alpha_i)t_0 c \end{aligned}$$

where $b := t_0 c$. Let $q_i := \eta(1-\alpha_i)$

Algorithm 10 Riemannian mixed effects models with $\psi_i(x)$

1: Calculate the Fréchet mean $\bar{y} \in \mathcal{M}$ of population.
2: Calculate the Fréchet mean for each subject $\bar{y}_i \in \mathcal{M}$.
3: Main longitudinal change direction η by algorithm (9).
4: Calculate subject-specfic base points (random effects) $B_i = \text{Exp}(B, U_i^*)$, where $U_i^* = \text{argmin}_{U_i} \, \text{d}(\bar{y}_i, \text{Exp}(B, U_i))^2 + \lambda_{U_i} \|U_i\|_B^2$.
5: $y_{ij}^{\wr} = \Gamma_{B_i \to I} \text{Log}(B_i, y_{ij})$.
6: **while** until convergence **do**
7: Calculate the common speed of change $V = c\boldsymbol{\eta}$ and common time intercept $t_0 = b/c$ with fixed all other variables by

$$\begin{bmatrix} \sum_{ij} q_i^T q_i & \sum_{ij} p_{ij}^T q_i \\ \sum_{ij} p_{ij}^T q_i & \sum_{ij} p_{ij}^T p_{ij} \end{bmatrix} \begin{bmatrix} b \\ c \end{bmatrix} = \begin{bmatrix} \sum_{ij} q_i^T y_{ij}^{\wr} \\ \sum_{ij} p_{ij}^T y_{ij}^{\wr} \end{bmatrix},$$

where $b := t_0 c$, $q_i := \eta(1 - \alpha_i)$,$p_{ij} := \eta(\alpha_i x_{ij} - \alpha_i \tau_i)$.

8: Given V, t_0, calculate the subject-specific acceleration α_i, and time-shift τ_i by generalized least square estimation with the priors for α_i and $\tau_i = d_i/\alpha_i$

$$\begin{bmatrix} \sum_j W_{ij}^T W_{ij} & -\sum_j W_{ij}^T V \\ \sum_j W_{ij}^T V & -\sum_j V^T V \end{bmatrix} \begin{bmatrix} \alpha_i \\ d_i \end{bmatrix} = \begin{bmatrix} \sum_j \text{Y}_{ij}^T W_{ij} \\ \sum_j \text{Y}_{ij}^T V \end{bmatrix},$$

where $\text{Y}_{ij} := y_{ij}^{\wr} - V t_0$, $W_{ij} := V(X_{ij} - t_0)$ and $d_i = \alpha_i \tau_i$.
9: **end while**

and $p_{ij} := \eta(\alpha_i x_{ij} - \alpha_i \tau_i)$.

$$\begin{aligned} &\underset{b,c}{\text{argmin}} \sum_{ij} (y_{ij}^{\wr} - p_{ij} c - q_i b)^T (y_{ij}^{\wr} - p_{ij} c - q_i b) \\ = &\underset{b,c}{\text{argmin}} \sum_{ij} (y_{ij}^{\wr})^T y_{ij}^{\wr} + c^2 p_{ij}^T p_{ij} + b^2 q_i^T q_i - 2c(y_{ij}^{\wr})^T p_{ij} \\ &\qquad + 2bc p_{ij}^T q_i - 2b q_i^T y_{ij}^{\wr} \end{aligned}$$

Take the partial derivatives

$$\frac{\partial}{\partial b} = -\sum_{ij} q_i^T y_{ij}^{\wr} + c\sum_{ij} p_{ij}^T q_i + b\sum_{ij} q_i^T q_i = 0 \tag{6.16}$$

$$\frac{\partial}{\partial c} = -\sum_{ij} p_{ij}^T y_{ij}^{\wr} + c\sum_{ij} p_{ij}^T p_{ij} + b\sum_{ij} p_{ij}^T q_i = 0 \tag{6.17}$$

So the system of equations from the KKT condition is,

$$\begin{bmatrix} \sum_{ij} q_i^T q_i & \sum_{ij} p_{ij}^T q_i \\ \sum_{ij} p_{ij}^T q_i & \sum_{ij} p_{ij}^T p_{ij} \end{bmatrix} \begin{bmatrix} b \\ c \end{bmatrix} = \begin{bmatrix} \sum_{ij} q_i^T y_{ij}^{\wr} \\ \sum_{ij} p_{ij}^T y_{ij}^{\wr} \end{bmatrix} \tag{6.18}$$

Derivation for Step 8. Given $V := c\eta$, and t_0,

$$\begin{aligned} y_{ij} &= \mathrm{Exp}(B_i, \Gamma_{I\to B_i}(V)(\alpha_i(X_{ij} - \tau_i - t_0) + t_0)) \\ y_{ij}^{\wr} &= V(\alpha_i(X_{ij} - \tau_i - t_0) + t_0), \text{the same trick} \\ y_{ij}^{\wr} &= V(\alpha_i X_{ij} - \alpha_i\tau_i - \alpha_i t_0 + t_0) && (6.19) \\ y_{ij}^{\wr} - Vt_0 &= V\alpha_i X_{ij} - V\alpha_i\tau_i - V\alpha_i t_0 && (6.20) \\ \mathrm{Y}_{ij} &= W_{ij}\alpha_i - Vd_i && (6.21) \\ & && (6.22) \end{aligned}$$

where $\mathrm{Y}_{ij} := y_{ij}^{\wr} - Vt_0$, $W_{ij} := V(X_{ij} - t_0)$ and $d_i = \alpha_i\tau_i$

$$\begin{aligned} &\operatorname{argmin} \sum_j (\mathrm{Y}_{ij} - W_{ij}\alpha_i + Vd_i)^T(\mathrm{Y}_{ij} - W_{ij}\alpha_i + Vd_i) && (6.23) \\ &= \operatorname{argmin} \sum_j \mathrm{Y}_{ij}^T\mathrm{Y}_{ij} + W_{ij}^T W_{ij}\alpha_i^2 + V^T V d_i^2 && (6.24) \\ &\quad - 2\mathrm{Y}_{ij}^T W_{ij}\alpha_i + 2\mathrm{Y}_{ij}^T V d_i - 2W_{ij}^T V\alpha_i d_i && (6.25) \end{aligned}$$

Take the derivatives

$$\frac{\partial}{\partial \alpha_i} = \sum \alpha_i W_{ij}^T W_{ij} - \Upsilon_{ij} W_{ij} - W_{ij}^T V d_i \tag{6.26}$$

$$\frac{\partial}{\partial d_i} = \sum V^T V d_i + \Upsilon_{ij}^T V - W_{ij}^T V \alpha_i \tag{6.27}$$

The system of equations is given as

$$\begin{bmatrix} \sum_j W_{ij}^T W_{ij} & -\sum_j W_{ij}^T V \\ \sum_j W_{ij}^T V & -\sum_j V^T V \end{bmatrix} \begin{bmatrix} \alpha_i \\ d_i \end{bmatrix} = \begin{bmatrix} \sum_j \Upsilon_{ij}^T W_{ij} \\ \sum_j \Upsilon_{ij}^T V \end{bmatrix} \tag{6.28}$$

6.6 Experiments

We first show the synthetic data experiments to demonstrate the different behaviors of the proposed models. Then, we perform analysis with real longitudinal data from a neuroimaging study.

6.6.1 Synthetic experiments

In this section, we demonstrate the difference between our Riemannian nonlinear mixed effects model in (6.29) and MGLMs.

$$y_{[\![ij]\!]} = \text{Exp}(\text{Exp}(B_i(r), \Gamma_{B \to B_i(r)}(V)(\boldsymbol{x}_{[\![ij]\!]} - \tau_i(r))), \epsilon). \tag{6.29}$$

The model in (6.29) reduce to MMGLM in Chapter 3 as $r \to 0$ since all the subjects have the same intercept and same slope V, see the trend in Fig. 6.3. This model uses the interpolation between a global mean and a subject-specific mean as a subject-specific intercept. In Fig. 6.3(d), the global intercept is completely ignored and the model uses the mean

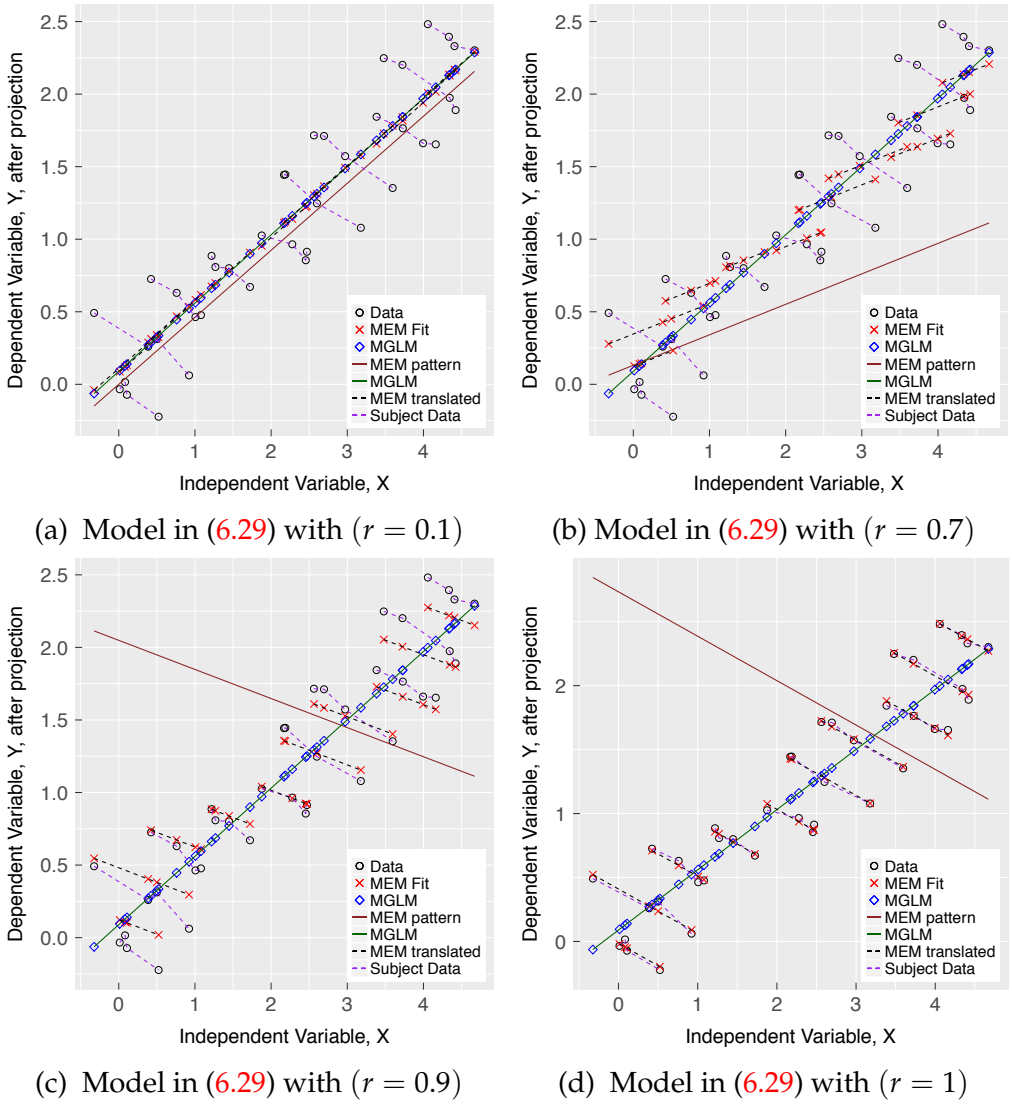
of measurements from a subject as the intercept for the subject. This allows a more flexible model that can learn the correct trajectories in the synthetic data. In Fig. 6.3(a), the interpolation between subject-specific mean and population mean with the weight $r = 0.1$, the subject-specific intercepts becomes closer to the population mean. The model behaves like a MMGLM and it fails to capture true trends with a poor fit.

6.6.2 Neuroimaging data experiments

Goals. The overarching goal of our experiments is to evaluate whether the proposed formulations can serve as core modules that drive longitudinal analysis of image datasets in neuroimaging. To this end, when conducting analysis of longitudinal data acquired in the context of a specific disease, the procedure should yield meaningful results for group analysis — for instance, when the population is split with a stratification variable (e.g., gender or disease risk factor), the "maps" of statistically significant group-wise differences in subject/voxel-specific "random" effects (especially, acceleration and spatial shift) should be scientifically interpretable, yet generally consistent with a baseline. Our experiments below show the extent to which the models satisfy this requirement.

Data. The CDT images (denoting subject-specific warps) were derived from a longitudinal neuroimaging study of pre-clinical Alzheimer's disease (AD). The longitudinal warps (or transformations) were obtained using *with-in* subject registration of T_1-weighted images between two consecutive visits i.e., $\Phi_{i,j} : \mathcal{I}_{i,j} \rightarrow \mathcal{I}_{i,j+1}$. Voxelwise CDTs were derived from the spatial derivatives $\nabla\Phi_{i,j}(\texttt{vox})$ of the deformation field. The details of calculation of CDTs and minimizing spatio-temporal biases in the CDT estimation are presented in Section A.3.2 and A.3.3.

CDT versus det(J). We first present a motivating experiment to demonstrate the rationale behind using CDTs instead of the determinants det(J) of Jacobian of the deformations, i.e., do CDTs actually carry more



(a) Model in (6.29) with $(r = 0.1)$

(b) Model in (6.29) with $(r = 0.7)$

(c) Model in (6.29) with $(r = 0.9)$

(d) Model in (6.29) with $(r = 1)$

Figure 6.3: When the variability over subjects is large than variability of a subject over time, the linear model (MMGLM in blue) captures the overall relationship between x and y. In this example, the trajectory estimated by a MMGLM is significantly different from the trajectory of each subject. Each subject is measured four times and the trajectory correctly captured by the mixed effects model (in gray dot lines). The mixed effects model control the variability between subjects as random effects and captured the common longitudinal pattern (in red). The common longitudinal change is translated to each subject (in gray dot lines).

information? We test for group differences in longitudinal changes of the brain between groups of middle versus old aged individuals using CDTs and compare these results to those obtained via determinants. In order to avoid confounding factors in this comparison, we use the Cramér's test, a nonparametric test for univariate as well as manifold-valued data since it does not require any specification of the null distribution (Baringhaus and Franz, 2004). Fig. 6.4 clearly shows the improvements in statistical differences across the groups (higher sensitivity) when using CDTs (instead of det(J) maps).

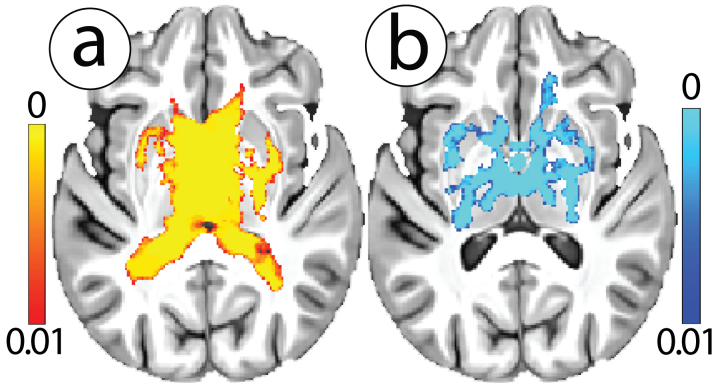


Figure 6.4: Results of Cramér's test showing voxels that are different between middle and old age groups ($p < 0.01$) from (a) CDTs and (b) det(J).

RNLMMs on longitudinal CDTs. We now present results using our Riemannian nonlinear mixed effects models (RNLMM) using subject specific transformation functions $\psi_i(x_{[\![ij]\!]})$ (6.6). Here, $x_{[\![ij]\!]} \in \mathbf{R}$ is used to represent the age of each subject at the previous visit and $y_{[\![ij]\!]} \in \mathcal{M}$, (the CDT image calculated from scans at two points). For these results, we used data from subjects who had at least three visits. We estimated our model at each voxel in the brain ($1.3M+$) using a total of $N = 228$ participants that had at least two CDT images. The maps for acceleration (α_i), spatial shift (U_i) and time shift (τ_i) for each of the subjects offer unique advantages. For instance, these maps are *not offered* by standard linear

mixed effects models where only a subject specific slope or intercept is used as the random-effects (independently noted in (Schiratti et al., 2015)). Fig. 6.5 shows four representative subject-specific acceleration maps. The regions where *this specific individual* has a faster (slower) aging (or disease progression) compared to the population average rate are colored in yellow (and blue) color-scales respectively. These RNLMM maps can be used to perform additional "downstream" statistical tests using parametric tests. Here, we cover two specific examples. In Fig. 6.5, we show the kind of results our model can offer at the **individual level**. Fig. 6.5(a)-(d) shows four results, each pertaining to a different participant in the study. Fig. 6.5(a)-(b) show maps for two females, whereas Fig. 6.5(c)-(d) show examples of two males. The color indicates the brain deformation over time (captured via acceleration), for this specific person, relative to the population. We see that a representative male (with no APOE risk) shows a slower acceleration rate (blue regions) compared to the population. Not many models in the literature can provide such *personalized assessment*.

Of course, such acceleration and spatial shift maps can also be used for **group level** analysis. We present results of Hotelling-T^2 tests on the group-wise U_i maps using the following two stratification variables: (a) males and females and (b) individuals with/without AD risk (due to APOE) (Tang et al., 1998; Corder et al., 1993). This enables us to identify longitudinal spatial shifts (deviations from population base points B) between these groups, shown in Fig. 6.6 for the gender and APOE stratification variables.

6.7 Summary

In this chapter, we extend nonlinear mixed effects models to the setting where the responses lie on curved spaces such as the manifold of

symmetric positive definite (SPD) matrices. By treating the subject-wise "non-linear warps" between consecutive time points as a field of Cauchy deformation tensors (CDT), we show how our model can facilitate longitudinal analysis that respects the geometry of such data. While the existing body of work dealing with regression models on manifold-valued data is inherently restricted to cross-sectional studies, the proposed mixed effects formulation significantly expands the operating range of the types of analyses we can conduct. For instance, the "random" effects in the construction parameterized by acceleration and spatial and time shifts offer interesting advantages. Not only can these quantities be directly used for downstream models but they also offer interpretability at the *level of individual subjects* — as an example, when conditioned on (or controlled for) race, sex and education, we can ask if a specific person's onset time of brain atrophy or *rate of atrophy*, at the level of individual voxels, deviates from the group. This capability is not currently available otherwise.

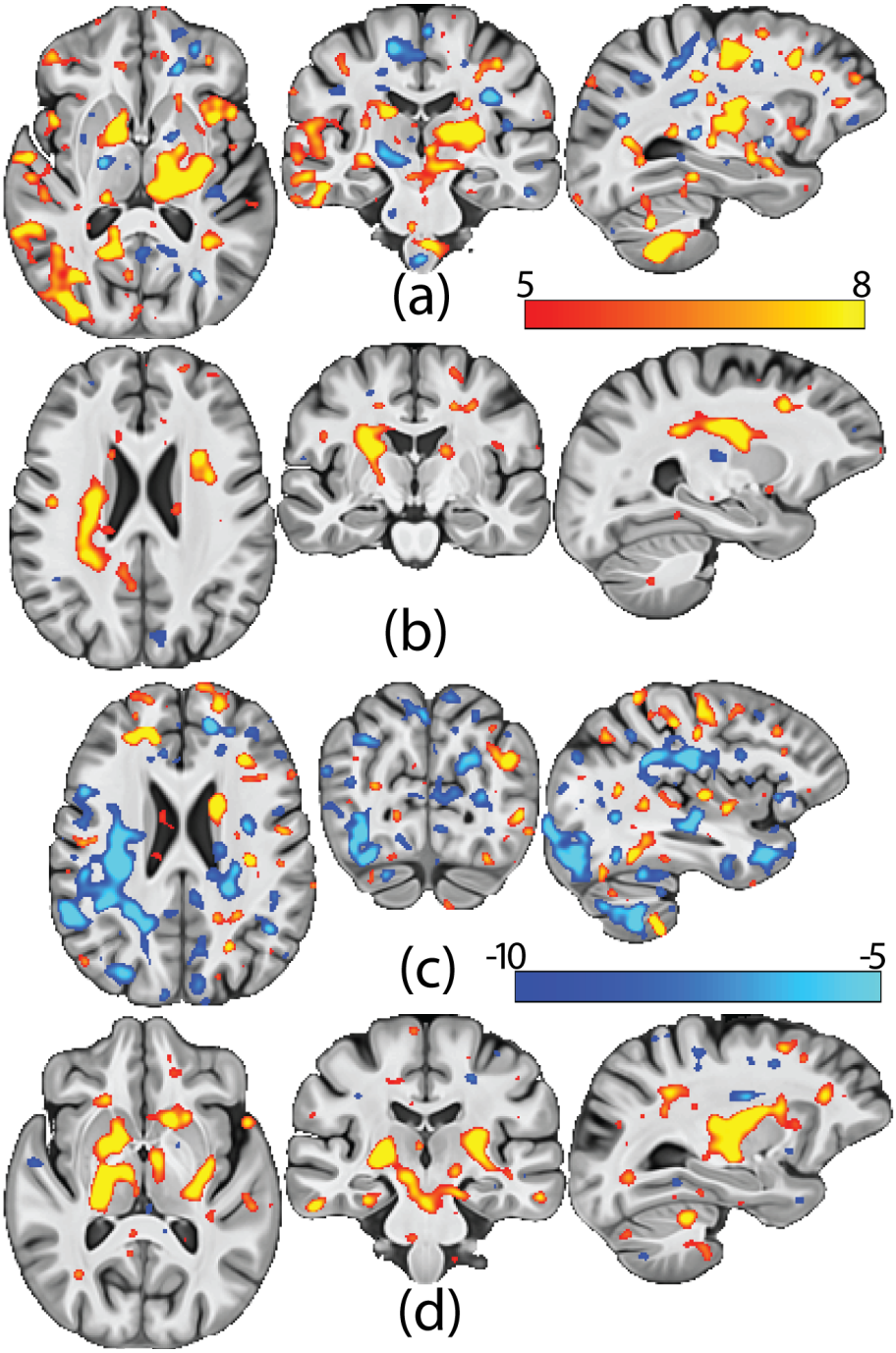


Figure 6.5: Representative acceleration (α_i) maps derived from our RNLMM. (a) Female, APOE-. (b) Female, APOE+. (c) Male, APOE-. (d) Male, APOE+. The male with no APOE risk shows slower progression (more blue regions) compared to the population average.

