Sparse Intrinsic Gaussian Processes in Complex Constrained Domains with Application of Bayesian Optimisation

Yuan Liu

Submitted in fulfilment of the requirements for the
Degree of Doctor of Philosophy

School of Mathematics & Statistics
College of Science and Engineering
University of Glasgow

October 2024

Abstract

As scientific research advances, more and more data are no longer limited to traditional Euclidean space, but extend to spaces with more complex geometric structures, such as complex constrained domains and Riemannian manifolds. Riemannian manifolds are increasingly being recognized as an important tool in data analysis and machine learning due to their widespread use in multiple scientific fields and in real-word contexts. For example, lakes can be modeled as manifolds to better understand their geographic structure and dynamics in environmental studies. In order to model such manifolds in real world situations, an increasing number of statistical tools are developed for estimation over a manifold. When considering regression on manifolds, inspired by the success of Gaussian Processes (GPs) in Euclidean spaces, this thesis aims to provide novel tools in order to efficiently and accurately estimate surfaces using GPs tailored for manifolds.

Traditional GPs typically use kernels that rely on Euclidean distance to define the covariance between data points on the target surface, such as the radial basis function (RBF) kernel. Traditional GPs cannot be directly applied to manifolds due to their failure to accurately capture the underlying structure, especially in the presence of gaps and complex boundaries. The heat kernel describes the heat diffusion on the manifold, which reflects the manifold's geometric properties, but only specific manifolds have closed-form expressions. Intrinsic GPs proposed in [114] use the transition density of Brownian motion (BM) on the manifold to approximate the heat kernel, thereby capturing the manifold's intrinsic geometric characteristics and enabling more accurate regression on manifolds. However, Intrinsic GPs face issues near boundaries due to resampling BM paths when crossing the boundary, causing inaccurate predictions near the boundary. According to the definition of the Neumann boundary condition, the BM path should be reflected when it crosses the boundary. This thesis proposes a "reflection" method to address

this issue, leading to more accurate predictions at the boundary.

Additionally, Intrinsic GPs are constrained by the computational complexity of simulating BM paths, especially on large-scale or highly complex manifolds, which make them highly computationally intensive. This thesis investigates the feasibility of sparse methods in Intrinsic GPs, which use inducing points as intermediaries to facilitate information transmission from training points to test points, aiming to simplify the computational complexity without sacrificing inference accuracy. This thesis first proposes Sparse Intrinsic GPs using a Deterministic Inducing Conditional approach (SI-GPDIC), which is straightforward to implement and computationally efficient; however, it is sensitive to the location of a small number of inducing points. The Sparse Intrinsic Gaussian Process using a Deterministic Training Conditional approach (SI-GPDTC) is then proposed, which is less sensitive to the location of inducing points, achieving a balance between computational efficiency and inference precision. Considering approximating the true posterior distribution with a simpler, more tractable distribution by minimizing the divergence metric between them, this thesis develops the Sparse Intrinsic Gaussian Process with Variational Inference (SI-GPVI), a powerful tool for regression on complex manifolds. Graph GPs, which utilize the graph Matérn kernel on the undirected graph constructed from the manifold, and Traditional GPs, which directly use the Euclidean distance-based RBF kernel, are employed for comparison with the three Sparse Intrinsic GPs developed in this thesis. The performance of the proposed methods is demonstrated using three examples: the 2D U-shape, the 3D Bitten-torus, and the real-world dataset of the Aral Sea, with SI-GPVI performing particularly well.

Finally, motivated by the success of Bayesian optimisation (BO) in Euclidean space, this thesis proposes novel approaches to construct Intrinsic BO on manifolds, building upon previous research. The proposed GPs (introduced earlier in the thesis), serve as surrogate models in the BO approach, providing the acquisition function the probability of improvement (PI), with accurate information about the underlying manifold structure. Benefiting from the surrogate models' ability to capture the structure of manifolds, the proposed BO algorithms—Intrinsic BO with DTC and Intrinsic BO with VI—achieve better results compared to Graph BO, based on Graph GPs, and Traditional BO, based on Traditional GPs. Among them, Intrinsic BO with DIC shows unstable performance due to its predictive variance providing inaccurate uncertainty when estimating points that are far from inducing points, whereas Intrinsic BO with VI demonstrates particularly strong performance, excelling in both accuracy and efficiency.

Contents

List of Tables

List of Figures

Acknowledgements

I sincerely thank my two supervisors, Dr. Mu Niu and Prof. Claire Miller. It is their guidance and companionship that have led me to where I am today. From their support and care during the pandemic to their instruction and mentorship after work routines were restored, everything that has happened over the past four years has been invaluable. Words cannot fully express my gratitude to them. Their mentorship has extended beyond research, as their work ethic and academic rigor will continue to influence my future career development.

I would also like to express my gratitude to the School of Mathematics & Statistics for providing such abundant academic resources and an excellent learning environment. The sunset by the office window is the most beautiful view. I am also thankful to the China Scholarship Council and the University of Glasgow's joint scholarship, which sponsored my PhD. It is with their support that I was able to gain such a valuable experience.

I am also grateful to my parents. It is their support and encouragement that have guided me on my academic journey. Our conversations, despite the time difference, accompanied me through countless late nights. Over the past two years, my family has faced some challenging times, but they have set an inspiring example for me. Family is always the source of courage.

Lastly, I want to thank the wonderful friends from various countries that I made during my time in the UK, including my colleagues, my roommates, and my neighbours. Wish everyone a bright and beautiful future. And a special 'thank you' to my cat Tuanzi, who always brightens my mood whenever I feel upset.

Declaration

I declare that all the work presented in this thesis has been done by myself under the supervision of Dr. Mu Niu and Prof. Claire Miller, except where otherwise explicitly stated. This thesis represents work that is done in the period (2020-2024) at the School of Mathematics and Statistics to achieve the degree of Doctor of Philosophy at university of Glasgow.

Part of the work in Chapter 4 has been presented as a poster in the 38th International Workshop on Statistical Modelling (IWSM) in Durham, UK and is published as

- Liu, Yuan, Mu Niu, and Claire Miller. "Sparse intrinsic Gaussian processes for prediction on manifolds: extending applications to environmental contexts." International Workshop on Statistical Modelling. Cham: Springer Nature Switzerland, 2024.

  https://doi.org/10.1007/978-3-031-65723-8_29

Part of the work in Chapter 7 has been presented in the Research Students' Conference 2022 (RSC) in Nottingham, UK.

List of Abbreviations

GPs	Gaussian Processes
RBF	Radial Basis Function
BM	Brownian Motion
BO	Bayesian Optimisation
SDE	Stochastic Differential Equation
PI	Probability of Improvement
DIC	Deterministic Inducing Conditional
DTC	Deterministic Training Conditional
VI	Variational Inference
SI-GPDIC	Sparse Intrinsic Gaussian Process with Deterministic Inducing Conditional
SI-GPDTC	Sparse Intrinsic Gaussian Process with Deterministic Training Conditional
SI-GPVI	Sparse Intrinsic Gaussian Process with Variational Inference
CDF	Cumulative Distribution Function
PDF	Probability Distribution Function
GL	Graph Laplacian
RMSE	Root Mean Square Error

SPDE	Stochastic Partial Differential Equation
KNN	K-Nearest Neighbors
PLL	Predictive Log Likelihood

List of Notations

M	the Riemannian manifold with boundary
∂M	the boundary of M
g	the metric tensor of Riemannian manifold with boundary
Δ_s	the Laplace–Beltrami operator of the Riemannian manifold with boundary
M'	the Riemannian manifold without boundary
g'	the metric tensor of Riemannian manifold without boundary
Δ'_g	the Laplace–Beltrami operator of the Riemannian manifold without boundary
S	the grid points set on M
G'	the number of the grid points set S
$\mathbf{f}_r$	the vector of $f(.)$ at all grid points set S
$\mathscr{D}$	the training points set on M
n	the number of the training points set $\mathscr{D}$
$\mathbf{f}_{\mathscr{D}}$	the vector of $f(.)$ at training points set $\mathscr{D}$
y	the observation value of the objective function to $\mathscr{D}$
z	the inducing points set on M
u	the vector of $f(.)$ at all inducing points set z

m	the number of the inducing points set z
ϕ	the smooth local mapping function from $\mathbb{R}^d$ to M
ς	the acquisition function
ε	the degree of exploration in the acquisition function
ε	the observation noise
σ_n^2	the variance of the noise ε
PI	the probability of improvement
φ	the cumulative distribution function
Σ	the covariance matrix
Q	the covariance matrix approximated using the information from the inducing points
μ	the predictive mean for different methods determined by specific subscripts
σ	the predictive variance for different methods determined by specific subscripts
α	the variance parameter in the RBF kernel
l	the length-scale parameter in the RBF kernel
t	the time parameter in the heat kernel
κ	the degree of dependency parameter in the Matérn kernel
Γ	the Gamma function used in the Matérn kernel
ν	the degree of the mean-square differentiability in the Matérn kernel
K_ν	the second kind of deformed Bessel function in the Matérn kernel
G	the undirected graph
V	the vertices of G
E	the edges of G

W	the adjacency matrix of G
$\tilde{\mathbf{A}}_{ij}$	the approximation of the adjacency matrix W
ζ	the bandwidth used in W
S_K	the K-Nearest Neighbors sparsification coefficient
$\tilde{D}$	the degree matrix of G
$\mathscr{W}$	the Gaussian white noise re-normalized by a certain constant
Δ_{rw}	the random walk normalized Laplacian
λ_i	the eigenvalues of Δ_{rw}
f_i	the eigenvectors of Δ_{rw}

Chapter 1

Introduction

This chapter is the introduction to this thesis. It begins with a discussion of the motivation for the work, then the specific goals the thesis is trying to achieve. The chapter also introduces three case studies used throughout the thesis: a 2D example of the U-shape, a 3D example of the Bitten-torus, and the real-world dataset about the Aral Sea. Finally, the outline of the thesis is presented.

1.1 Motivation

Euclidean space is the fundamental space of geometry, widely used in many fields such as physics and computer science. Traditional statistical methods used to assume that data usually comes from Euclidean space and measure the relationships between these data points using Euclidean distance. Although in many contexts these methods work, they do not perform well when handling data generated from more complex constrained domains and manifolds. A manifold is a space that in the neighborhood of each point resembles a Euclidean space [161]. It usually has a more intricate global structure and frequently arises in real-world scenarios, such as the surface of a lake or the field of medical imaging.

In many fields, such as physics, computer vision, medical imaging, and geology, manifolds play a critical role in representing complex data structures. In the field of computer vision,

manifold methods are often used in shape-related vision problems because individual shapes can typically be viewed as differential manifolds. The shape space can be treated as a mathematical object, represented by constructing a manifold structure over it [173]. The structural properties of manifolds is utilized to study shape space structures from various perspectives [26], [75], [140], [173]. In the medical imaging field, many structures in human imaging exhibit manifold-like characteristics, with complex intrinsic geometries and boundaries, for example, the 3D-heart model [152]. The following references [42], [88], [157] and [121] employ measuring a diffusion process on a manifold to simulate the diffusion of water molecules in tissues, as observed in Diffusion Tensor Imaging (DTI). Also, statistical analysis of manifold-valued data has gained a great deal of attention in neuroimaging applications [32], [174], [72], [67]. Manifolds have numerous applications in geological research. Advances in geostatistical modeling are essential for accurate subsurface characterization, as traditional methods often struggle with the complexity of geological media [118]. Treating geological formations as manifolds help capture many intrinsic geometric features and boundary information [66], [98]. Manifolds can also be used to model bodies of water, such as Lake Michigan, Gull Lake in [131], and the Aral Sea in [166]. This application will aid in monitoring and managing water resource pollution, contributing to environmental protection area. Thus, research on manifolds is crucial given the wide range of manifold-based applications.

On manifolds, Euclidean-based models usually fail to capture the intrinsic geometry, leading to poor performance in tasks such as regression, classification, and optimisation. Nowadays, there has been substantial interest in developing statistical methods suitable for manifolds, where data are best characterised as elements of a Riemannian manifold rather than points in Euclidean space [44]. For illustration, Pennec [120] focuses on the theoretical formulation of the statistical framework in geodesically complete Riemannian manifolds; Fletcher et al. [43] introduce Principal Component Analysis (PCA) to study shape analysis on manifolds, with the main application in modeling three-dimensional anatomical structures (e.g., kidneys); Then, Fletcher et al. [44] extend the concept of the geometric median, a robust estimator of centrality, to manifold-valued data, with applications in robust atlas estimation; Subbarao and Meer [144] extend the mean shift algorithm to Riemannian manifolds and provided detailed derivations for specific manifold examples, such as matrix Lie groups and Grassmann manifolds; Asgharbeygi and Maleki [4] proposed the geodesic K-means algorithm, extending the classical K-means clus-

tering method to manifolds; and so on.

Further research could explore the application of Gaussian processes (GPs). GPs are widely used in machine learning, offering flexible and interpretable models with uncertainty quantification [172]. The application of GPs in machine learning contains: robotics and control [33], time-series modelling [128], reinforcement learning [83], [154], survival analysis [40] and some Bayesian numerical methods, for example, Bayesian optimisation [109], [138], [23], Bayesian quadrature [18], [73] and Bayesian differential equations solvers [29]. In addition to machine learning, GPs are also used to solve a variety of problems, such as health monitoring [142], inverse problems [143], tsunami modelling [132], computer models [76] and engineering design [45]. In these applications, the input data typically comes from Euclidean space, where GPs perform well.

This thesis is interested in regression and optimisation on manifolds. Inspired by this, this thesis aims to develop GP methods suitable for manifolds. The challenge lies in how to handle data with complex intrinsic geometries. In Euclidean space, the kernels used in Traditional GPs typically rely on Euclidean distance for modelling, for example, the radial basis function (RBF) kernel and the Matérn kernel [165]. If Traditional GPs are directly applied to manifolds, they will overlook the complex intrinsic geometry of the manifold and fail to accurately capture the underlying structure, which can lead to poor performance. The most direct solution to this problem is to replace the Euclidean distance with geodesic distance. The geodesic distance refers to the shortest path between two points along the surface of the manifold [162]. For example, in Euclidean space, which is a special type of manifold, the geodesic distance is simply the straight-line distance (the Euclidean distance) between two points p, q. On a sphere, the geodesic is the shorter arc of the great circle that passes through two points along the surface of the sphere. However, it is challenging to accurately approximate the geodesic distance when the manifold has complex intrinsic geometric features, both regarding algorithmic complexity and inadequate accuracy [89]. Directly replacing Euclidean distance with the Riemannian geodesic distance often leads to ill-defined kernels in many cases of interest, subsequently causing deviations in the performance of GPs [39].

This motivates the need for more sophisticated approaches which can take the characteristics of manifolds into account. Lin et al. [92] propose extrinsic covariance kernels by embedding

manifolds into higher-dimensional Euclidean space, but this approach is challenging for complex spaces. Dunson et al. [36], Borovitskiy et al. [17], Fichera et al. [41] and Bolin [15] contributes to developing GPs on graphs and metric graphs formed by data observed on the manifold, but these approaches do not perform well when the constructed graph fails to capture the manifold's intrinsic features.

When considering the process of filling an artificial lake with water, the way the water spreads will be influenced by the internal geometric features of the lake, such as its irregular boundaries and the various islands scattered within the lake. The process of water spread is similar to the diffusion of heat along the surface of an object. Inspired by this, the heat kernel can be used to describe the heat diffusion along the surface of a manifold, reflecting the intrinsic geometric features of the manifold. Research on heat kernels covers various aspects, including: heat kernels in presence of group structure [1], [2], [3], heat kernels in infinite dimensional spaces [8], [35], heat kernels on fractals and fractal-like spaces [53], [7], [54], heat kernels of non-linear operators [34], [11], heat kernels of non-symmetric operators [37], [125] etc. The heat kernel expansion is widely studied in both physics and mathematics literature (see [153] for a review). Grigor'yan [55] studies the Heat Kernels on weighted manifolds and their applications. Castillo et al. [25] develop intrinsic GP models on Riemannian manifolds by rescaling solutions of heat equations, but the constructed intrinsic kernels are often impractical to implement. Lawler [85] discusses the probabilistic connections between Brownian motion (BM) and the heat equation. Based on this view, Kozdron [81] finds the transition density of BM satisfies the heat equation, providing a new perspective for solving the heat equation. Niu et al. [114] propose using transition density of BM to approximate the heat kernel on manifolds.

Based on [114], this thesis aims to improve the accuracy of capturing manifold geometric features using BM and expand the application of GPs to higher-dimensional manifolds through the use of sparse methods. GPs, due to the matrix inversion involved in their computations, are very challenging to apply to large datasets and high-dimensional data, where the computational cost increases exponentially with the growth of data size and dimension. To solve this problem, there are several approximate methods have been proposed for Traditional GP in Euclidean space. Williams and Seeger [164] use the Nyström Method to speed up the computational required for standard GP. The Subset of Regressors (SoR) is proposed by Wahba [155] and adapted by Smola and Bartlett [136] to propose a sparse greedy approximation to GPs. Csató and Opper

[31] and Seeger et al. [133] focus on the method Projected Latent Variables (PLV), which solves the problem of the SoR approximation, also named the Projected Process Approximation (PPA) by Rasmussen and Williams [165]. Snelson and Ghahramani [137] propose another likelihood approximation Sparse Pseudo-input Gaussian processes (SPGP) to speed up GPs. A unifying view of these sparse approximate GPs can be found in [126]. Titsias [150] introduce a variational formulation for sparse approximations GPs. Inspired by these studies, this thesis aims to construct sparse intrinsic GPs suitable for manifolds, which can take into account the intrinsic geometric features of the manifold and be applied to high dimensional manifolds.

There is also great interest in optimisation problems on manifolds. Similarly, in the case of water pollution control, it is often necessary to identify and prioritize the most polluted areas of a water body. This requires the measurement of certain indicators, such as chlorophyll level, nutrients, etc. However, in the presence of budget constraints, it is a challenge to find the target point in a limited measurements. Therefore, finding the target point efficiently and accurately is essential for effective decision-making. Determining the optimal drilling locations which contain rich oil resources is crucial for oil exploration in the desert. Due to the high cost of drilling and the huge area of desert to be explored, it is essential to efficiently and accurately identify suitable drilling locations.

Given the need to find the optimal point under budget constraints, Bayesian Optimisation (BO) offers a powerful approach. BO is a highly effective global optimisation algorithm, useful in situations where evaluations are expensive or time-consuming. The central idea of BO is to build a model that can be updated and queried to drive optimisation decisions. It mainly consists of two components: the surrogate model-used to approximate the unknown objective function and the acquisition function, which guides the selection of the next sampling point by balancing exploration and exploitation. Shahriari et al. [135] give a review of surrogate models and divide them into into two categories: parametric models, such as the beta-Bernoulli model, the linear models and the Generalized linear models (GLMs) and nonparametric models, such as the GPs, the random forests and the deep neural network (DNN). There are several acquisition functions used in BO: including the Probability of Improvement (PI)-selecting points with the highest probability of improving over the current best value [82], [70], the Expected Improvement (EI)-maximising the expected improvement over the current best solution [110], the upper confidence bound algorithm (UCB)-choosing points based on the upper bound of the confidence interval

[139] and the information based policies, the Thompson Sampling (TS) [149] and the entropy search [134].

BO is impacting a wide range of areas, including intelligent environmental monitoring [103], interactive user interfaces [19], experimental design [6], reinforcement learning [20], designing games [77], sensor set selection [49], and robotics [95], [104] etc. The data in these applications typically comes from Euclidean space, where BO is widely applied. Inspired by this research, this thesis seeks to explore optimisation problems on manifolds, expanding the application of BO to manifolds. However, directly applying BO from Euclidean space to manifolds can lead to incorrect optimisation results. This is because, in traditional BO, the surrogate model relies on Euclidean distance to define relationships between points, which does not account for the structure of the manifold. When the Euclidean distance differs significantly from the actual distance on the surface of manifold, BO is misguided during the iterations, leading to poor performance and inaccurate results. The quality of the surrogate model's approximation of the objective function directly impacts the effectiveness of BO. Thus, the surrogate model should take the intrinsic geometric features of the manifold into account. After constructing GPs suitable for manifolds, this thesis aims to utilise the proposed GPs as surrogate models to expand the application of BO on manifolds. Ideally, the proposed BO will achieve better performance compared to traditional BO by accurately capturing the manifold's intrinsic geometric features.

After introducing the motivation behind this work, the next section shows the thesis goals, detailing the specific objectives.

1.2 Thesis Goals

The aim of this thesis is to solve regression and optimisation problems on manifolds by developing novel statistical approaches that use the intrinsic geometric features of the manifolds. Specifically, the goals are as follows:

- The aim is to improve the heat kernel estimation closing to the boundary of manifolds in intrinsic GPs, designed for manifolds. The current "resample" method leads to reduced accuracy near the boundary due to a lack of exploration, causing the boundary information

conveyed by the BM paths to be smaller than the actual boundary of the manifold. To achieve this, this research proposes the novel "reflection" approach, which is designed and compared to enhance the accuracy of boundary exploration on manifolds.

- The goal is to address the high computational cost associated with simulating BM paths in intrinsic GPs, while also resolving the numerical instability and computational burden of matrix inversion within GPs. This research aims to develop GPs that are applicable to complex high-dimensional manifolds, ensuring both computational efficiency and accuracy. To achieve this, this work proposes three novel methodologies, termed sparse intrinsic GPs on manifolds, among which variational inference demonstrates outstanding performance across three representative examples.

- The aim is to solve optimisation problems on manifolds. It is essential due to the presence of many real-world optimisation problems on manifolds, such as in environmental monitoring or geological exploration. Traditional methods, which rely on Euclidean space, often overlook these intrinsic geometric features, leading to inaccurate results. This work proposes novel BO methods specifically designed for manifolds, which can effectively capture their intrinsic structures. An innovative application is presented in the field of water pollution management, where the proposed BO methods are applied to find the areas with the highest chlorophyll level in the Aral Sea.

After presenting the motivation and goals of this work, the next section will introduce the examples used through this work.

1.3 Case Studies Presented

This section introduces the case studies used in the research, including different dimensions as well as real-world dataset. These examples will help demonstrate the effectiveness of the proposed methods in the subsequent chapters.

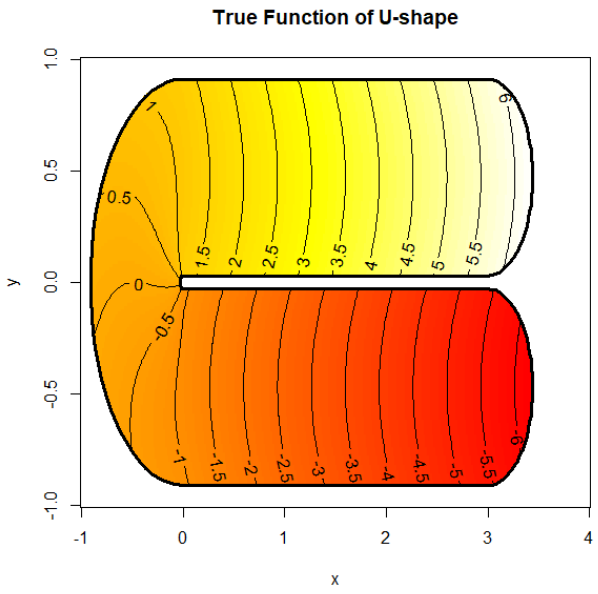


Figure 1.1: The U-shape domain with the test function shown as a colour map and contour plot over the region.

1.3.1 2D Example: the U-shape

The horseshoe-like U-shape defined as a subset of $\mathbb{R}^2$, serves as a 2-dimensional manifold simulated example for the GP approaches proposed and compared in this research. It modifies the simulation test presented in Wood et al. [168] and can be constructed through mgcv package in R [167]. Figure 1.1 gives the true function of the U-shape.

The U-shape presents a compact structure with two long arms separated by a narrow gap. When two points are located in the upper and lower arms, the Euclidean distance between them can be measured as the straight line between the two points. However, in reality, these points are separated by the boundary in the middle, and the actual distance, or path from one point to another requires going around the boundary within the domain. This difference emphasizes the limitations of using Euclidean distances to capture the real geometry of manifolds. Additionally, the value of the test function increases smoothly from the lower right to the upper right within the U-shape boundary ranging from -6.1188 to 6.1188, as shown by the color gradient and contour lines in Figure 1.1. The values between the upper and lower arms of the U-shape are significantly different due to the boundary separating the two regions. This gives the challenge in using

Euclidean distance for modelling on such manifolds since it fails to capture the true geometric separation. A total of 418 grid points S are uniformly distributed on the surface of the U-shape domain, ensuring an even representation of the space while respecting the boundary constraints. These points serve as the basis for applying the GP method. Details will be introduced in Chapter 3.

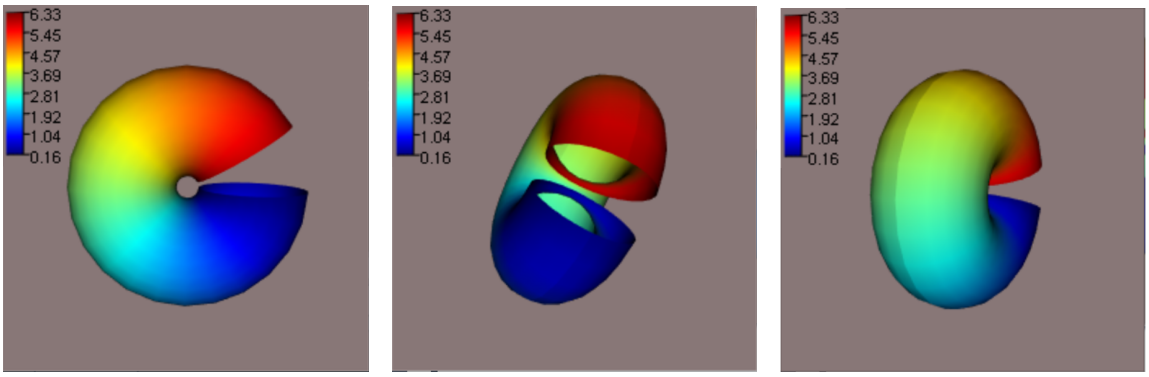
1.3.2 3D Example: the Bitten-torus

This research chooses the Bitten-torus as the representative of 3-dimensional manifolds. First, consider the torus, which is a surface of revolution generated by revolving a circle in three-dimensional space one full revolution about an axis that lies in the same plane as the circle [163]. It can be viewed as a two-dimensional manifold embedded in $\mathbb{R}^3$, which is constructed by four parameters: r radius of tube, R distance from center of the tube to the center of the torus, θ and ϕ are angles to make full circles while θ for angle of torus and ϕ for angle of tube.

This research keeps R and r fixed, while varying the angles θ and ϕ. By removing the lower right part of a torus, this research investigates problems on the Bitten-torus, which looks a donut with a bite in it. Figure 1.2 shows the Bitten-torus used in this work from three different perspectives. The value of the test function is displayed through colours in the figure, with low values shown in dark blue and high values in dark red, increasing smoothly from 0.1597541 to 6.334097922 on the surface of the Bitten-torus. A nonlinear function f is defined on the surface of Bitten-torus with

$$Y_i = f(x_i, y_i, z_i) + \varepsilon_i, \tag{1.1}$$

where x_i, y_i, z_i are the coordinates of a point on the surface and ε_i is the noise parameter. The detail for construction of the Bitten-torus and the derivation of the metric tensor are shown in the Chapter 6. On either side of the "bitten" region, although being geometrically close in Euclidean distance, the values are significantly different due to the separation by the boundary. If the modelling does not consider the manifold's intrinsic structure and boundary conditions, it will result in poor performance. There are 600 grid points uniformly distributed across the surface of the Bitten-torus, which will be used in the subsequent GPs and BO methods. Details will be elaborated in the following chapters.



(a) (b) (c)

Figure 1.2: The Bitten-torus with the test function shown as a colour map from three different perspectives.

1.3.3 Real World Dataset: the Aral Sea

The Aral Sea, as the real-world dataset used in this work, holds significant practical importance. The Aral Sea is a large inland lake located in Central Asia, lying between Kazakhstan to the north and Uzbekistan to the south. It was once the fourth-largest lake in the world, but due to poor water resource management, it began shrinking in the 1960s and largely dried up by the 2010s, becoming a prime example of an ecological disaster [160] [107]. How to effectively monitor water resource pollution and address it promptly is of great significance for environmental protection [158]. By analysing datasets related to the Aral Sea's chlorophyll levels, this work aims to develop models that offer useful predictions and guide decision-making for identifying the most polluted areas under budget constraints. This work consider an analysis of remotely-sensed chlorophyll data at 485 locations in the Aral Sea, which can be obtained from [166]. These 485 locations can be considered as grid points on the Aral Sea case study, which will be used for subsequent modelling. Figure 1.3 shows the chlorophyll levels of the Aral Sea, with the levels represented by the intensity of the color—the darker the color, the higher the chlorophyll level. The chlorophyll values range from 0 to 19.278724. The log of chlorophyll level is modelled as a function of the latitude and longitude coordinates of the measurement locations:

$$chl_i = f(\text{lon}_i, \text{lat}_i) + \varepsilon_i, \tag{1.2}$$

where lon_i and lat_i are longitude and latitude coordinates standardised by subtracting the mean [114], chl_i represents the Chlorophyll level, and ε_i is the noise parameter.

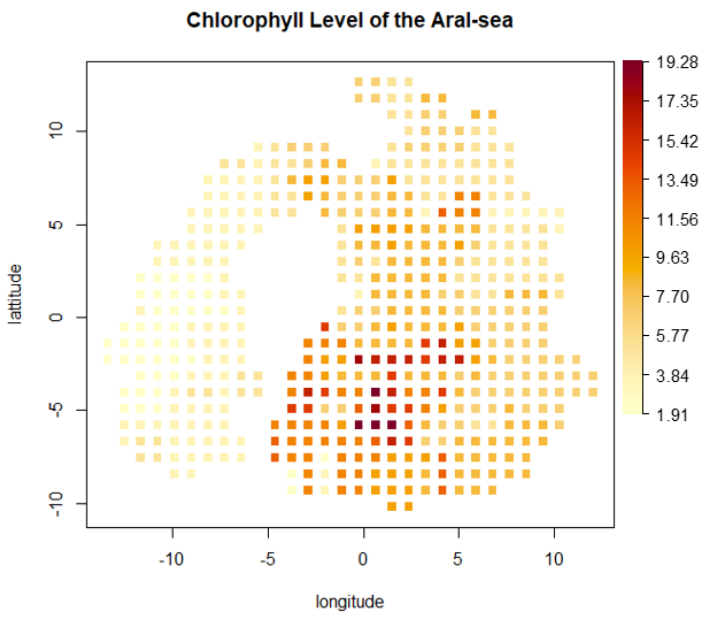


Figure 1.3: The chlorophyll levels of the Aral Sea shown as a colour map; the darker the colour, the higher the chlorophyll level.

The area with high chlorophyll level is located in the lower-central part on the left side of the Aral Sea. Separated by the gap corresponding to the isthmus of the peninsula, the overall chlorophyll level on the left side is relatively higher than on the right. Near the boundary of the gap despite the close Euclidean distance, the level on the left is significantly higher than on the right, varying smoothly within the boundary but not across the gap. Thus, considering the intrinsic geometric features when modelling on the Aral Sea is meaningful. Although real-world data is often more complex, potentially not adhering to the smoothness assumptions, and featuring various forms of noise, incompleteness, and nonlinear relationships, in the following research, the proposed methods demonstrate exceptional performance, outperforming some of the traditional methods used for comparisons.

The thesis is structured in chapters as described in the next section.

1.4 Outline of the Thesis

This thesis is divided into 8 chapters. A brief overview of each chapter and general structure of this thesis is shown below.

Chapter 2 provides the theoretical background for understanding GPs in Riemannian manifold, covering essential concepts from Riemannian geometry, Bayesian optimisation and graph theory. It explain how Bayesian optimisation works by giving an 1-dimensional example where GPs can be used as a surrogate model in Bayesian optimisation.

Chapter 3 presents the Intrinsic GPs for manifolds, beginning with a review of Traditional GPs and moving to intrinsic GPs that account for manifold geometry, including improvements in handling boundary conditions using BM.

Chapter 4 proposes Sparse Intrinsic GPs, which use sparse methods to address the computational issues associated with Intrinsic GPs. First, Sparse Intrinsic GPs with DIC is introduced. To correct the predictive variance of DIC, Sparse Intrinsic GPs with DTC is then presented. To further enhance prediction accuracy and stability, Sparse Intrinsic GPs with VI is introduced.

Chapter 5 explores Graph GPs, another approach of GPs on manifolds. It focuses on the use of the Graph Laplacian and Matérn kernels, adapted for graphs through its SPDE form, to model manifolds with complex boundaries.

Chapter 6 applies the proposed Sparse Intrinsic GPs to three representative examples introduced in Section 1.3. It begins by introducing the data analysis indicators used to evaluate and compare the performance of different GP methods. Following this, the chapter compares the performance of three Sparse Intrinsic GPs with Traditional GPs and Graph GPs, highlighting the strengths and weaknesses of each method.

Chapter 7 focuses on addressing optimisation problems on manifolds with complex boundaries, under budget constraints. Building on previous research, this chapter proposes five Bayesian Optimisation methods designed for manifolds. Then, it compares their effectiveness using the same examples discussed earlier, to evaluate how well each method performs in finding the optimal point.

Chapter 8 concludes the whole work in this thesis and highlights the main contribution. It also gives potential directions for future research.

Chapter 2

Theoretical Framework

In Chapter 1, the research motivation and thesis goals are outlined, providing the context and purpose of the study. The chapter also presents an overview of the case studies, followed by an outline of the thesis structure. In this thesis, the focus is on developing statistical methods, including GPs and BO, that are applicable to manifolds. This work not only extends the application of these methods beyond the limitations of Euclidean space but also provides crucial technical support for manifold-based research across various fields.

This chapter aims to provide a fundamental theoretical explanation for subsequent research. The chapter begins in Section 2.1 with an introduction to Riemannian geometry and its key concepts, with a particular focus on manifolds, which serve as the foundation for all subsequent research. Understanding these complex geometric structures is essential within this framework. Section 2.2 then presents the framework of Bayesian Optimisation (BO), which uses Gaussian Processes (GPs) as a surrogate model to provide information for the acquisition function at each iteration. GPs will be the focus of this research in the upcoming Chapters 3 and 4, where methodologies suitable for manifolds will be developed. In Chapter 6, these methods will be implemented and compared to evaluate their performance. Section 2.3 introduces graph theory, with a particular focus on the Graph Laplacian (GL), which will be used as another attempt to solve regression problems on manifolds, which will be further explored in Chapter 5.

2.1 Introduction to Riemannian Geometry

This section provides a broad introduction to Riemannian geometry and leads into the concept of manifolds, which are central to this study. Based on Gauss's intrinsic differential geometry of surfaces, the German mathematician G. F. B. Riemann proposed Riemannian geometry in the mid-19th century, it has evolved into an important and extensive field of study [9]. Riemannian geometry, as a branch of modern differential geometry, primarily investigates the geometric properties of Riemannian manifolds. Before introducing Riemannian manifolds, it is important to define some key notation fundamental to differential geometry. Let M denote the Riemannian manifold and C^∞ indicate smoothness (infinitely differentiable). At each point s on a Riemannian manifold M, there exists a tangent space which is denoted as T_sM. This tangent space can be regarded as a linear approximation of the Riemannian manifold at the point s. It is a standard vector space with its origin at the current point on the manifold, and the vectors are tangents to that point. Imagine a sphere, the vector space can be visualized as a sheet of stiff paper placed at some point s on the sphere, where all the vectors in the space can be drawn on this sheet. The tangent space T_sM is defined by equipping the manifold with an inner product, used to measure distances and angles between vectors. This inner product is given by a Riemannian metric tensor g, which can measure the angle and length between two tangent vectors.

A Riemannian manifold is then described as a smooth C^∞-manifold M (Hausdorff and second countable) with a Riemannian metric tensor g defined on each tangent space T_sM, as shown below [124], [51]:

Definition 2.1. *A **Riemannian metric** on a smooth manifold M is a symmetric positive definite smooth covariant 2-tensor field g. A smooth manifold M equipped with a Riemannian metric g is called a **Riemannian manifold**, and is denoted by (M,g).*

A compact Riemannian manifold M can be embedded in a larger, flat, Euclidean space with N dimensions, which can be explained by the Nash embedding theorems [112]:

Theory 2.1. *Any compact n-dimensional Riemannian manifold can be isometrically embedded in $\mathbb{R}^N$ for $N = \frac{n(3n+1)}{2}$.*

To explain it more intuitively, an isometric embedding means placing a curved space, such

as surface of a sphere, which is a 2-dimensional Riemannian manifold, into a flatter, higher-dimensional space, like an ordinary 3-dimensional space, without distorting distances. The theorem essentially provides a way to embed a compact Riemannian manifold in a larger Euclidean space, establishing a connection between Riemannian manifolds and Euclidean space. The presented theorem enables the analysis of complex manifold structures within the familiar Euclidean space, proposing new approaches to manifold research.

Suppose H represents a local open subset of the manifold M, with $H \subseteq M$, serving as the domain for the local chart. If $x : H \to \mathbb{R}^n$, the metric tensor g can be expressed as:

$$g = \sum_{i,j=1}^{n} g_{ij} dx^i \otimes dx^j,$$

in H, where

$$g_{ij} = g\left(\frac{\partial}{\partial x^i}, \frac{\partial}{\partial x^j}\right).$$

For example, $M = \mathbb{R}^n$, a euclidean n-space and $g = \sum_{i=1}^{n} dx^i \otimes dx^i$ define a Riemannian manifold, making Euclidean space a special type of manifold. The Riemannian metric tensor g is a symmetric, bilinear, positive definite function on M, which satisfies:

- $g(\mathbf{t}_1, \mathbf{t}_2)(s) = g(\mathbf{t}_2, \mathbf{t}_1)(s)$;
- $g(\mathbf{t}_1 + \mathbf{t}_2, \mathbf{t}_3)(s) = g(\mathbf{t}_1, \mathbf{t}_3) + g(\mathbf{t}_2, \mathbf{t}_3)(s)$;
- $g(\mathbf{t}_1, \mathbf{t}_1)(s) > 0$;

where $g(\mathbf{t}_1, \mathbf{t}_2)(s)$ is a smooth function of s from M to $\mathbb{R}$ and $\mathbf{t}_1, \mathbf{t}_2 \in T_s M$. The specific application and calculation of the metric tensor in this research can be found in Chapter 3 and Chapter 6. Given the metric tensor g, the Laplace-Beltrami operator Δ_s can be defined on the manifold M. It can be seen as a generalization of the Laplace operator in Euclidean space to Riemannian manifolds, depending on the metric structure of the manifold, which can be expressed as [117]:

$$\Delta_s f = \frac{1}{\sqrt{|g|}} \frac{\partial}{\partial x^i} \left(\sqrt{|g|} g^{ij} \frac{\partial f}{\partial x^j}\right),$$

where f is a smooth function defined on M, g^{ij} is the (i, j) element of its inverse, $|g|$ is the absolute value of the determinant of g and $\frac{\partial}{\partial x^i}$ represents the partial derivative with respect to

the coordinate x^i. It is a key operator in differential equations and physical systems on manifolds, describing the rate of change of a function on the manifold [69]. Chapter 5 explores how to implement GPs on manifolds by leveraging the graph structure when the Laplace-Beltrami operator Δ_s on M is unknown.

The geodesic distance $d(s_i, s_j)$ between two points s_i and s_j on the manifold M is defined as the infimum of the length of all smooth curves γ joining s_i and s_j, shown as:

$$d(p,q) = \inf_{\gamma} \int_0^1 \sqrt{g_{\gamma(t)}\left(\frac{d\gamma}{dt}, \frac{d\gamma}{dt}\right)} dt,$$

where $\gamma(t)$ is the parameterized arc, satisfying $\gamma(0) = s_i$ and $\gamma(1) = s_j$ [120].

The geodesic distance is easy to compute in special cases, such as for Euclidean space and sphere. However, for most complex manifolds, the computation can be extremely difficult. It often requires relying on other algorithms to study the geometric properties of the manifold. For example, the heat kernel can be used to explore the geometry of a manifold.

2.1.1 Heat Kernel

In Chapter 3, intrinsic GPs are developed by estimating the heat kernel on the manifold, as it captures the manifold's intrinsic geometric properties. The heat kernel is a fundamental concept in mathematics and physics, particularly in the study of partial differential equations (PDE). It represents the solution to the heat diffusion equation, a parabolic PDE [56]:

$$\partial_t u - \Delta_s u = 0, \tag{2.1}$$

where $\partial_t u$ shows the rate of change of heat over time, Δ_s is the Laplace-Beltrami operator defined before. The heat equation (2.1) expresses the relationship between the rate of change of the temperature field over time and the curvature of its spatial distribution in the absence of external heat sources. It provides valuable insights into the dynamics of heat transfer. For 1-dimensional Euclidean space $\mathbb{R}$, the fundamental solution to the heat equation (2.1) is shown as [38]:

Definition 2.2. *The function*

$$\Phi(x_1,\cdots,x_n,t)=\begin{cases}\frac{1}{(4\pi t)^{n/2}}e^{-\frac{|x|^2}{4t}} & (t>0)\\ 0 & (t\leq 0)\end{cases},$$

is called the fundamental solution of heat equation (2.1),

The process of deriving the fundamental solution of the heat equation (2.1) is provided in Appendix A.1. For n-dimensional Euclidean space $\mathbb{R}^n$, the heat kernel is the solution of the heat equation:

$$\frac{\partial K}{\partial t}(t,x,y)=\Delta_x K(t,x,y),$$

$$\lim_{t\to 0}K(t,x,y)=\delta(x-y)=\delta_x(y),$$

where time point $t>0$, $x,y\in\mathbf{R}^n$, Δ represents the Laplace operator defined before and δ is a Dirac delta function. Then, the heat kernel takes the form of a time-varying Gaussian function:

$$K(t,x,y)=\frac{1}{(4\pi t)^{n/2}}e^{-|x-y|^2/4t},$$

where $|x-y|$ means the euclidean distance between x and y. Figure 2.1 illustrates the heat kernel in 1-dimensional Euclidean space, with the initial point of heat diffusion at $x=0$. From Figure 2.1, it is easy to see that the heat diffusion is time-dependent. As time progresses, the peak of the heat kernel at the initial point $y=0$ decreases while the spread of the heat increases. At earlier times, such as $t=0.1$, $t=0.2$, the heat remains concentrated around the initial point. As t increases, the heat diffuses further outward, with the heat kernel flattening and extending over a larger range.

The above discussion of the heat equation does not take boundary conditions into account. From a thermodynamic perspective, if the temperature conditions (or heat exchange conditions) at the boundaries and the initial temperature conditions are known, the temperature of the medium at future moments can be determined. For manifolds with internal geometric structure and complex boundaries, considering boundary conditions is essential for calculating the heat kernel. In practice, the most common boundary conditions are as follows. For one-dimensional cases, specific expressions are given on the interval ($\mathscr{I}=(0,l)$) [148]:

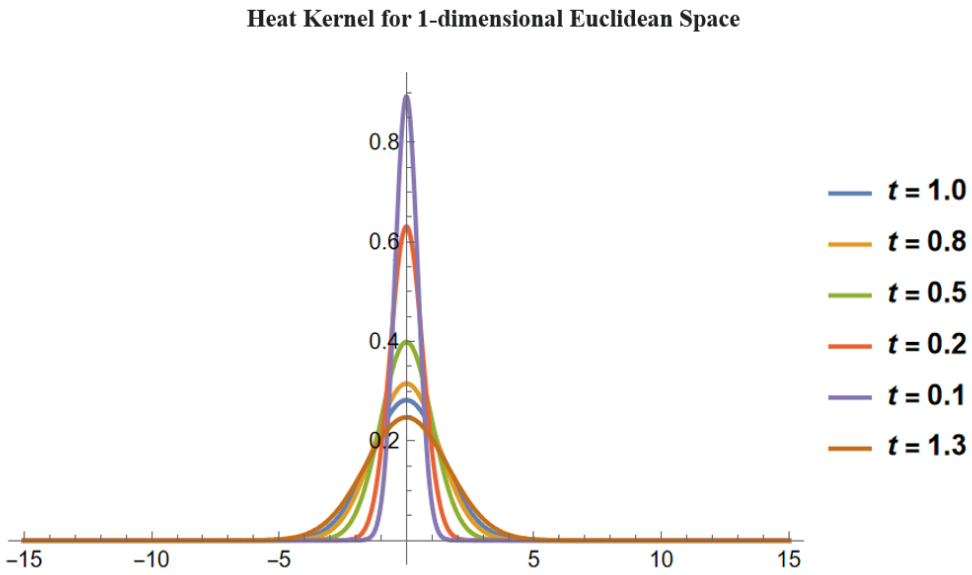


Figure 2.1: The different coloured lines represent the heat kernel in 1-dimensional Euclidean space at different moments t. The heat spreads out to both directions from initial position 0, without boundary limitation.

- The first-type boundary condition: the temperature of the medium at the boundary is known, also called Dirichlet boundary condition; the formula for $(\mathscr{I} = (0,l))$ is: $u(0,t) = c_1, u(l,t) = c_2$, where c_1 and c_2 specify the fixed temperature values at the boundaries.

- The second-type boundary condition: the temperature of the medium is unknown at the boundaries, but the normal derivative of temperature of the medium at the boundaries is known, also called Neumann boundary condition; the formula for $(\mathscr{I} = (0,l))$ is: $u_x(0,t) = c_1, u_x(l,t) = c_2$, where c_1 and c_2 define the normal derivative of temperature at the boundaries.

- The third-type boundary condition: the temperature of the medium at the boundary and the normal derivative of temperature of the medium at the boundaries satisfy certain relationship equations, also called Robin boundary condition; the formula for $(\mathscr{I} = (0,l))$ is: $u_x(0,t) - a_0 u(0,t) = c_1, u_x(l,t) - a_l u(l,t) = c_2$, where c_1 and c_2 are constants that appear in a linear combination of temperature and its derivative.

In this research, Neumann boundary conditions are adopted, ensuring that there is no heat exchange across the boundary by setting the constant term in the boundary condition to zero. The

first boundary condition is the simplest which directly states that the temperature remains constant at the boundary of the manifold M. The second boundary condition does not consider the specific temperature at the boundary, but the heat flow over the boundary, which specifies the amount of heat flowing at the boundary. It means how much heat passes through the boundary of the manifold per unit of time. The third boundary condition combines the first and second boundary conditions, which are affected by the temperature of the surroundings and consider the relationship between the temperature and the heat flow at the boundary. The heat equation with boundary and initial conditions is not easy to solve. For most manifolds with complex boundaries, finding an analytical solution to the heat equation is usually impractical. Therefore, numerical and approximation methods are often employed to solve it. Lawler [85] discusses the probabilistic connections between Brownian motion (BM) and the heat equation. The next section will introduce and explore BM in more detail.

2.1.2 Brownian motion

BM is a common natural phenomenon that describes the continuous, random movement of suspended small molecular objects in mediums such as liquids and air. It is utilized to provide an estimate for the heat kernel of manifold M in section 3.3. BM has continuous time parameters and continuous state space, and is closely linked to the normal distribution. As an important fundamental theory in the discipline of stochastic processes, it is widely used in physics, economics and other academic fields [50], [106], [116]. The BM can be defined as [111]:

Definition 2.3. *A real-valued stochastic process* $\{B(t) : t \geq 0\}$ *is called a (linear) BM starting at* $x \in \mathbb{R}$ *if the following holds:*

- $B(0) = x$,
- *the process has independent increments, i.e. for all times* $0 \leq t_1 \leq t_2 \leq ... \leq t_n$ *the increments* $B(t_n) - B(t_{n-1}), B(t_{n-1}) - B(t_{n-2}), \ldots, B(t_2) - B(t_1)$ *are independent random variables,*
- *for all* $t \geq 0$ *and* $h > 0$*, the increments* $B(t+h) - B(t)$ *are normally distributed with expectation zero and variance h,*

- *almost surely, the function $t \mapsto B(t)$ is continuous.*

If $x = 0$, $\{B(t) : t \geq 0\}$ is a standard BM. High-dimensional BM satisfies the following definition:

Definition 2.4. *Given that $B_1(t), B_2(t), ..., B_d(t)$, represent mutually independent standard BMs, $B(t) = (B_1(t), B_2(t), ..., B_d(t))$ is referred to as a d-dimensional BM.*

In other words, the projection of d-dimensional BM onto the spaces $\mathbb{R}, \mathbb{R}^2, ..., \mathbb{R}^{d-1}$ is also a BM. Figure 2.2 presents 5 BM paths starting from the same initial point $(0,0)$ in a 2-dimensional space, displayed in different colors. Each path represents an independent realization of Brownian motion, showcasing the random, unpredictable nature of the movement. Over time, the paths spread out in various directions. It effectively visualizes the stochastic nature of BM, where no two paths are the same despite originating from the same point. BM has many properties. For example, BM has the Markov property, i.e., given the current state, the future state does not depend on the past state. Also, BM demonstrates the Martingale Property, whereby at any given time point, the current value of the BM has the same expected value as its future values. Let $B(t)$ denote the location of the BM at time t. For any $s \leq t$, the Martingale Property can be expressed as:

$$\mathbb{E}[B(t) \,|\, \mathscr{F}_s] = B(s), \tag{2.2}$$

where $\mathscr{F}_s$ represents the information available at time s. The martingale property describes that the current value is the best predictor of future values. This means that at time s, the conditional expectation of the future value of the BM is equal to the current value $B(s)$. The detailed proof for the martingale property is shown in Appendix B.

This research aims to explore regression and optimisation problems on Riemannian manifolds. This section introduces some fundamental concepts related to manifolds, while the next section will provide an overview of commonly used statistical methods for regression and optimisation, including GPs and BO, where GPs can be used as surrogate models in BO.

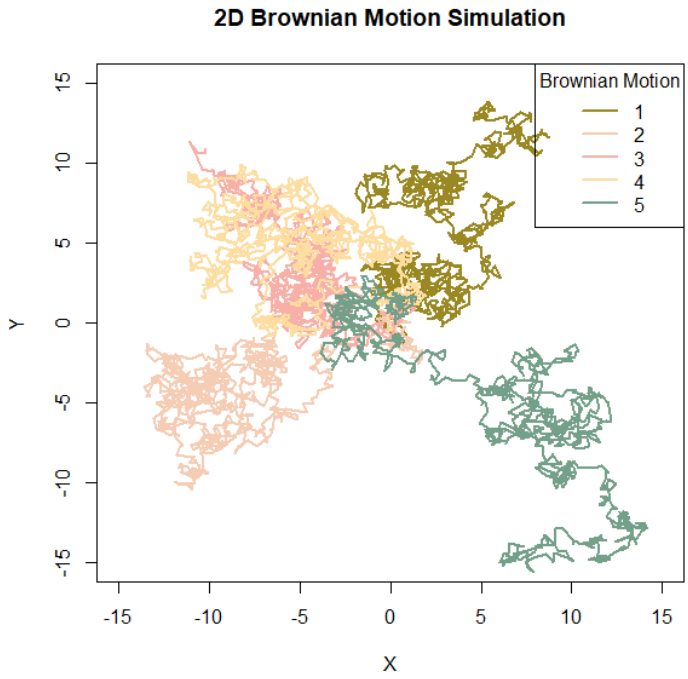


Figure 2.2: Five different BM paths in a 2-dimensional space, each represented by a different color; All paths originate from the same initial point at $(0,0)$ and spread out in various directions.

2.2 The Framework of Bayesian Optimisation

BO is an empirical global optimisation method used for optimising black-box functions that are costly and potentially noisy to evaluate. Motivated by the successful implementation of BO in Euclidean spaces, Chapter 7 will build on the previous research on GPs to extend the application of BO to manifolds, thus addressing optimisation problems on manifolds.

BO aims to iteratively search for the optimal points within the domain to be explored, based on the existing information. These optimal points can either be maxima or minima. The core problem in BO is determining how to select the next data point for evaluation based on the currently available information. Firstly, BO needs to consider how to make global predictions based on the limited available information. Secondly, once a global prediction is made, it must determine how to select the next point for exploration to update the current information set, thereby iteratively moving towards the optimal point. The BO framework consists of two key components: a probabilistic surrogate model and an acquisition function. The surrogate model is used to approximate the objective function, which cannot be explicitly or directly defined—in other

words, a clear mathematical formulation is not feasible. The acquisition function then guides the search for the next point to evaluate. The probabilistic surrogate model typically comprises a prior probability model $p(f)$ and an observation model. Let f represent the unknown objective function. The observation model describes the mechanism of observation data generation, that is, the likelihood distribution $p(\mathscr{D}_{1:i} \mid f)$, where $\mathscr{D}_{1:i} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_i, y_i)\}$ describes the observed data set. Updating the probabilistic proxy model means the posterior probability distribution $p(f \mid \mathscr{D}_{1:i})$ containing more data information; in other words, (x_{i+1}, y_{i+1}) has been added into dataset $\mathscr{D}_{1:i+1}$, which is used to calculate the posterior probability distribution $p(f \mid \mathscr{D}_{1:i+1})$. The process of updating the probabilistic proxy model is precisely Bayesian Inference.

