



HAL
open science

Bayesian Inference within the Framework of 2D and 3D Shape Analysis

Anis Fradi

► **To cite this version:**

Anis Fradi. Bayesian Inference within the Framework of 2D and 3D Shape Analysis. Machine Learning [cs.LG]. Université Clermont Auvergne; Université de Monastir (Tunisie), 2021. English. NNT : 2021UCFAC059 . tel-03633738

HAL Id: tel-03633738

<https://theses.hal.science/tel-03633738>

Submitted on 7 Apr 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITY OF CLERMONT AUVERGNE
UNIVERSITY OF MONASTIR

By

Anis FRADI

To obtain the degree of

DOCTOR OF UNIVERSITY

Specialty : **Computer Science**

Bayesian Inference within the Framework of 2D and 3D
Shape Analysis

Publicly defended on: July 07th, 2021 in front of the jury composed by

Pr. Julien JACQUES,	Lumière University Lyon 2, France	Reviewer
Pr. Sana LOUHICHI,	Grenoble Alpes University, France	Reviewer
Pr. Engelbert MEPHU NGUIFO,	University of Clermont Auvergne, France	President
Pr. Wen HUANG,	Xiamen University, China	Member
Pr. Chafik SAMIR,	University of Clermont Auvergne, France	Advisor
Pr. Leila BEN ABDELGHANI,	University of Monastir, Tunisia	Advisor

LIMOS - CNRS UMR 6158, 63178, Aubière, France
MAPSFA - LR11ES35, 4011, Hammam Sousse, Tunisia

Dedicace

I dedicate this work to:

My parents, my sisters, their husbands and their children: Zainouba, Fattouma, Aloulou and the younger niece Aryouma. I thank them for their patience, their moral and material support during these three years, without them I would not be where I am today.

My childhood friends Adem, Helmi and all my friends, especially Yassine Mkhinini, Mohamed Ayadi, Ala Mhala, Fatma Ben Khedher, Zaineb Samida and Salima Helali.

During this thesis, I lost two dear cousins Achraf and Mourad. May their souls rest in peace.

Clermont Ferrand, 2021

Anis FRADI

Acknowledgments

I address these few words of thanks to all people who have contributed directly or indirectly to this thesis.

First, I am indebted to my thesis supervisor Mr. Chafik Samir, Associate Professor at University of Clermont Auvergne, and member of LIMOS-CNRS laboratory for giving me the opportunity to join his team and to do my PhD. I am also grateful to him for the substantial time that he granted me, the nice discussions we had, his pedagogical and scientific qualities, his frankness and sympathy. I learned a lot from him and I address my gratitude for all this.

I would to express my deepest gratitude to my supervisor Mrs. Leila Ben Abdelghani, Professor at the University of Monastir, Tunisia and member of MAPSFA (LR11ES35) laboratory for her encouraging support to my work, her valuable suggestions and willingness to listen whose are major factors for the success of this thesis. Her energy and confidence have been essential for me.

I extend my thanks to Mrs. Anne Françoise Yao, Professor at University of Clermont Auvergne, France and member of CNRS-LMBP laboratory, for her involvement in the team. She created a vastly positive and enthusiastic working atmosphere.

I would like to thank my committee members, the reviewers Mrs. Sana Louhichi and Mr. Julien Jacques and furthermore the members Mr. Wen Huang and Mr. Engelbert Mephu Nguifo for their precious time, shared positive insight and guidance.

I would like also to show my sincerest gratitude to Mr. Mathias Bernanrd,

President of University of Clermont Auvergne and Mr. Hedi Bel Hadj Salah, President of the University of Monastir, for funding provided to this thesis under a joint supervision agreement between the two universities.

I would like to express my deepest appreciation to Mr. Mourad Baiou, Director of the LIMOS laboratory and Mr. Adel Daouas, Director of the MAPSFA laboratory, for the working conditions they provided for me.

I would like to express my special thanks to Mr. José Braga, Anthropobiology Professor at the University of Toulouse (Paul Sabatier), and Mr. Joshi Shantanu, Assistant Professor at the University of Florida for the collaborative works.

During those years, I had the pleasure of teaching. So I am thankful to all professors who accompanied me in my study for providing me expertise on every subject. Their enthusiasm and dedication to their students were truly inspiring.

I would also like to thank all members of LIMOS and IUT for their sympathies and friendships. I had a lot of fun to work with them. I can never forget the good moments with my colleagues Papa Mbaye, Yann Feunteun, Elliott Blot, Marouan Baklouti and Tien Tam Tran. They made my stay at the laboratory very pleasant.

My warmest thanks also to the members of MAPSFA, especially my colleagues Abdelhak Traiki, Mohamed Mabrouk, Foued Aloui, Mohamed Rahmani, Mohamed Louchaiech, Khalil Zahmoul, Amani Hammami, Maha Belhadj, Hadhami El Aini, Jihen Oueslati and Naoures Ayadi. We spent a great time together.

Abstract

The thesis is divided into two main parts: i) Nonparametric statistics on high-dimensional and functional spaces, and ii) Nonparametric statistics on Riemannian manifolds. In this part, we will summarize the major contributions of the thesis.

Nonparametric statistics on high-dimensional and functional spaces

In statistical learning, we introduce a new notion entitled: scalable Gaussian process classifier. The proposal is more general than the usual Gaussian process classifier for representing and classifying data lying on high-dimensional spaces. It is more advantageous for learning the hyper-parameters of the mapping (embedding) that maps initial data into a low-dimensional (feature) space and those of the Gaussian process classifier through its covariance function, jointly, with different optimization methods. The modified covariance function, depending on the embedding and operating on the feature space, is more expressive since the Euclidean metric is more informative in low-dimensional spaces. To summarize, our formulation takes care of non-linearity/high-correlation of data and increases the separability between them thanks to the Representer Theorem. In order to estimate the model's hyper-parameters we usually maximize the marginal likelihood. Unlike regression, the computation of the exact marginal likelihood remains difficult and even impossible in the classification case due to the discrete likelihoods. Thus, we introduce two methods to approximate the non-Gaussian posterior distribution by a Gaussian one in order to improve the efficiency and the scalability

of the Gaussian process.

For functional and even high-dimensional data, we also introduce the notion of Gaussian processes indexed by probability density functions. We will show how Gaussian processes can be defined into functional spaces, in particular that of density functions endowed with the Fisher-Rao metric. More precisely, we will extend the traditional methods of nonparametric statistics based on Gaussian processes from finite vectors in Euclidean spaces to constrained functions with Riemannian metrics. Our motivation is that several categories of observations can be represented by their density functions with more advantages than initial vector or functional inputs. This choice is very crucial for many reasons. First of all, density functions make the problem formulation more understood when identifying the initial vector inputs or functional data, which are hard to interpret, by their occurrences or their corresponding probabilities. Second, density functions improve the visualization of local data distributions. Finally, when dealing with high-dimensional datasets (set of repetitive features), we can visualize them using density functions which would be very helpful to explore the skewness of initial data.

Applications: Image classification (breast cancer/metallic boxes/growth charts /maize leaves/animal temperature) and video classification (violence detection).

[Nonparametric statistics on Riemannian manifolds](#)

In statistical learning on Riemannian manifold of curves, one of major problems is that of registration. For curve registration, we have to find the optimal deformation in terms of the best reparametrization function (or local distribution) between two curves. The space of reparametrization functions is a group of diffeomorphisms for the composition operation, which makes the optimization task quite complicated due to the structure of the group. In fact, there is no intuitive direction nor an underlying metric (or structure) in this group.

To handle these issues, we propose a new version of reparametrization functions. The main idea is based on the fact that any reparametrization can be viewed as an element of the manifold of cumulative distribution functions endowed with the Fisher-Rao metric, the only Riemannian metric invariant to reparametrizations. Then we can make the link with the Hilbert sphere endowed with the \mathbb{L}^2 metric for its simplicity and geometric advantages in statistics. Finally, we model the square-root of the probability density function, as an element of the Hilbert sphere, by a Gaussian process. Instead of estimating a reparametrization function directly in the group, we consider its truncated parametric version with coefficients belonging to the finite-dimensional unit sphere and resulting from the Loève expansion of the Gaussian process.

Given a finite set of observed curves, we are also interested in the curve clustering process in a Bayesian framework. We are able to find the sub-population means depending on their optimal local reparametrization functions. Compared to the state-of-the-art methods, the proposal has the advantage of computing the conditional probability that each curve belongs to a given sub-population.

A natural estimator of the unknown coefficients resulting from the Loève expansion is that maximizing the posterior density under spherical constraints. To find it we will consider the Hamiltonian Monte Carlo sampling. The samples are obtained when solving a system of differential equations describing the paths of Hamiltonian dynamics, controlling the position on the sphere and the velocity on the corresponding tangent space locally at each position, iteratively, until convergence.

Applications: Human cochlea clustering (male/female) and hominin cochlea clustering (paranthropus/gorilla/chimpanzee/australopithecus) discovered in South Africa.

Keywords: Nonparametric statistics; Bayesian inference; Gaussian processes; high-dimensional data; shape analysis of curves; Numerical optimization; MCMC sampling; HMC sampling; regression; classification; registration; Riemannian mani-

fold; Fisher-Rao metric, Hilbert sphere

Résumé

La thèse se décompose en deux parties principales: i) Statistiques non paramétriques sur les espaces en grande dimension et fonctionnels, et ii) Statistiques non paramétriques sur les variétés riemanniennes. Dans cette partie, nous allons résumer les contributions majeures de la thèse.

Statistiques non paramétriques sur les espaces en grande dimension et fonctionnels

Dans le domaine d'apprentissage statistique, nous introduisons une nouvelle notion intitulée: processus gaussien de classification évolutif. Le modèle proposé est plus général que le processus gaussien de classification standard pour représenter et classer des données appartenant à des espaces de grande dimension. Il a l'avantage d'apprendre les hyper-paramètres de la fonction qui transforme les données initiales sur un espace de dimension faible et ceux du processus gaussien de classification à travers sa fonction de covariance à la fois, avec plusieurs méthodes d'optimisation. La fonction de covariance modifiée, définie sur le nouveau espace des données transformées, est plus expressive car la métrique euclidienne devient plus informative. Pour résumer, notre formulation prend en considération la non-linéarité/forte corrélation des données et augmente la séparabilité entre elles grâce au Théorème du représentant. Afin d'estimer les hyper-paramètres du modèle proposé, nous maximisons la vraisemblance marginale. Contrairement à la régression, le calcul de la vraisemblance marginale exacte reste difficile et même impossible dans le cas de classification à cause des vraisemblances discrètes. Ainsi,

nous introduisons deux méthodes pour approximer une distribution a posteriori non gaussienne par une gaussienne afin d'améliorer l'efficacité et l'évolutivité du processus gaussien.

Pour les données fonctionnelles et même vectorielles en grande dimension, nous introduisons également la notion de processus gaussiens indexé par les fonctions de densité de probabilité. Nous montrerons comment les processus gaussiens peuvent être également définis sur des espaces fonctionnelles, en particulier celle de densités de probabilité muni de la métrique de Fisher-Rao. Plus précisément, nous étendrons les méthodes traditionnelles de statistiques non paramétriques par processus gaussiens de vecteurs finis dans les espaces euclidiens aux espaces des fonctions sous des contraintes munies des métriques riemanniennes. Notre motivation est que plusieurs catégories d'observations peuvent être représentées par des densités de probabilité avec plus d'avantages que des entrées vectorielles ou fonctionnelles brutes. Ce choix est très important pour plusieurs raisons. D'abord, les densités de probabilité permettent de simplifier la formulation du problème en identifiant les données vectorielles ou fonctionnelles initiales, qui sont difficiles à interpréter, par leurs occurrences ou leurs probabilités. Ensuite, les densités de probabilité améliorent la visualisation des distributions locales de données. Enfin, lorsqu'il s'agit des données fortement corrélées (caractéristiques répétitives) nous pouvons plutôt visualiser leurs densités de probabilité pour ajuster l'asymétrie des données initiales.

Applications: Classification d'images (cancer du sein/boîtes métalliques/courbes de croissance/feuilles de maïs/température des animaux) et des vidéos (détection de violence).

[Statistiques non paramétriques sur les variétés riemanniennes](#)

Dans le domaine d'apprentissage de courbes définies avec des structures riemanniennes, l'un des problèmes majeurs est celui de recalage. Pour recaler une

courbe par rapport à une autre, nous devons trouver la déformation optimale en terme de la meilleure fonction de reparamétrisation (ou distribution locale) entre les deux courbes. L'espace des fonctions de reparamétrisations est un groupe de difféomorphismes pour la loi de composition, ce qui rend la tâche d'optimisation assez compliquée à cause de la structure du groupe. En fait, il n'y a ni une direction intuitive ni une métrique (ou structure) sous-jacente dans ce groupe.

Pour résoudre ce problème, nous proposons une nouvelle version des fonctions de reparamétrisation. L'idée principale est basée sur le fait que toute fonction de reparamétrisation peut être vue comme un élément de la variété des fonctions de répartition munie de la métrique de Fisher-Rao, la seule métrique riemannienne invariante aux reparamétrisations. Ensuite, nous pouvons établir le lien avec la sphère de Hilbert munie de la métrique L^2 pour sa simplicité et ses avantages géométriques en statistiques. Enfin, nous modélisons la racine carrée de la fonction de densité de probabilité, comme un élément de la sphère de Hilbert, par un processus gaussien. Au lieu d'estimer une fonction de reparamétrisation directement dans le groupe, nous considérons sa version paramétrique tronquée avec des paramètres appartenant à la sphère unitaire de dimension finie et résultants de la décomposition de Loève du processus gaussien.

Étant donné un ensemble fini de courbes observées, nous nous intéressons également au regroupement non supervisé (clustering) de courbes dans un contexte bayésien. De plus, nous allons trouver les moyennes de toutes les sous-populations en fonction de leurs reparamétrisations locales optimales. Par rapport aux méthodes existantes, le modèle proposé a l'avantage de calculer la probabilité conditionnelle que chaque courbe appartienne à une sous-population donnée.

Un estimateur naturel des paramètres inconnus résultants de la décomposition de Loève est celui qui maximise la densité a posteriori sous des contraintes sphériques. Pour trouver cet estimateur nous considérerons l'échantillonnage par Hamiltonian Monte Carlo. Les simulations sont obtenues en résolvant un système d'équations différentielles décrivant les chemins de la dynamique hamiltonienne, contrôlant la position sur la sphère et la vitesse sur l'espace tangent associé locale-

ment en chaque position, d'une façon itérative, jusqu'à la convergence.

Applications: Regroupement des cochlées des humains (homme/femme) et celles des hominidés (paranthropus/gorille/chimpanzé/australopithèque) découvertes en Afrique du Sud.

Mots clés: Statistiques non paramétriques; inférence bayésienne; processus gaussiens; données en grande dimension; analyse de forme des courbes; optimisation numérique; échantillonnage par MCMC; échantillonnage par HMC; régression; classification; recalage; variété riemannienne; métrique de Fisher-Rao; sphère de Hilbert

Publications

We list submitted, revised and published papers during the thesis.

Published journal papers

1. **A. Fradi**, Y. Feunteun, C. Samir, M. Baklouti, F. Bachoc and J-M. Loubes. Bayesian regression and classification using Gaussian process priors indexed by probability density functions. *Journal of Information Sciences*-2020.
2. **A. Fradi**, C. Samir and F. Bachoc. A scalable approximate Bayesian inference for high-dimensional Gaussian processes. *Communication in Statistics - Theory and Methods*-2020.
3. **A. Fradi** and C. Samir. Bayesian cluster analysis for registration and clustering homogeneous subgroups in multidimensional functional data. *Communication in Statistics - Theory and Methods*-2020.
4. J. Braga, C. Samir, L. Risser, J. Dumoncel, D. Descouens, F. Thackeray, P. Balaresque, A. Oettlé, J-M. Loubes and **A. Fradi**. Cochlear shape reveals that the human organ of hearing is sex-typed from birth. *Scientific Reports*-2019.

Under review journal papers

1. **A. Fradi**, C. Samir, J. Braga, S.H. Joshi and J-M. Loubes. Nonparametric Bayesian regression and classification on manifolds, with applications to 3 D cochlear shapes. **Submitted** to: "IEEE Transactions on Image Processing".
2. **A. Fradi** and C. Samir. A new Bayesian framework for clustering and registration of vector-valued functions and surfaces. **Submitted** to: "Annals of the Institute of Statistical Mathematics".
3. J. Braga, C. Samir, **A. Fradi**, Y. Feunteun, K. Jakata, V.A. Zimmer, B. Zipfel, J.F. Thackeray, M. Macé, B.A. Wood and F.E. Grine. Unique cochlear

shape in *Paranthropus robustus* suggests a novel auditory ecology. **Submitted** to: "Science".

Published conference papers (peer-reviewed)

1. **A. Fradi**, C. Samir and A-F. Yao. Manifold-based inference for a supervised Gaussian process classifier. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)-2018.
2. **A. Fradi**, C. Samir and A-F. Yao. A Bayesian inference for a manifold Gaussian process classifier: applications to image classification. Société Française de Statistique (SFdS)-2019.
3. **A. Fradi** and C. Samir. Bayesian inference on local distributions of multi-dimensional curves. Société Française de Statistique (SFdS)-2020.

Communications with presentation

1. 49èmes Journées de Statistique (JDS)-2017. **Title:** "Manifold-based inference for a supervised Gaussian process classifier".
2. Joint Structures and Common Foundations of Statistical Physics, Information Geometry and Inference for Learning (SPIGL)-2020. **Title:** "Bayesian inference on local distributions of functions and multi-dimensional curves with spherical HMC sampling".

List of Abbreviations

Area under curve (AUC)
Australopithecus (AUS)
Broyden Fletcher Goldfarb Shanno (BFGS)
Breast cancer (BC)
Cumulative distribution function (CDF)
Computed-tomographie (CT)
Chimpanzees (PAN)
expectation propagation (EP)
Functional linear model (FLM)
Gaussian process (GP)
Gaussian process classifier (GPc)
Generalized Procrustes Analysis (GPA)
Gradient-descent (GD)
Gaussian process indexed by probability density functions (GPP)
Gaussian mixture model (GMM)
Gorillas (GOR)
Hamiltonian Monte Carlo (HMC)
Humans (HSS)
Jensen-Shannon (JS)
Karhunen-Loève (K-L)
Locally Linear Embedding (LLE)
Local Tangent Space Alignment (LTSA)
Maximum likelihood estimate (MLE)
Manifold Gaussian process (MGP)
Modified Locally Linear Embedding (MLLE)
Markov Chain Monte Carlo (MCMC)
Metropolis-Hastings (MH)

Manufacturing defect (MD)
Mean classification error (MCE)
Mean integrated squared error (MISE)
Newton-Raphson (NR)
Nonparametric kernel Wasserstein (NKW)
Negative log-marginal likelihood (NLML)
Paranthropus (PAR)
Principal Component Analysis (PCA)
Principal component (PC)
Pyramid of Histograms of Bag of Words (PHOW)
Probability density function (PDF)
Quasi-Newton (QN)
Reproducing Kernel Hilbert Space (RKHS)
Root mean square error (RMSE)
Square-root velocity function (SRVF)
Scalable Laplace approximation (SLA)
Scalable Expectation propagation (SEP)
Scalable Gaussian process classifier (SGPc)
Scene videos (SV)
t-distributed Stochastic Neighbor Embedding (t-SNE)
Tangent Principal Component Analysis (TPCA)
Truncated Fourier basis (TFB)
Violence Flows (ViF)



List of Figures

2.1	Predicting the function $f(x) = x \sin(x)$ for unobserved inputs in $[0, 10]$ from the GP regression.	9
2.2	The chart maps the part of the sphere with nonnegative z coordinate to a disc.	10
2.3	The trajectory of a Markov chain obtained from a MCMC sampling.	15
2.4	Four copies of the same shape under different Euclidean transformations.	17
2.5	An original shape (blue) transformed (red) with the action of translation (a), rotation (b) and scaling (c).	18
2.6	The i -th shape vector $\hat{\mathbf{x}}_i$ drawn from a population of mean \mathbf{x}_S (left), the tangent space projection by scaling (middle), and the tangent space projection by scaling and shape modification (right).	20
2.7	An example of three shape vectors \mathbf{x}_i (a) and their projections $\hat{\mathbf{z}}_i^t$ in the tangent space when applying the TPCA approach with two dimensions.	22
3.1	An example of two original images: non-defective (a) and defective (c). The associated extracted features: non-defective (b) and defective (d).	43
3.2	An Example of two video scene sequences where (top) non-violent and (bottom) violent. For each video: first frame (a,e), frame in the middle (b,f), last frame (c,g), and ViF descriptor (d,h).	43
3.3	Representing test data with contour plot colored as function of the predictive probability by region (a). The approximate predictor of class "+1" with optimal threshold (green line) (b). In all subfigures, normal tissues are red dotted and abnormal tissues are blue dotted.	45

3.4	Markov chain values of β_1 (a) and β_2 (c). Posterior distributions of β_1 (b) and β_2 (d).	45
3.5	Sigmoid function (red), SLA predictive distribution (green), and the product of sigmoid and predictive distribution (blue) (a). Probit function (red), SEP predictive distribution (green), and the product of probit and predictive distribution (blue) (b).	45
3.6	Representing and classifying data in two-dimensional feature space with SLA (top) and SEP (bottom). Test data (a,d), the approximate predictor of class "+1" with optimal threshold (green line) (b,e), and binary predicted outputs (c,f). In all subfigures, non-violent videos are red dotted and violent videos are blue dotted.	47
3.7	Performance evaluation as a mean classification error for different methods (a). Performance evaluation as a mean classification error when applying PCA to the baseline methods and the manifold embedding to the proposed SLA and SEP with several values of M for real data: BC (b), MD (c), and SV (d).	48
4.1	An illustration for representing a PDF p as an element g of the tangent space $\mathcal{T}_1(\mathcal{H})$	55
4.2	PDFs of TBF inputs for regression. The output with continuous value in $[-3, 4]$ is illustrated by a colorbar.	61
4.3	Synthetic PDFs for InvGamma (a) and Beta (b) with class 1 (red) and class 2 (blue). Semi-synthetic PDFs for Growth (c) with girls (red) and boys (blue). Real PDFs for Temp (d) with uninfected (red) and infected (blue). Real PDFs for Plants (e) with disease (red) and healthy (blue). In all subfigures, the Fréchet mean for each class is in black.	63
4.4	An example of two classes from maize plants dataset where healthy leaf (top) and leaf with disease (bottom). For each class: an original image (a,d), the extracted features (b,e), and the normalized histogram (c,f).	64

4.5	Boxplots of the classification accuracy (left) and AUC (right) on synthetic datasets: InvGamma (a) and Beta (b), and semi-synthetic dataset: Growth (c). In all subfigures, the performance is given for different methods: QN-GPP (red), HMC-GPP (light blue), W-GP (violet), and JS-GP (dark blue).	65
4.6	Boxplots of the classification accuracy (left) and AUC (right) on real data: Temp (a) and Plants (b). In all subfigures, the performance is given for different methods: QN-GPP (red), HMC-GPP (light blue), W-GP (violet), and JS-GP (dark blue).	66
5.1	An example of two different reparametrizations of the same class of curves with: first reparametrization (left) and second reparametrization (right).	70
5.2	Some examples of CDFs	72
5.3	A SRVF representation q (a) that is deformed using a number of CDFs F (b) giving the resulting q^* (c).	74
5.4	A first curve q_1 (a), a second curve q_2^* (b) and the resulting curve $\hat{q}_2^{*,m} \equiv \sqrt{\hat{F}_m} q_2 \circ \hat{F}_m$ depending on \hat{A} with the best matching of features between q_1 and q_2^* (c).	76
5.5	The true parameterized curves in \mathbb{R}^2 (a), the observed curves with $\sigma^2 = 0.01$ (b), and $\sigma^2 = 0.1$ (c). In all subfigures, the two clusters are illustrated with different colors: cluster 1 (blue) and cluster 2 (red).	81
5.6	The true parameterized curves in \mathbb{R}^3 (a), the observed curves with $\sigma^2 = 0.01$ (b), and $\sigma^2 = 0.1$ (c). In all subfigures, the two clusters are illustrated with different colors: cluster 1 (blue) and cluster 2 (red).	82
5.7	A medical CT image (a) and the extracted boundary surface with white cochlear curve (b).	84

5.8	The optimal CDF estimates $\hat{F}^{k,30}$ (a) and the Fréchet means of curves $\tilde{q}^{k,30}$ for above view (b) and front view (c). Cluster of female: $k = 1$ (blue) and cluster of male: $k = 2$ (red).	85
5.9	The conditional probability that i -th cochlea belongs to the first sub-population of female: $\mathbb{P}(C_i = 1 q_i)$ (a) and the resulting cluster (b). Cluster of female: $k = 1$ (blue) and cluster of male: $k = 2$ (red).	85
5.10	Markov chain values of a_1^1 (a), a_2^1 (b), and (a_1^1, a_2^1) (c). Posterior distributions of a_1^1 (d), a_2^1 (e), and (a_1^1, a_2^1) (f).	86
5.11	Some PAR cochlear hominin recovered from Kromdraai (South Africa) with blue cochlear curve.	87
5.12	The Fréchet mean of HSS (a), PAR (b), GOR (c), PAN (d) and AUS (e).	87
5.13	PC1 versus PC2 scatter plot obtained from TPCA (PC1 and PC2 represent 82.5% and 10.2% of variance, respectively) (a). The distance graph (normalized) with directional edges connecting the nodes of Fréchet means projected into the tangent space of the sphere formed by normalized curves (b).	88

List of Tables

3.1	Simulated datasets: Mean classification error.	42
3.2	Real data: Mean classification error.	43
3.3	Real data: Root mean square error.	43
4.1	Regression: Root mean square error.	62
4.2	Regression: Negative log-marginal likelihood.	62
4.3	Classification: Negative log-marginal likelihood.	67
5.1	Parameterized curves in \mathbb{R}^2 : Mean clustering error.	82
5.2	Parameterized curves in \mathbb{R}^3 : Mean clustering error.	83
5.3	Human cochlea: Mean clustering error (MCE), specificity (SP) and sensitivity (SE).	84
5.4	Hominin cochlea: Mean clustering error (MCE).	87

List of Algorithms

1	Gradient-descent.	13
2	Newton-Raphson.	13
3	Quasi-Newton.	14
4	Gibbs sampling.	16
5	Metropolis-Hasting.	17
6	GPA.	19
7	TPCA.	21
8	EP.	32
9	Leap-frog.	59
10	HMC sampling.	59
11	Newton-Raphson on the sphere.	77
12	Spherical HMC sampling.	80

Contents

1	General introduction	1
.1	Context and motivations	1
.1.1	High-dimensional data	1
.1.2	Shape analysis of curves	2
.2	Contributions	3
.2.1	High-dimensional and functional data	3
.2.2	Shape analysis of curves	4
.3	Outline	5
2	Background and basic notions	6
.1	Gaussian processes	6
.2	Manifold Gaussian processes	9
.3	Numerical algorithms and MCMC sampling	12
.3.1	Iterative optimization methods	12
.3.2	MCMC sampling	14
.4	Shape analysis of curves using landmarks	16
.5	Shape analysis of curves using a Riemannian structure	21
3	Bayesian classification using scalable Gaussian process priors for high-dimensional data	25
.1	Introduction	25
.2	Gaussian process classifier	27
.2.1	Problem formulation	27
.2.2	Laplace approximation	28
.2.2.1	Sampling functions and predictions	28
.2.2.2	Optimizing hyper-parameters	29
.2.3	Expectation propagation	30

	.2.3.1	Sampling functions and predictions	30
	.2.3.2	Optimizing hyper-parameters	31
.3		The embedded submanifold	31
.4		Scalable Gaussian process classifier	34
	.4.1	Posterior distribution approximations and predictions	34
		.4.1.1 Scalable Laplace approximation (SLA) . .	34
		.4.1.2 Scalable Expectation propagation (SEP) .	35
	.4.2	Optimizing hyper-parameters	36
		.4.2.1 The first partial derivatives of $\log \hat{\mathbb{P}}(\mathbf{y} \mathbf{Z})$.	36
		.4.2.2 The second partial derivatives of $\log \hat{\mathbb{P}}(\mathbf{y} \mathbf{Z})$	37
		.4.2.3 Numerical methods and sampling	39
.5		Applications	40
	.5.1	Synthetic datasets	41
	.5.2	Real data	42
		.5.2.1 Datasets of real data	44
		.5.2.2 Results on real data	44
	.5.3	Comparative study	47
.6		Conclusion	49
4		Bayesian regression and classification using Gaussian processes indexed by probability density functions	50
	.1	Introduction	50
	.2	Riemannian representation	52
	.3	Gaussian Processes on \mathcal{P}	54
	.4	Regression and classification on \mathcal{P}	55
		.4.1 Regression on \mathcal{P}	56
		.4.2 Classification on \mathcal{P}	56
	.5	Optimizing hyper-parameters	57
	.6	Applications	60
		.6.1 Regression	61
		.6.2 Classification	62

.6.2.1	Datasets for classification	62
.6.2.2	Classification results	66
.7	Conclusion	68
5	Bayesian registration and clustering of univariate functions and multidimensional curves	69
.1	Introduction	69
.2	Riemannian representation	71
.3	Spherical Gaussian process for curve registration	72
.3.1	Spherical Gaussian process decomposition	72
.3.2	Group actions, shape invariances and distance	74
.3.3	Optimal registration between curves	75
.4	Bayesian curve clustering	77
.5	Applications	81
.5.1	Synthetic datasets	82
.5.2	Real data	85
.6	Conclusion	88
6	Conclusion and prospects	89
.1	Summary of the contributions	89
.2	Future work and prospects	90
	Bibliography	91

Chapter 1: General introduction

This chapter summarizes the contents and describes the plan of the thesis. First, we highlight the motivations of this work. Then, we state the addressed issues.

1.1 Context and motivations

In this thesis, there are two main problems that we try to resolve: 1) the regression and the classification of high-dimensional and functional data, and 2) the registration and the clustering of shape of curves.

1.1.1 High-dimensional data

When dealing with high-dimensional data, there is various methods for linear dimensionality reduction. The two most widely used linear techniques for dimensionality reduction are: Principal Components Analysis [Journée et al. \(2010\)](#) and Multidimensional Scaling [Kruskal \(1964\)](#), both of which have solutions that are based upon the top few eigenvalues and associated eigenvectors of certain matrices. Other linear methods include projection pursuit which tries to explain the correlation structure of a set of variables by modeling those variables as a linear combination of a small number of unobserved latent variables or factors.

Many of the probability models used for machine learning have been interpreted as latent variable models [Ma and Fu \(2012\)](#). However, the nonlinear factor analysis has been informative in revealing inadequacies in linear relationships between variables. In order to explore the underlying nonlinear structure for multivariate data, we need to think about nonlinear manifold embedding. Manifold embedding approaches [Lee and Verleysen \(2007\)](#) became a new topic of research over two decades ago. It addressed the problem of how to recover a low-dimensional data from data located on that manifold, which is embedded within a higher dimensional ambient space [Gorban et al. \(2008\)](#). These approaches were the first attempts at nonlinear manifold learning, using spectral embedding methods.

On the other hand, Gaussian processes become useful in statistical modeling, benefiting from properties inherited from the normal distribution. However, methods based on Gaussian processes are successful with low dimensions, but are still limited in high-dimensions. While exact models often scale poorly as the amount of data increases, multiple approximation methods have been developed which often retain good accuracy while drastically reducing computation time. Among them, [Snelson and Ghahramani \(2006\)](#) has proposed an unsupervised dimensionality reduction by jointly learning a linear transformation of the input and a Gaussian process regression. More recently, [Calandra et al. \(2016\)](#) has introduced the notion of manifold Gaussian process for regression. The model profits from Gaussian processes properties and the advantages of the manifold embedding techniques for dimensionality reduction, jointly.

1.1.2 Shape analysis of curves

The original works in statistical analysis and modeling of shapes of objects came from [Kendall \(1984\)](#). The limitation of this work is the use of landmarks in defining shapes. Recently, there has been many focus on shape analysis of curves, albeit in the same spirit as Kendall's formulation. Consequently, there is some significant literatures on shapes of continuous curves as elements of Riemannian manifolds called pre-shape spaces. For instance, [Younes \(2000\)](#) defined pre-shape spaces of planar curves and imposed Riemannian metrics on them. In particular, he computed geodesic paths between curves in order to obtain deformations between them. Moreover, [Klassen et al. \(2004\)](#) restricted to arc-length parameterized planar curves and derived numerical algorithms for computing geodesics between curves. For shape analysis of curves, the elastic metric is widely accepted as the only metric invariant to reparameterizations. This is related to the Fisher-Rao metric used in information geometry. Curves can be represented in several ways where the form of elastic metric depends on the representation. With the square-root velocity function representation [Srivastava et al. \(2011\)](#) for curves, the pre-shape space is actually a subset of a unit sphere inside a Hilbert space. The use of geometry of

the sphere helps simplify computations to a large extent.

1.2 Contributions

Now, we will summarize the main contributions along this thesis for high-dimensional data, functional data and curves.

1.2.1 High-dimensional and functional data

In this thesis, we introduce a new concept of scalable Gaussian process classifier. The proposed model is closely to that of [Calandra et al. \(2016\)](#), but more general for representing and classifying high-dimensional data. It has the additional benefit to learn both the hyper-parameters of the adaptive embedding for dimensionality reduction and those of the Gaussian process classifier, jointly, with several optimization methods. Our formulation make it more easy to deal with nonlinearity of data and to create separability with mappings defined on a Reproducing Kernel Hilbert Space. In contrast to regression, computing the exact marginal likelihood remains difficult, if not impossible, for discrete likelihoods and high-dimensional inputs. So, we introduce two different methods to approximate a non-Gaussian posterior by a Gaussian one in order to improve the efficiency and the scalability of standard Gaussian process classifier.

For high-dimensional and functional data, we also introduce the notion of Gaussian processes indexed by probability density functions. We will particularly show how a Bayesian inference with Gaussian processes can be put into action on probability density spaces equipped with the Fisher-Rao metric. For more details, we will extend traditional machine learning methods from finite vectors to constrained functional instances. Our motivation is that many categories of observations can be represented by probability density functions with more advantages than working with vector/functional inputs directly.

1.2.2 Shape analysis of curves

For shape registration and particularly curves, one usually need to find the best reparametrization function, identified with the local distribution, between two shapes of curves. The set of reparametrization functions naturally forms a group of diffeomorphisms with group operation given by composition, which makes the optimization task very hard due to the group structure. In fact, there no underlying direction or metric that naturally arises in this group.

