Abstract

Gaussian Process Regression (GPR) has proven to be an extremely useful methodology in classification, machine learning, geostatistics, genetic analysis, and so on. Estimation of GPR hyperparameters can be done using empirical Bayes methods, that is, by maximizing the marginal likelihood function. A major potential problem, which appears to have received relatively little attention in the literature, is the possible existence of multiple local maxima of the marginal likelihood, as function of the hyperparameters, even for relatively simple problems. In fact, due to the non-convexity of the function, the optimization algorithm may not converge to the global maximum. This project will investigate the circumstances in which multiple local maxima can occur and strategies for finding the global maximum. Analyzing a first simple case, the work will be based on mathematical and numerical analysis of the marginal likelihood function and simulations. The results reveal that the estimated probability of having multiple local maxima, even for a very simple case, is positive.
Acknowledgements

To me, this thesis represents the final stage of my academic path started 6 years ago and which is most likely ending with the Master of Science degree in Stochastics and Financial Mathematics at the University of Amsterdam (UvA). This work was conducted during a research period as a visiting student at the London School of Economics and Political Science, Department of Statistics.

First of all, I would like to thank both my supervisors, Bas Kleijn (UvA) and Wicher Bergsma (LSE). I specifically want to thank them for the helpful and useful suggestions made, for reviewing my work and most of all for stimulating and pushing me to think when I was lost or got stuck while conducting this research. As it was my very first work and attempt to research, this thesis has been a big challenge for me.

Secondly, I would like to thank the LSE and the UvA for giving me the valuable opportunity to conduct my research and write my thesis in another university and environment and Sonja Cox, for her time and being my second reader.

Finally, I am grateful to my family, who has always supported me during my studies throughout all these years and made it possible for me to study abroad, to my best friend Federica, my person and my rock, who has never stopped encouraging me to pursue my goals, to my friend Isabella, who made of Amsterdam my second home, to my most faithful friends and fellow students of this master Caroline, Steven and Giovanni, without whom this master would not have been possible and to all my old and new friends who made this experience unique.
## Contents

Introduction .......................................................... 6

1. Gaussian Process Regression .................................. 10
   1.1. Gaussian processes ........................................... 10
   1.2. Gaussian process regression model ....................... 11
   1.3. Prediction .................................................... 13
   1.4. Covariance functions ...................................... 13

2. Estimating GPR hyperparameters ............................... 16
   2.1. Two different approaches .................................. 16
      2.1.1. The fully Bayes approach ............................ 17
      2.1.2. The empirical Bayes approach ..................... 18
   2.2. Existence of multiple local maxima ..................... 20
   2.3. Asymptotic behaviour of the empirical Bayes methods .. 22

3. Regression with a centered Brownian motion prior .......... 24
   3.1. Main results ................................................ 24
      3.1.1. The case n=2 ........................................ 30
   3.2. Proofs ...................................................... 32

4. Experimental Results ............................................. 37
   4.1. Methods .................................................... 37
      4.1.1. The line search method ............................. 38
      4.1.2. The trust region method ............................ 39
   4.2. Setting the problem ....................................... 40
   4.3. Results: number of local maxima and location of the global maximum . 41
   4.4. Comments and problems .................................... 43

5. Conclusion ........................................................ 46

A. Appendix A ....................................................... 48
   A.1. Bayes’s rule ............................................... 48
   A.2. Eigendecomposition of a matrix ......................... 49
   A.3. Matrix Derivatives ....................................... 50
   A.4. Gaussian Identities ...................................... 50

B. Appendix B ....................................................... 51
   B.1. The Linear Conjugate Gradient Method .................. 51
Introduction

Over the last few decades, Gaussian process regression (GPR) has proven to be an extremely useful methodology in classification, machine learning, geostatistics, genetics and so on, due to many interesting properties, such as e.g. "ease of obtaining and expressing uncertainty in predictions and the ability to capture a wide variety of behaviour through a simple parametrization" [13].