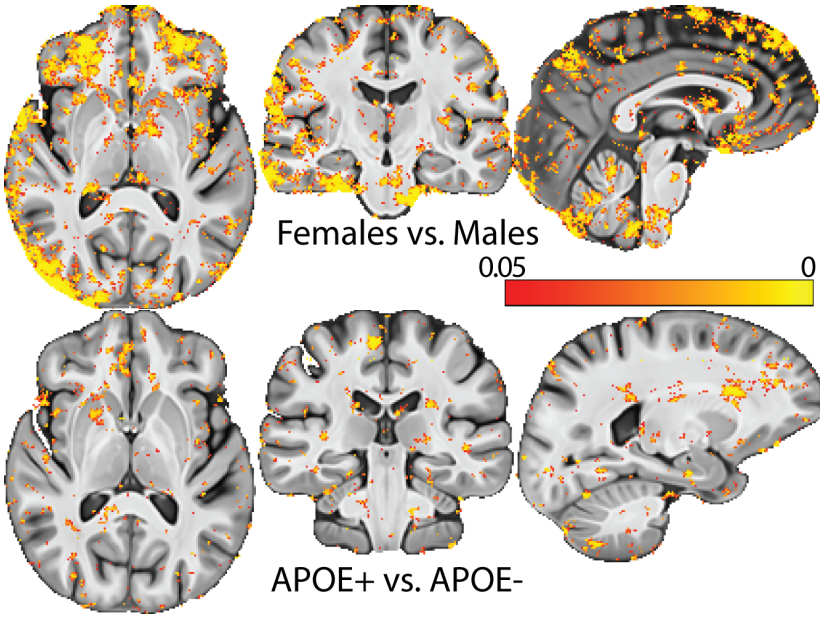


Figure 6.6: P-value maps of group differences in random effects (U_i). Top: Gender differences. Bottom: APOE group {APOE+, APOE-} differences. Gender differences can be effectively captured by our RNLMM.

7 INTERPOLATION ON THE MANIFOLD OF k COMPONENT GMMS

Probability density functions (PDFs) are fundamental objects in mathematics with numerous applications in computer vision, machine learning and medical imaging. Especially, in medical imaging, the diffusion of water molecules in diffusion weighted MRI is characterized by a PDF so-called ensemble average propagator (EAP) at each voxel. Diffusion tensor imaging (DTI) is a simple method to parameterize EAPs by a single Gaussian density function. But it is not capable to capture multiple orientations (fiber crossing) in a voxel. So, to address the problem, a mixture of Gaussian densities is often used. Motivated by applications, e.g., registration, denoising, and smoothing, in diffusion weighted MRI with GMMs, we study the parameterizations of Gaussian mixture models (GMMs) and their interpolation.

In this chapter, we study numerical algorithms to enable basic operations on such objects that strictly respect their underlying geometry controlling the model complexity. For instance, when operating with a set of K component GMMs, a first order expectation is that the result of simple operations like interpolation and averaging should provide an object that is also a K component GMM. The literature provides very little guidance on enforcing such requirements systematically. It turns out that these tasks are important internal modules for analysis and processing of a field of ensemble average propagators (EAPs), common in diffusion weighted magnetic resonance imaging. We provide proof of principle experiments showing how the proposed algorithms for interpolation can facilitate statistical analysis of such data, essential to many neuroimaging studies. Separately, we also derive interesting connections of our algorithm with functional spaces of Gaussians, that may be of independent interest.

7.1 Gaussian Mixture Models and applications

A K component GMM (K-GMM for short) is a probability density function given as a weighted sum of K Gaussian densities,

$$p(x|\Theta) = \sum_{j=1}^{K} \pi^j \mathcal{N}(x|\mu^j, \Sigma^j), \tag{7.1}$$

where the mean and covariance of the mixing components are given by μ^j and Σ^j respectively, π^j gives the corresponding weight and $\Theta = \{\mu^j, \Sigma^j\}_{j=1}^K$. Let $\mathbf{G} = \{\mathcal{G}_1^K, \cdots, \mathcal{G}_N^K\}$ denote a set of N K-GMMs. In this chapter, we study the problem of interpolating between $\mathcal{G}_1^K, \cdots, \mathcal{G}_N^K$ to derive an interpolant, $\hat{\mathcal{G}}$. Our main requirement on $\hat{\mathcal{G}}$ is that it should correspond to a K-GMM for a given K. In addition to this constraint, based upon the needs of the specific application, the interpolation task may correspond to an averaging operation over $\mathbf{G}$ or alternatively, when $|\mathbf{G}| = 2$, we may ask for a continuous interpolation $\Gamma(\mathcal{G}_i^K, t)$ such that $\Gamma(\mathcal{G}_i^K, 0) = \mathcal{G}_i^K$ and $\Gamma(\mathcal{G}_i^K, 1) = \mathcal{G}_j^K$ for any i, j and for any offset, $t \in [0, 1]$. The question of whether this problem permits efficient solution schemes is interesting enough in its own right to merit careful investigation. It turns out that such an algorithm, if available, will be immediately applicable to (or facilitate) a variety of tasks in computer vision, machine learning and medical imaging with minor changes. Besides the motivating examples of the interpolation of structured data discussed in Chapter 1.4, we provide specific applications of K-GMM interpolation below.

Problem 1: Spatial transformations of diffusion PDFs (Goh et al., 2009; Cheng et al., 2010, 2011). As we discussed in Chapter 1.4, interpolation of PDFs is a fundamental operation to handle images with diffusion PDFs. Here is an example. An important scientific frontier today is to establish a connectome of the human brain (Setsompop et al., 2013). Diffusion weighted magnetic resonance (MR) is one of the tools being used to help

answer the underlying analysis questions. It exploits the physical phenomenon of diffusion of water to image the microstructure of the white matter pathways in the brain (Cheng et al., 2010). An object estimated from such MR measurements is the so-called ensemble average propagator (EAP), a PDF describing the diffusivity profiles of water molecules on spheres of varying radii at the micrometer scale. The EAP can be conveniently represented as a K-GMM which can help resolve up to K crossing of white matter pathways at a voxel. Now, given two images (source and target) where each voxel has a K-GMM, the registration task involves applying a spatial transform to the source image to align it with the target image. Recall that the most basic routine needed in applying such a transformation is a way to estimate a 'value' for each voxel in the transformed image via interpolation (e.g., bi-linear). Since both the source and target images are a field of K-GMMs, an interpolation routine for K-GMMs is essential – in contrast, a naïve interpolation here will output a (NK)-GMM if $|\mathbf{G}| = N$, clearly blowing up the model complexity.

Problem 2: Matching point sets (Jian and Vemuri, 2011). Consider the problem of matching one point set to another where we seek the best alignment between the transformed "model" set and the target "scene" set — common in shape matching and model-based segmentation. In contrast to identifying point-to-point correspondence, a class of fairly successful recent approaches (Myronenko and Song, 2010) statistically model each of the two point sets by a PDF. Then, a suitable distance measure between the two distributions, $d(\cdot, \cdot)$ is minimized over the transformation parameters, τ. Kernel density based and GMM based representations are quite popular. Assume that the two point sets are defined as S and T. To align K-GMM$(\tau(S))$ and K-GMM(T), the optimization proceeds by taking incremental steps along $\nabla_\tau d$, until convergence. However, right after the first gradient update, we leave the feasibility region of K component GMMs. As a result, most methods are unable to provide intermediate

evolution steps along the transformation that are members of the same set as the source and the target models, i.e., a K-GMM. In contrast, with a minor modification (i.e., plugging in our method), this ability can be obtained with a nominal additional cost.

Problem 3: Statistical compressed sensing (Yu and Sapiro, 2011). Let $\mathbf{f} \in \mathbf{R}^p$ be a function (or signal) and $\Phi \in \mathbf{R}^{N \times p}$ denote the so-called sensing matrix. We are provided measurements $\boldsymbol{y} = \Phi\mathbf{f}$. The recovery of $\mathbf{f}$ from $\Phi\mathbf{f}$ is ill-posed in general when $N \ll p$. Compressed sensing significantly generalizes the regime under which such recovery is possible based on incoherence between the sensing and a certain 'representation' basis, see (Donoho, 2006). Statistical compressed sensing (SCS) takes this argument further by considering the situation where one is interested in reconstructing not just one but an entire sequence of signals, $\mathbf{f}_1, \mathbf{f}_2 \cdots$. Here, SCS assumes that $\mathbf{f}_i$ is drawn from a GMM — which enables additional improvements in recovery. When deployed in a 'streaming' setup, the current GMM prior in SCS (say, at time t) is incrementally updated based on the current measurement $(t+1)$. Our proposed algorithm offers a potential improvement: by providing a *moving average* version of the to-be-updated GMM prior by constructing a weighted (or unweighted) mean of the previous t GMMs. This will likely be immune to local fluctuations or noise in the streaming measurements.

In this chapter, we develop a systematic framework for performing interpolation on the manifold of K component GMMs. It will take in as input a set of GMMs and a specific interpolation task and provide a K-GMM as an output that optimizes the interpolation objective. While the primary focus of this work is theoretical, we provide experiments demonstrating the expected behavior of the algorithm. Separately, we highlight some interesting connections of this formulation with the *functional* spaces of Gaussians. Next, Section 7.2 introduces some basic concepts relevant to K-GMMs. With the ℓ_2-distance metric, Sections 7.3 studies the interpolation

of GMMs and Section 7.4 introduces a numerical scheme to identify the shortest path on the K-GMM manifold. Section 7.5 presents our main EM algorithm and a modified EM algorithm to minimize the KL-divergence between the learned K-GMM and given GMMs. Finally, experimental results and conclusions are discussed in Sections 7.6 and 7.7 respectively.

7.2 Parameterization and distance measures

To our knowledge, there are no existing algorithms for interpolating a set of K-GMMs which also control the number of components; on the other hand, there *is* a mature body of research for tackling the setting where the objects to be interpolated are probability density functions (PDFs) (Srivastava et al., 2007; Cetingul et al., 2012; Ncube et al., 2012; Li et al., 2014). So one might ask, why not simply use PDFs? We will present several specific reasons in the section below.

Observe that the actual formulation for interpolation will depend on the specific parameterization we choose to represent the PDF as well as the distance metric. To make this point concrete, let us review a few example parameterizations and distance metrics. With these two pieces, the corresponding interpolation/averaging operation is simple to derive. Evaluation of their advantages or limitations in the K-GMM setting will then become apparent.

7.2.1 PDF parameterizations and distances

Parameterization. First, let us consider a simple expression for computing the mean of probability densities $\{f_i\}_{i=1}^N$,

$$\hat{f} = \arg\min_{f \in \mathcal{F}} \sum_{i=1}^{N} w_i d(\Phi(f), \Phi(f_i))^2 \tag{7.2}$$

where $\Phi(\cdot)$ is for parameterizing the given probability densities, $d(\cdot,\cdot)$ is a distance metric and w_i is a weight for f_i. Some parameterizations will allow using tools from differential geometry for deriving efficient algorithms (Srivastava et al., 2007). Clearly, there are multiple options for parameterization but some specific ones form a set (the so called unit Hilbert sphere in ℓ_2-space) and are mathematically convenient. We can parameterize a given set of PDFs so that they lie in this set. The mapping is bijective when restricted to non-negative functions i.e., *every* element in the unit Hilbert sphere can be mapped back to a PDF. For example, the square root parameterization simply takes the square-root of the PDF value. If, for example, the PDF was parameterized using a K-GMM then,

$$f(x|\Theta) = \sqrt{p(x|\Theta)} = \sqrt{\sum_{j=1}^{K} \pi^j \mathcal{N}(x|\mu^j, \Sigma^j)} \tag{7.3}$$

By inspection, the ℓ_2-norm of f is always 1 since $\sqrt{\int f(x)f(x)dx} = \int p(x)dx = 1$. Notice that this is a *re*-parameterization of the original PDF (which was provided as a K-GMM).

Normalization. Alternatively, we can normalize the PDFs by dividing with the ℓ_2-norm, which only changes the scale and *not* the shape of the model.

$$p'(x) = p(x)/\|p(x)\|_2, \tag{7.4}$$

where $\|\cdot\|_2$ is the standard ℓ_2-norm for functions. For the special case of GMMs, we have

$$\|\mathcal{G}_i\|_2^2 = \sum_{j}^{K} \sum_{j'}^{K} \pi^j \pi^{j'} \mathcal{N}(\mu^j|\mu^{j'}, \Sigma^j + \Sigma^{j'}), \tag{7.5}$$

where $\mathcal{G}_i$ denotes a representative GMM.

Distances. Let us now consider the calculation of distances. Let

$p_i'(x) = p_i(x)/\|p_i(x)\|_2$. Recall that for two different functions, the ℓ_2-distance is given as

$$\|f_1 - f_2\|_2 = \left(\int_X |f_1(x) - f_2(x)|^2 d\mu(x)\right)^{1/2}. \tag{7.6}$$

Then, the normalized ℓ_2-distance ($d_{n\text{-}\ell_2}$) is simply the ℓ_2-distance between the normalized PDFs (Jensen et al., 2007),

$$\begin{aligned} d_{n\text{-}\ell_2}(p_1, p_2) &= \int (p_1'(x) - p_2'(x))^2 dx \\ &= 2(1 - \int_{\mathcal{X}} p_1'(x) p_2'(x) dx). \end{aligned} \tag{7.7}$$

Geodesics and Divergences. Instead of the ℓ_2-distance, we can also calculate the geodesic distance on the unit Hilbert sphere. Let $p_i'(x) = p_i(x)/\|p_i(x)\|_2$. Then, the geodesic distance between normalized PDFs is

$$d_{n\text{-geo}}(p_1, p_2) = \cos^{-1}\langle p_1', p_2'\rangle_2 = \cos^{-1}(\int_{\mathcal{X}} p_1'(x) p_2'(x) dx)$$

This is interesting because the geodesic distance here admits a closed form solution.

The KL-divergence (Kullback, 1997) is another possibility, albeit *not* a metric, that can be used as a information theoretic divergence between probability density functions $f(x)$ and $g(x)$. It is also known as relative entropy and given by

$$D(f\|g) := \int f(x) \log \frac{f(x)}{g(x)} dx. \tag{7.8}$$

The KL-divergence between two GMMs cannot be obtained analytically and so various approximations have been proposed (Hershey and Olsen, 2007). Shortly, we will discuss the relationship between the KL-divergence/cross entropy and the log likelihood which will suggest natural EM style algo-

rithms.

How many components? PDFs and K-GMMs. With these concepts in hand, it is easy to verify what happens when we seek to interpolate GMMs but the only tool we have available is an interpolation routine for PDFs. In general, given a set of GMMs, if we consider them simply as PDFs, the mean derived from the geodesic distance (with the square root parameterization) may not even be a GMM. However, it turns out that the simple arithmetic mean of PDFs, i.e., $\bar{f} = \sum_i^N f_i/N$ is optimal with respect to the ℓ_2-metric for PDFs. It is easy to check and the proof is provided shortly in Lemma 7.1.

Unfortunately, the main difficulty is that when given N GMMs with K components each, the arithmetic mean solution will *not* be a K component GMM (instead, a GMM with $N \times K$ components),

$$\bar{\mathcal{G}} = \sum_i^n \mathcal{G}_i/N = \underbrace{\sum_{i=1}^{N}\sum_{j=1}^{K}}_{N\times K \text{ components}} \frac{\pi_i^j}{N}\mathcal{N}(\mu_i^j, \Sigma_i^j).$$

If one needs the interpolation of K-GMMs to be a K-GMM, to our knowledge, there are no existing solutions. We address this problem in the later sections with a focus on ℓ_2-distance and KL-divergence/cross entropy which, roughly speaking, corresponds to the least squares and log-likelihood functions of a finite number of samples in the classical GMM setting.

7.3 Interpolation w.r.t. ℓ_2-distance

Let $\mathbf{G}^{(K)}$ denote the manifold of K-GMMs. We will first describe an optimization scheme to directly minimize the ℓ_2-distance in $\mathbf{G}^{(K)}$ which is used for the interpolation objective.

Computing the ℓ_2-mean in $\mathbf{G}^{(K)}$. First, we will derive an algorithm for calculating the mean for a set $\mathbf{F} = \{\mathcal{F}_1, \cdots, \mathcal{F}_n\}$ where $\forall j, \mathcal{F}_i \in \mathbf{G}^{(K)}$ w.r.t.ℓ_2 metric. Second, for the case where $|\mathbf{F}| = 2$, we will derive a 'path' from $\mathcal{F}_i$ to $\mathcal{F}_j$, which never leaves the feasibility region i.e., $\mathbf{G}^{(K)}$. This construction will provide a meaningful distance measure which respects the geometry of $\mathbf{G}^{(K)}$.

The ℓ_2-mean (arithmetic mean) of $\{\mathcal{F}_n\}_{n=1}^N$ minimizes the sum of squared ℓ_2-distances to each $\mathcal{F}_i \in \mathbf{F}$,

$$\bar{\mathcal{F}} = \arg\min_{\mathcal{G}} \sum_{n=1}^{N} \|\mathcal{G} - \mathcal{F}_n\|_2^2 \tag{7.9}$$

As discussed in Section 7.2, we have $\bar{\mathcal{F}} \in \mathbf{G}^{(NK)}$ (the blowup in the number of components). Instead, we require a GMM $\hat{\mathcal{G}} \in \mathbf{G}^{(K)}$. Our algorithm has two steps. First, we find $\bar{\mathcal{F}}$ and then find the closest K-component GMM to $\bar{\mathcal{F}}$, i.e., we will minimize (7.10)

$$\hat{\mathcal{G}} = \arg\min_{\mathcal{G} \in \mathbf{G}^{(K)}} \|\mathcal{G} - \bar{\mathcal{F}}\|_2^2 \tag{7.10}$$

This may seem like a very loose relaxation. In other words, is there a $\hat{\mathcal{G}}' \in \mathbf{G}^{(K)}$ that is farther from $\bar{\mathcal{F}}$ but achieves a lower objective function value for (7.9)? The following result shows that this cannot be the case.

Lemma 7.1. *The mean of a finite number of functions $\{\mathcal{F}_n\}_n^N$ with respect to ℓ_2 metric is the closest $\mathcal{G}^*$ to the ℓ_2-mean $\bar{\mathcal{F}} = \sum_n^N \frac{\mathcal{F}_n}{N}$.*

Proof.

$$
\begin{aligned}
\mathcal{G}^* &= \arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} \sum_n^N \|\mathcal{F}_n - \mathcal{G}\|_2^2 \\
&= \arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} \sum_n^N \|\mathcal{F}_n\|_2^2 - 2\sum_n^N \langle \mathcal{F}_n, \mathcal{G}\rangle_2 + N\|\mathcal{G}\|_2^2 \\
&= \arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} \frac{1}{N}\sum_n^N \|\mathcal{F}_n\|_2^2 - 2\left\langle \frac{\sum_n^N \mathcal{F}_n}{N}, \mathcal{G}\right\rangle_2 + \|\mathcal{G}\|_2^2 \\
&= \arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} -2\left\langle \frac{\sum_n^N \mathcal{F}_n}{N}, \mathcal{G}\right\rangle_2 + \|\mathcal{G}\|_2^2 \\
&= \arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} \|\frac{1}{N}\sum_n^N \mathcal{F}_n\|_2^2 - 2\langle \frac{1}{N}\sum_n^N \mathcal{F}_n, \mathcal{G}\rangle_2 + \|\mathcal{G}\|_2^2 \\
&= \arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} \|\frac{1}{N}\sum_n^N \mathcal{F}_n - \mathcal{G}\|_2^2 \\
&= \arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} \|\bar{\mathcal{F}} - \mathcal{G}\|_2^2
\end{aligned}
$$

□

This result suggests that (7.10) is indeed equivalent to (7.9) with the constraint $\mathcal{G} \in \mathbf{G}^{(K)}$. Also it shows that the arithmetic mean is the optimal solution in the ambient space (or L^2 space *not* $\mathbf{G}^{(K)}$), i.e., $\mathcal{G}^* = \bar{\mathcal{F}} = \frac{\sum_n^N \mathcal{F}_n}{N}$.

Optimization scheme. To optimize (7.10), we first initialize the solution and then perform incremental gradient descent steps. The main terms in the gradient update step are described below and are computed using $\bar{\mathcal{F}}$ and $\mathcal{G}$, the former has $L(= NK)$ components and the latter has K components.

Let $\mathcal{L}$ denote the objective function in (7.10). The three main variables to optimize over are the component weights $\pi_{\mathcal{G}}^j$, means $\mu_{\mathcal{G}}^j$ and covariances $\Sigma_{\mathcal{G}}^j$, where i and j index components in $\bar{\mathcal{F}}$ and $\mathcal{G}$ respectively. Let

$c^{j,i}_{\mathcal{G},\bar{\mathcal{F}}} := \mathcal{N}(\mu^j_{\mathcal{G}}|\mu^i_{\mathcal{F}}, \Sigma^j_{\mathcal{G}} + \Sigma^i_{\mathcal{F}})$. The derivative w.r.t. $\pi^j_{\mathcal{G}}$ takes the form,

$$\frac{\partial \mathcal{L}}{\partial \pi^j_{\mathcal{G}}} = 2\left(\sum_{j'=1}^{K} \pi^{j'}_{\mathcal{G}} c^{j,j'}_{\mathcal{G},\mathcal{G}} - \sum_{i=1}^{L} \pi^i_{\bar{\mathcal{F}}} c^{j,i}_{\mathcal{G},\bar{\mathcal{F}}}\right) \tag{7.11}$$

The derivative w.r.t. $\mu^j_{\mathcal{G}}$ is given as

$$\frac{\partial \mathcal{L}}{\partial \mu^j_{\mathcal{G}}} = 2\pi^j_{\mathcal{G}}\left(\sum_{j'\neq j}^{K} \pi^{j'}_{\mathcal{G}} \frac{\partial}{\partial \mu^j_{\mathcal{G}}} c^{j,j'}_{\mathcal{G},\mathcal{G}} - \sum_{i=1}^{L} \pi^i_{\bar{\mathcal{F}}} \frac{\partial}{\partial \mu^j_{\mathcal{G}}} c^{j,i}_{\mathcal{G},\bar{\mathcal{F}}}\right), \tag{7.12}$$

whereas the derivative $\frac{\partial \mathcal{L}}{\partial \Sigma^j_{\mathcal{G}}}$ is

$$\left(\pi^j_{\mathcal{G}}\right)^2 \frac{\partial}{\partial \Sigma^j_{\mathcal{G}}} c^{j,j}_{\mathcal{G},\mathcal{G}} + 2\pi^j_{\mathcal{G}}\left(\sum_{j'\neq j}^{K} \pi^{j'}_{\mathcal{G}} \frac{\partial}{\partial \Sigma^j_{\mathcal{G}}} c^{j,j'}_{\mathcal{G},\mathcal{G}} - \sum_{i=1}^{L} \pi^i_{\mathcal{F}} \frac{\partial}{\partial \Sigma^j_{\mathcal{G}}} c^{j,i}_{\mathcal{G},\bar{\mathcal{F}}}\right). \tag{7.13}$$

The gradient is calculated by putting together the three terms above and the step size is determined using a standard line search procedure (Nocedal and Wright, 2006a). We repeat until convergence.

For the detailed derivations of the derivatives above, we start with the partial derivatives of the Gaussian distribution.

7.3.1 Gaussian distribution and its derivatives

A Gaussian distribution has parameters (μ, Σ). Since Σ is matrix-valued, we need derivatives of functions w.r.t. a matrix. Suppose $X \in \mathbf{R}^{d\times d}$. We have the following.

(i) $\det(cX) = c^d \det(X)$, where A is a $n \times n$ matrix.

(ii) $\frac{\partial \det(X)}{\partial X} = \det(X)X^{-T}$, see (Petersen and Pedersen, 2012b).

(iii) $\frac{\mathbf{a}^T X^{-1}\mathbf{b}}{\partial X} = -X^T\mathbf{a}\mathbf{b}^T X^{-T}$, $\forall \mathbf{a}, \mathbf{b} \in \mathbf{R}^d$, see (Petersen and Pedersen, 2012b).