Bayesian Inference

Bayesian inference is a method of inferential statistics based on Bayes' theorem. Bayesian inference calculates posterior probabilities according to Bayes' theorem:

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)},$$

where $P(A)$ is an initial judgment of the probability of a hypothesis based on existing knowledge prior to observing the new data; $P(B|A)$ is the probability of observing the new data under a given hypothesis and measures the consistency of the data with the hypothesis; $P(B)$ is the marginal probability, representing the total probability of occurrence of the observed new data. Bayesian inference continuously updates the posterior probability by incorporating new data points, improving and correcting previous hypotheses in an iterative learning process.

The acquisition function is constructed based on the posterior probability distribution and selects the most "promising" point through a certain evaluation mechanism to update the dataset $\mathscr{D}$. Figure 2.3 shows the framework of BO, where the iteration continues until the termination condition is reached. The iterations of BO typically have several common termination conditions, such as a predefined number of iterations. A maximum number of iterations is set, and once this number is reached, the algorithm stops. Another condition is a time limit, where BO iterations must be completed within a certain time frame. Additionally, the convergence of

the objective function value can be a termination condition. If the optimal value found by BO shows very little change over several iterations (below a certain threshold), the iterations stop. This research considers the budget constraint that often exists in real-world problem-solving, particularly when the cost of each sample is very high. Therefore, in each case, the number of iterations is predefined based on the budget constraint. For both the surrogate model and acquisition function, there are various choices available. This research primarily uses GPs as the surrogate model. Building on the widespread application of GPs in Euclidean spaces, this work aims to develop GPs on manifolds that can be applied to higher-dimensional manifolds while maintaining feasible computational costs, developing from Niu et al. [114]. The next section will provide a basic introduction to GPs in Euclidean space.

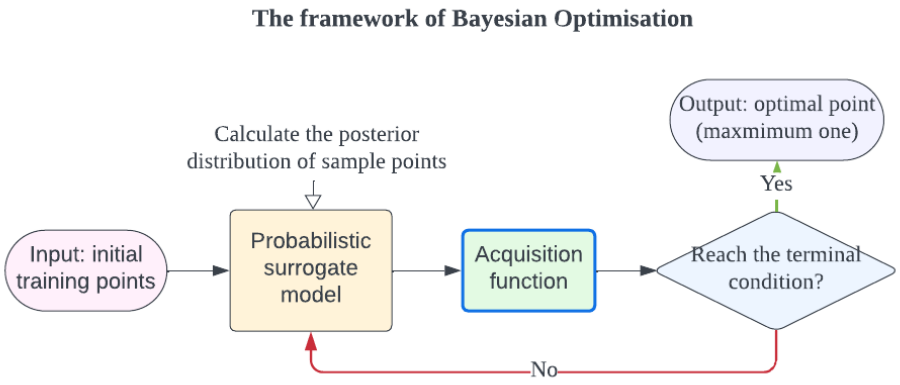


Figure 2.3: The flowchart illustrates the framework of BO; the process begins with inputting the initial training points and updating the probabilistic surrogate model by selecting the new observation point through the acquisition function; this iterative process continues until the termination condition is met.

2.2.1 Exploring Probabilistic Surrogate Models Focusing on Gaussian Processes

The common probabilistic surrogate models can be categorised into parametric models and non-parametric models as introduced in Chapter 1. This section will focus on GPs on Euclidean space, which are widely used due to the flexible prior distributions, effective uncertainty estimation, and efficient prediction. Meanwhile, GPs require less data than other non-parametric

models [135].

In statistics, a GP is a random process in which observations appear in a continuous domain (such as time or space). Each random variable within this domain follows a Gaussian distribution. The distribution of a GP is the joint distribution of all these (infinitely many) random variables. In other words, a GP is the normalization of the multivariate Gaussian probability distribution [165]. Let $\mathscr{X}$ be the input space, a Euclidean space in this section. A Gaussian process is specified by its mean function $\mu : \mathscr{X} \to \mathbb{R}$ and a positive semi-definite covariance function $K : \mathscr{X} \times \mathscr{X} \to \mathbb{R}$, where the diagonal elements represent the variance function σ. A valid covariance function must be positive semi-definite. The function $f(x)$ which is distributed as a GP with mean function μ and covariance function K can be written as:

$$f(x) \sim GP\left(\mu(x), K\left(x, x'\right)\right),$$

where the mean function is $\mu(x) = \mathbb{E}[f(x)]$ and $K(x,x') = \mathbb{E}\left[\left(f(x) - \mu(x)\right)\left(f\left(x'\right) - \mu\left(x'\right)\right)\right]$ represents the covariance function. This can be considered the prior in Bayesian inference. The focus then shifts to the posterior for making predictions at unknown test points. Using GPs to model the distribution of known data and make predictions for unknown data is also known as GP Regression. Let f represent the function values at known training points x from $\mathscr{X}$, and f^* denote the function values at the unknown testing points x^* from $\mathscr{X}$ which are unknown. According to the characteristics of the GP, the joint distribution between training objectives f and testing objectives f^* is as follows:

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}^* \end{bmatrix} \sim \mathscr{N}\left(\begin{bmatrix} \mu \\ \mu^* \end{bmatrix}, \begin{bmatrix} K(x,x) & K\left(x,x^*\right) \\ K\left(x^*,x\right) & K\left(x^*,x^*\right) \end{bmatrix}\right),$$

where μ is the mean function of training points while μ^* is the mean function of testing points. According to the conditional distribution properties of the multidimensional Gaussian distribution, the predictive distribution for GP regression can be obtained [165]:

$$\mathbf{f}^* \mid \mathbf{f} \sim \mathscr{N}\left(K\left(x^*,x\right)K(x,x)^{-1}\mathbf{f},\ K\left(x^*,x^*\right) - K\left(x^*,x\right)K(x,x)^{-1}K\left(x,x^*\right)\right).$$

The posterior mean and variance evaluated at any point from $\mathscr{X}$ represents the model's predic-

tion and uncertainty. The covariance function K, also known as the kernel function, is calculated to describe the relationship between different points in the input space, which is crucial in determining both the smoothness and structure of the predictions in GP regression. There are many kernel types that can be applied in Euclidean space, and a few of the more common ones are described below.

Common Kernel Functions

In the practical application of GP regression, selecting an appropriate covariance function $K(x,x)$ is essential for achieving good predictive performance. The covariance function defines the relationships between points during modelling, influencing how similarities between data points are measured. This directly impacts both prediction accuracy and uncertainty estimation. Commonly used kernel functions include the Matérn kernel and radial basis function (RBF) kernel, among others. The Matérn kernel cluster is a class of highly flexible covariance functions [165]. The specific function expressions are as follows:

$$k_\nu(x,y) = \sigma_n^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{\|x-y\|}{\kappa} \right)^\nu K_\nu \left(\sqrt{2\nu} \frac{\|x-y\|}{\kappa} \right), \tag{2.3}$$

among it, $\|x-y\|$ means the distance between x and y, κ controls the degree of dependency between neighbouring data points, Γ is Gamma function, ν is non-negative, controlling the smoothness of the function and σ_n^2 controls variability of the GP [165]. K_ν is the second kind of deformed Bessel function [52]. The Matérn kernel is parameterized by a smoothness parameter ν, which typically takes fixed values depending on the desired level of smoothness, such as $\frac{3}{2}$, $\frac{5}{2}$. As $\nu \to \infty$, the Matérn kernel converges to the RBF kernel, which produces infinitely smooth functions. The RBF kernel, also called the squared exponential kernel, can be expressed as:

$$k(x,y) = \alpha \exp\left(-\frac{\|x-y\|^2}{2\ell^2} \right),$$

which has two parameters, length-scale parameter l and variance parameter α. α intuitively affects the value of the kernel, the larger the α, the greater the fluctuation of the kernel. For l, the smaller the l, the higher the frequency of kernel fluctuations. The RBF kernel is Euclidean distance-based, producing significant similarity measures only for points closing to each other,

while the similarity rapidly decays for points that are farther apart. It is one of the most popular choices in GP modeling because it results in a smooth prior over the functions sampled from the GP. In this research, the Traditional GPs introduced in Chapter 3 adopt the RBF kernel, applying GPs from Euclidean space directly to manifolds, primarily as a baseline comparison, serving as a Euclidean distance-based reference rather than a focus of the study.

After introducing GPs as the surrogate model, the next section presents various acquisition functions used in BO.

2.2.2 Various types of Acquisition Functions

"Based on what we know so far, which point should we evaluate next?"

Due to the often high cost of sampling in real-life scenarios, such as drilling for oil in the desert, the "next point" to be explored must be chosen carefully. The acquisition function helps make this decision, which serves as a heuristic method to evaluate a point based on the current model. The acquisition function can be expressed as $\varsigma : \mathscr{X} \times \mathbb{R} \times \Theta \to \mathbb{R}$ mapping from the input space $\mathscr{X}$, the observation space $\mathbb{R}$, and the hyperparameter space Θ to the real number space. This function is constructed from the posterior distribution obtained from the observed data set $\mathscr{D}_{1:i}$, and guides the selection of the next sample point x_{i+1} by maximizing it:

$$x_{i+1} = \max_{x \in \chi} \varsigma_i \left(x; \mathscr{D}_{1:i}\right).$$

Commonly used acquisition functions can be divided into three categories: improvement-based strategies, optimistic Policies and information-based strategies [135].

Improvement-Based Strategy

The improvement-based strategy selects points that are expected to improve the current optimal objective function value, where improvement refers to values smaller or larger than the current objective, depending on the optimisation goal. Both the Expected Improvement (EI) and Probability of Improvement (PI) functions are commonly used in this approach. The PI function is

expressed as:

$$PI(x) = \varphi\left(\frac{\mu(x) - f(x^+) - \varepsilon}{\sigma(x)}\right), \tag{2.4}$$

among it, $x^+ = \text{argmax}_{x \in x_{1:i}} f(x)$ is the location of the current optimal value among the observed data set $\mathscr{D}_{1:i}$; $\mu(x)$ is the mean function while σ is the variance function in this iteration obtained from the surrogate model. ε regulates the trade-off between exploitation and exploration in BO and can be empirically tuned to determine the most suitable value for a given problem. PI quantifies the probability that the observed value of x may improve the current optimal objective function value [82]. Based on PI, Močkus et al. [108] proposed a new improvement-based strategy EI, which can be expressed as:

$$\begin{aligned} EI(x) &= \begin{cases} (\mu(x) - f(x^+) - \varepsilon)\,\varphi(Z) + \sigma(x)\psi(Z), & \text{if } \sigma(x) > 0 \\ 0, & \text{if } \sigma(x) = 0 \end{cases}, \\ Z &= \frac{\mu(x) - f(x^+) - \varepsilon}{\sigma(x)}, \end{aligned}$$

among them, $\varphi(\cdot)$ means cumulative distribution function (CDF) and $\psi(\cdot)$ means probability distribution function (PDF). Similar to PI, the amount of exploration of the EI can be adjusted by modifying ε. The x selected by EI not only integrates the improvement probability but also reflects the different improvement amount.

Optimistic Policies

Srinivas et al. [139] propose a confidence boundary strategy for GPs known as GP-UCB (Upper Confidence Bound). The UCB strategy focuses on maximising the objective function's upper confidence bound, which can be viewed as a weighted sum of the expected performance captured by $\mu(x)$ and of the uncertainty $\sigma(x)$. The acquisition function for the UCB strategy is given by:

$$UCB(x) = \mu(x) + \sqrt{\beta}\sigma(x),$$

where β balances the mean $\mu(x)$ and variance $\sigma(x)$. It is a dynamically adjusted parameter that controls the balance between exploration and exploitation. When BO is trying to find the minimum value of the objective function, the lower confidence bound (LCB) serves as the cor-

responding acquisition function. GP-LCB can be expressed as:

$$LCB(x) = -(\mu(x) - \sqrt{\beta}\sigma(x)),$$

where β plays the same role as in the UCB strategy and can be empirically tuned to determine the optimal value.

Information-Based Strategies

Different from the previous two categories, information-based strategies sample reward functions from the posterior distribution $p(f \mid \mathscr{D}_{1:i})$ of the surrogate function and optimise them. A notable example is Thompson sampling, first introduced by William R. Thompson [149]. Thompson sampling draws a random function from the posterior distribution of the surrogate model, representing a plausible version of the unknown objective function, and selects the point x_{i+1} that maximises this sampled function. In doing so, the algorithm treats the sampled function as if it were the true objective function. This process can be viewed as:

$$x_{i+1} = \underset{x \in X}{\operatorname{argmax}}\ \hat{f}(x),$$

where X is the input space and $\hat{f}(x)$ is the sampled function.

The next section will provide a simple example of BO in a one-dimensional case to help illustrate the process of BO.

2.2.3 Example for 1-Dimensional BO

When applying BO to practical problems, selecting an appropriate probabilistic surrogate model and acquisition function is crucial. In a simple one-dimensional Euclidean space, the GP serves as an excellent example of a surrogate model and is a key focus of this research. In this example, $f(x) = x^2 \sin^6(5\pi x)$, the surrogate model GP uses the RBF kernel, which is sufficient to capture the relationships between points. PI is used as the acquisition function, prioritising points with a high probability of improving the current optimal value. Figure 2.4 presents the BO process

by using the GPyOpt package in Python [5]. GPyOpt is a Python open-source library for BO developed by the Machine Learning group of the University of Sheffield [5]. The left graph in Figure 2.4 shows the plot of objective function $f(x)$, while the right graph shows the last iteration of the optimisation process, with the number of iterations limited to 50 steps. The red line represents the PI acquisition function, where the highest point corresponds to the selected exploration point for the next iteration. The optimisation process successfully identifies the global optimum. The predictive mean, represented by the black line, performs well and closely approximates the target function, shown by the blue line. After introducing the framework of

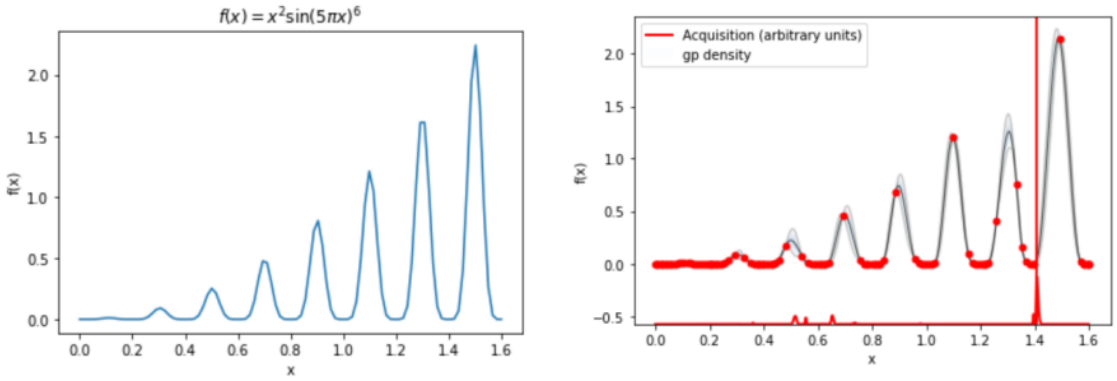


Figure 2.4: 1-Dimensional BO: The left graph shows the plot of the objective function $f(x) = x^2 \sin^6(5\pi x)$, while the right graph illustrates the BO process. The red points represent both the initial sampled points and those selected during optimisation. The red line indicates the acquisition function (PI), the black line represents the predictive mean, and the blue line shows the target function.

BO in Euclidean space, the next section will introduce the graph theory, which serves as the foundation for Chapter 5 and offers a new perspective for studying GPs on manifolds.

2.3 Introduction to Graph Theory

Graph theory is a branch of discrete mathematics that primarily studies graphs. A graph is a mathematical structure used to model pairwise relationships between objects, consisting of "nodes" and the "edges" that connect them. The combination of nodes and edges appears in various fields and is prevalent in real-world applications. For instance, in city planning, map drawing, and GPS navigation, road networks are often represented as graphs [57]. Social net-

works can also be modeled using graph theory [101], [59]. Additionally, infrastructure like power grids, water systems, and railways can be represented using graphs [68]. A graph can be defined as [175]:

Definition 2.5. *A graph $G = (V,E)$ is a mathematical structure consisting of two sets V and E. The elements of V are called vertices, and the elements of E are called edges. Each edge has a set of one or two vertices associated to it, which are called its endpoints.*

To illustrate, Figure 2.5 presents a simple two-dimensional undirected graph $G_a = (V_a, E_a)$, where $V_a = A,B,C,D,E$ and $E_a = a,b,c,d,e,f,g,h$. The structure of a graph can also be repre-

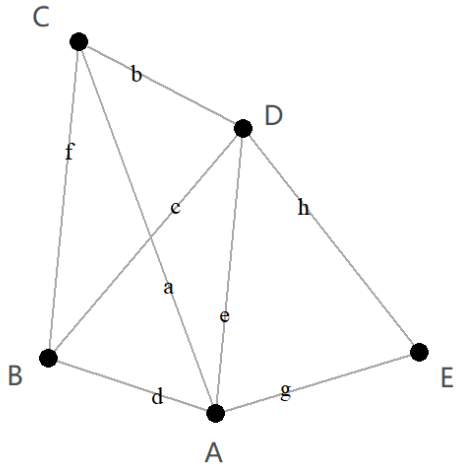


Figure 2.5: The undirected graph $G_a = (V_a, E_a)$ contains 5 vertices V_a and 8 edges E_a.

sented by an adjacency matrix, a binary matrix that indicates the presence or absence of edges between pairs of vertices. The definition of the adjacency matrix is as follows [13]:

Definition 2.6. *The adjacency matrix of graph G is the $n \times n$ matrix W whose entries w_{ij} are given by*

$$w_{ij} = \begin{cases} 1 & \text{if } v_i \text{ and } v_j \text{ are adjacent;} \\ 0 & \text{otherwise.} \end{cases}$$

The adjacency matrix is symmetric when the graph is undirected. For Figure 2.5, the adja-

cency matrix W_{G_a} is:

$$W_{G_a} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

2.3.1 Graph Laplacian

The Graph Laplacian (GL) matrix L, derived from the adjacency matrix W, provides a deeper representation of the graph's structural properties and information flow. It not only expresses the connections between nodes but also incorporates the degree of each node, effectively capturing the overall topology of the graph. Here, the most basic form of the GL is provided:

$$L(G) = D(G) - W(G), \tag{2.5}$$

where $D(G)$ is an $n \times n$ diagonal matrix, with each diagonal element equal to the sum of the elements in the corresponding row (or column) of the adjacency matrix W for that vertex. This matrix is called the degree matrix of graph G. For Figure 2.5, the GL matrix $L(G_a)$ is:

$$L(G_a) = D(G_a) - W(G_a) = \begin{bmatrix} 4 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix} - \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 4 & -1 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 & 0 \\ -1 & -1 & 3 & -1 & 0 \\ -1 & -1 & -1 & 4 & -1 \\ -1 & 0 & 0 & -1 & 2 \end{bmatrix}.$$

Some basic properties of the GL matrix are summarized as follows:

- $L(G)$ is symmetric, meaning that $L = L^\top$ and is positive semidefinite, meaning that for any non-zero vector $\mathbf{f}$, $\mathbf{f}^\top L\mathbf{f} \geq 0$;
- Since $L(G)$ is a symmetric matrix, all the eigenvalues λ are real, and both the eigenvalues and eigenvectors can be computed through the standard matrix diagonalisation process;

- The rank of $L(G)$ is $n-k$, where n is the number of vertices and k is the number of connected components of G. This means that the number of zero eigenvalues of $L(G)$ corresponds to the number of disconnected components in the graph;

- $L(G)$ can be expressed in the quadratic form:

$$\mathbf{f}^\top L\mathbf{f} = \sum_{i,j=1}^{n} \frac{1}{2} w_{ij}(f_i - f_j)^2,$$

where $\mathbf{f} \in \mathbb{R}^{n\times 1}$ is a vector of real values associated with the graph vertices, w_{ij} is the element of the adjacency matrix W and n represents the number of vertices. The proof follows from Equation 2.5:

$$\begin{aligned}
\mathbf{f}^\top L\mathbf{f} &= \mathbf{f}^\top (D-W)\mathbf{f} = \mathbf{f}^\top D\mathbf{f} - \mathbf{f}^\top W\mathbf{f} \\
&= \sum_{i=1}^{n} d_i f_i^2 - \sum_{i,j=1}^{n} w_{ij} f_i f_j \\
&= \frac{1}{2}\left(\sum_{i=1}^{n} d_i f_i^2 + \sum_{j=1}^{n} d_j f_j^2 - 2\sum_{i,j=1}^{n} w_{ij} f_i f_j \right) \\
&= \frac{1}{2}\left(\sum_{i=1}^{n} \left(\sum_{j=1}^{n} w_{ij} \right) f_i^2 + \sum_{j=1}^{n} \left(\sum_{i=1}^{n} w_{ij} \right) f_j^2 - 2\sum_{i,j=1}^{n} w_{ij} f_i f_j \right) \\
&= \sum_{i,j=1}^{n} \frac{1}{2} w_{ij}(f_i - f_j)^2.
\end{aligned}$$

The GL, as a key subject of study in graph theory, is widely applied across various fields, such as spectral clustering and spectral embedding in machine learning [28], and graph convolution in signal processing [47] and deep learning [12]. The introduction in this section serves as a foundation for the research presented in Chapter 5. The GPs method introduced in Chapter 5 for manifolds is built by constructing a graph on the manifold, enabling the use of the GL to analyse the structural properties of the manifold.

2.4 Conclusion

The purpose of this chapter is to help understand some terminology and basic concepts mentioned in subsequent chapters. Among these, the manifold in Riemannian geometry is the primary subject of study. This work aims to design GPs on manifolds, considering their intrinsic geometric properties (discussed in Chapters 3 and 4). Additionally, developing novel BO methods for manifolds addresses optimisation challenges (explored in Chapter 7). The graph theory provides the foundational framework for an alternative approach to GPs on manifolds introduced in Chapter 5. The next chapter will begin by introducing traditional GPs applied directly to manifolds. Then, based on the work of Niu et al. [114], Chapter 3 will introduce Intrinsic GPs and highlight the novel "reflection" method proposed to improve the accuracy of heat kernel estimation near the boundary.

Chapter 3

Intrinsic Gaussian Processes for Manifolds

This chapter aims to establish a framework for addressing the regression problem on manifolds. In recent years, an increasing amount of research has focused on manifolds, driven by their applicability in diverse fields such as geometry, environment, and medical imaging. Unlike Euclidean spaces, manifolds have complex intrinsic properties and boundary conditions, which make many research approaches that work well in Euclidean spaces not directly applicable to manifolds. This research aims to propose a model that can effectively measure the characteristics of manifolds, laying a foundation for more in-depth research. Inspired by the rapid development of Traditional GPs in Euclidean spaces, this chapter explores GPs on manifolds, Section 3.1 gives the brief introduction of this chapter. Section 3.2 introduces the simplest approach, namely the direct application of Traditional GPs on manifolds, which rely solely on Euclidean distance for regression. Then, Section 3.3 presents Intrinsic GPs specifically designed for manifolds, taking into account their intrinsic geometric structure, based on the work of Niu et al. [114]. Building on previous research, Section 3.4 introduces a novel "reflection" approach designed to improve the accuracy of the estimated heat kernel at boundaries through BM in Intrinsic GPs. Section 3.5 provides a summary of this chapter.

3.1 Introduction

Compared to simple geometries, manifolds usually exhibit more complex intrinsic geometric characteristics such as topology, connectivity and smoothness, as well as complex boundaries. These features tend to affect the modeling and prediction of manifolds, and hence the decision making in practical engineering fields. How to improve the consideration of the intrinsic geometric features of manifolds in the modeling and optimisation process is a major challenge in the study of manifolds. Chapter 1 details various methods and approaches that attempt to capture these geometric complexities. Given the wide application of GPs in Euclidean spaces, this research considers extending the application of GPs to the realm of manifolds.

In Euclidean spaces, GPs commonly use kernels such as the RBF kernel, as introduced in Chapter 2. The RBF kernel has demonstrated significant effectiveness across various kernel-based algorithms within Euclidean spaces. The RBF kernel maps data points from the original Euclidean space to an infinite-dimensional Hilbert space through the kernel function, where the structure of the data can be better represented and processed. However, the principle of the RBF kernel is based on Euclidean distance, which fails to recognize the intrinsic geometric features of manifolds, such as the complex boundaries of lakes. The next section will introduce Traditional GPs, applied in the same manner on both Euclidean space and manifolds to motivate the challenges arising in applying GPs on manifolds. Traditional GPs also serve as a baseline for comparison, validating the effectiveness of the methods proposed later in this work. Section 3.3 will introduce Intrinsic GPs designed specifically for manifolds [114], which use an approximate heat kernel to capture the manifold's structure. The term "approximate" is used here because, for most manifolds, an explicit analytical expression for the heat kernel typically does not exist. The complex geometric structure and topological properties of manifolds often make the heat kernel difficult to derive. Therefore, proposing a general "approximate" approach is meaningful. Intrinsic GPs will use the transition density of BM paths to approximate the heat kernel. The BM paths, through random walks within the manifold boundaries, provide information about the geometric features. However, due to the "resample" mechanism used at the boundaries, the information near the boundaries is often inaccurate. Section 3.4 proposes a novel "reflection" approach to address this issue in Intrinsic GPs, enhancing the accuracy of regression predictions, particularly near the boundaries.

3.2 Traditional Gaussian Processes

This section illustrates the Traditional GPs used on manifolds, which directly follow the same approach as in Euclidean spaces. As defined in previous chapters, let M be a d-dimensional complete and orientable Riemannian manifold and ∂M denote M's boundary which is continuous and C^1 almost everywhere. $S = \{s_i, i = 1, \ldots, G'\}$ is the grid points set defined on the manifold M, with G' the number of grid points, $s_i \in M$. $\mathbf{f}_r$ represents the vector of $f(.)$ at all grid points. Selecting $\mathscr{D}$ as training points set from grid points set and $\mathbf{f}_{\mathscr{D}}$ as the vector of $f(.)$ at training points set, y can be defined as the observation value of the objective function to $\mathscr{D}$: $y = \mathbf{f}_{\mathscr{D}} + \varepsilon$, where $\varepsilon \sim \mathscr{N}\left(0, \sigma_n^2\right)$ and σ_n^2 means the variance of the noise. The number of training points is chosen based on the scale of the manifold. The aim is to infer the unknown information $\mathbf{f}_r$ based on the given information y corresponding to the observed value of $\mathscr{D}$. In GPs, the grid points set S is also referred to as testing points. The joint distribution of the unknown $\mathbf{f}_r$ and the observed values y can be calculated as:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_r \end{bmatrix} \sim \mathscr{N}\left(0, \begin{bmatrix} \Sigma_{\mathscr{D}\mathscr{D}} + \sigma_n^2 I & \Sigma_{\mathscr{D}r} \\ \Sigma_{r\mathscr{D}} & \Sigma_{rr} \end{bmatrix}\right), \tag{3.1}$$

where $\Sigma_{\mathscr{D}r}$ is the covariance matrix for training data points $\mathscr{D}$ and test data points S. The covariance matrix for the joint distribution's entries can be computed using the equation: $\Sigma_{\mathscr{D}r}(i, j) = K(s_i, s_j)$, where $K(s_i, s_j)$ is the RBF kernel used to calculate the similarity between two points in this study. It can be expressed as:

$$K(s_i, s_j) = \alpha \exp\left(-\frac{\|s_i - s_j\|^2}{2\ell^2}\right) \tag{3.2}$$

where $\|s_i - s_j\|^2$ represents the squared Euclidean distance between the two data points. It's easy to see that the computation of the RBF kernel relies on the Euclidean distance. If the Euclidean distance does not accurately approximate the manifold's intrinsic distance, for instance, if it deviates significantly due to the manifold's geometric characteristics, it will fail to provide a reliable estimate of the manifold. The variance parameter, denoted as α, and the length-scale parameter, denoted as l, are critical kernel parameters. These parameters can be optimised by

maximizing the log marginal likelihood function,

$$\log p(Y \mid X, \theta) = -\frac{1}{2} Y^T \left(\Sigma + \sigma_n^2 I\right)^{-1} Y - \frac{1}{2} \log \left|\Sigma + \sigma_n^2 I\right| - \frac{n}{2} \log 2\pi, \tag{3.3}$$

where $\Sigma = K(s_i, s_j)$ and σ_n^2 denotes noise variance. θ represents the hyperparameters to be optimised, including α, l and σ_n^2. Knowing how to calculate the RBF kernel, the next step is to derive the predictive distribution based on Equation 3.1. The posterior predictive distribution of the unknown values $\mathbf{f}_r$ (at test points S) given the observed data y can be expressed as:

$$p\left(\mathbf{f}_r \mid y\right) = \mathcal{N}\left(\Sigma_{r\mathscr{D}} \left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_n^2 I\right)^{-1} y, \quad \Sigma_{rr} - \left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_n^2 I\right)^{-1} \Sigma_{\mathscr{D}r}\right). \tag{3.4}$$

The predictive mean and variance over all grid points on M can be calculated by Equation 3.4. The inverse term $\left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_n^2 I\right)^{-1}$ plays a critical role in GP predictions by updating the posterior distribution in GPs, with the computational complexity $O(n^3)$.

Traditional GPs is a common approach for predicting a smooth surface. The method will be used here as a "control" in order to compare the developments in this thesis proposed as more appropriate approaches for prediction over a manifold surface. The next section introduces Intrinsic GPs on manifolds, which are specifically designed for manifolds and utilize the manifold's intrinsic geometric structure for more accurate modeling [114].

3.3 Intrinsic Gaussian Processes on Manifolds

Since the RBF kernel used in Traditional GPs does not take into account the intrinsic geometric features of the manifold, this section introduces a kernel that better expresses the intrinsic geometric characteristics of the manifold. The heat kernel, which describes heat transfer across domains, whether in Euclidean space or more complex structures like manifolds, conveys essential structural information about the domains. By constructing a heat kernel on the manifold, the aim is to capture its geometric structure more effectively. As previously defined, M is the Riemannian manifold with ∂M representing its complex boundary. Let Δ_s denote the Laplace-Beltrami operator on M, and δ represent the Dirac delta function. A heat kernel of M is a smooth

function $K(x,y,t)$ on $M \times M \times \mathbb{R}^+$ that satisfies the heat equation:

$$\frac{\partial}{\partial t} K_{\text{heat}}\left(s_0,s,t\right) = \frac{1}{2}\Delta_s K_{\text{heat}}\left(s_0,s,t\right),$$
$$\lim_{t\to 0} K_{\text{heat}}\left(s_0,s,0\right) = \delta\left(s_0,s\right),$$

where $s_0, s \in S$ lie on the manifold M and the initial condition holds in a distributional sense [10]. The initial condition using the Dirac delta function δ means that at the initial moment, the heat is concentrated at the starting point and has not yet diffused. The heat kernel serves as a solution to the heat equation, discussed in Section 2.1.1. When ∂M is nonempty, multiple heat kernels exist. To make the heat kernel unique, this can be achieved by adding the Neumann boundary condition:

$$\frac{\partial K}{\partial \mathbf{n}} = 0 \quad \text{along } \partial M,$$

where $\mathbf{n}$ is a normal vector of ∂M, which means no heat transfer across the boundary ∂M. The heat kernel satisfies $K_{\text{heat}}\left(s_0,s,t\right) = K_{\text{heat}}\left(s,s_0,t\right)$ and for any fixed time t, it is a positive semi-definite kernel on M. Thus, the heat kernel is suitable for being the covariance matrix for GPs on M. Considering the form of the heat kernel when M is a d-dimensional Euclidean space $\mathbb{R}^d$, which is a special type of manifold,

$$K_{\text{heat}}\left(\mathrm{x}_0,\mathrm{x},t\right) = \frac{1}{(2\pi t)^{d/2}} \exp\left\{-\frac{\|\mathrm{x}_0-\mathrm{x}\|^2}{2t}\right\}, \mathrm{x} \in \mathbb{R}^d, \tag{3.5}$$

it has a similar structure to the RBF kernel as shown in Equation (3.2). The time parameter t corresponds to the lengthscale l of RBF kernel, influencing how quickly the covariance diminishes. Let $K^t_{\text{heat}}\left(s,s_0\right) = K_{\text{heat}}\left(s,s_0,t\right)$, the next challenge is how to determine the detailed heat kernel for a manifold. Given that explicit solutions for the heat kernel are primarily accessible for particular manifolds like Euclidean spaces and spheres, for most manifolds, the heat kernel cannot be directly derived and explicitly express. Considering that the random walk of BM in space is constrained by internal geometric features, such as the inability to move beyond boundaries, the transition density of BM within M is utilized to model the heat kernel on this manifold for constructing covariance kernels.

3.3.1 The Approximation of the Heat Kernel

The numerical approximation of the heat kernel K^t_{heat} can be derived as below. Each Riemannian manifold M has its own metric tensor g. Let $\phi : \mathbb{R}^d \to M$ be a smooth local function around point $s_0 \in M$ and $x(t_0) \in \mathbb{R}^d$ be such that $\phi(x(t_0)) = s_0$. The local parameter function ϕ provides a mapping that makes simulating a stochastic process in $\mathbb{R}^d$ with the starting point $x(t_0)$ equivalent to simulating a sample path of BM on M with the starting point s_0, as illustrated in Figure 3.1. In this work, it is assumed that the local parameterisation ϕ is known. The detail of ϕ and g is provided in Section 6.3, for the case of the Bitten-torus. If ϕ is unknown, an approach to obtain ϕ is provided by Tosi et al. [151], through nonlinear dimensionality reduction using latent variable models.

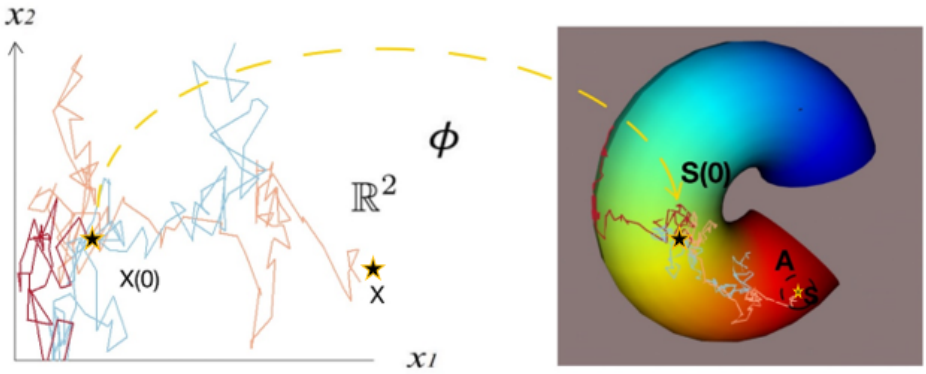


Figure 3.1: BM on the Bitten-torus and its equivalent stochastic process in $\mathbb{R}^2$: Three BM sample paths from same initial point s_0, shown in different colours; only the pink sample path reaches Borel set A (which can be considered a neighborhood of point s)at time t. $\phi : \mathbb{R}^2 \to M$ is a local parametrisation of M.

The BM on a Riemannian manifold is given as a system of stochastic differential equations (SDE) in the Ito form, refer to [64], [65]:

$$dx_i(t) = \frac{1}{2} G^{-1/2} \sum_{j=1}^{d} \frac{\partial}{\partial x_j} \left(g^{ij} G^{1/2} \right) dt + \left(g^{-1/2} dB(t) \right)_i, \tag{3.6}$$

where g is the metric tensor of Riemannian manifold M, g^{ij} is the (i, j) element of its inverse, G is the determinant of the matrix g and $B(t)$ represents an independent BM in the Euclidean space. The first term of Equation (3.6) corresponds to the local curvature of M, while the second term reflects the position-specific alignment of BM by transforming the standard BM $B(t)$ in $\mathbb{R}^d$

according to the metric tensor g. Using the Euler-Maruyama method [79], [84], the equation can be derived in:

$$dx_i(t) = \frac{1}{2}\sum_{j=1}^{d}\left(-g^{-1}\frac{\partial g}{\partial x_j(t)}g^{-1}\right)_{ij}dt + \frac{1}{4}\sum_{j=1}^{d}\left(g^{-1}\right)_{ij}\mathrm{tr}\left(g^{-1}\frac{\partial g}{\partial x_j(t)}\right)dt + \left(g^{-1/2}dB(t)\right)_i,$$

where dt is the diffusion time of each step of the BM simulation.

The BM path S(t) on M is defined starting from s_0 at time $t = 0$. The probability that S(t) enters any Borel set A on M at time t, i.e., S(t) ∈ A ⊂ M, is given by

$$\mathbb{P}\left[S(t) \in A \mid S(0) = s_0\right] = \int_A K^t_{\text{heat}}\,(s_0, s)\,ds,$$

where the integral is defined as the volume form of M. This establishes the connection between the heat kernel and the transition density of Brownian motion. The transition probability is approximated as

$$\mathbb{P}\left[S(t) \in A \mid S(0) = s_0\right] \approx \frac{k}{N},$$

where N is the total number of simulated BM sample paths and k is the number of BM paths which reach A at time t. Figure 3.1 shows three different sample BM paths originating from the same initial point s_0. Each sample path is represented by a distinct color. However, only the pink path successfully reaches neighborhood A of the target point s at time t. This process also corresponds to the stochastic process in $\mathbb{R}^2$ shown on the left, where only the pink path reaches the area near point x at time t. The estimate of the transition probability $p\left(S(t) \in A \mid S(0) = s_0\right)$ in this example is 1/3.

The transition density of $S(t)$ at s is approximated as:

$$K^t_{heat}\,(s_0, s) \approx \hat{K}^t = \mathbb{P}\left[S(t) \in A \mid S(0) = s_0\right] \approx \frac{1}{V(A)} \cdot \frac{k}{N}, \tag{3.7}$$

where $V(A)$ is the Riemannian volume of A, and $\widehat{K}^t$ is the estimated transition density, as well as the estimated heat kernel. t is the BM diffusion time. If t is large, the BM paths have a higher probability to reach A. In practical applications, N is typically set to $5e^4$ or higher. The median of the relative error decreases as N increases and stabilises after reaching $3e^4$ [114]. Note that the Neumann boundary condition corresponds to BM reflecting at the boundary. This can be

approximated by pausing time and resampling the next step until it falls into the interior of M. The next section will discuss the feasibility of the "resample" method and introduce the novel "reflection" approach proposed to improve the accuracy of the heat kernel near the boundaries.

Let Σ be the covariance matrix for all grid points on M. Given two grid points s_i and s_j on M, $\Sigma_{i,j}$ can be constructed by simulating N BM paths starting at s_i and numerically evaluating the transition density of the BM at s_j using Equation (3.7). Then,

$$\Sigma_{ij} = \sigma_h^2 K_{heat}^t \left(s_i, s_j\right), \tag{3.8}$$

where the rescaling hyperparameter σ_h^2 is used to add extra flexibility to control the magnitude of the kernel. Once the N BM paths are simulated, the heat kernel between s_i and other grid points can be evaluated from the existing simulated paths. Under the GP prior for the unknown objective function $f : M \to R$, the following holds, similar to Traditional GPs:

$$p(\mathbf{f}_r | s \in S) = \mathcal{N}(0, \Sigma),$$

where $\mathbf{f}_r$ is the vector of the realisation of f at the grid points. As previously defined, let $\mathbf{f}_{\mathscr{D}}$ represent the vector of the realisation of f at the observed data points, also named the training points dataset. The joint distribution of $\mathbf{f}_{\mathscr{D}}$ and $\mathbf{f}_r$ is :

$$p(\mathbf{f}_{\mathscr{D}}, \mathbf{f}_r) = \mathcal{N}\left(0, \begin{bmatrix} \Sigma_{\mathscr{D}\mathscr{D}} & \Sigma_{\mathscr{D}\mathrm{r}} \\ \Sigma_{\mathrm{r}\mathscr{D}} & \Sigma_{\mathrm{rr}} \end{bmatrix}\right)$$

where $\Sigma_{\mathscr{D}\mathscr{D}}$ is the covariance matrix for all data points in $\mathscr{D}$ and Σ_{rr} is the covariance matrix for all grid points S. Each element of the covariance matrix for the joint distribution can be computed using Equation (3.8). Then, the predictive distribution is derived as the same form with Equation (3.4):

$$p\left(\mathbf{f}_r \mid y\right) = \mathcal{N}\left(\Sigma_{\mathrm{r}\mathscr{D}}\left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_n^2 I\right)^{-1} y, \Sigma_{\mathrm{rr}} - \left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_n^2 I\right)^{-1} \Sigma_{\mathscr{D}\mathrm{r}}\right), \tag{3.9}$$

where σ_n^2 is the noise variance and the predictive mean and predictive variance take the form of:

$$\mu(S) = \Sigma_{\mathrm{r}\mathscr{D}} \left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_n^2 I\right)^{-1} y,$$
$$\sigma(S) = \Sigma_{\mathrm{rr}} - \left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_n^2 I\right)^{-1} \Sigma_{\mathscr{D}\mathrm{r}}.$$

After explaining how Intrinsic GPs use the transition density of BM paths to approximate the heat kernel, which leverages the intrinsic geometric features of the manifold, the next section proposes a novel "reflection" approach for improving heat kernel's accuracy near the boundaries, instead of the "resample" method.

3.4 Improvements to Heat Kernel Estimation at the Boundary

The heat kernel on a manifold is estimated by the transition density of BM paths. As introduced before, in Intrinsic GPs, when BM reaches the boundaries of the manifold, it employs a "resample" strategy. "Resample" means pausing the process and resampling the subsequent step until it is limited to the manifold's limits. However, the Neumann boundary condition adopted on the heat kernel corresponds to BM reflecting at the boundary [176]. "Reflection" refers to the behavior where, upon reaching a boundary, the BM path is reflected back into the domain instead of crossing the boundary. To better illustrate the difference between these two methods, Figure 3.2 presents how both methods keep BM within the boundary by using a new U-shape with a wider intermediate gap. It is easy to observe that BM paths remain within the manifold boundary in both the left and right figures.

In the circular insets on both figures, a zoomed-in view of the boundary region (highlighted by the blue dashed line) provides a simple schematic diagram at the boundary. The left one shows one BM path under the reflection method. In the circular inset, the blue line represents the boundary, and the black line represents the valid BM path within the boundary. If the BM path at step $i+1$ crosses the boundary, the part inside the boundary is preserved, and the BM path outside the boundary (indicated by the red dashed line, with arrows indicating the direction) is

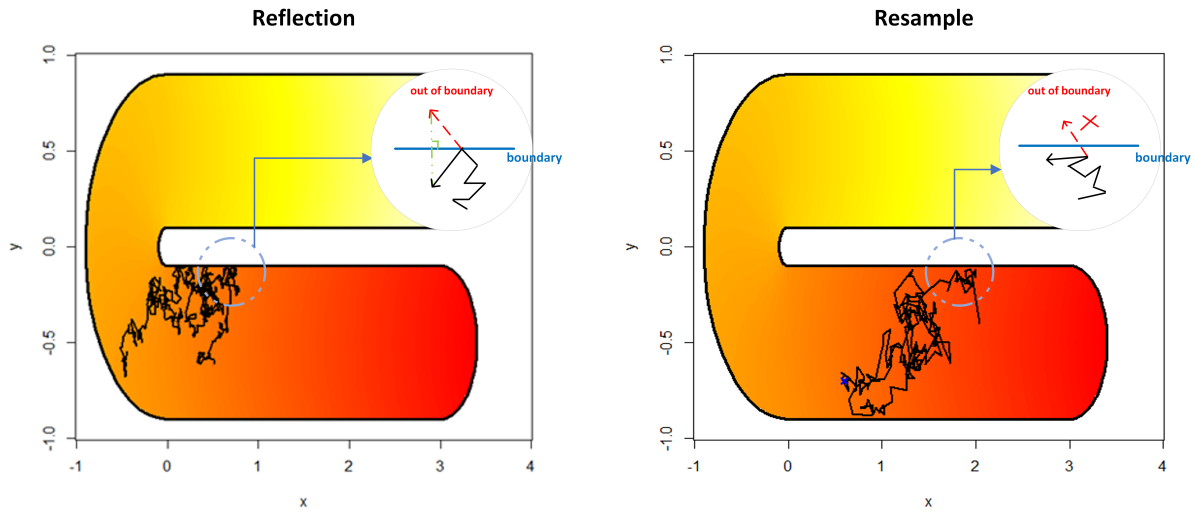


Figure 3.2: The left one shows the BM path under the reflection method while the right one shows the BM path under the resample method. The blue line represents the simplified boundaries, the black line is the BM path and the red dashed line is part of the BM path running out of the boundary not being accepted.

reflected into the manifold (indicated by the length of the black solid line, with arrows indicating the direction) using the boundary as the reflection axis. However, for the BM path under the resample method shown in the right part, if the BM path at step $i+1$ exceeds the boundary, the step $i+1$ is canceled and resampled until it stays within the boundary. The newly adopted BM path for step $i+1$ is shown as a black line, with arrows indicating the new direction. In the "reflection" method, BM paths retain the portion of step $i+1$ that remains within the manifold, even when the entire step $i+1$ would be "rejected" under the "resample" method. Since the heat kernel is estimated by the transition density of BM paths, the "reflection" method provides a more accurate estimation of the heat kernel near the boundary. In contrast, the "resample" method, while having lower computational requirements, lacks sufficient exploration near the boundary within a certain range, often underestimating the transition density at the boundary and resulting in a value lower than the true heat kernel.

Compared to the "reflection" method, the "resample" method used in the original Intrinsic GPs is much simpler to implement. "Reflection" requires to be discussed considering the complexity of the boundary in the calculation. BM paths not only need to determine whether the next step is outside the domain, but they also need to identify the specific boundary segment they have crossed and perform the necessary calculations. In some cases, after the initial reflection,

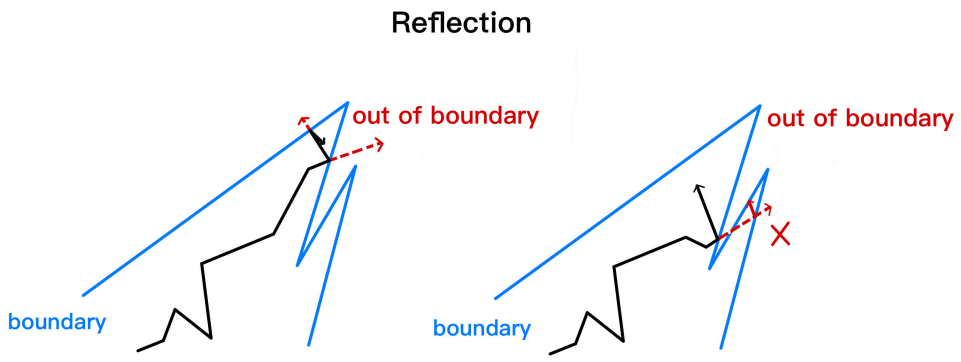


Figure 3.3: An illustration of two "reflection" examples for BM paths: The blue line represents the boundary, while the black lines show the BM paths inside the boundary. The red dashed line indicates the part of the BM path running out of the boundary not being accepted; In the left diagram, multiple "reflections" are required, whereas in the right diagram, it is necessary to determine which part of the boundary was crossed first.

the path may cross another boundary segment and still remain outside the manifold, requiring multiple reflections. Figure 3.3 illustrates a simplified schematic based on the above analysis. The blue line represents the boundary, and the black line shows a BM path within the boundary. The red dashed line represents the part of BM's step $i+1$ outside the boundary that is not accepted by the "reflection" method. In the left diagram of Figure 3.3, the red dashed line outside the boundary requires a second reflection after the first reflection still leaves it outside the boundary. During this "reflection" process, the parts within the boundary are retained, whereas the "resample" method would reject the entire step and resample a new step. The right diagram of Figure 3.3 illustrates a scenario where step $i+1$ of the BM path crosses multiple boundaries. It is necessary to determine which boundary was crossed first before applying the "reflection." Otherwise, it may result in an error, as indicated by the red solid line in the diagram. Figure 3.3 visually illustrates potential computational issues that may arise in the "reflection" method.

Building on the previous analysis, the next section aims to prove the differences between the two methods using the real heat kernel. However, since the true heat kernel cannot be directly calculated from Equation (3.5) due to the absence of boundary conditions, the proofs will first be conducted through a one-dimensional example.

3.4.1 Comparison in 1-D Example

The nonnegative real line serves as an example of a manifold with a boundary:

$$\mathbb{R}_{\geq 0} = \{x \in \mathbb{R} : x \geq 0\}.$$

In this case, the BM path starts at a point $x_0 = \tau$, where τ is a positive number (greater than zero), at the initial time $t = 0$. When considering only one step of the BM path, the PDF for "reflection" and "resample" can be written as follows:

Reflection at boundary: For each BM step ΔX, first sample from $\mathcal{N}(0, \Delta t)$, and if the path goes outside the boundary at $x = 0$, it is reflected back into $\mathbb{R}_{\geq 0}$. Notice that $X_{\Delta t} = x$ either if $\Delta X = x - \tau$ or if $\Delta X = -x - \tau$ accompanied by a reflection. Therefore the PDF of $X_{\Delta t}$ is

$$p_{\text{reflect}}(x) = \frac{1}{\sqrt{2\pi\Delta t}} \exp\left[-\frac{(x-\tau)^2}{2\Delta t}\right] + \frac{1}{\sqrt{2\pi\Delta t}} \exp\left[-\frac{(x+\tau)^2}{2\Delta t}\right] \quad \text{for } x > 0,$$

where the first part is the PDF of a new point that falls in a region greater than zero (i.e., the path remains within the boundary before reflection), and the second part is the PDF of a new point that falls in a region smaller than zero (i.e., the path is reflected back into the boundary after reflection). The formula for $p_{\text{reflect}}(x)$ is derived by separately handling the two cases: the path staying within the boundary and the path being reflected after crossing the boundary. This formula also represents the true heat kernel for a one-dimensional space with a boundary set at 0 when considering only one BM step.

Resample at boundary: Alternatively, "resample" is performed from $\mathcal{N}(0, \Delta t)$ until the path stays inside $\mathbb{R}_{\geq 0}$. In this case, the PDF of $X_{\Delta t}$ is represented by dividing the original distribution by the probability P, where P is the probability that the path remains within the boundary. This ensures that the adjusted probability density function describes the situation where the path remains within the boundary, as shown in:

$$p_{\text{resample}}(x) = P^{-1} \cdot \frac{1}{\sqrt{2\pi\Delta t}} \exp\left[-\frac{(x-\tau)^2}{2\Delta t}\right] \quad \text{for } x > 0,$$

where P is the probability that a single sample from $\mathcal{N}(0, \Delta t)$ gives a result that stays inside

$\mathbb{R}_{\geq 0}$, i.e.

$$P = \frac{1}{\sqrt{2\pi\Delta t}} \int_0^\infty \exp\left(-\frac{(x-\tau)^2}{2\Delta t}\right) dx = 1 - \varphi(-\tau) = \varphi(\tau),$$

where φ denotes the CDF of $\mathcal{N}(0, \Delta t)$.

Using the Taylor expansion around $\tau = 0$, the difference between $p_{\text{resample}}(x)$ and $p_{\text{reflect}}(x)$ is:

$$\lim_{\tau \to 0} p_{\text{resample}}(x) - p_{\text{reflect}}(x) = 0.$$

Further details can be found in Appendix C. This leads to the conclusion that, in the one-dimensional case, when considering only one step of the BM, the results of the two methods do not differ significantly as the initial point approaches zero. However, when the BM path moves more than one step, the above inference cannot be applied. The corresponding heat kernel can be derived by solving the partial differential equation:

$$\begin{aligned} \frac{\partial}{\partial t} K_{\text{heat}}(s_0, s, t) &= \frac{1}{2} \Delta_s K_{\text{heat}}(s_0, s, t), \\ \lim_{t \to 0} K_{\text{heat}}(s_0, s, 0) &= \delta(s_0, s), \quad s_0, s \in M, \\ \frac{\partial K}{\partial n} &= 0 \quad \text{along } \partial M, \end{aligned}$$

which can be simplified as:

$$\begin{cases} u_t = k u_{xx}, & 0 \leq x < \infty, 0 < t < \infty, \\ u(x,0) = g(x), & \text{(IC)}, \\ u_x(0,t) = 0, & \text{(BC)}, \end{cases}$$

where x represents s and $g(x)$ is the Dirac delta function $\delta(x)$ introduced in Section 3.3. $u(x,t)$ can be represented as the convolution of Green's function and the initial condition:

$$u(x,t) = \int_{-\infty}^{\infty} G(x,y,t) g(y)\, dy, \tag{3.10}$$

where $G(x,y,t) = \frac{1}{\sqrt{4\pi k t}} \exp\left(-\frac{(x-y)^2}{4kt}\right)$. To comply with the boundary condition, it is necessary to handle the effects of y symmetrically, ensuring that the derivative is zero at $x = 0$. Therefore,

considering the function is symmetric, Green's function should be modified as:

$$G(x,y,t) = \frac{1}{\sqrt{4\pi kt}}\left[\exp\left(-\frac{(x-y)^2}{4kt}\right) + \exp\left(-\frac{(x+y)^2}{4kt}\right)\right], \tag{3.11}$$

where the second term is symmetric about the origin with respect to y. Combining Equations (3.10) and (3.11) gives the following expression:

$$u(x,t) = \frac{1}{\sqrt{4\pi kt}}\int_0^\infty \left[\exp\left(-\frac{(x-y)^2}{4kt}\right) + \exp\left(-\frac{(x+y)^2}{4kt}\right)\right] g(y)dy.$$

Since x and y are both non-negative and G already contains the positive and negative parts of y, the integral is restricted to the non-negative real domain. The initial condition $\delta(x)$, which is also represented by $g(x)$, introduces discontinuities over the feasible domain, resulting in singularities that are difficult to handle in integrals. Consequently, this can make both analytical and numerical solutions highly complex. Moreover, the solutions may exhibit physically unrealistic behavior near discontinuity points.

