Estimation statistique dans les variétés Riemanniennes: implémentation et application à l'étude des déformations cardiaques

Nicolas Guigui

UNIVERSITÉ CÔTE D'AZUR

ÉCOLE DOCTORALE
SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

THÈSE DE DOCTORAT

Estimation statistique dans les variétés Riemanniennes: implémentation et application à l'étude des déformations cardiaques

Nicolas A. S. Guigui

Epione (Inria), ERC G-Statistics

Présentée le 18 novembre 2021 en vue de l'obtention du grade de docteur en informatique d'Université Côte d'Azur

Devant le jury, composé de :

Directeur :	Xavier Pennec	-	Epione (Inria)
Rapporteurs:	Joan Glaunès	-	Université Paris Descartes
	Sarang Joshi	-	University of Utah
Examinateurs:	Nicolas Duchateau	-	Insa Lyon
	Stanley Durrleman	-	Brain and Spine institute and Inria
	Stefan Sommer	-	University of Copenhaguen
	Francois-Xavier Vialard	-	Université Gustave Eiffel
Invitée:	Nina Miolane	-	University of California Santa Barbara

Inria Epione e-patient / e-medicine erc G-Statistics Geometric Statistics

Estimation statistique dans les variétés Riemanniennes: implémentation et application à l'étude des déformations cardiaques

Computational methods for statistical estimation on Riemannian manifolds and application to the study of cardiac deformations

Jury:

Rapporteurs:
Joan Glaunès, Maître de Conférences, Université Paris Descartes
Sarang Joshi, Professeur, University of Utah

Examinateurs:
Nicolas Duchateau, Maître de Conférences, Insa Lyon
Stanley Durrleman, Directeur de Recherche, Institut du Cerveau et de la Moelle
Xavier Pennec, Directeur de Recherche, Inria
Stefan Sommer, Professeur, University of Copenhaguen
Francois-Xavier Vialard, Professeur, Université Gustave Eiffel

Invitée:
Nina Miolane, Assistant Professor, University of California Santa Barbara

Estimation statistique dans les variétés Riemanniennes: implémentation et application à l'étude des déformations cardiaques

L'étude des formes anatomiques et de leur mouvement est au cœur des préoccupations dans de nombreuses spécialités de la médecine. En cardiologie, des pathologies telles que l'arythmie ou l'hypertension pulmonaire entraînent des anomalies, telles qu'une contraction plus lente ou le grossissement du myocarde, et dont la caractérisation en forme, et en déformation permet d'évaluer la gravité de la maladie ou l'impact d'un traitement. Cette caractérisation nécessite un cadre mathématique prenant en compte les non-linéarités et les invariances propres aux formes, et à leurs déformations, et qui permette de décorréler ces deux facteurs de variabilité.

Les variétés Riemanniennes offrent un cadre naturel qui répond à ces exigences, et le transport parallèle est communément utilisé comme stratégie de normalisation des déformations du cortex cérébral, permettant d'abstraire le mouvement de la forme. Plus généralement, de nombreux types de données sont modélisés dans des variétés Riemanniennes, mais peu d'outils numériques sont disponibles. L'objectif de cette thèse est d'une part de développer et mettre à disposition des outils de calcul robustes et modulables permettant de faire des statistiques dans des variétés Riemanniennes, et d'autre part d'appliquer ces concepts dans le but de modéliser les déformations du ventricule cardiaque droit lors de surcharge en volume et en pression.

Dans ce but, nous fournissons d'abord une exposition des notions de la géométrie Riemannienne, avec de nombreux exemples et en adoptant le point de vue des applications. Nous expliquons comment ces concepts sont implémentés dans le projet Python geomstats qui rassemble l'implémentation de la géométrie et des outils d'apprentissage statistique sur les variétés, et dont cette thèse constitue une des contributions majeures.

Une attention particulière est portée à l'implémentation des métriques invariantes sur les groupes de Lie pour lesquelles une nouvelle formulation du transport parallèle est donnée, ainsi qu'à une implémentation générique des métriques quotients, avec applications aux espaces de formes de Kendall. Cette nouvelle implémentation a permis des gains sur l'état de l'art en précision et en vitesse de convergence.

Nous nous concentrons ensuite sur le calcul du transport parallèle par les méthodes dites à échelle qui consistent à construire des petits parallélogrammes géodésiques et sont utilisées depuis les années 2000 dans des applications allant de l'imagerie médicale à la robotique. Cependant, on ne trouve pas dans la littérature d'analyse claire de ses propriétés de convergence. Nous comblons ces lacunes en fournissant un développement limité de la construction en fonction de la courbure de l'espace. Cela permet ensuite de montrer que le schéma répété converge et d'obtenir une vitesse de convergence quadratique. Des expériences numériques effectuées sur différentes variétés ont permis de valider très précisément ce résultat.

Ces résultats sont enfin appliqués au problème d'étude des déformations cardiaques. Le cadre des Larges Deformations Diffeomorphic Metric Mappings et le

transport parallèle sont utilisés pour estimer et réorienter les déformations du ventricule droit. Nous mettons cependant en évidence des effets indésirables dus au changements important de volumes entre les patients et le groupe témoin. Nous proposons une stratégie de normalisation pour l'amplitude des déformations par une mise à l'échelle de facon à préserver le changement de volume relatif au cours du temps. Cette approche est efficace et le coefficient de mise à l'echelle est significativement associé au volume initial. Les déformations normalisées sont ensuite résumées par une régression à splines, dont les paramètres révèlent des différences significatives entre chaque pathologie et le groupe témoin, marquant l'effet des maladies sur la contraction du ventricule.

Mot clés: Géométrie Différentielle, Anatomie Computationnelle, Statistiques Géométriques

Computational methods for statistical estimation on Riemannian manifolds and application to the study of cardiac deformations

The study of anatomical shapes and deformations is of major interest in cardiology where diseases such as arrhythmia or pulmonary hypertension cause abnormalities in the way the heart contracts or of its shape. The characterization of these abnormalities allows to evaluate the gravity of the disease or the impact of a treatment. A mathematical framework that accounts for the nonlinearity and invariance properties of shape spaces and shape deformations is needed to obtain such characterizations, perform population statistics and disentangle the two factors of variability: shape and deformation.

Riemannian manifolds are a natural setting that fill in these desiderata, and parallel transport is commonly used to normalize the deformations of the cerebral cortex, allowing to decorrelate its evolution from the initial shape. More generally, data lying on Riemannian manifolds are ubiquitous, yet, the development of computational tools from the basic theory of Riemannian geometry is tedious. One of the aims of this thesis is to provide the computational tools to perform statistics and machine learning on Riemannian manifolds to the community. The work presented here forms one of the main contributions to the open-source project geomstats, that consists in a Python package providing efficient implementations of the concepts of Riemannian geometry, both for mathematicians and for applied scientists for whom most of the difficulties are hidden under high-level functions.

An exposition of Riemannian geometry is first given, with illustrations and examples. We adopt a computational point of view to then demonstrate how these concepts are implemented in geomstats, explaining the choices made during its development. The use of geomstats to perform statistics on manifolds is briefly exemplified.

Particular attention is given to the implementation of invariant metrics on Lie groups for which a new formulation of the parallel transport is given, and on a general implementation of quotient metrics with applications to Kendall shape spaces. Our new implementation of parallel transport in these spaces outperforms the state of the art in speed and precision.

We then focus on the study of ladder methods that consist in reproducing the construction of small geodesic parallelograms to approximate parallel transport on manifolds. These are now used in applications such as robotics, medical imaging or computer vision. However, the literature lacks a clear analysis of their convergence performance. We give Taylor approximations of the constructions with respect to the curvature of the space and prove that these methods can be iterated to converge with quadratic speed, even when geodesics are approximated by numerical schemes. This rate is precisely observed on common manifolds. The special Euclidean group with an anisotropic invariant metric is of particular interest to illustrate the effect of the covariant derivative of curvature.

Finally, we apply these results to the problem of studying cardiac deformations. We use the Large Deformation Diffeomorphic Metric Mapping framework and par-

allel transport to reorient deformations of the right-ventricle. However, we find an undesirable effects of the large volume differences that occur between patients and the control group. We propose a normalization procedure for the amplitude of the deformation by a scaling step that conserves the relative volume change. This approach proves effective, and we exhibit a significant relation between the volume changes and the scaling parameter. The normalized deformations are then studied by summarizing each deformation with a spline regression, and performing statistics on the parameters of the splines that reveal significant differences between each disease and the control group, reflecting the dynamics of each disease.

Keywords: Differential Geometry, Computational Anatomy, Geometric Statistics

Acknowledgements

First and foremost I would like to address my sincere gratitude to Xavier Pennec for supervising my PhD with care and enthusiasm during the past three years. Granting me with trust, freedom, and yet always the right level of scientific expectations, he allowed me to thrive during this rich period. I am very grateful for the research environment he has built in the past years, from our growing G-Stats group and our contribution to the great Epione team to the inspiring world-wide community on Geometric Statistics, that we met in various places to exchange ideas in always friendly atmospheres. Many collaborations stemmed from these events and directly contributed to this work.

The most significant of these is the geomstats project that gained much momentum thanks to Xavier's support and long term vision, and that motivated a great part of this thesis. I would like to warmly thank Nina Miolane for starting this project and for her motivation and dedication to its improvement and dissemination. Not only did she set up a great infrastructure with high quality standards, but she is also able to welcome and guide all the new contributors to make this project truly open-source and collaborative.

I would like to thank Joan Glaunes and Sarang Joshi who accepted to review this manuscript, and Stanley Durrleman, Nicolas Duchateau, François-Xavier Vialard and Stefan Sommer for being part of my defense committee. I thank all the colleagues that interacted with me during this PhD, the contributors to geomstats, but also Emmanuel Chevallier for our discussions on symmetric spaces and related topics. I would also like to thank Pamela Moceri and Nicolas Duchateau for allowing me to work on cardiac data, and for the interactions we had related to that project.

More broadly, I want to thank my colleagues of the Epione team, and especially Maxime Sermesant for the discussions we had on the cardiac modeling project, and for the good atmosphere he brings, along with Marco Lorenzi and Nicolas Cedilnik, every time music instruments end up in their hands. I also thank all my fellow PhD students who contributed to make this lab a lively and inspiring place, and with whom I shared many experiences.

I am especially thankful to Yann Thanwerdas for his companionship during this PhD, for all the ideas exchanged during our daily discussions in each other's office and our diners in Antibes, but also for his generosity and hospitality during my nomadic year.

I would like to thank all my friends, *les frérot(e)s*, *les ami(e)s de toujours* and all the others who surrounded me and made these years full of travels, adventures and good times. My warm gratitude also goes to the Amram family, whose generosity allowed me to spend part of this time in excellent conditions in JLP.

I am grateful to my outstanding family, my parents, grand-parents, brother and sisters, for their joyful presence, their trust and their unwavering support.

Lastly, I want to thank Elsie for the time spent together during the writing of this manuscript, for her enthusiasm, her unconditional support and all the discoveries made together.

Funding

This work was funded by the ERC grant Nr. 786854 G-Statistics from the European Research Council under the European Union's Horizon 2020 research and innovation program. It was also supported by the French government through the 3IA Côte d'Azur Investments ANR-19-P3IA-0002 managed by the National Research Agency. The author is grateful to the OPAL infrastructure from Université Côte d'Azur for providing computing resources and support.

License

Contents

List of Figures

Chapter 1

Introduction

1.1 Data driven research in medical imaging

Motivation The study of anatomical shapes and deformations is of major interest in many specialties of medicine. In cardiology, pathologies such as arrhythmia or pulmonary hypertension may cause abnormalities of the heart's shape at a reference time of the cycle or in the way it contracts. The characterization of these abnormalities, either of shape or of deformation, allows to evaluate the gravity of the disease or the impact of a treatment. This thesis is motivated by the study of the cardiac right-ventricle under volume or pressure overload caused by three conditions: pulmonary hypertension, tetralogy of Fallot and atrial septal defect. In each case, the shape of the ventricle is modified, often with increased volume. The contraction is altered, with less amplitude and asynchrony. A mathematical framework that accounts for the non-linearities and invariance properties of shape spaces, and shape deformations is needed to characterise such alterations, and to be able to quantify the two factors of variability: shape and deformation.

With the advent of digital technology, our practice of scientific research has quickly evolved and is now at the intersection of two paradigms. On the one hand, the traditional model-based approach remains prominent, where scientists make hypotheses, assumptions, and design experiments to carefully evaluate them. On the other hand, data-driven approaches rely on optimization schemes to find the models that best suit empirical data. The latter, coined *Data Science*, *Machine Learning* or *Artificial Intelligence*, are now at the heart of strategic research investments.

This evolution is salient in medicine, where routinely acquired clinical data are now part of research projects to improve healthcare and to help gain novel medical knowledge. Indeed, the acquisition of medical data has greatly increased at all scales, from omics data thanks to high-throughput sequencing technologies, to images that are acquired by ultra-sound, scanners or magnetic resonance. Electronic health records complement this landscape with clinical data and experts' annotations. Together with the success of machine learning approaches, these data are now the raw material to produce new medical knowledge, by helping to discover biomarkers of a disease, to find new disease subtypes or to better identify risk factors. Moreover, data collected over the lifetime are key to understand disease onset and progression, to evaluate treatment outcomes and in the long term to provide personalized treatments. Finally machine learning algorithms can ease the burden of healthcare as they are the main building blocks of new automated diagnosis tools and interventional technologies. We shortly review the main ingredients to this rev-

olution and the issues that must be addressed to fully benefit from it in medicine.

Machine learning consists in minimizing a cost function with respect to a set of parameters. The cost is computed on historical data and the model with the optimal parameter can then be applied to new unseen inputs. In a supervised learning setting, pairs of input data $(x_i, y_i)_{i \leq n}$ are given, where x_i may be a vector of variables or *features*, or a more complex object such as a signal, a graph, an image, a shape, etc... y_i maybe a real number (in a regression task) or an integer (in a classification task). A model then consists in a function $\mathcal{F}$ with parameters θ that is used to map x_i to y_i, such that the average error is minimal:

$$\min_{\theta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\mathcal{F}(x_i, \theta), y_i) + \mathcal{R}(\theta), \tag{1.1}$$

where $\mathcal{L}$ is a *loss* function that measures the discrepancy between the model prediction and the target value, and $\mathcal{R}$ is a *regularization* penalty for unlikely values of θ. In medical imaging, many problems can be formulated this way, ranging from image registration and segmentation, to automated diagnosis.

The function $\mathcal{F}$ may range from elementary linear combinations, to compositions of complex nonlinear operations. Most of today's research on machine learning consists in defining adequate functions $\mathcal{F}$, coping with large-scale experiments - where $n \sim 10^6$ - or high-dimensional settings - $p \gg n$, and addressing methodological questions regarding the validation of the model, and the reproducibility of the experiments.

Some of these questions were tackled by the development of open-source Python packages such as `scikit-learn` (Pedregosa, Varoquaux, et al. 2011, paper cited more than 43k times as of 2021), with an estimated 600 000 monthly users[1], allowing to quickly test different models on various types of data, and allowing the democratization of these tools.

This evolution quickened with the striking results of deep learning (Krizhevsky, Sutskever, et al. 2012) which consists in composing elementary operations (called *neurons*) to compute increasingly higher-level features (LeCun, Bengio, et al. 2015), and the development of dedicated coding frameworks such as `TensorFlow` (Abadi, Agarwal, et al. 2015) and `PyTorch` (Paszke, Gross, et al. 2019).

Unfortunately, the main ingredient, the data, is more scarcely available in fields such as biomedical research, limiting the development and deployment of these tools to only a few applications, such as for example breast cancer detection, for which the Dream digital mammography challenge gathered 640 000 images from 86 000 subjects[2]. However, while deep networks in computer vision are trained on datasets such as ImageNet[3] that contains more than 14M images as of 2021, for many medical challenges the most widely used datasets are for example the Alzheimer's Disease

[1] Interview of core maintainer of scikit-learn Olivier Grisel by Claire Carroll, 2018, Dataiku blog.
[2] https://www.synapse.org/#!Synapse:syn4224222/wiki/401743
[3] https://www.image-net.org/index.php

Neuroimaging Initiative (ADNI, cited more than 80k times)[4], the Autism Brain Imaging Data Exchange (ABIDE)[5] or the Brain Tumor Segmentation (BraTS)[6] that count at most 1000 images each. The acquisition and curation of larger datasets is an ongoing process. In particular the creation of the *UK Biobank*[7] was started in 2006 and includes half a million UK participants. It gathers various data modalities such as omics, imaging and clinical covariates and is expected to become one of the largest available datasets in medical research.

However, even when small datasets are considered, heterogeneity is present as subjects may come from different imaging centers or different acquisition sequences are used. Moreover, these datasets are usually biased towards certain ethnicities, age population, comorbidity, etc. These aspects introduce additional challenges for researchers.

Moreover, deep networks have been designed for images and natural language processing at the detriment of more structured data types such as anatomical shapes that are represented as point clouds or meshes instead of dense arrays of color intensity. Indeed, their performance is well-established for the analysis of textures and detection of patterns at various scales, but these are less relevant for shape analysis over time, where we are interested in higher-level information such as strain or curvature. Although the emerging field of geometric deep learning (Bronstein, Bruna, et al. 2021) now allows to extract features from graphs and meshes, problems raised by the lower quality and the noise present in medical images are neglected. The gap between medical images and the use of such methods thus remains to be filled.

More generally, a prolific research avenue to deal with the lack of data remains to design tailored algorithms that account for the properties of the data.

Data on manifolds A quite general assumption is to consider data (or the parameters θ) as points of a manifold, i.e., a space that is defined by a set of constrains or invariance properties, and that only locally resembles usual Euclidean spaces. Indeed, as a first example, one may think of the circle or the unit sphere in $\mathbb{R}^3$ to model angles and solid angles. This field is traditionally known as directional statistics (Mardia and Jupp 2009), and useful in e.g. protein structure prediction in Boomsma, Mardia, et al. 2008, or in Sampath, Lewis, et al. 2015 to study the orientation of the femur and tibia in relationship with osteoarthritis of the knee, or for mining, in Peel, Whiten, et al. 2001 to study the orientations of parallel fracture planes in rocks.

Hyperbolic geometry, on the other hand, is used to model data with hierarchical structures, i.e., as embedding space for trees (Gromov 1987) and more generally graphs (Krioukov, Papadopoulos, et al. 2010; Nickel and Kiela 2017) or texts (Dhingra, Shallue, et al. 2018).

[4] http://adni.loni.usc.edu/
[5] http://fcon_1000.projects.nitrc.org/indi/abide/
[6] https://www.med.upenn.edu/cbica/brats2020/
[7] https://www.ukbiobank.ac.uk/

Another widespread research avenue is to leverage the geometric structure of the space of Symmetric Positive Definite (SPD) matrices to perform statistics and learning tasks on covariance and correlation matrices. This allowed major improvements over traditional methods designed for vector spaces, with to cite only a few of them, the works of Pennec, Fillard, et al. 2006 in diffusion tensor imaging, of Varoquaux, Baronnet, et al. 2010 in the study functional connectivity in the brain from resting state functional magnetic resonance images (fMRI), of Barachant, Bonnet, et al. 2012 in brain-computer interface classification or of Cabanes, Barbaresco, et al. 2019 for radar signal processing.

Furthermore, Lie groups of transformations such as rotations ($SO(3)$) or rigid-body transformation ($SE(3)$) are used in robotics, tracking (Hauberg, Lauze, et al. 2013), navigation systems (Barrau and Bonnabel 2017) and computer vision, for example, to estimate a camera pose (A. Kendall and Cipolla 2017; Hou, Miolane, et al. 2018).

The manifold setting additionally allows modelling more abstract concepts. A first example is that of a shape in the plane or in space. It may be described by a set of landmark points, but this does not encode any structure. Yet, it is intuitive to think that shapes that differ only by a rotation, translation or even a scaling transformation are the same. The invariance to these transformations defines a *quotient manifold*, usually referred to as Kendall shape space (D. G. Kendall 1984), and corresponding Riemannian metric.

This generalizes to a continuous curve, arising for example from the contour of an organ, and available at discrete points sampled along the curve, according to a segmentation process. Although a discrete curve boils down to a list of coordinates, that would be considered independent of one another by traditional statistics and machine learning methods, considering the infinite dimensional manifold of continuous curves allows defining Riemannian metrics that are invariant to e.g. rotation, translation, scaling or reparametrization, i.e. account for the discrete sampling process (Srivastava, Klassen, et al. 2011, and references therein).

Finally, shapes and images can also be modelled in an infinite dimensional manifold, and their variability modelled through a set of admissible transformations. The group of diffeomorphisms and more convenient approximations, are often considered and metrics that measure the amplitude of the deformation translate into metrics on shapes (Younes 2019). It allows to interpolate and extrapolate shape trajectories in a coherent way, avoiding folding, tearing or shearing of the surfaces, which is not the case as soon as large deformations occur if points are considered independent. The notion of distance defined by Riemannian geometric tools is then used to define loss functions $\mathcal{L}$ for (1.1) and to generalize common statistical distributions and learning algorithms to these datasets.

Geometric statistics encompass the generalization of statistics and machine learning methods to manifold-valued data. Firstly, in many cases this process is necessary for mathematical coherence. Indeed, manifolds are usually non-linear,

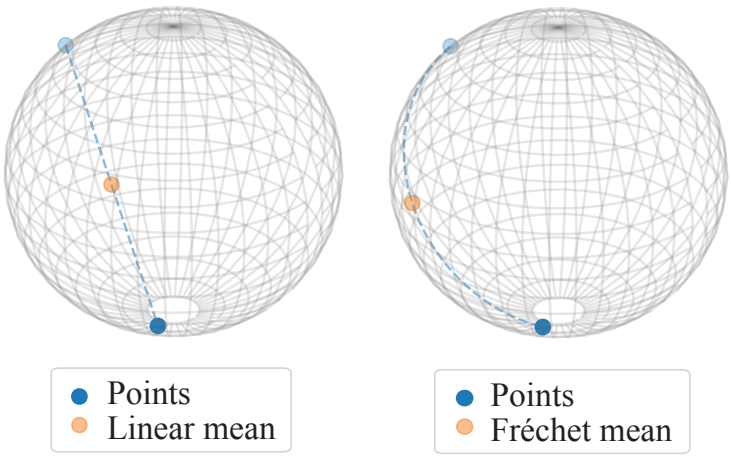


Figure 1.1: Illustration of non-linearity of the sphere: the mean of two points does not lie on the sphere. Geometric statistic methods cope with this problem by following geodesics. Figure reproduced from Miolane, Guigui, Zaatiti, et al. 2020.

meaning that the addition and subtraction operations are not defined, nor a unique coordinate system and corresponding inner product to measure distances between points. Consequently, the usual mean of two points may not lie on the manifold, hence not be a realistic summary of the data. This is the case for example of the sphere as depicted on Figure 1.1, or of anatomical shapes, whose topology is not preserved by linear operations. A more subtle example occurs in diffusion tensor imaging, where interpolation schemes are used to filter and reconstruct images of the brain, where at each voxel, an SPD matrix is given. It was shown that linear interpolation had a physically non-realistic effect because of the swelling of the eigenvalues, while a geometry-aware interpolation scheme did not show this effect (Pennec, Fillard, et al. 2006).

Secondly, the design of geometric statistic methods is necessary for statistical consistency as the curvature of the manifold introduces deviations from the usual statistical laws. This is the case for the law of large numbers and the central limit theorem, for which one can show that the speed of convergence of the usual estimators of the mean are faster or slower than in the Euclidean case (Hotz and Huckemann 2015; Pennec 2019). Correction terms must be implemented to define statistical tests with the desired level of confidence, and may change the results of an analysis, as in the study of the daily wind direction in Hundrieser, Eltzner, et al. 2021, where quantile based tests lead to reject the null hypothesis, while bootstrap based tests with correction do not lead to rejection.

Finally, geometry also serves as a prior when only small datasets are available, and e.g. Brooks, Schwander, et al. 2019 demonstrated faster convergence with respect to the number of samples used for training thanks to a batch normalization process relying on the Fréchet mean and parallel transport.

The Riemannian distance, volume measure and parallel transport are thus key

to generalize fundamental statistical operations. The first tool is to compute the mean of a set of points $(x_1, \dots, x_n)$. In the Euclidean case, it coincides with the minimizer of the sum-of-squared distances

$$\bar{x} = \arg\min_{x \in M} \sum_{i=1}^{n} d^2(x, x_i). \tag{1.2}$$

This characterization serves as a definition of the *Fréchet mean* in metric spaces, and in Riemannian manifolds in particular. However, the minimizer may not be unique or even exist (Pennec, Sommer, et al. 2020, Chapter 2). Furthermore there is no closed-form solution, and iterative algorithms have been derived to solve (1.2).

A widespread tool is then to lift the data points to the tangent space at the mean, by representing each points by the tangent vector to the geodesic i.e. the shortest-distance curve that joins the mean to that point. This corresponds to a first-order approximation of the manifold around the mean and has proven effective, provided the dispersion of the data is not too large. Parallel transport then allows moving different datasets to the same reference points, as a domain adaptation or transfer learning process. Going further, regression models are generalized by identifying geodesics with straight lines, and seeking the best geodesic to approximate given data.

These operations are defined by the Riemannian metric, that must be defined as an an additional structure to the differential structure of the manifold. From different metrics result different distance functions, geodesics and parallel transport maps, hence different statistical tools. Although for a few manifolds there exists a canonical metric, in most cases there is no such privileged choice and many metrics have been proposed in the literature. This raises the important question of choosing the metric for a given application. While some intuitions, invariance properties or constrains may guide this choice, trials and errors are often necessary to study the impact of the change of metric and choose the most appropriate. A modular framework is thus necessary to rapidly test different metrics.

However, this comes at a computational cost that may be prohibitive. Indeed, although for most of the finite-dimensional Riemannian manifolds mentioned above geodesics are available in closed form, they often require heavy matrix operations such as eigenvalue or singular value decompositions, matrix exponential or logarithms, that turn out intractable or not numerically stable when the dimension increases. More generally, when geodesics are not available in closed-form, numerical methods must be used to approximate the solution of differential equations, and optimization schemes to solve the boundary-value problems. This becomes very costly when these computations are only one element to compute the loss of a learning model. Furthermore, in order to derive second-order optimization schemes, the curvature of the space is essential, but it is a complex tool described as a $(1, 3)$-tensor at each point of the manifold, that is usually hard to compute.

These computations are now possible by taking advantage of the fast development of efficient numerical tools for deep learning, in particular of automatic

differentiation and optimization routines. Indeed, since the works of Kühnel, Sommer, and Arnaudon 2019; Bône, Louis, et al. 2018, automatic differentiation is used to compute Christoffel symbols and Jacobi fields, and allows to approximate geodesics, parallel transport and to differentiate geometric loss functions, opening a wide range of possible improvements to perform geometric statistics and machine learning. These tools are still at the level of proof-of-concepts, and one of the aim of this thesis is to contribute to their development and deployment driven by the application to medical images over time with a particular attention to reproducibility.

Reproducibility is fundamental to confirm discoveries, demonstrate their applicability and ensure the reliability of science. However, it is questioned in computer science by either the lack of open access to the original data with which a model was trained or to the code (Redish, Kummerfeld, et al. 2018). Open-source packages and Jupyter notebooks are thus essential tools to share code and enable other researchers to easily reproduce experiments. This is one of the motivations behind the developments of this thesis.

Indeed, as more papers are now submitted with the corresponding code, each use different frameworks and are hard to reproduce in a unified manner. Some Python packages do exist, but most of them fall under one of two categories. The first are the packages that focus on a single application, for instance on optimization: `Pymanopt` (Townsend, Koep, et al. 2016), `Geoopt` (Becigneul and Ganea 2019; Kochurov 2019), `TensorFlow RiemOpt` Smirnov 2021, and `McTorch` (Meghwanshi, Jawanpuria, et al. 2018) or on deep learning such as `PyG` (Fey and Lenssen 2019), where, in this case the geometry is often restricted to graph and mesh spaces. The other are the packages dedicated to a single manifold: `PyRiemann` on SPD matrices (Barachant 2015), `PyQuaternions` on 3D rotations (Wynn 2014), and `PyGeometry` on spheres, tori, 3D rotations and translations (Censi 2012). Lastly, others are not actively maintained anymore: `TheanoGeometry` (Kühnel and Sommer 2017). There is therefore a need for a unified open-source implementation of differential geometry and associated learning algorithms for manifold-valued data.

1.2 Objectives and outline of the manuscript

1.2.1 Objectives

The first aim of this thesis is to bridge the gap between the mathematical theory and the computational tools, by providing the framework to perform statistics and machine learning on Riemannian manifolds to the wider data-science community. The work presented here forms one of the main contributions to the open-source project geomstats, that consists in a Python package providing efficient implementations of the concepts of Riemannian geometry and geometric statistics, both for mathematicians and for applied scientists for whom most of the difficulties are hidden under high-level functions. With this package, we ambition to democratize the

use of geometric statistics in the data science community. For this purpose, all the geometric functions need to be implemented with a high-level interface, and to be modular in the sense that different structures can be combined. We aim at making available as many learning algorithms as possible, for as many geometries as we encounter and in an application-agnostic fashion. Finally the documentation of the package should be readable by engineers with no background in Riemannian geometry, with hands-on tutorials from the basics of Riemannian geometry to the training of learning algorithms on real datasets from popular applications.

Moreover, we identify many geometric structures defined by invariance properties, for example on Lie groups or on quotient spaces, and we investigate their common implementation. This is motivated by applications ranging from robotics to computer vision and medical imaging, with for example the manifolds of correlation matrices, of Kendall shapes, or of open and closed curves. Closed-form solutions are not always available but leveraging numerical approximation schemes, automatic differentiation and optimization, the properties of more geometries become at reach. We aim at implementing the computation of the geodesics, the Riemannian exponential and logarithm maps, parallel transport and curvature of each metric. All these operations should be available with a desired accuracy and differentiable by automatic differentiation software in order to serve as building blocks to learning algorithms. These developments could be highly beneficial to the community, as many results co-exist in the literature but lack unification.

A deeper study of the parallel transport is required as it is a fundamental operation in geometric statistics and in the study of longitudinal data in medical imaging. More generally, it is a geometric tool for transfer learning or domain adaptation, and it is gaining interest in the machine learning community. The second aim of this thesis is thus on the algorithmic point of view, to understand the different methods to compute parallel transport, either by integration of the parallel transport equation, or by approximation schemes such as ladder methods. These methods stem from Schild's ladder, which was introduced in the 70's by Alfred Schild, and consist in reproducing the construction of sufficiently small geodesic parallelograms, leveraging the fact that manifolds resemble Euclidean spaces when considering small sets. They are now used in applications such as robotics, medical imaging or computer vision. And yet, the literature lacks a clear analysis of their convergence performance. We shall prove that these schemes converge and give their speed of convergence with respect to the number of parallelograms used and the curvature of the Riemannian manifold. We also intend to synthesize the analysis of other existing methods under the same framework.

The ultimate goal is to investigate the use of parallel transport to normalize the cardiac cycles across patients, in order to remove the large volume differences between patients, to evaluate the cardiac function and the degree of adaption to remodelling. Although such methodology has been applied to brain data in the past, it has not been investigated in the context of cardiac modelling, where the amplitude of both inter-patients and longitudinal deformations are much higher, raising interesting challenges. The normalized deformations should allow to distinguish be-

tween diseases and the interpretation of the differences will provide new insight into each disease.

1.2.2 Overview

Chapter 2 The goal of the second chapter is on the one hand to give a brief exposition of the concepts of Riemannian Geometry, providing illustrations and examples at each step and adopting a computational point of view, and on the other hand to demonstrate how these concepts are implemented in geomstats, explaining the choices that were made and the conventions chosen. The use of geomstats to perform statistics on manifolds is then briefly exemplified. This work lead to an introductory presentation at the Scipy conference 2020 in Miolane, Guigui, Zaatiti, et al. 2020 and to a publication in the Journal of Machine Learning Research (Miolane, Guigui, Le Brigant, et al. 2020).

Chapter 3 Then, we focus on the implementation of invariant metrics on Lie groups and quotient manifolds. Particular attention is given to the formulation of the parallel transport on Lie groups equipped with an invariant metric, and to giving a general implementation for quotient metrics with applications to Kendall shape spaces, and the manifold of correlation matrices. This results in a new implementation of parallel transport on Kendall shape spaces that outperforms the state of the art in speed and precision. Part of the material presented in this chapter was presented at the conference on Geometric Science of Information (GSI) in Guigui and Pennec 2021a; Guigui, Maignant, et al. 2021. Together with the first chapter, this chapter will be submitted as an extended paper on the introduction to geometric statistics with geomstats and will complement the documentation of the package.

Chapter 4 This chapter is dedicated to the study of the ladder methods to approximate parallel transport on manifolds. A thorough analysis of the elementary construction of the schemes, and their convergence properties are given. Moreover, we focus on the case where geodesics are not available in closed form, and adapt the schemes to this case. We also investigate the link with existing methods such as the Fanning Scheme. All the results are illustrated on common manifolds, and we use the special Euclidean group with an anisotropic metric to exemplify the impact of the curvature on the convergence of the schemes. These results were published in Foundation of Computational Mathematics, in Guigui and Pennec 2021b.

Chapter 5 We apply the previous results to our initial problem: the study of cardiac deformation. We use the Large Deformation Diffeomorphic Metric Mapping (LDMMM) framework to model the shape space as a Riemannian manifold with a smooth action from a subgroup of diffeomorphisms. In this setting, parallel transport is used to reorient deformations of the right-ventricle. However, we exhibit undesirable effects of the large volume differences that occur between patient groups and the control group.

We thus propose a normalization procedure for the amplitude of the deformation by a scaling step that conserves the relative volume change that occurs during the time deformation. This method is compared to the approach of Niethammer and Vialard 2013 where a product metric is constructed with a volume preserving factor.

Then, to study the normalized deformations, we summarize each deformation with a spline regression, and then perform statistics on the parameters of the splines. This chapter extends the work that was presented at the IEEE International Symposium on Biomedical Imaging (ISBI) in Guigui, Moceri, et al. 2021 with additional experiments and comparisons between the models, both on simulated 2d data and on real 3d data.

Chapter 6 The last chapter concludes this thesis with a summary of the contributions, and give ideas for future works.

Chapter 2

Riemannian geometry: from basic theory to implementation

Contents

2.1 Introduction

Since the formal axiomization of Euclid in his famous *Elements* (dated around 300 BC), geometry was considered as the properties of figures in the plane or in space. The abstract notion of space as a mathematical object emerged in 1827 with C. F. Gauss' *Theorema Egrerium* proving that curvature is an intrinsic quantity of a surface, i.e. that can be computed without reference to a "larger" embedding space.

This notion was made precise by the cornerstone work of Riemann 1868[1] built around the intuitive idea that a mathematical space results from varying a number of independent quantities, later identified as coordinates and formalized in the definition of a manifold by Whitney 1936. Riemannian Geometry (RG) is the study of such differentiable manifolds equipped with an inner product at each point that smoothly varies between points. This allows to generalize the notions of angles, length and volumes, which can be integrated to global quantities highly coupled with the topology of the space.

Fruitful developments of these ideas allowed unifying previous examples of non-Euclidean geometries, i.e. that violate Euclid's parallel postulate which states that given a point and a straight-line, one and only one parallel straight-line can be drawn through the point. These ideas echoed with the developments of Lagrangian and Hamiltonian mechanics, and were instrumental in formalizing the modern theories of Physics, and especially Einstein's general relativity. They also made profound impact on many areas of mathematics such as group theory, representation theory, analysis, and algebraic and differential topology.

At the intersection of Physics and geometry, groups represent symmetries and transformations between states, and from the modern point of view of Klein's Erlangen programm, the study of geometry boils down to studying the action of groups on a space, and their invariants. The work of Elie Cartan allowed significant progress in this direction.

Riemannian geometry has thus become a vast subject that is not usually taught before graduate education in mathematics or physics, and that requires familiarity with many concepts from differential geometry. Hence, although some books on the topic can cover most of the pre-requisites and fundamental results of Riemannian geometry, the entry cost for applied mathematicians, computer scientists and engineers is extremely high.

Nowadays, as data is a predominant resource in applications, Riemannian geometry is a natural framework to model and unify complex nonlinear sources of data. However, the development of computational tools from the basic theory of Riemannian geometry is laborious due to often high dimensional and non-exhaustive coordinate systems, nonlinear and intractable differential equations, etc. One of the aims of this thesis is to provide the computational tools to perform statistics and machine learning on Riemannian manifolds to the wider data science community. The work presented here forms one of the main contributions to the open-source project geomstats, that consists in a Python package providing efficient implementations of the concepts of Riemannian geometry, both for mathematicians and for applied scientists for whom most of the difficulties are hidden under high-level functions.

The goal of this chapter is two-fold. First, to give a self-contained exposition of the basic concepts of Riemannian geometry, providing illustrations and examples at each step and adopting a computational point of view. We cover the basics of differentiable manifolds (Section 2.2), Riemannian metrics (Section 2.3) and Lie groups

[1] Riemann 1873, English Translation by W. K. Clifford.

(Section 2.4). Most proofs are omitted for brevity, and the interested reader may refer to the classic textbooks Lafontaine, Gallot, et al. 2004; Gallier and Quaintance 2020. Some mathematical definitions from the prerequisites can be found in the lexicon in Appendix A. Second, to demonstrate how these concepts are implemented in geomstats, explaining the choices that were made and the conventions chosen. This is mainly detailed in Subsection 2.2.4, and along the text and examples. The culmination of this implementation is to be able to perform statistics and machine learning on manifolds, with as few lines of codes as in wide-spread machine learning tools such as `scikit-learn`. We exemplify this in Section 2.5 with a brief introduction to geometric statistics.

2.2 Differentiable manifolds and tangent spaces

The differentiable manifold will be the structure underlying this entire thesis, yet its definition remains difficult for newcomers in the field. We start by that of an embedded manifold and generalize to the abstract case in a second part. The intuition behind the notion of smooth manifold is that around every point, it resembles $\mathbb{R}^d$, whose properties allow to define the notions of smooth functions, tangent vectors, etc. on the manifold.

2.2.1 Embedded manifolds

The fundamental examples of an embedded manifold are that of an open set of $\mathbb{R}^N$, and a vector subspace $\mathbb{R}^d \times \{0\} \subset \mathbb{R}^N$. These are 'deformed' via local diffeomorphisms to obtain an embedded manifold, which can be thought of a smooth surface in the ambient space. This was in fact one of the first motivations of the mathematical developments underlying the notion of manifold.

Definition 2.2.1. Let d, N be integers with $1 \leq d \leq N$. Then a d-dimensional smooth *embedded manifold* in $\mathbb{R}^N$ is a non-empty subset M of $\mathbb{R}^N$ such that for every point $p \in M$, there are two open subsets $U, V \subseteq \mathbb{R}^N$ with $p \in U$ and $0 \in V$, and a smooth diffeomorphism $f : U \to V$ such that $f(U \cap M) = V \cap (\mathbb{R}^d \times \{0\})$.

Thankfully there are equivalent definitions that give greater insights into what makes M a differentiable manifold (Gallier and Quaintance 2020, Theorem 3.6). We first need to define the notions of immersions and submersions.

Definition 2.2.2. Let $n \leq p$ be two integers, and $U \in \mathbb{R}^p, V \in \mathbb{R}^n$ two open sets.

- A differential map $f : U \to V$ is called a *submersion* at $x \in U$ if df_x is surjective. We say that f is a submersion if it is a submersion at every $x \in U$.
- A differential map $f : V \to U$ is called an *immersion* at $x \in V$ if df_x is injective. We say that f is an immersion if it is an immersion at every $x \in V$.

The fundamental example of an immersion is the injection $(x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_n, 0, \ldots, 0) \in \mathbb{R}^p$, while the projection $(x_1, \ldots, x_p) \in \mathbb{R}^p \mapsto (x_1, \ldots, x_n) \in$

$\mathbb{R}^n$ is that of a submersion. In fact, one can show that up to a local change of variables (i.e. composition with a diffeomorphism), these maps are respectively the only immersions and submersions.

Theorem 2.2.1. *A nonempty subset $M \subseteq \mathbb{R}^N$ is a d-dimensional manifold if and only if any of the following conditions hold:*

(1) (Local parametrization) For every $p \in M$, there are two open subsets $V \subseteq \mathbb{R}^d$ and $U \subseteq \mathbb{R}^N$ with $p \in U$ and $0 \in V$, and a smooth function $f : V \to \mathbb{R}^N$ such that $f(0) = p$, f is a homeomorphism between V and $U \cap M$, and f is an immersion at 0.

(2) (Local implicit function) For every $p \in M$, there exist an open set $U \in \mathbb{R}^N$ and a smooth map $f : U \to \mathbb{R}^{N-d}$ that is a submersion at p, such that $U \cap M = f^{-1}(\{0\})$.

(3) (Local graph) For every $x \in M$, there exist an open neighborhood $U \subseteq \mathbb{R}^N$ of x, a neighborhood $V \subseteq \mathbb{R}^d$ of 0 and a smooth map $f : V \to \mathbb{R}^{N-d}$ such that $U \cap M = \text{graph}(f)$[2]*.*

The characterization (2) encodes the notion of constraint: a manifold can be understood as the set of points that verify a constraint defined by an implicit equation, given by the function f. This is one of the reasons manifolds are ubiquitous in applications, we will give many examples of this case. The other characterizations can also be understood as follows. The first (1) implies that at every point of the manifold, a coordinate system defined on $\mathbb{R}^d$ exists to parametrize the manifold around that point. Finally (3) is the most common to think of surfaces in $\mathbb{R}^3$ as sets of points $(x, y, f(x, y))$.

Example 2.2.1: Hypersphere

The most simple manifold we will study is the hypersphere, or d-dimensional sphere. It is the set of unit-norm vectors of $\mathbb{R}^{d+1}$:

$$S^d = \{x \in \mathbb{R}^{d+1} \mid \|x\|_2^2 = 1\}.$$

Let $f : x \in \mathbb{R}^{d+1} \mapsto \|x\|^2 - 1 \in \mathbb{R}$. For $x \neq 0$, $df_x y = 2x^\top y$ is surjective, so (2) of Theorem 2.2.1 applies and S^d is a d-dimensional embedded manifold in $\mathbb{R}^{d+1}$. In dimension $d = 1$, this correspond to the circle, and for $d = 2$ this is the usual sphere. Both cases are common to represent angles and directions in space, and as such appear in the field of directional statistics (Mardia and Jupp 2009).

[2] the graph of f is the set $\text{graph}(f) = \{(x, f(x)) | x \in V\}$.

Example 2.2.2: Hyperbolic space

The fundamental counterpart to the hypersphere is the two-sheeted hyperboloid, defined by

$$H^d = \{x \in \mathbb{R}^{d+1} \mid -x_0^2 + \sum_{i=1}^{d} x_i^2 = -1\}.$$

It is one of the models of hyperbolic geometry, which is increasingly used to model hierarchical data, e .g. (Nickel and Kiela 2017).

Example 2.2.3: Special Orthogonal group

Matrix groups play an essential role in the theory of RG, and especially the special orthogonal group $SO(n)$, i .e. the set of unit determinant orthogonal matrices:

$$SO(n) = \{R \in M_n(\mathbb{R}) \mid R^\top R = I_n, \quad \det(R) = 1\}.$$

Consider the map $f : \begin{cases} GL^+(n) & \longrightarrow & S(n) \\ A & \longmapsto & A^\top A - I_n \end{cases}$ where $GL^+(n) \in M_n(\mathbb{R}) \simeq \mathbb{R}^{n^2}$ is the open set of invertible squared matrices with positive determinant, $S(n)$ is the set of symmetric matrices of size n, a vector subspace of $M_n(\mathbb{R})$ of dimension $\frac{n(n+1)}{2}$. It is straightforward to show that the differential of f at some R is

$$df_R(H) = R^\top H + H^\top R,$$

and we can see that it is surjective for all $R \in SO(n)$, as for any $S \in S(n)$, $df_R\left(\frac{RS}{2}\right) = S$. As $SO(n) = f^{-1}(0)$, we conclude that $SO(n)$ is indeed an embedded manifold of dimension $\frac{n(n-1)}{2}$.

One can represent the motion of a rigid-body in the referential of its barycenter as a curve with values in $SO(3)$, hence this group is widely used in e.g. robotics (Barczyk, Bonnabel, et al. 2015).

Example 2.2.4: Stiefel manifold

A generalization of both hypersphere and special orthogonal group is the Stiefel manifold, defined as the set of orthonormal k-frames of $\mathbb{R}^n$. If we represent each vector u_i of a k-frame $(u_1, \ldots, u_k)$ as the i^{th} column of a matrix U (in the canonical basis of $\mathbb{R}^n$), then the Stiefel manifold can be seen as a subset in $M_{n,k}(\mathbb{R})$:

$$St(k, n) = \{U \in M_{n,k}(\mathbb{R}) \mid U^\top U = I_k\}.$$

As in the previous example, we can consider the map $f : U \mapsto U^\top U - I$ on an

open subset of $M_{n,k}(\mathbb{R})$ and show that it is a submersion such that $S(k,n) = f^{-1}(0)$ to conclude that $S(k,n)$ is an embedded manifold of dimension $nk - \frac{k(k+1)}{2}$.

The Stiefel manifold naturally arises as the optimization domain in many problems related to matrix decompositions, in linear algebra, statistics, machine learning, computer vision, etc. see Absil, Mahony, et al. 2010, and references therein.

2.2.2 Manifolds

For generality, we now define a manifold in a more abstract way, i.e. as a topological space that is not a priori embedded in some $\mathbb{R}^N$. The idea is still that a manifold is a space that can be covered by open sets that each look like (i.e. are diffeomorphic to an open set of) the usual space $\mathbb{R}^d$. Of course one can verify that embedded manifolds are indeed manifolds with this more general definition, and in fact, Whitney 1936 proved that any manifold can be smoothly embedded in a larger space, showing that the two concepts are equivalent. We first motivate the need for a more general definition of manifold by the example of the Kendall size-and-shape space.

Example 2.2.5: Kendall size-and-shape space

The underlying idea is that a shape is what is left after removing the effects of translation and rotation. We first define the set of k landmarks of $\mathbb{R}^m$ as the space of $m \times k$ matrices $M_{m,k}(\mathbb{R})$. For $x \in M_{m,k}(\mathbb{R})$, let x_i denote the columns of x, i.e. points of $\mathbb{R}^m$ and let $\bar{x}$ be their barycenter. We remove the effects of translation by considering the matrix with columns $x_i - \bar{x}$ instead of x.

Now, in order to remove the effects of rotations, we would like to *identify* the landmark configurations that only differ by a rotation of all the landmarks. This defines an equivalence relation $\sim$:

$$\forall x, y \in M_{m,k}(\mathbb{R}), \quad x \sim y \iff \exists R \in SO(m), y = Rx.$$

A shape thus corresponds to an equivalence class $[x]$ of landmark configurations, and we can define the size-and-shape space as the quotient of the landmark space by the equivalence relation $\sim$ (or equivalently by $SO(n)$):

$$S\Sigma_m^k = \{[x] \mid x \in M_{m,k}(\mathbb{R})\}.$$

A quotient space of a manifold by another manifold may not even be a Hausdorff space, but we will give sufficient conditions in Section 2.4.3 to ensure that quotients resulting from a group action are indeed smooth manifolds.

In this case, $S\Sigma_m^k$ inherits a differentiable structure from the landmark space, that turns it into a smooth manifold, although we cannot see it explicitly as a subset of $\mathbb{R}^N$ for some N. Implementing tools to work with data

on such spaces is a challenging task that we tackled in geomstats. This is the subject of Chapter 3. Kendall size-and-shape spaces are a ubiquitous framework for the statistical analysis of data arising from medical imaging, computer vision, biology, chemistry and many more domains (Dryden and Mardia 2016).

Definition 2.2.3 (Atlas)**.** Let M be a topological space. A *C^k-atlas* $\mathcal{A}$ with values in $\mathbb{R}^d$ is a set of pairs (U, φ) called *charts* where $\varphi : U \to V$ is a homeomorphism between the open sets $U \subset M$ and $V \subset R^d$, such that $M \subset \bigcup_{U \in \mathcal{A}} U$ and for any (U, φ) and (U', φ') in $\mathcal{A}$, the *transition map*

$$\varphi' \circ \varphi : \varphi(U \cap U') \to \varphi'(U \cap U')$$

is a C^k-diffeomorphism.

Two atlases are *C^k-compatible* if their union is still a C^k-atlas. Compatibility defines an equivalence relation, and we will think of the equivalence class of an atlas whenever referring to one. There is a unique *maximal* atlas (for the inclusion) that contains a given atlas.

Note that the transition maps are defined between open sets of $\mathbb{R}^d$, the usual notions of differentiability are thus available, and will allow to define such notions on manifolds. We will always consider the case $k = \infty$ and say that C^∞ maps and atlases are *smooth*.

Definition 2.2.4 (Differentiable manifold)**.** We call *differentiable manifold* of class C^k and dimension d any topological space M that is Hausdorff and second-countable together with a maximal C^k-atlas $\mathcal{A}$ with values in $\mathbb{R}^d$.

We sometimes refer to the atlas of M as its differentiable structure, and to this definition of manifold as the *intrinsic* definition as the charts are defined on M rather than an extrinsic embedding space. A chart (U, φ) defines a set of *local coordinates* on U written $(x^1, \dots, x^d)$ for short, and defined by $x^i = pr_i(\varphi(x))$, where pr_i is the projection on the i^{th} coordinate of $\mathbb{R}^d$. Let's now use the intrinsic definition of a manifold to exemplify further how manifolds can be obtained from others.

Example 2.2.6: Product manifold

Let M, N be two manifolds with $(U_i, \varphi_i)_{i \in I}, (V_j, \psi_j)_{j \in J}$ their respective atlas. Define for $(i, j) \in (I \times J)$

$$\phi_{ij} : \begin{cases} U_i \times V_j & \longrightarrow & \varphi_i(U_i) \times \psi_j(V_j) \\ (x, y) & \longmapsto & (\varphi_i(x), \psi_j(y)) \end{cases}$$

Then it is easy to check that $(U_i \times V_j, \varphi_{ij})_{(i,j) \in I \times J}$ is an atlas for the product space $M \times N$. This atlas does not depend on the choice of original atlases of M, N (in the right equivalence class), and allows to endow $M \times N$ with

the structure of manifold. We call it the product manifold of M and N. Its dimension is $\dim(M \times N) = \dim M + \dim N$.

For a better grasp of the importance of the general notion of manifold, let's now consider some counter-examples.

Example 2.2.7: Cusp and Node

For more details on these two examples see Gallier and Quaintance 2020, chapter 7. First, we consider the classic example of a space that is not a manifold: the nodal cubic, shown in Figure 2.1. It is the set of points

$$M_1 = \{(x, y) \in \mathbb{R}^2 \mid y^2 = x^2 - x^3\}.$$

The self-intersection at the origin doesn't preserve the topology of $\mathbb{R}$, so no homeomorphism can exist between M and $\mathbb{R}$ around the origin. Thus, M is not a manifold.

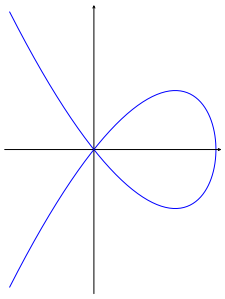


Figure 2.1: Nodal cubic

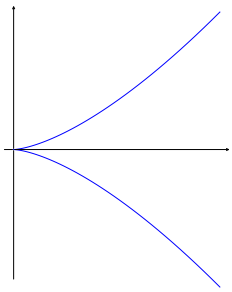


Figure 2.2: Cuspidal curve

Secondly, we consider the cuspidal curve displayed on Figure 2.2 and defined as the set

$$M_2 = \{(x, y) \in \mathbb{R}^2 \mid y^2 = x^3\}.$$

We can define the maps $\varphi : (x, y) \in M_2 \mapsto y^{1/3} \in \mathbb{R}$ and $\psi : (x, y) \in M_2 \mapsto y \in \mathbb{R}$, that each define a smooth atlas on M_2 and endow it with a differentiable structure of manifold. However, the two atlases (constituted of a single chart) are not compatible, so they define different manifolds.

For the next notions that will be introduced, we will use the convenient setting of embedded manifolds, but all these notions can be generalized to the abstract case by using charts to recover functions defined between vector spaces.

2.2.3 Tangent spaces and differentiable maps

We first define smooth curves on manifolds, using a local parametrization of M and the notion of smooth function from $\mathbb{R}$ to $\mathbb{R}^d$. The following definition does not depend on the choice of local parametrization.

Definition 2.2.5 (Smooth curve)**.** Let M be a d-dimensional manifold in $\mathbb{R}^N$. A *smooth curve* γ in M is any function $\gamma : I \to M$ where I is an open interval in $\mathbb{R}$, such that for any $t \in I$, $p = \gamma(t)$, there is a local parametrization $\varphi : V \to U$ of M at p and $\epsilon > 0$ such that $\varphi^{-1} \circ \gamma : (t - \epsilon, t + \epsilon) \to \mathbb{R}^d$ is smooth.

This definition is extended to smooth curves defined on a closed interval $I = [a, b]$ of $\mathbb{R}$ by requiring that γ be the restriction of some smooth curve defined on an open interval that contains $[a, b]$. A tangent vector along γ at some time $t \in I$ is obviously defined, as γ can be differentiated. This generalizes to tangent spaces to the manifold.

Definition 2.2.6 (Tangent vector)**.** Let $d \leq N \in \mathbb{N}$, M be an embedded manifold in $\mathbb{R}^N$ of dimension d and $p \in M$. A vector $v \in \mathbb{R}^N$ is *tangent* to M at p if there exists an open interval I centered around 0, and a curve $\gamma : I \to M$ such that

$$\gamma(0) = p \text{ and } \dot{\gamma}(0) = v.$$

We write T_pM for the set of tangent vectors at p.

Recall that $\mathbb{R}^d \times \{0\}$ is a fundamental example of an embedded manifold, it is clear that the tangent space at any $p \in \mathbb{R}^d \times \{0\}$ is the whole $\mathbb{R}^d \times \{0\}$. From this case we deduce the characterizations of tangent spaces equivalent to that of manifolds obtained in Theorem 2.2.1 (Paulin 2007, Proposition 3.1).

Theorem 2.2.2. *Let M be a manifold in $\mathbb{R}^N$ of dimension d.*

(1) If $U, V \subset \mathbb{R}^N$ are two open neighborhoods respectively around p and 0 in $\mathbb{R}^N$ and $f : U \to V$ is a diffeomorphism such that $f(p) = 0$ and $f(U \cap M) = V \cap (\mathbb{R}^d \times \{0\})$ then

$$T_pM = df_p^{-1}(\mathbb{R}^d \times \{0\}).$$

(2) (Local parametrization) If $U \subseteq \mathbb{R}^N$ is an open neighborhood of $p \in M$, $V \subseteq \mathbb{R}^d$ is an open neighborhood around 0 and $f : V \to \mathbb{R}^N$ is a smooth function such that $f(0) = p$, f is a homeomorphism between V and $U \cap M$, and f is an immersion at 0, then

$$T_pM = \operatorname{Im} df_0.$$

(3) (Local implicit function) If $U \subseteq \mathbb{R}^N$ is an open neighborhood around $p \in M$ and $f : U \to \mathbb{R}^{N-d}$ is a smooth map that is a submersion at p, such that $U \cap M = f^{-1}(\{0\})$, then

$$T_pM = \ker df_p.$$

(4) (Local graph) If $U \subseteq \mathbb{R}^N$ is an open neighborhood of $p \in M$, $V \subseteq \mathbb{R}^d$ a neighborhood of 0 and $f : V \to \mathbb{R}^{N-d}$ a smooth map such that $U \cap M = \operatorname{graph}(f)$ and $p = (0, f(0))$, then

$$T_pM = \operatorname{Im}\{v \mapsto (v, df_0(v))\}.$$

From (1) we can see that T_pM is a linear subspace of $\mathbb{R}^N$ of dimension d. Tangent spaces thus provide local linearizations of the manifold, a property that will be useful as a first way to handle data on manifolds. The previous theorem allows computing the tangent spaces of the common manifolds seen in the previous section.

Example 2.2.8: Hypersphere

Recall that the hypersphere is the embedded manifold defined by $S^d = f^{-1}(0)$ where $f : x \mapsto \|x\|^2 - 1$. This corresponds to (3) of Theorem 2.2.2, therefore for any $x \in S^d$

$$T_xS^d = \{v \in \mathbb{R}^{d+1} \mid \langle x, v\rangle = 0\}.$$

Example 2.2.9: Hyperbolic space

Similarly, as the hyperbolic space is defined as $H^d = f^{-1}(0)$ where $f : x \mapsto -x_0^2 + \sum_{i=1}^d x_i^2 + 1$, we obtain for any $x \in H^d$

$$T_xH^d = \{v \in \mathbb{R}^{d+1} \mid -x_0v_0 + \sum_{i=1}^d x_iv_i = 0\}.$$

Example 2.2.10: Special Orthogonal group

Recall $SO(n) = f^{-1}(0)$ with $f : A \mapsto A^\top A - I_n$, and for any $R \in SO(n), H \in M_n(\mathbb{R})$ we have $df_R(H) = R^\top H + H^\top R$. Therefore for any $R \in SO(n)$

$$T_RSO(n) = \{H \in M_n(\mathbb{R}) \mid R^\top H + H^\top R = 0\}.$$

Note the special case $R = I_n$, then $T_{I_n}SO(n) = \mathrm{Skew}(n)$, the set of skew-symmetric matrices of size n. The tangent space at the identity of a Lie group will play a particularly important role as will be exposed in Section 2.4.

Example 2.2.11: Stiefel manifold

Similarly, for any $U \in St(k, n)$,

$$T_USt(k, n) = \{H \in M_{n,k}(\mathbb{R}) \mid U^\top H + H^\top U = 0\}.$$

We can now define the notion of smooth maps between manifolds and their differential.

Definition 2.2.7 (Smooth map)**.** Let M, Q be two manifolds of dimensions $d_1, d_2 \in \mathbb{N}$ embedded in $\mathbb{R}^N$. A function $f : M \to Q$ is *smooth* if for every $p \in M$, there are parametrizations $\varphi : V_1 \to U_1$ of M at p and $\psi : V_2 \to U_2$ of Q at $f(p)$ such that

$f(U_1) \subseteq U_2$ and

$$\psi^{-1} \circ f \circ \varphi : V_1 \to V_2 \qquad \text{is smooth.}$$

Note that in the above definition V_1, V_2 are open sets of $\mathbb{R}^N$ so the notion of smooth map from V_1 to V_2 is already well known (see lexicon in Appendix A).

Definition 2.2.8 (Differential)**.** Let M, Q be two manifolds of dimensions $d_1, d_2 \in \mathbb{N}$ embedded in $\mathbb{R}^N$ and $f : M \to Q$ a smooth map. For any $p \in M$ and any $v \in T_pM$, let γ be a smooth curve through p such that $\dot{\gamma}(0) = v$ and define

$$df_p(v) = (f \circ \gamma)'(0).$$

This definition does not depend on the choice of curve γ and the map $df_p : T_pM \to T_{f(p)}Q$ is called the *differential* or *tangent map* of f at p. It is a linear map between tangent spaces.

It generalizes the differential map of a differentiable function defined from $\mathbb{R}^{N_1}$ to some $\mathbb{R}^{N_2}$ to functions defined on the manifold M instead of the embedding space. It coincides with the original differential (Definition A.0.10 in the lexicon page 191) when $M = \mathbb{R}^d$, hence the use of the same notation df_p. The set of real-valued smooth maps on M will be particularly useful. For short, we denote it $C^\infty(M) \triangleq C^\infty(M, \mathbb{R})$. $C^\infty(M)$ is clearly an infinite dimensional $\mathbb{R}$-vector space, and with point-wise multiplication, an algebra.

Next, it is convenient to consider the set of all the tangent spaces at all points

$$TM = \bigsqcup_{x \in M} \{x\} \times T_xM = \{(x, v) \,|x \in M, v \in T_xM\}.$$

and its natural projection

$$\pi : \left\{ \begin{array}{ccc} TM & \longrightarrow & M \\ (x, v) & \longmapsto & x \end{array} \right. .$$

This space is called the *tangent bundle* of M, and one can show that if M is a manifold of class C^{k+1} and dimension d, then TM is itself a manifold in $\mathbb{R}^N \times \mathbb{R}^N$, of class C^k and dimension $2d$. The tangent bundle is the domain of definition of the differential of smooth functions defined on manifolds:

$$f : M \to Q, \qquad df : TM \to TQ$$

It is also the space where *vector fields* are valued: a vector field X is a smooth assignment of a tangent vector to each point of a manifold, i.e. $X : M \to TM$ such that $\forall p \in M, \pi \circ X(p) = p$. $X(p)$ will be written X_p for convenience. Let $\Gamma(TM)$ denote the set of all vector fields (VF). It is clear that $\Gamma(TM)$ equipped with point-wise sum and multiplication by a scalar forms a vector space. Multiplication by a smooth function is also defined pointwise: for any $f \in C^\infty(M)$ and $X \in \Gamma(TM)$, fX is the vector field such that

$$\forall p \in M, (fX)_p = f(p)X_p.$$

This turns the set of vector fields into a $C^\infty(M)$-module. For a smooth map $f : M \to \mathbb{R}$ and a vector field X, we write $X(f)$ the function defined at every p by

$$X(f)(p) = df_p(X_p).$$

This leads to the following remark.

Remark 2.2.1. *We defined vector fields as sections of the tangent bundle, i.e., maps $\sigma : M \to TM$ such that $\pi \circ \sigma = Id$. Alternatively, vector fields can be defined as derivations over the algebra $C^\infty(M)$ of smooth real valued functions. A derivation $X : C^\infty(M) \to C^\infty(M)$ is a linear map that satisfies the Leibniz rule:*

$$\forall f, g \in C^\infty(M), \quad X(fg) = fX(g) + X(f)g. \tag{2.1}$$

One can check that a vector field as defined above indeed defines a derivation. However, applying the "composition" of two vector fields to a function f is not a derivation because of second-order derivatives of f. This leads to the definition of the Lie bracket of vector fields.

Definition 2.2.9 (Lie bracket over $\Gamma(TM)$)**.** The Lie bracket of vector fields is defined as the map

$$[\cdot,\cdot] : \left\{ \begin{array}{ccc} \Gamma(TM) \times \Gamma(TM) & \longrightarrow & \Gamma(TM) \\ (X, Y) & \longmapsto & f \mapsto X(Y(f)) - Y(X(f)) \end{array} \right. \tag{2.2}$$

A useful tool to handle vector fields locally is to use a basis of T_pM for p in some open set U.

Definition 2.2.10 (Frame)**.** Let M be a d-dimensional manifold. For any open set $U \subseteq M$, a family of vector fields $(X_1, \ldots, X_d)$ over U is called a *frame* over U if for every p in U, $\big(X_1(p), \ldots, X_d(p)\big)$ is a basis of T_pM.

Any chart (U, φ) defines a local frame that corresponds to its local coordinates: define the i^{th} curve $\gamma_i : t \in \mathbb{R} \mapsto \varphi^{-1}(0, \ldots, 0, t, 0, \ldots, 0) \in U$ and $X_i(\gamma_i(t)) = \dot{\gamma}_i(t)$. The vector field X_i defined on U is often written $X_i = \frac{\partial}{\partial x^i}$ or simply ∂_i and corresponds to $d\varphi^{-1}(e_i)$ where $(e_1, \ldots, e_d)$ is the canonical basis of $\mathbb{R}^d$. Then the family $(X_1, \ldots, X_d)$ is a local frame over U.

Remark 2.2.2. *If a family $(X_1, \ldots, X_d)$ is a frame over the whole manifold (i.e. $U = M$), we say that it is a global frame. Whether global frames exist depends on the topology of the manifold, and in that case the tangent bundle is called trivial, i.e. isomorphic to the direct product $M \times \mathbb{R}^N$. This is not the case of e.g. the sphere (of dimension 2), by the hairy ball theorem* [3].

Vector fields can be considered as infinitesimal generators of local maps, as we shall see in the following. These maps, called flows, usually supply strong information on global properties of the manifold. In this thesis, we will mainly focus on geodesic flows (sec. 2.3), and flows of left-invariant vector fields on Lie groups (sec. 2.4.2).

[3] https://en.wikipedia.org/wiki/Hairy_ball_theorem

Definition 2.2.11 (Integral curve)**.** Let $X \in \Gamma(M)$ and $p_0 \in M$. An *integral curve* for X with initial condition p_0 is a curve $\gamma : I \to M$ such that

$$\forall t \in I, \quad \dot{\gamma}(t) = X_{\gamma(t)} \quad \text{and} \quad \gamma(0) = p_0,$$

where $I = (a, b) \subseteq \mathbb{R}$ is an open interval containing 0.

An integral curve is thus a curve whose speed $\dot{\gamma}(t)$ coincide with X at any point along the curve. A collection of such integral curves is called a flow:

Definition 2.2.12 (Local flow)**.** Let $X \in \Gamma(M)$ and $p_0 \in M$. A *local flow* of X at p_0 is a map $\phi : I \times U \to M$ where I is an open interval containing 0 and U is an open set in M containing p, such that for every $p \in U$, the curve $t \mapsto \phi(t, p)$ is an integral curve of X starting from p.

Thanks to the theory of ordinary differential equations (ODE), one can prove that for any vector field, there is a local flow defined around any point, and if two such flows are defined on overlapping domains, they coincide on the intersection. For $t \in I$, we write $\phi_t : x \mapsto \phi(t, x)$. It is clear that ϕ_0 is the identity map, and for some $t \neq 0$, ϕ_t is a map defined locally on M.

Proposition 2.2.1. *Let X be a smooth vector field on M, $p_0 \in M$ and $\phi : I \times U \to M$ the local flow of X at p_0. For any $s, t \in I$ and $x \in U$,*

- *if $\phi_s(x) \in U$ and $t + s \in I$, then $\phi_t \circ \phi_s(x) = \phi_{t+s}(x)$;*
- *ϕ_t is a local diffeomorphism;*
- *ϕ_t preserves X, in the sense that $\forall t \in I, \forall x \in U$, $d(\phi_t)_x(X_x) = X_{\phi_t(x)}$.*

Definition 2.2.13 (Complete)**.** We say that X is *complete* if the domain of definition of its flow $(t, x) \mapsto \phi_t(x)$ is the entire $\mathbb{R} \times M$. If all the vector fields of M are complete, we say that M is complete.

In that case ϕ_t is a diffeomorphism of M, and $(\phi_t)_{t \in \mathbb{R}}$ is a *one parameter subgroup* of $\mathrm{Diff}(M)$.

We now have the ingredients to introduce geomstats.

2.2.4 Implementation in geomstats

Now that the fundamental notion of manifold has been exposed, we can delve more into the architecture of the geomstats package, and summarize the choices that we made in its development. Firstly the package is organized in different modules that distinguish between the geometric and the statistical operations. There is thus a geometry module, that gathers all the implementations of manifolds, connections and Riemannian metrics, and a learning module where estimation algorithms are implemented in a generic fashion and take the geometric structure and the data as inputs. The goal is that all the learning algorithms can be run seamlessly on different manifolds, and with different metrics. A sampling module is currently

being developed with the same spirit, and a visualization module allows to plot data on the common manifolds in dimension two or three. In this section, we focus on the geometry module, and more specifically on the objects that represent manifolds. The Riemannian metrics objects will be described in Section 2.3 along the mathematical definitions. The statistical and learning tools will be described in Section 2.5.

The package is object-oriented in the sense that all the tools are implemented as classes that contain all the methods related to a tool. The geometry module of geomstats is organized by classes that each represent a geometric structure. To guarantee the consistency of all the classes, we implement an abstract parent `Manifold` class, and the actual implementations of the usual manifolds are subclasses of this parent class. The aim of abstract base classes is to provide the minimal skeleton of attributes and methods expected in its subclasses. The methods of the abstract class are thus declared but contain no implementation, and they are overridden by the subclasses.

The manifold classes A subclass of `Manifold` gathers the methods to work with data lying on the considered manifold. Note that these classes do not explicitly provide a representation of the manifold (as e.g. a triangulated surface in 2d), but the tools to handle points and tangent vectors. As we work with embedded manifolds in most cases, points and tangent vectors are themselves represented by multi-dimensional arrays. These are either `NumPy` arrays, or `TensorFlow` of `PyTorch` tensors according to the backend that is being used.

Mathematically, the first attribute of a manifold is of course its dimension, called `dim` for brevity throughout the package. Then we use an attribute to inform on the expected type of the point: whether vectors (for e.g. the hypersphere and hyperbolic space) or matrices (e.g. SPD matrices, the special orthogonal group) should be used. This is called `default_point_type`.

Furthermore, a `Manifold` in geomstats should always implement a method that evaluates whether a given element belongs to that manifold, and whether a given vector is a tangent vector at a given point. These are the `belongs` and `is_tangent` methods. For practical reasons we also add a `random_point` method, to generate random points that belong to the manifold (regardless of the distribution). This is useful in particular to test the methods and the learning algorithms. We obtain the following base class:

```
class Manifold(abc.ABC):
    """Class for manifolds."""

    def __init__(
            self, dim, metric=None, default_point_type='vector',  **kwargs):
        super(Manifold, self).__init__(**kwargs)
        self.dim = dim
        self.default_point_type = default_point_type
        self.metric = metric
```

```
    @abc.abstractmethod
    def belongs(self, point, atol=gs.atol):
        """Evaluate if a point belongs to the manifold."""

    @abc.abstractmethod
    def is_tangent(self, vector, base_point, atol=gs.atol):
        """Check whether the vector is tangent at base_point."""

    @abc.abstractmethod
    def random_point(self, n_samples=1, bound=1.):
        """Sample random points on the manifold."""
```

Note that the `Manifold` class contains a metric attribute. This will be detailed in Section 2.3.

Implementation trick 2.1. *The methods decorated (@ symbol) with* `abc.abstractmethod` *are declared as abstract methods of the class. A class that contains abstract methods cannot be instantiated. This constrains the developer to implement these functions explicitly when writing subclasses of* `Manifold`*.*

Two elementary classes of manifolds Throughout the current chapter, we have met two elementary ways of defining a manifold, that correspond respectively to (1) and (2) of Theorem 2.2.1 (page 14):

1. As open sets of a d-dimensional vector space, called ambient space. In this case, we can add a `projection` method to project any d-dimensional vector to the manifold, and all the tangent spaces are identified with the ambient space. Therefore, the method `is_tangent` just checks if the input belongs to the ambient space, and we can add a method `to_tangent`, that calls the projection of the ambient space, to project to a tangent space, assuming the ambient space to be itself embedded in another space, or being a vector space, whose projection is the identity.

2. As pre-image of a submersion $f : \mathbb{R}^N \to \mathbb{R}^{N-d}$. We refer to such space as a level-set. In this case, specifying f, it is straightforward to implement the `belongs` and `is_tangent` method by evaluating f and its differential. It also makes sense to add `projection` and `to_tangent` methods from the embedding space to respectively the manifold and the tangent space at a point.

We thus implement two more abstract classes, the first for open sets:

```
class OpenSet(Manifold, abc.ABC):
    """Class for manifolds that are open sets of a vector space."""

    def __init__(self, dim, ambient_space, **kwargs):
        if 'default_point_type' not in kwargs:
            kwargs['default_point_type'] = ambient_space.default_point_type
        super().__init__(dim=dim, **kwargs)
        self.ambient_space = ambient_space
```

```
    def is_tangent(self, vector, base_point, atol=gs.atol):
        """Check whether the vector is tangent at base_point."""
        return self.ambient_space.belongs(vector, atol)

    def to_tangent(self, vector, base_point):
        """Project a vector to a tangent space of the manifold."""
        return self.ambient_space.projection(vector)

    def random_point(self, n_samples=1, bound=1.):
        """Sample random points on the manifold."""
        sample = self.ambient_space.random_point(n_samples, bound)
        return self.projection(sample)

    @abc.abstractmethod
    def projection(self, point):
        """Project a point in ambient manifold on manifold."""
```

And the second for level-sets:

```
class LevelSet(Manifold, abc.ABC):
    """Class for manifolds embedded in a vector space by a submersion."""

    def __init__(
            self, dim, embedding_space, submersion, value,
            tangent_submersion, **kwargs):
        super(LevelSet, self).__init__(
            dim=dim, default_point_type=embedding_space.default_point_type,
            **kwargs)
        self.embedding_space = embedding_space
        self.embedding_metric = embedding_space.metric
        self.submersion = submersion
        if isinstance(value, float):
            value = gs.array(value)
        self.value = value
        self.tangent_submersion = tangent_submersion

    def belongs(self, point, atol=gs.atol):
        """Evaluate if a point belongs to the manifold."""
        belongs = self.embedding_space.belongs(point, atol)
        if not gs.any(belongs):
            return belongs
        value = self.value
        constraint = gs.isclose(self.submersion(point), value, atol=atol)
        if value.ndim == 2:
            constraint = gs.all(constraint, axis=(-2, -1))
        elif value.ndim == 1:
            constraint = gs.all(constraint, axis=-1)
        return gs.logical_and(belongs, constraint)

    def is_tangent(self, vector, base_point, atol=gs.atol):
        """Check whether the vector is tangent at base_point."""
        belongs = self.embedding_space.belongs(vector, atol)
        tangent_sub_applied = self.tangent_submersion(vector, base_point)
```

```
        constraint = gs.isclose(tangent_sub_applied, 0., atol=atol)
        value = self.value
        if value.ndim == 2:
            constraint = gs.all(constraint, axis=(-2, -1))
        elif value.ndim == 1:
            constraint = gs.all(constraint, axis=-1)
        return gs.logical_and(belongs, constraint)

    @abc.abstractmethod
    def projection(self, point):
        """Project a point in embedding space on the manifold."""
```

To make sure that the attributes that represent the ambient/embedding space do implement the methods that are called in `OpenSet` and `LevelSet`, we also implemented an abstract `VectorSpace` class. Actual manifolds are then implemented as subclasses of the corresponding abstract manifold and must implement all the abstract methods. We give an example of each class below.