(iv) $\frac{\partial}{\partial X}\log|\det(X)| = X^{-T}$,

where for a matrix A, A^{-T} denotes an inverse followed by a transpose. We will use the facts above in our derivations. Places where they are used, are denoted by the number over the equality e.g. " $\overset{\text{(i)}}{=}$ ".

The density function of Gaussian is given by

$$f(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{\det(2\pi\Sigma)}} \exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]. \tag{7.14}$$

The derivative w.r.t. μ is given by

$$\frac{\partial f(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma)}{\partial \boldsymbol{\mu}} = f(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma)\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu}). \tag{7.15}$$

The derivative w.r.t. Σ is given by

$$
\begin{aligned}
\frac{\partial f(\boldsymbol{x}|\boldsymbol{\mu},\Sigma)}{\partial \Sigma} &= \frac{\partial}{\partial \Sigma}\left[\det(2\pi\Sigma)^{-1/2}\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]\right] \\
&\overset{\text{(i)}}{=} (2\pi)^{-d/2}\frac{\partial}{\partial \Sigma}\left[\det(\Sigma)^{-1/2}\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]\right] \\
&= (2\pi)^{-d/2}\left(\frac{\partial}{\partial \Sigma}\det(\Sigma)^{-1/2}\right)\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right] \\
&\quad + (2\pi)^{-d/2}\det(\Sigma)^{-1/2}\left(\frac{\partial}{\partial \Sigma}\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(x-\boldsymbol{\mu})\right]\right) \\
&\overset{\text{(ii)}}{=} (2\pi)^{-d/2}\left(-\frac{1}{2}\det(\Sigma)^{-3/2}\det(\Sigma)\Sigma^{-T}\right)\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right] \\
&\quad + (2\pi)^{-d/2}\det(\Sigma)^{-1/2}\left(\frac{\partial}{\partial \Sigma}\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]\right) \\
&= -\frac{1}{2}f(\boldsymbol{x}|\boldsymbol{\mu},\Sigma)\Sigma^{-T} \\
&\quad + (2\pi)^{-d/2}\det(\Sigma)^{-1/2}\left(\frac{\partial}{\partial \Sigma}\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]\right) \\
&\overset{\text{(iii)}}{=} -\frac{1}{2}f(\boldsymbol{x}|\boldsymbol{\mu},\Sigma)\Sigma^{-T} + \frac{1}{2}f(\boldsymbol{x}|\boldsymbol{\mu},\Sigma)\Sigma^{-T}(\boldsymbol{x}-\boldsymbol{\mu})(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-T}
\end{aligned}
$$

7.3.2 Derivatives for ℓ_2 minimization in (7.9)

The loss function (7.9) consists of the ℓ_2 norms of GMMs. We complete the derivation of (7.9) with the partial derivatives of $c^{j,i}_{\mathcal{G},\mathcal{F}} := \mathcal{N}(\boldsymbol{\mu}^j_{\mathcal{G}}|\boldsymbol{\mu}^i_{\bar{\mathcal{F}}}, \Sigma^i_{\mathcal{G}} + \Sigma^i_{\mathcal{F}})$ w.r.t $\boldsymbol{\mu}_{\mathcal{G}}$ and $\Sigma_{\mathcal{G}}$. The derivatives are obtained by derivatives of Gaussian in (7.15) and (7.16).

The derivatives w.r.t $\boldsymbol{\mu}^j_{\mathcal{G}}$ are given by

$$
\begin{aligned}
\frac{\partial c^{j,i}_{\mathcal{G},\bar{\mathcal{F}}}}{\partial \boldsymbol{\mu}^j_{\mathcal{G}}} &= -c^{j,i}_{\mathcal{G},\bar{\mathcal{F}}}(\Sigma^j_{\mathcal{G}} + \Sigma^i_{\bar{\mathcal{F}}})^{-1}(\boldsymbol{\mu}^j_{\mathcal{G}} - \boldsymbol{\mu}^i_{\bar{\mathcal{F}}}), \\
\frac{\partial c^{j,j'}_{\mathcal{G},\mathcal{G}}}{\partial \boldsymbol{\mu}^j_{\mathcal{G}}} &= -c^{j,j'}_{\mathcal{G},\mathcal{G}}(\Sigma^j_{\mathcal{G}} + \Sigma^{j'}_{\mathcal{G}})^{-1}(\boldsymbol{\mu}^j_{\mathcal{G}} - \boldsymbol{\mu}^{j'}_{\mathcal{G}}).
\end{aligned}
\tag{7.16}
$$

The derivatives w.r.t $\Sigma_{\mathcal{G}}^{j}$ are given by

$$\begin{aligned}\frac{\partial c_{\mathcal{G},\bar{\mathcal{F}}}^{j,i}}{\partial \Sigma_{\mathcal{G}}^{j}} &= -\frac{1}{2} c_{\mathcal{G},\bar{\mathcal{F}}}^{j,i} (\Sigma_{\mathcal{G}}^{j} + \Sigma_{\bar{\mathcal{F}}}^{i})^{-T} [I - (\boldsymbol{\mu}_{\mathcal{G}}^{j} - \boldsymbol{\mu}_{\bar{\mathcal{F}}}^{i})(\boldsymbol{\mu}_{\mathcal{G}}^{j} - \boldsymbol{\mu}_{\bar{\mathcal{F}}}^{i})^{T} (\Sigma_{\mathcal{G}}^{j} + \Sigma_{\bar{\mathcal{F}}}^{i})^{-T}] \\ \frac{\partial c_{\mathcal{G},\mathcal{G}}^{j,j'}}{\partial \Sigma_{\mathcal{G}}^{j}} &= -\frac{1}{2} c_{\mathcal{G},\mathcal{G}}^{j,j'} (\Sigma_{\mathcal{G}}^{j} + \Sigma_{\mathcal{G}}^{j'})^{-T} [I - (\boldsymbol{\mu}_{\mathcal{G}}^{j} - \boldsymbol{\mu}_{\mathcal{G}}^{j'})(\boldsymbol{\mu}_{\mathcal{G}}^{j} - \boldsymbol{\mu}_{\mathcal{G}}^{j'})^{T} (\Sigma_{\mathcal{G}}^{j} + \Sigma_{\mathcal{G}}^{j'})^{-T}].\end{aligned} \tag{7.17}$$

When $j = j'$, it is simplified as

$$\frac{\partial c_{\mathcal{G},\mathcal{G}}^{j,j}}{\partial \Sigma_{\mathcal{G}}^{j}} = -2^{-(d/2+1)} c_{\mathcal{G},\mathcal{G}}^{j,j} (\Sigma_{\mathcal{G}}^{j})^{-T} = -\frac{1}{2} \mathcal{N}(\boldsymbol{\mu}_{\mathcal{G}}^{j} | \boldsymbol{\mu}_{\mathcal{G}}^{j}, 2\Sigma_{\mathcal{G}}^{j}) (\Sigma_{\mathcal{G}}^{j})^{-1}. \tag{7.18}$$

7.4 Identifying a path in $\mathbf{G}^{(K)}$ between $\mathcal{F}_{start}$ and $\mathcal{F}_{end}$ w.r.t ℓ_2 distance

A special case for the interpolation scheme above is when we want to interpolate between just two K component GMMs, $\mathcal{F}_{\text{start}}$ and $\mathcal{F}_{\text{end}}$, and recover a shortest path $\{\mathcal{G}_t\}_{t=1}^{T}$ that does not leave the feasibility region, $\mathbf{G}^{(K)}$ and $\mathcal{G}_0 = \mathcal{F}_{\text{start}}$ and $\mathcal{G}_{T+1} = \mathcal{F}_{\text{end}}$. As can be expected, one can identify such a path with a minor change of the algorithm described above. Then, our objective function is,

$$\min_{\{\mathcal{G}_t\}_{t=1}^{T}} \sum_{t=0}^{T} \|\mathcal{G}_t - \mathcal{G}_{t+1}\|_2^2, \text{ s.t. } \mathcal{G}_t \in \mathbf{G}^{(K)} \; \forall t. \tag{7.19}$$

Letting $d_T := \sum_{t=0}^{T} \|\mathcal{G}_t^* - \mathcal{G}_{t+1}^*\|_2$, we have $\lim_{T\to} d_T = d(\mathcal{F}_{\text{start}}, \mathcal{F}_{\text{end}})$, the geodesic distance between $\mathcal{F}_{\text{start}}$ and $\mathcal{F}_{\text{end}}$ in $\mathbf{G}^{(K)}$. Given two GMMs, $\mathcal{F}_{\text{start}}$ and $\mathcal{F}_{\text{end}}$, we seek to find the shortest path which does not leave

the feasibility region, $\mathbf{G}^{(K)}$. The result from such a procedure will directly provide a potentially more meaningful distance measure between two samples in $\mathbf{G}^{(K)}$.

To do so, we will approximate it by a set of smaller paths along other GMMs with K components. By minimizing the sum of the squared distances between adjacent GMMs, we will approximate the shortest path. It is similar to the cumulative chordal distance (Ahlberg et al., 2016), approximation on the sphere, see Figure 7.1. In the limit, this will be the true shortest length.

On a Riemannian manifold $\mathcal{M}$ with metric tensor g, the length of a continuously differentiable curve $\gamma : [a, b] \to \mathcal{M}$ is defined by

$$L(\gamma) = \int_a^b \sqrt{g_{\gamma(t)}(\dot{\gamma}(t), \dot{\gamma}(t))}\, dt, \tag{7.20}$$

where L is its length and $g_{\gamma(t)}(\dot{\gamma}(t), \dot{\gamma}(t))$ is the inner product of $\dot{\gamma}(t)$ at $\gamma(t)$ w.r.t g. When γ is the shortest geodesic curve, it is called *geodesic distance*.

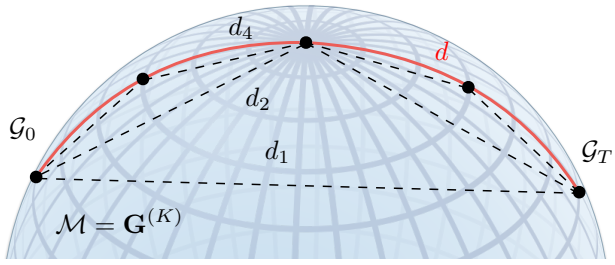


Figure 7.1: ℓ_2 distance between $\mathcal{G}_0$ and $\mathcal{G}_T$ is d_1. The geodesic distance between two GMMs can be approximated by the sum of ℓ_2 distances between many intermediate K-GMMs. It converges to the real path length as the number of chords t increases. Here $d_1 \leq d_2 \leq ... \leq d$, where d is the true distance.

Given γ, the arc length L of γ can be approximated by

$$L(C) = \sup_{a=t_0<t_1<\cdots<t_n=b} \sum_{i=0}^{n-1} d(\gamma(t_i), \gamma(t_{i+1})), \tag{7.21}$$

where the supremum is taken over all possible partitions of $[a, b]$ and n is unbounded. But we do not know the curve γ. To seek the γ on $\mathbf{G}^{(K)}$, we adopt the definition of the geodesic curve. A geodesic is a locally shortest metric curve (Lee, 2006; Deza and Deza, 2009). So we can approximate the geodesic with discretized line segments that minimizes the sum of squared distances of each chordal segment.

Let $\mathcal{G}_0$ ($\mathcal{G}_{T+1}$ resp.) be $\mathcal{G}_{\text{start}}$ ($\mathcal{G}_{\text{end}}$ resp.). Then, our objective function is given as

$$\min \mathcal{L} := \min_{\{\boldsymbol{\mu}_t, \boldsymbol{\pi}_t, \boldsymbol{\Sigma}_t\}_{t=1}^T} \sum_{t=0}^{T} \|\mathcal{G}_t - \mathcal{G}_{t+1}\|_2^2, \tag{7.22}$$

using the shorthand notations $\boldsymbol{\pi}_t := \{\pi_t^j\}_{j=1}^K$, $\boldsymbol{\mu}_t := \{\mu_t^j\}_{j=1}^K$ and $\boldsymbol{\Sigma}_t := \{\boldsymbol{\Sigma}_t^j\}_{j=1}^K$. Again, to compute the gradient, we take the derivative with respect to the relevant variables which include the component weights, means and their covariances. Similarly, define $c_{t,t'}^{j,j'} := \mathcal{N}(\boldsymbol{\mu}_t^j | \boldsymbol{\mu}_{t'}^{j'}, \Sigma_t^j + \Sigma_{t'}^{j'})$. The derivatives of (7.21) w.r.t π_t^j, μ_t^j, Σ_t^j are related to only the following terms

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}_t} &:= \frac{\partial}{\partial \boldsymbol{\mu}_t} \left[\|\mathcal{G}_{t-1} - \mathcal{G}_t\|_2^2 + \|\mathcal{G}_t - \mathcal{G}_{t+1}\|_2^2 \right] \\
&= \frac{\partial}{\partial \boldsymbol{\mu}_t} \left[\|\mathcal{G}_{t-1}\|_2^2 - 2\langle \mathcal{G}_{t-1}, \mathcal{G}_t \rangle_2 + 2\|\mathcal{G}_t\|_2^2 - 2\langle \mathcal{G}_t, \mathcal{G}_{t+1} \rangle_2 + \|\mathcal{G}_{t+1}\|_2^2 \right] \\
&= \frac{\partial}{\partial \boldsymbol{\mu}_t} \left[2\|\mathcal{G}_t\|_2^2 - 2\langle \mathcal{G}_t, \mathcal{G}_{t+1} \rangle_2 - 2\langle \mathcal{G}_t, \mathcal{G}_{t-1} \rangle_2 \right].
\end{aligned} \tag{7.23}$$

Recall that the inner product of two GMMs $\mathcal{G}_t$ and $\mathcal{G}_{t'}$ in $\mathbf{G}^{(K)}$ is given by

$$\langle \mathcal{G}_t, \mathcal{G}_{t'} \rangle_2 = \sum_{j=1}^{K} \sum_{j'=1}^{K} \pi_t^j \pi_{t'}^{j'} \mathcal{N}(\mu_t^j | \mu_{t'}^{j'}, \Sigma_t^j + \Sigma_{t'}^{j'}) = \sum_{j=1}^{K} \sum_{j'=1}^{K} \pi_t^j \pi_{t'}^{j'} c_{t,t'}^{j,j'} \tag{7.24}$$

Then the derivatives w.r.t π_t^j, μ_t^j, Σ_t^j can be written with $c_{t,t'}^{j,j'}$. The derivative w.r.t π_t^j is given as

$$\frac{\partial \mathcal{L}}{\partial \pi_t^j} = 4 \sum_{j'}^{K} \pi_t^{j'} c_{t,t}^{j,j'} - 2 \sum_{j'}^{K} \pi_{i+1}^{j'} c_{t,t+1}^{j,j'} - 2 \sum_{j'}^{K} \pi_{i-1}^{j'} c_{t,t-1}^{j,j'} \tag{7.25}$$

The derivative w.r.t μ_t^j is

$$\frac{\partial \mathcal{L}}{\partial \mu_t^j} = 4 \sum_{j' \neq j}^{K} \pi_t^j \pi_t^{j'} \frac{\partial}{\partial \mu_t^j} c_{t,t}^{j,j'} - 2 \sum_{j'=1}^{K} \pi_t^j \pi_{t+1}^{j'} \frac{\partial}{\partial \mu_t^j} c_{t,t+1}^{j,j'} - 2 \sum_{j'=1}^{K} \pi_t^j \pi_{i-1}^{j'} \frac{\partial}{\partial \mu_t^j} c_{t,t-1}^{j,j'} \tag{7.26}$$

The derivative w.r.t Σ_t^j

$$\frac{\partial \mathcal{L}}{\partial \Sigma_t^j} = 2\pi_t^j \pi_t^j \frac{\partial}{\partial \Sigma_t^j} c_{t,t}^{j,j} + 4 \sum_{j' \neq j}^{K} \pi_t^j \pi_t^{j'} \frac{\partial}{\partial \Sigma_t^j} c_{t,t}^{j,j'} - 2 \sum_{j'=1}^{K} \pi_t^j \pi_{t+1}^{j'} \frac{\partial}{\partial \Sigma_t^j} c_{t,t+1}^{j,j'} - 2 \sum_{j'=1}^{K} \pi_t^j \pi_{t-1}^{j'} \frac{\partial}{\partial \Sigma_t^j} c_{t,t-1}^{j,j'} \tag{7.27}$$

Interpolation path between two 2-GMMs in $\mathbf{G}^{(2)}$ is shown in Fig. 7.2 as a demonstration, see Fig. (7.22).

7.5 An EM algorithm for KL-divergence

Our initial experiments reveal that minimizing ℓ_2-distance via gradient descent with the constraint of staying on the $\mathbf{G}^{(K)}$ manifold is technically correct but prone to instability due to many local optima. For example, the gradient descent method works well when the covariance matrices are diagonally dominant (isotropic) but tends to yield unsatisfactory re-

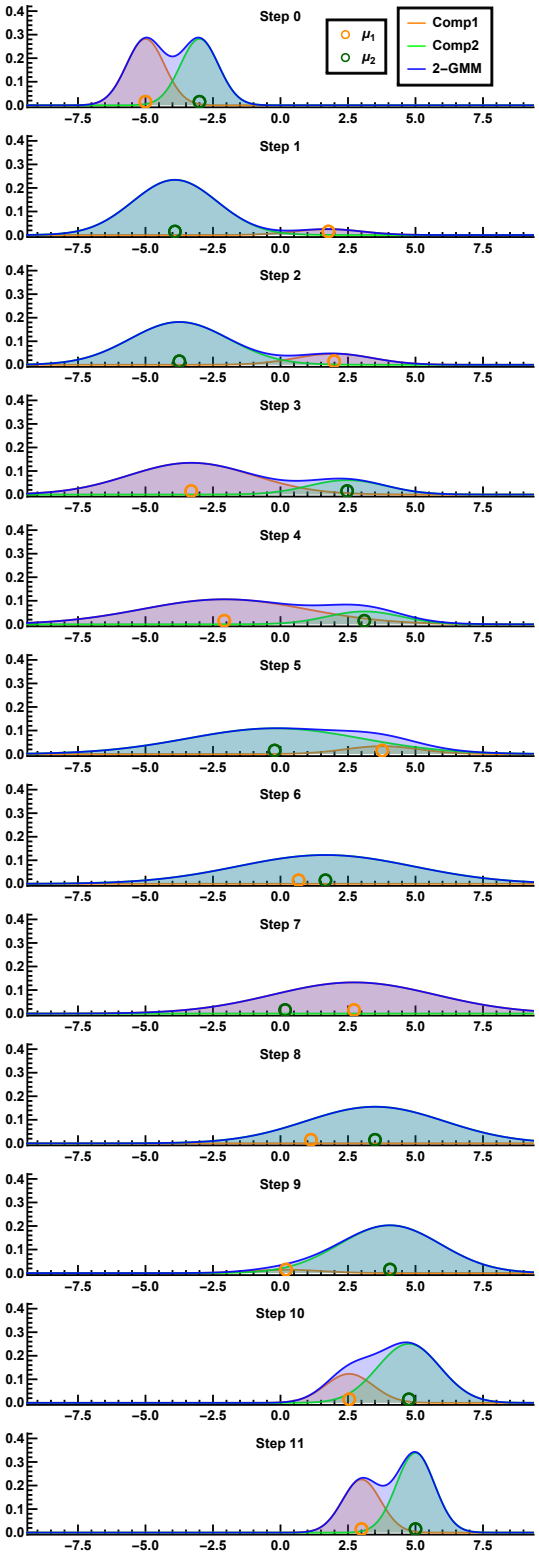


Figure 7.2: Interpolation path along 2-GMM manifold showing 10 steps from top (GMM_0) to bottom (GMM_{11}).

sults when the estimated covariances matrices need to be projected back to satisfy the "$\succeq 0$" constraint. To address this issue, we describe an alternate algorithm that avoids such a projection step. To motivate this setup, observe that in the preceding section, the overall interpolation task comprised of modules/steps for finding the closest K-GMM to a given L component GMM, see (7.10). So, any potential solution to the foregoing numerical issue must be addressed at the level of this module.

Consider a very special case of the module above where L is arbitrary but $K = 1$. Interestingly, it turns out that if we use KL-divergence instead of the ℓ_2-distance between GMMs, Lemma 7.2 suggests that that there *is* a closed form solution which involves no numerical difficulties. Notice that no such result exists for ℓ_2-distance. So, if we can extend this result to the case where $K > 1$, we can efficiently solve the problem while ensuring that the procedure is numerically stable. In fact, this idea will form the core of our proposal described next where we first decouple the components in the "E" step and use a closed form solution for each component in the "M" step. In fact, our scheme optimizes the KL-divergence which is equivalent to cross-entropy in this case.

The interpolation of multiple GMMs is obtained by minimizing,

$$\mathcal{G}^* = \arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} \sum_{n=1}^{N} D(\mathcal{F}_n \| \mathcal{G}). \tag{7.28}$$

We observe that the expression in (7.28) is equivalent to,

$$\begin{aligned} &\arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} D(\bar{\mathcal{F}}\|\mathcal{G}) \\ &= \arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} \int \bar{\mathcal{F}}(x) \log \frac{\bar{\mathcal{F}}(x)}{\mathcal{G}(x)} dx \\ &= \arg\min_{\mathcal{G}\in\mathbf{G}^{(K)}} -\int \bar{\mathcal{F}}(x) \log \mathcal{G}(x) dx. \end{aligned} \tag{7.29}$$

Letting $\mathcal{G}(x) = \sum_{j=1}^{K} w_j g_j(x)$, the objective function is given by

$$\begin{aligned} \mathcal{G}^* &= \arg\max_{\mathcal{G}\in\mathbf{G}^{(K)}} \int \bar{\mathcal{F}}(x) \log \sum_{j=1}^{K} w_j g_j(x) dx, \\ &= \arg\max_{g\in\mathbf{G}^{(K)}} \mathbb{E}_{\bar{\mathcal{F}}(x)}[\log \sum_{j=1}^{K} w_j g_j(x)]. \end{aligned} \tag{7.30}$$

We note that this formulation can also be interpreted as finding the best code book in $\mathbf{G}^{(K)}$, namely, $\mathcal{G}^*(x)$ to represent $\bar{\mathcal{F}}(x)$. The E and M steps are presented in Fig. 11. Detailed derivations are provided in Section 7.5.4.

Lemma 7.2. *Given GMM $f(x) := \sum_i^L \pi_i f_i(x)$, where $f_i(x)$ is a Gaussian distribution, the minimum cross entropy / KL-divergence between $f(x)$ and an unknown single Gaussian $g := \mathcal{N}(x; \mu, \Sigma)$, i.e.,*

$$(\mu^*, \Sigma^*) = \arg\min_{\mu,\Sigma} H(f(x), \mathcal{N}(x; \mu, \Sigma)), \tag{7.31}$$

is obtained by $\mu^* = \mathbb{E}_{f(x)}[x] = \sum_{i=1}^{NK} \pi_i' \mu_i$, and $\Sigma^* = \mathbb{E}_{f(x)}[(x-\mu^*)(x-\mu^*)^T] = \sum_{i=1}^{NK} \pi_i' \Sigma_i + \sum_{i=1}^{NK} \pi_i' (\mu_i - \mu_j)(\mu_i - \mu_j)^T$, where $\pi_i' = \frac{\pi_i \gamma_{ij}}{\sum_i \pi_i \gamma_{ij}}$, for fixed j.