Consider a dataset $\mathcal{D}$ of $n$ observations, $\mathcal{D} = \{(x_i, y_i) : i = 1, \ldots, n\}$, let us denote with $X$ the vector of input values and $Y$ the vector of the outputs or target values, which can be either continuous (in the regression case) or discrete (in the classification case), and we wish to make prediction for new inputs $X_*$ that are not in the training data. Then, the usual regression model is given by

$$y_i = f(x_i) + \epsilon_i, \text{ for } i = 1, \ldots, n,$$

where $\epsilon_i$ is the additive noise or error that we assume to be Gaussian distributed, namely such that for every $i = 1, \ldots, n$ the errors are identical independent normal distributed

$$\epsilon_i \sim \mathcal{N}(0, \sigma^2_n).$$

We now need a prediction function $f$ which can make predictions for all possible input values. A wide variety of methods have been proposed to deal with this problem. One approach consists of restricting our attention to a specific class of functions, let us say, for example, only polynomial functions of the inputs. However, an immediate issue comes along. We are deciding upon the richness of the class of functions. If we for instance choose to use the class of polynomial functions and the target function is not well modelled by this class, then the predictions will be poor. Hence, another approach consists of giving prior probability to predictive functions, where higher probabilities are given to functions considered to be more likely than others, for instance the class of all differentiable functions. Yet again a problem remains, i.e. there exists a much larger class of possible functions. This is where Gaussian process comes [1]. Gaussian processes are "widely used as building blocks for prior distributions of unknown functional parameters" [11]. Thus, in our case, for the prior distribution of the predictive function $f$, which can be seen as a parameter of the GPR model.

We have that a stochastic process indexed by a set $T$ is a collection of random variables $W = (W(t) : t \in T)$ defined on a common probability space [27]. Then, a Gaussian process $W$ is a stochastic process such that the finite-dimensional distributions of $(W(t_1), W(t_2), \ldots, W(t_m))$ for all $t_1, \ldots, t_m \in T$ and $m \in \mathbb{Z}$, are multivariate normally distributed [27]. In particular, we have that the finite dimensional-distributions
are completely specified by the mean function \( m : T \to \mathbb{R} \) and covariance function \( k(s, t) : T \times T \to \mathbb{R} \), given by
\[
m(t) = \mathbb{E}[W(t)] \quad \text{and} \quad k(s, t) = \text{cov}(W(s), W(t)).
\]

Furthermore, if we consider the Gaussian process \( W \), we can define the sample path \( t \mapsto W(t) \) of \( W \), which is a random function, and the law of \( W \) is then a prior distribution on the space of functions \( f : T \to \mathbb{R} \). Thus, we see that a stochastic process governs the properties of functions \([1]\).

If we go back again to the regression model (0.1) and the prediction function \( f \), since we can think of a function as a very long vector, such that each entry is the function value at a particular input \( x \), i.e. \( f(x) \), it comes out that Gaussian process is exactly what we needed. Then the joint distribution of \( \{f(x) : x \in \mathcal{X}\} \) is given by the law
\[
f(x) \sim \mathcal{N}(m(x), k(x, x'))
\]
for every \( x, x' \in \mathcal{X} \), where \( \mathcal{X} \) is the input space, and \( m(x) \) and \( k(x, x') \) are the mean function and covariance function of the process, respectively. Hence, we put a prior distribution on the functional parameter \( f \). A crucial property is that the joint distribution of any finite set of function values is still Gaussian. Indeed, we have a nonparametric regression model, but we always look at \( f \) at a finite number of input values. Moreover, Gaussian priors are conjugate in the GPR model, i.e. if we look at the posterior distribution, this is still Gaussian. In particular, the distribution of the vector of the target values \( Y \) given \( X \) and \( f \), will be Gaussian distributed.

Indeed, one of the main attraction of the Gaussian process framework is that it combines a consistent view with computational tractability \([1]\). For these reasons and others, such as flexibility and good performance of Gaussian processes, and what we have already said at the beginning, Gaussian process regression became an effective method for non-linear regression problems \([1, 13]\).