Example 2.2.12: Hypersphere

The hypersphere is implemented as embedded in $\mathbb{R}^{d+1}$, so it is a subclass of `LevelSet`.

```
class Hypersphere(LevelSet):
    """Class for the n-dimensional hypersphere."""

    def __init__(self, dim):
        super(Hypersphere, self).__init__(
            dim=dim, embedding_space=Euclidean(dim + 1),
            metric=HypersphereMetric(dim),
            submersion=lambda x: gs.sum(x ** 2, axis=-1), value=1.,
            tangent_submersion=lambda v, x: 2 * gs.sum(x * v, axis=-1))

    def projection(self, point):
        """Project a point on the hypersphere."""
        norm = gs.linalg.norm(point, axis=-1)
        if gs.any(norm < gs.atol):
            logging.warning('0 cannot be projected to the hypersphere')
        return gs.einsum('...,...i->...i', 1. / norm, point)

    def to_tangent(self, vector, base_point):
        """Project a vector to the tangent space."""
        sq_norm = gs.sum(base_point ** 2, axis=-1)
        inner_prod = self.embedding_metric.inner_product(base_point, vector)
        coef = inner_prod / sq_norm
        return vector - gs.einsum('...,...j->...j', coef, base_point)
```

Implementation trick 2.2. *All the methods of geomstats are vectorized*[4]*, in the*

[4] Vectorization may also be referred to as *array programming* on e.g. Wikipedia.

sense that they can take as argument either one input, or a collection of inputs corresponding to multiple samples. It is very useful to use the `einsum` *method for that purpose, with ellipses ('...') that represent an optional additional dimension.*

For example, the syntax `gs.einsum('...,...i->...i', coef, point)` *performs scalar multiplication between a list of scalars (*`coef`*) and a list of* `points`*, but it also works for a single scalar and a single point.*

Example 2.2.13: Poincaré Ball

The Poincare ball is one of the models of hyperbolic geometry, and is defined as the open unit disk of $\mathbb{R}^d$. It is then a subclass of `OpenSet`.

```
class PoincareBall(OpenSet):
    """Class for the n-dimensional hyperbolic space."""

    def __init__(self, dim, scale=1):
        super(PoincareBall, self).__init__(
            dim=dim, ambient_space=Euclidean(dim),
            metric=PoincareBallMetric(dim))

    def belongs(self, point, atol=gs.atol):
        """Test if a point belongs to the unit ball."""
        return gs.sum(point**2, axis=-1) < (1 - atol)

    def projection(self, point):
        """Project a point on the unit ball."""
        if point.shape[-1] != self.dim:
            raise NameError("Wrong dimension, expected ", self.dim)

        l2_norm = gs.linalg.norm(point, axis=-1)
        if gs.any(l2_norm >= 1 - gs.atol):
            projected_point = gs.einsum(
                '...j,...->...j', point * (1 - gs.atol), 1. / l2_norm)
            projected_point = -gs.maximum(-projected_point, -point)
            return projected_point
        return point
```

Manifolds can then be composed to define other manifolds by products (Example 2.2.6) or quotients (see Section 3.3.4). For products, we create the class `ProductManifold` that takes existing manifolds to construct a new one, and computations on each manifold can be done in parallel:

```
class ProductManifold(Manifold):
    r"""Class for a product of manifolds M_1 \times ... \times M_n."""

    def __init__(self, manifolds, n_jobs=1):
        self.dims = [manifold.dim for manifold in manifolds]
        super(ProductManifold, self).__init__(dim=sum(self.dims))
```

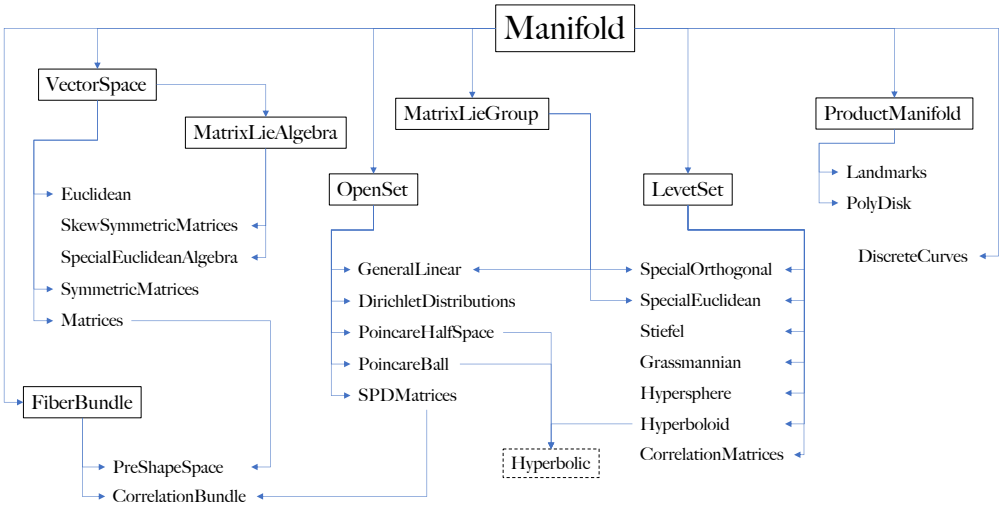


Figure 2.3: Architecture of the manifolds of geomstats. The abstract base classes are in black bounded boxes. Inheritance is shown by blue arrows. `Hyperbolic` is an exception as it is a common interface to the different representations of hyperbolic geometry.

To summarize, a diagram representing all the base classes for manifolds and all the manifolds is shown on Figure 2.3. The abstract base classes are shown with black bounding boxes. Inheritances occur in two cases:

- With an abstract base class as parent class;
- When both parent and child class represent the same manifold, as is the case for example in the `CorrelationBundle` of Example 3.3.8. A common interface `Hyperbolic` for the three representations of hyperbolic geometry follows this logic as any of the three representations can be chosen, but the instantiated object is either `Hyperboloid`, `PoincareBall` or `PoincareHalfSpace` as chosen by the user.

2.3 Riemannian metrics

2.3.1 Definition and examples

We now introduce a new structure on a differentiable manifold: the Riemannian metric, that allows to define the length of a curve, a distance function, a volume form, etc. Note that this additional structure may not be canonical, raising the thorny question of choosing the metric for the applications.

Definition 2.3.1 (Riemannian metric)**.** Let M be a smooth d-dimensional manifold. A *Riemannian metric* on M (or TM) is a family $(\langle \cdot, \cdot \rangle_p)_{p \in M}$ of inner products on

each tangent space T_pM, such that $\langle \cdot, \cdot \rangle_p$ depends smoothly on p. More formally, for any chart φ, U, and frame $(X_1, \dots, X_n)$ on U, the maps

$$p \mapsto \langle X_i(p), X_j(p) \rangle_p \quad 1 \leq i, j \leq n,$$

are smooth. A pair $(M, \langle \cdot, \cdot \rangle)$ is called a *Riemannian manifold*.

A metric is often written $g = (g_p)_{p \in M}$, where g_p is the symmetric, positive definite (SPD) matrix representing the inner-product in a chart, that is

$$g_{ij}(p) = \left\langle \left(\partial_i\right)_p, \left(\partial_j\right)_p \right\rangle_p.$$

Alternatively, we sometimes define a metric with the notation $g = f(dx_1, \dots, dx_n)$ where f is the quadratic form associated to g, and dx_i represent vector coordinates (as linear forms). For example the usual Euclidean metric is $g = \sum_{i=1}^n dx_i^2$. The following theorem ensures that a metric is indeed a general structure. A proof can be found e.g. in Lafontaine, Gallot, et al. 2004, Theorem 2.2.

Theorem 2.3.1 (Existence)**.** *Any smooth manifold admits a Riemannian metric.*

A Riemannian metric also defines a norm on TM, as usually defined by an inner-product on each tangent space:

$$\forall x \in M, \forall v \in T_xM, \quad \|v\|_x = \sqrt{g_x(v, v)} = \sqrt{\langle v, v \rangle_x}.$$

In geomstats In order to guaranty flexibility, we have decided to keep manifolds and metrics separated in different objects. Indeed, although there may exist a canonical Riemannian metric on a given manifold, the choice of metric is not always natural, and researchers have struggled to find criteria to choose the right metric for the application at hand. The aim of geomstats is to allow researchers to compare different metrics on their problem, and in the future, to allow to *learn*, or optimize the metric (Louis, Couronné, et al. 2019; Hauberg 2019).

We create an abstract `RiemannianMetric` class in geomstats to gather its basic attributes and methods. The most general way of defining a metric is to provide the `metric_matrix` method, that is $x \mapsto (g_{ij}(x))_{1 \leq i,j \leq d}$. By default, we use the identity matrix for all points, resulting in the Euclidean metric.

```
class RiemannianMetric(Connection):
    """Class for Riemannian and pseudo-Riemannian metrics."""

    def __init__(self, dim, signature=None):
        super(RiemannianMetric, self).__init__(dim=dim)
        if signature is None:
            self.signature = (dim, 0)

    def metric_matrix(self, base_point=None):
        """Inner product matrix at base point."""
        return gs.eye(self.dim)
```

```
    def inner_product(self, tangent_vec_a, tangent_vec_b, base_point=None):
        """Inner product between two tangent vectors at a base point."""
        inner_prod_mat = self.metric_matrix(base_point)
        inner_prod = gs.einsum(
            '...j,...jk,...k->...', tangent_vec_a, inner_prod_mat, tangent_vec_b)
        return inner_prod

    def squared_norm(self, vector, base_point=None):
        """Compute the square of the norm of a vector."""
        return self.inner_product(vector, vector, base_point)

    def norm(self, vector, base_point=None):
        """Compute norm of a vector."""
        sq_norm = self.squared_norm(vector, base_point)
        return gs.sqrt(sq_norm)
```

Remark 2.3.1.

1. *The above class inherits from* ***`Connection`****. We indeed chose a class for an affine connection as parent class to a Riemannian metric because it is a more general structure, as exposed in paragraph 2.3.2.1.*

2. *The attribute* ***`signature`*** *refers to the signature of the inner product, in case it is only a non-degenerate bilinear form, not necessarily positive. In this case the metric is called a pseudo-Riemannian metric.*

We also add a `metric` property to the `Manifold` class, meaning that it is an attribute of the class, that can be set externally, calling the `setter` that checks that the given argument is indeed an instance of a `RiemannianMetric` object. Of course, all manifolds studied in this thesis come with a default metric, but users can choose to use different metrics or implement new metrics with only a few minimal operations. When closed form solutions are available, the generic methods are overridden. With a metric attribute, a `Manifold` actually becomes a Riemannian manifold, but we did not think relevant to implement another layer of abstract class for Riemannian manifolds (i.e. a `RiemannianManifold` class), as all necessary operations are either in the `Manifold` object if they don't depend on the metric, or in the `RiemannianMetric` object if they do.

```
@property
def metric(self):
    """Riemannian Metric associated to the Manifold."""
    return self._metric

@metric.setter
def metric(self, metric):
    if metric is not None:
        if not isinstance(metric, RiemannianMetric):
            raise ValueError(
                'The argument must be a RiemannianMetric object')
        if metric.dim != self.dim:
```

```
            metric.dim = self.dim
    self._metric = metric
```

Example 2.3.1: Euclidean metric

Let $M = \mathbb{R}^d$ be the standard vector space of dimension d, that is trivially a smooth manifold, and consider its standard inner-product defined for all $x, y \in \mathbb{R}^d$ by

$$\langle x, y \rangle_2 = \sum_{i=1}^{d} x_i y_i = x^\top y.$$

As $T_x\mathbb{R}^d = \mathbb{R}^d$, it defines a metric on $\mathbb{R}^d$, which is referred to as the Euclidean metric.

Example 2.3.2: Product metric

Let (M, g) and (M', g') be two Riemannian manifolds, and recall from Example 2.2.6 (page 17) that the cartesian product $M \times M'$ is a manifold. It is also a Riemannian manifold. Indeed, define the *product metric* $g \oplus g'$ as the map defined at any $(x, x') \in M \times M'$ and $\forall (v, v'), (w, w') \in T_x M \times T_{x'} M'$ by

$$g \oplus g'_{(x,x')}((v, v'), (w, w')) = g(v, w) + g'(v', w').$$

In geomstats, it is possible to define such product metrics from existing objects of the class `RiemannianMetric`, and operations for each metric can be run in parallel if necessary.

```
class ProductRiemannianMetric(RiemannianMetric):
    """Class for product of Riemannian metrics."""

    def __init__(self, metrics, default_point_type='vector', n_jobs=1):
        self.n_metrics = len(metrics)
        dims = [metric.dim for metric in metrics]
        signatures = [metric.signature for metric in metrics]

        sig_pos = sum(sig[0] for sig in signatures)
        sig_neg = sum(sig[1] for sig in signatures)
        super(ProductRiemannianMetric, self).__init__(
            dim=sum(dims), signature=(sig_pos, sig_neg),
            default_point_type=default_point_type)
```

Let (N, g) be a Riemannian manifold, M a smooth manifold, and $f : M \to N$ a

map. Define the *pull-back* metric (f^*g) on M by

$$(f^*g)_x : \begin{cases} T_xM \times T_xM & \longrightarrow & \mathbb{R} \\ (v, w) & \longmapsto & g_{f(x)}(df_x v, df_x w) \end{cases} . \tag{2.3}$$

Now suppose that f is an immersion. If $(f^*g)_x$ is non-degenerate and of constant signature for all $x \in M$, then (M, f^*g) is a Riemannian metric.

Definition 2.3.2 (Isometry)**.** Let (M, g) and (M', g') be two Riemannian manifolds, and $f : M \to M'$. Then f is called an *isometry* if it is a bijection and $f^*g' = g$.

When $M = M'$ and $g = g'$, we write $\mathrm{Isom}(M)$ the set of isometries of M, and Myers and Steenrod 1939 showed that it is a Lie group that acts smoothly on M.

Now, consider that $M \subseteq N = \mathbb{R}^d$ is a submanifold of M, $f = i$ is the inclusion map. Then g is called the embedding metric and i^*g is its restriction to M. This case appears in many examples in geomstats.

Example 2.3.3: Hypersphere

The hypersphere $S^d \subset \mathbb{R}^{d+1}$ endowed with the restriction of the embedding Euclidean metric is a Riemannian manifold. We call this metric the standard spherical metric, and implement it by calling the embedding metric.

```
class HypersphereMetric(RiemannianMetric):
    """Class for the Metric on the Hypersphere."""

    def __init__(self, dim):
        super(HypersphereMetric, self).__init__(
            dim=dim, signature=(dim, 0))
        self.embedding_metric = EuclideanMetric(dim + 1)

    def metric_matrix(self, base_point=None):
        """Inner-product matrix at a base point."""
        return gs.eye(self.dim + 1)

    def inner_product(self, tangent_vec_a, tangent_vec_b, base_point=None):
        """Inner-product of two tangent vectors at a base point."""
        return self.embedding_metric.inner_product(
            tangent_vec_a, tangent_vec_b, base_point)
```

Let's now define the Lorentz bilinear form of $\mathbb{R}^{d+1}$. It is the canonical bilinear form with signature (1, d), i.e. for any $x, y \in \mathbb{R}^{d+1}$

$$\langle x, y \rangle_{\mathcal{L}} = -x_0 y_0 + \sum_{i=1}^{d} x_i y_i \tag{2.4}$$

And write $\|\cdot\|_{\mathcal{L}}$ for the associated quadratic form. This is the underlying embedding metric in the following example.

Example 2.3.4: Hyperbolic space

Using the Lorentz form, recall that $H^d \in \mathbb{R}^{d+1}$ is the set of points such that $\|x\|_{\mathcal{L}} = -1$. Consider now the open subset H^d_+ of H^d:

$$H^d_+ = \left\{ x \in R^{d+1} \mid x_0 > 0,\ \|x\|_{\mathcal{L}} = -1 \right\}.$$

Now, consider any $x \in H^d_+$ and two tangent vectors $v, w \in T_x M$. By Example 2.2.9, this means that $\langle v, x\rangle_{\mathcal{L}} = \langle w, x\rangle_{\mathcal{L}} = 0$. As $\langle , \rangle_{\mathcal{L}}$ is negative definite on $\mathbb{R}x$, and its signature is $(1, d)$, it is positive definite on the orthogonal of $\mathbb{R}x$ for the Lorentz metric, i.e. on $T_x M$. We can thus conclude that $(H^d_+, \langle \cdot, \cdot \rangle_{\mathcal{L}})$ is a Riemannian manifold.

In the implementation, we use the abstract class and simply override the `inner_product` function.

```
class HyperbolicMetric(RiemannianMetric):
    """Class for the hyperbolic metric."""

    def __init__(self, dim):
        super(HyperbolicMetric, self).__init__(
            dim=dim, signature=(1, dim))

    def metric_matrix(self, base_point=None):
        """Inner product matrix at base point."""
        diagonal = gs.array([-1.] + [1.] * self.dim)
        return from_vector_to_diagonal_matrix(diagonal)

    def inner_product(self, tangent_vec_a, tangent_vec_b, base_point=None):
        """Inner product between two tangent vectors at a base point."""
        diagonal = gs.array([-1.] + [1.] * self.dim)
        return gs.sum(diagonal * tangent_vec_a, tangent_vec_b, axis=-1)
```

Example 2.3.5: Frobenius metric

The analog of the Euclidean metric on matrix spaces is the *Frobenius* inner product defined by

$$\forall A, B \in M_{m,n}(\mathbb{R}), \quad \langle A, B\rangle_F = \mathrm{tr}(A^\top B) = \sum_{i,j} A_{ij} B_{ij}.$$

where tr is the trace operator. We use the right-hand-side expression in geomstats to avoid computing a matrix product ($O(mn^2)$ operations against $O(mn)$).

Endowed with this metric (or its restriction), $M_n(\mathbb{R}), GL(n), SO(n), SE(n)$ are Riemannian manifolds.

2.3.2 Affine connections and the Levi-Civita connection

A connection is an additional structure that can be defined independently of a Riemannian metric. It provides a way to compare tangent spaces from one point to another, by defining the notion of parallelism. For a detailed and historical account of the different approaches to defining connections, we refer to Marle 2005.

2.3.2.1 Connections

Definition 2.3.3 (Connection)**.** Let M be a smooth manifold. A *connection* on M is an $\mathbb{R}$-bilinear map $\nabla : \Gamma(TM) \times \Gamma(TM) \to \Gamma(TM)$ that verifies for all $X, Y \in \Gamma(TM), \forall f \in C^\infty(M)$:

1. (Linearity of 1st argument) $\nabla_{fX} Y = f \nabla_X Y$,
2. (Leibniz rule in 2nd argument) $\nabla_X (fY) = X(f)Y + f\nabla_X Y$.

The vector field $\nabla_X Y$ is called the *covariant derivative* of Y w.r.t. X.

In fact, $(\nabla_X Y)_p$ only depends on the value of X at p and not in its neighborhood. In contrast, it does depend on Y around p.

In local coordinates $(x^1, \dots, x^d)$, the *Christoffel symbols* are used to specify the connection. Recall that $\left(\partial_i\right)_i$ is a local frame, so we can decompose ∇ in this basis, and define $(\Gamma_{ij}^k)_{ijk}$ such that

$$\nabla_{\partial_i}\left(\partial_j\right) = \Gamma_{ij}^k \partial_k,$$

where we used Einstein summation convention, meaning that a sum occurs along the indices that appear both in subscript and superscript, here k.

Two vector fields $X, Y \in \Gamma(TM)$ can be decomposed locally in coordinates, writing $X = X^i \partial_i$ and $Y = Y^i \partial_i$ where X^i, Y^i are coordinate functions defined locally. Then using the properties of a connection,

$$\begin{aligned} \nabla_X Y &= X^i \nabla_{\partial_i}\left(Y^j \partial_j\right) \\ \nabla_X Y &= X^i \Big(\frac{\partial Y^j}{\partial x^i} \partial_j + Y^j \Gamma_{ij}^k \partial_j\Big). \end{aligned} \tag{2.5}$$

As this formula shows, a connection provides a correction term when compared to the directional derivative of Y with respect to X in a chart.

2.3.2.2 Parallel transport and geodesics

We now focus on vector fields that are defined along a curve $\gamma : [a, b] \to M$, i.e. a smooth map $X : [a, b] \to TM$ such that at any $t \in [a, b]$, $X(t) \in T_{\gamma(t)} M$. Note that such vector field need not be defined on the whole manifold, but can be locally extended to an open set around every point. Thankfully, one can show that there exists a *covariant derivative* that coincides with $(\nabla_{\dot\gamma(t)} X)_{\gamma(t)}$ at all $t \in [a, b]$ and for any X defined on a neighborhood of $\gamma(t)$ (Lafontaine, Gallot, et al. 2004, Theorem 2.68). We will skip these technicalities and admit that $\nabla_{\dot\gamma} X$ is well-defined for any vector field X along γ.

We now define the central notion of this manuscript: the parallel transport.

Definition 2.3.4 (Parallel vector field)**.** Let M be a smooth manifold and ∇ a connection on M. For any curve $\gamma : [a, b] \to M$ in M, a vector field X along γ is *parallel* if

$$\nabla_{\dot\gamma(t)} X(t) = 0. \tag{2.6}$$

In a local chart and using the Christoffel symbols and in particular using eq. (2.5), eq. (2.6) can be written $\forall t \in [a, b]$

$$\dot X^k(t) + \Gamma^k_{ij} X^i(t)\dot\gamma^j(t) = 0. \tag{2.7}$$

From the properties of ODEs, one can prove the following existence and uniqueness property

Proposition 2.3.1. *Let M be a smooth manifold and let ∇ be a connection on M. For every C^1 curve $\gamma : [a, b] \to M$ in M, for every $t \in [a, b]$ and every $v \in T_{\gamma(t)}M$, there is a unique parallel vector field X along γ such that $X(t) = v$.*

For such parallel vector field X and $s \in [a, b]$, we thus call $X(s)$ the *parallel transport* of v along γ from t to s, and write $X(s) = \Pi^s_{\gamma,t} v$. Another consequence of the properties of ODEs is that for any $s, t \in [a.b]$, $\Pi^s_{\gamma,t}$ is a linear isomorphism between the tangent spaces $T_{\gamma(t)}M$ and $T_{\gamma(s)}M$.

Intuitively, the parallel transport equation constrains the transported vector to keep a certain orientation w.r.t the speed $\dot\gamma$ of the curve while moving along it.

We now focus on a particular set of curves on M for which the velocity is parallel transported along the curve.

Definition 2.3.5. Let M be a smooth manifold endowed with a connection ∇. A curve $\gamma : [a, b] \to M$ is said to be *autoparallel* and is called a *geodesic* of (M, ∇) if it satisfies for all $t \in [a, b]$

$$\nabla_{\dot\gamma(t)} \dot\gamma(t) = 0. \tag{2.8}$$

Note that $\nabla_{\dot\gamma}\dot\gamma$ can be interpreted as the covariant acceleration of γ, so that equation (2.8) constrains geodesics to be zero acceleration curves. The flow of the geodesic equation is called geodesic flow, and is a fundamental example of dynamical system, that generalizes straight lines from Euclidean spaces.

The geodesic equation can be written in local coordinates like the parallel transport equation. A geodesic curve γ satisfies for all times $t \in [a, b]$

$$\ddot\gamma^k(t) + \Gamma^k_{ij}\dot\gamma^i(t)\dot\gamma^j(t) = 0. \tag{2.9}$$

From the properties of second-order differential equations, for any $(x, v) \in TM$, there exists a maximal interval $I_{x,v} \subseteq \mathbb{R}$ such that $\gamma_{x,v} : I_{x,v} \to M$ is the unique geodesic that verifies $\gamma_{x,v}(0) = x$ and $\dot\gamma_{x,v}(0) = v$. Moreover, by homogeneity of the equation (2.9), for any $s > 0$, $I_{x,sv} = \frac{1}{s} I_{x,v}$ and $\gamma_{x,sv}(t) = \gamma_{x,v}(st)$. We deduce that the set of vectors in T_xM such that $1 \in I_{x,v}$ is non empty, open, and contains 0. This leads to the following definition

Definition 2.3.6 (Exponential map)**.** Let ∇ be a connection on a smooth manifold M. The map $(x, v) \mapsto \gamma_{x,v}(1)$ defined on the open set $\{(x, v) \in TM, 1 \in I_{x,v}\}$ and with values in M is called the *exponential map* of ∇. We say that M is *geodesically complete* if the exponential map is defined on the whole of TM.

For any $x \in M$, we write the exponential map at x $\mathrm{Exp}_x : v \in T_xM \mapsto \gamma_{x,v}(1)$. The following properties of the exponential map can be proved.

Proposition 2.3.2. *Let ∇ be a connection on a smooth manifold M and $x \in M$.*

- *The differential of Exp_x at 0 is the identity.*
- *$(x, v) \mapsto (x, \mathrm{Exp}_x(v))$ is a smooth diffeomorphism from an open neighborhood of the null section (i.e. "$M \times \{0\}$") in TM to an open neighborhood of the diagonal of $M \times M$.*

Remark 2.3.2. *Note that although we introduced many notions in this paragraph, there were no examples. Indeed a connection or its Christoffel symbols are rarely explicit, except when the connection is compatible with a metric. We explain this notion of compatibility in the next paragraph, and will give several examples of geodesics and parallel transports.*

2.3.2.3 The Levi-Civita connection

First, given two vector fields Y, Z, recall that $\langle Y, Z\rangle$ is a smooth function on M so $X(\langle Y, Z\rangle)$ must be understood as $X(f) = df(X)$ for $f : p \mapsto \langle Y_p, Z_p\rangle_p$.

The following is considered the fundamental theorem of Riemannian geometry. It ensures that there exists a unique connection that is "compatible" with the metric.

Theorem 2.3.2. *Let (M, g) be a Riemannian manifold. There is a unique connection on M that verifies for all $X, Y, Z \in \Gamma(TM)$*

$$\text{1. (Torsion-free)} \quad \nabla_X Y - \nabla_Y X = [X, Y] \tag{2.10}$$

$$\text{2. (Compatibility)} \quad X(\langle X, Y\rangle) = \langle \nabla_X Y, Z\rangle + \langle Y, \nabla_X Z\rangle \tag{2.11}$$

This connection is called the Levi-Civita connection and is determined by the Koszul formula:

$$\begin{aligned} 2\langle \nabla_X Y, Z\rangle = {} & X(\langle Y, Z\rangle) + Y(\langle X, Z\rangle) - Z(\langle X, Y\rangle) \\ & - \langle Y, [X, Z]\rangle - \langle X, [Y, Z]\rangle - \langle Z, [Y, X]\rangle. \end{aligned} \tag{2.12}$$

The notion of compatibility is thus detailed in equations (2.10) and (2.11). The former is quite general but ensures unicity of the Levi-Civita connection, it is called the zero-torsion condition. The latter can be understood as a Leibniz rule where $\langle Y, Z\rangle$ is seen as a product and the derivative is ∇_X. More precisely, this condition means that the metric is parallel with respect to the connection. Indeed, although we introduced a connection as a map on vector fields, it can be extended to tensors of any order. For a 0-order tensor, i.e. a smooth function f, we have $\nabla_X f = X(f)$,

and for a (2, 0)-tensor such as g, we have (by generalising the Leibniz rule) for any $X, Y, Z \in \Gamma(TM)$

$$\begin{aligned}(\nabla_X g)(Y,Z) &= \nabla_X(g(Y,Z)) - g(\nabla_X Y, Z) - g(Y, \nabla_X Z). \\ &= X(\langle Y,Z\rangle) - \langle \nabla_X Y, Z\rangle - \langle Y, \nabla_X Z\rangle \end{aligned} \tag{2.13}$$

Therefore the combination of (2.11) and (2.13) results in $\nabla g = 0$.

In the general case, the Levi-Civita connection is characterized locally by its Christoffel symbols. Let $(x^1, \dots, x^n)$ be a local coordinate chart, by definition

$$\nabla_{\partial_i} \partial_j = \Gamma_{ij}^k \partial_k$$

The torsion-free condition (2.10) and Schwartz theorem ($[\partial_i, \partial_j] = 0$) thus imply the symmetry of the symbols: $\Gamma_{ij}^k = \Gamma_{ji}^k$ for all $1 \leq i, j, k \leq n$. Furthermore, together with (2.11), Koszul formula (2.12) translates into

$$2g(\nabla_{\partial_i}\partial_j, \partial_k) = \partial_i g_{jk} + \partial_j g_{ki} - \partial_k g_{ij}.$$

Writing $(g^{ij})_{ij} = (g_{ij})_{ij}^{-1}$ for the inverse of the metric matrix, we obtain

$$\Gamma_{ij}^k = \frac{1}{2} g^{lk}(\partial_i g_{jl} + \partial_j g_{li} - \partial_l g_{ij}) \tag{2.14}$$

Thus the Christoffel symbols can be computed from the metric g. This formula is rarely used by mathematicians for computations "by hand", but we shall use it in geomstats to implement pull back metrics.

The following characterization allows to compute the Levi-Civita connection in the case of embedded manifolds equipped with the embedding metric, as is the case in many of our examples.

Proposition 2.3.3. *Let M be a Riemannian manifold and N a submanifold of M endowed with the induced metric (i.e., the restriction of the embedding metric). If ∇^M and ∇^N are the Levi-Civita connections on M and N respectively defined by the metric on M. Then for any vector field $X, Y \in \Gamma(TN)$, we have*

$$\nabla_X^N Y = (\nabla_X^M Y)^{\|}, \tag{2.15}$$

where $(v)^{\|}$ is the orthogonal projection of any $v \in T_pM$ onto T_pN for any $p \in N$.

This proposition is particularly useful for embedded manifolds of $\mathbb{R}^N$ endowed with the ambient Euclidean metric, as it is straightforward to see (by unicity) that their Levi-Civita connection coincide with the directional derivative of functions from $\mathbb{R}^N$ to $\mathbb{R}^N$. Therefore, one only needs to compute the projection to the tangent spaces of any point of an embedded manifold to compute its connection.

Finally, thanks to the compatibility of the metric with the connection, the parallel transport preserves the metric, as stated in the following proposition.

Proposition 2.3.4. *Let $\gamma : I \to M$ be a smooth curve, and $s, t \in I$. Then the parallel transport map $\Pi^t_{\gamma,s} : T_{\gamma(s)}M \to T_{\gamma(t)}M$ along γ for the Levi-Civita connection is an isometry, i.e.*

$$\forall v, w \in T_{\gamma(s)}M, \quad \langle \Pi^t_{\gamma,s} v, \Pi^t_{\gamma,s} w \rangle_{\gamma(t)} = \langle v, w \rangle_{\gamma(s)}.$$

This justifies the use of parallel transport in statistical procedures to move data from one reference point to another while preserving distances.

Moreover, the compatibility with the metric also ensures that isometries preserve the Levi-Civita connection and its geodesics. The following can be deduced from the Kozsul formula (2.12).

Proposition 2.3.5. *Let (M, g) be a Riemannian connection and ∇ its Levi-Civita connection. Let $f \in \mathrm{Isom}(M)$ be an isometry of (M, g), then for any $X, Y \in \Gamma(TM)$ and $x \in M$, we have*

$$df \nabla_X Y = \nabla_{df X} df Y \tag{2.16}$$

$$\mathrm{Exp}_{f(x)} \circ df_x = f \circ \mathrm{Exp}_x \,. \tag{2.17}$$

We now come to the definition of distance and its link with geodesics.

2.3.3 Distance and Geodesics

Let (M, g) be a Riemannian manifold. As the Levi-Civita is uniquely defined, we call geodesic of (M, g) any geodesic of its Levi-Civita connection. Similarly, we define:

Definition 2.3.7 (Exp and Log maps)**.** We call *Riemannian exponential* the exponential map of the Levi-Civita connection, and define its *injectivity radius* $\mathrm{inj}_M(x)$ at x as the greatest $\epsilon > 0$ such that Exp is a diffeomorphism on the open ball of radius ϵ of $T_x M$. It may not be finite, as in e.g. the hyperbolic space.

The *Riemannian logarithm* is defined as the inverse of the Exp map on the *injectivity domain* $\mathrm{Inj}(x) \subseteq T_x M$, where the Exp map is injective, that contains the open ball of radius $\mathrm{inj}_M(x)$ but may be a larger star-shaped open set in general. The injectivity radius of M, inj_M, is then the smallest of all injectivity radii at x for $x \in M$.

By (2.11), the *energy* $E(t) = \frac{1}{2}\|\dot\gamma(t)\|^2$ of any geodesic γ is constant:

$$\frac{d}{dt}\langle \dot\gamma(t), \dot\gamma(t) \rangle = 2\langle \nabla_{\dot\gamma(t)} \dot\gamma(t), \dot\gamma(t) \rangle = 0$$

We say that a curve γ is parametrized with constant speed if the function $t \mapsto \|\dot\gamma(t)\|_{\gamma(t)}$ is constant. The above equation thus shows that geodesics are parametrized with constant speed.

Furthermore, if $N \subset M$ is an embedded manifold, (2.15) implies that the geodesics of N are the curves in M whose acceleration is normal to N.

Example 2.3.6: Hypersphere

For any $x \in S^d \subset \mathbb{R}^{d+1}$, and $v \in T_x S^d$, define the curve on S^d that parametrizes the great circle

$$\gamma : t \mapsto \cos(t\|v\|)x + \sin(t\|v\|)\frac{v}{\|v\|}. \tag{2.18}$$

Then, it is clear that the acceleration $\ddot{\gamma}(t) = -\|v\|^2\gamma(t)$ is normal to S^d. By unicity this curve is the geodesic from x with initial velocity v.

The Exp map at x, $v \mapsto \cos(t\|v\|)x + \sin(t\|v\|)\frac{v}{\|v\|}$ is thus well defined and is a diffeomorphism from the open ball of radius π to $S^d \setminus \{-x\}$, i.e. onto the entire sphere except the antipodal point of x. Its inverse is defined for any $y \notin \{x, -x\} \in S^d$ by

$$\mathrm{Log}_x(y) = \arccos(\langle y, x\rangle)\frac{y - \langle y, x\rangle x}{\|y - \langle y, x\rangle x\|}. \tag{2.19}$$

We implement the Riemannian Exp and Log maps in the `HypersphereMetric`.

Implementation trick 2.3. *To improve numerical stability around 0, we use a Taylor approximation of the sinc and cosine functions for inputs smaller than 10^{-6}. As both functions are even, the squared norm of v is computed and taking the square-root is not necessary.*

For a C^1-curve $\gamma : [a, b] \to M$, we now define its *length* L and total energy E (also called *action integral* in physics) by:

$$L(\gamma) = \int_a^b \|\dot{\gamma}(t)\|dt, \qquad E(\gamma) = \int_a^b \|\dot{\gamma}(t)\|^2 dt$$

Note that the length does not depend on the parametrization of the curve, while the energy does. Indeed, moving along a path from a to b does not require the same energy if the speed is increased, but the length remains the same. This finally leads to the definition of distance.

Definition 2.3.8 (Riemannian distance)**.** Let (M, g) be a Riemannian manifold, and $x, y \in M$. The *Riemannian distance* between x and y is the lower bound of the lengths of all piecewise smooth curves joining x to y:

$$d(x, y) = \inf\{L(\gamma) \mid \gamma : I \to M \text{ piecewise } C^1,\ \gamma(0) = x,\ \gamma(1) = y\}.$$

We say that γ is *minimizing* if $L(\gamma) = d(x, y)$.

If M is connected, this distance function is indeed a distance, and the induced topology coincides with that of M (Paulin 2014, Proposition 3.13). We say that a curve $\gamma : [a, b] \to M$ is parametrized by (or resp. proportional to) arc-length if $L(\gamma) = b - a$ (resp. $\exists \lambda > 0, L(\gamma) = \lambda(b - a)$). We now see that the geodesics of

(M, g) are the minimizing curves, and further minimize the total energy (Paulin 2014, Proposition 3.14).

Theorem 2.3.3. *Let (M, g) be a Riemannian manifold, and $\gamma : [a, b] \to M$ a C^1 curve. The following assertions are equivalent*

- *γ is a geodesic;*
- *γ is parametrized with constant velocity and locally minimizing;*
- *γ is locally energy minimizing.*

In particular, as for any $(x, v) \in TM$, the curve $\gamma : t \in \mathrm{Inj}(x) \mapsto \mathrm{Exp}_x(tv)$ is a geodesic, it is locally length-minimizing, so that for any $y \in \mathrm{Exp}_x(\mathrm{Inj}(x))$, $d(x, y) = L(\gamma) = \|v\|$. By definition, $v = \mathrm{Log}_x(y)$, so that

$$d(x, y) = \|\mathrm{Log}_x(y)\|.$$

We add this as a method to the `RiemannianMetric` class of geomstats.

```
def dist(self, point_a, point_b):
        """Riemannian distance between two points"""
        log = self.log(point_b, base_point=point_a)
        return self.norm(log, base_point=point_a)
```

Furthermore, the last point of Theorem 2.3.3 is a variational principle, i.e. the minimization of a function on the space of paths.

Recall from Definition 2.3.6 that a Riemannian manifold is called geodesically complete if the Exp map is defined on the whole of TM, meaning that geodesics are defined on all $\mathbb{R}$. On the other hand, the more usual notion of completeness of a metric space is defined as follows:

Definition 2.3.9 (Complete metric space)**.** A metric space is called complete if every Cauchy sequence converges in that space.

Example 2.3.7: Half-plane

Let the half-plane $P = \{(x, y) \in \mathbb{R}^2 \mid y > 0\}$ be equipped with the canonical Euclidean metric of $\mathbb{R}^2$. Then P is obviously not geodesically complete, as geodesics are straight lines exiting P.

Now define the metric $g(x, y) = \frac{dx^2+dy^2}{y^2}$. (P, g) is now a geodesically complete Riemannian manifold.

The following theorem gives a sufficient condition for Riemannian manifolds to share the same nice properties as complete metric spaces.

Theorem 2.3.4 (Hopf-Rinow)**.** *Let (M, g) be a Riemannian manifold. If M is geodesically complete, then any two points of M can be joined by a minimizing geodesic.*

Corollary 2.3.1. *Let (M, g) be a connected Riemannian manifold, and d the Riemannian distance. The following are equivalent*

- *(M, g) is geodesically complete;*
- *every closed and bounded subset of (M, d) is compact;*
- *the metric space (M, d) is complete (as a metric space).*

Remark 2.3.3 (Infinite dimension)**.** *The Hopf-Rinow theorem is not true in infinite dimension. Counter-examples have been constructed and exhibit manifolds that are complete as metric spaces but for which points exist that cannot be joined by a minimizing geodesic, or even by any geodesic at all. For more details, see Atkin 1975.*

In the previous example of the hypersphere, geodesics are great circles and are known in closed-form. To finish this paragraph, we detail our implementation of the Riemannian distance when no closed form solution is available for the geodesics. In the most general case, recall that geodesics are defined by an ODE expressed either in a chart or in the ambient space (e.g. (2.9)). Discrete integration methods can thus be used to approximate the exponential map, and optimization algorithms to compute the logarithm. Inspired by Kühnel, Sommer, and Arnaudon 2019, we implemented those methods in geomstats.

Consider the geodesic equation as a coupled system of first-order ODEs:

$$\begin{cases} v(t) &= \dot{\gamma}(t) \\ \dot{v}(t) &= f(v(t), \gamma(t), t) \end{cases} \tag{2.20}$$

where f is a smooth function. A method of the class `Connection` is used to compute the right-hand-side of (2.20), called `geodesic_equation` where the `state` variable is $(\gamma(t), \dot{\gamma}(t))$:

```
def geodesic_equation(self, state, time):
    """Return the right-hide-side of the geodesic equation."""
    position, velocity = state
    gamma = self.christoffels(position)
    equation = -gs.einsum('...kij,...i,...j->...k', gamma, velocity, velocity)
    return gs.stack([velocity, equation])
```

Given initial conditions, a first-order forward Euler scheme, or higher-order Runge-Kutta method can be used to integrate this system.

```
STEP_FUNCTIONS = {'euler': 'euler_step',
                  'rk2': 'rk2_step'}

def euler_step(force, state, time, dt):
    """Compute one step of the Euler approximation."""
    derivatives = force(state, time)
    new_state = state + derivatives * dt
    return new_state
```

```
def rk2_step(force, state, time, dt):
    """Compute one step of the rk2 approximation."""
    k1 = force(state, time)
    k2 = force(state + dt / 2 * k1, time + dt / 2)
    new_state = state + dt * k2
    return new_state

def integrate(
        function, initial_state, end_time=1.0, n_steps=10, step='euler'):
    """Compute the flow of a vector field."""

    dt = end_time / n_steps
    states = [initial_state]
    step_function = globals()[STEP_FUNCTIONS[step]]

    current_state = initial_state
    for i in range(n_steps):
        current_state = step_function(
            state=current_state, force=function, time=i * dt, dt=dt)
        states.append(current_state)
    return states
```

We can thus add the following method to the class `Connection` of geomstats to compute the exponential map

```
    def exp(self, tangent_vec, base_point, n_steps=10, step='euler', **kwargs):
        """Exponential map associated to the affine connection."""
        initial_state = gs.stack([base_point, tangent_vec])
        flow = integrate(
            self.geodesic_equation, initial_state, n_steps=n_steps, step=step)
        exp, velocity = flow[-1]
        return exp
```

On the other hand, the Log map corresponds to a boundary value problem (BVP), and an optimization procedure is required. We choose the *geodesic shooting* method, that solves the following problem ($\mathcal{P}$) which corresponds to an energy minimization. Working in a convex neighborhood, it admits a unique minimizer v^*:

$$\min d^2(\mathrm{Exp}_x(v), y) \tag{$\mathcal{P}$}$$

As in a coordinate chart or in the embedding space, all norms are equivalent, we use the appropriate Euclidean metric (written $\|.\|_2$) for practical purpose.

$$\|\mathrm{Exp}_x(v) - y\|_2^2. \tag{$\mathcal{P}'$}$$

and the problem is solved by gradient descent (GD) until a convergence tolerance ϵ is reached. We use the `minimize` function from the `scipy` function, and use automatic differentiation to compute the gradient of the exponential map.

```
def log(self, point, base_point, n_steps=N_STEPS, step='euler',
        max_iter=25, verbose=False, tol=gs.atol):
    """Compute logarithm map associated to the affine connection."""
    max_shape = point.shape if point.ndim > base_point.ndim else \
        base_point.shape

    def objective(velocity):
        """Define the objective function."""
        velocity = gs.array(velocity, dtype=base_point.dtype)
        velocity = gs.reshape(velocity, max_shape)
        delta = self.exp(velocity, base_point, n_steps, step) - point
        return gs.sum(delta ** 2)

    objective_with_grad = gs.autograd.value_and_grad(objective)
    tangent_vec = gs.flatten(gs.random.rand(*max_shape))
    res = minimize(
        objective_with_grad, tangent_vec, method='L-BFGS-B', jac=True,
        options={'disp': verbose, 'maxiter': max_iter}, tol=tol)

    tangent_vec = gs.array(res.x, dtype=base_point.dtype)
    tangent_vec = gs.reshape(tangent_vec, max_shape)
    return tangent_vec
```

2.3.4 Curvature

In a Euclidean space, a constant field is parallel along any curve. In a Riemannian manifold in general, parallel transport depends on the curve followed, and there may not exist fields that are parallel along all curves, not even localy. One can investigate the effect of parallel transport along small closed curves. Consider a point $x \in M$ and the closed curves whose tangent velocities at that point span a subspace of dimension two. They introduce a deviation of parallel transport from the identity map of the tangent space at that point, and this deviation can be shown to depend only on a basis of this plane. This is due to the commuting properties of the covariant derivative, i.e. the difference between evaluating ∇_X after ∇_Y and vice-versa. This difference is not sufficient however to define a tensor. The following lemma gives a sufficient condition for a map to define a tensor on a manifold (Lafontaine, Gallot, et al. 2004, Proposition 1.114).

Lemma 2.3.1 (Tensoriality). *Let $p \in \mathbb{N}$ and $A : \Gamma(TM)^p \to \Gamma(TM)$ be a $C^\infty(M)$-multilinear map, i.e. $\forall f_1, \dots, f_p \in C^\infty(M), \forall X_1, \dots X_p \in \Gamma(TM)$, $A(f_1 X_1, \dots f_p X_p) = f_1 \dots f_p A(X_1, \dots, X_p)$. Then for any $x \in M$, the value of $A(X_1, \dots, X_p)$ at x only depends on the value of the X_i's at x.*

Simple computations using the Leibniz rule from the definition of a connection (2) show that the following defines a C^∞-multilinear map.

Definition 2.3.10 (Curvature tensor). Let (M, ∇) be a manifold equipped with a connection. The *curvature* tensor of (M, ∇) is defined as the map from $\Gamma(TM)^3$ to

$\Gamma(TM)$ by

$$R(X,Y)Z = \nabla_X \nabla_Y Z - \nabla_Y \nabla_X Z - \nabla_{[X,Y]} Z \tag{2.21}$$

The curvature tensor of a Riemannian manifold is the curvature of its Levi-Civita connection.

In the above definition, the arguments X, Y, Z are vector fields, but the tensoriality Lemma 2.3.1 allows to write at any $x \in M$, R_x as a map defined on tangent vectors $u, v, w \in T_x M$.

Example 2.3.8: Euclidean space

In a Euclidean space $\mathbb{R}^d$ with the canonical inner-product, the connection coincides with the directional derivative, which commutes. Therefore $R = 0$.

The following properties help cope with the complexity of this tensor.

Proposition 2.3.6. *Let (M, g) be a Riemannian manifold and R its curvature tensor. The following properties hold*

1. *(Skew-symmetry) The map $(X, Y) \in \Gamma(TM)^2 \mapsto R(X, Y)$ is skew-symmetric.*
2. *(Bianchi's identity) For any $X, Y, Z \in \Gamma(TM)$, we have*
$$R(X,Y)Z + R(Y,Z)X + R(Z,X)Y = 0.$$
3. *(Bianchi's second identity) For any $X, Y, Z \in \Gamma(TM)$, we have*
$$\nabla_X R(Y,Z) + \nabla_Y R(Z,X) + \nabla_Z R(X,Y) = 0.$$
4. *For any isometry $f \in \mathrm{Isom}(M)$ and $X, Y, Z \in \Gamma(TM)$, $R(df X, df Y)(df Z) = df\big(R(X,Y)Z\big)$.*
5. *For all $X, Y \in \Gamma(TM)$, the field of linear maps $R(X, Y)$ is skew-symmetric with respect to the metric g, i.e. $\forall W, Z \in \Gamma(TM)$*
$$g(R(X,Y)Z, W) = -g(Z, R(X,Y)W)$$
6. *For all $X, Y, Z, W \in \Gamma(TM)$, we have*
$$g(R(X,Y)Z, W) = g(R(Z,W)X, Y) = g(R(W,Z)Y, X)$$

Note that the compatibility of the metric (equations (2.11), (2.10)) are required for the last two assertions. These allow to define the *sectional* curvature for any $x \in M$ and $u, v \in T_x M$ such that u, v are not collinear

$$\kappa_x(u,v) = \frac{\langle R_x(u,v)v, u\rangle}{\|u\|^2\|v\|^2 - \langle u, v\rangle^2} \tag{2.22}$$

From the above properties, the value of κ_x in fact only depends on the plane spanned by (u, v), i.e. for any non-vanishing linear combination $\alpha u + \beta v$, $\kappa(\alpha u + \beta v, v) = \kappa(u, v)$. In fact, this quantity is enough to characterize the curvature of M. Indeed, one can prove from algebraic computations only (see e.g. Kobayashi and Nomizu 1996a, Chapter V, Proposition 1.2) the following

Lemma 2.3.2. *Let two quadrilinear mappings A, B defined on a vector space V that both verify the properties $\forall u, v, w, z \in V$*

$$A(u,v,w,z) = -A(v,u,w,z) = -A(v,u,z,w)$$
$$A(u,v,w,z) + A(u,w,z,v) + A(u,z,v,w,) = 0$$

and coincide for any two variables $u, v \in V$ in the sense: $A(u,v,u,v) = B(u,v,u,v)$. Then $A = B$.

This applies to the map $(u,v,w,z) \mapsto g(R(w,z)v,u)$ and one can thus show the following theorem.

Theorem 2.3.5. *The sectional curvature determines the curvature tensor.*

In particular, if for every $x \in M$, κ has a constant value κ_x on every planes of T_xM, then we can compute the curvature up to this constant. Indeed from the definition (2.22) we have for any $u, v \in T_xM$

$$\langle R(u,v)v,u\rangle = \kappa_x\big(\|u\|^2\|v\|^2 - \langle u,v\rangle^2\big)$$

Consider the maps $A : (u,v,w,z) \mapsto \langle R(u,v)w,z\rangle$ and B defined by

$$B(u,v,w,z) = \big(\langle u,w\rangle\langle v,z\rangle - \langle u,z\rangle\langle v,w\rangle\big)$$

It is clear that the hypotheses of Lemma 2.3.2 are verified for A and $\kappa_x B$, so we conclude

$$\langle R(u,v)w,z\rangle = \left(\langle u,w\rangle\langle v,z\rangle - \langle u,z\rangle\langle v,w\rangle\right)\kappa_x. \tag{2.23}$$

For example, eq. (2.23) is always valid in dimension $d = 2$, as there is only one plane in each tangent space. We now state the surprising result of F. Schur that gives a sufficient condition for formula (2.23) to hold.

Theorem 2.3.6 (Schur)**.** *Let (M,g) be a connected Riemannian manifold of dimension d. If $d \geq 3$ and if the sectional curvature κ does not depend on the plane but only on the point $x \in M$, then κ is constant (i.e. does not even depend on x).*

Example 2.3.9: Hypersphere and hyperboloid

Recall that the rotation group $SO(d+1)$ acts on the hypersphere S^d, and suppose that $d \geq 2$. The stabilizer of this action on any $x \in S^d$ is isomorphic to $SO(d)$, whose action on T_xS^d is transitive on 2-planes, i.e. any 2-plane can be mapped by a rotation to any other 2-plane. By assertion 4 of Proposition 2.3.6, the sectional curvature is preserved from one plane to another by rotations, it is therefore constant on the whole T_xM, write this value κ_x. If $d \geq 3$, we can use Theorem 2.3.6 to conclude it is constant on S^d. Of course, it is also constant for $d = 2$, as we can use an isometry to map the curvature tensor from one point to any other.

The same argument, using this time the group $O(1,d)$ of isometries of the

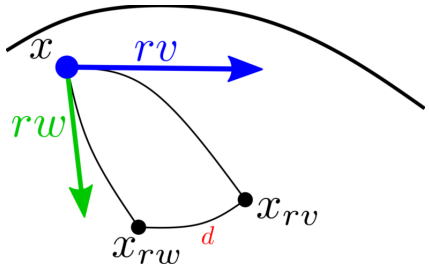


Figure 2.4: Drawing of the distortion of geodesics compared to Euclidean tangent vectors. Arrows represent tangent vectors, thin black lines represent geodesics and we use the notation $x_v = \mathrm{Exp}_x(v)$. Curvature modifies the geodesic distance d compared with the distance between rv and rw in T_xM (adapted from Paulin 2014, Section 3.6.3)

.

Lorentz product, that acts transitively on the hyperboloid H^d_+, shows that this space also has constant curvature.

Definition 2.3.11 (Constant curvature)**.** Let (M, g) be a Riemannian manifold. (M, g) is said to have *constant* (resp. *negative*, resp. *positive*) curvature if it has constant (resp. negative, resp. positive) sectional curvature.

In fact, the (complete simply connected) constant curvature spaces are all isometric to one of the Euclidean space (flat), the hypersphere (positive curvature) or the hyperbolic space (negative curvature) (see e.g . Lafontaine, Gallot, et al. 2004, Theorem 3.82). These three examples thus describe the entire class of constant curvature spaces (up to covering and isometry).

To finish this section, we may now explain how curvature modifies the correspondence between distances in tangent spaces and Riemannian distances on a manifold. Indeed, let (M, g) be a Riemannian manifolds and consider two orthogonal tangent vectors u, v at some $x \in M$. In the vector space T_xM, the distance (induced by the metric at x) between rv and rw for some $r > 0$ is of course $r\sqrt{2}$. Now map these vectors to the manifold using the exponential map, then the Riemannian distance between $\mathrm{Exp}_x(rv)$ and $\mathrm{Exp}_x(rw)$ may increase or decrease compared to the distance in the tangent space. Curvature is the fundamental tool that allows to quantify these variations, and the sign of the sectional curvature tells whether geodesics accumulate or grow apart (Figure 2.4).

Theorem 2.3.7. *Let (M, g) be a Riemannian manifold, R its curvature tensor, κ its sectional curvature, $x \in M$ and γ_v, γ_w two geodesics starting from x with initial velocities $v, w \in T_xM$. Then for $r \to 0$,*

$$d(\gamma_v(r), \gamma_w(r))^2 = r^2\|v - w\|^2 - \frac{r^4}{3}\langle R(v, w)w, v\rangle + O(r^5). \tag{2.24}$$

Moreover, if v, w are orthonormal

$$d(\gamma_v(r), \gamma_w(r)) = \sqrt{2}r\left(1 - \frac{\kappa(v,w)}{12}r^2\right) + O(r^4). \tag{2.25}$$

We now understand the importance of the sign of the sectional curvature: if $\kappa > 0$, geodesics get closer to one another, while if $\kappa < 0$, they grow apart. Using this theorem and formulas for geodesics on the hypersphere in Example 2.3.6, we can compute a Taylor expansion of $d(\gamma_v, \gamma_w)$ and identify the coefficients, to find that $\kappa = 1$. Similarly, for the hyperbolic space $\kappa = -1$. In chapter 4 we will derive similar Taylor expansions using the curvature tensor to characterise the defect of parallelism when constructing geodesic parallelograms.

Finally, we state two important theorems that give a hint about how the sign of the curvature determines the geometry. These theorems also show how curvature and topology may be intertwined.

Theorem 2.3.8 (Cartan-Hadamard)**.** *Let (M, g) be a complete connected Riemannian manifold with non-positive sectional curvature. Then the exponential map is a Riemannian covering, i.e. a covering that is a local isometry. This means that if M is simply connected, then M is diffeomorphic to $\mathbb{R}^d$, and any two points are joined by a unique minimizing geodesic.*

Remark 2.3.4 (Infinite dimension)**.** *This theorem carries over to infinite dimensional Riemannian Hilbert manifolds, i.e. whose atlases are valued in a Hilbert space (McAlpin 1965).*

On the contrary:

Theorem 2.3.9 (Particular case of Myers' theorem)**.** *Let (M, g) be a complete connected Riemannian manifold with sectional curvature bounded below $\kappa > \frac{1}{r^2}$ for some $r > 0$. Then the diameter of M (i.e. the largest distance between points in M) is bounded above by πr and M is compact.*

This ends our exposition of Riemannian metrics. In the next section, we focus on a particular class of manifolds that play a fundamental role in geometry and in many applications.

2.4 Lie groups

We first define the notions of Lie groups, subgroups and algebras, then the exponential map of Lie groups, and finally introduce smooth actions of Lie groups and homogeneous spaces. The metrics that can be defined on Lie groups, and their implementation in geomstats will be covered in chapter 3. We restrict to finite-dimensional Lie groups and only give a few remarks about infinite dimension.

2.4.1 Lie groups, Lie algebras and Lie subgroups

Definition 2.4.1 (Lie group)**.** A Lie group is a group $(G, \cdot)$ such that G is also a finite dimensional smooth manifold, and the group and differential structures are compatible, in the sense that the group law $\cdot$ and the inverse map $g \mapsto g^{-1}$ are smooth.

We generally omit $\cdot$ in the notation and simply write the group composition as a multiplication. Let e denote the neutral element, or identity of G.

Remark 2.4.1 (Infinite dimension)**.** *Infinite dimensional groups with a smooth structure appear naturally, for example as the set of diffeomorphisms of a smooth manifolds. However, although some properties of finite dimensional Lie groups generalise to infinite dimension, many don't. See Milnor 1984 for a good treatment of the topic. Unless otherwise specified, we restrict our exposition to finite dimension.*

For any $g \in G$, we define the *left* and *right translation* maps L_g and R_g by

$$L_g : h \in G \mapsto gh \in G \qquad R_g : h \in G \mapsto hg^{-1}.$$

By the definition of a Lie group, L_g and R_g are diffeomorphisms of G and their differential are linear isomorphisms of tangent spaces. This means that all the tangent spaces of G are identical to T_eG, and there is a canonical way of mapping vectors from T_gG to T_eG, by $L_{g^{-1}}$ for any $g \in G$.

This fact, that could be rephrased as "the tangent bundle of a Lie group is trivial", means that TG is diffeomorphic to the product $G \times T_eG$, and is of fundamental importance to characterize the geometry of a Lie group, and is very handy for the implementation. Furthermore, the maps $g \mapsto L_g$ and $g \mapsto R_g$ are group homomorphisms between $(G, \cdot)$ and $(\mathrm{Diff}(G), \circ)$, the group of diffeomorphisms of G with the composition of maps as group law.

Definition 2.4.2 (Invariant vector field)**.** A vector field $X \in \Gamma(TG)$ is *left-invariant* if $\forall g, h \in G$,

$$X_{gh} = dL_g X_h$$

Let $\mathcal{L}(G)$ denote the set of all left-invariant vector fields.

Example 2.4.1: General Linear group

The real general linear group $GL(n)$ plays a fundamental role as all the groups implemented in geomstats and encountered in the applications are subgroups of $GL(n)$. It is defined as the set of invertible matrices of size n

$$GL(n) = \{A \in M_n(\mathbb{R}) \mid \det(A) \neq 0\},$$

and as an open set of the vector space $M_n(\mathbb{R}) \simeq \mathbb{R}^{n^2}$ it is a smooth manifold.

The group law is the matrix multiplication, that can be written as a polynomial of the matrices coefficients and is thus smooth. Thanks to Cramer's formula $A^{-1} = \det(A)^{-1} Co(A)^T$, where Co denotes the matrix formed by all

cofactors, the inversion map is also smooth. Therefore, $GL(n)$ is a Lie group.

Definition 2.4.3 (Lie algebra)**.** A real *Lie algebra* is a real vector space $\mathfrak{g}$ equipped with a bilinear map $[\cdot, \cdot]$ that verifies

- (Skew-symmetry) $\forall x, y \in \mathfrak{g}, \quad [x, y] = -[y, x]$,
- (Jacobi identity) $\forall x, y \in \mathfrak{g}, \quad [x, [y, z]] + [y, [z, x]] + [z, [y, x]] = 0$.

Example 2.4.2: Vector fields

The space of vector fields $\Gamma(TM)$ of a manifold M equipped with the Lie bracket of vector field defined in (2.2) p. 22 is an infinite dimensional Lie algebra.

Example 2.4.3: Matrix algebra

The algebra of squared matrices $M_n(\mathbb{R})$ equipped with the commutator

$$\forall A, B \in M_n(\mathbb{R}), \quad [A, B] = AB - BA$$

is a Lie algebra.

In geomstats As the Lie groups we are working with are all matrix groups, we implement an abstract class `MatrixLieGroup` to gather the properties of a Lie group: its identity, composition law, translation map, exponential and logarithm maps. Morever, it is useful to have the Lie algebra implemented as a separate class and set as an attribute of the Lie group. This class is a subclass of `VectorSpace` and implements a `belongs` and `projection` method. This allows for example to write the methods `is_tangent` and `to_tangent` as follows:

```
def tangent_translation(self, point, left_or_right='left', inverse=False):
    """Return the differential map of the right or left translation map."""
    point_ = self.inverse(point) if inverse else point
    if left_or_right == 'left':
        return lambda tan_vec: self.compose(point_, tan_vec)
    return lambda tan_vec: self.compose(tan_vec, point_)

def is_tangent(self, vector, base_point=None, atol=gs.atol):
    """Check whether a vector is tangent at base point."""
    if base_point is None:
        base_point = self.identity

    if gs.allclose(base_point, self.identity):
        tangent_vec_at_id = vector
    else:
        tangent_vec_at_id = self.tangent_translation(
```

```
            base_point, inverse=True)(vector)
    return self.lie_algebra.belongs(tangent_vec_at_id, atol)

def to_tangent(self, vector, base_point=None):
    """Project a vector onto the tangent space at a base point."""
    if base_point is None:
        return self.lie_algebra.projection(vector)
    tangent_vec_at_id = self.tangent_translation(
        base_point, inverse=True)(vector)
    projected = self.lie_algebra.projection(tangent_vec_at_id)
    return self.tangent_translation(base_point)(projected)
```

Example 2.4.4: GeneralLinear

The general linear group (Example 2.4.1) is the archetype of matrix Lie group, and thus created as a subclass of `MatrixLieGroup`. It is defined as an open set of the matrix space, so it also inherits from `OpenSet`.

```
class GeneralLinear(MatrixLieGroup, OpenSet):
    """Class for the general linear group GL(n)."""

    def __init__(self, n, **kwargs):
        if 'dim' not in kwargs.keys():
            kwargs['dim'] = n ** 2
        super(GeneralLinear, self).__init__(
            ambient_space=Matrices(n, n), n=n, **kwargs)

    def belongs(self, point, atol=gs.atol):
        """Check if a matrix is invertible and of size n."""
        has_right_size = self.ambient_space.belongs(point)
        if gs.all(has_right_size):
            det = gs.linalg.det(point)
            return gs.abs(det) > atol
        return has_right_size

    def projection(self, point):
        """Project a matrix to the general linear group."""
        belongs = self.belongs(point)
        regularization = gs.einsum(
            '...,ij->...ij', gs.where(~belongs, gs.atol, 0.), self.identity)
        projected = point + regularization
        return projected
```

Theorem 2.4.1 (Left-invariant vector fields)**.** *Let G be a Lie group of finite dimension d.*

(1) The space of left-invariant vector fields $\mathcal{L}(G)$ of a Lie group G is a sub-algebra of $\Gamma(TG)$. This means that the bracket of two left-invariant vector fields is also left-invariant.

(2) As a vector space, $\mathcal{L}(G)$ is isomorphic to T_eG, the tangent space at the identity of G. This implies that $\mathcal{L}(G)$ is finite dimensional.

Indeed, the map $X \in \mathcal{L}(G) \mapsto X_e \in T_eG$ is linear and has inverse $x \mapsto \tilde{x}$ where we define $\tilde{x}$ to be the left-invariant vector field

$$\tilde{x} : g \in G \longmapsto dL_g x \in T_gG.$$

We can thus define the bracket on T_eG: $[x, y] \triangleq [\tilde{x}, \tilde{y}]_e$, which turns T_eG into a Lie algebra that is isomorphic (as Lie algebras) to $\mathcal{L}(G)$. We call T_eG *the* Lie algebra of G and denote it $\mathfrak{g} = T_eG$.

Thankfully, a linear representation of G allows to compute the Lie bracket without requiring the handling of vector fields. Indeed, define the conjugation map of G, for any $g \in G$ by $C_g : h \mapsto ghg^{-1}$, and its differential at the identity, called adjoint representation of G:

$$\mathrm{AD}_g : \mathfrak{g} \mapsto \mathfrak{g}.$$

Then the differential of $g \mapsto \mathrm{AD}_g$, written ad is called the adjoint representation of $\mathfrak{g}$, and one can show that it coincides with the Lie bracket of $\mathfrak{g}$:

$$\mathrm{ad}_x(y) = [x, y].$$

Example 2.4.5: General linear Lie algebra

Recall from Example 2.4.1 that $GL(n)$ is a Lie group, whose differentiable structure comes from its embedding in $M_n(\mathbb{R})$. Thus, by (1) of Theorem 2.2.2 (page 19), its tangent space at the identity is the entire matrix space, and as the Lie algebra of $GL(n)$ it is written $\mathfrak{gl}(n)$. We now compute explicitely AD and ad and verify that the bracket of $\mathfrak{gl}(n)$ coincides with the commutator defined in Example 2.4.3.

The conjugation map is the restriction to $GL(n)$ of the linear map $h \mapsto ghg^{-1}$ of $M_n(\mathbb{R})$ for any $g \in GL(n)$. Its differential is thus for any $X \in M_n(\mathbb{R})$

$$\mathrm{AD}_g(X) = gXg^{-1}.$$

Now consider a curve $c : (-\epsilon, \epsilon) \to GL(n)$ such that $c(0) = I_n$ and $c'(0) = X \in M_n(\mathbb{R})$. For any $Y \in M_n(\mathbb{R})$ we have

$$\mathrm{ad}_X Y = d(g \mapsto \mathrm{AD}_g(Y))_{I_n} = \frac{d}{dt}\bigg|_{t=0} (c(t)Yc(t)^{-1}) = XY - YX$$

To summarize,

- The Lie algebra of a Lie group is its tangent space at identity T_eG or equivalently, the algebra of left-invariant vector fields.
- The Lie bracket on T_eG is defined via the adjoint representation and coincides with that of left-invariant vector fields.

- In the case of linear Lie algebras (i.e. subalgebras of $M_n(\mathbb{R})$), the Lie bracket coincides with the matrix commutator.

Remark 2.4.2. *Note that right-invariant vector fields can be defined analogously as left-invariant vector fields, and their set forms a Lie algebra. However, the Lie bracket of right-invariant vector fields coincides with the opposite of the adjoint representation. For practical and historical reasons, right-invariant fields are used in infinite dimension while left-invariant fields are used in finite dimension.*

As all the next examples will be subgroups of the general linear group, we state a result by von Neumann and Cartan that gives a necessary and sufficient condition for a subgroup of a Lie group G to be an embedded Lie subgroup, i.e. a Lie group with differential structure agreeing with that of G, and such that the inclusion map is smooth (Cartan 1930, Section III, paragraph 26).

Theorem 2.4.2 (Cartan-von-Neumann)**.** *Let G be a Lie group. Any closed sugroup $H \subset G$ of G is a Lie subgroup of G. Conversely, any Lie subgroup of G is closed.*

One can also show that the Lie algebra of a subgroup G is a Lie subalgebra of the Lie algebra $\mathfrak{g}$ of G. Thus, they share the same bracket. Therefore, all Lie algebras considered in geomstats use the matrix commutator as bracket.

Example 2.4.6: Special Orthogonal group

Recall that the special orthogonal group is the set of orthogonal matrices of positive determinant. It is clear that it is stable by multiplication, hence it is a subgroup of $GL(n)$.

Furthermore, $SO(n) = f^{-1}(I_n) \cap \det^{-1}(1)$, where $f : A \mapsto A^\top A$ and det are continuous maps. $SO(n)$ is thus closed in $M_n(\mathbb{R})$, and is hence a Lie subgroup of $GL(n)$.

Moreover, from ex. 2.2.10 (page 20), its Lie algebra is

$$\mathfrak{so}(n) \triangleq T_{I_n} SO(n) = \{A \in M_n(\mathbb{R}) \mid A^\top + A = 0\} = \mathrm{Skew}(n).$$

In geomstats, $SO(n)$ is implemented as a subclass of the `MatrixLieGroup` class and as it is embedded in $GL_+(n)$ (the subgroup of $GL(n)$ formed by matrices with positive determinent), it also inherits from the `LevelSet` class. This allows to inherit the composition method, but the inverse method is overridden by the transposition.

```python
class SpecialOrthogonalMatrices(MatrixLieGroup, LevelSet):
    """Class for special orthogonal group."""

    def __init__(self, n):
        matrices = Matrices(n, n)
        gln = GeneralLinear(n, positive_det=True)
        super(SpecialOrthogonalMatrices, self).__init__(
            dim=int((n * (n - 1)) / 2), n=n, value=gs.eye(n),
            lie_algebra=SkewSymmetricMatrices(n=n), embedding_space=gln,
            submersion=lambda x: matrices.mul(matrices.transpose(x), x),
            tangent_submersion=lambda v, x: 2 * matrices.to_symmetric(
                matrices.mul(matrices.transpose(x), v)))
        self.bi_invariant_metric = BiInvariantMetric(group=self)
        self.metric = self.bi_invariant_metric

    @classmethod
    def inverse(cls, point):
        """Return the transpose matrix of point."""
        return cls.transpose(point)

    def projection(self, point):
        """Project a matrix on SO(n) by minimizing the Frobenius norm."""
        aux_mat = self.submersion(point)
        inv_sqrt_mat = SymmetricMatrices.powerm(aux_mat, - 1 / 2)
        rotation_mat = Matrices.mul(point, inv_sqrt_mat)
        det = gs.linalg.det(rotation_mat)
        return utils.flip_determinant(rotation_mat, det)
```

And the Lie algebra is implemented as follows.

```python
class SkewSymmetricMatrices(MatrixLieAlgebra):
    """Class for skew-symmetric matrices."""

    def __init__(self, n):
        dim = int(n * (n - 1) / 2)
        super(SkewSymmetricMatrices, self).__init__(dim, n)
        self.ambient_space = Matrices(n, n)

    def belongs(self, mat, atol=gs.atol):
        """Evaluate if mat is a skew-symmetric matrix."""
        has_right_shape = self.ambient_space.belongs(mat)
        if has_right_shape:
            return Matrices.equal(mat, - Matrices.transpose(mat), atol=atol)
        return False

    @classmethod
    def projection(cls, mat):
        """Compute the skew-symmetric component of a matrix."""
        return 1 / 2 * (mat - Matrices.transpose(mat))
```

Example 2.4.7: Special Euclidean group

We now introduce a group that is ubiquitous in applications and that shall be used to exemplify our theorem in Chapter 4. This group is defined as the set of direct isometries - or rigid-body transformations - of $\mathbb{R}^n$, i.e. the linear transformations of the affine space $\mathbb{R}^n$ that preserve its canonical inner-product. Such transformation ρ can be decomposed in a rotation part and a translation part: $\rho(x) = Rx + u$, where $R \in SO(n)$ and $x, u \in \mathbb{R}^n$. Define

$$SE(n) = \{(R, u) \mid R \in SO(n), u \in \mathbb{R}^n\}.$$

Now, the composition of two isometries $\rho = (R, u), \rho'(R', u')$ remains an isometry:

$$\rho \circ \rho'(x) = RR'x + Ru' + u,$$

and can be written $\rho \circ \rho' = (RR', Ru' + u)$. This suggests the representation of $SE(n)$ in *homogeneous* coordinates

$$\rho = \begin{pmatrix} R & u \\ 0 & 1 \end{pmatrix} \in GL(n+1) \tag{2.26}$$

The composition of isometries then corresponds to matrix multiplication, and $SE(n)$ is now a Lie subgroup of $GL(n+1)$. Its Lie algebra is

$$\mathfrak{se}(n) = \left\{ \begin{pmatrix} S & v \\ 0 & 0 \end{pmatrix} \mid S \in \text{Skew}(n), v \in |R^n \right\}.$$

Thus its dimension is $\frac{n(n+1)}{2}$. In geomstats, $SE(n)$ inherits from `MatrixLieGroup` and is embedded in $GL_+(n+1)$ with overridden `inverse` function. We used a utility function that builds a block matrix according to (2.26) (see appendix B.1). The submersion used is the map

$$\begin{pmatrix} R & u \\ v^\top & c \end{pmatrix} \in GL_+(n+1) \mapsto (R^\top R, v, c) \in M_n(\mathbb{R}) \times \mathbb{R}^n \times \mathbb{R},$$

so that $SE(n) = f^{-1}\{(I_n, 0, 1)\}$. It is not printed in the code below in the interest of space.

```python
class SpecialEuclideanMatrices(MatrixLieGroup, LevelSet):
    """Class for special Euclidean group."""

    def __init__(self, n):
        super().__init__(
            n=n + 1, dim=int((n * (n + 1)) / 2),
            embedding_space=GeneralLinear(n + 1, positive_det=True),
            submersion=submersion, value=gs.eye(n + 1),
            tangent_submersion=tangent_submersion,
            lie_algebra=SpecialEuclideanMatrixLieAlgebra(n=n))
        self.rotations = SpecialOrthogonal(n=n)
        self.translations = Euclidean(dim=n)
        self.n = n

        self.left_canonical_metric = \
            SpecialEuclideanMatrixCannonicalLeftMetric(group=self)
        self.metric = self.left_canonical_metric

    @property
    def identity(self):
        """Return the identity matrix."""
        return gs.eye(self.n + 1, self.n + 1)

    def inverse(self, point):
        """Return the inverse of a point."""
        n = self.n
        transposed_rot = self.transpose(point[..., :n, :n])
        translation = point[..., :n, -1]
        translation = gs.einsum(
            '...ij,...j', transposed_rot, translation)
        return homogeneous_representation(
            transposed_rot, -translation, point.shape)
```

Remark 2.4.3. *Note that as a manifold, $SE(n)$ defined in the above example corresponds to the product manifold $SO(n)\times\mathbb{R}^n$ (Ex. 2.2.6 p. 17), but as a Lie group, it is the semi-direct product $SO(n)\ltimes\mathbb{R}^n$, because the rotation part acts on the translation part in the composition rule.*

2.4.2 The exponential map

We now study the flow of left-invariant vector fields of a Lie group. It allows to canonically map the Lie algebra to its Lie group. First, the left-invariance translates into a commutation property of the flow with the left translation map:

Proposition 2.4.1. *Let G be a Lie group and $X\in\mathcal{L}(G)$. Then X is complete, and if $(\phi_t)_t$ is its flow, then for any $t\in\mathbb{R}$ and $g,g'\in G$ we have*

$$\phi_t(gg') = g\phi_t(g'), \quad i.e. \quad \phi_t\circ L_g = L_g\circ\phi_t.$$

This allows the following

Definition 2.4.4 (Exponential map)**.** We call exponential map, and write $\exp : \mathfrak{g} \to G$ the map defined by $x \mapsto \phi_1(e)$ where $(\phi_t)_t$ is the flow of the left-invariant vector fields $\tilde{x}$.