We propose a new version of reparametrization functions for curves, represented by their square-root velocity functions as elements on a Riemannian manifold. The main idea is to deal with the space of cumulative distribution functions induced with the Fisher-Rao metric as well as making the connection with the Hilbert sphere for its nice proprieties and geometries for statistics. Therefore, we model the square-root density function, as an element of the Hilbert sphere, with a Gaussian process prior. Instead of estimating a reparametrization function as a non-parametric element of the functional space directly, we will consider its truncated parametric version with coefficients belonging to the finite-dimensional sphere and resulting from the spherical Gaussian process decomposition.

We are also interested in the clustering process of observed curves in a unsupervised learning model. We are able to find the Fréchet mean of shapes for each sub-population depending on its local distribution. Compared to most of the previous methods, the proposed model gives us the conditional probability that each observed shape belongs to any given cluster.

When dealing with spherical constrained posteriors in a Bayesian framework, we will consider the spherical Hamiltonian Monte Carlo sampling [Lan et al. \(2014\)](#). The new samples are obtained by approximately solving a system of differential equations describing the paths of Hamiltonian dynamics controlling the position on the sphere and the velocity on its corresponding tangent space, iteratively, until convergence.

1.3 Outline

The remainder of this document is organized as follows. Chapter 2 presents some basic notions and related works that will be useful along this thesis. Chapter 3 provides a new Bayesian method for representing and classifying high-dimensional data based on Gaussian processes. In Chapter 4, we present the Bayesian regression and classification methods for high-dimensional and functional data. In the same context, we will introduce the notion of Gaussian processes indexed by probability density functions. We develop a nonparametric registration framework for univariate functions and multidimensional curves in Chapter 5. We also propose a Bayesian clustering model with local distributions modeled with spherical Gaussian processes. Concluding points and a presentation of future work make the body of Chapter 6.

Chapter 2: Background and basic notions

Before we give details of our frameworks and main contributions, we will recall some background and basic notions needed throughout the thesis.

2.1 Gaussian processes

In Gaussian processes (GPs) [Rasmussen and Williams \(2006\)](#), we focus directly on such distributions over functions. A GP defines a distribution over functions such that, if we pick any two or more points in a function (i.e., different input-output pairs), observations of the outputs at these points follow a joint (multivariate) Gaussian distribution. In GP regression, we assume the output $y \in \mathbb{R}$ of a function f at input $x \in \mathbb{R}^d$ can be written as

$$y = f(x) + \eta \tag{2.1}$$

where $\eta \sim \mathcal{N}(0, \sigma^2)$ refers to the noise term of variance σ^2 . Note that this is similar to the assumption made in linear regression, in that we assume an observation consists of an independent "signal" term $f(x)$ and a "noise" term η . The function $f(x)$ is distributed as

$$f(x) \sim \mathcal{GP}(m(x), c(x, x')) \tag{2.2}$$

A GP is fully defined by a mean function and a covariance function. The mean function $m(x)$ reflects the expected function value at input x

$$m(x) = \mathbb{E}[f(x)] \tag{2.3}$$

i.e., the average of all functions in the distribution evaluated at input x . The prior mean function is often set to $m(x) = 0$ in order to avoid expensive posterior computations and only do inference via the covariance function. Empirically, setting the prior to 0 is often achieved by subtracting the (prior) mean from all observations. The covariance function $c(x, x')$ models the dependence between the

function values at different input points x and x' as follows

$$c(x, x') = \mathbb{E}[(f(x) - m(x))(f(x') - m(x')))] \quad (2.4)$$

If $m(x) = 0$ then the covariance function simply takes $c(x, x') = \mathbb{E}[f(x)f(x')]$. The choice of an appropriate covariance function is based on assumptions such as smoothness and likely patterns to be expected in the data. A sensible assumption is usually that the correlation between two points decays with the distance between the points. This means that closer points are expected to behave more similarly than points which are further away from each other.

Sampling and predicition. Having set out the conditions on the covariance function, we can observe all realizations from

$$y_i = f(x_i) + \eta_i, \quad i = 1, \dots, N \quad (2.5)$$

With above of notations used in the introduction the likelihood term at $\mathbf{y} = (y_1, \dots, y_N)^T$ given $\mathbf{f} = (f_1, \dots, f_N)^T = (f(x_1), \dots, f(x_N))^T$ is

$$\begin{aligned} \mathbb{P}(\mathbf{y}|\mathbf{f}) &= \prod_{i=1}^N \mathbb{P}(y_i|f_i) \\ &= \prod_{i=1}^N \mathcal{N}(f_i, \sigma^2) \\ &= \mathcal{N}(\mathbf{f}, \sigma^2 \mathcal{I}) \end{aligned} \quad (2.6)$$

where \mathcal{I} is the $N \times N$ identity matrix. From (2.2), the prior on \mathbf{f} is

$$\mathbb{P}(\mathbf{f}|\mathbf{X}) = \mathcal{N}(0, \mathbf{C}) \quad (2.7)$$

with $\mathbf{C} = c(\mathbf{X}, \mathbf{X})$ and $\mathbf{X} = [x_1, \dots, x_N]^T$ is the $N \times d$ matrix of observed inputs. Inference in the Bayesian model is based on the posterior distribution, computed by Bayes' rule

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{marginal likelihood}} \quad (2.8)$$

which is updated in our case as

$$\mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{f})\mathbb{P}(\mathbf{f}|\mathbf{X})}{\mathbb{P}(\mathbf{y}|\mathbf{X})} \quad (2.9)$$

Now, we can draw samples from the distribution of functions evaluated at any an unobserved x^* . Then we generate a random Gaussian vector with this covariance

matrix as

$$f^* = f(x^*) \sim \mathcal{N}(0, \mathbf{C}_{**}) \quad (2.10)$$

for $\mathbf{C}_{**} = c(x^*, x^*)$. We can write the joint distribution of the observed target values and the function values at the test locations under the prior as

$$\begin{bmatrix} \mathbf{y} \\ f^* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \mathbf{C} + \sigma^2 \mathcal{I} & \mathbf{C}_* \\ \mathbf{C}_*^T & \mathbf{C}_{**} \end{bmatrix}\right) \quad (2.11)$$

where $\mathbf{C}_* = c(\mathbf{X}, x^*)$. By deriving the conditional distribution, we arrive at the key predictive equation

$$\mathbb{P}(f^* | \mathbf{X}, \mathbf{y}, x^*) = \mathcal{N}(f^* | \mu(x^*), \sigma^2(x^*)) \quad (2.12)$$

with

$$\begin{cases} \mu(x^*) = \mathbf{C}_*^T (\mathbf{C} + \sigma^2 \mathcal{I})^{-1} \mathbf{y} \\ \sigma^2(x^*) = \mathbf{C}_{**} - \mathbf{C}_*^T (\mathbf{C} + \sigma^2 \mathcal{I})^{-1} \mathbf{C}_* \end{cases} \quad (2.13)$$

To predict f^* , we can simply use the mean function $\mu(x^*)$ or sample functions from the GP with this mean function and variance $\sigma^2(x^*)$.

Optimizing hyper – parameters. The covariance function $c(\cdot, \cdot)$ usually depends on a vector of hyper-parameters θ_c , which is unknown and need to be inferred from the data. A common practice is to obtain point estimates of the hyper-parameters by maximizing the marginal (log) likelihood. This is similar to parameter estimation by maximum likelihood and is also referred to as type-II maximum likelihood estimate (MLE). The normalizing constant in (2.9), also known as the marginal likelihood, is the integral of the likelihood times the prior depending on θ_c

$$\mathbb{P}(\mathbf{y} | \mathbf{X}) = \int_{\mathbb{R}^N} \mathbb{P}(\mathbf{y} | \mathbf{f}) \mathbb{P}(\mathbf{f} | \mathbf{X}) d\mathbf{f} \quad (2.14)$$

The term marginal likelihood refers to the marginalization over the function values \mathbf{f} . We use the product of likelihood and prior terms to perform the integration yielding the log-marginal likelihood

$$l(\theta_c) = -\frac{1}{2} \mathbf{y}^T (\mathbf{C} + \sigma^2 \mathcal{I})^{-1} \mathbf{y} - \frac{1}{2} \log |\mathbf{C} + \sigma^2 \mathcal{I}| - \frac{N}{2} \log 2\pi \quad (2.15)$$

The marginal log likelihood can be viewed as a penalized fit measure, where the term $-\frac{1}{2} \mathbf{y}^T (\mathbf{C} + \sigma^2 \mathcal{I})^{-1} \mathbf{y}$ measures the data fit that is how well the current co-

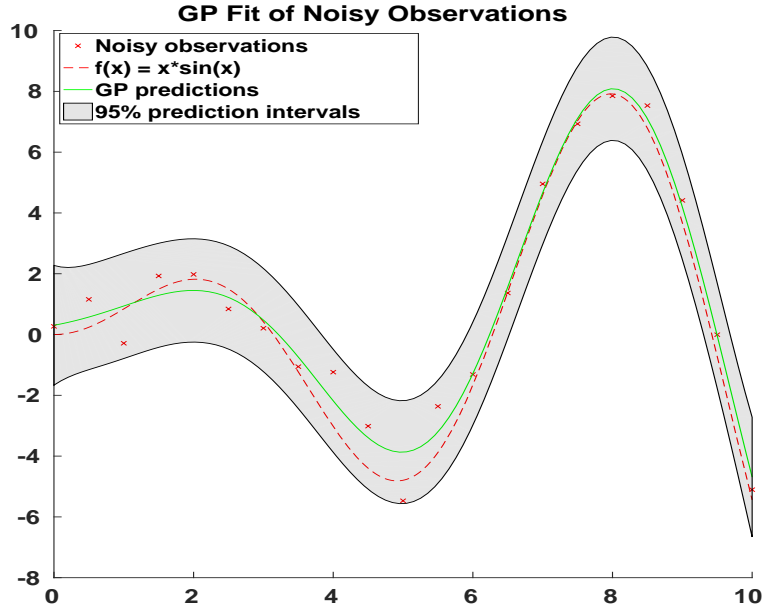


Figure 2.1: Predicting the function $f(x) = x \sin(x)$ for unobserved inputs in $[0, 10]$ from the GP regression.

variance parametrization explains the dependent variable and $-\frac{1}{2} \log |\mathbf{C} + \sigma^2 \mathcal{I}|$ is a complexity penalization term. The final term $-\frac{N}{2} \log 2\pi$ is a normalization constant. The marginal likelihood is normally maximized through many optimization tools that will be detailed in Section 2.3. These routines make use of the partial derivatives of $l(\theta_c)$ with respect to θ_c . Let $\theta_c = \{\theta_c^j\}_{j=1}^p \in \mathbb{R}^p$ denote the set of hyper-parameters of the covariance function $c(\cdot, \cdot)$. The partial derivative of $l(\theta_c)$ with respect to θ_c^j is

$$\frac{\partial l(\theta_c)}{\partial \theta_c^j} = \frac{1}{2} \mathbf{y}^T (\mathbf{C} + \sigma^2 \mathcal{I})^{-1} \frac{\partial \mathbf{C}}{\partial \theta_c^j} (\mathbf{C} + \sigma^2 \mathcal{I})^{-1} \mathbf{y} - \frac{1}{2} \text{tr} \left[(\mathbf{C} + \sigma^2 \mathcal{I})^{-1} \frac{\partial \mathbf{C}}{\partial \theta_c^j} \right] \quad (2.16)$$

We give an example of predicting inputs in $[0, 10]$ of an unknown function $f(x) = x \sin(x)$ in Figure 2.1. This is achieved from the noisy observations while training the GP model. We also compute the prediction intervals using the trained model.

2.2 Manifold Gaussian processes

First of all, what is a manifold ?

Definition 2.1

A manifold is a topological space that locally resembles Euclidean space near each point. More precisely, an d -dimensional manifold, or d -manifold

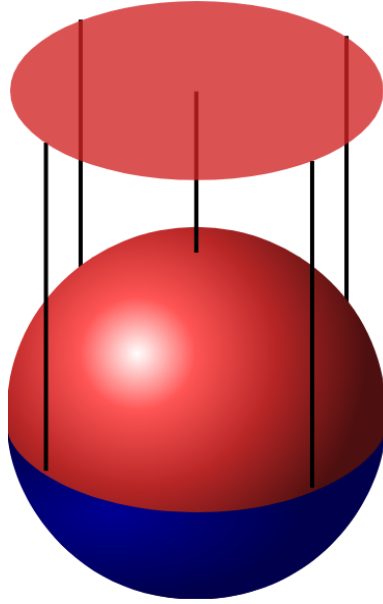



Figure 2.2: The chart maps the part of the sphere with nonnegative z coordinate to a disc.

for short, is a topological space with the property that each point has a neighborhood that is homeomorphic to the Euclidean space of dimension d . 

The simplest way to construct a manifold is the sphere. A sphere is just the surface (not the solid interior), which can be defined as a subset of \mathbb{R}^d . For $d = 3$, the sphere is two-dimensional satisfying

$$\mathcal{S}^2 = \left\{ (x, y, z) \in \mathbb{R}^3 \mid \|(x, y, z)\|_2^2 = x^2 + y^2 + z^2 = 1 \right\} \quad (2.17)$$

so each chart will map part of the sphere to an open subset of \mathbb{R}^2 . Consider the upper-hemisphere, which is the part with nonnegative z coordinate (colored red in Figure 2.2). The function ξ (called chart) defined by $\xi(x, y, z) = (x, y)$ maps the upper-hemisphere to the open unit disc by projecting it on the (x, y) plane. This can be easily generalized to higher-dimensional spheres.

Although GPs can be applied to a great range of problems, they are still limited in high-dimensions where the common smoothness assumptions are violated. In fact, if the input's dimension increases the covariance function becomes less informative. Two common ideas can overcome the limitations of the covariance functions. The first approach combines multiple standard covariance functions to form a new covariance function [Wilson and Adams \(2013\)](#). However, the resulting

covariance function remains limited by the properties of the combined covariance functions. The second approach is based on data transformation, after which the data can be modeled with standard covariance functions. Transforming the input space and subsequently applying GP with a standard covariance function is equivalent to GP with a new covariance function depending on the data transformation [MacKay \(1998\)](#). Common transformations of the inputs include data dimensionality reduction. Multiple related approaches in the literature attempt joint supervised learning of features and regression. Generally, these transformed inputs are good heuristics or optimize an unsupervised objective. However, they may be suboptimal for the overall regression task.

Recently, [Snelson and Ghahramani \(2006\)](#) has proposed an unsupervised dimensionality reduction, e.g., Principal Component Analysis (PCA) [Journée et al. \(2010\)](#) by jointly learning a linear transformation of the inputs and a GP. More recently, [Calandra et al. \(2016\)](#) has proposed the manifold GP (MGP) with flexible covariance functions for GPs. The GP model is equivalent to jointly learning a data transformation into a feature space (manifold) followed by a GP regression with off-the-shelf covariance functions from feature space to observed space. The model profits from standard GP properties, such as a straightforward incorporation of a prior mean function and a faithful representation of model uncertainty. In comparison with linear transformations, the purpose of a non-linear transformation (embedding) Ψ is to account for skewness in the data, while MGP allows for a more general class of transformations. If r denotes the MGP, it satisfies

$$r(x) \sim \mathcal{GP}(\tilde{m}(x), \tilde{c}(x, x')) \quad (2.18)$$

where $\tilde{m}(x) = m(\Psi(x))$ is the transformed prior mean and $\tilde{c}(x, x') = c(\Psi(x), \Psi(x'))$ refers to the modified covariance function.

For selecting a non-linear embedding Ψ , there exists many manifold embedding approaches for non-linear dimensionality reduction. Algorithms adopted for this task are based on the idea that the dimensionality of many data sets is only artificially high. The underlying idea is based on the fact that high-dimensional datasets can be very difficult to visualize. While data in two or three dimensions can be

plotted to show the inherent structure of the data, equivalent high-dimensional plots are much less intuitive. To aid visualization of the structure of a dataset, the dimension must be reduced in some way.

Manifold embedding approaches can be thought of as an attempt to generalize linear frameworks like PCA to be sensitive to non-linear structure in data. Though supervised variants exist, the typical manifold embedding problem is unsupervised: it learns the high-dimensional structure of the data from the data itself, without the use of predetermined classifications. Among manifold embedding approaches, we can cite for example: Isomap [Tenenbaum et al. \(2000\)](#), Locally Linear Embedding (LLE) [Roweis and Saul \(2000\)](#), Modified Locally Linear Embedding (MLLE) [Zhang and Wang \(2007\)](#), Spectral Embedding [Belkin and Niyogi \(2003\)](#), Local Tangent Space Alignment (LTSA) [Zhang and Zha \(2005\)](#) and t-distributed Stochastic Neighbor Embedding (t-SNE) [Van Der Maaten \(2014\)](#).

2.3 Numerical algorithms and MCMC sampling

In this section, we present two main categories of algorithms established in this manuscript based on iterative optimization methods and Monte Carlo sampling [Van Ravenzwaaij et al. \(2018\)](#).

2.3.1 Iterative optimization methods

Let $f : \mathbb{R}^p \rightarrow \mathbb{R}$ be a smooth function (cost function). Optimization algorithms tend to be iterative procedures in order to find a local minimum of $f(\cdot)$. Starting from a given point x_0 , they generate a sequence $(x_t)_t$ of iterates (or trial solutions) that converge to a solution or at least they are designed to be so.

Gradient – descent. is a first-order iterative optimization algorithm based on the observation that $f(x)$ decreases fastest if one goes from x in the direction of the negative gradient of f at x : $-\nabla f(x)$. It follows that if $x_{t+1} = x_t - \epsilon \nabla f(x_t)$, for $\epsilon \in \mathbb{R}^+$ then $f(x_t) \geq f(x_{t+1})$. Therefore, we have a monotonic sequence $f(x_0) \geq f(x_1) \geq f(x_2) \geq \dots$, so hopefully the sequence $(x_t)_t$ converges to the

desired local minimum. The gradient-descent is summarized in Algorithm 1.

Algorithm 1: Gradient-descent.

Require: cost function $f(\cdot)$ and its gradient vector $\nabla f(\cdot)$

- 1: repeat
 - 2: Evaluate $\nabla f(x_t)$
 - 3: Find the step size ϵ (e.g., by backtracking line search)
 - 4: Compute $x_{t+1} := x_t - \epsilon \nabla f(x_t)$
 - 5: Set $t := t + 1$
 - 6: until Convergence
-

Newton – Raphson. allows a numerical resolution of the score equation. We start from an arbitrary initial value of x , denoted x_0 and we designate by $x_1 = x_0 + h$ as a candidate value to be a solution of $\nabla f(x) = 0$, that is to say $\nabla f(x_0 + h) = 0$. Applying a first order Taylor series to the function $\nabla f(\cdot)$, we get

$$\nabla f(x_0 + h) \approx \nabla f(x_0) + h \nabla^2 f(x_0) \quad (2.19)$$

As $\nabla f(x_0 + h) = 0$ then h takes the following value

$$h = -[\nabla^2 f(x_0)]^{-1} \nabla f(x_0) \quad (2.20)$$

implying

$$x_1 = x_0 - [\nabla^2 f(x_0)]^{-1} \nabla f(x_0) \quad (2.21)$$

We update the last equation for any t and we express x_{t+1} in terms of x_t , as illustrated in Algorithm 2.

Algorithm 2: Newton-Raphson.

Require: cost function $f(\cdot)$, its gradient vector $\nabla f(\cdot)$ and its Hessian matrix $\nabla^2 f(\cdot)$

- 1: repeat
 - 2: Evaluate $\nabla f(x_t)$ and $\nabla^2 f(x_t)$
 - 3: Find the step size ϵ (e.g., by backtracking line search)
 - 4: Compute $x_{t+1} := x_t - \epsilon [\nabla^2 f(x_t)]^{-1} \nabla f(x_t)$
 - 5: Set $t := t + 1$
 - 6: until Convergence
-

Quasi – Newton. is an alternative to Newton's method used to either find zeros or local minimum of functions. It can be used if the gradient or the Hessian

is unavailable or is too expensive to be computed at every iteration. There is many popular update formulas to approximate the Hessian matrix, for instance the Broyden Fletcher Goldfarb Shanno (BFGS) method. Algorithm 3 summarizes the quasi-Newton method with BFGS updates.

Algorithm 3: Quasi-Newton.

Require: cost function $f(\cdot)$ and its gradient vector $\nabla f(\cdot)$

- 1: repeat
 - 2: Evaluate $\rho := -H_t^{-1}\nabla f(x_t)$
 - 3: Find the step size ϵ (e.g., by backtracking line search)
 - 4: Evaluate $x_{t+1} := x_t + \epsilon\rho$
 - 5: $a := x_{t+1} - x_t$ and $b := \nabla f(x_{t+1}) - \nabla f(x_t)$
 - 6: $H_{t+1}^{-1} := H_t + \frac{bb^T}{b^T a} - \frac{H_t a a^T H_t}{a^T H_t a}$
 - 7: Set $t := t + 1$
 - 8: until Convergence
-

2.3.2 MCMC sampling

When performing Bayesian inference, we aim to compute and use the full posterior joint distribution over a set of random variables. Unfortunately, this often requires calculating intractable integrals. In such cases, we may give up on solving the analytical equations and proceed with sampling techniques based upon Markov Chain Monte Carlo (MCMC) methods. When using MCMC methods, we estimate the posterior distribution and the intractable integrals using simulated samples from the posterior distribution. The underlying logic of MCMC sampling is that we can estimate any desired expectation by ergodic averages. That is, we can compute any statistic of a posterior distribution as long as we have T simulated samples from that distribution

$$\mathbb{E}[f(X)]_\pi \approx \frac{1}{T} \sum_{t=1}^T f(x_t) \quad (2.22)$$

where π is the posterior distribution of interest, $f(X)$ is the desired expectation, and $f(x_t)$ is the t -th simulated sample from π . For example, we can estimate the mean by $\mathbb{E}[X]_\pi \approx \frac{1}{T} \sum_{t=1}^T (x_t)$ as a particular case of (2.22) for the identity func-

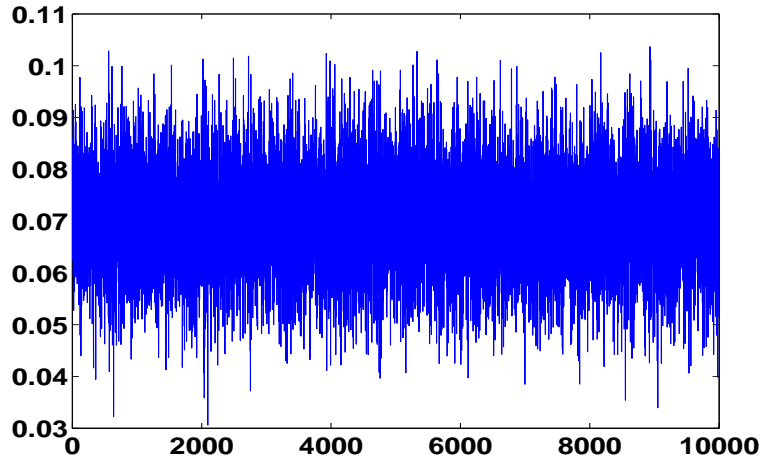


Figure 2.3: The trajectory of a Markov chain obtained from a MCMC sampling.

tion.

The theory of MCMC guarantees that the stationary distribution of the samples is the target joint posterior. For this reason, MCMC algorithms are typically run for a large number of iterations (in the hope that convergence to the target posterior will be achieved). An example of Markov chain values sampled from the MCMC algorithm is illustrated in Figure 2.3. Because samples from the early iterations are not from the target posterior, it is common to discard these samples. The discarded iterations are often referred to as the "Burn-in" period. To reduce autocorrelation between samples we also need the "Thinning" period. The question that arises now is: How do we obtain samples from the posterior distribution?

Ulam and Metropolis introduced the Metropolis algorithm and its impact was enormous. Afterwards, MCMC was introduced to statistics and generalized with the Metropolis-Hastings algorithm [Hastings \(1970\)](#) and the Gibbs sampling [Geman and Geman \(1984\)](#).

Gibbs sampling. is one MCMC technique suitable for the task. The idea in Gibbs sampling is to generate posterior samples by sweeping through each variable (or block of variables) to sample from its conditional distribution with the remaining variables fixed to their current values. For instance, we consider the random variables X^1 , X^2 , and X^3 and we start by setting these variables to their initial values x_0^1 , x_0^2 and x_0^3 . At iteration t , we sample $x_t^1 \sim \pi(X^1 = x^1 | X^2 = x_{t-1}^2, X^3 = x_{t-1}^3)$, sample $x_t^2 \sim \pi(X^2 = x^2 | X^1 = x_t^1, X^3 = x_{t-1}^3)$, and sample $x_t^3 \sim \pi(X^3 = x^3 | X^1 =$

$x_t^1, X^2 = x_t^2$). This process continues until convergence. Algorithm 4 details a generic Gibbs sampling generated for p variables.

Algorithm 4: Gibbs sampling.

Require: posterior distribution $\pi(\cdot)$

- 1: Initialize $x_0 = (x_0^1, \dots, x_0^p)$
- 2: for $t = 1, 2, \dots, T$ do
- 3: $x_t^1 \sim \pi(X^1 = x^1 | X^2 = x_{t-1}^2, X^3 = x_{t-1}^3, \dots, X^p = x_{t-1}^p)$
- 4: $x_t^2 \sim \pi(X^2 = x^2 | X^1 = x_t^1, X^3 = x_{t-1}^3, \dots, X^p = x_{t-1}^p)$
- \vdots
- 5: $x_t^p \sim \pi(X^p = x^p | X^1 = x_t^1, X^2 = x_t^2, \dots, X^{p-1} = x_t^{p-1})$
- 6: end for

Metropolis – Hastings (MH). simulates samples from a probability distribution by making use of the full joint density function and (independent) proposal distributions for each of the variables of interest. The first step is to initialize the sample value for each random variable. This value is often sampled from the variable’s prior distribution. The main loop of HM algorithm consists of three steps:

1. Generate a proposal (or a candidate) sample x_{cand} from the proposal distribution $q(x_{\text{cand}} | x_{t-1})$.
2. Compute the acceptance probability via the acceptance function $\alpha(x_{\text{cand}} | x_{t-1})$ based upon the proposal distribution and the full joint density $\pi(\cdot)$.
3. Accept the candidate sample with probability α or reject it with probability $1 - \alpha$.

Algorithm 5 provides the details of a generic MH algorithm.

2.4 Shape analysis of curves using landmarks

In shape analysis [Dryden and Mardia \(1998, 2016\)](#); [Gower \(1975\)](#), we usually need a technique that involves transformations (i.e., translation, rotation and isotropic scaling) of individual data matrices to provide optimal comparability. This notion is widely applicable to many areas such as shape of curves. According to Kendall, the shape of a curve can be represented with n landmarks in \mathbb{R}^d , where $n > d$. The n landmarks are points located in d dimensions which represent the important

Algorithm 5: Metropolis-Hasting.

 Require: posterior distribution $\pi(\cdot)$ and proposal distribution $q(\cdot, \cdot)$

- 1: Initialize x_0
- 2: for $t = 1, 2, \dots, T$ do
- 3: Propose $x_{\text{cand}} \sim q(x_t | x_{t-1})$
- 4: Acceptance probability

$$\alpha(x_{\text{cand}} | x_{t-1}) := \min \left\{ 1, \frac{q(x_{t-1} | x_{\text{cand}}) \pi(x_{\text{cand}})}{q(x_{\text{cand}} | x_{t-1}) \pi(x_{t-1})} \right\}$$

- 5: Simulate $u \sim \mathcal{U}([0, 1])$
 - 6: if $u < \alpha$ then
 - 7: Accept the proposal $x_t := x_{\text{cand}}$
 - 8: else
 - 9: Reject the proposal $x_t := x_{t-1}$
 - 10: end if
 - 11: end for
-



Figure 2.4: Four copies of the same shape under different Euclidean transformations.

features of the objects under study. What do we actually understand by the concept of shape and landmark?

Definition 2.2

Shape is all the geometrical information that remains when location, scale and rotational effects are filtered out from an object. ♣

In other words, shape is invariant to Euclidean similarity transformations. This is reflected in Figure 2.4. A second example of an original shape moved with the action of several Euclidean transformations is given in Figure 2.5. One way to describe a shape is by locating a finite number of points on the outline.

Definition 2.3

A landmark is a point of correspondence on each object that matches between and within populations. ♣

A mathematical representation of a shape formed by n points in d dimensions could

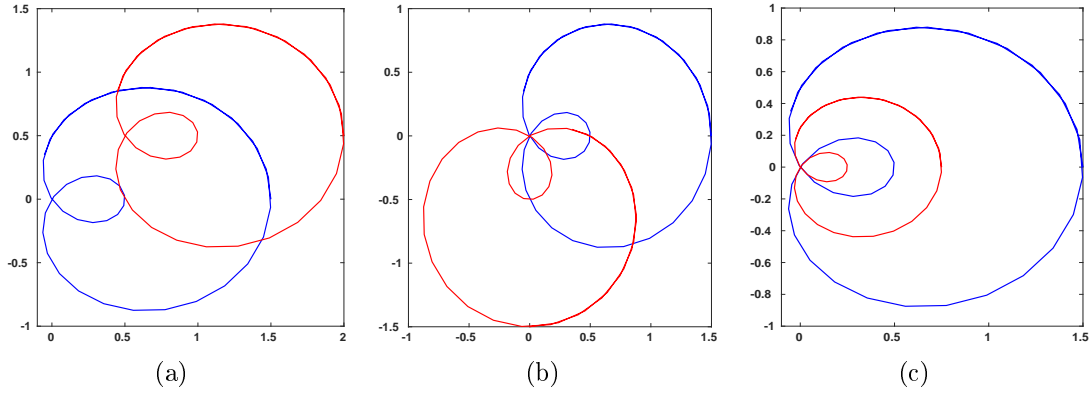


Figure 2.5: An original shape (blue) transformed (red) with the action of translation (a), rotation (b) and scaling (c).

be to concatenate each dimension into a nd -vector. The vector representation for planar shapes ($d = 2$) would then be

$$\mathbf{x} = (x_1, \dots, x_n, y_1, \dots, y_n)^T \quad (2.23)$$

To obtain a true shape representation according to the definition location, scale and rotational effects need to be filtered out. This is usually carried out by establishing a coordinate reference around position, scale and rotation. This is commonly known as pose for which all shapes are aligned.

Definition 2.4

A Shape space is the set of all possible shapes of the object in question. Formally, the shape space denoted by Σ_d^n is the orbit shape of the non-coincident n point set configurations in \mathbb{R}^d under the action of the Euclidean similarity transformations.



The question that arises now is: What is the dimension spanned by the shape space? If we have n point vectors in d Euclidean dimensions, the shape space's dimension is nd . But the alignment procedure peels of dimensionality i.e., the data now spans only a subspace of dimension less than nd . The translation removes d dimensions, the uniform scaling 1 dimension and the rotation $\frac{1}{2}d(d-1)$ dimensions. Thus, if k denotes the shape space dimensionality then it satisfies

$$k = nd - d - 1 - \frac{1}{2}d(d-1) \quad (2.24)$$

This can be achieved by the Generalized Procrustes Analysis (GPA) approach.

Summary of GPA. Although an analytic solution exists to the alignment of a set of two planar shapes, Algorithm 6 will suffice for any dimension d and any set of N shapes.

Algorithm 6: GPA.

Input: $\mathbf{x}_1, \dots, \mathbf{x}_N \in \mathbb{R}^{nd}$

Output: $\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_N \in \mathcal{S}^k$

- 1: Compute the centroids of all shapes: $\bar{\mathbf{x}}_i, i = 1, \dots, N$
- 2: Align w.r.t. position all shapes at their centroids

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i - \bar{\mathbf{x}}_i, \quad i = 1, \dots, N$$

- 3: Re-scale all shapes to have equal size

$$\tilde{\mathbf{x}}_i^* = \frac{\tilde{\mathbf{x}}_i}{\|\tilde{\mathbf{x}}_i\|_2} = \frac{\mathbf{x}_i - \bar{\mathbf{x}}_i}{\|\mathbf{x}_i - \bar{\mathbf{x}}_i\|_2}, \quad i = 1, \dots, N$$

- 4: Compute the Fréchet mean of $\tilde{\mathbf{x}}_1^*, \dots, \tilde{\mathbf{x}}_N^*$ on the finite-dimensional unit sphere \mathcal{S}^k denoted \mathbf{x}_S since $\|\tilde{\mathbf{x}}_i^*\|_2 = 1$
- 5: Find the optimal rotation for all shapes

$$R_i = \operatorname{argmin}_{R \in SO(d)} \|\tilde{\mathbf{x}}_i^* - \mathbf{x}_S R\|_2, \quad i = 1, \dots, N$$

where $SO(d)$ is the special orthogonal group of $d \times d$ rotation matrices through the Singular Value Decomposition (SVD) [Berge \(1977\)](#)

- 6: Set the i -th aligned shape to $\hat{\mathbf{x}}_i = \tilde{\mathbf{x}}_i^* R_i$
-

Tangent space projection. The notion of tangent space projection is based on modifying the shape vectors from a hyper-sphere to a hyper-plane. Further, the Euclidean distance in this plane can be employed as a shape metric. Suppose that $\hat{\mathbf{x}}_i$ is the i -th aligned shape and \mathbf{x}_S is the Fréchet mean of all shapes belonging to \mathcal{S}^k , as illustrated in Figure 2.6 (left). The remaining two sketches in Figure 2.6 show two tangent space projections of which we will focus on the former. Let $\hat{\mathbf{x}}_i^t$ be the projection of $\hat{\mathbf{x}}_i$ into the tangent space of \mathcal{S}^k at \mathbf{x}_S . From Figure 2.6 (middle), we see that $\hat{\mathbf{x}}_i^t$'s projection onto \mathbf{x}_S is \mathbf{x}_S , i.e.,

$$\mathbf{x}_S = \frac{\langle \mathbf{x}_S, \hat{\mathbf{x}}_i^t \rangle_2}{\|\mathbf{x}_S\|_2^2} \mathbf{x}_S = \langle \mathbf{x}_S, \hat{\mathbf{x}}_i^t \rangle_2 \mathbf{x}_S = \beta \mathbf{x}_S \quad (2.25)$$

Substituting $\hat{\mathbf{x}}_i^t$ with $\alpha \hat{\mathbf{x}}_i$ in the scaling factor β , we get

$$\beta = 1 = \langle \mathbf{x}_S, \hat{\mathbf{x}}_i^t \rangle_2 = \alpha \langle \mathbf{x}_S, \hat{\mathbf{x}}_i \rangle_2 \quad (2.26)$$

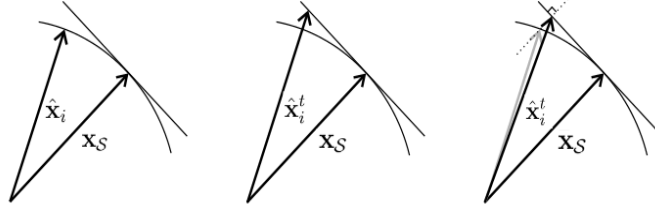


Figure 2.6: The i -th shape vector $\hat{\mathbf{x}}_i$ drawn from a population of mean \mathbf{x}_S (left), the tangent space projection by scaling (middle), and the tangent space projection by scaling and shape modification (right).