Based on the properties of $\delta(x)$, this work proposes an approximation using a normal distribution $\mathcal{N}(\mu, 0.1)$, as shown in Figure 3.4.1. When the variance is sufficiently small, the normal distribution exhibits a sharp peak with a narrow shape. As demonstrated in Figure 3.4.1, the distribution reaches its maximum value near μ (corresponding to the initial point position), with fewer observations in the tails of the distribution. This substitution allows for an analytic expression of the heat kernel:

$$u(x,t) = \frac{1}{\sqrt{4\pi kt}}\int_0^\infty \left[\exp\left(-\frac{(x-y)^2}{4kt}\right) + \exp\left(-\frac{(x+y)^2}{4kt}\right)\right] \frac{1}{\sqrt{0.2\pi}}\exp\left(-\frac{(y-\mu)^2}{0.2}\right) dy,$$

where μ here corresponds to the initial point position at $t = 0$.

Considering the boundary condition at 0, either the "resample" or "reflection" method is applied when the BM path reaches the boundary in one dimension. Figure 3.5 compares the true heat kernel with the heat kernels estimated using the reflection and resample methods, respectively. Among them, the purple lines represent the true heat kernel aimed to approximate in the one-dimensional case, while the red and blue lines correspond to the reflection method and the resample method, respectively. In both Figure 3.5(a) and Figure 3.5(b), the estimated heat kernels from both methods closely match the true heat kernel. These figures depict BM paths

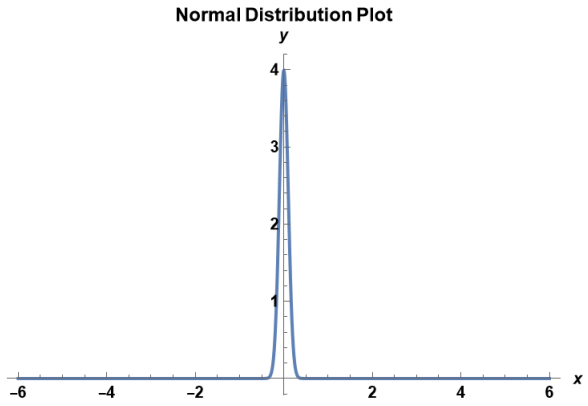


Figure 3.4: Normal distribution $\mathcal{N}(\mu, 0.1)$ with $\mu = 0$ and $\delta = 0.1$ used to instead the Dirac delta function $\delta(x)$.

originating from positions 1 and 2, respectively, with parameters $T_{max} = 1$ and $T_{len} = 13$, where T_{max} represents the maximum path length for the BM and T_{len} denotes the number of steps taken by the BM. These parameters determine the length of each step in the BM path. For Figures 3.5 (c)-(e), all BM paths start from position 2 with different settings for T_{max} and T_{len}. The reflection method's estimated heat kernel closely aligns with the true heat kernel and significantly outperforms the resample method. Notably, near the boundary, the resample method fails to fit the true heat kernel accurately. As analyzed in Section 3.4, the resample method struggles to place BM paths near the boundary, creating gaps and reducing the number of BM paths calculated in this region. This leads to estimated heat kernel values that are smaller than the true values. However, when the BM path starts from position 4, both methods yield accurate estimates because the starting point is far enough from the boundary, making the boundary's influence negligible.

In this section, a comparison is made between the estimated heat kernel under the reflection method and the resample method, alongside the true heat kernel, focusing on both single-step and multi-step scenarios in one-dimensional space. When BM paths take only a single step, the difference between the two methods is minimal, particularly as the starting point approaches zero. However, with multiple steps, the reflection method demonstrates a clear advantage over the resample method. The next section will extend this exploration to two-dimensional space.

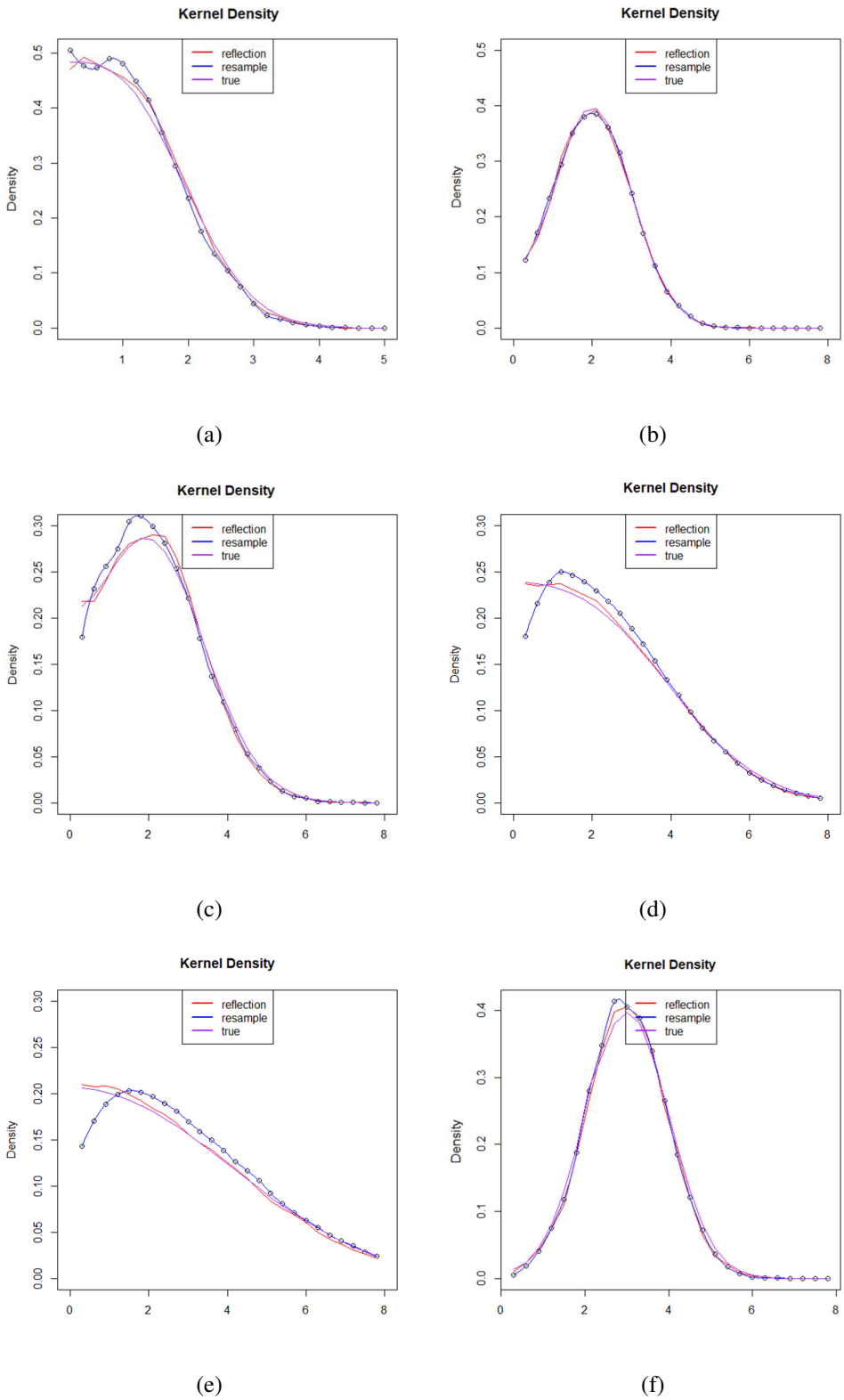


Figure 3.5: The comparison between the true heat kernel and the estimated heat kernel using the reflection method and resample method at the boundary separately. The boundary is set at zero. (a) BM starts from 1, Tmax=1, Tlen=13, win=0.3; (b) BM starts from 2, Tmax=1, Tlen=13, win=0.3; (c) BM starts from 2, Tmax=5, Tlen=13, win=0.3; (d) BM starts from 2, Tmax=10, Tlen=13, win=0.3; (e) BM starts from 2, Tmax=10, Tlen=20, win=0.3; (f) BM starts from 4, Tmax=10, Tlen=20, win=0.1.

3.4.2 Comparison in 2-D Example

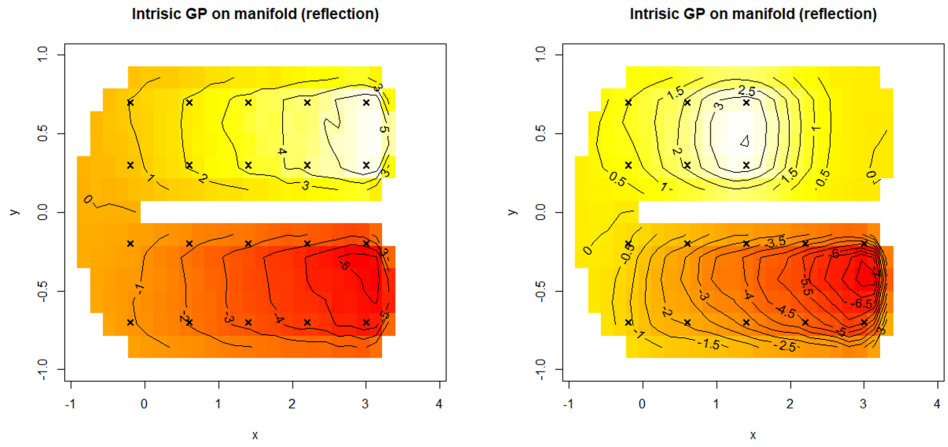
In two-dimensional space, the true heat kernel often lacks a specific expression, making direct comparisons of heat kernels impossible. In light of this, the accuracy of the estimated heat kernel under two methods will be compared by predicting the true function values through Intrinsic GPs, as discussed in Section 3.3.

To better observe the differences between the "reflection" method and the "resample" method near the boundary, this comparison, like in Figure 3.2, will also be conducted on a new U-shape domain with a wider intermediate gap. Different from the U-shape introduced in Chapter 1 and used in Chapters 6 and 7, the new U-shape shown in Figure 3.6(e) has a wider gap in the middle, making it easier to observe the regression performance near the central boundary. The rest of the set-up of the U-shape remains the same. This design helps to compare how effectively the "reflection" and "resample" methods handle boundary interactions. Figures 3.6 (a)-(d) display the predictive mean for two different sets of training points: the left side corresponds to 20 training points uniformly distributed across the manifold, while the right side corresponds to 16 training points, excluding the four points in the upper right corner compared to the first set. Figures 3.6 (a) and (b) show the results using the Intrinsic GPs with the "reflection" method, while Figures 3.6 (c) and (d) illustrate the "resample" method. When the 20 training points are uniformly distributed across the manifold, it is evident that the Intrinsic GP under the "reflection" method yields a noticeably smoother predictive mean, especially near the boundaries. However, with 16 training points, the difference in smoothness is less pronounced.

To provide a numerical comparison, the Root Mean Square Error (RMSE) is chosen as the metric. The RMSE is a standard way to measure the accuracy of a model in predicting quantitative data, widely used in both numerical and statistical analyses. It is used to compare the prediction results from the Intrinsic GPs under "reflection" and "resample" methods. The equation of the RMSE is:

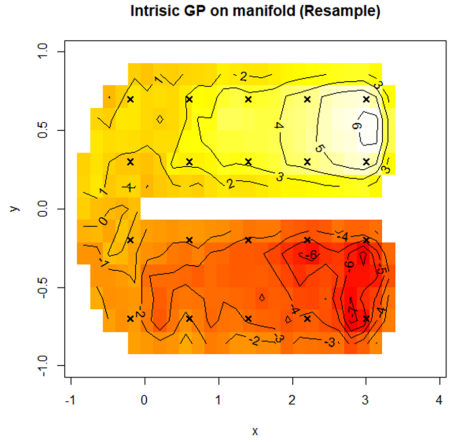
$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{\mu} - \mu)^2}, \tag{3.12}$$

where n is the number of testing points, μ is the true value of the testing points and $\hat{\mu}$ is the predictive mean calculated by different model. The RMSE results in Table 3.1 compare the performance of Traditional GPs and Intrinsic GPs using the "reflection" and "resample" methods

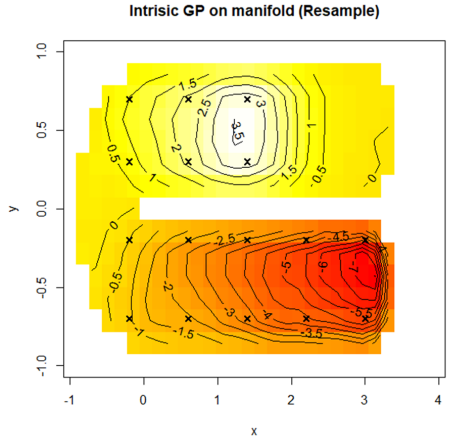


(a)

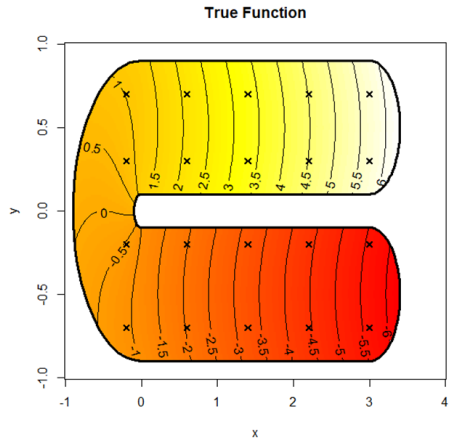
(b)



(c)



(d)



(e)

Figure 3.6: Predictive means for different methods on the U-shape domain: (a) Intrinsic GP using "reflection" method with 20 training points; (b) Intrinsic GP using "reflection" method with 16 training points; (c) Intrinsic GP using "resample" method with 20 training points; (d) Intrinsic GP using "resample" method with 16 training points; (e) The true function of the new U-shape with wide intermediate gap.

RMSE	Traditional GPs	Intrinsic GPs (Reflection)	Intrinsic GPs (Resample)
20 training points	1.149192	0.9593393	1.147957
16 training points	3.004057	2.183035	2.196994

Table 3.1: Comparison of the RMSE of predictive means for different methods on the new U-shape (with wide intermediate gap).

for two sets of training points (20 and 16 points). For 20 training points, the Intrinsic GP with the “reflection” method achieves the lowest RMSE of 0.96, outperforming both the Traditional GP and the Intrinsic GP with the “resample” method. For 16 training points, the Intrinsic GP with the “reflection” method again performs the best with an RMSE of 2.183035. These results also validate the earlier findings in one dimension, confirming the superiority of the “reflection” method over the “resample” method.

Computational Efficiency

This section discusses the trade-offs between the "reflection" and "resample" methods for handling boundaries in Intrinsic Gaussian Processes (GPs), especially when applied to manifolds with complex boundaries in two or higher-dimensional spaces. The "reflection" method, while more accurate in capturing boundary interactions, incurs significantly greater computational cost due to the usually complex boundaries of manifolds. As simplified in Figure 3.3, two scenarios that need to be considered under the “reflection” method are illustrated. In contrast, the "resample" method offers a simpler, less computationally expensive approach by resampling paths which cross the boundary.In this U-shape case, there is no difference in memory usage between the two methods. Using a system with an 11th Gen Intel i5-1145G7 CPU (2.60 GHz) and 16 GB RAM, simulating BM paths 50,000 times from a single point using the “reflection” method took approximately 10.67 minutes on average. In contrast, the “resample” method required only 0.87 minutes under the same conditions, representing nearly a 12-fold improvement in computational efficiency. This efficiency gap becomes increasingly significant as the sample size or data volume grows, resulting in greater time savings by the “resample” method. This is because, in addition to the two scenarios described in Figure 3.3, the “reflection” method requires more computations at the boundary—for instance, determining the direction of reflection. In contrast, the resampling method involves no such calculations and only needs to check whether the next step falls outside the boundary and resample if necessary.

From the perspective of Intrinsic GPs on manifolds, the accuracy loss of the estimated heat kernel at the boundaries does not significantly disrupt the overall predictive outcomes generated by the Intrinsic GPs. Therefore, the choice between "reflection" and "resample" depends on the specific task objective. In tasks where boundary behavior is crucial, the "reflection" method is more suitable. If the boundary is less critical or exploration inside the domain is prioritized, the "resample" method may be more appropriate. The choice also depends on computational complexity and accuracy requirements. The "resample" method is generally simpler to implement and may have lower computational overhead, though with some loss of accuracy.

3.5 Conclusion

This chapter first reviews the Traditional GPs, which are based on the RBF kernel. The RBF kernel is a widely used covariance function in Euclidean space, providing smooth and stationary predictions. However, due to the complex intrinsic geometry of manifolds, the RBF kernel based on Euclidean distance is not well-suited for capturing the intrinsic structure of such spaces. It does not respect the actual distances between points on the manifold, especially when these points are separated by gaps and boundaries. This motivates the need for more sophisticated approaches which can take the characteristics of manifolds into account. Consequently, the chapter introduces the Intrinsic GPs. The kernel Intrinsic GP uses an approximate heat kernel. Except for very special manifolds, such as Euclidean spaces and spheres, which have explicit solutions, such solutions do not exist for general Riemannian manifolds. Therefore, an approximation of the heat kernel is necessary.

The approximate heat kernel used here is interpreted as the transition densities of BM on the manifold M [114]. BM paths simulate from each grid point and use a "resample" method when reaching the boundary. "Resample" here means to generate a new step until it stays within the boundary. This chapter addresses improvements to the modelling of BM as it approaches boundaries, offering more accurate handling of edge conditions through the "reflection" method. "Reflection" is the symptom of the Neumann boundary condition on the heat kernel, ensuring that BM reflects at the boundary rather than crossing it. "Reflection" allows for more accurate predictions at the boundaries and ensures better handling of the manifold's complex bound-

ary characteristics. In one-dimensional space, the "reflection" method clearly outperforms the "resample" method, particularly in estimating the true heat kernel near boundaries. In two-dimensional space, the accuracy of these methods is assessed by comparing the RMSE of predictive means of the Intrinsic GPs against true function values. The "reflection" method, due to its better handling of boundary conditions, consistently produces smoother and more accurate estimates than the "resample" method. However, the improvement in accuracy comes with additional computational overhead, which increases as the boundary complexity grows, as discussed in detail within the text. The choice of method primarily depends on the specific task objectives, as well as balancing computational complexity and approximation accuracy.

Overall, this chapter proposes Intrinsic GPs suitable for manifolds, where the use of the approximated heat kernel effectively captures the intrinsic geometric features of the manifolds. Additionally, the "reflection" approach is introduced to improve prediction accuracy near boundaries. However, Section 3.3 points out that it presents significant challenges for Intrinsic GPs to be directly utilized in manifolds. In Intrinsic GPs, for the same row in the covariance matrix, all elements can be estimated with the same batch of BM simulations which come from the same starting points. The calculation of the predictive mean only requires the BM simulations originating from all training points $\mathscr{D}$, where $\Sigma_{\mathrm{r}\mathscr{D}} = \Sigma_{\mathscr{D}\mathrm{r}}$ (symmetric). However, calculating the predictive variance requires BM paths simulated from all grid points for N times ($N = 50000$ in this research). For instance, in the examples used in this study, the U-shape, the Bitten-torus, and the Aral Sea, there are 418, 600, and 485 grid points, respectively. This leads to a significant computational burden. Although the BM simulations can be executed in parallel, the computational cost remains high. Moreover, GPs face the well-known problem of high computational complexity $O(n^3)$, where n corresponds to the number of training points, due to the inversion of the covariance matrix.

Chapter 4 will introduce new sparse Intrinsic GPs, which utilize different sparse methods to ensure feasibility while preserving the intrinsic geometric structure of the manifold, efficiently managing large datasets and high-dimensional manifolds.

Chapter 4

Sparse Intrinsic Gaussian Processes

Chapter 3 highlights the computational complexity inherent in Intrinsic GPs. This chapter aims to reduce computational challenges by introducing approximation methods, also known as sparse methods, into the model. The Deterministic Inducing Conditional (DIC) approximation is initially chosen to improve the model due to its ease of use (described in Section 4.2), but it presents certain issues. Consequently, the Deterministic Training Conditional (DTC) approximation is employed to address the problems associated with DIC (outlined in Section 4.3). However, given the limitations of DTC, the Variational Inference (VI) is introduced to further enhance the model's performance (explained in Section 4.4).

This chapter begins by establishing the theoretical foundations of inducing variables in Section 4.1, which are a key component of sparse methods, and then provides a concise introduction to the sparse methods used subsequently. Section 4.2 discusses the Intrinsic GPs improved by using the DIC approximation. Sections 4.3 provides the new sparse Intrinsic GPs combined with the DTC approximation. Finally, Section 4.4 explores how to use the VI to enhance Intrinsic GPs from a new perspective, transforming the prediction problem into an optimisation problem. Through these sections, the chapter systematically explores the progression from the DIC to DTC, and ultimately to VI, demonstrating how each approximation method reduces computational complexity and improves model performance. Section 4.5 provides a summary of the applications of the three sparse methods discussed in this chapter in the context of the Intrinsic GP.

4.1 Introduction of Inducing Variables

In Chapter 3, the discussion begins with the traditional GPs based on the RBF kernel as a powerful non-parametric Bayesian method, possessing excellent performance in handling complex regression problems. On this basis, the Intrinsic GPs built on the approximated heat kernel takes into account the complex boundary and internal geometric features on manifolds to solve the predictive regression problem on manifolds. As proposed in Chapter 3, the Intrinsic GPs face certain computational challenges, making the direct application on manifolds impractical. For example, for a training dataset $\mathscr{D}$ of size n, the GP regression process involves inverting an $n \times n$ covariance matrix, which has a computational complexity of $O(n^3)$. When the dataset is large, the computational and storage overhead becomes very high, making it difficult to handle large-scale datasets. Additionally, with a large training dataset, the inversion process may encounter numerical instability issues. Moreover, since the intrinsic GP uses the transition density of BM to model the heat kernel of the manifold, calculating the predictive variance requires running BM paths from each grid point as the initial point N times (in this research, $N = 5e^4$), which will bring high computational cost and lengthy computational process. Although parallel computing has been employed to reduce computational complexity, the need for substantial storage remains an urgent issue to address.

In Euclidean space, to address the computational limitations of GP regression, many researchers have dedicated their efforts to the study of sparse approximation methods, as discussed in Chapter 1. Sparse approximation is a technique for handling large datasets by using smaller subsets to approximate the original data, thereby reducing computational intensity. Although these sparse methods vary in their implementation and theoretical basis, they share a common characteristic: they treat a set of data points exactly while applying computationally efficient approximations to the remaining observation points. This approach reduces computational costs while preserving the predictive capabilities of GPs. These sparse approximation methods enable GPs to be applied to larger-scale and higher-dimensional datasets in Euclidean space. Quionero-Candela et al. [126] points out that these algorithms can be understood as "*exact inference with an approximate prior*," enabling the approximations to be directly formulated based on prior assumptions about the function.

A set of latent variables $u = [f(z_1), \ldots, f(z_m)]$ is first introduced to establish sparse approx-

imation methods on manifolds. These latent variables represent values of the GP (similar to f), corresponding to a set of m inducing points $z = [z_1, z_2, ..., z_m], z_i \in M$. The inducing points used in this study are not a subset of the training set; they are predefined on the manifold. This is because, on the manifold, the approximation of the heat kernel in Intrinsic GPs involves simulating BM. By utilizing the principles of sparse methods, the use of inducing points ensures that BM simulations are only required from these specific points, which will be discussed in detail in the following sections. By defining the inducing points on the manifold and completing the BM simulation in advance, the intrinsic geometric features of the manifold can be effectively captured and stored. The subsequent Intrinsic GPs and further applications can then simply retrieve the information stored from the BM simulations, significantly improving computational efficiency.

Since the grid points in GPs are equally spaced on the manifold, the selection of inducing points here also aims to distribute them as uniformly as possible across the manifold. This ensures that the same amount of information is provided for each part of the manifold M. The graphical model of relationships between the inducing points, training points and testing points is shown in Figure 4.1, where the training points and testing points are distinct. There is no direct communication path between the training and testing points. As defined earlier, $\mathbf{f}_{\mathscr{D}}$ represents the vector of $f(.)$ at the training points set $\mathscr{D}$, while $\mathbf{f}_r$ denotes the vector of $f(.)$ at all grid points set S (also referred to as test points). Information from $\mathbf{f}_{\mathscr{D}}$ can only be transmitted to $\mathbf{f}_r$ through the inducing variables u. u therefore induces the dependencies between training and test cases. This also explains why the term "inducing" is used, as these points serve as intermediaries that "induce" or facilitate the transfer of information. Once the inducing variables u are established, the DIC, DTC, and VI models introduced in the following sections will all leverage u to reduce computational complexity. The upcoming sections will provide a detailed exploration of the advantages, disadvantages, and limitations of each method.

4.2 Sparse Intrinsic Gaussian Process with DIC

Section 3.3 introduces the construction of the Intrinsic GPs and highlights the computational limitations. This section proposes Sparse Intrinsic Gaussian Process with DIC (SI-GPDIC) on

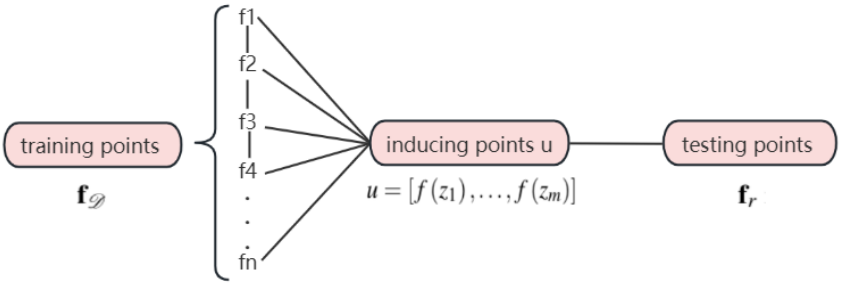


Figure 4.1: The graphical model of relationships between the inducing points, training points and testing points: training points are connected to inducing points; testing points are also connected to inducing points; no connection between training points and testing points; shown that information from $\mathbf{f}_{\mathscr{D}}$ can only be transmitted to $\mathbf{f}_r$ through the inducing variables u.

manifolds to address these challenges and improve computational efficiency, by utilizing the DIC sparse method.

As outlined in Section 4.1, it is necessary to introduce inducing points to address these computational challenges. The GP prior can be augmented with m inducing points on M denoted as $z = [z_1, z_2, ..., z_m]$, where $z_i \in M$, which are independent of the training points. The realisation of the objective function at the inducing points can be represented as the vector $u = [f(z_1), \ldots, f(z_m)]$. The marginal prior distribution $p(\mathbf{f}_r, \mathbf{f}_{\mathscr{D}})$ can be written in terms of the prior distribution $p(u)$ and the conditional distribution $p(\mathbf{f}_r, \mathbf{f}_{\mathscr{D}}|u)$.

$$p(\mathbf{f}_r, \mathbf{f}_{\mathscr{D}}) = \int p(\mathbf{f}_r, \mathbf{f}_{\mathscr{D}}|u)p(u)du, \quad p(u) = \mathscr{N}(0, \Sigma_{zz}), \tag{4.1}$$

where Σ_{zz} is the covariance matrix for all inducing points. Assuming that $\mathbf{f}_{\mathscr{D}}$ and $\mathbf{f}_r$ are conditionally independent given $\mathbf{u}$, see Figure 4.1, the GP joint prior can be approximated as

$$p(\mathbf{f}_r, \mathbf{f}_{\mathscr{D}}) \approx q(\mathbf{f}_r, \mathbf{f}_{\mathscr{D}}) = \int q(\mathbf{f}_r|u)q(\mathbf{f}_{\mathscr{D}}|u)p(u)du. \tag{4.2}$$

DIC models are finite models that are linear in their parameters, with a specific prior on the weights [126]. For input grid points dataset S, the corresponding function value $\mathbf{f}_r$ is expressed as:

$$\mathbf{f}_r = \sum_{i=1}^{m} k(s, z_i)\, \mathrm{w}_{\mathrm{z}}^{\mathrm{i}} = \Sigma_{\mathrm{rz}}\mathbf{w}_{\mathrm{z}}, \quad \text{with} \quad p(\mathbf{w}_{\mathrm{z}}) = \mathscr{N}\left(0, \Sigma_{\mathrm{zz}}^{-1}\right), \tag{4.3}$$

where each inducing input z_i corresponds to a different weight $\mathrm{w}_{\mathrm{z}}^{\mathrm{i}}$. Compared with Equation (4.1), the covariance matrix for the prior on the weights is the inverse of the covariance matrix for u. This guarantees that the GP prior on u can be precisely obtained, which is a Gaussian distribution with zero mean and covariance:

$$\left\langle uu^{\top} \right\rangle = \Sigma_{\mathrm{z,z}} \left\langle \mathbf{w}_{\mathrm{z}} \mathbf{w}_{\mathrm{z}}^{\top} \right\rangle \Sigma_{\mathrm{z,z}} = \Sigma_{\mathrm{z,z}} (\Sigma_{\mathrm{z,z}})^{-1} \Sigma_{\mathrm{z,z}} = \Sigma_{\mathrm{z,z}},$$

where $u = \Sigma_{\mathrm{zz}} \mathbf{w}_{\mathrm{z}}$ and $\left\langle uu^{\top} \right\rangle$ represents the expectation of the outer product $uu^{\top}$. Then, by taking $\mathbf{w}_{\mathrm{z}} = (\Sigma_{\mathrm{zz}})^{-1} u$, the function value $\mathbf{f}_r$ can be redefined as:

$$\mathbf{f}_r = \Sigma_{\mathrm{rz}} (\Sigma_{\mathrm{zz}})^{-1} u, \quad \text{with} \quad u \sim \mathcal{N}(0, \Sigma_{\mathrm{zz}}).$$

Similarly, given input training points dataset $\mathscr{D}$, the corresponding function value $\mathbf{f}_{\mathscr{D}}$ can be written as:

$$\mathbf{f}_{\mathscr{D}} = \Sigma_{\mathscr{D}\mathrm{z}} (\Sigma_{\mathrm{zz}})^{-1} u, \quad \text{with} \quad u \sim \mathcal{N}(0, \Sigma_{\mathrm{zz}}). \tag{4.4}$$

Through these derivations, the approximate conditional distributions in the integral (4.2) is given by:

$$\begin{aligned} q(\mathbf{f}_{\mathscr{D}} | u) &= \mathcal{N}\left(\Sigma_{\mathscr{D}\mathrm{z}} (\Sigma_{\mathrm{zz}})^{-1} u, 0\right), \\ q(\mathbf{f}_r | u) &= \mathcal{N}\left(\Sigma_{\mathrm{rz}} (\Sigma_{\mathrm{zz}})^{-1} u, 0\right), \end{aligned} \tag{4.5}$$

with zero conditional covariance. The GP approximate joint prior can be shown as:

$$\begin{aligned} q(\mathbf{f}_{\mathscr{D}}, \mathbf{f}_r) &= \mathcal{N}\left(0, \begin{bmatrix} \Sigma_{\mathscr{D}\mathrm{z}} \Sigma_{\mathrm{zz}}^{-1} \Sigma_{\mathrm{z}\mathscr{D}} & \Sigma_{\mathscr{D}\mathrm{z}} \Sigma_{\mathrm{zz}}^{-1} \Sigma_{\mathrm{zr}} \\ \Sigma_{\mathrm{rz}} \Sigma_{\mathrm{zz}^{-1}} \Sigma_{\mathrm{z}\mathscr{D}} & \Sigma_{\mathrm{rz}} \Sigma_{\mathrm{zz}}^{-1} \Sigma_{\mathrm{zr}} \end{bmatrix}\right) \\ &= \mathcal{N}\left(0, \begin{bmatrix} Q_{\mathscr{D}\mathscr{D}} & Q_{\mathscr{D}\mathrm{r.}} \\ Q_{\mathrm{r}\mathscr{D}} & Q_{\mathrm{rr}} \end{bmatrix}\right), \end{aligned}$$

note that $Q_{\mathscr{D}\mathscr{D}} = \Sigma_{\mathscr{D}\mathrm{z}} \Sigma_{\mathrm{zz}}^{-1} \Sigma_{\mathrm{z}\mathscr{D}}$. This model possesses only m degrees of freedom, meaning that only m linearly independent functions can be drawn from the prior. This constraint underscores the finite nature of the model's flexibility, ensuring that the additional functions do not introduce new independent variations but rather depend on the established set of m functions. In an intrinsic GP, Σ_{zz}, $\Sigma_{\mathscr{D}\mathrm{z}}$, and Σ_{zr} can all be obtained by evaluating the transition densities from the BM simulations with inducing points as the starting points (due to its symmetric properties,

$\Sigma_{\mathscr{D}\mathrm{z}} = \Sigma_{\mathrm{z}\mathscr{D}}$). The number of inducing points (m) is significantly smaller than the number of grid data points (G'). As a result, BM paths only need to be simulated from the inducing points in advance, rather than from every data point, which greatly reduces computational effort.

As defined earlier in Chapter 3, $\mathbf{y}$ represents the observations of the objective function in the training set $\mathscr{D}$. Using this approximation, the marginal likelihood can be expressed as:

$$p(\mathbf{y}) \approx q(\mathbf{y}) = \mathscr{N}(0, \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{z}\mathscr{D}} + \sigma_n^2\mathbf{I}),$$

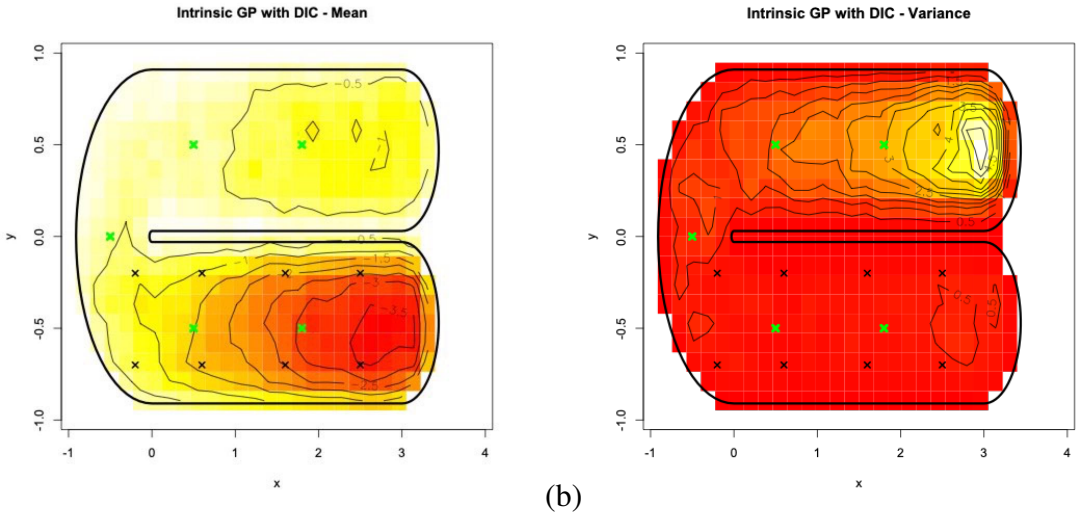
where σ_n^2 represents the noise variance. The diffusion time t and the magnitude parameter σ_h^2 (introduced in Equation (3.8)) can be optimised as the hyperparameter of the kernel by maximising this approximate likelihood. The predictive distribution is shown below:

$$\begin{aligned} q(\mathbf{f}_r \mid \mathbf{y}) &= \mathscr{N}\left(Q_{\mathrm{r}\mathscr{D}}\left(Q_{\mathscr{D}\mathscr{D}} + \sigma_n^2\mathbf{I}\right)^{-1}\mathbf{y}\,,\, Q_{\mathrm{rr}} - Q_{\mathrm{r}\mathscr{D}}\left(Q_{\mathscr{D}\mathscr{D}} + \sigma_n^2\mathbf{I}\right)^{-1}Q_{\mathscr{D}\mathrm{r}}\right) \\ &= \mathscr{N}\left(\sigma_n^{-2}\Sigma_{\mathrm{rz}}\mathrm{K}\Sigma_{\mathrm{z}\mathscr{D}}\mathbf{y}, \Sigma_{\mathrm{rz}}\mathrm{K}\Sigma_{\mathrm{zr}}\right), \end{aligned} \tag{4.6}$$

where $\mathrm{K} = (\sigma_n^{-2}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}} + \Sigma_{\mathrm{zz}})^{-1}$. The first line of Equation (4.6) can be seen as a transformation of Equation (3.9), where the covariance matrix Σ has been replaced everywhere by Q, which also reflects the relative change in the GP approximate joint prior compared to Chapter 3. While the second line of Equation (4.6) is computationally cheaper, based on Equation (4.3). Here, K can be viewed as the covariance of the posterior distribution of the weights $\mathbf{w}_{\mathrm{z}}$. By using the inducing points, the complexity of inverting the covariance matrix is decreased from $O(n^3)$ to $O(nm^2)$, where $m << n$. The application of the SI-GPDIC method on manifolds, along with comparisons to other methods, will be discussed in detail in Chapter 6.

Although the DIC method offers computational efficiency, it comes with certain limitations. Due to the degeneracy of the prior, the predictive distributions present inaccurate results. As noted by Liu et al. [94], for covariance functions that diminish to zero for pairs of distant inputs, Q approaches zero, specifically, as test points move further from the inducing points. As a result, the predictive variance approaches zero as the test points move further away from the inducing inputs. This is unrealistic, as the predictive variance represents uncertainty; it should increase, reflecting greater uncertainty when moving away from the known inducing and training points, rather than converging to zero. Quionero-Candela et al. [126] demonstrate this limitation of

the DIC method in Euclidean space, while Liu et al. [94] provide evidence of this issue on manifolds. By applying the SI-GPDIC on the U-shaped domain (introduced in Section 1.3.1), this issue is illustrated through a set of examples, as shown in Figure 4.2. On the U-shape domain, 5 inducing points, shown as green crosses in Figure 4.2, are symmetrically spaced within the boundary, while eight training points are positioned along the lower half of the U-shape, marked as black crosses. Figure 4.2(a) presents the corresponding predictive mean, while Figure 4.2(b) displays the predictive variance. Both are calculated using Equation (3.9). Figure 4.2(b) proves that in the upper-right region of the U-shape, the variance decreases to zero as the distance from the inducing points increases.



(a) (b)

Figure 4.2: SI-GPDIC on the U-shape, with 8 training points (black crosses) and 5 inducing points (green crosses): (a) the predictive mean on the U-shape domain; (b) the predictive variance on the U-shape domain.

To address the problem of inaccurate variance estimation, a new sparse method called DTC is considered. Proposed by Seeger et al. (2003), the DTC does not suffer from the nonsensical predictive uncertainties associated with the DIC approximation [86]. This method improves the reliability of predictive variances and offers a more robust solution for handling sparse approximations in large datasets.

4.3 Sparse Intrinsic Gaussian Process with DTC

This section introduces the Sparse Intrinsic Gaussian Process with DTC (SI-GPDTC) to address the issues identified in the previously discussed SI-GPDIC method. The DTC approach, also named as Projected Latent Variables (PLV), is based on the projection from $\mathbf{f}_{\mathscr{D}}$ to u using Equation (4.2). The corresponding likelihood approximation is given by:

$$p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) \simeq q(\mathbf{y} \mid \mathbf{u}) = \mathscr{N}\left(\Sigma_{\mathscr{D}\mathrm{z}}(\Sigma_{\mathrm{zz}})^{-1}u, \sigma_{\mathrm{n}}^2\mathbf{I}\right),$$

where σ_{n}^2 represents the noise variance and $\Sigma_{\mathscr{D}\mathrm{z}}$ is the covariance matrix between training points $\mathscr{D}$ and inducing points z. The DTC method retains the same likelihood as the DIC but applies a deterministic training conditional and an exact test conditional, referred to as $q(\mathbf{f}_{\mathscr{D}}|u)$ and $q(\mathbf{f}_r|u)$ respectively, and is characterized by:

$$\begin{aligned} q(\mathbf{f}_{\mathscr{D}}|u) &= \mathscr{N}\left(\Sigma_{\mathscr{D}\mathrm{z}}(\Sigma_{\mathrm{zz}})^{-1}u, 0\right), \\ q(\mathbf{f}_r|u) &= p(\mathbf{f}_r|u), \end{aligned}$$

where $q(\mathbf{f}_r|u)$ is exact, instead of the deterministic relation between $\mathbf{f}_r$ and u of the DIC method, as shown in Equation (4.5). This is also why it is named the Deterministic Training Conditional. This reformulation allows for maintaining strict requirements for the model, specifically exact inference and exact likelihood functions, while employing approximate methods in other areas, namely for the prior distribution. The advantage of this approach lies in its ability to simplify the computational complexity without sacrificing inference accuracy. This method achieves a balance between computational efficiency and inference precision. The joint prior implied by the DTC is given by:

$$\begin{aligned} q\left(\mathbf{f}_{\mathscr{D}}, \mathbf{f}_r\right) &= \mathscr{N}\left(0, \begin{bmatrix} \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{z}\mathscr{D}} & \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}} \\ \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}^{-1}}\Sigma_{\mathrm{z}\mathscr{D}} & \Sigma_{\mathrm{rr}} \end{bmatrix}\right) \\ &= \mathscr{N}\left(0, \begin{bmatrix} Q_{\mathscr{D}\mathscr{D}} & Q_{\mathscr{D}\mathrm{r}.} \\ Q_{\mathrm{r}\mathscr{D}} & \Sigma_{\mathrm{rr}} \end{bmatrix}\right), \end{aligned}$$

which has the closed form of the effective prior implied by the DIC method. Under the DTC approximation, the key difference is that f_r possesses its own prior variance, denoted by Σ_{rr}. This

prior variance corrects the behavior of the predictive uncertainties, making them reasonable and reliable. Then, the predictive distribution is now given by:

$$\begin{aligned} q(\mathbf{f}_r \mid \mathbf{y}) &= \mathscr{N}\left(Q_{\mathrm{r}\mathscr{D}}\left(Q_{\mathscr{D}\mathscr{D}}+\sigma_n^2\mathbf{I}\right)^{-1}\mathbf{y}\,,\,\Sigma_{\mathrm{rr}}-Q_{\mathrm{r}\mathscr{D}}\left(Q_{\mathscr{D}\mathscr{D}}+\sigma_n^2\mathbf{I}\right)^{-1}Q_{\mathscr{D}\mathrm{r}}\right) \\ &= \mathscr{N}\left(\sigma_n^{-2}\Sigma_{\mathrm{rz}}\mathrm{K}\Sigma_{\mathrm{z}\mathscr{D}}\mathbf{y},(\Sigma_{\mathrm{rr}}-Q_{\mathrm{rr}})+\Sigma_{\mathrm{rz}}\mathrm{K}\Sigma_{\mathrm{zr}}\right), \end{aligned} \tag{4.7}$$

where $\mathrm{K}=(\sigma_n^{-2}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}}+\Sigma_{\mathrm{zz}})^{-1}$. The predictive mean of the DTC method is the same as that of the DIC method, while for the predictive variance, in the first line of Equation (4.7), Q_{rr} is replaced with Σ_{rr}. This replacement occurs because f_r has its exact prior variance Σ_{rr}, which also corresponds to the difference in the joint distribution between the DIC method and the DTC method. In the second line of Equation (4.7), an additional term $(\Sigma_{\mathrm{rr}}-Q_{\mathrm{rr}})$ is included. Σ_{rr} is the full covariance matrix between all the testing points S, containing all possible information, while Q_{rr} is the covariance matrix approximated using the information from the inducing points z, actually underestimating the total variance because it does not account for all information. This ensures that $(\Sigma_{\mathrm{rr}}-Q_{\mathrm{rr}})$ is always positive semidefinite. This additional term increases to the Σ_{rr}, the exact prior, as the testing points move farther from the inducing inputs z, due to Σ_{rz} and Σ_{zr} in $Q_{\mathrm{rr}}=\Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}}$ tending to zero. The presence of this term effectively resolves the variance estimation issues present in the DIC method.

4.3.1 How to Compute Σ_{rr}

To use the DTC method in an Intrinsic GP, another issue that needs to be addressed is how to compute Σ_{rr}. The computation of Σ_{rr} requires simulating BM paths from each testing point. As previously discussed, one of the reasons for using the sparse method and introducing inducing points is to avoid the computational complexity encountered in simulating BM paths. The additional term $(\Sigma_{\mathrm{rr}}-Q_{\mathrm{rr}})$ in predictive variance ensures that the variance accurately reflects uncertainty when the testing points are far from both the inducing points and training points, preventing it from decreasing to zero as the distance increases. Given that Σ_{rr} cannot be directly computed, it is necessary to approximate Σ_{rr} in a way that ensures this property continues to hold.

The predictive variance considers only the diagonal elements of the covariance matrix. Here,

Σ_{rr} can be approximated as the maximum value of the diagonal elements in Σ_{zz}, which can be expressed as:

$$\Sigma^*_{\text{rr}} = \max(\text{diag}(\Sigma_{\text{zz}}))\mathbf{I}.$$

To ensure the additional term is positive definite, the absolute value is taken during the calculation process. Then, the predictive variance shown in Equation (4.7) can be rewritten as:

$$\sigma_{DTC} = \max\left[\text{diag}(\Sigma^*_{\text{rr}}) - \text{diag}(Q_{\text{rr}}), 0\right] + \text{diag}(\Sigma_{\text{rz}}\text{K}\Sigma_{\text{zr}}),$$

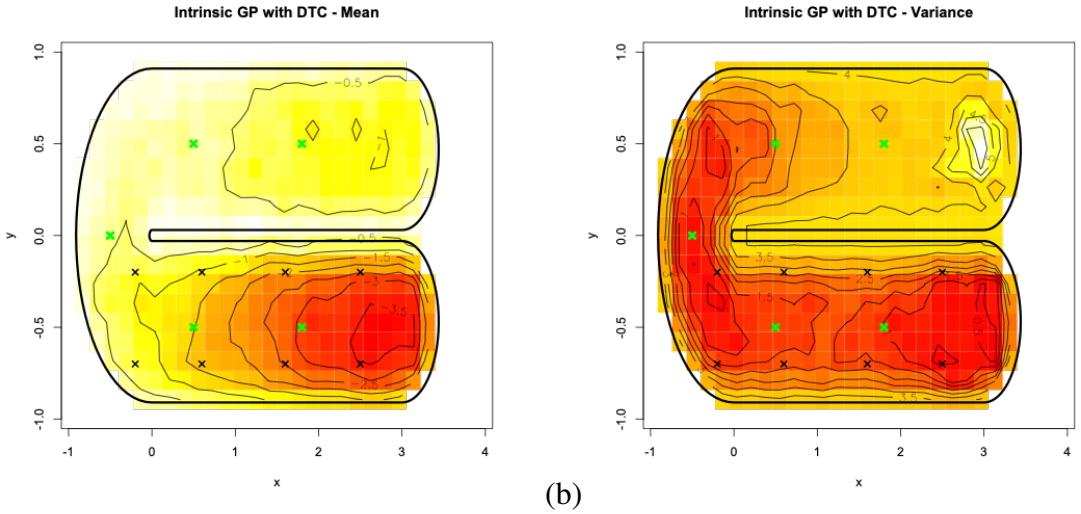
where $Q_{\text{rr}} = \Sigma_{\text{rz}}\Sigma_{\text{zz}}^{-1}\Sigma_{\text{zr}}$ and $\text{K} = (\sigma_n^{-2}\Sigma_{\text{z}\mathscr{D}}\Sigma_{\mathscr{D}\text{z}} + \Sigma_{\text{zz}})^{-1}$. The use of $\max()$ ensures that the added term is positive definite.

4.3.2 How to Select Inducing Points

In sparse intrinsic GPs, BM paths need to be simulated from inducing points in advance. This constraint prevents treating the position and number of inducing points as tunable parameters in each experiment. Thus, the inducing points z used in this work are predefined and approximately uniformly distributed across the manifold. For different manifolds, both the location and number of inducing points need to be adjusted accordingly. The preference for a uniform distribution ensures that the BM paths originating from these points can comprehensively explore the entire domain of the manifold, thereby capturing its geometric structure as completely as possible for later predictions. The inducing variables u induces the dependencies between the training cases $\mathbf{f}_{\mathscr{D}}$ and testing cases $\mathbf{f}_r$. As the number of inducing points increases, the information provided becomes more comprehensive, leading to improved accuracy in the approximations obtained by methods both DIC and DTC. However, this also results in a higher computational burden. Therefore, the number of inducing points must be carefully chosen to balance accuracy and computational efficiency. The focus of this study is primarily on comparing the performance of different sparse intrinsic GP methods under the same number of inducing points, rather than specifically optimizing the number of inducing points. In the later chapters, the impact of different numbers of inducing points on each method will also be discussed.

The application and comparison of SI-GPDTC on manifolds will be detailed in Chapter 6. This section highlights the improvement in predictive variance achieved by the DTC method

compared to the DIC method, demonstrated using the same U-shape example, as shown in Figure 4.3. The U-shape setup has been introduced in Section 4.2. Figure 4.3(a) shows the predictive mean calculated using the DTC method, while Figure 4.3(b) illustrates the corresponding predictive variance. Under the DTC method, the predictive mean remains the same as in the DIC method. However, it is clear that the DTC method resolves the variance issues observed in DIC. When the test points are far from the inducing points, where information is known, the variance—representing uncertainty—remains appropriately large and does not drop to zero.



(a) (b)

Figure 4.3: SI-GPDTC on the U-shape, with 8 training points (black crosses) and 5 inducing points (green crosses) (a) the predictive mean on the U-shape domain; (b) the predictive variance on the U-shape domain.

Currently, both SI-GPDIC and SI-GPDTC use inducing points to approximate the covariance matrix of a GP, thus making the inference process more manageable. The modified model produced by each of them is not a strictly approximate procedure because there is no minimum distance between the exact model p and the modified model q. This means the approximation is not optimised based on a specific criterion, such as a distance measure between probability distributions. The next section proposes the SI-GPVI, which utilizes VI, approaching the problem from a new perspective. It cleverly transforms the posterior inference problem into an optimisation problem. The core idea of the VI is to introduce a metric, the Kullback-Leibler (KL) divergence, to measure the difference between the exact posterior p and the modified model q. By minimizing this divergence, the model is optimised, offering a more theoretically rigorous approximation.

4.4 Sparse Intrinsic Gaussian Process with Variational Inference

VI is a major category of methods in Bayesian approximate inference, offering improved convergence and scalability [71], [156]. It has been applied to various fields, such as computational biology, where probabilistic models provide important building blocks for analyzing genetic data [96], [127], [141]. VI has been important to computer vision and and robotics [14], [91], [145]. There have been many applications of variational inference to neuroscience, especially for autoregressive processes [122], [123], hierarchical models of multiple subjects [169] and brain-computer interfaces [147], among others.

Instead of relying on exact or sampling-based methods, VI approximates the true posterior distribution with a simpler, more tractable distribution by minimizing the divergence metric between them. This makes it possible to perform efficient inference in high-dimensional and complex models where traditional methods may struggle due to computational constraints. Figure 4.4 provides a simple example where the purple shaded area represents the original target distribution p, resembling a GP. The blue and green lines represent the distributions q_1 and q_2, derived from Gaussian distributions. The overlap between each q and p is calculated, and by minimizing the uncovered areas—i.e., minimizing the divergence metric—the distribution q_2 is selected as the approximate distribution for p. VI offers a flexible and scalable approach by allowing the selection of simpler distributions to closely approximate the true posterior. This is particularly beneficial in large-scale applications where the data volume and model complexity make exact inference infeasible.

This section proposes the novel SI-GPVI on manifolds. Instead of modifying the exact GP model, it minimizes the distance between the exact posterior GP and the variational approximation. The inducing points now become variational parameters [150]. Based on the principles of VI, the process begins without modifying the GP prior and demonstrates how to directly approximate the posterior GP mean and covariance functions. During this process, the variational parameters to be optimized are defined. Subsequently, the method for optimising hyperparameters is derived by maximizing a lower bound to the exact marginal likelihood, achieved by minimizing the KL divergence.

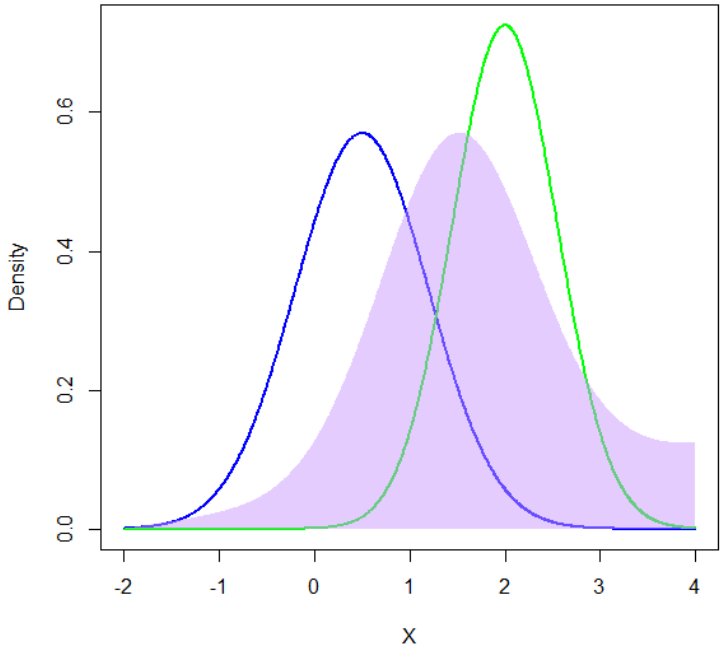


Figure 4.4: The graphical model of the original target distribution p, the distributions q_1 and q_2 used to approximate p: the purple shaded area represents p, the blue and green lines represent the distributions q_1 and q_2 respectively.