Remark 2.4.4. *This is the second definition of exponential map we encounter, where the first was the Riemannian exponential defined by the geodesic flow of a metric. We will refer to the one canonically defined on a Lie group as the group exponential and use a lowercase exp.*

Example 2.4.8: General linear

The fundamental case is of course again that of the general linear group. Recall that its Lie algebra $\mathfrak{gl}(n) = M_n(\mathbb{R})$ is the set of square matrices, and the group law is the (linear) matrix multiplication. Therefore, a left-invariant vector field $\tilde{X}$ associated to $X \in \mathfrak{gl}(n)$ is defined by

$$g \mapsto gX$$

Let γ be the integral curve from the identity I_n. This means that γ is solution to the ODE defined on $\mathbb{R}$ with initial condition $\gamma(0) = I_n$

$$\gamma'(t) = \gamma(t)X$$

It is well known that the unique solution is the matrix exponential, defined by the series

$$e^{tX} = \sum_{k=0}^{\infty} \frac{t^k}{k!} X^k.$$

Thus $\exp(X) = e^X$.

We now state some of the fundamental properties of the exponential map (see e.g. Gallier and Quaintance 2020, Proposition 18.6-18 .7-18.13).

Proposition 2.4.2. *For any Lie group G, the exponential map $\exp : \mathfrak{g} \to G$ is smooth and a local diffeomorphism at 0.*

The inverse of the group exponential, defined locally, is called *logarithm* map and is valued in $\mathfrak{g} = T_eG$. It thus allows to map data defined on G to its Lie algebra $\mathfrak{g}$, which is a vector space! This fact is at the basis of many algorithms that handle Lie group data.

Remark 2.4.5 (Infinite dimension)**.** *Proposition 2.4.2 is not true in infinite dimension. For example, if $G = \mathrm{Diff}(M)$, the exp map is not even surjective in any neighborhood of the identity (Schmid 2004).*

Proposition 2.4.3. *Let G, H be a Lie group, and $f : G \to H$ a Lie group homomorphism. Then*

$$f \circ \exp = \exp \circ df_e \tag{2.27}$$

In particular, if $G = H$ and $f = C_g$, we have

$$\exp(t\,\mathrm{AD}_g(u)) = g\exp(tu)g^{-1} = C_g(\exp(tu)). \tag{2.28}$$

The commutation property (2.27) can be depicted by saying that the following diagram commutes, meaning both paths leading from the top left space to the bottom right one are equivalent:

$$\begin{array}{ccc} \mathfrak{g} & \xrightarrow{\exp} & G \\ \downarrow{\scriptstyle df_e} & & \downarrow{\scriptstyle f} \\ \mathfrak{h} & \xrightarrow{\exp} & H \end{array}$$

Using $f = L_g$, or equivalently (thanks to (2.28)) $f = R_g$, we define the exponential map at any point $g \in G$ so that it fulfills the above properties. For any $v \in T_gG$

$$\exp_g(v) = L_g \circ \exp \circ (dL_g)^{-1} = R_{g^{-1}} \circ \exp \circ dR_g.$$

where differentiation is at e, and so that the group exp is now understood as $\exp_e$. In geomstats we use the implementation of the matrix exponential and logarithm from the backends and include it in the `MatrixLieGroup` class.

Moreover, we can recover the whole connected component of the identity from the Lie algebra:

Theorem 2.4.3. *If G is a Lie group and G_0 is the connected component of e, then G_0 is generated by* $\exp(\mathfrak{g})$.

In particular (see Gallier and Quaintance 2020, Theorem 1.6 and 1.12), we have the following

Proposition 2.4.4. *$SO(n)$ and $SE(n)$*

- *The exponential map* $\exp : \mathfrak{so}(n) \to SO(n)$ *is surjective,*
- *The exponential map* $\exp : \mathfrak{se}(n) \to SE(n)$ *is surjective.*

Note that by applying the commutation property (2.27) to the inclusion map, the exponential maps of $SE(n)$ and $SO(n)$ coincide with the restriction of the exponential map of $GL(n)$, i.e. the matrix exponential.

Finally, recall that flows of complete vector fields are one-parameters subgroups of $\mathrm{Diff}(M)$. Define now a one-parameter subgroup of G as a Lie group homomorphism $t \in \mathbb{R} \mapsto \gamma_t \in G$, written $(\gamma_t)_{t\in\mathbb{R}}$. Then by a change of variable in the flow equation, it is clear that $t \mapsto \exp(tX)$ is a one-parameter subgroup for any $X \in \mathfrak{g}$. In fact, all one-parameter subgroups are of this form and the maps:

$$\theta : (\gamma_t)_{t\in\mathbb{R}} \mapsto \frac{d}{dt}\Big|_{t=0}\gamma_t; \qquad \theta' : X \mapsto \big(\exp(tX)\big)_{t\in\mathbb{R}}$$

are inverse to each other, between $\mathfrak{g}$ and the set of one-parameter subgroups.

Before closing this section, we illustrate it with a representation of the one-parameter subgroups of $SE(2)$.

2.4.3 Group action and homogeneous spaces

To conclude this section, we define the notion of group action, that will be fundamental in Section 3. It allows to model groups as generators of transformations of any manifold. In Example 2.2.5 (page 16), the action of $SO(m)$ on $M_{m,k}(\mathbb{R})$ is in fact the main factor to study the geometry of the quotient space $S\Sigma_m^k$.

Definition 2.4.5 (Group action)**.** Given a set M and a group G, a *left-action* of G on M is a function $\triangleright : G \times M \to M$, such that:

- $\forall g, h \in G, \forall x \in M, \quad g \triangleright (h \triangleright x) = (gh) \triangleright x$,
- $\forall x \in M, \quad e \triangleright x = x$.

If furthermore M is a smooth manifold and G is a Lie group, we say that this action is smooth if the map $\triangleright$ is smooth from the product manifold $G \times M$ to M. In this case, for any $g \in G$, $x \mapsto g \triangleright x$ is a diffeomorphism of M, whose inverse is $x \mapsto g^{-1} \triangleright x$. We call this map *left translation* by g and write it L_g by analogy with the group law.

Remark 2.4.6. *Note that the map $g \mapsto L_g$ is a group homomorphism between G and* $\mathrm{Diff}(M)$*. In fact, if this mapping is injective, we say that the action is faithful and we can see G as an immersed subgroup of* $\mathrm{Diff}(M)$*.*

We say that an action is

- *free* if for all $g \in G$ and $x \in M$, if $g \triangleright x = x$, then $g = e$;
- *faithful* if for all $g \in G$, if $\forall x \in M, g \triangleright x = x$ then $g = e$;
- *transitive* if for all $x, y \in M$, there exists $g \in G$ such that $y = g \triangleright x$;
- *proper* if for all compact set $K \subset M$, the set $\{g \in G \mid K \cap gK \neq \emptyset\}$ is compact. This is always the case if G is compact.

Note that a free action means that $G_x = e$ for all $x \in M$. In contrast, if an action is faithful, then the map $x \mapsto g \triangleright x$ is different from the identity of M for all $g \in G$. These notions play an essential role in determining the geometry of M. For example, we shall see in Kendall shape spaces that the action of $SO(n)$ is free if we remove certain points, that are otherwise considered as singularities.

For any $x \in M$, define the *orbit* of x as $[x] = G \triangleright x = \{g \triangleright x \mid g \in G\}$. As in Ex. 2.2.5 (p. 16), the orbits are the equivalence classes of the following relation

$$x \sim y \iff \exists g \in G, y = g \triangleright x \iff y \in g \triangleright x.$$

Definition 2.4.6 (Quotient space)**.** The set of orbits is denoted M/G and is called *quotient* of M by G. This set is obtained by identifying all the points in an orbit. Define the *canonical projection*

$$\pi : \left\{ \begin{array}{ccc} M & \longrightarrow & M/G \\ x & \longmapsto & [x] \end{array} \right. \tag{2.29}$$

We have the following sufficient conditions on the action to ensure that a quotient space is indeed a smooth manifold (Lafontaine, Gallot, et al. 2004, Theorem 1.95).

Theorem 2.4.4 (Quotient manifold)**.** *Let M be a smooth manifold, G a Lie group with a smooth left-action on M that is free and proper. Then the canonical projection (2.29) is a submersion and there is a unique differential structure on M/G with this property.*

This case occurs in many manifolds considered until here, e.g. the Kendall size-and-shape space of Example 2.2.5 and the Stiefel manifold (Example 2.2.4 page 15). We now focus on a particular case of the action of a subgroup of G on G. At any $x \in M$, we can define the *isotropy* subgroup, or *stabilizer* of x as

$$G_x = \{g \in G \mid g \triangleright x = x\}.$$

Then G_x is a Lie subgroup of G. We consider the action of G_x on G, or more generally of a Lie subgroup H of G, obviously defined by the group law. We have (Gallier and Quaintance 2020, Corollary 22.10):

Theorem 2.4.5. *The action of a Lie subgroup H of a Lie group G on G is free and proper.*

The previous Theorem 2.4.4 thus applies to the right action $(g, h) \in G \times H \mapsto gh \in G$ and G/H is a smooth manifold such that $\pi : G \to G/H$ is a submersion. The orbits of this right action are the left cosets $\{\mathrm{gH} \mid g \in G\}$ Note that G acts on G/H by $g_1 \triangleright g_2 H = g_1 g_2 H$, and it is clear that this action is transitive. In fact all transitive left actions yield to a quotient space of orbits from a right action.

Definition 2.4.7 (Homogeneous space)**.** We say that M is homogeneous if there exists a smooth transitive action of a Lie group G on M.

Remark 2.4.7. *Note that there might be several such Lie groups with a transitive action on a given homogeneous space M.*

All homogenous spaces correspond (up to a diffeomorphism) to quotient spaces of the form G/H, as stated in the following (Gallier and Quaintance 2020, Theorem 22.13).

Theorem 2.4.6. *Let G be a connected Lie group acting smoothly and transitively on a smooth manifold M, so that M is homogeneous. Then for any $x \in M$, writing $H = G_x$, the map*

$$\begin{cases} G/H & \longrightarrow & M \\ gH & \longmapsto & g \triangleright x \end{cases}$$

is a diffeomorphism, so that $M \simeq G/H$.

Note that the choice of reference point x above does not matter, as for two different choices x and y, the action is transitive so there exists $g \in G$ such that

$y = g \triangleright x$, and the stabilizers G_x and $G_y = G_{g \triangleright x} = gG_xg^{-1}$ are conjugate, hence isomorphic.

We exemplify all these notions with the sphere, the Stiefel manifold and the manifold of symmetric positive definite (SPD) matrices, although other manifolds in geomstats such as hyperbolic spaces and the Grassmannian are also homogeneous.

Example 2.4.10: Hypersphere

The hypersphere embedded in $\mathbb{R}^{d+1}$ can be seen as a homogeneous space by considering the action of the special orthogonal group $SO(d+1)$ of the embedding space. For $R \in SO(d+1), x \in S^d$:

$$R \triangleright x = Rx \in S^d \tag{2.30}$$

Indeed, the sphere is stable by the action of rotation matrices, and this action is transitive. Consider a pole $x_0 = (1, 0, \ldots, 0) \in S^d$. Its stabilizer is the set of rotations whose axis is x_0, i.e.

$$H = \left\{ \begin{pmatrix} 1 & 0 \\ 0 & R \end{pmatrix} \mid R \in SO(d) \right\} \simeq SO(d).$$

Then by Theorem 2.4.6, $S^d = SO(d+1)/SO(d)$. Intuitively this corresponds to identifying all the rotations that share the same axis where rotation axes are described by a unit vector, i.e., a point on the sphere.

Example 2.4.11: Stiefel manifold

Recall that the Stiefel manifold $St(k, n)$ is the set of orthonormal k-frames of $\mathbb{R}^n$. The special orthogonal group $SO(n)$ thus acts on $St(k, n)$ by

$$R \triangleright (u_1, \ldots, u_k) = (Ru_1, \ldots, Ru_k)$$

It is clear that this action is transitive. Furthermore, as a frame can be represented by an orthogonal matrix of size $n \times k$, the above action corresponds to matrix multiplication and is thus smooth.

For any $U \in St(k, n)$, the stabilizer of U is

$$H = \left\{ \begin{pmatrix} I_k & 0 \\ 0 & R \end{pmatrix} \mid R \in SO(n-k) \right\} \simeq SO(n-k).$$

Then by Theorem 2.4.6, $St(k, n) = SO(n)/SO(n-k)$. Intuitively, this corresponds to the fact that a k-frame can be completed into an orthonormal basis of $\mathbb{R}^n$, i.e. a matrix in $SO(n)$, and matrices in H map one such completion to all the other completions. Thus quotienting $SO(n)$ by $SO(n-k)$ amounts to identifying the orthonormal bases that agree on the first k vectors.

Example 2.4.12: Grassmann manifold

Define the Grassmann manifold $Gr(n,k)$ as the set of subspaces of $\mathbb{R}^n$ of dimension k. On any k-dimensional subspace $\mathcal{U}$ is defined a unique orthogonal projector, i.e. a linear map p defined in $\mathbb{R}^n$ for which $p \circ p = p$ and $\mathrm{Im}(p) = \mathcal{U}$ and $\ker(p) = \mathcal{U}^\perp$. Any such projector is represented by a symmetric matrix P of size n, rank k and such that $P^2 = P$. We thus adopt the representation of $Gr(n,k)$ as the set

$$Gr(n,k) = \left\{P \in \mathrm{Sym}(n) \mid P^2 = P \text{ and } \mathrm{rank}(P) = k\right\}.$$

Intuitively, any k dimensional subspace can be represented by an equivalence class of orthonormal bases, so that the Grassmannian is a quotient space of the orthogonal group $O(n)$. To consider a connected group, let $G = SO(n)$, and let the action of G on $Gr(k,n)$ correspond to a change of basis: $Q \triangleright P = QPQ^\top$. It is clear that any rank-k projector can be represented by a matrix of the form

$$P_k = \begin{pmatrix} I_k & 0 \\ 0 & 0 \end{pmatrix}.$$

This means exactly that the action is transitive, and $Gr(k,n)$ is the orbit of P_k. Let H be its stabilizer, i.e. the set

$$H = \left\{ \begin{pmatrix} Q & 0 \\ 0 & R \end{pmatrix} \mid Q \in SO(k), R \in SO(n-k) \right\} \simeq SO(k) \times SO(n-k).$$

$Gr(k,n) = SO(n)/(SO(k) \times SO(n-k))$ is therefore a homogeneous space with canonical projection $\pi : Q \in SO(n) \mapsto QP_kQ^\top$ and dimension $k(n-k)$. Of course this manifold also be described as a quotient of $St(k,n)$ by $SO(p)$, where the action is by right multiplication and the projection is identified with $U \in St(k,n) \mapsto UU^\top \in Gr(k,n)$. This quotient can be more practical than that of $SO(n)$ when $n \gg k$. The Grassmann manifold is widely used in applications both in numerical problems such as low-rank matrix decomposition or optimization, and in higher-level applications in machine learning, computer vision and image processing. We refer to Bendokat, Zimmermann, et al. 2020 for a very complete exposition of this manifold.

Example 2.4.13: SPD matrices

Recall that a symmetric matrix Σ is *positive definite* if it is invertible and $\forall x \in \mathbb{R}^n, x^\top \Sigma x \geq 0$. The set $SPD(n)$ of such matrices is an open set of the vector space $\mathrm{Sym}(n)$ of symmetric matrices, hence it is an embedded manifold and the canonical immersion $\mathrm{id} : SPD(n) \to \mathrm{Sym}(n)$ defines a global chart. Furthermore, for any $\Sigma \in SPD(n)$ the tangent space $T_\Sigma SPD(n)$ is identified with $\mathrm{Sym}(n)$.

Now, define the action of $GL(n)$ on $SPD(n)$ by:

$$\triangleright : (A, \Sigma) \in GL(n) \times SPD(n) \longmapsto A\Sigma A^\top.$$

This action is sometimes called action by *congruence*. It is smooth and transitive. Indeed, let $\Sigma = PDP^\top, \Sigma' = Q\Delta Q^\top \in SPD(n)$ be two SPD matrices with their eigenvalue decomposition given by the spectral theorem, then $A = Q\Delta^{-1/2}D^{-1/2}P^\top$ is such that $A \triangleright \Sigma = \Sigma'$. The manifold of SPD matrices is thus a homogeneous space.

Finally, the isotropy group of the identity matrix I_n is the orthogonal group $O(n) \subset GL(n)$:

$$G_{I_n} = \{A \in GL(n) \mid A \triangleright I_n = I_n\} = \{A \in GL(n) \mid AA^\top = I_n\} = O(n).$$

It is indeed a closed subgroup of $GL(n)$, so its right action on $GL(n)$ is free and proper (Theorem 2.4.5), and by Theorem 2.4.6, $SPD(n)$ is isomorphic to the orbit space $GL(n)/O(n)$. The canonical projection $\pi : \begin{cases} GL(n) & \longrightarrow & SPD(n) \\ A & \longmapsto & AA^\top \end{cases}$ is a submersion.

In geomstats, we implement a class for symmetric matrices and a class for the manifold of SPD matrices that inherits from `OpenSet`.

```
class SPDMatrices(OpenSet):
    """Class for the manifold of symmetric positive definite matrices."""

    def __init__(self, n):
        super(SPDMatrices, self).__init__(
            dim=int(n * (n + 1) / 2),
            ambient_space=SymmetricMatrices(n))
        self.n = n

    def belongs(self, mat, atol=gs.atol):
        """Check if a matrix is symmetric with positive eigenvalues."""
        is_symmetric = super(SPDMatrices, self).belongs(mat, atol)
        eigvalues = gs.linalg.eigh(mat, eigvals_only=True)
        is_positive = gs.all(eigvalues > 0, axis=-1)
        belongs = gs.logical_and(is_symmetric, is_positive)
        return belongs
```

As the covariance matrix of a multivariate random variable is an positive semi-definite matrix, this manifold is ubiquitous in applications such as signal processing, neuroscience, etc. (see Pennec, Sommer, et al. 2020, Chapter 3. and references therein), although some issues arise when the matrices are degenerate.

2.5 Statistics and machine learning with Geomstats

To conclude this chapter, we demonstrate the use of geomstats to perform statistics on manifold-valued data. The strength of the package is that learning algorithms are defined as external estimators that take the geometric object as input. This happens thanks to the standardized interface of the classes, and ensures that all learning tools be available for all the manifolds and metrics. The obtained flexibility allows to compare the impact of the different metrics on the learning results. We give a brief introduction to geometric statistics, and the interested reader is referred to Pennec, Sommer, et al. 2020, Chapter 2 for the theoretical exposition. An introductory paper gathering more examples using geomstats was presented at the Scipy Conference 2020 in Austin, Texas (Miolane, Guigui, Zaatiti, et al. 2020).

2.5.1 Probability distributions and sampling

Given a probability measure, one can define random variables valued in a manifold M as follows.

Definition 2.5.1. Let $(\Omega, \mathcal{B}(\Omega), \Pr)$ be a probability space, with $\mathcal{B}(\Omega)$ the Borel σ-algebra of (Ω) and Pr a unit mass measure. Then a *random variable* in the Riemannian manifold M is a Borel measurable function $X : \Omega \to M$.

A Riemannian metric offers a convenient framework to define probability distributions on manifolds, as it defines a volume measure.

Definition 2.5.2 (Riemannian volume form)**.** An oriented (pseudo)-Riemannian manifold (M, g) has a natural *volume form* $d\,\mathrm{Vol}(x) = \det(g(x))$.

The volume form provides the way to define integrals on M, i.e. it defines a measure. so that some probability distributions can be expressed by their densities with respect to that measure.

Definition 2.5.3. The random variable X has *density* f if $\forall \mathcal{X} \in \mathcal{B}(M)$, $\Pr(X \in \mathcal{X}) = \int_{\mathcal{X}} f(y) dv(y)$ and $\Pr(M) = 1$.

Example 2.5.1: Uniform distribution

Let M be a compact Riemannian manifold, for example the hypersphere, the special orthogonal group of the Grassmann manifold. Then a uniform distribution on M has density

$$f(x) = \frac{1}{\mathrm{Vol}(M)}.$$

Example 2.5.2: Gaussian distribution

Let M be a Riemannian Symmetric space. A Gaussian distribution with mean and precision (μ, Γ) is defined by the density

$$f(x) = \alpha(\Gamma, \mu) \exp(-\log_\mu(x)^\top \Gamma \log_\mu(x)).$$

It is the entropy maximizing distribution (Pennec 2006), and in the isotropic, non-positive curvature, complete and simply-connected case, the maximum likelihood estimator coincides with the Frechet Mean (Said, Hajri, et al. 2018).

Sampling A common task in statistics is to draw samples from a given probability distribution on a manifold. This might be motivated by inference tasks where the posterior distributions has constrained parameters, in testing goodness of fit for exponential families, or to generate samples of data to test our learning algorithms (Diaconis, Holmes, et al. 2013; Barp, Kennedy, et al. 2019, and references therein).

However this is does not reduce to sampling from usual distributions even when a parametrization of the manifold is available. Indeed, the curvature of the space generally deforms or stretches the densities. This is illustrated in the following examples.

Example 2.5.3: Uniform distribution on the sphere

Consider the unit sphere $S^2 \subset \mathbb{R}^3$, with the spherical coordinates from the north pole $e_0 = (0, 0, 1)$: $x = \sin(\phi)\cos(\theta), y = \sin(\phi)\cos(\theta), z = \cos(\phi)$. The volume element (here the area) is $d\,\mathrm{Vol}(\theta, \phi) = \sin(\phi) d\theta d\phi$. A naive attempt at sampling from the uniform distribution on the sphere would be to sample θ uniformly in $[0, 2\pi)$ and ϕ uniformly in $[0, \pi)$.

However, near $\phi = 0$ and $\phi = \pi$, as $\sin(\phi)$ there is less density than with respect to the usual measure of the 2d-plane. We thus expect points to accumulate in a non-uniform way around the poles of the sphere.

In fact, if Y be a Gaussian vector in $\mathbb{R}^d$ with mean 0 and covariance identity. Then $X = \frac{Y}{\|Y\|}$ is uniformly distributed in S^d. The two sampling strategies are tested with geomstats and shown on Figure 2.6.

```
space = Hypersphere(2)
n_samples = 5000
uniform_param = gs.random.rand(
    n_samples, 2) * gs.pi * gs.array([1., 2.])[None, :]
naive_samples = space.spherical_to_extrinsic(uniform_param)
uniform_samples = space.random_uniform(n_samples)
```

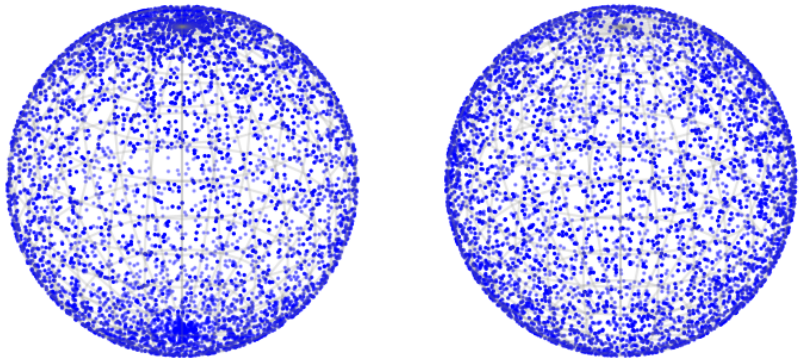


Figure 2.6: Comparison of the naive sampling from a uniform distribution on the spherical coordinates (left) and a more appropriate scheme that respects the anisotropy of the scheme (right). In the naive case, points accumulate near the poles and are sparser around the equator.

Example 2.5.4: The Special Orthogonal group

Let $SO(n)$ be the group of unit determinant orthogonal matrices of size $n \times n$. Let Y be a matrix of size $n \times n$ with independent standard normal distribution on each coefficient. The QR decomposition of Y defines $Y = XR$ where $X \in SO(n)$ and R upper triangular. Then X is uniformly distributed on $SO(n)$ w.r.t. to the Haar measure, which coincides with the Riemannian measure (Eaton 1983).

Example 2.5.5: Stiefel

Recall the Stiefel manifold is the set of orthonormal k-frames in $\mathbb{R}^n$. It can be represented by the set of matrices $\{U \in \mathbb{R}^{n\times k} | U^\top U = I_k\}$. Let Z be a matrix with i.i.d standard normal distribution on the entries. Then $X = Z(Z^\top Z)^{-1/2}$ is uniformly distributed (Chikuse 2003, Theorem 2.2.1). Note that X is a factor of the polar decomposition of Z, representing the *orientation* of Z.

Example 2.5.6: Grassmann

The Grassmann manifold is the set of k-dim subspaces of $\mathbb{R}^n$. It can be represented by the the set of projection matrices of size $n \times n$, i.e. symmetric rank-k P such that $P^2 = P$. Let Z be a matrix with i.i.d standard normal distribution on the entries. Then $X = Z(Z^\top Z)^{-1/2}Z$ is uniformly distributed (Chikuse 2003, Theorem 2.2.2). Note that $X = YY^\top$ for Y uniformly distributed on the Stiefel manifold.

These examples are particular cases where simple recipes were available to sample from the uniform distribution. They are implemented in geomstats. Sampling from non-uniform distribution is usually intractable and we resort to simulation methods from the literature. In particular, in geomstats we focused on rejection sampling, that consists in sampling from a proposal distribution whose density is greater than the target density, and accepting the samples with probability corresponding to the ratio of the two densities[5]. We do not detail this procedure in this manuscript but refer to Wood 1994; Hauberg 2018 for examples of rejection sampling algorithms for non-uniform distributions on the hypersphere.

2.5.2 The Fréchet mean

Let (M, g) be a Riemannian manifold and let $x_1, \dots, x_n \in M$ be independent identically distributed (i.i.d.) sample data points. The Euclidean sample mean $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ is not defined unless we consider $M \subset \mathbb{R}^N$, but in this case $\bar{x}$ may not lie on M.

Definition, Existence, Uniqueness In fact, the mean is characterized by the property that it minimises the sum of squared distances to the data points. This property was used by Fréchet 1948 to generalize the notion of mean to metric spaces, and later by Hermann Karcher in Riemannian manifolds. For a historical note and corresponding references, we refer to Karcher 2014.

Definition 2.5.4 (Fréchet mean)**.** Let (M, g) be a Riemannian manifold with Riemannian distance function d and let $x_1, \dots, x_n \in M$ be an i.i.d. data set. The sample *Fréchet mean* is defined as the set of minimizers of the sum-of-squared distances, i.e.,

$$\bar{x} = \arg\min_{x \in M} \sum_{i=1}^{n} d(x_i, x)^2.$$

With some nuance in the requirements on the minimum (local or global), the Fréchet mean is also referred to as Karcher mean or Riemannian barycenter, or even Wasserstein barycenter in optimal transport. Note that as it is defined by an optimization, the Fréchet mean may not exist, or not be unique, depending on the properties of the distance function. For instance, the completeness of the metric is a sufficient condition to guarantee the existence of the Fréchet mean of a finite set of points (Pennec, Sommer, et al. 2020, Chapter 2).

Theorem 2.5.1. *Let M be a complete metric space. Then the Fréchet mean of any finite set of points $x_1, \dots, x_n$ exists.*

On the other hand, uniqueness depends on the convexity of the distance function and can be related to the sign of the curvature (Karcher 1977; W. S. Kendall 1990; Afsari 2011).

[5] see the Wikipedia page for more details.

Theorem 2.5.2. *Let M be a complete Riemannian manifold with sectional curvature bounded above by some δ, and let* $\mathrm{inj}(M)$ *be its injectivity radius. If samples $x_1, \dots, x_n \in M$ are contained in a geodesic ball of radius*

$$r = \frac{1}{2} \min\left(\mathrm{inj}(M), \frac{\pi}{\sqrt{\delta}}\right)$$

then the Fréchet mean $\bar{x}$ is unique.

Note that in the above, $\frac{\pi}{\sqrt{\delta}}$ is interpreted as ∞ if $\delta \leq 0$. This means that in complete manifolds of non-positive curvature, the Fréchet mean is always uniquely defined, as in the Euclidean case. On the other hand, this is not the case in spaces of positive curvature. For example, on the sphere, Theorem 2.5.2 ensures that the mean is unique if all the data points lie on the same open hemisphere, but the simple data set constituted by two antipodal points ($\{x, -x\}$ for any $x \in S^d$) exemplifies non-uniqueness of the mean: a great circle in this case.

The asymptotic properties of the Fréchet mean were studied in Bhattacharya and Patrangenaru 2005, where a law of large numbers and a central limit theorem are established, and prove the relevance of this notion of mean.

Characterization and Estimation Suppose that the data points $x_1, \dots, x_n$ lie in a ball B of radius $r < \frac{\mathrm{inj}(M)}{2}$ as in Theorem 2.5.2. Then the squared distance is obtained by measuring the length of a minimizing geodesic and can be written $d^2(x, y) = \|\operatorname{Log}_x(y)\|_x^2$ for any $x, y \in B$. One can show that the gradient of $y \mapsto d^2(x, y)$ is $-2\operatorname{Log}_y(x) \in T_yM$ (see the first variation formula in Lafontaine, Gallot, et al. 2004, Theorem 3.31).

Remark 2.5.1. *Recall that the gradient of a function $f : \mathbb{R}^d \to \mathbb{R}$ is defined as the adjoint to its differential, i.e.*

$$\forall (x, v) \in TM, \quad \langle \mathrm{grad}_x(f), v \rangle \triangleq df_x v.$$

This definition extends to functions defined on Riemannian manifolds, but the gradient thus depends on the metric, and is sometimes referred to as the Riemannian gradient. It is also named natural gradient in information geometry, in the case of manifolds of parameters of families of distributions with the Fisher-Rao metric. If the manifold is embedded in some $\mathbb{R}^N$ and equipped with the pullback metric, it can be obtained by projecting the usual gradient to the tangent space at the point where the function is being differentiated, similarly to the Levi-Civita connection.

By definition, the Fréchet mean must be a critical point of the sum-of-squared distances function, so that it verifies

$$\sum_{i=1}^{n} \operatorname{Log}_{\bar{x}}(y) = 0. \tag{2.31}$$

A simple strategy to estimate the sample Fréchet mean is thus to use a fixed-point iterative algorithm until (2.31) is verified. An update at iteration t is performed

along a geodesic: $x^{t+1} = \mathrm{Exp}_{x^t}(\gamma \sum_{i=1}^{n} \mathrm{Log}_{x^t}(y))$, where γ is a step size. Higher-order methods use adaptive step size, approximates of the Hessian function or its link with curvature to improve the performance of the estimation. Some of these are implemented in the `FrechetMean` module of geomstats.

Example 2.5.7: Fréchet mean on the sphere

A scientific expedition of 15 boats is collecting samples from the Pacific ocean to measure the quantity of micro-plastics in the water. Every month, they gather all the samples in one boat that goes back to their home harbour for the analyses. As they have been wandering around the Pacific for some time, we sample their positions from a spherical normal distribution around their last meeting point.

```
space = Hypersphere(2)
n_samples = 15
last_meeting_point = gs.array([0., 1., 0.])
samples = space.random_riemannian_normal(
    mean=last_meeting_point, precision=10, n_samples=15)
```

The geometers onboard compute the Fréchet mean of their positions to choose the meeting spot that minimises the cost of the fuel for this gathering (supposing no wind or currents).

```
estimator = FrechetMean(space.metric)
estimator.fit(samples)
new_meeting_point = estimator.estimate_
```

Their position (black) and the next meeting point (red) are displayed in Figure 2.7. The trajectories they have to follow are drawn in green, and are geodesics.

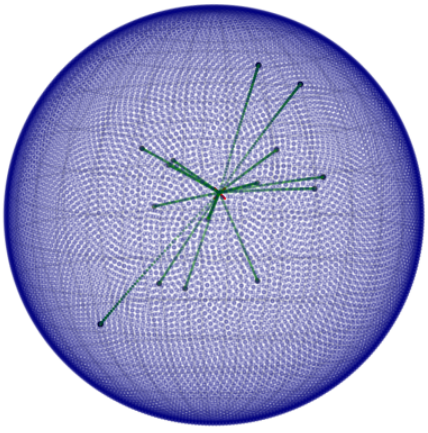


Figure 2.7: Fréchet mean on the sphere

The Fréchet mean $\bar{x}$ is key to define many other statistical and learning tools, e.g. the sample covariance matrix:

$$\Sigma = \frac{1}{n}\sum_{i=1}^{n} \mathrm{Log}_{\bar{x}}\, x_i \otimes \mathrm{Log}_{\bar{x}}\, x_i. \tag{2.32}$$

One can then parametrize densities from their mean and covariance, as in our collaboration with Emmanuel Chevallier in Chevallier and Guigui 2020. The proposed model allows to define a simple density estimation scheme.

One can also generalize the K-means clustering algorithm and the minimum distance to mean (MDM) classification algorithm with Fréchet means. These algorithms are implemented in geomstats in the classes `RiemannianKMeans` and `RiemannianMinimumDistanceToMeanClassifier`. Moreover, one can use the Fréchet mean to linearize the data by lifting it to the tangent space at the Fréchet mean, i .e. to consider the transformed data set $(\tilde{x}_1, \ldots, \tilde{x}_n)$ where

$$\tilde{x}_i = \mathrm{Log}_{\bar{x}}(x_i).$$

This is equivalent to using a Taylor expansion at the first order of the Riemannian exponential map around 0: $x_i = \mathrm{Exp}_{\bar{x}}(\tilde{x}_i) = \bar{x} + \tilde{x}_i + O(t^2)$. As $T_{\bar{x}}M$ is a vector space, all the usual statistical and machine learning tools can be used off-the-shelf on the transformed data set. This is implemented as a `Transformer` from the scikit-learn package in the `ToTangent` class of geomstats, and can be used in scikit-learn's pipeline.

2.5.3 Generalizations of PCA

The other fundamental tools to analyse data in vector spaces are the sample covariance matrix, and Principal Component Analysis (PCA). The aim of PCA is to find the sequence of subspaces such that data projected on these subspaces have maximum variance, or equivalently minimum reconstruction error (i.e. sum of squared distance to the original points). This equivalence is no longer true in Riemannian manifolds, as Pythagorean theorem is not true.

We describe the principal geodesic analysis (PGA), that aims at minimizing the reconstruction error when projecting data on a geodesic submanifold (Fletcher, Lu, et al. 2004). In the simplest forward fashion, the mean is first computed, and the first component is a geodesic from the mean. This is close to a slightly different procedure called Geodesic PCA (Huckemann, Hotz, et al. 2010b). The projection of a data point x on a geodesic γ with initial velocity v at the mean $\bar{x}$ is defined by

$$\pi_{\bar{x},v}(x) = \arg\min_{t\in\mathbb{R}} d^2\left(x, \mathrm{Exp}_{\bar{x}}(tv)\right).$$

There is no guaranty that this projection exists, but if the data is not too spread out, one can hope that there exists a convex neighborhood where the exponential map is injective that contains x and a portion of the geodesic. The projection is

solved by a gradient descent, as in the case of the Fréchet mean, where the gradient of the exponential is computed by automatic differentiation.

The next step is to minimize the overall reconstruction error, given a dataset $x_1, \dots, x_n$, we look for the initial velocity of a geodesic as:

$$v^* = \arg\min_{v \in T_{\bar{x}}M} \sum_{i=1}^{n} d^2\left(x_i, \pi_{\bar{x},v}(x_i)\right). \tag{2.33}$$

This time computing the gradient of the objective function seems more complicated, as n minimization problems are already solved by a gradient descent to evaluate the function. However Ablin, Peyré, et al. 2020 show that the gradient of a function defined by a minimum could be computed efficiently by automatic differentiation of the gradient descent approximation. We exemplify this in the example of the sphere. Of course this example is particularly simple as both Exp and Log maps are computed with closed form solutions whose derivatives are known, and some work remains to use automatic differentiation when the Log map itself is hard to compute.

Example 2.5.8: Principal geodesic analysis on earth

The boats of the same scientific exploration of Example 2.5.7 are in fact too busy collecting data and cannot make route to the Fréchet mean. Thankfully, the geometers onboard are expert users of geomstats and compute a PGA of their positions, so that one boat can visit the rest of the fleet.

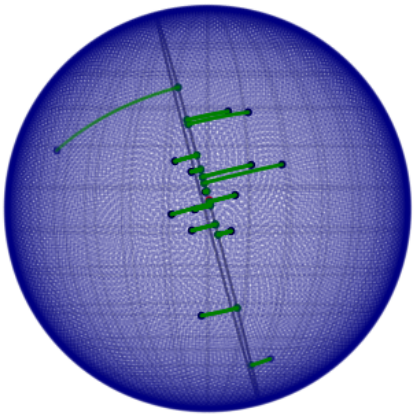


Figure 2.8: Example of PGA on the sphere. The target data are the blue dots, the projected values are in green, the first geodesic subspace is in black, and the mean in red.

A general simplification if the data is not too far from the mean is to approximate the projection operator by the Log map: $\mathrm{Log}_{\bar{x}}(\pi_{\bar{x},v}(x_i)) \approx \mathrm{Log}_{\bar{x}}(x_i)$. With this approximation, PGA resumes to computing the logarithms of all the data points

and computing a usual PCA of the obtained tangent vectors. This procedure is more generally called *tangent PCA* and is implemented in the `TangentPCA` class of geomstats.

2.5.4 Geodesic Regression

Similarly to the mean, the linear regression can be generalized to *geodesic regression* on Riemannian manifolds by solving a least-square fitting problem. Given target points $y_1, \dots, y_n \in M$, and data $t_1, \dots, t_n \in \mathbb{R}$, we seek the geodesic that best approximates the data:

$$\min_{(p,v)\in TM} \sum_{i=1}^{n} d^2\left(\mathrm{Exp}_p(t_i v), y_i\right), \tag{2.34}$$

where d and Exp are the Riemannian distance and exponential maps. When the metric is Euclidean, this coincides with the usual linear regression problem. However, there is no closed-form solution in general and the problem must be solved by optimization.

To differentiate the objective function of (2.34), one needs to compose the gradients of the squared distance with that of the exponential map. The gradient of the squared distance is proportional to the Riemannian logarithm, as noted above. On the other hand, the gradient of the Exp map is usually computed via Jacobi fields. However, to avoid implementing those in geomstats we chose to leverage automatic differentiation tools to compute the extrinsic gradient, and to project it to the right tangent space, as explained in Remark 2.5.1.

To solve the optimization problem, either a Riemannian gradient descent, or an extrinsic one with `scipy`'s solver is used.

Example 2.5.9: Geodesic Regression above earth

The European Space Agency sends a mission to run some maintenance on their geo-stationary satellites. As these are all at the same altitude, they lie on a sphere, and the geometers seek an optimal trajectory.

```
space = Hypersphere(2)
metric = space.metric
```

The maintenance should be done as close as possible to the end of the fuel tanks, which are known from previous missions. One vessel will leave earth, reach the first point of the trajectory called point γ, and split in two parts that will each go in opposite directions given by $\beta \in T_\gamma S^2$.

We use data from the previous mission to generate random satellite positions:

```
n_samples = 50
data = gs.random.rand(n_samples)
data -= gs.mean(data)

previous_gamma = space.random_uniform()
beta = space.to_tangent(5. * gs.random.rand(3), previous_gamma)
target = metric.exp(data[:, None] * beta, previous_gamma)
```

And add some noise because satellites did not stay on the previous misison's trajectory

```
normal_noise = gs.random.normal(size=(n_samples, 3))
noise = space.to_tangent(normal_noise, target) / gs.pi / 2
target = metric.exp(noise, target)
```

The optimal trajectory is computed by fitting a geodesic regression model with the Riemannian gradient descent:

```
gr = GeodesicRegression(space, algorithm='riemannian')
gr.fit(data, target, compute_training_score=True)
gamma_new, beta_new = gr.intercept_, gr.coef_
```

We can measure the mean squared error (MSE) with respect to the previous γ and β, and the determination coefficient from the noise level. We obtain MSEs around 10^{-3} and $R^2 = 0.96$.

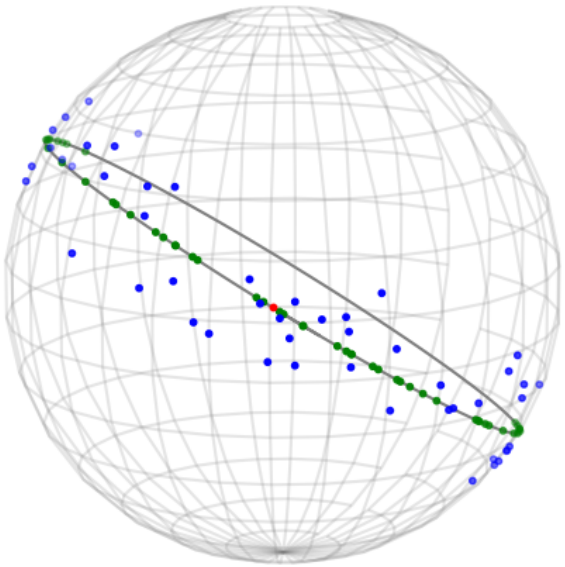


Figure 2.9: Example of geodesic regression on the sphere. The target data are the blue dots, the fitted values are in green, the regression geodesic in black, and the intercept (or γ in this example) in red.

2.6 Conclusion

In this chapter, we introduced the main notions of differential geometry and Riemannian metrics for an audience with a general background in mathematics. We focused on embedded manifolds of $\mathbb{R}^N$ and highlighted the different ways of defining a manifold, and a Riemannian metric. These are key to the architecture of the geomstats library and drove our recent contributions to the package, that aimed at making it more faithful to mathematical theory, more robust and more modular. Furthermore, we detailed the notions of curvature, geodesic, distance and parallel transport, that are the essential building blocks of geometric statistics. All the notions were exemplified with the most common manifolds encountered in mathematical text books as well as in applications. Finally, we gave an introduction to geometric statistic tools such as the Fréchet mean, principal geodesic analysis and geodesic regression, and illustrated these on toy examples. They are all implemented in the geomstats package with a common high-level interface, so that together with this chapter, geometric statistics become available to any data scientist on a wide variety of problems!

We also introduced the concepts of Lie groups, smooth group actions and quotient space, that will play a major role in the next chapter, as we focus on the implementation of Riemannian metrics that are invariant to a group action.

Chapter 3

Metrics defined by invariance properties

Contents

3.1 Introduction

As seen in the previous chapter, group actions play a key role in determining the geometric properties of a manifold. Designing operations that are either invariant or equivariant to a group action has been a fruitful source of improvement to deal with data lying on manifolds (e.g. Pennec, Fillard, et al. 2006), and more recently to generalize convolutional neural networks (Cohen, Geiger, et al. 2019, and references therein).

In this chapter, we focus on the implementation of Riemannian metrics that verify invariance properties, either on Lie groups, or in the definition of quotient manifolds and quotient metrics. In the first case (Section 3.2), the invariance of the metric allows computing explicitly its Levi-Civita connection, yielding simplified

formulas for the geodesic equation and curvature. Although this work is largely inspired by the pioneering work of Milnor 1976 and Arnold 1966, we contribute to this topic by leveraging the well known equations of the connection to formulate the parallel transport of tangent vectors as a differential equation in the Lie algebra, in the spirit of the Euler-Poincaré equation for geodesics. This results in a stable and efficient implementation that is fully integrated into geomstats and is valid for both left and right invariant metrics. We exemplify these results with an anisotropic metric on the special Euclidean groups $SE(2)$ and $SE(3)$. The material and results of this section were presented at the GSI 2021 conference in Guigui and Pennec 2021a.

In the second case (Section 3.3), we propose a generic framework to implement quotient metrics, that are defined by a free and proper group action, and a metric that is invariant to this group action. This framework is exemplified in the case of Kendall shape spaces, and results in an efficient implementation that allows to compute parallel transport and curvature. This is, up to our knowledge, the first open-source Python implementation of such spaces. It was presented at the GSI 2021 conference in Guigui, Maignant, et al. 2021. Likewise, we implement the quotient metric on correlation matrices studied by Thanwerdas and Pennec 2021a.

This chapter thus gathers two contributions to the formulation and implementation of the parallel transport in the cases of invariant metrics and quotient metrics. Our original implementations allow to apply these results to numerous spaces. For the completeness of our exposure, we briefly present the particular cases of the invariant metrics on homogeneous and symmetric spaces in sections 3.4 and 3.5, and discuss the impact of these properties on the implementation.

3.2 Invariant metrics on Lie groups

Different geometric structures are compatible with the group structure. For instance, the group exponential defined in Section 2.4 can be defined as the exponential map of a connection on G, implying that geodesics are one-parameter subgroups. This connection is known as the canonical Cartan connection, and is of practical interest as in linear groups geodesics can be computed in closed-form using the matrix exponential.

The canonical Cartan connection may however not be the Levi-Civita connection of any metric, for more details we refer the reader to Pennec, Sommer, et al. 2020, Chapter 5. In this section we focus on the study of invariant metrics on Lie groups, i.e. metrics for which the left (or right) translation map is an isometry. This case is fundamental in geometric mechanics (Kolev 2004) and has been studied in depth since the foundational papers of Arnold 1966 and Milnor 1976. Using left-invariant vector fields, one can compute explicitly the LC connection, allowing to rewrite the geodesic equation. This fundamental idea, known as Euler-Poincaré reduction, is that the geodesic equation can be expressed entirely in the Lie algebra thanks to the symmetry of left-invariance (Marsden and Ratiu 2009), alleviating the burden

of coordinate charts.

We derive here a similar reduction of the parallel transport equation, resulting in a stable and efficient implementation of parallel transport in geomstats. Finally, knowing the connection, one can also compute algebraically the curvature of the space.

3.2.1 Definitions and computations of the connection

Let G be a Lie group. Recall that the Lie algebra of left-invariant vector fields $\mathcal{L}(G)$ and the tangent space at the identity $\mathfrak{g}$ can be identified, and in this section we will write $\tilde{x}$ the left-invariant field generated by $x \in \mathfrak{g}$: $\forall g \in G$, $\tilde{x}_g = dL_g x$.

Definition 3.2.1 (Invariant metric)**.** Let G be a Lie group. A Riemannian metric $\langle \cdot, \cdot \rangle$ on G is called *left-invariant* if the differential map of the left translation is an isometry between tangent spaces, that is

$$\forall g, h \in G, \forall u, v \in T_g G, \quad \langle u, v \rangle_g = \langle dL_h u, dL_h v \rangle_{hg}.$$

It is thus uniquely determined by an inner product on the tangent space at the identity $T_e G = \mathfrak{g}$ of G. Similarly, a *right-invariant* metric is such that the differential of the right translation is an isometry. A metric is called *bi-invariant* if it is both left and right-invariant.

Moreover, let $(e_1, \ldots, e_n)$ be an orthonormal basis of $\mathfrak{g}$, and the associated left-invariant vector fields $\tilde{e}_i$. As dL_g is an isomorphism, $(\tilde{e}_{1,g}, \ldots, \tilde{e}_{n,g})$ form a basis of $T_g G$ for any $g \in G$, so for any $X \in \Gamma(G)$ one can write $X_g = f^i(g)\tilde{e}_{i,g}$ where for $i = 1, \ldots, n$ $g \mapsto f^i(g)$ is a smooth real-valued function on G. Any vector field on G can thus be expressed as a linear combination of the $\tilde{e}_i$ with functional coefficients.

Example 3.2.1: Special Orthogonal

Consider the special orthogonal group endowed with the restriction of the Frobenius metric of Example 2.3.5. Let $P, Q \in SO(n)$ and $A, B \in T_P SO(n) = P\,\mathrm{Skew}(n)$. Then it is clear that

$$\langle dL_Q A, dL_Q B \rangle = \mathrm{tr}((QA)^\top QB) = \mathrm{tr}(A^\top B) = \langle A, B \rangle,$$

and similarly that $\langle dR_Q A, dR_Q B \rangle = \langle A, B \rangle$. Therefore, the Frobenius metric is bi-invariant on $SO(n)$.

Example 3.2.2: Special Euclidean

Now, consider the special Euclidean group endowed with the restriction of the Frobenius metric. As in the previous example, it is easy to show that it is left-invariant. However, it is not right-invariant. Indeed, let $g = (P, t) \in$

$SE(n)$ and $x = (S, u), y = (T, v) \in \mathfrak{se}(n) = \mathrm{Skew}(n) \oplus \mathbb{R}^n$. Then

$$\begin{aligned}\langle dR_g x, dR_g y\rangle &= \langle \begin{pmatrix} SP^\top & u - SP^\top t \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} TP^\top & v - TP^\top t \\ 0 & 0 \end{pmatrix} \rangle \\ &= \mathrm{tr}(S^\top T) + u^T v + t^\top P S^\top T P^\top t \\ &= \langle x, y\rangle + t^\top P S^\top T P^\top t\end{aligned}$$

So that one can find $t \neq 0$ and $S, T \in \mathrm{Skew}(n)$ such that $\langle dR_g x, dR_g y\rangle \neq \langle x, y\rangle$. Therefore the metric is not bi-invariant. In fact one can show that there does not exist any bi-invariant metric on $SE(n)$ for $n \neq 1$, and there exists a bi-invariant pseudo-metric for $n = 3$ (Miolane and Pennec 2015).

Definition 3.2.2 (Dual adjoint)**.** Define the metric dual adjoint map on $\mathfrak{g}$ as the unique map that verifies

$$\forall a, b, c \in \mathfrak{g}, \langle \mathrm{ad}_a^*(b), c\rangle = \langle b, \mathrm{ad}_a(c)\rangle = \langle [a, c], b\rangle.$$

As the bracket can be computed explicitly in the Lie algebra, so can ad^* thanks to the orthonormal basis of $\mathfrak{g}$. Now let ∇ be the Levi-Civita connection associated to the metric. It is also left-invariant and can be characterized by a bi-linear form on $\mathfrak{g}$ that verifies $\forall x, y \in \mathfrak{g}$ (Pennec and Arsigny 2012; Gallier and Quaintance 2020):

$$\alpha(x, y) \triangleq (\nabla_{\tilde{x}} \tilde{y})_e = \frac{1}{2}\big([x, y] - \mathrm{ad}_x^*(y) - \mathrm{ad}_y^*(x)\big) \tag{3.1}$$

Indeed by the left-invariance, for two left-invariant vector fields $X = \tilde{x}, Y = \tilde{y} \in \mathcal{L}(G)$, the map $g \mapsto \langle X, Y\rangle_g$ is constant, so for any vector field $Z = \tilde{z}$ we have $Z(\langle X, Y\rangle) = 0$. Kozsul formula thus becomes

$$\begin{aligned}2\langle \nabla_X Y, Z\rangle &= \langle [X, Y], Z\rangle - \langle [Y, Z], X\rangle - \langle [X, Z], Y\rangle \\ 2\langle \nabla_X Y, Z\rangle_e &= \langle [x, y], z\rangle_e - \langle \mathrm{ad}_y(z), x\rangle_e - \langle \mathrm{ad}_x(z), y\rangle_e \\ 2\langle \alpha(x, y), z\rangle_e &= \langle [x, y], z\rangle_e - \langle \mathrm{ad}_y^*(x), z\rangle_e - \langle \mathrm{ad}_x^*(y), z\rangle_e.\end{aligned} \tag{3.2}$$

Definition 3.2.3 (Structure constants)**.** Let G be a Lie group and $g = \langle \cdot, \cdot\rangle$ be a left-invariant metric on G. Let $(e_1, \ldots, e_d)$ be an orthonormal basis of $\mathfrak{g}$ for g. Define the *structure constants* as

$$C_{ij}^k = \langle [e_i, e_j], e_k\rangle.$$

Example 3.2.3: SO(3)

An orthonormal basis of $\mathfrak{so}(3)$ endowed with the Frobenius metric is

$$a_1 = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} \qquad a_2 = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix} \qquad a_3 = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

.

And as the Lie bracket is the matrix commutator, we can compute the structure constants to obtain

$$C_{ij}^k = \frac{1}{\sqrt{2}} \text{ if } ijk \text{ is a direct cycle of } \{1,2,3\},$$

and 0 otherwise.

Example 3.2.4: SE(3)

We define a left-invariant metric on the special Euclidean group by defining an inner product in its Lie algebra. Let the metric matrix at the identity be diagonal: $g = \mathrm{diag}(1,1,1,\beta,1,1)$ for some $\beta > 0$, an anisotropy parameter. For $\beta = 1$, this metric coincides with the Frobenius metric. An orthonormal basis of the Lie algebra $\mathfrak{se}(3)$ for this metric is

$$e_1 = \begin{pmatrix} a_1 & 0 \\ 0 & 0 \end{pmatrix} \qquad e_2 = \begin{pmatrix} a_2 & 0 \\ 0 & 0 \end{pmatrix} \qquad e_3 = \begin{pmatrix} a_3 & 0 \\ 0 & 0 \end{pmatrix}$$
$$e_4 = \frac{1}{\sqrt{\beta}}\begin{pmatrix} 0 & \epsilon_1 \\ 0 & 0 \end{pmatrix} \qquad e_5 = \begin{pmatrix} 0 & \epsilon_2 \\ 0 & 0 \end{pmatrix} \qquad e_6 = \begin{pmatrix} 0 & \epsilon_3 \\ 0 & 0 \end{pmatrix}.$$

where (a_1, a_2, a_3) is the basis of $\mathfrak{so}(3)$ defined above, and $(\epsilon_1, \epsilon_2, \epsilon_3)$ is the canonical basis of $\mathbb{R}^3$. As the Lie bracket is the usual matrix commutator, it is straightforward to compute

$$C_{ij}^k = \frac{1}{\sqrt{2}} \text{ if } ijk \text{ is a direct cycle of } \{1,2,3\}; \tag{3.3}$$
$$C_{15}^6 = -C_{16}^5 = -\sqrt{\beta}C_{24}^6 = \frac{1}{\sqrt{\beta}}C_{26}^4 = \sqrt{\beta}C_{34}^5 = -\frac{1}{\sqrt{\beta}}C_{35}^4 = \frac{1}{\sqrt{2}}. \tag{3.4}$$

and all others that cannot be deduced by skew-symmetry of the bracket are equal to 0.

The structure constants allow to compute ad* and formula (3.1) in practice:

$$\alpha(e_i, e_j) = \nabla_{e_i} e_j = \frac{1}{2}\sum_k (C_{ij}^k - C_{jk}^i + C_{ki}^j)e_k.$$

Note however that (3.1) is the connection for left-invariant vector fields only. We

will now generalize to any vector field defined along a smooth curve on G, using the left-invariant basis $(\tilde{e}_1, \dots, \tilde{e}_n)$.

Let $\gamma : [0,1] \to G$ be a smooth curve, and Y a vector field defined along γ. Write $Y = h^i \tilde{e}_i$, $\dot{\gamma} = f^i \tilde{e}_i$. Let's also define the *left-angular velocities* $\omega = dL_\gamma^{-1}\dot{\gamma} = f^i \circ \gamma e_i \in \mathfrak{g}$ and $\zeta = dL_\gamma^{-1} Y = h^j \circ \gamma e_j \in \mathfrak{g}$. Then the covariant derivative of Y along γ is

$$\begin{aligned}
\nabla_{\dot{\gamma}(t)} Y &= (f^i \circ \gamma)(t) \nabla_{\tilde{e}_i}\big(h^j \tilde{e}_j\big) \\
&= (f^i \circ \gamma)(t) \tilde{e}_i(h^j) \tilde{e}_j + (f^i \circ \gamma)(t)(h^j \circ \gamma)(t) (\nabla_{\tilde{e}_i} \tilde{e}_j)_{\gamma(t)} \\
dL^{-1}_{\gamma(t)} \nabla_{\dot{\gamma}(t)} Y &= (f^i \circ \gamma)(t) \tilde{e}_i(h^j) e_j + (f^i \circ \gamma)(t)(h^j \circ \gamma)(t) dL^{-1}_{\gamma(t)} (\nabla_{\tilde{e}_i} \tilde{e}_j)_{\gamma(t)} \\
&= (f^i \circ \gamma)(t) \tilde{e}_i(h^j) e_j + (f^i \circ \gamma)(t)(h^j \circ \gamma)(t) \nabla_{e_i} e_j
\end{aligned}$$

where the Leibniz formula and the invariance of the connection is used in $(\nabla_{\tilde{e}_i} \tilde{e}_j) = dL_{\gamma(t)} \nabla_{e_i} e_j$. Therefore, for $k = 1..n$

$$\begin{aligned}
\langle dL^{-1}_{\gamma(t)} \nabla_{\dot{\gamma}(t)} Y, e_k \rangle &= (f^i \circ \gamma)(t) \tilde{e}_i(h^j) \langle e_j, e_k \rangle \\
&\quad + (f^i \circ \gamma)(t)(h^j \circ \gamma)(t) \langle \nabla_{e_i} e_j, e_k \rangle
\end{aligned}$$

but on the one hand

$$\begin{aligned}
\zeta(t) &= (h^j \circ \gamma)(t) e_j \\
\dot{\zeta}(t) &= (h^j \circ \gamma)'(t) e_j = dh^j_{\gamma(t)} \dot{\gamma}(t) e_j \\
&= dh^j_{\gamma(t)} \Big((f^i \circ \gamma)(t) \tilde{e}_{i,\gamma(t)} \Big) e_j \\
&= (f^i \circ \gamma)(t) d_{\gamma(t)} h^j \tilde{e}_{i,\gamma(t)} e_j \\
&= (f^i \circ \gamma)(t) \tilde{e}_i(h^j)_{\gamma(t)} e_j
\end{aligned}$$

and on the other hand, using (3.1):

$$\begin{aligned}
(f^i \circ \gamma)(h^j \circ \gamma) \langle \nabla_{e_i} e_j, e_k \rangle &= (f^i \circ \gamma)(h^j \circ \gamma) \Big(\langle [e_i, e_j], e_k \rangle - \langle \mathrm{ad}^*_{e_i} e_j, e_k \rangle - \langle \mathrm{ad}^*_{e_j} e_i, e_k \rangle \Big) \\
&= \frac{1}{2} (f^i \circ \gamma)(h^j \circ \gamma) \Big(\langle [e_i, e_j], e_k \rangle - \langle [e_j, e_k], e_i \rangle - \langle [e_i, e_k], e_j \rangle \Big) \\
&= \frac{1}{2} (\langle [(f^i \circ \gamma) e_i, (h^j \circ \gamma) e_j], e_k \rangle \\
&\qquad - \langle [(h^j \circ \gamma) e_j, e_k], (f^i \circ \gamma) e_i \rangle \\
&\qquad - \langle [(f^i \circ \gamma) e_i, e_k], (h^j \circ \gamma) e_j \rangle) \\
&= \langle \frac{1}{2} \left([\omega, \zeta] - \mathrm{ad}^*_\omega \zeta - \mathrm{ad}^*_\zeta \omega \right), e_k \rangle = \langle \alpha(\omega, \zeta), e_k \rangle.
\end{aligned}$$

Thus, we obtain an algebraic expression for the covariant derivative of any vector field Y along a smooth curve γ. It is the main ingredient of this section.

Lemma 3.2.1. *Let G be a Lie group and ∇ be the Levi-Civita connection of a left-invariant metric on G. Let γ be a smooth curve on G, and Y a vector field defined along γ. Consider the left-angular velocities $\omega = dL_\gamma^{-1}\dot{\gamma}$ and $\zeta = dL_\gamma^{-1} Y$. Then*

$$dL^{-1}_{\gamma(t)} \nabla_{\dot{\gamma}(t)} Y(t) = \dot{\zeta}(t) + \alpha(\omega(t), \zeta(t)) \tag{3.5}$$

A similar expression can be found in Arnold 1966; Gay-Balmaz, Holm, et al. 2012. As all the variables of the right-hand side are defined in $\mathfrak{g}$, they can be computed with matrix operations and an orthonormal basis.

We now focus on two particular cases of (3.5) to derive the equations of geodesics and of parallel transport along a curve.

3.2.2 Geodesic equation

The first particular case is for $Y(t) = \dot{\gamma}(t)$. It is then straightforward to deduce from (3.5) the Euler-Poincaré equation for a geodesic curve (e.g Kolev 2004; Cendra, Holm, et al. 1998). Indeed in this case, recall that $\omega = dL_\gamma^{-1}\dot{\gamma}$ is the left-angular velocity, $\zeta = \omega$ and $\alpha(\omega, \omega) = -\operatorname{ad}^*_\omega(\omega)$. Hence γ is a geodesic if and only if $dL^{-1}_{\gamma(t)}\nabla_{\dot{\gamma}(t)}\dot{\gamma}(t) = 0$ i.e. setting the left-hand side of (3.5) to 0. We obtain

$$\begin{cases} \dot{\gamma}(t) &= dL_{\gamma(t)}\omega(t) \\ \dot{\omega}(t) &= \operatorname{ad}^*_{\omega(t)}\omega(t). \end{cases} \tag{3.6}$$

Remark 3.2.1.

- *A similar treatment of a right-invariant metric is straightforward. Indeed, in this case the Lie bracket of right-invariant vector fields is the opposite of the adjoint representation of the Lie algebra, therefore the expressions are all the same with $-\alpha$ instead of α.*

- *One can show that the metric is bi-invariant if and only if the adjoint map is skew-symmetric (see Pennec and Arsigny 2012 or Gallier and Quaintance 2020, Prop. 20.7). In this case $\operatorname{ad}^*_\omega(\omega) = 0$ and (3.6) coincides with the equation of one-parameter subgroups on G.*

In the class `InvariantMetric` of geomstats, the `geodesic_equation` method is modified to implement (3.6), and the integrators of Section 2.3.3 (page 42) are used to compute the exponential map.

Example 3.2.5: SE(2)

We use the same visualization as in Example 2.4.9 (page 64) to plot the geodesics of a left-invariant metric on $SE(2)$. We first compare the left and right-invariant metrics with same inner-product at the identity, and compare their geodesics to one parameter subgroups. These curves are shown on Figure 3.1.

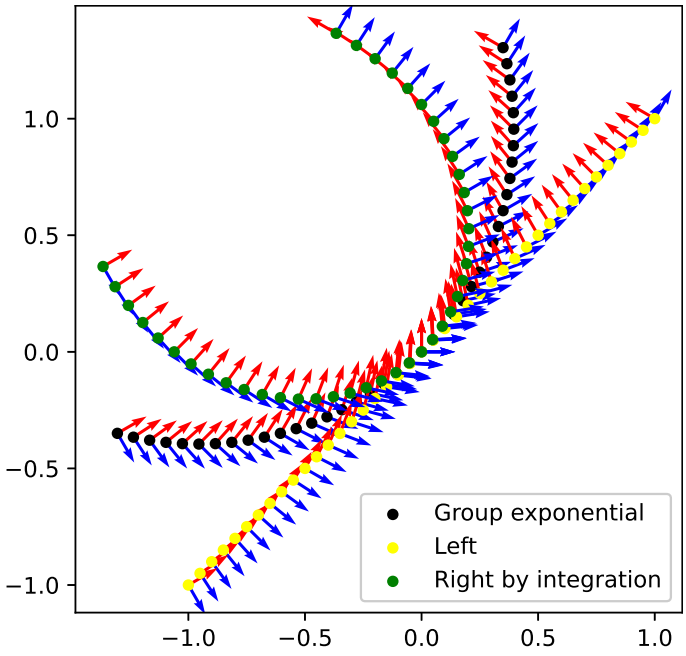


Figure 3.1: Visualization of geodesics and one parameter subgroups on $SE(2)$.

Next, we define the metric as in Example 3.2.4, by the bilinear form defined in $\mathfrak{se}(2)$ by the matrix $g = \mathrm{diag}(1, \beta, 1)$ for some $\beta > 0$, an anisotropy parameter, and extended by left-invariance.

For $\beta = 1$, the metric coincides with the Frobenius metric, and there is no interaction between the rotation and translation part. This metric is thus isomorphic to the direct product metric of the bi-invariant metric on $SO(2)$ and the usual inner product of $\mathbb{R}^2$. The geodesics are thus straight-lines on the translation part, and rotations with constant speed (as one-parameter subgroups) on the rotation part.

When $\beta \neq 1$, the rotation part remains the same, but there is a direction towards which less movement occurs. This is shown on Figure 3.2.

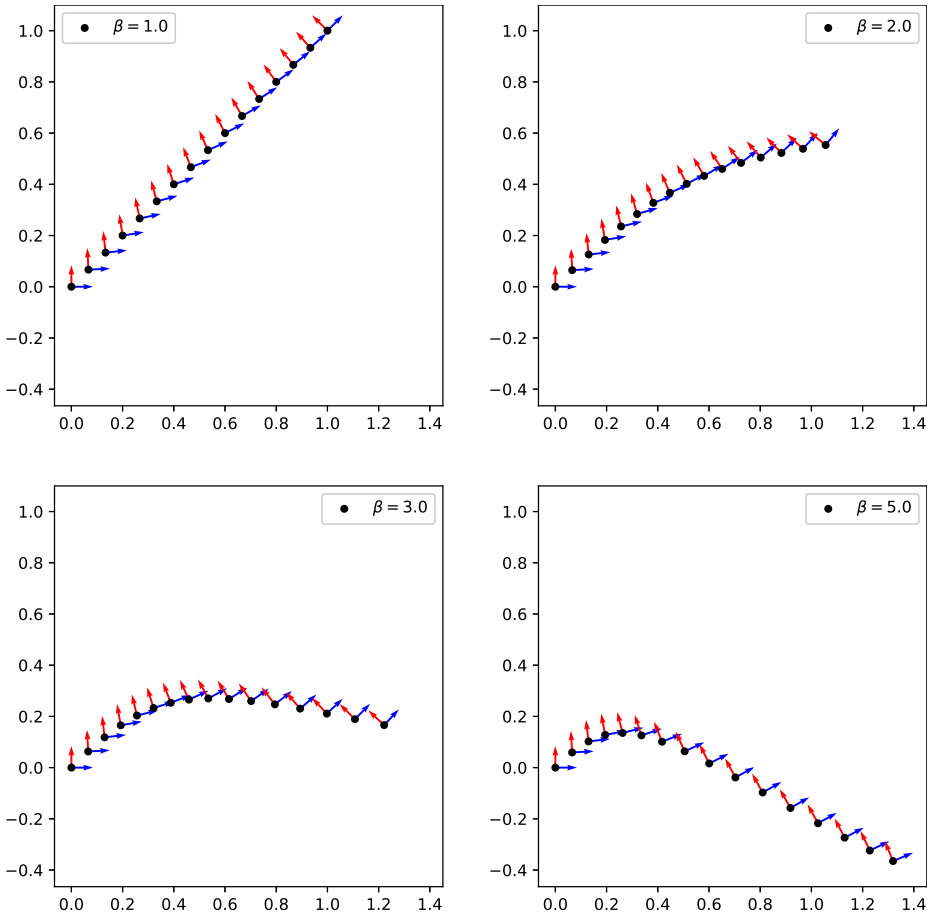


Figure 3.2: Visualization of geodesics on $SE(2)$ with the same initial conditions and different values of β.

3.2.3 Reduced parallel transport equation

The second case is for a vector field Y that is parallel along the curve γ, that is, $\forall t, \nabla_{\dot\gamma(t)} Y(t) = 0$. Similarly to the geodesic equation, we deduce from (3.5) the parallel transport equation expressed in the Lie algebra. To the best of our knowledge, this formulation of the parallel transport on Lie groups with an invariant metric is original.

Theorem 3.2.1. *Let γ be a smooth curve on G. The vector Y is parallel along γ if and only if it is solution to*

$$\begin{cases} \omega(t) &= dL^{-1}_{\gamma(t)}\dot\gamma(t) \\ Y(t) &= dL_{\gamma(t)}\zeta(t) \\ \dot\zeta(t) &= -\alpha(\omega(t), \zeta(t)) \end{cases} \tag{3.7}$$

Note that in order to parallel transport along a geodesic curve, (3.6) and (3.7) are solved jointly. We add the corresponding method to the `InvariantMetric` class of geomstats.

```
def geodesic_equation(self, state, _time):
    """Compute the right-hand side of the geodesic equation."""
    sign = 1. if self.left_or_right == 'left' else -1.
    basis = self.normal_basis(self.lie_algebra.basis)

    point, vector = state
    velocity = self.group.tangent_translation_map(
        point, left_or_right=self.left_or_right)(vector)
    coefficients = gs.array([self.structure_constant(
        vector, basis_vector, vector) for basis_vector in basis])
    acceleration = gs.einsum('i...,ijk->...jk', coefficients, basis)
    return gs.stack([velocity, sign * acceleration])

def parallel_transport(
        self, tangent_vec_a, tangent_vec_b, base_point, n_steps=10,
        step='rk4', **kwargs):
    """Compute the parallel transport of a tangent vector along a geodesic."""
    group = self.group
    translation_map = group.tangent_translation_map(
        base_point,
        left_or_right=self.left_or_right, inverse=True)
    left_angular_vel_a = group.to_tangent(translation_map(tangent_vec_a))
    left_angular_vel_b = group.to_tangent(translation_map(tangent_vec_b))

    def acceleration(state, time):
        point, omega, zeta = state
        gam_dot, omega_dot = self.geodesic_equation(state[:2], time)
        zeta_dot = - self.connection_at_identity(omega, zeta)
        return gs.stack([gam_dot, omega_dot, zeta_dot])

    if (base_point.ndim == 2 or base_point.shape[0] == 1) and \
            (tangent_vec_a.ndim == 3 or tangent_vec_b.ndim == 3):
        n_sample = tangent_vec_a.shape[0] if tangent_vec_a.ndim == 3 else\
            tangent_vec_b.shape[0]
        base_point = gs.stack([base_point] * n_sample)

    initial_state = gs.stack([
        base_point, left_angular_vel_b, left_angular_vel_a])
    flow = integrate(
        acceleration, initial_state, n_steps=n_steps, step=step, **kwargs)
    gamma, gamma_dot, zeta_t = flow[-1]
    transported = group.tangent_translation_map(
        gamma, left_or_right=self.left_or_right, inverse=False)(zeta_t)
    return transported
```

Example 3.2.6: Parallel Transport in $SE(3)$

We exemplify the use of the reduced parallel transport equation on $SE(3)$ endowed with an anisotropic left-invariant metric, as in Examples 3.2.4 and 3.2.5. The metric is defined by the matrix $g = \mathrm{diag}(1, 1, 1, \beta, 1, 1)$ for some $\beta > 0$ at the identity and extended by left-invariance.

We randomly generate a point in $x \in SE(3)$ and two tangent vectors $v, w \in T_x SE(3)$, and transport the vector v along the geodesic $t \mapsto \mathrm{Exp}_x(tw)$.

When $\beta = 1$, it coincides with the product metric between the bi-invariant metric of $SO(3)$ and of $\mathbb{R}^3$, so the geodesics and parallel transport are known in closed form. For $\beta > 1$, the curvature grows away from 0 (this is proved in Appendix C.3), and so does its covariant derivative, so that parallel transport becomes more complex to compute. we use $n = 1100$ steps in the discretization of $[0, 1]$ for integration scheme to compute a reference value that is then used to measure the error of the method for $10 \leq n \leq 1000$. We use different values of β but keep the same initial vectors regardless of the value of β.