Finally, we show that

$$\hat{\mathbf{x}}_i^t = \alpha \hat{\mathbf{x}}_i = \frac{1}{\langle \mathbf{x}_S, \hat{\mathbf{x}}_i \rangle_2} \hat{\mathbf{x}}_i \quad (2.27)$$

Tangent Principal Component Analysis. Once all configurations have been aligned (or registered) to a common coordinate frame filtering out similarity transformations, they represent the shape of each structure. For instance, we could use PCA [Journée et al. \(2010\)](#) onto aligned data to find a (small) set of orthonormal directions that explain most of the shape variability. We show how the PCA can be derived by means of simple linear algebra and used for modeling shape variation. We consider the case of having N shapes consisting of n points in d dimension where i -th shape is represented as $\mathbf{x}_i \in \mathbb{R}^{nd}$, aligned by $\hat{\mathbf{x}}_i \in \mathcal{S}^k$ and projected into the tangent linear space at $\hat{\mathbf{x}}_i^t \in \mathbb{R}^{k+1}$. If we consider a set covering a certain class of shapes then we will always observe some degrees of inter-point correlation. If not, i) the set either contains no variation, or ii) the points are purely random, which implies that the points are not landmarks. This argumentation leads to the suspicion that there could exist a shape representation accounting for correlation between points. If some point movements were to be totally correlated, this could be exploited to reduce the dimensionality. The central idea of Tangent Principal Component Analysis (TPCA) is to reduce the dimensionality of projected shape vectors into the tangent space of \mathcal{S}^k . This is achieved when transforming to a new set of variables, known as principal components (PCs), so that the first few retain most of the variation present in all of the original variables. The whole process of obtaining principle components from a raw dataset of projected shape

vectors in the tangent space can be simplified in Algorithm 7. An example that

Algorithm 7: TPCA.

Input: $\hat{\mathbf{x}}_1^t, \dots, \hat{\mathbf{x}}_N^t \in \mathbb{R}^{k+1}$

Output: $\hat{\mathbf{z}}_1^t, \dots, \hat{\mathbf{z}}_N^t \in \mathbb{R}^M$

- 1: Compute the Euclidean mean of $\hat{\mathbf{x}}_1^t, \dots, \hat{\mathbf{x}}_N^t$ denoted $\hat{\mathbf{x}}_{\text{mean}}^t$
- 2: Compute the covariance matrix $\mathbf{C}_{\hat{\mathbf{x}}^t}$ of the whole dataset of projected shape vectors
- 3: Find the eigen-vectors v_1, \dots, v_{k+1} of $\mathbf{C}_{\hat{\mathbf{x}}^t}$ and the corresponding eigen-values $\lambda_1, \dots, \lambda_{k+1}$. The eigen-values of $\mathbf{C}_{\hat{\mathbf{x}}^t}$ are roots of the characteristic equation

$$\det(\mathbf{C}_{\hat{\mathbf{x}}^t} - \lambda \mathcal{I}) = 0$$

- 4: Sort the eigen-vectors by decreasing the eigen-values and choose M eigen-vectors with the largest eigen-values

$$\lambda_{k+1} < \lambda_k < \dots < \lambda_{M+1} < \underbrace{\lambda_M < \dots < \lambda_1}_{\text{kept eigen-values}}$$

to form a $(k+1) \times M$ dimensional matrix: $W = [v_1, v_2, \dots, v_M]$

- 5: Use W to transform the samples of projected shape vectors onto the new sub-space

$$\hat{\mathbf{z}}_i^t = W^T \hat{\mathbf{x}}_i^t$$

illustrates this approach is given in Figure 2.7. Accordingly, we keep the two most important directions for the TPCA approach when reducing the dimensionality of the projected shape vectors into the tangent space.

2.5 Shape analysis of curves using a Riemannian structure

In order to develop a formal framework for analyzing shapes of curves, one needs a mathematical representation of curves that is natural, general and efficient. We describe one such representation that allows a simple framework for shape analysis with a Riemannian structure. Let $\beta : I = [0, 1] \rightarrow \mathbb{R}^d$, $d \geq 1$. If $d = 1$ then β is an univariate function, else β is a multidimensional curve.

Pre – shape space and metric. Assume that for all $\xi \in I$, $\dot{\beta}(\xi) \neq 0$. We then define $\varphi : I \rightarrow \mathbb{R}$ by $\varphi(\xi) = \ln(\|\dot{\beta}(\xi)\|_2)$, and $\theta : I \rightarrow \mathcal{S}^{d-1}$ by $\theta(\xi) = \frac{\dot{\beta}(\xi)}{\|\dot{\beta}(\xi)\|_2}$. Clearly, φ and θ completely specify $\dot{\beta}$ since $\dot{\beta}(\xi) = \exp(\varphi(\xi))\theta(\xi)$. Thus, we have

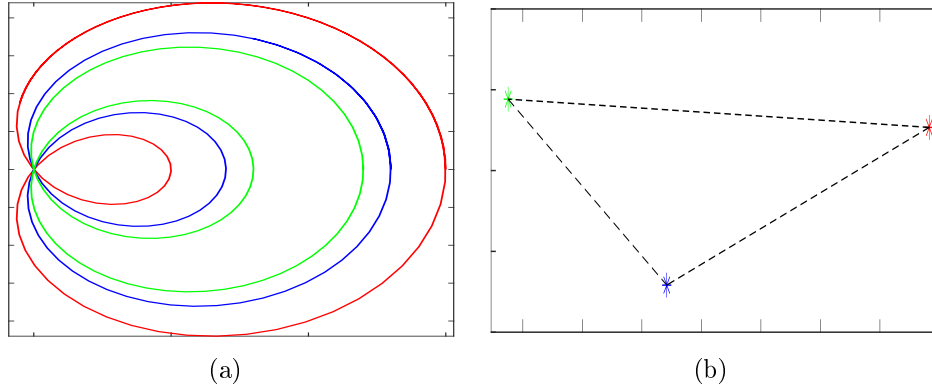


Figure 2.7: An example of three shape vectors \mathbf{x}_i (a) and their projections $\hat{\mathbf{z}}_i^t$ in the tangent space when applying the TPCA approach with two dimensions.

defined a map from the space of smooth curves in \mathbb{R}^d to $\Phi \times \Theta$ where $\Phi = \left\{ \varphi : I \rightarrow \mathbb{R} \right\}$ and $\Theta = \left\{ \theta : I \rightarrow \mathcal{S}^{d-1} \right\}$. This map is surjective and not injective, but two curves are mapped to the same pair (φ, θ) if and only if they are translates of each other, i.e., if they differ by an additive constant. Intuitively, φ tells us the speed of traversal of the curve, while θ tells us the direction of the curve at each time ξ .

In order to quantify the magnitudes of perturbations of β , we wish to impose a Riemannian metric on the space of curves that is invariant under translation by putting a metric on $\Phi \times \Theta$. First, we state that the tangent of space of $\Phi \times \Theta$ at any point (φ, θ) is given by

$$\mathcal{T}_{(\varphi, \theta)}(\Phi \times \Theta) = \left\{ (u, v) \mid u \in \Phi, v : I \rightarrow \mathbb{R}^d, \text{ and } v(\xi) \perp \theta(\xi); \forall \xi \in I \right\} \quad (2.28)$$

Suppose (u_1, v_1) and (u_2, v_2) are both elements of $\mathcal{T}_{(\varphi, \theta)}(\Phi \times \Theta)$. Let define an inner product by

$$\begin{aligned} \left\langle (u_1, v_1), (u_2, v_2) \right\rangle_{(\varphi, \theta)} &= \frac{1}{4} \int_I u_1(\xi) u_2(\xi) \exp(\varphi(\xi)) d\xi \\ &+ \int_I \left\langle v_1(\xi), v_2(\xi) \right\rangle_2 \exp(\varphi(\xi)) d\xi \end{aligned} \quad (2.29)$$

The first integral measures the amount of "stretching" since u_1 and u_2 are variations of the speed φ of the curve, while the second integral measures the amount of "bending" since v_1 and v_2 are variations of the direction φ of the curve.

Square – root velocity function. We now define the square-root velocity function (SRVF) representation [Srivastava et al. \(2011\)](#).

Definition 2.5

The SRVF $q : I \rightarrow \mathbb{R}^d$ of β is defined as

$$q(\xi) = \frac{\dot{\beta}(\xi)}{\sqrt{\|\dot{\beta}(\xi)\|_2}} \quad (2.30)$$



Now, we will prove that the \mathbb{L}^2 metric in the SRVF representation is (2.29). Relating this to the (φ, θ) representation of the curve, gives $q(\xi) = \exp(\frac{1}{2}\varphi(\xi))\theta(\xi)$. A couple of simple differentiations show that if $(u, v) \in \mathcal{T}_{(\varphi, \theta)}(\Phi \times \Theta)$ then the corresponding tangent vector to $\mathbb{L}^2(I, \mathbb{R}^d)$ at q is given by

$$\delta q(\xi) = \frac{1}{2} \exp\left(\frac{1}{2}\varphi(\xi)\right) \left\langle u(\xi), \theta(\xi) \right\rangle_2 + \exp\left(\frac{1}{2}\varphi(\xi)\right) v(\xi) \quad (2.31)$$

Now, let (u_1, v_1) and (u_2, v_2) denote two elements of $\mathcal{T}_{(\varphi, \theta)}(\Phi \times \Theta)$ and let δq_1 and δq_2 denote the corresponding tangent vectors to $\mathbb{L}^2(I, \mathbb{R}^d)$ at q . Computing the \mathbb{L}^2 inner product of δq_1 and δq_2 , yields

$$\begin{aligned} \langle \delta q_1, \delta q_2 \rangle &= \int_I \left\langle \frac{1}{2} \exp\left(\frac{1}{2}\varphi(\xi)\right) \left\langle u_1(\xi), \theta(\xi) \right\rangle_2 + \exp\left(\frac{1}{2}\varphi(\xi)\right) v_1(\xi), \right. \\ &\quad \left. \frac{1}{2} \exp\left(\frac{1}{2}\varphi(\xi)\right) \left\langle u_2(\xi), \theta(\xi) \right\rangle_2 + \exp\left(\frac{1}{2}\varphi(\xi)\right) v_2(\xi) \right\rangle_2 d\xi \\ &= \frac{1}{4} \int_I \exp(\varphi(\xi)) u_1(\xi) u_2(\xi) d\xi + \int_I \exp(\varphi(\xi)) \left\langle v_1(\xi), v_2(\xi) \right\rangle_2 d\xi \end{aligned} \quad (2.32)$$

In the above term, we have used the fact that $\left\langle \theta(\xi), \theta(\xi) \right\rangle_2 = 1$ since $\theta(\xi)$ is an element of the unit sphere \mathcal{S}^{d-1} as well as $\left\langle \theta(\xi), v_j(\xi) \right\rangle_2 = 0$ since $v_j(\xi)$ is a tangent vector to \mathcal{S}^{d-1} at $\theta(\xi)$. This clearly shows that the \mathbb{L}^2 metric on the space of SRVFs corresponds precisely to the elastic metric (2.29) on $\Phi \times \Theta$.

Geodesics. Conversely, for $q \in \mathbb{L}^2(I, \mathbb{R}^d)$ there exists a curve β (unique up to a translation) such that q is the SRVF of β . In fact, β can be rewritten as $\beta(\xi) = \beta(0) + \int_0^\xi q(t) \|q(t)\|_2 dt$. To remove scaling variability, we rescale all curves to be of unit length. This restriction comes from the fact that $\int_I \|q(\xi)\|_2^2 d\xi = \int_I \|\dot{\beta}(\xi)\|_2^2 d\xi = 1$. Let \mathcal{M} denote the space of all qs . The standard metric on $\mathbb{L}^2(I, \mathbb{R}^d)$ restricts to Riemannian structure on \mathcal{M} . This structure can then be used to determine geodesic and geodesic length between elements of this space [Lang \(1998\)](#).

Let $\alpha : I \rightarrow \mathcal{M}$ be a geodesic path such that $\alpha(0) = q_1$ and $\alpha(1) = q_2$. Then, the

length of α is defined by

$$L[\alpha] = \int_I \langle \dot{\alpha}(\xi), \dot{\alpha}(\xi) \rangle_2^{1/2} d\xi \quad (2.33)$$

In addition, α is said to be a length-minimizing geodesic if $L[\alpha]$ achieves the infimum over all such paths. The length of the geodesic path becomes a distance

$$d_{\mathcal{M}}(q_1, q_2) = \inf_{\alpha | \alpha(0)=q_1, \alpha(1)=q_2} L[\alpha] \quad (2.34)$$

Since \mathcal{M} is a linear subspace of $\mathbb{L}^2(I, \mathbb{R}^d)$, the geodesic between q_1 and q_2 becomes straightforward.

Lemma 2.1

Given q_1 and q_2 in \mathcal{M} , a geodesic path between them is given by

$$\alpha(\xi) = (1 - \xi)q_1 + \xi q_2 \quad \text{for all } 0 \leq \xi \leq 1 \quad (2.35) \heartsuit$$

Chapter 3: Bayesian classification using scalable Gaussian process priors for high-dimensional data

In this chapter, we adopt a Bayesian point of view, based on Gaussian processes, for classifying high-dimensional data. Since computing the exact marginal likelihood remains difficult, if not impossible, for discrete likelihoods and high-dimensional inputs, we introduce two different methods to improve the efficiency and the scalability of standard Gaussian processes for classification: scalable Laplace approximation and scalable expectation propagation, together with a proposed non-linear embedding for dimensionality reduction.

3.1 Introduction

In this chapter, we assume we are given N observations $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$, in which $x_i \in \mathcal{X} \subseteq \mathbb{R}^d, i \in \{1, \dots, N\}$, are the high-dimensional inputs (predictors) and $y_i, i \in \{1, \dots, N\}$, are the associated responses. Throughout this chapter, we consider the binary classification case where y_i takes values in $\{-1, +1\}$. In a Bayesian framework, we model f by a zero mean GP classifier (GPC) with a covariance function $c(\cdot, \cdot)$ controlling its underlying structure. This implies a multivariate Gaussian density function $\mathbb{P}(\mathbf{f}|\mathbf{X}) = \mathcal{N}(\mathbf{f}|0, c(\mathbf{X}, \mathbf{X}))$ where \mathbf{X} is the $N \times d$ matrix $[x_1, \dots, x_N]^T$ and \mathbf{f} is the $N \times 1$ Gaussian vector $(f(x_1), \dots, f(x_N))^T$. The conditional density of $\mathbf{y} = (y_1, \dots, y_N)^T$ given \mathbf{f} that we write $\mathbb{P}(\mathbf{y}|\mathbf{f})$ refers to the likelihood term. From the Bayes' rule, we write the posterior distribution, that is the conditional density of \mathbf{f} given \mathbf{y} and \mathbf{X} , as $\mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y}) \propto \mathbb{P}(\mathbf{f}|\mathbf{X})\mathbb{P}(\mathbf{y}|\mathbf{f})$.

In general, the covariance function $c(\cdot, \cdot)$ relies on a set of unknown hyper-parameters $\theta_c = \{\theta_c^j\}_{j=1}^p \in \mathbb{R}^p$. Thus, the hyper-parameters have to be estimated from observations in an optimal way in order to achieve best predictions. It is, indeed, very common to maximize the marginal likelihood $\mathbb{P}(\mathbf{y}|\mathbf{X})$ depending on θ_c . It can be achieved when marginalizing the posterior on \mathbf{f} , i.e.,

$\mathbb{P}(\mathbf{y}|\mathbf{X}) = \int_{\mathbb{R}^N} \mathbb{P}(\mathbf{f}|\mathbf{X})\mathbb{P}(\mathbf{y}|\mathbf{f})d\mathbf{f}$. This task is not trivial when the likelihood term $\mathbb{P}(\mathbf{y}|\mathbf{f})$ deviates from standard forms, which is the case for the classification framework. Ideally, the non-Gaussian posteriors can be approximated by Gaussian distributions [Minka \(2001\)](#); [Williams and Barber \(1998\)](#).

Methods based on GPCs are successful with low and medium dimensions, but are limited in high-dimensions ($N \ll d$) [Djolonga et al. \(2013\)](#). The most prominent weakness is usually the computational cost due to the inversion and the determinant of the $N \times N$ covariance matrix needed for evaluating any marginal likelihood. Furthermore, when d is large, the Euclidean distance between x and x' in \mathbb{R}^d : $\|x - x'\|_2$ becomes less informative. To overcome those issues, we suppose that the inputs lie on an embedded submanifold \mathcal{M} of low dimension [Calandra et al. \(2016\)](#); [Fradi et al. \(2018\)](#). We then propose to perform inference on embedded inputs $\mathbf{Z} = \Psi(\mathbf{X})$ where Ψ is a mapping (embedding) defined on a RKHS (Reproducing Kernel Hilbert Space) \mathcal{H}_K . The family of kernels $K(.,.)$ is again controlled by a vector of hyper-parameters $\theta_K = \{\theta_K^j\}_{j=1}^k \in \mathbb{R}^k$. One of the main advantages of this formulation is that the embedded submanifold \mathcal{M} (which is the image of \mathcal{X} by Ψ) has a lower dimension than \mathcal{X} , i.e., $M \ll d$ with an adaptive geometrical structure. As a consequence, we let the covariance function $c(.,.)$ operate directly on \mathcal{M} . To summarize, this formulation solves non-linearity from initial inputs, creates separability in the embedded submanifold and reduces the dimensionality.

The main goal in this chapter is to provide an efficient scalable approximation of the Bayesian inference related to the above dimension reduction framework. In this context, we will estimate the hyper-parameters of the covariance function $c(.,.)$ and the embedding with $K(.,.)$, jointly, by two different techniques. We now give more details about the proposed methods:

1. A novel set of equations and algorithms is developed in order to evaluate marginalized log-likelihoods. All details of the gradient vectors and the Hessian matrices are given. For selecting the hyper-parameters, we maximize the approximate marginal likelihoods using two iterative techniques:

gradient-descent and Newton-Raphson. Nonetheless, these optimization techniques usually have some limitations when dealing with non-convex cost functions [Carlin and Louis \(1997\)](#), in particular, the approximate marginal likelihood. We solve this problem and we show that our formulation requires a small number of iterations for maximizing marginal likelihoods [Karaboga and Basturk \(2008\)](#).

2. As an alternative to marginalization, we can simply maximize the posteriors on these hyper-parameters when computational resources increase. A common approach to tackle the non-convexity issue of the marginal likelihood is to use multiple starting points randomly selected from specific prior laws for the model hyper-parameters. Following this idea, we simulate from the posterior distribution that we are able to factorize across two separate posterior distributions for θ_c and θ_K . We proceed with sampling based on MCMC [Van Ravenzwaaij et al. \(2018\)](#). Finally, it is important to mention that the problem of high complexity and computational time, usually needed for ensuring the stationarity of Markov chains, will be efficiently solved thanks to the proposed embedded submanifold.

3.2 Gaussian process classifier

For GPc, we are interested in the target class "+1" probability satisfying $\pi(x) = \mathbb{P}(y = +1|f(x)) = \sigma(f(x))$ with an activation function $\sigma : \mathbb{R} \rightarrow [0, 1]$ usually refers to the sigmoid ($\sigma(t) = \frac{1}{1+\exp(-t)}$) or the probit ($\sigma(t) = F(t)$). Here F refers to the standard Gaussian cumulative distribution function.

3.2.1 Problem formulation

The likelihood term is the product of individual likelihoods, i.e.,

$$\begin{aligned} \mathbb{P}(\mathbf{y}|\mathbf{f}) &= \prod_{i=1}^N \mathbb{P}(y_i|f_i) \\ &= \prod_{i=1}^N \sigma(y_i f_i) \end{aligned} \tag{3.1}$$

According to our case, the posterior distribution of \mathbf{f} given \mathbf{X} and \mathbf{y} is proportional (\propto) to

$$\begin{aligned}\mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y}) &= \frac{\mathbb{P}(\mathbf{y}|\mathbf{f})\mathbb{P}(\mathbf{f}|\mathbf{X})}{\mathbb{P}(\mathbf{y}|\mathbf{X})} \\ &\propto \mathbb{P}(\mathbf{y}|\mathbf{f})\mathbb{P}(\mathbf{f}|\mathbf{X})\end{aligned}\quad (3.2)$$

where $\mathbb{P}(\mathbf{y}|\mathbf{X})$ is the exact marginal likelihood. From (3.2), the posterior is analytically intractable due to the likelihood term and need to be approximated, for instance, by a Gaussian distribution. As a solution, we introduce the Laplace approximation and the Expectation propagation methods. We also give the approximate marginal likelihood and the predictive distribution in both cases.

3.2.2 Laplace approximation

We firstly discuss details of Laplace approximation method for GPc.

3.2.2.1 Sampling functions and predictions

From the GPc definition, the prior law on \mathbf{f} satisfies

$$\mathbb{P}(\mathbf{f}|\mathbf{X}) = \mathcal{N}(\mathbf{f}|0, \mathbf{C}) \quad (3.3)$$

where $\mathbf{C} = c(\mathbf{X}, \mathbf{X})$. From (3.2), the log-posterior is simply proportional to

$$\log \mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y}) \propto \log \mathbb{P}(\mathbf{y}|\mathbf{f}) - \frac{1}{2} \mathbf{f}^T \mathbf{C}^{-1} \mathbf{f} \quad (3.4)$$

For the Laplace approximation, we firstly find the maximum a posteriori (MAP) estimator denoted $\hat{\mathbf{f}} = (\hat{f}_1, \dots, \hat{f}_N)^T$ from the Newton-Raphson method, with the iteration

$$\mathbf{f}^{t+1} = (\mathbf{C}^{-1} + \mathbf{W})^{-1}(\mathbf{W}\mathbf{f}^t + \nabla \mathbb{P}(\mathbf{y}|\mathbf{f}^t)) \quad (3.5)$$

for $t = 2, 3, \dots$. \mathbf{W} is the negative Hessian matrix of the likelihood term, i.e., \mathbf{W} is a $N \times N$ diagonal matrix with entries $\mathbf{W}_{ii} = - \left. \frac{\partial^2 \log p(y_i|f_i)}{\partial^2 f_i} \right|_{f_i=\hat{f}_i}$ and $\mathbb{P}(y_i|f_i) = \sigma(y_i f_i) = \frac{1}{1+e^{-y_i f_i}}$. Once we have estimated the MAP $\hat{\mathbf{f}}$, we can specify a Gaussian approximation of the posterior, when doing a second order Taylor expansion of $\log \mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y})$ around $\hat{\mathbf{f}}$, as

$$\hat{\mathbb{P}}(\mathbf{f}|\mathbf{y}, \mathbf{X}) = \mathcal{N}(\mathbf{f}|\hat{\mathbf{f}}, (\mathbf{C}^{-1} + \mathbf{W})^{-1}) \quad (3.6)$$

Given an unobserved input x^* , the predictive distribution at $f^* = f(x^*)$ can be approximated by

$$\hat{\mathbb{P}}(f^*|\mathbf{X}, \mathbf{y}, x^*) = \mathcal{N}(f^*|\mu(x^*), \sigma^2(x^*)) \quad (3.7)$$

with

$$\begin{cases} \mu(x^*) = \mathbf{C}_*^T \mathbf{C}^{-1} \hat{\mathbf{f}} \\ \sigma^2(x^*) = \mathbf{C}_{**} - \mathbf{C}_*^T (\mathbf{C} + \mathbf{W}^{-1})^{-1} \mathbf{C}_* \end{cases} \quad (3.8)$$

Using the moments of prediction, we approximate the predictor for $y^* = +1$ by

$$\bar{\pi}(x^*) = \int_{\mathbb{R}} \sigma(f^*) \hat{\mathbb{P}}(f^*|\mathbf{X}, \mathbf{y}, x^*) df^* \quad (3.9)$$

3.2.2.2 Optimizing hyper-parameters

We evaluate the approximate marginal likelihood $\hat{\mathbb{P}}(\mathbf{y}|\mathbf{X})$ instead of the exact marginal likelihood $\mathbb{P}(\mathbf{y}|\mathbf{X})$ given in the denominator of (3.2). This term usually depends on a vector of hyper-parameters $\theta_c = \{\theta_c^j\}_{j=1}^p \in \mathbb{R}^p$ associated to the covariance function $c(\cdot, \cdot)$, unknown and to be inferred. A common practice is to obtain point estimates of the hyper-parameters by maximizing the marginal (log) likelihood. Integrating out \mathbf{f} , the log-marginal likelihood is approximated by

$$l(\theta_c) = -\frac{1}{2} \hat{\mathbf{f}}^T \mathbf{C}^{-1} \hat{\mathbf{f}} + \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{f}}) - \frac{1}{2} \log |\mathcal{I} + \mathbf{W}^{\frac{1}{2}} \mathbf{C} \mathbf{W}^{\frac{1}{2}}| \quad (3.10)$$

Optimizing over θ_c requires the evaluation of first partial derivatives. The first partial derivatives of $l(\theta_c)$ with respect to θ_c^j satisfy

$$\frac{\partial l(\theta_c)}{\partial \theta_c^j} = \left. \frac{\partial l(\theta_c)}{\partial \theta_c^j} \right|_{\hat{\mathbf{f}}} + \sum_{i=1}^N \frac{\partial l(\theta_c)}{\partial \hat{f}_i} \frac{\partial \hat{f}_i}{\partial \theta_c^j} \quad (3.11)$$

The first term (explicit), obtained when we assume that $\hat{\mathbf{f}}$ (as well as \mathbf{W}) does not depend on θ_c , satisfies

$$\left. \frac{\partial l(\theta_c)}{\partial \theta_c^j} \right|_{\hat{\mathbf{f}}} = \frac{1}{2} \hat{\mathbf{f}}^T \mathbf{C}^{-1} \frac{\partial \mathbf{C}}{\partial \theta_c^j} \mathbf{C}^{-1} \hat{\mathbf{f}} - \frac{1}{2} \text{tr} \left[(\mathbf{C} + \mathbf{W}^{-1})^{-1} \frac{\partial \mathbf{C}}{\partial \theta_c^j} \right] \quad (3.12)$$

The second term (implicit), obtained when we suppose that only $\hat{\mathbf{f}}$ (as well as \mathbf{W}) depends on θ_c , is fully determined by

$$\frac{\partial l(\theta_c)}{\partial \hat{f}_i} = -\frac{1}{2} \left[(\mathbf{C}^{-1} + \mathbf{W})^{-1} \right]_{ii} \frac{\partial^3 \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{f}})}{\partial^3 \hat{f}_i} \quad (3.13)$$

and

$$\frac{\partial \hat{\mathbf{f}}}{\partial \theta_c^j} = (\mathcal{I} + \mathbf{C}\mathbf{W})^{-1} \frac{\partial \mathbf{C}}{\partial \theta_c^j} \nabla \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{f}}) \quad (3.14)$$

where ∇ is the gradient w.r.t. \mathbf{f} .

3.2.3 Expectation propagation

In this section, we give details for another method based on the Expectation propagation.

3.2.3.1 Sampling functions and predictions

The Expectation propagation is usually used to approximate marginal moments of the posterior [Minka \(2001\)](#). The key idea is to replace individual likelihoods by unnormalized Gaussian distributions. We then use the same notations and rewrite the posterior distribution over \mathbf{f} as the product of the prior and the likelihood terms

$$\mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}) = \frac{1}{L} \mathbb{P}(\mathbf{f}|\mathbf{X}) \prod_{i=1}^N \mathbb{P}(y_i|f_i) \quad (3.15)$$

where the normalization term is

$$\begin{aligned} L &= \mathbb{P}(\mathbf{y}|\mathbf{X}) \\ &= \int_{\mathbb{R}^N} \mathbb{P}(\mathbf{f}|\mathbf{X}) \prod_{i=1}^N \mathbb{P}(y_i|f_i) d\mathbf{f} \end{aligned} \quad (3.16)$$

We consider that $\mathbb{P}(y_i|f_i) = F(y_i f_i)$. To build this framework, we can approximate each individual likelihood by

$$\begin{aligned} \mathbb{P}(y_i|f_i) &\approx t_i(f_i|L_i, \tilde{\mu}_i, \tilde{\sigma}_i^2) \\ &= L_i \times \mathcal{N}(f_i|\tilde{\mu}_i, \tilde{\sigma}_i^2) \end{aligned} \quad (3.17)$$

However, the likelihood approximation should not be normalized since the exact likelihood do not have this property. The product of individual likelihood approximations is then $\prod_{i=1}^N L_i \times \mathcal{N}(\mathbf{f}|\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\Sigma}})$ where $\tilde{\boldsymbol{\mu}} = (\tilde{\mu}_1, \dots, \tilde{\mu}_N)^T$ and $\tilde{\boldsymbol{\Sigma}} = \text{diag}(\tilde{\sigma}_1^2, \dots, \tilde{\sigma}_N^2)$. Based on local approximations, the true posterior distribution

is approximated by

$$\begin{aligned}\hat{\mathbb{P}}(\mathbf{f}|\mathbf{y}, \mathbf{X}) &= \frac{1}{L} \mathbb{P}(\mathbf{f}|\mathbf{X}) \prod_{i=1}^N t_i(f_i|L_i, \tilde{\mu}_i, \tilde{\sigma}_i^2) \\ &= \mathcal{N}(\mathbf{f}|\mu, \Sigma)\end{aligned}\quad (3.18)$$

with $\mu = \Sigma \tilde{\Sigma}^{-1} \tilde{\mu}$ and $\Sigma = (\mathbf{C}^{-1} + \tilde{\Sigma}^{-1})^{-1}$.

To summarize, we give the main steps of the expectation propagation (EP) in Algorithm 8. Once we have estimated $\tilde{\mu}$ and $\tilde{\Sigma}$, the prediction moments are

$$\begin{cases} \mu(x^*) = \mathbf{C}_*^T \mathbf{C}^{-1} \mu = \mathbf{C}_*^T (\mathbf{C} + \tilde{\Sigma})^{-1} \tilde{\mu} \\ \sigma^2(x^*) = \mathbf{C}_{**} - \mathbf{C}_*^T (\mathbf{C} + \tilde{\Sigma})^{-1} \mathbf{C}_* \end{cases}\quad (3.19)$$

Therefore, the approximate predictor for $y^* = 1$ is

$$\bar{\pi}(x^*) = F\left(\frac{\mathbf{C}_*^T (\mathbf{C} + \tilde{\Sigma})^{-1} \tilde{\mu}}{\sqrt{1 + \mathbf{C}_{**} - \mathbf{C}_*^T (\mathbf{C} + \tilde{\Sigma})^{-1} \mathbf{C}_*}}\right)\quad (3.20)$$

3.2.3.2 Optimizing hyper-parameters

The marginal likelihood can be found from the normalization of (3.18) as

$$L_{\text{EP}} = \hat{\mathbb{P}}(\mathbf{y}|\mathbf{X}) = \int_{\mathbb{R}^N} p(\mathbf{f}|\mathbf{X}) \prod_{i=1}^N (L_i \times \mathcal{N}(f_i|\tilde{\mu}_i, \tilde{\sigma}_i^2)) d\mathbf{f}\quad (3.21)$$

Consequently, its logarithm satisfies

$$l(\theta_c) = \log(L_{\text{EP}}) = -\frac{1}{2} \log |\mathbf{C} + \tilde{\Sigma}| - \frac{1}{2} \tilde{\mu}^T (\mathbf{C} + \tilde{\Sigma})^{-1} \tilde{\mu} + B\quad (3.22)$$

where B comes from the normalization constants L_i , satisfying $B = \sum_{i=1}^N \log L_i$. Luckily, it turns out that implicit terms in the derivatives, being a function of θ_c , is exactly zero. More details are given in Seeger (2005). Consequently, we only have to take account of the explicit term

$$\frac{\partial l(\theta_c)}{\partial \theta_c^j} = \frac{1}{2} \tilde{\mu}^T (\mathbf{C} + \tilde{\Sigma})^{-1} \frac{\partial \mathbf{C}}{\partial \theta_c^j} (\mathbf{C} + \tilde{\Sigma})^{-1} \tilde{\mu} - \frac{1}{2} \text{tr} \left[(\mathbf{C} + \tilde{\Sigma})^{-1} \frac{\partial \mathbf{C}}{\partial \theta_c^j} \right]\quad (3.23)$$

3.3 The embedded submanifold

We partially introduce some technical results and we move to functions (mappings) that can be expressed in terms of expansions. Let us consider a nonnegative real-valued kernel $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+$ with its corresponding RKHS \mathcal{H}_K . We restrict ourselves to a class of kernels satisfying $K(x, x') = \langle \Psi(x), \Psi(x') \rangle_2$, depending on

Algorithm 8: EP.

- 1: Choose one $t_i(f_i|L_i, \tilde{\mu}_i, \tilde{\sigma}_i^2)$ to update.
- 2: Compute the cavity distribution of f_i

$$\hat{\mathbb{P}}_{-i}(f_i) \propto \frac{\hat{\mathbb{P}}(f_i|\mathbf{y}, \mathbf{X})}{t_i(f_i|L_i, \tilde{\mu}_i, \tilde{\sigma}_i^2)} = \mathcal{N}(f_i|\mu_{-i}, \sigma_{-i}^2)$$

where

$$\hat{\mathbb{P}}(f_i|\mathbf{y}, \mathbf{X}) = \mathcal{N}(f_i|\mu_i, \sigma_i^2 = \Sigma_{ii})$$

$$\sigma_{-i}^2 = (\sigma_i^{-2} - \tilde{\sigma}_i^{-2})^{-1}$$

$$\mu_{-i} = \sigma_{-i}^2(\sigma_i^{-2}\mu_i - \tilde{\sigma}_i^{-2}\tilde{\mu}_i)$$

- 3: Define $\mathbb{P}_i(f_i)$, the pseudo-exact posterior marginal distribution of f_i , as

$$\mathbb{P}_i(f_i) = \mathbb{P}(y_i|f_i)\hat{\mathbb{P}}_{-i}(f_i)$$

- 4: Compute $\hat{\mathbb{P}}(f_i) = \hat{L}_i \times \mathcal{N}(f_i|\hat{\mu}_i, \hat{\sigma}_i^2)$ by minimizing the Kullback-Leibler (K-L) divergence

$$(\hat{L}_i, \hat{\mu}_i, \hat{\sigma}_i^2) = \underset{(\hat{L}_i, \hat{\mu}_i, \hat{\sigma}_i^2)}{\operatorname{argmin}} \operatorname{K-L}(\mathbb{P}_i(f_i) || \hat{\mathbb{P}}(f_i))$$

giving the following marginal moments

$$\hat{L}_i = F(l_i)$$

$$\hat{\sigma}_i^2 = \sigma_{-i}^2 - \frac{\sigma_{-i}^4 \mathcal{N}(l_i|0,1)}{(1+\sigma_{-i}^2)F(l_i)} (l_i + \frac{\mathcal{N}(l_i|0,1)}{F(l_i)})$$

$$\hat{\mu}_i = \mu_{-i} + \frac{y_i \sigma_{-i}^2 \mathcal{N}(l_i|0,1)}{F(l_i) \sqrt{1+\sigma_{-i}^2}}$$

$$l_i = \frac{y_i \mu_{-i}}{\sqrt{1+\sigma_{-i}^2}}$$

- 5: Update $(L_i, \tilde{\mu}_i, \tilde{\sigma}_i^2)$ with $t_i(f_i|L_i, \tilde{\mu}_i, \tilde{\sigma}_i^2) = \frac{\hat{\mathbb{P}}(f_i)}{\hat{\mathbb{P}}_{-i}(f_i)}$ so that

$$\tilde{\mu}_i = \tilde{\sigma}_i^2(\hat{\sigma}_i^{-2}\hat{\mu}_i - \sigma_{-i}^{-2}\mu_{-i})$$

$$\tilde{\sigma}_i^2 = (\hat{\sigma}_i^{-2} - \sigma_{-i}^{-2})^{-1}$$

$$L_i = \hat{L}_i \sqrt{2\pi(\sigma_{-i}^2 + \tilde{\sigma}_i^2)} \exp\left(\frac{1}{2} \frac{(\mu_{-i} - \tilde{\mu}_i)^2}{\sigma_{-i}^2 + \tilde{\sigma}_i^2}\right)$$

local shape hyper-parameters $\theta_K = \{\theta_K^j\}_{j=1}^k \in \mathbb{R}^k$ where Ψ is an unknown non-linear mapping defined from \mathcal{X} to \mathcal{M} . The "Representer Theorem" [Schölkopf et al. \(2001\)](#) states that solutions of a large class of optimization problems can be expressed as kernel expansions over the sample points. Since our observations are high-correlated, that issue does not allow us to search a good configuration from the "Representer Theorem" directly. However, the proposed technique succeeds to find the optimal partitions with their local structures. For this purpose, we adopt the following result deduced from the "Representer Theorem".