4.4.1 The Approximate Posterior Intrinsic GPs

The posterior intrinsic GP uses an approximate heat kernel constructed from the transition density of the BM paths. It can be described by the predictive Gaussian distribution:

$$p(\mathbf{f}_r \mid \mathbf{y}) = \int p(\mathbf{f}_r \mid \mathbf{f}_{\mathscr{D}}) p(\mathbf{f}_{\mathscr{D}} \mid \mathbf{y}) d\mathbf{f}_{\mathscr{D}},$$

where $p(\mathbf{f}_r \mid \mathbf{f}_{\mathscr{D}})$ is the conditional intrinsic GP prior while $p(\mathbf{f}_{\mathscr{D}} \mid \mathbf{y})$ is the posterior distribution over the training function value y (where $y = \mathbf{f}_{\mathscr{D}} + \varepsilon$ as defined in Section 3.2). When introducing inducing points z and their corresponding function values u (the inducing points dataset z is independent of the training points dataset $\mathscr{D}$), in the augmented probability space, this helps decompose the complex posterior distribution into a more manageable form:

$$p(\mathbf{f}_r \mid \mathbf{y}) = \int p(\mathbf{f}_r \mid u, \mathbf{f}_{\mathscr{D}}) \, p(\mathbf{f}_{\mathscr{D}} \mid u, \mathbf{y}) \, p(u \mid \mathbf{y}) \, d\mathbf{f}_{\mathscr{D}} du.$$

Assume that given u, all information contained in the function values $\mathbf{f}_{\mathscr{D}}$ is fully captured, and any additional information from $\mathbf{f}_{\mathscr{D}}$ will not affect other variables $\mathbf{f}_r$. In other words, $\mathbf{f}_r$ and $\mathbf{f}_{\mathscr{D}}$ are independent given u. From that, $p(\mathbf{f}_r \mid u, \mathbf{f}_{\mathscr{D}}) = p(\mathbf{f}_r \mid u)$. Since y is the noisy version of $\mathbf{f}_{\mathscr{D}}$, $p(\mathbf{f}_{\mathscr{D}} \mid u, \mathbf{y}) = p(\mathbf{f}_{\mathscr{D}} \mid u)$. To derive the mean and covariance functions of $p(\mathbf{f}_r \mid \mathbf{y})$, first regard the following:

$$p(\mathbf{f}_r, u) = \mathscr{N}\left(0, \begin{bmatrix} \Sigma_{\mathrm{rr}} & \Sigma_{\mathrm{rz}} \\ \Sigma_{\mathrm{zr}} & \Sigma_{\mathrm{zz}} \end{bmatrix}\right), \tag{4.8}$$

$$p(\mathbf{f}_r, \mathbf{f}_{\mathscr{D}}, u) = \mathscr{N}\left(0, \begin{bmatrix} \Sigma_{r\mathscr{D}} & \Sigma_{rr} & \Sigma_{rz} \\ \Sigma_{\mathscr{D}\mathscr{D}} & \Sigma_{\mathscr{D}r} & \Sigma_{\mathscr{D}\mathrm{z}} \\ \Sigma_{\mathrm{z}\mathscr{D}} & \Sigma_{zr} & \Sigma_{\mathrm{zz}} \end{bmatrix}\right), \tag{4.9}$$

then, the conditional distribution of $\mathbf{f}_r$ given u can be derived as follows:

$$p(\mathbf{f}_r \mid u) = \mathscr{N}\left(\Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}u, \Sigma_{\mathrm{rr}} - \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}}\right).$$

The conditional covariance of $\mathbf{f}_r$ and $\mathbf{f}_{\mathscr{D}}$ given u is:

$$\mathrm{Cov}[\mathbf{f}_{\mathscr{D}}, \mathbf{f}_r \mid u] = \Sigma_{\mathscr{D}\mathrm{r}} - \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}}. \tag{4.10}$$

Here, let $\phi(u)$ be a free variational Gaussian distribution with its mean and variance being, respectively, μ and A. Based on this, the mean functions of the approximate posterior Intrinsic GPs is obtained:

$$\mathbb{E}[\mathbf{f}_r \mid y] = \mathbb{E}_{\phi(u)}\left[\mathbb{E}_{p(\mathbf{f}_{\mathscr{D}}|u,\mathbf{y})}[\mathbf{f}_r|u, \mathbf{f}_{\mathscr{D}}]\right] = \mathbb{E}_{\phi(u)}[\mathbb{E}_{p(\mathbf{f}_{\mathscr{D}}|u)}[\mathbf{f}_r \mid u]] = \mathbb{E}_{\phi(u)}[\Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}u],$$

where $\phi(u) \sim \mathscr{N}(\mu, A)$ and u is a sufficient statistic for the vector $\mathbf{f}_r$ and $\mathbf{f}_{\mathscr{D}}$, allowing $\mathbb{E}[\mathbf{f}_r \mid u, \mathbf{f}_{\mathscr{D}}] = \mathbb{E}[\mathbf{f}_r \mid u]$. The conditional covariance of $\mathbf{f}_{\mathscr{D}}$ and $\mathbf{f}_r$ given u has already been provided by Equation (4.10). When considering the variational distribution $\phi(u)$, the conditional covariance needs to incorporate the covariance of the variational distribution, using the law of total covariance [129]:

$$\mathrm{Cov}[\mathbf{f}_{\mathscr{D}}, \mathbf{f}_r] = \mathbb{E}_{\phi(u)}[\mathrm{Cov}[\mathbf{f}_{\mathscr{D}}, \mathbf{f}_r \mid u]] + \mathrm{Cov}_{\phi(u)}[\mathbb{E}[\mathbf{f}_{\mathscr{D}} \mid u], \mathbb{E}[\mathbf{f}_r \mid u]],$$

among it, the first part expands as follows:

$$\mathbb{E}_{\phi(u)}[\mathrm{Cov}[\mathbf{f}_{\mathscr{D}}, \mathbf{f}_r \mid u]] = \Sigma_{\mathscr{D}\mathrm{r}} - \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}}.$$

The second part expands to:

$$\mathrm{Cov}_{\phi(u)}[\mathbb{E}[\mathbf{f}_{\mathscr{D}} \mid u], \mathbb{E}[\mathbf{f}_r \mid u]] = \mathrm{Cov}_{\phi(u)}[\Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}u, \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}u],$$

which since $\mathbb{E}[\mathbf{f}_{\mathscr{D}} \mid u] = \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}u$, then

$$\mathrm{Cov}_{\phi(u)}[\Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}u, \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}u] = \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}\mathrm{Cov}_{\phi(u)}[u]\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{rz}}.$$

Given $\phi(u) \sim \mathscr{N}(\mu, A)$, which is predefined, the predictive mean function and covariance function of the SI-GPVI can be expressed as:

$$\begin{aligned} \mu_{VI}(S) &= \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}\mu, \\ \mathrm{Cov}(\mathscr{D}, S) &= \Sigma_{\mathscr{D}\mathrm{r}} - \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}} + \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}A\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{rz}}, \end{aligned} \tag{4.11}$$

where the predictive variance function of $p(\mathbf{f}_r \mid \mathbf{y})$ takes the form of:

$$\begin{aligned} \sigma_{VI}(S) &= \Sigma_{\mathrm{rr}} - \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}} + \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}A\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{rz}} \\ &= (\Sigma_{\mathrm{rr}} - Q_{\mathrm{rr}}) + \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}A\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{rz}}, \end{aligned} \tag{4.12}$$

where $Q_{\mathrm{rr}} = \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}}$. In many applications, the focus is solely on the variance at each grid point. In this research, the predictive variance of each grid point can be simplified to:

$$\sigma_{VI}(S) = \mathrm{diag}(\Sigma_{\mathrm{rr}}) - \mathrm{diag}(Q_{\mathrm{rr}}) + \mathrm{diag}(\Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}A\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{rz}}).$$

In the Intrinsic GP, directly calculating Σ_{rr} would require simulating BM paths from each point in the grid points dataset S leading to significant computational cost. To circumvent the need of simulating BM paths from S, an approximate $\mathrm{diag}(\Sigma_{\mathrm{rr}})$ is constructed by:

$$\mathrm{diag}(\Sigma_{\mathrm{rr}}^{*}) = \max(\mathrm{diag}(\Sigma_{\mathrm{zz}})). \tag{4.13}$$

As a special type of manifold, the Euclidean space allows the approximation $\mathrm{diag}(\Sigma^*_{\mathrm{rr}})$ to be the same as $\mathrm{diag}(\Sigma_{\mathrm{rr}})$. To ensure the positive definiteness of the covariance matrix, the predictive variance function can be replaced with:

$$\sigma_{VI}(S) = \max\left[\mathrm{diag}(\Sigma^*_{\mathrm{rr}}) - \mathrm{diag}(Q_{\mathrm{rr}}), 0\right] + \mathrm{diag}(\Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1} A \Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{rz}}). \tag{4.14}$$

With the basic form of the sparse posterior intrinsic GP established, the next step is to determine how to choose μ and A of the $\phi(u)$. In the following section, the variational distribution $q(\mathbf{f}_r, u)$ proposed will be used to approximate the exact posterior distribution $p(\mathbf{f}_r \mid y)$. After deriving the lower bound on the exact log marginal likelihood, model hyperparameters (t, σ_h^2) will be optimised by maximizing it. Also, $\phi(u)$ can be optimised through these processes.

4.4.2 Variational lower bound

The variational lower bound provides a clear optimisation objective. By maximizing the variational lower bound, the variational distribution can be made as close as possible to the true posterior distribution. This section will demonstrate the derivation and construction of the variational lower bound. In an augmented probability space which contains $\mathbf{f}_{\mathscr{D}}, \mathbf{f}_r$ and inducing variables u, the initial joint model $p(y, \mathbf{f}_r)$ is extended by incorporating the variables u, resulting in the augmented model:

$$p\left(\mathbf{y}, \mathbf{f}_r, u\right) = p(\mathbf{y} \mid \mathbf{f}_r) p\left(\mathbf{f}_r \mid u\right) p\left(u\right),$$

where $p\left(\mathbf{f}_r \mid u\right) = \mathscr{N}\left(\mathbf{f}_r \mid \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1} u, \Sigma_{\mathrm{rr}} - \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}}\right)$ represents the conditional intrinsic GP prior. The values of the hyperparameters (t, σ_h^2) for the posterior intrinsic GP can be estimated by maximizing the log marginal likelihood, which is given by:

$$\log p(\mathbf{y}) = \log[\mathscr{N}(\mathbf{y} \mid 0, \sigma^2\mathbf{I} + \Sigma_{\mathscr{D}\mathscr{D}})]. \tag{4.15}$$

According to the marginalization property of Intrinsic GPs, Equation (4.15) also takes another form:

$$\log p(\mathbf{y}) = \log \int p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) p\left(\mathbf{f}_{\mathscr{D}} \mid u\right) p\left(u\right) d\mathbf{f}_{\mathscr{D}} du.$$

Instead of directly calculating the true posterior distribution $p(\mathbf{f}_{\mathscr{D}}, u \mid y)$, a parametric model is introduced to approximate the complex posterior distribution with a simpler distribution $q(\mathbf{f}_{\mathscr{D}}, u)$. Transforming complex posterior inference problems into optimisation problems requires only minimizing the distance between the approximate distribution and the true distribution to solve for the parameters, thereby enabling efficient computation. Then, the KL divergence is used to measure the similarity between these two distributions. The KL divergence is a measure from information theory that measures the amount of information lost when $q(\mathbf{f}_{\mathscr{D}}, u)$ is used to approximate $p(\mathbf{f}_{\mathscr{D}}, u \mid y)$ [61]. The formula for KL divergence is shown as:

$$\mathrm{KL}\left(q(\mathbf{f}_{\mathscr{D}}, u) \| p(\mathbf{f}_{\mathscr{D}}, u \mid y)\right) = \int q(\mathbf{f}_{\mathscr{D}}, u) \log \frac{q(\mathbf{f}_{\mathscr{D}}, u)}{p(\mathbf{f}_{\mathscr{D}}, u \mid y)} d\mathbf{f}_{\mathscr{D}} du, \tag{4.16}$$

where $q\left(\mathbf{f}_{\mathscr{D}}, u\right)$ defined as $q\left(\mathbf{f}_{\mathscr{D}}, u\right) = p\left(\mathbf{f}_{\mathscr{D}} \mid u\right) \phi\left(u\right)$. $\phi\left(u\right)$ is the unconstrained variational distribution over u defined before and $p\left(\mathbf{f}_{\mathscr{D}} \mid u\right)$ is the conditional GP prior. The $p(\mathbf{f}_{\mathscr{D}}, u \mid y)$ can be expanded using Bayesian methods as follows:

$$p(\mathbf{f}_{\mathscr{D}}, u \mid y) = \frac{p\left(y \mid \mathbf{f}_{\mathscr{D}}, u\right) p\left(\mathbf{f}_{\mathscr{D}}, u\right)}{p(y)} = \frac{p(y \mid \mathbf{f}_{\mathscr{D}}) p\left(\mathbf{f}_{\mathscr{D}} \mid u\right) p\left(u\right)}{p(y)}. \tag{4.17}$$

Combining with Equation (4.17), the KL divergence shown in Equation (4.16) can be decomposed as:

$$\begin{aligned}
\mathrm{KL}\left(q(\mathbf{f}_{\mathscr{D}}, u) \| p(\mathbf{f}_{\mathscr{D}}, u \mid y)\right) &= \int q(\mathbf{f}_{\mathscr{D}}, u) \log \frac{q(\mathbf{f}_{\mathscr{D}}, u) p(y)}{p(y \mid \mathbf{f}_{\mathscr{D}}) p\left(\mathbf{f}_{\mathscr{D}} \mid u\right) p\left(u\right)} d\mathbf{f}_{\mathscr{D}} du \\
&= -\int q(\mathbf{f}_{\mathscr{D}}, u) \log \frac{p(y \mid \mathbf{f}_{\mathscr{D}}) p\left(\mathbf{f}_{\mathscr{D}} \mid u\right) p\left(u\right)}{q(\mathbf{f}_{\mathscr{D}}, u) p(y)} d\mathbf{f}_{\mathscr{D}} du \\
&= -\int q(\mathbf{f}_{\mathscr{D}}, u) \log \frac{p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) p\left(\mathbf{f}_{\mathscr{D}} \mid u\right) p\left(u\right)}{q(\mathbf{f}_{\mathscr{D}}, u)} d\mathbf{f}_{\mathscr{D}} du + \int q(\mathbf{f}_{\mathscr{D}}, u) \log p(\mathbf{y}) d\mathbf{f}_{\mathscr{D}} du \\
&= -F_V(\phi) + \log p(\mathbf{y}).
\end{aligned}$$

Then,

$$\log p(\mathbf{y}) = \mathrm{KL}\left(q(\mathbf{f}_{\mathscr{D}}, u) \| p(\mathbf{f}_{\mathscr{D}}, u \mid y)\right) + F_V(\phi).$$

Minimizing the KL divergence is equivalently expressed as the maximization of $F_V(\phi)$. Due to the non-negativity of the KL divergence, it always holds that $\log p(\mathbf{y}) \geq F_V(\phi)$. Thus, $F_V(\phi)$ can be regarded as the variational lower bound on the true log marginal likelihood. The vari-

ational lower bound can also be derived directly from the true log marginal likelihood. In this approach, Jensen's inequality $\log(\mathbb{E}[x]) \geq \mathbb{E}[\log(x)]$ is utilized to find the lower bound:

$$\begin{aligned}
\log p(\mathbf{y}) &= \log \int p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) p(\mathbf{f}_{\mathscr{D}} \mid u)\, p(u)\, d\mathbf{f}_{\mathscr{D}} du \\
&= \log \int q(\mathbf{f}_{\mathscr{D}}, u) \frac{p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) p(\mathbf{f}_{\mathscr{D}} \mid u)\, p(u)}{q(\mathbf{f}_{\mathscr{D}}, u)} d\mathbf{f}_{\mathscr{D}} du \\
&\geq \int q(\mathbf{f}_{\mathscr{D}}, u) \log \frac{p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) p(\mathbf{f}_{\mathscr{D}} \mid u)\, p(u)}{q(\mathbf{f}_{\mathscr{D}}, u)} d\mathbf{f}_{\mathscr{D}} du \\
&= F_V(\phi).
\end{aligned}$$

Substituting the decomposed form of $q(\mathbf{f}_{\mathscr{D}}, u)$ into it, $F_V(\phi)$ can be expressed as:

$$\begin{aligned}
F_V(\phi) &= \int p(\mathbf{f}_{\mathscr{D}}|u)\phi(u) \log \frac{p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) p(\mathbf{f}_{\mathscr{D}} \mid u)\, p(u)}{p(\mathbf{f}_{\mathscr{D}}|u)\phi(u)} d\mathbf{f}_{\mathscr{D}} du \\
&= \int p(\mathbf{f}_{\mathscr{D}}|u)\phi(u) \log \frac{p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) p(u)}{\phi(u)} d\mathbf{f}_{\mathscr{D}} du \\
&= \int p(\mathbf{f}_{\mathscr{D}}|u)\phi(u) \left\{ \log p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) + \log \frac{p(u)}{\phi(u)} \right\} d\mathbf{f}_{\mathscr{D}} du \\
&= \int \phi(u) \left\{ \int p(\mathbf{f}_{\mathscr{D}}|u) \log p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) d\mathbf{f}_{\mathscr{D}} + \log \frac{p(u)}{\phi(u)} \right\} du.
\end{aligned} \tag{4.18}$$

Among it, $\log p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}})$ is expanded based on the form of the Gaussian distribution:

$$\begin{aligned}
\log p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}}) &= \log \left\{ \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2} \|y - \mathbf{f}_{\mathscr{D}}\|^2 \right) \right\} \\
&= \log \left\{ \frac{1}{(2\pi\sigma^2)^{n/2}} \right\} + \log \left\{ \exp\left(-\frac{1}{2\sigma^2} \|y - \mathbf{f}_{\mathscr{D}}\|^2 \right) \right\} \\
&= -\frac{n}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \|y - \mathbf{f}_{\mathscr{D}}\|^2 \\
&= -\frac{n}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \operatorname{Tr}\left[\mathbf{y}\mathbf{y}^T - 2\mathbf{y}\mathbf{f}_{\mathscr{D}}{}^T + \mathbf{f}_{\mathscr{D}}\mathbf{f}_{\mathscr{D}}{}^T \right].
\end{aligned} \tag{4.19}$$

By combining this expansion, the first term in the parentheses can be expressed as:

$$\begin{aligned}\int p(\mathbf{f}_{\mathscr{D}}|u)\log p(\mathbf{y}\mid\mathbf{f}_{\mathscr{D}})d\mathbf{f}_{\mathscr{D}} &= \int p(\mathbf{f}_{\mathscr{D}}|u)\left\{-\frac{n}{2}\log(2\pi\sigma^2)-\frac{1}{2\sigma^2}\operatorname{Tr}\left[\mathbf{y}\mathbf{y}^T-2\mathbf{y}\mathbf{f}_{\mathscr{D}}{}^T+\mathbf{f}_{\mathscr{D}}\mathbf{f}_{\mathscr{D}}{}^T\right]\right\}d\mathbf{f}_{\mathscr{D}}\\ &= -\frac{n}{2}\log(2\pi\sigma^2)\int p(\mathbf{f}_{\mathscr{D}}|u)d\mathbf{f}_{\mathscr{D}}\\ &-\frac{1}{2\sigma^2}\int p(\mathbf{f}_{\mathscr{D}}|u)\operatorname{Tr}\left[\mathbf{y}\mathbf{y}^T-2\mathbf{y}\mathbf{f}_{\mathscr{D}}{}^T+\mathbf{f}_{\mathscr{D}}\mathbf{f}_{\mathscr{D}}{}^T\right]d\mathbf{f}_{\mathscr{D}}\\ &= -\frac{n}{2}\log(2\pi\sigma^2)-\\ &\frac{1}{2\sigma^2}\left\{\operatorname{Tr}(yy^T)-2\operatorname{Tr}\left[y\int p(\mathbf{f}_{\mathscr{D}}|u)\mathbf{f}_{\mathscr{D}}^T d\mathbf{f}_{\mathscr{D}}\right]+\operatorname{Tr}\left[\int p(\mathbf{f}_{\mathscr{D}}|u)\mathbf{f}_{\mathscr{D}}\mathbf{f}_{\mathscr{D}}^T d\mathbf{f}_{\mathscr{D}}\right]\right\},\end{aligned} \tag{4.20}$$

among it, setting

$$\alpha = \mathbb{E}[\mathbf{f}_{\mathscr{D}}|u] = \int p(\mathbf{f}_{\mathscr{D}}|u)\mathbf{f}_{\mathscr{D}}d\mathbf{f}_{\mathscr{D}} = \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}u,$$

$$\sigma(\mathbf{f}_{\mathscr{D}}|u) = \operatorname{Cov}[\mathbf{f}_{\mathscr{D}},\mathbf{f}_{\mathscr{D}}\mid u] = \Sigma_{\mathscr{D}\mathscr{D}}-Q_{\mathscr{D}\mathscr{D}},$$

where $Q_{\mathscr{D}\mathscr{D}} = \Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{z}\mathscr{D}}$ has already been defined in the previous section. Then, each part of Equation (4.20) can be simplified as follows:

$$-2\operatorname{Tr}\left[y\int p(\mathbf{f}_{\mathscr{D}}|u)\mathbf{f}_{\mathscr{D}}^T d\mathbf{f}_{\mathscr{D}}\right] = -2\operatorname{Tr}\left[y\alpha^T\right],$$

$$\begin{aligned}\operatorname{Tr}\left[\int p(\mathbf{f}_{\mathscr{D}}|u)\mathbf{f}_{\mathscr{D}}\mathbf{f}_{\mathscr{D}}^T d\mathbf{f}_{\mathscr{D}}\right] = \operatorname{Tr}\left[\mathbb{E}[\mathbf{f}_{\mathscr{D}}\mathbf{f}_{\mathscr{D}}^T|u]\right] &= \operatorname{Tr}\left[\sigma(\mathbf{f}_{\mathscr{D}}|u)+\mathbb{E}[\mathbf{f}_{\mathscr{D}}|u]\mathbb{E}[\mathbf{f}_{\mathscr{D}}|u]^T\right]\\ &= \operatorname{Tr}\left[\sigma(\mathbf{f}_{\mathscr{D}}|u)+\alpha\alpha^T\right] = \operatorname{Tr}\left[\Sigma_{\mathscr{D}\mathscr{D}}-Q_{\mathscr{D}\mathscr{D}}+\alpha\alpha^T\right].\end{aligned}$$

Then, Equation (4.20) takes form of:

$$\begin{aligned}\int p(\mathbf{f}_{\mathscr{D}}|u)\log p(\mathbf{y}\mid\mathbf{f}_{\mathscr{D}})d\mathbf{f}_{\mathscr{D}} &= -\frac{n}{2}\log(2\pi\sigma^2)-\frac{1}{2\sigma^2}\operatorname{Tr}\left[\mathbf{y}\mathbf{y}^T-2\mathbf{y}\alpha^T+\alpha\alpha^T+\Sigma_{\mathscr{D}\mathscr{D}}-Q_{\mathscr{D}\mathscr{D}}\right]\\ &= -\frac{n}{2}\log(2\pi\sigma^2)-\frac{1}{2\sigma^2}(y-\alpha)^T(y-\alpha)-\frac{1}{2\sigma^2}\operatorname{Tr}(\Sigma_{\mathscr{D}\mathscr{D}}-Q_{\mathscr{D}\mathscr{D}})\\ &= \log\left[\mathscr{N}(\mathbf{y}|\alpha,\sigma^2\mathbf{I})\right]-\frac{1}{2\sigma^2}\operatorname{Tr}(\Sigma_{\mathscr{D}\mathscr{D}}-Q_{\mathscr{D}\mathscr{D}}).\end{aligned} \tag{4.21}$$

$\Sigma_{\mathscr{D}\mathscr{D}}$ cannot be directly calculated in Intrinsic GPs due to the high computational cost of simulating BM paths. Similarly to approximation of Σ_{rr} in section 4.4.1, the trace only concerns

the values on the diagonal of $\Sigma_{\mathscr{D}\mathscr{D}}$. An approximation method is then used here:

$$\Sigma^*_{\mathscr{D}\mathscr{D}} = \max(\text{diag}(\Sigma_{zz}))\mathbf{I}. \tag{4.22}$$

Due to $\text{Tr}(\Sigma_{\mathscr{D}\mathscr{D}} - Q_{\mathscr{D}\mathscr{D}})$ being non-negative, Equation (4.21) can be rewritten as:

$$\int p(\mathbf{f}_{\mathscr{D}}|u)\log p(\mathbf{y} \mid \mathbf{f}_{\mathscr{D}})d\mathbf{f}_{\mathscr{D}} = \log\left[\mathscr{N}(\mathbf{y}|\alpha,\sigma^2\mathbf{I})\right] - \frac{1}{2\sigma^2}\text{Tr}\left[\max(\Sigma^*_{\mathscr{D}\mathscr{D}} - Q_{\mathscr{D}\mathscr{D}},0)\right].$$

After expanding the first part, Equation (4.18) $F_V(\phi)$ can be written as:

$$F_V(\phi) = \int \phi(u)\log\left[\frac{\mathscr{N}(\mathbf{y}|\alpha,\sigma^2\mathbf{I})p(u)}{\phi(u)}\right]du - \frac{1}{2\sigma^2}\text{Tr}\left[\max(\Sigma^*_{\mathscr{D}\mathscr{D}} - Q_{\mathscr{D}\mathscr{D}},0)\right]. \tag{4.23}$$

By reversing Jensen's inequality, which involves moving the logarithm outside of the integral, the maximum value of the lower bound can be computed:

$$\begin{aligned} F_V(\phi) &\leq \log\int N(\mathbf{y}|\alpha,\sigma^2\mathbf{I})p(u)du - \frac{1}{2\sigma^2}\text{Tr}\left[\max(\Sigma^*_{\mathscr{D}\mathscr{D}} - Q_{\mathscr{D}\mathscr{D}},0)\right] \\ &= \log\left[N(\mathbf{y}|0,\sigma^2\mathbf{I}+Q_{\mathscr{D}\mathscr{D}})\right] - \frac{1}{2\sigma^2}\text{Tr}\left[\max(\Sigma^*_{\mathscr{D}\mathscr{D}} - Q_{\mathscr{D}\mathscr{D}},0)\right]. \end{aligned}$$

From this equation, it is evident that maximizing the lower bound of the log marginal likelihood function is also equivalent to minimizing the trace $\text{Tr}\left[\max(\Sigma^*_{\mathscr{D}\mathscr{D}} - Q_{\mathscr{D}\mathscr{D}},0)\right]$. This term is equal to the variance of $p(\mathbf{f}_{\mathscr{D}}|u)$, which is the error in predicting $\mathbf{f}_{\mathscr{D}}$ from u. Then, the model parameters can be computed by maximizing the lower bound log likelihood $F_V(\phi)$. The next step is to consider how to compute the predictive mean and variance. In Section 4.4.1, expressions for the predictive mean and variance in terms of $\phi(u) \sim \mathscr{N}(\mu,A)$ have been derived, seeing Equation (4.11) and (4.14). In the following section, the optimal distribution of $\phi(u)$ will be obtained through the derivation of $F_V(\phi)$.

4.4.3 Optimise $\phi(u)$

The optimisation of $\phi(u)$ can be achieved through maximizing the lower bound of the marginal likelihood function. Taking the derivative of $F_V(\phi)$ with respect to $\phi(u)$ in Equation (4.23),

takes the form of:

$$\frac{\partial F_V(\phi)}{\partial \phi(u)} = \frac{\partial}{\partial \phi(u)} \left[\int \phi(u) \log \left[\frac{\mathcal{N}(\mathbf{y}|\alpha, \sigma^2 \mathbf{I}) p(u)}{\phi(u)} \right] du - \frac{1}{2\sigma^2} \operatorname{Tr} \left[\max(\Sigma^*_{\mathscr{D}\mathscr{D}} - Q_{\mathscr{D}\mathscr{D}}, 0) \right] \right].$$

To find the optimal value of $\phi(u)$, the derivative has been set to 0,

$$\frac{\partial F_V(\phi)}{\partial \phi(u)} = 0.$$

The specific derivation proceeds as follows. Let G denote the logarithmic part of the integral,

$$G = \log \left(\frac{N(y|\alpha, \sigma^2 I) p(u)}{\phi(u)} \right).$$

Also, $\frac{\partial G}{\partial \phi(u)} = -\frac{1}{\phi(u)}$. Thus,

$$\begin{aligned} \frac{\partial}{\partial \phi(u)} \int \phi(u) G \, du &= 0, \\ G + \phi(u) \frac{\partial G}{\partial \phi(u)} &= 0, \\ G &= 1. \end{aligned}$$

Rearranging the expression of G gives

$$\log \phi(u) = \log N(y|\alpha, \sigma^2 I) + \log p(u) - 1.$$

Taking the exponential results in:

$$\phi(u) = \frac{N(y|\alpha, \sigma^2 I) p(u)}{Z},$$

where Z is a normalization constant.

Then, the optimal $\phi(u)$ is proportional to the product of the Gaussian likelihood and the prior:

$$\phi(u) \propto N(\mathbf{y}|\alpha, \sigma^2 \mathbf{I}) p(u),$$

where

$$\begin{aligned}\mathcal{N}(y|\alpha,\sigma^2 I) &\propto \exp\left(-\frac{1}{2\sigma^2}(y-\alpha)^T(y-\alpha)\right)\\ &= \exp\left(-\frac{1}{2\sigma^2}(y-\Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}u)^T(y-\Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}u)\right),\\ p(u) &\propto \exp\left(-\frac{1}{2}u^T\Sigma_{\mathrm{zz}}^{-1}u\right).\end{aligned}$$

Then, it can be rewritten as:

$$\phi(u) \propto c\exp\left\{-\frac{1}{2}u^T(\Sigma_{\mathrm{zz}}^{-1}+\frac{1}{\sigma^2}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1})u+\frac{1}{\sigma^2}\mathbf{y}^T\Sigma_{\mathscr{D}\mathrm{z}}\Sigma_{\mathrm{zz}}^{-1}u\right\},$$

where c is a constant. By completing the quadratic form, it is recognized that this expression represents a Gaussian distribution. Consequently, the mean and covariance of the Gaussian distribution for $\phi(u)$ are identified:

$$\phi(u) = N(u|\sigma^{-2}\Sigma_{\mathrm{zz}}(\Sigma_{\mathrm{zz}}+\sigma^{-2}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}})^{-1}\Sigma_{\mathrm{z}\mathscr{D}}\mathbf{y}, \Sigma_{\mathrm{zz}}(\Sigma_{\mathrm{zz}}+\sigma^{-2}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}})^{-1}\Sigma_{\mathrm{zz}}).$$

Then,

$$\begin{aligned}\mu &= \sigma^{-2}\Sigma_{\mathrm{zz}}(\Sigma_{\mathrm{zz}}+\sigma^{-2}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}})^{-1}\Sigma_{\mathrm{z}\mathscr{D}}\mathbf{y}\\ A &= \Sigma_{\mathrm{zz}}(\Sigma_{\mathrm{zz}}+\sigma^{-2}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}})^{-1}\Sigma_{\mathrm{zz}}).\end{aligned}$$

The predictive mean and variance in Equation (4.11) and (4.14) of SI-GPVI can be calculated as:

$$\begin{aligned}\mu_{VI}(S) &= \sigma^{-2}\Sigma_{\mathrm{rz}}\mathrm{K}\Sigma_{\mathrm{z}\mathscr{D}}\mathbf{y},\\ \sigma_{VI}(S) &= \max\left[\mathrm{diag}(\Sigma_{\mathrm{rr}}^*)-\mathrm{diag}(Q_{\mathrm{rr}}),0\right]+\mathrm{diag}(\Sigma_{\mathrm{rz}}\mathrm{K}\Sigma_{\mathrm{zr}}),\end{aligned} \tag{4.24}$$

where $\mathrm{K} = (\Sigma_{\mathrm{zz}}+\sigma^{-2}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}})^{-1}$, $Q_{\mathrm{rr}} = \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}}$ and $\mathrm{diag}(\Sigma_{\mathrm{rr}}^*) = \max(\mathrm{diag}(\Sigma_{\mathrm{zz}}))$.

The proposal of SI-GPVI offers a novel approach for regression on manifolds, effectively addressing the computational challenges associated with Intrinsic GPs. This method can be applied to large datasets and high-dimensional manifolds, ensuring both efficiency and accuracy. Benefiting from its ability to minimize the difference between the original target p and the approximate distribution q, SI-GPVI achieves favorable results in handling complex manifolds and providing more accurate predictions. The specific application of SI-GPVI will be presented in Chapter 6. Through the three examples outlined in Section 1.3, SI-GPVI is validated by comparison with other GPs, demonstrating its effectiveness and advantages.

4.4.4 Enhancing Computational Efficiency via Inducing Points

As mentioned before, in the intrinsic GP framework, the introduction of inducing points allows the reduction of time complexity from $O(n^3)$ to $O(nm^2)$, while also improving numerical stability. Moreover, since the intrinsic GP models the heat kernel on the manifold using the transition density of BM, the use of inducing points significantly reduces the computational burden: BM paths only need to be simulated from the inducing points rather than from each grid point N times. This results in a computational saving roughly proportional to G'/m, where G' denotes the total number of grid points and m is the number of inducing points. Similarly, storage requirement is also reduced by approximately the same factor.

Chapter 6 presents a comparison of these GP methods. For the U-shape example, 5 inducing points are evenly distributed along the U-shape, with a total of 418 grid points. Simulating BM paths 50,000 times from the 5 inducing points took approximately 11.73 minutes, with a memory usage of 723 MB. Compared to the non-sparse version, this setup achieved an 83.6-fold improvement in computational efficiency, along with a proportional reduction in memory consumption. In the Bitten-torus example, 6 inducing points are uniformly distributed over the surface with 600 grid points. The BM path simulations from 6 inducing points 50,000 times took around 48.11 minutes, occupying 1.48 GB of memory. This yielded a 100-fold improvement in both computation time and memory usage. For the Aral Sea example, due to the presence of real-world noise and a more complex boundary structure, a higher number of inducing points is needed. 10 inducing points are uniformly distributed across the surface, with 485 grid points in total. To accommodate the higher computational load, parallel computing was employed. The simulation took approximately 84 minutes, with 1.40 GB of memory used, resulting in a 48.5-fold improvement in efficiency compared to the non-sparse version. These results highlight the advantages of the sparse intrinsic GP approach in both computational efficiency and storage requirement. The detailed applications of these examples are presented in Chapter 6.

4.5 Conclusion of Proposed Methods

This chapter explores approaches to enhance the computational efficiency of Intrinsic GPs by implementing sparse approximation techniques. Three key approaches are proposed: SI-GPDIC, SI-GPDTC, and SI-GPVI. These methods address the computational challenges of Intrinsic GPs by approximating the process using a subset of data points, referred to as inducing points, rather than relying on the entire training dataset.

SI-GPDIC focuses on using a small number of inducing points to approximate the full dataset, thereby reducing computational demands. It combines the Intrinsic GPs with the DIC method, where information from $\mathbf{f}_{\mathscr{D}}$ can only be transmitted to $\mathbf{f}_r$ via the inducing variables u, effectively addressing the computational burden. However, this method exhibits errors when calculating the predictive variance in some regions. The accuracy of the predictive variance in SI-GPDIC is significantly affected by the placement of the inducing points. When moving away from the inducing points and training points, the predictive variance incorrectly decreases instead of increasing, resulting in inaccurate uncertainty estimates. A set of examples on the U-shape domain illustrates this issue, demonstrating that the variance drops to zero in regions far from the inducing points, which poses a significant problem.

Then, SI-GPDTC is used to account for this problem. It combines the Intrinsic GPs with the DTC method and provides a more accurate estimation by incorporating its own prior variance Σ_{rr}. This prior variance adjusts the predictive uncertainties, making them more reliable. The predictive mean remains the same as in SI-GPDIC, but the key difference lies in the predictive variance. It includes an additional term $(\Sigma_{\text{rr}} - Q_{\text{rr}})$, which accounts for the full covariance matrix between all testing points, thereby preventing underestimation of the total variance. This term ensures positive semidefiniteness and effectively resolves the variance estimation issues seen in the DIC method. For the Intrinsic GPs, Σ_{rr} cannot be directly calculated. It is approximated as the maximum value of the diagonal elements in Σ_{zz}, which can be expressed as $\Sigma^*_{\text{rr}} = \max(\text{diag}(\Sigma_{\text{zz}}))\mathbf{I}$. Using the same U-shaped domain example, it is demonstrated that, unlike SI-GPDIC, SI-GPDTC maintains a high variance for distant test points, preventing it from dropping to zero and ensuring more reliable predictions.

The chapter also explores SI-GPVI. VI efficiently approximates complex probability distri-

butions by transforming the estimation of posterior distributions into an optimisation problem. This optimization mechanism ensures the accuracy of SI-GPVI. This section first derives the form of the predictive mean and variance by introducing $\phi(u)$, a variational intrinsic Gaussian distribution. During this process, the Intrinsic GPs simplify the variance function for practical applications and explain the process of approximating the diagonal elements to avoid high computational costs. Then, Section 4.4 discusses how the lower bound of marginal log-likelihood is utilized as the optimisation objective to approximate the posterior distribution. Building on this, Section 4.4 derives the predictive distribution for SI-GPVI, simplifying the variance function for practical applications and explains the process of approximating the diagonal elements to avoid high computational costs.

Overall, Chapter 4 presents a comprehensive discussion of three Sparse Intrinsic GPs. These methods reduce the computational complexity, making Intrinsic GPs more practical for large-scale and complex applications. The three approaches provide robust frameworks for implementing Intrinsic GPs in various constrained and irregular-shaped domains. In Chapter 6, the application of these methods in various scenarios will be provided. Building on this foundation, extensions in Bayesian optimisation (BO) will be discussed in Chapter 7.

The next chapter introduces the Graph GPs as a comparable but alternative approach. This process is constructed based on the Graph Matérn Kernel. When applied to manifolds, it treats them uniformly as graphs.

Chapter 5

Graph Gaussian Processes

The previous chapter presents the GP regression method on manifolds from two theoretical frameworks. One approach is a direct extension of GP applications in Euclidean space, referred to as Traditional GPs in Section 3.2. This method is based on the RBF kernel relying on the Euclidean distance for modeling, which does not account for the intrinsic structure of manifolds. The second framework considers the geometric structure of the manifold. Section 3.3 introduced the Intrinsic GPs, utilizing the approximate heat kernel to model the manifold's intrinsic geometric properties. Since not all manifolds have a well-defined analytical expression for the heat kernel, Intrinsic GPs use the transition density of Brownian motion (BM) paths simulated on manifolds to approximate the heat kernel. To mitigate the computational challenges in Intrinsic GPs, Chapter 4 proposed three distinct sparse intrinsic GP methods, which are: SI-GPDIC, SI-GPDTC and SI-GPVI. This chapter will explore a third theoretical framework, which learns manifolds with complex boundaries by constructing undirected graphs on them and utilizing the graph Matérn kernel for modeling. The quality of the constructed graph significantly influences the Graph GP's performance. Section 5.1 provides an overview of the development of Graph GPs. Section 5.2 introduces the approximation of the Laplace–Beltrami operator, which transforms information on the manifold into inputs used by the Matérn kernel on graphs. Section 5.3 constructs the graph Matérn kernel, presenting parameter tuning and the limitations of Graph GPs.

5.1 Introduction

A Graph GP is a method designed to handle graph-structured data, such as social networks [21], [105], molecular structures [99], [58], and sensor networks [102]. GPs are no longer limited to Euclidean space and have also been extended to graphs. Furthermore, Graph GP is not limited to graphs; it can also serve as an approach to develop GPs on manifolds by leveraging graph structures to model complex dependencies between data points on the manifold. This method of transforming regression problems on manifolds into undirected graphs enables representation of the geometry of the manifold by using graphical connections, which provides a new approach to implementing GPs on manifolds with complex boundaries. This makes Graph GPs a valuable point of comparison against the three methods proposed in this work. Through comparison with Graph GPs, the effectiveness of the proposed methods is more strongly validated.

Section 2.3 provides a brief introduction to the fundamental concepts of graph theory. A number of approaches have been proposed to define GPs over a weighted undirected graph $G = (V, E)$, as defined in Definition 2.4. These include the works of Kondor and Lafferty [80], Rue and Held [130], and others, in areas such as Gaussian Markov Random Fields (GMRFs) and diffusion kernels. Whittle [159] finds that Matérn GPs satisfy the stochastic partial differential equation (SPDE) and Lindgren et al. [93] uses this perspective and defines a Matérn GP as the solution to a specific SPDE driven by white noise. Inspired by previous works, Borovitskiy et al. [16] explore the development of GP models using the SPDE form of the Matérn kernel for manifolds M' defined without a boundary, under the assumption that the Laplace–Beltrami operator Δ'_g and its eigenpairs are known. Expanding on this, Borovitskiy et al. [17] explore GPs indexed by the vertices of undirected graphs G, assuming that the Graph Laplacian (GL) and its eigenpairs are known. Their work studies the graph Matérn GPs based on the graph Matérn kernels, analogous to the SPDE form of the Matérn kernels used in Euclidean space, and applies them to graphs derived from manifolds with boundaries.

To implement GPs on manifolds, M, with a boundary using the graph structure, one approach, proposed by Dunson et al. [36], is the Graph Laplacian-based GP (GL-GP). When the Laplace– Beltrami operator, denoted as Δ_s, of a manifold M is unknown, the GL-GP approximates it by constructing a graph on the manifold and uses the GL to approximate the heat kernel of M, relying on a finite number of eigenpairs from the GL. The approximate heat kernel is

tied to Euclidean distances between graph vertices V. Inspired by the convergence of graph Matérn GPs between graphs and their manifold counterparts [17], Fichera et al. [41] extend graph Matérn GPs to implicit manifolds, addressing cases where the Laplace–Beltrami operator is unknown. They follow Dunson et al. [36] in constructing a graph on the manifold, calculating the random walk normalized Laplacian Δ_{rw} and its eigenpairs, and subsequently using the Graph Matérn kernel to implement GPs on M.

This research targets known manifolds with complex boundaries and intrinsic structures, aiming to solve regression problems. The presence of complex boundaries adds an additional level of complexity to the problem, which cannot be directly compared to the work of Boroviskiy et al. [16] which has not considered this situation. The Graph GPs used for comparison with Tradtional GPs and SI-GPDIC, SI-GPDTC and SI-GPVI in this thesis study the graph Matérn GPs mentioned in [41]. This Graph GP extends the work of Borovitskiy et al. on GPs over graphs [17] and their earlier research on manifolds without boundaries [16], adapting it to handle manifolds with complex boundaries. The following sections will introduce the construction of Graph GPs, illustrating how they can be used to model regression tasks on manifolds.

5.2 Approximation of Laplace–Beltrami Operator

Suppose M is a connected Riemannian manifold with a boundary, and S denotes the grid points uniformly distributed across the manifold, as previously defined in Sections 2.1 and 3.2. This manifold can be approximated by a weighted undirected graph with the node set S and weights determined by the squared Euclidean distances $\|s_i - s_j\|^2$, where $s_i, s_j \in M$. The most common way to define the graph is using a Gaussian-like kernel function to represent the adjacency matrix W:

$$W_{ij} = \exp(-\frac{||s_i - s_j||^2}{4\zeta^2}),$$

where $\zeta > 0$, ζ is the bandwidth. By using the K-Nearest Neighbors Graph (KNN), the sparsification of the matrix W is effectively achieved, significantly reducing its density when $K << G'$

[41]. The matrix W can be reformulated as:

$$\tilde{\mathbf{A}}_{ij} = S_K\left(s_i, s_j\right) \exp\left(-\frac{\left\|s_i - s_j\right\|^2}{4\zeta^2}\right),$$

where $S_K\left(s_i, s_j\right)$ is the K-Nearest Neighbors sparsification coefficient, ensuring that only the K-nearest neighbors of each point contribute to the adjacency matrix. Suppose $\tilde{\mathbf{D}}$ is the degree matrix of the graph, which can be expressed as:

$$\tilde{\mathbf{D}}_{ij} = \begin{cases} \sum_m \tilde{\mathbf{A}}_{im} & i = j \\ 0 & i \neq j \end{cases}, \tag{5.1}$$

where each diagonal element corresponds to the sum of the weights of the edges connected to node s_i. The GL can be used to approximate the Laplace-Beltrami operator on the manifold, as introduced in Section 2.1. There are three popular notions of GL, and this study utilizes the random walk normalized Laplacian, which takes the form:

$$\Delta_{\text{rw}} = \mathbf{I} - \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{A}}. \tag{5.2}$$

Also, there are two other forms of GL, which are the Unnormalized GL and the Symmetric Normalized Laplacian, defined below respectively:

$$\Delta_{\text{un}} = \tilde{\mathbf{D}} - \tilde{\mathbf{A}}, \quad \Delta_{\text{svm}} = \mathbf{I} - \tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1/2},$$

where $\tilde{\mathbf{A}}$ is the adjusted adjacency matrix defined in Equation (5.2) and $\tilde{\mathbf{D}}$ represents the diagonal degree matrix defined in Equation (5.1).

When the grid points set S are uniformly distributed on the manifold, all of the GLs, each multiplied by an appropriate power of ζ, converge to the Laplace–Beltrami operator, both pointwise [60] and spectrally [48]. When the grid points in S are sampled non-uniformly from the manifold, none of the GLs converge to the Laplace–Beltrami operator [60]. By taking $\mathbf{A} = \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1}$, the random walk normalized Laplacian Δ_{rw} can be rewritten as:

$$\Delta_{\text{rw}} = \mathbf{I} - \tilde{\mathbf{D}}^{-1}\mathbf{A}.$$

This transformation proposed by Coifman and Lafon [30] helps address the pointwise convergence of non-uniformly distributed data to the Laplace–Beltrami operator, which corresponds to normalizing by using the kernel density estimator to cancel out the unknown density. However, this approach is not effective for Δ_{un} and Δ_{svm}. The grid points S used in this thesis are uniformly distributed on the manifold. This allows for a more comprehensive representation of the manifold, and, as illustrated in Chapter 7, the uniform distribution of grid points facilitates efficient exploration of the manifold in BO under a budget constraint. Equation (5.2) can serve as the effective approximation of the Laplace–Beltrami operator. The eigenpairs of Δ_{rw} consist of the eigenvalues and their corresponding eigenvectors, denoted as (λ_i, f_i), satisfying:

$$\Delta_{\text{rw}} \lambda_i = \Delta_{\text{rw}} f_i.$$

These eigenpairs play a crucial role in spectral analysis and will be used for constructing Graph GPs on manifolds with complex boundaries in next section.

5.3 Graph Matérn Gaussian Processes

The previous section provided a random walk normalized Laplacian Δ_{rw} constructed from the grid points using the graph structure, which corresponds to the Laplace–Beltrami operator Δ_s on the manifold M. This section will demonstrate how to implement GPs on manifolds with complex boundaries by utilizing the eigenpairs (λ_i, f_i) of Δ_{rw}.

5.3.1 Stochastic Partial Differential Equations

The Matérn kernel is a widely used kernel in Euclidean space, as shown in Equation (2.3). Whittle [159] has demonstrated that Matérn GPs on $X = \mathbb{R}^d$ satisfy the following SPDE:

$$\left(\frac{2\nu}{\kappa^2} - \Delta\right)^{\frac{\nu}{2}+\frac{d}{4}} f = \mathscr{W}, \tag{5.3}$$

where ν is non-negative, controlling the smoothness of the function, κ shows the degree of dependency, Δ is the Laplacian and $\mathscr{W}$ is Gaussian white noise [90] re-normalized by a certain

constant. This SPDE form directly allows the Matérn kernel to be applied to Riemannian manifolds, which is achieved by replacing Δ with the Laplace–Beltrami operator. The term $\mathscr{W}$ can be replaced by a canonical white noise process. As given in Definition 2.1, suppose (M', g) is a compact Riemannian manifold without boundary, and let Δ'_g be its Laplace–Beltrami operator. To solve the covariance kernel of the GP in Equation (5.3) within this setting, Boroviskiy et al. [16] introduce an appropriate form that allows the use of spectral theory to compute the required expressions. There exists a countable number of non-negative eigenvalues, following a non-decreasing sequence with $0 = \lambda_0 < \lambda_1 \leq ... \leq \lambda_n$, where $\lim_{n\to\infty} \lambda_n = \infty$. The corresponding eigenfunctions form an orthonormal basis $\{f_n\}_{n\in\mathbb{Z}_+}$. Following the Sturm–Liouville decomposition [27], $-\Delta'_g$ can be written as:

$$-\Delta'_g f = \sum_{n=0}^{\infty} \lambda_n \langle f, f_n \rangle f_n. \tag{5.4}$$

It is natural to define the operators $\Phi\left(-\Delta'_g\right) = \left(\frac{2\nu}{\kappa^2} - \Delta'_g\right)^{\frac{\nu}{2}+\frac{d}{4}}$. By replacing $\Phi(\lambda_n) = \left(\frac{2\nu}{\kappa^2} + \lambda_n\right)^{\frac{\nu}{2}+\frac{d}{4}}$. Then, Equation (5.4) can be expressed as:

$$\left(\frac{2\nu}{\kappa^2} - \Delta'_g\right)^{\frac{\nu}{2}+\frac{d}{4}} f = \sum_{n=0}^{\infty} \left(\frac{2\nu}{\kappa^2} + \lambda_n\right)^{\frac{\nu}{2}+\frac{d}{4}} \langle f, f_n \rangle f_n.$$

Then, the spectral decomposition of the left-hand side of the SPDE (5.3) is obtained.

The right part of SPDE (5.3), the canonical white noise process $\mathscr{W}$, can be substituted with the appropriate generalization of the Gaussian white noise [93]. The Matérn kernel on a manifold without boundary can be summarized as:

Theorem 5.1. *Let λ_n be eigenvalues of $-\Delta'_g$, and let f_n be corresponding eigenfunctions. The Matérn kernel on a manifold without boundary M' are given by [16]:*

$$k_\nu\left(s, s'\right) = \frac{\sigma^2}{C_\nu} \sum_{n=0}^{\infty} \left(\frac{2\nu}{\kappa^2} + \lambda_n\right)^{-\nu-\frac{d}{2}} f_n(s) f_n\left(s'\right)$$

where C_ν is normalizing constant chosen so that the average variance over the manifold satisfies $vol_g(M)^{-1} \int_X k(\cdot)(x,x)\,\mathrm{d}x = \sigma^2$.

Proof. See Borovitskiy [**Theorem 5**, 16]. □

5.3.2 Graph Matérn Gaussian Processes on Manifolds with Complex Boundaries

The previous section introduced the Matérn kernel for manifolds M' without boundary and derived the kernel equation based on the SPDE formulation of original Matérn kernel in Euclidean space. To extend the Matérn kernel to manifolds with complex boundaries, the internal structure and boundary characteristics of the manifold can be captured through their analogous undirected graphs, as introduced in Section 5.2. This method utilizes the random walk normalized Laplacian Δ_{rw}, as defined in Equation (5.2) and constructed in Section 5.2 to replace the Laplace–Beltrami operator Δ_g on the manifold M. Denoting its eigenvalues, ordered from smallest to largest, by $0 = \lambda_0 \leq \lambda_1 \leq \ldots \leq \lambda_{N-1}$, and its eigenvectors, orthonormal under the modified inner product $\langle \cdot, \cdot \rangle_D$, by $f_0, f_1, \ldots, f_{N-1}$, the Matérn kernel is defined as:

$$k_{\nu,\kappa,\sigma^2}\left(s_i, s_j\right) = \sigma^2 \sum_{l=0}^{L-1} \left(\frac{2\nu}{\kappa^2} + \lambda_l\right)^{-\nu} f_l\left(s_i\right) f_l\left(s_j\right), \tag{5.5}$$

where L is not equal to the actual number N of eigenpairs, which means cutting off high frequency eigenvectors (f_l when l large). Since high-frequency eigenvectors contribute less to the sum, this approach can effectively reduce computational costs. The value of L can be treated as a parameter, which will be discussed in the next section. Then, the posterior distribution of Graph GPs is:

$$p\left(\mathbf{f}_r \mid y\right) = \mathcal{N}\left(\Sigma_{\mathrm{r}\mathscr{D}}\left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_{\mathrm{n}}^2 I\right)^{-1} y, \Sigma_{\mathrm{rr}} - \left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_{\mathrm{n}}^2 I\right)^{-1} \Sigma_{\mathscr{D}\mathrm{r}}\right),$$

where $\mathbf{f}_r$ is the corresponding function value for the grid points dataset S, $\mathscr{D}$ is the training point datasets and σ_{n}^2 is the noise variance of GPs. $\Sigma_{\mathrm{r}\mathscr{D}}$ is calculated from Equation (5.5), the Matérn kernel defined on manifolds with complex boundaries.