The results are plotted in a loglog plot, with the error shown with respect to the number of steps used to integrate (3.6) and (3.7), with an $RK2$ or $RK4$ scheme. As expected, the speed of convergence depends on the order of the scheme used: quadratic speed is reached for $RK2$ and speed of order 4 for $RK4$. The method is very stable for large n, and shows little sensibility to a change of metric (i.e. when β grows).

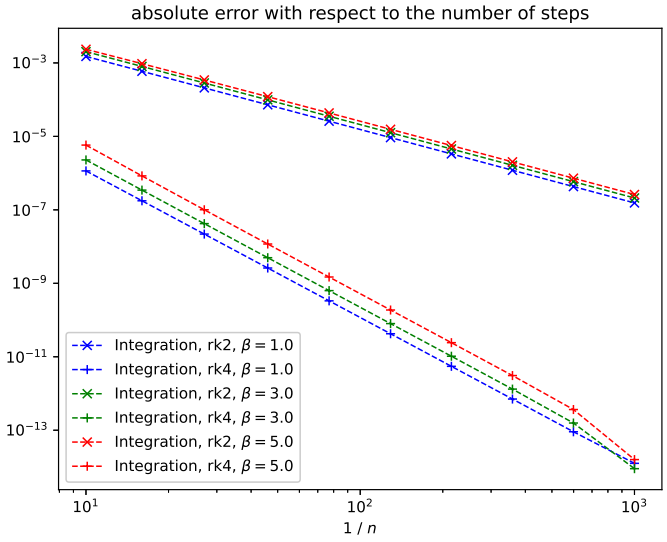


Figure 3.3: Norm of the absolute error represented with respect to the number of steps.

3.2.4 Curvature

By definition of the left-invariance of the metric, for any $g \in G$, the left-translation map L_g is an isometry. It is thus sufficient to compute the curvature tensor at the identity, that is, in the Lie algebra $\mathfrak{g}$, and to map it to any other point. Using an orthonormal basis and formula (3.1), this reduces to simple algebra. The covariant derivative of the curvature tensor can also be computed using Leibniz rule, for any $u, v, w, z \in \mathfrak{g}$:

$$(\nabla_z R)(u,v)w = \nabla_z\big(R(u,v)w\big) - R(\nabla_z u, v)w - R(u, \nabla_z v)w - R(u,v)\nabla_z w \quad (3.8)$$

Of course this is also valid for right-invariant metrics. This is implemented in geomstats in the class `InvariantMetric` as follows.

```
def connection_at_identity(self, tangent_vec_a, tangent_vec_b):
    """Compute the Levi-Civita connection at identity."""
    sign = 1. if self.left_or_right == 'left' else -1.
    return sign / 2 * (Matrices.bracket(tangent_vec_a, tangent_vec_b)
                       - self.dual_adjoint(tangent_vec_a, tangent_vec_b)
                       - self.dual_adjoint(tangent_vec_b, tangent_vec_a))

def curvature_at_identity(
        self, tangent_vec_a, tangent_vec_b, tangent_vec_c):
    """Compute the curvature at identity."""
    bracket = Matrices.bracket(tangent_vec_a, tangent_vec_b)
    bracket_term = self.connection_at_identity(bracket, tangent_vec_c)

    left_term = self.connection_at_identity(
        tangent_vec_a, self.connection_at_identity(
            tangent_vec_b, tangent_vec_c))

    right_term = self.connection_at_identity(
        tangent_vec_b, self.connection_at_identity(
            tangent_vec_a, tangent_vec_c))

    return left_term - right_term - bracket_term

def sectional_curvature_at_identity(self, tangent_vec_a, tangent_vec_b):
    """Compute the sectional curvature at identity."""
    curvature = self.curvature_at_identity(
        tangent_vec_a, tangent_vec_b, tangent_vec_b)
    num = self.inner_product(curvature, tangent_vec_a)
    denom = (
            self.squared_norm(tangent_vec_a)
            * self.squared_norm(tangent_vec_a)
            - self.inner_product(tangent_vec_a, tangent_vec_b) ** 2)
    condition = gs.isclose(denom, 0.)
    denom = gs.where(condition, 1., denom)
    return gs.where(~condition, num / denom, 0.)

def curvature_derivative_at_identity(
        self, tangent_vec_a, tangent_vec_b, tangent_vec_c, tangent_vec_d):
```

```
    """Compute the covariant derivative of the curvature at identity."""
    first_term = self.connection_at_identity(
        tangent_vec_a,
        self.curvature_at_identity(
            tangent_vec_b, tangent_vec_c, tangent_vec_d))

    second_term = self.curvature_at_identity(
        self.connection_at_identity(tangent_vec_a, tangent_vec_b),
        tangent_vec_c,
        tangent_vec_d)

    third_term = self.curvature_at_identity(
        tangent_vec_b,
        self.connection_at_identity(tangent_vec_a, tangent_vec_c),
        tangent_vec_d)

    fourth_term = self.curvature_at_identity(
        tangent_vec_b,
        tangent_vec_c,
        self.connection_at_identity(tangent_vec_a, tangent_vec_d))

    return first_term - second_term - third_term - fourth_term
```

If the metric is bi-invariant, it is well known that these formulas greatly simplify (Lafontaine, Gallot, et al. 2004, Proposition 3 .17) or (Gallier and Quaintance 2020, Proposition 20.19).

Proposition 3.2.1. *If G is a Lie group equipped with a bi-invariant metric, and if $X, Y, Z, T \in \mathcal{L}(G)$, then*

$$\langle R(X,Y)Z, T\rangle = \frac{1}{4}\langle [X,Y], [Z,T]\rangle \tag{3.9}$$

In particular, G has non-negative sectional curvature.

We now turn to the case of submersions defined by the action of a Lie group on a manifold, and the properties of the induced quotient metric.

3.3 Submersions and quotient metrics

A particularly interesting setting is when there exists a submersion $\pi : E \to M$ between two Riemannian manifolds E and M, such that π is compatible with the two metrics. In that case, most of the geometry of M can be deduced from the geometry of E and the knowledge of π. Such submersions arise for homogeneous spaces, and for quotients by a Lie group, hence we shall revisit some examples from the previous sections with this viewpoint. We first introduce the general notions, then focus on quotient metrics. Metrics on homogeneous spaces will be treated in Section 3.4.

Although the mathematical notions exposed in this section are well known since the works of O'Neill 1966 and Le and D. G. Kendall 1993, this work is original by the common implementation of these structures in geomstats. This implementation allowed computing parallel transport on Kendall shape spaces (Subsection 3.3.3), outperforming the state-of-the-art implementations, and are available on other quotient spaces with limited additional software development. The generic implementation is presented in Subsection 3.3.4, and constitutes one of the main contributions of this section.

To exemplify this section, we extensively use the work done in collaboration with Elodie Maignant on Kendall shape spaces, and presented at the GSI 2021 conference in Guigui, Maignant, et al. 2021. We also use the Bures-Wasserstein metric, with results from Bhatia, Jain, et al. 2019 and following Thanwerdas and Pennec 2021b.

3.3.1 Riemannian submersions

Throughout this section, let (E, g) and (M, h) be two Riemannian manifolds, and $\pi : E \to M$ be a submersion. Recall that this means that at every $p \in E$, $d\pi_p$ is surjective. We adopt the vocabulary of principal fiber bundles, referring to E as the *total* space, and to M as the *base* manifold. In particular, for every $x \in M$, $\pi^{-1}(x) \subset E$ is a submanifold in E usually called *fiber* above x, and the tangent space at any $p \in \pi^{-1}(x)$ is $T_p\pi^{-1}(x) = \ker d\pi_p$.

Definition 3.3.1 (Section)**.** A *section* σ of $\pi : E \to M$ is a smooth map $\sigma : U \to E$ defined on an open set $U \subseteq M$ of M, such that $\forall x \in M, \quad \pi(\sigma(x)) = x$. If $U = M$, we say that σ is a *global* section.

A section allows to choose a point in a fiber above any x in a small open set. In the sequel, we assume that π always admits local sections. This is true for Riemannian submersions in particular by the inverse function theorem.

3.3.1.1 Geodesics

Definition 3.3.2 (Horizontal - Vertical subspaces)**.** Let $x \in M$ and $p \in \pi^{-1}(x)$. The *vertical* subspace of T_pE is defined as $V_p = \ker d\pi_p$ and is the tangent space to the fiber through p. The *horizontal* subspace is its orthogonal complement in T_pE: $H_p = (V_p)^{\perp}$.

Any tangent vector $u \in T_pE$ can thus be decomposed into a vertical and a horizontal component, and we write ver and hor respectively the orthogonal projections on the vertical and horizontal subspace. Note that this decomposition depends on the metric on E by the definition of the orthogonal complement for the horizontal subspace. Furthermore, we say that u is horizontal if $u \in H_p$, and a curve $c : I \to E$ is called horizontal if at every time $t \in I$, the velocity of c is horizontal: $c'(t) \in H_{c(t)}$.

Example 3.3.1: SPD matrices with Bures-Wasserstein metric

Recall from Example 2.4.13 (page 62) that the map $A \in GL(n) \mapsto AA^\top \in SPD(n)$ is a smooth submersion. Its differential at any A is for any $H \in M_n(\mathbb{R})$:

$$d\pi_A H = AH^\top + HA^\top,$$

and its kernel is $\ker d\pi_A = \{H \mid HA^\top \in \mathrm{Skew}(n)\} = \mathrm{Skew}(n)A^{-\top}$. This defines the vertical subspace V_A.

Now, consider $GL(n)$ endowed with the restriction of the Frobenius metric. The orthogonal complement to V_A is

$$\begin{aligned} H_A &= \{H \in M_n(\mathbb{R}) \mid \forall B \in \mathrm{Skew}(n),\ \ \langle H, BA^{-\top}\rangle = 0\} \\ &= \{H \in M_n(\mathbb{R}) \mid \forall B \in \mathrm{Skew}(n),\ \ \mathrm{tr}(BA^{-\top}H^\top) = 0\} \\ &= \{H \in M_n(\mathbb{R}) \mid HA^{-1} \in \mathrm{Sym}(n)\} = \mathrm{Sym}(n)A. \end{aligned}$$

Note that in Section 3.5, we will consider the same manifolds and submersion, but instead endow $GL(n)$ with a left-invariant metric. This will result in another geometry on the manifold of SPD matrices.

Recall that as π is a submersion, $d\pi_p$ is a linear isomorphism between H_p and $\mathrm{Im}(\pi) = T_xM$. If this identification is additionally isometric, then geodesics and curvature on M will be deduced from E.

Definition 3.3.3 (Riemannian submersion)**.** Let $\pi : E \to M$ be a smooth submersion between two Riemannian manifolds. π is called a *Riemannian submersion* if for every $p \in E$, the map $d\pi_p$ is an isometry between the horizontal subspace H_p of T_pE and T_xM where $x = \pi(p)$.

If π is also surjective, this allows to identify tangent vectors of M to horizontal vectors of E.

Definition 3.3.4 (Horizontal lift)**.** Let $\pi : E \to M$ be a Riemannian submersion that is surjective onto M. Let X be a vector field on M, then the unique *horizontal lift* $\bar{X}$ in E is defined such that for every $x \in M$ and every $p \in \pi^{-1}(x)$,

$$d\pi_p \bar{X}_p = X_x.$$

If we have a local section σ, we speak about the *corresponding* horizontal lift such that $\forall (x, v) \in TM, \bar{v} \in H_{\sigma(x)} \subset T_{\sigma(x)}E$.

Example 3.3.2: SPD matrices

Following Example 3.3.1, it is known from linear algebra that the submersion $\pi : A \in GL(n) \mapsto AA^\top \in SPD(n)$ is surjective. To compute the horizontal lift of a tangent vector X at $\Sigma = AA^\top$, we seek a symmetric matrix S such that $d\pi_A(SA) = X$. This is true if and only if $AA^\top S + SAA^\top = X$,

i.e. if $S \in \mathrm{Sym}(n)$ solves the Sylvester equation

$$\Sigma S + S\Sigma = X. \tag{3.10}$$

More precisely, using the eigenvalue decomposition of $\Sigma = PDP^\top$, S solves (3.10) iff

$$\begin{aligned} & \Sigma S + S\Sigma = X \\ \iff \quad & DP^\top SP + P^\top SPD = P^\top XP \\ \iff \quad & (P^\top SP)_{ij} = \frac{(P^\top XP)_{ij}}{d_i + d_j}. \end{aligned} \tag{3.11}$$

Define the map $S_\Sigma : \mathrm{Sym}(n) \to \mathrm{Sym}(n)$ that solves Sylvester equation using equation (3.11). It uniquely defines the horizontal lift $\bar{X}$ of X at A by $\bar{X} = S_{AA^\top}(X)A$.

Proposition 3.3.1. *Similarly, if $c : [0,1] \to M$ is a piecewise smooth curve in M, and $p \in \pi^{-1}(c(0))$, then there exists a unique curve $\bar{c} : [0,1] \to E$ such that $\pi \circ \bar{c} = c$ and $\bar{c}' \in H$. The curve $\bar{c}$ is called horizontal lift of c.*

This leads to the main theorem of this section, that is usually attributed to O'Neill, and relates geodesics of M with horizontal geodesics of E (Gallier and Quaintance 2020, Prop. 17.8).

Theorem 3.3.1. *Let $\pi : E \to M$ be a Riemannian submersion between two Riemannian manifolds E and M.*

(1) If γ is a geodesic in E such that $c'(0)$ is horizontal, then c is horizontal, and its projection $\gamma = \pi \circ c$ is a geodesic in M of same length as c.

(2) For every $p \in E$, if γ is a geodesic in M such that $\gamma(0) = \pi(p)$, then there exists $\epsilon > 0$ such that there exists a unique horizontal lift c of the restriction of γ to $[-\epsilon, \epsilon]$ and c is a geodesic of E through p.

(3) If furthermore π is surjective, for any vector fields $X, Y \in \Gamma(TM)$, $\langle \bar{X}, \bar{Y} \rangle = \langle X, Y \rangle \circ \pi$ (where $\bar{\cdot}$ is the horizontal lift).

The first property of this theorem can be written by the commutative diagram below:

$$\begin{array}{ccc} H_p & \xrightarrow{\mathrm{Exp}_p} & E \\ \downarrow{\scriptstyle d\pi_p} & & \downarrow{\scriptstyle \pi} \\ T_xM & \xrightarrow{\mathrm{Exp}_x} & M \end{array}$$

where $p \in E$ and $x = \pi(p)$. Furthermore, as the restriction of $d\pi$ on horizontal spaces is an isomorphism, choosing a section σ of E and the corresponding horizontal lift

$\bar{\cdot}$ allows to compute the exponential map of M from the one of E:

$$\forall x \in M, \forall v \in T_x M, \quad \mathrm{Exp}_x(v) = \pi \circ \mathrm{Exp}_{\sigma(x)}(\bar{v}) \tag{3.12}$$

Remark 3.3.1.

- *One can also show that the connection $\nabla_{\bar{X}}\bar{Y}$ on E verifies (see e.g. Lafontaine, Gallot, et al. 2004, Proposition 3.55)*

$$\nabla_{\bar{X}}\bar{Y} = \overline{\nabla_X Y} + \frac{1}{2}\,\mathrm{ver}[\bar{X}, \bar{Y}].$$

  Consequently, one cannot hope to obtain a similar commutation rule between the parallel transport in E and the one in M. Indeed, if $\bar{Y}(t)$ is the horizontal lift of a parallel vector field $Y(t)$ along a curve γ (whose horizontal lift is $\bar{\gamma}$), then $\nabla_{\bar{\gamma}'}\bar{Y}$ has a non-zero vertical component given above, which vanishes in the case of a geodesic. Computing parallel transport is thus not as straightforward as computing geodesics in this case.

- *A Riemannian submersion shortens distances, i.e. for $p, q \in E$, and writing d_E and d_M the Riemannian distances on respectively E and M, we have*

$$d_M\big(\pi(p), \pi(q)\big) \leq d_E(p, q).$$

Example 3.3.3: Bures-Wasserstein geodesics

There is a unique metric that turns $\pi : A \in GL(n) \mapsto AA^\top \in SPD(n)$ into a Riemannian submersion between $GL(n)$ endowed with the Frobenius metric and $SPD(n)$. This metric is called the Bures-Wasserstein (BW) metric. Let's apply Theorem 3.3.1 to compute the geodesics of this metric. Recall that $(GL(n), \text{Frobenius})$ is flat, so the geodesics are "straight" lines: $\mathrm{Exp}_A(tX) = A + tX$. Let $\Sigma = AA^\top \in SPD(n)$ and $X \in T_\Sigma SPD(n)$, then $\forall t > 0$

$$\begin{aligned}
\mathrm{Exp}_\Sigma(tX) &= \pi \circ \mathrm{Exp}_A(t\bar{X}) = \pi(A + tS_\Sigma(X)A) \\
&= (A + tS_\Sigma(X)A)(A + tS_\Sigma(X)A)^\top \\
&= AA^\top + t(S_\Sigma(X)\Sigma + \Sigma S_\Sigma(X)) + t^2 S_\Sigma(X)\Sigma S_\Sigma(X) \\
&= AA^\top + tX + t^2 S_\Sigma(X)\Sigma S_\Sigma(X)
\end{aligned}$$

Note that a geodesic hits the boundaries of the manifold when $-1/t$ is in an eigenvalue of $S_\Sigma(X)$. $(SPD(n), \text{BW})$ is therefore not complete. All the properties of this metric are derived using the equations of Riemannian submersions in Thanwerdas and Pennec 2021b.

The relation between the connection of the total space and that of the base manifold also translates into a relation between the curvatures of each space.

3.3.1.2 Curvature

O'Neill 1966 showed that the curvatures of the total space and of the base space were related, and that those of the base space could be computed using two fundamental tensors defined by the horizontal and vertical projections of the connection. We state here the main result, that shows that sectional curvature can only increase after a Riemannian submersion.

Theorem 3.3.2 (O'Neill)**.** *Let $\pi : E \to M$ be a Riemannian submersion, and X, Y be orthonormal vector fields on M, with horizontal lifts $\bar{X}, \bar{Y}$. Then*

$$\kappa(X, Y) = \kappa(\bar{X}, \bar{Y}) + \frac{3}{4}\| \operatorname{ver}[\bar{X}, \bar{Y}]\|^2. \tag{3.13}$$

Example 3.3.4: Bures-Wasserstein

As $GL(n)$ with the Frobenius metric is flat, O'Neill's theorem implies that the space of SPD matrices endowed with the Bures-Wasserstein metric has non-negative curvature.

We now apply these results to the special case of submersions defined as the canonical projection to a quotient space.

3.3.2 Quotient metric

Recall from Theorem 2.4.4 (page 60) that if G is a Lie group and E a smooth manifold such that G acts on E and the action is smooth, free and proper, then the canonical projection $\pi : x \in E \mapsto [x] \in M = E/G$ is a submersion.

In this case, the action of G allows to move in the fibers. More precisely, as the fibers are defined as orbits $G \triangleright p = \{g \triangleright p \mid g \in G\}$ for $p \in E$, the fibers are stable by the action of G, i.e. $\forall x \in M, \forall g \in G, \forall p \in \pi^{-1}(x), g \triangleright p \in \pi^{-1}(x)$, which can also be written $\pi(g \triangleright p) = \pi(p) = x$. Moreover, this action is transitive on fibers, i.e. for any $q \in E$ such that $\pi(q) = x$, there exists $g \in G$ such that $q = g \triangleright p$.

Definition 3.3.5 (Invariant metric)**.** A Riemannian metric on E is *G-invariant* if for any $g \in G$, $L_g : p \in E \mapsto g \triangleright p \in E$ is an isometry.

Now suppose that E is equipped with a G-invariant. Then one can define a metric on the quotient manifold $M = E/G$ such that π is a Riemannian submersion (Lafontaine, Gallot, et al. 2004, Proposition 2.28).

Proposition 3.3.2 (Quotient metric)**.** *Let E be a smooth manifold, G a Lie group acting smoothly, properly and freely on G, and $\langle \cdot, \cdot \rangle$ be a G-invariant metric on E. Let $\pi : E \to M$ be the canonical projection. Then there exists a unique Riemannian metric on M such that π is a Riemannian submersion. Let $x \in M$ and $p \in \pi^{-1}(x)$, for any $u, v \in T_x M$, it is given by*

$$\langle u, v \rangle_x \triangleq \langle \bar{u}, \bar{v} \rangle_p. \tag{3.14}$$

We refer to this metric on M as the quotient metric.

Thanks to the invariance of the metric on E, the quotient metric on M is well defined and does not depend on the choice of p in the fiber above x. The quotient metric is the unique metric on M such that π is a Riemannian submersion. Note that we do not use different notations for the two metrics for simplicity, but the subscripts indicate in which space it is defined: the character p is preferred for a point in the total space E and x in the quotient manifold M. Let d_E and d_M denote the Riemannian distance of respectively $(E, \langle\cdot,\cdot\rangle)$ and $(M, \langle\cdot,\cdot\rangle)$.

Proposition 3.3.3. *Let $p, q \in E$. We have the following relation*

$$d_M\big(\pi(p), \pi(q)\big) = \inf_{q' \in \pi^{-1}(\pi(q))} \{d_E(p, q')\} = \inf_{g \in G} \{d_E(p, gq)\}. \tag{3.15}$$

This leads to the following definition.

Definition 3.3.6 (Align)**.** Let $p, q \in E$, we say that q is *aligned* or *well-positioned* with p if $d_E(p, q) = d_M(\pi(p), \pi(q))$. If p and q are close enough, then there exists a unique q' aligned with p, and the geodesic between p and q is horizontal.

We define the *align* map ω on a sufficiently small subset of $D \subseteq E^2$, such that for any $(x, y) \in D$, $\omega(x, y) \in E$ is aligned with x. In general, the optimization problem corresponding to (3.15) must be solved in G to compute the align map. This procedure is often referred to as alignment, Procrustes analysis or registration in the literature. We implement a general optimization procedure using the `minimize` method of scipy, that implements a gradient descent. If the group and its Lie algebra are explicitly given, we optimize in the Lie algebra and use the group exponential map to compute the action of a group element. A parameter represents the coefficient in a given basis of the Lie algebra, and `matrix_representation` maps these coefficient to the corresponding vector by linear combination.

```
def wrap(param):
    """Wrap a parameter vector to a group element."""
    algebra_elt = gs.array(param)
    algebra_elt = gs.cast(algebra_elt, dtype=base_point.dtype)
    algebra_elt = group.lie_algebra.matrix_representation(algebra_elt)
    group_elt = group.exp(algebra_elt)
    return group_action(point, group_elt)
```

If only the group action is given, we optimize without precautions on the group structure:

```
def wrap(param):
        vector = gs.array(param)
        vector = gs.cast(vector, dtype=base_point.dtype)
        return group_action(vector, point)
```

In both cases, automatic differentiation is used to compute the gradient of the objective function.

```
    def align(self, point, base_point,
              max_iter=25, verbose=False, tol=gs.atol):
        """Align point to base_point by minimization."""
                ...
        objective_with_grad = gs.autograd.value_and_grad(
            lambda param: self.ambient_metric.squared_dist(
                wrap(param), base_point))

        init_param = gs.flatten(gs.random.rand(*max_shape))
        res = minimize(
            objective_with_grad, init_param, method='L-BFGS-B', jac=True,
            options={'disp': verbose, 'maxiter': max_iter}, tol=tol)

        return wrap(res.x)
```

This map is particularly useful because it allows to compute the Log map of the quotient space from the one of the total space. Indeed, choosing a section σ and the corresponding horizontal lift $\bar{\cdot}$, we can deduce from Theorem 3.3.1 the following relation:

$$\forall x, y \in M, \mathrm{Log}_x(y) = d\pi_{\sigma(x)} \mathrm{Log}_x \left(\omega(x, y)\right). \tag{3.16}$$

Example 3.3.5: Bures-Wasserstein metric

Continuing Example 3.3.2 where the submersion $A \mapsto AA^\top$ from $GL(n)$ to $SPD(n)$, and explaining why the metric of Example 3.3.3 is correctly defined, we define the quotient metric of $GL(n)$ with the Frobenius metric by the orthogonal group. Indeed, consider the right-action of $O(n)$ on $GL(n)$ by matrix multiplication. As shown in Example 2.4.13 (page 62), the canonical projection of $GL(n)/O(n)$ coincides with $\pi : A \mapsto AA^\top$. The metric hereby defined on the manifold of SPD matrices is called the Bures-Wasserstein metric (Bhatia, Jain, et al. 2019). It arises naturally as the L_2-Wasserstein distance on multivariate centred normal random variables. As $GL(n)$ is Euclidean under the Frobenius metric, the geodesics and curvature are known and can be projected to the quotient space. The minimization problem of (3.15) can be solved in closed-form to obtain (Thanwerdas and Pennec 2021b).

$$d(\Sigma, \Sigma')^2 = \mathrm{tr}(\Sigma) + \mathrm{tr}(\Sigma') - 2\,\mathrm{tr}((\Sigma\Sigma')^{1/2}).$$

In the next subsection, we focus on Kendall shape spaces, as most of the geometry of these spaces can be computed using the results of this section. Our implementation in geomstats is, to the best of our knowledge, the first open-source Python implementation of Kendall shape spaces, and allows efficient computation of parallel transport.

3.3.3 Application to Kendall shape spaces

The following subsection is inspired by Nava-Yazdani, Hege, et al. 2020 but constitutes an original contribution by putting all the code open source in our package

geomstats, and by improving the implementation of parallel transport. It is a collaboration with Elodie Maignant and was presented at GSI 2021 in Guigui, Maignant, et al. 2021.

We revisit Example 2.2.5 (page 16) with the additional idea that the size of a set of landmarks may be filtered out to define a shape. The study of these spaces, including their mathematical structure, the properties of statistical distributions and estimation methods for shape data and their applications to many scientific fields goes back to the late 1970's with the works of Mardia, Bookstein and Kendall among others. For historical notes on this research areas and an introduction to the applications see Dryden and Mardia 2016, Preface and Section 1 .4. Recall that we consider the set of centred matrices of size $m \times k$ as the space of configurations (or landmarks). We assume that $m \geq 2$, and refer the reader to Le and D. G. Kendall 1993 for more details.

To further remove the effects of scaling, we restrict to non-zero x (i.e. at least two landmarks are different), and divide x by its Frobenius norm (written $\|\cdot\|$). This defines the pre-shape space

$$\mathcal{S}_m^k = \{x \in M(m,k) \mid \sum_{i=1}^{k} x_i = 0, \quad \|x\| = 1\},$$

which is identified with the hypersphere of dimension $m(k-1)-1$. The pre-shape space is therefore a differential manifold whose tangent space at any $x \in \mathcal{S}_m^k$ is given by

$$T_x\mathcal{S}_m^k = \{w \in M(m,k) \mid \sum_{i=1}^{k} w_i = 0, \ \mathrm{tr}(w^\top x) = 0\}.$$

Moreover, the rotation group $SO(m)$ acts on $\mathcal{S}_m^k$ by matrix multiplication, and this action corresponds to applying a rotation to each landmark individually. As $SO(m)$ is compact, this action is proper. However, this action is not free everywhere if $m \geq 3$. This makes the orbit space

$$\Sigma_m^k = \{[x] \mid x \in \mathcal{S}_m^k\} = \mathcal{S}_m^k / SO(m)$$

a "differential manifold with singularities where the action is not free", and these points correspond to matrices of rank $m-2$ or less (i.e. some landmarks are aligned).

By Theorem 2.4.4 (page 60), the canonical projection $\pi : x \mapsto [x]$ is a submersion. For any $x \in \mathcal{S}_m^k$ and $A \in \mathrm{Skew}(m)$, as for all $t \in \mathbb{R}$, $[\exp(tA)x] = x$, the curve $t \mapsto \exp(tA)x$ is a curve in the fiber through x, so the vertical space at x is

$$V_x = \{Ax \mid A \in \mathrm{Skew}(m)\} = \mathrm{Skew}(m)x.$$

The pullback of Frobenius metric on the pre-shape space allows to define the horizontal spaces:

$$\begin{aligned} H_x &= \{w \in T_x\mathcal{S}_m^k \mid Tr(Axw^\top) = 0 \quad \forall A \in \mathrm{Skew}(m)\} \\ &= \{w \in T_x\mathcal{S}_m^k \mid xw^\top \in \mathrm{Sym}(m)\} \end{aligned}$$

where $\mathrm{Sym}(m)$ is the space of symmetric matrices of size m. Lemma 1 from Nava-Yazdani, Hege, et al. 2020 allows to compute the vertical component of any tangent vector.

Lemma 3.3.1. *For any $x \in \mathcal{S}_m^k$ and $w \in T_x\mathcal{S}_m^k$, the vertical component of w can be computed as $V_x(w) = Ax$ where A solves the Sylvester equation:*

$$Axx^\top + xx^\top A = wx^\top - xw^\top \tag{3.17}$$

If $\mathrm{rank}(x) \geq m - 1$, the skew-symmetric solution A of (3.17) is unique.

In practice, the Sylvester equation can be solved by an eigenvalue decomposition of $xx^\top$ (see Example 3.3.2). This defines ver_x, the orthogonal projection on V_x. As $T_x\mathcal{S}_m^k = V_x \oplus H_x$, any tangent vector w at $x \in \mathcal{S}_m^k$ may be decomposed into a horizontal and a vertical component, by solving (3.17) to compute $\mathrm{ver}_x(w)$, and then $\mathrm{hor}_x(w) = w - \mathrm{ver}_x(w)$.

As the Frobenius metric is invariant to the action of $SO(m)$, we can define the quotient metric on Σ_m^k, and this makes π a Riemannian submersion. Furthermore, the Riemannian distances d on $\mathcal{S}_m^k$ and d_Σ on Σ_m^k are related by

$$d_\Sigma(\pi(x), \pi(y)) = \inf_{R \in SO(m)} d(x, Ry).$$

The optimal rotation R between any x, y is unique in a subset U of $\mathcal{S}_m^k \times \mathcal{S}_m^k$, which allows to define the align map $\omega : U \to \mathcal{S}_m^k$ that maps (x, y) to Ry. In this case, $d_\Sigma(\pi(x), \pi(y)) = d(x, \omega(x, y))$ and $x\omega(x, y)^\top \in \mathrm{Sym}(m)$. It is useful to notice that $\omega(x, y)$ can be directly computed by a pseudo-singular value decomposition of $xy^\top$(W. S. Kendall and Le 2009).

Remark 3.3.2. *The alignment problem is similar to the canonical correlation analysis problem (CCA) between two data sets:*

$$\max_{U,V \in O(m)} \langle Ux, Vy \rangle.$$

Finally, in the case of the Kendall shape spaces, the quotient space cannot be seen explicitly as a submanifold of some $\mathbb{R}^N$. Moreover, the projection π and its derivative $d\pi$ cannot be computed. However, the align map amounts to identifying the shape space with a local horizontal section of the pre-shape space, and thanks to the characteristics of Riemannian submersions mentioned in the previous subsections, all the computations can be done in the pre-shape space.

Let Exp, Log, and d denote the operations of the pre-shape space $\mathcal{S}_m^k$, that are given in (2.18),(2.19) (page 40). We obtain from Theorem 3.3.1 for any $x, y \in \mathcal{S}_m^k$ and $v \in T_x\mathcal{S}_m^k$

$$\begin{aligned}
\mathrm{Exp}_{\Sigma,[x]}(d\pi_x v) &= \pi(\mathrm{Exp}_x(\mathrm{hor}_x(v))), \\
\mathrm{Log}_{\Sigma,[x]}([y]) &= d\pi_x \,\mathrm{Log}_x(\omega(x, y)), \\
d_\Sigma([x], [y]) &= d(x, \omega(x, y)).
\end{aligned}$$

To end this section, we state Proposition 2 of Kim, Dryden, et al. 2020, that echoes with Remark 3.3.1 for the computation of parallel transport.

Proposition 3.3.4 (Kim, Dryden, et al. 2020)**.** *Let γ be a horizontal C^1-curve in $\mathcal{S}_m^k$ and v be a horizontal tangent vector at $\gamma(0)$. Assume that* $\operatorname{rank}(\gamma(s)) \geq m-1$ *except for finitely many s. Then the vector field $s \mapsto v(s)$ along γ is horizontal and the projection of $v(s)$ to $T_{[\gamma(s)]}\Sigma_m^k$ is the parallel transport of $d\pi_x v$ along $[\gamma(s)]$ if and only if $s \mapsto v(s)$ is the solution of*

$$\dot{v}(s) = -\operatorname{tr}(\dot{\gamma}(s)v(s)^\top)\gamma(s) + A(s)\gamma(s), \qquad v(0) = v \tag{3.18}$$

where for every s, $A(s) \in \operatorname{Skew}(m)$ is the unique solution to

$$A(s)\gamma(s)\gamma(s)^\top + \gamma(s)\gamma(s)^\top A(s) = \dot{\gamma}(s)v(s)^\top - v(s)\dot{\gamma}(s)^\top. \tag{3.19}$$

Equation (3.18) means that the covariant derivative of $s \mapsto v(s)$ along γ must be a vertical vector at all times, defined by the matrix $A(s) \in \operatorname{Skew}(m)$. These equations can be used to compute parallel transport in the shape space. To compute the parallel transport of $d\pi_x w$ along $[\gamma]$, Kim, Dryden, et al. propose the following method: one first chooses a discretization time-step $\delta = \frac{1}{n}$, then repeat for every $s = \frac{i}{n}, i = 0 \dots n$

1. Compute $\gamma(s)$ and $\dot{\gamma}(s)$,
2. Solve the Sylvester equation (3.19) to compute $A(s)$ and the r.h.s. of (3.18),
3. Take a discrete Euler step to obtain $\tilde{v}(s+\delta)$
4. Project $\tilde{v}(s+\delta)$ to $T_{\gamma(s)}\mathcal{S}_m^k$ to obtain $\hat{v}(s+\delta)$,
5. Project to the horizontal subspace: $v(s+\delta) \leftarrow \operatorname{hor}(\hat{v}(s+\delta))$
6. $s \leftarrow s+\delta$

We notice that this method can be accelerated by a higher-order integration scheme, such as Runge-Kutta (RK) by directly integrating the system $\dot{v} = f(v,s)$ where f is a smooth map given by (3.18) and (3.19). In this case, steps 4. and 5. are not necessary. The precision and complexity of this method is then bound to that of the integration scheme used, as shown in the plot below (Figure 3.4) for randomly generated orthogonal tangent vectors of unit norm.

3.3.4 Implementation

We now present an abstract class to construct a quotient metric. There are two scenarios. In the first one, a submersion $\pi : E \to M$ is given between the total space and the base manifold, together with a metric on E whose restriction to horizontal spaces is preserved by π. This is the case of the Bures-Wasserstein metric (Examples 3.3.2, 3.3.3 and 3.3.5).

In the second scenario, we are only given

1. a total space E,
2. a group G acting freely, properly and smoothly on E,
3. a metric on E invariant to the action of G.

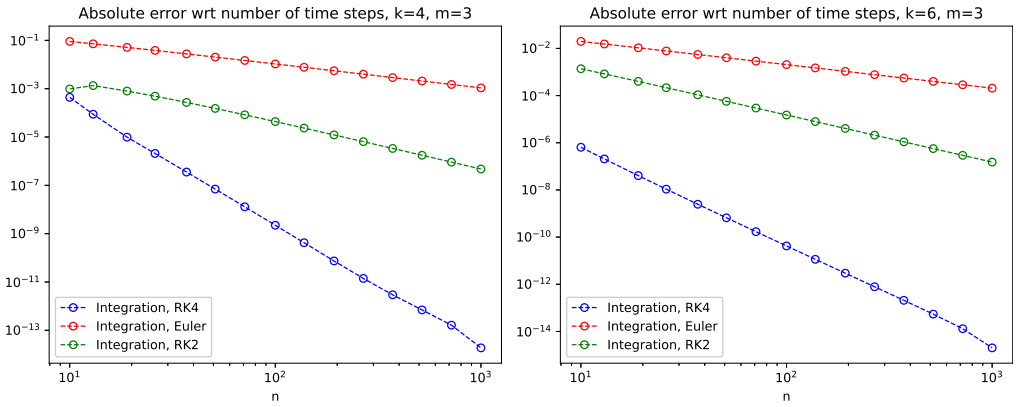


Figure 3.4: Convergence speed of the integration of (3.18) for two orthonormal initial vectors v and $\dot{\gamma}(0)$, with respect to the number of steps. The curve γ is a horizontal geodesic here.

In this case, the base manifold cannot be represented explicitly, that is, as an embedded submanifold of $\mathbb{R}^N$. Thus, we cannot implement it as an `OpenSet` or `LevelSet` class on its own. However, we can use the properties of the canonical projection, that is a Riemannian submersion, to construct a metric on the quotient space. This metric is in fact defined on horizontal spaces of the total space, and the base manifold is locally identified with a section of the total space. This is the case of Kendall shape spaces (Section 3.3.3).

To model both cases, we use the structure of fiber bundle. The first scenario is a fiber bundle by definition. For the second scenario, we notice that $\pi : E \to M$ and $(G, \cdot)$ form a principal fiber bundle. Conversely, if we endow a principal fiber bundle with a G-invariant metric, we can define a quotient metric on B. We thus choose to implement an abstract `FiberBundle` class, as main ingredient of a `QuotientMetric` class.

As the goal is to compute the Riemannian Exp, Log and distance maps according to (3.12), (3.16) and (3.15), we need to specify a section and the corresponding horizontal lift, and the align map. Then the quotient metric is implemented as follows

```
class QuotientMetric(RiemannianMetric):
    """Quotient metric."""

    def __init__(self, fiber_bundle: FiberBundle, dim: int = None):
        if dim is None:
            if fiber_bundle.base is not None:
                dim = fiber_bundle.base.dim
            elif fiber_bundle.group is not None:
                dim = fiber_bundle.dim - fiber_bundle.group.dim
            else:
                raise ValueError('Either the base manifold, '
                                 'its dimension, or the group acting on the '
```

```
                                'total space must be provided.')
        super(QuotientMetric, self).__init__(
            dim=dim, default_point_type=fiber_bundle.default_point_type)

        self.fiber_bundle = fiber_bundle
        self.group = fiber_bundle.group
        self.ambient_metric = fiber_bundle.ambient_metric

    def inner_product(
            self, tangent_vec_a, tangent_vec_b, base_point=None,
            point_above=None):
        """Compute the inner-product of two tangent vectors at a base point."""
        if point_above is None:
            if base_point is not None:
                point_above = self.fiber_bundle.lift(base_point)
            else:
                raise ValueError(
                    'Either a point (of the total space) or a base point (of '
                    'the quotient manifold) must be given.')
        hor_a = self.fiber_bundle.horizontal_lift(tangent_vec_a, point_above)
        hor_b = self.fiber_bundle.horizontal_lift(tangent_vec_b, point_above)
        return self.ambient_metric.inner_product(hor_a, hor_b, point_above)

    def exp(self, tangent_vec, base_point, **kwargs):
        """Compute the Riemannian exponential of a tangent vector."""
        lift = self.fiber_bundle.lift(base_point)
        horizontal_vec = self.fiber_bundle.horizontal_lift(tangent_vec, lift)
        return self.fiber_bundle.riemannian_submersion(
            self.ambient_metric.exp(horizontal_vec, lift))

    def log(self, point, base_point, **kwargs):
        """Compute the Riemannian logarithm of a point."""
        point_fiber = self.fiber_bundle.lift(point)
        bp_fiber = self.fiber_bundle.lift(base_point)
        aligned = self.fiber_bundle.align(point_fiber, bp_fiber, **kwargs)
        return self.fiber_bundle.tangent_riemannian_submersion(
            self.ambient_metric.log(aligned, bp_fiber), bp_fiber)

    def squared_dist(self, point_a, point_b, **kwargs):
        """Squared geodesic distance between two points."""
        lift_a = self.fiber_bundle.lift(point_a)
        lift_b = self.fiber_bundle.lift(point_b)
        aligned = self.fiber_bundle.align(lift_a, lift_b, **kwargs)
        return self.ambient_metric.squared_dist(aligned, lift_b)
```

Example 3.3.6: Bures-Wasserstein metric

In the setting of Examples 3.3.2, 3.3.3 and 3.3.5, that corresponds to the first scenario as $E = GL(n)$ and $M = SPD(n)$, we create a new class that corresponds to $GL(n)$ with the fiber bundle structure:

```
class BuresWasserteinBundle(GeneralLinear, FiberBundle):
    def __init__(self, n):
        super(BuresWasserteinBundle, self).__init__(
            n=n, base=SPDMatrices(n), group=SpecialOrthogonal(n),
            ambient_metric=MatricesMetric(n, n))

    @staticmethod
    def riemannian_submersion(point):
        return Matrices.mul(point, Matrices.transpose(point))

    @staticmethod
    def lift(point):
        return gs.linalg.cholesky(point)

    def tangent_riemannian_submersion(self, tangent_vec, base_point):
        product = Matrices.mul(
            base_point, Matrices.transpose(tangent_vec))
        return 2 * Matrices.to_symmetric(product)

    def horizontal_lift(self, tangent_vec, point_above=None, base_point=None):
        if base_point is None and point_above is not None:
            if point_above is not None:
                base_point = self.riemannian_submersion(point_above)
            else: raise ValueError(
                'Either a point in the fiber or a base point in base manifold)'
                ' must be given.')
        sylvester = gs.linalg.solve_sylvester(
            base_point, base_point, tangent_vec)
        return Matrices.mul(sylvester, point_above)
```

Note that all computations can actually be carried out in closed form (see Bhatia, Jain, et al. 2019), and this specific examples allows to test our general implementation. In this case we used the minimization procedure to compute the align map.

In the second scenario, we only compute in E so both submersion and lift maps are the identity of E. This can be set by default and overridden as in Example 3.3.6. Furthermore, in that case, tangent vectors to M are identified with horizontal vectors of E, so we need to compute the horizontal decomposition of a tangent vector to E. Thus, either the horizontal or vertical projection needs to be implemented, and the other one can be deduced.

In the first scenario, these are given by using $d\pi$ and the horizontal lift, as vertical vectors are in the kernel of $d\pi$ by definition.

This explains the following `FiberBundle` class (the align method is not duplicated from above in the interest of space):

```
class FiberBundle(Manifold, abc.ABC):
    """Class for (principal) fiber bundles."""
```

```
    def __init__(
            self, dim: int, base: Manifold = None,
            group: LieGroup = None, ambient_metric: RiemannianMetric = None,
            group_action=None, **kwargs):

        super(FiberBundle, self).__init__(dim=dim, **kwargs)
        self.base = base
        self.group = group
        self.ambient_metric = ambient_metric

        if group_action is None and group is not None:
            group_action = group.compose
        self.group_action = group_action

    @staticmethod
    def riemannian_submersion(point):
        """Project a point to base manifold."""
        return point

    @staticmethod
    def lift(point):
        """Lift a point to total space."""
        return point

    def align(self, point, base_point,
              max_iter=25, verbose=False, tol=gs.atol):
        """Align point to base_point by optimization in the Lie algebra."""
        pass

    def tangent_riemannian_submersion(self, tangent_vec, base_point):
        """Project a tangent vector to base manifold."""
        return self.horizontal_projection(tangent_vec, base_point)

    def horizontal_lift(self, tangent_vec, point_above=None, base_point=None):
        """Lift a tangent vector to a horizontal vector in the total space."""
        if point_above is None:
            if base_point is not None:
                point_above = self.lift(base_point)
            else:
                raise ValueError(
                    'Either a point (of the total space) or a base point (of'
                    'the base manifold) must be given.')
        return self.horizontal_projection(tangent_vec, point_above)

    def horizontal_projection(self, tangent_vec, base_point):
        """Project to horizontal subspace."""
        try:
            return tangent_vec - self.vertical_projection(
                tangent_vec, base_point)
        except (RecursionError, NotImplementedError):
            return self.horizontal_lift(
                self.tangent_riemannian_submersion(
                    tangent_vec, base_point), base_point)
```

```
    def vertical_projection(self, tangent_vec, base_point, **kwargs):
        """Project to vertical subspace."""
        try:
            return tangent_vec - self.horizontal_projection(
                tangent_vec, base_point)
        except RecursionError:
            raise NotImplementedError
```

Example 3.3.7: Kendall shape metric

With the above construction of the second scenario, Kendall shape metric is just a subclass of the `QuotientMetric` class. We also add the parallel transport method discussed above and used in Figure 3.4.

```
class KendallShapeMetric(QuotientMetric):
    """Quotient metric on the shape space."""

    def __init__(self, k_landmarks, m_ambient):
        bundle = PreShapeSpace(k_landmarks, m_ambient)
        super(KendallShapeMetric, self).__init__(
            fiber_bundle=bundle,
            dim=bundle.dim - int(m_ambient * (m_ambient - 1) / 2))

    def parallel_transport(
            self, tangent_vec_a, tangent_vec_b, base_point, n_steps=100,
            step='rk4'):
        """Compute the parallel transport of a tangent vec along a geodesic."""
        horizontal_a = self.fiber_bundle.horizontal_projection(
            tangent_vec_a, base_point)
        horizontal_b = self.fiber_bundle.horizontal_projection(
            tangent_vec_b, base_point)

        def force(state, time):
            gamma_t = self.ambient_metric.exp(time * horizontal_b, base_point)
            speed = self.ambient_metric.parallel_transport(
                horizontal_b, time * horizontal_b, base_point)
            coef = self.inner_product(speed, state, gamma_t)
            normal = gs.einsum('...,...ij->...ij', coef, gamma_t)

            align = gs.matmul(Matrices.transpose(speed), state)
            right = align - Matrices.transpose(align)
            left = gs.matmul(Matrices.transpose(gamma_t), gamma_t)
            skew_ = gs.linalg.solve_sylvester(left, left, right)
            vertical_ = - gs.matmul(gamma_t, skew_)
            return vertical_ - normal

        flow = integrate(force, horizontal_a, n_steps=n_steps, step=step)
        return flow[-1]
```

Example 3.3.8: Full rank correlation matrices

As a last example, we implement the metric on the set of Correlation matrices described in Thanwerdas and Pennec 2021a using our abstract `FiberBundle` and `QuotientMetric` classes.

The set of full-rank correlation matrices $Corr(n)$ is a submanifold of $SPD(n)$ formed by matrices with unit diagonal. Define the action of positive diagonal matrices on $SPD(n)$ by congruence:

$$\triangleright : (D, \Sigma) \in Diag_+(n) \times SPD(n) \mapsto D\Sigma D \in SPD(n).$$

This action is smooth, free, and proper (David and Gu 2019). The quotient manifold $SPD(n)/Diag_+(n)$ is thus well defined, and can be identified with $Corr(n)$ by the map $[\Sigma] \mapsto D_\Sigma^{-1/2} \triangleright \Sigma$ where D_Σ is the diagonal matrix with coefficients Σ_{ii}.

Let $\pi : SPD(n) \to Corr(n)$ be the canonical projection composed with this map, i.e. $\Sigma \mapsto D_\Sigma^{-1/2} \triangleright \Sigma$, and consider the affine-invariant metric defined on $SPD(n)$ by

$$g_\Sigma(V, W) = tr(\Sigma^{-1} V \Sigma^{-1} W).$$

This metric will be detailed in Example 3.5.4 in Section 3.5. It is clear that it is invariant by the action of diagonal matrices.

Therefore we can define the quotient metric such that π is a Riemannian submersion. The horizontal and vertical spaces are given in Thanwerdas and Pennec 2021a, Theorem 1. The submersion and its differential, the vertical projection and horizontal lifts are computed in closed form, so we are in the first scenario, and the quotient metric is simply

```
class FullRankCorrelationAffineQuotientMetric(QuotientMetric):
    """Class for the quotient of the affine-invariant metric."""

    def __init__(self, n):
        super(FullRankCorrelationAffineQuotientMetric, self).__init__(
            fiber_bundle=CorrelationMatricesBundle(n=n))
```

where the `FiberBundle` class is used to define `CorrelationMatricesBundle` as follows:

```python
class CorrelationMatricesBundle(SPDMatrices, FiberBundle):
    """Fiber bundle for the quotient metric on correlation matrices."""

    def __init__(self, n):
        super(CorrelationMatricesBundle, self).__init__(
            n=n, base=CorrelationMatrices(n),
            ambient_metric=SPDMetricAffine(n), group_dim=n,
            group_action=CorrelationMatrices.diag_action)

    @staticmethod
    def riemannian_submersion(point):
        """Compute the correlation matrix associated to an SPD matrix."""
        diagonal = Matrices.diagonal(point) ** (-.5)
        aux = gs.einsum('...i,...j->...ij', diagonal, diagonal)
        return point * aux

    def tangent_riemannian_submersion(self, tangent_vec, base_point):
        """Compute the differential of the Riemannian submersion."""
        diagonal_bp = Matrices.diagonal(base_point)
        diagonal_tv = Matrices.diagonal(tangent_vec)

        diagonal = diagonal_tv / diagonal_bp
        aux = base_point * (diagonal[..., None, :] + diagonal[..., :, None])
        mat = tangent_vec - .5 * aux
        return CorrelationMatrices.diag_action(diagonal_bp ** (-.5), mat)

    def vertical_projection(self, tangent_vec, base_point, **kwargs):
        """Compute the vertical projection wrt the affine-invariant metric."""
        n = self.n
        inverse_base_point = GeneralLinear.inverse(base_point)
        operator = gs.eye(n) + base_point * inverse_base_point
        inverse_operator = GeneralLinear.inverse(operator)
        vector = gs.einsum(
            '...ij,...ji->...i', inverse_base_point, tangent_vec)
        diagonal = gs.einsum('...ij,...j->...i', inverse_operator, vector)
        return base_point * (diagonal[..., None, :] + diagonal[..., :, None])

    def horizontal_lift(self, tangent_vec, base_point=None, fiber_point=None):
        """Compute the horizontal lift wrt the affine-invariant metric."""
        if fiber_point is None and base_point is not None:
            return self.horizontal_projection(tangent_vec, base_point)
        diagonal_point = Matrices.diagonal(fiber_point) ** 0.5
        lift = CorrelationMatrices.diag_action(diagonal_point, tangent_vec)
        hor_lift = self.horizontal_projection(lift, base_point=fiber_point)
        return hor_lift
```

3.4 Homogeneous spaces

We now come back to the particular case of homogeneous spaces, that are quotient spaces where the total space is a Lie group $E = G$ under the action of a subgroup

$H \subset G$ of G, and $M = G/H$. We focus on Riemannian metrics that are invariant to the group action, that is, for which the action of any $g \in G$ on M is an isometry. This case is fundamental as it is the simplest way in which all the geometry of M is determined by that of G. If additionnally the metric on G is bi-invariant, we say the metric on G/H is normal, and all the computations can be carried in closed-form.

3.4.1 Characterization

Recall that a homogeneous space $M \simeq G/H$ is a manifold with a smooth transitive action of G on M, and corresponds to the orbits of the right-action of a closed subgroup H of G. As the map $\pi : G \mapsto G/H$ is a submersion (see Theorem 2.4.4 page 60), we shall see that invariant metrics on G/H are in fact particular cases of the quotient metrics of the previous section. Indeed, a particular case of Theorem 3.3.2 (page 93) gives the following proposition.

Proposition 3.4.1. *Let G be a Lie group and $H \subseteq G$ a closed subgroup of G. Write $o = eH = H \in G/H$, and $\mathfrak{g}, \mathfrak{h}$ the Lie algebras of respectively G, H. Let g be a left-invariant Riemannian metric on G, that is also right-invariant by H. Then there exists a unique Riemannian metric on G/H that is invariant to the action of G and such that $d\pi_e$ is an isometry between $\mathfrak{h}^\perp \subset \mathfrak{g}$ and T_oG/H. In fact, π is a Riemannian submersion.*

Indeed, Theorem 3.3.2 applies with $E = G$ and $G = H$ as the action of H on G is smooth, free and proper (by Theorem 2.4.5), and the metric on G is invariant to this action. This defines a metric on G/H, and we can show that it is invariant to G. Indeed, for any $x, y \in G$, write $M_{yx^{-1}} : G/H \to G/H$ that maps some $[p]$ to $[yx^{-1}p]$, and $L_{yx^{-1}} : G \to G$ the left translation of G by yx^{-1}. Then by definition of the action of G on M, we have the following commutation

$$\pi \circ L_{yx^{-1}} = M_{yx^{-1}} \circ \pi$$

Differentiating the above expression at x and as $d\pi_x, d\pi_y$ and $d(L_{yx^{-1}})_x$ are isometries, we conclude that $d(M_{yx^{-1}})_{[x]}$ is also an isometry.

3.4.2 Existence

The above proposition thus states that we need a left-invariant metric on G that is also right-invariant by H to define an invariant metric on G/H. There are necessary and sufficient conditions on G and H for these to exist (see e.g. Gallier and Quaintance 2020, Proposition 22 .22).

Proposition 3.4.2. *If G acts faithfully on G/H and if $\mathfrak{g}$ admits a decomposition $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$ with $\mathrm{AD}_H(\mathfrak{m}) \subset \mathfrak{m}$, then G-invariant metrics on G/H are in one-to-one correspondence with AD_H-invariant scalar products on $\mathfrak{m}$. These exist if and only if the closure of the group $\mathrm{AD}_H(\mathfrak{m})$ is compact. Conversly, if G/H admits a G-invariant metric, then G admits a left-invariant metric which is right invariant under H, and the restriction of this metric to H is bi-invariant. The decomposition of $\mathfrak{g}$ is given by $\mathfrak{m} = \mathfrak{h}^\perp$ in this case.*

Note that if H is connected, the condition $\mathrm{AD}_H(\mathfrak{m}) \subset \mathfrak{m}$ is equivalent to $[h, \mathfrak{m}] \subset \mathfrak{m}$. If H is compact, then $\mathrm{AD}_H(\mathfrak{m})$ is compact, so a G-invariant metric exists. The above proposition suggests the following definitions due to Nomizu 1954.

Definition 3.4.1 (Reductive Homogeneous space)**.** Let G be a Lie group and H a closed subgroup of G. Write $\mathfrak{g}, \mathfrak{h}$ their Lie algebra. We say that the homogeneous space G/H is *reductive* if there exists some subspace $\mathfrak{m} \subset \mathfrak{g}$ such that

$$\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m} \qquad \text{and} \qquad \mathrm{AD}_h(\mathfrak{m}) \subseteq \mathfrak{m} \quad \forall h \in H.$$

In this case, $\mathfrak{m}$ is isomorphic to T_oG/H via $d\pi_e$. In fact, $\mathfrak{h}$ is the vertical subspace at e, and $\mathfrak{m}$ is the horizontal space. The notion of reductive homogeneous space is important because it is a sufficient condition for $M = G/H$ to admit a G-invariant connection, whose geodesics are projections (by π) of the one-parameter subgroups of G. This is not sufficient however to obtain such property from the Levi-Civita connection of some metric, and an additional condition will be given in the next subsection.

Example 3.4.1: Stiefel space

Recall from Example 2.4.11 (page 61) that with $G = SO(n)$ and

$$H = \left\{ \begin{pmatrix} I_k & 0 \\ 0 & R \end{pmatrix} \mid R \in SO(n-k) \right\} \simeq SO(n-k),$$

we obtain the Stiefel manifold $S(n,k) = G/H$, the set of orthonormal k-frames represented by $n \times k$ orthogonal matrices. The equivalence class of some $Q = (U, U_\perp) \in S(k,n)$ is $[Q] = QH = (U, U_\perp R)$ where R is any matrix in $SO(n-k)$. The canonical projection is therefore given by $\pi : Q = (U, U_\perp) \mapsto U$ and the origin $o = eH$ of G/H is $\pi(I_n) = I_k$. Thus we can write π as the projection on the first k columns. Moreover, $\mathfrak{g}$ is the set of skew-symmetric matrices, so that we have $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$ with

$$\mathfrak{h} = \left\{ \begin{pmatrix} 0 & 0 \\ 0 & S \end{pmatrix}, \ S \in \mathfrak{so}(n-k) \right\}$$
$$\mathfrak{m} = \left\{ \begin{pmatrix} T & -A^\top \\ A & 0 \end{pmatrix}, \ T \in \mathfrak{so}(k) A \in M_{n-k,k}(\mathbb{R}) \right\}$$

It is straightforward to check that $[\mathfrak{h}, \mathfrak{m}] \subseteq \mathfrak{m}$, and as H is connected, $S(k,n)$ is a reductive homogeneous space.

As H is compact, by Proposition 3.4.2 there exists a G-invariant metric on $S(k,n)$. Indeed, as the pullback of the Frobenius metric on G is bi-invariant on G, hence on H, Proposition 3.4.1 applies and the metric on $S(k,n)$ is defined such that $\pi : Q = (U, U_\perp) \mapsto U$ is a Riemannian submersion. For

$X = \begin{pmatrix} T & -A^\top \\ A & 0 \end{pmatrix} \in \mathfrak{m}$, $d\pi_o X = \begin{pmatrix} T \\ A \end{pmatrix}$ and we obtain

$$\left\langle \begin{pmatrix} T \\ A \end{pmatrix}, \begin{pmatrix} S \\ B \end{pmatrix} \right\rangle \triangleq \frac{1}{2} \operatorname{tr}\left(\begin{pmatrix} T & -A^\top \\ A & 0 \end{pmatrix}^\top \begin{pmatrix} S & -B^\top \\ B & 0 \end{pmatrix} \right) = \frac{1}{2} \operatorname{tr}(T^\top S) + \operatorname{tr}(A^\top B) \tag{3.20}$$

A more convenient way of representing a tangent vector at U is by $X = US + (I - UU^\top)A$. Then one can show that the $SO(n)$-invariant metric defined in (3.20) can be written

$$\langle X_1, X_2 \rangle_U = \operatorname{tr}(X_1^\top (I_k - \frac{1}{2} UU^\top) X_2).$$

For more on this metric, see Gallier and Quaintance 2020, Section 22.5.

3.4.3 Properties

By using the formulas of the connection of an invariant metric on a Lie group (see Section 3.2) and of a Riemannian submersion (Section 3.3), one can show the following (see e.g. Gallier and Quaintance 2020, Propositions 22.25 and 22.27).

Theorem 3.4.1. *Let G be a Lie group and H a closed subgroup of G. If there exists an AD_H-invariant inner product on $\mathfrak{m}$, then the Levi-Civita connection of the induced metric on G/H is given by $\nabla_X Y = -\frac{1}{2}[X, Y]_\mathfrak{m}$ and the geodesics are projections of one-parameter subgroups if and only if*

$$\langle x, [z, y]_\mathfrak{m} \rangle + \langle [z, x]_\mathfrak{m}, y \rangle = 0 \quad \forall x, y, z \in \mathfrak{m}. \tag{3.21}$$

In this case, we say that G/H is naturally reductive

This property allows to derive closed-form expressions for the geodesics in many cases, as one parameter subgroups are given by the matrix exponential in classical Lie groups. A similar behavior holds for curvature and parallel transport. Many formulas implemented in geomstats can be retrieved this way.

Example 3.4.2: Stiefel Exponential map

Following Example 3.4.1 and applying theorem 3.4.1, we can verify that the metric verifies (3.21). Therefore, the exponential map is given by

$$\operatorname{Exp}_U \left(\begin{pmatrix} S \\ A \end{pmatrix} \right) = (U, U_\perp) \exp \left(\begin{pmatrix} S & -A^\top \\ A & 0 \end{pmatrix} \right) \begin{pmatrix} I_k \\ 0 \end{pmatrix}$$

This expression can be simplified with a QR decomposition of A, and its inverse (the Logarithm) can be computed recursively see Zimmermann 2017, for more details.

3.5 Symmetric spaces

To conclude this chapter, we briefly expose symmetric spaces, as they can be defined from a homogeneous space G/H with an additional tool defined on G. This results in one of the most simple geometries, where the geodesics, parallel transport and curvature can be computed in closed form. We start with a more intrinsic definition and will connect with this description after. Similarly to that of homogeneous space, the structure of symmetric space does not necessarily require a Riemannian metric but only an affine connection. We first focus on the Riemannian case for simplicity, and give a few remarks on the more general case, referring to it as the affine symmetric case. This notion was introduced by Cartan 1926 who fully achieved a classification of symmetric spaces. The most complete reference is Helgason 1979, and a good exposition of the non Riemannian case is given in Kobayashi and Nomizu 1996b, Chapter XI and Postnikov 2001, Chapters 4–10.

We first define the geodesic symmetries, the maps defined locally that revert geodesics.

Definition 3.5.1 (Geodesic symmetry)**.** Let (M, g) be a Riemannian manifold. Let $x \in M$ and $U \subset M$ be an open neighborhood of x such that the exponential map is injective on U. The *geodesic symmetry* at x is the map defined by

$$s_x : \left\{ \begin{array}{ccc} U & \longrightarrow & M \\ y & \longmapsto & \mathrm{Exp}_x(-\mathrm{Log}_x(y)) \end{array} \right. .$$

It is clear that for any $x \in M$, x is an isolated fixed point of s_x, and that $(ds_x)_x = -\mathrm{id}$, where id is the identity transformation of T_xM.

Definition 3.5.2 (Locally symmetric space)**.** Let (M, g) be a Riemannian manifold. M is called *locally symmetric* if for any $x \in M$, the geodesic symmetry s_x is an isometry.

Remark 3.5.1. *This definition is valid in an affine space where the notion of isometry is replaced by the notion of affine map, i.e., that preserve the connection.*

The property that defines locally symmetric spaces has direct consequences on its curvature tensor (Kobayashi and Nomizu 1996b, Chapter XI, Theorem 6.2).

Theorem 3.5.1. *A Riemannian manifold M is locally symmetric if and only if $\nabla R = 0$.*

We shall say that the curvature of M is *covariantly* constant, and this will simplify the parallel transport on locally symmetric spaces.

Definition 3.5.3 (Symmetric Space)**.** (M, g) is called (globally) symmetric if the geodesic symmetries are defined on the whole manifold M and are isometries.

The two notions are equivalent up to topological constrains, as stated in the following theorem (Kobayashi and Nomizu 1996b, Chapter XI, Theorems 6.3-6.4).

Theorem 3.5.2. *A geodesically complete, simply connected, locally symmetric space is globally symmetric. Conversely, every globally symmetric space is geodesically complete.*

We now come to the interesting theorem that relates symmetric spaces to homogeneous spaces (Kobayashi and Nomizu 1996b, Chapter XI, Theorem 6.5).

Theorem 3.5.3. *The group of isometries of M $\mathrm{Isom}(M)$ is a Lie group that acts transitively on M. M is thus a homogeneous space. Let G be the largest connected group of isometries of M, and consider a reference point $o \in M$. Let H its stabilizer by the action of G. Then H is compact and $M \simeq G/H$.*

Define G as in the above theorem. We can define an additional structure on G that will be the essence that differentiate symmetric spaces from homogeneous spaces.

Remark 3.5.2.

- *The composition of two geodesic symmetries belongs to G, and is called a transvection.*
- *Let γ be a geodesic through o, then transvections $s_{\gamma(t)} \circ s_o$ are one-parameter subgroups of G.*

Theorem 3.5.4. *Let $M = G/H$ be a symmetric space, define on G the map $\sigma : g \mapsto s_o \circ g \circ s_o$.*

- *σ is involutive: $\sigma \circ \sigma = Id$,*
- *σ is a group homeomorphism, i.e. $\sigma(g \circ h) = \sigma(g) \circ \sigma(h)$, thus an automorphism,*
- *The set G^σ of fixed point of σ is a closed subgroup, and with G_0^σ its connected component, we have $G_0^\sigma \subset H \subset G^\sigma$ (implying $\dim(H) = \dim(G^\sigma)$).*

Define $\mathfrak{h} = \{v \in \mathfrak{g}, d\sigma_e(v) = v\}$ and $\mathfrak{m} = \{v \in \mathfrak{g}, d\sigma_e(v) = -v\}$. Then G/H is naturally reductive with $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$, i.e. $[\mathfrak{h}, \mathfrak{h}] \subset \mathfrak{h}$ and $[\mathfrak{h}, \mathfrak{m}] \subset \mathfrak{m}$, and we have the additional property:

$$[\mathfrak{m}, \mathfrak{m}] \subset \mathfrak{h}.$$

As a consequence (Theorem 3.4.1), the geodesics of a symmetric space are thus projections of one parameter subgroups. In fact symmetric spaces are the only homogeneous spaces with an involutive automorphism as described in Theorem 3.5.4.

Definition 3.5.4 (Symmetric pair)**.** A *symmetric pair* is a triplet (G, H, σ) where G is a connected Lie group, H a closed subgroup of G and σ is an involutive automorphism of G, such that its set of fixed points G^σ satisfies $G_0^\sigma \subset H \subset G^\sigma$.

Remark 3.5.3.

- *If there exists $a \in G$ s.t. $a^2 = e$, then $\sigma : g \mapsto a \circ g \circ a^{-1}$ is an involutive automorphism, and its set of fixed points G^σ is a closed (normal) subgroup of G, so that (G, G^σ, σ) forms a symmetric pair.*

- *The inversion $g \mapsto g^{-1}$ is an automorphism if and only if G is commutative. This is very restrictive, and in general the involution is not the inversion map.*
- *In fact one can show (see e.g. Cheeger and Ebin 1975, Proposition 3.37) that a simply connected group G possesses an involutive automorphism σ if and only if its Lie algebra admits a decomposition $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$ with $[\mathfrak{h}, \mathfrak{h}] \subset \mathfrak{h}$, $[\mathfrak{h}, \mathfrak{m}] \subset \mathfrak{m}$ and $[\mathfrak{m}, \mathfrak{m}] \subset \mathfrak{h}$.*

We now see how to recover a symmetric space from a symmetric pair. Recall that $\pi : g \in G \to gH \in G/H$ is the canonical projection of the quotient G/H, and that G acts on G/H by $g_1 \triangleright (g_2 H) = (g_1 g_2)H$. Let $o = eH = H$, and $s_o : gH \mapsto \sigma(g)H$, i.e. $s_o \circ \pi = \pi \circ \sigma$.

Theorem 3.5.5. *Let (G, H, σ) be a symmetric pair such that H and G_0^σ are compact. Define $\mathfrak{h} = \{v \in \mathfrak{g}, d\sigma_e(v) = v\}$ and $\mathfrak{m} = \{v \in \mathfrak{g}, d\sigma_e(v) = -v\}$. Then $M = G/H$ is naturally reductive with the decomposition $g = \mathfrak{h} \oplus \mathfrak{m}$. Furthermore, with the family of symmetries defined at any $x = gH \in M$ by*

$$s_x = g \circ s_o \circ g^{-1},$$

M is a globally symmetric space.

Remark 3.5.4. *The assumption that H and G_0^σ are compact is sufficient to ensure the existence of a G-invariant metric on G/H such that it is naturally reductive. It is not necessary however to show that G/H is affine symmetric, but the connection may not be the Levi-Civita connection of any metric. If a G-invariant metric exists however on an affine symmetric space $M = G/H$, its Levi-Civita connection coincides with the connection of the affine symmetric structure (Kobayashi and Nomizu 1996b, Chapter XI, Theorem 3.3).*

The last theorem along with Remark 3.5.3 now make it easier to exemplify the notion of symmetric space.

Example 3.5.1: Hypersphere

Recall from Example 2.4.10 (page 61) that the hypersphere S^d can be seen as the quotient of $G = SO(d+1)$ by $H \simeq SO(d)$ defined by

$$H = \left\{ \begin{pmatrix} 1 & 0 \\ 0 & R \end{pmatrix} \mid R \in SO(d) \right\}.$$

Define $J = \begin{pmatrix} -1 & 0 \\ 0 & I_d \end{pmatrix}$. Obviously $J^2 = I_{d+1}$ so that the map defined by $\sigma : P \in G \mapsto JPJ \in G$ is an involutive automorphism, and it is straightforward to check that $G^\sigma = H$. Then (G, H, σ) is a symmetric pair and by Theorem 3.5.5, S^d is a symmetric space. The Lie algebra is decomposed in

$\mathfrak{so}(d+1) = \mathfrak{h} \oplus \mathfrak{m}$ with

$$\mathfrak{h} = \left\{ \begin{pmatrix} 0 & 0 \\ 0 & S \end{pmatrix}, \ S \in \mathfrak{so}(d) \right\}$$
$$\mathfrak{m} = \left\{ \begin{pmatrix} 0 & -u^\top \\ u & 0 \end{pmatrix}, \ u \in \mathbb{R}^d \right\}.$$

From the expressions of the Exp and Log map, we can compute the symmetry at $x \in S^d$. For any $y \in S^d$

$$s_x(y) = 2(\langle x, y \rangle)x - y.$$

Similarly, the upper hyperboloid and the Euclidean space R^d are also symmetric. Thus, all constant-curvature spaces are symmetric.

Example 3.5.2: Grassmann manifold

The Grassmann manifold $Gr(k,n)$ (Example 2.4.12 page 62) is the set of k-dimensional subspaces of $\mathbb{R}^n$, and is identified with $O(n)/(O(k)\times O(n-k))$. Define $J = \begin{pmatrix} I_k & 0 \\ 0 & -I_{n-k} \end{pmatrix}$. Obviously $J^2 = I_n$ so that the map defined by $\sigma : P \in G \mapsto JPJ \in G$ is an involutive automorphism with fixed points

$$G^\sigma = \left\{ \begin{pmatrix} Q & 0 \\ 0 & R \end{pmatrix} \mid Q \in O(k), R \in O(n-k), \ \det(Q)\det(R) = 1 \right\},$$

and $G^\sigma = S(O(k) \times O(n-k))$ with $G_0^\sigma = SO(k) \times SO(n-k)$. Note that if we use $H = G_0^\sigma$, we restrict to *oriented* k-subspaces, and consider in this case the oriented Grassmann manifold. In both cases, we have

$$\mathfrak{h} = \left\{ \begin{pmatrix} S & 0 \\ 0 & T \end{pmatrix}, \ S \in \mathfrak{so}(k), T \in \mathfrak{so}(n-k) \right\}$$
$$\mathfrak{m} = \left\{ \begin{pmatrix} 0 & -A^\top \\ A & 0 \end{pmatrix}, \ A \in M_{n-k,k}(\mathbb{R}) \right\}.$$

Then (G, H, σ) is a symmetric pair and by Theorem 3.5.5, $Gr(k,n)$ is a symmetric space. Given any $P = QP_kQ^\top \in Gr(k,n)$, the symmetry at P is $s_P : \tilde{P} \mapsto (QJQ^\top)\tilde{P}(QJQ^\top)$. The symmetric space structure allows to deduce many properties of $Gr(k,n)$, namely that it is geodesically complete, hence a complete metric space (by the Hopf-Rinow theorem page 41), and its exponential map is surjective at all points.

Example 3.5.3: Stiefel

We now give an example of homogeneous space that is not symmetric: the Stiefel manifold $St(k,n)$. Indeed recall that we have $St(k,n) = G/H$ with $G = SO(n)$ and $H = SO(n-k)$ and reductive decomposition $\mathfrak{so}(n) = \mathfrak{h} \oplus \mathfrak{m}$ with

$$\mathfrak{h} = \left\{ \begin{pmatrix} 0 & 0 \\ 0 & T \end{pmatrix}, \ T \in \mathfrak{so}(n-k) \right\}$$
$$\mathfrak{m} = \left\{ \begin{pmatrix} S & -A^\top \\ A & 0 \end{pmatrix}, \ S \in \mathfrak{so}(k) A \in M_{n-k,k}(\mathbb{R}) \right\}$$

We can check that $[\mathfrak{m}, \mathfrak{m}] \not\subset \mathfrak{h}$, so that $St(k.n)$ is not symmetric with this decomposition.

Example 3.5.4: Affine-Invariant metric on SPD

Recall that $SPD(n)$ is a homogenous space with (restricting to a connected component) $G = GL^+(n)$ and $H = SO(n)$ and canonical projection that coincides with $\pi : A \mapsto AA^\top$. Define $\sigma : A \in G \mapsto A^{-\top}$. It is clear that σ is an involutive automorphism with $G^\sigma = H$. Then (G, H, σ) is a symmetric pair and by Theorem 3.5.5, $SPD(n)$ is a symmetric space. From σ and π we deduce the symmetry for $\Sigma, \Lambda \in SPD(n)$: $s_\Sigma(\Lambda) = \Sigma\Lambda^{-1}\Sigma$.