Lemma 3.1

We assume that there exists a partition of \mathcal{X} with centers $\{c_i\}_i$ and an arbitrary empirical risk function E , defined on \mathcal{H}_K , controlled by data and regularization terms. Then, any function satisfying $\hat{\Psi} = \operatorname{argmin}_{\Psi \in \mathcal{H}_K} E(\Psi)$, admits a representation of the form

$$\hat{\Psi}(\cdot) = \sum_{i=1}^N \alpha_i K(\cdot, c_i) \quad (3.24)$$



Note that the mapping $\hat{\Psi}$ is a linear combination of some kernels $K(\cdot, c_i)$ centered on $\{c_i\}_i$. A proposed approximation of $\hat{\Psi}(x)$ is then $\Psi(x) = \sum_{j=1}^M \alpha_j K(x, c_j)$ for $M < N$. Now, considering the parametrized version $\phi_j : \mathcal{X} \rightarrow \mathbb{R}_+$; $x \mapsto K(x, c_j)$, we rewrite

$$\Psi(x) = \sum_{j=1}^M \alpha_j \phi_j(x) \quad (3.25)$$

From the last expansion, we only need to find α_j in order to determine an approximation of $\hat{\Psi}$. If we note $\alpha = [\alpha_1, \dots, \alpha_M]$ and $\phi(x) = (\phi_1(x), \dots, \phi_M(x))^T$, we can reformulate the approximation as $\Psi(x) = \alpha \phi(x)$, which implies that $\alpha^T \Psi(x) = (\alpha^T \alpha) \phi(x)$. If α is orthonormal (i.e., $\alpha^T \alpha = \mathcal{I}$), we simply get $\phi(x) = \alpha^T \Psi(x)$. We suppose that the configuration of centers $\{c_j\}_{j=1}^M$ can be obtained by any unsupervised clustering method (e.g., k-means [Hamerly and Drake \(2015\)](#)) applied to the inputs \mathbf{X} . Given $\{c_j\}_{j=1}^M$, the orthonormality of α (i.e., $\langle \alpha_j, \alpha_h \rangle_2 = \delta_{jh}$) heavily depends on the kernels basis ϕ_j (and their hyper-parameters θ_K as well) since $\langle \alpha_j, \alpha_h \rangle_2 = \langle \Psi(c_j), \Psi(c_h) \rangle_2 = K(c_j, c_h) = \phi_h(c_j)$. Finally, the function Ψ detailed in (3.25), maps the d -dimensional inputs x_1, \dots, x_N

into the M -dimensional vectors denoted by z_1, \dots, z_N where $M \ll d$ (since $N \ll d$ and $M < N$).

3.4 Scalable Gaussian process classifier

We define a scalable GPc (SGPc) $r : \mathcal{M} \subseteq \mathbb{R}^M \rightarrow \mathbb{R}$, as a classical GPc $f = r \circ \Psi : \mathcal{X} \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$. The covariance function of r denoted $\tilde{c}(\cdot, \cdot)$ is more expressive than $c(\cdot, \cdot)$, satisfying $\tilde{c}(x, x') = c(\Psi(x), \Psi(x'))$. Besides, it operates on \mathcal{M} and depends on θ_c and θ_K , jointly. For the reminder, we recover the $N \times M$ matrix of transformed inputs $\mathbf{Z} = [z_1, \dots, z_N]^T = \Psi(\mathbf{X})$ and the $N \times 1$ transformed Gaussian vector $\mathbf{r} = (r_1, \dots, r_N)^T = r(\Psi(\mathbf{X}))$. Accordingly, the posterior distribution of \mathbf{f} , given in (3.2), becomes a posterior at \mathbf{r} satisfying

$$\begin{aligned} \mathbb{P}(\mathbf{r}|\mathbf{Z}, \mathbf{y}) &= \frac{\mathbb{P}(\mathbf{r}|\mathbf{Z})\mathbb{P}(\mathbf{y}|\mathbf{r})}{\mathbb{P}(\mathbf{y}|\mathbf{Z})} \\ &\propto \mathbb{P}(\mathbf{r}|\mathbf{Z})\mathbb{P}(\mathbf{y}|\mathbf{r}) \end{aligned} \quad (3.26)$$

where the term $\mathbb{P}(\mathbf{y}|\mathbf{Z})$ is called again the true marginal likelihood.

3.4.1 Posterior distribution approximations and predictions

We give details on how to approximate the posterior distribution in connection with the proposed mapping (embedding) Ψ .

3.4.1.1 Scalable Laplace approximation (SLA)

Let $\hat{\mathbf{r}}$ be the MAP of $\mathbb{P}(\mathbf{r}|\mathbf{Z}, \mathbf{y})$, i.e., $\hat{\mathbf{r}} = (\hat{r}_1, \dots, \hat{r}_N)^T = \arg \max_{\mathbf{r}} \mathbb{P}(\mathbf{r}|\mathbf{Z}, \mathbf{y})$. As for the Laplace approximation, we can find $\hat{\mathbf{r}}$, iteratively, according to

$$\mathbf{r}^{t+1} = (\tilde{\mathbf{C}}^{-1} + \tilde{\mathbf{W}})^{-1}(\tilde{\mathbf{W}}\mathbf{r}^t + \nabla \log \mathbb{P}(\mathbf{y}|\mathbf{r}^t)) \quad (3.27)$$

Let $\tilde{\mathbf{C}} = \tilde{c}(\mathbf{X}, \mathbf{X})$ denote the $N \times N$ matrix with element i, j equal to $\tilde{c}(x_i, x_j)$ and $\tilde{\mathbf{W}}$ is a $N \times N$ diagonal matrix with entries $\tilde{W}_{ii} = \frac{\exp(-\hat{r}_i)}{(1+\exp(-\hat{r}_i))^2}$. Therefore, a Gaussian approximation of the posterior distribution in (3.26) is given by

$$\hat{\mathbb{P}}(\mathbf{r}|\mathbf{Z}, \mathbf{y}) = \mathcal{N}(\mathbf{r}|\hat{\mathbf{r}}, (\tilde{\mathbf{C}}^{-1} + \tilde{\mathbf{W}})^{-1}) \quad (3.28)$$

Given a test input $z^* = \Psi(x^*)$, the rules for conditioning of multivariate Gaussian distributions allow us to derive the approximate predictive distribution of $r^* =$

$r(z^*)$ given \mathbf{Z} , \mathbf{y} and z^* as

$$\hat{\mathbb{P}}(r^*|\mathbf{Z}, \mathbf{y}, z^*) = \mathcal{N}(r^*|\mu(z^*), \sigma^2(z^*)) \quad (3.29)$$

with

$$\begin{cases} \mu(z^*) = \tilde{\mathbf{C}}_*^T \tilde{\mathbf{C}}_*^{-1} \hat{\mathbf{r}} \\ \sigma^2(z^*) = \tilde{\mathbf{C}}_{**} - \tilde{\mathbf{C}}_*^T (\tilde{\mathbf{C}} + \tilde{\mathbf{W}}^{-1})^{-1} \tilde{\mathbf{C}}_* \end{cases} \quad (3.30)$$

where $\tilde{\mathbf{C}}_{**} = \tilde{c}(x^*, x^*)$ and $\tilde{\mathbf{C}}_* = \tilde{c}(\mathbf{X}, x^*)$ is the $N \times 1$ vector with i -th element equal to $\tilde{c}(x_i, x^*)$. Then, we can approximate the predictor of first class $\mathbb{P}(y^* = +1|\mathbf{Z}, \mathbf{y}, z^*)$, that is the conditional probability that $y^* = +1$, by

$$\bar{\pi}(z^*) = \int_{\mathbb{R}} \sigma(r^*) \hat{\mathbb{P}}(r^*|\mathbf{Z}, \mathbf{y}, z^*) dr^* \quad (3.31)$$

3.4.1.2 Scalable Expectation propagation (SEP)

As for Expectation propagation, SEP consists in using un-normalized Gaussian approximations of individual likelihoods $\mathbb{P}(y_i|r_i) = F(y_i r_i)$, $i \in \{1, \dots, N\}$. The i -th individual likelihood is approximated by un-normalized Gaussian distribution $\tilde{L}_i \times \mathcal{N}(r_i|\tilde{v}_i, \tilde{s}_i)$ where $\tilde{L}_i > 0$. Using the principle of Expectation propagation, the posterior distribution in (3.26) can be approximated by

$$\hat{\mathbb{P}}(\mathbf{r}|\mathbf{Z}, \mathbf{y}) = \mathcal{N}(\mathbf{r} | (\mathcal{I} + \tilde{\mathbf{S}} \tilde{\mathbf{C}}^{-1})^{-1} \tilde{\mathbf{v}}, (\tilde{\mathbf{C}}^{-1} + \tilde{\mathbf{S}}^{-1})^{-1}) \quad (3.32)$$

where $\tilde{\mathbf{v}} = (\tilde{v}_1, \dots, \tilde{v}_N)^T$ and $\tilde{\mathbf{S}} = \text{diag}(\tilde{s}_1, \dots, \tilde{s}_N)$ are obtained from Algorithm 8. From (3.32), the conditional mean and variance of predictive distribution r^* given \mathbf{Z} , \mathbf{y} and z^* are

$$\begin{cases} \mu(z^*) = \tilde{\mathbf{C}}_*^T (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \tilde{\mathbf{v}} \\ \sigma^2(z^*) = \tilde{\mathbf{C}}_{**} - \tilde{\mathbf{C}}_*^T (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \tilde{\mathbf{C}}_* \end{cases} \quad (3.33)$$

Then, the conditional probability $\mathbb{P}(y^* = +1|\mathbf{y}, \mathbf{Z}, z^*)$ is approximated by

$$\bar{\pi}(z^*) = F\left(\frac{\tilde{\mathbf{C}}_*^T (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \tilde{\mathbf{v}}}{\sqrt{1 + \tilde{\mathbf{C}}_{**} - \tilde{\mathbf{C}}_*^T (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \tilde{\mathbf{C}}_*}}\right) \quad (3.34)$$

To summarize, we have defined a family of classifiers using a latent SGPC and approximations of the exact posterior distribution depending on a set of hyper-parameters θ_c and θ_K typically unknown to be estimated. We state that some properties such as the stationarity of the GPc and the correlation between trans-

formed inputs in the feature space \mathcal{M} are controlled by θ_c and θ_K , respectively.

3.4.2 Optimizing hyper-parameters

Let $\theta = (\theta_c, \theta_K) \in \mathbb{R}^{p+k}$ denote the hyper-parameters of the covariance function $c(\cdot, \cdot)$ and those of the kernel $K(\cdot, \cdot)$, jointly. We will provide useful formulas and simplifications for evaluating the approximate marginal likelihoods $\hat{\mathbb{P}}(\mathbf{y}|\mathbf{Z})$ for both SLA and SEP.

3.4.2.1 The first partial derivatives of $\log \hat{\mathbb{P}}(\mathbf{y}|\mathbf{Z})$

In this section, we focus on terms required for computing first partial derivatives.

First case : SLA :

Integrating out the latent values \mathbf{r} in (3.28), yields an approximation of the log-marginal likelihood $\log \mathbb{P}(\mathbf{y}|\mathbf{Z})$

$$l(\theta) = \log \hat{\mathbb{P}}(\mathbf{y}|\mathbf{Z}) = -\frac{1}{2} \hat{\mathbf{r}}^T \tilde{\mathbf{C}}^{-1} \hat{\mathbf{r}} + \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{r}}) - \frac{1}{2} \log |\mathcal{I} + \tilde{\mathbf{W}}^{\frac{1}{2}} \tilde{\mathbf{C}} \tilde{\mathbf{W}}^{\frac{1}{2}}| \quad (3.35)$$

The first partial derivative of $l(\theta)$ with respect to θ_c^j satisfies

$$\frac{\partial l(\theta)}{\partial \theta_c^j} = \left. \frac{\partial l(\theta)}{\partial \theta_c^j} \right|_{\hat{\mathbf{r}}} + \sum_{i=1}^N \frac{\partial l(\theta)}{\partial \hat{r}_i} \frac{\partial \hat{r}_i}{\partial \theta_c^j} \quad (3.36)$$

Here

$$\left. \frac{\partial l(\theta)}{\partial \theta_c^j} \right|_{\hat{\mathbf{r}}} = \frac{1}{2} \hat{\mathbf{r}}^T \tilde{\mathbf{C}}^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} \tilde{\mathbf{C}}^{-1} \hat{\mathbf{r}} - \frac{1}{2} \text{tr} [(\tilde{\mathbf{C}} + \tilde{\mathbf{W}}^{-1})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j}] \quad (3.37)$$

means that the partial derivatives of $l(\theta)$ are calculated as if $\hat{\mathbf{r}}$ (and thus $\tilde{\mathbf{W}}$) does not depend on θ_c . The second term in (3.36) is determined by

$$\frac{\partial l(\theta)}{\partial \hat{r}_i} = -\frac{1}{2} [(\tilde{\mathbf{C}}^{-1} + \tilde{\mathbf{W}})^{-1}]_{ii} \frac{\partial^3 \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{r}})}{\partial^3 \hat{r}_i} \quad (3.38)$$

and

$$\frac{\partial \hat{\mathbf{r}}}{\partial \theta_c^j} = (\mathcal{I} + \tilde{\mathbf{C}} \tilde{\mathbf{W}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} \nabla \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{r}}) \quad (3.39)$$

We now provide the partial derivatives of $l(\theta)$ w.r.t. θ_K . The first partial derivative of $l(\theta)$ with respect to θ_K^j satisfies

$$\frac{\partial l(\theta)}{\partial \theta_K^j} = \left. \frac{\partial l(\theta)}{\partial \theta_K^j} \right|_{\hat{\mathbf{r}}} + \sum_{i=1}^N \frac{\partial l(\theta)}{\partial \hat{r}_i} \frac{\partial \hat{r}_i}{\partial \theta_K^j} \quad (3.40)$$

Here

$$\left. \frac{\partial l(\theta)}{\partial \theta_K^j} \right|_{\hat{\mathbf{r}}} = \frac{1}{2} \hat{\mathbf{r}}^T \tilde{\mathbf{C}}^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \mathbf{Z}} \frac{\partial \mathbf{Z}}{\partial \theta_K^j} \tilde{\mathbf{C}}^{-1} \hat{\mathbf{r}} - \frac{1}{2} \text{tr} \left[(\tilde{\mathbf{C}} + \tilde{\mathbf{W}}^{-1})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \mathbf{Z}} \frac{\partial \mathbf{Z}}{\partial \theta_K^j} \right] \quad (3.41)$$

which means that the partial derivatives are calculated as if $\hat{\mathbf{r}}$ (and thus $\tilde{\mathbf{W}}$) does not depend on θ_K . The second term relies on

$$\frac{\partial \hat{\mathbf{r}}}{\partial \theta_K^j} = (\mathcal{I} + \tilde{\mathbf{C}} \tilde{\mathbf{W}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \mathbf{Z}} \frac{\partial \mathbf{Z}}{\partial \theta_K^j} \nabla \log \mathbb{P}(\mathbf{y} | \hat{\mathbf{r}}) \quad (3.42)$$

The quantity $\frac{\partial \tilde{\mathbf{C}}}{\partial \mathbf{Z}}$ is the gradient of the modified covariance matrix with respect to the M -dimensional inputs $\mathbf{Z} = \Psi(\mathbf{X})$.

Second case : SEP :

From (3.32), the approximate log-marginal likelihood for SEP approximation is

$$l(\theta) = -\frac{1}{2} \log |\tilde{\mathbf{C}} + \tilde{\mathbf{S}}| - \frac{1}{2} \tilde{\mathbf{v}}^T (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \tilde{\mathbf{v}} + \tilde{B} \quad (3.43)$$

where \tilde{B} comes from the normalization constants \tilde{L}_i , satisfying $\tilde{B} = \sum_{i=1}^N \log \tilde{L}_i$.

Proposition 3.1

For SEP, it holds that

$$\frac{\partial l(\theta)}{\partial \theta_c^j} = \frac{1}{2} \tilde{\mathbf{v}}^T (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \tilde{\mathbf{v}} - \frac{1}{2} \text{tr} \left[(\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} \right] \quad (3.44)$$

and

$$\frac{\partial l(\theta)}{\partial \theta_K^j} = \frac{1}{2} \tilde{\mathbf{v}}^T (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \mathbf{Z}} \frac{\partial \mathbf{Z}}{\partial \theta_K^j} (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \tilde{\mathbf{v}} - \frac{1}{2} \text{tr} \left[(\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \mathbf{Z}} \frac{\partial \mathbf{Z}}{\partial \theta_K^j} \right] \quad (3.45)$$

Proof We use the fact that implicit terms vanish when differentiating $l(\theta)$ and the chain rule $\frac{\partial \tilde{\mathbf{C}}}{\partial \theta_K^j} = \frac{\partial \tilde{\mathbf{C}}}{\partial \mathbf{Z}} \frac{\partial \mathbf{Z}}{\partial \theta_K^j}$.

3.4.2.2 The second partial derivatives of $\log \hat{\mathbb{P}}(\mathbf{y} | \mathbf{Z})$

In this section, we focus on terms required for computing second partial derivatives.

First case : SLA :

Proposition 3.2

For SLA, the second partial derivatives of the approximate log-marginal likelihood with respect to $\{\theta_c^h, \theta_c^j\}$ can be analytically composed as

$$\frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_c^j} = \left. \frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_c^j} \right|_{\hat{\mathbf{r}}} + \sum_{i=1}^N \frac{\partial^2 l(\theta)}{\partial^2 \hat{r}_i} \frac{\partial^2 \hat{r}_i}{\partial \theta_c^h \partial \theta_c^j} \quad (3.46)$$

The first term is

$$\begin{aligned} \left. \frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_c^j} \right|_{\hat{\mathbf{r}}} &= -\frac{1}{2} \hat{\mathbf{r}}^T \left(\tilde{\mathbf{C}}^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^h} \tilde{\mathbf{C}}^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} \tilde{\mathbf{C}}^{-1} - \tilde{\mathbf{C}}^{-1} \frac{\partial^2 \tilde{\mathbf{C}}}{\partial \theta_c^h \partial \theta_c^j} \tilde{\mathbf{C}}^{-1} \right. \\ &\quad \left. + \tilde{\mathbf{C}}^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} \tilde{\mathbf{C}}^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^h} \tilde{\mathbf{C}}^{-1} \right) \hat{\mathbf{r}} \\ &\quad + \frac{1}{2} \text{tr} \left[(\tilde{\mathbf{C}} + \tilde{\mathbf{W}}^{-1})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^h} (\tilde{\mathbf{C}} + \tilde{\mathbf{W}}^{-1})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} \right. \\ &\quad \left. - (\tilde{\mathbf{C}} + \tilde{\mathbf{W}}^{-1})^{-1} \frac{\partial^2 \tilde{\mathbf{C}}}{\partial \theta_c^h \partial \theta_c^j} \right] \end{aligned} \quad (3.47)$$

which means that the partial derivatives of $l(\theta)$ are calculated as if $\hat{\mathbf{r}}$ (and thus $\tilde{\mathbf{W}}$) did not depend on θ_c . The second term depends on

$$\begin{aligned} \frac{\partial^2 l(\theta)}{\partial^2 \hat{r}_i} &= \frac{1}{2} \left[(\tilde{\mathbf{C}}^{-1} + \tilde{\mathbf{W}})^{-1} \frac{\partial \tilde{\mathbf{W}}}{\partial \hat{r}_i} (\tilde{\mathbf{C}}^{-1} + \tilde{\mathbf{W}})^{-1} \right]_{ii} \frac{\partial^3 \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{r}})}{\partial^3 \hat{r}_i} \\ &\quad - \frac{1}{2} \left[(\tilde{\mathbf{C}}^{-1} + \tilde{\mathbf{W}})^{-1} \right]_{ii} \times \frac{\partial^4 \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{r}})}{\partial^4 \hat{r}_i} \end{aligned} \quad (3.48)$$

and

$$\frac{\partial^2 \hat{\mathbf{r}}}{\partial \theta_c^h \partial \theta_c^j} = (\mathcal{I} + \tilde{\mathbf{C}} \frac{\partial \tilde{\mathbf{W}}}{\partial \hat{\mathbf{r}}})^{-1} \left(\frac{\partial^2 \tilde{\mathbf{C}}}{\partial \theta_c^h \partial \theta_c^j} \nabla \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{r}}) - \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} \tilde{\mathbf{W}} \frac{\partial \hat{\mathbf{r}}}{\partial \theta_c^h} - \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^h} \tilde{\mathbf{W}} \frac{\partial \hat{\mathbf{r}}}{\partial \theta_c^j} \right) \quad (3.49)$$

Proof For $\left. \frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_c^j} \right|_{\hat{\mathbf{r}}}$ and $\frac{\partial^2 l(\theta)}{\partial^2 \hat{r}_i}$, we differentiate the first partial derivatives given in (3.37) and (3.38), respectively. For $\frac{\partial^2 \hat{\mathbf{r}}}{\partial \theta_c^h \partial \theta_c^j}$, we differentiate $\hat{\mathbf{r}} = \tilde{\mathbf{C}} \nabla \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{r}})$ two times and we use the following chain rules:

$$\begin{aligned} \frac{\partial}{\partial \theta_c^h} &= \frac{\partial}{\partial \hat{\mathbf{r}}} \frac{\partial \hat{\mathbf{r}}}{\partial \theta_c^h}, \quad \frac{\partial}{\partial \theta_c^j} = \frac{\partial}{\partial \hat{\mathbf{r}}} \frac{\partial \hat{\mathbf{r}}}{\partial \theta_c^j}, \quad \frac{\partial^2}{\partial \theta_c^h \partial \theta_c^j} = \frac{\partial^2}{\partial \hat{\mathbf{r}} \partial \theta_c^h \partial \theta_c^j}, \quad \frac{\partial \nabla \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{r}})}{\partial \hat{\mathbf{r}}} = -\tilde{\mathbf{W}} \text{ and } \frac{\partial^2 \nabla \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{r}})}{\partial^2 \hat{\mathbf{r}}} \\ &= -\frac{\partial \tilde{\mathbf{W}}}{\partial \hat{\mathbf{r}}}. \end{aligned}$$

We Keep the same steps for evaluating the second partial derivatives $\frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_c^j}$ when replacing $\frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j}$ by $\frac{\partial \tilde{\mathbf{C}}}{\partial \mathbf{Z}} \frac{\partial \mathbf{Z}}{\partial \theta_c^j}$ and $\frac{\partial^2 \tilde{\mathbf{C}}}{\partial \theta_c^h \partial \theta_c^j}$ by $\frac{\partial^2 \tilde{\mathbf{C}}}{\partial^2 \mathbf{Z}} \frac{\partial^2 \mathbf{Z}}{\partial \theta_c^h \partial \theta_c^j}$ in (3.47) and (3.49).

For mixed derivatives $\frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_c^j}$ (likewise $\frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_c^j}$), we just replace $\frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j}$ by $\frac{\partial \tilde{\mathbf{C}}}{\partial \mathbf{Z}} \frac{\partial \mathbf{Z}}{\partial \theta_c^j}$ and $\frac{\partial^2 \tilde{\mathbf{C}}}{\partial \theta_c^h \partial \theta_c^j}$ by $\frac{\partial}{\partial \theta_c^h} \left(\frac{\partial \tilde{\mathbf{C}}}{\partial \mathbf{Z}} \frac{\partial \mathbf{Z}}{\partial \theta_c^j} \right)$ in the same equations.

First case : SEP :

Proposition 3.3

For SEP, the second partial derivatives of the approximate log-marginal like-

likelihood with respect to $\{\theta_c^h, \theta_c^j\}$ satisfy

$$\begin{aligned} \frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_c^j} &= -\frac{1}{2} \tilde{\mathbf{v}}^T \left((\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^h} (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \right. \\ &\quad - (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial^2 \tilde{\mathbf{C}}}{\partial \theta_c^h \partial \theta_c^j} (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \\ &\quad \left. + (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^h} (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \right) \tilde{\mathbf{v}} \\ &\quad + \frac{1}{2} \text{tr} \left[(\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^h} (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial \tilde{\mathbf{C}}}{\partial \theta_c^j} - (\tilde{\mathbf{C}} + \tilde{\mathbf{S}})^{-1} \frac{\partial^2 \tilde{\mathbf{C}}}{\partial \theta_c^h \partial \theta_c^j} \right] \end{aligned} \quad (3.50)$$

We can also get $\frac{\partial^2 l(\theta)}{\partial \theta_K^h \partial \theta_K^j}$ and $\frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_K^j}$ (likewise $\frac{\partial^2 l(\theta)}{\partial \theta_K^h \partial \theta_c^j}$) for SEP by using the same chain rules as for SLA only for the first term since the second term vanished when differentiating the approximate log-marginal likelihood.

3.4.2.3 Numerical methods and sampling

In the marginal likelihood estimation, point estimates of the hyper-parameters are obtained by maximizing the log-marginal likelihood with respect to $\theta = (\theta_c, \theta_K)$, i.e., finding

$$\hat{\theta} = \underset{\theta}{\text{argmax}} l(\theta) \quad (3.51)$$

We refer to the resulting hyper-parameters as type-II MLEs. Let $\nabla l(\theta) = \left(\left\{ \frac{\partial l(\theta)}{\partial \theta_c^j} \right\}_{j=1}^p, \left\{ \frac{\partial l(\theta)}{\partial \theta_K^j} \right\}_{j=1}^k \right)^T$ denote the gradient vector of $l(\theta)$. Since there is no analytic solution when solving $\nabla l(\theta) = 0$, we make use of two iterative methods. We first consider the gradient-descent according to Algorithm 1 where the cost function to be minimized is the negative approximate log-marginal likelihoods $-l(\theta)$ given in (3.35) and (3.43) for both SLA and SEP, respectively. The second procedure consists in finding an explicit expression of the Hessian matrix $\nabla^2 l(\theta)$ formed by second partial derivatives of the log-marginal likelihood, i.e.,

$$\nabla^2 l(\theta) = \begin{bmatrix} \left\{ \frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_c^j} \right\}_{j,h=1}^p & \left\{ \frac{\partial^2 l(\theta)}{\partial \theta_c^h \partial \theta_K^j} \right\}_{j,h=1}^{p,k} \\ \left\{ \frac{\partial^2 l(\theta)}{\partial \theta_K^h \partial \theta_c^j} \right\}_{j,h=1}^{k,p} & \left\{ \frac{\partial^2 l(\theta)}{\partial \theta_K^h \partial \theta_K^j} \right\}_{j,h=1}^k \end{bmatrix} \quad (3.52)$$

which allows the use of the Newton-Raphson given in Algorithm 2.

Note that conventional optimizations might not find the best local maximum, thus failing to find the most appropriate value of θ . Moreover, by selecting only one

candidate for θ robustness and uncertainty quantification are lost in the process. Hence, we adopt a Bayesian point of view on $\theta = (\theta_c, \theta_K)$ and assign priors denoted by $\mathbb{P}(\theta_c)$ and $\mathbb{P}(\theta_K)$ in order to find the type II-MAP that maximizes the posterior distributions on θ (since the notation MAP was used for $\hat{\mathbf{r}}$). We then sample posterior values of θ with MCMC. We simulate from $\mathbb{P}(\theta_c, \theta_K | \mathbf{r}, \mathbf{y})$ which factorizes across θ_c and θ_K . From (3.26) we get two separate conditional posterior distributions

$$\mathbb{P}(\theta_K | \mathbf{r}, \mathbf{y}) \propto \mathbb{P}(\mathbf{y} | \mathbf{r}) \mathbb{P}(\theta_K) \quad (3.53)$$

and

$$\mathbb{P}(\theta_c | \mathbf{r}) \propto \mathbb{P}(\mathbf{r} | \mathbf{Z}) \mathbb{P}(\theta_c) \quad (3.54)$$

For SEP, we simply replace $\mathbb{P}(\mathbf{y} | \mathbf{r})$ by its approximation $\prod_{i=1}^N \tilde{L}_i \times \mathcal{N}(\mathbf{r} | \tilde{\mathbf{v}}, \tilde{\mathbf{S}})$ given in (3.53). Sampling from any of these distributions is carried out by using some proposal distribution, for instance a Gaussian in the Metropolis-Hastings (MH) illustrated in Algorithm 5, which is updated during the early iterations of SLA and SEP algorithms in order to tune the acceptance rate. Furthermore, the local shape hyper-parameters $\{\theta_K^j\}_{j=1}^k$ exhibit additional conditional independencies so that we can sample them independently in separate blocks Neal (1997). An accepted state for SGPc hyper-parameters requires an update of the proposal distribution when updating $(\hat{\mathbf{r}}, \tilde{\mathbf{W}})$ for SLA and $(\tilde{\mathbf{v}}, \tilde{\mathbf{S}})$ for SEP. This holds for algorithms described in Section 3.4.1.1 and Section 3.4.1.2 in order to find the posterior approximation $\hat{\mathbb{P}}(\mathbf{r} | \mathbf{Z}, \mathbf{y})$. The prior laws of θ_c and θ_K will be carefully fixed for applications in connection with the kernel $K(., .)$ and the covariance function $c(., .)$ choices.

3.5 Applications

The main goal of our experiments is to evaluate the proposed SGPc using both synthetic and real data.

Covariance function and kernel. In our experiments, the covariance function of GPc is the squared exponential: $c_{\delta^2, \gamma}(x, x') = \delta^2 \exp\left(-\frac{\|x-x'\|_2^2}{2\gamma}\right)$ where δ^2

controls the variance and γ is the length-scale, which determines the fall-off in correlation with distance $\|x - x'\|_2$. For the mapping Ψ , we use a set of Gaussian kernels: $\phi_j(x) = K(x, c_j) = \exp\left(-\beta_j \|x - c_j\|_2^2\right)$ with local shape constants β_j , $j \in \{1, \dots, M\}$. Consequently, the number of hyper-parameters θ_K coincides with the dimension of \mathcal{M} , i.e., $k = M$. Then, our model hyper-parameters become $\theta_c = \{\delta, \gamma\}$ and $\theta_K = \{\beta_1, \dots, \beta_M\}$. To apply the MCMC sampling, we need to define prior distributions over unknown hyper-parameters. The variance δ^2 is fixed to one, for simplicity, which avoid the identifiability problem of the proposed SGPc model. The length-scale γ and the shape constants β_j , often take nonnegative values and can be sampled in the log-space, are assigned to inverse-gamma and gamma priors, respectively.

Baselines. We focus on several comparisons:

- We compare the proposed approximation methods (SLA and SEP) among themselves with hyper-parameters estimated by: gradient-descent (GD-SLA and GD-SEP), Newton-Raphson (NR-SLA and NR-SEP) and MCMC (MCMC-SLA and MCMC-SEP).
- We also compare the proposed SLA and SEP against some state-of-the-art methods before and after dimensionality reduction using a standard technique.

Performance criteria. As accuracy criteria, we consider the mean classification error (MCE) where the classification error for any unobserved data (z^*, y^*) and fixed threshold s is: $\text{CE} = \mathbf{1}_{\left(\{y^*=+1, \bar{\pi}(z^*) \geq s\} \cup \{y^*=-1, \bar{\pi}(z^*) < s\}\right)}$. Here $\bar{\pi}$ is defined in (3.31) and (3.34), whereas the optimal threshold is reached by the ROC curve. We also consider the root mean square error (RMSE) as a precision criteria where the square error is defined by: $\text{SE} = \left(y^* - \bar{\pi}(z^*)\right)^2$ for $y^* \in \{0, 1\}$.

3.5.1 Synthetic datasets

To illustrate the practical use of the proposed SGPc with different optimization techniques, we conduct a set of controlled synthetic studies for avouching our

Table 3.1: Simulated datasets: Mean classification error.

Datasets	Methods					
	GD-SLA	GD-SEP	NR-SLA	NR-SEP	MCMC-SLA	MCMC-SEP
Step function	14.6%	9.8%	14.6%	8.4%	7.2%	0%
Weighted mixture	12%	10%	9%	8%	6%	5%

theoretical results. It is important to mention that the goal of the embedded submanifold technique, here, is to search an adaptable representation for data rather than dimensionality reduction.

Datasets. The first simulation is from the univariate step function defined by: $g(x_i) = \begin{cases} -1 & \text{if } x_i \leq 0 \\ +1 & \text{if } x_i > 0 \end{cases}$. The training set is composed of 200 inputs sampled from $\mathcal{N}(0, 1)$ while 500 inputs are uniformly distributed between -2 and 2 for test. An error will occur when $g(x^*) \neq y^*$ for an unobserved data (x^*, y^*) . The second simulation is determined by a mixture of two four-dimensional Gaussian distributions i.e., $x_i = y_i * \mathcal{N}(0, \mathcal{I}) + (1 - y_i) * \mathcal{N}(1, \mathcal{I})$ where y_i are produced by sampling directly from a Bernoulli law $\mathcal{B}(\lambda)$ of a fixed parameter $\lambda \in]0, 1[$. For this example, 200 samples were generated for training whereas 400 are for test.