5.3.3 Hyperparameter optimisation and Limitations

When calculating the Matérn kernel defined on a manifold with a complex boundary, there exists hyperparameters $\hat{\theta} = \left(\hat{\zeta}, \hat{\kappa}, \hat{\sigma}^2, \hat{\sigma}_{\mathrm{n}}^2\right)$ which determine the graph structure, GP prior and the noise variance that fits the observations y best. The smoothing parameter ν of the Matérn kernel

function is predefined to avoid unreasonable parameter values. It usually takes specific half-integer values, such as 0.5, 1.5, or 2.5. The hyperparameters $\hat{\theta}$ can be estimated by maximizing the log marginal likelihood:

$$\log p\left(\mathbf{y} \mid \zeta, \kappa, \sigma^2, \sigma_{\mathrm{n}}^2\right) = -\frac{1}{2}\mathbf{y}^{\top}\left(\Sigma + \sigma_{\mathrm{n}}^2 I_{m\times m}\right)^{-1}\mathbf{y} - \frac{1}{2}\log\left(\det\left(\Sigma + \sigma_{\mathrm{n}}^2 I_{m\times m}\right)\right) - \frac{n}{2}\log(2\pi), \tag{5.6}$$

where $\Sigma = k_{\nu,\kappa,\sigma^2}\left(s_i, s_j\right)$, as defined in Equation (5.5). This is similar to the hyperparameter optimisation method used in Traditional GPs as described in Equation (3.3). The effectiveness of Graph GPs is highly dependent on the choice of the parameter ζ. It should be sufficiently large to ensure that there are enough points within an ζ-sized Euclidean neighborhood around s to capture the local geometry, while avoiding the inclusion of distant points that could distort the representation [36]. Since the ζ-sized Euclidean neighborhood is based on Euclidean distance, it tends to overlook the intrinsic geometric features of the manifold. When dealing with data from complex manifolds or exhibiting highly non-uniform density, there is generally a loss of accuracy at the local level. This is because the single ζ is optimised on all of the global data points, therefore the consistent choice of ζ may not adequately capture local variations in the manifold's structure.

The number of neighbors K in the KNN method (introduced in Section 5.2) and the number of eigenpairs L, which determines the cutoff for high-frequency eigenvectors, are assumed to be manually fixed parameters [41]. Higher values of K and L may improve the quality of the approximation of the manifold kernel, but could substantially increase computational costs. Furthermore, higher parameter values do not always result in better estimation of the kernel on manifolds. Larger datasets coupled with high parameter values can lead to numerical stability issues, potentially causing inefficiencies in the model estimation. Fichera et al. [41] does not provide clear recommendations for determining the values of K and L, leaving these to be the subject of further exploration. Dunson et al. [36] and Luo et al. [97] also discuss the selection of these parameter values, but neither provides a clear method for determining them.

As mentioned by Fichera et al. [41], the quality of the constructed graph significantly influences the technique's performance. When dealing with data from complex manifolds, simplistic KNN method might fail to capture the manifold structure relying on a single graph bandwidth ζ. The selection of these parameters K, L, ν significantly affects the quality of the constructed

graph. Additionally, when the number of data points used to construct the graph is insufficient, it often results in a poor approximation of the manifold, which in turn affects the stability and robustness of the Graph GPs method.

This chapter proposes an alternative perspective using Graph GPs compared to Traditional GPs and Sparse Intrinsic GPs on manifolds. Graph GPs construct graphs capturing the intrinsic structure of the manifolds. The Graph Matérn kernel is constructed by using the SPDE form of the Matérn kernel from Euclidean space, where the Laplace operator Δ is replaced by the Graph Laplacian to apply it to graphs [41]. The Graph Matérn kernel is indirectly applied to the manifold with complex boundaries by constructing a graph structure on the manifold, allowing for the modelling of the manifold's geometry through the graph. However, this method has several drawbacks. Since Graph GPs rely on the graph structure constructed on the manifold, if this graph fails to accurately capture the intrinsic geometric structure of the manifold, it can lead to significant errors in the predictions made by Graph GPs.

Graph-based methods have recently gained increasing attention in the study of manifolds, providing a approach to modeling complex structures. Given their relevance, it is necessary to compare this research with Graph GPs to evaluate their relative advantages and limitations in manifold-based learning. In the next chapter, the five manifold-based GPs proposed so far—Traditional GP, SI-GPDIC, SI-GPDTC, SI-GPVI, and Graph GPs—will be applied to three representative examples presented in Section 1.3. These three examples aim to provide a comparative analysis of the effectiveness of these GP methods through their predictive distributions, while also highlighting the strengths and limitations of each approach.

Chapter 6

Applications of Proposed Gaussian Processes on Manifolds

Based on the intrinsic GPs, Chapter 4 constructs three sparse intrinsic GPs: SI-GPDIC, SI-GPDTC, and SI-GPVI. Among them, SI-GPDIC is significantly affected by the position of the inducing points in variance prediction. Specifically, as test points move further from the inducing points, the predictive variance incorrectly approaches zero. In this chapter, the three GPs developed in Chapter 4—SI-GPDIC, SI-GPDTC, and SI-GPVI—will be applied to three distinct examples, as presented in Section 1.3. These examples are selected to emphasize the advantages and limitations of each model. The models will also be evaluated and compared against Traditional GPs and Graph GPs for a thorough analysis.

Section 6.1 presents two specific data analysis methods used to compare different GP models. Section 6.2 evaluates the application of these methods on the U-shape domain. Section 6.3 extends the comparison to a three-dimensional context using the Bitten-torus as an example, starting with the derivation of the metric tensor for the Bitten-torus. In Section 6.4, the focus shifts to a real-world dataset, illustrating the practical significance of the research by predicting chlorophyll levels in the Aral Sea. Finally, Section 6.5 provides a comprehensive summary of the regression results discussed throughout the chapter.

6.1 Data Analysis Methods

The Root Mean Square Error (RMSE) as introduced in Section 3.4.2 is employed to evaluate and compare the performance of various GPs which are fitted to manifolds of varying shape and dimenionality. RMSE is a standard metric used to measure the differences between values predicted by a model and the actual observed values [62]. It offers a unified measure of predictive accuracy by calculating the square root of the average squared differences between predicted and observed values, as shown in Equation (3.12). By assessing the predictive mean through RMSE, it becomes possible to determine which model offers the most accurate predictions, thus identifying the most effective approximation method.

Predictive Log-Likelihood

Since the RMSE only considers the average difference between the predictive mean's value and the actual values, without taking into account the predictive variance's performance, the predictive log-likelihood (PLL) can consider both the predictive mean and the predictive variance. The PLL is a statistical measure used to assess the performance of predictive models, especially in probabilistic prediction. This approach is represented as a distribution over possible outcomes rather than a single deterministic value. It compares the predictive probability density function based on two predicted vectors, assessing the likelihood that the observations are within the predicted distribution of the model. The PLL for a Gaussian distribution is shown as:

$$P(\hat{\mu} \mid \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(\hat{\mu}-\mu)^2}{2\sigma}\right),$$

where μ is the true value of the objective function corresponding to grid points, $\hat{\mu}$ and σ represent the predictive mean vector and predictive variance vector separately. Since the predictive mean of SI-GPDIC and SI-GPDTC are identical, only one plot will be drawn for both methods in the subsequent sections, and they also share the same RMSE. Additionally, due to the disproportionately small predictive variance of Graph GP, its PLL values are often extremely low (less than -10,000), making them not meaningful for comparison. Therefore, Graph GPs are excluded from the PLL comparison.

Wilcoxon Signed-Rank Test

In the following three case studies, 20 datasets are randomly selected from the grid points for each case. Different GP methods are applied to perform predictions on these datasets, and the corresponding RMSE and PLL values are computed for each dataset. To assess whether the differences in performance between methods are statistically significant, pairwise paired t-tests are first considered. This statistical test requires several assumptions to be met: (1) the differences between paired observations should be approximately normally distributed, (2) the observations must be paired appropriately, and (3) the paired differences should be independent of one another [78]. In each case study, the sample size of 20 is insufficient for the Central Limit Theorem to justify the normality assumption; therefore, the normality assumption is tested using the Shapiro-Wilk test. The results indicate that in some cases, the assumption of normality is violated. Consequently, the Wilcoxon Signed-Rank Test is employed as a non-parametric alternative for paired samples. This test is suitable for comparing differences between two paired groups when the assumption of normality is not satisfied [170]. Given that the data are paired and the observations within each pair are independent, the Wilcoxon Signed-Rank Test is applied to evaluate the statistical significance of the performance differences between methods.

The next section will present the implementation of the proposed methods on the U-shape, demonstrating how each GP method performs on this two-dimensional manifold.

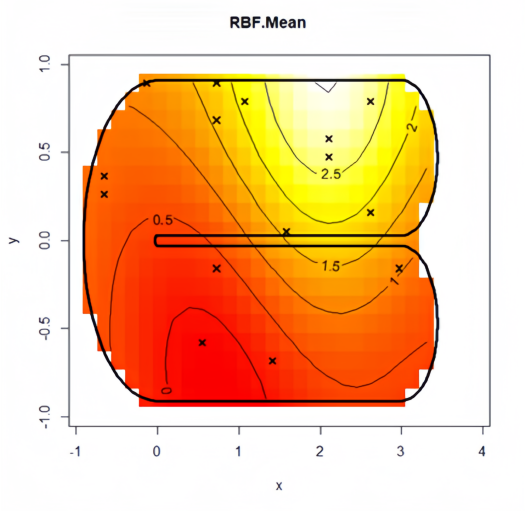
6.2 The Implementation of Proposed Methods on the U-shape

The U-shape, introduced in Section 1.3.1, is used as an example for the two-dimensional case to evaluate the proposed methods. The 418 grid points are uniformly distributed across the U-shape, providing a sufficiently dense grid to capture its structural details. From these 418 grid points, 20 datasets are randomly selected, and each dataset contains 15 training points. This approach allows for a robust comparison of the performance of different GP methods, ensuring that the results are not skewed by any single set of training points. Figures 6.1 and 6.2 illustrate one of these selected datasets, showcasing the predictive mean and predictive variance for each GP method. This helps provide a visual comparison of how each method performs on the U-

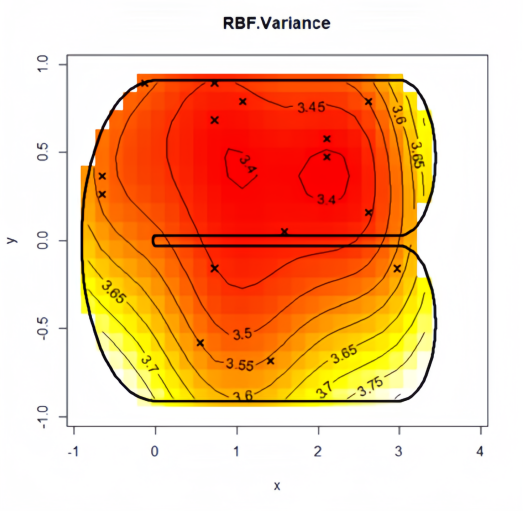
shape. In the figures, 15 black crosses denote the training points and 5 green crosses represent the inducing points used in Sparse Intrinsic GPs.

Figures 6.1 (a) and (b) show the predictive mean and variance of the Traditional GP, respectively. The Traditional GP clearly smooths over the gap between the two arms of the U-shape, due to the proximity of the upper and lower arms in Euclidean space. It fails to account for the boundary information, leading to a high covariance between points that are close in Euclidean distance, despite being far apart in intrinsic distance. Figure 6.2 (d) shows the predictive mean of the Graph GP, which also exhibits a similar situation. As discussed in Chapter 5, the Graph GPs rely on the graph structure constructed on the manifold, where the connectivity between vertices is measured through the adjacency matrix W, as shown in Equation (5.2). This matrix is based on the Euclidean distance between grid points and a bandwidth parameter ζ, optimised on all of the global data points. Due to the influence of Euclidean distance and the global bandwidth parameter ζ, the constructed graph fails to fully capture the intrinsic geometric features of the U-shape, resulting in the predictive mean smoothing across the middle gap, which does not reflect the true structure of the U-shape. Especially when the number of grid points is too small or their distribution is uneven, this can amplify the issue and lead to poor results. Figures 6.1 (c) and (d) show the predictive mean and variance of SI-GPVI. Considering the boundary factors, this method does not smooth across the middle gap. The predictive mean transitions from the red region (with low values) in the lower right, following the U-shape to the yellow region (with higher values) in the upper right, mirroring the trend of the true values. This behavior occurs because, given a fixed diffusion time, the transition probability of Brownian motion (BM) from inducing points in the lower arm to points in the upper arm within the boundary is relatively small. Consequently, the correlation between these regions is very low. Figure 6.2 (a) is the predictive mean of SI-GPDIC and SI-GPDTC. The predictive mean for these two method is same. Benefiting from the approximate heat kernel, it also does not cross the boundary. Figure 6.2 (b) and (c) are the predictive variance for SI-GPDIC and SI-GPDTC respectively.

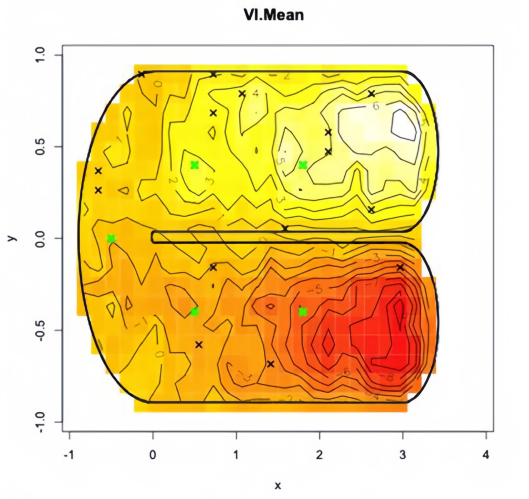
The specific comparison between methods will be presented through numerical analysis, utilizing two comparison methods, the RMSE and PLL introduced in the previous section. Table 6.1 provides descriptive statistical analysis for these results, highlighting key metrics like the minimum, median, mean, maximum, and range. From the RMSE results, it is evident that the predictive mean of the Traditional GP performs the worst compared to other methods. The table



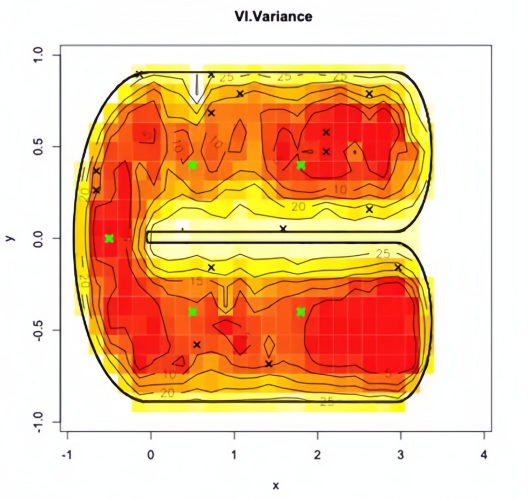
(a) Mean of Traditional GP



(b) Variance of Traditional GP

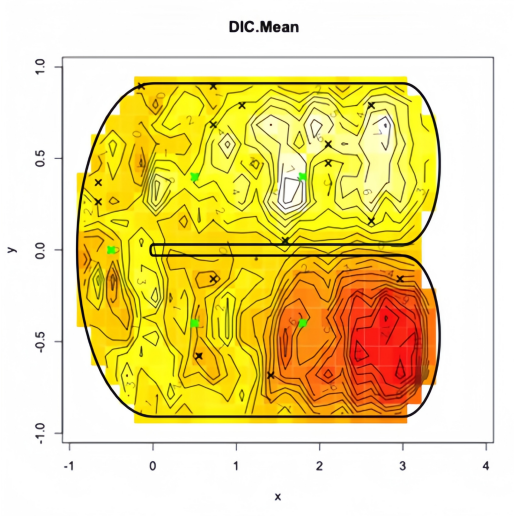


(c) Mean of SI-GPVI

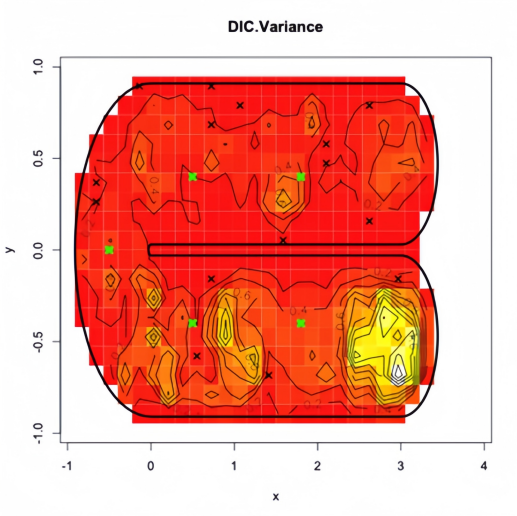


(d) Variance of SI-GPVI

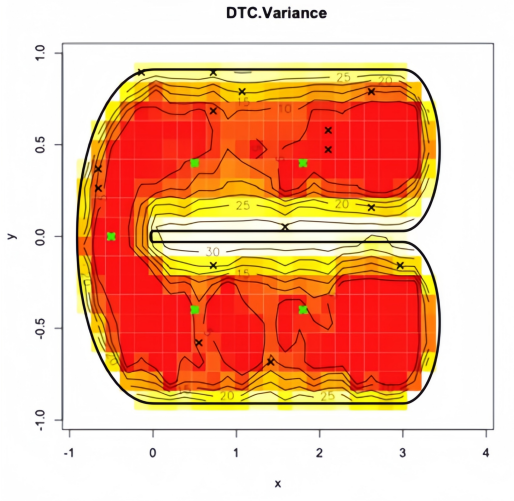
Figure 6.1: With 15 training points randomly selected, shown as black crosses on the U-shape, and green crosses as inducing points: (a)-(b) show the predictive mean and predictive variance of the Traditional GP; (c)-(d) show the predictive mean and predictive variance of SI-GPVI.



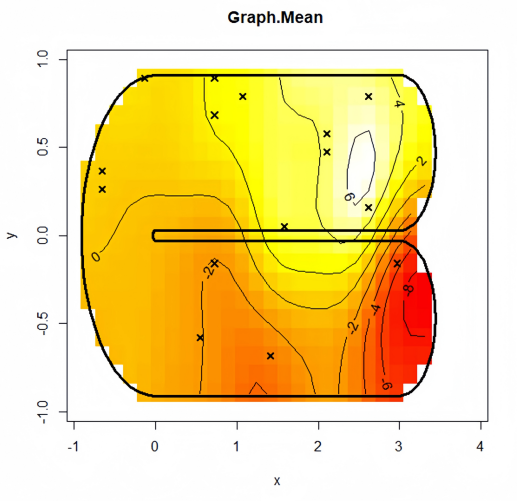
(a) Mean of SI-GPDIC & SI-GPDTC



(b) Variance of SI-GPDIC



(c) Variance of SI-GPDTC



(d) Mean of Graph GP

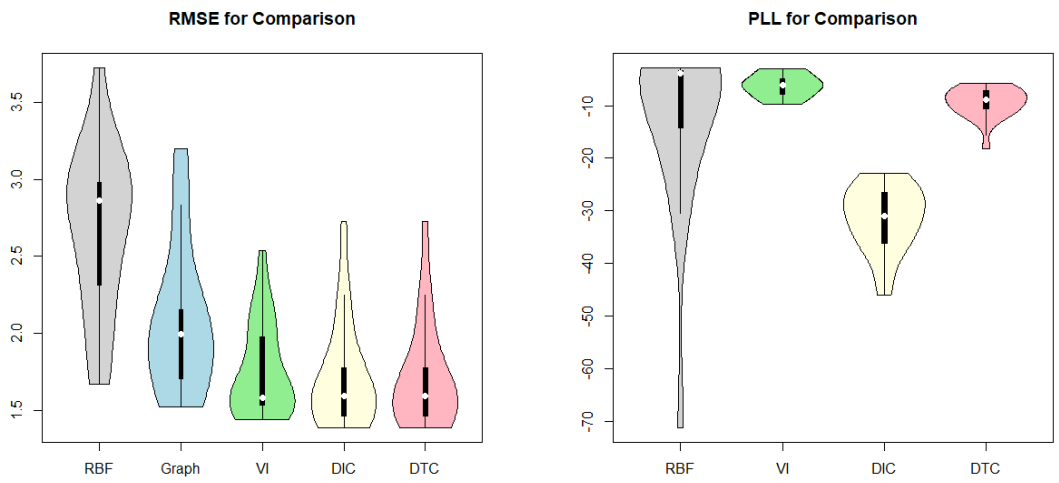
Figure 6.2: With 15 training points randomly selected, shown as black crosses on the U-shape, and green crosses as inducing points: (a) Predictive mean of SI-GPDIC & SI-GPDTC; (b) Predictive variance of SI-GPDIC; (c) Predictive variance of SI-GPDTC; (d) Predictive mean of the Graph GP.

clearly shows that the three proposed Sparse Intrinsic GPs yield better RMSE results than both the Traditional GP and Graph GP methods in terms of minimum, median, mean, and maximum values. Figure 6.3 (a) shows the violin plot of RMSE for comparison among the different methods, further illustrating these findings. The dots at each end of the bold black lines represent the first and third quartiles, while the white dot marks the median. The PLL results indicate that the Traditional GP also performs poorly in PLL. The proposed Sparse Intrinsic GPs (SI-GPVI, SI-GPDIC, and SI-GPDTC) have better PLL results compared to the Traditional GP. Among these three sparse method, due to the inaccuracy of DIC's variance, DIC's PLL results perform worse than others. Figure 6.3 (b) shows the violin plot of PLL for comparison among the different methods, supporting these observations.

Summary (RMSE)	Min.	Median	Mean	Max.	Range
Tra	1.669	2.864	2.663	3.726	2.057
Graph	1.523	1.991	2.077	3.199	1.676
VI	1.437	1.578	1.750	2.538	1.101
DIC&DTC	1.386	1.591	1.713	2.730	1.344
Summary (PLL)	Min.	Median	Mean	Max.	Range
Tra	-12.503	-1.758	-2.397	-1.563	10.94
VI	-3.438	-2.093	-2.060	-1.433	2.005
DIC	-54.583	-6.014	-10.987	-3.136	51.447
DTC	-1.907	-1.782	-1.794	-1.681	0.226

Table 6.1: Statistical summary of RMSE & PLL for all GP methods on the U-shape domain.

The Wilcoxon Signed-Rank Test is used to compare the performance between each pair of GP methods on paired data. The Wilcoxon Signed-Rank Test results for the RMSE and PLL comparisons between different GP methods are displayed in Table 6.2. If the p-value is less than 0.05, it is judged as "significant" in statistics. The Wilcoxon Signed-Rank Test results provide strong statistical evidence supporting the superiority of the proposed Sparse Intrinsic GPs over the Traditional GPs and Graph GPs. The significant differences in RMSE confirms that the Sparse Intrinsic GPs offer improved predictive performance and robustness. The Wilcoxon Signed-Rank Test comparison among SI-GPDIC, SI-GPDTC, and SI-GPVI offers valuable insights into their relative strengths and effectiveness. While there is no significant difference in RMSE, SI-GPVI stands out in terms of PLL, significantly outperforming both SI-GPDIC and SI-GPDTC. SI-GPDTC method also outperforms SI-GPDIC in PLL. These findings suggest that while all three methods are comparable in RMSE, SI-GPVI method is the preferred choice for



(a) (b)

Figure 6.3: Violin plot of RMSE & PLL for all GP methods on the U-shape domain: the dots at each end of the bold black lines represent the first and third quartiles and the white dot represents the median.

achieving higher predictive accuracy and reliability.

Wilcoxon Signed-Rank Test (RMSE)	GRAPH	VI	DIC&DTC
RBF	0.0005856	<0.0001	<0.0001
GRAPH		0.01718	0.003153
VI			0.33
Wilcoxon Signed-Rank Test (PLL)	VI	DIC	DTC
RBF	0.4749	0.0005856	0.8983
VI		<0.0001	0.0003223
DIC			<0.0001

Table 6.2: Wilcoxon Signed-Rank Test results for RMSE and PLL among different GP methods on the U-shape domain.

In conclusion, the U-shape has a very small gap between the upper and lower arms. The Euclidean distance between points on the upper and lower arms differs significantly from the true distance on the manifold. This difference leads to poor performance of Traditional GPs, which rely on Euclidean distance through RBF kernels for predictions. Similarly, Graph GP is based on a graph constructed from the manifold related to Euclidean distance, as shown in Equation (5.2). As a result, Graph GP also performs unsatisfactory. These two GP methods constrained by Euclidean distance and their assumption of smooth predictions, tends to assume that the

values of nearby points are similar, which differ significantly in the true function. This issue is effectively resolved by the three sparse intrinsic GP methods proposed in Chapter 4, which rely on the approximated heat kernel based on the manifold. Since the heat kernel does not have an explicit expression for most manifolds, it can be approximated using the transition densities of BM paths simulated from the inducing points z. These BM paths explore the manifold along its surface, capturing its intrinsic geometric features. The presence of the middle boundary of the U-shape ensures that these sparse intrinsic GPs provide very different predictions for the upper and lower arms. From Figures 6.1 and 6.2, the conclusion is clearly prove that the contour lines of Traditional GP and Graph GP smoothly cross the boundary, while SI-GPDIC, SI-GPDTC and SI-GPVI not. In the experiment, 20 sets of training points are randomly selected from grid points S on the U-shape, with each set containing 15 points. The sparse intrinsic GP methods performed better than the others, with SI-GPVI demonstrating superior performance overall. The results of descriptive statistics and Wilcoxon Signed-Rank Tests regarding RMSE and PLL for the various GP methods further confirm this observation.

6.3 The Implementation of Proposed Methods on the Bitten-torus

The Bitten-torus is used to study the effectiveness of the proposed Sparse Intrinsic GP methods in three-dimensional applications, as introduced in Section 1.3.2.

6.3.1 Construction of the Bitten-torus

In the Bitten torus example, "Bitten" refers to a portion of the Torus being removed, resembling a donut with a bite taken out of it, as shown in Figure 1.2. To construct the Bitten-torus, let the radius of tube $r = 5$, the distance from centre of the tube to the centre of the torus $R = 6$, the angle of torus $\theta = (0, 2\pi)$ and the angle of tube $\phi = (0.0625\pi, 1.98\pi)$. The Bitten-Torus can be expressed as:

$$\mathbf{X}(\theta, \phi) = ((R + r\cos\theta)\cos\phi, (R + r\cos\theta)\sin\phi, r\sin\theta).$$

By computing the partial derivatives for θ and ϕ:

$$\mathbf{X}_\phi = ((R+r\cos\theta)(-\sin\phi),(R+r\cos\theta)\cos\phi,0),$$
$$\mathbf{X}_\theta = (r\cos\phi(-\sin\theta),r\sin\phi(-\sin\theta),r\cos\theta).$$

Through

$$(\mathbf{X}_\theta\cdot\mathbf{X}_\theta)\,d\theta^2+2\left(\mathbf{X}_\theta\cdot\mathbf{X}_\phi\right)d\theta d\phi+\left(\mathbf{X}_\phi\cdot\mathbf{X}_\phi\right)d\phi^2 = r^2d\theta^2+(R+r\cos\theta)^2d\phi^2,$$

the metric tensor g for describing the Bitten-torus is easily to get:

$$g=\begin{bmatrix} r^2 & 0 \\ 0 & (R+r\cos\theta)^2 \end{bmatrix},$$

where r^2 represents the metric component in the direction of the small circle (along the θ direction) and $(R+r\cos\theta)^2$ describes the metric component in the direction of the large circle (along the ϕ direction). Then,

$$g^{-1}=\begin{bmatrix} \frac{1}{r^2} & 0 \\ 0 & \frac{1}{(R+r\cos\theta)^2} \end{bmatrix},$$
$$\frac{\partial g}{\partial\theta}=\begin{bmatrix} 0 & 0 \\ 0 & -2(R+r\cos\theta)r\sin\theta \end{bmatrix},\frac{\partial g}{\partial\phi}=\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Using the metric tensor of the Bitten-torus, the BM paths on the Bitten-torus can be constructed via the stochastic differential equations:

$$\begin{aligned} d\theta(t) &= \frac{1}{2}\left(-g^{-1}\frac{\partial g}{\partial\theta}g^{-1}\right)_{11}dt+\frac{1}{4}\left(g^{-1}\right)_{11}\mathrm{tr}\left(g^{-1}\frac{\partial g}{\partial\theta}\right)dt+\left(g^{-1/2}\right)_{11}dB_1(t) \\ &= -\frac{1}{2}r^{-1}\sin\theta(R+r\cos\theta)^{-1}dt+r^{-1}dB_1(t) \end{aligned},$$

$$\begin{aligned} d\phi(t) &= \frac{1}{2}\left(-g^{-1}\frac{\partial g}{\partial\phi}g^{-1}\right)_{22}dt+\frac{1}{4}\left(g^{-1}\right)_{22}\mathrm{tr}\left(g^{-1}\frac{\partial g}{\partial\phi}\right)dt+\left(g^{-1/2}\right)_{22}dB_2(t) \\ &= 0+0+\left|(R+r\cos\theta)^{-1}\right|dB_2(t) \\ &= \left|(R+r\cos\theta)^{-1}\right|dB_2(t) \end{aligned},$$

where $d\theta(t)$ and $d\phi(t)$ represent the BM paths in the $\theta(t)$ direction and $\phi(t)$ direction, respectively.

6.3.2 Implementation on the Bitten-torus

The 600 grid points are uniformly distributed on the surface of the Bitten-torus, providing a sufficiently dense grid. The BM paths are simulated from six inducing points selected from 600 grid points, evenly distributed along the ridge of the outer circumference of the Bitten-torus. Figure 6.4 (a) and (b) provide a top-down perspective and a side view, respectively, to present the Bitten-torus and show the positions of the inducing points (indicated by black dots). By using sparse method, the number of BM sample paths is decreased from $600 \times N$ to $6 \times N$, and in this case, $N = 50000$.

For 20 training points primarily distributed in the middle and lower regions of the Bitten-torus, Figure 6.4 provides the predictive mean of all GP methods. Among them, Figure 6.4 (c) shows the predictive mean of the Traditional GP. It is evident that the Traditional GP does not account for boundary effects, as it smoothly crosses the boundary of the bitten area. The upper boundary of the bitten area, which should originally display a high value in red, is incorrectly predicted as blue due to the influence of the training points below, indicating a lower value. This error occurs because the two ends of the "bitten" section are very close in Euclidean space. The Traditional GP considers only Euclidean distance, ignoring the intrinsic geometric structure of this manifold, causing the upper area to be influenced by the nearby training points below. Figure 6.4 (d) shows the predictive mean of the Graph GP, which makes predictions similar to the Traditional GP near the bitten area in the upper region. This is due to the constructed graph failing to accurately capture the local geometric features of the manifold, leading to inaccurate predictions. Additionally, the results of the Graph GP are not as smooth as those of the Traditional GP. Figure 6.4 (e) and (f) display the predictive means of SI-GPVI and SI-GPDIC & SI-GPDTC, respectively. In both cases, the predictions do not cross the boundary. Due to boundary effects, the BM paths simulated from inducing points in the lower region have difficulty reaching the upper boundary area along the surface within the given time and lengthscale constraints. All Sparse Intrinsic GPs effectively consider the intrinsic geometric features of the manifold. Table 6.3 provides the RMSE and PLL results for different GP methods in this ex-

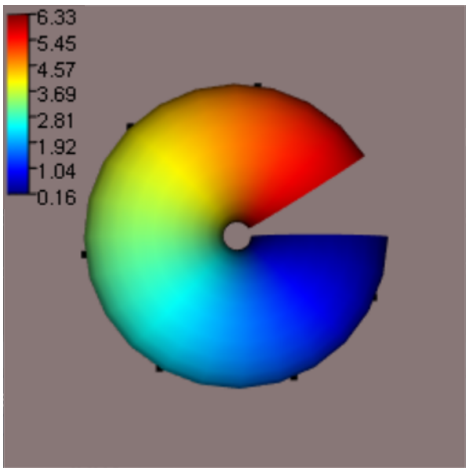
periment. The results in Table 6.3 demonstrate that the sparse methods (VI, DIC, and DTC) outperform the Traditional GPs and Graph GPs in terms of RMSE, with SI-GPVI achieving the lowest RMSE.

	Tra	Graph	VI	DIC	DTC
RMSE	2.265784	2.153284	1.372703	1.376375	1.376375
PLL	-4.96996		-2.629837	-5.728875	-1.79337

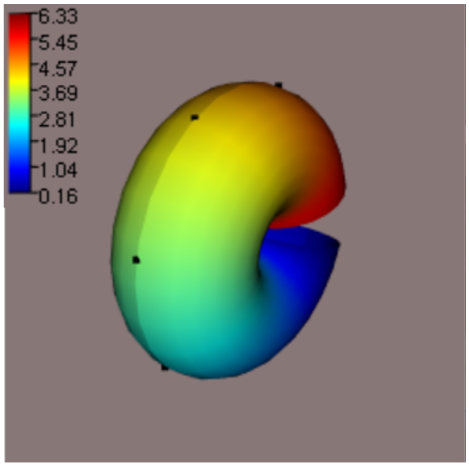
Table 6.3: RMSE and PLL among different GP methods on the Bitten-torus with 20 training points primarily distributed in the middle and lower regions of the Bitten-torus.

Next, 20 datasets will be randomly selected for the experiment to compare different methods. In the Bitten-torus example, as the number of grid points increases to 600, each dataset is set to contain 20 training points to maintain a proportionate sample size. Figure 6.5 and Figure 6.6 show the experimental results on one of these datasets. Figure 6.5 (a) shows the predictive mean of the Traditional GP. It can be seen that the inner area near the lower boundary is influenced by the training points from the upper region, turning red. This indicates that the Traditional GP crosses the boundary, thus affecting the prediction accuracy. Figure 6.6 (d), which shows the predictive mean of the graph GP, also exhibits the previously discussed issue, with the inner area near the lower region turning red. The affected area is more prominent compared to the Traditional GP. Figure 6.5 (b) presents the predictive variance of the Traditional GP, which shows no apparent color difference due to the very small variance values. Figure 6.5 (c) and (d) show the predictive mean and variance of the SI-GPVI. Consistent with the analysis from previous experiments, the Sparse Intrinsic GP does not cross the boundary and effectively considers the manifold's boundary conditions. The results for SI-GPDIC and SI-GPDTC are similar, as shown in Figure 6.6 (a). The predictive variance of SI-GPDIC method (shown in Figure 6.6 (c)), in particular, shows minimal color difference due to very small predictive variance, which also leads to poorer performance in subsequent PLL comparison.

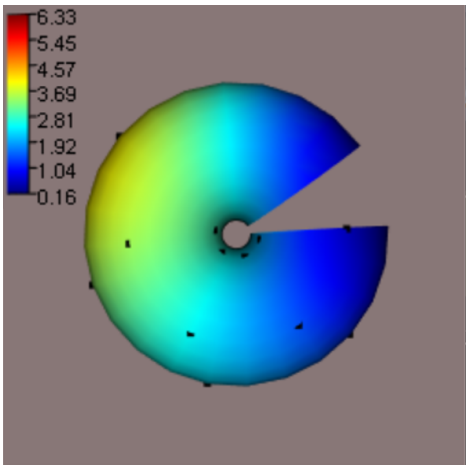
Table 6.4 provides an analysis of RMSE and PLL values across different GP methods on the Bitten-torus example. It shows that Sparse Intrinsic GP methods as a whole outperform Traditional and Graph GP methods in terms of both RMSE and PLL on the Bitten-torus manifold. In comparison, Traditional GPs and Graph GPs exhibit higher RMSE values and greater variability, indicating less precise and consistent performance. Although SI-GPDIC method has a lower RMSE value, its poorer PLL performance (mean of -8.066, range of 22.077) suggests



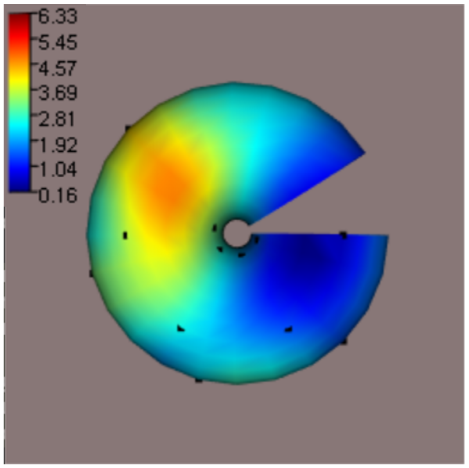
(a) Top-down perspective of objective function



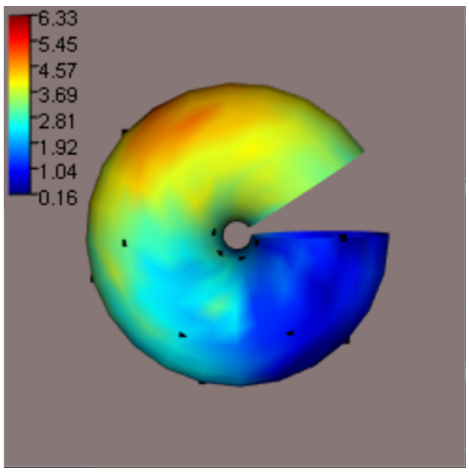
(b) side perspective of objective function



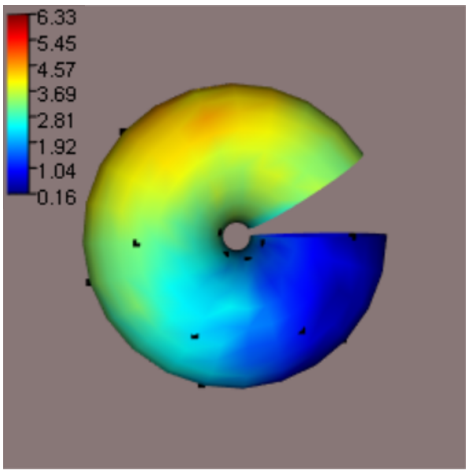
(c) Mean of Traditional GP



(d) Mean of Graph GP

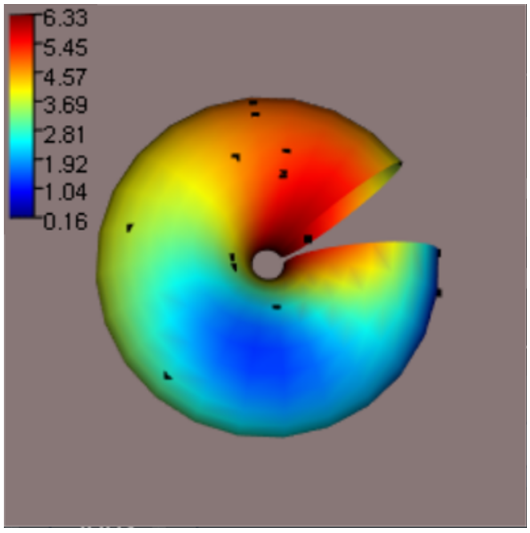


(e) Mean of SI-GPVI

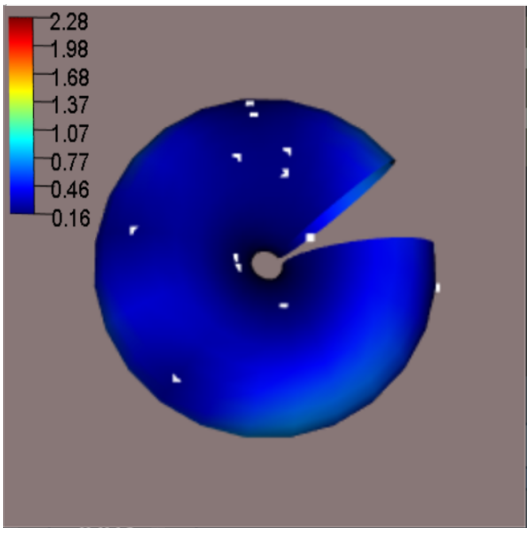


(f) Mean of SI-GPDIC & SI-GPDTC

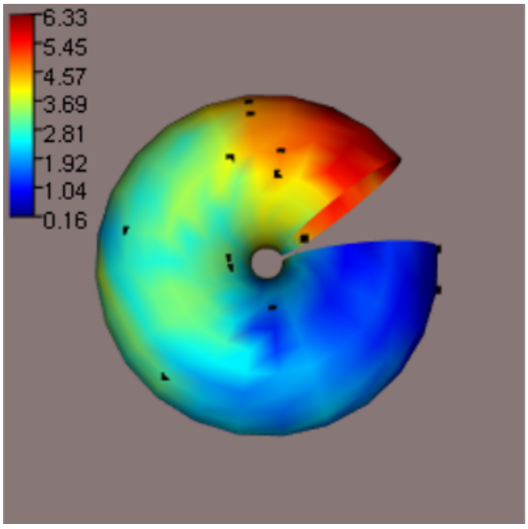
Figure 6.4: With 20 training points primarily distributed in the middle and lower regions of the Bitten-torus: (a) and (b) show the positions of the 6 inducing points used in the sparse intrinsic GP, represented by black dots; in the remaining figures, the black dots represent the positions of the 20 training points. (c) the predictive mean of Traditional GP; (d) the predictive mean of Graph GP; (e) the predictive mean of SI-GPVI; (f) the predictive mean of SI-GPDIC & SI-GPDTC.



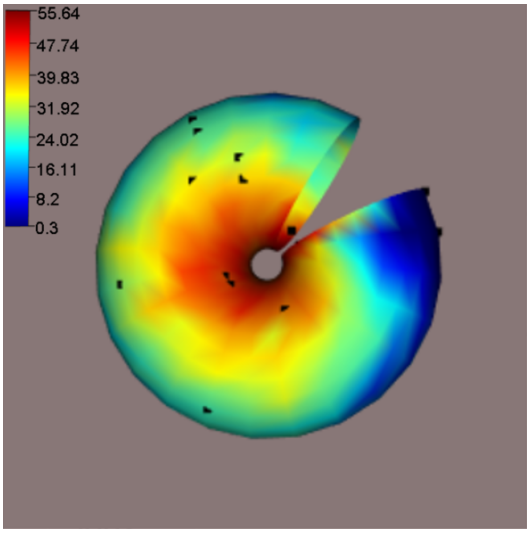
(a) Mean of Traditional GP



(b) Variance of Traditional GP

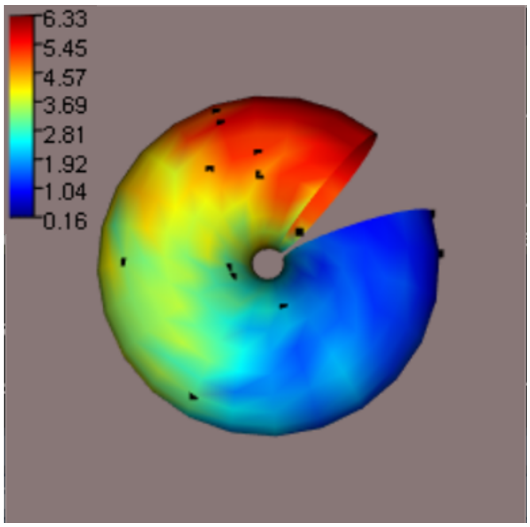


(c) Mean of SI-GPVI

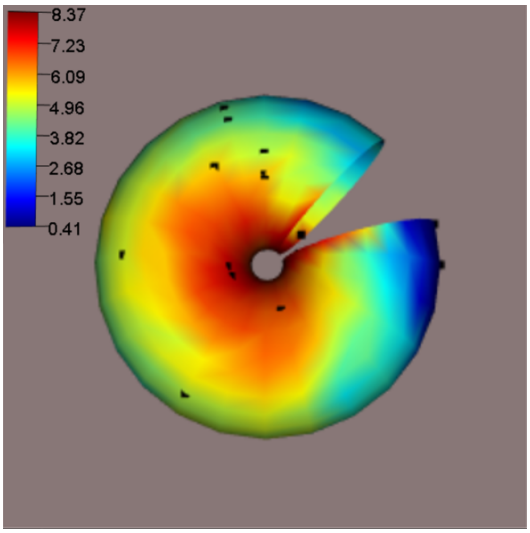


(d) Variance of SI-GPVI

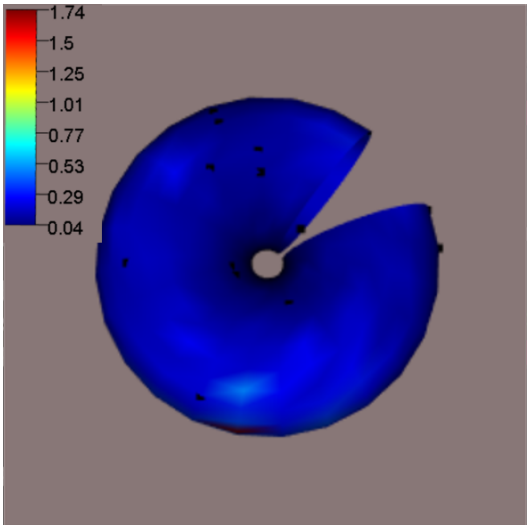
Figure 6.5: With 20 training points randomly selected shown as black dots on the Bitten-torus: (a)-(b) Predictive mean and predictive variance of Traditioanl GP; (c)-(d) Predictive mean and predictive variance of SI-GPVI



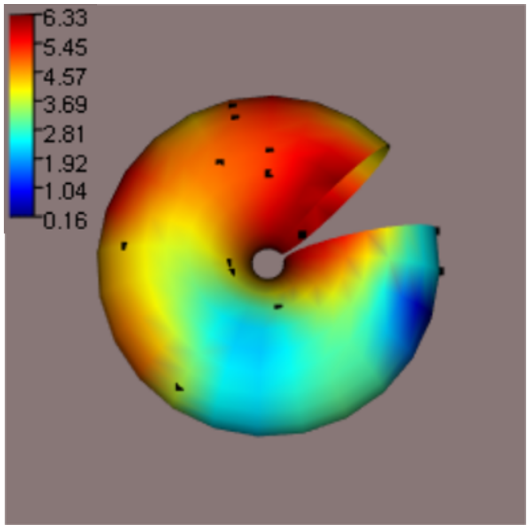
(a) Mean of SI-GPDIC & SI-GPDTC



(b) Variance of SI-GPDTC



(c) Variance of SI-GPDIC



(d) Mean of Graph GP

Figure 6.6: With 20 training points randomly selected shown as black dots on the Bitten-torus: (a) Predictive mean of SI-GPDIC & SI-GPDTC; (b) Predictive variance of SI-GPDTC; (c) Predictive variance of SI-GPDIC; (d) Predictive mean of Graph GP.

lower reliability. SI-GPVI achieves the lowest mean RMSE (0.9500) and the smallest range (0.1469). The reliability of SI-GPVI is emphasized by the PLL results, which show consistent probabilistic predictions with a mean PLL value of -2.663 and a range of 0.185. To substantiate these findings, a series of Wilcoxon Signed-Rank Tests were conducted to validate the statistical significance of the differences in RMSE and PLL between different GPs. The Wilcoxon Signed-Rank Test results shown in Table 6.5 confirm that the Sparse Intrinsic GP methods are significantly superior to both Traditional GP and Graph GP in terms of RMSE. Among them, SI-GPVI further demonstrates a statistically significant advantage over the other methods in RMSE performance.

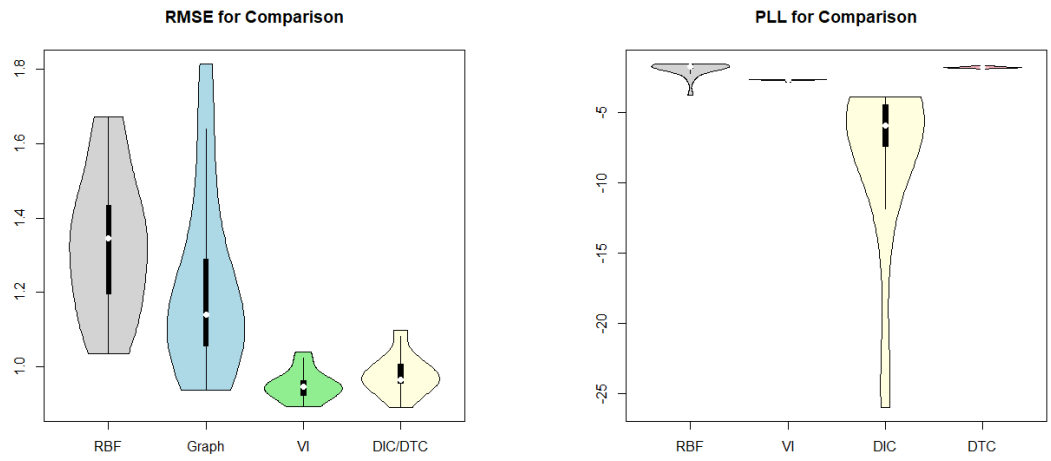
Summary (RMSE)	Min.	Median	Mean	Max.	Range
Tra	1.035	1.345	1.329	1.672	0.637
Graph	0.9368	1.1394	1.2147	1.8147	0.835
VI	0.8930	0.9452	0.9500	1.0399	0.1469
DIC/DTC	0.8908	0.9661	0.9780	1.0986	0.3542
Summary (PLL)	Min.	Median	Mean	Max.	Range
Tra	-3.749	-1.734	-1.888	-1.540	2.209
VI	-2.817	-2.650	-2.663	-2.632	0.185
DIC	-25.967	-5.923	-8.066	-3.890	22.077
DTC	-1.897	-1.790	-1.783	-1.690	0.207

Table 6.4: Statistical summary of RMSE & PLL for all GP methods on the Bitten-torus.

Wilcoxon Signed-Rank Test (RMSE)	GRAPH	VI	DIC/DTC
RBF	0.01069	<0.0001	<0.0001
GRAPH		<0.0001	<0.0001
VI			0.01069
Wilcoxon Signed-Rank Test (PLL)	VI	DIC	DTC
RBF	0.0007076	<0.0001	0.5706
VI		<0.0001	<0.0001
DIC			<0.0001

Table 6.5: Wilcoxon Signed-Rank Test results for RMSE and PLL among different GP methods on the Bitten-torus.

These results are shown visually in Figure 6.7, where (a) is the violin plot for the RMSE results, and (b) corresponds to the PLL results. The dots at each end of the bold black lines represent the first and third quartiles, while the white dot marks the median. SI-GPVI exhibits the lowest values in RMSE, and smallest spread in both RMSE and PLL, indicating its robustness and accuracy.



(a) (b)

Figure 6.7: Violin plot of RMSE & PLL for all GP methods on the Bitten-torus: the bold black lines at each end represent the first and third quartiles and the white dot represents the median.

In conclusion, the Bitten-torus, as a representation of a 3-dimensional manifold, is similar to the U-shape example in that points on either side of the "bite" may be close in Euclidean distance but far apart in the manifold's intrinsic geometry. By comparing the five GP methods on 20 randomly selected training sets (each containing 20 points), the Traditional GP and Graph GP, both relying on Euclidean distance, perform worse than the sparse intrinsic GP methods. Among them, SI-GPVI reconfirms its superior performance. Descriptive statistics and Wilcoxon Signed-Rank Tests used in this section further help to confirm this conclusion.

6.4 The Implementation of Proposed Methods on the Aral Sea

The example of real-world data is predicting chlorophyll level in the Aral Sea, which has been introduced in Section 1.3.3. Chlorophyll level is an important parameter for measuring the biomass and primary productivity of marine phytoplankton. It is also a key indicator of the degree of eutrophication and the occurrence of red tides in water bodies. Monitoring chlorophyll level in water bodies is of significant importance for environmental protection. This case also adds significance to the research on sparse intrinsic GPs on manifolds, highlighting the potential for numerous future applications. For sparse intrinsic GPs, given that the Aral Sea serves as a

real-world dataset, with its objective function being non-smooth and noisy, it is essential to set a larger number of inducing points to provide better coverage across the manifold and capture the intricate patterns more effectively. Then, 10 inducing points are selected from 485 grid points, evenly distributed within the boundary of the Aral Sea, represented by the little circles in Figure 6.8 (a), ensuring that the BM paths originating from them can adequately explore the entire domain of the manifold. Consequently, the starting points for the BM are reduced from 485 grid points to only 10 inducing points. The number of BM sample paths is decreased from $485 \times N$ to $10 \times N$, where $N = 50000$.

For 35 training points nearly equally spaced in the right half of the Aral Sea, Figures 6.8 and 6.9 present the results of each GP methods. Figure 6.8 (a) shows the true values of chlorophyll level in the Aral Sea, with white circles representing the 10 inducing points. The chlorophyll level values range from 0 to 19.278724, with colors in Figure 6.8 (a) transitioning from light yellow to dark red. The lower part of the Aral Sea is divided by the isthmus of the central peninsula, with lower chlorophyll concentrations on the left and higher concentrations on the right. In the remaining panels of Figure 6.8 and in subsequent figures of the Aral Sea, the white circles denote the training points. Figure 6.8 (b) and (c) respectively show the predictive mean and variance of the Traditional GP. It can be observed that due to the smooth nature of the RBF kernel, the predictive mean and variance in the lower left region near the right boundary are unreasonably influenced by the training points in the right region. The RBF kernel does not account for the existence of physical boundaries, leading to an interaction of values across different regions separated by land. Similarly, the Graph GP shown in Figure 6.8 (d), which first constructs a graph on the manifold and then applies the Graph Matérn kernel, fails to accurately capture the manifold's intrinsic geometric features. As a result, it exhibits a similar trend of incorrect predictions in the lower left region near the right boundary. In Figure 6.9, the results of different sparse intrinsic GP methods for 35 training points are presented. Specifically, Figure 6.9 (a) displays the predictive mean of SI-GPVI. Unlike the issues observed with Traditional GP and Graph GP, SI-GPVI effectively avoids these problems. Due to boundary effects, when calculating the heat kernel in the left region, very few BM paths from the right region's inducing points contribute, minimizing the influence from the right region. This method effectively utilizes the internal geometric features of the manifold. Figure 6.9 (c) shows the predictive mean of SI-GPDIC and SI-GPDTC, following the same principle. In Figures 6.9 (b) and (e), which show

the predictive variance of SI-GPVI and SI-GPDTC respectively, the variance in the left region shows high values due to boundary-induced separation from training points, indicating high uncertainty. However, in Figure 6.9 (d), the predictive variance of SI-GPDIC reveals that the left region does not show similarly high uncertainty. This difference arises from the sensitivity of SI-GPDIC to the positions of the inducing points, causing the variance to notably decrease as it approaches the left boundary, moving further from the inducing points.