The affine-invariant (AI) metric is defined as the quotient metric on G/H of the left-invariant metric on G that coincides with the Frobenius metric at I. Its expression is thus at any $\Sigma \in SPD(n)$, for all $V, W \in \mathrm{Sym}(n)$

$$g_\Sigma(V,W) = \mathrm{tr}(\Sigma^{-1}V\Sigma^{-1}W).$$

From the projection of one parameter subgroups we deduce $\forall \Sigma, \Sigma_1, \Sigma_2 \in SPD(n), \forall W \in \mathrm{Sym}(n)$

$$\mathrm{Exp}_\Sigma(W) = \Sigma^{\frac{1}{2}} \exp(\Sigma^{-\frac{1}{2}} W \Sigma^{-\frac{1}{2}}) \Sigma^{\frac{1}{2}},$$
$$\mathrm{Log}_{\Sigma_1}(\Sigma_2) = \Sigma_1^{\frac{1}{2}} \log(\Sigma_1^{-\frac{1}{2}} \Sigma_2 \Sigma_1^{-\frac{1}{2}}) \Sigma_1^{\frac{1}{2}},$$

where when not indexed, exp and log refer to the matrix operators. Finally, let

$$P_t = \Sigma^{\frac{1}{2}} \exp(\frac{t}{2} \Sigma^{-\frac{1}{2}} W \Sigma^{-\frac{1}{2}}) \Sigma^{-\frac{1}{2}}.$$

The parallel transport from Σ along the geodesic with initial velocity $W \in \mathrm{Sym}(n)$ of $V \in \mathrm{Sym}(n)$ a time t is (Yair, Ben-Chen, et al. 2019)

$$\Pi^t_{0,W} V = P_t V P_t^\top.$$

We now focus on the case of Lie groups themselves, that can be seen as symmetric spaces. However, one must be cautious on the structure, either metric or affine that

is used. Let G be a connected Lie group. Consider the product group $\tilde{G} = G \times G$ with the involution $\sigma : (g, h) \mapsto (h, g)$. The subgroup of fixed points is the diagonal of $\tilde{G}$, $H = \{(g, g), g \in G\}$, and its Lie algebra is $\mathfrak{h} = \{(x, x), x \in \mathfrak{g}\} \simeq \mathfrak{g}$. We thus see from Proposition 3.4.2, that for $G = G \times G/G$ to be Riemannian symmetric hence reductive homogeneous, it must admit a bi-invariant metric. There are still three possible reductive decompositions $\mathfrak{m} \oplus \mathfrak{h}$ of the Lie algebra $\tilde{\mathfrak{g}} = \mathfrak{g} \times \mathfrak{g}$ of $\tilde{G}$:

$$\mathfrak{m} = \{(x, 0), x \in \mathfrak{g}\}$$
$$\mathfrak{m} = \{(0, x), x \in \mathfrak{g}\}$$
$$\mathfrak{m} = \{(x, -x), x \in \mathfrak{g}\}$$

With either of these, G is homogeneous reductive, and each decomposition lead to a different connection, called respectively *left, right or mean* connection. These are known as the Cartan-Schouten connections (Lorenzi and Pennec 2011). On can show that with the last connection, G is an affine symmetric space with the symmetries $\forall g \in G,\ s_g : h \mapsto gh^{-1}g$ (Pennec, Sommer, et al. 2020, Theorem 5.8). If G admits a bi-invariant metric, this connection is the Levi-Civita connection of the bi-invariant metric and G is a Riemannian symmetric space. An example of this case is the group of rotation matrices $SO(n)$. On the contrary, $SE(n)$ does not admit any bi-invariant metric, so it does not admit a Riemannian symmetric structure that coincides with its Cartan-Schouten connection.

The properties of symmetric spaces allow to compute the geodesics and parallel transport in closed-form, by the projection of the one-parameter subgroups of the Lie group G. We rarely use explicitly this structure in geomstats, as all the results were derived and implemented case by case. The structure is however useful to compute the curvature. Indeed, using the identification of $\mathfrak{m}$ with $T_o(M)$ induced by the restriction of $d\pi$ to $\mathfrak{m}$, we have for all $X, Y, Z \in \mathfrak{m}$

$$R(X, Y)Z = -[[X, Y], Z]$$

Moreover under topological conditions (M is simply connected and irreducible), O'Neill's formula (3.13) (page 93) simplifies and the sectional curvature is for $u, v \in \mathfrak{m}$ orthogonal

$$\kappa(u, v) = \frac{1}{2}\langle[[u, v], v], u\rangle - \frac{1}{2}\langle[[v, u], u], v\rangle.$$

To conclude this section, symmetric spaces offer a useful framework for statistics on manifolds beyond the convenience of the closed form solutions for the geodesics and parallel transport. Firstly, the normal distribution can be defined on all symmetric spaces, by generalizing the property that its maximum likelihood estimate coincides with the least-square problem of the Frechet mean (Said, Hajri, et al. 2018). The normalizing factor (or partition function) does not depend on the mean and can be computed in closed form thanks a decomposition of the Lie algebra.

Moreover, many computations such as interpolation or sampling can be performed in the subspace $\mathfrak{m}$ of the Lie algebra and be projected back to the symmetric space as in Gawlik and Leok 2018; Munthe-Kaas, Quispel, et al. 2014; Barp,

Kennedy, et al. 2019. Finally, there is a large theory of harmonic analysis on symmetric spaces (see Terras 1988), and limit theorems for stochastic processes such as Brownian motion.

3.6 Conclusion

In this chapter, we focused on the common implementation of Riemannian metrics that arise from group actions. The first case is for invariant metrics on Lie groups themselves, for which the invariance allows to reformulate all the geometric operations as algebraic operations in the Lie algebra. In particular, we applied this reasoning to derive a parallel transport equation, that lead to a stable implementation in geomstats.

The second case is that of quotient metrics, that arise on the orbits of the group actions. We identified the key ingredients to a common implementation in geomstats, and focused on the Kendall shape spaces and Bures-Wasserstein metric to exemplify these. This formulation allowed to easily implement parallel transport on Kendall shape spaces, and is to the best of our knowledge the only open-source Python implementation of these spaces. The space of correlation matrices seen as the quotient of SPD manifolds by the action of diagonal matrices is also implemented in geomstats, and others spaces such as the space of positive semi-definite matrices will be implemented in the near future.

Two particular instances of quotient spaces were then introduced: reductive homogeneous spaces and symmetric spaces. They are in fact direct generalizations of Lie groups and many operations from quotient spaces simplify. These concepts are however not used in the present implementation, as their properties actually allows to derive more efficient closed-form solutions.

Chapter 4

Numerical accuracy of ladder schemes for parallel transport

Parallel transport is a fundamental tool to perform statistics on Riemannian manifolds. Since closed formulae don't exist in general, practitioners often have to resort to numerical schemes. In this chapter, we focus on *ladder* methods, a popular class of algorithms that rely on iterative constructions of geodesic parallelograms. However, the literature lacks a clear analysis of their convergence performance. We give Taylor approximations of the elementary constructions of Schild's ladder and of the pole ladder with respect to the Riemann curvature of the underlying space. We then prove that these methods can be iterated to converge with quadratic speed, even when geodesics are approximated by numerical schemes. We also contribute a new link between Schild's ladder and the Fanning Scheme which explains why the latter naturally converges only linearly. Illustrations on usual manifolds of geomstats show that the theoretical errors we have established are measured with a high accuracy in practice. The special Euclidean group with an anisotropic left-invariant metric is of particular interest as it is a tractable example of a non-symmetric space in general, which reduces to a Riemannian symmetric space in a particular case. The results presented in this chapter are published in Guigui and Pennec 2021b.

Contents

4.1 Introduction

In many applications, it is natural to model data as points that lie on a manifold. Consequently, there has been a growing interest in defining a consistent framework to perform statistics and machine learning on manifolds (Pennec, Sommer, et al. 2020). A fruitful approach is to locally linearize the data by associating to each point a tangent vector. The parallel transport of tangent vectors then appears as a natural tool to compare tangent vectors across tangent spaces. For example, Brooks, Schwander, et al. 2019 use it on the manifold of symmetric positive definite (SPD) matrices to centralize batches when training a neural network. In Yair, Ben-Chen, et al. 2019, parallel transport is used again on SPD matrices for domain adaptation. In Kim, Dryden, et al. 2020, it is a key ingredient to spline-fitting in a Kendall shape space. In computational anatomy, it allows to compare longitudinal, intra-subject evolution across populations (Lorenzi, Ayache, et al. 2011; Lorenzi and Pennec 2014; Cury, Lorenzi, et al. 2016; Schiratti, Allassonnière, et al. 2017). It is also used in computer vision in e.g. Hauberg, Lauze, et al. 2013; Freifeld, Hauberg, et al. 2014. Finally, parallel transport is used to generalize statistical tests and sampling schemes on manifolds (Huckemann, Hotz, et al. 2010a).

However, there is usually no closed-form solution to compute parallel transport and one must use approximation schemes. Two classes of approximations have been developed. The first intends to solve ordinary or partial differential equations (ODE/PDE) related to the definition of parallel transport itself or to the Jacobi fields (Younes 2007; Louis, Charlier, et al. 2018). For instance, Kim, Dryden, et al. 2020 derived a homogeneous first-order differential equation stemming from the structure of quotient space that defines Kendall shape spaces. Louis, Charlier, et al. 2018 leveraged the relation between Jacobi fields and parallel transport to derive a numerical scheme that amounts to integrating the geodesic equations. They prove that a convergence speed of order one is reached. This scheme is particularly appealing as only the Hamiltonian of the metric is required, and both the main geodesic and the parallel transport are computed simultaneously when no-closed form solution is available for the geodesics.

The second class of approximations, referred to as *ladder* methods (Lorenzi, Ayache, et al. 2011; Lorenzi and Pennec 2014), consists in iterating elementary constructions of geodesic parallelograms (Figure 4.1). The most famous, Schild's ladder (SL), was originally introduced by Alfred Schild in 1970 although no published reference exist[1]. Its first appearance in the literature is in Misner, Thorne, et al. 1973 where it is used to introduce the Riemannian curvature. A proof that the construction of a geodesic parallelogram —i.e. one *rung* of the ladder— is an approximation of parallel transport was first given in Kheyfets, W. A. Miller, et al. 2000. However there is currently no formal proof that it converges when iterated along the geodesic along which the vector is to be transported, as prescribed in every description of the method, and because only the first order is given in Kheyfets, W. A. Miller, et al.

[1] Ehlers, Pirani, et al. 1972 is often cited but no mention of the scheme is made in this work

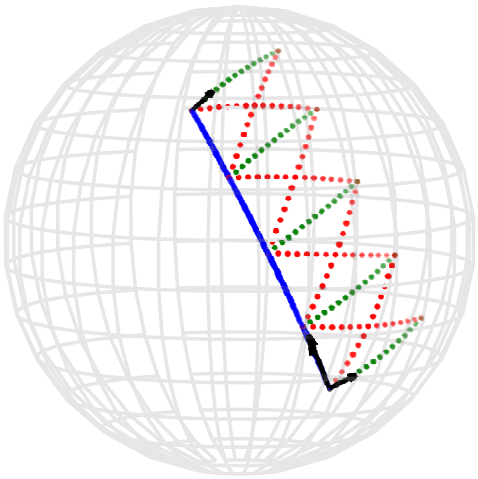


Figure 4.1: Schild's ladder on the sphere. The main geodesic is in blue, while the sides of the intermediate geodesic parallelograms are in dotted green and the diagonals in dotted red. The tangent vectors are represented by (scaled) black arrows. The construction is detailed in section 4.2.1

2000, Schild's ladder is considered to be a first-order method in the literature.

The first aim of this paper is at filling this gap. Using a Taylor expansion of the *double exponential* introduced by Gavrilov 2006; Gavrilov 2013, we compute an expansion of the SL construction up to order three, with coefficients that depend on the curvature tensor and its covariant derivatives. Understanding this expansion then allows to give a proof of the convergence of the numerical scheme with iterated constructions, and to improve the scheme to a second-order method. We indeed prove that an arbitrary speed of convergence in $\frac{1}{n^\alpha}$ can be obtained where $1 \leq \alpha \leq 2$ and n is the number of parallelograms, or rungs of the ladder. This improves over the assumption made in Louis, Charlier, et al. 2018 that only a first-order convergence speed can be reached with SL. These results are observed with a high accuracy in numerical experiments performed using geomstats.

A slight modification of this scheme, Pole Ladder (PL), was introduced in Lorenzi and Pennec 2014. It turns out that this scheme is exact in symmetric spaces, and in general that an elementary construction of PL is more precise by an order of magnitude than that of SL Pennec 2018. With the same method as for SL, we give a new proof of this result. By applying the previous analysis to PL, we demonstrate that a convergence speed of order two can be reached. Furthermore, we introduce a new construction which consists in averaging two SL constructions, and show that it is of the same order as the PL.

In most cases however, the exponential and logarithm maps are not available in closed form, and one also has to resort to numerical integration schemes. As for the Fanning Scheme (FS) (Louis, Charlier, et al. 2018), we prove that the ladder methods converge when using approximate geodesics and that all geodesics of the construction may be computed in one loop — i.e. using one integration step (e.g.

Runge-Kutta) per parallelogram construction, thus reducing the computational cost. We study the FS under the hood of ladder methods and show that its implementation make it very close to SL. However, their elementary steps differ at the third order, so that the FS cannot be improved to a second-order method like SL or the PL.

To observe the impact of the different geometric structures on the convergence, we study the Lie group of isometries of $\mathbb{R}^d$, the special Euclidean group $SE(d)$. Endowed with a left-invariant metric, this space is a Riemannian symmetric space if the metric is isotropic (Zefran, Kumar, et al. 1998). However, we show that it is no longer symmetric when using anisotropic metrics, and that geodesics may be computed by integrating the Euler-Poincaré equations. The same implementation thus allows to observe both geometric structures. Furthermore, using the results of Section 3.2 the curvature and its derivative may be computed explicitly and confirm our predictions. We treat this example in detail to demonstrate the impact of curvature on the convergence, and the code for the computations, and all the experiments of the paper are available online at https://github.com/nguigs/ladder-methods.

The first part of this paper is dedicated to Schild's ladder, while the second part applies the same methodology to the modified constructions. All the results are illustrated by numerical experiments along the way. The main proofs are in the text, and remaining details can be found in the appendices.

4.1.1 Notations and assumptions

We consider a complete Riemannian manifold (M, g) of finite dimension $d \in \mathbb{N}$. The associated Levi-Civita connection defines a covariant derivative ∇ and the parallel transport map. Denote Exp the Riemannian exponential map, and Log its inverse, defined locally. For $x \in M$, let T_xM be the tangent space at x. The map Exp_x sends T_xM to (a subset of) M, and we will often write $x_w = \mathrm{Exp}_x(w) \in M$ for $w \in T_xM$.

Let $\gamma : [0, 1] \to M$ be a smooth curve with $\gamma(0) = x$. For $v \in T_xM$, the parallel transport $\Pi^t_{\gamma,0} v \in T_{\gamma(t)}M$ of v along γ is defined as the unique solution at time $t \leq 1$ to the ODE $\nabla_{\dot\gamma(s)} X(s) = 0$ with $X(0) = v$. In general, parallel transport depends on the curve followed between two points. In this work however, we focus on the case where γ is a geodesic starting at x, and let w be its initial velocity, i.e. $\gamma(t) = \mathrm{Exp}_x(tw)$. Thus the dependence on γ in the notation Π_γ will be omitted, and we instead write Π^y_x for the parallel transport along the geodesic joining x to y when it exists and is unique. The methods developed below can be extended straightforwardly to piecewise geodesic curves.

We denote by $\|\cdot\|$ the norm defined on each tangent space by the metric g and R the Riemann curvature tensor that maps for any $x \in M$, $u, v, w \in T_xM$ to $R(u, v)w \in T_xM$. Throughout this paper, we consider γ to be contained in a compact set $K \subset M$ of diameter $\delta > 0$, and thus R and all its covariant derivatives $\nabla^n R$ can be uniformly bounded.

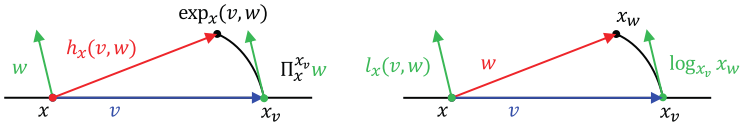


Figure 4.2: The double exponential and its inverse (left) and the neighboring logarithm (right) represented in a normal coordinate system centred at x. Geodesics are in black and tangent vectors are coloured

4.1.2 Double exponential and neighboring log

We now introduce the main tool to compute a Taylor approximation of the ladder constructions, the double exponential (also written Exp) (Gavrilov 2006; Gavrilov 2013). It is defined for $x \in M$, $(v,w) \in (T_xM)^2$ by first applying Exp_x to v, then Exp_{x_v} to the parallel transport of w along the geodesic from x to x_v:

$$\mathrm{Exp}_x(v,w) = \mathrm{Exp}_{x_v}(\Pi_x^{x_v} w).$$

As the composition of smooth maps, it is also a smooth map. As the exponential map is locally one-to-one and the parallel transport is an isomorphism, we may define the function $h_x : U_x \to T_xM$ on an open neighborhood U_x of $(0,0) \in T_xM \times T_xM$ such that $\mathrm{Exp}_x(U_x)$ is contained in a convex neighborhood of M and (Figure 4.2):

$$\mathrm{Exp}_x(h_x(v,w)) = \mathrm{Exp}_x(v,w) \quad \forall v, w \in T_xM.$$

As it is a smooth map, we can apply the Taylor theorem to $(s,t) \mapsto h_x(sv,tw)$. As stated in Gavrilov 2007, *"The function $h_x(v,w)$ is defined invariantly; therefore, the terms of its Taylor series are invariants of the connection. Such invariants must be expressed in terms of the curvature tensor, torsion tensor, and their derivatives."*

In this work we will require an approximation at order four (Gavrilov 2006, Section 8), although higher order terms can be found in Gavrilov 2013. For $s,t \in \mathbb{R}$ small enough:

$$\begin{aligned} h_x(sv,tw) &= \mathrm{Log}_x(\mathrm{Exp}_{x_{sv}}(\Pi_x^{x_{sv}} tw)) \\ &= sv + tw + \frac{st}{6} R(w,v)(sv + 2tw) \\ &\quad + \frac{st}{24}\Big((\nabla_v R)(w,sv)(5tw + 2sv) \\ &\quad + (\nabla_w R)(tw,v)(sv + 2tw) \Big) + q_5(sv,tw), \end{aligned} \tag{4.1}$$

where $q_5(sv,tw)$ contains homogeneous terms of degree five and higher, whose coefficients can be bounded uniformly in K (as they can be expressed in terms of the curvature and its covariant derivatives). Because we consider in this paper the Levi-Civita connection of a Riemannian manifold, there is no torsion term appearing here. Thus, taking $s,t \sim \frac{1}{n}$ for some $n \in \mathbb{N}$ large enough, $q_5(sv,tw) = O(\frac{1}{n^5})$. To simplify the notations, we write $O(5)$ in that sense. Formally (4.1) is similar to the BCH formula in Lie groups.

Similarly, Pennec 2019 introduced the *neighboring log* and computed its Taylor approximation. It is defined by applying the Exp_x map to small enough $v, w \in T_xM$ to obtain x_v, x_w then computing the log of x_w from x_v, and finally parallel transporting this vector back to x (see Figure 4.2). This defines $l_x : U_x \to T_xM$ by:

$$l_x(v, w) = \Pi_{x_v}^{x} \mathrm{Log}_{x_v}(x_w),$$

which relates to the double exponential by solving

$$w = h_x(v, l_x(v, w)).$$

Using (4.1), one can solve for the first terms of a Taylor expansion and obtains for $(v, w) \in U_x \subset T_xM \times T_xM$ Pennec 2019:

$$\begin{aligned} l_x(v, w) &= \Pi_{x_v}^{x} \mathrm{Log}_{x_v}(x_w) \\ &= w - v + \frac{1}{6} R(v, w)(2w - v) \\ &\quad + \frac{1}{24}\Big((\nabla_v R)(v, w)(3w - 2v) \\ &\quad + (\nabla_w R)(v, w)(2w - v) \Big) + O(5), \end{aligned} \tag{4.2}$$

where we have implicitly assumed v, w small enough and taken the time variables $s = t = 1$. We will do so in the next section to simplify the notations.

4.2 Schild's ladder

We now turn to the construction of Schild's ladder and to the analysis of this numerical scheme.

4.2.1 Elementary construction

The construction to parallel transport $v \in T_xM$ along the geodesic γ with $\gamma(0) = x$ and $\dot{\gamma}(0) = w \in T_xM$ (such that $(v, w) \in U_x$) is given by the following steps (Figure 4.3):

1. Compute the geodesics from x with initial velocities v and w until time $s = t = 1$ to obtain x_v and x_w. These are the sides of the parallelogram.
2. Compute the geodesic between x_v and x_w and the midpoint m of this geodesic. i.e.
$$m = \mathrm{Exp}_{x_v}\big(\frac{1}{2}\mathrm{Log}_{x_v}(x_w)\big).$$
This is the first diagonal of the parallelogram.
3. Compute the geodesic between x and m, let $a \in T_xM$ be its initial velocity. Extend it beyond m for the same length as between x and m to obtain z, i.e.
$$a = \mathrm{Log}_x(m); \qquad z = \mathrm{Exp}_x(2a) = x_{2a}.$$
This is the second diagonal of the parallelogram.

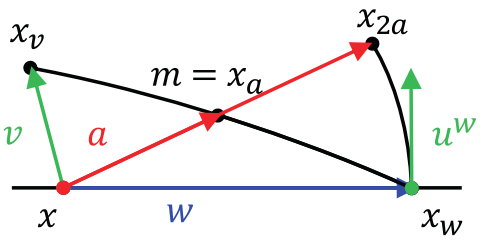


Figure 4.3: Construction of a geodesic parallelogram in Schild's ladder, represented in a normal coordinate system centred at x. Geodesics are in black and can be identified with tangent vectors (colored) when going through x.

4. Compute the geodesic between x_w and z. Its initial velocity u^w is an approximation of the parallel transport of v along the geodesic from x to x_w, i.e.
$$u^w = \mathrm{Log}_{x_w}(x_{2a}).$$

By assuming that there exists a convex neighborhood that contains the entire parallelogram, all the above operations are well defined. In the literature, this construction is then iterated along γ without further precision.

4.2.2 Taylor expansion

We can now reformulate this elementary construction in terms of successive applications of the double exponential and the neighboring logarithm maps. Let $b^v = \mathrm{Log}_{x_v}(x_w) \in T_{x_v}M$ be the initial velocity of the geodesic computed in step 2 of the above construction, and b its parallel transport to x. Therefore, we have $b = \Pi^x_{x_v} \mathrm{Log}_{x_v}(x_w) = l_x(v, w)$. The midpoint is now computed by $m = \mathrm{Exp}_{x_v}(\frac{b^v}{2})$, i.e.
$$m = \mathrm{Exp}_{x_v}(\Pi^{x_v}_x \frac{b}{2}).$$

Thus step 3. is equivalent to $a = h_x(v, \frac{1}{2}l_x(v, w))$. Combining the Taylor approximations (4.1) and (4.2), we obtain an expansion of $2a$. The computations are detailed in appendix C.1, and we only report here the third order, meaning that all the terms of the form $\nabla.R(\cdot,\cdot)\cdot$ are summarized in the term $O(4)$:
$$2a = w + v + \frac{1}{6}R(v, w)(w - v) + O(4) \tag{4.3}$$

We notice that this expression is symmetric in v and w, as expected. Furthermore, the deviation from the Euclidean mean of v, w (the parallelogram law) is explicitly due to the curvature. Accounting for this correction term is a key ingredient to reach a quadratic speed of convergence.

Now, in order to compute the error e_x made by this construction to parallel transport v, $e_x = \Pi^{x_w}_x v - u^w$, we parallel transport it back to x: define $u = \Pi^x_{x_w} u^w$ and $e = \Pi^x_{x_w} e_x = v - u \in T_xM$. Now as $u^w = \mathrm{Log}_{x_w}(x_{2a})$, we have $u = l_x(w, 2a)$.

Combining (4.2) with (4.3) (see appendix C.1), we obtain the first main result of this paper, a third order approximation of the Schild's ladder construction:

Theorem 4.2.1. *Let (M, g) be a finite dimensional Riemannian manifold. Let $x \in M$ and $v, w \in T_xM$ sufficiently small. Then the output u of one step of Schild's ladder parallel transported back to x is given by*

$$u = v + \frac{1}{2}R(w, v)v + O(4) \tag{4.4}$$

The fourth order and a bound on the remainder are detailed in appendix C.1. This theorem shows that Schild's ladder is in fact a second-order approximation of parallel transport. Furthermore, this shows that splitting the main geodesic into segments of length $\frac{1}{n}$ and simply iterating this construction n times will in fact sum n error terms of the form $R(\frac{w_i}{n}, v_i)v_i$, hence by linearity of R, the error won't necessarily vanish as $n \to \infty$. To ensure convergence, it is necessary to also scale v in each parallelogram. The second main contribution of this paper is to clarify this procedure and to give a proof of convergence of the numerical scheme when scaling both v and w. This is detailed in the next subsection.

4.2.3 Numerical scheme and proof of convergence

With the notations of the previous subsection, let us define $\text{schild}(x, w, v) = u^w \in T_{x_w}M$. We now divide the geodesic γ into segments of length $\frac{1}{n}$ for some $n \in \mathbb{N}^*$ large enough so that the previous Taylor expansions can be applied to $\frac{w}{n}$ and $\frac{v}{n}$. As mentioned before, v needs to be scaled as w. In fact let $\alpha \geq 1$ and consider the sequence defined by (see Figure 4.4)

$$\begin{aligned} v_0 &= v \\ v_{i+1} &= n^\alpha \cdot \text{schild}(x_i, \frac{w_i}{n}, \frac{v_i}{n^\alpha}), \end{aligned} \tag{4.5}$$

where $x_i = \gamma(\frac{i}{n}) = \text{Exp}_x\left(\frac{i}{n}w\right)$, $w_i = n\,\text{Log}_{x_i}(x_{i+1}) = \Pi_x^{x_i}w$. We now establish the following result, which ensures convergence of Schild's ladder to the parallel transport of v along γ at order at most two.

Theorem 4.2.2. *Let $(x_i, v_i, w_i)_{(i\leq n)}$ be the sequence defined as above. Then $\exists \tau > 0, \exists \beta > 0, \exists N \in \mathbb{N}, \forall n > N$,*

$$\|v_n - \Pi_x^{x_n}v\| \leq \frac{\tau}{n^\alpha} + \frac{\beta}{n^2}.$$

Moreover, τ is bounded by a bound on the sectional curvature, and β by a bound on the covariant derivative of the curvature tensor.

Proof. To compute the accumulated error, we parallel transport back to x the results v_n of the algorithm after n rungs. The error is then written as the sum of the errors

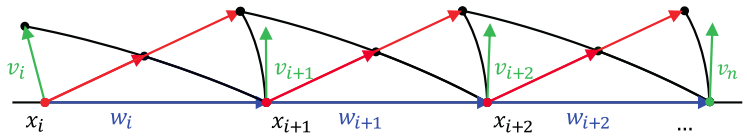


Figure 4.4: Schild's ladder, consisting in iterative constructions of geodesic parallelograms

at each rung, parallel transported to x (isometrically):

$$\begin{aligned}\frac{1}{n^\alpha}\|v_0 - \Pi^x_{x_n} v_n\| &= \|\sum_{i=0}^{n-1} \Pi^x_{x_i}\big(\frac{v_i}{n^\alpha} - \Pi^{x_i}_{x_i+1}\frac{v_{i+1}}{n^\alpha}\big)\| \\ &\leq \sum_{i=0}^{n-1} \|\Pi^x_{x_i}\big(\frac{v_i}{n^\alpha} - \Pi^{x_i}_{x_i+1}\frac{v_{i+1}}{n^\alpha}\big)\| \leq \sum_{i=0}^{n-1}\|e_i\|. \end{aligned} \tag{4.6}$$

By (4.4) and Lemma C.1.1, the error at each step can be written

$$e_i = \frac{v_i}{n^\alpha} - \Pi^{x_i}_{x_i+1}\frac{v_{i+1}}{n^\alpha} = \frac{1}{2n^{(1+2\alpha)}}R(v_i, w_i)v_i + r_4(\frac{w_i}{n}, \frac{v_i}{n^\alpha}). \tag{4.7}$$

where r_4 contains the fourth order residual terms, and is given in appendix C.1. We start as in Louis 2019 by assuming that v_i doesn't grow too much, i.e . there exists $s \in \{1, \dots, n\}$ such that $\forall i < s, \|v_i\| \leq 2\|v\|$. We will then show that the control obtained on v_n is tight enough so that when n is large enough, $\forall k \leq n, \|v_k\| \leq 2\|v\|$. With this assumption each term of the right-hand side (r.h.s.) can be bounded. As $\alpha \geq 1$, we apply Lemma C.1.1 (in appendix C.1): $\exists \beta > 0$ such that for n large enough:

$$\forall i \leq n, \|r_4(v_i, w_i)\| \leq \frac{\beta}{n^{\alpha+3}}.$$

Similarly, as $\forall i \leq n,\ \|w_i\| = \|w\|$ and by assumption $\|v_i\| \leq 2\|v\|$, we have

$$\|R(v_i, w_i)v_i\| \leq \|R\|_\infty \|v_i\|^2 \|w_i\| \leq 4\|v\|^2\|R\|_\infty\|w\|$$

Let $\tau = 2\|v\|^2\|R\|_\infty\|w\|$. This gives

$$\|e_i\| \leq \frac{\tau}{n^{(1+2\alpha)}} + \frac{\beta}{n^{\alpha+3}}. \tag{4.8}$$

As the r.h.s. does not depend on n, it can be plugged into (4.6) to obtain by summing for $i = 0, \dots, s-1 \leq n$:

$$\frac{1}{n^\alpha}\|v_0 - \Pi^x_{x_s} v_s\| \leq \sum_{i=0}^{s-1}\|e_i\| \leq \frac{\tau}{n^{2\alpha}} + \frac{\beta}{n^{\alpha+2}},$$

and finally

$$\|v - \Pi^x_{x_s} v_s\| \leq \frac{\tau}{n^\alpha} + \frac{\beta}{n^2}. \tag{4.9}$$

Now suppose for contradiction that for arbitrarily large n, there exists $k \leq n$ such that $\|v_k\| > 2\|v\|$ and choose $u(n) \in \{1, \dots, n\}$ minimal with this property, i.e. $\|v_{u(n)}\| > 2\|v\|$ so that (4.9) can be used with $s = u(n)$. Then we have:

$$\|v\| < \left|\|v\| - \|v_{u(n)}\|\right| \leq \|v - \Pi^x_{x_{u(n)}} v_{u(n)}\| \leq \frac{\tau}{n^\alpha} + \frac{\beta}{n^2}. \tag{4.10}$$

But the r.h.s. goes to 0 as n goes to infinity, leading to a contradiction. Therefore, for n large enough, $\forall i \leq n$, $\|v_i\| \leq 2\|v\|$, and the previous control on $\|v - \Pi^x_{x_s} v_s\|$ given by (4.9) is valid for $s = n$. The result follows by parallel transporting this error to x_n, which doesn't change its norm. □

Remark 4.2.1. *The proof allows to grasp the origin of the bounds in Theorem. 4.2.1. The term $O(n^{-\alpha})$ comes from $R(w,v)v$ in (4.4), as it is bilinear in v, so that the division by n^α is squared in the error, while we only need to multiply by n^α at the final step to recover the parallel transport of v. Similarly, as $R(w,v)v$ is linear in w, the division by n and summing of n terms doesn't appear in the error.*

On the other hand, the $O(n^{-2})$ comes from $\nabla_w R(w,v)w$ (from Lemma C.1.1). Indeed, this term is invariant by the division/multiplication by n^α, unlike other error terms where v appears several times, and the multilinearity w.r.t. w implies that it is a $O(n^{-3})$ that is then summed n times. We therefore use $\alpha = 2$ in practice.

Remark 4.2.2. *The bound on the curvature that we use can be related to a bound on the sectional curvature κ as follows. Recall*

$$\kappa(X,Y) = \frac{< R(Y,X)X, Y >}{\|X\|^2\|Y\|^2 - < X, Y >^2}. \tag{4.11}$$

Suppose it is bounded on the compact set K: $\forall X, Y \in \Gamma(K), |\kappa(X,Y)| < \|\kappa\|_\infty$ Therefore, for two linearly independent X, Y, write $\delta_{X,Y} = \frac{<X,Y>^2}{\|X\|^2\|Y\|^2} \in [0,1)$,

$$\begin{aligned} | < R(Y,X)X, Y > | &= |\kappa(X,Y)|(\|X\|^2\|Y\|^2 - < X, Y >^2) \\ &= |\kappa(X,Y)|\|X\|^2\|Y\|^2(1 - \delta_{X,Y}) \\ &\leq \|\kappa\|_\infty \|X\|^2\|Y\|^2. \end{aligned}$$

Now, the linear map $Y \mapsto R(Y,X)X$ is self-adjoint, so if $|\lambda_1(X)| \leq \ldots \leq |\lambda_d(X)|$ are its eigenvalues, we have on the first hand $|\lambda_d(X)| \leq \|\kappa\|_\infty \|X\|^2$ and on the second hand $\|R(Y,X)X\| \leq |\lambda_d(X)|\|Y\|$. Thus

$$\|R(Y,X)X\| \leq \|\kappa\|_\infty \|X\|^2 \|Y\|. \tag{4.12}$$

We then could have used $\tau = \frac{\delta^3 \|\kappa\|_\infty}{2}$ in the proof of theorem 4.2.2.

We notice in this proof and remark 4.2.1 that the terms of the Taylor expansion where v appears several times vanish faster thanks to the arbitrary exponent $\alpha \geq 1$. Together with Lemma C.1.1, this implies that the dominant term is the one where v appears only once (given explicitly in appendix C.1), which imposes here a speed

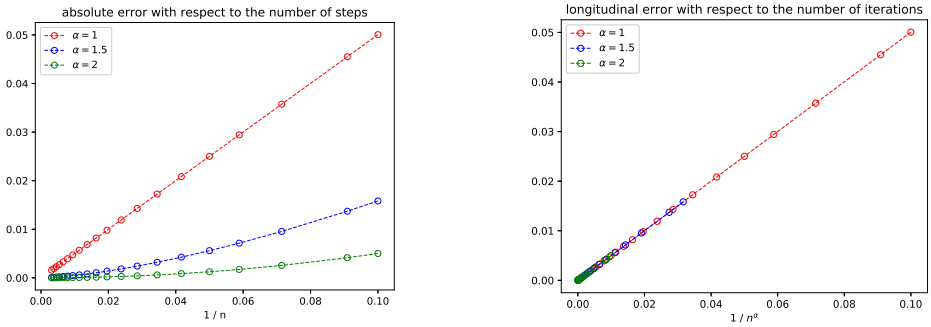


Figure 4.5: Error of the parallel transport of v along the geodesic with initial velocity w where v and w are orthonormal. Accordingly with our main result, any speed of convergence $1 \leq \alpha \leq 2$ can be reached. Left : Absolute error w.r.t $\frac{1}{n}$. Right: Longitudinal error w.r.t. $\frac{1}{n^\alpha}$: on S^2 this is a straight line with slope $\frac{1}{2}$.

of convergence of order two. We will use this key fact to compute the speed of convergence of the other schemes.

We now present numerical simulations that show the convergence bounds in two simple cases: the sphere S^2 and the space of 3×3 symmetric positive-definite matrices $SPD(3)$.

Example 4.2.1: The sphere

We consider the sphere S^2 as the subset of unit vectors of $\mathbb{R}^3$, endowed with the canonical ambient metric $< \cdot, \cdot >$. Recall that it is a Riemannian manifold of constant curvature, and the geodesics and parallel transport are available in closed form (Example 2.3.6 page 40).

Because the sectional curvature κ is constant, the fourth-order terms vanish and theorem 4.2.2 becomes $\|v - \Pi_{x_n}^x v_n\| \leq \frac{\tau}{n^\alpha}$ for $\alpha \geq 1$. This is precisely observed on figure 4.5. Moreover, using eq.(4.7), we have at each rung:

$$\begin{aligned} < \Pi_{x_i}^x e_i, w > =< e_i, w_i > &= \frac{1}{2n^{(1+2\alpha)}} < R(v_i, w_i)v_i, w_i > + O(\frac{1}{n^{(4+\alpha)}}) \\ &= \frac{-1}{2n^{(1+2\alpha)}}\kappa(v_i, w_i) = \frac{-1}{2n^{(1+2\alpha)}} + O(\frac{1}{n^{(4+\alpha)}}). \end{aligned}$$

Therefore, by summing as in the proof of theorem 4.2.2 we obtain:

$$< v_n - \Pi_x^{x_n} v, w_n >= \frac{1}{2n^\alpha} + O(\frac{1}{n^3}).$$

Thus the projection of the error onto w, which we call longitudinal error, allows to verify our theory: we can measure the slope of the decay with respect to $n^{-\alpha}$ and it should equal $\frac{1}{2}$. This is very precisely what we observe (Figure 4.5, right).

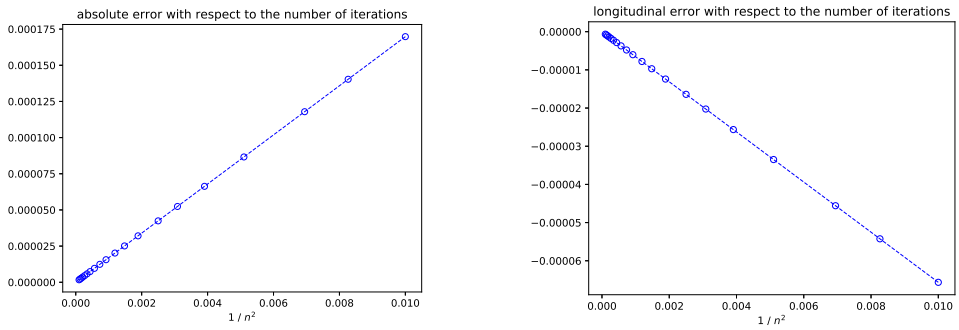


Figure 4.6: Error of the parallel transport of v along the geodesic with initial velocity w where v and w are orthonormal. Accordingly with our main result, a quadratic speed of convergence $\alpha = 2$ can be reached. Left : Absolute error w.r.t $\frac{1}{n^2}$. Right: Longitudinal error w.r.t. $\frac{1}{n^2}$: with the AI metric, this is a straight line with negative slope.

Example 4.2.2: SPD

We now consider $SPD(3)$ endowed with the affine-invariant metric (Pennec 2020, Section 3.3). Again, the formulas for the exp, log and parallel transport maps are available in closed form and given in Example 3.5.4 (page 113). Because the sectional curvature is always non-positive, the slope of the longitudinal error is negative, and this is observed on Figure 4.6.

4.3 Variations of Schild's ladder

We now turn to variations of SL. First, we revisit the Fanning Scheme using the neighboring log and show how close to SL it in fact is. Then we examine the pole ladder, and introduce the averaged SL. Furthermore, we introduce infinitesimal schemes, where the exp and log maps are replaced by one step of a numerical integration scheme. We give convergence results and simulations for the PL in this setting. This allows to probe the convergence bounds in settings where the underlying space is not symmetric, and therefore observe the impact of a non-zero ∇R.

4.3.1 The fanning scheme, revisited

The Fanning Scheme Louis, Charlier, et al. 2018 leverages an identity given in Younes 2007 between Jacobi fields and parallel transport. As proved in Louis, Charlier, et al. 2018, it converges linearly when dividing the main geodesic into n segments and using second-order integration schemes of the Hamiltonian equations. We recall the algorithm here with our notations and compare it to SL. For $v, w \in T_xM$, define the Jacobi field (JF) along $\gamma : t \mapsto \mathrm{Exp}_x(tw)$ at time h small

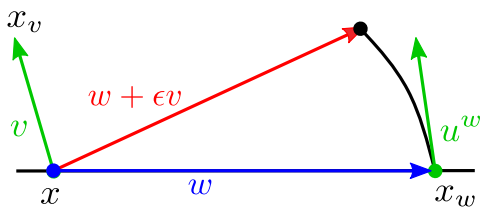


Figure 4.7: Elementary construction of the fanning scheme. For $\epsilon = \frac{1}{n}$, it is similar to that of SL (with $\alpha = 2$) except that the midpoint is not computed but $w + \epsilon v$ is used instead of $\text{Log}_x(m)$.

enough by:

$$J^v_{\gamma(t)}(h) = \frac{\partial}{\partial \epsilon}\bigg|_{\epsilon=0} \text{Exp}_{\gamma(t)}\big(h(\dot\gamma(t) + \epsilon v)\big).$$

One can then show that $J^v_{\gamma(t)}(h) = h\Pi^{t+h}_{\gamma,t} v + O(h^2)$ (Younes 2007). The Fanning Scheme consists in computing *perturbed* geodesics, i.e. with initial velocity $w + \epsilon v$ for some $\epsilon > 0$ and to approximate the associated JF by

$$J^v_{\gamma(t)}(h) = \frac{\text{Exp}_{\gamma(t)}\big(h(\dot\gamma(t) + \epsilon v)\big) - \text{Exp}_{\gamma(t)}\big(h\dot\gamma(t)\big)}{\epsilon}, \tag{4.13}$$

and the parallel transport of v along γ between 0 and h by

$$u^w = \frac{\text{Exp}_x(h(w + \epsilon v)) - \text{Exp}_x(hw)}{h\epsilon}.$$

This defines an elementary construction that is iterated n times with time-steps $h = \frac{1}{n}$, and Louis, Charlier, et al. 2018 showed that using $\epsilon = h$ yields the desired convergence. Let's consider that the finite difference approximation of (4.13) is in fact a first-order approximation of the log of $\text{Exp}_x\big(h(w+\epsilon v)\big)$ from $\text{Exp}_x(hw)$. Now this scheme is similar to SL, except that it uses $2a = w + \epsilon v$ instead of (4.3), i.e. the correction term according to curvature is not accounted for (compare figures 4.3 and 4.7). When using $h = \epsilon = \frac{1}{n}$, this corresponds to $\alpha = 2$ in our analysis of the sequence defined by 4.5. Similarly, we can compute a third order Taylor approximation of the error made at each step of the FS. To do so, introduce as in sec. 4.2.2 $u = \Pi^x_{x_{hw}} u^w$. Then

$$\begin{aligned} u &= l_x\big(hw, h(w + \epsilon v)\big) \\ &= h\epsilon v + \frac{h^2}{6} R(w, w + \epsilon v)\,(2h(w + \epsilon v) - hw) + O(h^4\epsilon) \\ &= h\epsilon v + \frac{h^3\epsilon}{6} R(w, v)(w + 2\epsilon v) + O(h^4\epsilon). \end{aligned}$$

But as $\epsilon = h$, the last term including v disappears in the $O(h^4\epsilon)$ and we obtain:

$$\frac{u}{h\epsilon} = v + \frac{h^2}{6} R(w, v)w + O(h^3). \tag{4.14}$$

We notice that unlike the expression given by Theorem 4.2.1, this approximation of FS contains a term linear in v (i.e. where v appears only once), and bilinear in w, so that when applied to $h = \frac{1}{n}$, summing error terms yields a global error of order $\frac{1}{n}$. This shows that a better speed cannot be reached. Moreover, we recognize the coefficient $\frac{1}{6n}$ computed explicitly on the sphere in Louis, Charlier, et al. 2018, sec. 2.3. As in the previous section, define $u^w = \mathrm{fs}(x, w, v)$ the result of the FS construction, and the sequence

$$\begin{aligned} v_0 &= v, \\ v_{i+1} &= n^2 \cdot \mathrm{fs}\left(x_i, \frac{w_i}{n}, \frac{v_i}{n^2}\right), \end{aligned} \tag{4.15}$$

where $x_i = \gamma(\frac{i}{n}) = \mathrm{Exp}_x\left(\frac{i}{n}w\right)$, $w_i = n\,\mathrm{Log}_{x_i}(x_{i+1}) = \Pi_x^{x_i} w$. Then it is straightforward to reproduce the proof of thm. 4.2.2 with (4.14) instead of (4.4) to show that $\exists \beta > 0, \exists N \in \mathbb{N}, \forall n > N$,

$$\|v_n - \Pi_x^{x_n} v\| \leq \frac{\beta}{n}.$$

This corroborates the result of Louis, Charlier, et al. 2018, although at this point geodesics are assumed to be available exactly. An improvement of the FS is to use a second-order approximation of the Jacobi field by computing two perturbed geodesics, but this doesn't change the precision of the approximation of the parallel transport. We implemented this scheme with this improvement and the result is compared with ladder methods in paragraph 4.3.4.3.

4.3.2 Averaged Schild's ladder

Focusing on eq. (4.4) of Theorem 4.2.1, we notice that the error term is even with respect to v. We therefore propose to average the results of a step of SL applied to v and to $-v$ to obtain $u^{(+)}$ and $u^{(-)}$. Using (C.2) of appendix C.1, we have:

$$u^{(+)} = v + \frac{1}{2}R(w,v)v + \frac{7}{48}\Big((\nabla_v R)(w,v)v + (\nabla_w R)(w,v)(\frac{8}{7}v + w)\Big) + O(5), \tag{4.16}$$

$$u^{(-)} = -v + \frac{1}{2}R(w,v)v - \frac{7}{48}\Big((\nabla_v R)(w,v)v + (\nabla_w R)(w,v)(w - \frac{8}{7}v)\Big) + O(5). \tag{4.17}$$

Hence

$$\frac{u^{(+)} - u^{(-)}}{2} = v + \frac{7}{48}\Big((\nabla_v R)(w,v)v + (\nabla_w R)(w,v)w\Big) + O(5). \tag{4.18}$$

This scheme thus allows to cancel out the third order term in an elementary construction. However, when iterating this scheme, the dominant term will be $(\nabla_w R)(w,v)w$, so that the speed of convergence remains quadratic anyway. Being also heavier to compute, this scheme has thus very little practical advantage to offer, while the pole ladder, detailed in the next section, present both advantages of being exact at the third order, and being much cheaper to compute.

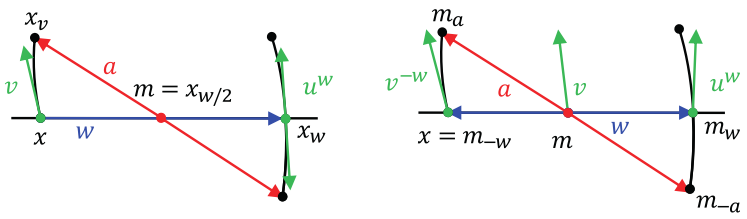


Figure 4.8: Elementary construction of the pole ladder with the previous notations (left), and new notations (right), in a normal coordinate system at m.

4.3.3 Pole ladder

The pole ladder was introduced by Lorenzi and Pennec 2014 to reduce the computational burden of SL. They showed that at the first order, the constructions where equivalent. Pennec 2018 latter gave a Taylor approximation of the elementary construction showing that each rung of pole ladder is a third order approximation of parallel transport, and additionally showed that PL is exact in affine (hence in Riemannian) locally symmetric spaces (the proof is reproduced in Guigui, Jia, et al. 2019). We first present the scheme and derive the Taylor approximation using the neighboring log.

4.3.3.1 Elementary construction

The construction to parallel transport $v \in T_xM$ along the geodesic γ with $\gamma(0) = x$ and $\dot\gamma(0) = w \in T_xM$ (such that $(v, w) \in U_x$) is given by the following steps (see Figure 4.8):

1. Compute the geodesics from x with initial velocities v and w until time $s = t = 1$ to obtain x_v and x_w, and $t = \frac{1}{2}$ to obtain the midpoint m. The main geodesic γ is a diagonal of the parallelogram.

2. Compute the geodesic between x_v and m, let $a \in T_mM$ be its initial velocity. Extend it beyond m for the same length as between x_v and m to obtain z, i.e.

$$a = \mathrm{Log}_m(x_v); \qquad z = \mathrm{Exp}_m(-a) = m_{-a}.$$

   This is the second diagonal of the parallelogram.

3. Compute the geodesic between x_w and z. The opposite initial velocity u^w is an approximation of the parallel transport of v along the geodesic from x to x_w, i.e.

$$u^w = -\mathrm{Log}_{x_w}(m_{-a}).$$

By assuming that there exists a convex neighborhood that contains the entire parallelogram, all the above operations are well defined.

4.3.3.2 Taylor approximation

We now introduce new notations, and centre the construction at the midpoint m (see Fig . 4.8, right). Both v, w are now tangent vectors at m, and v is the parallel transport of the vectors v^{-w} that we wish to transport between m_{-w} and m_w. With these new notations,

$$u = \Pi_{m_w}^m u^w = -l_m(w, -a)$$

where $a = h_m(-w, v)$. Therefore, using (4.1) and (4.2), we obtain the following (see appendix C.2 for the computations)

Theorem 4.3.1. *Let (M, g) be a finite dimensional Riemannian manifold. Let $m \in M$ and $v, w \in T_mM$ sufficiently small. Then the output u of one step of the pole ladder parallel transported back to m is given by*

$$u = v + \frac{1}{12}\big((\nabla_w R)(w, v)(5v - w) + (\nabla_v R)(w, v)(2v - w)\big) + O(5). \tag{4.19}$$

We notice that an elementary construction of pole ladder is more precise than that of Schild's ladder in the sense that there is no third order term $R(\cdot, \cdot)\cdot$. Using n^{-1} to scale w and $n^{-\alpha}$ to scale v as in SL, the term $(\nabla_w R)(w, v)w$ is the dominant term when iterating this construction. Indeed, as it is linear in v, the scaling does not influence the convergence speed as one needs to multiply the result by n^α at the final rung of the ladder to recover the parallel transport of v. The other terms are multilinear in v, so if $\alpha > 1$, they are small compared to $\frac{1}{n^3}(\nabla_w R)(w, v)w$. Summing n terms of the form $\frac{1}{n^3}(\nabla_w R)(w, v)w$ we thus obtain a quadratic speed of convergence, as for Schild's ladder, for any $\alpha \geq 1$. Note that using $1 < \alpha \leq 2$ will reduce the impact of the other terms in the error without changing the overall order of convergence. We henceforth use $\alpha = 1$.

As in Sec. 4.3.2, one could think of applying pole ladder to linear combinations of v, w to achieve a better performance, but this strategy is doomed as no linear combination can cancel out the dominant term $(\nabla_w R)(w, v)w$. As in the previous section, define $u^w = \text{pole}(m, w, v)$ the result of the PL construction (with the new notations), and the sequence

$$\begin{aligned} v_0 &= v \\ v_{i+1} &= n \cdot \text{pole}(m_i, \frac{w_i}{n}, \frac{v_i}{n}), \end{aligned} \tag{4.20}$$

where $m_i = \gamma(\frac{2i+1}{2n}) = \text{Exp}_m\left(\frac{2i}{n}w\right)$, $w_i = \dot\gamma(\frac{2i+1}{2n}) = \frac{n}{2}\,\text{Log}_{m_i}(m_{i+1}) = \Pi_m^{m_i} w$. Then it is straightforward to reproduce the proof of thm. 4.2.2 with (4.19) instead of (4.4) to show that $\exists \beta > 0, \exists N \in \mathbb{N}, \forall n > N$,

$$\|v_n - \Pi_m^{m_n} v\| \leq \frac{\beta}{n^2}, \tag{4.21}$$

and β can be bounded by the covariant derivative of the curvature.

This proves the convergence of the scheme with the same speed as SL. Note however that when iterating the scheme, unlike SL, the v_i's and the geodesics that

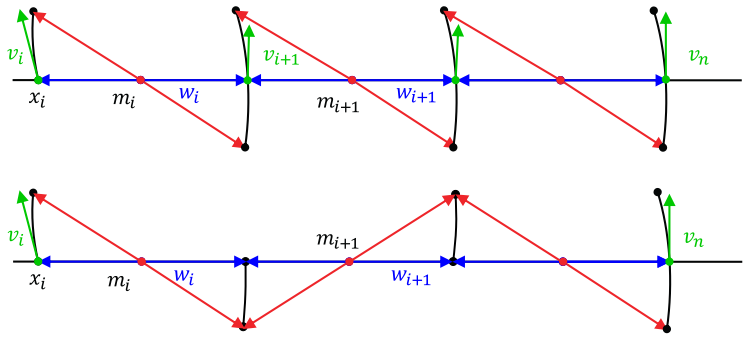
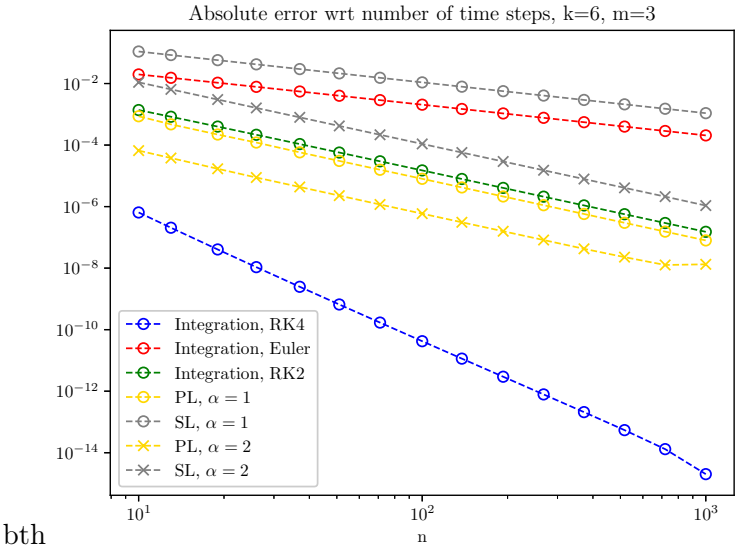


Figure 4.9: The pole ladder, consisting in iterations of elementary constructions (top). It can be simplified to lighten the computational burden of computing geodesics (bottom), only one log and one exp need to be computed from m_i at each iteration.

correspond to the rungs of the ladder need not being computed explicitly, except at the last step. This is shown Figure 4.9, bottom row. This greatly reduces the computational burden of the PL over SL.

Example 4.3.1: Kendall Shape Spaces

We now reproduce the experiments of Examples 4.2.1 and 4.2.2 on Kendall shape spaces using the setting of Section 3.3.3 and Example 3.4 (page 99). The results are displayed in a log-log plot on Figure 4.10 for $k = 6$ landmarks in $\mathbb{R}^3$. It is one of the examples of non-symmetric spaces where PL and SL can be compared. PL reaches quadratic convergence regardless the scaling exponent α, but with a different constant that translates into a different intercept in log scale, while SL's convergence speed is directly related to α. Ladder methods thus compare with the RK method of order two, and are slower than the order-four scheme.



bth

Figure 4.10: Comparison of the integration of parallel transport equation in the pre-shape space with the ladder methods. The norm of the absolute error is represented with respect to the number of steps in a log-log scale. Although PL, SL and RK2 reach the same slope, the speed of convergence differs by a multiplicative constant and PL converges faster than SL and RK2 but slower than RK4.

4.3.4 Infinitesimal schemes with geodesic approximations

When geodesics are not available in closed form, we replace the exp map by a fourth-order numerical scheme (e.g. Runge-Kutta (RK)), and the log is obtained by gradient descent over the initial condition of exp. It turns out that only one step of the numerical scheme is sufficient to ensure convergence, and keeps the computational complexity reasonable. As only one step of the integration schemes is performed, we are no longer computing geodesic parallelograms, but infinitesimal ones, and thus refer to this variant as *infinitesimal scheme*. We here detail the proof for the PL, but it could be applied to SL as well as the other variants.

Remark 4.3.1. *While in this section we use the geodesic equation written with Christoffel symbols, any formulation of the geodesic equation (such as the Euler-Poincaré equation on Lie groups, or the Hamilton Jacobi equations) can be used. In practice, as the number of Christoffel symbols is n^3 for an n-dimensional manifold, these are rarely computed. This prohibits the direct integration of the parallel transport equation written with Christoffel symbols. On the other hand, as there is a wide literature on variational problems and symplectic integrators, these methods are often preferred to compute geodesics. The infinitesimal ladder schemes thus allow to leverage state-of-the art geodesic solvers for the purpose of computing parallel transport.*

However, in some cases the parallel transport equation can be written with known quantities, and a direct integration with high-order integration scheme will then provide faster solutions to compute parallel transport. This is explored e.g. on Lie groups in Guigui and Pennec 2021a and on Kendall shape spaces in Guigui, Maignant, et al. 2021 and reported in paragraph 4.3.4.2 for the former and Example 4.3.1 for the latter.

Furthermore, infinitesimal schemes are used in computational anatomy, on approximations of the infinite-dimensional group of diffeomorphisms (Lorenzi and Pennec 2014). This section thus provides a proof that this approach is valid in finite dimension. Finally, infinitesimal ladder schemes may also be useful in the context of discrete geodesics (Berkels, Effland, et al. 2015).

4.3.4.1 Derivation and proof

More precisely, consider the geodesic equation written as a first order equation of two variables in a global chart Φ of M, that defines for any $p \in M$ a basis of T_pM, written $B_x^\Phi = \frac{\partial}{\partial x^i}\Big|_x$, $i = 1, \dots, d$:

$$\begin{cases} \dot{x}^k(t) = v^k(t), \\ \dot{v}^k(t) = -\Gamma_{ij}^k v^i(t) v^j(t), \end{cases} \tag{4.22}$$

where Γ_{ij}^k are the Christoffel symbols, and we use Einstein summation convention. Let $rk : TM \times \mathbb{R}_+ \to TM$ be the map that performs one step of a fourth-order numerical scheme (e.g. RK4), i.e. it takes as input $(x(t), v(t)), h)$ and returns an approximation of $(x(t+h), v(t+h))$ when (x, v) is solution of the system (4.22). In our case, the step-size $h = \frac{1}{n}$ is used. By fourth-order, we mean that we have the following local *truncation* error relative to the 2-norm of the global coordinate chart Φ:

$$\|x(t+h) - rk_1(x(t), v(t), h)\|_2 \leq \tau_1 h^5,$$

and global accumulated error after n steps with step size $h = \frac{1}{n}$:

$$\|x(1) - \tilde{x}_n\|_2 \leq \frac{\tau_1}{n^4},$$

where by rk_1 we mean the projection on the first variable, and $\tau_1 > 0$ is a constant.

This means that any point on the geodesic is approximated with error $O(\frac{1}{n^4})$. As we are working in a compact domain, and all the norms are equivalent, the previous bounds can be expressed in Riemannian distance d:

$$d\big(x(t+h), rk_1(x(t), v(t), h)\big) \leq \tau_2 h^5 \tag{4.23}$$

$$d\big(x(1), \tilde{x_n}\big) \leq \frac{\tau_2}{n^4}. \tag{4.24}$$

Furthermore, the inverse of $v \mapsto rk_1(x, v, h)$ can be computed for any x by gradient descent, and we assume[2] that any desired accuracy can be reached. We

[2] see remark 4.3.2 thereafter about the validity of this hypothesis

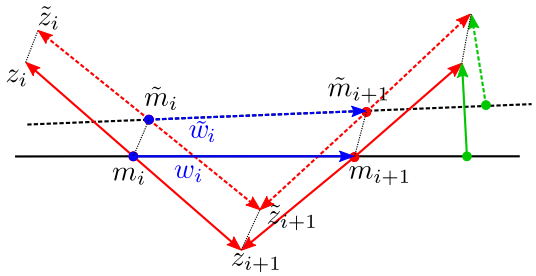


Figure 4.11: Representation of one iteration of the infinitesimal PL scheme. The exact geodesics and logs are plain lines, while their approximations with a numerical scheme are dashed.

write $\tilde{v} = rk_x^{-1}(y)$ for the optimal v such that $d(rk(x, v, \frac{1}{n}), y) = o(\frac{1}{n^5})$, and assume that $\|rk_x^{-1}(y) - \operatorname{Log}_x(y)\| \leq \frac{\tau_3}{n^4}$. Note that the step size does not appear explicitly in the notation rk^{-1}, but $h = \frac{1}{n}$ is always used. As in practice, $x \in M, v, w \in T_xM$ are given, the initial w_0 is tangent at x, and then for $i > 0$, w_i will be tangent at m_i as in (4.20), but w_i maps m_i to m_{i+1} and this must correspond to one step of size $\frac{1}{n}$ in the discrete scheme, so there is a factor two that differs from the definition of w_i in (4.20). Now define also $\tilde{m}_i, \tilde{w}_i, \tilde{v}_i$, using rk instead of exp and log, that is (see figure 4.11):

$$\begin{aligned} m_0 &= \operatorname{Exp}_x(\frac{w}{2n}) & z_0 &= \operatorname{Exp}_x(\frac{v}{n}) \\ m_{i+1} &= \operatorname{Exp}_{m_i}(\frac{w_i}{n}) & z_{i+1} &= \operatorname{Exp}_{m_i}(-\operatorname{Log}_{m_i}(z_i)) \\ \tilde{m}_0, \frac{\tilde{w}_0}{2} &= rk(x, \frac{w}{2}, \frac{1}{n}) & \tilde{z}_0 &= rk_1(x, v, \frac{1}{n}) \\ \tilde{m}_{i+1}, \tilde{w}_{i+1} &= rk(\tilde{m}_i, \tilde{w}_i, \frac{1}{n}) & \tilde{z}_{i+1} &= rk_1(\tilde{m}_i, -rk_{\tilde{m}_i}^{-1}(\tilde{z}_i), \frac{1}{n}). \end{aligned}$$

Finally let $x_n = \operatorname{Exp}(w), v_n = (-1)^n n \operatorname{Log}_{x_n}(z_n)$ and their approximations $\tilde{x}_n = rk(\tilde{m}_n, \frac{\tilde{w}_n}{2}, \frac{1}{n})$ and $\tilde{v}_n = n \cdot rk_{\tilde{x}_n}^{-1}(\tilde{z}_n)$.

We will prove that this approximate sequence converges to the true parallel transport of v:

Theorem 4.3.2. *Let $(\tilde{v}_n)_n$ be the sequence defined above, corresponding to the result of the pole ladder with approximate geodesics computed by a fourth-order method in a global chart. Then*

$$\|\Pi_x^{x_n} v - \tilde{v}_n\| = O(\frac{1}{n^2}).$$

Proof. It suffices to show that there exists $\beta' > 0$ such that for n large enough $\|\tilde{v}_n - v_n\| \leq \frac{\beta'}{n^2}$. Indeed, by (4.21), for n large enough:

$$\|\Pi_x^{x_n} v - \tilde{v}_n\| \leq \|\Pi_x^{x_n} v - v_n\| + \|\tilde{v}_n - v_n\| \leq \frac{\beta + \beta'}{n^2}.$$

The approximations made when computing the geodesics with a numerical scheme accumulate at three steps: (1) the RK scheme compared to the true geodesic, this

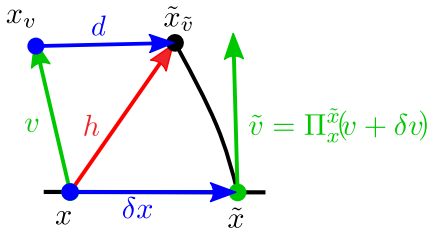


Figure 4.12: Visualization for the two lemmas: Lemma 4.3.1 seeks to bound the norm of d given δx and δv while Lemma 4.3.2 bounds the norm of δv given d and δx. Here h is the shorthand for $h_x(\delta x, v + \delta v)$ and d for $\operatorname{Log}_{x_v}(\tilde{x}_{\tilde{v}})$.

is controlled with the above hypotheses, (2) the distance between the results of the exp map when both the footpoint and the input vector vary a little, this is handled in Lemma 4.3.1 below, and (3) the difference between the results of the log map when both the foot-point and the input vary a little, this is similar to (2) and is handled in Lemma 4.3.2. See figure 4.12 for a visual intuition of those lemmas.

Lemma 4.3.1. *$\forall x, \tilde{x} \in M$ such that x and $\tilde{x}$ are close enough, for all $v \in T_xM, \tilde{v} \in T_{\tilde{v}}M$ such that both v and $\delta v = \Pi_{\tilde{x}}^x \tilde{v} - v$ are small enough, we have:*

$$d\big(\operatorname{Exp}_x(v), \operatorname{Exp}_{\tilde{x}}(\tilde{v})\big) \leq d(x, \tilde{x}) + \|\Pi_{\tilde{x}}^x \tilde{v} - v\|.$$

Proof. Let $\delta x = \operatorname{Log}_x(\tilde{x})$, $x_v = \operatorname{Exp}_x(v)$ and $\tilde{x}_{\tilde{v}} = \operatorname{Exp}_{\tilde{x}}(\tilde{v})$. By the definition of the double exp and δv, $\tilde{x}_{\tilde{v}} = \operatorname{Exp}_x(h_x(\delta x, v + \delta v))$. Then by the definition of the neighboring log, we have

$$\operatorname{Log}_{x_v}(\tilde{x}_{\tilde{v}}) = \operatorname{Log}_{x_v}(\operatorname{Exp}_x(h_x(\delta x, v + \delta v))) = \Pi_x^{x_v} l_x(v, h_x(\delta_x, v + \delta v)). \tag{4.25}$$

Using the Taylor approximations (4.1) and (4.2) truncated at the order of $\|v\|$, we obtain

$$\Pi_{x_v}^x \operatorname{Log}_{x_v}(\tilde{x}_{\tilde{v}}) = l_x(v, \delta x + v + \delta v) = \delta x + \delta v.$$

So that the Riemannian distance d between x_v and $\tilde{x}_{\tilde{v}}$ is

$$d = \|\operatorname{Log}_{x_v}(\tilde{x}_{\tilde{v}})\| \leq \|\delta x\| + \|\delta v\|,$$

and the result follows. □

Lemma 4.3.2. *$\forall x, \tilde{x}, z \in M$ close enough to one another, we have in the metric norm*

$$\|\operatorname{Log}_x(z) - \Pi_{\tilde{x}}^x \operatorname{Log}_{\tilde{x}}(\tilde{z})\| \leq d(x, \tilde{x}) + d(z, \tilde{z}). \tag{4.26}$$

Proof. The proof is similar to that of Lemma 4.3.1 except that this time it is the norm of $\|\delta v\|$ that needs to be bounded by $d(x_v, \tilde{x}_{\tilde{v}})$. □

Now, we first show that the sequence $\big(\delta_i = d(\tilde{z}_i, z_i)\big)_i$ verifies an inductive relation, such that it is bounded by $\frac{1}{n^3}$, and then we will use Lemma 4.3.2 to conclude. We first write

$$\|\operatorname{Log}_{m_i}(z_i) - rk^{-1}_{\tilde{m}_i}(\tilde{z}_i)\| \leq \|\operatorname{Log}_{m_i}(z_i) - \operatorname{Log}_{\tilde{m}_i}(\tilde{z}_i)\| + \|\operatorname{Log}_{\tilde{m}_i}(\tilde{z}_i) - rk^{-1}_{\tilde{m}_i}(\tilde{z}_i)\|.$$

The second term on the r.h.s. corresponds to the approximation of the log by gradient descent and is bounded by hypothesis. For the first term, $d(m_i, \tilde{m}_i) \leq \frac{\tau_2}{n^4}$ by hypothesis on the scheme (4.24). Suppose for a proof by induction on i that $\delta_i = d(\tilde{z}_i, z_i) \leq \frac{\tau_2}{n}$ and $d(m_i, z_i) \leq \frac{\|w\|+2\|v\|}{n}$. This is verified for $i = 0$ and allows to apply Lemma 4.3.2 for n large enough, so

$$\|\operatorname{Log}_{m_i}(z_i) - rk^{-1}_{\tilde{m}_i}(\tilde{z}_i)\| \leq d(\tilde{m}_i, m_i) + \delta_i + \frac{\tau_3}{n^4}. \tag{4.27}$$

Furthermore, by Lemma 4.3.1 applied to $v = -\operatorname{Log}_{m_i}(z_i)$ and $\tilde{v} = -rk^{-1}_{\tilde{m}_i}(\tilde{z}_i)$, which are sufficiently small by the induction hypothesis,

$$\begin{aligned}
\delta_{i+1} = d(z_{i+1}, \tilde{z}_{i+1}) &= d\big(rk_1(\tilde{m}_i, -rk^{-1}_{\tilde{m}_i}(\tilde{z}_i), \frac{1}{n}), \operatorname{Exp}_{m_i}(-\operatorname{Log}_{m_i}(z_i))\big) \\
&\leq d\big(rk_1(\tilde{m}_i, -rk^{-1}_{\tilde{m}_i}(\tilde{z}_i), \frac{1}{n}), \operatorname{Exp}_{\tilde{m}_i}(-rk^{-1}_{\tilde{m}_i}(\tilde{z}_i))\big) \\
&\qquad + d\big(\operatorname{Exp}_{\tilde{m}_i}(-rk^{-1}_{\tilde{m}_i}(\tilde{z}_i)), \operatorname{Exp}_{m_i}(-\operatorname{Log}_{m_i}(z_i))\big) \\
&\leq \frac{\tau_2}{n^5} + d(m_i, \tilde{m}_i) + \|\operatorname{Log}_{m_i}(z_i) - rk^{-1}_{\tilde{m}_i}(\tilde{z}_i)\|.
\end{aligned}$$

And combining the two results, we obtain $\delta_{i+1} \leq \frac{(2n+1)\tau_2 + n\tau_3}{n^5} + \delta_i$, which completes the induction for δ_i. For $d(z_i, m_i)$ we have:

$$d(z_{i+1}, m_{i+1}) = d(z_i, m_{i+1}) \leq d(x_i, z_i) + d(x_i, m_i) = \|\frac{v_i}{n}\| + \|\frac{w_i}{n}\|. \tag{4.28}$$

In the section on SL, we proved that $\|v_i\| \leq 2\|v\|$ for n large enough. This applies here as well, and the fact that w_i is the parallel transport of w completes the proof by induction.

By summing the terms for $i = 0, \ldots, n-1$, and using $\delta_0 \leq \frac{\tau_2}{n^5}$ by (4.23), we thus have $\delta_n \leq \frac{\tau_3 + 2\tau_2}{n^3} + \frac{n+1}{n^5}\tau_2$. We finally apply Lemma 4.3.2 to the scaled $v_n, \tilde{v}_n$, as x_n and $\tilde{x}_n$ are close enough:

$$\begin{aligned}
\frac{1}{n}\|v_n - \tilde{v}_n\| &= \|\operatorname{Log}_{x_n}(z_n) - rk^{-1}_{\tilde{x}_n}(\tilde{z}_n)\| \\
&\leq \|\operatorname{Log}_{x_n}(z_n) - \operatorname{Log}_{\tilde{x}_n}(\tilde{z}_n)\| + \|\operatorname{Log}_{\tilde{x}_n}(\tilde{z}_n) - rk^{-1}_{\tilde{x}_n}(\tilde{z}_n)\| \\
&\leq d(x_n, \tilde{x}_n) + \delta_n + \frac{\tau_2}{n^5}.
\end{aligned}$$

So that for n large enough $\|\tilde{v}_n - v_n\| \leq \frac{\beta'}{n^2}$ for some $\beta' > 0$. □

Remark 4.3.2. *To justify the hypothesis on rk^{-1}, namely, $\|rk^{-1}_x(y) - \operatorname{Log}_x(y)\|_2 \leq \frac{\tau}{n^5}$, we consider the problem $(\mathcal{P})$ which corresponds to an energy minimization. Working in a convex neighborhood, it admits a unique minimizer v^*:*

$$\min \frac{1}{2}\|v\|_2^2 \qquad s.t. \qquad \operatorname{Exp}_x(v) = y. \tag{$\mathcal{P}$}$$

The constraint can be written with Lagrange multipliers,

$$\min \frac{1}{2}\|v\|_2^2 + \lambda \| \operatorname{Exp}_x(v) - y\|_2^2. \tag{$\mathcal{P}'$}$$

Now as the Exp *map is approximated by rk_1 at order 5 locally, writing for any v and h small enough, $\| \operatorname{Exp}_x(hv) - y\|_2 \leq \|rk_1(x, v, h) - y\|_2 + \tau_1 h^5$, ($\mathcal{P}'$) is equivalent to*

$$\min \|rk_1(x, v, h) - y\|_2^2 + \frac{1}{2\lambda}\|v\|_2^2, \tag{$\mathcal{Q}$}$$

which is solved by gradient descent (GD) until a convergence tolerance $\epsilon \leq h^5$ is reached.

4.3.4.2 Numerical simulations

It is not relevant to compare the PL with SL on spheres or SPD matrices as in the previous section, as theses spaces are symmetric and thus the PL is exact (Guigui, Jia, et al. 2019). We therefore use the result of section 3.2 and focus on the Lie group of isometries of $\mathbb{R}^3$, endowed with a left-invariant metric g with the diagonal matrix at identity:

$$G = \operatorname{diag}(1, 1, 1, \beta, 1, 1),$$

where the first three coordinates correspond to the basis of the Lie algebra of the group of rotations, while the last three correspond to the translation part, and $\beta > 0$ is a coefficient of anisotropy. These metrics were considered in Zefran, Kumar, et al. 1998 in relation with kinematics, and visualization of the geodesics was provided, but no result on the curvature was given. Using the tools of section 3.2, we compute explicitly the covariant derivative of the curvature and deduce (proof given in appendix C.3)

Lemma 4.3.3. *$(SE(3), g)$ is locally symmetric, i.e. $\nabla R = 0$, if and only if $\beta = 1$.*

This allows to observe the impact of ∇R on the convergence of the PL. In the case where $\beta = 1$, the Riemannian manifold $(SE(3), g)$ corresponds to the direct product $SO(3) \times \mathbb{R}^3$ with the left-invariant product metric formed by the canonical bi-invariant metric on the group of rotations $SO(3)$ and the canonical inner-product of $\mathbb{R}^3$. Therefore the geodesics can be computed in closed form. Note that the left-invariance refers to the group law which encompasses a semi-direct action of the rotations on the translations. When $\beta \neq 1$ however, geodesics are no longer available in closed form and the infinitesimal scheme is used, with the geodesic equations computed numerically (detailed in the previous chapter, section 3.2). In this case, we compute the pole ladder with an increasing number of steps, and then use the most accurate computation as reference to measure the empirical error.