Moreover, Neal \([18]\) showed that many Bayesian regression models based on neural networks converge to Gaussian processes in the limit of an infinite network, and so Gaussian processes have been used as a replacement for supervised neural networks in non-linear regression and classification \([13, 17]\). In addition, empirical studies showed that Gaussian process regression have better predictive performance than other nonparametric models \(\text{\cite{Rasmussen1996}}\) such as Support Vector Machine (SVM) \([24, 25]\).

Gaussian process regression models rely on an appropriate selection of kernel, namely on the covariance function and the hyperparameters on which it depends. The covariance function encodes the assumptions we made about the function we wish to learn and it defines the similarity or nearness between datapoints \([1]\). Thus, the choice of the kernel has a big impact on the performance, and hence, predictive value of the GPR model.

Once the kernel, which controls the characteristics of the GPR model, has been chosen, the unknown hyperparameters on which it relies need to be determined. The common approach is to estimate them from the training data. Two methods can be adopted, the fully Bayes method which, through a hierarchical specification of models, gives prior
distribution to the hyperparameters (often called hyperprior) and the empirical Bayes (EB) method, which estimates the hyperparameters directly from the data. The first approach sees the implementation of Markov Chain Monte Carlo (MCMC) methods, that can perform the Gaussian process regression without the need of estimating hyperparameters [14, 17]. However, due to the high computational cost of MCMC methods, the hyperparameters are often estimated by means of maximum marginal likelihood [13, 18] (the EB approach), which considers the marginal likelihood as function of the hyperparameters and estimates them by maximizing it. Unfortunately, the marginal likelihood is not a convex function of the hyperparameters. A potential issue (which will form the focus of this thesis but received relatively little attention in the literature) is the existence of multiple local maxima. In fact, multiple local optima can exist [16] and the optimized hyperparameters may not be the global maxima [15, 17].

The main focus of this thesis will be the analytical and numerical study of the marginal likelihood of a specific Gaussian process regression model. We will focus our attention on a very simple case. Let us consider the regression model given at the beginning (0.1). Thus, we are going to put on $f$ a prior given by the distribution of a centered Brownian motion with scale parameter $\alpha$ and we are going to assume that $\epsilon \sim \mathcal{N}(0, e^{\beta})$. Therefore, the research topic will concern properties of the marginal likelihood as function of two hyperparameters (the error variance $\beta$ and the scale parameter $\alpha$ of the Gaussian process). These are estimated by means of empirical Bayes methods, i.e. by maximizing the marginal likelihood. Even in this very simple and restricted case, a major issue is given by the existence of multiple local maxima, and thus, the difficulty to find the global maximum. Therefore, we investigate the theoretical properties of the marginal likelihood and using the software Wolfram Mathematica 11.2 we run simulations to investigate the issues that may theoretically be intractable. We then find interesting results, such as the asymptotic behaviour of the marginal likelihood, and properties of its ridges, where it is more likely to find the local maxima. Moreover, simulations allow us to compute the estimated probability to have one or more local maxima and to locate the global maximum. Finally, and importantly, we find a good automatic and systematic method of finding a number of (and possibly all) the local maxima, through the implementation of the Conjugate Gradient optimisation algorithm. This study could provide help to researchers for finding the global maximum of the likelihood in order to retrieve the most useful information from the data.

This thesis is organized as follows. In the first chapter a theoretical framework for Gaussian process regression is presented, necessary for the next chapters. Basic definitions are given, such as the definition of a Gaussian process, and the main ingredients of this thesis are described. Gaussian process regression model is then characterized by its covariance function depending on the hyperparameters.

The main purpose of chapter 2 is to introduce the problem of estimating the kernel hyperparameters. Two approaches are presented and explained, the fully Bayes and
empirical Bayes methods. Then, the main problem and focus of this thesis is introduced, the existence of multiple local maxima of the marginal likelihood as function of the GPR hyperparameters. Finally, we look at some asymptotic properties of the empirical Bayes methods.