The closed form of (μ^*, Σ^*) is used in (7.38) and (7.39).

Proof. We can easily observe that

$$\arg\min_g D(f\|g) = \arg\min_g H(f, g), \tag{7.32}$$

since f is fixed. Recall that cross entropy is given by

$$H(f, g) := E_f[-\log g(x)] = -\int f(x) \log g(x) dx. \tag{7.33}$$

Take the derivative of objective function $H(f,g)$ w.r.t μ and set it to zero. Then we get,

$$\begin{aligned}
-\frac{\partial}{\partial \boldsymbol{\mu}} \int f(\boldsymbol{x}) \log g(\boldsymbol{x}) dx &= \kappa \frac{\partial}{\partial \boldsymbol{\mu}} \int f(\boldsymbol{x})(\boldsymbol{x}-\boldsymbol{\mu})^T \Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu}) dx \\
&= \kappa' \int f(\boldsymbol{x}) \Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu}) dx = c' \Sigma^{-1}\Big(\int f(\boldsymbol{x}) \boldsymbol{x} dx - \boldsymbol{\mu}\Big) = 0 \\
&\Leftrightarrow \boldsymbol{\mu} = \int f(\boldsymbol{x}) \boldsymbol{x} dx,
\end{aligned}$$

where κ and κ' are some constants. Therefore $\mu^* = \int f(x) x dx$, since Σ is invertible. Now take the derivative of objective function $H(f,g)$ w.r.t. Σ we get,

$$\begin{aligned}
&-\frac{\partial}{\partial \Sigma} \int f(x) \left(\log \left(\frac{2\pi^{-d/2}}{\det(\Sigma)^{1/2}} \right) - \frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu) \right) dx \\
&= c \int f(x) \frac{\partial}{\partial \Sigma} \left(\log \det(\Sigma) + (x-\mu)^T \Sigma^{-1}(x-\mu) \right) dx \\
&\overset{\text{(iii \& iv)}}{=} c \int f(x) \left(\Sigma^{-T} - \Sigma^{-T}(x-\mu)(x-\mu)^T \Sigma^{-T} \right) dx, \quad \because X \succ 0.
\end{aligned}$$

Set the derivative to zero we get,

$$\begin{aligned}
\Sigma^{-T} &= \int \Sigma^{-T}(x-\mu)^T (x-\mu) \Sigma^{-T} f(x) dx \\
&= \Sigma^{-T} \int (x-\mu)^T (x-\mu) f(x) dx \Sigma^{-T}.
\end{aligned} \tag{7.34}$$

Then,

$$\Sigma = \int (x-\mu)^T (x-\mu) f(x) dx, \quad \because \Sigma = \Sigma^T. \tag{7.35}$$

□

To perform the EM algorithm in Alg. 11, we need each module in a

Algorithm 11 EM algorithm minimizing cross entropy.

E-step: Let $\Theta = \{w_j, \mu_j, \Sigma_j\}_{j=1}^{K}$, $\bar{\mathcal{F}}(x) = \sum_{i=1}^{NK} \pi_i f_i(x)$ and X_i be a set of points with density function $f_i(x)$. Then we have,

$$\gamma_{ij} := p(z_i = j | X_i, \Theta) = \frac{w_j \exp\left[-H(f_i, g_j)\right]}{\sum_{j'}^{K} w_{j'} \exp[-H(f_i, g_{j'})]} \tag{7.36}$$

Note that γ_{ij} is the likelihood that the i^{th} component of $\bar{\mathcal{F}}$ corresponds to j^{th} in $\mathcal{G}$ $H(f_i, g_j)$ is analytically obtained as,

$$\frac{1}{2}\{k \log 2\pi + \log|\Sigma_j| + \mathrm{tr}[\Sigma_j^{-1}\Sigma_i] + (\mu_i - \mu_j)^T \Sigma_j^{-1}(\mu_i - \mu_j)\}$$

M-step:

$$w_j = \frac{\sum_{i=1}^{NK} \pi_i \gamma_{ij}}{\sum_{j'=1}^{K} \sum_{i'=1}^{NK} \pi_{i'} \gamma_{i'j'}} \tag{7.37}$$

$$\mu_j = \mathbb{E}_{\bar{\mathcal{F}}'(x)}[x] = \sum_{i=1}^{NK} \pi_i' \mu_i \tag{7.38}$$

$$\Sigma_j = \mathbb{E}_{\bar{\mathcal{F}}'(x)}[(x - \mu_j)(x - \mu_j)^T] \tag{7.39}$$

$$= \sum_{i=1}^{NK} \pi_i' \Sigma_i + \sum_{i=1}^{NK} \pi_i' (\mu_i - \mu_j)(\mu_i - \mu_j)^T \tag{7.40}$$

where $\bar{\mathcal{F}}' = \sum_{i=1}^{NK} \pi_i' f_i(x)$, and $\pi_i' = \frac{\pi_i \gamma_{ij}}{\sum_i \pi_i \gamma_{ij}}$, for fixed j.

closed form. We first introduce how to calculate the cross entropy between two Gaussian distributions used in (7.36) for the E-step. And then we derive the closed forms for the mean and variance of a GMM used in (7.38) and (7.39) for the M-step.

7.5.1 Cross entropy between two Gaussians

The cross entropy used in (7.36) is the optimal code length given data p and codebook q, which is as

$$H(p,q) := \mathbb{E}_p[-\log q(x)] = -\int p(x)\log q(x)dx. \tag{7.41}$$

The cross entropy between two Gaussian distributions has an analytic form. Let $\mathcal{N}_p$ and $\mathcal{N}_q$ be multivariate Gaussian distributions with (μ_p, Σ_p) and (μ_q, Σ_q) respectively. The cross entropy $H(p,q)$ is given as

$$\begin{aligned}\mathbb{E}_p[-\log q(x)] = \frac{1}{2}\Big\{&k\log 2\pi + \log|\Sigma_q| + \mathrm{tr}[\Sigma_q^{-1}\Sigma_p] \\ &+(\mu_p-\mu_q)^T\Sigma_q^{-1}(\mu_p-\mu_q)\Big\}\end{aligned} \tag{7.42}$$

Proof. First, the Gaussian density function is given by

$$q(x) = \frac{1}{\sqrt{(2\pi)^k|\Sigma_q|}}\exp\left(-\frac{1}{2}(x-\mu_q)^T\Sigma_q^{-1}(x-\mu_q)\right) \tag{7.43}$$

Let us take the log of $q(x)$.

$$\log q(x) = -\frac{k}{2}\log 2\pi - \frac{1}{2}\log|\Sigma_q| - \frac{1}{2}(x-\mu_q)^T\Sigma_q^{-1}(x-\mu_q) \tag{7.44}$$

Then, the cross entropy $H(p,q)$ is

$$\begin{aligned}\int -p(x)\log q(x)dx = &\frac{k}{2}\log 2\pi + \frac{1}{2}\log|\Sigma_q| \\ &+\frac{1}{2}[\mu_q^T\Sigma_q^{-1}\mu_q - 2\mu_p^T\Sigma_q^{-1}\mu_q + \int x^T\Sigma_q^{-1}xp(x)dx]\end{aligned} \tag{7.45}$$

We know that $\mathrm{tr}(\cdot)$ and $\mathbb{E}[\cdot]$ are linear operators, so $\mathrm{tr}\circ E = E\circ\mathrm{tr}$. Using this fact, we have

$$\begin{aligned}\int x^T\Sigma_q^{-1}xp(x)dx &= \mathbb{E}_p[x^T\Sigma_q^{-1}x] = \mathbb{E}_p[\mathrm{tr}[x^T\Sigma_q^{-1}x]] \\ &= \mathbb{E}_p[\mathrm{tr}[\Sigma_q^{-1}xx^T]] = \mathrm{tr}[\mathbb{E}_p[\Sigma_q^{-1}xx^T]] \\ &= \mathrm{tr}[\Sigma_q^{-1}\mathbb{E}_p[xx^T]] = \mathrm{tr}[\Sigma_q^{-1}(\Sigma_p + \mu_p\mu_p^T)] \\ &= \mathrm{tr}[\Sigma_q^{-1}\Sigma_p] + \mu_p\Sigma_q^{-1}\mu_p^T\end{aligned} \tag{7.46}$$

Replacing $\int x^T\Sigma_q^{-1}xp(x)dx$ in (7.45) with $\mathrm{tr}[\Sigma_q^{-1}\Sigma_p] + \mu_p\Sigma_q^{-1}\mu_p^T$ completes the proof. □

7.5.2 Mean and covariance of samples from a GMM

Let $f'(x) = \sum_{i=1}^{L}\pi_i' f_i(x)$ be a GMM, namely, each $f_i(x;\mu_i,\Sigma_i)$ is a Gaussian distribution and $\sum_{i=1}^{L}\pi_i' = 1$. We provide the derivation of the mean and covariance of GMM $f'(x)$ used in (7.38) and (7.39).

The mean used in (7.38) is obtained by

$$\begin{aligned}\mathbb{E}_{f'(x)}[x] &= \int xf'(x)dx = \int_x x\sum_{i=1}^{L}\pi_i' f_i(x)dx \\ &= \sum_{i=1}^{L}\pi_i'\int_x xf_i(x)dx = \sum_{i=1}^{L}\pi_i'\mu_i =: \bar{\mu}\end{aligned} \tag{7.47}$$

Now, the covariance used in (7.39) is obtained by

$$
\begin{aligned}
&\mathbb{E}_{f'(x)}[(x-\bar{\mu})(x-\bar{\mu})^T] = \int (x-\bar{\mu})(x-\bar{\mu})^T f'(x)dx \\
&= \int_x (x-\bar{\mu})(x-\bar{\mu})^T \sum_{i=1}^{L} \pi_i' f_i(x)dx \\
&= \sum_{i=1}^{L} \pi_i' \int_x (x-\bar{\mu})(x-\bar{\mu})^T f_i(x)dx \\
&= \left(\sum_{i=1}^{L} \pi_i' \int_x xx^T f_i(x)dx - 2\pi_i' \int x\bar{\mu}^T f_i(x)dx \right) + \bar{\mu}\bar{\mu}^T \\
&= \left(\sum_{i=1}^{L} \pi_i' \int_x xx^T f_i(x)dx \right) - \bar{\mu}\bar{\mu}^T \\
&= \left(\sum_{i=1}^{L} \pi_i' [\Sigma_i + \mu_i \mu_i^T] \right) - \bar{\mu}\bar{\mu}^T \\
&= \sum_{i=1}^{L} \pi_i' \Sigma_i + \sum_{i=1}^{L} \pi_i' (\mu_i - \bar{\mu})(\mu_i - \bar{\mu})^T
\end{aligned}
\tag{7.48}
$$

Remarks: our proposed method can be interpreted as a functional clustering algorithm for a set of Gaussian distributions $\{f_i\}_{=1}^{L}$ whereas the classical GMM clusters a set of points $\{x_i\}_{=1}^{L}$. In our case, γ_{ij} represents the soft assignment of f_i in $\bar{\mathcal{F}}$ to g_j in $\mathcal{G}$. M-step can be interpreted as searching parameters for one representative Gaussian $g_j \in \mathcal{G}$ (roughly speaking, for each cluster a mean function (centroid) is restricted to a Gaussian) for a set of assigned Gaussians $\{f_i\}_{i=1}^{L} \in \bar{\mathcal{F}}$ with $\{\gamma_{ij}\}_{i=1}^{L}$.

7.5.3 Comparison with EM for the classical GMM

EM-algorithm w.r.t. cross entropy. Similar to the EM-algorithm for classical GMM, this proposed method comprises of two steps: E-step and M-step. Our result shows that M-step maximizes the *negative* cross entropy between reweighted data GMM $\sum_{i=1}^{L} \pi' f_i$ and a Gaussian component

in a model GMM g_j as the classical GMM increases the likelihood of reweighted samples. Let us compare each step of classical GMM and our proposed method.

E-step in classical GMM is given by

$$\gamma_{ij} := p(z_i = j|x_i, \theta) = \frac{p(z_i = j|\theta)g(x_i|z_i = j, \theta)}{\sum_{j'}^{K} p(z_i = j'|\theta)g(x_i|z_i = j', \theta)} \tag{7.49}$$

where i is the index for instance and j is the index for component in g (K-GMM).

E-step with cross entropy, we estimate the responsibilities between f_i and g_j rather than a point x_i and g_j in the classical GMM. Let X_i be a set of points which belong to set i with density function $f_i(x)$. j is defined as above. Then, the responsibilities γ_{ij} of f_i to g_j is given by

$$\begin{aligned}
\gamma_{ij} &:= p(z_i = j|X_i, \theta) \\
&= \frac{p(z_i = j|\theta)p(X_i|z_i = j, \theta)}{\sum_{j'=1}^{K} p(z_i = j'|\theta)p(X_i|z_i = j', \theta)} \\
&= \frac{p(z_i = j|\theta)\prod_{x\in X_i} g(x|z_i = j, \theta)^{f_i(x)}}{\sum_{j'}^{K} p(z_i = j'|\theta)\prod_{x\in X_i} p(x|z_i = j', \theta)^{f_i(x)}} \\
&= \frac{p(z_i = j|\theta)\exp\left[\sum_{x\in X_i} f_i(x)\log g(x|z_i = j, \theta)\right]}{\sum_{j'}^{K} p(z_i = j'|\theta)\exp\left[\sum_{x\in X_i} f_i(x)\log g(x|z_i = j', \theta)\right]} \\
&= \frac{p(z_i = j|\theta)\exp\left[\int_x f_i(x)\log g(x|z_i = j, \theta)dx\right]}{\sum_{j'}^{K} p(z_i = j'|\theta)\exp\left[\int_x f_i(x)\log g(x|z_i = j', \theta)dx\right]} \\
&= \frac{w_j\exp\left[\int_x f_i(x)\log g_j(x|\theta)dx\right]}{\sum_{j'}^{K} w_{j'}\exp\left[\int_x f_i(x)\log g_{j'}(x|\theta)dx\right]} \\
&= \frac{w_j\exp\left[-H(f_i, g_j)\right]}{\sum_{j'}^{K} w_{j'}\exp[-H(f_i, g_{j'})]}
\end{aligned} \tag{7.50}$$

Note that γ_{ij} denotes the membership of the i-th Gaussian distribution in l-GMM to the j-th component in k-GMM. $H(f_i, g_j)$ is analytically obtained as (7.42).

M-step in classical GMM is,

$$\begin{aligned} w_j &= \frac{1}{L}\sum_i \gamma_{ij} \\ \mu_j &= \frac{\sum_{i=1}^{L}\gamma_{ij}x_i}{\sum_{i'=1}^{L}\gamma_{i'j}} \\ \Sigma_j &= \frac{\sum_{i=1}^{L}\gamma_{ij}(x_i-\mu_j)(x_i-\mu_j)^T}{\sum_{i'=1}^{L}\gamma_{i'j}} \end{aligned} \tag{7.51}$$

M-step with cross entropy is,

$$\begin{aligned} w_j &= \sum_{i=1}^{L}\pi_i\gamma_{ij}, \text{since } \sum_{i=1}^{L}\pi_i = 1 \\ \mu_j &= \mathbb{E}_{f'}(x)[x] = \sum_{i=1}^{L}\pi_i'\mu_i \\ \Sigma_j &= \mathbb{E}_{f'}(x)[(x-\mu_j)(x-\mu_j)^T] \\ &= \sum_i \pi_i'\Sigma_i + \sum_i \pi_i'(\mu_i-\mu_j)(\mu_i-\mu_j)^T \end{aligned} \tag{7.52}$$

where $f' = \sum_{i=1}^{L}\pi_i' f_i(x)$, and $\pi_i' = \frac{\pi_i\gamma_{ij}}{\sum_i \pi_i\gamma_{ij}}$, for fixed j.

M step in classical GMM can be interpreted as expectations: x and $(x-\mu)(x-\mu)^T$ over a discrete probability distribution $\{\gamma_{ij}/\gamma_j\}_{i=1}^{L}$, where $\gamma_j := \sum_{i=1}^{L}\gamma_{ij}$. Similarly, M-step in our proposed method also can be interpreted as expectations over a reweighted GMM f', which is a continuous probability distribution.

7.5.4 Detailed derivation of EM algorithm

In this section, we provide full derivation of EM algorithm for our method. Let $f = \sum_i \pi_i f_i$ and $g = \sum_j \pi_j f_j$ be GMMs. Our EM algorithm maximizes the *negative* cross entropy $-H(f,g) := \int f(x) \log g(x)$. First, we derive the Q function from the objective function.

$$
\begin{aligned}
&\arg\max_{g \in \mathbf{G}(K)} \int \sum_{i=1}^{L} \pi_i f_i(x) \log \sum_{j=1}^{K} w_j g_j(x) dx \\
&= \arg\max_{g \in \mathbf{G}(K)} \int \sum_{i=1}^{L} \pi_i f_i(x) \log \sum_{j=1}^{K} P(z_i = j | X_i, \theta) \frac{w_j g_j(x)}{P(z_i = j | X_i, \theta)} dx \\
&\geq \arg\max_{g \in \mathbf{G}(K)} \int \sum_{i=1}^{L} \sum_{j=1}^{K} \pi_i f_i(x) P(z_i = j | X_i, \theta) \log \frac{w_j g_j(x)}{P(z_i = j | X_i, \theta)} dx \qquad (7.53) \\
&= \arg\max_{g \in \mathbf{G}(K)} \int \sum_{i=1}^{L} \sum_{j=1}^{K} \pi_i f_i(x) P(z_i = j | X_i, \theta) \log w_j g_j(x) dx \\
&\quad - \int \sum_{i=1}^{L} \sum_{j=1}^{K} \pi_i f_i(x) P(z_i = j | X_i, \theta) \log P(z_i = j | X_i, \theta) dx
\end{aligned}
$$

The inequality above is obtained by Jensen's inequality. Now, we define $Q(\theta|\theta_n)$ with the first term of the last equation as

$$
Q(\theta|\theta_n) := \int \sum_{i=1}^{L} \sum_{j=1}^{K} \pi_i f_i(x) P(z_i = j | X_i, \theta_n) \log w_j g_j(x) dx. \qquad (7.54)
$$

Once we define $Q(\theta|\theta_n)$, we are ready to derive EM algorithm. First, E step is merely to estimate $P(z_i = j|X_i, \theta) =: \gamma_{ij}$ by (7.50). Second, we derive M step. To do so, we will maximize $Q(\theta|\theta_n)$ w.r.t. $\{w_j, \boldsymbol{\mu}_j, \Sigma_j\}_{j=1}^{K}$.

The Q function can be rewritten as

$$Q(\theta|\theta_n) = \int \sum_{i=1}^{L}\sum_{j=1}^{K} \pi_i f_i(x)\gamma_{ij} \log g_j(x)dx + \sum_{i=1}^{L}\sum_{j=1}^{K} \pi_i\gamma_{ij}\log w_j. \quad (7.55)$$

To maximize over μ_j and Σ_j, one needs to maximize the following.

$$\underset{\{\boldsymbol{\mu}_j\}_{j=1}^K,\{\Sigma_j\}_{j=1}^K}{\operatorname{argmax}} Q(\theta|\theta_n) = \underset{\{\boldsymbol{\mu}_j\}_{j=1}^K,\{\Sigma_j\}_{j=1}^K}{\operatorname{argmax}} \int \sum_{i=1}^{L}\sum_{j=1}^{K} \pi_i\gamma_{ij} f_i(x)\log g_j(x)dx \quad (7.56)$$

Since, given γ_{ij}, one can decompose the maximization into for each component j, one has

$$\underset{\boldsymbol{\mu}_j,\Sigma_j}{\operatorname{argmax}} \int \sum_{i=1}^{L} \pi_i\gamma_{ij} f_i(x)\log g_j(x)dx = \underset{\boldsymbol{\mu}_j,\Sigma_j}{\operatorname{argmax}} \int \frac{\sum_{i=1}^{L}\pi_i\gamma_{ij}}{\sum_{i'=1}^{L}\pi_{i'}\gamma_{i'j}} f_i(x)\log g_j(x)dx \quad (7.57)$$

since dividing the objective function by a constant doesn't change the problem. Now, let $\pi_i^{'} := \frac{\pi_i\gamma_{ij}}{\sum_{i'=1}^{L}\pi_{i'}\gamma_{i'j}}$. Then the maximization over μ_j and Σ_j reduces to Lemma 7.2 that maximizes the *negative* entropy between a reweighted GMM $f'(x) := \sum_{i=1}^{L}\pi_i^{'} f_i(x)$ and a Gaussian with μ_j and Σ_j. The optimal solution μ_j^*, Σ_j^* is given in (7.52).

To maximize over w_j, one needs to maximize $Q(\theta|\theta_n)$ with a constraint $\sum_{j=1}^{K} w_j = 1$. So the objective function with a Lagrange multiplier is defined by $\mathcal{L} := Q(\theta|\theta_n) + \lambda(\sum_{j=1}^{K} w_j - 1)$. Now, take the derivative of $\mathcal{L}$ w.r.t w_j.

$$\frac{\partial\mathcal{L}}{\partial w_j} = \frac{\partial}{\partial w_j}\sum_{i=1}^{L}\pi_i\gamma_{ij}\log w_j + \lambda \quad (7.58)$$

Let's set the derivative above to zero. Then, one gets

$$w_j = -\frac{\sum_{i=1}^{L} \pi_i \gamma_{ij}}{\lambda}. \tag{7.59}$$

The primal feasibility $\sum_{j=1}^{K} w_j = 1$ and the result above yields $\lambda = -\sum_{j=1}^{K} \sum_{i=1}^{L} \pi_i \gamma_{ij}$. Hence the optimal weight is given by

$$w_j^* = \frac{\sum_{i=1}^{L} \pi_i \gamma_{ij}}{\sum_{j'=1}^{K} \sum_{i'=1}^{L} \pi_{i'} \gamma_{i'j'}}. \tag{7.60}$$

This completes the derivation of our EM algorithm. For more details on the theory of EM algorithms, we refer the reader to (Mak et al., 2004).

7.5.5 Functional clustering and construction of modified EM for EAPs

In the case of EAPs (more details in section 7.6), the GMMs have a special property that all μ_js are zero. It is easy to verify that if the input EAPs are comprised of zero mean Gaussians, the algorithm in Fig. 11 does yield a valid EAP (k-GMM with zero means). However, our goal is not to merely obtain 'valid' EAPs but to minimize the potential change in anisotropy (of the EAPs) using our interpolation. EAPs (with zero mean Gaussians) imply that their components overlap significantly at their modes. We found that their differences are much less accurately captured by cross-entropy. In practice, this may lead the algorithm towards inaccurately big ellipsoids since it averages different Gaussians (in the EAPs) with relatively similar responsibility γ_{ij}.

This problem is directly addressed by our modified EM algorithm in Fig 12. First, we use the ℓ_2 distance for the E-step to capture the differences. In addition, by introducing the simplest covariance function C_j for each component, we allow each component to have different densities in the

Algorithm 12 Modified EM for operations on EAPs.