Results are displayed on Figure 4.13. Firstly, we observe very precisely the quadratic convergence of the infinitesimal scheme, as straight lines are obtained when plotting the error against $\frac{1}{n^2}$. Secondly, we see how the slope varies with β: accordingly with our results, it cancels for $\beta = 1$ and increases as β grows.

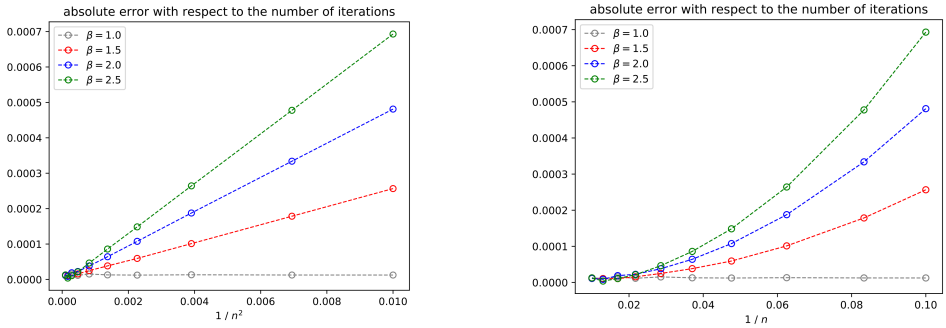


Figure 4.13: Error of the parallel transport of v along the geodesic with initial velocity w where v and w are orthonormal basis vectors (they stay the same for all values of β). In accordance with our main result, a quadratic speed of convergence is reached, and the speed depends on the asymmetry of the space, itself induced by the anisotropy β of the metric. Left : Absolute error w.r.t $\frac{1}{n^2}$: these are straight lines with slopes growing with β. Right: Absolute error w.r.t. $\frac{1}{n}$ showing the quadratic convergence.

Finally, for completeness, we compare Schild's ladder and the pole ladder in this context. We cannot choose two basis vectors v, w (that don't change when β changes) such that $R(w,v)v \neq 0$ when $\beta = 1$ and $\nabla_w R(w,v)w \neq 0$ when $\beta \neq 1$. Indeed, the former condition implies that v, w are infinitesimal rotations, which implies $\forall \beta, \nabla_w R(w,v)w = 0$. Therefore, we choose v, w such that the latter condition is verified, so that the SL error is of order four in our example (but it is not exact), and it cannot be distinguished from PL, but this is only a particular case. The results are shown on Figure 4.14. Note that due to the larger number of operations required for SL and as smaller quantities are involved as $\alpha = 2$, our implementation is less stable and diverges when n grows too much (~ 50). As expected when $\beta > 1$, the speeds of convergence are of the same order for PL and SL, but the multiplicative constant is smaller for PL.

We also compare ladder methods to the Fanning Scheme, and as expected the quadratic speed of convergence reached by ladder methods yields a far better accuracy even for small n. For completeness, we also compare the pole ladder with the integration of our reduced parallel transport equation on Lie groups with a left-invariant metric (Subsection 3.2.3). The precision reached by the latter depends on the order of the numerical scheme used for integration. The results are displayed on Figure 4.15 in a log-log plot representing the error between a reference value for the parallel transport and the value obtained with n steps in the discretization of the integral scheme, or in the pole ladder. The reference value is the one obtained by the integration scheme with RK steps of order four for $n = 1000$. As expected, we reach convergence speeds of order two for the pole ladder and the RK2 scheme, while the RK4 schemes is of order four. Both integration methods are more stable, than the pole ladder.

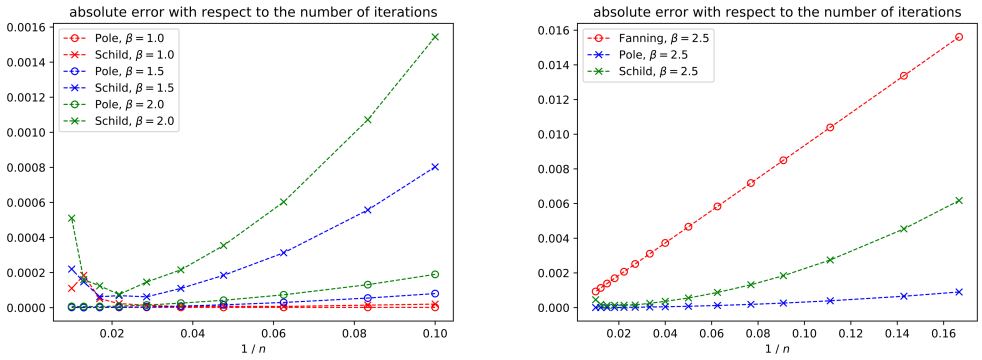


Figure 4.14: left: Comparison of the speed of convergence of SL and PL for two orthonormal basis vectors of $T_xSE(3)$ depending on the coefficient of anisotropy β of the metric. PL converges faster and is much more stable right: Comparison of SL, the PL and the FS in the same setting with fixed β. (Note that absolute and relative errors are the same here as v has unit norm.)

We now compare the infinitesimal SL, PL and the FS in terms of computational cost.

4.3.4.3 Remarks on complexity

At initialization, ladder methods require to compute x_v, with one call to the numerical scheme rk (e.g. Runge-Kutta). Then the main geodesic needs to be computed, for SL and FS it requires n calls to rk. For PL, we only take a half step at initialization, to compute the first midpoint, then compute all the m_is so that one final call to rk is necessary to compute the final point of the geodesic x_n, totalling $n+1$ calls. Then at every iteration, considering given m_i, z_i, one needs to compute an inversion of rk, and then shoot with twice (or minus for PL) the result. In practice the inversion of rk by gradient descent (i.e. shooting) converges in about five or six iterations, each requiring one call to rk. This operation thus requires less than ten calls to rk. Additionally, for SL only, the midpoint needs to be computed by one inversion and one call to rk, thus adding ten calls to the total. At the final step, another inversion needs to be computed, adding less than ten calls. Therefore, SL requires $21n + 11$ calls to rk, the PL $11(n + 1)$. In contrast, the FS only requires to compute one (or two) perturbed geodesic and finite differences instead of the approximation of the log at every step. It therefore require $3n$ calls to rk.

Moreover as the FS is intrinsically a first-order scheme, a second-order rk step is sufficient to guarantee the convergence, thus requiring only two calls to the geodesic equation —the most significantly expensive computation at each iteration— while ladder methods require a fourth-order scheme, which is twice as expensive. However, this additional cost is quickly balanced as only $O(\sqrt{n})$ steps are needed to reach a given accuracy, where $O(n)$ are required for the FS. Comparing the values of $n \mapsto 4 * \frac{11}{\sqrt{n+1}}$ and $n \mapsto 2 * \frac{3}{n}$, we conclude that PL is the cheapest option as soon as

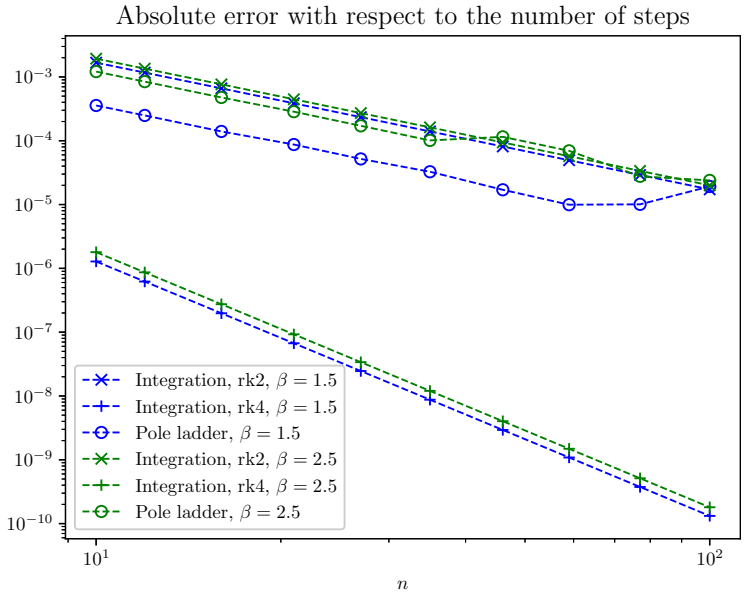


Figure 4.15: Comparison of the integration of the reduced equation with the pole ladder. The norm of the absolute error is represented with respect to the number of steps in a log-log scale. The pole ladder is less stable than the methods by integration.

we want to achieve a relative accuracy better than 2.10^{-2}. Note that this doesn't take into account the constants β, that may differ, but the previous numerical simulations show that these estimates are valid: a regression on the cost for the PL gives $y = 41n + 45$. Finally, in the experiment of Figure 4.14, the PL with $n = 6$ yields a precision of 8.10^{-4} for a cost of 304 calls to the equations, while $n = 250$ is required to reach that precision with the FS, for a cost of 1500 calls. For a similar computational budget, the PL reaches a precision $\sim 2.10^{-5}$ with $n = 37$.

4.4 Conclusion

In this chapter, we analyzed the behaviour of ladder methods to compute parallel transport. We first gave a Taylor expansion of one step of Schild's Ladder. Then, we showed that when scaling the vector to transport by $\frac{1}{n^2}$, a quadratic speed of convergence is reached. The same results were deduced for the pole ladder, and as less geodesics are required for the iterative construction, the overall computational cost is reduced and our numerical experiments illustrate that these bounds are indeed reached. In the same framework, we bridged the gap between the Fanning Scheme and Schild's Ladder, shedding light on how SL could be turned into a second-order method while the FS cannot.

For manifolds with no closed-form geodesics, we introduced the infinitesimal ladder schemes and showed that the PL converges with the same order as its counterpart with exact exp and log maps. The same exercise can be realized with SL.

Numerical experiments were performed on $SE(3)$ with an anisotropic metric, and allowed to observe the role of ∇R in the convergence of the PL in a non-symmetric space. This result corroborates our theoretical developments, and shows that the bounds on the speed of convergence are reached.

Our last comparison of SL and PL shows that, although more popular, SL is far less appealing that the PL as (i) it is more expensive to compute, (ii) it converges slower, (iii) it is less stable when using approximate geodesics and (iv) it is not exact in symmetric spaces. Pole ladder is also more appealing than the FS because of its quadratic speed of convergence, which allows to reach mild convergence tolerance at a much lower overall cost despite the higher complexity of each step.

The ladder methods are restricted to transporting along geodesics, but this not a major drawback as this is common to other methods, and one may approximate any curve by a piecewise geodesic curve.

In our work on infinitesimal schemes, we only used basic integration schemes of ODEs in charts, while there is a wide literature on numerical methods on manifolds. In future work, it would be very interesting to consider specifically adapted RK schemes on Lie groups and homogeneous spaces (Munthe-Kaas and Verdier 2016; Munthe-Kaas and Zanna 1997), or symplectic integrators (Hairer, Wanner, et al. 2002; Dedieu and Nowicki 2005) in order to reduce the computational burden and to improve the stability of the scheme. More precisely, the computations of the PL may be hindered by the approximation of the log, while in fact, only a geodesic symmetry $y \mapsto \mathrm{Exp}_x(-\mathrm{Log}_x(y))$ is necessary and may be computed more accurately with a symmetric scheme.

Chapter 5

Cardiac motion modelling with parallel transport

In this chapter, we apply the tools discussed in the previous chapters to the study of the motion of the cardiac right ventricle under pressure or volume overload. The difficulty of this task lies in the interactions between shape and deformation. The central idea of this work is to filter out these interactions using the parallel transport of deformations to a reference shape, where deformations are considered in the Large Deformations Diffeomorphic Metric Mapping (LDDMM) framework. It appears that parallel transport alone is not sufficient to normalize deformations when large volume differences occur. We thus propose a normalization procedure for the amplitude of the deformation, and compare it with volume-preserving metrics. After normalization, we compare a geodesic regression and a second-order spline model to represent the full cardiac contraction. The statistical analysis of the parameters of the model reveal insights into the dynamics of each disease. The method is applied to 3D meshes of the right ventricle extracted from echocardiographic sequences of 314 patients divided into three disease categories and a control group. This chapter is an extension of the work that was presented at the ISBI 2021 conference in Guigui, Moceri, et al. 2021. The comparison with volume preserving metrics and the experiments on simulated data are new and not published.

Contents

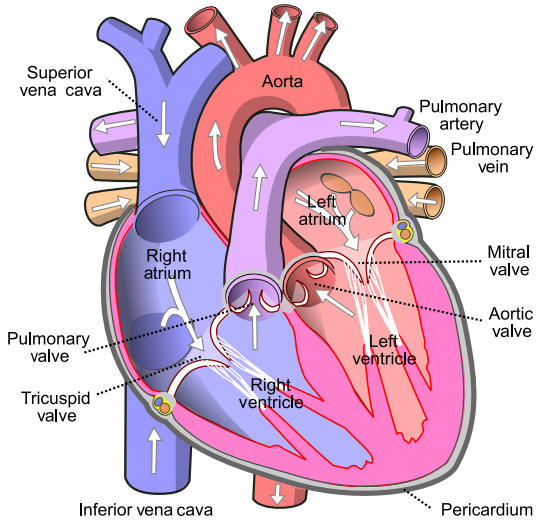


Figure 5.1: Diagram of the human heart reproduced from Wikipedia.

5.1 Introduction to Cardiac Modelling

Spatio-temporal shape analysis is of growing importance in the study of cardiac diseases. In particular, the assessment of cardiac function requires the measurement and analysis of cardiac motion beyond scalar indicators such as volume, ejection fraction, pressure-volume loops or area strain. We focus on the case of the right ventricle (RV), that is of particular interest as it has been shown to have a large capacity to adapt to overload by remodelling (Sanz, Sánchez-Quintana, et al. 2019), raising the issue of disentangling the deformation from the initial anatomy.

5.1.1 The Right Ventricle

The right ventricle is one of the two discharging chambers of the heart, receiving blood from the right-atrium through the tricuspid valve and ejecting blood into the main pulmonary artery (see Figure 5.1). Although neglected in the assessment of normal blood circulation of adult patients for decades, its relevance to determine symptoms and outcomes is now well established in many pathologies such as congenital heart diseases (CHD) and pulmonary hypertension (PHT). Three such pathologies are studied in this manuscript, PHT, Atrial Septal Defect (ASD) and Tetralogy of Fallot (ToF).

Pulmonary hypertension is a severe condition characterized by increased blood pressure in the pulmonary arteries, appearing between the ages of thirty and forty for about 4 new cases per million[1]. It is associated with symptoms and premature deaths (Moceri, Duchateau, Baudouy, et al. 2018) and the most common source of pressure overload in the right-ventricle, resulting in hypertrophy, flattening of

[1] https://en.wikipedia.org/wiki/Pulmonary_hypertension#Epidemiology

the interventricular septum, progressive dilation and dysfunction (Sanz, Sánchez-Quintana, et al. 2019).

Atrial septal defect is a CHD (i.e. a heart abnormality present at birth) in which blood flows between the left and right atrium. This may lead to lower oxygen levels in the blood that supplies the other organs, resulting in cyanosis. It affects the RV with volume overload, marked by dilation and hypertrophy and leftward septal shift, but it may be tolerated for years, as many studies report no affectation of the function of the RV under volume overload (Moceri, Duchateau, Gillon, et al. 2020; Sanz, Sánchez-Quintana, et al. 2019).

Tetralogy of Fallot is another CHD characterized by pulmonary stenosis (i.e. narrowing of the pulmonary valve) and an overriding aorta, allowing blood flows between the left and right ventricles, and resulting in hypertrophy of the RV, reduced oxygen level and cyanosis. This is the unique case where the RV is under both volume and pressure overload (Sanz, Sánchez-Quintana, et al. 2019). Untreated, ToF patients rarely survive to adulthood (Chessa and Giamberti 2012). It is treated by surgical repair during the first year of life, greatly improving survival, but post-surgery defects such as pulmonary regurgitation may occur, requiring other operations [2].

We now briefly review some indicators of cardiac function relevant to the right-ventricle that will be used in this chapter.

The ejection fraction (EF) is the relative volume change between end-diastolic (ED) time and end-systolic (ES) time, i.e. during one contraction:

$$EF = \frac{\text{Vol}(ED) - \text{Vol}(ES)}{\text{Vol}(ED)} \tag{5.1}$$

It represents the volume of blood pumped by the heart to the body circulation, and is used in clinical routines to evaluate heart failures. The measure is extracted from imaging modalities such as 2d or 3d echocardiography and magnetic resonance imaging (MRI). Values for the right-ventricle range around $55 \pm 6\%$ ((Moceri, Duchateau, Baudouy, et al. 2018)).

The area strain (AS) is the relative area change of each cell of the mesh between ED time and ES time. As it depends on the quality of the mesh, it is usually filtered by computing the mean at each vertex over the neighboring cells, and visualized as a colormap on the mesh, or averaged by regions of the ventricle. It represents the amount of stretching local tissues undergo during the deformation, and is a common descriptor of cardiac motion (Kleijn, Aly, et al. 2011; Di Folco, Clarysse, et al. 2019; Moceri, Duchateau, Gillon, et al. 2020).

[2] https://en.wikipedia.org/wiki/Tetralogy_of_Fallot#Prognosis

5.1.2 Computational Anatomy

This chapter builds on the framework of computational anatomy that aims at studying temporal evolution and cross- sectional variability of anatomical shapes through the action of a certain group of transformations of the space in which the shapes reside. This setting is different from the point of view of Kendall shape spaces (Section 3.3.3) in which shapes are defined by their invariance properties, and a metric is defined directly on the quotient space, identifying two shapes if one can be reached by a similarity transformation of the other. On the other hand, computational anatomy builds on the idea of D'Arcy Thompson from 1917 that the distance between two shapes is captured by the quantity of deformation needed to reach one from the other, and puts deformations as the quantity of interest instead of noise that must be filtered out (Durrleman 2010). This framework is compatible with the study of different shape modalities that all describe organs and their movements: images, landmark sets, currents, surfaces, etc.

Various frameworks have been proposed to model the relevant set of deformations in computational anatomy. A list of references is given in Pennec, Sommer, et al. 2020, Section 4.8. In this work we focus on the Large Deformations Diffeomorphic metric mapping (LDDMM) framework. An introduction to the mathematical framework is given in section 5.2, for a more thorough discussion of the related framework, the reader is referred to Younes 2019 and M. I. Miller, Trouvé, et al. 2015.

Longitudinal models are a key tool to study time-series of shapes in both disease progression modelling and in the study of dynamical systems such as the heart. In computational anatomy, two paradigms that rely on parallel transport have been proposed. The first, proposed by Schiratti, Allassonnière, et al. 2015, advocates to compute a reference trajectory, to register the reference time point of a given trajectory on the reference time point of the reference trajectory, and to transport this deformation along the reference trajectory. This corresponds to the assumption that all the trajectories are within a small set of perturbations of the reference trajectory that are parallel along the trajectory. Reconstructions may be obtained by shooting with the transported perturbation at all times of the reference trajectory. This procedure is called exp-parallelization.

This model is too restrictive at first sight, as different trajectories may be present in the same dataset. A recent refinement of this model has been proposed, that allows to cluster trajectories into distinct classes that each follow a different trajectory (Debavelaere, Bône, et al. 2019). Although interesting, this model does not serve the same purpose, as it allows to infer representative trajectories for each disease group, but not directly to compare them as deformations of the same initial anatomy.

We therefore focus on the second approach proposed by Lorenzi and Pennec 2011, which consists in first estimating a patient-specific trajectory, and to normalize it with respect to the patient-to-atlas deformation at a reference time.

The need to normalize deformations is salient in the example of Figure 5.2, where the deformation between the patients' ED and ES frames are applied without any normalization to another reference shape, the atlas. These result in meshes of poor quality, with irregularities, especially near the valves and on the septum. These abnormalities are salient when considering ventricles with pressure and volume overload as the amplitude of both the intra-patient and patient-to-atlas deformations may be large. This motivates the need to normalize the deformations in order to perform population statistics of deformations. In this thesis, we focused on parallel transport as a tool to reorient the deformations (Section 5.2.2) and investigated different normalization strategies for the amplitude of the intra-patient deformation (Section 5.3).

Related work The problem of comparing cardiac deformations across patients has been partially addressed in previous work. Young and Frangi 2009 use the Point Distribution Model (PDM) framework, which amounts to performing principal component analysis of the nodes' coordinates in the Euclidean 3d space. This framework fails to preserve the organ's structure as soon as large deformations are encountered and thus does not allow to extrapolate deformations. Peyrat, Delingette, et al. 2010 use trajectory constraints in the Diffeomorphic Demons algorithm. The constraints are given by another signal such as electrocardiograph recordings, and they are used in a registration pipeline that locates each frame with respect to a reference sequence by composing the intra-patient deformations of a patient k, $\phi^k_{t,0}$ and the reference trajectory $\phi^{ref}_{0,t}$, with the patient-to-reference deformation $\phi_0^{k\to ref}$ at the reference time: $\phi_t^{k\to ref} = \phi^{ref}_{0,t} \circ \phi_0^{k\to ref} \circ \phi^k_{t,0}$. In this framework, a reference trajectory must be given. Such trajectory, referred to as *statistical atlas*, is computed for example in Durrleman, Pennec, et al. 2009 where a similar chaining of intra-patient and patient-to-atlas deformations are used to compute the likelihood of proposed reference trajectories. The procedure is however only valid for small deformations. Duchateau, De Craene, et al. 2011 propose a similar approach but use velocities between frames computed with local differences and normalized by the push-forward action by the composed transformation $\phi^{ref}_{0,t} \circ \phi_0^{k\to ref} \circ \phi^k_{t,0}$.

Qiu, Younes, et al. 2008; Qiu, Albert, et al. 2009 propose the first implementation of parallel transport by using Jacobi fields (Younes 2007) and compare the evolution of shapes along different sequences. This method is used in Cury, Lorenzi, et al. 2016 for the analysis of the thalamus in frontotemporal dementia. The method is improved by the Fanning Scheme Louis, Charlier, et al. 2018, which is applied to shape analysis of brain structures Louis, Bône, et al. 2017. In those works, the authors claim that the Jacobi Field approach is more precise and less expensive computationally than ladder methods. We have proved that this is not the case in chapter 4.

Ladder methods have been proposed in shape analysis in Lorenzi, Ayache, et al. 2011 with the Stationary Velocity Field (SVF) framework. It was then improved into the pole ladder in Lorenzi and Pennec 2014 and applied in Lorenzi, Pennec, et

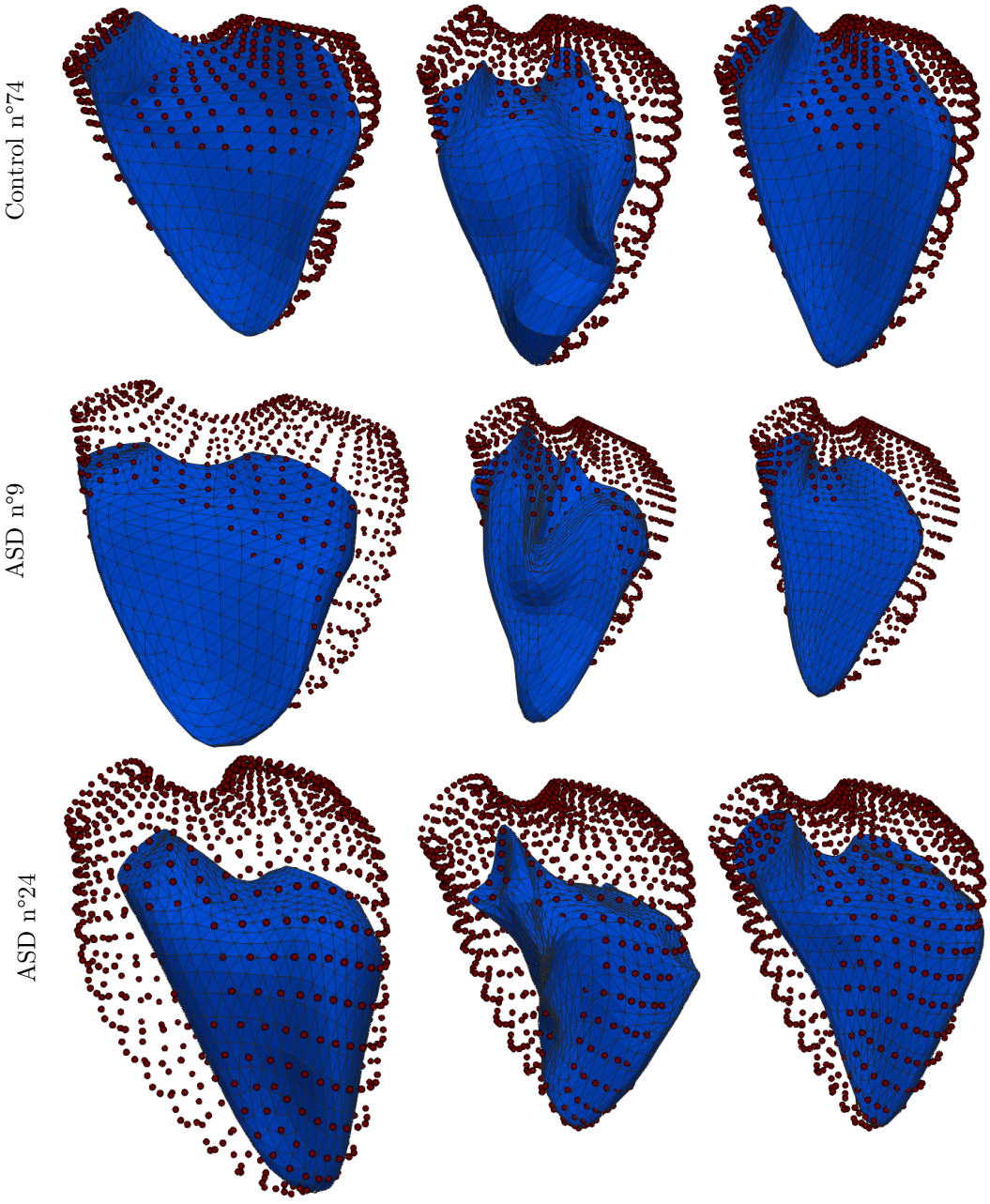


Figure 5.2: Example of a systolic deformation estimated on a patient and applied to the atlas without normalization (middle) and with normalization by the method of Section 5.3.1 (right). The ED frame is a red point cloud, while the ES frame is the blue mesh. The ES frames obtained without normalization show irregularities and cannot be used in downstream analyses.

al. 2015 to distinguish between normal aging and pathological evolution. In the SVF framework, the Baker-Campbell-Hausdorff allows to approximate the neighboring log in the space of velocity fields, which reduces the computational cost of the method. An efficient computational pipeline is provided by Hadj-Hamou, Lorenzi, et al. 2016 and used in Sivera, Delingette, et al. 2019; Sivera, Capet, et al. 2020. To the best of our knowledge, ladder methods have not been used in the LDDMM framework, for which this approximation is not available.

Furthermore, previous applications of ladder methods in medical imaging have restricted to the study of the evolution of the brain, whose amplitudes are much lower than that encountered in the cardiac cycle. This raises methodological questions that will be addressed in this chapter, in particulat in section 5.3.

5.2 LDDMM

5.2.1 Self-contained exposition of the LDDMM framework

The LDDMM framework encompasses both algorithms for shape matching (also referred to as registration) and a Riemannian geometric structure on the space of deformations, that projects to the space of shapes. The former allows computing shape descriptors, and to parameterize diffeomorphisms. The latter, provides a distance to compare deformations, and an associated notion of parallelism to transport them. We present here the formulation of Durrleman, Allassonnière, et al. 2013; Durrleman, Prastawa, et al. 2014 and their implementation in Bône, Louis, et al. 2018, focusing on the case of landmarks. For a thorough treatment of the topic, we refer the interested reader to Younes 2019.

In this chapter, we restrict to shapes defined as $d-1$-dimensional submanifolds of $\mathbb{R}^d$, $d = 1, 2$, with the same given topology, and approximated by triangulated meshes, defined by a set of points in $\mathbb{R}^d$, called landmarks, and connectivity between those. The set of shapes is denoted $\mathcal{S}$. The model of computational anatomy stipulates that such a shape is fully described by a diffeomorphism $\varphi \in \text{Diff}(\mathbb{R}^d)$, where the interaction between shapes and diffeomorphisms is encoded by the action of $\text{Diff}(\mathbb{R}^d)$ on $\mathcal{S}$. This turns the shape space into a homogeneous space with group $\text{Diff}(\mathbb{R}^d)$. The isotropy subgroup of a given shape is the set of re-parameterization of this shape, i.e. diffeomorphims that leave the shape unchanged, but move particles along the shape. These diffeomorphisms are infinite dimensional nuisance parameters in a statistical analysis, and one feature of LDDMM is to provide methods that are invariant to such nuisance (M. I. Miller, Trouvé, et al. 2015).

A natural and efficient computational construction of diffeomorphisms is obtained by flows associated to ordinary differential equations (ODEs) $\partial\phi_t(\cdot) = v_t[\phi_t(\cdot)]$, with the initial condition $\phi_0 = Id$. The time-dependent vector field v_t can be interpreted as the instantaneous speed of the points during deformation, and must verify certain regularity conditions to ensure that solutions to the ODE are indeed diffeomorphisms. An efficient way to enforce these conditions is to restrict to vector fields obtained by the convolution of a number N_c of momentum vectors

carried by distinct control points:

$$v_t(x) = \sum_{k=1}^{N_c} K(x, c_k^{(t)})\mu_K^{(t)}, \tag{5.2}$$

where K is a kernel function, e.g. the Gaussian kernel: $K(x,y) = \exp\big(-\frac{\|x-y\|^2}{\sigma^2}\big)$. The effect of the choice of kernel is studied in Micheli and Glaunès 2014, but we restrict to the Gaussian kernel in this study. The (closure of the) set of such vector fields forms a reproducing kernel Hilbert space H_K, with the associated norm

$$\|v\|_K^2 = \sum_{i,j} K(c_i, c_j)\mu_i^T \mu_j.$$

We write Diff_K the subgroup of diffeomorphisms obtained this way and use matrix notation with a $dN_c \times dN_c$ block matrix $\mathbf{K}(c,c') = \big(K(c_i, c'_j)I_d\big)_{ij}$ and flat vectors of size dN_c for landmarks sets, velocities and momentum, so that $v(x) = \mathbf{K}(c,x)\mu$ and $\|v\|_K^2 = \mu^\top \mathbf{K}(c,c)\mu$. The total cost, or energy of the deformation can be defined as $\int_0^1 \|v_t\|_K^2 dt$.

It can be shown that the momentum vectors that minimize this energy, considering $c_k^{(0)}, c_k^{(1)}$, $k = 1 \dots N_c$ fixed, together with the equation driving the motion of the control points, follow a Hamiltonian system of ODEs:

$$\begin{cases} \dot{c}^{(t)} &= \mathbf{K}(c^{(t)}, c^{(t)})\mu^{(t)} \\ \dot{\mu}^{(t)} &= -\nabla_1 \mathbf{K}(c^{(t)}, c^{(t)})\mu^{(t)^T}\mu^{(t)} \end{cases} \tag{5.3}$$

A diffeomorphism ϕ_1 is thus uniquely parameterized by the initial conditions $c_k^{(0)}, \mu_k^{(0)}$, $k = 1 \dots N_c$, and a shape registration criterion between a template $\bar{q}$ and target q can be defined as

$$C(c,\mu) = \|q - \phi_1^{c,\mu}(\bar{q})\|_2^2 + \alpha^2 \|v_0^{c,\mu}\|_K^2. \tag{5.4}$$

where α is a regularization parameter that penalises large deformations. Minimizing C therefore amounts to finding the transformation that best deforms $\bar{q}$ to match q. The gradient of C can be computed through automatic differentiation, to perform gradient descent. Note that in (5.4), the L_2 norm between landmark positions is used to evaluate the data-attachment term. When corresponding landmarks are not available, fidelity metrics relying on currents, varifolds or normal cycles have been derived and allow to compute metrics between curves or surfaces with different parametrizations. These are reviewed in Charon, Charlier, et al. 2020.

Two examples of registration solutions are shown, the first is on simple parametric shapes, a circle and an ellipse (Figure 5.3). In this case the full trajectory of each landmark is represented (with a color scheme from blue to red). These differ from a linear interpolation especially where the curvature of the shape increases. The second example is on real RV data (Figure 5.4). The distribution of momentum vectors on the shape is not intuitive, and it is easier to inspect the velocity field evaluated at the control points. The matching is very efficient on these meshes, thanks to

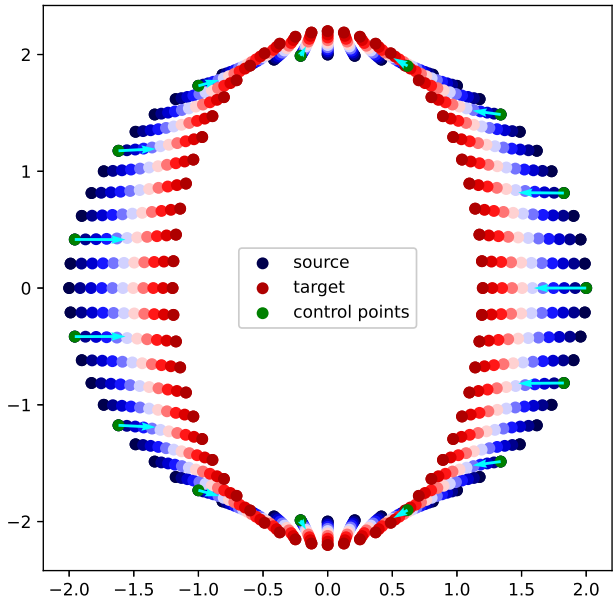


Figure 5.3: Example of registration and optimal path between a circle (source) and an ellipse (target) with 60 landmarks, 15 control points (green) and corresponding momemtum vectors (cyan) and kernel width 1. The progression from blue to red shows the time trajectories of the landmarks under the minimizing geodesic deformation.

the quality of the correspondences between landmarks and to the smoothness of the shapes.

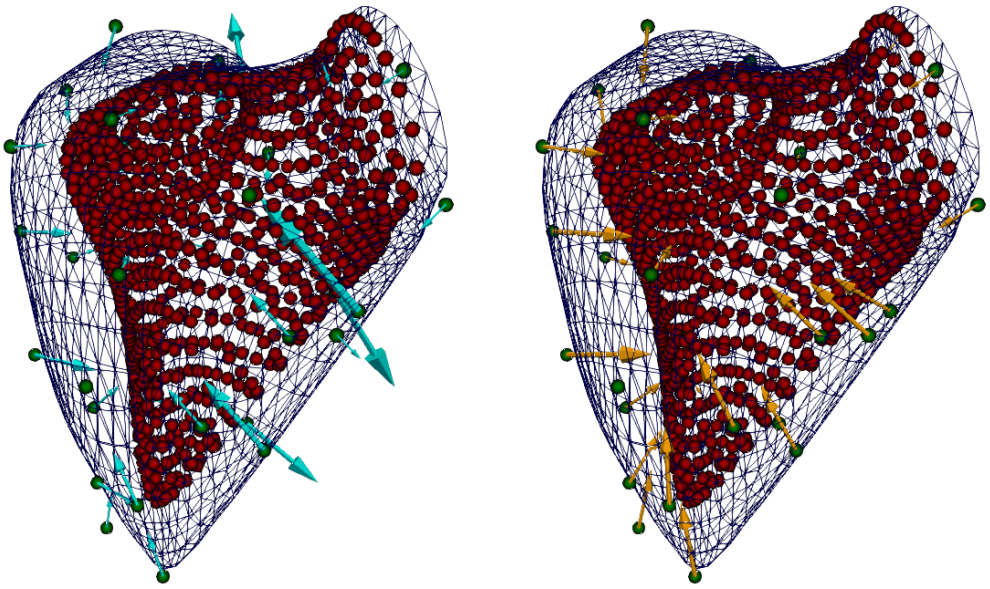
The optimal value of C defines a distance between ϕ_1 and the identity. In fact this distance derives from a right-invariant Riemannian metric on the group Diff_K and the path $t \mapsto \phi_t$ is a minimizing geodesic for this metric. Integrating the system (5.3) computes the exponential map of this metric, and is often called *shooting* in this section, while minimizing (5.4) approximates the logarithm, and is called *registering*. By considering the action of diffeomorphisms on shapes, this distance projects to a distance between the shapes q and $\bar{q}$, and we have

$$d_K(q,\bar{q})^2 = \inf\left\{\frac{1}{2}\int_0^1 \dot{q}_t \mathbf{K}(c,c)^{-1}\dot{q}_t dt \quad | \quad q_0 = \bar{q}, q_1 = q\right\}$$

This in fact turns the space of shapes into a homogeneous space, with invariant metric given by the group, as in Section 3.4 of chapter 3 (page 105).

5.2.2 Ladder methods with the LDDMM framework

Along with a distance, the Riemannian metric provides a notion of parallel transport by its Levi-Civita connection (Subsection 2.3.2 page 35). A Hamiltonian formulation of parallel transport may be given by an ODE that allows to transport a set of momentum vectors along a path of diffeomorphisms ϕ_t (Younes 2019, section 13.3.3).



(a) Systolic deformation with momenta (b) Systolic deformation with velocity field

Figure 5.4: Example of the registration between ED (blue wireframe) and ES (red points) for a control right-ventricle. There are 938 landmarks, 34 control points with kernel width 10mm (the overall ventricle at ED is about 55mm high). Note how momentum vectors are noisy due to strong interactions between control points (left). The velocity field at the control points offers a better summary of the deformation (right).

Previous implementations of parallel transport of deformations in the context of LDDMM are however based on Jacobi fields (Younes 2007; Cury, Lorenzi, et al. 2016), and in particular on the *fanning scheme* (Louis, Charlier, et al. 2018).

Ladders method in contrast, were proposed in the framework of stationary velocity fields (SVF), where the vector fields generating diffeomorphims is the subset of constant vector fields . This results in a Lie group structure, where the registration problem amounts to computing one-parameter subgroups of the considered group (Lorenzi and Pennec 2011), allowing to formulate ladder methods with the BCH formula instead of the double exponential, and bringing algorithmic simplifications.

We propose a new implementation of the pole ladder in the context of LDDMM, that leverages the lower dimensional parametrization and the Hamiltonian formulation, and is grounded by the results of Chapter 4. Indeed, the geodesics are not known in closed-form and we are in the case of infinitesimal ladder schemes, requiring the use of fourth-order integration schemes (Section 4.3.4 page 133). Our implementation is largely inspired by that of the fanning scheme and leverages the open-source software Deformetrica (Bône, Louis, et al. 2018).

Throughout this chapter, we consider a reference shape q_A, and wish to transport

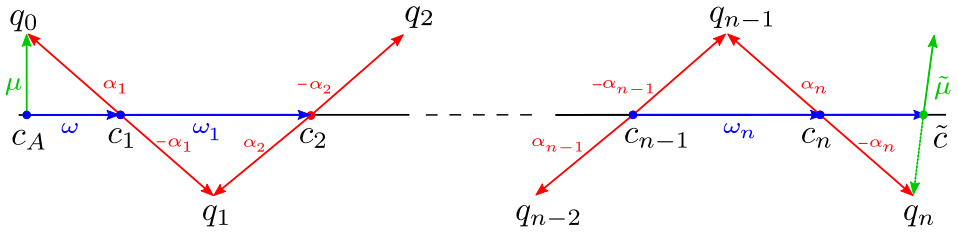


Figure 5.5: Representation of the pole ladder for diffeomorphisms with an odd number of steps n. The exponential maps are computed with a RK4 scheme, and the log by gradient descent.

the deformation (c, μ) obtained by registration of a shape q_1 on another shape q_0 to q_A, that is, along the geodesic between q_0 and q_A, parametrized by (c_A, ω) and also obtained by registration. We suppose that registration outputs a set of control points such that $\mathbf{K}(c, c)$ is a symmetric positive definite matrix.

Recall from the homogeneous space structure that the tangent space a q_0 is identified with the horizontal subspace of the Lie algebra of Diff_K, which is here the reproducing Kernel Hilbert space H_K that contains all vector fields defined by the convolution (5.2). It defines a one-to-one correspondence between the tangent space at q_0, and its co-tangent space, meaning that we can transport the momentum μ and flow the points c instead of transporting a full vector field v. Similarly, we don't need to flow the full shape q_0 along the ladder, but only the control points.

We describe the pole ladder in the case where the momentum to transport μ and the momentum along which we transport ω are carried by the same initial control points $c = c_A$. If it is not the case, μ is projected to c_A by solving a linear system with a Cholesky decomposition of $\mathbf{K}(c_A, c_A)$: $\mu' = \mathbf{K}(c_A, c_A)^{-1}\mathbf{K}(c, c_A)\mu$.

As in Paragraph 4.3.4.1, define $\mathrm{rk} : T\mathcal{S}^* \times \mathbb{R}_+ \to T\mathcal{S}^*$ the map that performs one step of the fourth-order Runge-Kutta numerical scheme on the Hamiltonian system (5.3), and rk_1 the projection on the control points only, and rk_x^{-1} its approximate inverse by gradient descent, such that $\|\mathrm{rk}_1(\mathrm{rk}_x^{-1}(y), h) - y\| \leq h^5$. Then, pole ladder is performed by the following algorithm.

Choose $n \in \mathbb{N}$ and $1 \leq \alpha \leq 2$, and divide $[0, 1]$ into n intervals of length $h = \frac{1}{n}$. Compute the first midpoint $c_1, \omega_1 = \mathrm{rk}(c_A, \omega, \frac{h}{2})$ along the main geodesic, and the first rung $q_0 = \mathrm{rk}_1(\mu, \frac{1}{n^\alpha})$. Then iterate for $i = 1 \dots n$ (see Figure 5.5)

1. Compute the momentum $\alpha_i = n\,\mathrm{rk}_{c_i}^{-1}(q_{i-1})$;
2. Compute the flow of its inverse $q_i = \mathrm{rk}_1(c_i, -\alpha_i, h)$;
3. Compute the next midpoint $c_{i+1}, \omega_{i+1} = \mathrm{rk}(c_i, \omega_i, h)$ except at the last step where $h \leftarrow \frac{h}{2}$ to compute $\tilde{c}$;

Return $\tilde{\mu} = n^\alpha(-1)^n\,\mathrm{rk}_{\tilde{c}}^{-1}(q_n)$.

Two examples of parallel transport solutions are shown, the first on simple parametric shapes, the evolution from a circle to an ellipse is transported to a smaller circle (Figure 5.6). We retrieve a smaller ellipse as expected. The second example on

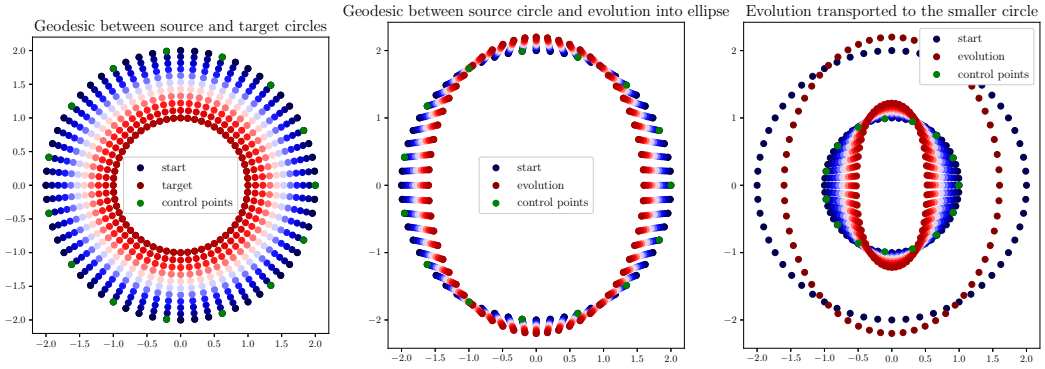
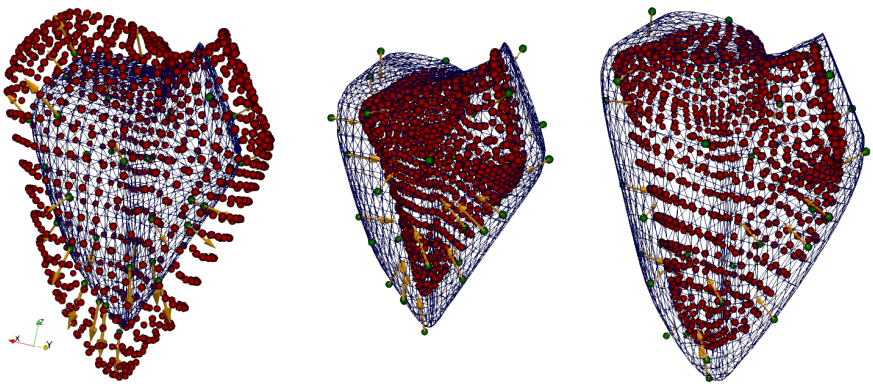


Figure 5.6: Parallel transport of the evolution from a circle to an ellipse along the deformation to a smaller circle.

real RV data (Figure 5.7), shows the ED-to-ES deformation transported to a third reference shape, the atlas. The obtained reconstruction is a personalized estimate of an ES frame for the atlas, specific to the considered patient. The transported frame is much more acceptable than those of Figure 5.2.



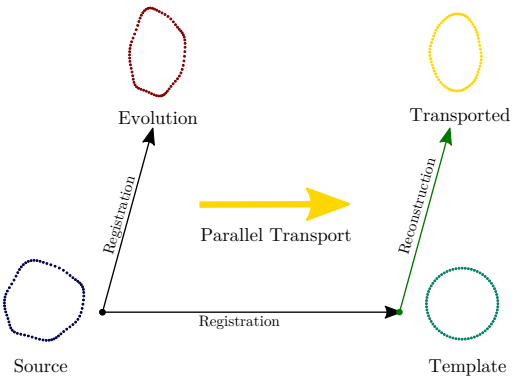
ED to Atlas deformation ED to ES deformation Transported deformation

Figure 5.7: Example ED-to-ES deformation transported along the ED-to-atlas path.The source frames are represented by a blue wireframe, the final frames by red landmarks, the control points by green landmarks and the initial velocities at the control points by orange arrows. The transported frame is much more acceptable than those of Figure 5.2.

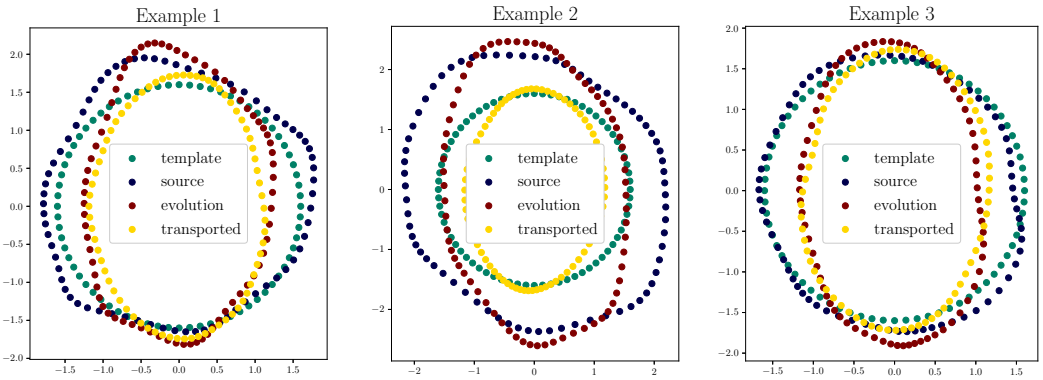
Validation Recall that parallel transport is an isometry, so the norm of the transported deformation must equal that of the initial deformation. We use this property as a first step to validate our implementation on a population of simulated 2d-shapes. These are generated by shooting from a circle with random Gaussian momentum vectors. This defines the source shape. Then an affinity is applied to the source shape to define its evolution (i.e. scaling with different coeficients along each axis).

This deformation is then estimated by registration and transported to the template. Some examples are displayed on Figure 5.8. The reconstruction obtained by deforming the circle with the transported deformation (yellow shape on the figure) resembles an ellipse, as expected by the application of the affinity.

For $n = 10$ rungs, we obtain a root mean squared error (RMSE) of 8.3×10^{-3} and less than $1.3 \pm 0.9\%$ relative error (in absolute value) for 100 samples. We also apply the affinity to the template and register the template to the result to compute the expected momentum after transport. The deviation of the transported momentum to this expected value is measured with the kernel norm of the difference and averaged over the samples. We obtain an error of 0.34 while the fanning scheme achieves 0.33. The two implementations are therefore very similar on this set of shapes. We however find cases where the fanning scheme does not behave as well as the pole ladder for more complex deformations on the heart data. This is shown in the next subsection.



(a) Experimental design



(b) Examples of deformations used to validate our implementation.

Figure 5.8: We can check visually that the affinity used as evolution (which maps a circle to an ellipsis) is retrieved by the parallel transport (between template and transported).

5.2.3 Application to cardiac motion

Dataset We use 3D meshes extracted from 314 echocardiographic sequences from patients examined at the CHU of Nice (study protocol ID-RCB:2017-A02077-46). More details about the cohorts are given in Moceri, Duchateau, Baudouy, et al. 2018; Moceri, Duchateau, Gillon, et al. 2020. The meshes were extracted with a commercial software (4D RV Function 2.0, TomTec Imaging Systems, GmbH, DE) with point-to-point correspondences across time and patients. These are formed of 938 points and 1872 triangles. All the shapes were realigned with a subject-specific rigid-body deformation. An atlas was computed from the end diastolic meshes of the control group, after rigid-body alignment. We warmly thank Pamela Moceri and Nicolas Duchateau for collecting and curating the data, and for the helpful interactions that we had during this project.

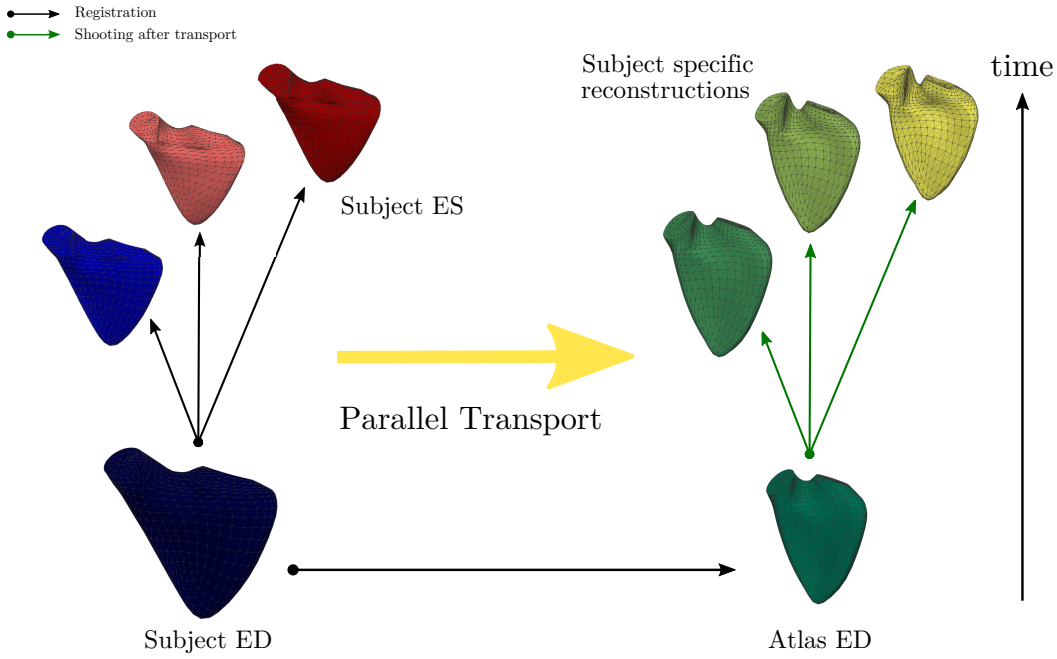


Figure 5.9: Framework using registration and parallel transport to normalize cardiac deformations.

We now use parallel transport to study the motion of the right ventricle during the systolic phase of the cardiac cycle. This phase is observed at a discrete set of sampling times between ED and ES, and we treat each observation independently for its reorientation by parallel transport. The framework is summarized in Figure 5.9. For each observation time t_i, the subject's frame at time t_i is registered to this subject's ED frame, transported along the path between ED and the atlas, and reconstructed by shooting with the transported momenta. We reproduce on Figure 5.10 the reconstructions of the ES frame for a patient of each disease group and of the control group. These are obtained by the pole ladder algorithm (right) as well as the fanning scheme (middle). On these examples the two methods

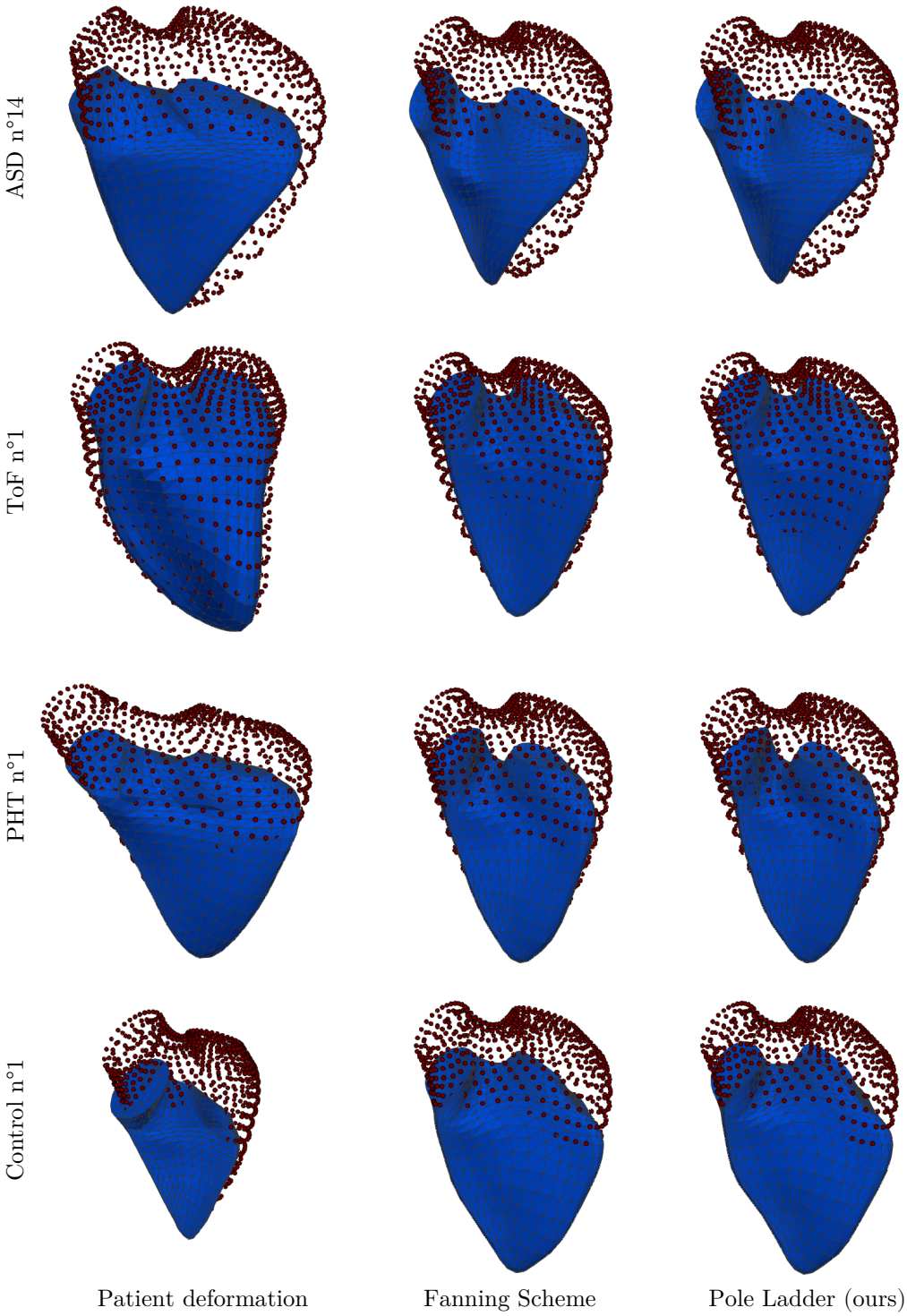


Figure 5.10: Examples of reconstructions of the ES frame (blue meshes) after parallel transport of the ED to ES deformation (point cloud to mesh) along the deformation from ED to atlas. Parallel transport is computed by the fanning scheme (middle) and our implementation of pole ladder (right). In most cases, little difference can be observed visually.

produce very similar results, which validates our implementation. Moreover, the obtained reconstructions are smooth (compare with Figure 5.2) and seem to adapt the deformation pattern of each patient to the atlas.

In most cases, little difference can be observed qualitatively. We find however a few examples where the fanning scheme did not achieve to transport the deformation, and some points diverge or collapse in the reconstructions. This occurs when the original shapes are significantly larger than the atlas, and with strong spherification of the apex, causing large deformations as for the ASD patient of Figure 5.11.

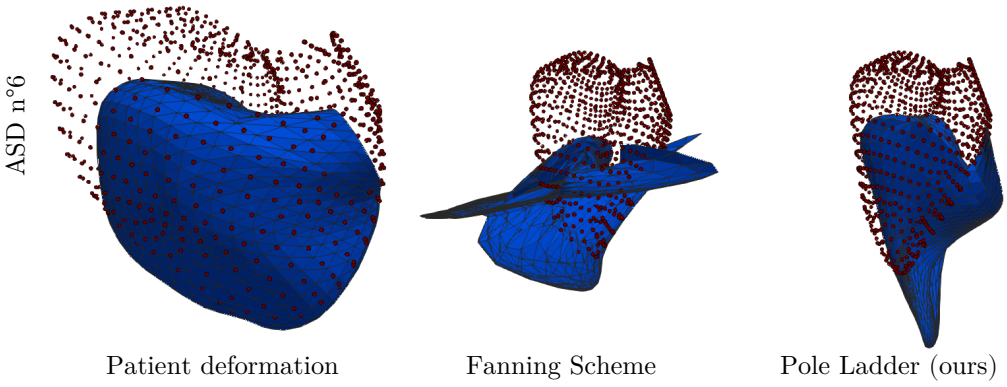
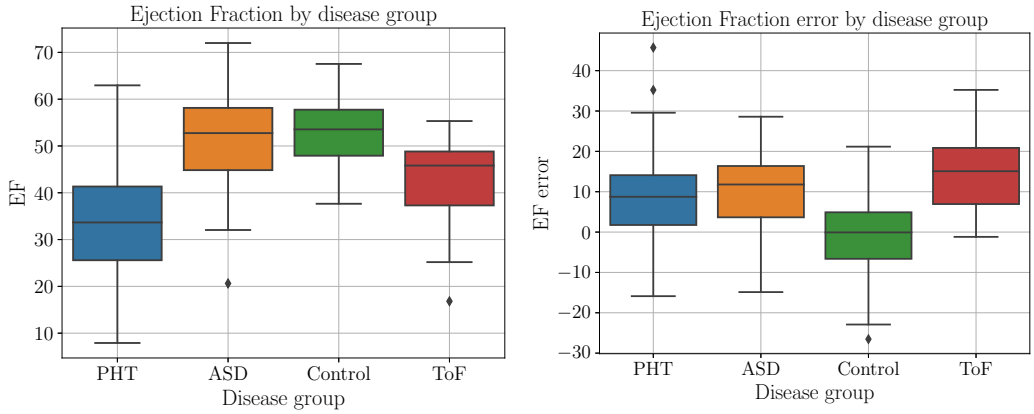


Figure 5.11: Examples of reconstruction where the fanning scheme did not achieve to transport the deformation, while pole ladder did, although it is not realistic.



(a) EF per disease group.

(b) Alteration of EF by parallel transport.

Scale problem Niethammer and Vialard 2013 showed that parallel transport in LDDMM does not conserve global properties such as scale or volume changes. This is visible in our previous experiment (Figure 5.6), where the anisotropy of the original evolution (from circle to ellipse) is 1.4 whereas the one obtained after parallel transport is 2.3. In the case of cardiac deformations, the magnitude of the temporal deformation is comparable to that of the subject's reference to atlas deformation,

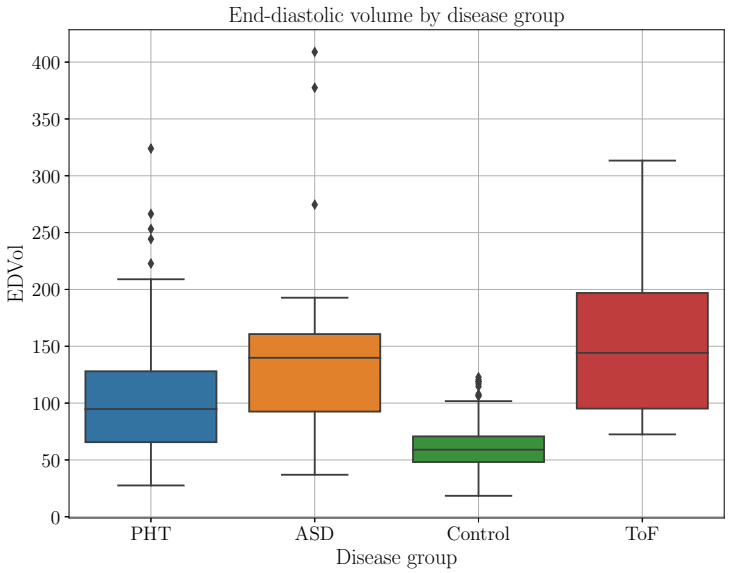


Figure 5.13: End-diastolic volume by disease group

and substantial volume changes are observed. Thus the lack of scale-invariance is crucial.

Furthermore, there is a significant correlation between the magnitude of the systolic deformation and the ED volume ($\rho = 0.42$ in the dataset considered in this thesis). Additionally, there are significant differences of ED volumes by disease groups (Figure 5.13). As the parallel transport is isometric, this deformation may be too large for the atlas. Examples of the obtained ES frames are shown on Figure 5.14 for patients whose RV volume is greater than that of the atlas, resulting in non-realistic ES frames.

To quantify this phenomenon, we measure the EF and the AS before and after transport and report the RMSE in Table 5.1. We observe a large relative RMSE EF, representing 31% of the mean value, and equal to the standard deviation of the data. Furthermore, plotting the absolute value of the EF error by disease group, we retrieve a strong correlation between the error and the EF (Figure 5.12b). This effect is quite undesirable for a normalization procedure!

	Original Values	RMSE PT
AS	-0.24 ± 0.08	0.18
EF	0.42 ± 0.13	0.13

Table 5.1: Validation of parallel transport with EF and AS.

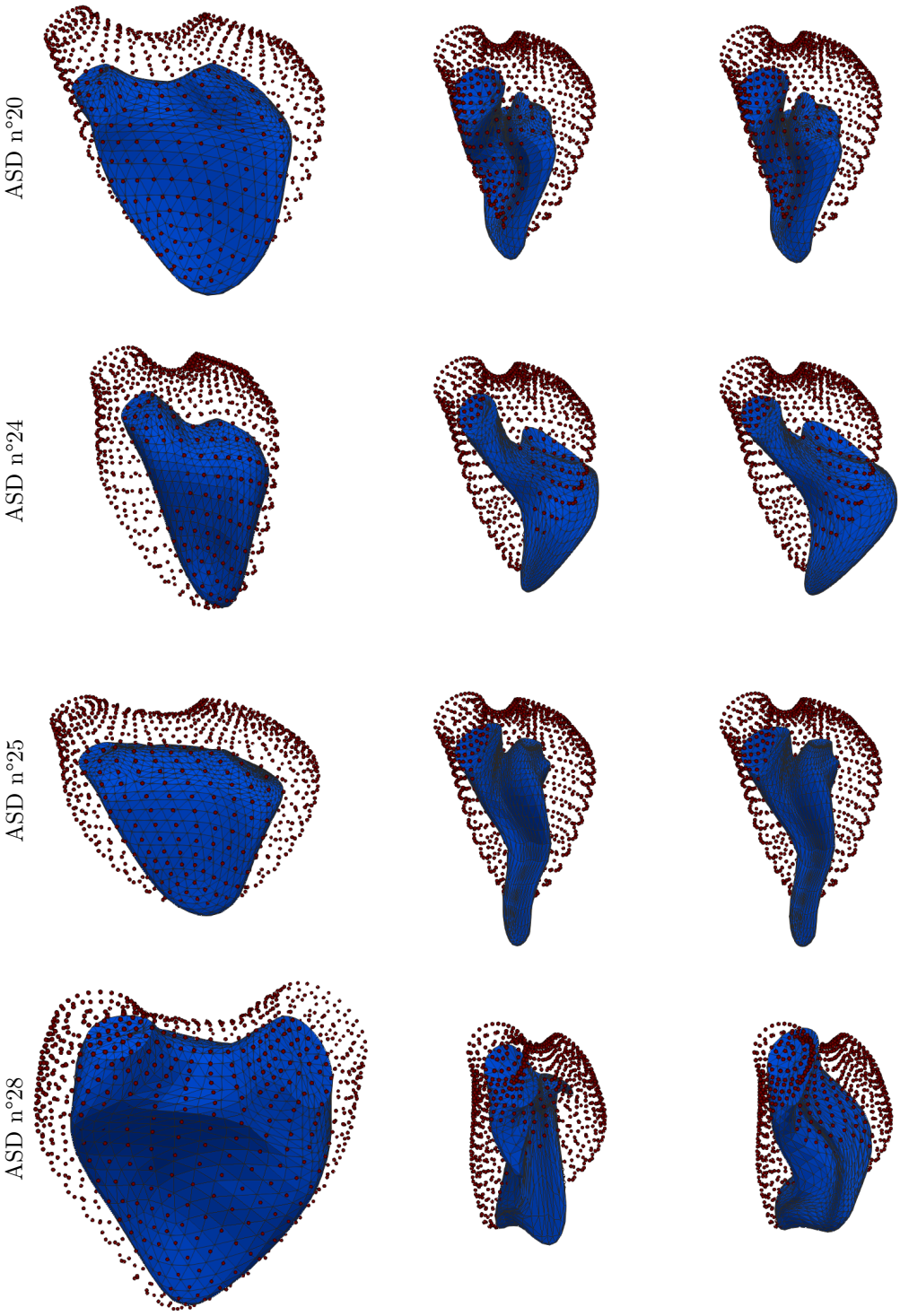


Figure 5.14: Examples of reconstructions of the ES frame (blue meshes) after parallel transport of the ED to ES deformation (point cloud to mesh) along the deformation from ED to atlas. The subjects belong to the ASD group, hence show large volume changes that result in unrealistic ES frames.

5.3 Relative volume preservation strategies

In the previous section we showed that parallel transport alone was able to reorient the deformation to a different frame, but was not sufficient to fully normalize it in the presence of large volume differences. We now investigate two strategies to achieve proper normalization for a broader range of volume changes and to preserve the relative volume changes.

The first strategy investigated in Subsection 5.3.1 was presented at ISBI 2021 in Guigui, Moceri, et al. 2021. It hypothesises that the amplitude of the ED to atlas deformation i.e., the one along which we transport, acts on the transported deformation by scaling to construct a normalization strategy invariant to initial volume changes. The second strategy is the proposed by Niethammer and Vialard 2013 and implemented thanks to the precious advice of François-Xavier Vialard. It consists in using a metric derived from the LDMMM metric that decouples the volume change from the deformation. We present it, our implementation, and the results in Section 5.3.2.

5.3.1 Scaling after transport

Hypothesis We now propose to introduce a rescaling step after the parallel transport in our framework of Figure 5.9. More precisely, instead of using $R_t = \mathrm{Exp}_A(\Pi^A_{ED} \mathrm{Log}_{ED}(S_t))$ as the subject specific reconstruction of time-frame S_t, we introduce a parameter $\lambda > 0$ and use

$$R_t(\lambda) = \mathrm{Exp}_A\left(\lambda \Pi^A_{ED} \mathrm{Log}_{ED}(S_t)\right).$$

where Π^A_{ED} is the parallel transport map along the geodesic that joins the ED frame to the atlas. Based on our observation of the results of parallel transport (Figures 5.10 and 5.14) and of the relationship between the ejection fraction error and the disease group (Figure 5.12b), we hypothesise that this rescaling should be patient-specific and depend on the volume of the ED frame. Furthermore, rescaling should not be necessary if the volume of the ED frame matches that of the atlas. The relevant quantity is thus the ratio $\mathrm{Vol}_{ED} \,/\, \mathrm{Vol}_A$.

Criterion and Estimation of λ Given a patient's sequence, we now need a criterion to find λ. As discussed in the introduction, a clinically relevant quantity that should be conserved is the ejection fraction, as it is a straightforward scalar indicator of cardiac function. A perfect preservation of the ejection fraction between each time point and the ED frame constraints the volume of each reconstruction:

$$\mathrm{Vol}_{new}(t) = \frac{\mathrm{Vol}_A}{\mathrm{Vol}_{ED}} \mathrm{Vol}_{S_t}.$$

A straightforward loss function for λ is therefore

$$\mathcal{L}(\lambda) = \sum_{i=1}^{k} \left(\mathrm{Vol}(R_{t_i}(\lambda)) - \mathrm{Vol}_{new}(t_i)\right)^2.$$

For each patient, we then minimize this loss function with respect to λ to find the patient-specific scaling parameter. We solve this problem by gradient descent, where the gradient is computed by automatic differentiation. The parameter λ is regularized to be close to 1 to avoid poor solutions.

Results To validate the method, we first check that the ejection fraction is effectively conserved. The errors on ejection fraction and area strain are reported in Table 5.2. We indeed observe a three-fold decrease of the RMSE on the ejection fraction. Additionally, the distribution of this error is no longer related to the disease group (Figure 5.15). Visually, the reconstructions obtained by the scaled parallel transport are more realistic (Figure 5.16). Interestingly, it is possible to obtain a low error on the EF after the scaled transport, but this does not preserve the area strain (AS) . This shows that although these quantities are related, they carry different information and the AS depends on the initial shape, itself related to the pathology.

	Original Values	RMSE PT	RMSE SPT
AS	-0.24 ± 0.08	0.18	0.13
EF	0.42 ± 0.13	0.13	0.04

Table 5.2: Validation of scaled parallel transport (SPT) with EF and AS.

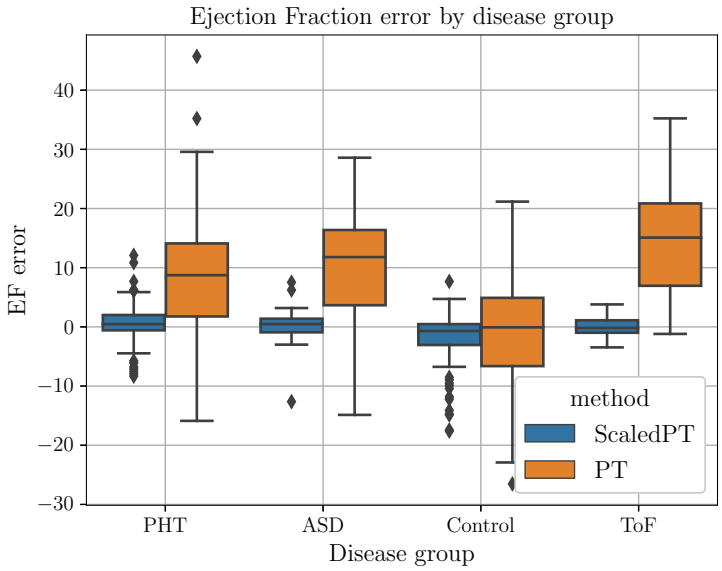


Figure 5.15: Alteration of the ejection fraction per disease group. The scaled parallel transport achieves an error that is not related to the disease group.