Results. The results from Table 3.1 suggest that NR-SEP performs as well as GD-SEP, but there is the same MCE for SLA with the step function. Moreover, MCMC sampling gives improved performance when compared to iterative optimization methods. Accordingly, one can observe that SEP achieves a better accuracy than SLA with a significant margin for both datasets. From various conducted tests, we showed that the quality of the proposed SGPC strongly depends on the hyper-parameters selection method.

3.5.2 Real data

In order to assess the computational complexity for our proposed procedures, a real study was conducted for a range of datasets. Unlike the previous experiments, the embedded submanifold technique has an important and crucial role to reduce the dimensionality of data. We randomly choose 80% of the dataset to form the training set whereas the rest is maintained for test.

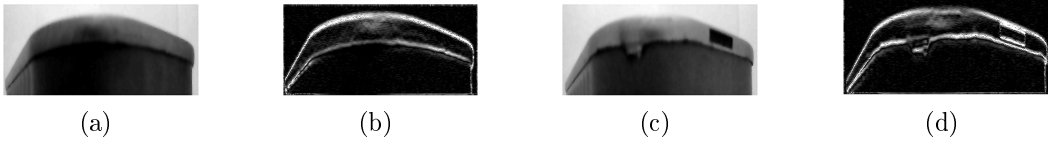


Figure 3.1: An example of two original images: non-defective (a) and defective (c). The associated extracted features: non-defective (b) and defective (d).



Figure 3.2: An Example of two video scene sequences where (top) non-violent and (bottom) violent. For each video: first frame (a,e), frame in the middle (b,f), last frame (c,g), and ViF descriptor (d,h).

Table 3.2: Real data: Mean classification error.

Datasets	Methods					
	GD-SLA	GD-SEP	NR-SLA	NR-SEP	MCMC-SLA	MCMC-SEP
BC	12.08%	11.67%	10%	9.58%	7.08%	7.08%
MD	13.07%	8.91%	12.39%	8.32%	11.31%	7.49%
SV	14.29%	12.24%	14.29%	10.2%	6.12%	4.08%

Table 3.3: Real data: Root mean square error.

Datasets	Methods					
	GD-SLA	GD-SEP	NR-SLA	NR-SEP	MCMC-SLA	MCMC-SEP
BC	0.213	0.202	0.188	0.154	0.175	0.120
MD	0.256	0.211	0.225	0.186	0.200	0.159
SV	0.213	0.197	0.194	0.163	0.175	0.137

3.5.2.1 Datasets of real data

Breast cancer (BC) from 2D images. We first apply the proposed approaches for classifying 1200 images representing tissues (normal or abnormal). We note that breast cancer have specific tissues and we will use them to extract pertinent features in order to test if a patient is infected or not. Motivated by this application, we represent each image with a descriptor of length $d = 6300$, called Pyramid of Histograms of Bag of Words (PHOW), based on the concatenation of three regional histograms determined in subdivisions of the image. More details about this dataset are given in [Gao et al. \(2015\)](#).

Manufacturing defect (MD). The second dataset contains 2042 images of manufacturing defects (defectives or non-defectives) [Fradi et al. \(2018\)](#). Each original image is represented with its extracted feature using the vertical gradient of length $d = 23205$. The main goal is to learn the relationship between defect and directional change in the intensity of each image. We display some examples in [Figure 3.1](#) for original images: non-defective (a) and defective (c) and their extracted features: non-defective (b) and defective (d).

Learning and classifying motion information from scene videos (SV). The third example comes from a dataset containing 246 videos of crowd violence (violent or non-violent scenes), also presented in [Hassner et al. \(2012\)](#). Given a video sequence, we selected a Violence Flows (ViF) descriptor of length $d = 756$ based on estimating the optical flow between consecutive frames. [Figure 3.2](#) illustrates an example of two video scene sequences: non-violent (a,b,c) and violent (e,f,g) with their corresponding ViF descriptor: (d) and (h).

3.5.2.2 Results on real data

The mean classification and root square errors are summarized in [Table 3.2](#) and [Table 3.3](#), respectively. Accordingly, one can observe that SEP achieves the lowest MCE and RMSE values with a significant margin. It is also notable that both SLA

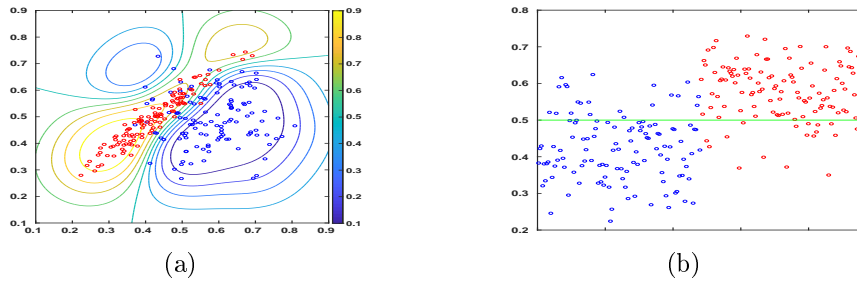


Figure 3.3: Representing test data with contour plot colored as function of the predictive probability by region (a). The approximate predictor of class "+1" with optimal threshold (green line) (b). In all subfigures, normal tissues are red dotted and abnormal tissues are blue dotted.

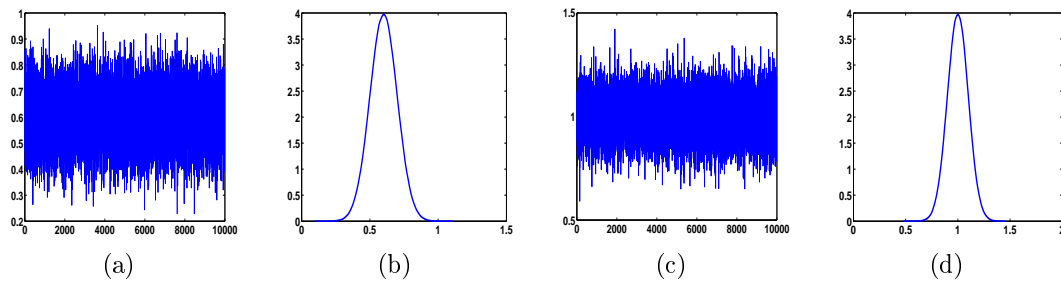


Figure 3.4: Markov chain values of β_1 (a) and β_2 (c). Posterior distributions of β_1 (b) and β_2 (d).

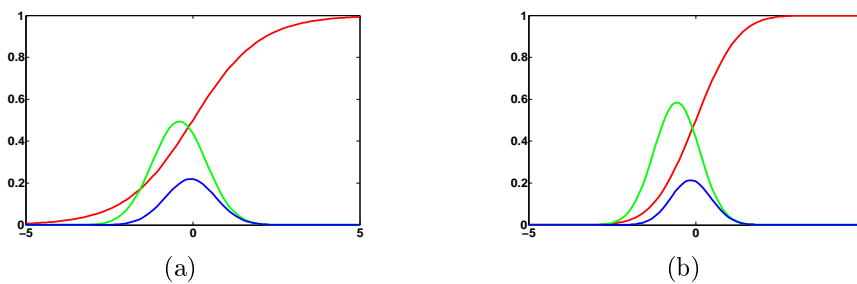


Figure 3.5: Sigmoid function (red), SLA predictive distribution (green), and the product of sigmoid and predictive distribution (blue) (a). Probit function (red), SEP predictive distribution (green), and the product of probit and predictive distribution (blue) (b).

and SEP with MCMC sampling improve those with iterative optimization methods with better results for SEP. Furthermore, both SLA and SEP give a good precision when focusing on the RMSE criteria with a better margin for SEP.

Results on breast cancer (BC). More details about MCMC-SLA are given in Figure 3.3 (a). Accordingly, test data are represented in two-dimensional feature space $\mathcal{M} \subseteq \mathbb{R}^2$ with contour plot colored as function of predictive probabilities of class "+1" in regions. This plot still shows that predictive probabilities revert to one half if we move away from the data. This can be also confirmed in Figure 3.3 (b) where predictive probabilities are near/far to 0.5.

We give an illustration of MCMC-SEP. The evolution of the length-scale γ and the local shape parameters $\theta_K = \{\beta_1, \beta_2\}$ were sampled using 10^4 simulations in each update for the algorithms described in Section 3.4.1.1 and Section 3.4.1.2. Fortunately, the experiments have shown that the problem of big iterations (practically 10^6) usually needed to simulate Markov chains for complex inputs is partially solved by considering the submanifold structure. We show the chain values of θ_K for the last update in Figure 3.4 (a,c). The sampled values are centered near the sample means: 0.6 and 1 for β_1 and β_2 , respectively, but also contain values that are less common. To estimate the posterior distributions of θ_K , we simply take the nonparametric kernel density of sampled values Botev et al. (2010) evaluated at equally-spaced points that cover the range of the data in β_1 and β_2 . Figure 3.4 (b,d) displays these absolutely smooth posteriors.

Results on manufacturing defect (MD). Figure 3.5 illustrates the key steps needed to classify an unobserved input $z^* = \Psi(x^*)$ where the true output is $y^* = -1$. The area between the x-axis and the blue line is the approximate predictor for $y^* = +1$, $\bar{\pi}(z^*)$ with respective values: 0.42 and 0.33 for MCMC-SLA and MCMC-SEP. This means that we reach more precision with SEP since 0.33 is further from 0.5 than 0.42.

Results on scene videos (SV). We consider a particular example of mapping a

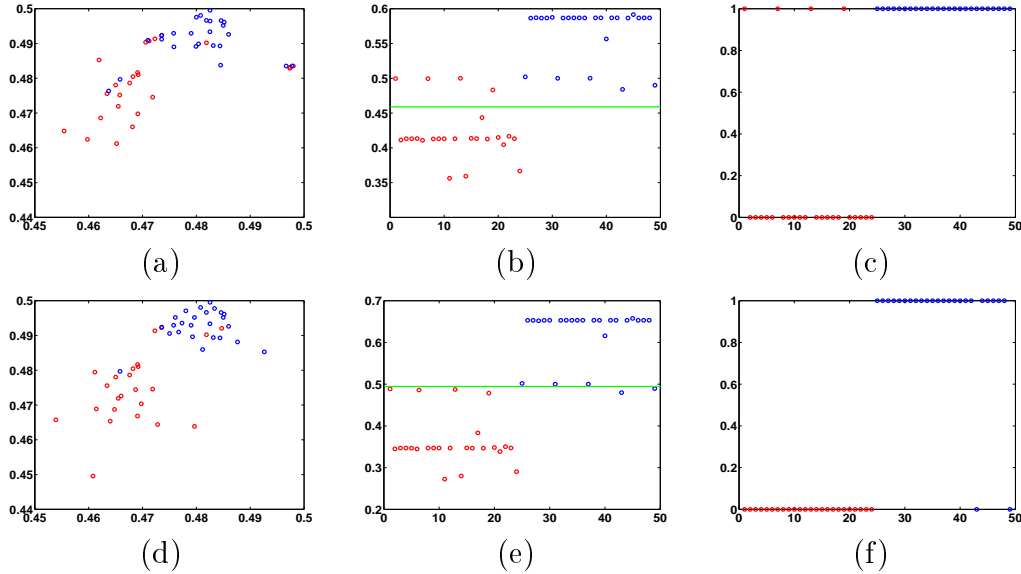


Figure 3.6: Representing and classifying data in two-dimensional feature space with SLA (top) and SEP (bottom). Test data (a,d), the approximate predictor of class "+1" with optimal threshold (green line) (b,e), and binary predicted outputs (c,f). In all subfigures, non-violent videos are red dotted and violent videos are blue dotted.

subset of test data from the input space $\mathcal{X} \subseteq \mathbb{R}^{756}$ to the feature one $\mathcal{M} \subseteq \mathbb{R}^2$. In Figure 3.6 (a,d), we plot the transformed inputs with group labels "-1" as red dots and those with group labels "+1" as blue dots. These results were obtained when finding the type-II MAP estimator of $\theta_K = \{\beta_1, \beta_2\}$ for both MCMC-SLA and MCMC-SEP, respectively. A great separability between the two classes is clearly visible when reducing the dimensionality and the complexity of initial inputs. Hence, the mapping from \mathcal{M} to the output space $\{-1, +1\}$ is smooth and can be easily managed by the SGPc after estimating the length-scale γ . The solid green line in Figure 3.6 (b,e) represents the optimal threshold obtained by the ROC curve for SLA and SEP, respectively. From Figure 3.6 (c,f), it can be seen that SEP has a better predictor than SLA where we have four misclassified videos for SLA and only two for SEP.

3.5.3 Comparative study

We first compare the proposed SLA and SEP with the type II-MAP estimator to some baseline methods: linear SVM, RBF SVM, standard GPc with Laplace approximation, logistic regression and logistic ridge regression. Following the same

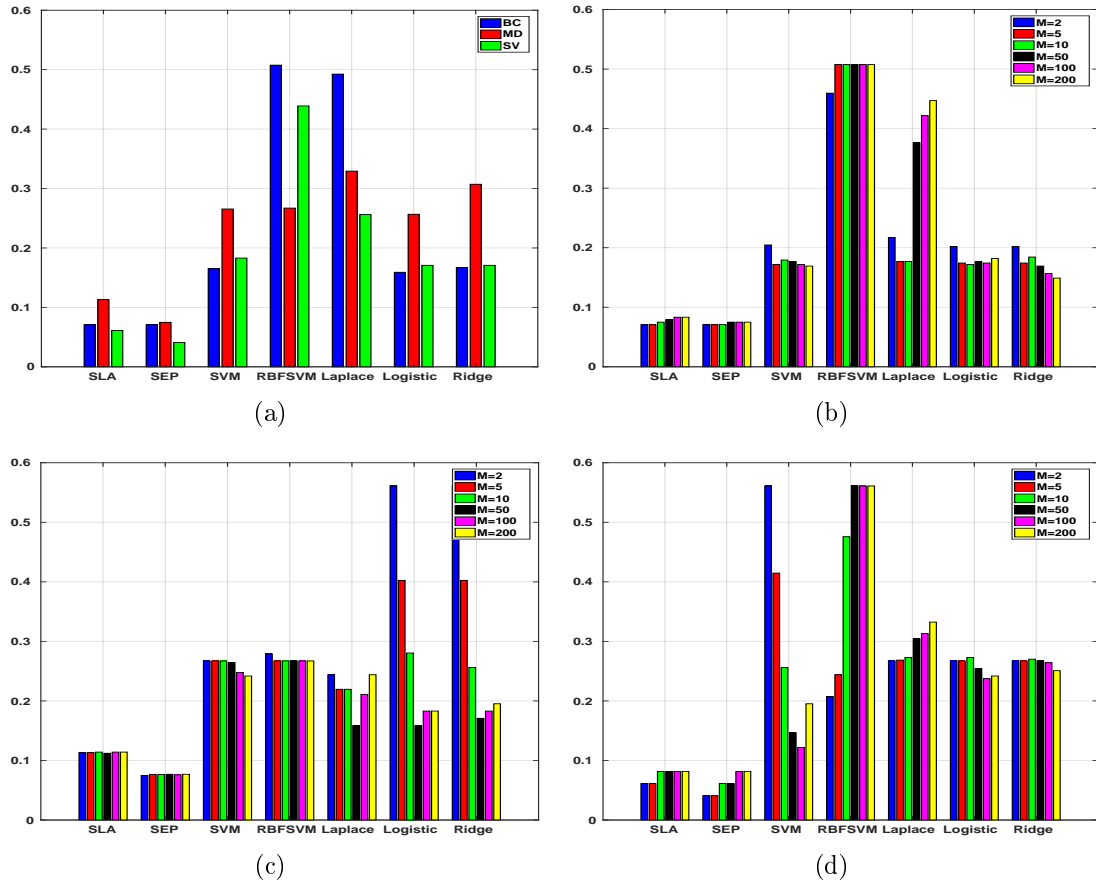


Figure 3.7: Performance evaluation as a mean classification error for different methods (a). Performance evaluation as a mean classification error when applying PCA to the baseline methods and the manifold embedding to the proposed SLA and SEP with several values of M for real data: BC (b), MD (c), and SV (d).

idea, we show results of the baseline methods in Figure 3.7 (a). According to these results, we state that SLA and SEP are more efficient than all baseline methods when minimizing the MCE criteria. It is also important to compare the proposed manifold embedding for dimensionality reduction to standard techniques based on modifying the covariance function. For instance, Snelson and Ghahramani (2006) have learned a linear projection \mathcal{P} of the data points in a supervised manner (e.g., principal component analysis (PCA) extraction) with a covariance function: $c(\mathcal{P}(x), \mathcal{P}(x'))$, yielding the following modified squared exponential covariance: $\tilde{c}_{\delta, \gamma}(x, x') = \delta^2 \exp\left(-\frac{\|\mathcal{P}(x) - \mathcal{P}(x')\|_2^2}{2\gamma}\right)$ for any pair of inputs (x, x') . We also give results after reducing the dimensionality with PCA as function of different projection dimensions ($M = 2, 5, 10, 50, 100, 200$) in Figure 3.7 (b,c,d). We use the same values of M for our proposed manifold embedding technique when dealing with SLA and SEP. Accordingly, we state that both SLA and SEP perform better than the baseline methods with a significant margin for all values of M .

3.6 Conclusion

Although Gaussian processes are very flexible, they are still limited in high-dimensions. In this chapter, we have suggested to perform an embedded submanifold with a mapping (embedding) defined on a Reproducing Kernel Hilbert Space for dimensionality reduction. The hyper-parameters of the kernel and the covariance function were estimated jointly. A set of new identities were also derived in this purpose yielding a reduced time complexity. To summarize, our proposed scalable Gaussian process classifier can be viewed as a valid Gaussian process classifier for classifying complex and high-dimensional data with a more expressive covariance function. This provides new data representations in the feature space (manifold) allowing more advantages than dealing with initial inputs. Finally, our proposed method successfully modeled highly complex data (e.g., images and video sequences) where other baseline methods have failed.

Chapter 4: Bayesian regression and classification using Gaussian processes indexed by probability density functions

In this chapter, we introduce the notion of GPs indexed by probability density functions. We particularly show how a Bayesian inference with GPs can be put into action on functional spaces. We discuss some improvements of covariance function selection and hyper-parameters estimation from Chapter 3. Our framework has the capacity of inferring and classifying both high-dimensional and functional inputs. Extensive experiments on multiple synthetic, semi-synthetic and real data demonstrate the effectiveness and the efficiency of the proposed method.

4.1 Introduction

In functional data analysis [Srivastava and Klassen \(2016\)](#) and medical imaging [Belle et al. \(2015\)](#), it is very common to compare/classify functions. The mathematical formulation leads to a wide range of applications, but it is crucial to characterize a population or to build predictive models. For instance, probability density functions (PDFs) are inherently infinite-dimensional objects so that it is not straightforward to extend traditional machine learning methods from finite vectors to functional instances [Pistone and Sempi \(1995\)](#). In particular, multiple frameworks exist for comparing PDFs in different representations by their covariance matrices including Frobenius, Fisher-Rao, log-Euclidean, Jensen-Shannon and Wasserstein metrics [Bachoc et al. \(2018\)](#); [Nguyen and Vreeken \(2015\)](#); [Srivastava et al. \(2007\)](#).

Many categories of observations can be represented by PDFs and then studied as elements of a Riemannian manifold equipped with the Fisher-Rao metric [Rao \(1945\)](#). This setting is important for many reasons: First, PDFs make the problem

formulation simpler by identifying data originally lying on an vector space, that are hard to interpret, by their categories and their corresponding probabilities. Second, PDFs improve the visualization of local distributions of data. Finally, when dealing with high-dimensional datasets (set of repetitive features), we can visualize them using PDFs which would be very helpful to explore the skewness of data. In particular, the consistency of regression and classification with PDFs as inputs was established in [Oliva et al. \(2014\)](#); [Póczos et al. \(2013\)](#); [Sutherland et al. \(2016\)](#) with the help of the nonparametric kernel density estimation.

Throughout this chapter, our main aim is to learn GPs indexed by PDFs. For instance, one can think of a GP as defining PDFs and inference is taking place directly in the function-space. Moreover, the index space becomes that of PDFs when choosing the underlying metric in order to evaluate the dissimilarity between PDFs. The only drawback is usually that performing Kriging [Vignès et al. \(2017\)](#) on the PDFs space \mathcal{P} is not straightforward due to its geometry [ichi Amari \(1983\)](#); [Jost \(2011\)](#). For this end, we will exploit an isometry from \mathcal{P} to the tangent space of the Hilbert upper-hemisphere [Srivastava et al. \(2007\)](#). This allows inference to be made in a sub-linear space.

In order to capture all variations from PDFs and to perform optimal predictions, we thus define a zero mean GP on \mathcal{P} : $Z \sim \mathcal{GP}(0, c(.,.))$ of a covariance function $c(.,.)$. Let $(p_1, y_1), (p_2, y_2), \dots, (p_N, y_N)$ be a finite set of observations in which $p_i \in \mathcal{P}$ are PDFs inputs and $y_i \in \mathbb{R}$ are the associated responses, $i \in \{1, \dots, N\}$. We define an estimate of the conditional predictive expectation by $\mathbb{E}[y^* | y_1, \dots, y_N, p_1, \dots, p_N, p^*]$ for an unobserved PDF p^* . For the classification model, we assume that $y_i \in \{-1, +1\}$ and we are interested in the probability of one of two given classes $\mathbb{P}(y^* = \pm 1 | y_1, \dots, y_N, p_1, \dots, p_N, p^*)$.

For estimating the covariance function hyper-parameters, we focus on several methods maximizing the marginal likelihood. Our goal is then to select those optimizing different performance criteria for both regression and classification:

1. The first method is the quasi-Newton with BFGS updates, based on the gradient vector and an approximation of the Hessian matrix, in order to find

a local maximum of the marginal likelihood. This choice is very crucial and is an improvement from Chapter 3. One of the chief advantages of quasi-Newton method over Newton's method is that the Hessian matrix does not need to be computed but only approximated. In addition, Newton's method and its derivatives, such as interior point methods, require the Hessian to be inverted which is typically implemented by solving a system of linear equations and is often quite costly.

2. The second method is a special case of MCMC methods, called Hamiltonian Monte-Carlo (HMC) [Duane et al. \(1987\)](#). The objective is to perform sampling from a probability distribution for which the marginal likelihood and its gradient are known. The HMC has the advantage of simulating from a physical system governed by Hamiltonian dynamics, which performs the number of iterations usually needed for MCMC sampling.

4.2 Riemannian representation

Let p be a PDF of a real-valued random variable X defined on $I = [0, 1]$. The set of all PDFs forms

$$\mathcal{P} = \left\{ p : I \rightarrow \mathbb{R} \mid p \text{ is nonnegative and } \int_I p(t) dt = 1 \right\} \quad (4.1)$$

The tangent space of \mathcal{P} at p is

$$\mathcal{T}_p(\mathcal{P}) = \left\{ f : I \rightarrow \mathbb{R} \mid \int_I f(t) dt = 0 \right\} \quad (4.2)$$

Note that \mathcal{P} is viewed as a Riemannian manifold equipped with the Fisher-Rao metric defined as follows: for any $p \in \mathcal{P}$ and tangent vectors $f_1, f_2 \in \mathcal{T}_p(\mathcal{P})$, the inner-product is given by

$$\langle f_1, f_2 \rangle_p = \int_I \frac{f_1(t) f_2(t)}{p(t)} dt \quad (4.3)$$

As a second choice of Riemannian representations is the space of square-root density functions, satisfying

$$\mathcal{H} = \left\{ \psi : I \rightarrow \mathbb{R} \mid \psi \text{ is nonnegative, and } \|\psi\|_{\mathbb{L}^2} = \left(\int_I \psi(t)^2 dt \right)^{\frac{1}{2}} = 1 \right\} \quad (4.4)$$

The tangent space of \mathcal{H} at ψ is

$$\mathcal{T}_\psi(\mathcal{H}) = \left\{ g : I \rightarrow \mathbb{R} \mid \int_I \psi(t)g(t)dt = 0 \right\} \quad (4.5)$$

For any two tangent vectors $g_1, g_2 \in \mathcal{T}_\psi(\mathcal{H})$, we state that the Fisher-Rao metric is simply reduced to the \mathbb{L}^2 metric defined by

$$\langle g_1, g_2 \rangle_{\mathbb{L}^2} = \int_I g_1(t)g_2(t)dt \quad (4.6)$$

Moreover, associated with each $p \in \mathcal{P}$ is a unique $\psi \in \mathcal{H}$ (isometrically) expressed as

$$\psi(t) = \sqrt{p(t)}, \quad t \in I \quad (4.7)$$

Note that \mathcal{H} results to be the Hilbert upper-hemisphere (nonnegative-only) with the \mathbb{L}^2 metric. The advantage of the representation $\psi \in \mathcal{H}$ is that it greatly simplifies the Fisher-Rao metric placed on \mathcal{P} with some nice statistical tools on the Hilbert sphere. We list some analytical expressions that are useful for statistical analysis:

Geodesic path. Given $\psi \in \mathcal{H}$ and a vector $g \in \mathcal{T}_\psi(\mathcal{H})$, the geodesic path with initial condition ψ and velocity g at any time instant t can be parameterized in terms of a direction in $\mathcal{T}_\psi(\mathcal{H})$ as

$$\psi(t) = \cos(t\|g\|_{\mathbb{L}^2})\psi + \sin(t\|g\|_{\mathbb{L}^2})\frac{g}{\|g\|_{\mathbb{L}^2}} \quad (4.8)$$

Geodesic distance. The arc length of the geodesic path in \mathcal{H} between two functions ψ_1 and ψ_2 , called geodesic distance, is given by

$$d_{\mathcal{H}}(\psi_1, \psi_2) = \arccos\left(\langle \psi_1, \psi_2 \rangle_{\mathbb{L}^2}\right) \quad (4.9)$$

Exponential map. Let ψ be any element of \mathcal{H} and $g \in \mathcal{T}_\psi(\mathcal{H})$. We define the exponential map as the geodesic path at $t = 1$, which is an isometry from $\mathcal{T}_\psi(\mathcal{H})$ to \mathcal{H} , satisfying

$$\exp_\psi(g) = \cos(\|g\|_{\mathbb{L}^2})\psi + \sin(\|g\|_{\mathbb{L}^2})\frac{g}{\|g\|_{\mathbb{L}^2}} \quad (4.10)$$

The exponential map is a bijection between the tangent space and the unit sphere if we restrict g so that $\|g\|_{\mathbb{L}^2} \in [0, \pi[$.

Log map. For $\psi_1, \psi_2 \in \mathcal{H}$, we define $g \in \mathcal{T}_{\psi_1}(\mathcal{H})$ to be the inverse exponential (log) map of ψ_2 if $\exp_{\psi_1}(g) = \psi_2$. We then use the notation

$$g = \log_{\psi_1}(\psi_2) \quad (4.11)$$

where $g = \frac{\beta}{\|\beta\|_{\mathbb{L}^2}} d_{\mathcal{H}}(\psi_1, \psi_2)$ and $\beta = \psi_2 - \langle \psi_2, \psi_1 \rangle_{\mathbb{L}^2} \psi_1$.

Fréchet mean. The Fréchet mean of $\psi_1, \dots, \psi_N \in \mathcal{H}$ is the function $\psi_{\mathcal{H}}$ belonging to \mathcal{H} and minimizing the Fréchet variance [Karcher \(1977\)](#), i.e.,

$$\psi_{\mathcal{H}} = \operatorname{argmin}_{\psi \in \mathcal{H}} \sum_{i=1}^N d_{\mathcal{H}}^2(\psi, \psi_i) \quad (4.12)$$

For the search of $\psi_{\mathcal{H}}$, we consider iterative algorithms on \mathcal{H} . For simplicity, we consider the gradient-descent on the sphere according to:

- $\rho \leftarrow \exp_{\rho}(\epsilon\tau)$ with a step size $\epsilon > 0$.
- $\tau \leftarrow \frac{1}{N} \sum_{i=1}^N \log_{\rho}(\psi_i)$ for direction update.

In addition, the curvature of the unit sphere is equal to one, the injectivity and the convexity radius are π and $\frac{\pi}{2}$, respectively. This means that the Fréchet mean is unique particularly in the Hilbert upper-hemisphere \mathcal{H} . More details were given in [Krakowski and Manton \(2007\)](#).

4.3 Gaussian Processes on \mathcal{P}

In this section, we focus on constructing GPs on \mathcal{P} . A GP Z on \mathcal{P} is a random field indexed by \mathcal{P} so that $(Z(p_1), \dots, Z(p_N))^T$ is a multivariate Gaussian vector for $p_1, \dots, p_N \in \mathcal{P}$. A zero mean GP is completely specified by its covariance function $c : \mathcal{P} \times \mathcal{P} \rightarrow \mathbb{R}$ of a real process Z defined as

$$c(p_i, p_j) = \mathbb{E}[Z(p_i)Z(p_j)] \quad (4.13)$$

The covariance function $c(.,.)$ on \mathcal{P} must satisfy the following conditions: For any $N \geq 1$ and $p_1, \dots, p_N \in \mathcal{P}$, the matrix $\mathbf{C} = c(\mathbf{p}, \mathbf{p})$ is symmetric nonnegative definite for $\mathbf{p} = (p_1, \dots, p_N)^T$. Furthermore, $c(.,.)$ is called non-degenerate when the above matrix is invertible whenever p_1, \dots, p_N are two-by-two distinct.

For \mathcal{P} and \mathcal{H} detailed in (4.1) and (4.4), we state that there is an isometry between \mathcal{P} and \mathcal{H} ; $p \mapsto \psi \equiv \sqrt{p}$ from (4.7) and a second one between \mathcal{H} and $\mathcal{T}_{\psi_1}(\mathcal{H})$ detailed

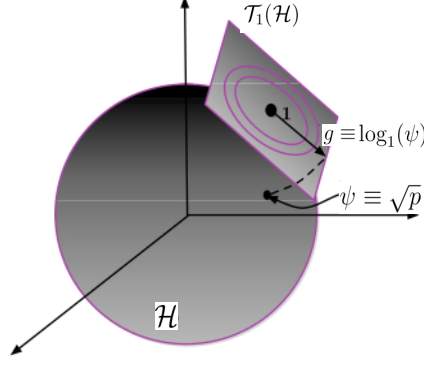


Figure 4.1: An illustration for representing a PDF p as an element g of the tangent space $\mathcal{T}_1(\mathcal{H})$.

in (4.5); $\psi \mapsto g \equiv \log_{\psi_1}(\psi)$ for any $\psi_1 \in \mathcal{H}$ from (4.11). Consequently, we get an isometry between \mathcal{P} and $\mathcal{T}_{\psi_1}(\mathcal{H})$; $p \mapsto \log_{\psi_1}(\sqrt{p})$ by composition. As a special case, we note $\mathcal{E} = \mathcal{T}_1(\mathcal{H})$ the tangent space of \mathcal{H} at the unity pole ($\psi_1 \equiv 1$), as illustrated in Figure 4.1. The strategy that we adopt to construct covariance functions is to exploit the isometric map \log_1 based on the linear tangent space \mathcal{E} . That is, we construct covariance functions with (i, j) component as

$$c(p_i, p_j) = K(\|\log_1(\sqrt{p_i}) - \log_1(\sqrt{p_j})\|_2) \quad (4.14)$$

Proposition 4.1

Let $K : \mathbb{R}^+ \rightarrow \mathbb{R}$ be a covariance function on \mathcal{E} satisfying the stationarity condition: $K(u_i, u_j) = K(\|u_i - u_j\|_2)$ and $c(.,.)$ be defined as in (4.14). Then, $c(.,.)$ is a covariance function. Furthermore, if $\left\{K(\|u_i - u_j\|_2)\right\}_{i,j=1}^N$ is invertible then $c(.,.)$ is non-degenerate. ♠

More details are given in Bachoc et al. (2018); Fradi et al. (2020). From (4.14), the covariance function with (i, j) component is expressed as $c(p_i, p_j) = K(\|g_i - g_j\|_2)$ for $g_i \equiv \log_1(\sqrt{p_i})$. In addition, the covariance function $K(\cdot)$ usually relies on a set of hyper-parameters denoted $\theta = \{\theta^j\}_{j=1}^p$. For this reason, we use the notation $K_\theta(\cdot)$ to emphasize the dependence on θ .

4.4 Regression and classification on \mathcal{P}

In this section, we give details of both regression and classification on \mathcal{P} .

4.4.1 Regression on \mathcal{P}

Having set out the conditions on the covariance function, we can define the regression model on \mathcal{P} by

$$y_i = Z(p_i) + \eta_i, \quad i = 1, \dots, N \quad (4.15)$$

where Z is a zero mean GP indexed by \mathcal{P} of a covariance function $c(\cdot, \cdot)$ and $\eta_i \stackrel{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$. Here σ^2 is the observation noise variance supposed to be known for simplicity. Moreover, if we note $\mathbf{y} = (y_1, \dots, y_N)^T$, the likelihood term is then $\mathbb{P}(\mathbf{y}|Z(\mathbf{p})) = \mathcal{N}(Z(\mathbf{p}), \sigma^2\mathcal{I})$, whereas the prior on $Z(\mathbf{p})$ is $\mathbb{P}(Z(\mathbf{p})) = \mathcal{N}(0, \mathbf{C})$ with $\mathbf{C} = c(\mathbf{p}, \mathbf{p}) = \left\{ K_\theta(\|g_i - g_j\|_2) \right\}_{i,j=1}^N$ from the definition of Z .

For an unobserved PDF p^* and by deriving the conditional distribution, we arrive at the key predictive equation at $Z^* = Z(p^*)$ so that

$$\mathbb{P}(Z^*|\mathbf{p}, \mathbf{y}, p^*) = \mathcal{N}(Z^*|\mu(p^*), \sigma^2(p^*)) \quad (4.16)$$

with

$$\begin{cases} \mu(p^*) = \mathbf{C}_*^T (\mathbf{C} + \sigma^2\mathcal{I})^{-1} \mathbf{y} \\ \sigma^2(p^*) = \mathbf{C}_{**} - \mathbf{C}_*^T (\mathbf{C} + \sigma^2\mathcal{I})^{-1} \mathbf{C}_* \end{cases} \quad (4.17)$$

where $\mathbf{C}_* = c(\mathbf{p}, p^*) = \left\{ K_\theta(\|g_i - g^*\|_2) \right\}_{i=1}^N$ and $\mathbf{C}_{**} = c(p^*, p^*) = K_\theta(\|g^* - g^*\|_2)$ for $g^* \equiv \log_1(\sqrt{p^*})$.