In the third chapter we present the GPR model that we are going to study. There will be a first introductory part, where some things already seen in the first chapter are recalled. The real aim of this chapter is to theoretically study the marginal likelihood as a function of the hyperparameters, deriving for instance, the asymptotic behaviour of the function, in order to understand where the local maxima of the function are and how many local maxima the function can have. The main results are explained in two propositions, and the proofs follow.

Chapter 4 covers the numerical study of the marginal likelihood. A brief explanation of the numerical methods implemented by the software Mathematica is given. Then, using the main results of the previous chapters, and in particular of chapter 3, a systematic method for finding the multiple local maxima of the marginal likelihood is given. Simulations are run in order to estimate the probability of having one or more local maxima, and to find the location of the global maximum. Finally, the experimental results are presented, and related problems are discussed.

We finish with the conclusion, where we briefly summarize what we saw in the previous chapters, the main results we accomplished, the main problems we faced and lastly we look at possible developments and open problems for further research.
1. Gaussian Process Regression

In this chapter we set up a theoretical framework for Gaussian process regression, necessary for the following chapters. The main concepts and ingredients will be presented. What is a Gaussian process is explained and the special regression model is given. Moreover, we are going to see how the Gaussian process regression model can be characterized and its main purpose, making prediction. Finally, we will briefly introduce what is the main goal of this thesis.

1.1. Gaussian processes

Since Gaussian processes are the main ingredient of the regression model, it is important to spend a section on them, explaining what a Gaussian process consists of. Hence, first of all we start with the definition of a stochastic process [10] and its finite-dimensional distributions [8].

Definition 1.1. Let \((\Omega, \mathcal{F}, \mathbb{P})\) be a probability space. A d-dimensional stochastic process indexed by \(I \subset [0, \infty)\) is a family of random variables \(X_t : \Omega \to \mathbb{R}^d\), \(t \in I\). We write \(X = (X_t)_{t \in I}\). \(I\) is called the index set and \(\mathbb{R}^d\) the state space of the process.

Definition 1.2. Let \((X_t)_{t \in T}\) be a real-valued stochastic process. The distributions of the finite-dimensional vectors of the form \((X_{t_1}, X_{t_2}, \ldots, X_{t_n})\), for \(t_1 < t_2 < \cdots < t_n\), are called the finite-dimensional distributions (fdd’s) of the process.

A Gaussian process is a type of stochastic process with specific properties, and it is defined as it follows.

Definition 1.3. A real-valued stochastic process is called Gaussian if all its fdd’s are multivariate normal distributions.

In particular, the definition of Gaussian process implies a consistency requirement, in other words, the distribution of a subset of variables is marginal to the distribution of a larger set of variables [1]. In fact, the Kolmogorov’s existence theorem guarantees that a suitably consistent collection of finite-dimensional distributions will define a stochastic process. Thus, let us give the definition of consistent fdd’s and the Kolmogorov’s existence theorem, both taken from [8].

Definition 1.4. Let \(T \subset \mathbb{R}\) and let \((E, \mathcal{E})\) be a measurable space. For all \(n \in \mathbb{Z}_+\) and all \(t_1 < t_2 < \cdots < t_n\), \(t_i \in T\), \(i = 1, \ldots, n\), let \(\mu_{t_1, \ldots, t_n}\) be a probability measure on \((E^n, \mathcal{E}^n)\). This collection of measures is called consistent if it has the property that

\[
\mu_{t_1, \ldots, t_{i-1}, t_{i+1}, \ldots, t_n} (A_1 \times \cdots \times A_{i-1} \times A_{i+1} \times \cdots \times A_n) = \\
\mu_{t_1, \ldots, t_n} (A_1 \times \cdots \times A_{i-1} \times E \times A_{i+1} \times \cdots \times A_n),
\]

10
for all $A_1, \ldots, A_{i-1}, A_{i+1}, \ldots, A_n \in \mathcal{E}$.