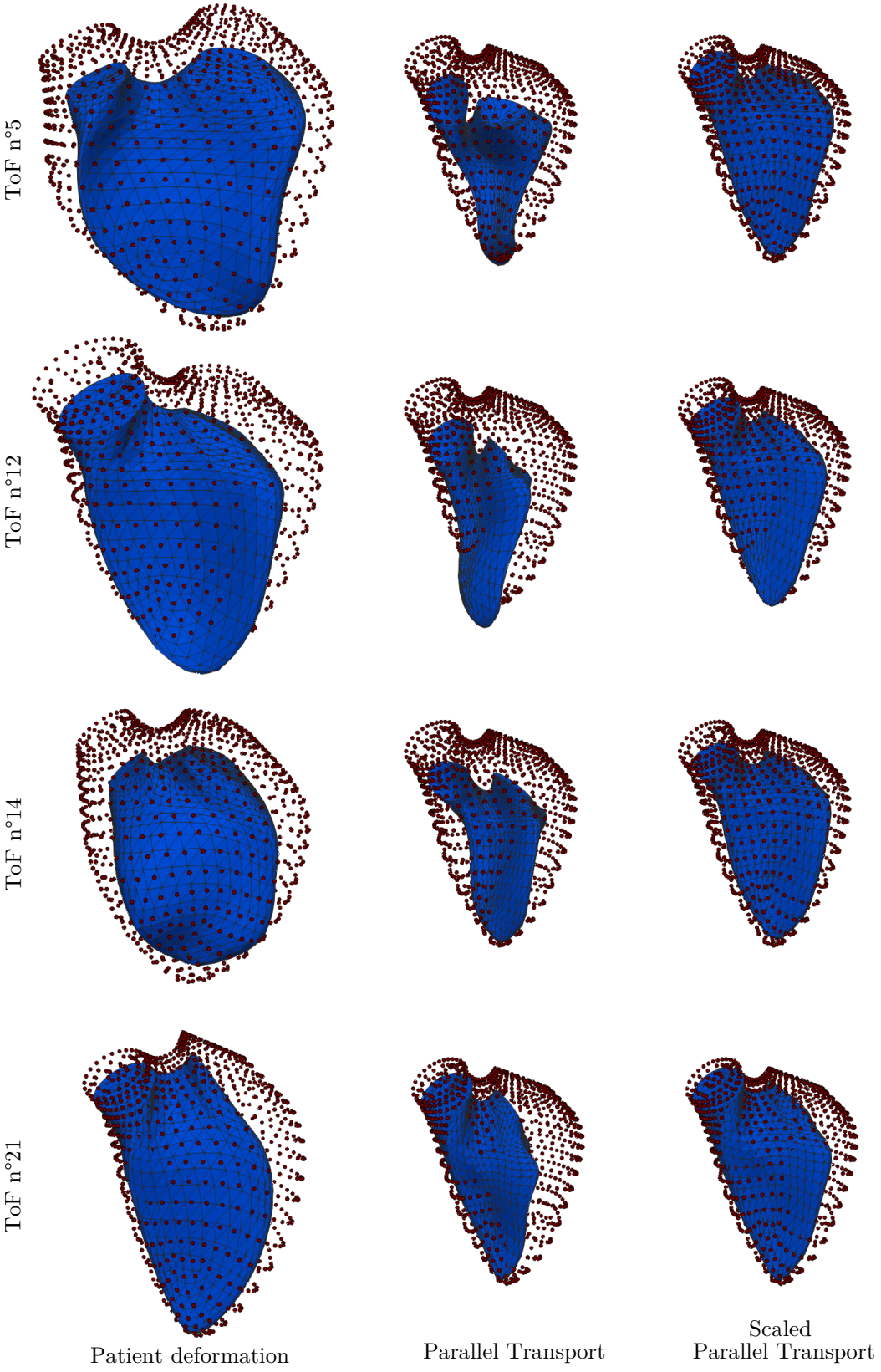


Figure 5.16: Examples of reconstructions of the ES frame (blue meshes) after parallel transport of the ED-to-ES deformation (point cloud to mesh) along ED-to-atlas deformation. The large volume changes between subjects and the atlas result in unrealistic ES frames (middle column). This problem is well addressed by the scaling strategy (right column).

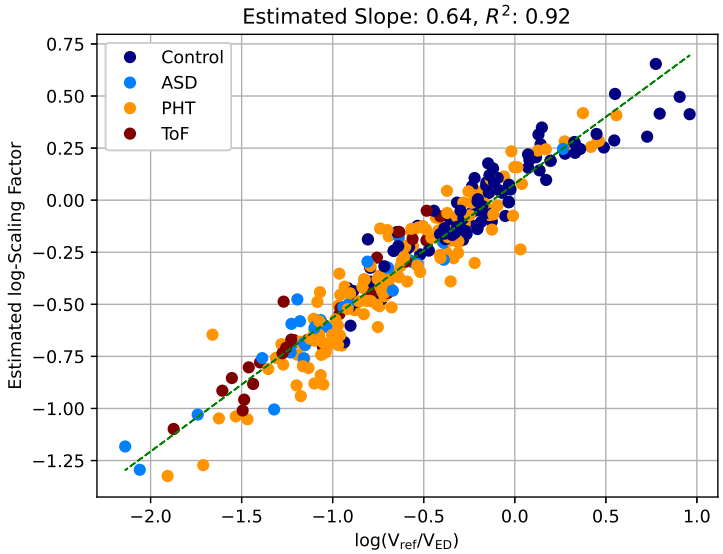


Figure 5.17: Scaling Parameter λ with respect to the ED volume.

Relationship between λ and Vol_{ED} As expected, the scaling coefficient is closely related to the ED volume. We use a linear regression to identify a linear relation between $\log(\lambda)$ and $\log(\mathrm{Vol}_A \,/\, \mathrm{Vol}_{ED})$. This is displayed on Figure 5.17. The relationship seems highly significant as we measure a coefficient of determination $R^2 = 0.92$. The measured slope is $\alpha = 0.64$ and corresponds to the relationship

$$\lambda = \beta \left(\frac{\mathrm{Vol}_A}{\mathrm{Vol}_{ED}} \right)^{\alpha}. \tag{5.5}$$

where β is the exponential of the intercept, but is found close to 1, with $\beta = 1.08$

In order to better understand this relationship, we reproduce the experiment on 2d simulated shapes, with the same experimental set-up as in the validation experiment. It is striking that we also find a highly significant relationship with $R^2 = 0.99$ and almost the same slope $\alpha = 0.67$ (Figure 5.18) and $\beta = 1.02$. The values measured for α are close to $2/3$ and surprisingly do not seem to depend on the dimension of the ambient space.

Our experiments thus give strong evidence in favour of relation (5.5), and raise many questions. First is α a constant, or does it depend on the data, the dimension, the kernel, or the deformation model? From our results it seems that it does not depend on the data or the dimension. Performing the same experiments with different kernels could bring further information, this is ongoing work. Similarly, experiments in the stationary velocity field (SVF) framework (Arsigny, Commowick, et al. 2006) could be performed to test whether this effect is specific to LDDMM or more general. These experiments could also clarify wether β significantly differs from one.

Furthermore, this effect suggests that all the experiments that have been performed with parallel transport should be revised with a scaling step. Most of these

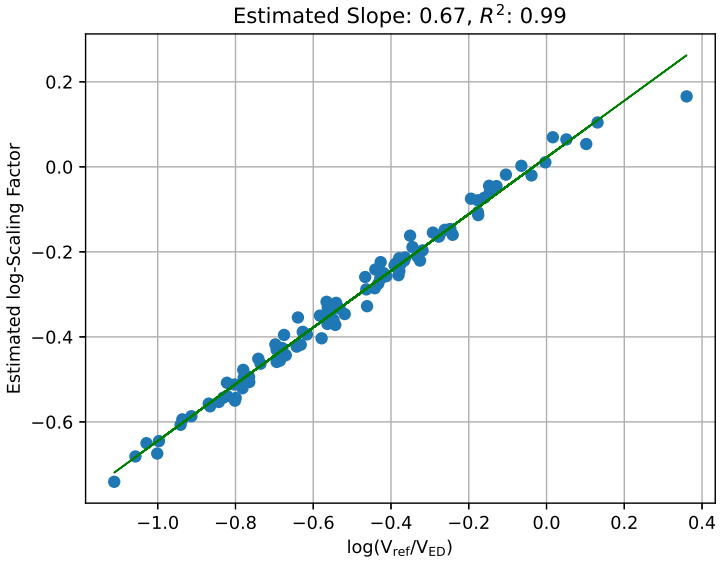


Figure 5.18: Scaling Parameter λ with respect to the ED volume on synthetic 2d data.

in the medical imaging field concern the evolution of brain images (Cury, Lorenzi, et al. 2016; Lorenzi, Pennec, et al. 2015). However, longitudinal deformations of the brain are very different from that of the right ventricle, and we don't expect the scaling step to be necessary. Indeed, brain deformations are characterized by very localized strong Jacobian determinant, but very little overall volume changes. A more relevant type of data to reproduce this effect could be to evaluate tumor evolution during treatment, or the evolution of more plastic organs such as the liver, whose shape is affected by all the liver diseases, whether viral, metabolic or due to alcohol.

Finally, a last hypothesis is that by scaling the parallel transport of the LDDMM metric, we are approximating the parallel transport of a direct product metric with a volume preserving factor and a metric on the volume changes as will be introduced in the next section. This relation would translate in setting the metric on the volume part.

5.3.2 Preserving metric

We now present the second method that we investigated to define a normalization procedure that preserves relative volume changes. This method is based on the approach of Niethammer and Vialard 2013 that aims to construct a scale invariant metric. More generally, they show that a Riemannian metric on the space of shapes preserves a global non-degenerate function f (such as volume) if and only if it can be decomposed into a product metric on $\mathrm{Im}(f)$ and $f^{-1}(0)$ (see Niethammer and Vialard 2013, Theorem 41).

Model We apply their result as follows. Restricting to the space of shapes that are embeddings of the circle in $\mathbb{R}^2$ or of the sphere in $\mathbb{R}^3$, let $M = \mathrm{Emb}(S^d, \mathbb{R}^{d+1})$ be the set of shapes, endowed with a metric g. As we are interested in relative volume changes, we consider the function $f = \log\left(\frac{\mathrm{Vol}}{\mathrm{Vol}_A}\right) : M \to \mathbb{R}$ (so that $df = \frac{d\,\mathrm{Vol}}{\mathrm{Vol}}$), and let $M_0 = f^{-1}(0)$, the space of shapes whose volume equals the volume of the atlas Vol_A. Then, by Niethammer and Vialard 2013, Theorem 41, (M, g) can be decomposed into a direct product of Riemannian manifolds $(M, g) = (\mathbb{R}, dt^2) \times (M_0, g_0)$ where g_0 is a Riemannian metric on the submanifold M_0. This metric can be constructed by choosing a projection $\pi : M \to M_0$ and the restriction of g to M_0. However, there is no canonical projection and two schemes are proposed in Niethammer and Vialard 2013:

- by gradient flow: we follow the flow of $(d\,\mathrm{Vol})^{\#}$ where $\#$ depends on the choice of metric on M, in this case the LDDMM metric.
- by scaling: we centre the shape around 0 and divide all the landmarks by $\mathrm{Vol}_q^{1/3}$. This choice depends on the centre (barycenter) of the shape which may be unnatural.

In both cases, the framework is slightly modified by applying this projection first to all the time frames, so that their volume matches that of the atlas. Then the previous framework of Figure 5.9 is applied with volume preserving geodesics and parallel transport. This corresponds to the parallel transport on the factor M_0 of $M = \mathbb{R} \times M_0$. Finally, the (Euclidean) transport on $\mathbb{R}$ corresponds to applying the vector $\log\left(\frac{\mathrm{Vol}_{S_t}}{\mathrm{Vol}_{ED}}\right)$ to 0, so that the new volume is $\mathrm{Vol}_{new}(t) = \frac{\mathrm{Vol}_A}{\mathrm{Vol}_{ED}}\,\mathrm{Vol}_{S_t}$ as in the previous section, and the ejection fraction is preserved by construction. This volume is obtained either by scaling of by gradient flow, consistently with the projection.

Implementation We now give more details on the computation of geodesics and our implementation in the LDDMM framework of Section 5.2. Recall that tangent spaces are described by the Hilbert space H_K of vector fields obtained by convolution. As we restrict to $M_0 = f^{-1}(0)$, a vertical tangent subspace at any shape q is defined as the set of vector fields of H_K that are volume preserving, i.e.

$$\ker(df_q) = \{v \in H_K | d\,\mathrm{Vol}_q((v(q)) = 0\},$$

To avoid any confusion, we distinguish the linear form $d\,\mathrm{Vol}_q$ from its representation in the canonical basis $\partial\,\mathrm{Vol}_q$, such that $d\,\mathrm{Vol}_q(v(q)) = v(q)^\top \partial\,\mathrm{Vol}_q$. The orthogonal projection on V_q (which depends on the LDDMM metric) can be computed by

$$\begin{aligned} \pi_q : T_qM &\longrightarrow V_q \\ v &\longmapsto v - d\,\mathrm{Vol}_q(v)\vec{n} \end{aligned} \tag{5.6}$$

where $\vec{n} = \frac{(d\,\mathrm{Vol}_q)^{\#}}{\|(d\,\mathrm{Vol}_q)^{\#}\|^2} = \frac{H_K^* \mathbf{K}(\cdot,q)\partial\,\mathrm{Vol}_q}{\partial\,\mathrm{Vol}_q^\top \mathbf{K}(q,q)\partial\,\mathrm{Vol}_q}$. Note that the map $\#$ associates a tangent vector to the linear form $d\,\mathrm{Vol}$ such that $d\,\mathrm{Vol}(v) = \langle (d\,\mathrm{Vol}_q)^{\#}, v\rangle_K$, so that $(d\,\mathrm{Vol}_q)^{\#} = \mathbf{K}(\cdot, q)\partial\,\mathrm{Vol}_q$. Define also the dual $\pi_q^* : T_q^*M \to V_q^*$ to π_q by the relation

$\mu(\pi_q(v)) = \pi_q^*(\mu)(v)$, i.e. in matrix notations $\mu^\top \pi_q(v) = v^\top \pi_q^*(\mu)$. From (5.6), we obtain

$$\begin{aligned} \pi_q^* : T_q^*M &\longrightarrow V_q^* \\ (c,\mu) &\longmapsto \left(c, \mu - \frac{\mu^\top \mathbf{K}(c,q)\partial \operatorname{Vol}_q}{\partial \operatorname{Vol}_q^\top \mathbf{K}(q,q)\partial \operatorname{Vol}_q} d\operatorname{Vol}_q \right). \end{aligned} \tag{5.7}$$

Geodesics A geodesic between $q_0, q_1 \in \mathrm{Emb}_1(S_2, \mathbb{R}^3)$ is solution of:

$$\inf \int_0^1 \|v_t\|_K^2 dt \tag{5.8}$$

under the constraints $\dot{q} = v_t(q(t))$ where $v_t \in V_{q(t)}$. The constraints can be relaxed to $\dot{q} = \pi_{q(t)}(v_t(q(t))$ where $v_t \in V$. This allows solving Problem (5.8) by solutions of Hamiltonian equations given by $H(\mu, q) = \frac{1}{2} < \pi_q^*(\mu), K(q)\pi_q^*(\mu) >$.

In order to implement these geodesics in the LDDMM framework presented in Section 5.2, we consider that both the control points and the landmark points are part of the system, and write the Hamiltonian

$$H(\mu, q, c) = \frac{1}{2}\mu^T \mathbf{K}(c,c)\mu - \frac{(\mu^T \mathbf{K}(c,q)\partial \operatorname{Vol}_q)^2}{2\|(d\operatorname{Vol}_q)^\#\|_K^2}.$$

This results in the modified flow equations

$$\begin{cases} \dot{c} &= \mathbf{K}(c,c)\mu - \lambda \mathbf{K}(c,q)\partial \operatorname{Vol}_q \\ \dot{q} &= \mathbf{K}(q,c)\mu - \lambda \mathbf{K}(q,q)\partial \operatorname{Vol}_q \\ \dot{\mu} &= -\mu^T \nabla_1 \mathbf{K}(c,c)\mu + \lambda \mu^T \nabla_1 \mathbf{K}(c,q)\partial \operatorname{Vol}_q, \end{cases} \tag{5.9}$$

where $\lambda = \frac{\mu^T \mathbf{K}(c,q)\partial \operatorname{Vol}_q}{\|(d\operatorname{Vol}_q)^\#\|_K^2}$.

This system can readily replace (5.3) for registration and in the geodesics of the pole ladder constructions. We can thus conduct the same experiment as in the previous section with this metric.

Results We first validate our implementation by measuring the area strain and ejection fraction errors, and complete Table 5.2 into Table 5.3. Similarly, we reproduce the boxplot of Figure 5.12b with the two projection and the previous volume constrained scaling strategy. As in the previous section, there is no relation between the error and the disease group (Figure 5.19). We show some reconstructions on Figure 5.20, along the reconstructions obtained by the scaled parallel transport of the previous section. The two metrics, and the two projection methods result in different reconstructions. The projection by gradient flow seems less stable than the two other methods, as the thinning near the valve is exaggerated on the first row, and the apex of the second row is unrealistically spherical.

The values of area strain obtained by this normalization procedure were included in the analysis of Di Folco, Guigui, et al. that leveraged dimensionality reduction

	Original Values	PT	SPT	grad VPPT	scaling VPPT
AS	-0.24 ± 0.08	0.18	0.13	0.13	0.09
EF	0.42 ± 0.13	0.13	0.04	0.02	0.02

Table 5.3: Validation of the volume preserving parallel transport (VPPT), with the projection performed either by gradient flow or by scaling.

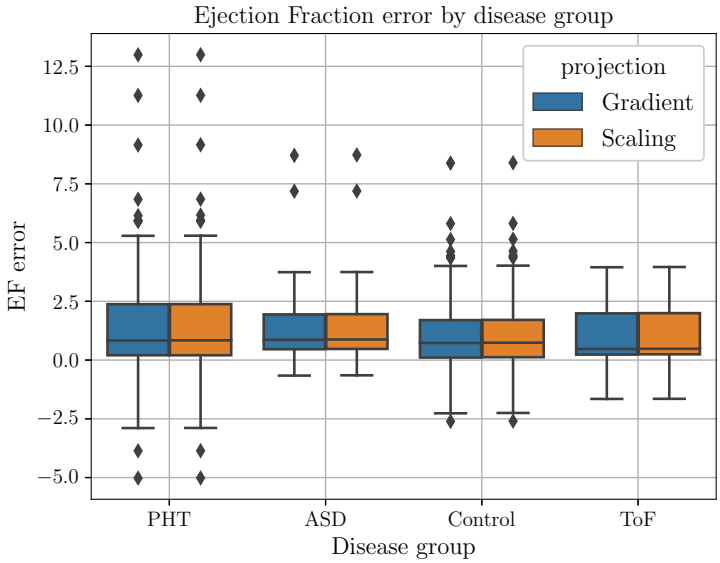


Figure 5.19: Analysis of the alteration of EF by our three normalization strategies after parallel transport, against the disease group. There is no difference between the errors in the different disease groups and between the two projection methods.

methods to compare the different normalization procedures and explore the interactions between shape and deformations. Shape descriptors were computed by pointwise differences between the ED mesh and the atlas after a rigid-body alignment, or alignment by a similarity. By comparing the latent spaces obtained with normalized and unnormalized data, they show that both methods reduce the bias introduced by the alignment of the ED shapes (Di Folco, Guigui, et al. 2021).

A few area strain maps before and after transport by the different methods are shown on Figure 5.21 and 5.22. Beside filtering outliers, it is not clear what is expected of the effect of the normalization strategy on the area strain. This figure also confirms that the volume preserving parallel transport, and the rescaled parallel transport of Section 5.3.1 are indeed different. The volume preserving with projection by scaling seems best at conserving the area strain maps, and this is confirmed in Table 5.3.

Now we come to the original goal of this chapter: performing groupwise analysis of the systolic deformation. We have proposed several ways to normalize each individual time deformation, we now need to summarize the complete systolic tra-

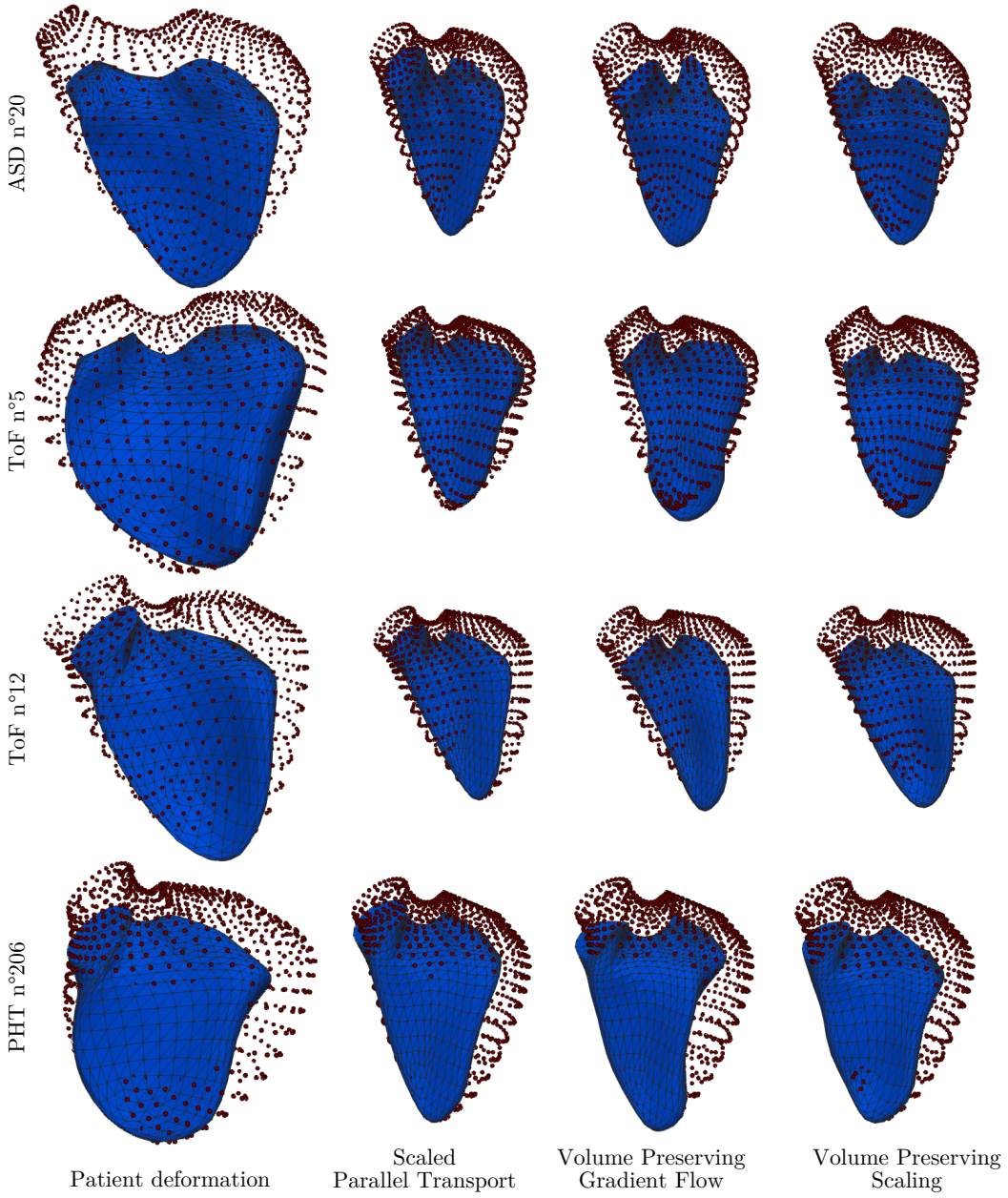
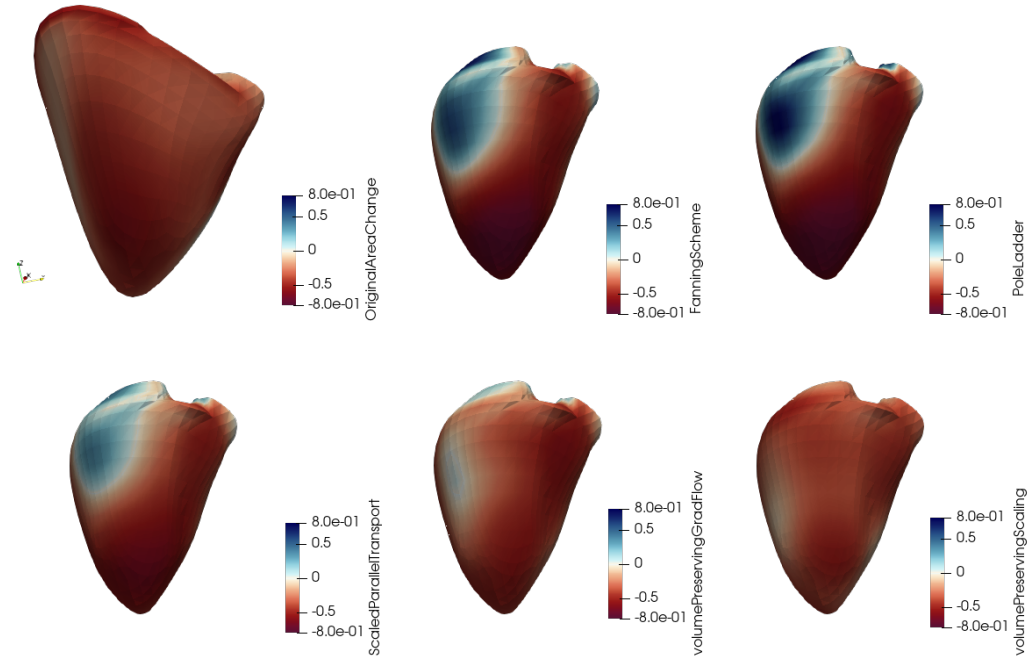
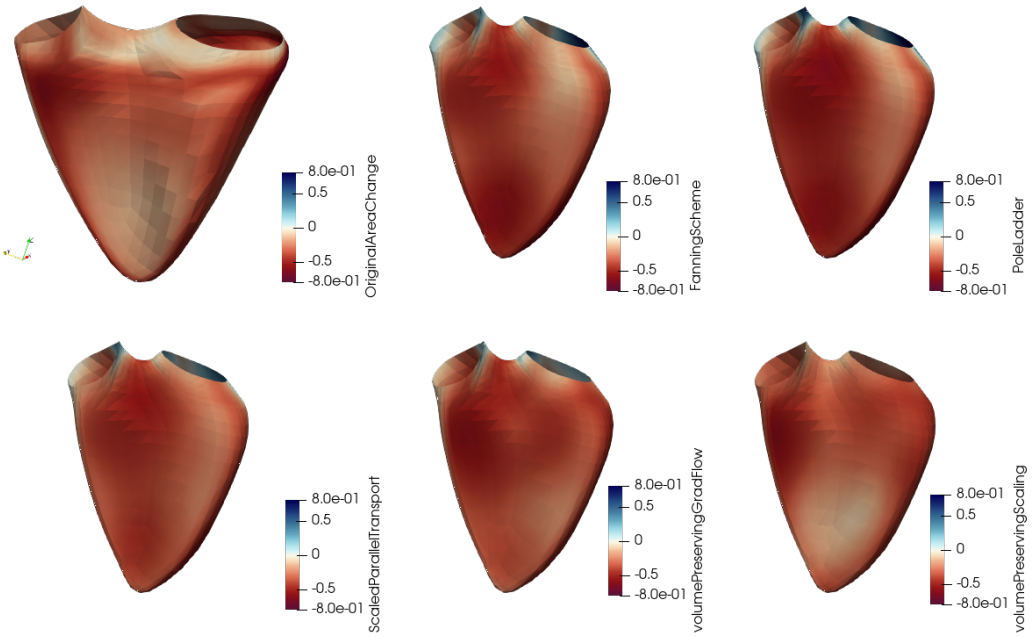


Figure 5.20: Examples of reconstructions of the ES frame (blue meshes) after parallel transport of the ED to ES deformation (point cloud to mesh) along the deformation from ED to atlas. Comparison of the three methods. The gradient flow seems less stable (first two rows) than the projection by scaling. The scaling method results in more vertical movement of the base than the two other methods. This may be due to the centering step required before scaling.

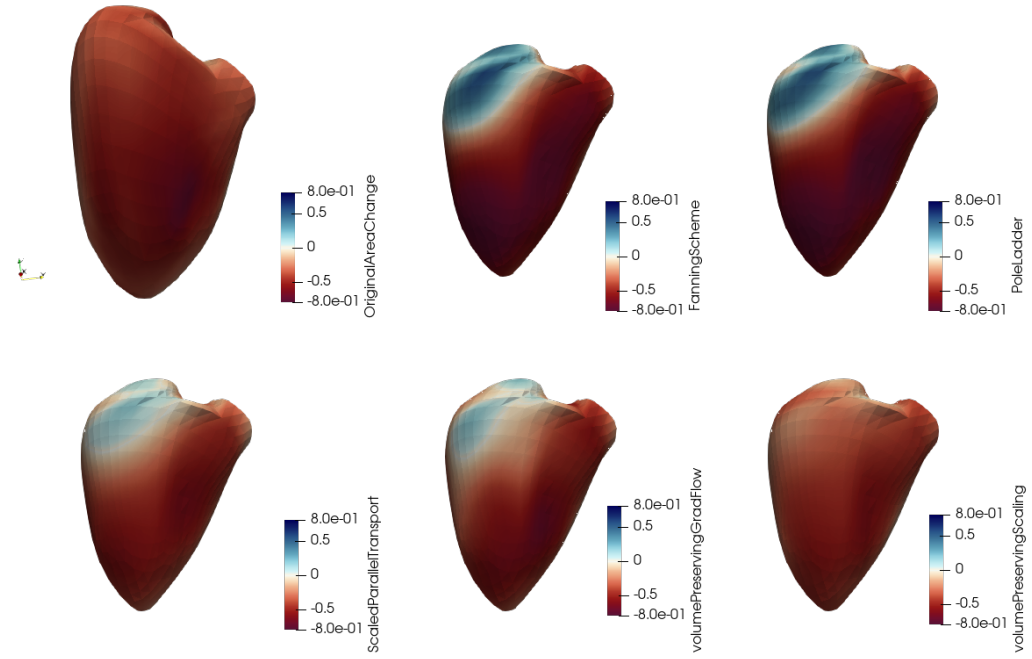


(a) View of the free wall

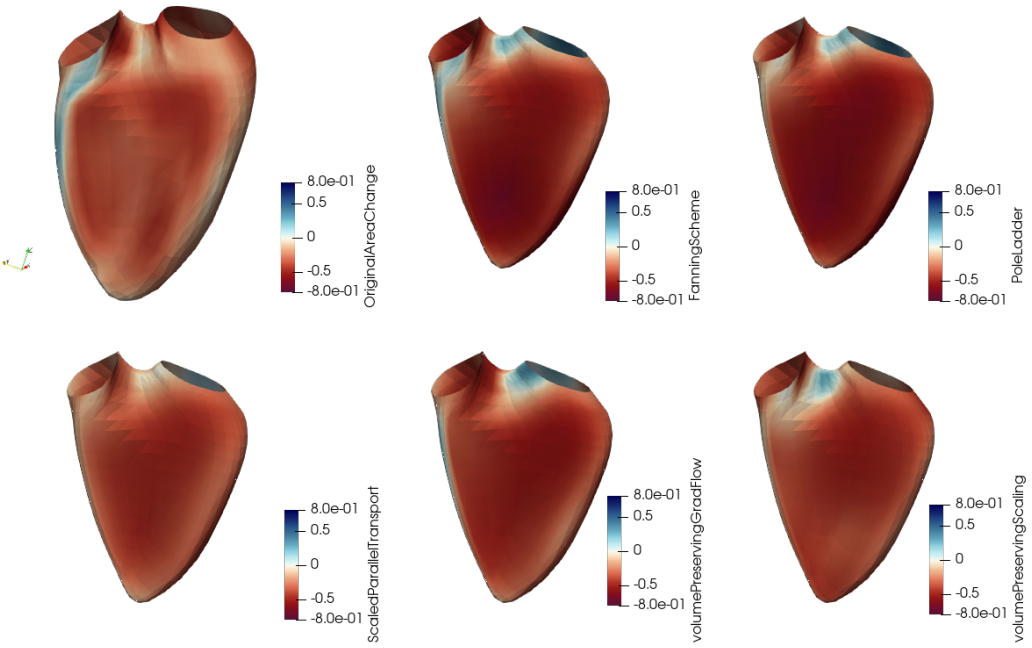


(b) View of the septum

Figure 5.21: Area strain maps for patient ASD $n°8$, before transport (OriginalAreaChange) and after normalization by each method. In this case unrealistic positive values appear near the tricuspid valve for nearly all methods except the volume preserving metric with projection by scaling. This method seems best at preserving the area strain map.



(a) View of the free wall



(b) View of the septum

Figure 5.22: Area strain maps for patient PHT n°17, before transport (OriginalAreaChange) and after normalization by each method. In this case the volume preserving scaling method seems to preserve the area strain map better than the other methods, except between the valves.

jectory. We propose to use a geodesic - or higher order- regression. This is detailed in the next section dedicated to the downstream analyses.

5.4 Analysis of the normalized deformations

5.4.1 Geodesic and Spline regression

In order to compactly represent the full systolic trajectory in the same space and to proceed with linear statistics, we propose to fit a regression model in the shape space as considered in the registration framework of Section 5.2. It estimates a geodesic path between two shapes (the equivalent of a uniform motion). For a trajectory such as the contraction of the cardiac RV, one may expect to find second-order dynamics, making a (first-order) geodesic regression ill-suited. We thus propose to use the second-order model defined in Trouvé and Vialard 2012 to account for the motion of the RV during systole. It is similar to a geodesic regression (Fishbaugh, Durrleman, et al. 2017) in the shape space (see Section 2.5.4), with an additional acceleration term in the model of the trajectory. We therefore compare it to the simpler geodesic regression.

The second-order terms u_t can be interpreted as random external forces smoothly perturbing the trajectory around a mean geodesic. They modify the continuous-time system of equations (5.3) as follows: $\forall t \in [0,1]$,

$$\begin{cases} \dot{c_k}^{(t)} &= \sum_j K(c_k^{(t)}, c_j^{(t)}) \mu_j^{(t)} \\ \dot{\mu_k}^{(t)} &= -\sum_j \nabla_1 K(c_k^{(t)}, c_j^{(t)}) \mu_k^{(t)^T} \mu_j^{(t)} + u_k^{(t)} \end{cases} \tag{5.10}$$

If we consider a discrete sequence of observation times $t_1 = 0, t_2, \dots t_d = 1$ and configurations $x_{t_1}, \dots, x_{t_d}$, one seeks to find the path ϕ_t that minimizes the new cost

$$C_S(c, \mu, u_t) = \frac{1}{\alpha^2 d} \sum_{i=1}^{d} \|x_{t_i} - \phi_{t_i}(x_{t_0})\|_2^2 + \int_0^1 \|u^{(t)}\|^2 dt + \|v_0^{c,\mu}\|_K^2. \tag{5.11}$$

In practice, the ODEs (5.3) and (5.10) are discretized in n time steps and an integration method such as Euler or Runge-Kutta is used. We define all the patients trajectories between $t = 0$ and $t = 1$, and use the same discretization for all the patients to ensure that $u_0, \dots, u_{N_c}$ are estimated at corresponding times. Along with $\mu^{(0)}$, these are estimated by gradient descent as in the case of registration. Setting $u_0 = \dots = u_{N_c} = 0$ at all times recovers a geodesic trajectory. We use a kernel bandwidth $\sigma = 15$ in all the experiments, and 60 control points for all the deformations of the atlas. The initial control points are fixed for the entire dataset so that the initial momenta can be compared consistently. They have been optimized to register the atlas on all the transported ES frames.

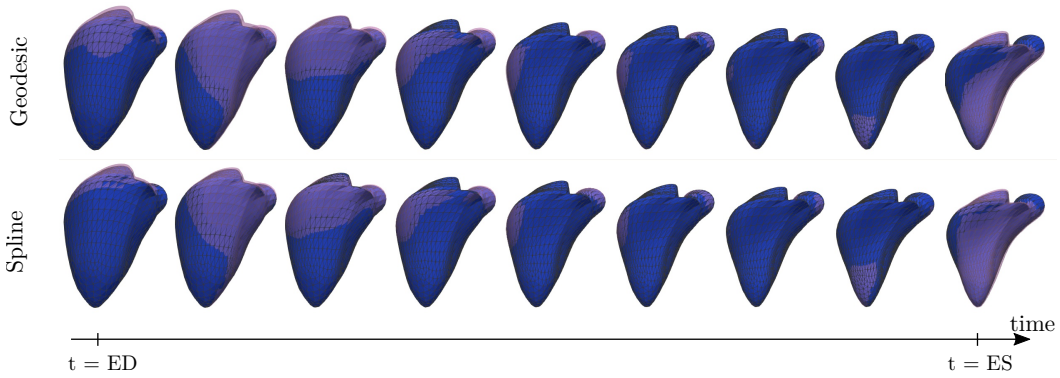


Figure 5.23: Fit of the normalized sequence of an ASD patient by geodesic (top) and spline (bottom) regression. Fits are the purple transparent meshes, normalized data are the blue meshes with edges. Upon visual inspection, there is barely any difference between the two fits.

Results Visually we only notice very slight differences between the geodesic and spline regressions (Figure 5.23). We do observe that the fit is slightly less precise in the intermediate frames for the geodesic regression, this is confirmed by measuring the overall data attachment term (left term of (5.11)) that is reported in Table 5.4, regardless of the normalization method. The spline regression therefore yields a more faithful representation of the normalized cardiac deformations than the geodesic regression, at the cost of a larger set of parameters, encompassing the external time-dependent forces u_t whose interpretation and analysis is difficult. Indeed, the size of the acceleration term u_t is $(d-1) \times N_c \times 3 = 1620$.

	SPT	grad VPPT	scaling VPPT
Geodesic	1384 ± 952	1659 ± 906	1483 ± 864
Spline	549 ± 413	681 ± 390	554 ± 324

Table 5.4: Comparison of the mean data attachment term for each method.

5.4.2 Statistics on velocities

Hotelling tests The sequences are normalized by one of the three proposed methods: scaled parallel transport (SPT), volume-preserving parallel transport (VPPT) with gradient projection and VPPT with scaling projection, and summarized by a regression with one of the two deformation models: geodesic or spline. In this paragraph, we focus on the result obtained with the scaled parallel transport and spline regression. This procedure results in a set of descriptive parameters for each individual that lie in a linear space: the co-tangent space at the atlas. For more stable descriptors of the deformation, we compute the initial velocity vectors at the initial control points with $v = \mathbf{K}(c, c)\mu$. We now have all the ingredients to perform statistics on cardiac deformations described by the initial velocities.

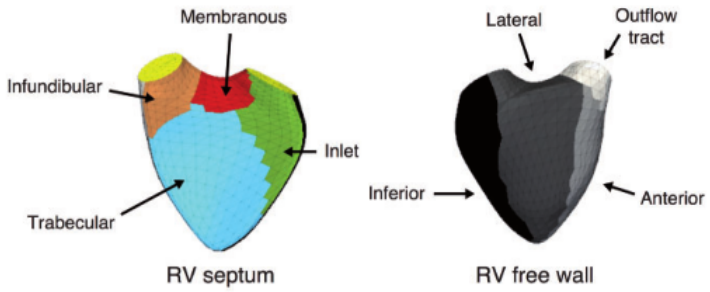


Figure 5.24: Regions of the right ventricle. Reproduced from Moceri, Duchateau, Baudouy, et al. 2018 with permission of Oxford University Press.

We compare the impact of the different diseases by performing pairwise Hotelling tests between each disease group and the control group. One statistic is computed at each control point, that tests if the mean velocity at that point is significantly different. A Bonferroni correction is then applied for multiple testing across the full set of control points, to maintain type I error risk at $\alpha = 0.05$.

The results for the velocities of the spline regression with the scaled PT are displayed Figure 5.25 for the ASD, ToF and PHT groups and show significant differences between each disease and the control group. The differences are superimposed with the group mean velocity fields, and show that these differences are mainly of magnitude, with different orientations at a few points along the free wall. The differences observed near the tricuspid valve mainly reflect the difference of magnitude of the deformations, and one should be cautious before drawing further conclusions as the quality of the mesh may vary near this region. However it is interesting to notice that very little differences are observed for the ASD group, which corroborates previous results (Moceri, Duchateau, Gillon, et al. 2020). Similarly, only small differences are observed on the septum for the PHT group, showing that the shape differences usually observed on the PHT group have been filtered out. This makes the differences observed on the free wall interesting and other markers such as the circumferential strain will be studied to confirm these effects. These experiments also highlight the difference between the PHT and ToF group as significant differences are localized on the inferior part of the free wall and on the inlet for the Tof group while they are distributed across the whole shape for the PHT group (see Figure 5.24 for the labels of the regions).

However, from prior knowledge, we would also expect to find differences near the infandibular for the ToF group as this region was modified by surgery. There is no such significant difference on the velocities, and it was previously reported to be absent from the area strain maps as well. This might be due to the segmentation process that was used between the raw 3d echocardigraphies and the meshes, that uses a template shape that is too constraining for patients with different shapes, so that these differences are smoothened. Indeed Ferraro, Adar, et al. 2020 reported variability in the quality of the acquisition for patients with altered ventricular shapes. The segmentation is here done by the TomTec software and we don't have

access to the raw images of this study.

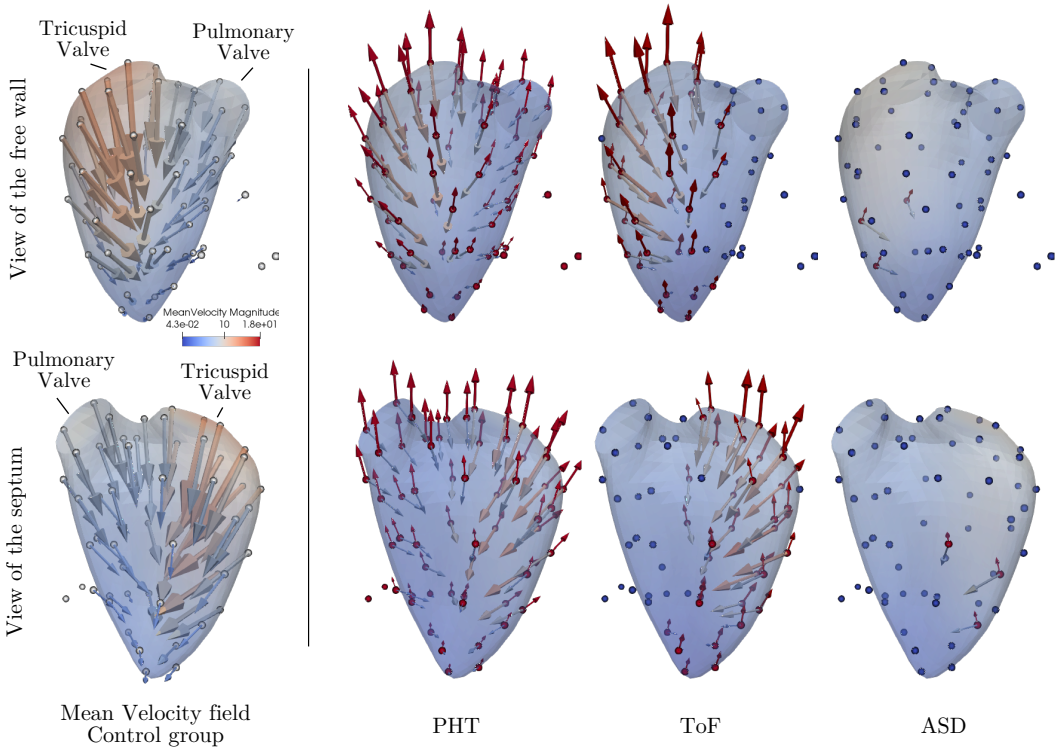


Figure 5.25: Results of the Hotelling tests between velocities. Where the differences are significant, control points and difference with the control group are in red, while coloured arrows represent the mean velocity field for the disease group. All the arrows have been scaled by a factor 2.3 for visualization purposes. The color map of the meshes reflects the norm of the velocity field at that point, if it is significantly different from the control group. As for the area strain, there is little difference for the ASD group, showing that the volume differences have well been filtered out by the normalization. Moreover the differences for the PHT and ToF group reflect deformations with less amplitude than the control group, as observed on the ejection fraction.

The same visualization is provided for the mean acceleration term by disease group, across the group-wise mean trajectory (Figure 5.26). Recall that these can be interpreted as external forces. They seem to increase the movement towards contraction at the beginning of motion while slowing it down towards the end. These terms thus provide more insights into the dynamics of each disease.

Similarly to the analysis of the velocities, the mean external forces of the ASD group is not significantly different from that of the control group. This shows that the volume differences reported in Figure 5.13 were successfully normalized by the scaled parallel transport. For the ToF and PHT group, many points have significantly different values of acceleration, and those differences are shown by red arrows. They are distributed across the whole RV for the PHT group while their locations

vary in time for the ToF group, and concentrate around the apex towards the end of the deformation. For the ToF group, we do find differences on the infandibular at times $ED + 1$ and $ED + 2$ as expected from the consequences of the surgery. This is in favor of our method on the analysis of shape deformations, as these differences were not observed with the traditional analysis of strain maps. As in the case of velocities, the differences mainly reflect amplitude differences, especially near ED and ES times. They thus reflect the late contraction of the RV and the distribution of the differences across the ventricle and in time reflect a longer and asynchronous contraction introduced by the disease. Indeed, the deformation of the control group is more uniformly distributed on the shape, and the introduction of heterogeneity across the shape has been previously reported and associated with the disease, and is known to be a factor of arrhythmia with direct consequences on survival.

Figure 5.26: Results of the Hotelling tests between accelerations. Where the differences are significant, control points and differences with the control group are in red, while coloured arrows represent the mean acceleration for the disease group. As for the velocity fields, there are no differences for the ASD group. For the Tof group, the differences are localized, but these sites vary with time, while for the PHT group, differences are distributed on the entire shape.

Linear discriminant analysis (LDA) We now investigate more deeply the differences observed in the velocities in order to evaluate the separation of the different disease groups. We use a simple linear method to classify and reduce the dimension of the whitened velocity fields. This procedure is similar to the kernel PCA used in Cury, Glaunès, et al. 2018 but we use a supervised LDA instead to investigate the relations between disease groups. The whitening step corresponds to multiplying the velocities by $\mathbf{K}(c,c)^{-1/2}$ and allows accounting for the metric when computing the covariances. We apply the model to the parameters of the geodesic and spline regression models fitted after the raw parallel transport and each normalization strategies. The projections of the data on the first three axes are displayed on Figure 5.27 for the parallel transport only, on Figure 5.28 for the spline regression parameters, and on Figure 5.29 for the geodesic regression parameters, with each time the three different normalization strategies.

With all the methods, the first two components achieve a good separation of the PHT and Control group. The ToF group is also well separated from the control group by the first component, from the PHT group by the second, and from the ASD group by the third. These differences may be further analysed by shooting in the direction of each component.

This separation is slightly better for the geodesic regression as shown by the confusion matrices reported in Figure 5.5. This might be explained by the fact that all the parameters of the geodesic regression are used in this model, while when using splines, some information captured by the external forces is not included.

From the previous results (Figures 5.12a and 5.13 (page 159) and the Hotelling tests, we know that the ASD group is on average not different from the control group in terms of ejection fraction and area strain, while it is in terms of volume. However, when normalizing by volume, at constant ejection fraction, the muscular effort is increased with consequences on the deformation. These were not significant on the mean parameters but are captured by the LDA model, explaining that the ASD group remains separated from the control group after normalization, and showing that the direction of these differences are coherent and interpretable although not large enough compared to the covariance of the data.

Finally, this experiment partially allows to compare the different normalization methods. Although not visible on the global classification accuracy, the error on the classification between the PHT and control group is well reduced by the scaled parallel transport (SPT) and the volume preserving metric with gradient projection (VPPTgrad) compared with the parallel transport only. The opposite is observed on the classification of the ASD and ToF subjects. As these groups were very different in terms of distribution of ED volumes (Figure 5.13), this shows that the volume information is less dominant in the data with a normalized parallel transport (SPT or VPPT).

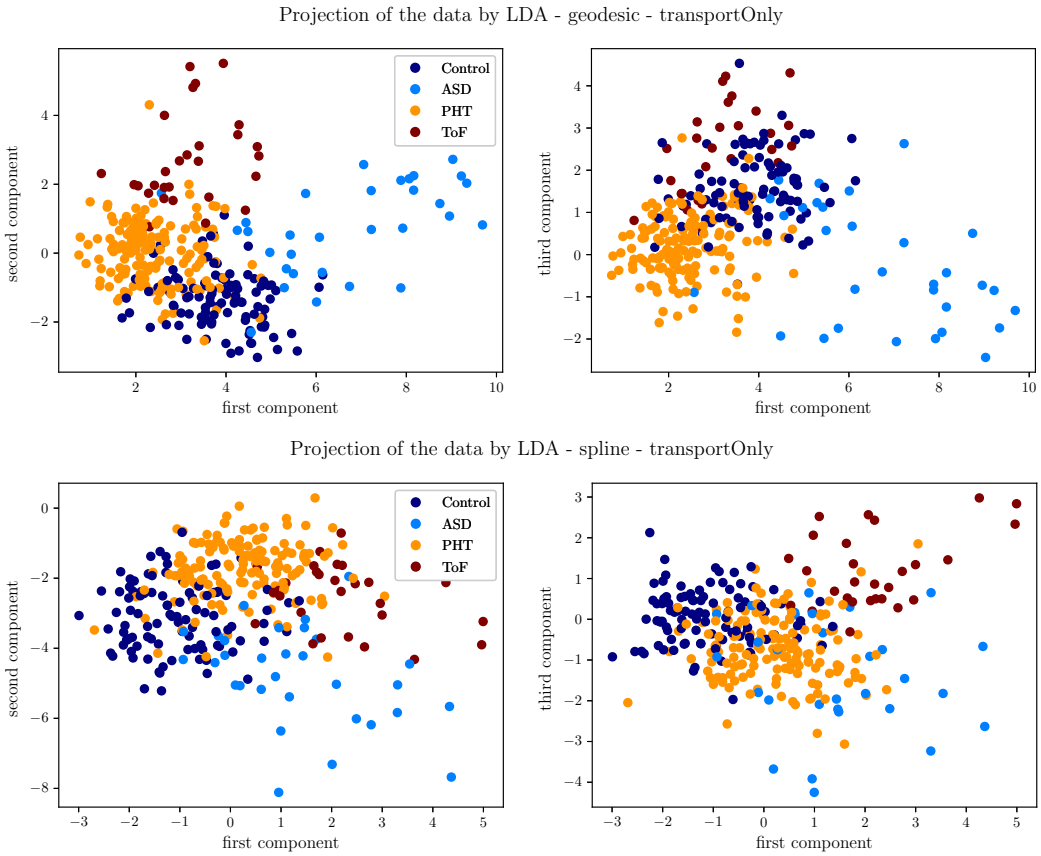


Figure 5.27: LDA on spline and geodesic regression parameters after parallel transport only

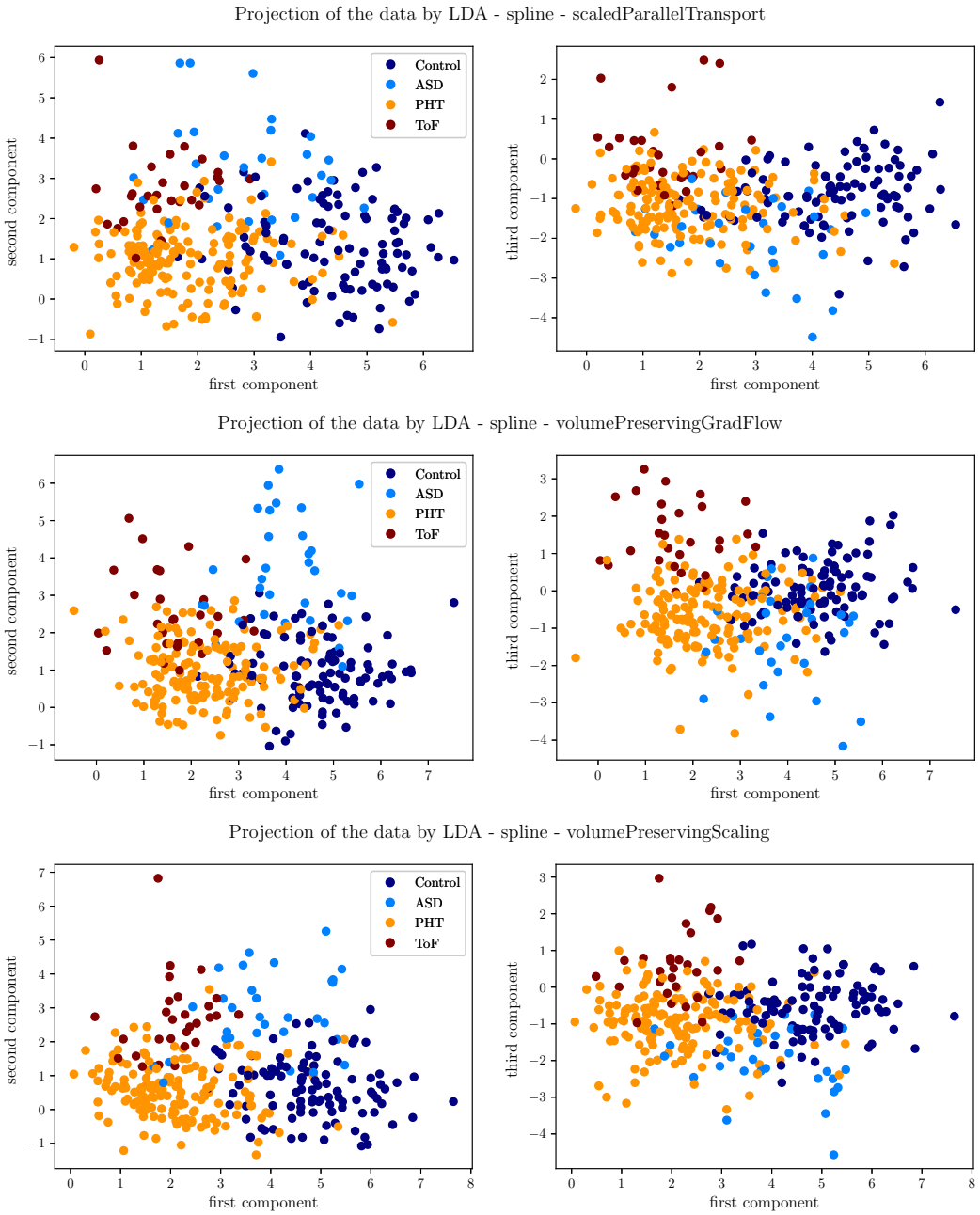


Figure 5.28: LDA for the different normalization methods, on spline regression parameters.

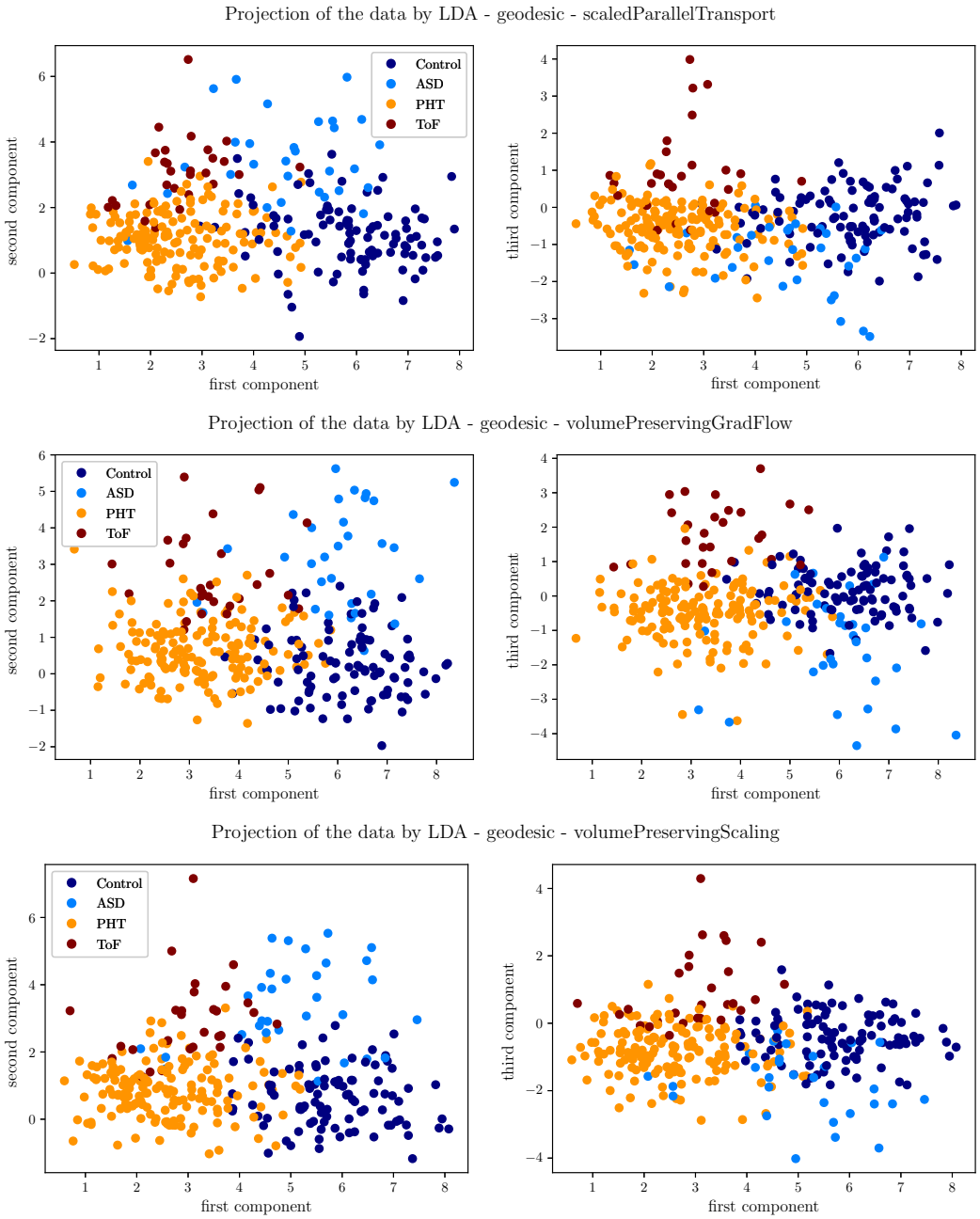


Figure 5.29: LDA for the different normalization methods, on geodesic regression parameters.

Predicted / True	Control	ASD	PHT	ToF
Control	83	0	17	0
ASD	8	20	3	0
PHT	15	0	138	2
ToF	0	0	6	22

(a) Geodesic, PT, Accuracy: 0.84

Predicted / True	Control	ASD	PHT	ToF
Control	80	1	19	0
ASD	6	19	4	2
PHT	18	0	134	3
ToF	1	0	7	20

(b) Spline, PT, Accuracy: 0.81

Predicted / True	Control	ASD	PHT	ToF
Control	87	4	9	0
ASD	4	21	5	1
PHT	8	2	143	2
ToF	1	0	10	17

(c) Geodesic, SPT, Accuracy: 0.85

Predicted / True	Control	ASD	PHT	ToF
Control	80	5	15	0
ASD	3	19	8	1
PHT	12	1	139	3
ToF	0	0	12	16

(d) Spline, SPT, Accuracy: 0.81

Predicted / True	Control	ASD	PHT	ToF
Control	91	0	9	0
ASD	6	21	2	2
PHT	11	0	142	2
ToF	1	0	6	21

(e) Geodesic, VPPTgrad, Accuracy: 0.88

Predicted / True	Control	ASD	PHT	ToF
Control	83	1	16	0
ASD	7	19	5	0
PHT	19	1	131	4
ToF	1	0	9	18

(f) Spline, VPPTgrad, Accuracy: 0.80

Predicted / True	Control	ASD	PHT	ToF
Control	89	2	9	0
ASD	4	22	4	1
PHT	9	1	142	3
ToF	0	0	9	19

(g) Geodesic, VPPTscale, Accuracy: 0.87

Predicted / True	Control	ASD	PHT	ToF
Control	84	3	13	0
ASD	4	22	5	0
PHT	11	2	138	4
ToF	0	1	7	20

(h) Spline, VPPTscale, Accuracy: 0.84

Table 5.5: Confusion matrices for the LDA classifiers of each model.

5.5 Conclusion

In this chapter, we leveraged the results of the previous chapters to propose a new implementation of the pole ladder scheme as a reorientation step in the normalization of longitudinal shape data. This procedure exhibited a strong bias due to the volume differences at a reference time point. Two strategies were investigated to correct for this bias: scaling the parallel transport, and using a metric that decomposes volume changes from volume preserving deformations. Both methods were successful at preserving the ejection fraction, and reduced the bias due to volume differences. When scaling the parallel transport, we discovered a significant relationship between the scaling parameter and the initial volume ratio, however its interpretation must be understood further. This will be investigated in future work, together with the possible mathematical relation to the volume preserving metric.

The analysis of the deformations was meaningful and coherent with previous knowledge, but it was not sufficient to produce new findings. More subtle differences could be observed with more data or by studying directly the raw 3d images, but those are hardly available in echographic studies.

This study was neither sufficient to decide which method should be preferred. The scaling is of course computationally simpler, hence faster. The volume-preserving parallel transport with the projection by scaling was the most successful at filtering outliers in terms of area strain. We have investigated other criteria such as comparisons of the curvature of the reconstructed shapes or of the area strain maps with distance functions on densities, but we did not find any relevant difference between the methods. In the SPT strategy, the LDDMM method is used and the volume changes are taken into account after parallel transport while in the VPPT strategy, a metric that takes into account the volume changes is used. As the first phase of the systolic contraction occurs with constant volume, there is a physical motivation for using the latter metric, but this only concerns a fraction of the deformation. This raises the question whether a finer modelling with different metrics would be meaningful.

This comes back to the more general question of choosing the metric. In this implementation of the LDDMM framework, this boils down to the choice of kernel, but the framework itself is questioned. As data-driven methods have gained popularity in this field, more papers have proposed strategies to learn the metric from data (Vialard and Risser 2014; Louis, Couronné, et al. 2019; Niethammer, Kwitt, et al. 2019). These solutions are attractive to remove the arbitrary aspect of the choice of metric. Yet, beyond the computational complexity, these approaches face the problem of defining 'natural' criteria to assess a model. It will be interesting to test the parallel transport normalization method with a metric learned from data.

Finally, the framework proposed in this chapter is readily usable to evaluate the impact of a heart surgery on the deformation. It can also be used to simulate new cardiac sequences for a given shape from a population of cardiac sequences, by performing e.g. a principal component analysis of the momentum vectors of the spline or geodesic regression, and shooting along the principal modes, or by sampling

coefficients from a multivariate normal law to mix the principal components. We provided such examples to collaborators of the SimCardioTest consortium that aims at enhancing numerical experiments to reduce the burden of clinical trials in the drug validation process. Their feed-back will help assess the relevance of the parallel transport framework and may provide new criteria to improve the normalization of deformations.

Chapter 6

Conclusion and future works

6.1 Summary of the contributions

This Ph.D thesis focuses on the computational aspects of geometric statistics with application to the study of cardiac deformations. It gives a self-contained introduction to the field, covering the notions of manifold and Riemannian metric, with a focus on the additional structures encountered in applications: Lie groups, homogeneous spaces, quotient spaces, and the metrics that are defined with respect to these structures. All these notions are illustrated by the most common examples and have been implemented in the open-source Python package geomstats.

During this Ph.D, the author directly contributed to geomstats with more than 800 commits, adding new geometries as well as major geometric and learning tools such as the parallel transport and computation of geodesics with numerical schemes, the computations with invariant metrics and quotient manifolds, geodesic regression, principal geodesic analysis and many others. A strong attention to efficiency was given, and we intended to match the performance of the existing dedicated implementations with our generic modular implementations whenever possible. Our contributions also focused on structuring the library in order to make it more intuitive and more faithful to the mathematical theory, resulting in a more modular and versatile package, with more code reuse, as described in section 2.2.4. Finally, thanks to the support of the ERC G-Stats on geometric statistics, we organized coding events branded *hackathons*, both physically at Inria and online via video conferencing tools. These gathered more than twenty contributors, directly animating the community around geometric statistics and fostering interactions between communities. A coding challenge was also organized by Nina Miolane at the workshop on computational geometry and topology of the International Conference on Learning Representations (ICLR 2021), during which more than fifteen international teams participated intending to propose the best data analysis, computational method or numerical experiment based on geomstats and Giotto-tda, a topological data analysis package. All the contributions were of high quality and showed a number of new use-cases where geomstats had a direct impact. The design of the challenge, the submissions and the feedbacks from participants are summarized in the white paper Miolane, Caorsi, et al. 2021.

Our work on geomstats also lead to collaborations, where our personal contribution was not in the scientific hypothesis itself but on the numerical experiments. This is the case of the publication on density estimation on Lie groups with Emmanuel Chevallier (Chevallier and Guigui 2020), and of the work with Alice Le Brigant on

classification in the space of β distributions on brain and cardiac data (Le Brigant, Guigui, et al. 2021). These works further show the potential of geomstats for applications in numerous domains and the interest from members of the community.

Beyond these pedagogical and engineering contributions, we gave the first thorough analysis of ladder methods for parallel transport. We derived Taylor expansions of the constructions with respect to the curvature tensor and its derivatives, and showed that with an appropriate scaling of the initial vector to transport, both Schild's ladder and the pole ladder converge with a quadratic speed. We further showed that these guarantees are still valid even when geodesics are not available in closed form, greatly opening the range of applications of the schemes, and answering questions that had been opened for the past decades. We also contributed a new link between Schild's ladder and the Fanning scheme, that bridges the two very different formulations of the same problem. This link also explains why the latter naturally converges only linearly. The extra computational cost of ladder methods is thus easily compensated by a drastic reduction of the number of steps needed to achieve the requested accuracy. The results were illustrated in common manifolds and highlighted for the first time the effect of the covariant derivative of the curvature.

This analysis also allowed reimplementing the pole ladder as a normalization technique for cardiac motion modelling, within the LDDMM framework. We exhibited a strong residual bias due to the volume differences and proposed an additional correction strategy based on scaling the transported vector field and the conservation of the ejection fraction. This strategy was successful to conserve the ejection fraction and produced more realistic normalized image frames in the transported sequence. The individual scaling coefficient is significantly related to the volume at the reference time, and surprisingly this relation is the same in dimension two and three, and was reproduced on very different data.

We also compared this approach with two versions of a volume preserving metric. Although the resulting shapes were different, we did not find a criterion to decide which method should be preferred. The volume preserving metric with projection by scaling was the best in terms of preservation of the area strain. The problem of disentangling shape and deformation is not new and this work constitutes a step forward even in the absence of a definite answer. More specifically, we highlighted the impact of the volume differences at the reference time on the transported deformation, and showed that with a normalization of the amplitude, finer analysis of the parameters of the deformation was possible. Our normalized data was also used by Maxime Di Folco and Nicolas Duchateau who studied the impact of the different normalization procedures and the relation with shape descriptors before and after normalization (Di Folco, Guigui, et al. 2021).

6.2 Future work

geomstats The participants of the ICLR 2021 challenge came across some weaknesses of the package and made some enhancement suggestions. Thus, geomstats is still a project going under great development with an active set of contributors, and there remains a lot of work maintaining the package and animating the community both on Github and with live events. In particular, an interesting contribution would be to explicitly use the curvature of the space to speed-up optimization algorithms. This is at reach thanks to the recent development on embedded manifolds and quotient spaces, that allow to compute the curvature of a given manifold by projecting the curvature of a larger space. With this procedure we can start from 0-curvature Euclidean spaces and deduce the curvature of most of the manifolds in geomstats. Making this procedure efficient enough to include it in second-order gradient descents remains challenging but could turn out worth the cost.

Furthermore, geomstats works mainly with Riemannian manifolds, but some applications fail to fall under this realm and there is a clear lack of practical framework to deal with those. This is the case of Kendall shape spaces, where we restricted to the main stratum of the space and did not deal with the singularities. This is also the case for example of semi-definite positive matrices, that belong to an orbit space, where the orbits of same dimension are positive matrices with fixed rank and each form a smooth Riemannian manifold (Alekseevsky, Kriegl, et al. 2003). Orbit spaces are a particular type of stratified spaces, that are spaces that can be decomposed into strata where each stratum is a manifold. There are famous examples of stratified spaces in applications and performing statistics in such spaces is a challenging research topic (see Hotz, Skwerer, et al. 2013, for an example of statistics on the open book). The space of phylogenetic trees has been a strong motivation for this research in the past decades, or more generally the analysis of populations of networks with different numbers of nodes (Billera, Holmes, et al. 2001; Feragen, Owen, et al. 2013; Garba, Nye, et al. 2021). A geometric framework to encompass all these cases is an active field of research and geomstats could directly contribute to test these developments.

Finally, geomstats contains a few manifolds that arise in information geometry, that is the study of the geometry of parametric families of probability distributions. This is the case for example of the 2d-hyperbolic space that models the univariate normal distribution, of the SPD manifold with the affine-invariant metric that encodes centred-multivariate normal distributions, or of the space of parameters of the β and Dirichlet distributions. All these distributions belong to the exponential family, and are particular geometries of hessian manifolds, whose metric tensor is given as the hessian matrix of a potential function of the parameters (Amari 2016). A generic implementation of these Riemannian metrics is within close reach by leveraging automatic differentiation tools. We paved the way by implementing a proof-of-concept on the family of β distributions, and in the future this could be generalized to the entire exponential family. Note that, to the best of our knowledge, geomstats is already the first open-source framework that allows to use information

geometry explicitly and these future developments would open practical research avenues in information geometry, that has a growing impact in artificial intelligence.

Ladder methods In the future, the implementation could be improved with more adequate numerical schemes that exist on Lie groups and homogeneous spaces (Munthe-Kaas and Zanna 1997; Munthe-Kaas and Verdier 2016), or symplectic integrators (Hairer, Wanner, et al. 2002; Dedieu and Nowicki 2005). The main challenge remains the computational cost of the Riemannian logarithm. A potential workaround could be to leverage the literature on optimization on manifolds to use a high-order retraction and its inverse instead of the exponential and logarithm maps.

Cardiac modelling The physical interpretation of the relation between the scaling parameter and the volume at end-diastole remains an open question that needs to be investigated in future works on different datasets, and with different deformation models. Our normalization method could be used to evaluate the impact of reconstructive surgery on the cardiac cycle. A finer analysis of the temporal heterogeneity during the cycle would also be beneficial to study more in depth asynchronous contractions and predict the risk of arrhythmia. In the longer term, our framework could also make its way into the clinic in the context of heart failures where fluid retention may cause shape changes, hindering the assessment of cardiac function by the clinicians.

6.3 Publications and Awards

Journal papers

- Nicolas Guigui and Xavier Pennec (June 2021b). "Numerical Accuracy of Ladder Schemes for Parallel Transport on Manifolds". *Foundations of Computational Mathematics*. ISSN: 1615-3383. DOI: 10.1007/s10208-021-09515-x.
- Nicolas Guigui, Nina Miolane, and Xavier Pennec (2021). "A Hands-on Introduction to Riemannian Geometry and Geometric Statistics with Geomstats". *In preparation*.
- Nina Miolane, Nicolas Guigui, Alice Le Brigant, et al. (Dec. 2020). "Geomstats: A Python Package for Riemannian Geometry in Machine Learning". *Journal of Machine Learning Research* 21.223, pp. 1–9.
- Emmanuel Chevallier and Nicolas Guigui (2020). "A Bi-Invariant Statistical Model Parametrized by Mean and Covariance on Rigid Motions". *Entropy* 22.4, p. 432. DOI: 10.3390/e22040432.

Conference articles

- Nicolas Guigui, Shuman Jia, Maxime Sermesant, and Xavier Pennec (Aug. 2019). "Symmetric Algorithmic Components for Shape Analysis with Diffeomorphisms". *GSI 2019 - 4th Conference on Geometric Science of Information*. Vol. 11712. Lecture Notes in Computer Science. Springer, Cham, pp. 759–768. DOI: 10.1007/978-3-030-26980-7_79.
- Nicolas Guigui and Xavier Pennec (July 2021a). "A Reduced Parallel Transport Equation on Lie Groups with a Left-Invariant Metric". *GSI 2021 - 5th Conference on Geometric Science of Information*. Vol. 12829. Lecture Notes in Computer Science. Springer, Cham, pp. 119–126. DOI: 10.1007/978-3-030-80209-7_14.
- Nicolas Guigui, Elodie Maignant, Alain Trouvé, and Xavier Pennec (July 2021). "Parallel Transport on Kendall Shape Spaces". *GSI 2021 - 5th Conference on Geometric Science of Information*. Vol. 12829. Lecture Notes in Computer Science. Springer, Cham, pp. 103–110. DOI: 10.1007/978-3-030-80209-7_12.
- Nina Miolane, Nicolas Guigui, Hadi Zaatiti, et al. (July 2020). "Introduction to Geometric Learning in Python with Geomstats". *SciPy 2020 - 19th Python in Science Conference*, p. 48. DOI: 10.25080/Majora-342d178e-007.
- Nicolas Guigui, Pamela Moceri, Maxime Sermesant, and Xavier Pennec (Apr. 2021). "Cardiac Motion Modeling With Parallel Transport And Shape Splines". *ISBI 2021 - IEEE 18th International Symposium on Biomedical Imaging*, pp. 1394–1397. DOI: 10.1109/ISBI48211.2021.9433887.
- Alice Le Brigant, Nicolas Guigui, Sana Rebbah, and Stéphane Puechmorel (Jan. 2021). "Classifying Histograms of Medical Data Using Information Geometry of Beta Distributions". *IFAC-PapersOnLine*. MTNS 2020 - 24th International Symposium on Mathematical Theory of Networks and Systems 54.9, pp. 514–520. ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2021.06.110.
- Maxime Di Folco, Nicolas Guigui, Patrick Clarysse, Pamela Moceri, and Nicolas Duchateau (June 2021). "Investigation of the Impact of Normalization on the Study of Interactions Between Myocardial Shape and Deformation". *Functional Imaging and Modeling of the Heart*. Springer, Cham, pp. 223–231. DOI: 10.1007/978-3-030-78710-3_22.