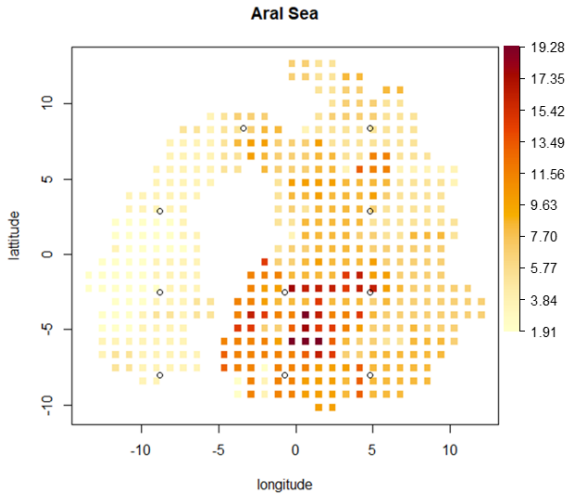
Table 6.6 presents the comparison of RMSE and PLL values for different GP methods under this set of experiments. Sparse methods, particularly SI-GPVI, demonstrate superior performance in terms of RMSE. While SI-GPDTC method excels in PLL, SI-GPVI remains robust across both metrics.

	Tra	Graph	VI	DIC	DTC
RMSE	2.816694	2.816725	2.669898	2.729834	2.729834
PLL	-2.754867		-3.900971	-4.776345	-2.459444

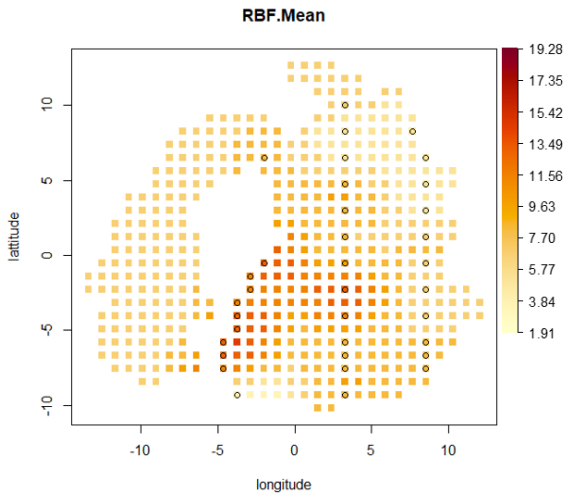
Table 6.6: RMSE and PLL among different GP methods on the Aral Sea with 35 training points.

Given the non-smooth and noisy nature of the objective function, the number of randomly selected training points is increased to 30 in 20 experimental sets to allow for a comparison of the prediction accuracy across different methods. Figures 6.10 and 6.11 display the results of one of these groups. Figures 6.10 (a) and (b) show the predictive mean and variance of the Traditional GP, respectively. Similar to the predictive mean in Figure 6.11 (d) for the Graph GP, these figures exhibit the same characteristics observed with the previous 35 training points case. However, due to the randomness in selecting training points, the training points in the left region now provide information about that area, reducing the influence of values from the right region on the predictions for the left region. Figures 6.10 (c) and (d) show the predictive mean and variance corresponding to SI-GPVI, respectively. Due to the separation by land, the values on either side do not influence each other. Figures 6.11 (a), (b), and (c) display the predictive mean and variance for SI-GPDIC and SI-GPDTC, respectively, and exhibit similar behavior to SI-GPVI.

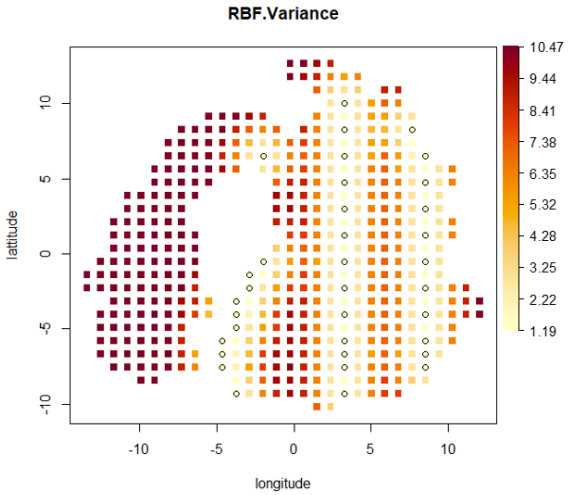
Table 6.7 provides an analysis of RMSE and PLL values across different GP methods, offering a comprehensive evaluation of their performance on the Aral Sea dataset. Table 6.7 indicates that SI-GPVI achieves the lowest mean RMSE value of 2.370, which means it offers the highest



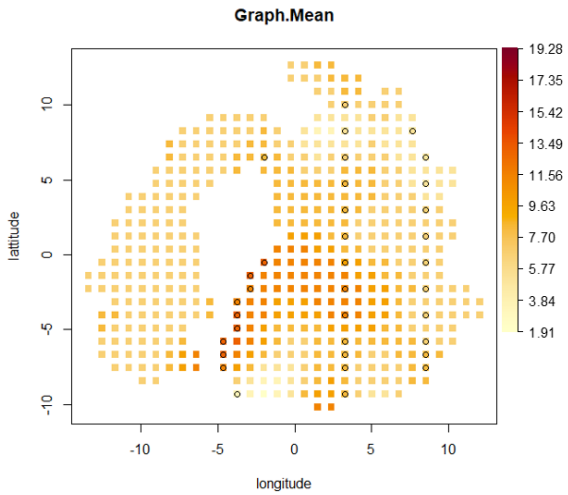
(a) Objective function



(b) Mean of Traditional GP

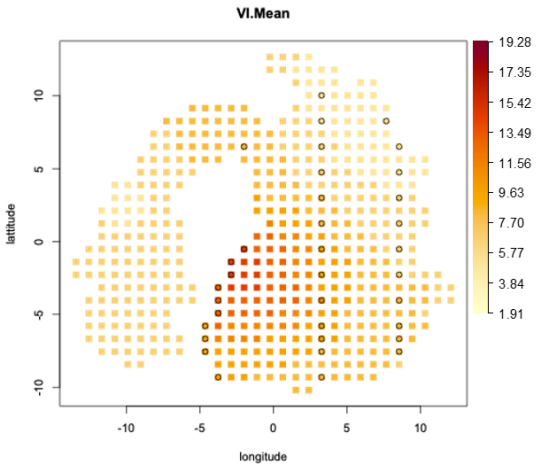


(c) Variance of Traditional GP

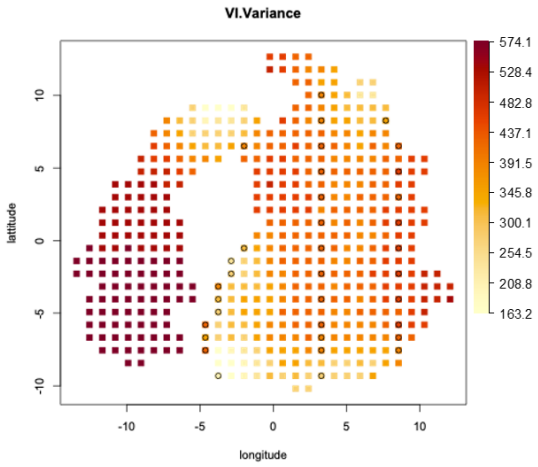


(d) Mean of Graph GP

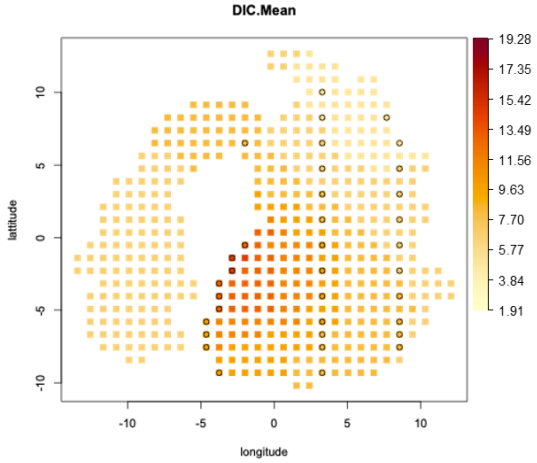
Figure 6.8: With 35 training points nearly evenly distributed across the right half of the Aral Sea, separated by land, (a) displays the true chlorophyll levels of the Aral Sea, with circles representing the inducing points used in the sparse intrinsic GPs. In (b)-(f), circles mark the positions of the 35 training points. Specifically, (b) and (c) show the predictive mean and variance of the Traditional GP, respectively, while (d) illustrates the predictive mean generated by the Graph GP method.



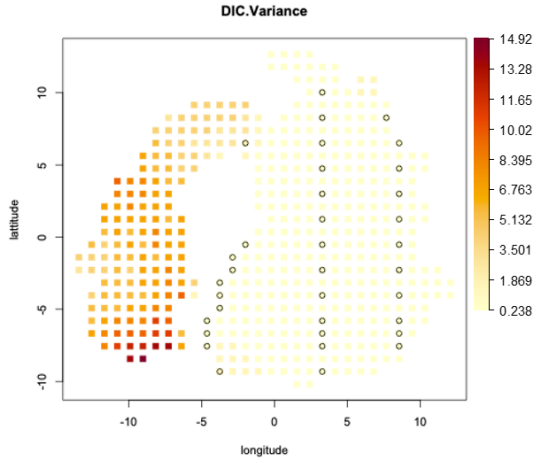
(a) Mean of SI-GPVI



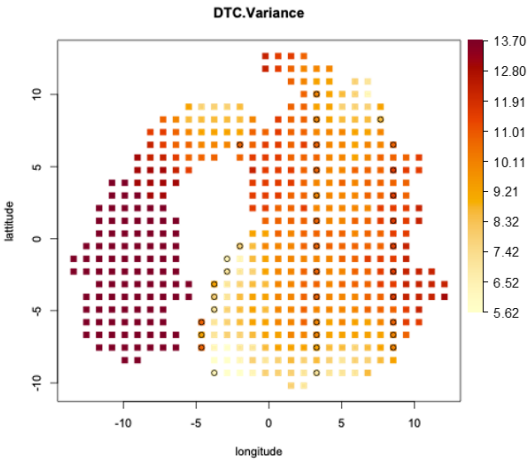
(b) Variance of SI-GPVI



(c) Mean of SI-GPDIC & SI-GPDTC

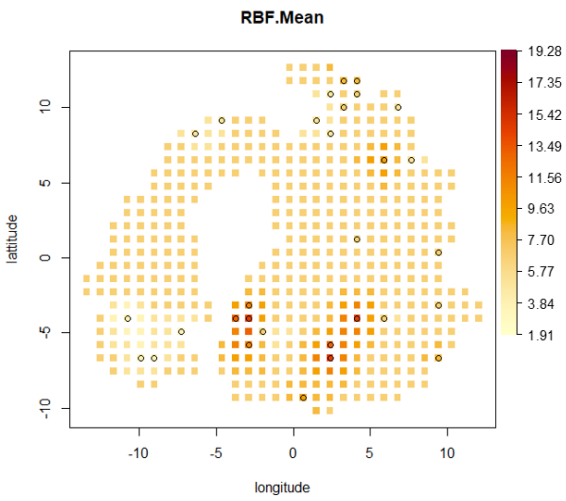


(d) Variance of SI-GPDIC

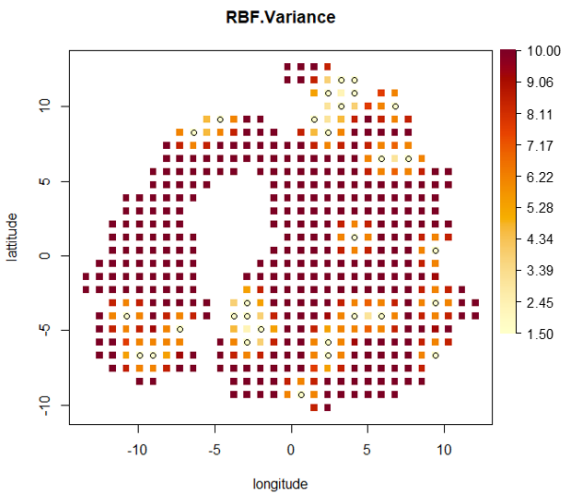


(e) Variance of SI-GPDTC

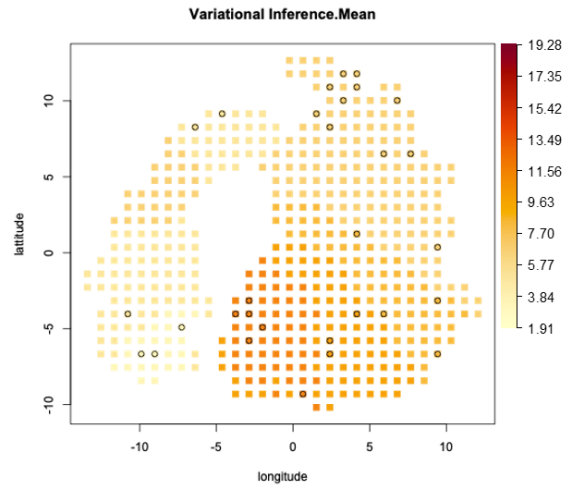
Figure 6.9: With 35 training points nearly evenly distributed in the right half of the Aral Sea separated by land, represented by circles on plots: (a) and (b) illustrate the predictive mean and variance of SI-GPVI; (c) is the Predictive mean of SI-GPDIC and SI-GPDTC; (d) and (e) represent the predictive variance of SI-GPDIC and SI-GPDTC respectively.



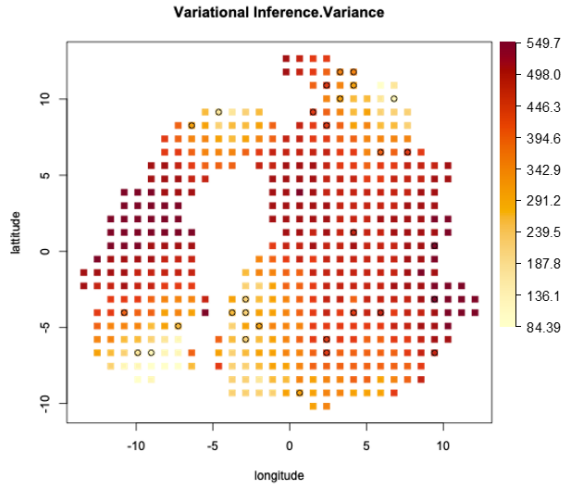
(a) Mean of Traditional GP



(b) Variance of Traditional GP

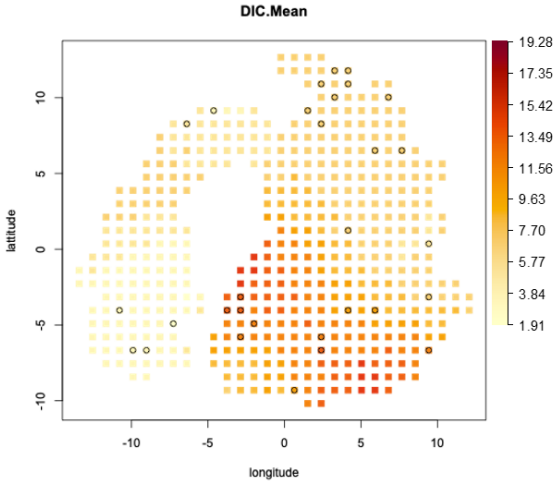


(c) Mean of SI-GPVI

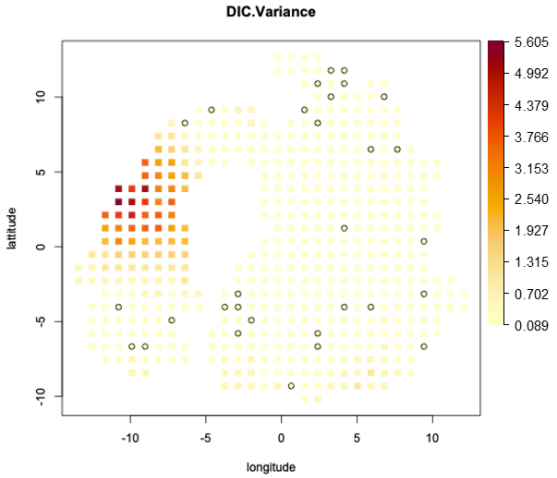


(d) Variance of SI-GPVI

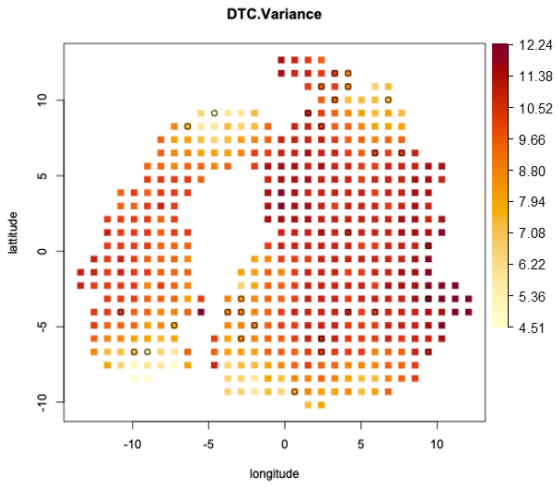
Figure 6.10: With 30 training points randomly selected shown as circles on the Aral Sea: (a)-(b), Predictive mean and predictive variance of Traditional GP; (c)-(d), Predictive mean and predictive variance of SI-GPVI



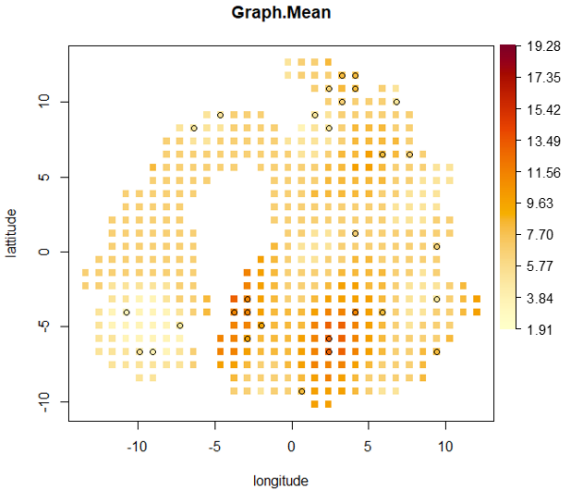
(a) Mean of SI-GPDIC & SI-GPDTC



(b) Variance of SI-GPDIC



(c) Variance of SI-GPDTC



(d) Mean of Graph GP

Figure 6.11: With 30 training points randomly selected shown as circles on the Aral Sea: (a), Predictive mean of SI-GPDIC & SI-GPDTC; (b), Predictive variance of SI-GPDIC; (c), Predictive variance of SI-GPDTC; (d), Predictive mean of Graph GP.

prediction accuracy among all methods. The range of RMSE values for SI-GPVI is also the smallest (0.577), indicating stable performance with less variability in prediction accuracy. As for probabilistic predictions, SI-GPVI maintains a mean PLL of -3.860 with a specifically narrow range of 0.051, indicating reliable and balanced performance comparing with other methods. The Wilcoxon Signed-Rank Test results in Table 6.8 further support the robustness of SI-GPVI. The results show a statistically significant improvement in RMSE for SI-GPVI compared to the Traditional GP (p = 0.0002098), proving a higher prediction accuracy. Other comparisons, such as the RMSE between Graph GP and SI-GPVI (p = 0.6477), do not show significant differences, indicating that these methods perform equally well in certain aspects. However, the probabilistic predictions of the Graph GP cannot be used. This limitation must be taken into account when considering its overall effectiveness. As a result, while the Graph GP demonstrates comparable accuracy in some indices, its utility is constrained by the unreliability of its probabilistic outputs. In the Aral Sea, SI-GPDIC and SI-GPDTC offer improve better RMSE results over Traditional GP, but lagging behind SI-GPVI in both accuracy and consistency. SI-GPDTC shows better probabilistic prediction performance compared to SI-GPVI and SI-GPDIC, with a mean PLL of -2.544. However, it exhibits greater variability in probabilistic predictions, as reflected by a wider PLL range compared to SI-GPVI.

Summary (RMSE)	Min.	Median	Mean	Max.	Range
Tra	2.162	2.485	2.554	3.375	1.213
Graph	2.086	2.318	2.407	2.921	0.835
VI	2.164	2.318	2.370	2.741	0.577
DIC/DTC	2.094	2.432	2.504	3.339	1.245
Summary (PLL)	Min.	Median	Mean	Max.	Range
Tra	-3.614	-2.422	-2.514	-2.082	1.532
VI	-3.892	-3.860	-3.860	-3.841	0.051
DIC	-10.542	-3.483	-4.017	-2.484	8.058
DTC	-4.004	-2.400	-2.544	-2.215	1.789

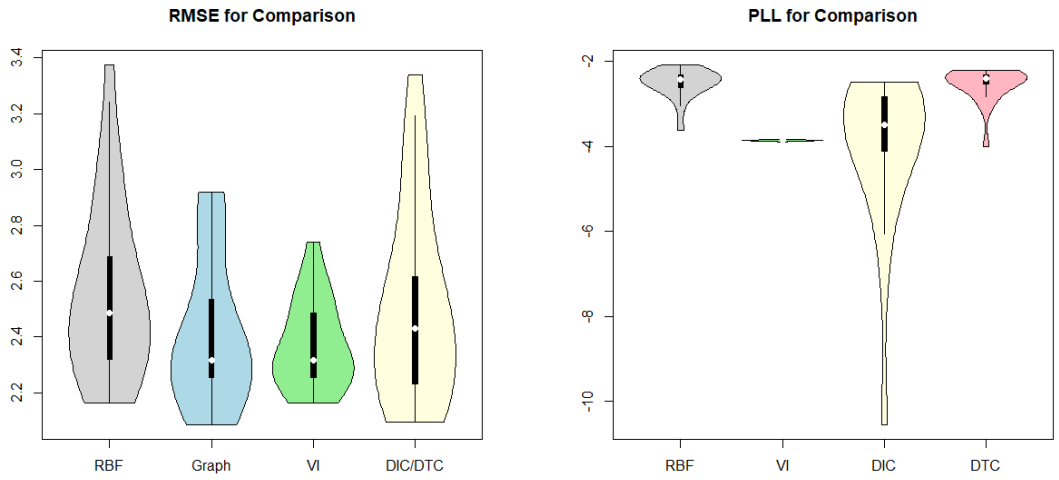
Table 6.7: Statistical summary of RMSE & PLL for all GP methods on the Aral Sea.

The violin diagram in Figure 6.12 visualizes the above findings, where (a) is the violin plot for RMSE, and (b) corresponds to the PLL results. Among it, the dots at each end of the bold black lines represent the first and third quartiles, while the white dot represents the median. The plots indicate that SI-GPVI has a more concentrated distribution for both RMSE and PLL, reflecting consistent performance.

Wilcoxon Signed-Rank Test (RMSE)	GRAPH	VI	DIC/DTC
RBF	0.01718	0.0002098	0.1429
GRAPH		0.6477	0.6215
VI			0.165
Wilcoxon Signed-Rank Test (PLL)	VI	DIC	DTC
RBF	<0.0001	<0.0001	0.8408
VI		0.5958	<0.0001
DIC			<0.0001

Table 6.8: Wilcoxon Signed-Rank Test results for RMSE and PLL among different GP methods on the Aral Sea.

In this example, the reason that the differences between the methods are not particularly large is that real-world data has sudden changes, noise, and irregularities that are less than smooth. When the data is highly unsmooth, the predictive power of the various GP methods decreases, thus reducing the apparent difference in prediction accuracy. However, under these conditions, SI-GPVI demonstrates better performance, with smaller ranges in RMSE and PLL values, indicating more consistent and stable performance.



(a) (b)

Figure 6.12: Violin plot of RMSE & PLL for all GP methods on the Aral Sea: the dots at each end of the bold black lines represent the first and third quartiles and the white dot represents the median.

6.5 Conclusion

In the previous chapters, several sparse intrinsic GP methods tailored for manifolds have been proposed. These methods are designed to efficiently handle regression on complex manifolds. In addition to these methods, Traditional GP and Graph GP are introduced for comparison.

This chapter uses three representative examples as introduced in Section 1.3: the two-dimensional U-shape, the three-dimensional Bitten-torus, and a real-world dataset, Aral Sea. Various GP methods are applied to these three examples to evaluate and compare their performance. Section 6.1 introduces two statistical metrics used for comparison: RMSE to evaluate the measurement deviation and PLL to evaluate the predictive probability density function which additionally considers the predictive variance.

Across all three examples—the U-shape, Bitten-torus, and Aral Sea—the performance of the GP methods are consistent. The proposed sparse intrinsic GPs, particularly SI-GPVI, outperform both the Traditional GP and Graph GP in terms of RMSE and PLL. Statistical analysis and Wilcoxon Signed-Rank Tests confirm the superiority of SI-GPVI, highlighting its greater predictive accuracy and reliability. SI-GPDIC and SI-GPDTC also demonstrate notable improvements over the Traditional GP, but are less consistent than SI-GPVI. The small differences between GPs in the Aral Sea can be attributed to the inherent noise and irregularity of real-world data, which brings challenges to prediction. Despite this, SI-GPVI remains the most robust and stable of all the examples. Through these experiments, SI-GPVI proves to be a highly effective and powerful method for regression on manifolds.

Given the superiority of the proposed methods in predicting over manifolds, exploring the implementation of Bayesian Optimisation (BO) on manifolds becomes feasible. BO is a global optimisation method based on Bayesian statistics and GPs, which is mainly used to optimise costly black-box functions, and performs particularly well when dealing with computationally expensive or experimentally costly optimisation problems. In the past, BO is usually used for optimisation problems in Euclidean space, while the method proposed in this thesis extends the application of BO to manifolds. In the next chapter, the implementation of BO on manifolds with SI-GPVI, SI-GPDIC, and SI-GPDTC as proxy functions will be developed. This is not only an extension of the application of the BO algorithm itself, but also effectively solves the

optimisation problem on manifolds, providing substantive practical importance.

Chapter 7

Intrinsic Bayesian Optimisation on Manifolds

When considering the mathematical optimisation problem, how to successfully achieve optimisation goals on manifolds presents unique challenges when constrained by complex internal geometry. The problem exists in how to appropriately incorporate the geometric properties of manifolds into the optimisation process. For example, measuring and monitoring pollutants in lakes is an initiative in environmental protection. How to consume less resources to find the most polluted areas can effectively reduce the cost of environmental remediation. Lakes can be viewed as manifolds with complex geometrical features, and the problem can be viewed as how to find a specific area within a limited budget, such as detecting the area of highest pollution levels. Chapter 4 proposes several Sparse Intrinsic GPs, which are shown to be suitable for predicting the surface of measurements after appropriately accounting for the geometry of a manifold. With this foundation, exploring the implementation of Bayesian Optimisation (BO) on manifolds becomes feasible. While Chapter 6 demonstrated that SI-GPVI is superior in estimating the smooth surface and related variance over the manifold, it is also important to explore and compare the performance of all approaches—SI-GPDIC, SI-GPDTC, and SI-GPVI, as well as the Graph GPs under the BO framework. The iteration of BO begins with a very limited number of initial points. This approach causes each point selected by the acquisition function in the early regression model to significantly influence the outcomes of subsequent iterations, introducing a degree of uncertainty in the early stages. Consequently, it is essential to explore

and compare different methods when uncertainty appears in the early iterations.

This chapter aims to propose several frameworks to support the optimisation problem over manifolds, which begins with a brief introduction of BO and defines the optimisation problems to be addressed. Then, Section 7.2 details the Traditional BO, which has already been proficiently implemented in Euclidean space, describing its algorithmic structure on manifolds. Section 7.3 proposes the algorithm of Intrinsic BO with DIC, and also gives the introduction of Intrinsic BO with DTC method. In Section 7.4, when using SI-GPVI as a proxy function, the Intrinsic BO with VI method is proposed, and is expected to demonstrate excellent performance due to its surrogate model. Section 7.5 introduces Graph BO, which considers the manifold as a graph composed solely of points. Through comparative analysis and simulation studies, Section 7.6 evaluates the performance of all algorithms across various complex manifolds, similar to the examples in Chapter 6, which also includes one application in a real-world scenario. Section 7.7 provides a summary of this chapter.

7.1 Introduction of Bayesian Optimisation

Optimisation problems arise in all quantitative disciplines from statistics [100], [6] and computer science [63], [138], [146] to engineering [104], [49] and economics [24]. In recent years, the optimisation objective is no longer limited to Euclidean space with simple spatial structure. Optimisation problems on manifolds have gradually become a research hot topic of interest. For example, measuring and monitoring pollutants in lakes is an environmental protection initiative. The health of lakes can be effectively assessed through the measurement of a number of pollution indicators, such as the concentration of algal blooms, the level of dissolved oxygen and the concentration of nutrients such as nitrogen and phosphorus [22]. Using these data, the most polluted areas can be located so that targeted treatment measures can be taken with the aim of doing this in the most cost effective way. BO is an effective solution to solve optimisation problems in Euclidean space when the objective function is unknown [135]. It outperforms other leading-edge global optimisation algorithms on a wide range of challenging optimisation benchmark functions [70]. Given the wide application of BO in Euclidean spaces for optimising expensive black-box functions to explore global optima, this chapter will consider extending the

application of BO based on GPs to the realm of manifolds. Compared to simple geometries, manifolds usually exhibit more complex intrinsic geometric characteristics such as topology, connectivity and smoothness, as well as complex boundaries. These features will impact how data points are related to each other, tending to affect the modeling and prediction of surfaces over the manifolds, and hence the decision making in practical application fields. Failing to account appropriately for the structure of the manifold in BO may result in significant relationships and dependencies within the data being missed, resulting in less reliable results. Accounting for intrinsic geometric features of manifolds in the modeling and optimisation process is a major challenge in the study of manifolds and the main focus of the work that follows.

In practical applications, data collection capabilities are often limited by budget constraints, resulting in investigation of minimal sampling designs required in order to estimate variables of interest across a manifold. Statistical modelling including GP regression and BO, in particular, are useful tools to estimate e.g. a pollutant surface, which means it is unnecessary to perform continuous dense sampling around each point. For example, when placing sensors in lakes or fields, a single sensor typically covers a small surrounding area, removing the need to waste resources by placing sensors multiple times in the area. For BO, when the context is a smooth latent continuous surface, the search process of BO may sometimes involve repeatedly exploring a small neighborhood around a point. Due to the smoothness assumption of the latent function, it is generally expected that points within a small neighborhood exhibit similar characteristics. When a point is identified as the current optimum, BO may conduct dense sampling in that region. Such repetitive sampling can become inefficient, as it consumes computational resources without yielding proportionate gains in information or performance, ultimately leading to resource waste and an increase in unnecessary computational burden. As a result, the optimisation context shifts from a manifold with a continuous surface to one where observations can only be collected from grid points that are densely and uniformly distributed across the manifold, which can be seen as an exploration constraint that effectively reduces resource waste. For example, in Figure 1.3, the Aral Sea is represented by grid points, helping to illustrate this approach. However, the number of grid points is finite, and when there is no limit to the number of iterations of BO, BO can definitely find the optimal point. In an extreme case where the number of iterations approaches the total number of grid points, it is equivalent to sampling from all grid points, which is practically meaningless. Considering budget constraints, it is expected that BO can

find the optimal point from grid points on the manifold within an effective number of iterations. For example, the goal might be to find the optimal point successfully within sampling iterations that are 5% of the total number of grid points.

Then, the problem can be posed as trying to find the optimal point within these grid points in a limited number of iterations (5% $G^{'}$, where $G^{'}$ is the number of grid points), so that the corresponding implicit objective function reaches its global maximum (or minimum). M is previously defined as a d dimensional complete Riemannian manifold, which is also the submanifold of a higher dimensional Euclidean space R^p, $d \leq p$. This problem can be described as:

Given: The objective function $f(s)$ is defined on the manifold M and lacks an explicit analytical expression.

Objective: Identify the optimal point s_M by solving the following optimisation problem:

$$s_M = \text{argmax}_{s \in S} f(s),$$

where S represents a finite set of grid points discretely sampled on the manifold M.

To better illustrate the problem, consider the task of finding the location with the highest chlorophyll levels in the Aral Sea [115] in Figure 7.1 which is introduced in Section 6.4. The distribution of the chlorophyll levels in the constrained domain is unknown and can be treated as a black-box function. The geometry of the constrained domain is also different from the Euclidean space R^2. Two locations that have a close Euclidean distance on a map may be intrinsically far apart if they are separated by a land barrier, for example, in the lower part of the Aral Sea. In BO, if very few initial points are located in the left region near the right boundary, the Euclidean distance might cause the predictive mean in the right region, which is separated by a land barrier, to be smoothly wrong influenced by the left side. Additionally, the predictive variance in the right region may be relatively small. This can easily lead to BO missing an appropriate exploration of the right region.

In Chapter 4, the SI-GPDIC, SI-GPDTC and SI-GPVI methods are proposed, where the kernels used in the regression are a heat kernel approximated via the transition density of BM

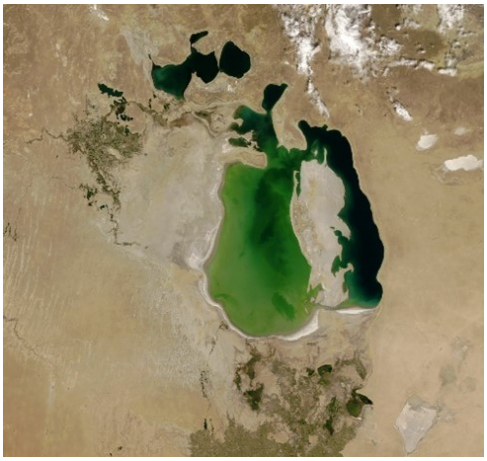


Figure 7.1: Satellite imagery of the Aral Sea (shaded green to black in colour), an endorheic basin (saltwater lake) in Central Asia [115].

paths simulating from inducing points. These approaches effectively incorporate the intrinsic geometric structure and boundary information of the manifold. Chapter 6 demonstrates the effectiveness of these methods through the application to three different examples, where SI-GPVI achieved better results, as evidenced by a comprehensive analysis of RMSE and PLL. Compared to the Traditional GP introduced in Chapter 3 and the Graph GP introduced in Chapter 5, which show dependency on the distribution of training points, the three sparse intrinsic GP methods are more robust, exhibiting less sensitivity to the distribution of training points. Building on these foundations, this chapter introduces three methods: Intrinsic BO with DIC, DTC, and VI. Additionally, Traditional BO and Graph BO methods are developed based on the Traditional GPs and Graph GPs discussed in earlier chapters. The performance of all these methods is rigorously analyzed through practical applications in Section 7.6. Among them, Intrinsic BO with VI demonstrates superior results within a limited number of iterations compared to the other methods.

7.2 Algorithm for Traditional Bayesian Optimisation

Traditional BO is widely applied in Euclidean spaces, including two core components: the probabilistic surrogate model and the acquisition function. The probabilistic surrogate model used

here is the Traditional GP based on the RBF kernel, as introduced in Section 3.2. Chapter 6 presents the application of the Traditional GPs on manifolds. For example, Figure 6.1 (a), (b) shows the predictive mean and predictive variance of Traditional GP on the U-shape domain, respectively. The values on either side of the middle gap influence each other due to their close Euclidean distance.

Based on the Traditional GPs obtained in Chapter 3, the acquisition function focuses on the Probability of Improvement (PI), which evaluates and identifies the most promising point within $s^* \in S$ (where S is previously defined as the grid points on the manifold M) to serve as the target for the subsequent exploration [82]. The "promising" refers to a new point that has the highest likelihood of yielding a better outcome than the best result observed so far (if the global optimum being attempted is the maximum value, then the "better outcome" refers to the generation of a new maximum value). The acquistion function PI is used here to achieve this goal:

$$PI(S) = \varphi\left(\frac{\mu(S) - \mathbf{f}(s^+) - \varepsilon}{\sigma(S)}\right), \tag{7.1}$$

where φ is CDF of the standard normal distribution, $\mathbf{f}(s^+)$ represents the corresponding maximum value among the existing training points $\mathscr{D}$ and ε helps PI to balance exploration and exploitation, which refers to balancing the exploration of unvisited regions of the space (where uncertainty is high) against the exploitation of regions known to have promising results (but maybe not the absolute best). This balance is crucial for efficiently finding global optima in complex domains. The predictive mean μ and variance σ of grid points S by the Traditional GP are:

$$\mu_{RBF}(S) = \Sigma_{\mathrm{r}\mathscr{D}}\left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_{\mathrm{n}}^2 I\right)^{-1} y, \tag{7.2}$$

$$\sigma_{RBF}(S) = \Sigma_{\mathrm{rr}} - \left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_{\mathrm{n}}^2 I\right)^{-1} \Sigma_{\mathscr{D}\mathrm{r}}, \tag{7.3}$$

where σ_{n}^2 represents the noise variance, $\Sigma_{\mathrm{r}\mathscr{D}}$ is the covariance matrix for training data points $\mathscr{D}$ and test data points S, as calculated according to Equation (3.2). y is the observed value corresponding to training data points $\mathscr{D}$.

According to the origin of PI, the next evaluation point calculated from Equation (7.1) is the point $s_{t+1} = \arg\max(PI(S))$. This new point will then be used to update the training points

set $\mathscr{D}$ and the end of this process will be controlled by the stopping criteria. Common stopping criteria include setting a maximum number of iterations, setting a total optimisation duration, or setting accuracy requirements. Considering practical applications and budget constraints, the stopping criteria adopted here is when the number of new training points generated by BO iterations equals 5% of the total number of grid points. The focus of this study is to compare the performance of different BO methods under the same budget constraints, rather than adjusting the constraints to optimise the performance of all BO methods proposed. After the optimisation stops upon reaching the maximum number of iterations, the optimal point will be selected from the final set of training points as the global optimisation result.

Močkus [109] provided a derivation of the global optimum for BO. After introducing the principles of Traditional BO, it is summarized in Algorithm 1. Initially, a set of training points $\mathscr{D} = s_1, s_2, ..., s_n$ from the grid points S is randomly selected. After calculating the predictive mean and predictive variance from Traditional GP, the PI acquisition function is used to search for the next 'best location' s_i. In every iteration, (s_i, y_i) has been added to $\mathscr{D}_{i-1}$ to update the posterior distribution of Traditional GP. When the maximum number of iterations is reached, the iteration process stops. Finally, the global optimum point identified by Traditional BO will be selected from the set $\mathscr{D}$.

Algorithm 1 Traditional Bayesian optimisation on manifolds

Selecting $\mathscr{D}_0 = s_1, s_2, ..., s_n$, as training points $\mathscr{D}$ from grid points set S
for $i = 1, \ldots, I$ { I is the number of iterations, equals to 5% of the number of S } **do**
  1.1 calculate the predictive mean and predictive variance based on $\mathscr{D}_{i-1}$ using Equation (7.2) and (7.4);
  1.2 select new $s_i = \arg\max(PI(S))$ by optimising PI function using Equation (7.1);
  1.3 query objective function to obtain y_i;
  1.4 augment data $\mathscr{D}_i = \{\mathscr{D}_{i-1}, (s_i, y_i)\}$.
end for
Selecting the global optimum point from the set $\mathscr{D}$.

The Traditional BO is based on the Traditional GP. Chapter 6 shows the problems when the Traditional GP is applied on manifolds. One of the major problems is that Traditional GPs neglect the boundary conditions. This results wrongly in distance between points being estimated across boundaries, instead of being constrained by the boundary. Additionally, the effectiveness of Traditional GP is limited by the location of training points, as discussed in Section 6.1. In particular, Traditional BO, with a very small amount of initial points becomes more sensitive to the distribution of these initial points, resulting in unstable results. This will be specifically ana-

lyzed in Section 7.6. Section 7.3 and 7.4 propose the use of Intrinsic BO to effectively addresses this issue.

7.3 Algorithm for Intrinsic Bayesian Optimisation with DIC & DTC

In Section 3.3, Intrinsic GPs on manifolds were introduced, which take into account the intrinsic geometric features of the manifold. Building on this foundation, Section 4.2 proposes SI-GPDIC, which improves the computational efficiency of Intrinsic GPs through utilizing m inducing points z. On one hand, BM paths only need to be simulated from the inducing points z rather than from all grid points S. On the other hand, it reduces the computational complexity of the inversion in the calculation of the predictive mean and variance from $O(n^3)$ to $O(nm^2)$, where $m < n$. This corresponds to the inversion of $(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_{\mathrm{n}}^2 I)^{-1}$ in Equation (3.9) and $(\sigma_{noise}^{-2}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}} + \Sigma_{\mathrm{zz}})^{-1}$ in Equation (4.6), respectively.

Chapter 6 applies this method to three examples. While it may not perform as well as other sparse methods, it remains an effective surrogate model in BO. This is because, in each BO iteration, inherent uncertainty plays a key role, and the point selected by the acquisition function PI directly influences subsequent iterations—especially when the number of training points is quite low in the early stages. This uncertainty is not solely determined by the performance of the predictive mean and predictive variance. Additionally, the results from this approach provide a useful benchmark for comparison with other methods. The predictive mean and predictive variance obtained from SI-GPDIC are expressed as:

$$\mu_{DIC}(S) = \sigma_n^{-2}\Sigma_{\mathrm{rz}}\mathrm{K}\Sigma_{\mathrm{z}\mathscr{D}}\mathbf{y}, \tag{7.4}$$

$$\sigma_{DIC}(S) = \Sigma_{\mathrm{rz}}\mathrm{K}\Sigma_{\mathrm{zr}}, \tag{7.5}$$

where σ_{n}^2 represents the noise variance, $\mathrm{K} = (\sigma_n^{-2}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}} + \Sigma_{\mathrm{zz}})^{-1}$ and Σ_{rz} is the covariance matrix for training data points $\mathscr{D}$ and inducing points z, calculated as Equation (3.8). With this foundation, Intrinsic BO with DIC uses the PI as the acquisition function, as shown in Equation (7.1). In each iteration, PI calculates the point most likely to yield an improved outcome based

on the regression results, adds this point to the set of training points $\mathscr{D}$, and initiates a new iteration. This process continues until the budget constraint is reached.

Intrinsic BO with DIC is summarized in Algorithm 2. The initial phase begins with selecting inducing points on the manifold. In this work, the inducing points are selected from the grid points which are equally spaced on the manifold. The BM paths are simulated using Equation (3.6) starting from the inducing point. The heat kernel is estimated as the BM transition density using Equation (3.7). The covariance matrices of the inducing points and all grid points can be constructed using the heat kernel estimates at different diffusion time. This concludes the initial phase. In the iterative phase, the training set $\mathscr{D}$ is initialised by randomly choosing initial locations from the grid points. The PI acquisition function uses the predictive mean and variance from Equation (7.4) and (7.5) to search for the next 'best location', s_{i+1}. In every iteration, s_{i+1} is added into training set $\mathscr{D}$ to update the posterior distribution of SI-GPDIC. The updating process is repeated until the maximum number of iterations is reached. Throughout the process, the BM paths only need to be simulated once in the initial phase.

Algorithm 2 Intrinsic Bayesian Optimisation with DIC on manifolds

Initial Phase

1.1 Select m inducing points from the grid points S on M;
1.2 Simulate BM paths starting with inducing points z:
for $i = 1,\ldots,m$ { m is the size of inducing points } **do**
  for $j = 1,\ldots,N$ {N is No. of paths } **do**
    Simulate BM sample paths starting at ith inducing point using Equation (3.6);
  end for
end for
1.3 Estimate the transition density of BM on M using Equation (3.7);
1.4 Construct Σ_{zz}, Σ_{zr} using the heat kernel as in Equation (3.8).

Iterative Phase

2.1 Initialise the Intrinsic BO by selecting the initial locations from the grid points S on M; $\mathscr{D}_0$ is initialised as $\mathscr{D}_0 = \{\mathbf{s}_0, \mathbf{y}_0\}$
for $i = 1,\ldots,I$ { I is the number of iterations } **do**
  1. Calculate predict mean and predict variance on grid points using Equation (7.4) and (7.5); $\Sigma_{z\mathscr{D}}$ can be constructed from Σ_{zr} by selecting the corresponding rows and columns;
  2. Find s_{i+1} by optimising PI function as in Equation (7.1);
  3. The training set is augmented $\mathscr{D}_i = \{\mathscr{D}_{i-1}, (s_i, y_i)\}$;
end for
Selecting the global optimum point from the set $\mathscr{D}$.

Due to the fact that the predicted variance in DIC approaches zero as test points move farther from the inducing inputs, leading to inaccurate uncertainty predictions at certain points, Section

4.3 introduces SI-GPDTC method. Similar to DIC in terms of predictive mean, DTC addresses the shortcomings of DIC by improving the reliability of predictive variances. The predictive mean and variance of SI-GPDTC can be expressed as:

$$\mu_{DTC}(S) = \sigma_n^{-2}\Sigma_{\mathrm{rz}}\mathrm{K}\Sigma_{\mathrm{z}\mathscr{D}}\mathbf{y}, \tag{7.6}$$

$$\sigma_{DTC}(S) = \max\left[\mathrm{diag}(\Sigma^*_{\mathrm{rr}}) - \mathrm{diag}(Q_{\mathrm{rr}}), 0\right] + \mathrm{diag}(\Sigma_{\mathrm{rz}}\mathrm{K}\Sigma_{\mathrm{zr}}), \tag{7.7}$$

where σ_{n}^2 represents the noise variance, $\mathrm{K} = (\sigma_n^{-2}\Sigma_{\mathrm{z}\mathscr{D}}\Sigma_{\mathscr{D}\mathrm{z}} + \Sigma_{\mathrm{zz}})^{-1}$ and $Q_{\mathrm{rr}} = \Sigma_{\mathrm{rz}}\Sigma_{\mathrm{zz}}^{-1}\Sigma_{\mathrm{zr}}$ is the covariance matrix approximated using the information from the inducing points z. Σ_{rz} is the covariance matrix for training data points $\mathscr{D}$ and inducing points z, calculated as Equation (3.8).

This improvement makes SI-GPDTC a robust solution with a more reliable predictive distribution. Intrinsic BO with DTC is proposed using SI-GPDTC as the surrogate model and PI as the acquisition function. It is expected that as the performance of the surrogate model improves, BO will correspondingly yield better results, although there exists some uncertainty. Algorithm 3 provides a summary of Intrinsic BO with DTC. This optimisation process is still divided into two parts. The Initial Phase does not involve iteration but provides the foundational information required for the subsequent Iterative Phase. In the Initial Phase, BM paths simulate from inducing points z and the necessary covariance matrices $\Sigma_{\mathrm{zz}}, \Sigma_{\mathrm{zr}}$ are precomputed and stored. In the Iterative Phase, the PI function plays a critical role by determining the next point to explore. This point is identified based on the predictive mean and predictive variance generated by the SI-GPDTC. Once the next exploration point is settled, it is incorporated into the training set, thereby updating the model with new data. This iteration process continues until the iteration budget is reached.

7.4 Algorithm for Intrinsic Bayesian Optimisation with VI

Section 4.4 introduces SI-GPVI from a new perspective, transforming the posterior inference problem into minimizing the distance between the exact model p and the modified model q, achieved via KL divergence, as shown in Equation (4.16). This approach incorporates the variational lower bound $F_V(\phi)$ into the true log marginal likelihood. The resulting predictive mean

Algorithm 3 Intrinsic Bayesian Optimisation with DTC on manifolds

Initial Phase

1.1 Select m inducing points from the grid points S on M;

1.2 Simulate BM paths starting with inducing points z:

for $i = 1, \ldots, m$ { m is the size of inducing points } **do**

  for $j = 1, \ldots, N$ {N is No. of paths } **do**

    Simulate BM sample paths starting at ith inducing point using Equation (3.6);

  end for

end for

1.3 Estimate the transition density of BM on M using Equation (3.7);

1.4 Construct Σ_{zz}, Σ_{zr} using the heat kernel as in Equation (3.8).

Iterative Phase

2.1 Initialise the Intrinsic BO by selecting the initial locations from the grid points on M. $\mathscr{D}_0$ is initialised as $\mathscr{D}_0 = \{\mathbf{s}_0, \mathbf{y}_0\}$

for $i = 1, \ldots, I$ { I is the number of iterations } **do**

  1. Calculate predict mean and predict variance on grid points using Equation (7.6) and (7.7); $\Sigma_{z\mathscr{D}}$ can be constructed from Σ_{zr} by selecting the corresponding rows and columns;

  2. Find s_{i+1} by optimising PI function as in Equation (7.1);

  3. The training set is augmented $\mathscr{D}_i = \{\mathscr{D}_{i-1}, (s_i, y_i)\}$;

end for

Selecting the global optimum point from the set $\mathscr{D}$.

and variance are provided in Equation (4.24). Chapter 6 demonstrates, through three examples, that SI-GPVI outperforms other sparse methods (SI-GPDIC and SI-GPDTC) as well as Traditional GPs and Graph GPs. Given SI-GPVI's strong performance, it is reasonable to expect that using it as the surrogate model in BO can lead to superior results, thanks to enhanced prediction accuracy in each iteration. Consequently, Intrinsic BO with VI has been established and summarized in Algorithm 4.

Intrinsic BO with VI is composed of two main components: SI-GPVI, which serves as the surrogate model for modeling the objective function, and the acquisition function PI, as shown in Equation (7.1). Before the iterative process begins, there are some preparatory steps for SI-GPVI. Specifically, BM paths need to be simulated first from the inducing points z. Following this, the covariance matrix for the inducing points themselves Σ_{zz}, as well as the covariance matrix between the inducing points and all grid points Σ_{zr}, can be precomputed and stored. This approach allows the predictive mean and variance to be retrieved from precomputed information Σ_{zz} and Σ_{zr} during each iteration when the surrogate model SI-GPVI is called, significantly improving computational efficiency by eliminating the need to recompute the covariance matrix $\Sigma_{z\mathscr{D}}$ at each iteration. The acquisition function PI uses the information returned by the surrogate model to determine the position of the next sampling point, identifying the location most worth

Algorithm 4 Intrinsic Bayesian Optimisation with VI on manifolds

Initial Phase

1.1 Select m inducing points from the grid points S on M;

1.2 Simulate BM paths starting with inducing points z:

for $i = 1, \ldots, m$ { m is the size of inducing points } **do**

  for $j = 1, \ldots, N$ {N is No. of paths } **do**

    Simulate BM sample paths starting at ith inducing point using Equation (3.6);

  end for

end for

1.3 Estimate the transition density of BM on M using Equation (3.7);

1.4 Construct Σ_{zz}, Σ_{zr} using the heat kernel as in Equation (3.8).

Iterative Phase

2.1 Initialise the Intrinsic BO by selecting the initial locations from the grid points on M. $\mathscr{D}_0$ is initialised as $\mathscr{D}_0 = \{\mathbf{s}_0, \mathbf{y}_0\}$

for $i = 1, \ldots, I$ { I is the number of iterations } **do**

  1. Calculate predict mean and predict variance on grid points using Equation (4.24); $\Sigma_{z\mathscr{D}}$ can be constructed from Σ_{zr} by selecting the corresponding rows and columns;

  2. Find s_{i+1} by optimising PI function as in Equation (7.1);

  3. The training set is augmented $\mathscr{D}_i = \{\mathscr{D}_{i-1}, (s_i, y_i)\}$;

end for

Selecting the global optimum point from the set $\mathscr{D}$.

exploring. In each iteration, the selected next sampling point is used to update the training set $\mathscr{D}$, initiating a new iteration of the optimisation process. The introduction of this method represents a powerful and efficient approach to optimising functions on manifolds. Section 7.6 will demonstrate its application in three examples and present a comparison of the results with other methods.

7.5 Algorithm of Graph Bayesian Optimisation

Chapter 5 introduces the Graph GP on manifolds, which differs from other methods by considering the manifold as an undirected graph structure, where the relationships between points are determined not by distance, as is typically the case, but by whether or not they are connected. On the undirected graph, inputs and outputs are indexed by vertices, with connections between vertices represented by assigned weights, which to some extent rely on Euclidean distance. The kernel used is an extension of the Matérn kernel from Euclidean space, known as the Graph Matérn kernel, which is derived from the SPDE formulation of the Matérn kernel. The

predictive mean and variance of Graph GPs can be calculated as:

$$8\mu_{Graph}(S) = \Sigma_{r\mathscr{D}}\left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_n^2 I\right)^{-1} y, \tag{7.8}$$

$$\sigma_{Graph}(S) = \Sigma_{rr} - \left(\Sigma_{\mathscr{D}\mathscr{D}} + \sigma_n^2 I\right)^{-1} \Sigma_{\mathscr{D}r}, \tag{7.9}$$

where σ_n^2 represents the noise variance, $\Sigma_{r\mathscr{D}}$ is the covariance matrix for training data points $\mathscr{D}$ and test data points S, calculated using the Graph Matérn kernel defined on the graph structure according to Equation (5.5).