E-step: Estimate the responsibilities of data PDFs to components of our model,

$$\gamma_{ij} = \frac{w_j C_j^{-1} \exp\left(-\frac{1}{2C_j^2}\|f_i - g_j\|_2^2\right)}{\sum_{k=1}^{K} w_k C_k^{-1} \exp\left(-\frac{1}{2C_k^2}\|f_i - g_k\|_2^2\right)} \tag{7.61}$$

M-step: Maximize cross entropy given assignments over model parameters (a weight w_j, mean function $\mathcal{N}(\mu_j, \Sigma_j)$ and a covariance function C_j).

$$C_j^2 = \sum_{i=1}^{NK} \gamma_{ij}\pi_i \|f_i - g_j\|_2^2 / \sum_{i'=1}^{NK} \gamma_{i'j}\pi_{i'} \tag{7.62}$$

w_j and μ_j, Σ_j are updated using Eqs. (7.37)-(7.39).

functional space. In other words, even though some Gaussians within an EAP may overlap substantially, if C_j is small enough, our algorithm is still able to distinguish them nicely and assign significantly different responsibility. This makes our approach very robust.

This modified algorithm, which estimates *four* parameters for each component $(w_j, \mu_j, \Sigma_j, C_j)$, drives the EAP experiments presented in this chapter. Note that as a by-product of EM algorithms, all our EM-algorithms described in Alg. 11 and 12 cluster the weighted component Gaussian distributions f_i of $\bar{\mathcal{F}}$ in the functional space.

7.6 Experiments

In this section, we introduce the diffusion PDF of interest (EAP) and demonstrate the results of various operations such as upsampling resolution, denoising, spatial transformations on the EAP field where the basic underlying module is interpolation. We also show experiments showing that interpolation on the K-GMM manifold provides benefits in terms

of controlling the number of components when one needs to perform repeated interpolations. Controlling the number of components has a direct impact on our ability to resolve the peaks in the EAP profiles which is crucial in generating tractography, a key component in deriving brain connectivity information from such imaging data (Bastiani et al., 2012).

Ensemble average propagator (EAP). White matter architecture can be probed by analyzing thermal diffusivity profiles of water molecules in the brain. Thermal diffusion of water causes signal decay in the measured MR signal. The decay, under certain assumptions of the MR pulse sequencing used to acquire the signal satisfies the following relationship

$$E(q\mathbf{u}) = \int_{\mathbb{R}^3} P(R\mathbf{r})\exp(2\pi i q R\mathbf{u}^T\mathbf{r})\mathrm{d}R\mathbf{r}, \tag{7.63}$$

where $\mathbf{u}, \mathbf{r}$ are unit vectors in $\mathbb{R}^3$, q is proportional to the amplitude of the magnetic field gradient along $\mathbf{u}$ and $P(R\mathbf{r})$ is called the ensemble average propagator (EAP) describing the probability of diffusion displacements of water molecules at radius R (Stejskal and Tanner, 1965; Callaghan, 1991). Assuming antipodal (radial) symmetries for the signal decay (i.e., $E(q\mathbf{u}) = E(-q\mathbf{u})$) and EAP ($P(R\mathbf{r}) = P(-R\mathbf{r})$), the following relationship holds (Cheng et al., 2010)

$$P(R\mathbf{r}) = \int_{\mathbb{R}^3} E(q\mathbf{u})\cos(2\pi q R\mathbf{u}^T\mathbf{r})\mathrm{d}q\mathbf{u}. \tag{7.64}$$

The EAP is a PDF whose domain is $\mathbb{R}^3$. In our experiments, we use a K-GMM representation of the EAP (Jian and Vemuri, 2007b). However, we would like to note that other tensor distribution models may be used to tackle fiber crossings without the use of finite mixture of Gaussians (Jian et al., 2007; Jian and Vemuri, 2007a,b).

Upsampling and denoising. Signal to noise ratio (SNR) of the MR signal is proportional to the volume size of a voxel. Diffusion weighted MRI

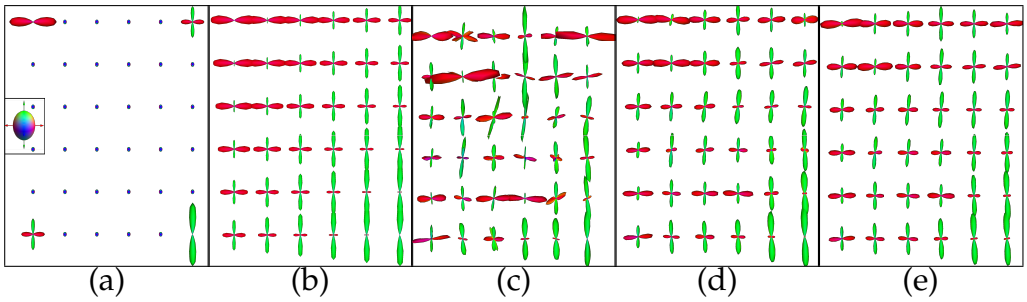


Figure 7.3: (a) Input data with just four voxels in the foreground, (a)-(e) are all at the same scale. The color mapping scheme used to visualize the profile at each voxel is shown in the box overlaid on the background voxels which are set to have isotropic diffusivity. (b) Result of upsampling with bi-linear interpolation. (c) Noisy EAPs. (d) Gaussian filtering. (e) Anisotropic filtering.

faces challenges in terms of achieving high SNR due to rapid acquisitions and hence the voxel resolution acquired on typical scanners is usually 8 mm^3. For applications like tractography, recent investigations recommend a resolution of 1.95313 mm^3(Setsompop et al., 2013). But acquiring such a scan requires drastic improvements to the scanner gradient capabilities and adds significant scanning time (~55 mins. vs. ~10 mins.) (Setsompop et al., 2013). Hence providing an upsampling algorithm and a denoising modules that can reconstruct the EAPs respecting its native geometry can be practically very useful. We simulate EAP profiles at $R = 15\mu$m in voxels at the four corners of a 6×6 grid as shown in Fig. 7.3(a) and fill in such severely undersampled data in the remainder of the grid with our algorithm. We perform a simple bi-linear interpolation to fill in the grid as shown in Fig. 7.3(b) using the operations introduced in Section 7.3. We can observe that the diffusion PDFs are smoothly interpolated respecting the geometry of the crossing fibers. To demonstrate the denoising capabilities of our algorithm we add Wishart noise to the EAPs (Fig. 7.3(c)). The denoised EAPs using Gaussian filtering and anisotropic filtering are shown in Figs. 7.3(d) and (e).

Since EAP profiles are affected by the architecture of the white mat-

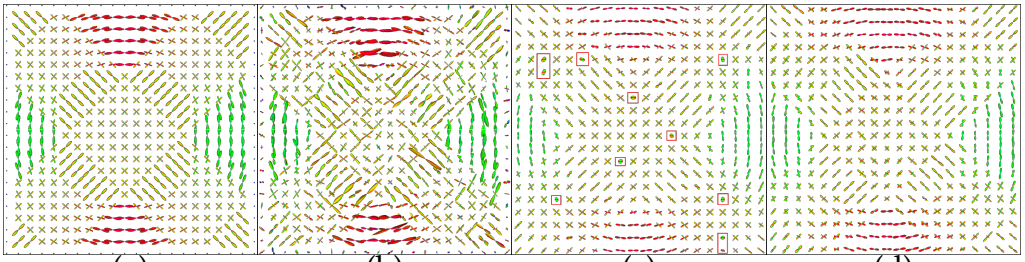


Figure 7.4: (a) Simulated EAP profiles. (b) EAP profiles with added Wishart noise. (c) Gaussian filtering. (d) Anisotropic filtering.

ter pathways we additionally simulate EAP data to reflect crossing and curving pathways (Cheng et al., 2010) and demonstrate Gaussian and anisotropic filtering as shown in Fig. 7.4(a). We can observe that the anisotropic filtering does near perfect recovery of the underlying signal. The red boxes in (c) highlight the differences between Gaussian and anisotropic filtering.

Spatial transformations. One of the key steps in statistical analysis of neuroimaging data is to spatially normalize the images from different subjects i.e., transform/warp each of the individual subject's image data onto a group-level standard grid. Although spatial transformations of diffusion tensor images (single component GMMs) is widely studied and used in clinical studies (Zhang et al., 2006b), currently there are no widely available tools for advanced diffusion PDFs such as EAPs. Note that there *are* Riemmannian interpolation schemes available (Goh et al., 2009; Cheng et al., 2009, 2011) in the literature but not specifically for K-GMMs. Using our algorithm, we rotate two EAP fields by 30° and also apply affine transformations. The results are shown in Fig. 7.5. When performing non-orthonormal transformations on the EAP fields, one needs to extract the rotation transformation to reorient the profiles. To do so, we use the finite strain method (Alexander et al., 2001). We observe that even in cases of really complex architecture our interpolation and reorientation preserve

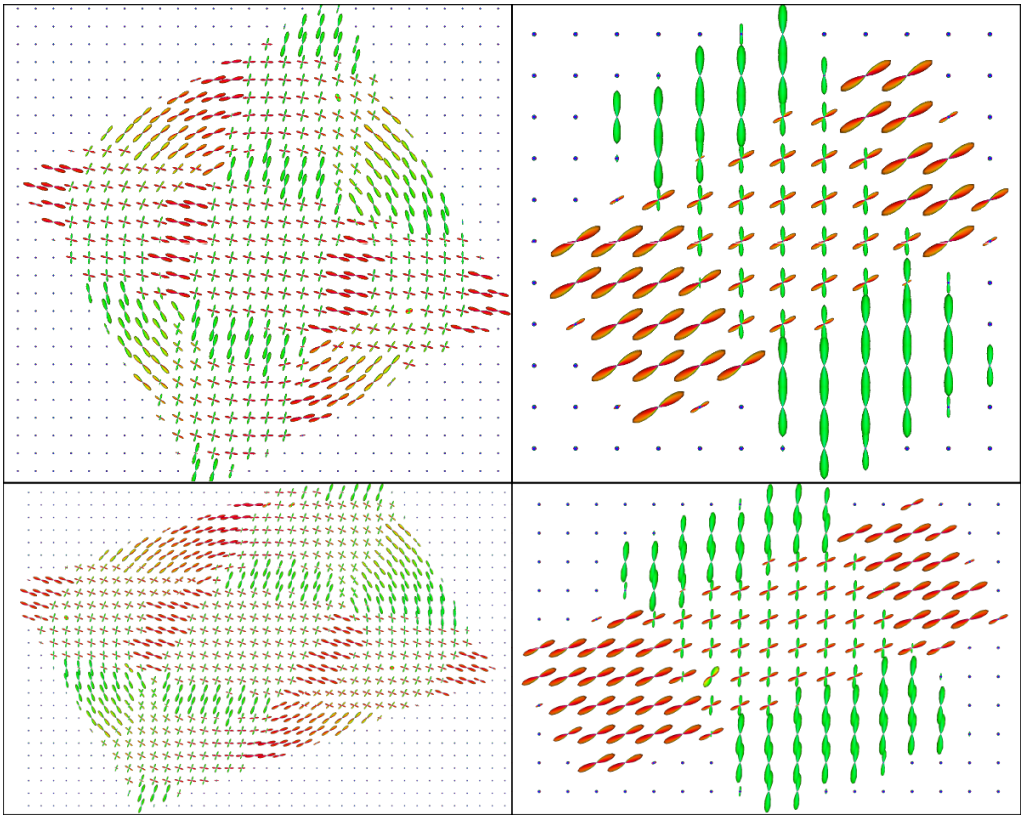


Figure 7.5: **Top row:** Rotated EAP profiles. **Bottom row:** Results of affine transformation of the EAP fields.

the organizational features (crossing and circular nature of the profiles) of the profiles. The shearing effects where the crossing fiber region stretches increasing the number of crossing fibers and the circular organization becomes elliptical.

Peak preserving complexity reduction. In this experiment, we demonstrate that model complexity can interfere with simple peak finding algorithms and hence it is advantageous to operate on a fixed K-GMM manifold. The error in peak detection is computed as follows,

Let K^* be the true number of peaks in the simulated EAP field. Then, the error at each voxel in an estimated/interpolated EAP field is measured

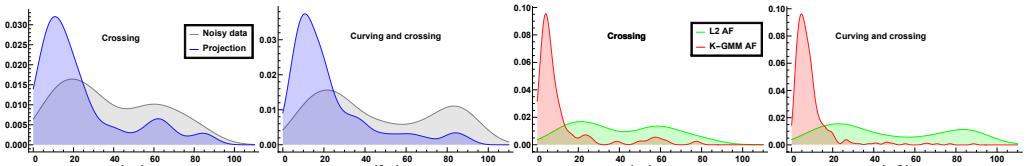


Figure 7.6: The distributions of angular deviations of the peaks. Comparing projected and noisy data in (a) crossing fiber phantom, and (b) curving and crossing phantom. Comparing anisotropic filtering with K-GMM (ours) and ℓ_2 interpolation in (c) crossing fiber phantom, and (d) curving and crossing phantom.

by

$$\epsilon = \min_{\Pi} \sum_{i=1}^{K^*} \cos^{-1} |V_i^T U_{\Pi(i)}|, \tag{7.65}$$

where V_i and U_i are eigen vectors of the K^* largest weight components of ground truth and estimated EAP, respectively. $\Pi(i)$ is the best permutation which has the minimum error, i.e., when $K^* = 2$, ϵ is the minimum of angular errors between $\{V_1, V_2\}$ and $\{U_1, U_2\}$ with all possible permutations. Hence the range of ϵ in each voxel is $[0, K^* \times 90^\circ]$. We first add Wishart noise to the numerically simulated crossing and curving EAP profiles (see Fig. 7.5). Figs. 7.6(a) and (b) show the deviations of the peaks detected by projecting (*without any filtering*) from $\mathbf{G}^{(10)}$ to $\mathbf{G}^{(2)}$ the noisy data for crossing and curving phantoms respectively. The distribution corresponds to errors in all voxels in an image. As we can see, the errors are reduced just by reducing the number of components. Figs. 7.6(c) and (d) show the distributions of the angular deviations for crossing and curving after anisotropic filtering with K-GMM and ℓ_2 method. We can observe that the K-GMM method significantly outperforms the ℓ_2 method. The K-GMM method deviates on average about 10° while the errors with ℓ_2 are spread further especially in crossing fiber regions (i.e. $\epsilon > 90^\circ$).

7.7 Summary

This chapter describes a numerically robust scheme for performing interpolation on the manifold of K component GMMs, where few solutions are available in the literature today. Such operations are needed to perform theoretically sound processing of a field of EAPs, fundamental objects in diffusion weighted magnetic resonance imaging. We first derive a gradient descent scheme and then use those ideas towards an efficient and numerically stable EM style method. The algorithm is general and applicable to other situations where interpolation is needed for objects such as functions, probability distributions and so on (though for some special cases, more specialized algorithms are known). Separately, notice that operating directly on the functional space of Gaussians (and their mixtures) suggested insights that were useful in obtaining our numerical procedures. Some of these issues are briefly mentioned in passing in the chapter (see last paragraphs of Section 7.2 and Section 7.5). We believe that with the growing interest in using advanced image analysis and statistical techniques for analyzing and making sense of rich datasets being collected worldwide (e.g., the Human Connectome project), algorithms such as the one proposed here will be valuable in ensuring that the underlying processing remains faithful to the geometry/structure of the data. Doing so will not only improve the statistical analysis but put us in the best position to extract scientifically interesting hypotheses from such images. Code is available online [1].

[1] https://github.com/MLman/kgmm_interpolation

8 DISCUSSION AND FUTURE DIRECTIONS

This chapter summarizes the main themes and contributions of this thesis and discusses future directions.

8.1 Main ideas and contributions

The main motivation of this thesis is to develop statistical machine learning algorithms for structured data motivated by applications in computer vision and neuroimaging. The proposed methods expand the operating range of Euclidean multivariate statistics to more general nonlinear spaces such as Riemannian manifolds and a structured functional space. Specifically, when the data space is known a priori, we studied how to exploit the geometry of data space. Towards new learning models that respect the geometry of data space and enable more statistically powerful and accurate inference. The main contributions of this thesis are the following:

- We developed Manifold-valued Multivariate General Linear Models (MMGLMs) for a structured response variable and multiple covariates, i.e., $f : \mathbf{R}^n \rightarrow \mathcal{M}$ with efficient estimation schemes (Kim et al., 2014b). The companion open-source code is available for large scale analysis on Amazon Web Service[1] and HTCondor[2].

- We proposed a principled generalization of CCA to the Riemannian setting that handles multi-modal images with structured measurements and offers feature selection based on correlation (Kim et al., 2014a, 2016b).

- For more flexible regression models for manifold-valued data, we extended MMGLMs to the mixtures of MMGLMs on manifolds using

[1] https://github.com/MLman/MMGLMAWS
[2] https://github.com/MLman/MMGLM_HTCONDOR

a nonparametric Bayesian approach. We studied a new distribution to sample manifold-valued parameters and extended the Hamiltonian Monte Carlo sampling method for manifold-valued parameters. The model captures more complex patterns than MMGLMs and offers nonparametric clustering based on the relationship between covariates and response variables (Kim et al., 2015b).

- We studied nonlinear mixed effects models that handles manifold-valued response variables and analyzed the trajectories of local morphometric changes of brains longitudinally. The model captures subject-specific random effects as well as population-level trends. It offers interpretability of learned models (Kim et al., 2017b).

- We studied how to interpolate Gaussian mixture models, which are structured probability density functions, restricting the complexity of resulting interpolants. This framework allows more robust interpolation and compression of GMMs with a far fewer number of Gaussian components (Kim et al., 2015a).

8.2 Future Directions

Structured data analysis has been shown to be effective and a variety of theoretical results are emerging under umbrella topics such as Object-oriented data analysis (OODA) (Marron and Alonso, 2014), *manifold statistics* and so on. But the impact of these works still remains somewhat limited due to three main reasons: 1) small-scale experiments, which are not convincing to practitioners, 2) this line of work focuses on exploring new models and their implementations are limited for a full suite of downstream analysis, and 3) the lack of standard materials (textbooks) often makes this field less accessible to newcomers.

We believe that it is crucial to develop computationally scalable, efficient and easily deployable frameworks. As a part of this effort, open source projects such as MANOPT (Boumal et al., 2014) are an excellent jumping off point. But it is focused on optimization rather than learning models and it may not be efficient or convenient enough for data scientists to apply directly to large-scale real world data. On our end, we provided software for some of the proposed methods in this thesis but they may need to be improved for large-scale data in some applications.

Beyond efficient estimation of the models or scalable implementation, additional statistical downstream analysis such as p-value calculations, group difference, confidence interval, null distributions are still rudimentary and rely on more computationally expensive methods such as *permutation tests*. Theoretical development focused on these aspects is important to understand the models and computational gain from such developments is likely to be significant. Developing a full suite of machine learning methods and efficient inference frameworks are sorely needed to maximize the potential impact of this line of work. Motivated by the needs of real-world data analysis problems, we plan to devote effort towards both the theory and computational development of this topic, so that these tools are effective at addressing the types of problems many vision and machine learning researchers want to solve.

Further, one of the biggest barriers to entry is that no widely used standard material to start studying manifold frameworks. Unlike other topics, these concepts are not covered in a standard data science course. We believe that an online resource/compendium of differential geometry for machine learning and computer vision aimed at entry-level graduate students/undergrads including calculus, and numerical optimization with constraints will be extremely useful and improve the accessibility of manifold frameworks. One common issue for non-experts is that identifying applications which benefit from manifold frameworks is not always

easy. While the scope of such frameworks has been growing, it is slow except some specific fields. We believe that the introductory materials with examples online will help readers understand the framework and apply them to their research properly.

Manifold frameworks have advanced the understanding of structured data and data space. Further, it plays a crucial role to generalize statistical models for multiple different types of structured data. The methods in this thesis are also general and applicable to a wide range of manifold-valued data directly or with minor changes. One natural question is what other applications may benefit from our proposed methods? We discuss some open problems directly related to the proposed models.

- Manifold-valued multivariate general linear models in Chapter 3 are applied for structured measurements in brain images. But the model can be used for any structured data as long as they lie on Riemannian manifolds. A recent project in collaboration with Ronak Mehta in progress studies time-varying graphs, which can be on SPD manifolds, with MMGLMs to identify a subgraph that shows significant group difference in trends or abnormal trends. Also, this can be used for shape analysis as a form of geodesic regression for image time-series (Niethammer et al., 2011).

- Mixed effects models in Chapter 6 were used to enable the longitudinal analysis controlling for subject-specific random effects, since samples from a particular subject may have its own bias. The same idea/framework can be used for data integration. Brain images from multiple sites often have their own site-specific random effects (Zhou et al., 2017). A collaborative effort of international Alzheimer disease centers will benefit from mixed effects models in analyzing neuroimaging data from with a much larger sample size controlling for heterogeneity of data.

- Riemannian Canonical Correlation Analysis in Chapter 4 can be extended to feature selection while controlling for covariates/nuisance variables, namely partial canonical correlation analysis. This is crucial to remove trivially correlated regions by factors such as age, and gender for brain changes. Further, feature selection in a product space, where we handle images with structured measurement at each voxel, should be studied with more general sparsity regularizations (e.g., group lasso) so that the feature selection algorithms can find sparsity patterns at a voxel-level rather than at a dimension-level of the product space.

- Cauchy deformation tensors are not limited to brain image analysis. It can be derived from any registration algorithms with natural images, shape analysis, medical imaging and so on. One concrete example is optical flow (Nagaraj et al., 2014). We can derive CDTs or Jacobian matrices from the optical flow (vector field) between two consecutive frames and may be able to find the intrinsic lower dimensional spaces for more robust optical flow estimation and segmentation using intrinsic metrics.

- We briefly discussed the unbiased estimation of brain structural change and parallel transport of the morphometric changes to a template space. If we start from registered images in a template space, then a large portion of subject-specific structured change will be lost. Estimating the structural changes within each subject and bring the trajectories in a common space is preferable. In this problem, modeling the individual trajectories and transporting them to a common space are addressed by recent works (Lorenzi et al., 2011b; Lorenzi and Pennec, 2014) but still various open problems remain.

- We plan to evaluate the proposed methods on another dataset (or

diseases) such as Human Connection Projects (HCP), Alzheimer's Disease Neuroimaging Initiative (ADNI), and Dominantly Inherited Alzheimer Network (DIAN). Further, we plan to apply the methods to other diseases. This will ensure that the reproducibility of the models with limited samples.

We end this discussion with another interesting line of open problems. Most manifold frameworks assume that the geometry of data spaces is known a priori. However, even in known structured data spaces, data may form a submanifold since data in the Euclidean space can be viewed as points from a lower dimensional space embedded in the ambient space. This topic is studied by relatively simple models (Fletcher et al., 2004; Zhang and Fletcher, 2013; Sommer, 2013; Damon and Marron, 2014; Harandi et al., 2017). Estimations of more general submanifolds needs to be studied. Further, we know that a Riemannian manifold has three layers of structure: topological space, differentiable space, and Riemannian metric. Most works including the proposed methods in this thesis use the well-studied Riemannian metric for a particular smooth manifold. For example, when we have unit vectors then we often simply use a unit sphere with a canonical metric, which is induced by the metric in the ambient space. But on a smooth manifold, there may be multiple Riemannian metrics and it will be ideal if this were optimally chosen by the data. A more challenging problem may be structured data analysis in general and time-varying spaces beyond manifolds: such spaces may not have differential structures or well-defined metrics: examples include trees, graphs, deformable shape spaces, and permutations. These objects are useful to model pathway connectivity (biology), social networks (social science), reference networks (library & information), annotations with hierarchy, permutation space for ranking (recommender system, multi-label classification), and so on. Also, it is relevant to my work (Kim et al., 2016c) (but not included in this thesis) for graph structure estimation.