**Theorem 1.5** (Kolmogorov’s existence theorem). Suppose that $E$ is a Polish space and $\mathcal{E}$ its Borel-$\sigma$-algebra. Let $T = \mathbb{R}$ and for all $n \in \mathbb{Z}_+$ and all $t_1 < t_2 < \cdots < t_n \in T$, $t_i \in T$, $i = 1, \ldots, n$, let $\mu_{t_1, \ldots, t_n}$ be a probability measure on $(E^n, \mathcal{E}^n)$. If the measures $\mu_{t_1, \ldots, t_n}$ form a consistent system, then there exists a probability measure $P$ on $E^T$, such that the canonical (or co-ordinate variable) process $(X_t)_{t \in T}$ on $(\Omega = E^T, \mathcal{F} = \mathcal{E}^T, P)$, defined by

$$X(\omega) = \omega, \quad X_t(\omega) = \omega_t,$$

has fdd’s $\mu_{t_1, \ldots, t_n}$.

A detailed proof can be found e.g. in Billingsley 1995 [26].

Let us now suppose that we have a Gaussian process $X = (X_t)_{t \in T}$, thus we can define $m(t) = \mathbb{E}X_t$, $t \in T$, as the mean function of the process, and $k(s, t) = \text{cov}(X_s, X_t)$, $(s, t) \in T \times T$, as the covariance function of the process. Then, we have that a Gaussian process is completely specified by its mean function and covariance function, and we write $X \sim \mathcal{GP}(m(t), k(s, t))$. In general a Gaussian process $X$ with $\mathbb{E}X_t = 0$ for all $t \in T$ is called centered.

In particular, if a Gaussian process $(X_t)_{t \in T}$ is such that the mean function is $m(t) = 0$ and the covariance function is $r(s, t) = s \wedge t$, then $X$ is a Brownian motion. Thus, let us give another way to characterize this special stochastic process that we will see again in chapter 3.

**Definition 1.6.** The stochastic process $W = (W_t)_{t \in T}$, with $T = [0, \tau]$, is called a (standard) Brownian motion or Wiener process, if

1) $W_0 = 0$ a.s.;

2) $W$ is a stochastic process with stationary, independent increments, namely $W_t - W_s \sim W_{t-s} - W_0$ for $s \leq t \leq \tau$;

3) $W_t - W_s \sim \mathcal{N}(0, t-s)$;

4) almost all sample paths are continuous.

### 1.2. Gaussian process regression model

Now that we have introduced what is a Gaussian process we can start talking about the regression model. There are many ways to interpret Gaussian process regression models. One is the weight-space view, or parametric model, which is more familiar and accessible to many and the other is the function-space view, or non-parametric model, which is the one we are going to follow. Indeed, we can see a Gaussian process as a distribution over functions and the inference is taking place directly in the functions space. We are going to do this in a Bayesian framework [1].
Let us suppose we have a training set \( D = \{(x_i, y_i) : i = 1, \ldots, n\} \) where \( x_i \) are the input values and \( y_i \) the output or target values, such that they follow the regression model given by

\[
y_i = f(x_i) + \epsilon_i. \tag{1.1}
\]

We have assumed that the target values \( y_i \) differ from the function values, \( f(x_i) \), by a factor \( \epsilon_i \), the additive noise. Furthermore, we assume that this noise follows an independent, identically distributed Gaussian distribution with mean zero and variance \( \sigma_n^2 \),

\[
\epsilon_i \sim \mathcal{N}(0, \sigma_n^2). \tag{1.2}
\]

One of the main purposes is making inference about the relationship between inputs and targets, namely the conditional distribution of the targets given the inputs. We denote \( X = (x_1, \ldots, x_n) \) and \( Y = (y_1, \ldots, y_n) \), the vectors of input and target values, respectively. By the noise (1.2) and model (1.1) assumptions, we have that the following holds,

\[
p(Y | X, f) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma_n}} \exp \left( -\frac{(y_i - f(x_i))^2}{2\sigma_n^2} \right) 
= \frac{1}{(2\pi\sigma_n^2)^{n/2}} \exp \left( -\frac{1}{2\sigma_n^2} |Y - f|^2 \right) = \mathcal{N}(f, \sigma_n^2 I), \tag{1.3}
\]

where \( |z| \) denotes the Euclidean length of the vector \( z \) and \( I \) is the identity matrix.