We use the product of likelihood and prior terms to perform the integration over $Z(\mathbf{p})$ yielding the following log-marginal likelihood

$$l(\theta) = \log \mathbb{P}(\mathbf{y}|\mathbf{p}, \theta) = -\mathbf{y}^T (\mathbf{C} + \sigma^2\mathcal{I})^{-1} \mathbf{y} - \log |\mathbf{C} + \sigma^2\mathcal{I}| - \frac{N}{2} \log 2\pi \quad (4.18)$$

The partial derivative of the log-marginal likelihood with respect to θ^j is then

$$\frac{\partial l(\theta)}{\partial \theta^j} = \frac{1}{2} \mathbf{y}^T \mathbf{C}^{-1} \frac{\partial \mathbf{C}}{\partial \theta^j} \mathbf{C}^{-1} \mathbf{y} - \text{tr} \left[\mathbf{C}^{-1} \frac{\partial \mathbf{C}}{\partial \theta^j} \right] \quad (4.19)$$

4.4.2 Classification on \mathcal{P}

For classification, we focus on the case of binary outputs, i.e., $y_i \in \{-1, +1\}$ and the GP Z is now referred to as a GPc indexed by \mathcal{P} . We simply update the Laplace approximation detailed in Section 3.2.2 to GPc indexed by PDFs p_1, \dots, p_N instead of vectors x_1, \dots, x_N . Let $\hat{Z}(\mathbf{p}) = (\hat{Z}(p_1), \dots, \hat{Z}(p_N))$ be the

MAP estimator resulting from the Laplace approximation and \mathbf{W} the negative Hessian matrix of the likelihood term, i.e., \mathbf{W} is a $N \times N$ diagonal matrix with entries $\mathbf{W}_{ii} = \frac{\exp(-\hat{Z}(p_i))}{(1+\exp(-\hat{Z}(p_i)))^2}$. The approximate log-marginal likelihood is

$$l(\theta) = -\frac{1}{2}\hat{\mathbf{Z}}(\mathbf{p})^T \mathbf{C}^{-1} \hat{\mathbf{Z}}(\mathbf{p}) + \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{Z}}(\mathbf{p})) - \frac{1}{2} \log |\mathcal{I} + \mathbf{W}^{\frac{1}{2}} \mathbf{C} \mathbf{W}^{\frac{1}{2}}| \quad (4.20)$$

Following the same idea, the partial derivatives of the log-marginal likelihood with respect to θ^j satisfy

$$\frac{\partial l(\theta)}{\partial \theta^j} = \left. \frac{\partial l(\theta)}{\partial \theta^j} \right|_{\hat{\mathbf{Z}}(\mathbf{p})} + \sum_{i=1}^N \frac{\partial l(\theta)}{\partial \hat{Z}(p_i)} \frac{\partial \hat{Z}(p_i)}{\partial \theta^j} \quad (4.21)$$

The first term, obtained when we assume that $\hat{\mathbf{Z}}(\mathbf{p})$ (as well as \mathbf{W}) does not depend on θ , satisfies

$$\left. \frac{\partial l(\theta)}{\partial \theta^j} \right|_{\hat{\mathbf{Z}}(\mathbf{p})} = \frac{1}{2} \hat{\mathbf{Z}}(\mathbf{p})^T \mathbf{C}^{-1} \frac{\partial \mathbf{C}}{\partial \theta^j} \mathbf{C}^{-1} \hat{\mathbf{Z}}(\mathbf{p}) - \frac{1}{2} \text{tr}[(\mathbf{C} + \mathbf{W}^{-1})^{-1} \frac{\partial \mathbf{C}}{\partial \theta^j}] \quad (4.22)$$

The second term, obtained when we suppose that only $\hat{\mathbf{Z}}(\mathbf{p})$ (as well as \mathbf{W}) depends on θ , is fully determined by

$$\frac{\partial l(\theta)}{\partial \hat{Z}(p_i)} = -\frac{1}{2} [(\mathbf{C}^{-1} + \mathbf{W})^{-1}]_{ii} \frac{\partial^3 \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{Z}}(\mathbf{p}))}{\partial^3 \hat{Z}(p_i)} \quad (4.23)$$

and

$$\frac{\partial \hat{\mathbf{Z}}(\mathbf{p})}{\partial \theta^j} = (\mathcal{I} + \mathbf{C} \mathbf{W})^{-1} \frac{\partial \mathbf{C}}{\partial \theta^j} \nabla \log \mathbb{P}(\mathbf{y}|\hat{\mathbf{Z}}(\mathbf{p})) \quad (4.24)$$

4.5 Optimizing hyper-parameters

The resulting log-marginal likelihoods $l(\theta)$ given in (4.18) and (4.20) for both regression and classification depend on the covariance function hyper-parameters controlling the stationarity of GPs on \mathcal{P} . We can optimize all hyper-parameters based on prior expert knowledge or directly from data, which depends on the data type to be collected.

For maximizing the log-marginal likelihoods with respect to θ , we first make use of an iterative optimization method: quasi-Newton, detailed in to Algorithm 3. This task is equivalent to minimizing the cost function taking the negative log-marginal likelihoods $-l(\theta)$.

In a Bayesian context, weak prior distributions are commonly used for $\theta = \{\theta^j\}_{j=1}^p$.

Such weak prior has the form

$$\mathbb{P}(\theta) = \prod_{j=1}^p \mathbb{P}(\theta^j) \quad (4.25)$$

where we assume that all θ^j s are independent. From Bayes' rule, the log-marginal posterior of θ satisfies

$$\begin{aligned} l_{\text{post}}(\theta) &= \log \mathbb{P}(\theta | \mathbf{y}, \mathbf{p}) \\ &\propto \log \mathbb{P}(\mathbf{y} | \mathbf{p}, \theta) + \log \mathbb{P}(\theta) \\ &= l(\theta) + \sum_{j=1}^p \log \mathbb{P}(\theta^j) \end{aligned} \quad (4.26)$$

When sampling from continuous variables, HMC can prove to be a more powerful tool than the usual MCMC sampling. It avoids random walk behavior by simulating from a physical system governed by Hamiltonian dynamics. In HMC, particles are characterized by a position vector or state $\theta = \{\theta^j\}_{j=1}^p$ and a velocity vector $s = \{s^j\}_{j=1}^p$. The Hamiltonian is the sum of potential energy and kinetic energy, defined as follows

$$H(\theta, s) = H^1(\theta) + H^2(s) = -l_{\text{post}}(\theta) + \frac{1}{2} \sum_{j=1}^p s^j{}^2 \quad (4.27)$$

which means that $s \sim \mathcal{N}(0, \mathcal{I})$. Instead of sampling from $\exp(l_{\text{post}}(\theta))$ directly, HMC operates by sampling from the distribution $\exp(-H(\theta, s))$. State a position θ and a velocity s are modified such that $H(\theta, s)$ remains constant throughout the simulation process. The differential equations are given by

$$\frac{d\theta^j}{dt} = \frac{\partial H}{\partial s^j} = s^j \quad \text{and} \quad \frac{ds^j}{dt} = -\frac{\partial H}{\partial \theta^j} = -\frac{\partial H^1}{\partial \theta^j}, \quad j = 1, \dots, p \quad (4.28)$$

To maintain invariance of the Markov chain, however, care must be taken to preserve the volume conservation and time reversibility. The leap-frog algorithm, summarized in Algorithm 9, maintains these properties Neal (2010).

Algorithm 9: Leap-frog.

- 1: for $t = 1, 2, \dots$ do
 - 2: Find the step size ϵ (e.g., by backtracking line search)
 - 3: $s_{t+\frac{\epsilon}{2}}^j := s_t^j - \frac{\epsilon}{2} \frac{\partial}{\partial \theta^j} H^1(\theta_t)$
 - 4: $\theta_{t+\epsilon}^j := \theta_t^j + \epsilon s_{t+\frac{\epsilon}{2}}^j$
 - 5: $s_{t+\epsilon}^j := s_{t+\frac{\epsilon}{2}}^j - \frac{\epsilon}{2} \frac{\partial}{\partial \theta^j} H^1(\theta_{t+\epsilon})$
 - 6: end for
-

We thus perform a half-step update of the velocity at time $t + \frac{\epsilon}{2}$, which is then used to compute $\theta_{t+\epsilon}^j$ and $s_{t+\epsilon}^j$. HMC also needs an acceptance test to accept/reject stage after T leap-frog steps. We summarize the HMC sampling in Algorithm 10.

Algorithm 10: HMC sampling.

Require: log-marginal posterior l_{post} and its gradient vector

$$\nabla l_{\text{post}}(\theta) = \nabla l(\theta) + \nabla \log \mathbb{P}(\theta)$$

Ensure: $\hat{\theta}$

- 1: Initialize θ_0
 - 2: Sample a velocity $s_0 \sim \mathcal{N}(0, \mathcal{I})$
 - 3: Perform T Leap-frog steps to obtain the new state θ_T and velocity s_T from Algorithm 9
 - 4: Acceptance probability

$$\alpha := \min \left\{ 1, \frac{\exp(-H(\theta_T, s_T))}{\exp(-H(\theta_0, s_0))} \right\}$$
 - 5: Simulate $u \sim \mathcal{U}([0, 1])$
 - 6: if $u < \alpha$ then
 - 7: Accept the proposal $\hat{\theta} := \theta_T$
 - 8: else
 - 9: Reject the proposal $\hat{\theta} := \theta_0$
 - 10: end if
-

4.6 Applications

In this section, we test and illustrate the proposed methods using synthetic, semi-synthetic and real data. For all experiments, we study the empirical results of GPs indexed by PDFs for both regression and classification.

Covariance function. In practice, we can select the covariance function $K_\theta(\cdot)$ from the Matérn family: $K_\theta(x) = \frac{\delta^2}{\Gamma(\nu)2^{\nu-1}} \left(\frac{2\sqrt{\nu x}}{\gamma}\right)^\nu K_\nu\left(\frac{2\sqrt{\nu x}}{\gamma}\right)$ where δ^2 is the variance parameter, γ is the length-scale parameter and ν is the smoothness parameter. Here, K_ν is the modified Bessel function of the second kind and Γ refers to the gamma function. From Proposition 4.1, the Matérn covariance function with (i, j) component defined by $c(p_i, p_j) = K_\theta(\|g_i - g_j\|_2)$ is, indeed, non-degenerate. The Matérn form has the desirable property that GPs have realizations (sample paths) that are $(\nu - 1)$ times differentiable, which prove its smoothness as function of ν . As $\nu \rightarrow \infty$, the Matérn covariance function approaches the squared exponential form, whose realizations are infinitely differentiable [Minasny and Mcbratney \(2005\)](#). So, the Matérn covariance function is more general than the squared exponential adopted in Chapter 3. To apply the HMC sampling, we need to define prior distributions over unknown hyper-parameters. Following [Gelman \(2006\)](#), δ^2 will be assigned to a half-Cauchy (nonnegative-only) prior distribution and γ is assumed to be an inverse-gamma distribution, whereas ν is simply estimated by cross-validation [Neal \(1997\)](#).

Baselines. We compare results of GPs indexed by PDFs (GPP) where the hyper-parameters are estimated by quasi-Newton (QN-GPP) and HMC (HMC-GPP) to:

- Functional linear model (FLM) for regression [Ramsay and Dalzell \(1991\)](#).
- Nonparametric kernel Wasserstein (NKW) for regression [Sriperumbudur et al. \(2010\)](#).
- A GP based on the Wasserstein distance (W-GP) for classification [Mallasto and Feragen \(2017\)](#).

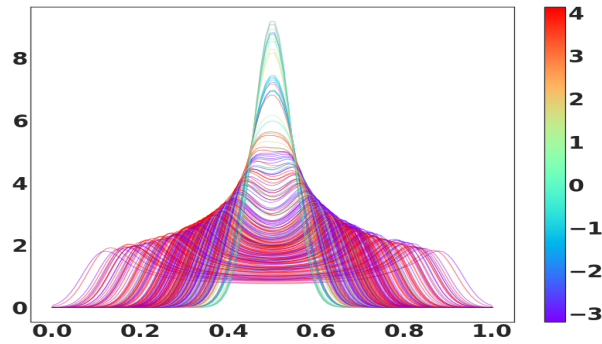


Figure 4.2: PDFs of TBF inputs for regression. The output with continuous value in $[-3, 4]$ is illustrated by a colorbar.

- A GP based on the Jensen-Shannon (JS-GP) divergence for classification [Nguyen and Vreeken \(2015\)](#).

Performance criteria. For regression, we illustrate the performance of proposed framework in terms of root mean square error (RMSE) where the square error at an unobserved data (p^*, y^*) is defined by: $SE = (y^* - \mu(p^*))^2$ and the negative log-marginal likelihood (NLML). For classification, we consider three criteria: accuracy, area under curve (AUC) and NLML.

4.6.1 Regression

We first consider a synthetic dataset for regression.

Dataset. We observe a finite set of functions simulated from (4.15) with

$$Z(p_i) = 0.5 * \langle \sqrt{p_i}, \sqrt{\tilde{p}} \rangle_2 + 0.5$$

For this example, we consider a truncated Fourier basis (TFB) with random Gaussian coefficients to form the original functions satisfying $v_i(t) = \delta_{i,1}\sqrt{2}\sin(2\pi t) + \delta_{i,2}\sqrt{2}\cos(2\pi t)$ where $\delta_{i,1}, \delta_{i,2} \sim \mathcal{N}(0, 1)$. We also take $\tilde{v}(t) = -0.5\sqrt{2}\sin(2\pi t) + 0.5\sqrt{2}\cos(2\pi t)$ as a reference function. We suppose that \tilde{p} and p_i s are referred to the corresponding PDFs of \tilde{v} and v_i s estimated using the nonparametric kernel method (bandwidths were selected using the method given in [Botev et al. \(2010\)](#)). An example of $N = 100$ estimates is displayed in Figure 4.2 with colors depending on their output levels. Note that PDFs defined on I can also be treated as those

on any interval in \mathbb{R} via wrapping I for analysis.

Results. The experimental results of the TFB regression dataset, when focusing on the RMSE values, are shown in Table 4.1. According to these results, we remark that the proposed QN-GPP gives better precision than FLM. On the other hand, HMC-GPP substantially outperforms NKW with a significant margin. As illustrated in Table 4.2, we state that the proposed methods are more efficient than the baseline FLM when maximizing the log-marginal likelihood. Again, this is an explanation on how the quality of GPP strongly depends on hyper-parameters estimation method. In addition, QN-GPP stated in Algorithm 3 is very effective from a computational point of view.

Table 4.1: Regression: Root mean square error.

QN-GPP	HMC-GPP	FLM	NKW
0.07	0.13	0.10	0.28

Table 4.2: Regression: Negative log-marginal likelihood.

QN-GPP	HMC-GPP	FLM
73.28	21.89	329.66

4.6.2 Classification

Now, we perform some extensive experiments to evaluate the proposed methods using a second category of datasets for classification.

4.6.2.1 Datasets for classification

Synthetic datasets. We consider two datasets of synthetic PDFs: beta and inverse-gamma distributions. This choice is very crucial for many reasons since beta is defined on $I = [0, 1]$ by default, parametrized by two positive parameters, and has been widely used to represent a large family of PDFs with finite support in various fields. Increasingly, the inverse-gamma plays an important role to characterize random fluctuations affecting wireless channels [Atapattu et al. \(2011\)](#). We refer to these datasets as Beta and InvGamma, respectively. We performed this exper-

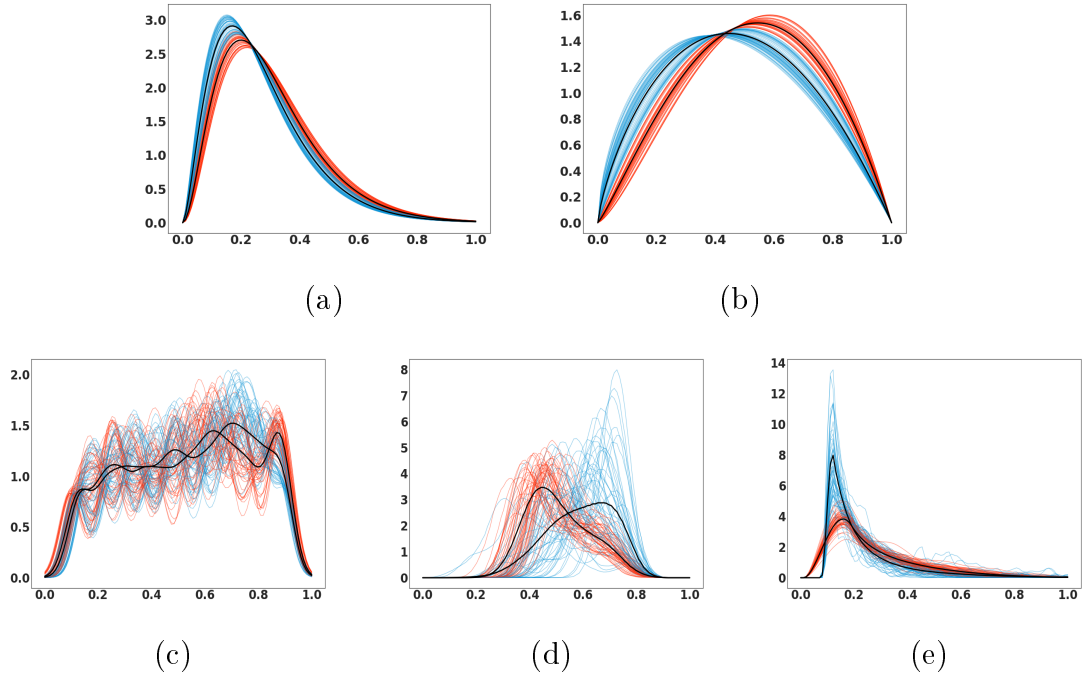


Figure 4.3: Synthetic PDFs for InvGamma (a) and Beta (b) with class 1 (red) and class 2 (blue). Semi-synthetic PDFs for Growth (c) with girls (red) and boys (blue). Real PDFs for Temp (d) with uninfected (red) and infected (blue). Real PDFs for Plants (e) with disease (red) and healthy (blue). In all subfigures, the Fréchet mean for each class is in black.

iment by simulating $N = 200$ PDFs uniformly divided and slightly different for both classes. Each observation is represented as a density when we add a random white noise to initial parameters of Beta and InvGamma. We show some examples of nonparametric PDFs with different random parameters in Figure 4.3 (a,b). We also illustrate the Fréchet mean for each class in black from (4.12) when dealing with the Hilbert upper-hemisphere \mathcal{H} .

Semi – synthetic dataset. Semi-synthetic data represent clinical growth charts for children from 2 to 12 years. We refer to this dataset as Growth. We simulate the charts from centers for disease control and prevention [Kuczmarski et al. \(2002\)](#) through the available quantile values. The main goal is to classify observations by gender. Each observation represents the size growth of a child as function of to his age (120 months). We represent observations as nonparametric PDFs with some examples displayed in Figure 4.3 (c). For each class, we plot girls in red and boys in blue as well as we show the Fréchet mean in black.

Real datasets.

A first public dataset with 1500 images represent maize leaves [DeChant et al. \(2017\)](#). We note that maize leaves have specific textures used to extract pertinent features in order to test if a plant has disease or not. We refer to this dataset as Plants. Motivated by this application, we first represent each image, with its wavelet-deconvolved version, by a vector of length 262144. Figure 4.4 illustrates

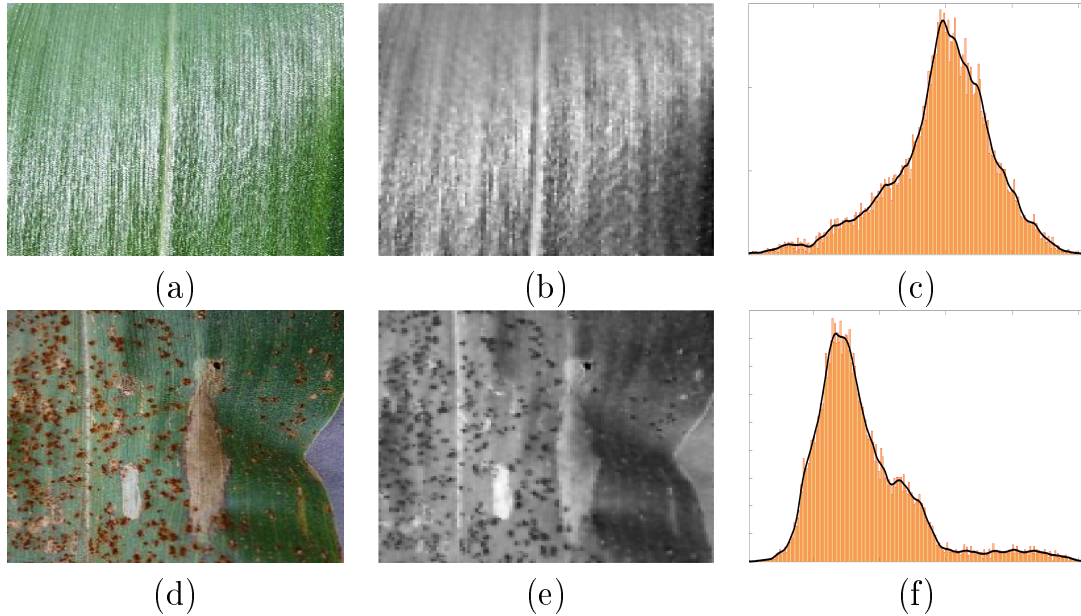


Figure 4.4: An example of two classes from maize plants dataset where healthy leaf (top) and leaf with disease (bottom). For each class: an original image (a,d), the extracted features (b,e), and the normalized histogram (c,f).

an example of two original images: healthy plant (a) and a plant with disease (d), their wavelet-deconvolved versions (b,e) and the corresponding normalized histograms (c,f). We also display PDFs from histograms in black.

We remind that high-dimensional inputs (here, 262144) make traditional machine learning techniques fail to solve the problem at hand. However, the spectral histograms as marginal distributions of the wavelet-deconvolved versions can be used to represent/classify original images. In fact, instead of comparing the histograms, a better way to compare two images (here, a set of repetitive features) would be to compare their corresponding PDFs.

A second real dataset with 1717 observations gives the body temperature of dogs [Kumar and Kumar \(2018\)](#), for which temporal measures of infected and uninfected

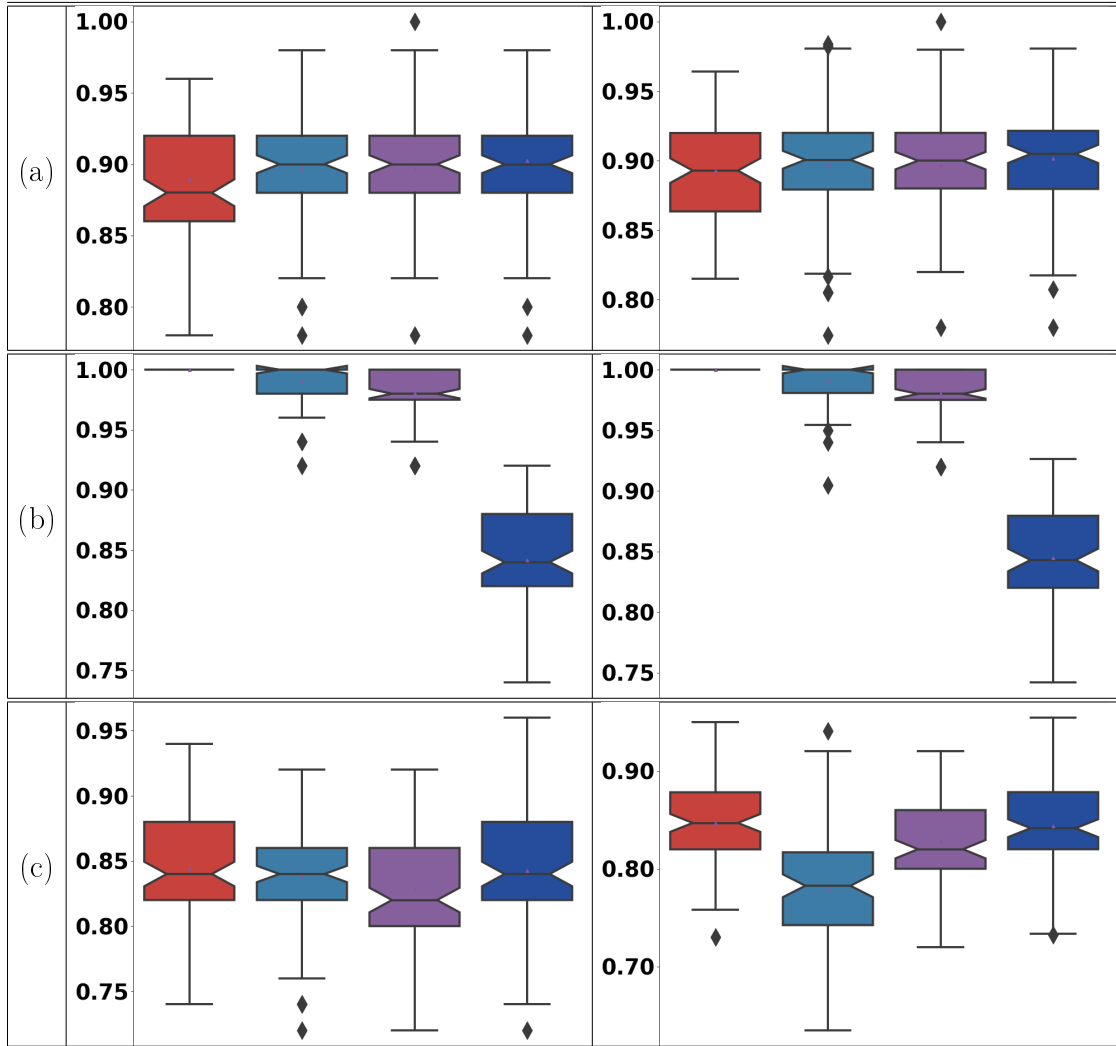


Figure 4.5: Boxplots of the classification accuracy (left) and AUC (right) on synthetic datasets: InvGamma (a) and Beta (b), and semi-synthetic dataset: Growth (c). In all subfigures, the performance is given for different methods: QN-GPP (red), HMC-GPP (light blue), W-GP (violet), and JS-GP (dark blue).

dogs are stored during 24 hours. The infection by a parasite is suspected to cause persistent fever. The main goal is to learn the relationship between the infection and a dominant pattern from temporal temperatures. We refer to this dataset as Temp.

For these two examples, the kernel choice for nonparametric PDFs is very crucial. Since non-Gaussian PDFs do not give good test performances when dealing with Gaussian kernels, we consider the Epanechnikov kernel which has the lowest RMSE for a compact support. The PDF estimates were obtained using an automatic bandwidth selection method described in Botev et al. (2010). We illustrate some examples of PDFs from real datasets in Figure 4.3 (d,e).

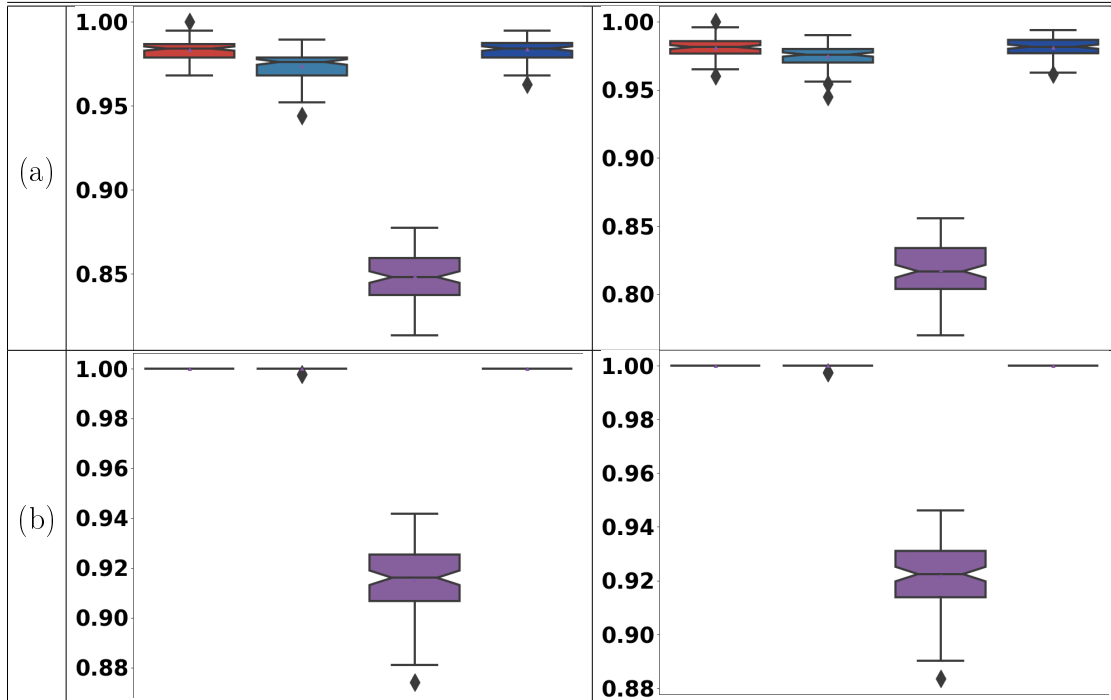


Figure 4.6: Boxplots of the classification accuracy (left) and AUC (right) on real data: Temp (a) and Plants (b). In all subfigures, the performance is given for different methods: QN-GPP (red), HMC-GPP (light blue), W-GP (violet), and JS-GP (dark blue).

4.6.2.2 Classification results

We learn the model from 80% of the dataset whereas the rest is kept for test. This subdivision has been performed randomly 100 times. The performance is given as a mean and the corresponding standard deviation (std) in order to reduce the bias (class imbalance and sample representativeness) introduced by the random train/test split.

Results on synthetic datasets. We summarize all evaluation results on synthetic datasets in Figure 4.5 (a,b). Accordingly, one can observe that both HMC-GPP, W-GP and JS-GP reach the best accuracy values for InvGamma with a little margin for the proposed HMC-GPP. On the other hand, QN-GPP and HMC-GPP heavily outperform W-GP and JS-GP for Beta. Again, this simply shows how the hyper-parameters estimation method impacts the quality of the predictive distributions.

Results on semi – synthetic datasets. We summarize all results on the Growth

Table 4.3: Classification: Negative log-marginal likelihood.

Methods	Datasets									
	Synthetic				Semi-synthetic		Real data			
	InvGamma		Beta		Growth		Temp		Plants	
	mean	std	mean	std	mean	std	mean	std	mean	std
QN-GPP	30.50	2.43	4.41	0.06	68.03	3.43	98.66	0.73	98.65	0.72
HMC-GPP	105.35	0.22	105.28	0.21	61.65	2.24	105.36	0.22	9.33	0.21
JS-GP	32.2	2.38	42.87	2.73	62.0	3.02	116.65	4.13	10.26	0.12

semi-synthetic dataset in Figure 4.5 (c) where we show accuracy and AUC values as boxplots from 100 tests. One can observe that QN-GPP gives the best accuracy with a significant margin. Note that we have used $T = 10^4$ HMC iterations in Algorithm 10. Furthermore, we set the "Burn-in" and "Thinning" parameters in order to ensure a fast convergence of the Markov chains and to reduce the correlation between samples.

Results on real data. We further investigate whether our proposed methods can be applied to real data. Figure 4.6 (a,b) shows the boxplots of accuracy and AUC values for Temp and Plants, respectively. In short, we highlight that the proposed methods successfully modeled real data with improved results in comparison to the baseline W-GP.

Fortunately, the experiments have shown that the problem of big number of iterations, usually needed to simulate Markov chains for complex inputs, e.g., images, is partially solved by considering the proposed HMC sampling detailed in Algorithm 10. In closing, we can state that the leap-frog algorithm, based on Hamiltonian dynamics, lets to early search the best directions giving the best local minimum of the Hamiltonian defined in (4.27).

We also confirm all previous results from Table 4.3, which summarizes the mean and the std of NLML values for classification datasets. It clearly shows that at least one of the proposed methods (QN-GPP or HMC-GPP) better minimizes the NLML than the baseline method JS-GP. This brings more quite accurate estimates, which prove the predictive power of our proposed approaches.

4.7 Conclusion

In this chapter, we have extended the classical Bayesian models by introducing the notion of Gaussian processes indexed by probability density functions. We detailed and applied two different numerical methods to learn both regression and classification models. Furthermore, we showed new theoretical results for the covariance function defined on the space of density functions thanks to the Riemannian geometry and the Fisher-Rao information. Extensive experiments on multiple and varied datasets have demonstrated the effectiveness of proposed methods against current state-of-the-art methods.

Chapter 5: Bayesian registration and clustering of univariate functions and multidimensional curves

In this chapter, we develop a nonparametric registration framework for functions and multidimensional curves. We also propose a Bayesian clustering model based on Gaussian processes priors, which enable us to define distributions (subpopulations) over the sub-sets of observations that naturally arise in this problem. In our framework, nonparametric inference includes the generation of posterior samples on coefficients resulting from the Karhunen-Loève expansion. It usually requires integrating over infinitely many parameters but we efficiently solve such issue with Hamiltonian dynamics.

5.1 Introduction

For shape analysis of functions and curves, one usually need to introduce the notion of reparametrizations. A reparametrization is a differentiable diffeomorphism, defined from $I = [0, 1]$ into itself and preserving the boundary constraints. In practice, one of major problems is that of registration since when collecting data many phenomena can explain the fact that there is a time difference. For functional data and curve registration, data pre-processing is required before focusing on amplitude variation. In the literature, there are many existing algorithms for registration. In this context, we can cite [Kneip and Ramsay \(2008\)](#); [Liu and Muller \(2004\)](#); [Ramsay and Li \(1998\)](#). In order to find the optimal registration between two curves, several variations have been proposed. Among them, we can cite the dynamic programming [Bernal et al. \(2016\)](#); [Cai and Judd \(2010\)](#) and the quasi-Newton [Huang et al. \(2016\)](#). Recently, shape registration of functions and curves becomes very interesting in many fields especially in medical applications [Grogan](#)

and Dahyot (2017); Ying et al. (2016).

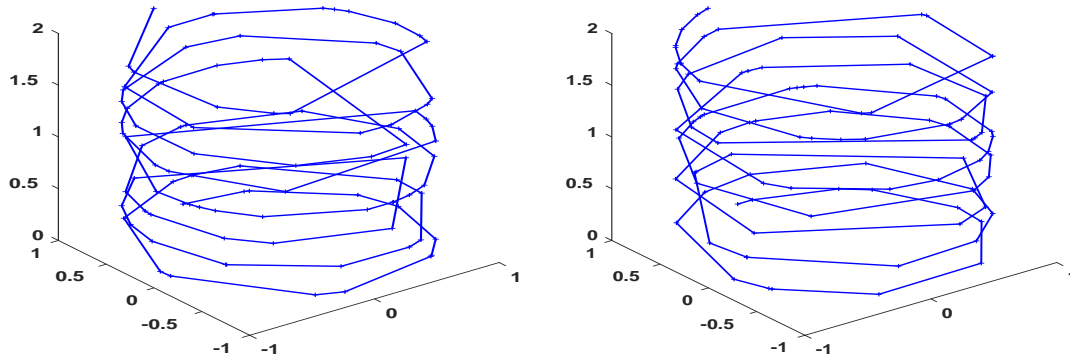


Figure 5.1: An example of two different reparametrizations of the same class of curves with: first reparametrization (left) and second reparametrization (right).