Often, these structured data (including graphs) lie in time-varying spaces. Analyzing the structured data in time-varying spaces is useful for many different fields dealing with longitudinal data. For example, landmark-based shape analysis on shape manifolds assumes that the number of landmarks is consistent over time (Kendall, 1984). But this is rarely true in practice. Longitudinal analysis of time-varying networks is an exciting area (Ahmed and Xing, 2009; Zhou et al., 2010; Qiu et al., 2016) with broad cutting applications that are related to the ideas described here. Towards a deeper understanding of time-varying data spaces, we studied temporal graphical models where the graph structure varies over time.

Lastly, multi-resolution analysis and deep learning for structured data are important directions but less explored. Wavelets have been studied on Riemannian manifolds by (Dahlke, 1994) and specifically harmonic analysis on symmetric space such as $SO(n)$ and S^n have been studied by (Antoine and Vandergheynst, 1998; Terras, 2012). The harmonic basis on Riemannian manifolds allow generalizing tools (or objects) to Riemannian manifolds, for example, distributions (exponential family) on $SO(n)$ and S^n (Cohen and Welling, 2015) and deep neural networks (Shaham et al., 2016) for provable function approximation using deep neural networks with wavelet for data on Riemannian manifolds. We believe that many theories from manifold statistics including the proposed methods in this thesis will be useful to make building blocks of deep neural networks for structured data. For example, the output layers for manifold-valued measurements can benefit from MMGLMs in Chapter 3 and deep learning for correlation analysis, the so-called Deep CCA in (Andrew et al., 2013), can be generalized for manifold-valued data using the idea of RCCA in Chapter 4. In this direction, there are recent attempts to incorporate in deep learning the geometric priors (or geometry of data space) including a distance metric, and invariance (or equivariance) w.r.t. scale, rotation, translation and local deformation, e.g., Group Equivariant Convolutional

Network (Cohen and Welling, 2016) and Invariant Scattering Convolution Networks (Bruna and Mallat, 2013). These emerging techniques to generalized deep networks for non-Euclidean and structured data are called geometric deep learning models (Bronstein et al., 2016). Recently, neural networks and deep learning have been studied on graphs (Bruna et al., 2013), Riemannian manifolds (Shaham et al., 2016) specifically Deep learning on SPD manifolds (Huang and Van Gool, 2017) and Grassmannian manifolds (Huang et al., 2016). Also Geodesic CNN (Masci et al., 2015) and Anisotropic CNN (Boscaini et al., 2016) have been proposed for mesh or point cloud. Though Riemannian manifold frameworks have been studied in the community for decades, these ideas are *not* well studied in terms of how to identify the best geometry of given data. Also, the framework is not suitable for data on mathematically not well-defined spaces. We believe that its marriage with deep neural networks allows handling more complex data spaces such as time-varying spaces and pseudo-Riemannian manifolds.

A APPENDIX

A.1 Distributions for manifold-valued variables

A.1.1 Prior distributions for SPD matrix

Wishart distribution over $n \times n$ SPD X with V a (fixed) positive definite matrix and df degrees of freedom.

$$\begin{aligned} f(X|V,df) &= \frac{1}{2^{\frac{n\times df}{2}}|V|^{\frac{df}{2}}\Gamma_n(\frac{df}{2})}|X|^{\frac{df-n-1}{2}}\exp\left(-\frac{1}{2}\mathrm{tr}(V^{-1}X)\right) \\ \log f(X|V,df) &= -\log Z(V,df) + \frac{df-n-1}{2}\log\det(X) - \frac{1}{2}\mathrm{tr}(V^{-1}X) \\ \frac{\partial}{\partial X}\log f(X|V,df) &= \frac{df-n-1}{2}X^{-1} - \frac{1}{2}V^{-1} \end{aligned} \tag{A.1}$$

Since X is a symmetric positive definite matrix, we have $\frac{\partial}{\partial X}\log\det(X) = X^{-1}$, see A.4.1 in (Boyd and Vandenberghe, 2004). Also we know that $\frac{\partial}{\partial X}\mathrm{tr}(AX^T) = A$ (Petersen and Pedersen, 2012a). So we can ensure that the derivative is a symmetric matrix.

Generalized normal distribution over $n \times n$ SPD X with (fixed) mean positive definite matrix M and $\sigma \in \mathbf{R}$.

$$\begin{aligned} f(X|M,\sigma) &= \frac{1}{Z(M,\sigma)}\exp\left(-\frac{1}{2\sigma^2}\mathrm{d}(X,M)^2\right) \\ \log f(X|M,\sigma) &= -\log Z(\sigma) - \frac{1}{2\sigma^2}\mathrm{d}(X,M)^2 \\ \nabla_X \log f(X|V,n) &= \frac{\mathrm{Log}(X,M)}{\sigma^2} \end{aligned} \tag{A.2}$$

The second equality holds since Z is constant w.r.t M on SPD manifolds. Note that this derivative is in $T_p\mathcal{M}$.

Log normal distribution 1. (definition 3.3.3 in (Schwartzman, 2006)) over symmetric positive definite matrix X with parameters $M \in \text{sym}^+(n)$ and covariance $\Sigma \in Sym^+(q)$, where $q = \frac{n(n+1)}{2}$. In other words, $Y = \text{Log}(M, X)$ has symmetric matrix variate normal distribution.

First, we define a $n(n+1)/2 \times 1$ column vector $vecd(X)$ as the concatenation of the on-diagonal and off-diagonal elements of X, i.e.

$$\text{vecd}(X) = \begin{pmatrix} \text{diag}(X) \\ \text{offdiag}(X) \end{pmatrix} \tag{A.3}$$

Then, the log normal distribution is defined by

$$\begin{aligned} vecd(Y) = vecd(\text{Log}(M, X)) &\sim N(0, \Sigma_{q\times q}) \\ f(X; M, \sigma^2 = 1) = \frac{J(G^{-1}XG^{-T})}{(2\pi)^{q/2}|GG^T|^{-(n+1)/2}} &\exp\left(-\frac{1}{2}\text{tr}(\log(G^{-1}XG^{-T}))^2\right) \end{aligned} \tag{A.4}$$

where $Y = \text{Log}(M, X)$, $M = GG^T$, and $J(\cdot) = \mathcal{J}(Y \to X) = |\partial Y/\partial X|$ is Jacobian of the log transformation $Y = \log X$. Let $\lambda_1 > \ldots > \lambda_p$ be the eigenvalues of X. S Then the Jacobina of the transformation $Y = \log X$ is equal to $J(X) = \mathcal{J}(Y \to X) = \frac{1}{\lambda_1 \ldots \lambda_p} \prod_{i<j} \frac{\log \lambda_j - \log \lambda_i}{\lambda_j - \lambda_i}$. However, the Riemannian log-normal distribution is not symmetric. In general, $f(X; Y, \sigma^2) \neq f(Y; X, \sigma^2)$.

Log normal distribution 2. (definition 3.3.4 in (Schwartzman, 2006)) We say that $X \in Sym^+(p)$ has a positive definite matrix variate Riemannian log normal distribution with parameter $M \in \text{Sym}^+(n)$, if the Riemannian logarithm map $Y = \text{Log}(M, X) \in Sym(n)$ has a symmetric matrix variate

normal distribution. Specifically,

$$Y = \mathrm{Log}(M, X) \sim N_{sym}(0, I) \tag{A.5}$$

Two log normal distribution on SPD manifolds are easy to sample but calculating the gradient has difficulty to deal with $J(\cdot)$ which depends on the sample X.

A.1.2 Prior distributions for symmetric matrix

Normal distribution 1. (definition 3.1.2 in (Schwartzman, 2006)) over $X \in Sym(n)$ with mean matrix 0 and covariance matrix I with respect to Lebesque measure on $\mathbf{R}^q$ is given by

$$f(X) = \frac{1}{(2\pi)^{q/2}} \exp(-\frac{1}{2}\mathrm{tr}(X^2)) \tag{A.6}$$

where $q = n(n+1)/2$. This is equivalent to multivariate normal distribution with the appropriate reshaping function. For example, for $p = 3$, Z is constructed as

$$Z = \begin{pmatrix} N(0,1) & N(0,1/2) & N(0,1/2) \\ * & N(0,1) & N(0,1/2) \\ * & * & N(0,1) \end{pmatrix} \tag{A.7}$$

Normal distribution 2. (definition 3.1.3 in (Schwartzman, 2006)) over $X \in Sym(p)$ with mean matrix M and covariance matrix Σ

$$
\begin{aligned}
f(X; M, \Sigma) &= \frac{1}{(2\pi)^{q/2}|\Sigma|^{(p+1)/2}} \exp\left(-\frac{1}{2}\mathrm{tr}((X-M)\Sigma^{-1})^2\right) \\
\log f(X|M,\Sigma) &= -\log Z(\Sigma) - \frac{1}{2}\mathrm{tr}[((X-M)\Sigma^{-1})^2] \\
\frac{\partial}{\partial X}\log f(X|M,\Sigma) &= -\frac{1}{2}\frac{\partial}{\partial X}\mathrm{tr}\left[(X-M)\Sigma^{-1}(X-M)\Sigma^{-1}\right] \\
&= -\frac{1}{2}\frac{\partial}{\partial X}\left[\mathrm{tr}(X\Sigma^{-1}X\Sigma^{-1}) - 2\mathrm{tr}(\Sigma^{-1}M\Sigma^{-1}X) + \mathrm{tr}(M\Sigma^{-1}M\Sigma^{-1})\right] \\
&= \Sigma^{-1}(M-X)\Sigma^{-1}
\end{aligned}
\tag{A.8}
$$

The last equality is obtained by $\frac{\partial}{\partial X}\mathrm{tr}(AX^T) = A$ and $\frac{\partial}{\partial X}\mathrm{tr}(AXBX) = A^TX^TB^T + B^TX^TA^T$.

A.2 Differentiation related to Riemannian CCA

The iterative method Algorithm 4 for Riemannian CCA with exact projection needs first and second derivative of g in (4.8). We provide more details here.

First derivative of g for SPD

Given SPD(n), the gradient of g with respect to t is obtained by the following proposition in (Moakher, 2005).

Proposition 3. *Let $F(t)$ be a real matrix-valued function of the real variable t. We assume that, for all t in its domain, $F(t)$ is an invertible matrix which does*

not have eigenvalues on the closed negative real line. Then

$$\frac{d}{dt}tr[\log^2 F(t)] = 2tr[\log F(t)F(t)^{-1}\frac{d}{dt}F(t)] \tag{A.9}$$

The derivation of $\frac{d}{dt_i}g(t_i, \boldsymbol{w}_x)$ proceeds as,

$$\begin{aligned}\frac{d}{dt_i}g(t_i, \boldsymbol{w}_x) &= \frac{d}{dt_i}\|\mathrm{Log}(\mathrm{Exp}(\mu_x, t_i W_x), X_i)\|^2 \\ &= \frac{d}{dt_i}\mathrm{tr}[\log^2(X_i^{-1}S(t_i))]\end{aligned} \tag{A.10}$$

where $S(t_i) = \mathrm{Exp}(\mu_x, t_i W_x) = \mu_x^{1/2}\exp^{t_i A}\mu_x^{1/2}$ and $A = \mu_x^{-1/2}W_x\mu_x^{-1/2}$.

In our formulation, $F(t) = X_i^{-1}S(t_i)$. Then we have $F(t)^{-1} = S(t_i)^{-1}X_i$ and $\frac{d}{dt}F(t) = X_i^{-1}\dot{S}(t_i)$. Hence, the derivative of g with respect to t_i is given by

$$\begin{aligned}\frac{d}{dt_i}g(t_i, \boldsymbol{w}_x) &= 2\mathrm{tr}[\log(X_i^{-1}S(t_i))S(t_i)^{-1}X_i X_i^{-1}\dot{S}(t_i)], \text{ according to proposition 1} \\ &= 2\mathrm{tr}[\log(X_i^{-1}S(t_i))S(t_i)^{-1}\dot{S}(t_i)]\end{aligned} \tag{A.11}$$

where $\dot{S}(t_i) = \mu_x^{1/2}A\exp^{t_i A}\mu_x^{1/2}$.

Numerical expression for the second derivative of g

Riemannian CCA with exact projection can be optimized by Algorithm 4. Observe that the objective function of the proposed augmented Lagrangian method, $\mathcal{L}_A$ includes the term ∇g in (4.8). The gradient of $\mathcal{L}_A$ involves the second derivative of g. More precisely, we need $\frac{d^2}{dwdt}g$ and

$\frac{d^2}{dt^2}g$. These can be estimated by a finite difference method,

$$f^{'}(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h} \tag{A.12}$$

Obviously, $\frac{d^2}{dt^2}g$ can be obtained by the expression above using the analytical first derivative $\frac{d}{dt}g$. For $\frac{d^2}{dwdt}g$, we use the orthonormal basis in $T_{\mu_x}\mathcal{M}$ to approximate the derivative. By definition of directional derivative, we have

$$\lim_{h \to 0} \sum_{i}^{d} \left(\frac{f(x+hu_i) - f(x)}{h} \right) u_i = \sum_{i}^{d} \langle \nabla_x f(x), u_i \rangle u_i = \nabla_x f(x) \tag{A.13}$$

where $x \in \mathcal{X}$, d is dimension of $\mathcal{X}$, and $\{u_i\}$ is orthonormal basis of $\mathcal{X}$. Hence, perturbation along the orthonormal basis enables us to approximate the gradient. For example, on SPD(n) manifolds, the orthonormal basis in arbitrary tangent space $T_p\mathcal{M}$ can be obtained by following three steps.

Step a) Pick an orthonormal basis $\{e_i\}$ of $R^{n(n+1)/2}$,

Step b) Convert $\{e_i\}$ into n-by-n symmetric matrices $\{u_i\}$ in $T_I\mathcal{M}$, i.e., $\{u_i\} =$mat$(\{e_i\})$,

Step c) Transform basis $\{u_i\}$ from $T_I\mathcal{M}$ to $T_p\mathcal{M}$.

A.3 Mixed effect models and longitudinal analysis

A.3.1 Standard Euclidean Mixed Effects Models

We briefly discussed in Chapter 6 how to estimate mixed effects models in the Euclidean space. For linear mixed effects models in the Euclidean space, multiple numerical techniques have been proposed such as EM

algorithms, Newton-Raphson methods, and MCMC. Particularly, EM algorithms treat random effects $\boldsymbol{u}$ as unobserved hidden variables and can naturally handle missing data.

As we discussed in Chapter 6, a linear mixed effects model is given as

$$\boldsymbol{y} = X\boldsymbol{\beta} + Z\boldsymbol{u} + \sigma_\epsilon^2 I, \tag{A.14}$$

where $\boldsymbol{u} \sim \mathcal{N}(0, \tilde{\Sigma})$ and $\tilde{\Sigma} = \text{diag}(\Sigma_1, \Sigma_2, \ldots, \Sigma_N) = \Sigma \otimes I$ (when $\Sigma_i = \Sigma, \quad \forall i$), and $Z = \text{diag}(Z_1, Z_2, \ldots, Z_N)$.

Especially, when the variances are known, regression coefficients $\boldsymbol{\beta}$ (fixed effects) and random effects $\boldsymbol{u}$ can be estimated by a closed form solution. When $\tilde{\Sigma}$ and σ_ϵ^2 are known (or can be estimated), the mixed effects model can be re-written as Henderson's mixed model equations (MME),

$$\begin{pmatrix} X'X & X'Z \\ Z'X & Z'Z + \sigma_\epsilon^2 \tilde{\Sigma}^{-1} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{u}} \end{pmatrix} = \begin{pmatrix} X'\boldsymbol{y} \\ Z'\boldsymbol{y} \end{pmatrix}.$$

Thus when the variances are known, the *generalized least squares estimation* can be used as

$$\hat{\boldsymbol{\beta}} = (X^T S^{-1} X)^{-1} X^T S^{-1} \boldsymbol{y} \tag{A.15}$$

$$\hat{\boldsymbol{u}} = \tilde{\Sigma} Z^T S^{-1} (\boldsymbol{y} - X\hat{\boldsymbol{\beta}}) \tag{A.16}$$

where $S := \sigma_\epsilon^2 I + Z\tilde{D}Z^T$.
Applying the Gauss-Markov theorem, $\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{u}}$ are the best linear unbiased estimates (BLUE) and predictors (BLUP), respectively (Lindstrom and Bates, 1988). Depending on the applications, the structure of covariance matrix may be differently specified as a function of some parameters, e.g., $\Sigma(\theta)$ rather than estimating the whole matrix (Demidenko, 2013).

A.3.2 Cauchy deformation tensors (CDTs)

Recall that Cauchy deformation tensors are derived from a nonlinear deformation $\Phi(\mathtt{vox})$ for voxels (spatial locations) $\mathtt{vox} \in \Omega$ for each image (rather, for each $(\mathcal{I}_{i,j+1}, \mathcal{I}_{i,j})$ pair) is given as

$$\begin{aligned} &\Phi : \mathcal{I}_{i,j+1} \rightarrow \mathcal{I}_{i,j} \\ &\Phi(\mathtt{vox} + d\mathtt{vox}) = \Phi(\mathtt{vox}) + J(\mathtt{vox}) d\mathtt{vox} + \mathcal{O}(d\mathtt{vox}^2), \end{aligned} \tag{A.17}$$

where $J(\mathtt{vox})$ denotes the Jacobian of the deformations at position $\mathtt{vox}$. A nice property of CDTs is that it preserves the determinant of $J(\mathtt{vox})$, since $\det(J(\mathtt{vox})) > 0$. So, a CDT representation introduced in the main, nicely symmetrizes $J(\mathtt{vox})$ without affecting the volumetric change information, i.e., $\det(J) = \det(\sqrt{J^T J})$. To prove this there are few assumptions which are commonly made by registraion algorithms to get a nonlinear deformation. In neuroimaging applications, registration algorithms generally assume that deformations are *diffeomorphic* and *orientation preserving*. A *diffeomorphism* requires a Jacoian matrix of deformation to be invertible. Orientation preserving implies that the determinants of Jacobian matrices are positive. Hence, the spatial gradient of deformation, (Jacobian $J(\mathtt{vox})$), forms a subgroup of general linear group, $\mathrm{GL}^+(n)$, which is a subgroup of invertible matrices with positive determinants, where n is the number of rows (or columns) of a matrix.

More explicitly each Jacobian matrix can be written as

$$J(\mathtt{vox}) = \mathcal{D}\Phi|_{\mathtt{vox}} = \begin{pmatrix} \partial_1\phi^1|_{\mathtt{vox}} & \partial_2\phi^1|_{\mathtt{vox}} & \partial_3\phi^1|_{\mathtt{vox}} \\ \partial_1\phi^2|_{\mathtt{vox}} & \partial_2\phi^2|_{\mathtt{vox}} & \partial_3\phi^2|_{\mathtt{vox}} \\ \partial_1\phi^3|_{\mathtt{vox}} & \partial_2\phi^3|_{\mathtt{vox}} & \partial_3\phi^3|_{\mathtt{vox}} \end{pmatrix}. \tag{A.18}$$

where $\mathcal{D}$ is the Jacobian operator (derivative of vector field) and $\partial_i\phi^j$ is a derivative along i and j component of Ω and Γ respectively.

Now, CDTs can be derived from $J(x)$ with matrix operation

$$\text{CDT}(\texttt{vox}) = \sqrt{J(\texttt{vox})^T J(\texttt{vox})}, \tag{A.19}$$

where $\sqrt{(\cdot)}$ is matrix square root. As mentioned above, one nice property of CDTs is that it preserves the determinant of $J(\texttt{vox})$, since $\det(J(\texttt{vox})) > 0$. So, CDT transformation nicely symmetrizes $J(\texttt{vox})$ without changing information of volumetric changes, i.e., $\det(J) = \det(\sqrt{J^T J})$.

Lemma A.1. $\det(J) = \det(\sqrt{J^T J})$

Proof. In general, the square root of a matrix can be multiple. Fortunately, positive (semi) definite matrix has a unique positive (semi) definite square root matrix (Horn and Johnson, 2012). Also the square root matrix of a symmetric positive (semi) definite matrix can be written as

$$X^{1/2} = VD^{1/2}V^T, \text{where } X = X^T, X \succeq 0, X = VDV^T \tag{A.20}$$

So, let $X = J^T J$. Then, since $X \succeq 0, X^T = X$, we have $\det(\sqrt{J^T J}) = \det(X^{1/2}) = \prod_i \sqrt{D_{ii}} = \sqrt{\prod_i D_{ii}} = \sqrt{\det(X)} = \sqrt{\det(J^T)\det(J)} = \sqrt{\det(J)\det(J)} = \det(J)$. □

A.3.3 Unbiased estimation of CDTs

We present experimental details on deriving the deformation fields and then the CDT images which capture the subject-wise longitudinal changes. These CDT images will be in a least biased coordinate system to allow for voxel-wise analysis of morphometric longitudinal changes. We first estimate an unbiased global template space as shown in Fig. A.1. While estimating an unbiased atlas (coordinate system) is well investigated in cross-sectional imaging studies, there are fewer validation studies for

the 3D+time regime. The additional bias which we must restrict in a longitudinal study is the interpolation asymmetry that can arise when selecting only one of the time points as a temporal representative in generating the population/study level coordinate system. Based on the current best practices, we first estimate a subject-specific average that is temporally unbiased. The subject-specific averages are then used to generate an unbiased population level average template space (Fig. A.1). Each of the curved black lines represents a combination of affine and non-linear diffeomorphic transformations. These transformations and the spatial averages are estimated iteratively until convergence using ANTS (Avants et al., 2008; Tustison and Avants, 2013) based implementation.

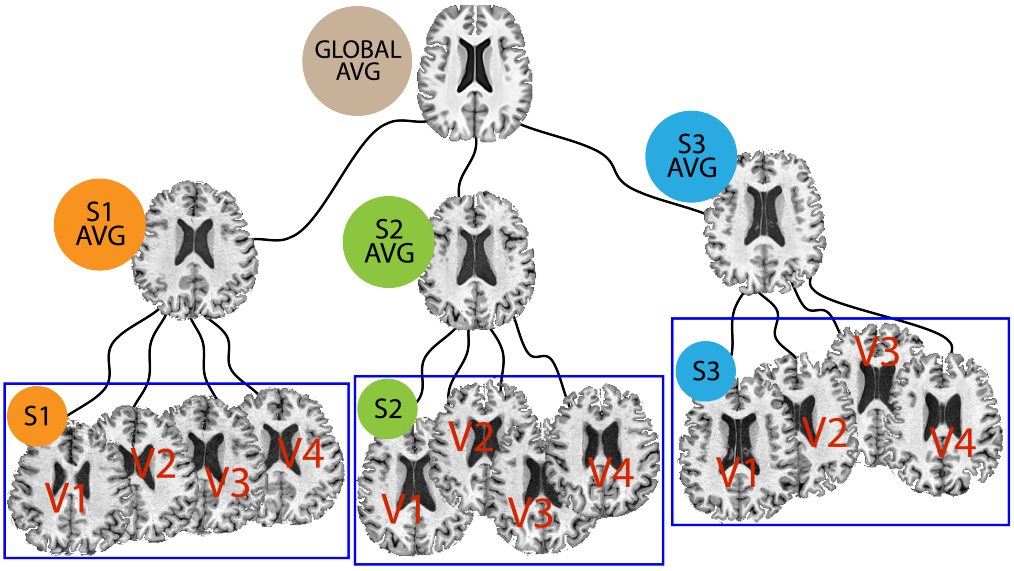


Figure A.1: Schematic for generating least biased global coordinate system for the longitudinally acquired imaging data. Visits V1-V4 are averaged first which are then used to estimate the global average.