Finally, we assume that for every \( x, x' \in \mathcal{X} \),

\[
f(x) \sim \mathcal{GP}(m(x), k(x, x')) \tag{1.5}
\]

which is the so-called prior distribution of \( f \). We see that this time the Gaussian process is given by the function values at each input \( x \in \mathcal{X} \), i.e. \( f(x) \) for \( x \in \mathcal{X} \). Hence, the stochastic process is no more defined over time, as in definition 1.1. In fact, here instead of having \( T \) as the index set, we now have \( \mathcal{X} \), the set of possible inputs. For notational convenience we denote \( f := f(X), f_i := f(x_i) \) and we suppose in (1.5) \( m(x) \) to be zero for every \( x \in \mathcal{X} \), even if it is not necessary. Note that using a zero mean function represents the lack of prior knowledge, and it does not mean that we expect the predictive function \( f \) distributing in equal way around zero, namely taking positive and negative values over equal parts of its range.

We can then compute the posterior distribution of \( f \), given the target values \( Y \) and the input values \( X \), which, by Bayes’s rule (A.4), writing only the terms of the likelihood (1.4) and the prior (1.2) which depend on \( f \), and completing the square, is given by

\[
p(f|X, Y) \propto \exp \left( -\frac{1}{2\sigma_n^2} (Y - f)^T (Y - f) \right) \exp \left( -f^T K^{-1} f \right)
\propto \exp \left( -\frac{1}{2} (f - \bar{f})^T \left( \frac{1}{\sigma_n^2} I + K^{-1} \right) (f - \bar{f}) \right),
\]
where $K$ is the $n \times n$ covariance matrix of $f$ of which the $(i,j)$-element is given by $K_{ij} = k(x_i, x_j)$, $\bar{f} = \frac{1}{\sigma_n^2} (\frac{1}{\sigma_n^2} I + K)^{-1} Y$, and we recognize the form of the posterior distribution as Gaussian with mean $\bar{f}$ and covariance matrix $A^{-1}$

$$p(f|X,Y) = \mathcal{N}(\bar{f} = \frac{1}{\sigma_n^2} A^{-1} Y, A^{-1}),$$

(1.6)

where $A = \frac{1}{\sigma_n^2} I + K^{-1}$.

1.3. Prediction

Even if prediction is not the main purpose of this thesis, for completeness it is important to briefly see it.

Again, let us suppose we have a dataset $D$ of $n$ observations, $D = \{(x_i, y_i) : i = 1, \ldots, n\}$. Given the training data, let us suppose we have new input values $X^*$, such that $X^* \not\in D$. Thus, we wish to make predictions for $X^*$. Let us denote $f(X^*) := f^*$. To predict the function values $f^*$, the joint probability distribution of the observed target values $Y$ and predictive targets $f^*$ is given by

$$\begin{bmatrix} Y \\ f^* \end{bmatrix} \sim \mathcal{N} \left( 0, \begin{bmatrix} K(X,X) + \sigma_n^2 I & K(X,X^*) \\ K(X,X^*) & K(X^*,X^*) \end{bmatrix} \right).$$

Indeed, it follows from (1.5) where we assumed $m(x) = 0$ for every $x \in X$, and from the fact that $Y \sim \mathcal{N}(0, K(X,X) + \sigma_n^2 I)$, since it is the sum of two multivariate normal random vectors, $f$ and $\epsilon$. From this we can derive the conditional distribution of $f^*$, the key predictive equation for Gaussian process regression, given by (see A.15)

$$f^*|X,Y,X^* \sim \mathcal{N}(\bar{f}^*, \text{cov}(f^*)), \quad (1.7)$$

where

$$\bar{f}^* = \mathbb{E}[f^*|X,Y,X^*] = K(X^*,X)[K(X,X) + \sigma_n^2 I]^{-1} Y; \quad (1.8)$$

$$\text{cov}(f^*) = K(X^*,X^*) - K(X^*,X)[K(X,X) + \sigma_n^2 I]^{-1} K(X,X^*). \quad (1.9)$$

We see that (1.8), the predictive mean, is a linear combination of the output observed values $Y$, that is why sometimes is called linear predictor. By contrast, (1.9), the predictive variance, does not depend on $Y$ but only on the input values $X$ and $X^*$. Moreover, we can notice that (1.9) is the difference between $K(X^*,X^*)$ which is simply the prior covariance, and $K(X^*,X)[K(X,X) + \sigma_n^2 I]^{-1} K(X,X^*)$, representing the information the observations give us about the function $f^*$.