Awards

- 2nd prize at the 2020 Pierre Laffitte competition, among 2nd-year PhD students of Sophia Antipolis, organized by Fondation MINES ParisTech.
- 3rd poster prize at MOMI2021 workshop, organized by the Inria PhD Seminar.

Appendix A

Lexicon

Definition A.0.1 (Topology)**.** Let M be a set and $\mathcal{P}(M)$ the set of subsets of M. Then a *topology* on M is a set $\Theta \in \mathcal{P}(M)$ such that

- $\emptyset \in \Theta$ and $M \in \Theta$
- $\forall U, V \in \Theta, U \cap V \in \Theta$
- $C \subseteq \Theta \implies \bigcup_{U \in C} U \in \Theta$

The sets in Θ are called *open* sets, and a set S is said to be *closed* if and only if $M \setminus S \in \Theta$. Such a pair (M, Θ) is called a topological space.

Definition A.0.2 (Hausdorff)**.** A topological space (M, Θ) is said to be *Hausdorff* if, for any two distinct points $p, q \in M$, there exist open neighborhoods of p and q with empty intersection.

Definition A.0.3. A topological space (M, Θ) is second-countable if there exists some countable collection $\mathcal{A} = \{U_i\}_{i \in \mathbb{N}}$ of open sets of M such that any open set can be written as a union of elements of $\mathcal{A}$

Definition A.0.4 (Connected)**.** A topological space (M, Θ) is said to be *connected* unless there exist two non empty open sets $A, B \in \Theta$ such that $A \cap B \neq 0$ and $M = A \cup B$.

Equivalently, M is connected if and only if the only subsets that are both open and closed are M itself and the empty set $\emptyset$.

Definition A.0.5 (Group)**.** A *group* is a couple $(G, \cdot)$ where G is a nonempty set, and $\cdot : G \times G \to G$ is a map such that

- $\exists e \in G, \forall g \in G,\ e \cdot g = g \cdot e = g$,
- $\forall g \in G, \exists h,\ g \cdot h = h \cdot g = e$. We define the inverse of g for $\cdot$ as $g^{-1} = h$ in this case.

Definition A.0.6 (Algebra)**.** An *algebra* over a field K is a vector space $(A, +, \cdot)$ over K equipped with a bilinear multiplicative law $\otimes : A \times A \to A$ such that

- (distributivity) $\forall x, y, z \in A,\ (x + y) \otimes z = x \otimes z + y \otimes z$ and $z \otimes (x + y) = z \otimes x + z \otimes y$
- (compatibility with scalars) $\forall a, b \in K, \forall x, y \in A,\ (ax) \otimes (by) = (ab)x \otimes y$.

Definition A.0.7 (Injective-Surjective-Bijective map)**.** Let E, F be two sets and $f : E \to F$ a map between E and F. Then we say that

- f is *injective* if for every $x, x' \in E,\ x \neq x' \implies f(x) \neq f(x')$,
- f is *surjective* if for every $y \in F$, there exists $x \in E$ such that $y = f(x)$,

- f is *bijective* if it is both injective and surjective.

Definition A.0.8 (Continuous map)**.** Let E, F be two topological spaces. $f : E \to F$ is *continuous* of C^0 if for every open set $U \subset F$, its preimage $f^{-1}(U)$ by f is an open set of E.

Definition A.0.9. Homeomorphism Let $f : E \to F$ be a map between two topological spaces. f is called a *homeomorphism* if it has the following properties:

- f is a bijection,
- f is continuous,
- the inverse f^{-1} of f is continuous.

Definition A.0.10 (Differential map)**.** Let $p, n \in \mathbb{N}$, $f : U \subset \mathbb{R}^n \to \mathbb{R}^p$ a map defined on an open set U, and $x_0 \in U$. We say that f is *differentiable* at x_0 if there exists a linear map L defined in $\mathbb{R}^n$ such that

$$\forall h, \quad f(x_0 + h) = f(x_0) + L(h) + o(\|h\|).$$

In that case, L is unique and is called the differential of f at x_0, and written df_{x_0}.

Definition A.0.11 (Class C^k)**.** Let $p, n \in \mathbb{N}$, $f : U \subset \mathbb{R}^n \to R^p$ a map defined on an open set U. f is C^1 if it is differentiable on U and the map $x \mapsto df_x$ is continuous on U. Similarly we say that f is C^k or of class C^k for $k \in \mathbb{N} \cup \{\infty\}$ if f is k-times differentiable.

Definition A.0.12 (C^k-diffeomorphism)**.** Let $k \in (\mathbb{N}\backslash\{0\}) \cup \{\infty\}$ and let $f : U \to V$ be a map between two open sets of $\mathbb{R}^n$. f is called a *diffeomorphism* of class C^k if it has the following properties:

- f is a bijection,
- f is of class C^k,
- the inverse f^{-1} of f is C^k.

Definition A.0.13 (σ-algebra)**.** Let M be a set and $\mathcal{P}(M)$ the set of subsets of M. A subset $\Sigma \subseteq \mathcal{P}(M)$ is called *σ-algebra* if it has the three following properties:

- It is closed under complementation: for any set $S \in \Sigma$, $M \setminus S \in \Sigma$;
- M is an element of Σ: $M \in \Sigma$;
- Σ is closed under countable unions: $\forall (S_i)_{i \in \mathbb{N}}$, $\bigcup_i S_i \in \Sigma$.

Definition A.0.14 (Borel σ-algebra)**.** Let M be a topological set. The *Borel* σ-algebra $\mathcal{B}(M)$ on M is the smallest σ-algebra that contains all the open sets of M.

Appendix B

More geomstats code

B.1 Utility functions in geomstats

```
def homogeneous_representation(
        rotation, translation, output_shape, constant=1.):
    r"""Embed rotation, translation couples into n+1 square matrices.

    Construct a block matrix of size (n+1, n+1)

    Parameters
    ----------
    rotation : array-like, shape=[..., n, n]
        Square Matrix.
    translation : array-like, shape=[..., n]
        Vector.
    output_shape : tuple of int
        Desired output shape. This is need for vectorization.
    constant : float
        Constant to use at the last line and column of the square matrix.
        Optional, default: 1.

    Returns
    -------
    mat: array-like, shape=[..., n + 1, n + 1]
        Square Matrix of size n + 1. It can represent an element of the
        special euclidean group or its Lie algebra.
    """
    mat = gs.concatenate((rotation, translation[..., None]), axis=-1)
    last_line = gs.zeros(output_shape)[..., -1]
    last_line = gs.concatenate(
        [last_line[..., :-1],
         constant * gs.ones_like(translation)[..., None, -1]], axis=-1)
    mat = gs.concatenate((mat, last_line[..., None, :]), axis=-2)
    return mat
```

B.2 One parameter subgroups of the special Euclidean group

```
import matplotlib.pyplot as plt

import geomstats.backend as gs
```

```
import geomstats.visualization as visualization
from geomstats.geometry.special_euclidean import SpecialEuclidean

SE2_GROUP = SpecialEuclidean(n=2, point_type='matrix')
N_STEPS = 30
end_time = 2.7

theta = gs.pi / 3
initial_tangent_vecs = gs.array([
    [[0., - theta, 2], [theta, 0., 2], [0., 0., 0.]],
    [[0., - theta, 1.2], [theta, 0., 1.2], [0., 0., 0.]],
    [[0., - theta, 1.6], [theta, 0., 1.6], [0., 0., 0.]]])
t = gs.linspace(-end_time, end_time, N_STEPS + 1)

fig = plt.figure(figsize=(6, 6))
for tv, col in zip(initial_tangent_vecs, ['black', 'y', 'g']):
    tangent_vec = gs.einsum('t,ij->tij', t, tv)
    group_geo_points = SE2_GROUP.exp(tangent_vec)
    ax = visualization.plot(
        group_geo_points, space='SE2_GROUP', color=col)
ax = visualization.plot(
    gs.eye(3)[None, :, :], space='SE2_GROUP', color='slategray')
ax.set_aspect('equal')
ax.axis("off")
plt.savefig('../figures/exponential_se2.eps')
plt.show()
```

B.3 Anisotropic geodesics of the special Euclidean group

```
import matplotlib.pyplot as plt

import geomstats.backend as gs
import geomstats.visualization as visualization
from geomstats.algebra_utils import from_vector_to_diagonal_matrix
from geomstats.geometry.invariant_metric import InvariantMetric
from geomstats.geometry.special_euclidean import SpecialEuclidean

SE2_GROUP = SpecialEuclidean(n=2, point_type='matrix')
N_STEPS = 15


def main():
    """Plot geodesics on SE(2) with different structures."""
    theta = gs.pi / 4
    initial_tangent_vec = gs.array([
        [0., - theta, 1],
        [theta, 0., 1],
        [0., 0., 0.]])
    t = gs.linspace(0, 1., N_STEPS + 1)
```

```
    tangent_vec = gs.einsum('t,ij->tij', t, initial_tangent_vec)

    fig = plt.figure(figsize=(10, 10))
    maxs_x = []
    mins_y = []
    maxs = []
    for i, beta in enumerate([1., 2., 3., 5.]):
        ax = plt.subplot(2, 2, i + 1)
        metric_mat = from_vector_to_diagonal_matrix(gs.array([1, beta, 1.]))
        metric = InvariantMetric(SE2_GROUP, metric_mat, point_type='matrix')
        points = metric.exp(tangent_vec, base_point=SE2_GROUP.identity)
        ax = visualization.plot(
            points, ax=ax, space='SE2_GROUP', color='black',
            label=r'$\beta={}$'.format(beta))
        mins_y.append(min(points[:, 1, 2]))
        maxs.append(max(points[:, 1, 2]))
        maxs_x.append(max(points[:, 0, 2]))
        plt.legend(loc='best')

    for ax in fig.axes:
        x_lim_inf, _ = plt.xlim()
        x_lims = [x_lim_inf, 1.1 * max(maxs_x)]
        y_lims = [min(mins_y) - .1, max(maxs) + .1]
        ax.set_ylim(y_lims)
        ax.set_xlim(x_lims)
        ax.set_aspect('equal')
    plt.savefig('../figures/geo-se2.eps', bbox_inches='tight', pad_inches=0)
    plt.show()


if __name__ == '__main__':
    main()
```

Appendix C

Proofs of chapter 4

C.1 Computation of the expansion of Schild's ladder

The details of the Taylor approximation for SL are given below at the fourth order, and a lemma to bound the fourth order terms is proved. First we combine (4.1) and (4.2) to compute an approximation of $a = \log_x(m)$ where m is the midpoint between x_v and x_w. That is, $a = h_x(v, \frac{1}{2}b)$ where $b = l_x(v, w)$:

$$\begin{aligned}
2a &= 2v + b + \frac{1}{6}R(b,v)(v+b) + \frac{1}{24}\big((\nabla_v R)(b,v)(\frac{5}{2}b + 2v) + (\nabla_b R)(\frac{1}{2}b, v)(v+b)\big) + O(5)\\
&= 2v + w - v + \frac{1}{6}R(v,w)(2w-v) + \frac{1}{24}\Big((\nabla_v R)(v,w)(3w-2v)\\
&\quad + (\nabla_w R)(v,w)(2w-v)\Big) + \frac{1}{6}R(\frac{1}{2}(w-v), v)(v+w-v)\\
&\quad + \frac{1}{24}\big((\nabla_v R)(w-v,v)(\frac{5}{2}(w-v) + 2v) + (\nabla_{w-v} R)(\frac{1}{2}(w-v), v)(v+w-v)\big) + O(5)\\
&= w + v + \frac{1}{6}R(v,w)(2w-v-w) + \frac{1}{24}\Big((\nabla_v R)(v,w)(3w-2v-2v-\frac{5}{2}(w-v))\\
&\quad + (\nabla_w R)(v,w)(2w-v) + (\nabla_{w-v} R)(v,w)(-\frac{1}{2}w)\Big) + O(5).
\end{aligned}$$

Therefore,

$$\begin{aligned}
2a &= w + v + \frac{1}{6}R(v,w)(w-v)\\
&\quad + \frac{1}{24}\Big((\nabla_v R)(v,w)(w - \frac{3}{2}v) + (\nabla_w R)(w,v)(v - \frac{3}{2}w)\Big) + O(5). \qquad \text{(C.1)}
\end{aligned}$$

Now we compute $u = l_x(w, 2a)$:

$$\begin{aligned}
u &= 2a - w + \frac{1}{6}R(2a, w)(w - 4a) \\
&\quad + \frac{1}{24}\Big((\nabla_w R)(w, 2a)(3 * 2a - 2w) + (\nabla_{2a} R)(w, 2a)(2 * 2a - w)\Big) + O(5) \\
&= w + v + \frac{1}{6}R(v, w)(w - v) \\
&\quad + \frac{1}{24}\Big((\nabla_v R)(v, w)(w - \frac{3}{2}v) + (\nabla_w R)(w, v)(v - \frac{3}{2}w)\Big) \\
&\quad - w + \frac{1}{6}R\big(w + v, w\big)(w - 2(w + v)) + \frac{1}{24}\Big((\nabla_w R)(w, w + v)(3w + 3v - 2w) \\
&\quad + (\nabla_{w+v} R)(w, w + v)(2w + 2v - w)\Big) + O(5) \\
&= v + \frac{1}{6}R(v, w)(w - v - w - 2v) \\
&\quad + \frac{1}{24}\Big((\nabla_v R)(v, w)(w - \frac{3}{2}v - 2v - w) + (\nabla_w R)(v, w)(v - \frac{3}{2}w - w - 3v - 2v - w)\Big).
\end{aligned}$$

Thus

$$\begin{aligned}
u &= v - \frac{1}{2}R(v, w)v \\
&\quad + \frac{1}{24}\Big((\nabla_v R)(v, w)(-\frac{7}{2}v) + (\nabla_w R)(v, w)(-4v - \frac{7}{2}w)\Big) + O(5). \qquad \text{(C.2)}
\end{aligned}$$

We deduce the following

Lemma C.1.1. *With the previous notations, at x in a compact set K, u can be written $u = v + \frac{1}{2}R(w, v)v + r_4(v, w)$ and there exists $C > 0$ such that $\exists \eta > 0, \forall v, w \in B_0(\delta) \subset T_xM, \forall |s|, |t| \leq \eta$, r_4 verifies:*

$$\|r_4(sv, tw)\| \leq C(s^3t + s^2t^2 + st^3)\delta^4.$$

If furthermore $|s| \leq |t|$, then this reduces to $\|r_4(sv, tw)\| \leq Cst^3\delta^4$. Moreover, C can be bounded by bounds on the covariant derivatives of the curvature tensor.

Proof. Let $t' = \min(|t|, |s|)$. By (C.2) we get

$$r_4(sv, tw) = \frac{st}{24}\Big((\nabla_v R)(w, sv)(\frac{7}{2}sv) + (\nabla_w R)(tw, v)(4sv + \frac{7t}{2}w)\Big) + O(5).$$

Each term of the form $\nabla_{\cdot} R(tw, sv)\cdot$ can be bounded, for example:

$$\|st(\nabla_v R)(w, sv)(\frac{7}{2}sv)\| \leq s^3t\|\nabla R\|_\infty \|w\|\|v\|^3,$$

where the infimum norm on ∇R is taken on the compact set K (thus it exists and it is finite). Similarly, as $\|w\| \leq \delta$ and $\|v\| \leq \delta$

$$\|s(\nabla_v R)(tw, sv)(\frac{7}{2}sv)\| \leq s^3t\|\nabla R\|_\infty \delta^3.$$

By dealing in a similar fashion with the two other terms, we obtain:

$$\|r_4(sv,tw)\| \leq \|\nabla R\|_\infty (s^3t + s^2t^2 + st^3)\delta^4 + O(5), \tag{C.3}$$

where $O(5)$ is a combination of terms of the form $q_5(sa,tb)$, which are homogeneous polynomials of degree at least five and variables in the ball of radius δ, hence they can be bounded above by $C_1(s^3t + s^2t^2 + st^3)\delta^4$ for some C_1. The result follows with $C = \|\nabla R\|_\infty + C_1$. □

C.2 Computation of the expansion of the pole ladder

As in the previous appendix, we give here the computations for the fourth-order Taylor approximation of the PL construction. Recall (Fig. 4.8) that $v, w \in T_mM$ where $m = \gamma(\frac{1}{2})$ is the midpoint between $x = \gamma(0)$ and $\gamma(1)$. The result of the construction parallel transported back to m is u such that:

$$\begin{aligned}
-u &= l_m(w, -h_m(-w,v)) \\
&= w + h_m(-w,v) + \frac{1}{6}R(w,-h_m(-w,v))(-2h_m(-w,v) - w) \\
&\quad + \frac{1}{24}\big((\nabla_w R)(w,-h_m(-w,v))(-3h_m(-w,v) - 2w) \\
&\quad + (\nabla_{-h_m(-w,v)}R)(w,-h_m(-w,v))(-2h_m(-w,v) - w)\big) + O(5).
\end{aligned}$$

Now we plug (4.1), but it reduces to $-h_m(-w,v) = w - v + O(3)$ when it appears in a term including curvature (because we restrict to fourth order terms overall). Hence

$$\begin{aligned}
-u &= -(-w + v + \frac{1}{6}R(v,-w)(-w+2v) + \frac{1}{24}\big((\nabla_{-w}R)(v,-w)(5v-2w) \\
&\quad + (\nabla_v R)(v,-w)(-w+2v)\big) + q_5(w,v)) + \frac{1}{6}R(w,w-v)(2(w-v)-w) \\
&\quad + \frac{1}{24}\big((\nabla_w R)(w,w-v)(3(w-v)-2w) + (\nabla_{w-v}R)(w,w-v)(2(w-v)-w)\big) + O(5) \\
&= -v + \frac{1}{6}R(v,w)(-w+2v) + \frac{1}{6}R(w,-v)(w-2v) \\
&\quad + \frac{1}{24}\big((\nabla_w R)(v,-w)(5v-2w) + (\nabla_v R)(v,w)(-w+2v)\big) \\
&\quad + \frac{1}{24}\big((\nabla_w R)(v,w)(w-3v) + (\nabla_{w-v}R)(v,w)(w-2v)\big) + O(5) \\
&= -v + \frac{1}{12}\big((\nabla_w R)(v,w)(5v-w) + (\nabla_v R)(v,w)(2v-w)\big) + O(5).
\end{aligned}$$

C.3 The special Euclidean group with an anisotropic metric

This metric is used already in Example 3.2.4. From the structure constants and equation (3.1), we can compute the associated Christoffel symbols at identity for

the frame $(\tilde{e_1}, \dots \tilde{e_6})$. Let $\tau = (\sqrt{\beta} + \frac{1}{\sqrt{\beta}})$. We obtain

$$\Gamma_{ij}^k = \frac{1}{2\sqrt{2}} \text{ if } ijk \text{ is a cycle of [1,2,3]}, \tag{C.4}$$

$$\Gamma_{15}^6 = -\Gamma_{16}^5 = -\frac{2}{\tau}\Gamma_{24}^6 = \frac{2}{\tau}\Gamma_{26}^4 = \frac{2}{\tau}\Gamma_{34}^5 = -\frac{2}{\tau}\Gamma_{35}^4 = \frac{1}{\sqrt{2}}, \tag{C.5}$$

and all the others are null.

We can now prove lemma 4.3.3, formulated as: $(SE(3), g)$ is locally symmetric, i.e. $\nabla R = 0$, if and only if $\beta = 1$. This is valid for any dimension $d \geq 2$ provided that the metric matrix G is diagonal, of size $d(d+1)/2$, with ones everywhere except one coefficient of the translation part.

Proof. For $\beta = 1$, $(SE(d), g)$ is isometric to $(SO(d) \times \mathbb{R}^d, g_{rot} \oplus g_{trans})$. As the product of two symmetric spaces is again symmetric, $(SE(d), g)$ is symmetric.

We prove the contraposition of the necessary condition. Let $\beta \neq 1$. We give i, j, k, l s.t. $(\nabla_{e_i} R)(e_j, e_k)e_l \neq 0$:

$$\begin{aligned}(\nabla_{e_3} R)(e_3, e_2)e_4 &= \nabla_{e_3}(R(e_3, e_2)e_4) - R(e_3, \nabla_{e_3} e_2)e_4 - R(e_3, e_2)\nabla_{e_3} e_4 \\ &= \nabla_{e_3}(R(e_3, e_2)e_4) + \frac{1}{\sqrt{2}} R(e_3, e_1)e_4 - \frac{\tau}{2\sqrt{2}} R(e_3, e_2)e_5.\end{aligned}$$

And from the above

$$\begin{aligned}R(e_3, e_2)e_4 &= \nabla_{e_3}\nabla_{e_2} e_4 - \nabla_{e_2}\nabla_{e_3} e_4 - \nabla_{[e_3, e_2]} e_4 \\ &= -\frac{\tau}{2\sqrt{2}}\nabla_{e_3} e_6 - \nabla_{e_2} e_5 + \frac{1}{\sqrt{2}}\nabla_{e_1} e_4 \\ &= 0. \\ R(e_3, e_1)e_4 &= \nabla_{e_3}\nabla_{e_1} e_4 - \nabla_{e_1}\nabla_{e_3} e_4 - \nabla_{[e_3, e_1]} e_4 \\ &= -\frac{\tau}{2\sqrt{2}}\nabla_{e_1} e_5 - \frac{1}{\sqrt{2}}\nabla_{e_2} e_4 \\ &= -\frac{\tau}{4} e_6 + \frac{\tau}{4} e_6 = 0. \\ R(e_3, e_2)e_5 &= \nabla_{e_3}\nabla_{e_2} e_5 - \nabla_{e_2}\nabla_{e_3} e_5 - \nabla_{[e_3, e_2]} e_5 \\ &= \frac{\tau}{2\sqrt{2}}\nabla_{e_2} e_4 + \frac{1}{\sqrt{2}}\nabla_{e_1} e_5 \\ &= -\frac{\tau^2}{8} e_6 + \frac{1}{2} e_6 = \frac{1}{2}(1 - \frac{\tau^2}{4}) e_6.\end{aligned}$$

And therefore

$$\beta \neq 1 \implies \tau = (\sqrt{\beta} + \frac{1}{\sqrt{\beta}}) \neq 2 \implies (\nabla_{e_3} R)(e_3, e_1)e_4 = -\frac{\tau}{4\sqrt{2}}(1 - \frac{\tau^2}{4}) e_6 \neq 0,$$

which proves Lemma 4.3.3. □

Bibliography

Abadi, Martín, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, et al. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*.

Ablin, Pierre, Gabriel Peyré, and Thomas Moreau (July 2020). "Super-Efficiency of Automatic Differentiation for Functions Defined as a Minimum". *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, pp. 32–41.

Absil, Pierre-Antoine, Robert Mahony, and Rodolphe Sepulchre (2010). "Optimization On Manifolds: Methods and Applications". *Recent Advances in Optimization and Its Applications in Engineering*. Ed. by Moritz Diehl, Francois Glineur, Elias Jarlebring, and Wim Michiels. Berlin, Heidelberg: Springer, pp. 125–144. ISBN: 978-3-642-12597-3. DOI: 10.1007/978-3-642-12598-0_12.

Afsari, Bijan (2011). "Riemannian Lp Center of Mass: Existence, Uniqueness, and Convexity". *Proceedings of the American Mathematical Society* 139.2, pp. 655–673. ISSN: 0002-9939.

Alekseevsky, Dmitri, Andreas Kriegl, Mark Losik, and Peter W Michor (2003). "The Riemannian Geometry of Orbit Spaces. The Metric, Geodesics, and Integrable Systems". *Publicationes Mathematicae Debrecen* 62.

Amari, Shun-ichi (2016). *Information Geometry and Its Applications*. Vol. 194. Applied Mathematical Sciences. Springer Japan. ISBN: 978-4-431-55977-1. DOI: 10.1007/978-4-431-55978-8.

Arnold, Vladimir (1966). "Sur la géométrie différentielle des groupes de Lie de dimension infinie et ses applications à l'hydrodynamique des fluides parfaits". *Annales de l'institut Fourier* 16.1, pp. 319–361. ISSN: 0373-0956. DOI: 10.5802/aif.233.

Arsigny, Vincent, Olivier Commowick, Xavier Pennec, and Nicholas Ayache (2006). "A Log-Euclidean Framework for Statistics on Diffeomorphisms". *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2006*. Ed. by Rasmus Larsen, Mads Nielsen, and Jon Sporring. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 924–931. ISBN: 978-3-540-44708-5.

Atkin, Christopher J. (1975). "The Hopf-Rinow Theorem Is False in Infinite Dimensions". *Bulletin of the London Mathematical Society* 7.3, pp. 261–266. ISSN: 1469-2120. DOI: 10.1112/blms/7.3.261.

Barachant, Alexandre (2015). *PyRiemann: Python Package for Covariance Matrices Manipulation and Biosignal Classification with Application in Brain Computer Interface*.

Barachant, Alexandre, Stéphane Bonnet, Marco Congedo, and Christian Jutten (Mar. 2012). "Multiclass Brain-Computer Interface Classification by Riemannian Geometry". *IEEE Transactions on Biomedical Engineering* 59.4, pp. 920–928. DOI: 10.1109/TBME.2011.2172210.

Barczyk, Martin, Silvère Bonnabel, Jean-Emmanuel Deschaud, and Francois Goulette (Nov. 2015). "Invariant EKF Design for Scan Matching-Aided Localization". *IEEE Transactions on Control Systems Technology* 23.6, pp. 2440–2448. ISSN: 1558-0865. DOI: 10.1109/TCST.2015.2413933.

Barp, Alessandro, Anthony Kennedy, and Mark Girolami (Mar. 2019). "Hamiltonian Monte Carlo on Symmetric and Homogeneous Spaces via Symplectic Reduction". arXiv: 1903.02699.

Barrau, Axel and Silvère Bonnabel (Apr. 2017). "The Invariant Extended Kalman Filter as a Stable Observer". *IEEE Transactions on Automatic Control* 62.4, pp. 1797–1812. ISSN: 1558-2523. DOI: 10.1109/TAC.2016.2594085.

Becigneul, Gary and Octavian Ganea (May 2019). "Riemannian Adaptive Optimization Methods". *7th International Conference on Learning Representations (ICLR)*.

Bendokat, Thomas, Ralf Zimmermann, and Pierre-Antoine Absil (Nov. 2020). *A Grassmann Manifold Handbook: Basic Geometry and Computational Aspects*. Technical Report UCL-INMA-2020.07. arXiv: 2011.13699.

Berkels, Benjamin, Alexander Effland, and Martin Rumpf (Jan. 2015). "Time Discrete Geodesic Paths in the Space of Images". *SIAM Journal on Imaging Sciences* 8.3, pp. 1457–1488. DOI: 10.1137/140970719.

Bhatia, Rajendra, Tanvi Jain, and Yongdo Lim (2019). "On the Bures–Wasserstein Distance between Positive Definite Matrices". *Expositiones Mathematicae* 37.2, pp. 165–191. DOI: 10.1016/j.exmath.2018.01.002.

Bhattacharya, Rabi and Vic Patrangenaru (June 2005). "Large Sample Theory of Intrinsic and Extrinsic Sample Means on Manifolds—II". *The Annals of Statistics* 33.3, pp. 1225–1259. ISSN: 0090-5364, 2168-8966. DOI: 10.1214/009053605000000093.

Billera, Louis J., Susan Holmes, and Karen Vogtmann (Nov. 2001). "Geometry of the Space of Phylogenetic Trees". *Advances in Applied Mathematics* 27.4, pp. 733–767. ISSN: 0196-8858. DOI: 10.1006/aama.2001.0759.

Bône, Alexandre, Maxime Louis, Benoit Martin, and Stanley Durrleman (Sept. 2018). "Deformetrica 4: An Open-Source Software for Statistical Shape Analysis". DOI: 10.1007/978-3-030-04747-4_1.

Boomsma, Wouter, Kanti V. Mardia, Charles C. Taylor, Jesper Ferkinghoff-Borg, Anders Krogh, and Thomas Hamelryck (July 2008). "A Generative, Probabilistic Model of Local Protein Structure". *Proceedings of the National Academy of Sciences of the United States of America* 105.26, pp. 8932–8937. ISSN: 0027-8424. DOI: 10.1073/pnas.0801715105.

Bronstein, Michael M., Joan Bruna, Taco Cohen, and Petar Veličković (May 2021). "Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges". arXiv: 2104.13478.

Brooks, Daniel, Olivier Schwander, Frederic Barbaresco, Jean-Yves Schneider, and Matthieu Cord (2019). "Riemannian Batch Normalization for SPD Neural Networks". *Advances in Neural Information Processing Systems 32*. Ed. by H. Wal-

lach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., pp. 15489–15500.

Cabanes, Yann, Frédéric Barbaresco, Marc Arnaudon, and Jérémie Bigot (Nov. 2019). "Unsupervised Machine Learning for Pathological Radar Clutter Clustering: The P-Mean-Shift Algorithm". *C&ESAR 2019*. Rennes, France.

Cartan, Elie (1926). "Sur une classe remarquable d'espaces de Riemann". *Bulletin de la Société Mathématique de France* 54, pp. 214–264. DOI: 10.24033/bsmf.1105.

— (1930). *La théorie des groupes finis et continus et l'analysis situs*. Vol. 42. Mémorial des sciences mathématiques. Gauthier-Villars et cie.

Cendra, Hernán, Darryl D. Holm, Jerrold E. Marsden, and Tudor S. Ratiu (1998). "Lagrangian Reduction, the Euler-Poincaré Equations, and Semidirect Products". *American Mathematical Society Translations* 186.1, pp. 1–25. ISSN: 0065-9290.

Censi, Andrea (2012). *PyGeometry: Library for Handling Various Differentiable Manifolds*.

Charon, Nicolas, Benjamin Charlier, Joan A. Glaunès, Pietro Gori, and Pierre Roussillon (2020). "Fidelity Metrics between Curves and Surfaces: Currents, Varifolds, and Normal Cycles". *Riemannian Geometric Statistics in Medical Image Analysis*. Ed. by Xavier Pennec, Stefan Sommer, and Tom Fletcher. Academic Press, pp. 441–477. ISBN: 978-0-12-814725-2. DOI: 10.1016/B978-0-12-814725-2.00021-2.

Cheeger, Jeff and David Ebin (1975). *Comparison Theorems in Riemannian Geometry*. Vol. 9. North-holland Amsterdam.

Chessa, Massimo and Alessandro Giamberti, eds. (2012). *The Right Ventricle in Adults with Tetralogy of Fallot*. Mailand: Springer-Verlag. ISBN: 978-88-470-2357-4. DOI: 10.1007/978-88-470-2358-1.

Chevallier, Emmanuel and Nicolas Guigui (2020). "A Bi-Invariant Statistical Model Parametrized by Mean and Covariance on Rigid Motions". *Entropy* 22.4, p. 432. DOI: 10.3390/e22040432.

Chikuse, Yasuko (2003). *Statistics on Special Manifolds*. Lecture Notes in Statistics. New York: Springer-Verlag. ISBN: 978-0-387-00160-9. DOI: 10.1007/978-0-387-21540-2.

Cohen, Taco S, Mario Geiger, and Maurice Weiler (2019). "A General Theory of Equivariant CNNs on Homogeneous Spaces". *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc.

Cury, Claire, Joan A. Glaunès, Roberto Toro, Marie Chupin, Gunter Schumann, et al. (2018). "Statistical Shape Analysis of Large Datasets Based on Diffeomorphic Iterative Centroids". *Frontiers in Neuroscience* 12, p. 803. ISSN: 1662-453X. DOI: 10.3389/fnins.2018.00803.

Cury, Claire, Marco Lorenzi, David Cash, Jennifer Nicholas, Alexandre Routier, et al. (Sept. 2016). "Spatio-Temporal Shape Analysis of Cross-Sectional Data for Detection of Early Changes in Neurodegenerative Disease". *SeSAMI 2016 - First International Workshop Spectral and Shape Analysis in Medical Imaging*. Vol. 10126. LNCS. Springer, pp. 63–75. DOI: 10.1007/978-3-319-51237-2_6.

David, Paul and Weiqing Gu (2019). "A Riemannian Structure for Correlation Matrices". *Operators and Matrices* 13.3, pp. 607–627.

Debavelaere, Vianney, Alexandre Bône, Stanley Durrleman, Stéphanie Allassonnière, and for the Alzheimer's Disease Neuroimaging Initiative (Oct. 2019). "Clustering of Longitudinal Shape Data Sets Using Mixture of Separate or Branching Trajectories". *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*. Springer, Cham, pp. 66–74. DOI: 10.1007/978-3-030-32251-9_8.

Dedieu, Jean-Pierre and Dmitry Nowicki (Aug. 2005). "Symplectic Methods for the Approximation of the Exponential Map and the Newton Iteration on Riemannian Submanifolds". *Journal of Complexity* 21.4, pp. 487–501. ISSN: 0885064X. DOI: 10.1016/j.jco.2004.09.010.

Dhingra, Bhuwan, Christopher Shallue, Mohammad Norouzi, Andrew Dai, and George Dahl (2018). "Embedding Text in Hyperbolic Spaces". *Proceedings of the Twelfth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-12)*. New Orleans, Louisiana, USA: Association for Computational Linguistics, pp. 59–69. DOI: 10.18653/v1/W18-1708.

Di Folco, Maxime, Patrick Clarysse, Pamela Moceri, and Nicolas Duchateau (2019). "Learning Interactions between Cardiac Shape and Deformation: Application to Pulmonary Hypertension". *10th International Statistical Atlases and Computational Modeling of the Heart (STACOM) Workshop, Held in Conjunction with MICCAI 2019*. Lecture Notes in Computer Science. Shenzen, China.

Di Folco, Maxime, Nicolas Guigui, Patrick Clarysse, Pamela Moceri, and Nicolas Duchateau (June 2021). "Investigation of the Impact of Normalization on the Study of Interactions Between Myocardial Shape and Deformation". *Functional Imaging and Modeling of the Heart*. Springer, Cham, pp. 223–231. DOI: 10.1007/978-3-030-78710-3_22.

Diaconis, Persi, Susan Holmes, and Mehrdad Shahshahani (Jan. 2013). "Sampling from a Manifold". *Advances in Modern Statistical Theory and Applications: A Festschrift in honor of Morris L. Eaton*, pp. 102–125. DOI: 10.1214/12-IMSCOLL1006.

Dryden, Ian L. and Kanti V. Mardia (Sept. 2016). *Statistical Shape Analysis: With Applications in R*. John Wiley & Sons. ISBN: 978-0-470-69962-1.

Duchateau, Nicolas, Mathieu De Craene, Gemma Piella, Etelvino Silva, Adelina Doltra, et al. (June 2011). "A Spatiotemporal Statistical Atlas of Motion for the Quantification of Abnormal Myocardial Tissue Velocities". *Medical Image Analysis* 15.3, pp. 316–328. ISSN: 1361-8423. DOI: 10.1016/j.media.2010.12.006.

Durrleman, Stanley (Mar. 2010). "Statistical Models of Currents for Measuring the Variability of Anatomical Curves, Surfaces and Their Evolution". PhD thesis. Université Nice Sophia Antipolis.

Durrleman, Stanley, Stéphanie Allassonnière, and Sarang Joshi (Jan. 2013). "Sparse Adaptive Parameterization of Variability in Image Ensembles". *International*

Journal of Computer Vision 101.1, pp. 161–183. ISSN: 1573-1405. DOI: 10.1007/s11263-012-0556-1.

Durrleman, Stanley, Xavier Pennec, Alain Trouvé, Guido Gerig, and Nicholas Ayache (Sept. 2009). "Spatiotemporal Atlas Estimation for Developmental Delay Detection in Longitudinal Datasets". *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2009*. Springer, Berlin, Heidelberg, pp. 297–304. DOI: 10.1007/978-3-642-04268-3_37.

Durrleman, Stanley, Marcel Prastawa, Nicolas Charon, Julie R. Korenberg, Sarang Joshi, et al. (Nov. 2014). "Morphometry of Anatomical Shape Complexes with Dense Deformations and Sparse Parameters". *NeuroImage* 101, pp. 35–49. ISSN: 1053-8119. DOI: 10.1016/j.neuroimage.2014.06.043.

Eaton, Morris L. (1983). *Multivariate Statistics: A Vector Space Approach*. Probability and Statistics Series. Wiley. ISBN: 978-0-471-02776-8.

Ehlers, Jürgen, Felix A. E. Pirani, and Alfred Schild (1972). "The Geometry of Free Fall and Light Propagation". *General Relativity: Papers in Honour of J. L. Synge*. Ed. by Lochlainn O'Raifeartaigh. Oxford : Clarendon Press, pp. 63–84. ISBN: 978-0-19-851126-7.

Feragen, Aasa, Megan Owen, Jens Petersen, Mathilde M. W. Wille, Laura H. Thomsen, et al. (2013). "Tree-Space Statistics and Approximations for Large-Scale Analysis of Anatomical Trees". *Information Processing in Medical Imaging: Proceedings of the ... Conference* 23, pp. 74–85. ISSN: 1011-2499. DOI: 10.1007/978-3-642-38868-2_7.

Ferraro, Alessandra M., Adi Adar, Sunil J. Ghelani, Lynn A. Sleeper, Philip T. Levy, et al. (May 2020). "Speckle Tracking Echocardiographically-Based Analysis of Ventricular Strain in Children: An Intervendor Comparison". *Cardiovascular Ultrasound* 18, p. 15. ISSN: 1476-7120. DOI: 10.1186/s12947-020-00199-x.

Fey, Matthias and Jan E. Lenssen (Apr. 2019). "Fast Graph Representation Learning with PyTorch Geometric". arXiv: 1903.02428.

Fishbaugh, James, Stanley Durrleman, Marcel Prastawa, and Guido Gerig (July 2017). "Geodesic Shape Regression with Multiple Geometries and Sparse Parameters". *Medical Image Analysis* 39, pp. 1–17. ISSN: 1361-8415. DOI: 10.1016/j.media.2017.03.008.

Fletcher, 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, pp. 995–1005.

Fréchet, Maurice (1948). "Les éléments aléatoires de nature quelconque dans un espace distancié". *Annales de l'institut Henri Poincaré* 10.4, pp. 215–310.

Freifeld, Oren, Søren Hauberg, and Michael J. Black (2014). "Model Transport: Towards Scalable Transfer Learning on Manifolds". *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 1378–1385.

Gallier, Jean and Jocelyn Quaintance (2020). *Differential Geometry and Lie Groups: A Computational Perspective*. Geometry and Computing. Springer International Publishing. ISBN: 978-3-030-46039-6. DOI: 10.1007/978-3-030-46040-2.

Garba, Maryam K., Tom M. Nye, Jonas Lueg, and Stephan Huckemann (Feb. 2021). "Information Geometry for Phylogenetic Trees". *Journal of Mathematical Biology* 82.3, p. 19. ISSN: 1432-1416. DOI: 10.1007/s00285-021-01553-x.

Gavrilov, Aleksei Vladimirovich (2006). "Algebraic Properties of Covariant Derivative and Composition of Exponential Maps". *Siberian Adv. Math.* 16.3, pp. 54–70.

— (Jan. 2007). "The Double Exponential Map and Covariant Derivation". *Siberian Mathematical Journal* 48.1, pp. 56–61. ISSN: 1573-9260. DOI: 10.1007/s11202-007-0006-4.

— (Jan. 2013). "The Affine Connection in the Normal Coordinates". *Siberian Advances in Mathematics* 23.1, pp. 1–19. ISSN: 1934-8126. DOI: 10.3103/S105513441301001X.

Gawlik, Evan S. and Melvin Leok (June 2018). "Interpolation on Symmetric Spaces Via the Generalized Polar Decomposition". *Foundations of Computational Mathematics* 18.3, pp. 757–788. ISSN: 1615-3375, 1615-3383. DOI: 10.1007/s10208-017-9353-0.

Gay-Balmaz, François, Darryl D. Holm, David M. Meier, Tudor S. Ratiu, and François-Xavier Vialard (Aug. 2012). "Invariant Higher-Order Variational Problems II". *Journal of Nonlinear Science* 22.4, pp. 553–597. ISSN: 0938-8974, 1432-1467. DOI: 10.1007/s00332-012-9137-2. arXiv: 1112.6380.

Gromov, Mikhail (1987). "Hyperbolic Groups". *Essays in Group Theory*. New York, NY: Springer New York, pp. 75–263. DOI: 10.1007/978-1-4613-9586-7_3.

Guigui, Nicolas, Shuman Jia, Maxime Sermesant, and Xavier Pennec (Aug. 2019). "Symmetric Algorithmic Components for Shape Analysis with Diffeomorphisms". *GSI 2019 - 4th Conference on Geometric Science of Information*. Vol. 11712. Lecture Notes in Computer Science. Springer, Cham, pp. 759–768. DOI: 10.1007/978-3-030-26980-7_79.

Guigui, Nicolas, Elodie Maignant, Alain Trouvé, and Xavier Pennec (July 2021). "Parallel Transport on Kendall Shape Spaces". *GSI 2021 - 5th Conference on Geometric Science of Information*. Vol. 12829. Lecture Notes in Computer Science. Springer, Cham, pp. 103–110. DOI: 10.1007/978-3-030-80209-7_12.

Guigui, Nicolas, Nina Miolane, and Xavier Pennec (2021). "A Hands-on Introduction to Riemannian Geometry and Geometric Statistics with Geomstats". *In preparation*.

Guigui, Nicolas, Pamela Moceri, Maxime Sermesant, and Xavier Pennec (Apr. 2021). "Cardiac Motion Modeling With Parallel Transport And Shape Splines". *ISBI 2021 - IEEE 18th International Symposium on Biomedical Imaging*, pp. 1394–1397. DOI: 10.1109/ISBI48211.2021.9433887.

Guigui, Nicolas and Xavier Pennec (July 2021a). "A Reduced Parallel Transport Equation on Lie Groups with a Left-Invariant Metric". *GSI 2021 - 5th Conference on Geometric Science of Information*. Vol. 12829. Lecture Notes in Computer Science. Springer, Cham, pp. 119–126. DOI: 10.1007/978-3-030-80209-7_14.

— (June 2021b). "Numerical Accuracy of Ladder Schemes for Parallel Transport on Manifolds". *Foundations of Computational Mathematics*. ISSN: 1615-3383. DOI: 10.1007/s10208-021-09515-x.

Hadj-Hamou, Mehdi, Marco Lorenzi, Nicholas Ayache, and Xavier Pennec (June 2016). "Longitudinal Analysis of Image Time Series with Diffeomorphic Deformations: A Computational Framework Based on Stationary Velocity Fields". *Frontiers in Neuroscience* 10, p. 236. ISSN: 1662-4548. DOI: 10.3389/fnins.2016.00236.

Hairer, Ernst, Gerhard Wanner, and Christian Lubich (2002). "Symplectic Integration of Hamiltonian Systems". *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Ed. by Ernst Hairer, Gerhard Wanner, and Christian Lubich. Springer Series in Computational Mathematics. Berlin, Heidelberg: Springer, pp. 167–208. ISBN: 978-3-662-05018-7. DOI: 10.1007/978-3-662-05018-7_6.

Hauberg, Søren (July 2018). "Directional Statistics with the Spherical Normal Distribution". *2018 21st International Conference on Information Fusion (FUSION)*, pp. 704–711. DOI: 10.23919/ICIF.2018.8455242.

— (Sept. 2019). "Only Bayes Should Learn a Manifold (on the Estimation of Differential Geometric Structure from Data)". arXiv: 1806.04994.

Hauberg, Søren, François Lauze, and Kim Steenstrup Pedersen (May 2013). "Unscented Kalman Filtering on Riemannian Manifolds". *Journal of Mathematical Imaging and Vision* 46.1, pp. 103–120. ISSN: 1573-7683. DOI: 10.1007/s10851-012-0372-9.

Helgason, Sigurdur (Feb. 1979). *Differential Geometry, Lie Groups, and Symmetric Spaces*. Academic Press. ISBN: 978-0-08-087396-1.

Hotz, Thomas and Stephan Huckemann (Feb. 2015). "Intrinsic Means on the Circle: Uniqueness, Locus and Asymptotics". *Annals of the Institute of Statistical Mathematics* 67.1, pp. 177–193. ISSN: 1572-9052. DOI: 10.1007/s10463-013-0444-7.

Hotz, Thomas, Sean Skwerer, Stephan Huckemann, Huiling Le, J. S. Marron, et al. (Dec. 2013). "Sticky Central Limit Theorems on Open Books". *The Annals of Applied Probability* 23.6, pp. 2238–2258. ISSN: 1050-5164, 2168-8737. DOI: 10.1214/12-AAP899.

Hou, Benjamin, Nina Miolane, Bishesh Khanal, Matthew C. Lee, Amir Alansary, et al. (2018). "Computing CNN Loss and Gradients for Pose Estimation with Riemannian Geometry". *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*. Ed. by Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger. Lecture Notes in Computer Science. Springer International Publishing, pp. 756–764. ISBN: 978-3-030-00928-1.

Huckemann, Stephan, Thomas Hotz, and Axel Munk (Apr. 2010a). "Intrinsic MANOVA for Riemannian Manifolds with an Application to Kendall's Space of Planar Shapes". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.4, pp. 593–603. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2009.117.

Huckemann, Stephan, Thomas Hotz, and Axel Munk (2010b). "Intrinsic Shape Analysis: Geodesic PCA for Riemannian Manifolds modulo Isometric Lie Group Actions". *Statistica Sinica* 20.1, pp. 1–58. ISSN: 10170405.

Hundrieser, Shayan, Benjamin Eltzner, and Stephan Huckemann (July 2021). "Finite Sample Smeariness of Fr\'echet Means and Application to Climate". arXiv: 2005.02321.

Karcher, Hermann (1977). "Riemannian Center of Mass and Mollifier Smoothing". *Communications on Pure and Applied Mathematics* 30.5, pp. 509–541. ISSN: 1097-0312. DOI: 10.1002/cpa.3160300502.

— (July 2014). "Riemannian Center of Mass and so Called Karcher Mean". arXiv: 1407.2087.

Kendall, Alex and Roberto Cipolla (2017). "Geometric Loss Functions for Camera Pose Regression with Deep Learning". *CVPR*. DOI: 10.1109/CVPR.2017.694.

Kendall, David G. (1984). "Shape Manifolds, Procrustean Metrics, and Complex Projective Spaces". *Bulletin of the London Mathematical Society* 16.2, pp. 81–121. ISSN: 1469-2120. DOI: 10.1112/blms/16.2.81.

Kendall, Wilfrid S. (1990). "Probability, Convexity, and Harmonic Maps with Small Image I: Uniqueness and Fine Existence". *Proceedings of the London Mathematical Society* s3-61.2, pp. 371–406. ISSN: 1460-244X. DOI: 10.1112/plms/s3-61.2.371.

Kendall, Wilfrid S. and Huiling Le (2009). "Statistical Shape Theory". *New Perspectives in Stochastic Geometry*, pp. 348–373.

Kheyfets, Arkady, Warner A. Miller, and Gregory A. Newton (Dec. 2000). "Schild's Ladder Parallel Transport Procedure for an Arbitrary Connection". *International Journal of Theoretical Physics* 39.12, pp. 2891–2898. ISSN: 1572-9575. DOI: 10.1023/A:1026473418439.

Kim, Kwang-Rae, Ian L. Dryden, Huiling Le, and Katie E. Severn (Dec. 2020). "Smoothing Splines on Riemannian Manifolds, with Applications to 3D Shape Space". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*. ISSN: 1467-9868.

Kleijn, Sebastiaan A., Mohamed F. A. Aly, Caroline B. Terwee, Albert C. van Rossum, and Otto Kamp (Mar. 2011). "Three-Dimensional Speckle Tracking Echocardiography for Automatic Assessment of Global and Regional Left Ventricular Function Based on Area Strain". *Journal of the American Society of Echocardiography: Official Publication of the American Society of Echocardiography* 24.3, pp. 314–321. ISSN: 1097-6795. DOI: 10.1016/j.echo.2011.01.014.

Kobayashi, Shoshichi and Katsumi Nomizu (Feb. 1996a). *Foundations of Differential Geometry, Volume 1*. Wiley. ISBN: 978-0-471-15733-5.

— (Feb. 1996b). *Foundations of Differential Geometry, Volume 2*. Wiley. ISBN: 978-0-471-15732-8.

Kochurov, Maxim (2019). *Geoopt: Riemannian Adaptive Optimization Methods with Pytorch Optim*.

Kolev, Boris (Jan. 2004). "Lie Groups and Mechanics: An Introduction". *Journal of Nonlinear Mathematical Physics* 11.4, pp. 480–498. ISSN: 1402-9251, 1776-0852. DOI: 10.2991/jnmp.2004.11.4.5. arXiv: math-ph/0402052.

Krioukov, Dmitri, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, and Marián Boguñá (Sept. 2010). "Hyperbolic Geometry of Complex Networks". *Physical Review E* 82.3, p. 036106. DOI: 10.1103/PhysRevE.82.036106.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey Hinton (2012). "ImageNet Classification with Deep Convolutional Neural Networks". *Advances in Neural Information Processing Systems*. Vol. 25. Curran Associates, Inc.

Kühnel, Line and Stefan Sommer (Sept. 2017). "Computational Anatomy in Theano". *Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics*, pp. 164–176. DOI: 10.1007/978-3-319-67675-3_15.

Kühnel, Line, Stefan Sommer, and Alexis Arnaudon (Sept. 2019). "Differential Geometry and Stochastic Dynamics with Deep Learning Numerics". *Applied Mathematics and Computation* 356, pp. 411–437. ISSN: 0096-3003. DOI: 10.1016/j.amc.2019.03.044.

Lafontaine, Jacques, Sylvestre Gallot, and Dominique Hulin (2004). *Riemannian Geometry*. Springer Verlag.

Le, Huiling and David G. Kendall (1993). "The Riemannian Structure of Euclidean Shape Spaces: A Novel Environment for Statistics". *The Annals of Statistics* 21.3, pp. 1225–1271. ISSN: 0090-5364.

Le Brigant, Alice, Nicolas Guigui, Sana Rebbah, and Stéphane Puechmorel (Jan. 2021). "Classifying Histograms of Medical Data Using Information Geometry of Beta Distributions". *IFAC-PapersOnLine*. MTNS 2020 - 24th International Symposium on Mathematical Theory of Networks and Systems 54.9, pp. 514–520. ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2021.06.110.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton (May 2015). "Deep Learning". *Nature* 521.7553, pp. 436–444. ISSN: 1476-4687. DOI: 10.1038/nature14539.

Lorenzi, Marco, Nicholas Ayache, and Xavier Pennec (2011). "Schild's Ladder for the Parallel Transport of Deformations in Time Series of Images". *Information Processing in Medical Imaging*. Ed. by Gábor Székely and Horst K. Hahn. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 463–474. ISBN: 978-3-642-22092-0. DOI: 10.1007/978-3-642-22092-0_38.

Lorenzi, Marco and Xavier Pennec (Sept. 2011). "Geodesics, Parallel Transport & One-Parameter Subgroups for Diffeomorphic Image Registration". *Proceedings of the 3rd Workshop on Math Fund of Comp Anat*. Ed. by Xavier Pennec, Sarang Joshi, and Mads Nielsen. Toronto, Canada, pp. 64–74.

— (Sept. 2014). "Efficient Parallel Transport of Deformations in Time Series of Images: From Schild to Pole Ladder". *Journal of Mathematical Imaging and Vision* 50.1, pp. 5–17. ISSN: 1573-7683. DOI: 10.1007/s10851-013-0470-3.

Lorenzi, Marco, Xavier Pennec, Giovanni B. Frisoni, and Nicholas Ayache (Jan. 2015). "Disentangling Normal Aging from Alzheimer's Disease in Structural Magnetic Resonance Images". *Neurobiology of Aging* 36, S42. DOI: 10.1016/j.neurobiolaging.2014.07.046.

Louis, Maxime (Oct. 2019). "Computational and Statistical Methods for Trajectory Analysis in a Riemannian Geometry Setting". PhD thesis. Sorbonnes universités.

Louis, Maxime, Alexandre Bône, Benjamin Charlier, and Stanley Durrleman (Nov. 2017). "Parallel Transport in Shape Analysis : A Scalable Numerical Scheme". *Geometric Science of Information*. Paris, France.

Louis, Maxime, Benjamin Charlier, Paul Jusselin, Susovan Pal, and Stanley Durrleman (2018). "A Fanning Scheme for the Parallel Transport Along Geodesics on Riemannian Manifolds". *SIAM Journal on Numerical Analysis* 56.4, pp. 2563–2584. DOI: 10.1137/17M1130617.

Louis, Maxime, Raphaël Couronné, Igor Koval, Benjamin Charlier, and Stanley Durrleman (June 2019). "Riemannian Geometry Learning for Disease Progression Modelling". *Information Processing in Medical Imaging*. Springer, Cham, pp. 542–553. DOI: 10.1007/978-3-030-20351-1_42.

Mardia, Kanti V. and Peter E. Jupp (Sept. 2009). *Directional Statistics*. John Wiley & Sons. ISBN: 978-0-470-31781-5.

Marle, Charles-Michel (May 2005). "The Works of Charles Ehresmann on Connections: From Cartan Connections to Connections on Fibre Bundles". *Geometry and Topology of Manifolds*. Vol. 76. Bedlewo, Poland, pp. 65–86.

Marsden, Jerrold E. and Tudor S. Ratiu (2009). "Mechanical Systems: Symmetries and Reduction". *Encyclopedia of Complexity and Systems Science*. Ed. by Robert A. Meyers. New York, NY: Springer, pp. 5482–5510. ISBN: 978-0-387-30440-3. DOI: 10.1007/978-0-387-30440-3_326.

McAlpin, John Harris (1965). *Infinite Dimensional Manifolds and Morse Theory*. Columbia University.

Meghwanshi, Mayank, Pratik Jawanpuria, Anoop Kunchukuttan, Hiroyuki Kasai, and Bamdev Mishra (2018). *McTorch, a Manifold Optimization Library for Deep Learning*.

Micheli, Mario and Joan A. Glaunès (2014). "Matrix-Valued Kernels for Shape Deformation Analysis". *Geometry, Imaging and Computing* 1.1, pp. 57–139. ISSN: 2328-8884. DOI: 10.4310/GIC.2014.v1.n1.a2.

Miller, Michael I., Alain Trouvé, and Laurent Younes (Dec. 2015). "Hamiltonian Systems and Optimal Control in Computational Anatomy: 100 Years Since D'Arcy Thompson". *Annual Review of Biomedical Engineering* 17.1, pp. 447–509. ISSN: 1523-9829, 1545-4274. DOI: 10.1146/annurev-bioeng-071114-040601.

Milnor, John (Sept. 1976). "Curvatures of Left Invariant Metrics on Lie Groups". *Advances in Mathematics* 21.3, pp. 293–329. ISSN: 00018708. DOI: 10.1016/S0001-8708(76)80002-3.

— (1984). "Remarks on Infinite-Dimensional Lie Groups". *Relativity, groups and topology. 2*.

Miolane, Nina, Matteo Caorsi, Umberto Lupo, Marius Guerard, Nicolas Guigui, et al. (Aug. 2021). "ICLR 2021 Challenge for Computational Geometry & Topology: Design and Results". arXiv: 2108.09810.

Miolane, Nina, Nicolas Guigui, Alice Le Brigant, Johan Mathe, Benjamin Hou, et al. (Dec. 2020). "Geomstats: A Python Package for Riemannian Geometry in Machine Learning". *Journal of Machine Learning Research* 21.223, pp. 1–9.

Miolane, Nina, Nicolas Guigui, Hadi Zaatiti, Christian Shewmake, Hatem Hajri, et al. (July 2020). "Introduction to Geometric Learning in Python with Geomstats". *SciPy 2020 - 19th Python in Science Conference*, p. 48. DOI: 10.25080/Majora-342d178e-007.

Miolane, Nina and Xavier Pennec (Apr. 2015). "Computing Bi-Invariant Pseudo-Metrics on Lie Groups for Consistent Statistics". *Entropy* 17.4, pp. 1850–1881. DOI: 10.3390/e17041850.

Misner, Charles W., Kip S. Thorne, and John Archibald Wheeler (1973). *Gravitation*. Ed. by Freeman. Princeton University Press. ISBN: 978-0-691-17779-3.

Moceri, Pamela, Nicolas Duchateau, Delphine Baudouy, Elie-Dan Schouver, Sylvie Leroy, et al. (Apr. 2018). "Three-Dimensional Right-Ventricular Regional Deformation and Survival in Pulmonary Hypertension". *European Heart Journal - Cardiovascular Imaging* 19.4, pp. 450–458. ISSN: 2047-2404. DOI: 10.1093/ehjci/jex163.

Moceri, Pamela, Nicolas Duchateau, Stephane Gillon, Lolita Jaunay, Delphine Baudouy, et al. (2020). "3D Right Ventricular Shape and Strain in Congenital Heart Disease Patients with Right Ventricular Chronic Volume Loading". *European Heart Journal - Cardiovascular Imaging*.

Munthe-Kaas, Hans, Reinout Quispel, and Antonella Zanna (Mar. 2014). "Symmetric Spaces and Lie Triple Systems in Numerical Analysis of Differential Equations". *BIT Numerical Mathematics* 54.1, pp. 257–282. ISSN: 1572-9125. DOI: 10.1007/s10543-014-0473-5.

Munthe-Kaas, Hans and Olivier Verdier (Aug. 2016). "Integrators on Homogeneous Spaces: Isotropy Choice and Connections". *Foundations of Computational Mathematics* 16.4, pp. 899–939. ISSN: 1615-3375, 1615-3383. DOI: 10.1007/s10208-015-9267-7. arXiv: 1402.6981.

Munthe-Kaas, Hans and Antonella Zanna (1997). "Numerical Integration of Differential Equations on Homogeneous Manifolds". *Foundations of Computational Mathematics*. Ed. by Felipe Cucker and Michael Shub. Berlin, Heidelberg: Springer, pp. 305–315. ISBN: 978-3-642-60539-0. DOI: 10.1007/978-3-642-60539-0_24.

Myers, Sumner B. and Norman E. Steenrod (1939). "The Group of Isometries of a Riemannian Manifold". *Annals of Mathematics* 40.2, pp. 400–416. ISSN: 0003-486X. DOI: 10.2307/1968928.

Nava-Yazdani, Esfandiar, Hans-Christian Hege, Tim J. Sulllivan, and Christoph von Tycowicz (May 2020). "Geodesic Analysis in Kendall's Shape Space with Epidemiological Applications". *Journal of Mathematical Imaging and Vision* 62.4, pp. 549–559. ISSN: 1573-7683. DOI: 10.1007/s10851-020-00945-w.

Nickel, Maximillian and Douwe Kiela (2017). "Poincaré Embeddings for Learning Hierarchical Representations". *Advances in Neural Information Processing Sys-*

tems 30. Ed. by I Guyon, U V Luxburg, S Bengio, H Wallach, R Fergus, et al. Curran Associates, Inc., pp. 6338–6347.

Niethammer, Marc, Roland Kwitt, and François-Xavier Vialard (2019). "Metric Learning for Image Registration". *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8463–8472.

Niethammer, Marc and François-Xavier Vialard (2013). "Riemannian Metrics for Statistics on Shapes : Parallel Transport and Scale Invariance". *Proceedings of Miccai Workshop, MFCA*.

Nomizu, Katsumi (1954). "Invariant Affine Connections on Homogeneous Spaces". *American Journal of Mathematics* 76.1, pp. 33–65. ISSN: 0002-9327. DOI: 10.2307/2372398.

O'Neill, Barrett (Dec. 1966). "The Fundamental Equations of a Submersion". *Michigan Mathematical Journal* 13.4, pp. 459–469. ISSN: 0026-2285, 1945-2365. DOI: 10.1307/mmj/1028999604.

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, et al. (2019). "PyTorch: An Imperative Style, High-Performance Deep Learning Library". *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., pp. 8024–8035.

Paulin, Frédéric (2007). *Géométrie différentielle élémentaire*.

— (2014). *Géométrie Riemanienne, cours de second année de mastère*.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, et al. (2011). "Scikit-Learn: Machine Learning in Python". *Journal of Machine Learning Research* 12.85, pp. 2825–2830. ISSN: 1533-7928.

Peel, David, William J Whiten, and Geoffrey J McLachlan (Mar. 2001). "Fitting Mixtures of Kent Distributions to Aid in Joint Set Identification". *Journal of the American Statistical Association* 96.453, pp. 56–63. ISSN: 0162-1459. DOI: 10.1198/016214501750332974.

Pennec, Xavier (July 2006). "Intrinsic Statistics on Riemannian Manifolds: Basic Tools for Geometric Measurements". *Journal of Mathematical Imaging and Vision* 25.1, pp. 127–154. ISSN: 0924-9907, 1573-7683. DOI: 10.1007/s10851-006-6228-4.

— (May 2018). "Parallel Transport with Pole Ladder: A Third Order Scheme in Affine Connection Spaces Which Is Exact in Affine Symmetric Spaces".

— (June 2019). "Curvature Effects on the Empirical Mean in Riemannian and Affine Manifolds: A Non-Asymptotic High Concentration Expansion in the Small-Sample Regime".

— (Jan. 2020). "Manifold-Valued Image Processing with SPD Matrices". *Riemannian Geometric Statistics in Medical Image Analysis*. Ed. by Xavier Pennec, Stefan Sommer, and Tom Fletcher. Academic Press, pp. 75–134. ISBN: 978-0-12-814725-2. DOI: 10.1016/B978-0-12-814725-2.00010-8.

Pennec, Xavier and Vincent Arsigny (May 2012). "Exponential Barycenters of the Canonical Cartan Connection and Invariant Means on Lie Groups". *Matrix Information Geometry*. Springer, pp. 123–168. ISBN: 978-3-642-30231-2.

Pennec, Xavier, Pierre Fillard, and Nicholas Ayache (Jan. 2006). "A Riemannian Framework for Tensor Computing". *International Journal of Computer Vision* 66.1, pp. 41–66. ISSN: 0920-5691, 1573-1405. DOI: 10.1007/s11263-005-3222-z.

Pennec, Xavier, Stefan Sommer, and Tom Fletcher, eds. (2020). *Riemannian Geometric Statistics in Medical Image Analysis*. Vol. 3. The Elsevier and MICCAI Society Book Series. Elsevier. ISBN: 978-0-12-814725-2. DOI: 10.1016/C2017-0-01561-6.

Peyrat, Jean-Marc, Hervé Delingette, Maxime Sermesant, Chenyang Xu, and Nicholas Ayache (July 2010). "Registration of 4D Cardiac CT Sequences Under Trajectory Constraints With Multichannel Diffeomorphic Demons". *IEEE Transactions on Medical Imaging* 29.7, pp. 1351–1368. ISSN: 1558-254X. DOI: 10.1109/TMI.2009.2038908.

Postnikov, Mikhail Mikhailovich (2001). *Geometry VI: Riemannian Geometry*. Encyclopaedia of Mathematical Sciences, Geometry. Berlin Heidelberg: Springer-Verlag. ISBN: 978-3-540-41108-6.

Qiu, Anqi, Marilyn Albert, Laurent Younes, and Michael I. Miller (Mar. 2009). "Time Sequence Diffeomorphic Metric Mapping and Parallel Transport Track Time-Dependent Shape Changes". *NeuroImage*. Mathematics in Brain Imaging 45.1, Supplement 1, S51–S60. ISSN: 1053-8119. DOI: 10.1016/j.neuroimage.2008.10.039.

Qiu, Anqi, Laurent Younes, Michael I. Miller, and John G. Csernansky (Mar. 2008). "Parallel Transport in Diffeomorphisms Distinguishes the Time-Dependent Pattern of Hippocampal Surface Deformation Due to Healthy Aging and the Dementia of the Alzheimer's Type". *NeuroImage* 40.1, pp. 68–76. ISSN: 1053-8119. DOI: 10.1016/j.neuroimage.2007.11.041.

Redish, David A., Erich Kummerfeld, Rebecca L. Morris, and Alan C. Love (May 2018). "Reproducibility failures are essential to scientific inquiry". *Proceedings of the National Academy of Sciences of the United States of America* 115.20, pp. 5042–5046. ISSN: 0027-8424. DOI: 10.1073/pnas.1806370115.

Riemann, Bernhard (1868). "Über Die Hypothesen, Welche Der Geometrie Zu Grunde Liegen. (Mitgetheilt Durch r. Dedekind)". *Abhandlungen der Königlichen Gesellschaft der Wissenschaften in Göttingen* 13, pp. 133–152.

— (May 1873). "On the Hypotheses Which Lie at the Bases of Geometry". *Nature* 8.184, pp. 36–37. ISSN: 1476-4687. DOI: 10.1038/008036a0.

Said, Salem, Hatem Hajri, Lionel Bombrun, and Baba C. Vemuri (Feb. 2018). "Gaussian Distributions on Riemannian Symmetric Spaces: Statistical Learning With Structured Covariance Matrices". *IEEE Transactions on Information Theory* 64.2, pp. 752–772. ISSN: 0018-9448. DOI: 10.1109/TIT.2017.2713829.

Sampath, Shameem A., Sandra Lewis, Matteo Fosco, and Domenico Tigani (Apr. 2015). "Trabecular Orientation in the Human Femur and Tibia and the Relationship with Lower-Limb Alignment for Patients with Osteoarthritis of the Knee". *Journal of Biomechanics* 48.6, pp. 1214–1218. ISSN: 0021-9290. DOI: 10.1016/j.jbiomech.2015.01.028.

Sanz, Javier, Damián Sánchez-Quintana, Eduardo Bossone, Harm J. Bogaard, and Robert Naeije (Apr. 2019). "Anatomy, Function, and Dysfunction of the Right Ventricle: JACC State-of-the-Art Review". *Journal of the American College of Cardiology* 73.12, pp. 1463–1482. ISSN: 0735-1097, 1558-3597. DOI: 10.1016/j.jacc.2018.12.076.

Schiratti, Jean-Baptiste, Stéphanie Allassonnière, Olivier Colliot, and Stanley Durrleman (2015). "Learning Spatiotemporal Trajectories from Manifold-Valued Longitudinal Data". *Advances in Neural Information Processing Systems* 28.

— (2017). "A Bayesian Mixed-Effects Model to Learn Trajectories of Changes from Repeated Manifold-Valued Observations". *Journal of Machine Learning Research* 18.133, p. 33.

Schmid, Rudolf (Jan. 2004). "Infinite Dimentional Lie Groups with Applications to Mathematical Physics". *Journal of Geometry and Symmetry in Physics* 1, pp. 54–120. ISSN: 1312-5192, 1314-5673. DOI: 10.7546/jgsp-1-2004-54-120.

Sivera, Raphaël, Nicolas Capet, Valeria Manera, Roxane Fabre, Marco Lorenzi, et al. (Oct. 2020). "Voxel-Based Assessments of Treatment Effects on Longitudinal Brain Changes in the Multidomain Alzheimer Preventive Trial Cohort". *Neurobiology of Aging* 94, p. 50. DOI: 10.1016/j.neurobiolaging.2019.11.020.

Sivera, Raphaël, Hervé Delingette, Marco Lorenzi, Xavier Pennec, and Nicholas Ayache (Sept. 2019). "A Model of Brain Morphological Changes Related to Aging and Alzheimer's Disease from Cross-Sectional Assessments". *NeuroImage* 198, p. 255. DOI: 10.1016/j.neuroimage.2019.05.040.

Smirnov, Oleg (July 2021). "TensorFlow RiemOpt: A Library for Optimization on Riemannian Manifolds". arXiv: 2105.13921.

Srivastava, Anuj, Eric Klassen, Shantanu H. Joshi, and Ian H. Jermyn (July 2011). "Shape Analysis of Elastic Curves in Euclidean Spaces". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.7, pp. 1415–1428. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2010.184.

Terras, Audrey (1988). *Harmonic Analysis on Symmetric Spaces and Applications, Vol. II*. New York: Springer-Verlag.

Thanwerdas, Yann and Xavier Pennec (July 2021a). "Geodesics and Curvature of the Quotient-Affine Metrics on Full-Rank Correlation Matrices". *GSI 2021 - 5th Conference on Geometric Science of Information*. Proceedings of Geometric Science of Information. Paris, France.

— (Sept. 2021b). "O(n)-Invariant Riemannian Metrics on SPD Matrices".

Thompson, D'Arcy Wentworth (1917). *On Growth and Form*. Ed. by John Tyler Bonner. Canto. Cambridge: Cambridge University Press. DOI: 10.1017/CBO9781107325852.

Townsend, James, Niklas Koep, and Sebastian Weichwald (2016). "Pymanopt: A Python Toolbox for Optimization on Manifolds Using Automatic Differentiation". *Journal of Machine Learning Research* 17.137, pp. 1–5. DOI: 10.5555/2946645.3007090.

Trouvé, Alain and François-Xavier Vialard (2012). "Shape Splines and Stochastic Shape Evolutions: A Second Order Point of View". *Quarterly of Applied Mathematics* 70.2, pp. 219–251. ISSN: 0033-569X.

Varoquaux, Gaël, Flore Baronnet, Andreas Kleinschmidt, Pierre Fillard, and Bertrand Thirion (Sept. 2010). "Detection of Brain Functional-Connectivity Difference in Post-Stroke Patients Using Group-Level Covariance Modeling". *Medical Image Computing and Computer Added Intervention*. Springer. DOI: 10.1007/978-3-642-15705-9_25.

Vialard, François-Xavier and Laurent Risser (Sept. 2014). "Spatially-Varying Metric Learning for Diffeomorphic Image Registration: A Variational Framework". *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014*. Springer, Cham, pp. 227–234. DOI: 10.1007/978-3-319-10404-1_29.

Whitney, Hassler (1936). "Differentiable Manifolds". *Annals of Mathematics* 37.3, pp. 645–680. ISSN: 0003-486X. DOI: 10.2307/1968482.

Wood, Andrew T. A. (1994). "Simulation of the von Mises Fisher Distribution". *Communications in Statistics - Simulation and Computation* 23.1, pp. 157–164. DOI: 10.1080/03610919408813161.

Wynn, Kieran (2014). *PyQuaternions: A Fully Featured, Pythonic Library for Representing and Using Quaternions*.

Yair, Or, Mirela Ben-Chen, and Ronen Talmon (Apr. 2019). "Parallel Transport on the Cone Manifold of SPD Matrices for Domain Adaptation". *IEEE Transactions on Signal Processing*. Vol. 67, pp. 1797–1811. DOI: 10.1109/TSP.2019.2894801.

Younes, Laurent (Feb. 2007). "Jacobi Fields in Groups of Diffeomorphisms and Applications". *Quarterly of Applied Mathematics* 65.1, pp. 113–134. ISSN: 0033-569X, 1552-4485. DOI: 10.1090/S0033-569X-07-01027-5.

— (2019). *Shapes and Diffeomorphisms*. Vol. 171. Applied Mathematical Sciences. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN: 978-3-662-58495-8. DOI: 10.1007/978-3-662-58496-5.

Young, Alistair A. and Alejandro F. Frangi (2009). "Computational Cardiac Atlases: From Patient to Population and Back". *Experimental Physiology* 94.5, pp. 578–596. ISSN: 1469-445X. DOI: 10.1113/expphysiol.2008.044081.

Zefran, Milos, Vijay Kumar, and Christopher Croke (Aug. 1998). "On the Generation of Smooth Three-Dimensional Rigid Body Motions". *IEEE Transactions on Robotics and Automation* 14.4, pp. 576–589. ISSN: 1042296X. DOI: 10.1109/70.704225.

Zimmermann, Ralf (Jan. 2017). "A Matrix-Algebraic Algorithm for the Riemannian Logarithm on the Stiefel Manifold under the Canonical Metric". *SIAM Journal on Matrix Analysis and Applications* 38.2, pp. 322–342. ISSN: 0895-4798. DOI: 10.1137/16M1074485.