So far, we have only described constructing a common coordinate system. N ext we describe how the longitudinal deformation fields are generated in the global coordinate system for voxel-wise analyses. This

step turns out to be non-trivial for CDT images. Most existing publicly available pipelines such as SPM, FreeSurfer, AFNI, FSL do not generate such results and instead provide scalar/univariate images representing longitudinal magnitude of Jacobians or divergences of the deformation fields. Note that the widely used Freesurfer processing streams allows subject-level longitudinal analysis of features such as rate of change of thickness on the cortical surfaces but not the morphological changes themselves. While such "summary" measures of structures are relevant, in the case of preclinical AD, they are complementary to morphological changes captured at the *voxel-level*. Fortunately based on the work in gravitation theory Lorenzi and Pennec recently developed a computationally efficient framework for parallel transport of stationary velocity fields along other stationary velocity fields (SVFs) (Lorenzi and Pennec, 2014). Using this framework we can obtain *longitudinal* deformations in a global coordinate system. We first register the set of longitudinal images (using rigid transformations - rotations and translations) from all visits (V_i) to the global average (GA) estimated as described in the previous paragraph i.e. $V_i \mapsto_R GA, \quad \forall i$. We thus have V_i^R and GA in the same global coordinate system. Now the key non-linear symmetric diffeomorphic deformations are generated using (Lorenzi et al., 2013). Images are registered pairwise between consecutive visits, i.e. $V_{i+1}^R \hookrightarrow_{SVF} V_i^R$) resulting in a stationary velocity field (SVF) ($\mathcal{V}^{SVF}_{(i+1)\mapsto i}$) representing longitudinal progression between visits $i+1$ and i. The individual visit images V_i^R are non-linearly registered to GA ($V_i^R \hookrightarrow_{SVF} GA$) resulting in a "subject-to-template" SVF, $\mathcal{V}_i^{SVF}$. The $\mathcal{V}^{SVF}_{(i+1)\mapsto i}$ are then parallel transported in the direction of $\mathcal{V}_i^{SVF}$ resulting in SVFs that represents longitudinal progression in the global coordinate system. Cauchy deformation tensor fields are constructed from Jacobian matrices of these final transported vector fields.

REFERENCES

Absil, P-A, Christopher G Baker, and Kyle A Gallivan. 2007. Trust-region methods on riemannian manifolds. *Foundations of Computational Mathematics* 7(3):303–330.

Absil, P-A, Robert Mahony, and Rodolphe Sepulchre. 2009. *Optimization algorithms on matrix manifolds*. Princeton University Press.

Aganj, Iman, Christophe Lenglet, and Guillermo Sapiro. 2009. Odf reconstruction in q-ball imaging with solid angle consideration. In *Biomedical imaging: From nano to macro, 2009. isbi'09. ieee international symposium on*, 1398–1401. IEEE.

Ahlberg, J Harold, Edwin Norman Nilson, and Joseph Leonard Walsh. 2016. *The theory of splines and their applications: Mathematics in science and engineering: A series of monographs and textbooks*, vol. 38. Elsevier.

Ahmed, Amr, and Eric P Xing. 2009. Recovering time-varying networks of dependencies in social and biological studies. *Proceedings of the National Academy of Sciences* 106(29):11878–11883.

Akaho, Shotaro. 2006. A kernel method for canonical correlation analysis. *arXiv preprint cs/0609071*.

Alexander, Andrew L, Jee Eun Lee, Mariana Lazar, and Aaron S Field. 2007. Diffusion tensor imaging of the brain. *Neurotherapeutics* 4(3):316–329.

Alexander, Daniel C, Carlo Pierpaoli, Peter J Basser, and James C Gee. 2001. Spatial transformations of diffusion tensor magnetic resonance images. *IEEE Transactions on Medical Imaging* 20(11):1131–1139.

Amari, S, and H Nagaoka. 2000. Methods of information geometry.

Amari, Shun-ichi. 1985. Differential-geometrical methods in statistics.

Andrew, Galen, Raman Arora, Jeff Bilmes, and Karen Livescu. 2013. Deep canonical correlation analysis. In *International conference on machine learning*, 1247–1255.

Antoine, J-P, and P Vandergheynst. 1998. Wavelets on the n-sphere and related manifolds. *Journal of mathematical physics* 39(8):3987–4008.

Avants, B. B., D. J. Libon, K. Rascovsky, A. Boller, C.T. McMillan, L. Massimo, H. Coslett, A. Chatterjee, R.G. Gross, and M. Grossman. 2014. Sparse canonical correlation analysis relates network-level atrophy to multivariate cognitive measures in a neurodegenerative population. *NeuroImage* 84(1):698—-711.

Avants, Brian B, Charles L Epstein, et al. 2008. Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. *Medical image analysis* 12(1): 26–41.

Aydın, Burcu, Gábor Pataki, Haonan Wang, Alim Ladha, Elizabeth Bullitt, and James Stephen Marron. 2012. New approaches to principal component analysis for trees. *Statistics in Biosciences* 4(1):132–156.

Bach, Francis R, and Michael I Jordan. 2002. Kernel independent component analysis. *Journal of machine learning research* 3(Jul):1–48.

Banerjee, Monami, Rudrasis Chakraborty, Edward Ofori, Michael S Okun, David E Viallancourt, and Baba C Vemuri. 2016. A nonlinear regression technique for manifold valued data with applications to medical image analysis. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 4424–4432.

Baringhaus, Ludwig, and Carsten Franz. 2004. On a new multivariate two-sample test. *Journal of multivariate analysis* 88(1):190–206.

Barmpoutis, Angelos, Baba C Vemuri, Timothy M Shepherd, and John R Forder. 2007. Tensor splines for interpolation and approximation of dt-mri with applications to segmentation of isolated rat hippocampi. *IEEE transactions on medical imaging* 26(11):1537–1546.

Basser, Peter J, James Mattiello, and Denis LeBihan. 1994. Mr diffusion tensor spectroscopy and imaging. *Biophysical journal* 66(1):259–267.

Bastiani, Matteo, Nadim Jon Shah, Rainer Goebel, and Alard Roebroeck. 2012. Human cortical connectome reconstruction from diffusion weighted MRI: the effect of tractography algorithm. *Neuroimage* 62(3): 1732–1749.

Benjamini, Yoav, and Yosef Hochberg. 1995. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)* 289–300.

Bertsekas, Dimitri P. 1999. *Nonlinear programming*. Athena scientific Belmont.

Bonferroni, Carlo E. 1936. *Teoria statistica delle classi e calcolo delle probabilita*. Libreria internazionale Seeber.

Bonnabel, Silvere, and Rodolphe Sepulchre. 2009. Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank. *SIAM Journal on Matrix Analysis and Applications* 31(3):1055–1070.

Boscaini, Davide, Jonathan Masci, Emanuele Rodolà, and Michael Bronstein. 2016. Learning shape correspondence with anisotropic convolutional neural networks. In *Advances in neural information processing systems*, 3189–3197.

Boumal, Nicolas. 2014. Optimization and estimation on manifolds. Ph.D. thesis, Catholic University of Louvain, Louvain-la-Neuve, Belgium.

Boumal, Nicolas, and Pierre-antoine Absil. 2011. Rtrmc: A riemannian trust-region method for low-rank matrix completion. In *Advances in neural information processing systems*, 406–414.

Boumal, Nicolas, Bamdev Mishra, Pierre-Antoine Absil, Rodolphe Sepulchre, et al. 2014. Manopt, a matlab toolbox for optimization on manifolds. *Journal of Machine Learning Research* 15(1):1455–1459.

Bourgon, Richard, Robert Gentleman, and Wolfgang Huber. 2010. Independent filtering increases detection power for high-throughput experiments. *Proceedings of the National Academy of Sciences* 107(21):9546–9551.

Boyd, Stephen P, and Lieven Vandenberghe. 2004. *Convex optimization*. Cambridge university press.

Bronstein, Michael M, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. 2016. Geometric deep learning: going beyond euclidean data. *arXiv preprint arXiv:1611.08097*.

Bruna, Joan, and Stéphane Mallat. 2013. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence* 35(8):1872–1886.

Bruna, Joan, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. 2013. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*.

Callaghan, Paul T. 1991. *Principles of nuclear magnetic resonance microscopy*. Oxford University Press.

Candes, Emmanuel, and Benjamin Recht. 2012. Exact matrix completion via convex optimization. *Communications of the ACM* 55(6):111–119.

Cetingul, H Ertan, Bijan Afsari, Margaret J Wright, Paul M Thompson, and René Vidal. 2012. Group action induced averaging for HARDI

processing. In *Ieee international symposium on biomedical imaging*, 1389–1392.

Chakraborty, Rudrasis, and Baba C Vemuri. 2015. Recursive fréchet mean computation on the grassmannian and its applications to computer vision. In *Proceedings of the ieee international conference on computer vision*, 4229–4237.

Cheng, Guang, Jeffrey Ho, Hesamoddin Salehian, and Baba C Vemuri. 2016. Recursive computation of the fréchet mean on non-positively curved riemannian manifolds with applications. In *Riemannian computing in computer vision*, 21–43. Springer.

Cheng, Guang, Hesamoddin Salehian, and Baba Vemuri. 2012. Efficient recursive algorithms for computing the mean diffusion tensor and applications to dti segmentation. *Computer Vision–ECCV 2012* 390–401.

Cheng, Guang, and Baba C Vemuri. 2013. A novel dynamic system in the space of SPD matrices with applications to appearance tracking. *SIAM journal on imaging sciences* 6(1):592–615.

Cheng, Jian, , Aurobrata Ghosh, Tianzi Jiang, and Rachid Deriche. 2010. Model-free and analytical EAP reconstruction via spherical polar fourier diffusion MRI. In *MICCAI*, 590–597.

———. 2011. Diffeomorphism invariant Riemannian framework for ensemble average propagator computing. In *MICCAI*, 98–106.

Cheng, Jian. 2012. Estimation and processing of ensemble average propagator and its features in diffusion mri. Ph.D. thesis, Université Nice Sophia Antipolis.

Cheng, Jian, Aurobrata Ghosh, Tianzi Jiang, and Rachid Deriche. 2009. A Riemannian framework for orientation distribution function computing. In *Miccai*, 911–918. Springer.

Chikuse, Yasuko. 2003. Statistics on special manifolds.

Chung, MK, KJ Worsley, T Paus, C Cherif, DL Collins, JN Giedd, JL Rapoport, and AC Evans. 2001. A unified statistical approach to deformation-based morphometry. *NeuroImage* 14(3):595–606.

Cohen, Taco, and Max Welling. 2016. Group equivariant convolutional networks. In *International conference on machine learning*, 2990–2999.

Cohen, Taco S, and Max Welling. 2015. Harmonic exponential families on manifolds. *arXiv preprint arXiv:1505.04413*.

Cook, PA., Y. Bai, S. Nedjati-Gilani, K. K. Seunarine, M. G. Hall, G. J. Parker, and D. C. Alexander. 2006. Camino: Open-source diffusion-MRI reconstruction and processing. In *Ismrm*, 2759.

Corcuera, José Manuel, and Wilfrid S Kendall. 1999. Riemannian barycentres and geodesic convexity. In *Mathematical proceedings of the cambridge philosophical society*, vol. 127, 253–269. Cambridge University Press.

Corder, EH, AM Saunders, et al. 1993. Gene dose of apolipoprotein E type 4 allele and the risk of alzheimer's disease in late onset families. *Science* 261(5123):921–923.

Cornea, Emil, Hongtu Zhu, et al. 2016. Regression models on Riemannian symmetric spaces. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*.

Cramér, Harald. 1928. On the composition of elementary errors: First paper: Mathematical deductions. *Scandinavian Actuarial Journal* 1928(1): 13–74.

Dahl, David B, and Michael A Newton. 2007. Multiple hypothesis testing by clustering treatment effects. *Journal of the American Statistical Association* 102(478):517–526.

Dahlke, Stephan. 1994. Multiresolution analysis, haar bases and wavelets on riemannian manifolds. *Wavelets: Theory, Algorithms, and Applications (Taormina 1993), Wavelet Analysis and Its Applications* 5:33–52.

Damon, James, and JS Marron. 2014. Backwards principal component analysis and principal nested relations. *Journal of mathematical imaging and vision* 50(1-2):107–114.

Davis, BC, PT Fletcher, E Bullitt, and S Joshi. 2007. Population shape regression from random design data. In *2007 ieee 11th international conference on computer vision*.

Demidenko, Eugene. 2013. *Mixed models: theory and applications with r*. John Wiley & Sons.

Deza, Michel Marie, and Elena Deza. 2009. *Encyclopedia of distances*, vol. 94. Springer.

Do Carmo, Manfredo P. 1992. *Riemannian geometry*. Springer.

Dominici, Francesca, Aidan McDermott, Scott L Zeger, and Jonathan M Samet. 2002. On the use of generalized additive models in time-series studies of air pollution and health. *American journal of epidemiology* 156(3): 193–203.

Donoho, David L. 2006. Compressed sensing. *IEEE Transactions on Information Theory* 52(4):1289–1306.

Dreisigmeyer, David W. 2006. Direct search algorithms over riemannian manifolds. *Submitted to SIOPT*.

Du, Jia, Alvina Goh, Sergey Kushnarev, and Anqi Qiu. 2013. Geodesic regression on ODFs with its application to an aging study. *NeuroImage* 13:1053–8119.

———. 2014. Geodesic regression on orientation distribution functions with its application to an aging study. *NeuroImage* 87:416–426.

Durrleman, Stanley, Xavier Pennec, Alain Trouvé, Guido Gerig, and Nicholas Ayache. 2009. Spatiotemporal atlas estimation for developmental delay detection in longitudinal datasets. In *International conference on medical image computing and computer-assisted intervention*, 297–304. Springer.

Durrleman, Stanley, Xavier Pennec, et al. 2013. Toward a comprehensive framework for the spatiotemporal statistical analysis of longitudinal shape data. *IJCV* 103(1):22–59.

Elhamifar, Ehsan, and René Vidal. 2009. Sparse subspace clustering. In *Computer vision and pattern recognition, 2009. cvpr 2009. ieee conference on*, 2790–2797. IEEE.

Feragen, Aasa, François Lauze, and Soren Hauberg. 2015. Geodesic exponential kernels: When curvature and linearity conflict. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 3032–3042.

Ferreira, Ricardo, Joao Xavier, Joao Paulo Costeira, and Victor Barroso. 2006. Newton method for riemannian centroid computation in naturally reductive homogeneous spaces.

Fishbaugh, James, Marcel Prastawa, Stanley Durrleman, Joseph Piven, and Guido Gerig. 2012. Analysis of longitudinal shape variability via subject specific growth modeling. *Medical Image Computing and Computer-assisted Intervention–MICCAI 2012* 731–738.

Fletcher, P Thomas. 2013. Geodesic regression and the theory of least squares on Riemannian manifolds. *IJCV* 105(2):171–185.

Fletcher, P Thomas, Conglin Lu, Stephen M Pizer, and Sarang Joshi. 2004. Principal geodesic analysis for the study of nonlinear statistics of shape. *IEEE transactions on medical imaging* 23(8):995–1005.

Fletcher, P Thomas, Suresh Venkatasubramanian, and Sarang Joshi. 2009. The geometric median on riemannian manifolds with application to robust atlas estimation. *NeuroImage* 45(1):S143–S152.

Frackowiak, Richard SJ. 2004. *Human brain function*. Academic press.

Frank, Lawrence R. 2002. Characterization of anisotropy in high angular resolution diffusion-weighted mri. *Magnetic Resonance in Medicine* 47(6): 1083–1099.

Freeborough, Peter A, and Nick C Fox. 1998. Modeling brain deformations in alzheimer disease by fluid registration of serial 3d mr images. *Journal of computer assisted tomography* 22(5):838–843.

Freifeld, Oren. 2013. Statistics on manifolds with applications to modeling shape deformations. Ph.D. thesis, Brown University.

Friston, K.J., J. Ashburner, S.J. Kiebel, T.E. Nichols, and W.D. Penny, eds. 2007. *Statistical parametric mapping: The analysis of functional brain images*. Academic Press.

Gabay, Daniel. 1982. Minimizing a differentiable function over a differential manifold. *Journal of Optimization Theory and Applications* 37(2): 177–219.

Garrido, Lucia, Nicholas Furl, Bogdan Draganski, Nikolaus Weiskopf, John Stevens, Geoffrey Chern-Yee Tan, Jon Driver, Ray J Dolan, and Bradley Duchaine. 2009. Voxel-based morphometry reveals reduced grey matter volume in the temporal cortex of developmental prosopagnosics. *Brain* 132(12):3443–3455.

Girolami, Mark, and Ben Calderhead. 2011. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73(2):123–214.

Goh, Alvina. 2010. *Estimation and processing of orientation distribution functions for high angular resolution diffusion images*. THE JOHNS HOPKINS UNIVERSITY.

Goh, Alvina, Christophe Lenglet, Paul M Thompson, and René Vidal. 2009. A nonparametric Riemannian framework for processing high angular resolution diffusion images (HARDI) 2496–2503.

———. 2011. A nonparametric Riemannian framework for processing HARDI and its applications to ODF-based morphometry. *NeuroImage* 56(3):1181–1201.

Gower, John C, and Garmt B Dijksterhuis. 2004. *Procrustes problems*, vol. 30. Oxford University Press on Demand.

Grenander, Ulf, and Gabor Szegö. 2001. *Toeplitz forms and their applications*, vol. 321. Univ of California Press.

Guo, Guodong, Rui Guo, and Xin Li. 2013. Facial expression recognition influenced by human aging. *IEEE Trans. Affective Computing* 4(3):291–298.

Hamelryck, Thomas, John T Kent, and Anders Krogh. 2006. Sampling realistic protein conformations using local structural bias. *PLoS Computational Biology* 2(9):e131.

Hamm, Jihun, and Daniel D Lee. 2008. Grassmann discriminant analysis: a unifying view on subspace-based learning. In *Proceedings of the 25th international conference on machine learning*, 376–383. ACM.

Hannah, Lauren A, David M Blei, and Warren B Powell. 2011. Dirichlet process mixtures of generalized linear models. *JMLR* 12:1923–1953.

Harandi, Mehrtash, Mathieu Salzmann, and Richard Hartley. 2017. Dimensionality reduction on SPD manifolds: The emergence of geometry-aware methods. *IEEE transactions on pattern analysis and machine intelligence*.

Hardoon, David R, Janaina Mourao-Miranda, Michael Brammer, and John Shawe-Taylor. 2007. Unsupervised analysis of fmri data using kernel canonical correlation. *NeuroImage* 37(4):1250–1259.

Hardoon, David R, Sandor Szedmak, and John Shawe-Taylor. 2004. Canonical correlation analysis: An overview with application to learning methods. *Neural computation* 16(12):2639–2664.

Hastie, Trevor, and Robert Tibshirani. 1993. Varying-coefficient models. *Journal of the Royal Statistical Society. Series B (Methodological)* 757–796.

Hershey, John R, and Peder A Olsen. 2007. Approximating the Kullback Leibler divergence between Gaussian mixture models. In *Ieee international conference on acoustics, speech and signal processing*, vol. 4, IV–317.

Hinkle, Jacob, Prasanna Muralidharan, PTHOMAS Fletcher, and Sarang Joshi. 2012. Polynomial regression on Riemannian manifolds. *ECCV* 1–14.

Ho, Jeffrey, Guang Cheng, et al. 2013. Recursive karcher expectation estimators and geometric law of large number. In *Aistats*, 325–332.

Hoerl, Arthur E, and Robert W Kennard. 1970. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* 12(1):55–67.

Hong, Yi, Nikhil Singh, Roland Kwitt, and Marc Niethammer. 2014. Time-warped geodesic regression. In *Miccai*, vol. 17, 105. NIH Public Access.

Horn, Roger A, and Charles R Johnson. 2012. *Matrix analysis*. Cambridge university press.

Hotelling, Harold. 1936. Relations between two sets of variates. *Biometrika* 28(3/4):321–377.

Hsieh, William W. 2000. Nonlinear canonical correlation analysis by neural networks. *Neural Networks* 13(10):1095–1105.

Hsu, Jason. 1996. *Multiple comparisons: theory and methods*. CRC Press.

Hua, Xue, Alex D Leow, Neelroop Parikshak, Suh Lee, Ming-Chang Chiang, Arthur W Toga, Clifford R Jack, Michael W Weiner, Paul M Thompson, Alzheimer's Disease Neuroimaging Initiative, et al. 2008. Tensor-based morphometry as a neuroimaging biomarker for alzheimer's disease: an mri study of 676 ad, mci, and normal subjects. *Neuroimage* 43(3):458–469.

Huang, Zhiwu, and Luc J Van Gool. 2017. A riemannian network for spd matrix learning. In *Aaai*, vol. 2, 6.

Huang, Zhiwu, Jiqing Wu, and Luc Van Gool. 2016. Building deep networks on grassmann manifolds. *arXiv:1611.05742*.

Huckemann, Stephan, Thomas Hotz, and Axel Munk. 2010a. Intrinsic shape analysis: Geodesic PCA for Riemannian manifolds modulo isometric lie group actions. *Statistica Sinica* 1–58.

———. 2010b. Intrinsic shape analysis: Geodesic PCA for Riemannian manifolds modulo isometric Lie group actions. *Statistica Sinica* 20:1–100.

Jae Hwang, Seong, Maxwell D Collins, Sathya N Ravi, Vamsi K Ithapu, Nagesh Adluru, Sterling C Johnson, and Vikas Singh. 2015. A projection

free method for generalized eigenvalue problem with a nonsmooth regularizer. In *Proceedings of the ieee international conference on computer vision*, 1841–1849.

Jayasumana, Sadeep, Richard Hartley, Mathieu Salzmann, Hongdong Li, and Mehrtash Harandi. 2013. Kernel methods on the Riemannian manifold of symmetric positive definite matrices. In *Cvpr*, 73–80.

Jenkinson, Mark, Christian F Beckmann, Timothy EJ Behrens, Mark W Woolrich, and Stephen M Smith. 2012. FSL. *Neuroimage* 62(2):782–790.

Jensen, Jesper Hojvang, Daniel PW Ellis, Mads G Christensen, and Soren Holdt Jensen. 2007. Evaluation distance measures between Gaussian mixture models of MFCCs. In *International conference on music information retrieval*, 107–108.

Jian, Bing, and Baba Vemuri. 2007a. Multi-fiber reconstruction from diffusion mri using mixture of wisharts and sparse deconvolution. In *Information processing in medical imaging*, 384–395. Springer Berlin/Heidelberg.

———. 2011. Robust point set registration using Gaussian mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33(8):1633–1645.

Jian, Bing, and Baba C Vemuri. 2007b. A unified computational framework for deconvolution to reconstruct multiple fibers from diffusion weighted mri. *IEEE transactions on medical imaging* 26(11):1464–1471.

Jian, Bing, Baba C Vemuri, Evren Özarslan, Paul R Carney, and Thomas H Mareci. 2007. A novel tensor distribution model for the diffusion-weighted mr signal. *NeuroImage* 37(1):164–176.