Chapter 6 demonstrates the application of Graph GP across three examples, where the predictive mean shows improved performance compared to Traditional GP but falls short of the sparse intrinsic GP, especially SI-GPVI. However, in practice, the predictive variance it provides tends to be too small, leading to abnormally large values in the calculation of the PLL, which renders the comparison meaningless. Nevertheless, it is still useful to construct BO with Graph GPs as the surrogate model. This approach offers a different perspective on optimisation over manifolds. By leveraging the inherent graph structure to model relationships between points on the manifold, Graph GP introduces an alternative way to explore the search space. Although the predictive variance may present some problems, the overall framework can still provide valuable information for the optimisation process. This also represents a further exploration of the application of Graph GPs on manifolds.

Graph BO is summarized in Algorithm 5. The PI, as shown in Equation (7.1), is used to determine the next optimal location. The initial phase begins with the construction of the graph structure, which serves as the preparatory step for a Graph GP. Before the iteration begins, it is first necessary to compute the $n \times n$ affinity matrix W and derive the random walk normalized Laplacian Δ_{rw}. The eigenvalues λ and eigenvectors f of the Laplacian matrix Δ_{rw} are crucial components in defining the GP on the graph structure. In each iteration, Graph GP calculates the predictive mean and variance using the current training points $\mathscr{D}$ first. Based on this foundation, the PI function identifies the next point to explore, which is then used to update the training points $\mathscr{D}$. This updating process continues until the maximum number of iterations is reached. In the next section, Graph BO will be implemented in three examples and compared to the results from the other BO methods.

Algorithm 5 Graph Bayesian Optimisation on manifolds

Initial Phase

1.1 Calculate $n \times n$ affinity matrix $\tilde{\mathbf{A}}$ and derive the random walk normalized Laplacian Δ_{rw}, using the approach proposed by [41], as in Equation (5.2);
1.2 Calculate eigenvalues λ, eigenvectors f of Δ_{rw};
1.3 Build the Graph Matérn kernel from eigenvalues λ and eigenvectors f using Equation (5.5);
1.4 Construct the Graph GP model.

Iterative Phase

2.1 Initialise the Graph BO by selecting the initial locations from the grid points S on M. $\mathscr{D}_0$ is initialised as $\mathscr{D}_0 = \{\mathbf{s}_0, \mathbf{y}_0\}$
2.2 Update the posterior distribution by finding s_{i+1} to update training set $\mathscr{D}$:
for $i = 1, \ldots, I$ { I is the number of iterations } **do**
1. optimise the hyperparameters of Graph GPs using Equation (5.6);
2. Calculate predictive mean and variance on grid points S from the Graph GPs as in Equation (7.8) and (7.9);
3. Find s_{i+1} by optimising PI function as in Equation (7.1);
4. The training set is augmented $\mathscr{D}_i = \{\mathscr{D}_{i-1}, (s_i, y_i)\}$;

end for
Selecting the global optimum point from the set $\mathscr{D}$.

7.6 The Application and Comparison of All Proposed BO Methods

Given the successful application of BO in Euclidean spaces, this chapter aims to extend its application to manifolds. In the previous sections, five distinct BO approaches on manifolds are introduced. The main difference between these methods lies in the different surrogate models. Each of these methods provides a unique approach to optimising functions on manifolds. In the application of BO, particularly on real-world datasets, it is often important to consider not only the ability to identify the optimum value, but also the ability to identify its location. Since the optimum value appears at a specific location in the domain, it is also meaningful to examine how close the point found by each method is to the true optimal location, both in terms of distance and surrounding area. However, as discussed in Chapter 1 and Chapter 2, unlike in Euclidean space, there is no common and suitable notion of distance on a manifold that fully reflects intrinsic geometric properties and complex boundaries. Consequently, this section not only evaluates the optimum value found by each method, but also analyzes whether the identified point lies near the true optimal location and whether the corresponding region of the true optimum has been effectively explored.

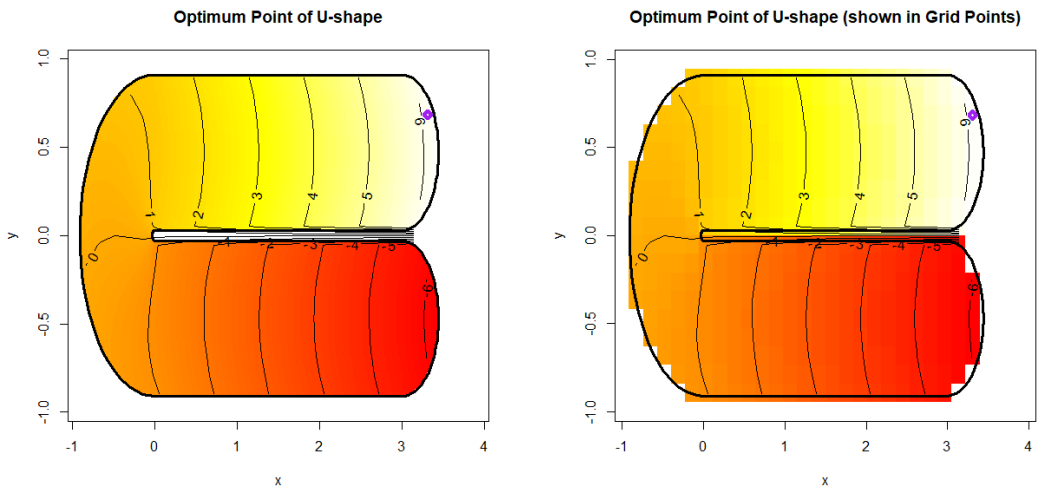
The main focus of BO remains on optimising the function value, so comparisons are primarily based on the gap between the obtained and true optimal value. This section will use the U-shape domain, the Bittern-torus, and the real-world dataset of the Aral Sea to demonstrate and compare the performance of different BO methods.

7.6.1 The Implementation of Proposed BO on the U-shape

The U-shape, previously introduced in Section 1.3.1 as an example of a 2-dimensional manifold, will be used to implement the proposed BO methods. Figure 7.2 shows the original form of the U-shape as well as the U-shape constructed from 418 grid points. By applying BO to observations chosen from evenly distributed grid points on the manifold (rather than from a continuous surface), resource wastage could be effectively minimized. Considering the budget constraints, all BO methods are expected to find the optimal point on the U-shape using training points that make up only 5% of the total grid points. With 418 grid points and an initial set of 3 training points, the comparison focuses on how closely each BO method's optimal values align with the true optimal value after only 18 iterations, utilizing a total of 21 points (5% of the 418 grid points) to explore the U-shape. The optimal point to be identified in this example corresponds to the maximum value point. The true optimal value on the U-shape is 6.1188, as indicated by the purple dot in Figure 7.2.

From the 418 grid points of the U-shape, 20 training sets are randomly selected, each containing 3 points as the initial locations for the optimisation. Conducting 20 experiments ensures robustness and reliability by averaging the outcomes across different initial training points, reducing the influence of randomness and leading to more generalizable conclusions. One of the training sets is selected for demonstration, as shown in Figure 7.3. The figure illustrates the predictive mean in the final step of the BO process for each method using this set, allowing for the observation of the distinct characteristics of the surrogate models GPs, associated with each BO method. The blue points denote the three initial points, while the black crosses mark the exploration points identified by the PI function during the BO iterations. The purple point indicates the optimal point found by the current BO method.

In Figure 7.3, all three initial points are located in the lower part of the U-shape. In panel (a),



(a) (b)

Figure 7.2: Original U-shape (a) and U-shape shown in grid points (b), with optimal point indicating as the purple dot near the boundary.

the optimal point found by Traditional BO is located on the far left side of the U-shape, which is notably distant from the true optimum located in the upper-right region. This misidentification reflects not only a suboptimal function value but also a failure in locating the correct region where the global optimum resides. The reason for this failure is that the surrogate model used by Traditional BO, the Traditional GP, fails to recognize the boundaries of the U-shape. During the iterations, the predictive mean and variance generated for the upper-right region are influenced by the lower-right region, leading the model to mistakenly assume that the upper-right region also has relatively low values. Additionally, because the upper-right region is closer to the initial points in the lower-right region in Euclidean space, the model predicts a smaller variance for this area. As a result, the acquisition function PI evaluates the upper-right region as having low exploration value, leading to no exploration in that area. This decision not only causes the algorithm to miss the global optimum in terms of value but also reveals its inability to infer or explore the correct spatial location of the optimum. In panel (b), Graph BO also fails to find the optimal point, and the reason is similar to that of Traditional BO. The surrogate model, Graph GP, is unable to recognize the intrinsic geometric features of the U-shape, leading it to assume that the upper-right and lower-right regions have similar values. Its inability to capture the intrinsic geometric features of the U-shape leads to a misinterpretation of where high-value regions may lie. This poor judgment prevents the method from identifying the true optimal

point located at the boundary of the upper-right region. Panels (c), (d), and (e) correspond to the results of Intrinsic BO with VI, Intrinsic BO with DIC, and Intrinsic BO with DTC, respectively. The core of their surrogate model is the Intrinsic GP, which is proposed in Chapter 3. By using the transition density of BM paths to approximate the heat kernel on the manifold, Intrinsic GP adopts the intrinsic geometric information of the manifold to create a predictive model that better suits the manifold's structure. The differences among these BO methods lie in the various sparse methods they employ, specifically SI-GPDIC, SI-GPDTC, and SI-GPVI, as introduced in Chapter 4. In Panel (c), Intrinsic BO with VI is the only one among the five methods that successfully finds the optimal point. The surrogate model, SI-GPVI, as with the excellent performance in Chapter 6, provides more accurate predictions in each iteration of BO, leading to it finding the optimal point. By contrast, although the other two Intrinsic BO methods in Panels (d) and (e) do not find the optimal point, they benefit from the utilization of the manifold geometric features to find the upper-right region, the correct location of the optimum. This indicates partial success in identifying the optimal region, even if the exact value is not achieved. This represents a significant improvement over Traditional and Graph BO, which failed to explore this area at all. Their predictions for the upper-right region are not influenced by the lower regions.

A detailed comparison of the various methods will be made through numerical analysis. Table 7.1 summarizes the optimal values found by each BO method across 20 experiments. The optimal points found by Traditional BO have a wide range (5.6335) and a lowest mean and median, suggesting instability and poor performance on the manifold. Graph BO performs better than Traditional BO, but there are still cases where the upper right region cannot be explored. Intrinsic BO with VI outperforms other methods with the highest median (5.924) and the smallest range (0.81), indicating both accuracy and reliability. Given the sample size ($n = 20$) and the violation of the normality assumption, the paired t-test is not suitable for evaluating statistical significance. Consequently, the Wilcoxon Signed-Rank Test is employed. It demonstrates statistically significant superior performance compared to both Traditional BO and Intrinsic BO with DIC, as confirmed by the Wilcoxon Signed-Rank Test (p = 0.002995 and 0.002978, respectively). Due to the presence of tied and zero differences, the p-value is computed using a normal approximation rather than the exact method. Intrinsic BO with DTC also performs well but is slightly inferior to the VI method in all aspects. The optimal points found by Intrinsic BO with

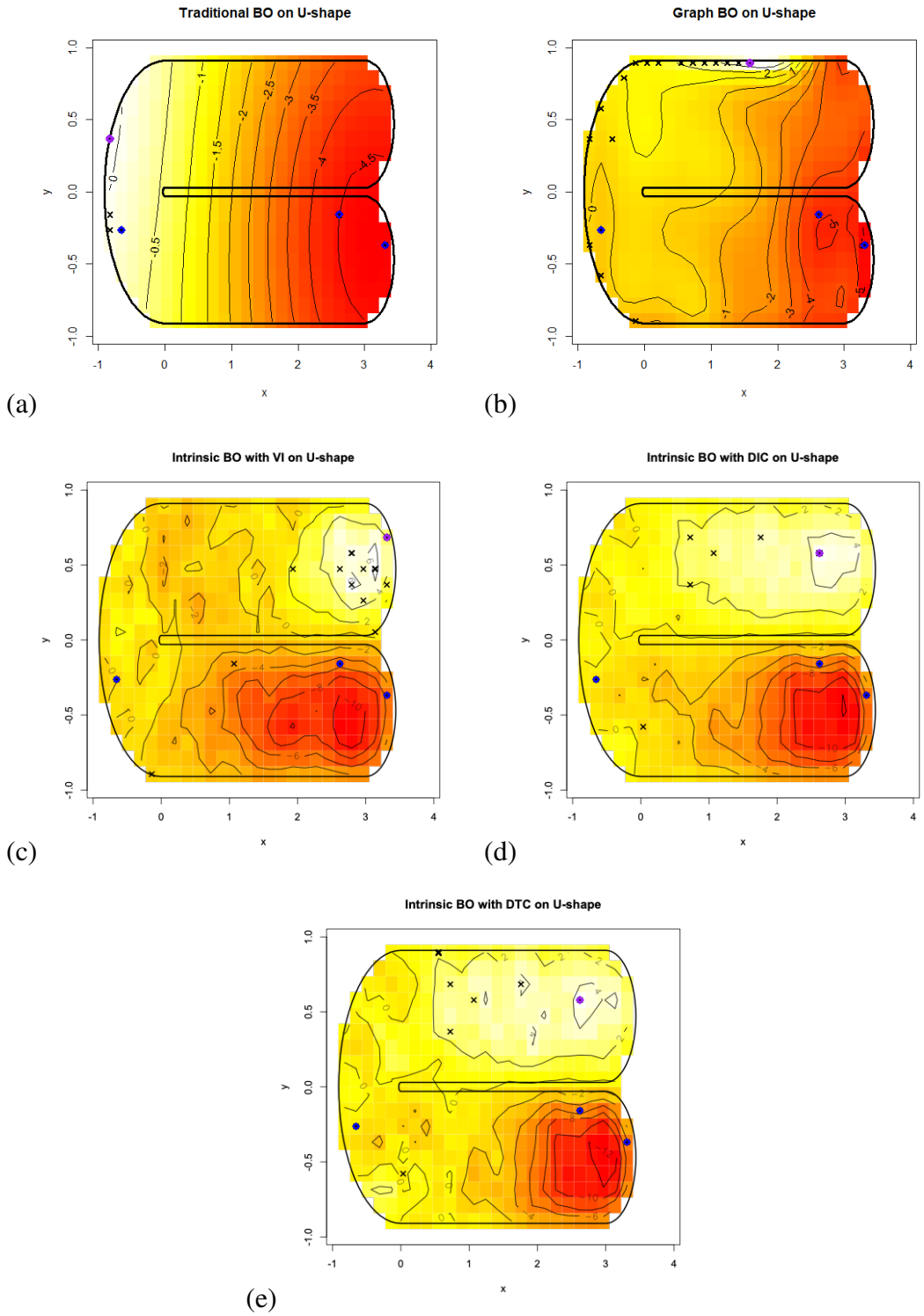


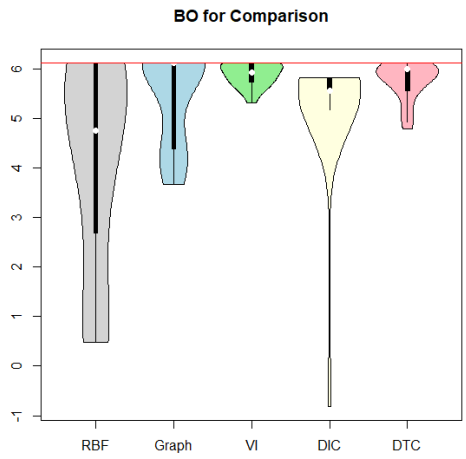
Figure 7.3: Different BO methods applied on the U-shape with same initial points and number of iterations. The blue dots represent the initial points, the black crosses indicate the points explored during the BO process, and the purple dot marks the optimal point found by each method. (a) the implementation of Traditional BO on the U-shape domain; (b) the implementation of Graph BO on the U-shape domain; (c) the implementation of Intrinsic BO with VI on the U-shape domain; (d) the implementation of Intrinsic BO with DIC on the U-shape domain; (e) the implementation of Intrinsic BO with DTC on the U-shape domain.

DIC display a wide range and relatively low mean, indicating poor performance.

Summary (Optimum: 6.1188)	Min.	Median	Mean	Max.	Range
Tra	0.4853	4.7606	4.1568	6.1188	5.6335
Graph	3.667	6.1188	5.445	6.1188	2.452
VI	5.309	5.924	5.882	6.1188	0.81
DIC	-0.8101	5.5676	5.2481	5.8262	6.6363
DTC	4.792	5.989	5.758	6.116	1.324

Table 7.1: Statistical summary of optimal point found by each BO method on the U-shape.

Figure 7.4 visualizes the statistical analysis results using a violin plot. The red horizontal line indicates the true global value. Among all methods, only Intrinsic BO with DIC failed to find the optimal point across all 20 sets. This is attributed to the predictive variance issues in its surrogate model SI-GPDIC. Since the optimal point lies near the boundary, far from the inducing points, the variance representing uncertainty incorrectly approaches zero, which leads to a lack of exploration in this region. This issue is discussed in detail in Section 4.2. From Figure 7.4, it is evident that Intrinsic BO with VI consistently outperforms the other methods in finding the optimal point (the maximum value) on the U-shape domain, making it a robust choice for manifold-based optimisation tasks.



(a)

Figure 7.4: Violin plot of optimal points found by different BO methods on the U-shape domain; the red horizontal line is the true global value; the dots at each end of the bold black lines represent the first and third quartiles; the white dot represents the median.

7.6.2 The Implementation of Proposed BO on the Bitten-torus

The Bitten-torus is used as a representative example of a three-dimensional manifold as introduced in Section 1.3.2. The proposed BO methods aim to select the optimal point from the 600 grid points evenly distributed on the Bitten-torus, which adequately cover the surface of the Bitten-torus. In this example, BO attempts to find the minimum value point on the Bitten Torus, with the optimal point value being 0.1598, as shown as pink square in Figure 7.5(a). While the previous U-shape example focuses on maximizing the objective function, this example helps demonstrate the flexibility of BO in addressing both minimization and maximization problems. From the 600 grid points on the Bitten Torus, 20 training sets are randomly selected, each consisting of three points. In Figure 7.5(a), the purple dots represent one of these sets, showing the initial points selected from the Bitten Torus. Given the 5% budget constraints, with 600 grid points and 3 initial points, all BO methods are expected to explore using a total of 30 points (i.e., 5% of the grid points), meaning the optimal point needs to be found within 27 iterations. The comparison will focus on how the optimal values found by each BO method align with the true optimal value.

Figure 7.5 compares the performance of each BO method using the same set of initial points, shown as purple dots. In panel (b), Traditional BO fails to find the optimal value, and the optimal point it finds is far from the true optimal point. This result shows not only suboptimal performance in value but also a lack of ability to locate the optimal region on the manifold. This issue arises because the surrogate model, Traditional GP, cannot recognize the manifold's intrinsic geometric features and boundaries. When the optimal point (with the minimum value), is located at the boundary, the predictions made by Traditional GP in this region are influenced by the initial points in the red area which have higher values. As a result, it fails to recognize the boundary as a potentially optimal location, limiting both value discovery and spatial localization. It incorrectly estimates this region as having higher values and lower uncertainty, leading to the neglect of exploration in this area and producing an overstated predictive mean. This neglect prevents the model from identifying not just the value but also the spatial location of the optimum. This causes Traditional BO to incorrectly judge the optimal point to be located in the central region of the Bitten Torus, leading to failure in identifying the true optimal point. In panel (c), Graph BO has similar issues as Traditional BO and fails to find the optimal point.

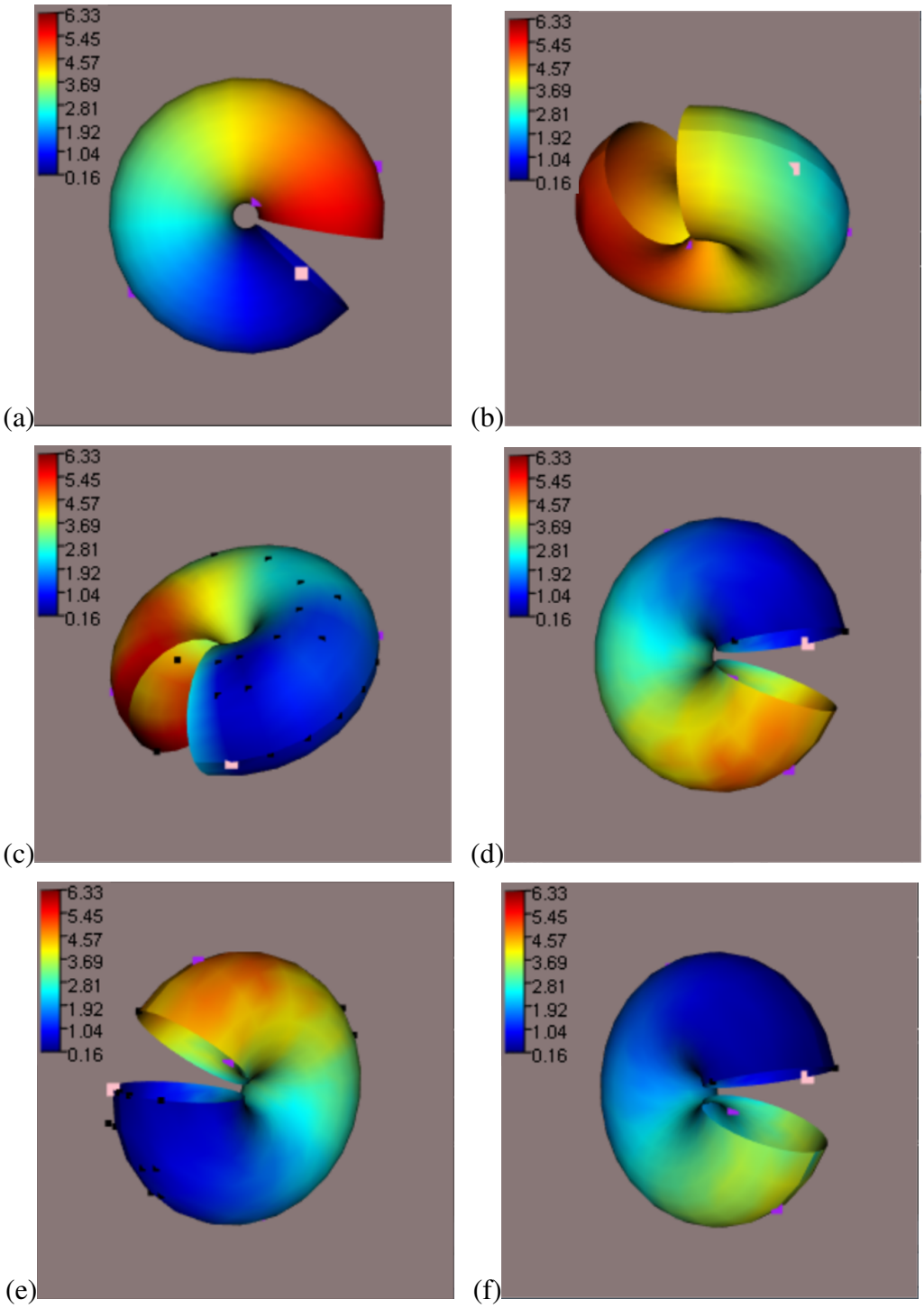


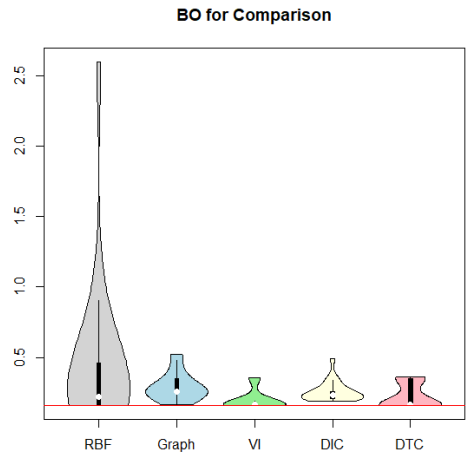
Figure 7.5: Comparison of different BO methods applied to the Bitten-torus using the same number of iterations and initial points, represented by the purple dots. The black dots indicate the exploration points during the BO process, and in (a), the pink square marks the true global optimal point (minimum), which all BO processes aim to find. In the remaining figures, the pink square shows the optimal point found by each method: (a) the original Bitten-torus with the true optimal point; (b) Traditional BO; (c) Graph BO; (d) Intrinsic BO with VI; (e) Intrinsic BO with DIC; (f) Intrinsic BO with DTC.

However, in this set, the impact on Graph GP is less severe because there is still some distance from the initial point in the red region to the blue region boundary. However, the method still does not demonstrate an effective ability to localize the boundary region containing the true optimum. Graph GP transforms the manifold into a graph structure, where the influence between points diminishes as their connections weaken. In contrast, when using Sparse Intrinsic GPs as surrogate models, which account for the manifold's intrinsic structure, the BO methods in panels (d), (e), and (f) deliver strong results. Specifically, Intrinsic BO with VI (panel d) and Intrinsic BO with DTC (panel f) both successfully locate the optimal value at the boundary. While Intrinsic BO with DIC (panel e) does not find the exact optimal point, it reaches a point close to the optimal value. Its exploration pattern covers the boundary area well, suggesting that it captures the right region even if the precise point is missed. The use of Intrinsic GPs, which capture the geometric properties of the manifold, indicates not just convergence in function value, but also a strong ability to guide exploration toward the correct region where the optimum lies. The figures clearly illustrate that for Intrinsic BO, the two sides of the "bitten" region remain independent, which closely resembles the true structure of the Bitten-torus.

Table 7.2 shows a statistical summary of the optimal points found by different BO methods on the Bitten Torus, and Figure 7.6 uses the violin plot to visualize these results, where the red horizontal line shows the true global optimal value. It is obvious that Traditional BO has the widest range (2.4377). The mean optimal value (0.4551) found by Traditional BO is significantly higher than other BO methods, indicating ineffectiveness in finding the optimal point. Graph BO also displays a high mean value (0.2911) and a relatively wide range. The three Intrinsic BO methods, VI, DIC, and DTC, all outperform Traditional BO and Graph BO. Among them, Intrinsic BO with VI shows the best performance, which has the lowest mean optimal value (0.1972) and the smallest range (0.1965). It shows significantly superior performance compared to Traditional BO, Graph BO, and Intrinsic BO with DIC, as confirmed by the Wilcoxon Signed-Rank Tests, with p-values of 0.004954, 0.006687, and 0.02342, respectively. Intrinsic BO with DTC also shows strong results, though slightly less consistent than VI. These results show that incorporating intrinsic geometric characteristics into the BO method enhances its effectiveness on manifolds, with Intrinsic BO with VI, driven by SI-GPVI, proving to be the most reliable and effective approach.

Summary (Optimum: 0.1598)	Min.	Median	Mean	Max.	Range
Tra	0.1598	0.2218	0.4551	2.5975	2.4377
Graph	0.1632	0.2598	0.2911	0.5195	0.3563
VI	0.1598	0.1632	0.1972	0.3563	0.1965
DIC	0.1890	0.2339	0.2528	0.4936	0.3046
DTC	0.1598	0.1632	0.2248	0.3598	0.2

Table 7.2: Statistical summary of optimal point found by each BO method on the Bitten-torus.



(a)

Figure 7.6: Violin plot of optimal points found by different BO methods on the Bitten-torus; the red horizontal line is the true global optimal value; the dots at each end of the bold black lines represent the first and third quartiles; the white dot represents the median.

7.6.3 The Implementation of Proposed BO on the Aral Sea

The previous section has demonstrated and compared the application of five proposed BO methods on both two-dimensional and three-dimensional manifolds. As analyzed in Section 7.4, Intrinsic BO with VI performs more effectively on manifolds compared to other methods, which is attributed to the superior performance of its surrogate model, SI-GPVI. This section will utilize the real-world dataset of the Aral Sea as introduced in Section 1.3.3 to further evaluate and compare the proposed BO methods. It considers remotely sensed chlorophyll data in the Aral Sea to investigate sites with the highest chlorophyll concentration. Chlorophyll level serves as an indicator of water quality, making this study valuable for monitoring and predicting environmental pollution. Additionally, it contributes to the protection of the fragile ecosystem of the Aral Sea. With 485 grid points scattered within the complex boundary [166], the location

with highest level of chlorophyll concentration (19.278724) is shown as a purple dot in Figure 7.7(a), which also being the optimal point that the BO methods aim to explore. When conducting inference and prediction tasks, it's essential to consider the intrinsic geometry of the sea and its complex boundaries. The Euclidean distance between two points, which represents the 'straight-line' distance, may not accurately reflect the true separation when accounting for land barriers or other obstructions. Locations separated by such boundaries often have distinct chlorophyll levels, which the surrogate models need to capture to ensure accurate predictions and exploration outcomes. In such cases, regions that appear close in Euclidean space may in fact be disconnected or difficult to reach on the manifold, making it harder for models relying solely on Euclidean proximity to correctly identify the location of the optimum.

When the budget is set to 5%, with 3 initial points, the BO methods are expected to explore using 24 points (5% of 485 grid points), i.e., find the optimal point within 21 iterations. The 20 training sets are randomly selected from the 485 grid points of the Aral Sea, with each set containing 3 initial points. Figure 7.7 illustrates the results of the five proposed BO methods using one of the selected training sets, with three initial points located in the left region. In panel (b), influenced by the initial point positioned in the lower part of the west side, the surrogate model in Traditional BO (i.e., Traditional GP) fails to account for the land barrier in the middle. As a result, it assumes that nearby Euclidean locations are smoothly connected, which misleads the model into focusing exploration away from the disconnected region containing the true optimum — thus impairing both the value estimation and the ability to identify the correct location. Traditional BO incorrectly assumes that the region near the left boundary on the east side has similarly low values, leading to insufficient exploration in the lower-left area of the east side of the Aral Sea. Consequently, Traditional BO's final chosen optimum lies in a more northern region of the Aral Sea rather than in the true optimal area (the lower-eastern basin). Similarly, in panel (c), Graph BO also overlooks the lower-left area of the east side of the sea, leading to lower predicted values in that region and causing the optimal point found to be higher than the true optimal point for the Aral Sea. However, in this dataset, certain real-world noise or sampling deviations appeared to unintentionally benefit Traditional BO and Graph BO, helping them explore the region near the true optimum that would otherwise have been overlooked. In panels (d), (e), and (f), it is evident that the Intrinsic BO methods using Sparse Intrinsic GP as the surrogate model successfully avoid the issues associated with Traditional BO and Graph

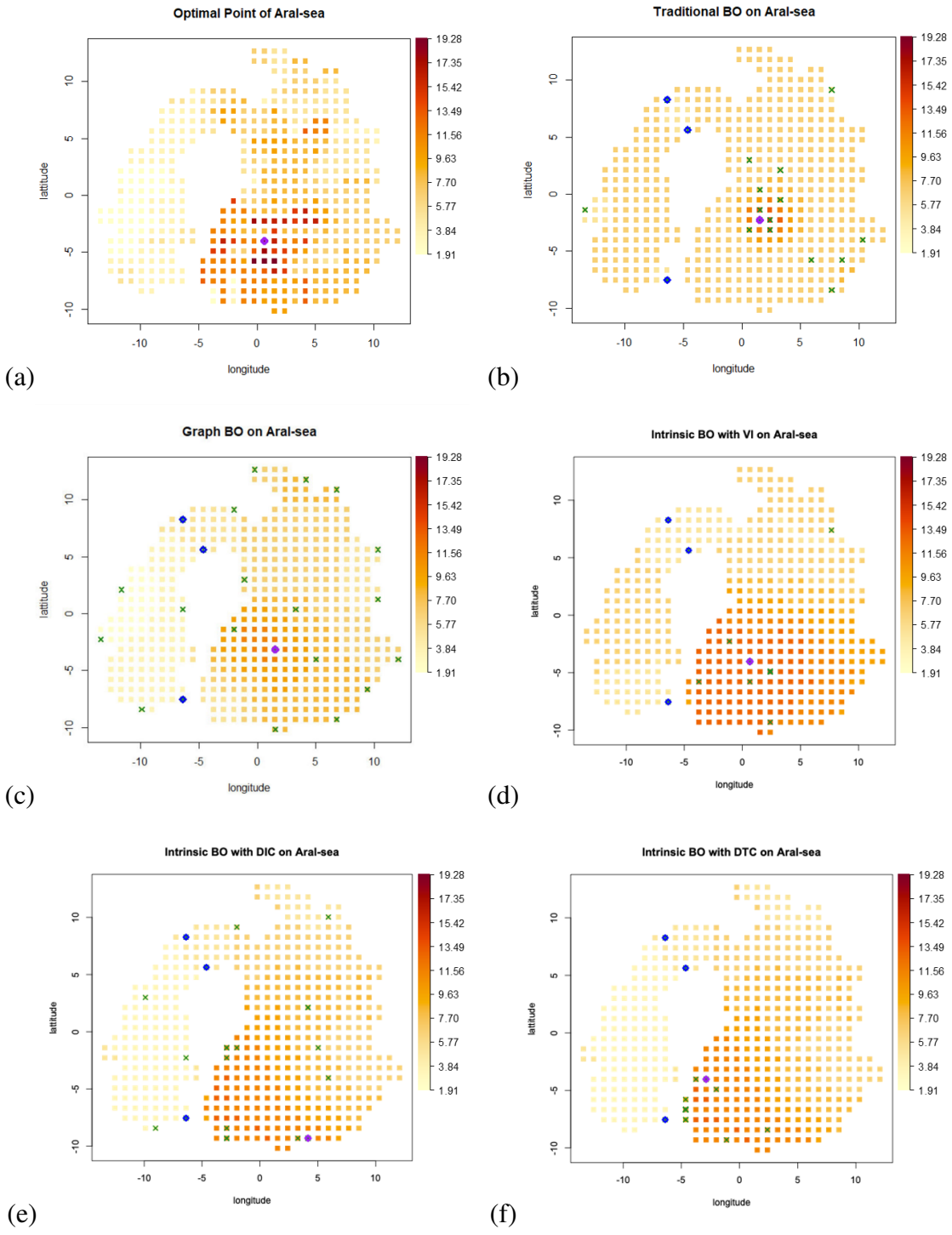


Figure 7.7: Different BO methods applied on the Aral Sea with same number of iterations and initial points, represented by the blue dots. The green crosses indicate the points explored during the BO process, and in (a), and the purple dot marks the true optimal point (maximum), which all BO processes aim to find. In the remaining figures, the purple dot marks the optimal point found by each method: (a) the true chlorophyll level of the Aral Sea; (b) the implementation of Traditional BO; (c) the implementation of Graph BO; (d) the implementation of Intrinsic BO with VI; (e) the implementation of Intrinsic BO with DIC; (f) the implementation of Intrinsic BO with DTC.

BO. Sparse Intrinsic GP utilize the transition density of BM paths to estimate the heat kernel on the manifold, incorporating the manifold's intrinsic geometric structure into the predictive model. This allows for more accurate predictions on manifolds. In particular, panel (d) shows that Intrinsic BO with VI is the only method among the five methods that successfully identifies the true optimal point in this case, demonstrating not only convergence in function value, but also a strong ability to guide exploration toward the correct region where the optimum lies.

Figure 7.8 provides a clearer illustration of the impact of the boundary on each method by showing the predictive mean of the surrogate models after only one iteration, using the same initial points as those in Figure 7.7. Since Intrinsic BO with DIC, DTC, and VI all rely on the principle of simulating BM paths, it is sufficient to present the results of Intrinsic BO with VI. Figure 7.8 (a) and (b) show the one-step iteration results for Traditional BO and Graph BO, respectively. It is more evident here that the surrogate models of these two methods smooth across the isthmus of the central peninsula, providing similar predictions on both sides of the isthmus due to the close spatial vicinity of points in Euclidean space. In contrast, Figure 7.8 (c) shows that Intrinsic BO with VI clearly provides different predictive results on either side of the isthmus. The presence of the isthmus causes very large actual distance between the two sides of the sea, which is accurately reflected in the model's predictions.

Table 7.3 provides a statistical summary of the optimal points found by all BO methods after 21 iterations on the Aral Sea. The violin plot in Figure 7.9 visually compares the distribution of these optimal points found by different BO methods. The red horizontal line is the true global optimal value 19.278724. Graph BO is the worst-performing method among all, with the most significant range (9.943) and the lowest mean (13.682), reflecting great prediction instability and inaccuracy. Traditional BO fails to successfully identify the optimal point in all 20 training sets when applied to the Aral Sea. Although it shows relatively good predictive results compared to other methods, it still falls short compared to the Intrinsic BO with VI. As in the case of the U-shape and Bitten-torus applications, Intrinsic BO with VI performs better than other methods in all aspects, offering more stable and accurate performance. It significantly exceeds the performance of both Graph BO and Intrinsic BO with DIC, as validated by the Wilcoxon Signed-Rank Tests, with p-values of 0.0007208 and 0.005346, respectively. Given that the common significance threshold for the p-value is 0.05, these results are highly significant.

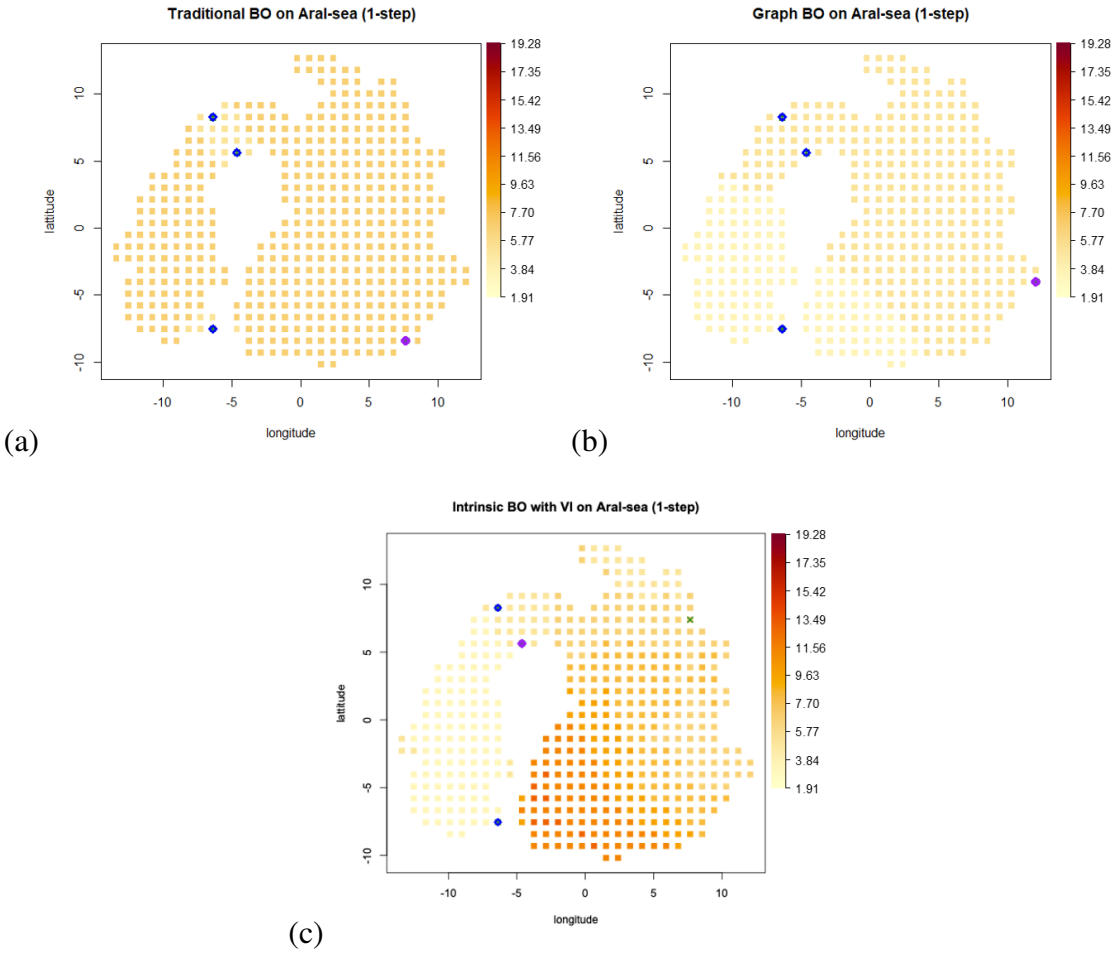
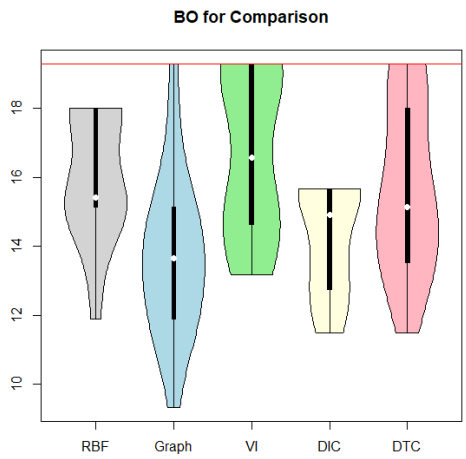


Figure 7.8: Different BO methods with only one iteration, the blue dots represent the initial points, and the purple dot is the optimal point found by each BO method, the green cross represents the optimal point found after one iteration: (a) the implementation of Traditional BO; (b) the implementation of Graph BO; (c) the implementation of Intrinsic BO with VI.

Summary (Optimum: 19.279)	Min.	Median	Mean	Max.	Range
Tra	11.89	15.41	16.05	17.99	6.1
Graph	9.336	13.649	13.682	19.279	9.943
VI	13.19	16.57	16.71	19.279	6.089
DIC	11.49	14.90	14.22	15.67	4.18
DTC	11.49	15.14	15.69	19.279	7.789

Table 7.3: Statistical summary of optimal point found by each BO method with 21 iterations on the Aral Sea.



(a)

Figure 7.9: Violin plot of optimal points found by different BO methods on the Aral Sea; the red horizontal line is the true global optimal value; the dots at each end of the bold black lines represent the first and third quartiles; the white dot represents the median.

7.6.4 Impact of the Number of Inducing Points

Due to the abrupt changes, noise, and irregularities inherent in real-world data, the Aral Sea dataset lacks smoothness, resulting in instability in the performance of BO methods based on sparse intrinsic GPs. For example, in the case of Intrinsic BO with Variational Inference, the range of optimal values identified is 0.81 for the U-shape and 0.1965 for the Bittern-torus, whereas in the Aral Sea case, it increases significantly to 6.089, indicating substantial variability. As increasing the number of inducing points generally leads to a more accurate approximation and enhances the model's ability to fit complex functional structures, it is expected that varying the number of inducing points may improve the BO algorithm's ability to identify the true optimum. This consideration motivates a sensitivity analysis on the number of inducing points, with the aim of improving the performance of intrinsic BO methods in challenging, non-smooth data settings such as the Aral Sea case.

In Section 7.6.3, the Aral Sea experiments use 10 inducing points, with their locations shown in Figure 6.8(a). To investigate the impact of the number of inducing points, additional experiments are conducted using 5, 15, 20, and 42 inducing points, while keeping other experimental settings fixed. All configurations are evaluated using the same 20 sets of initial training points to

ensure a fair comparison. The distributions of the inducing points for each setting are provided in Appendix D.

The results show that increasing the number of inducing points does not consistently lead to significant improvements in performance on the Aral Sea dataset. When the number of inducing points increases from 10 to 15 and 20, no clear performance gain is observed, while an improvement appears at 42 inducing points. However, this improvement results in a 4.2-fold increase in computational time, and the dense distribution of inducing points at this level undermines the fundamental advantage of using the sparse method. Although slight variations in outcomes are observed across different settings with 5, 10, 15, 20, and 42 inducing points, these differences are not statistically significant. The Wilcoxon Signed-Rank Test is conducted on paired results obtained from the same 20 initial point sets across different inducing point settings. The test results show that only the comparison between 5 and 10 inducing points is statistically significant (where the p-value is less than 0.0001). All other comparisons with 10 inducing points have p-values greater than 0.05. This suggests that increasing the number of inducing points beyond 10 does not lead to statistically significant improvements in optimisation performance. The observed fluctuations are likely due to noise and randomness in the dataset rather than systematic gains from using more inducing points. In light of this, increasing the number of inducing points does not necessarily yield better performance. A moderate number of inducing points is sufficient to explore the manifold effectively without bringing unnecessary computational burden. Future work may consider alternative strategies to improve the performance of BO methods on real-world cases.

7.7 Conclusion of All BO Methods Proposed and Applied

In Chapter 6, the U-shape is used as the representative example of two-dimensional manifolds to compare the performance of five GP methods: Traditional GPs, Graph GPs, SI-GPDIC, SI-GPDTC, and SI-GPVI. These methods have already been discussed in previous chapters. The comparison also extends to a three-dimensional manifold, the Bitten-torus, and a real-world dataset, the Aral Sea. Three Sparse Intrinsic GPs stand out because their kernel functions account for the intrinsic geometric structure of the manifold, leading to more accurate predictions

when compared to Traditional GPs and Graph GPs, according to the root mean square error (RMSE) indicator. Additionally, SI-GPVI and SI-GPDTC methods generally provide better predictive distributions, as evaluated by the predictive log likelihood (PLL) indicator, which utilizes both predictive mean and variance. Among these methods, SI-GPVI consistently outperforms the others across all three examples. Encouraged by the successful of these methods in regression tasks on manifolds, this chapter explores BO on manifolds. BO is an advanced optimisation technique in Euclidean space, and this chapter extends its application to address optimisation challenges on manifolds.

Section 7.1 provides the background of BO and discusses its significance for optimisation problems on manifolds. The core task is defined as finding the optimal point from a finite set of grid points on the manifold while operating under a budget constraint. In this case, the budget constraint is expressed as a limited number of iteration steps, set at 5% of $G^{'}$, the total number of grid points S. BO consists of two main parts: the surrogate model and the acquisition function. The five GP methods mentioned earlier can serve as the surrogate models to predict the objective function. PI function acts as the acquisition function, selecting the next point to explore in the following iteration. This newly selected point updates the training set, starting the next cycle of iterations. The process repeats until the predefined number of iterations is completed, ensuring an efficient and targeted search for the optimal point on the manifold.

Section 7.2-7.5 propose five different BO algorithms based on different surrogate models mentioned above, namely Traditional BO, Graph BO, Intrinsic BO with DIC, Intrinsic BO with DTC, and Intrinsic BO with VI. Obtaining knowledge of the geometry and intricate boundaries of the manifold through the heat kernel allows BO to more effectively navigate the manifold and enhance the precision of resolving the optimisation problem, when using SI-GPDIC, SI-GPDTC, and SI-GPVI as surrogate models. However, for Traditional BO and Graph BO, the surrogate model is related to Euclidean distance, which provides less accurate predictions for the iterative process. Especially in the initial stages of iteration, due to the small number of training points offering limited information, inaccurate GP predictions may lead the PI to explore incorrect regions. Although these errors may be corrected to some extent after multiple iterations, they can still affect the results of BO, especially when existing a budget constraint.

Section 7.6 applies all proposed BO methods to the three examples used in Chapter 6, provid-

ing a comprehensive evaluation of their performance. The performance of these BO methods in the three examples is generally consistent with the performance of their corresponding surrogate models. The shortcomings of the surrogate model on manifolds continue to emerge throughout the iterative process, though these problems may be relieved partially. For instance, for BO, its surrogate model, Traditional GP, lacks consideration of the manifold's intrinsic features, depending on Euclidean distances. This limitation in capturing the true geometric properties of the manifolds can seriously affect the whole optimisation process, especially when the optimal point is near the boundary. Both Traditional BO and Graph BO are highly sensitive to initialisation and often get trapped in central regions of the U-shape and Bitten-torus. This occurs due to inaccurate estimations near the boundary, influenced by the Euclidean distances between the two ends. Correspondingly, BO methods also benefit from the strengths of their surrogate models. For example, Intrinsic GPs consider the geometric features of the manifolds, enhancing the accuracy and effectiveness of the optimisation process. Intrinsic BO with VI performs more effectively on manifolds compared to other methods, owing to the exceptional performance of its surrogate model, SI-GPVI. When the implicit objective function is non-differentiable, multi-peaked and non-convex, Intrinsic BO with VI still presents advantages compared to others in the Aral Sea example. Furthermore, the analysis of the number of inducing points suggests that simply increasing the number of inducing points does not guarantee better performance.

In conclusion, this chapter expands the application of BO to the field of manifolds. The chapter proposes five BO methods suitable for manifolds. Among these methods, Intrinsic BO with VI is prove to be the most promising, consistently obtaining accurate and dependable optimisation results across various manifold structures. It is a powerful tool for solving complex optimisation problems in manifold settings. The next chapter will give a comprehensive discussion throughout this work, highlighting the contributions made and suggesting the potential directions for future studies.

Chapter 8

Conclusion

Considering the more complex geometric structures on manifolds compared to Euclidean space, Traditional GPs cannot be directly applied to manifolds. This thesis proposes several GP approaches dedicated to manifolds and compares them with existing GP methods when applying to manifolds. The superiority of the proposed methods is demonstrated through three representative examples.

8.1 Summary

Standard statistical and machine learning tools typically require input data to lie in Euclidean space. However, many types of data are better represented in more general nonlinear metric spaces, such as Riemannian manifolds. This work aims to explore regression and optimisation on manifolds, focusing on the application and extension of GP methods in the field of manifolds. GP is a well-established theory in Euclidean space, but one of the most challenging tasks to explore GP effectiveness in manifold is how to capture the intrinsic geometric characteristics of the manifold and its complex boundaries. This study proposes several GPs customised for manifolds, specifically SI-GPDIC, SI-GPDTC, and SI-GPVI, and compares them with Traditional GP and Graph GP across three examples, demonstrating the superiority of the proposed methods. Based on these GP methods, several BO approaches for solving optimisation problems on manifolds are developed. The details are presented below.

In chapter 1, this thesis begins by introducing the research motivation and the background behind the topic selection. It outlines the key goals and contributions of the thesis. Additionally, it presents an overview of the case studies that will be explored throughout the thesis, setting the stage for the subsequent chapters.

Chapter 2 provides the theoretical background necessary for deriving and solving the new models. It begins with an introduction to Riemannian geometry which are crucial for understanding the manifold structure. Then, Chapter 2 moves on to the framework of Bayesian optimisation and provides an example of 1-dimensional Bayesian optimisation. This framework lays the foundation for the Bayesian optimisation on manifolds discussed in Chapter 7. Among its topics, the introduction to GPs also establishes the groundwork for GPs on manifolds in Chapters 3, 4 and 6. Finally, Chapter 2 introduces graph theory, with a particular focus on the Graph Laplacian (GL), used in Chapter 5 for Graph GPs.

Chapter 3 explores the application of GPs on manifolds. It begins by introducing Traditional GPs used in Euclidean space based on the RBF kernel, highlighting their limitations in effectively handling the complex boundaries of manifolds. As a result, the chapter introduces the Intrinsic GPs, which considers the geometric features of manifolds, thereby improving the accuracy of evaluations. Intrinsic GPs use the transition density of BM to simulate the heat kernel on manifolds. Furthermore, the chapter addresses the challenge of BM paths at the boundary and proposes a novel "reflection" method based on the Neumann boundary condition. It offers a comparison of the "reflection" and "resample" methods, along with their respective advantages and disadvantages. In one-dimensional space, the "reflection" method clearly outperforms the "resample" method, particularly in estimating the heat kernel near boundaries. In two-dimensional space, the accuracy of these methods is assessed by comparing the predictive means of the Intrinsic GP against true function values. In Section 3.3, the challenges in calculating the predictive variance for the Intrinsic GP—specifically the need to simulate BM from each grid point—are highlighted. The next chapter will propose three new GPs by utilizing sparse methods to address this issue.

Chapter 4 proposes the use of sparse methods to address the computational challenges inherent in Intrinsic GPs discussed earlier. This chapter aims to improve the computational efficiency while maintaining accuracy in GP models on manifolds. Sparse methods greatly reduce the

computational complexity of Intrinsic GPs by introducing a set of m inducing points z, making it feasible for large-scale datasets and high-dimensional manifolds. BM paths only need to be simulated from the inducing points instead of the grid points, where $m << G^{'}$. Additionally, the computational complexity of the inversion in Intrinsic GPs, which is typically $O(n^3)$, is reduced to $O(nm^2)$ using sparse methods, where $m < n$, significantly reducing the computational cost. This chapter proposes three Sparse Intrinsic GP methods:

- Sparse Intrinsic GP with DIC (SI-GPDIC) - DIC ensures that information flows from training points to testing points exclusively through the inducing points. However, DIC has limitations in predictive accuracy due to the degeneracy of the prior. As test points move further from inducing points, the predicted variance unreasonably approaches zero.
- Sparse Intrinsic GP with DTC (SI-GPDTC) - It shares the same mean as SI-GPDIC but resolves the issues present in the DIC method. The key difference between these two approaches is that f_{r^*} possesses its own prior variance, denoted by Σ_{rr}, instead of $\Sigma_{rz}\Sigma_{zz}^{-1}\Sigma_{zr}$. This prior variance corrects the uncertainty problem found in DIC.
- Sparse Intrinsic GP with VI (SI-GPVI) - It provides more accurate and stable predictions. Instead of relying on exact or sampling-based methods, SI-GPVI transforms this problem into an optimisation problem, making it more efficient and tractable. It approximates the true posterior distribution with a simpler distribution by minimizing the KL divergence between them. Minimizing the KL divergence, as derived in Section 4.4, is equivalent to maximizing the lower bound of the log marginal likelihood function. Building on this, Section 4.4 derives the predictive distribution for SI-GPVI, simplifying the variance function for practical applications.

This section also explores how the inducing points should be selected. Further applications and comparisons among these methods are provided in Chapter 6.