We give an example of two different reparametrizations of the same class of curves in Figure 5.1. Accordingly, we can observe that the L^2 distance between the two shapes (left and right) is not zero although they belong to the same class of curves. Consequently, the reparametrization invariance is a serious task in shape analysis of curves. Compared to the state-of-the-art methods mentioned above, the SRVF representation detailed in Section 2.5 is very efficient for analyzing shapes of curves in Euclidean spaces Srivastava et al. (2011). The SRVF representation has several advantages: The well-known elastic metric of shapes simplifies to the L^2 metric, the reparameterization function acts by isometries, and the space of unit length curves results to be the familiar unit sphere.

In terms of statistical modeling and inference, the set of all reparametrizations forms a group of diffeomorphisms and working with objects in the group is more challenging due to its complicated geometry. Our work differs from previous methods since we study a reparametrization as a cumulative distribution function (CDF). In this chapter, our aim consists in reformulating the registration problem of curves represented by their shapes as elements on a Riemannian manifold. Besides, we are interested in the clustering process of a finite set of observed curves. By setting K sub-populations and given N curves, estimating the optimal local distribution (identified with CDF) for k -th cluster, $k = 1, \dots, K$, is necessary before assigning each curve to its cluster Fradi and Samir (2020).

Our motivation is to establish the link between the space of CDFs and the

Hilbert sphere due to its nice properties and explicit geometrical tools. In fact, one of the advantages is that the Riemannian metric is the Fisher-Rao, the only metric invariant to reparametrizations [Brigant et al. \(2015\)](#). To handle such task of clustering, the inference on CDFs F^1, \dots, F^K becomes more affordable on the coefficients resulting from the Karhunen-Loève (K-L) expansion [Ghanem and Spanos \(1991\)](#) A^1, \dots, A^K . This can be performed with the Hamiltonian dynamic on a finite-dimensional sphere using the spherical HMC sampling [Lan et al. \(2014\)](#).

5.2 Riemannian representation

We recall some tools about the geometry of Riemannian representations defined on a compact interval $I = [0, 1]$ with their corresponding Fisher-Rao metrics.

Let F be a CDF of a real-valued random variable X . The space of CDFs, defined on I , is a Riemannian representation satisfying

$$\mathcal{F} = \left\{ F : I \rightarrow I \mid \dot{F} \text{ is nonnegative, } F(0) = 0, \text{ and } F(1) = 1 \right\} \quad (5.1)$$

\mathcal{F} forms a group with the group operation given by composition, i.e., for $F_1, F_2 \in \mathcal{F}$, the group operation is given by $F_2(F_1(\xi))$. The identity element of \mathcal{F} is the function $F(\xi) = \xi$. The tangent space of \mathcal{F} at F is

$$\mathcal{T}_F(\mathcal{F}) = \left\{ f : I \rightarrow \mathbb{R} \mid \int_I \dot{f}(\xi) d\xi = 0 \right\} \quad (5.2)$$

where the Fisher-Rao metric is stated as follow: for any two tangent vectors $f_1, f_2 \in \mathcal{T}_F(\mathcal{F})$, the inner product is given by

$$\langle f_1, f_2 \rangle_F = \int_I \frac{\dot{f}_1(\xi) \dot{f}_2(\xi)}{\dot{F}(\xi)} d\xi \quad (5.3)$$

A second choice of Riemannian representations is the space of square-root density functions, satisfying

$$\mathcal{H} = \left\{ \psi : I \rightarrow \mathbb{R} \mid \psi \text{ is nonnegative, and } \|\psi\|_{\mathbb{L}^2} = \left(\int_I \psi(t)^2 dt \right)^{\frac{1}{2}} = 1 \right\} \quad (5.4)$$

\mathcal{H} results to be the Hilbert upper-hemisphere (nonnegative part) with the \mathbb{L}^2 metric. Besides, associated with each $\psi \in \mathcal{H}$ is a unique CDF $F \in \mathcal{F}$ (isometrically) satisfying

$$F(\xi) = \int_0^\xi \psi(t)^2 dt, \quad \xi \in I \quad (5.5)$$

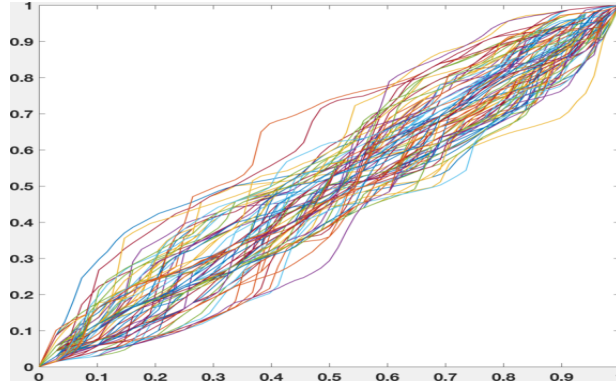


Figure 5.2: Some examples of CDFs

Note that the reparametrization functions in shape analysis of curves can be directly written as elements of \mathcal{F} . Moreover, \mathcal{H} is somewhat easier than \mathcal{F} to analyze in view of its group structure with some nice statistical tools already mentioned in Section 4.2. We illustrate some examples of CDFs in Figure 5.2.

5.3 Spherical Gaussian process for curve registration

In this section, we shall describe some methods for computing distances and carrying out inference in quotient spaces for curves. We also propose to reformulate the usual problem of curve registration.

5.3.1 Spherical Gaussian process decomposition

In order to simplify the estimation of CDFs, we propose to use the K-L expansion of ψ as a linear sum of basis functions in $\mathbb{L}^2(I)$ with random coefficients. Under this respect, we model ψ as a random function, itself drawn from a second order GP Williams and Rasmussen (1996) with a continuous, square-integrable, symmetric, and nonnegative definite covariance function $c(.,.)$ over $I \times I$, i.e.,

$$\psi(t) \sim \mathcal{GP}(0, c(t, t')), \text{ where } \psi \in \mathcal{H} \quad (5.6)$$

Let $(\phi_l)_l$ denote a system of orthonormal eigen-functions in $\mathbb{L}^2(I)$ and $(\lambda_l)_l$ the associated nonnegative eigen-values of $c(.,.)$. We define the Hilbert-Schmidt integral operator as a mapping from $\mathbb{L}^2(I)$ into itself, expressed by $L : \phi_l \mapsto L\phi_l$ and

satisfying

$$(L\phi_l)(t') = \int_I c(t, t')\phi_l(t)dt \quad (5.7)$$

By Mercer's theorem, the covariance function can be expressed as

$$c(t, t') = \sum_{l=1}^{\infty} \lambda_l \phi_l(t)\phi_l(t') \quad (5.8)$$

and the Fredholm integral equation is

$$(L\phi_l)(t') = \lambda_l \phi_l(t') \quad (5.9)$$

To maintain the constraint that $\psi \in \mathcal{H}$, we only focus on the restriction that $\|\psi\|_{\mathbb{L}^2} = 1$ since ψ is nonnegative does not impose any additional constraint.

Therefore, the K-L expansion of ψ is

$$\psi(t) = \sum_{l=1}^{\infty} a_l \phi_l(t), \quad \text{with } a_l \stackrel{\text{ind}}{\sim} \mathcal{N}(0, \lambda_l) \quad (5.10)$$

We take into account a truncated version at order m of the K-L expansion given by

$$\psi_m(t) = \sum_{l=1}^m a_l \phi_l(t) \quad (5.11)$$

with the approximation error

$$e_m(t) = \sum_{l=m+1}^{\infty} a_l \phi_l(t) \quad (5.12)$$

This choice results from the fact that among all versions expressed in (5.11), the truncated K-L expansion is optimal in the sense of minimizing the mean integrated squared error (MISE) given by $\int_I \mathbb{E}[(e_m(t))^2]dt$. From (5.5), we get

$$\begin{aligned} F_m(\xi) &= \int_0^\xi \psi_m(t)^2 dt, \quad \forall \xi \in I \\ &= \sum_{l=1}^m a_l^2 \int_0^\xi \phi_l(t)^2 dt + 2 \sum_{l=1}^m \sum_{r=l+1}^m a_l a_r \int_0^\xi \phi_l(t)\phi_r(t) dt, \quad \forall \xi \in I \end{aligned} \quad (5.13)$$

Theorem 5.1

The truncated version F_m is a CDF if and only if $A = (a_1, \dots, a_m) \in \mathcal{S}^{m-1}$

where

$$\mathcal{S}^{m-1} = \left\{ A = (a_1, \dots, a_m) \in \mathbb{R}^m \mid \|A\|_2 = \left(\sum_{l=1}^m a_l^2 \right)^{1/2} = 1 \right\}$$



Proof For the proof of this result, we are able to check:

- $\xi \mapsto F_m(\xi)$ is a C^1 mapping on I , since $t \mapsto \psi_m(t)$ is a continuous one for all

$t \in I$.

- $F_m(0) = \int_0^0 \psi_m(t)^2 dt = 0$, by definition.
- $F_m(1) = \int_0^1 \psi_m(t)^2 dt = 1$, if and only if $A \in \mathcal{S}^{m-1}$.
- $\dot{F}_m(\xi) = \psi_m(\xi)^2$ is nonnegative for all $\xi \in I$.

Consequently, the resulting version makes it easier to check that F_m is a CDF instead of F . It translates directly to a finite-dimensional spherical constraint on the random coefficients a_l , $l = 1, \dots, m$.

5.3.2 Group actions, shape invariances and distance

In order to develop a formal framework for analyzing shapes of curves, one needs a mathematical representation of curves that is natural, general and efficient. We establish the SRVF representation detailed in Section 2.5 for its advantages in shape analysis of curves. By representing a curve $\beta : I \rightarrow \mathbb{R}^d$ by its SRVF $q \in \mathcal{M}$ satisfying $q(\xi) = \frac{\dot{\beta}(\xi)}{\sqrt{\|\dot{\beta}(\xi)\|_2}}$, we have taken care of the translation and the scaling variability, but the rotation and the reparameterization variability still remain. When $d \geq 2$, a rotation is an element of $SO(d)$ and a reparameterization is an element of \mathcal{F} for any $d \geq 1$. The rotation and reparameterization of a curve β are denoted by the actions of $SO(d)$ and \mathcal{F} on its SRVF. While the action of $SO(d)$ is the usual $O \mapsto Oq$, the action of \mathcal{F} is derived as $F \mapsto (q, F) = \sqrt{F}q \circ F \equiv q^*$.

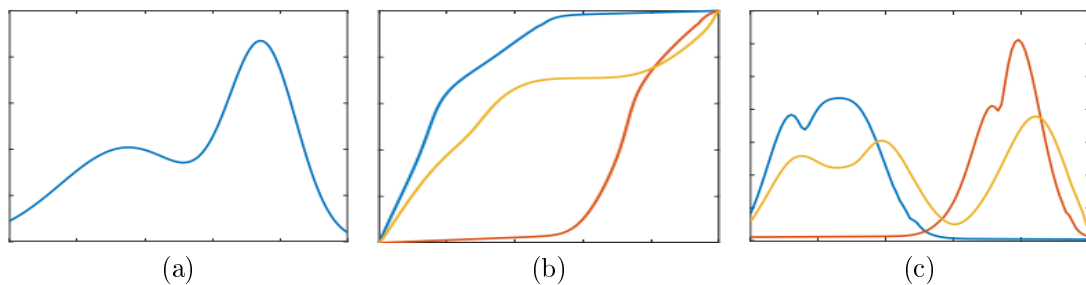


Figure 5.3: A SRVF representation q (a) that is deformed using a number of CDFs F (b) giving the resulting q^* (c).

Figure 5.3 shows an example of different actions of CDFs on a SRVF representation defined on I . In order for shape analysis to be invariant to these transformations, it is important for these groups to act by isometries. We present the following properties of these actions.

Lemma 5.1

The actions of $SO(d)$ and \mathcal{F} commute on both \mathcal{M} and \mathbb{L}^2 . ♡

Proof This follows directly from the definitions of the two group actions. Therefore, we can form a joint action of the product group $\mathcal{G} = SO(d) \times \mathcal{F}$ according to $((O, q), F) = O\sqrt{\dot{F}}q \circ F$.

Lemma 5.2

The action of the product group \mathcal{G} on \mathcal{M} is by isometries. ♡

Proof Since $\langle Oq_1, Oq_2 \rangle = \langle q_1, q_2 \rangle$ for all $O \in SO(d)$, the proof for $SO(d)$ follows. Besides, we have

$$\begin{aligned}
\langle q_1^*, q_2^* \rangle &= \int_I \langle q_1^*(\xi), q_2^*(\xi) \rangle_2 d\xi & (5.14) \\
&= \int_I \langle \sqrt{\dot{F}}(\xi)q_1 \circ F(\xi), \sqrt{\dot{F}}(\xi)q_2 \circ F(\xi) \rangle_2 d\xi \\
&= \int_I \langle q_1(\tilde{\xi}), q_2(\tilde{\xi}) \rangle_2 d\tilde{\xi}; \quad \tilde{\xi} = F(\xi) \\
&= \langle q_1, q_2 \rangle
\end{aligned}$$

which completes the proof for \mathcal{F} .

Therefore, we can define a quotient space of \mathcal{M} modulo \mathcal{G} . The orbit of a function $q \in \mathcal{M}$ is given by

$$[q] = \left\{ O\sqrt{\dot{F}}q \circ F \mid (O, F) \in \mathcal{G} \right\} \quad (5.15)$$

Consequently, we have to define a distance on $\mathcal{Q} = \{[q] \mid q \in \mathcal{M}\}$ by

$$d_{\mathcal{Q}}([q_1], [q_2]) = \inf_{(O, F) \in \mathcal{G}} \|q_1 - ((O, q_2), F)\| \quad (5.16)$$

For the remainder, the rotation invariance will be performed using the SVD [Berge \(1977\)](#) where we only focus on the reparametrization invariance. Thus, the distance defined in (5.16) becomes $d_{\mathcal{Q}}([q_1], [q_2]) = \inf_{F \in \mathcal{F}} \|q_1 - (q_2, F)\| = \inf_{F \in \mathcal{F}} \|q_1 - q_2^*\|$ where $\|q_1 - q_2^*\|^2 = \int_I \|q_1(\xi) - q_2^*(\xi)\|_2^2 d\xi$.

5.3.3 Optimal registration between curves

The optimal registration between two curves q_1 and q_2 is given by the best reparametrization minimizing the deformation between them, i.e.,

$$\hat{F} = \operatorname{arginf}_{F \in \mathcal{F}} \|q_1 - q_2^*\|^2; \quad q_2^* \equiv \sqrt{\dot{F}}q_2 \circ F \quad (5.17)$$

Given a random sample q_1, \dots, q_N , we obtain the sample Fréchet mean \tilde{q} when optimizing over F_i , i.e.,

$$\tilde{q} = \operatorname{argmin}_{q \in \mathcal{Q}} \sum_{i=1}^N \inf_{F_i \in \mathcal{F}} \|q - q_i^*\|^2 \quad (5.18)$$

Since the shape space of curves results to be a linear subspace of $L^2(I, \mathbb{R}^d)$ with the SRVF representation, the Fréchet mean becomes the arithmetic mean satisfying

$$\tilde{q}(\xi) = \frac{1}{N} \sum_{i=1}^N q_i^*(\xi) \quad (5.19)$$

where each F_i is then updated in an iterative algorithm until convergence.

Now, we will identify F by F_m detailed in (5.13) under its corresponding assumption. This simplifies the initial registration problem on F given in (5.17) with an equivalent problem on A , from the uniqueness of the K-L expansion, such that

$$\hat{A} = \operatorname{arginf}_{A \in \mathcal{S}^{m-1}} \|q_1 - q_2^{*,m}\|^2; \quad q_2^{*,m} \equiv \sqrt{\hat{F}_m} q_2 \circ F_m \quad (5.20)$$

Let $l(A) = \|q_1 - q_2^{*,m}\|^2$ denote the cost function defined in (5.20). Algorithm 11 summarizes the Newton-Raphson on the sphere established in order to minimize the cost function. More details about this algorithm were given in [Holbrook et al. \(2017\)](#). We give an example of finding an optimal registration between two curves in Figure 5.4. Likewise, the Fréchet mean in (5.18) becomes

$$\tilde{q}^m = \operatorname{argmin}_{q \in \mathcal{Q}} \sum_{i=1}^N \inf_{A_i \in \mathcal{S}^{m-1}} \|q - q_i^{*,m}\|^2 \quad (5.21)$$

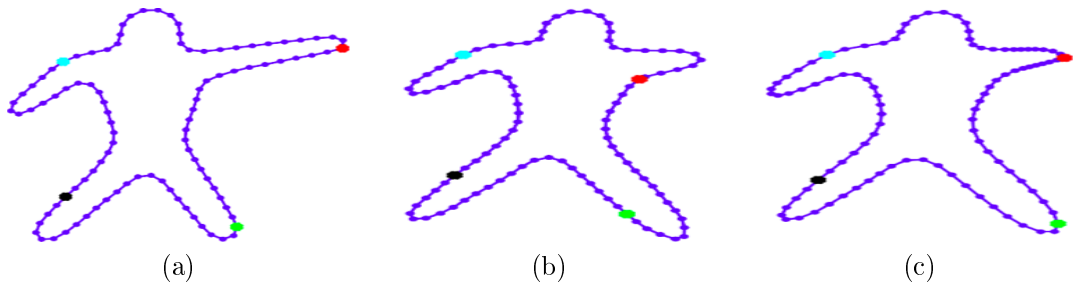


Figure 5.4: A first curve q_1 (a), a second curve q_2^* (b) and the resulting curve $\hat{q}_2^{*,m} \equiv \sqrt{\hat{F}_m} q_2 \circ \hat{F}_m$ depending on \hat{A} with the best matching of features between q_1 and q_2^* (c).

Algorithm 11: Newton-Raphson on the sphere.

- Require: cost function $l(\cdot)$, its gradient vector $\nabla l(\cdot)$ and its Hessian matrix $\nabla^2 l(\cdot)$
- 1: repeat
 - 2: Evaluate Hess $l(A_t) := \nabla^2 l(A_t) - \nabla l(A_t)^T A_t \mathcal{I}_m$
 - 3: Compute $W_t := (\mathcal{I}_m - A_t A_t^T) [\text{Hess } l(A_t)]^{-1} (\mathcal{I}_m - A_t A_t^T)$
 - 4: Compute $v_t := -W_t \nabla l(A_t)$
 - 5: Progress along geodesic given in (4.8) with velocity v_t and position A_t to recover A_{t+1}
 - 6: Set $t := t + 1$
 - 7: until Convergence
-

5.4 Bayesian curve clustering

Now, we are ready to formulate the problem of curve clustering with the Gaussian mixture model (GMM). We assume that we have a finite set of N curves q_1, \dots, q_N to be grouped into K populations with $K < N$. For each q_i^* , we draw a cluster C_i with values in $\{1, \dots, K\}$ under the probability $\mathbb{P}(C_i = k) = \pi_k$ where $\sum_{k=1}^K \pi_k = 1$. Consequently, the density of i -th curve that defines the components of k -th sub-population is given by $\mathbb{P}(q_i^* | C_i = k)$. We consider that a discretization $q_i^*(\xi_h) \in \mathbb{R}^d$, $h = 1, \dots, n$ is observed and we note $\boldsymbol{\xi} = (\xi_1, \dots, \xi_n)$. We can model $q_i^*(\boldsymbol{\xi}) | C_i = k$ with a multivariate Gaussian since q_i^* is a continuous function. For simplicity, we assume that $q_i^*(\boldsymbol{\xi}) | C_i = k \sim \mathcal{N}(\tilde{q}^k(\boldsymbol{\xi}), \sigma^2 \mathcal{I})$ where $\sigma^2 > 0$ is the variance parameter and \mathcal{I} is the $nd \times nd$ identity matrix. This work deals with estimating the optimal CDF for the k -th sub-population. Let F^k denote this unknown function and F_m^k its truncated version. The density of q_i with components of sub-population k is

$$\mathbb{P}(q_i | F^k, \tilde{q}^k(\boldsymbol{\xi}), \sigma^2) \propto \exp\left(-\frac{1}{2\sigma^2} \|q_i^*(\boldsymbol{\xi}) - \tilde{q}^k(\boldsymbol{\xi})\|_2^2\right) \quad (5.22)$$

For reasons mentioned above, we use F_m as detailed in (5.13) instead of F so that the prior on F^k becomes a simple prior on $A^k = (a_1^k, \dots, a_m^k) \in \mathcal{S}^{m-1}$, satisfying

$$\mathbb{P}(A^k) \propto \exp\left(-\sum_{l=1}^m \frac{a_l^{k2}}{2\lambda_l}\right) \times \delta_{A^k \in \mathcal{S}^{m-1}} \quad (5.23)$$

where δ refers to the Kronecker delta function. We have all the ingredients to propose the following result.

Theorem 5.2

Given $D = \{q_i\}_{i=1}^N, \pi_1, \dots, \pi_K, \tilde{q}^{1,m}(\boldsymbol{\xi}), \dots, \tilde{q}^{K,m}(\boldsymbol{\xi})$ and σ^2 , the log-posterior of A^1, \dots, A^K is

$$\begin{aligned} & \log p(A^1, \dots, A^K | D, \pi_1, \dots, \pi_K, \tilde{q}^{1,m}(\boldsymbol{\xi}), \dots, \tilde{q}^{K,m}(\boldsymbol{\xi}), \sigma^2) \quad (5.24) \\ & \propto \sum_{i=1}^N \log \left(\sum_{k=1}^K \pi_k \exp \left(-\frac{1}{2\sigma^2} \|q_i^*(\boldsymbol{\xi}) - \tilde{q}^{k,m}(\boldsymbol{\xi})\|_2^2 \right) \right) - \frac{1}{2} \sum_{k=1}^K \sum_{l=1}^m \frac{a_l^{k^2}}{\lambda_l} \end{aligned}$$

under the constraint that A^1, \dots, A^K belong to \mathcal{S}^{m-1} . ♡

Proof The complete likelihood term is

$$\begin{aligned} & \mathbb{P}(D | A^1, \dots, A^K, \pi_1, \dots, \pi_K, \tilde{q}^{1,m}(\boldsymbol{\xi}), \dots, \tilde{q}^{K,m}(\boldsymbol{\xi}), \sigma^2) \quad (5.25) \\ & = \prod_{i=1}^N \left(\sum_{k=1}^K \pi_k \mathbb{P}(q_i | A^k, \tilde{q}^{k,m}(\boldsymbol{\xi}), \sigma^2) \right) \\ & \propto \prod_{i=1}^N \left(\sum_{k=1}^K \pi_k \exp \left(-\frac{1}{2\sigma^2} \|q_i^*(\boldsymbol{\xi}) - \tilde{q}^{k,m}(\boldsymbol{\xi})\|_2^2 \right) \right) \end{aligned}$$

and its logarithm satisfies

$$\begin{aligned} & \log \mathbb{P}(D | A^1, \dots, A^K, \pi_1, \dots, \pi_K, \tilde{q}^{1,m}(\boldsymbol{\xi}), \dots, \tilde{q}^{K,m}(\boldsymbol{\xi}), \sigma^2) \quad (5.26) \\ & \propto \sum_{i=1}^N \log \left(\sum_{k=1}^K \pi_k \exp \left(-\frac{1}{2\sigma^2} \|q_i^*(\boldsymbol{\xi}) - \tilde{q}^{k,m}(\boldsymbol{\xi})\|_2^2 \right) \right) \end{aligned}$$

We also write the constrained prior as

$$\mathbb{P}(A^1, \dots, A^K) = \prod_{k=1}^K \mathbb{P}(A^k) \propto \exp \left(-\sum_{k=1}^K \sum_{l=1}^m \frac{a_l^{k^2}}{2\lambda_l} \right) \times \delta_{A^1, \dots, A^K \in \mathcal{S}^{m-1}} \quad (5.27)$$

where we assume that A^k s are independent and resulting from the same integral operator defined in (5.7). Besides, its logarithm satisfies

$$\log \mathbb{P}(A^1, \dots, A^K) \propto -\frac{1}{2} \sum_{k=1}^K \sum_{l=1}^m \frac{a_l^{k^2}}{\lambda_l} \quad (5.28)$$

under the constraint that A^1, \dots, A^K belong to \mathcal{S}^{m-1} . The desired result follows by plugging (5.26) and (5.28) into the log-posterior probability term.

We use the spherical HMC on \mathcal{S}^{m-1} for simulating from the posterior of $\mathbf{A} = (A^1, \dots, A^K)$. We add an extra Gibbs sampling to update $\pi_1, \dots, \pi_K, \sigma^2$ and the Fréchet means $\tilde{q}^{1,m}(\boldsymbol{\xi}), \dots, \tilde{q}^{K,m}(\boldsymbol{\xi})$ when minimizing the Fréchet variance of observations in each cluster from (5.21), iteratively, until convergence. The HMC sampling augments the state space with an auxiliary velocity variable $\mathbf{v} = (v^1, \dots, v^K)$ satisfying $v^k A^k = 0, k = 1, \dots, K$. It also simulates from a Hamiltonian dynamic (H) splitted into two terms ($H = H^1 + H^2$) with a potential energy defined as the

negative log-posterior

$$H^1(\mathbf{A}) = -\log \mathbb{P}(\mathbf{A}|D, \pi_1, \dots, \pi_K, \tilde{q}^{1,m}(\boldsymbol{\xi}), \dots, \tilde{q}^{K,m}(\boldsymbol{\xi}), \sigma^2) \quad (5.29)$$

and a kinetic energy, satisfying

$$H^2(\mathbf{v}) = \frac{1}{2} \sum_{k=1}^K v^k T G v^k \quad (5.30)$$

where G refers to the canonical spherical metric. According to [Lan et al. \(2014\)](#), the spherical HMC establishes the link between the unit sphere in \mathbb{R}^m denoted \mathcal{S}^{m-1} and the unit ball in \mathbb{R}^{m-1} denoted \mathcal{B}_0^{m-1} . If $\bar{A}^k = (a_1^k, \dots, a_{m-1}^k)$ takes the $(m-1)$ first components of A^k then $\bar{A}^k \in \mathcal{B}_0^{m-1}$. Therefore, we can rewrite A^k as $A^k = (\bar{A}^k, \sqrt{1 - \|\bar{A}^k\|_2^2})$. The spherical HMC is then detailed in [Algorithm 12](#) in terms of $\mathbf{A} = (A^1, \dots, A^K)$, $\bar{\mathbf{A}} = (\bar{A}^1, \dots, \bar{A}^K)$ and the block diagonal matrix

$$\mathbf{M} = \underbrace{\left(\begin{array}{cccc} \left[\begin{array}{c} \mathcal{I}_{m-1} \\ 0 \end{array} \right] & 0 & 0 & \dots \\ 0 & \left[\begin{array}{c} \mathcal{I}_{m-1} \\ 0 \end{array} \right] & 0 & \dots \\ 0 & \dots & 0 & \left[\begin{array}{c} \mathcal{I}_{m-1} \\ 0 \end{array} \right] \end{array} \right)}_{(m-1)K} \Bigg\} mK$$

Once we have estimated all model's parameters, we can evaluate the conditional probability that the i -th curve belongs to k -th sub-population by

$$\begin{aligned} \mathbb{P}(C_i = k|q_i) &= \frac{\mathbb{P}(C_i = k, q_i)}{\mathbb{P}(q_i)} & (5.31) \\ &= \frac{\mathbb{P}(C_i = k)\mathbb{P}(q_i|C_i = k)}{\mathbb{P}(q_i)} \\ &= \frac{\pi_k \mathbb{P}(q_i|C_i = k)}{\sum_{k=1}^K \pi_k \mathbb{P}(q_i|C_i = k)} \\ &= \frac{\pi_k \exp\left(-\frac{1}{2\sigma^2} \|q_i^*(\boldsymbol{\xi}) - \tilde{q}^{k,m}(\boldsymbol{\xi})\|_2^2\right)}{\sum_{k=1}^K \pi_k \exp\left(-\frac{1}{2\sigma^2} \|q_i^*(\boldsymbol{\xi}) - \tilde{q}^{k,m}(\boldsymbol{\xi})\|_2^2\right)} \end{aligned}$$

Algorithm 12: Spherical HMC sampling.

Require: Negative log-posterior $H^1(\cdot)$ and its gradient $\nabla H^1(\cdot)$

Ensure: $\hat{\mathbf{A}}$

- 1: Initialize \mathbf{A}_0
- 2: Sample a new momentum value $\mathbf{v}_0 \sim \mathcal{N}(0, \mathcal{I})$ where \mathcal{I} is the $mK \times mK$ identity matrix
- 3: $\mathbf{v}_0 := \mathbf{v}_0 - \mathbf{A}_0 \mathbf{A}_0^T \mathbf{v}_0$
- 4: Calculate $H(\mathbf{A}_0, \mathbf{v}_0) := H^1(\bar{\mathbf{A}}_0) + H^2(\mathbf{v}_0)$
- 5: for $t = 1, 2, \dots, T$ do
- 6: $\mathbf{v}_{t-\frac{1}{2}} := \mathbf{v}_{t-1} - \frac{\epsilon}{2} \left(\mathbf{M} - \mathbf{A}_{t-1} \bar{\mathbf{A}}_{t-1}^T \right) \nabla H^1(\bar{\mathbf{A}}_{t-1})$
- 7: $\mathbf{A}_t := \mathbf{A}_{t-1} \cos(\|\mathbf{v}_{t-\frac{1}{2}}\|_2 \epsilon) + \frac{\mathbf{v}_{t-\frac{1}{2}}}{\|\mathbf{v}_{t-\frac{1}{2}}\|_2} \sin(\|\mathbf{v}_{t-\frac{1}{2}}\|_2 \epsilon)$
- 8: $\mathbf{v}_{t-\frac{1}{2}} := -\mathbf{A}_{t-1} \|\mathbf{v}_{t-\frac{1}{2}}\|_2 \sin(\|\mathbf{v}_{t-\frac{1}{2}}\|_2 \epsilon) + \mathbf{v}_{t-\frac{1}{2}} \cos(\|\mathbf{v}_{t-\frac{1}{2}}\|_2 \epsilon)$
- 9: $\mathbf{v}_t := \mathbf{v}_{t-\frac{1}{2}} - \frac{\epsilon}{2} \left(\mathbf{M} - \mathbf{A}_t \bar{\mathbf{A}}_t^T \right) \nabla H^1(\bar{\mathbf{A}}_t)$
- 10: end for
- 11: Calculate $H(\mathbf{A}_T, \mathbf{v}_T) := H^1(\bar{\mathbf{A}}_T) + H^2(\mathbf{v}_T)$
- 12: Acceptance probability

$$\alpha := \min \left\{ 1, \frac{\exp(-H(\mathbf{A}_T, \mathbf{v}_T))}{\exp(-H(\mathbf{A}_0, \mathbf{v}_0))} \right\}$$

- 13: Simulate $u \sim \mathcal{U}([0, 1])$
 - 14: if $u < \alpha$ then
 - 15: Accept the proposal $\hat{\mathbf{A}} := \mathbf{A}_T$
 - 16: else
 - 17: Reject the proposal $\hat{\mathbf{A}} := \mathbf{A}_0$
 - 18: end if
-

5.5 Applications

In this section, we demonstrate the effectiveness of our method to a clustering task of curves in order to assign each observed curve to its sub-population. All results are drawn from $T = 10^4$ iterations when using the spherical HMC sampling and we set the truncation order of K-L expansion to $m = 30$.

Covariance function. For the covariance function, we model ψ by a GP with a Hilbert-Schmidt operator satisfying $L = \delta^2(\gamma - \partial_x^2)^{-\nu}$ of variance parameter δ^2 , length-scale γ and smoothness parameter ν . By solving (5.9), one can check that the corresponding eigen-values and eigen-functions are $\lambda_l = \delta^2(\gamma + l^2\pi^2)^{-\nu}$ and $\phi_l(t) = \sqrt{2}\cos(l\pi t)$. The hyper-parameter setting of the Hilbert-Schmidt operator is fixed to $(\delta^2, \gamma, \nu) = (1, 0.5, 1)$.

Baselines. We compare results of our method with coefficients estimated by spherical HMC sampling in a Bayesian framework against:

- The GPA-kmeans and GPA-kmedoids when applying the GPA method detailed in Algorithm 6. We update the classical kmeans and kmedoids clustering with the geodesic distance computed on the embedded sphere $\mathcal{S}^{(n-1)d-1-\frac{1}{2}d(d-1)}$. This results in a representation which is invariant under the effects of translation, scaling and rotation.
- The TPCA-kmeans and the TPCA-GMM when applying the TPCA method detailed in Algorithm 7. We update the classical kmeans and GMM clustering with the Euclidean distance computed on the tangent space of the

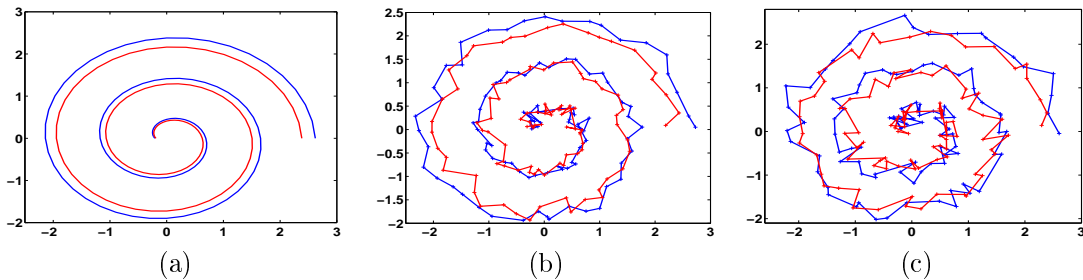


Figure 5.5: The true parameterized curves in \mathbb{R}^2 (a), the observed curves with $\sigma^2 = 0.01$ (b), and $\sigma^2 = 0.1$ (c). In all subfigures, the two clusters are illustrated with different colors: cluster 1 (blue) and cluster 2 (red).

Table 5.1: Parameterized curves in \mathbb{R}^2 : Mean clustering error.