1.4. Covariance functions

The covariance function (or kernel), $k(\cdot, \cdot)$, plays a crucial role in Gaussian process regression. So far we did not specify any particular covariance function or matrix. However,
from equations (1.8) and (1.9) we can see how crucial the choice of $k$ in both predictive mean and variance is. Indeed, according to [1], it encodes the assumptions we made about the function we wish to learn and it defines the similarity or nearness between datapoints. In fact, close datapoints $X$ are more likely to have the same or closer target points $Y$, and so training points close to a test point $x_*$ become informative about $f_*$, the prediction at that point. In particular, in Gaussian process regression what describes and defines this closeness between the training points is the covariance function. As a result the choice of the covariance function has an important impact on the prediction.

Kernel covariance functions are defined as $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$. Anyway, not any arbitrary function of two input points $x, x' \in \mathcal{X}$ can serve as a covariance function. First of all, a function of two arguments, mapping a pair of inputs $x, x' \in \mathcal{X}$ into $\mathbb{R}$, is called kernel. In particular, a real kernel is said to be symmetric if $k(x, x') = k(x', x)$. Then, given the set of inputs $\mathcal{X} = \{x_1, \ldots, x_n\}$, we can define the matrix whose entries are defined such that $K_{ij} = k(x_i, x_j)$. If $k$ is a covariance function we call the matrix $K$ covariance matrix, and it is clear that in this case, by definition, $k$ is symmetric. Moreover, $K$ is called positive semidefinite (PSD) if for every vector $v \in \mathbb{R}^n$ $v^TKv \geq 0$, and if a matrix is symmetric then it is PSD if and only if all its eigenvalues are non-negative. Since $K$ is a covariance matrix, then it must be positive semidefinite. In the same way, we say that the kernel $k$ is positive semidefinite if

$$\int k(x, x') f(x) f(x') d\mu(x') \geq 0,$$

for all $f \in L_2(\mathcal{X}, \mu)$, where $\mu$ denotes a measure.

We are now going to list some of the most important and used kernel functions, taken from [1]. However, you will see that in Chapter 3 we will not use any of them, since we will restrict our attention to a simpler case.

The squared exponential. The squared exponential kernel is defined as

$$k_{SE}(x, x') = \exp\left(-\frac{(x - x')^2}{2l^2}\right), \quad (1.10)$$

with parameter $l$ defining the characteristic length-scale. This covariance function is infinitely differentiable, which means that the Gaussian process with this covariance function is very smooth. Moreover, it is the most widely-used kernel within the kernel machine field [1].

The Matérn class. The Matérn class of covariance functions is given by

$$k_{Matern}(x, x') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu(x - x')^2}}{l}\right)^\nu K_\nu \left(\frac{\sqrt{2\nu(x - x')^2}}{l}\right), \quad (1.11)$$

with positive parameters $\nu$ and $l$, where $K_\nu$ is a modified Bessel function [1].
The $\gamma$-exponential. The $\gamma$-exponential kernel is given by

$$k(x, x') = \exp\left(-\left((x - x')^2/l\right)^\gamma\right) \text{ for } 0 < \gamma \leq 2,$$

(1.12)

with $l$ again a positive parameter.

The Rational quadratic. The rational quadratic covariance function is defined as

$$k_{QR}(x, x') = \left(1 + \frac{(x - x')^4}{2\alpha l^2}\right)^{-\alpha},$$

(1.13)

with $\alpha, l > 0$.