Chapter 5 explores another theoretical framework Graph GPs on manifolds, with the core idea of representing the manifold as a graph structure. By utilizing the graph Matérn kernel, it extends the application of GPs to manifolds with complex boundaries. The construction of this graph is limited by several factors, such as the number of vertices, the similarity between intrinsic distances on the manifold and Euclidean distances, and the differences between local and

global characteristics of the manifold. These limitations can significantly affect the performance of Graph GPs.

Chapter 6 uses the three cases introduced in Chapter 1—namely, the U-shape, the Bitten-torus, and the real-world dataset Aral Sea—to compare the five GPs on manifolds proposed in the previous chapters. The final example is a real-world dataset, the chlorophyll levels in the Aral Sea. The application of this example is specifically significant because it shows the feasibility and applicability of the proposed methodology in the real world. It also highlights the utility of these GP methods in environmental studies. RMSE and PLL are two metrics used for comparison in this research. In each example, 20 sets of experimental groups are randomly selected from the grid points S on the manifold. These examples demonstrate the limitations of the Traditional GP and Graph GP, which rely on Euclidean distance and perform poorly in capturing the true manifold structure. The Sparse Intrinsic GP methods perform better than the others. Among them, SI-GPVI demonstrates superior performance, showcasing its accuracy and robustness in applications on manifolds. This makes it a strong choice for real-world applications.

To address optimisation problems on manifolds, Chapter 7 proposes several BO methodologies, building on the previous works discussed in earlier chapters. BO uses these GPs as surrogate models and the PI as the acquisition function to iteratively search for the optimal point (maximum or minimum, depending on the requirements) within a limited number of iterations. This limitation is due to budget constraints. Understanding the geometry and intricate boundaries of the manifold enables BOs to navigate it more effectively and improve optimisation accuracy. Chapter 7 presents five BO algorithms: Traditional BO, Graph BO, Intrinsic BO with VI, Intrinsic BO with DIC, and Intrinsic BO with DTC. These methods are applied to the same examples used in Chapter 6 for comparative analysis. Intrinsic BO with VI performs the best across all cases, consistent with the superior performance of its surrogate model, SI-GPVI. This not only helps effectively solve optimisation problems on the manifold by providing the most promising solutions, but also highlights the effectiveness of SI-GPVI in capturing the intrinsic geometry of the manifold. In addition, the analysis of inducing point selection suggests that simply increasing their number does not necessarily lead to better performance, emphasising the importance of choosing an appropriate quantity to balance accuracy and computational efficiency.

8.1.1 Main Contribution

The main contribution of this work can be divided into three aspects:

- Chapter 3 improves the method for handling BM paths when reaching the boundary of manifolds in Intrinsic GP. The "resample" method, where the next step of the BM path falls outside the boundary, resamples the step until it within the boundary. However, this approach results in lacking of exploration near the boundary, which leads to reduced accuracy for Intrinsic GP close to the boundary. To address this issue, this thesis introduces the "reflection" method. When the next step falls outside the boundary, the path is reflected back into the manifold while still preserving a small segment of the path that reached the boundary, as shown in Figure 3.2. This method provides more accurate boundary information and helps Intrinsic GP to better approximate the true heat kernel at the boundary, which is particularly evident in the one-dimensional example shown in Figure 3.4. The limitation of this "reflection" method arises from its computational difficulty, which increases with boundary complexity. The choice between "reflection" and "resample" depends on the specific task objective, and the requirements for computational complexity and accuracy.

- This thesis proposes three GPs methods specifically designed for manifolds: SI-GPDIC, SI-GPDTC, and SI-GPVI , increasing the feasibility of applying GPs to larger datasets or higher-dimensional manifolds. These methods hold significant importance as they address the computational challenges posed by Intrinsic GP, as well as retain the ability to capture the intrinsic geometry of the manifold, which is essential for accurate predictions, especially in domains where the Euclidean distance between points fails to reflect their true relationship on the manifold. These methods not only relieved the simulation from a large number $G^{'}$ of grid points S for N_{bm} times, but instead, require simulations from only a small number m of inducing points z, where $m << G^{'}$. This greatly reduces the computational burden associated with BM path simulations. Additionally, they reduce the high computational complexity $O(n^3)$ of inverting the covariance matrix in GPs to $O(nm^2)$, where n is the number of training points and $m << n$. Among these methods, SI-GPDIC, while achieving good results for the predictive mean, is highly sensitive to the location of the inducing points in its predictive variance, often providing incorrect results

when far from the inducing points. SI-GPDTC addresses this issue, offering more reliable predictive variance while maintaining the same mean as SI-GPDIC. SI-GPVI incorporates variational inference, transforming the problem of posterior inference into an optimisation problem. It simplifies the lower bound of the log marginal likelihood and the predictive variance function by approximating the diagonal elements. SI-GPVI is proved to outperform the other methods, making it the most reliable and effective algorithm for use on manifolds.

- This thesis expands the application of BO on manifolds, making significant contributions to solving optimisation problems on manifolds. While BO is widely used in Euclidean space, directly applying it to manifolds often fails to find the optimum point, because it tends to overlook the intrinsic geometric structure of the manifold. Inspired by SI-GPDIC, SI-GPDTC, and SI-GPVI's ability to capture the geometric structure of manifolds, this thesis proposes several BO methods suited for manifolds: Intrinsic BO with DIC, Intrinsic BO with DTC, and Intrinsic BO with VI. Additionally, this thesis also proposes Traditional BO and Graph BO, which mainly rely on Euclidean distances on the manifold. Thanks to the superiority of SI-GPVI, Intrinsic BO with VI consistently shows the best performance in finding the optimum point on manifolds compared to other BO methods. The limitation is that when applied in real-world scenarios, where the objective function is often non-smooth and noisy, Intrinsic BO with VI, although superior to other methods, may struggle to maintain its performance.

8.1.2 Limitation

Although the methodologies proposed in this research have demonstrated their effectiveness in the text, there are still some limitations in practical applications. The three examples presented in this thesis contain two-dimensional as well as three-dimensional manifolds, and the application of the proposed methodologies to higher dimensions would involve more complex modeling and more factors to be considered, which remain to be explored. The "reflection" method faces limitations due to its increasing computational complexity as boundary conditions become more intricate. BM paths not only need to determine whether the next step is outside the domain, but also identify the specific boundary segment they have crossed and perform the necessary calcu-

lations. Sometimes, a single reflection may not be enough to ensure that the BM path goes back inside the boundary, after the initial reflection, the path might cross another boundary, remaining outside the manifold and requiring multiple reflections. However, given the computational complexity of the "reflection" method and its limited contribution to improving overall regression accuracy, the "resample" method, with its easier implementation and simplicity, remains a strong choice in practical applications. Additionally, although SI-GPVI and Intrinsic BO with VI have demonstrated superiority over other methods, their advantages are limited when dealing with real-world cases like the Aral Sea, where the data is non-smooth and noisy. Further exploration is required to improve the robustness of SI-GPVI and Intrinsic BO with VI when dealing with this type of data.

8.2 Future work

This work proposes effective methods to address regression and optimisation problems on manifolds. However, various extensions can be made to this work. For example, this work anticipates applications in higher-dimensional spaces, as the examples currently provided remain in three dimensions. Expanding to handle more complex, higher-dimensional manifolds would further demonstrate the robustness and applicability of these proposed methods and help address more challenges in manifold study. Sometimes, a single reflection may not be enough to ensure the path returns inside the boundary. After the initial reflection, the BM path might cross another boundary, remain outside the manifold, and require multiple reflections. Additionally, in some cases, the BM path may cross more than one boundary, making it necessary to accurately determine which boundary segment is reached first.

Below are additional potential directions for future research:

- The inducing points z used in this work are predefined and approximately uniformly distributed across the manifold. The inducing variables u induces the dependencies between the training cases $\mathbf{f}_{\mathscr{D}}$ and testing cases $\mathbf{f}_r$, leaving an imprint on the final solution [126]. Despite the promising results of SI-GPVI on manifolds, optimising the location of inducing points z would theoretically improve the model's accuracy and efficiency. Several greedy selection approaches have been suggested for GPs on Euclidean space. Csató et al.

[31] use an online selection process that updates z by adding new points to the inducing set only when their projection error exceeds a threshold. Keerthi et al. [74] use greedy forward selection to choose z by adding points that maximize the reduction in negative log-posterior. Snelson et al. [137] treat z as a parameter and optimise it by maximising the marginal likelihood with respect to z. Smola et al. [136] maximise the effective posterior instead of the marginal likelihood to find the optimal z. However, these methods are not suitable for sparse intrinsic GPs, as the computational burden caused by simulating BM paths makes it infeasible to determine the optimal z through greedy selection approaches. Within the framework of sparse intrinsic GPs, how to optimise the location of inducing points remains an area for further research.

- This thesis focuses on manifolds with complex boundaries, assuming that the intrinsic geometric features of the manifold are known. There has been abundant interest in learning of In-BO on unknown manifolds, which means no intrinsic geometry available in advance. This is a more practical approach for real-world applications, as accurately measuring the geometry of data manifolds is often either impossible or expensive. Fichera et al. [41] propose the Graph GPs that are capable of inferring the implicit structure of the manifold directly from data. Peach et al. [119] propose Riemannian manifold vector field GP (RVGP), which models vector fields on implicit manifolds. The most closely related research is by Mu et al. [113], which propose the GP on Unknown Manifolds (GPUM) methodology to learn the manifold's geometry using probabilistic latent variable models, where BM simulated paths work. Inspired by these works, this research also extends to address regression and optimisation problems on unknown manifolds, offering a wide range of applications.

- This work uses the Aral sea dataset as a real world case study, which provides the chlorophyll levels of the region. Both SI-GPVI and Intrinsic BO with VI achieve promising results in this case. These applications provide concrete examples for environmental protection and water pollution management. However, the real-world applications of manifolds extend far beyond these area. In various branches of medical imaging analysis and computer vision, manifolds play a crucial role, for example, 3D Human model in [46], white matter geometry in [88], diffusion tensor imaging in [121], diffusion magnetic resonance images in [157] and so on. Research on manifolds can also be applied in geology

[66] [171]. For instance, geological formations, such as terrain surfaces and subsurface structures, often exhibit complex, non-Euclidean geometries that can be modelled as manifolds. Ignoring the intrinsic geometric features of these manifolds often leads to inaccurate modelling and suboptimal results. Although many challenges remain with manifolds in these domains, this work's ability to capture the internal structure of manifolds holds significant potential for application in these fields.

Appendix A

Fundamental Solution to the Heat Kernel

This appendix aims to derive the fundamental solution to the heat equation:

$$\partial_t u - \Delta u = 0. \tag{A.1}$$

Suppose $u(x,t)$ is a solution to PDE (A.1) which takes form of $u(x,t) = w(t)v\left(\frac{x^2}{t}\right)$, a straightforward calculation gives

$$\begin{aligned}
u_t(x,t) &= w'(t)v\left(\frac{x^2}{t}\right) - w(t)v'\left(\frac{x^2}{t}\right)\frac{x^2}{t^2};\\
u_x(x,t) &= w(t)v'\left(\frac{x^2}{t}\right)\frac{2x}{t};\\
u_{xx}(x,t) &= w(t)v''\left(\frac{x^2}{t}\right)\frac{4x^2}{t^2} + w(t)v'\left(\frac{x^2}{t}\right)\frac{2}{t}.
\end{aligned}$$

According to (A.1), it satisfies:

$$w'(t)v\left(\frac{x^2}{t}\right) - \frac{w(t)}{t}\left[v''\left(\frac{x^2}{t}\right)\frac{4x^2}{t} + 2v'\left(\frac{x^2}{t}\right) + v'\left(\frac{x^2}{t}\right)\frac{x^2}{t}\right] = 0.$$

Then,

$$\frac{w'(t)t}{w(t)} = \frac{4\frac{x^2}{t}v''(\frac{x^2}{t}) + 2v'(\frac{x^2}{t}) + \frac{x^2}{t}v'(\frac{x^2}{t})}{v(s)}. \tag{A.2}$$

To ensure that both sides of Equation A.2 remain equal, both sides of this equality must be constant. Let this constant be denoted as λ, then

$$\frac{\frac{4x^2}{t}v''(\frac{x^2}{t})+2v'(\frac{x^2}{t})+\frac{x^2}{t}v'(\frac{x^2}{t})}{v(s)}=\lambda,$$
$$\frac{x^2}{t}\left(4v''+v'\right)+\frac{1}{2}\left(4v'-2\lambda v\right)=0.$$

By giving $\lambda=-\frac{1}{2}$ and $4v'+v=0$, the equation is satisfied and $v(\frac{x^2}{t})=e^{-\frac{x^2}{4t}}$. Correspondingly,

$$\frac{w'(t)t}{w(t)}=-\frac{1}{2},$$
$$w(t)=t^{-\frac{1}{2}}.$$

Therefore,

$$u=u_1(x,t)=\frac{1}{\sqrt{t}}e^{-\frac{x^2}{4t}}$$

is a solution of $u_t=u_{xx}$. Then,

$$u(x_1,x_2,\cdots,x_n,t)=u_1(x_1,t)u_2(x_2,t)...u_n(x_n,t)=\frac{1}{t^{n/2}}e^{-\frac{|x|^2}{4t}}.$$

To ensure that the integral of the solution over the entire space equals 1, that is, for each $t>0$:

$$\int_{\mathbb{R}^n}\Phi(x_1,\ldots,x_n,t)\,dx_1\cdots dx_n=1,$$

it is necessary to use $\frac{1}{(4\pi)^{n/2}}$ as a normalization factor. Then, the fundamental solution to the heat equation (A.1) is expressed as:

$$\Phi(x_1,\cdots,x_n,t)=\begin{cases}\frac{1}{(4\pi t)^{n/2}}e^{-\frac{|x|^2}{4t}} & (t>0)\\ 0 & (t\leq 0)\end{cases}.$$

Appendix B

Martingale Properties of Brownian Motion

BM has many important properties. Here, the proof of its martingale property is presented. First, the definition of the martingale property is as follows [87]:

Definition B.1. *A real-valued stocastic process* $\{X(t)\}_{t\geq 0}$ *is a martingale with respect to a filtration* $\{\mathscr{F}(t)\}$ *if it is adapted, that is,* $X(t) \in \mathscr{F}(t)$ *for all* $t \geq 0$*, if* $E|X(t)| < +\infty$ *for all* $t \geq 0$*, and if*

$$\mathbb{E}[X(t) \mid \mathscr{F}(s)] = X(s)$$

almost surely, for all $0 \leq s \leq t$.

To prove the martingale property of BM, it is only necessary to show that the following equation holds:

$$\mathbb{E}[B(t) \,|\, \mathscr{F}_s] = B(s), \tag{B.1}$$

where $\{B(t)\}$ is a standard BM. Due to the independence of $B(t) - B(s)$ from $\mathscr{F}(s)$,

$$E\left[B(t) - B(s) \mid \mathscr{F}_s\right] = E[B(t) - B(s)],$$

it is easy to obtain:

$$\begin{aligned} E\left[B(t) \mid \mathscr{F}_s\right] &= E\left[B(s) + (B(t) - B(s)) \mid \mathscr{F}_s\right] \\ &= E\left[B(s) \mid \mathscr{F}_s\right] + E\left[B(t) - B(s) \mid \mathscr{F}_s\right] \\ &= B(s) + E[B(t) - B(s)] = B(s), \end{aligned}$$

which proves the martingale property of BM.

Appendix C

The Taylor Expansion for $p_{\text{resample}}(x)$ And $p_{\text{reflect}}(x)$

Section 3.4.1 has provided the formulas for $p_{\text{resample}}(x)$ nd $p_{\text{reflect}}(x)$. Thus, the difference between them is given by:

$$p_{\text{resample}}(x) - p_{\text{reflect}}(x) = \left(\frac{1}{\varphi(\tau)} - 1\right) \cdot \frac{1}{\sqrt{2\pi\Delta t}} \exp\left(-\frac{(x-\tau)^2}{2\Delta t}\right) - \frac{1}{\sqrt{2\pi\Delta t}} \exp\left(-\frac{(x+\tau)^2}{2\Delta t}\right)$$

The Taylor expansion for the standard normal CDF near $\tau = 0$ is used:

$$\varphi(\tau) = \frac{1}{2} + \frac{\tau}{\sqrt{2\pi}} + O\left(\tau^3\right).$$

From this, it follows that:

$$\frac{1}{\varphi(\tau)} = 2 - \frac{2\tau}{\sqrt{2\pi}} + O\left(\tau^2\right).$$

Thus,

$$\frac{1}{\varphi(\tau)} - 1 = 1 - \frac{2\tau}{\sqrt{2\pi}} + O\left(\tau^2\right).$$

Next, the exponentials are expanded:

$$\exp\left(-\frac{(x \pm \tau)^2}{2\Delta t}\right) = \exp\left(-\frac{x^2}{2\Delta t}\right)\left(1 \pm \frac{x\tau}{\Delta t} + O\left(\tau^2\right)\right).$$

Using these expansions, the difference between the two PDFs becomes:

$$p_{\text{resample}}(x) - p_{\text{reflect}}(x) = \frac{1}{\sqrt{2\pi\Delta t}} \exp\left(-\frac{x^2}{2\Delta t}\right) \left[\left(1 - \frac{2\tau}{\sqrt{2\pi}}\right)\left(1 + \frac{x\tau}{\Delta t}\right) - \left(1 - \frac{x\tau}{\Delta t}\right)\right] + O\left(\tau^2\right).$$

The terms inside the brackets can be expanded as follows:

$$\left(1 - \frac{2\tau}{\sqrt{2\pi}}\right)\left(1 + \frac{x\tau}{\Delta t}\right) = 1 + \frac{x\tau}{\Delta t} - \frac{2\tau}{\sqrt{2\pi}} - \frac{2x\tau^2}{\sqrt{2\pi}\Delta t}.$$

Then,

$$p_{\text{resample}}(x) - p_{\text{reflect}}(x) = \frac{1}{\sqrt{2\pi\Delta t}} \exp\left(-\frac{x^2}{2\Delta t}\right) \left[\frac{2x\tau}{\Delta t} - \frac{2\tau}{\sqrt{2\pi}} + O\left(\tau^2\right)\right]$$

The term τ can be factored out from the expression as follows:

$$p_{\text{resample}}(x) - p_{\text{reflect}}(x) = \tau \cdot \frac{1}{\sqrt{2\pi\Delta t}} \exp\left(-\frac{x^2}{2\Delta t}\right) \left[\frac{x}{\Delta t} - \frac{2}{\sqrt{2\pi}} + O(\tau)\right].$$

Since τ is an explicit factor in the expression, taking the limit yields:

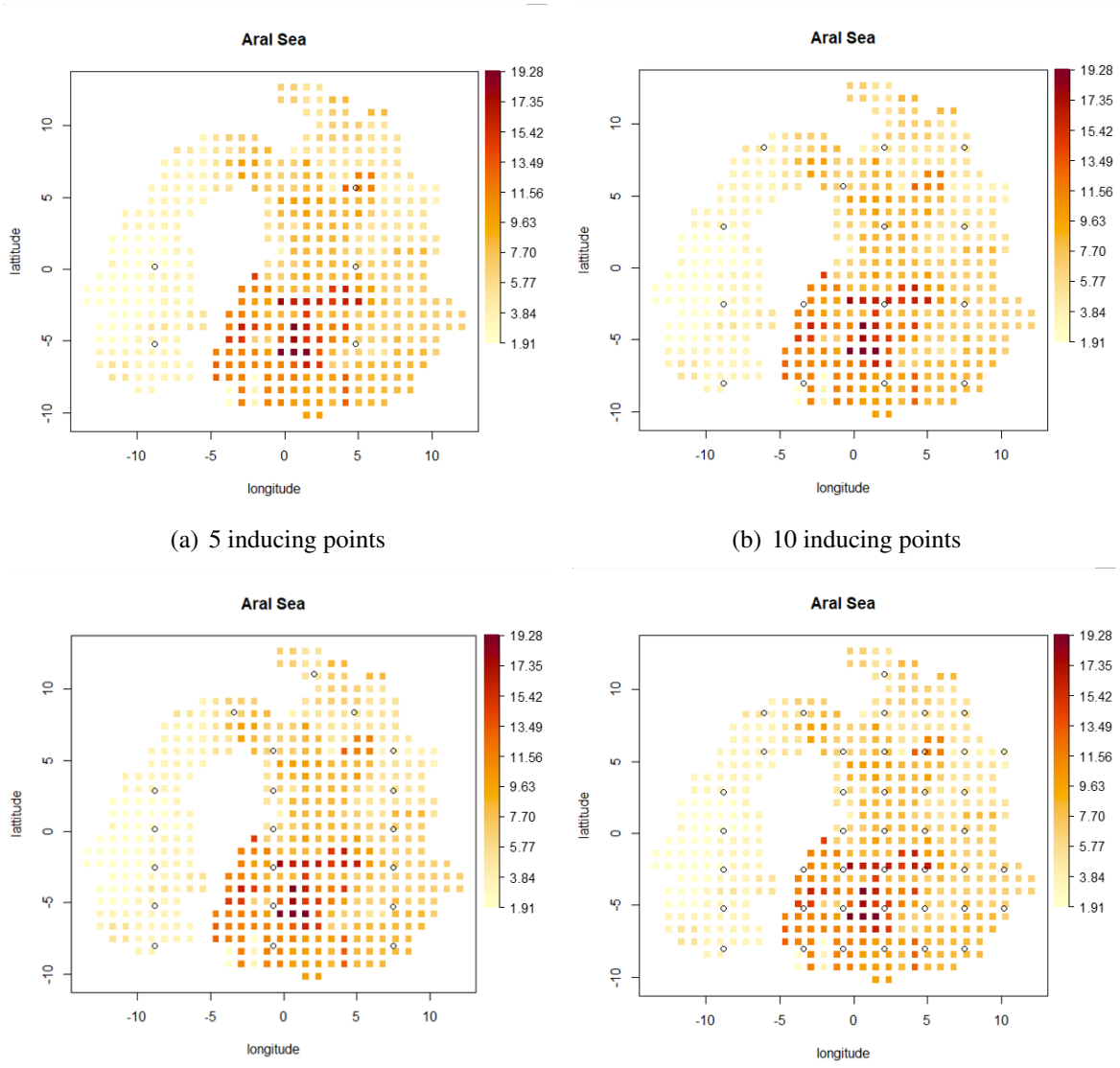
$$\lim_{\tau \to 0^+} \left(p_{\text{resample}}(x) - p_{\text{reflet}}(x)\right) = 0,$$

which leads to the conclusion presented in Section 3.4.1.

Appendix D

Inducing Points Placement in the Aral Sea

The following illustrates the locations of inducing points with different quantities in the Aral Sea dataset.



(a) 5 inducing points

(b) 10 inducing points

(c) 15 inducing points

(d) 42 inducing points

Figure D.1: Comparison of inducing point distributions in the Aral Sea dataset with varying numbers of inducing points (5, 10, 15, and 42).

Bibliography

[1] G Alexopoulos. “An application of homogenization theory to harmonic analysis on solvable Lie groups of polynomial growth”. In: *Pacific Journal of Mathematics* 159.1 (1993), pp. 19–45.

[2] Jean-Philippe Anker and Lizhen Ji. “Heat kernel and Green function estimates on noncompact symmetric spaces”. In: *Geometric and Functional Analysis* 9 (1999), pp. 1035–1091.

[3] Jean-Philippe Anker and Patrick Ostellari. “The heat kernel on noncompact symmetric spaces”. In: *Lie groups and symmetric spaces* (2003), pp. 27–46.

[4] Nima Asgharbeygi and Arian Maleki. “Geodesic k-means clustering”. In: *2008 19th International Conference on Pattern Recognition*. IEEE. 2008, pp. 1–4.

[5] The GPyOpt authors. *GPyOpt: A Bayesian Optimization framework in python*. `http://github.com/SheffieldML/GPyOpt`. 2016.

[6] Javad Azimi, Ali Jalali, and Xiaoli Fern. “Hybrid batch Bayesian optimization”. In: *arXiv preprint arXiv:1202.5597* (2012).

[7] Martin T Barlow. “Heat kernels and sets with fractal structure”. In: *Contemporary Mathematics* 338 (2003), pp. 11–40.

[8] Alexander Bendikov and L Saloff-Coste. “On-and off-diagonal heat kernel behaviors on certain infinite dimensional local Dirichlet spaces”. In: *American Journal of Mathematics* 122.6 (2000), pp. 1205–1263.

[9] Marcel Berger. *Riemannian geometry during the second half of the twentieth century*. Vol. 17. American Mathematical Soc., 2000.

[10] Nicole Berline, Ezra Getzler, and Michele Vergne. *Heat kernels and Dirac operators*. Springer Science & Business Media, 2003.

[11] Francisco Bernis and Avner Friedman. "Higher order nonlinear degenerate parabolic equations". In: *Journal of differential equations* 83.1 (1990), pp. 179–206.

[12] Uzair Aslam Bhatti et al. "Deep learning with graph convolutional networks: An overview and latest applications in computational intelligence". In: *International Journal of Intelligent Systems* 2023.1 (2023), pp. 1–28.

[13] Norman Biggs. *Algebraic graph theory*. 67. Cambridge university press, 1993.

[14] Christopher M Bishop and John M Winn. "Non-linear Bayesian image modelling". In: *Computer Vision-ECCV 2000: 6th European Conference on Computer Vision Dublin, Ireland, June 26–July 1, 2000 Proceedings, Part I 6*. Springer. 2000, pp. 3–17.

[15] David Bolin, Alexandre B Simas, and Jonas Wallin. "Gaussian Whittle–Matérn fields on metric graphs". In: *Bernoulli* 30.2 (2024), pp. 1611–1639.

[16] Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, et al. "Matérn Gaussian processes on Riemannian manifolds". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 12426–12437.

[17] Viacheslav Borovitskiy et al. "Matérn Gaussian Processes on Graphs". In: *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*. Vol. 130. PMLR, 2021, pp. 2593–2601.

[18] François-Xavier Briol et al. "Rejoinder: Probabilistic integration: A role in statistical computation". In: *Statistical Science* 34.1 (2019), pp. 38–42.

[19] Eric Brochu, Tyson Brochu, and Nando De Freitas. "A Bayesian interactive optimization approach to procedural animation design". In: *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 2010, pp. 103–112.

[20] Eric Brochu, Vlad M Cora, and Nando De Freitas. "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning". In: *arXiv preprint arXiv:1012.2599* (2010).

[21] Andrei Broder et al. "Graph structure in the web". In: *Computer networks* 33.1-6 (2000), pp. 309–320.

[22] Bryan W Brooks et al. "Are harmful algal blooms becoming the greatest inland water quality threat to public health and aquatic ecosystems?" In: *Environmental toxicology and chemistry* 35.1 (2016), pp. 6–13.

[23] Adam D Bull. "Convergence rates of efficient global optimization algorithms". In: *Journal of Machine Learning Research* 12.10 (2011), pp. 2879–2904.

[24] Sait Cakmak et al. "Bayesian optimization of risk measures". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 20130–20141.

[25] Ismaël Castillo, Gérard Kerkyacharian, and Dominique Picard. "Thomas Bayes' walk on manifolds". In: *Probability Theory and Related Fields* 158.3 (2014), pp. 665–710.

[26] Nicolas Charon and Alain Trouvé. "The varifold representation of nonoriented shapes for diffeomorphic registration". In: *SIAM journal on Imaging Sciences* 6.4 (2013), pp. 2547–2580.

[27] Isaac Chavel. *Eigenvalues in Riemannian geometry*. Academic press, 1984.

[28] Fan RK Chung. *Spectral graph theory*. Vol. 92. American Mathematical Soc., 1997.

[29] Jon Cockayne et al. "Probabilistic meshless methods for partial differential equations and Bayesian inverse problems". In: *arXiv preprint arXiv:1605.07811* (2016).

[30] Ronald R Coifman and Stéphane Lafon. "Diffusion maps". In: *Applied and computational harmonic analysis* 21.1 (2006), pp. 5–30.

[31] Lehel Csató and Manfred Opper. "Sparse on-line Gaussian processes". In: *Neural computation* 14.3 (2002), pp. 641–668.

[32] Brad C Davis et al. "Population shape regression from random design data". In: *International journal of computer vision* 90 (2010), pp. 255–266.

[33] Marc Peter Deisenroth, Dieter Fox, and Carl Edward Rasmussen. "Gaussian processes for data-efficient learning in robotics and control". In: *IEEE transactions on pattern analysis and machine intelligence* 37.2 (2013), pp. 408–423.

[34] SAJ Dekkers. "Finite propagation speed for solutions of the parabolic $\{p\}$-Laplace equation on manifolds". In: *Communications in Analysis and Geometry* 13.4 (2005), pp. 741–768.

[35] Bruce K Driver. “Heat kernels measures and infinite dimensional analysis”. In: *Contemporary Mathematics* 338 (2003), pp. 101–142.

[36] David B Dunson, Hau-Tieng Wu, and Nan Wu. “Graph based Gaussian processes on restricted domains”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 84.2 (2022), pp. 414–439.

[37] Luis Escauriaza. “Bounds for the fundamental solutions of elliptic and parabolic equations: In memory of Eugene Fabes”. In: *Communications in Partial Differential Equations* 25.5-6 (2000), pp. 821–845.

[38] Lawrence C Evans. *Partial differential equations*. Vol. 19. American Mathematical Society, 2022.

[39] Aasa Feragen, Francois Lauze, and Soren Hauberg. “Geodesic exponential kernels: When curvature and linearity conflict”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3032–3042.

[40] Tamara Fernández, Nicolás Rivera, and Yee Whye Teh. “Gaussian processes for survival analysis”. In: *Advances in Neural Information Processing Systems* 29 (2016), pp. 5021–5029.

[41] Bernardo Fichera et al. “Implicit manifold Gaussian process regression”. In: *Advances in Neural Information Processing Systems* 36 (2024), pp. 67701–67720.

[42] P Thomas Fletcher and Sarang Joshi. “Principal geodesic analysis on symmetric spaces: Statistics of diffusion tensors”. In: *International Workshop on Mathematical Methods in Medical and Biomedical Image Analysis*. Springer. 2004, pp. 87–98.

[43] P Thomas Fletcher, Conglin Lu, and Sarang Joshi. “Statistics of shape via principal geodesic analysis on Lie groups”. In: *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings*. Vol. 1. IEEE. 2003, pp. 95–101.

[44] P Thomas Fletcher, Suresh Venkatasubramanian, and Sarang Joshi. “The geometric median on Riemannian manifolds with application to robust atlas estimation”. In: *NeuroImage* 45.1 (2009), pp. 143–152.

[45] Alexander Forrester, Andras Sobester, and Andy Keane. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.

[46] Oren Freifeld and Michael J Black. "Lie bodies: A manifold representation of 3D human shape". In: *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part I 12*. Springer. 2012, pp. 1–14.

[47] Fernando Gama et al. "Graphs, convolutions, and neural networks: From graph filters to graph neural networks". In: *IEEE Signal Processing Magazine* 37.6 (2020), pp. 128–138.

[48] Nicolás Garcıa Trillos et al. "Error estimates for spectral convergence of the graph Laplacian on random geometric graphs toward the Laplace–Beltrami operator". In: *Foundations of Computational Mathematics* 20.4 (2020), pp. 827–887.

[49] Roman Garnett, Michael A Osborne, and Stephen J Roberts. "Bayesian optimization for sensor set selection". In: *Proceedings of the 9th ACM/IEEE international conference on information processing in sensor networks*. 2010, pp. 209–219.

[50] Arthur Genthon. "The concept of velocity in the history of Brownian motion: From physics to mathematics and back". In: *The European Physical Journal H* 45.1 (2020), pp. 49–105.

[51] Leonor Godinho and José Natário. "An introduction to riemannian geometry". In: *With Applications* (2012).

[52] Izrail Solomonovich Gradshteyn and Iosif Moiseevich Ryzhik. *Table of integrals, series, and products*. Academic press, 2014.

[53] Alexander Grigor'yan. "Heat kernels and function theory on metric measure spaces". In: *Contemporary Mathematics* 338 (2003), pp. 143–172.

[54] Alexander Grigor'yan and Andras Telcs. "Sub-Gaussian estimates of heat kernels on infinite graphs". In: *Duke Mathematical Journal* 109.3 (2001), pp. 451–510.

[55] Alexander Grigor'yan. "Heat kernels on weighted manifolds and applications". In: *Cont. Math* 398.2006 (2006), pp. 93–191.

[56] Alexander Grigoryan. *Heat kernel and analysis on manifolds*. Vol. 47. American Mathematical Soc., 2009.

[57] Jose M Gutierrez, Michael Jensen, and Tahir Riaz. “Applied graph theory to real smart city logistic problems”. In: *Procedia Computer Science* 95 (2016), pp. 40–47.

[58] Ivan Gutman and Nenad Trinajstić. “Graph theory and molecular orbitals”. In: *New Concepts II*. Springer, 2005, pp. 49–93.

[59] Frank Harary and Robert Z Norman. “Graph theory as a mathematical model in social science”. In: *Research Center for Group Dynamics* (1953), pp. 1–27.

[60] Matthias Hein, Jean-Yves Audibert, and Ulrike von Luxburg. “Graph laplacians and their convergence on random neighborhood graphs”. In: *Journal of Machine Learning Research* 8.6 (2007), pp. 1325–1368.

[61] John R Hershey and Peder A Olsen. “Approximating the Kullback Leibler divergence between Gaussian mixture models”. In: *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP’07*. Vol. 4. IEEE. 2007, pp. 317–320.

[62] Timothy O Hodson. “Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not”. In: *Geoscientific Model Development Discussions* (2022), pp. 1–10.

[63] Matthew Hoffman, Bobak Shahriari, and Nando Freitas. “On correlation and budget constraints in model-based bandit optimization with application to automatic machine learning”. In: *Artificial Intelligence and Statistics*. PMLR. 2014, pp. 365–374.

[64] Elton P Hsu. “A brief introduction to Brownian motion on a Riemannian manifold”. In: *lecture notes* (2008).

[65] Pei Hsu. “Brownian motion and Riemannian geometry”. In: *Contemp. Math* 73 (1988), pp. 95–104.

[66] Mengsu Hu and Jonny Rutqvist. “Numerical manifold method modeling of coupled processes in fractured geological media at multiple scales”. In: *Journal of Rock Mechanics and Geotechnical Engineering* 12.4 (2020), pp. 667–681.

[67] Stephan Huckemann, Thomas Hotz, and Axel Munk. “Intrinsic shape analysis: Geodesic PCA for Riemannian manifolds modulo isometric Lie group actions”. In: *Statistica Sinica* (2010), pp. 1–58.

[68] Iroshani Jayawardene, Pramod Herath, and Ganesh Kumar Venayagamoorthy. “A graph theory-based clustering method for power system networks”. In: *2020 Clemson University Power Systems Conference (PSC)*. IEEE. 2020, pp. 1–8.

[69] X Jane Jiang and Paul J Scott. *Advanced metrology: freeform surfaces*. Academic Press, 2020.

[70] Donald R Jones. “A taxonomy of global optimization methods based on response surfaces”. In: *Journal of global optimization* 21.4 (2001), pp. 345–383.

[71] Michael I Jordan et al. “An introduction to variational methods for graphical models”. In: *Machine learning* 37 (1999), pp. 183–233.

[72] Peter E Jupp and John T Kent. “Fitting smooth paths to spherical data”. In: *Journal of the Royal Statistical Society Series C: Applied Statistics* 36.1 (1987), pp. 34–46.

[73] Motonobu Kanagawa, Bharath K Sriperumbudur, and Kenji Fukumizu. “Convergence analysis of deterministic kernel-based quadrature rules in misspecified settings”. In: *Foundations of Computational Mathematics* 20 (2020), pp. 155–194.

[74] Sathiya Keerthi and Wei Chu. “A matching pursuit approach to sparse gaussian process regression”. In: *Advances in neural information processing systems* 18 (2005), pp. 643–650.

[75] David G Kendall. “Shape manifolds, procrustean metrics, and complex projective spaces”. In: *Bulletin of the London mathematical society* 16.2 (1984), pp. 81–121.

[76] Marc C Kennedy and Anthony O’Hagan. “Bayesian calibration of computer models”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63.3 (2001), pp. 425–464.

[77] Mohammad M Khajah et al. “Designing engaging games using Bayesian optimization”. In: *Proceedings of the 2016 CHI conference on human factors in computing systems*. 2016, pp. 5571–5582.

[78] Tae Kyun Kim. “T test as a parametric statistic”. In: *Korean journal of anesthesiology* 68.6 (2015), pp. 540–546.

[79] Peter E Kloeden and Eckhard Platen. "Higher-order implicit strong numerical schemes for stochastic differential equations". In: *Journal of statistical physics* 66.1 (1992), pp. 283–314.

[80] R Kondor and J Lafferty. "Diffusion kernels on graphs and other discrete input spaces". In: *ICML*. 2002, pp. 315–322.

[81] Michael J Kozdron. "Brownian Motion and the Heat Equation". In: *University of Regina* (2008), pp. 1–6.

[82] Harold J Kushner. "A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise". In: *Basic Engineering* 86 (1964), pp. 97–106.

[83] Malte Kuss and Carl Rasmussen. "Gaussian processes in reinforcement learning". In: *Advances in neural information processing systems* 16 (2003), pp. 751–759.

[84] Damien Lamberton and Bernard Lapeyre. *Introduction to stochastic calculus applied to finance*. CRC press, 2011.

[85] Gregory F Lawler. *Introduction to stochastic processes*. Chapman and Hall/CRC, 2018.

[86] Neil Lawrence, Matthias Seeger, and Ralf Herbrich. "Fast sparse Gaussian process methods: The informative vector machine". In: *Advances in neural information processing systems* 15 (2002), pp. 625–632.

[87] Jean-François Le Gall. *Brownian motion, martingales, and stochastic calculus*. Springer, 2016.

[88] Christophe Lenglet, Rachid Deriche, and Olivier Faugeras. "Inferring white matter geometry from diffusion tensor MRI: Application to connectivity mapping". In: *Computer Vision-ECCV 2004: 8th European Conference on Computer Vision, Prague, Czech Republic*. Springer. 2004, pp. 127–140.

[89] Didong Li and David B Dunson. "Geodesic distance estimation with spherelets". In: *arXiv preprint arXiv:1907.00296* (2019).

[90] Mikhail Lifshits and Mikhail Lifshits. *Lectures on Gaussian processes*. Springer, 2012.

[91] Aristidis C Likas and Nikolas P Galatsanos. "A variational approach for Bayesian blind image deconvolution". In: *IEEE transactions on signal processing* 52.8 (2004), pp. 2222–2233.

[92] Lizhen Lin et al. “Extrinsic Gaussian processes for regression and classification on manifolds”. In: *Bayesian Analysis* 14.3 (2019), pp. 887–906.

[93] Finn Lindgren, Håvard Rue, and Johan Lindström. “An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 73.4 (2011), pp. 423–498.

[94] Yuan Liu, Mu Niu, and Claire Miller. “Sparse intrinsic Gaussian processes for prediction on manifolds: extending applications to environmental contexts”. In: *International Workshop on Statistical Modelling*. Springer. 2024, pp. 185–190.

[95] Daniel J Lizotte et al. “Automatic Gait Optimization With Gaussian Process Regression”. In: *IJCAI*. Vol. 7. 2007, pp. 944–949.

[96] Benjamin A Logsdon, Gabriel E Hoffman, and Jason G Mezey. “A variational Bayes algorithm for fast and accurate multiple locus genome-wide association analysis”. In: *BMC bioinformatics* 11 (2010), pp. 1–13.

[97] Hengrui Luo, Jeremy E Purvis, and Didong Li. “Spherical rotation dimension reduction with geometric loss functions”. In: *arXiv preprint arXiv:2204.10975* (2022).

[98] GW Ma et al. “Segmented two-phase flow analysis in fractured geological medium based on the numerical manifold method”. In: *Advances in Water Resources* 121 (2018), pp. 112–129.

[99] Pierre Mahé et al. “Graph kernels for molecular structure- activity relationship analysis with support vector machines”. In: *Journal of chemical information and modeling* 45.4 (2005), pp. 939–951.

[100] Nimalan Mahendran et al. “Adaptive MCMC with Bayesian optimization”. In: *Artificial Intelligence and Statistics*. PMLR. 2012, pp. 751–760.

[101] Abdul Majeed and Ibtisam Rauf. “Graph theory: A comprehensive survey about graph theory applications in computer science and social networks”. In: *Inventions* 5.1 (2020), pp. 10–49.

[102] Yongyi Mao et al. “A factor graph approach to link loss monitoring in wireless sensor networks”. In: *IEEE Journal on Selected Areas in Communications* 23.4 (2005), pp. 820–829.

[103] Roman Marchant and Fabio Ramos. “Bayesian optimisation for intelligent environmental monitoring”. In: *2012 IEEE/RSJ international conference on intelligent robots and systems*. IEEE. 2012, pp. 2242–2249.

[104] Ruben Martinez-Cantin et al. “Active policy learning for robot planning and exploration under uncertainty”. In: *Robotics: Science and systems*. Vol. 3. 2007, pp. 321–328.

[105] Robert Meusel et al. “Graph structure in the web—revisited: a trick of the heavy tail”. In: *Proceedings of the 23rd international conference on World Wide Web*. 2014, pp. 427–432.

[106] Bernard De Meyer and Hadiza Moussa Saley. “On the strategic origin of Brownian motion in finance”. In: *International Journal of Game Theory* 31 (2003), pp. 285–319.

[107] Philip Micklin. “The Aral sea disaster”. In: *Annu. Rev. Earth Planet. Sci.* 35.1 (2007), pp. 47–72.

[108] Jonas Mockus. “The application of Bayesian methods for seeking the extremum”. In: *Towards global optimization* 2 (1998), pp. 117–130.

[109] Jonas Mockus. “The Bayesian approach to global optimization”. In: *System Modeling and Optimization: Proceedings of the 10th IFIP Conference New York City, USA*. Springer. 2005, pp. 473–481.

[110] Jonas Močkus. “On Bayesian methods for seeking the extremum”. In: *Optimization techniques IFIP technical conference: Novosibirsk*. Springer. 1975, pp. 400–404.

[111] Peter Mörters and Yuval Peres. *Brownian motion*. Vol. 30. Cambridge University Press, 2010.

[112] John Nash. “The imbedding problem for Riemannian manifolds”. In: *Annals of mathematics* 63.1 (1956), pp. 20–63.

[113] Mu Niu et al. “Intrinsic Gaussian process on unknown manifolds with probabilistic metrics”. In: *Journal of Machine Learning Research* 24.104 (2023), pp. 1–42.

[114] Mu Niu et al. “Intrinsic Gaussian processes on complex constrained domains”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 81.3 (2019), pp. 603–627.

[115] The Earth Observatory. *World of Change: Shrinking Aral Sea*. 2022. URL: https://earthobservatory.nasa.gov/ContentWOC/images/aral/aralsea_tmo_2000238_lrg.jpg.

[116] Venkatesh Padmanabhan and Wm E Souder. "A Brownian motion model for technology transfer: Application to a machine maintenance expert system". In: *Journal of Product innovation management* 11.2 (1994), pp. 119–133.

[117] Jin Suk Pak and Shinsuke Yorozu. "The Laplace-Beltrami operator on a Riemannian manifold". In: *Annals of science the College of Liberal Arts* 26 (1989), pp. 13–15.

[118] Eungyu Park. "Manifold embedding based on geodesic distance for nonstationary spatial estimation in higher dimensions". In: *Journal of Hydrology* 640 (2024), pp. 1–11.

[119] Robert L Peach et al. "Implicit Gaussian process representation of vector fields over arbitrary latent manifolds". In: *arXiv preprint arXiv:2309.16746* (2023).

[120] Xavier Pennec. "Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements". In: *Journal of Mathematical Imaging and Vision* 25 (2006), pp. 127–154.

[121] Xavier Pennec, Pierre Fillard, and Nicholas Ayache. "A Riemannian framework for tensor computing". In: *International Journal of computer vision* 66 (2006), pp. 41–66.

[122] Will Penny, Stefan Kiebel, and Karl Friston. "Variational Bayesian inference for fMRI time series". In: *NeuroImage* 19.3 (2003), pp. 727–741.

[123] William D Penny, Nelson J Trujillo-Barreto, and Karl J Friston. "Bayesian fMRI time series analysis with spatial priors". In: *NeuroImage* 24.2 (2005), pp. 350–362.

[124] Peter Petersen. *Riemannian geometry*. Vol. 171. Springer, 2006.

[125] Yehuda Pinchover. "Large time behavior of the heat kernel and the behavior of the Green function near criticality for nonsymmetric elliptic operators". In: *Journal of functional analysis* 104.1 (1992), pp. 54–70.

[126] Joaquin Quinonero-Candela, Carl Edward Rasmussen, and Christopher KI Williams. "Approximation methods for Gaussian process regression". In: *Large-scale kernel machines*. MIT Press, 2007, pp. 203–223.

[127] Anil Raj, Matthew Stephens, and Jonathan K Pritchard. "fastSTRUCTURE: variational inference of population structure in large SNP data sets". In: *Genetics* 197.2 (2014), pp. 573–589.

[128] Stephen Roberts et al. "Gaussian processes for time-series modelling". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 371.1984 (2013), pp. 1–27.

[129] Matthew R Rudary. *On predictive linear Gaussian models*. University of Michigan, 2009.

[130] Havard Rue and Leonhard Held. *Gaussian Markov random fields: theory and applications*. Chapman and Hall/CRC, 2005.

[131] Ammar Safaie et al. "Manifold methods for assimilating geophysical and meteorological data in Earth system models and their components". In: *Journal of Hydrology* 544 (2017), pp. 383–396.

[132] Andria Sarri, Serge Guillas, and Frederic Dias. "Statistical emulation of a tsunami model for sensitivity analysis and uncertainty quantification". In: *Natural Hazards and Earth System Sciences* 12.6 (2012), pp. 2003–2018.

[133] Matthias W Seeger, Christopher KI Williams, and Neil D Lawrence. "Fast forward selection to speed up sparse Gaussian process regression". In: *International Workshop on Artificial Intelligence and Statistics*. PMLR. 2003, pp. 254–261.

[134] Bobak Shahriari et al. "An entropy search portfolio for Bayesian optimization". In: *arXiv preprint arXiv:1406.4625* (2014).

[135] Bobak Shahriari et al. "Taking the human out of the loop: A review of Bayesian optimization". In: *Proceedings of the IEEE* 104.1 (2015), pp. 148–175.

[136] Alex Smola and Peter Bartlett. "Sparse greedy Gaussian process regression". In: *Advances in neural information processing systems* 13 (2000), pp. 619–625.

[137] Edward Snelson and Zoubin Ghahramani. "Sparse Gaussian processes using pseudo-inputs". In: *Advances in neural information processing systems* 18 (2005), pp. 1259–1266.

[138] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. "Practical bayesian optimization of machine learning algorithms". In: *Advances in neural information processing systems* 25 (2012), pp. 2951–2959.

[139] Niranjan Srinivas et al. "Gaussian process optimization in the bandit setting: No regret and experimental design". In: *arXiv preprint arXiv:0912.3995* (2009).

[140] Anuj Srivastava et al. "Statistical shape analysis: Clustering, learning, and testing". In: *IEEE Transactions on pattern analysis and machine intelligence* 27.4 (2005), pp. 590–602.

[141] Oliver Stegle et al. "A Bayesian framework to account for complex non-genetic factors in gene expression levels greatly increases power in eQTL studies". In: *PLoS computational biology* 6.5 (2010), pp. 1–11.

[142] Oliver Stegle et al. "Gaussian process robust regression for noisy heart rate data". In: *IEEE Transactions on Biomedical Engineering* 55.9 (2008), pp. 2143–2151.

[143] Andrew M Stuart. "Inverse problems: a Bayesian perspective". In: *Acta numerica* 19 (2010), pp. 451–559.

[144] Raghav Subbarao and Peter Meer. "Nonlinear mean shift over Riemannian manifolds". In: *International journal of computer vision* 84 (2009), pp. 1–20.

[145] Erik Sudderth and Michael Jordan. "Shared segmentation of natural scenes using dependent Pitman-Yor processes". In: *Advances in neural information processing systems* 21 (2008), pp. 1585–1592.

[146] Kevin Swersky, Jasper Snoek, and Ryan P Adams. "Multi-task bayesian optimization". In: *Advances in neural information processing systems* 26 (2013), pp. 2004–2012.

[147] Peter Sykacek, Stephen J Roberts, and Maria Stokes. "Adaptive BCI based on variational Bayesian Kalman filtering: an empirical evaluation". In: *IEEE Transactions on biomedical engineering* 51.5 (2004), pp. 719–727.

[148] Michael Thambynayagam. *The Diffusion Handbook: Applied Solutions for Engineers*. McGraw-Hill, 2011.

[149] William R Thompson. "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples". In: *Biometrika* 25.3-4 (1933), pp. 285–294.

[150] Michalis Titsias. "Variational learning of inducing variables in sparse Gaussian processes". In: *Artificial intelligence and statistics*. PMLR. 2009, pp. 567–574.

[151] Alessandra Tosi et al. "Metrics for probabilistic geometries". In: *arXiv preprint arXiv: 1411.7432* (2014).

[152] Primoz Trunk et al. "3D heart model for computer simulations in cardiac surgery". In: *Computers in Biology and Medicine* 37.10 (2007), pp. 1398–1403.

[153] Dmitri V Vassilevich. "Heat kernel expansion: user's manual". In: *Physics reports* 388.5-6 (2003), pp. 279–360.

[154] Julia Vinogradska. "Gaussian processes in reinforcement learning: Stability analysis and efficient value propagation". In: *Technische Universität Darmstadt* (2018), pp. 1–110.

[155] Grace Wahba. *Spline models for observational data*. SIAM, 1990.

[156] Martin J Wainwright, Michael I Jordan, et al. "Graphical models, exponential families, and variational inference". In: *Foundations and Trends® in Machine Learning* 1.1–2 (2008), pp. 1–305.

[157] Yuanxiang Wang et al. "Tracking on the product manifold of shape and orientation for tractography from diffusion MRI". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 3051–3056.

[158] Marcel R Wernand, Hendrik J van der Woerd, and Winfried WC Gieskes. "Trends in ocean colour and chlorophyll concentration from 1889 to 2000, worldwide". In: *PLOS one* 8.6 (2013), pp. 1–20.

[159] P. Whittle. *Stochastic processes in several dimensions*. Vol. 40(2). Bulletin of the International Statistical Institute, 1963, pp. 974–994.

[160] Wikipedia contributors. *Aral Sea — Wikipedia, The Free Encyclopedia*. [Online; accessed 14-September-2024]. 2024. URL: https://en.wikipedia.org/w/index.php?title=Aral_Sea&oldid=1245182055.

[161] Wikipedia contributors. *Euclidean space — Wikipedia, The Free Encyclopedia*. [Online; accessed 15-September-2024]. 2024. URL: https://en.wikipedia.org/w/index.php?title=Euclidean_space&oldid=1244616018.

[162] Wikipedia contributors. *Geodesic — Wikipedia, The Free Encyclopedia*. [Online; accessed 16-September-2024]. 2024. URL: https://en.wikipedia.org/w/index.php?title=Geodesic&oldid=1222113871.

[163] Wikipedia contributors. *Torus — Wikipedia, The Free Encyclopedia*. [Online; accessed 14-September-2024]. 2024. URL: https://en.wikipedia.org/w/index.php?title=Torus&oldid=1245330799.

[164] Christopher Williams and Matthias Seeger. "Using the Nyström method to speed up kernel machines". In: *Advances in neural information processing systems* 13 (2000), pp. 682–688.

[165] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. Vol. 2. 3. MIT press Cambridge, MA, 2006.

[166] Simon N Wood. *Generalized additive models: an introduction with R*. chapman and hall/CRC, 2017.

[167] Simon N Wood. "mgcv: GAMs and generalized ridge regression for R". In: *R news* 1.2 (2001), pp. 20–25.

[168] Simon N Wood, Mark V Bravington, and Sharon L Hedley. "Soap film smoothing". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 70.5 (2008), pp. 931–955.

[169] Mark W Woolrich et al. "Multilevel linear modelling for FMRI group analysis using Bayesian inference". In: *Neuroimage* 21.4 (2004), pp. 1732–1747.

[170] Robert F Woolson. "Wilcoxon signed-rank test". In: *Encyclopedia of biostatistics* 8 (2005), pp. 1–3.

[171] Zhijun Wu et al. "Micro-mechanical modeling of the macro-mechanical response and fracture behavior of rock using the numerical manifold method". In: *Engineering geology* 225 (2017), pp. 49–60.

[172] George Wynne, François-Xavier Briol, and Mark Girolami. "Convergence guarantees for Gaussian process means with misspecified likelihoods and smoothness". In: *Journal of Machine Learning Research* 22.123 (2021), pp. 1–40.

[173] Laurent Younes. “Spaces and manifolds of shapes in computer vision: An overview”. In: *Image and Vision Computing* 30.6-7 (2012), pp. 389–397.

[174] Paul Yushkevich et al. “Continuous medial representations for geometric object modeling in 2D and 3D”. In: *Image and Vision Computing* 21.1 (2003), pp. 17–27.

[175] P Zhang and G Chartrand. *Introduction to graph theory*. Tata McGraw-Hill, 2006.

[176] Yijing Zhou, Wei Cai, and Elton Hsu. “Computation of local time of reflecting Brownian motion and probabilistic representation of the Neumann problem”. In: *arXiv preprint arXiv:1502.01319* (2015).