Methods	$\sigma^2 = 0.01$	$\sigma^2 = 0.1$
TPCA-GMM	27%	30%
TPCA-kmeans	26%	28%
GPA-kmeans	18%	23%
GPA-kmedoids	16%	22%
Proposed	9%	17%

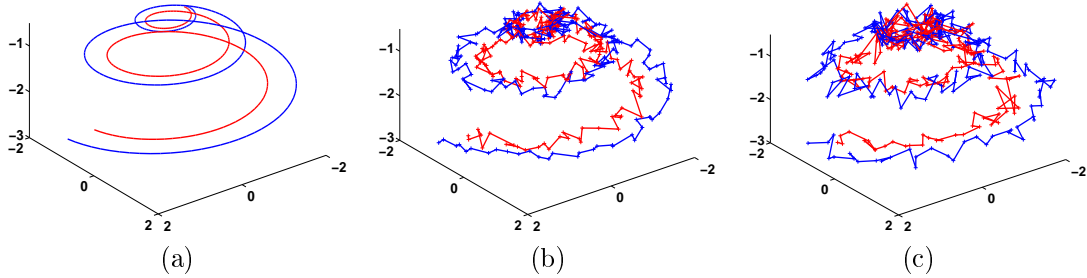


Figure 5.6: The true parameterized curves in \mathbb{R}^3 (a), the observed curves with $\sigma^2 = 0.01$ (b), and $\sigma^2 = 0.1$ (c). In all subfigures, the two clusters are illustrated with different colors: cluster 1 (blue) and cluster 2 (red).

embedded sphere.

Datasets. The proposed methods will be evaluated on two different datasets:

- Two simulations with 2D and 3D parametric curves that have been used for simulating human cochlear implants [Dang et al. \(2015\)](#); [McDonnell et al. \(2010\)](#).
- Two real data of 3D cochlear curves extracted from computed-tomographic (CT) images for human and hominin evolution [Braga et al. \(2019\)](#).

5.5.1 Synthetic datasets

We first validate the performance of the proposed framework in term of accuracy using two simulated datasets. We perform experiments on two examples of parameterized curves in \mathbb{R}^2 and \mathbb{R}^3 . Each curve is most likely in cluster \hat{k} which maximizes the conditional probability given in (5.31), i.e., $\hat{k} = \operatorname{argmax}_k \mathbb{P}(C_i = k | q_i)$ for each i . An error occurs if the observed cluster and the true cluster are different.

Parameterized curves in \mathbb{R}^2 . To simulate data in \mathbb{R}^2 , we first generate two para-

Table 5.2: Parameterized curves in \mathbb{R}^3 : Mean clustering error.

Methods	$\sigma^2 = 0.01$	$\sigma^2 = 0.1$
TPCA-GMM	27%	36%
TPCA-kmeans	29%	40%
GPA-kmeans	16%	25%
GPA-kmedoids	16%	24%
Proposed	12%	19%

metric curves: For all $\xi \in I$,

$$\beta^1(\xi) = \begin{cases} \frac{1}{3}(7(\xi + 0.1)) \cos(\varphi(5\pi\xi)) \\ -\frac{1}{3}(7(\xi + 0.1)) \cos(\varphi(5\pi\xi)) \end{cases} \quad \text{and} \quad \beta^2(\xi) = \begin{cases} \frac{3}{10}(7(\xi + 0.1)) \cos(\varphi(5\pi\xi)) \\ -\frac{3}{10}(7(\xi + 0.1)) \cos(\varphi(5\pi\xi)) \end{cases}$$

The two clusters of curves are displayed (β^1 in blue and β^2 in red) in Figure 5.5 (a). We simulate $N = 100$ curves per cluster using a Gaussian perturbation model where the i -th configuration is obtained as follows

$$\beta_i(\boldsymbol{\xi})|C_i = k \sim \mathcal{N}(\beta^k \circ F(\boldsymbol{\xi}), \sigma^2 \mathcal{I}), \quad i = 1, \dots, 100, \quad k = 1, 2$$

where the discretization of the unit interval I is $n = 50$, i.e., $\boldsymbol{\xi} = (\xi_1, \dots, \xi_{50})$. The true CDF is the identity function, i.e., $F(\xi) = \xi$ for all $\xi \in I$ and \mathcal{I} is the 100×100 identity matrix. The plots in Figure 5.5 (b,c) illustrate the noisy data forming the simulated curves with two variance levels $\sigma^2 = 0.01$ and $\sigma^2 = 0.1$, respectively. For our proposed approach, we need to use the transformed curves $q_i^*(\boldsymbol{\xi})$ in order to search the optimal CDF per cluster. For comparison methods, we apply the GPA and the TPCA approaches to the observed curves $\beta_i(\boldsymbol{\xi})$ directly. From Table 5.1 and focusing on the mean clustering error criteria, our approach performs better than TPCA-GMM, TPCA-kmeans, GPA-kmeans and GPA-kmedoids with a significant margin.

Parameterized curves in \mathbb{R}^3 . In this part, we illustrate our method using three dimensions (3D) curves. We consider two clusters of curves expressed as

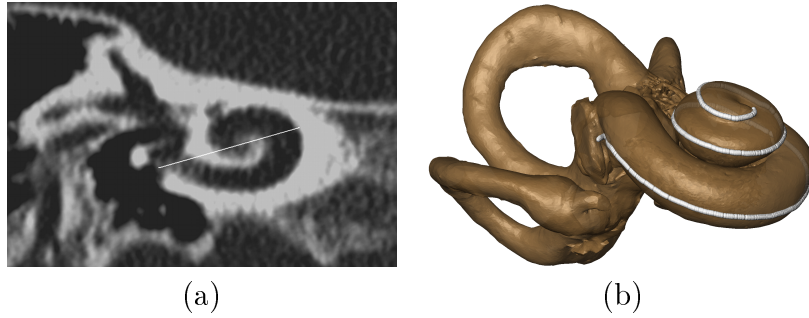


Figure 5.7: A medical CT image (a) and the extracted boundary surface with white cochlear curve (b).

$$\beta^1(\xi) = \begin{cases} \frac{1}{3}(7(\xi + 0.1)) \cos(\varphi(5\pi\xi)) \\ -\frac{1}{3}(7(\xi + 0.1)) \cos(\varphi(5\pi\xi)) \\ -\xi(\xi + 1) \end{cases} \quad \text{and} \quad \beta^2(\xi) = \begin{cases} \frac{1}{4}(7(\xi + 0.1)) \cos(\varphi(5\pi\xi)) \\ -\frac{1}{4}(7(\xi + 0.1)) \cos(\varphi(5\pi\xi)) \\ -\xi(\xi + 1) \end{cases}$$

The two curves are displayed (β^1 in blue and β^2 in red) in Figure 5.6 (a). We simulate $N = 100$ curves per cluster using the same model as for parameterized curves in \mathbb{R}^2 where the discretization of I is $n = 100$ and \mathcal{I} is the 300×300 identity matrix. An example of simulated data forming the observed curves is given in Figure 5.6 (b,c) with two variance levels $\sigma^2 = 0.01$ and $\sigma^2 = 0.1$, respectively. Table 5.2 summarizes the mean clustering errors at each level where it is shown that our method is very accurate and has a better predictor. This result clearly shows the utility of estimating the optimal CDF when maximizing the log-posterior distribution on their associated coefficients on \mathcal{S}^{m-1} ($m = 30$) to reach good results.

Table 5.3: Human cochlea: Mean clustering error (MCE), specificity (SP) and sensibility (SE).

Methods	MCE	SP	SE
TPCA-GMM	41.75%	58.07%	58.5%
TPCA-kmeans	41.54%	58.44%	58.5%
GPA-kmeans	25.11%	74.4%	75.45%
GPA-kmedoids	10.85%	89.8%	88.41%
Proposed	4.26%	94%	97.73%

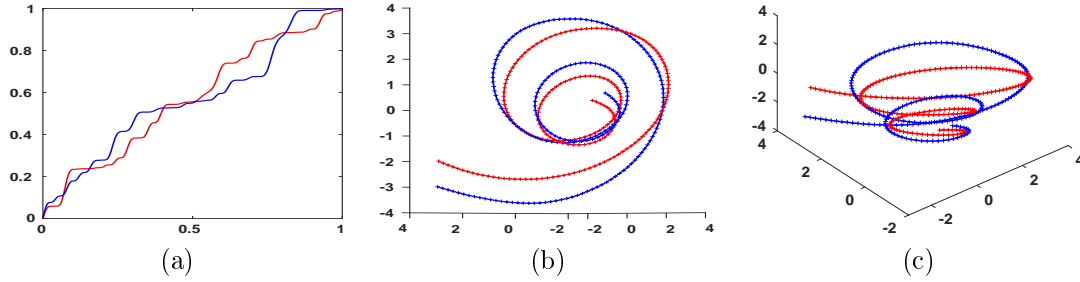


Figure 5.8: The optimal CDF estimates $\hat{F}^{k,30}$ (a) and the Fréchet means of curves $\tilde{q}^{k,30}$ for above view (b) and front view (c). Cluster of female: $k = 1$ (blue) and cluster of male: $k = 2$ (red).

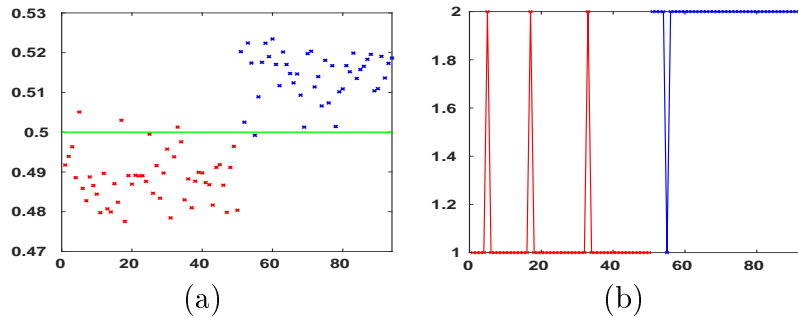


Figure 5.9: The conditional probability that i -th cochlea belongs to the first sub-population of female: $\mathbb{P}(C_i = 1|q_i)$ (a) and the resulting cluster (b). Cluster of female: $k = 1$ (blue) and cluster of male: $k = 2$ (red).

5.5.2 Real data

This section shows the importance of cluster analysis in 3D real cochlea, the main organ of hearing.

Human cochlea. For the first real data, we use a total of 94 X-ray medical CT images representing adult individuals (see Figure 5.7 for an example) Braga et al. (2015). In order to, first, detect the presence of differences in 3D cochlear shape and, second, assess the reliability of our method to form homogeneous subgroups, we overcame the drawbacks of all previously proposed approaches that ignored the intrinsic nonlinearity of the cochlear geometry.

We validate the proposed method to cluster humans (male or female) from their cochlea. The search of the optimal threshold minimizing the clustering error is based on the ROC curve where we consider the False Negatives (FN: female but classified as male) and the False Positives (FP: male but classified as female). The Mean clustering error (MCE) as well as the sensibility (SE) and the specificity

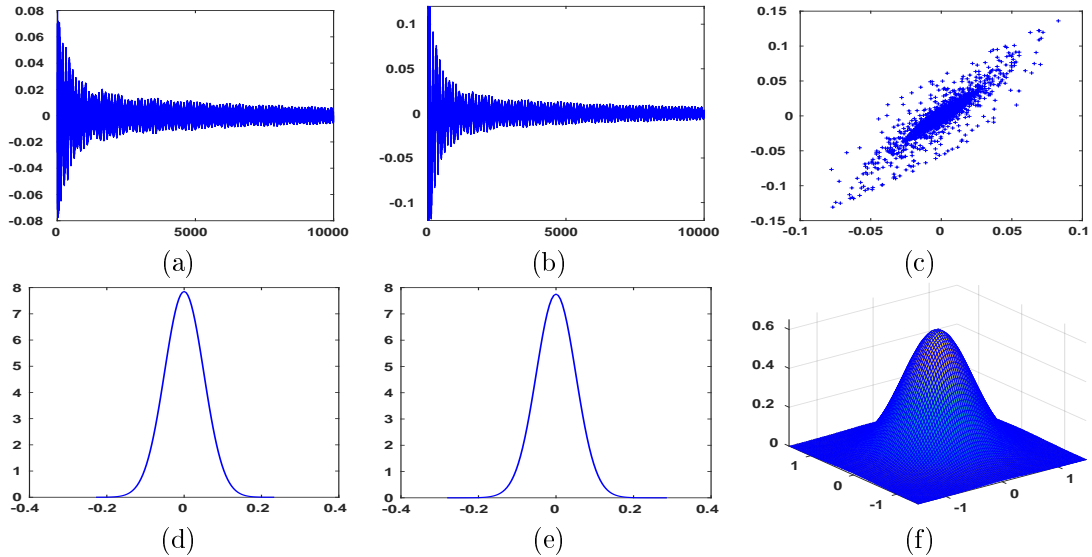


Figure 5.10: Markov chain values of a_1^1 (a), a_2^1 (b), and (a_1^1, a_2^1) (c). Posterior distributions of a_1^1 (d), a_2^1 (e), and (a_1^1, a_2^1) (f).

(SP) are reported in Table 5.3. We point out that our Bayesian method performs much better than the four baseline methods mentioned above. We can also observe that the GPA-kmedoids achieves a better accuracy than GPA-kmeans and both TPCA methods with a great margin. Accordingly, we prove the utility of finding the optimal CDF for each cluster when selecting the MAP estimate $\hat{\mathbf{A}} = (\hat{A}^1, \hat{A}^2)$. Figure 5.8 (a) illustrates the CDF estimate $\hat{F}^{k,30}$ for k -th cluster as function of the corresponding coefficients \hat{A}^k , $k = 1, 2$. In Figure 5.8 (b,c), we plot the Fréchet mean $\tilde{q}^{k,30}$ for both clusters from two different views.

For more details, we plot the conditional probability that each cochlea belongs to the first sub-population, assumed to be of female, as red dotted and those of male as blue dotted in Figure 5.9 (a). A good separability between the two clusters is clearly visible when estimating the optimal CDF for each sub-population. Hence, the mapping from the probability interval $[0, 1]$ to the output space, taking values in $\{1, 2\}$, is smooth and can be easily managed. Figure 5.9 (b) represents the resulting clusters when fixing the optimal threshold from the ROC curve.

We give an illustration of the spherical HMC sampling particularly for the two first components of A^1 : a_1^1 , a_2^1 and both jointly. We show the trajectory of Markov chains in Figure 5.10 (a,b,c) where sampled values are centered near the means: 0.02 and 10^{-3} , respectively. Figure 5.10 (d,e,f) displays their posterior distribu-

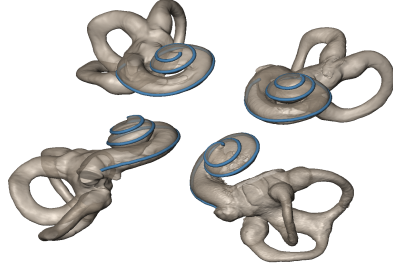


Figure 5.11: Some PAR cochlear hominin recovered from Kromdraai (South Africa) with blue cochlear curve.

Table 5.4: Hominin cochlea: Mean clustering error (MCE).

Methods	MCE
TPCA-GMM	15%
TPCA-kmeans	20%
GPA-kmeans	12.5%
GPA-kmedoids	10%
Proposed	0%

tions.

Hominin cochlea. A second real data is formed by 80 X-ray medical CT images representing newly discovered fossil hominin specimens. For this dataset, the hominin set will be grouped into $K = 5$ clusters: Humans (HSS), Paranthropus (PAR), Gorillas (GOR), Chimpanzees (PAN) and Australopithecus (AUS). Some examples of PAR cochlear hominin are given in Figure 5.11.

The mean clustering errors are reported in Table 5.4 where optimal values are reached by our proposal with a significant margin. Moreover, Figure 5.12 illustrates the Fréchet mean of each sub-population resulting from the Gibbs sampling algorithm. When comparing existing methods, TPCA-kmeans is less accurate than GPA when projecting shape vectors into the tangent space of the unit sphere. We

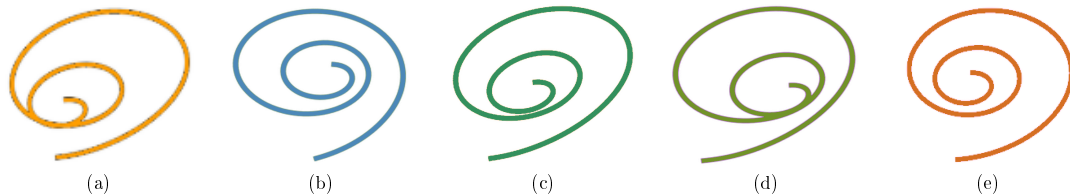


Figure 5.12: The Fréchet mean of HSS (a), PAR (b), GOR (c), PAN (d) and AUS (e).

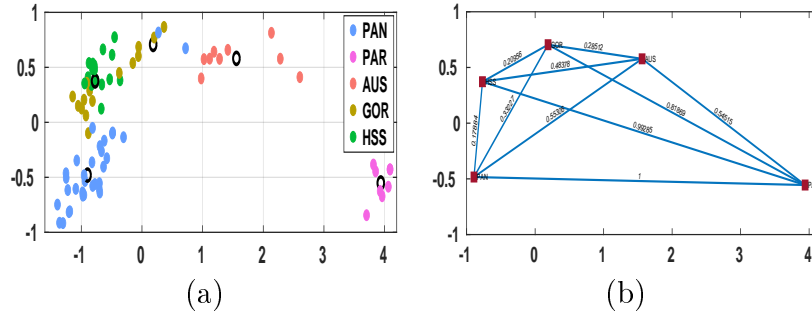


Figure 5.13: PC1 versus PC2 scatter plot obtained from TPCA (PC1 and PC2 represent 82.5% and 10.2% of variance, respectively) (a). The distance graph (normalized) with directional edges connecting the nodes of Fréchet means projected into the tangent space of the sphere formed by normalized curves (b).

give an illustration of projecting cochlea into a two-dimensional linear sub-space in Figure 5.13 (a). We add the normalized distance graph between projected Fréchet means in Figure 5.13 (b). For this experiment, we only keep the most important directions for TPCA (PC1 and PC2) having the biggest variance values: 82.5% and 10.2%, respectively. This confirms results obtained in Table 5.4 due to the overlap between projected observations especially between GOR and HSS. This makes the Euclidean distance, operating on the tangent space of the sphere, enable to give a good clustering performance.

5.6 Conclusion

We have introduced a new population background with registration and Bayesian clustering frameworks for curves. We have considered that each sub-population depends on its optimal local distribution and we have formulated the problem to estimate all of them jointly. Thanks to the Riemannian geometry and the Fisher-Rao metric, the proposed method solves the optimization task, originally defined on the infinite-dimensional and complex group of reparametrizations, with the use of a more practical optimization. We have tested our model on multiple simulated and real datasets. Compared to some existing methods, we showed several benefits and better accuracy when estimating the optimal local distribution of each sub-population.

Chapter 6: Conclusion and prospects

In this chapter, we conclude the thesis by summarizing our main contributions and results. We also highlight the ongoing works we are conducting as an extension.

6.1 Summary of the contributions

Throughout this thesis, we have shown that the quality of a Gaussian process model strongly depends on an appropriate covariance function and its hyper-parameters selection. We can summarize our work in the following items:

- To model such complex and high-dimensional inputs, we introduced the scalable Gaussian process classifier. The key idea is to decompose the overall classification into learning a feature space mapping (embedding) and a Gaussian process classifier that maps from this feature space to the observed space. We have introduced a new technique of manifold embedding for dimensionality reduction with a mapping defined on a Reproducing Kernel Hilbert Space. Both the input transformations and the Gaussian process classifier are learned jointly by maximizing the approximate log-marginal likelihood.
- We have extended the classical notion of Gaussian process from vector inputs to constrained functional inputs when introducing the Gaussian process indexed by probability density functions. We showed a theoretical result that the covariance function is well defined thanks to the underlying Riemannian geometry. This framework has the capacity of inferring and classifying both high-dimensional and univariate functional inputs.
- We have proposed a framework for registration and Bayesian clustering of shapes of curves as elements of a Riemannian manifold. We took advantages of a representation due to its invariance proprieties to Euclidean transformations in shape analysis. Thanks to the Fisher-Rao metric and the Gaussian process benefits, we reduced the complexity of estimating reparametrization functions, identified with local distributions of shapes, directly in an

infinite-dimensional group of diffeomorphisms. Using our proposed version, the inference become more affordable on the resulting coefficients belonging to the finite-dimensional sphere. Our problem was performed with the Hamiltonian dynamic on the sphere using the spherical HMC sampling.

6.2 Future work and prospects

In this thesis, we always make the link between Riemannian representations and the Hilbert sphere due to its nice statistical and geometric tools. In fact, it simplifies many basic notions where we have all analytic expressions of geodesics, exponential maps, log maps, Fréchet means, ect. However, it would be interesting to generalize the proposed models for more complex Riemannian manifolds with their corresponding metrics. Especially, we can extend our idea to 2D reparametrization functions, identified with local distributions for shapes defined on bivariate domains, e.g., shape analysis of surfaces.

Throughout this thesis, the metric was fixed to be the Fisher-Rao, the only metric invariant to reparametrizations. What happens if we change this metric ? If we take a less advantageous metric, can we revise it to check several proprieties usually needed in shape analysis ?

Bibliography

- Atapattu, S., Tellambura, C., and Jiang, H. (2011). A mixture Gamma distribution to model the snr of wireless channels. *IEEE Transactions on Wireless Communications*, 10:4193–4203.
- Bachoc, F., Gamboa, F., Loubes, J.-M., and Venet, N. (2018). A Gaussian process regression model for distribution inputs. *IEEE Transactions on Information Theory*, 64:6620–6637.
- Belkin, M. and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15:1373–1396.
- Belle, A., Thiagarajan, R., Soroushmehr, S. M. R., Navidi, F., Beard, D., and Najarian, K. (2015). Big data analytics in healthcare. *BioMed Research International*.
- Berge, J. M. F. T. (1977). Orthogonal procrustes rotation for two or more matrices. *Psychometrika*, 42:267–276.
- Bernal, J., Dogan, G., and Hagwood, C. R. (2016). Fast dynamic programming for elastic registration of curves. In *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1066–1073.
- Botev, Z. I., Grotowski, J. F., and Kroese, D. P. (2010). Kernel density estimation via diffusion. *The Annals of Statistics*, 38:2916–2957.
- Braga, J., Loubes, J.-M., Descouens, D., Dumoncel, J., Thackeray, J. F., Kahn, J.-L., de Beer, F., Riberon, A., Hoffman, K., Balaesque, P., and Gilissen, E. (2015). Disproportionate cochlear length in genus homo shows a high phylogenetic signal during apes’ hearing evolution. *PLOS ONE*, 10:1–23.

- Braga, J., Samir, C., Risser, L., Dumoncel, J., Descouens, D., Thackeray, F., Balaesque, P., Oettlé, A., Loubes, J.-M., and Fradi, A. (2019). Cochlear shape reveals that the human organ of hearing is sex-typed from birth. *Scientific Reports*, 9:1–9.
- Brigant, A. L., Arnaudon, M., and Barbaresco, F. (2015). Reparameterization invariant metric on the space of curves. In *Geometric Science of Information (GSI)*, pages 140–149, Palaiseau, France.
- Cai, Y. and Judd, K. L. (2010). Stable and efficient computational methods for dynamic programming. *Journal of the European Economic Association*, 8:626–634.
- Calandra, R., Peters, J., Rasmussen, C. E., and Deisenroth, M. P. (2016). Manifold Gaussian processes for regression. In *International Joint Conference on Neural Networks (IJCNN)*, pages 3338–3345. IEEE.
- Carlin, B. P. and Louis, T. A. (1997). Bayesian and empirical Bayes methods for data analysis. *Statistics and Computing*, 7:153–154.
- Dang, K., Clerc, M., Vandersteen, C., Guevara, N., and Gnansia, D. (2015). In situ validation of a parametric model of electrical field distribution in an implanted cochlea. In *Conference on Neural Engineering (NER)*, pages 667–670, Montpellier, France.
- DeChant, C., Wiesner-Hanks, T., Chen, S., Stewart, E. L., Yosinski, J., Gore, M. A., Nelson, R. J., and Lipson, H. (2017). Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning. *Phytopathology*, 107:1426–1432.
- Djlonga, J., Krause, A., and Cevher, V. (2013). High-dimensional Gaussian process bandits. In *Advances in Neural Information Processing Systems 26*, pages 1025–1033. Curran Associates, Inc.
- Dryden, I. L. and Mardia, K. V. (1998). *Statistical shape analysis*. Wiley, Chichester.

- Dryden, I. L. and Mardia, K. V. (2016). Statistical shape analysis with applications in R. 2nd edition.
- Duane, S., Kennedy, A., Pendleton, B., and Roweth, D. (1987). Hybrid Monte Carlo. *Physics Letters B*, pages 216 – 222.
- Fradi, A., Feunteun, Y., Samir, C., Baklouti, M., Bachoc, F., and Loubes, J.-M. (2020). Bayesian regression and classification using gaussian process priors indexed by probability density functions. *Information Sciences*, 548:56–68.
- Fradi, A. and Samir, C. (2020). Bayesian cluster analysis for registration and clustering homogeneous subgroups in multidimensional functional data. *Communications in Statistics - Theory and Methods*, 49:1–17.
- Fradi, A., Samir, C., and Yao, A. (2018). Manifold-based inference for a supervised Gaussian process classifier. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4239–4243.
- Gao, H., Chen, W., and Dou, L. (2015). Image classification based on support vector machine and the fusion of complementary features. *ArXiv*.
- Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models. *Bayesian Analysis*, pages 515–533.
- Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 6:721–741.
- Ghanem, R. G. and Spanos, P. D. (1991). *Stochastic finite elements: A spectral approach*. Springer-Verlag, Berlin, Heidelberg.
- Gorban, A. N., Kgl, B., Wunsch, D. C., and Zinovyev, A. (2008). *Principal manifolds for data visualization and dimension reduction*. Springer Publishing Company, Incorporated, 1st

- edition.
- Gower, J. C. (1975). Generalized procrustes analysis. *Psychometrika*, 40:33–51.
- Grogan, M. and Dahyot, R. (2017). Shape registration with directional data. *Pattern Recognition*, 79:452–466.
- Hamerly, G. and Drake, J. (2015). Accelerating lloyd’s algorithm for k-means clustering. Springer International Publishing, pages 41–78.
- Hassner, T., Itcher, Y., and Kliper-Gross, O. (2012). Violent flows: Real-time detection of violent crowd behavior. pages 1–6. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops.
- Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57:97–109.
- Holbrook, A., Lan, S., Streets, J., and Shahbaba, B. (2017). The nonparametric Fisher information geometry and the chi-square process density prior. arXiv:1707.03117.
- Huang, W., Gallivan, K. A., Srivastava, A., and Absil, P.-A. (2016). Riemannian optimization for registration of curves in elastic shape analysis. *Journal of Mathematical Imaging and Vision*, 54:320–343.
- Amari, S. (1983). Differential geometry of statistical inference. In *Probability Theory and Mathematical Statistics*, pages 26–40, Berlin, Heidelberg. Springer.
- Jost, J. (2011). Riemannian geometry and geometric analysis, chapter 3, pages 89–131. Springer-Verlag, Berlin, Heidelberg.
- Journée, M., Nesterov, Y., Richtárik, P., and Sepulchre, R. (2010). Generalized power method for sparse principal component analysis. *J. Mach. Learn. Res.*, 11:517–553.
- Karaboga, D. and Basturk, B. (2008). On the performance of artificial bee colony (abc) algorithm.

- Appl. Soft Comput., 8:687–697.
- Karcher, H. (1977). Riemannian center of mass and mollifier smoothing. *Communications on Pure and Applied Mathematics*, 30:509–541.
- Kendall, D. G. (1984). Shape manifolds, procrustean metrics, and complex projective spaces. *Bulletin of the London Mathematical Society*, 16(2):81–121.
- Klassen, E., Srivastava, A., Mio, W., and Joshi, S. H. (2004). Analysis of planar shapes using geodesic paths on shape spaces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26:372–383.
- Kneip, A. and Ramsay, J. O. (2008). Combining registration and fitting for functional models. *Journal of the American Statistical Association*, 103:1155–1165.
- Krakowski, K. A. and Manton, J. H. (2007). On the computation of the Karcher mean on spheres and special orthogonal groups. In *Proc. Workshop Robot. Math.*
- Kruskal, J. B. (1964). Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29:115–129.
- Kuczumarski, R. J., Ogden, C. L., Guo, S. S., Grummer-Strawn, L. M., Flegal, K. M., Mei, Z., Wei, R., Curtin, L. R., Roche, A. F., and Johnson, C. L. (2002). 2000 CDC growth charts for the united states: methods and development. *Vital and Health Statistics* 11, 246:1–190.
- Kumar, P. and Kumar, A. (2018). Haemato-biochemical changes in dogs infected with babesiosis. *International Journal of Chemical Studies*, pages 25–28.
- Lan, S., Zhou, B., and Shahbaba, B. (2014). Spherical Hamiltonian Monte Carlo for constrained target distributions. In *Proceedings of the 31st International Conference on Machine Learning (PMLR)*, pages 629–637, Beijing, China.
- Lang, S. (1998). *Fundamentals of differential geometry*. Graduate texts in mathematics. Springer, Island.

- Lee, J. A. and Verleysen, M. (2007). Nonlinear dimensionality reduction. Springer Publishing Company, Incorporated, 1st edition.
- Liu, X. and Muller, H.-G. (2004). Functional convex averaging and synchronization for time-warped random curves. *Journal of the American Statistical Association*, 99:687–699.
- Ma, Y. and Fu, Y. (2012). *Manifold learning theory and applications*. CRC Press.
- MacKay, D. J. C. (1998). Introduction to Gaussian processes. In *Neural Networks and Machine Learning*, NATO ASI Series, pages 133–166. Kluwer Academic Press.
- Mallasto, A. and Feragen, A. (2017). Learning from uncertain curves: the 2-Wasserstein metric for Gaussian processes. In *Advances in Neural Information Processing Systems 30*, pages 5660–5670. Curran Associates, Inc.
- McDonnell, M. D., Burkitt, A. N., Grayden, D. B., Meffin, H., and Grant, A. J. (2010). A channel model for inferring the optimal number of electrodes for future cochlear implants. *IEEE Transactions on Information Theory*, 56:928–940.
- Minasny, B. and Mcbratney, A. (2005). The Matérn function as a general model for soil variograms. *Geoderma*, 128:192–207.
- Minka, T. P. (2001). Expectation propagation for approximate Bayesian inference. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, page 362–369. Morgan Kaufmann Publishers Inc.
- Neal, R. M. (1997). Monte Carlo implementation of Gaussian process models for Bayesian regression and classification. Technical report, USA.
- Neal, R. M. (2010). MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 54:113–162.
- Nguyen, H.-V. and Vreeken, J. (2015). Non-parametric Jensen-Shannon divergence. In *Ma-*

- chine Learning and Knowledge Discovery in Databases, pages 173–189. Springer International Publishing.
- Oliva, J. B., Neiswanger, W., Poczos, B., Schneider, J., and Xing, E. (2014). Fast distribution to real regression. In Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, pages 706–714, Scottsdale, Arizona, USA. JMLR.org.
- Pistone, G. and Sempi, C. (1995). An infinite-dimensional geometric structure on the space of all the probability measures equivalent to a given one. *The Annals of Statistics*, 23:1543–1561.
- Póczos, B., Singh, A., Rinaldo, A., and Wasserman, L. A. (2013). Distribution-free distribution regression. In Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics, volume 31, pages 507–515, Scottsdale, Arizona, USA. JMLR.org.
- Ramsay, J. O. and Dalzell, C. J. (1991). Some tools for functional data analysis. *The Royal Statistical Society*, 53:539–572.
- Ramsay, J. O. and Li, X. (1998). Curve registration. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60:351–363.
- Rao, C. R. (1945). Information and the accuracy attainable in the estimation of statistical parameters. *Bulletin of Calcutta Mathematical Society*, 37:81–91.
- Rasmussen, C. E. and Williams, C. K. I. (2006). Gaussian processes for machine learning. The MIT Press.
- Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326.
- Schölkopf, B., Herbrich, R., and Smola, A. J. (2001). A generalized representer theorem. In Proceedings of the 14th Annual Conference on Computational Learning Theory and and 5th European Conference on Computational Learning Theory, page 416–426, Berlin, Heidelberg.

- Springer-Verlag.
- Seeger, M. (2005). Expectation propagation for exponential families. Technical report.
- Snelson, E. and Ghahramani, Z. (2006). Variable noise and dimensionality reduction for sparse Gaussian processes. In Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence, page 461–468, Arlington, Virginia, USA. AUAI Press.
- Sriperumbudur, B. K., Fukumizu, K., Gretton, A., Bernhard, and Lanckriet, G. R. G. (2010). Non-parametric estimation of integral probability metrics. In 2010 IEEE International Symposium on Information Theory, pages 1428–1432.
- Srivastava, A., Jermyn, I., and Joshi, S. (2007). Riemannian analysis of probability density functions with applications in vision. In 2007 IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8.
- Srivastava, A. and Klassen, E. (2016). Functional and shape data analysis. Springer-Verlag New York.
- Srivastava, A., Klassen, E., Joshi, S., and Jermyn, I. (2011). Shape analysis of elastic curves in euclidean spaces. IEEE transactions on pattern analysis and machine intelligence, pages 1415–1428.
- Sutherland, D. J., Oliva, J. B., Póczos, B., and Schneider, J. (2016). Linear-time learning on distributions with approximate kernel embeddings. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, pages 2073–2079. AAAI Press.
- Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. Science, 290:2319–2323.
- Van Der Maaten, L. (2014). Accelerating t-SNE using tree-based algorithms. J. Mach. Learn. Res., 15:3221–3245.

- Van Ravenzwaaij, D., Cassey, P., and Brown, S. (2018). A simple introduction to Markov Chain Monte Carlo sampling. *Psychonomic Bulletin & Review*, 25:143–154.
- Vigsnes, M., Kolbjørnsen, O., Hauge, V. L., Dahle, P., and Abrahamsen, P. (2017). Fast and accurate approximation to kriging using common data neighborhoods. *Mathematical Geosciences*, 49:619–634.
- Williams, C. K. I. and Barber, D. (1998). Bayesian classification with Gaussian processes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20:1342–1351.
- Williams, C. K. I. and Rasmussen, C. E. (1996). Gaussian processes for regression. pages 514–520. MIT-press.
- Wilson, A. G. and Adams, R. P. (2013). Gaussian process kernels for pattern discovery and extrapolation. In *ICML (3), JMLR Workshop and Conference Proceedings*, pages 1067–1075. JMLR.org.
- Ying, S., Wang, Y., Wen, Z., and Lin, Y. (2016). Nonlinear 2d shape registration via thin-plate spline and lie group representation. *Neurocomputing*, 195:129–136.
- Younes, L. (2000). Computable elastic distances between shapes. *SIAM Journal on Applied Mathematics*, 58:565–586.
- Zhang, Z. and Wang, J. (2007). MLLE: modified locally linear embedding using multiple weights. In *Advances in Neural Information Processing Systems 19*, pages 1593–1600. MIT Press.
- Zhang, Z. and Zha, H. (2005). Principal manifolds and nonlinear dimension reduction via local tangent space alignment. *SIAM J. Sci. Comput.*, 26:313–338.