Joshi, Sarang, Brad Davis, Matthieu Jomier, and Guido Gerig. 2004. Unbiased diffeomorphic atlas construction for computational anatomy. *NeuroImage* 23:S151–S160.

Jung, Sungkyu, Ian L Dryden, JS Marron, et al. 2012. Analysis of principal nested spheres. *Biometrika* 99(3):551–568.

Jung, Sungkyu, Xiaoxiao Liu, JS Marron, and Stephen M Pizer. 2010. Generalized pca via the backward stepwise approach in image analysis. In *Brain, body and machine*, 111–123. Springer.

Jupp, Peter E, and John T Kent. 1987. Fitting smooth paths to speherical data. *Applied statistics* 34–46.

Karcher, Hermann. 1977. Riemannian center of mass and mollifier smoothing. *Communications on pure and applied mathematics* 30(5):509–541.

Kendall, David G. 1984. Shape manifolds, procrustean metrics, and complex projective spaces. *Bulletin of the London Mathematical Society* 16(2):81–121.

Kilian, Martin, Niloy J Mitra, and Helmut Pottmann. 2007. Geometric modeling in shape space. In *Acm transactions on graphics (tog)*, vol. 26, 64. ACM.

Kim, Hyunwoo, Nagesh Adluru, Sterling C Johnson, and Vikas Singh. 2016a. Manifold-valued statistical models for longitudinal morphometric analysis in preclinical alzheimer's disease. *Alzheimer's & Dementia: The Journal of the Alzheimer's Association* 12(7):P529–P530.

Kim, Hyunwoo J, Nagesh Adluru, Monami Banerjee, Baba C Vemuri, and Vikas Singh. 2015a. Interpolation on the manifold of k component GMMs. In *Proceedings of international conference on computer vision (iccv)*. Santiago, Chile.

Kim, Hyunwoo J, Nagesh Adluru, Barbara B Bendlin, Sterling C Johnson, Baba C Vemuri, and Vikas Singh. 2014a. Canonical correlation analysis on Riemannian manifolds and its applications. In *Eccv*, 251–267.

———. 2016b. Canonical correlation analysis on SPD (n) manifolds. In *Riemannian computing in computer vision*, 69–100. Springer.

Kim, Hyunwoo J, Nagesh Adluru, Maxwell D Collins, Moo K Chung, Barbara B Bendlin, Sterling C Johnson, Richard J Davidson, and Vikas Singh. 2014b. Multivariate general linear models (MGLM) on Riemannian manifolds with applications to statistical analysis of diffusion weighted images. In *Cvpr*, 2705–2712.

Kim, Hyunwoo J., Nagesh Adluru, Maxwell D. Collins, Moo K. Chung, Barbara B. Bendlin, Sterling C. Johnson, Richard J. Davidson, and Vikas Singh. 2014c. Multivariate general linear models (MGLM) on Riemannian manifolds with applications to statistical analysis of diffusion weighted images. In *Cvpr*.

Kim, Hyunwoo J., Nagesh Adluru, Heemanshu Suri, Baba C. Vemuri, Sterling C. Johnson, and Vikas Singh. 2017a. Riemannian nonlinear mixed effects models: Analyzing longitudinal deformations in neuroimaging. In *Cvpr*. Hawaii, Honolulu.

Kim, Hyunwoo J, Nagesh Adluru, Heemanshu Suri, Baba C Vemuri, Sterling C Johnson, and Vikas Singh. 2017b. Riemannian nonlinear mixed effects models: Analyzing longitudinal deformations in neuroimaging. In *Cvpr*, 1–8.

Kim, Hyunwoo J., Jia Xu, Baba C. Vemuri, and Vikas Singh. 2015b. Manifold-valued Dirichlet processes. In *Proceedings of the 32nd international conference on machine learning*, 1199–1208.

Kim, Won Hwa, Hyunwoo J. Kim, Nagesh Adluru, and Vikas Singh. 2016c. Latent variable graphical model selection using harmonic analysis: Applications to the human connectome project (HCP). In *The ieee conference on computer vision and pattern recognition (cvpr)*. Las Vegas, Nevada, USA.

Klassen, Eric, Anuj Srivastava, M Mio, and Shantanu H Joshi. 2004. Analysis of planar shapes using geodesic paths on shape spaces. *IEEE transactions on pattern analysis and machine intelligence* 26(3):372–383.

Klein, Arno, Jesper Andersson, et al. 2009. Evaluation of 14 nonlinear deformation algorithms applied to human brain mri registration. *Neuroimage* 46(3):786–802.

Kullback, Solomon. 1997. *Information theory and statistics*. Courier Corporation.

Lai, Pei Ling, and Colin Fyfe. 1999. A neural implementation of canonical correlation analysis. *Neural Networks* 12(10):1391–1397.

Laird, Nan M, and James H Ware. 1982. Random-effects models for longitudinal data. *Biometrics* 963–974.

Larsen, Richard J, Morris L Marx, et al. 1986. *An introduction to mathematical statistics and its applications*, vol. 2. Prentice-Hall Englewood Cliffs, NJ.

Le Bihan, Denis, Jean-François Mangin, Cyril Poupon, Chris A Clark, Sabina Pappata, Nicolas Molko, and Hughes Chabriat. 2001. Diffusion tensor imaging: concepts and applications. *Journal of magnetic resonance imaging* 13(4):534–546.

Lebanon, Guy, et al. 2005. Riemannian geometry and statistical machine learning. Ph.D. thesis, Carnegie Mellon University, Language Technologies Institute, School of Computer Science.

Lee, John M. 2006. *Riemannian manifolds: an introduction to curvature*, vol. 176. Springer Science & Business Media.

———. 2012. *Introduction to smooth manifolds*. Springer.

Lenglet, Christophe, Mikaël Rousson, Rachid Deriche, and Olivier Faugeras. 2006. Statistics on the manifold of multivariate normal distributions: Theory and application to diffusion tensor mri processing. *Journal of Mathematical Imaging and Vision* 25(3):423–444.

Leow, Alex D, Siwei Zhu, Liang Zhan, Katie McMahon, Greig I de Zubicaray, Matthew Meredith, MJ Wright, AW Toga, and PM Thompson. 2009. The tensor distribution function. *Magnetic Resonance in Medicine* 61(1):205–214.

Lepore, Natasha, Caroline Brun, Yi-Yu Chou, Ming-Chang Chiang, Rebecca Dutton, Kiralee M Hayashi, Eileen Luders, Oscar L Lopez, Howard J Aizenstein, Arthur W Toga, et al. 2008. Generalized tensor-based morphometry of hiv/aids using multivariate statistics on deformation tensors. *Medical Imaging, IEEE Transactions on* 27(1):129–141.

Li, Junning, Yonggang Shi, and Arthur W Toga. 2014. Diffusion of fiber orientation distribution functions with a rotation-induced Riemannian metric. In *Medical image computing and computer-assisted intervention*, 249–256.

Lindstrom, Mary J, and Douglas M Bates. 1988. Newton—raphson and em algorithms for linear mixed-effects models for repeated-measures data. *Journal of the American Statistical Association* 83(404):1014–1022.

———. 1990. Nonlinear mixed effects models for repeated measures data. *Biometrics* 673–687.

Lorenzi, Marco, Nicholas Ayache, Giovanni Frisoni, and Xavier Pennec. 2011a. Mapping the effects of aβ 1- 42 levels on the longitudinal

changes in healthy aging: hierarchical modeling based on stationary velocity fields. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2011* 663–670.

Lorenzi, Marco, Nicholas Ayache, Giovanni B Frisoni, Xavier Pennec, Alzheimer's Disease Neuroimaging Initiative (ADNI, et al. 2013. Lcc-demons: a robust and accurate symmetric diffeomorphic registration algorithm. *NeuroImage* 81:470–483.

Lorenzi, Marco, Nicholas Ayache, and Xavier Pennec. 2011b. Schild's ladder for the parallel transport of deformations in time series of images. In *Information processing in medical imaging*, 463–474. Springer.

Lorenzi, Marco, and Xavier Pennec. 2014. Efficient parallel transport of deformations in time series of images: from schild's to pole ladder. *Journal of Mathematical Imaging and Vision* 50(1-2):5–17.

Mak, M.W, S.Y. Kung, and S.H. Lin. 2004. Expectation Maximization Theory. *Biometric Authentication: A Machine Learning Approach* 61(1): 503–512.

Mammasis, Konstantinos, and RobertW Stewart. 2010. Spherical statistics and spatial correlation for multielement antenna systems. *EURASIP Journal on Wireless Communications and Networking* 2010(1):307265.

Manton, Jonathan H. 2004. A globally convergent numerical algorithm for computing the centre of mass on compact lie groups. In *Control, automation, robotics and vision conference, 2004. icarcv 2004 8th*, vol. 3, 2211–2216. IEEE.

Mardia, Kantilal Varichand, and Peter E Jupp. 1999. Directional statistics.

Marron, J Steve, and Andrés M Alonso. 2014. Overview of object oriented data analysis. *Biometrical Journal* 56(5):732–753.

Masci, Jonathan, Davide Boscaini, Michael Bronstein, and Pierre Vandergheynst. 2015. Geodesic convolutional neural networks on Riemannian manifolds. In *Cvpr workshops*, 37–45.

McCulloch, Charles E, and John M Neuhaus. 2001. *Generalized linear mixed models*. Wiley Online Library.

Meza, Cristian, Florence Jaffrézic, et al. 2007. REML estimation of variance parameters in nonlinear mixed effects models using the saem algorithm. *Biometrical Journal* 49(6):876–888.

Minear, Meredith, and Denise C Park. 2004. A lifespan database of adult facial stimuli. *Behavior Research Methods, Instruments, & Computers* 36(4): 630–633.

Moakher, Maher. 2005. A differential geometric approach to the geometric mean of symmetric positive-definite matrices. *SIAM Journal on Matrix Analysis and Applications* 26(3):735–747.

———. 2006. A riemannian framework for the averaging, smoothing and interpolation of some matrix-valued data. In *Processing of the 17th international symposium on mathematical theory of networks and systems*, 1720–1729.

Moakher, Maher, and Philipp G Batchelor. 2006. Symmetric positive-definite matrices: From geometry to applications and visualization. In *Visualization and processing of tensor fields*, 285–298. Springer.

Montgomery, Douglas C, Elizabeth A Peck, and G Geoffrey Vining. 2015. *Introduction to linear regression analysis*. John Wiley & Sons.

Montillo, Albert, and Haibin Ling. 2009. Age regression from faces using random forests. In *Icip*, 2465–2468.

Mostow, G.D. 1973. *Strong rigidity of locally symmetric spaces*. 78, Princeton University Press.

Mukhopadhyay, Saurabh, and Alan E Gelfand. 1997. Dirichlet process mixed generalized linear models. *JASA* 92(438):633–639.

Mumford, David. 1994. Elastica and computer vision. In *Algebraic geometry and its applications*, 491–506. Springer.

Myronenko, Andriy, and Xubo Song. 2010. Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32(12):2262–2275.

Nagaraj, Sriram, Chinmay Hegde, Aswin C Sankaranarayanan, and Richard G Baraniuk. 2014. Optical flow-based transport on image manifolds. *Applied and Computational Harmonic Analysis* 36(2):280–301.

Ncube, Sentibaleng, Qian Xie, and Anuj Srivastava. 2012. A geometric analysis of ODFs as oriented surfaces for interpolation, averaging and denoising in HARDI data. In *Ieee workshop on mathematical methods in biomedical image analysis*, 1–6.

Neal, R. 2011. MCMC using Hamiltonian dynamics. *Handbook of MCMC* 113–162.

Neal, Radford M. 2000. Markov chain sampling methods for Dirichlet process mixture models. *Journal of computational and graphical statistics* 9(2):249–265.

Nichols, Thomas, and Satoru Hayasaka. 2003. Controlling the familywise error rate in functional neuroimaging: a comparative review. *Statistical methods in medical research* 12(5):419–446.

Niethammer, Marc, Yang Huang, and François-Xavier Vialard. 2011. Geodesic regression for image time-series. In *International conference*

on medical image computing and computer-assisted intervention, 655–662. Springer.

Nocedal, Jorge, and Stephen J Wright. 2006a. *Least-squares problems*. Springer.

———. 2006b. *Numerical optimization 2nd*. Springer.

Özarslan, Evren, and Thomas H Mareci. 2003. Generalized diffusion tensor imaging and analytical relationships between diffusion tensor imaging and high angular resolution diffusion imaging. *Magnetic resonance in Medicine* 50(5):955–965.

Papadopoulos, Athanase. 2005. *Metric spaces, convexity and nonpositive curvature*, vol. 6. European Mathematical Society.

Park, FC, and Bahram Ravani. 1995. Bezier curves on riemannian manifolds and lie groups with kinematics applications. *Journal of Mechanical Design* 117(1):36–40.

Pennec, Xavier. 2006. Intrinsic statistics on riemannian manifolds: Basic tools for geometric measurements. *Journal of Mathematical Imaging and Vision* 25(1):127–154.

Perneger, Thomas V. 1998. What's wrong with bonferroni adjustments. *British Medical Journal* 316(7139):1236.

Petersen, K. B., and M. S. Pedersen. 2012a. The matrix cookbook.

Petersen, Kaare Brandt, and Michael Syskind Pedersen. 2012b. The matrix cookbook (version november 15, 2012).

Petz, Dénes. 2005. Means of positive matrices: Geometry and a conjecture. In *Annales mathematicae et informaticae*, vol. 32, 129–139.

Pflaum, Markus. 2001. *Analytic and geometric study of stratified spaces: contributions to analytic and geometric aspects*. 1768, Springer Science & Business Media.

Pinheiro, José C, and Edward C Chao. 2012. Efficient laplacian and adaptive gaussian quadrature algorithms for multilevel generalized linear mixed models. *Journal of Computational and Graphical Statistics*.

Qiu, Huitong, Fang Han, Han Liu, and Brian Caffo. 2016. Joint estimation of multiple graphical models from high dimensional time series. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 78(2):487–504.

Quirk, Chris, Arul Menezes, and Colin Cherry. 2005. Dependency treelet translation: Syntactically informed phrasal smt. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, 271–279. Association for Computational Linguistics.

Rapcsák, T. 1991. Geodesic convexity in nonlinear optimization. *Journal of Optimization Theory and Applications* 69(1):169–183.

Riddle, William R, Rui Li, J Michael Fitzpatrick, Susan C DonLevy, Benoit M Dawant, and Ronald R Price. 2004. Characterizing changes in mr images with color-coded jacobians. *Magnetic resonance imaging* 22(6): 769–777.

Said, S., L. Bombrun, Y. Berthoumieu, and J. H. Manton. 2017. Riemannian Gaussian distributions on the space of symmetric positive definite matrices. *IEEE Transactions on Information Theory* 63(4):2153–2170.

Said, Salem, Nicolas Courty, Nicolas Le Bihan, Stephen J Sangwine, et al. 2007. Exact principal geodesic analysis for data on SO(3). In *Proceedings of the 15th european signal processing conference*, 1700–1705.

Salehian, Hesamoddin, Rudrasis Chakraborty, Edward Ofori, David Vaillancourt, and Baba C Vemuri. 2015. An efficient recursive estimator of the fréchet mean on a hypersphere with applications to medical image analysis. *Mathematical Foundations of Computational Anatomy*.

Schiratti, Jean-Baptiste, Stéphanie Allassonniere, Olivier Colliot, and Stanley Durrleman. 2015. Learning spatiotemporal trajectories from manifold-valued longitudinal data. In *Advances in neural information processing systems*, 2404–2412.

Schölkopf, Bernhard, and Alexander J Smola. 2002. Learning with kernels: support vector machines, regularization, optimization, and beyond.

Schwartzman, Armin. 2006. Random ellipsoids and false discovery rates: Statistics for diffusion tensor imaging data. Ph.D. thesis, Stanford University.

Setsompop, Kawin, R Kimmlingen, E Eberlein, Thomas Witzel, Julien Cohen-Adad, Jennifer A McNab, Boris Keil, M Dylan Tisdall, P Hoecht, P Dietz, et al. 2013. Pushing the limits of in vivo diffusion MRI for the Human Connectome Project. *Neuroimage* 80:220–233.

Shaham, Uri, Alexander Cloninger, and Ronald R Coifman. 2016. Provable approximation properties for deep neural networks. *Applied and Computational Harmonic Analysis*.

Shahbaba, B., and R. Neal. 2009. Nonlinear models using Dirichlet process mixtures. *JMLR* 10:1829–1850.

Singh, Nikhil, Jacob Hinkle, et al. 2013. A hierarchical geodesic model for diffeomorphic longitudinal shape analysis. In *Ipmi*, 560–571.

Smith, Steven T. 1994. Optimization techniques on riemannian manifolds. *Fields institute communications* 3(3):113–135.

Sommer, Stefan. 2013. Horizontal dimensionality reduction and iterated frame bundle development. In *Geometric science of information*, 76–83. Springer.

Sommer, Stefan, François Lauze, and Mads Nielsen. 2014a. Optimization over geodesics for exact principal geodesic analysis. *Advances in Computational Mathematics* 40(2):283–313.

———. 2014b. Optimization over geodesics for exact principal geodesic analysis. *Advances in Computational Mathematics* 1–31.

Spivak, Michael. 1981. comprehensive introduction to differential geometry. vol. iv.[a].

Srivastava, Anuj, Ian Jermyn, and Shantanu Joshi. 2007. Riemannian analysis of probability density functions with applications in vision 1–8.

Stejskal, EO, and JE Tanner. 1965. Spin diffusion measurements: spin echoes in the presence of a time-dependent field gradient. *The journal of chemical physics* 42(1):288–292.

Straub, Julian, Jason Chang, Oren Freifeld, and John Fisher III. 2015. A dirichlet process mixture model for spherical data. In *Artificial intelligence and statistics*, 930–938.

Tang, Ming-Xin, Yaakov Stern, et al. 1998. The apoe 4 allele and the risk of alzheimer disease among african americans, whites, and hispanics. *Jama* 279(10):751–755.

Terras, Audrey. 2012. *Harmonic analysis on symmetric spaces and applications ii*. Springer Science & Business Media.

Tsay, Ruey S. 2005. *Analysis of financial time series*, vol. 543. John Wiley & Sons.

Tuch, David S, RM Weisskoff, JW Belliveau, and VJ Wedeen. 1999. High angular resolution diffusion imaging of the human brain. In *Proceedings of the 7th annual meeting of ismrm, philadelphia*, vol. 321.

Tuch, D.S. 2004. Q-ball imaging. *Magn. Resn. Med.* 52(6):1358–1372.

Turaga, Pavan, Ashok Veeraraghavan, Anurag Srivastava, and Rama Chellappa. 2011. Statistical computations on grassmann and stiefel manifolds for image and video-based recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33(11):2273–2286.

Tustison, Nicholas J, and Brian B Avants. 2013. Explicit b-spline regularization in diffeomorphic image registration. *Front. Neuroinform* 7(39).

Tuzel, Oncel, Fatih Porikli, and Peter Meer. 2006a. Region covariance: A fast descriptor for detection and classification. In *Computer vision–eccv 2006*, 589–600. Springer.

———. 2006b. Region covariance: A fast descriptor for detection and classification. In *Eccv*, 589–600.

———. 2008. Pedestrian detection via classification on Riemannian manifolds. *IEEE transactions on pattern analysis and machine intelligence* 30(10):1713–1727.

Udriste, Constantin. 1994. *Convex functions and optimization methods on riemannian manifolds*, vol. 297. Springer Science & Business Media.

Vandereycken, Bart. 2013. Low-rank matrix completion by riemannian optimization. *SIAM Journal on Optimization* 23(2):1214–1236.

Wackerly, Dennis, William Mendenhall, and Richard Scheaffer. 2007. *Mathematical statistics with applications*. Nelson Education.

Walpole, Ronald E, Raymond H Myers, Sharon L Myers, and E Ye Keying. 2016. *Probability & statistics for engineers & scientists, mystatlab*. Pearson Higher Ed.

Wang, Zhizhou, Baba C Vemuri, Yunmei Chen, and Thomas H Mareci. 2004. A constrained variational principle for direct estimation and smoothing of the diffusion tensor field from complex dwi. *IEEE transactions on Medical Imaging* 23(8):930–939.

Witten, Daniela M, Robert Tibshirani, and Trevor Hastie. 2009. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics* 10(3):515–534.

Wolfinger, Russ, and Michael O'connell. 1993. Generalized linear mixed models a pseudo-likelihood approach. *Journal of statistical Computation and Simulation* 48(3-4):233–243.

Xie, Yuchen, Baba Vemuri, and Jeffrey Ho. 2010. Statistical analysis of tensor fields. *MICCAI* 682–689.

Xu, Jia, Vamsi K Ithapu, Lopamudra Mukherjee, James M Rehg, and Vikas Singh. 2013. Gosus: Grassmannian online subspace updates with structured-sparsity. In *Proceedings of the ieee international conference on computer vision*, 3376–3383.

Yang, Yaguang. 2007. Globally convergent optimization algorithms on riemannian manifolds: Uniform framework for unconstrained and constrained optimization. *Journal of Optimization Theory and Applications* 132(2):245.

Yoon, Uicheul, Vladimir S Fonov, Daniel Perusse, Alan C Evans, Brain Development Cooperative Group, et al. 2009. The effect of template

choice on morphometric analysis of pediatric brain data. *Neuroimage* 45(3):769–777.

Yu, Guoshen, and Guillermo Sapiro. 2011. Statistical compressed sensing of gaussian mixture models. *IEEE Transactions on Signal Processing* 59(12): 5842–5858.

Zacur, Ernesto, Matias Bossa, and Salvador Olmos. 2014. Multivariate tensor-based morphometry with a right-invariant Riemannian distance on GL+ (n). *Journal of mathematical imaging and vision* 50(1-2):18–31.

Zhang, H., P.A. Yushkevich, D.C. Alexander, and J.C. Gee. 2006a. Deformable registration of diffusion tensor MR images with explicit orientation optimization. *Med. Img. Analysis* 10:764–785.

Zhang, Hui, Paul A Yushkevich, Daniel C Alexander, and James C Gee. 2006b. Deformable registration of diffusion tensor MR images with explicit orientation optimization. *Medical image analysis* 10(5):764–785.

Zhang, Miaomiao, and P Thomas Fletcher. 2013. Probabilistic principal geodesic analysis. In *Advances in neural information processing systems (nips)*, 1178–1186.

Zhang, Zhihua, Dakan Wang, Guang Dai, and Michael I Jordan. 2014. Matrix-variate Dirichlet process priors with applications. *Bayesian Analysis* 9:259–289.

Zheng, Ligang, Hyunwoo J Kim, Nagesh Adluru, Michael A Newton, and Vikas Singh. 2017. Riemannian variance filtering: An independent filtering scheme for statistical tests on manifold-valued data. In *Computer vision and pattern recognition workshops (cvprw), 2017 ieee conference on*, 699–708. IEEE.

Zhou, Hao, Yilin Zhang, Vamsi Ithapu, Sterling Johnson, Grace Wahba, and Vikas Singh. 2017. When can multi-site datasets be pooled for regression? hypothesis tests, l2-consistency and neuroscience applications.

Zhou, Shuheng, John Lafferty, and Larry Wasserman. 2010. Time varying undirected graphs. *Machine Learning* 80(2):295–319.