Periodic. The periodic covariance function is especially used to model functions which show a periodic behaviour, and it is given by

$$k_{PER}(x, x') = \exp\left(-\frac{2 \sin^2\left(\frac{\pi(x - x')}{p}\right)}{l^2}\right),$$

(1.14)

where $p$ is the period of the function and $l$ has the same meaning as for the squared exponential (SE) kernel.

There are several other covariance functions, like linear, polynomial, dot-product, and of course constant, but it is not of our interest in this thesis.

Instead, we can look at the free parameters of the covariance function, e.g. the characteristic length-scale $l$ in (1.10). In general we call them hyperparameters, since they are parameters of the prior distribution (rather than the model distribution), but it will be clearer later (chapter 2). Such hyperparameters can control "the amount of noise in a regression model, the scale of variation in a regression function, the degree to which various input variables are relevant, and the magnitudes of different additive components of a model" [14].

Gaussian process regression (GPR) hyperparameters can vary from two or three for a very simple regression model up to several dozen or more for a model with many inputs and, in particular, they can influence the predictive value of the model [13]. There is plenty of scope for variation even inside only one family of covariance functions. However, more than exploring the effect of varying the GPR hyperparameters and see how these affect the predictability of the model, we are interested in the estimate of the GPR hyperparameters. Indeed, in the next chapter we will consider different methods for determining the hyperparameters from the training data.
2. Estimating GPR hyperparameters

In chapter 1 we have seen how to do regression using Gaussian processes with a given covariance function. In particular, we have considered some of the most important and used kernels, many of which depend on several hyperparameters whose values need to be determined. Since covariance functions control properties of prediction function, such as smoothness, rate of variability, periodicity, we see how important it is to select and properly determine the parameters.

In this chapter we are going to introduce two different approaches to estimation of the GPR hyperparameters from the training data: the fully Bayes and empirical Bayes methods. In particular, in chapters 3 and 4, we will use the second approach, that is, by maximizing the marginal likelihood function. Thus, we will explain why one is preferred over the other. Moreover, we will describe one of the major possible problems that one can encounter, the existence of multiple local maxima, the main focus of this thesis. Finally, we will present some asymptotic properties of the empirical Bayes methods.

2.1. Two different approaches

In Gaussian process regression models the parameters need to be estimated from the training data. We can distinguish two different approaches. The fully Bayes approach, which uses a hierarchical specification of priors, giving prior distributions to the hyperparameters and the empirical Bayes (EB) approach, which estimates the prior from the data [11]. The EB method uses the maximum likelihood for estimating the hyperparameters [18], namely the marginal likelihood as function of the hyperparameters is maximized using optimization algorithms, such as the Newton or Conjugate Gradient algorithm. In general, in the fully Bayes methods, predictions are made by averaging the posterior distribution for the hyperparameters, implementing e.g. Markov Chain Monte Carlo (MCMC) methods, that can perform without the need of estimating hyperparameters [14, 17] or, due to the high computational cost of MCMC methods, it estimates the GPR hyperparameters by means of maximum marginal likelihood [13].

However, "common practice suggests that a fully Bayesian approach may not be worthwhile" [15]. There are several reasons sampling kernel hyperparameters is not yet popular, as empirical Bayes approach actually is. Some of them (taken from [15]) include:

- Historically, tuning (kernel) hyperparameters has not received much attention, even if they can significantly affect the performance and predictive value of the GPR model. Therefore, the literature is relatively "poor" regarding this topic.
This is partly due to the fact that the more well-known SVM framework only provides heuristics for kernel hyperparameters tuning, and these are not really studied and discussed in paper writing.

- Sampling methods are not typically as computationally efficient to implement as optimization. Moreover, it is not clear which sampling method should be implemented for kernel hyperparameters, even if MCMC and slice sampling are two common methods.

- The marginal likelihood as function of the kernel hyperparameters is often peaked for the kernels that are most widely used, such as the squared exponential (SE) kernel. Rasmussen (1996) [19], through some experiments using the SE kernel, found no strong empirical advantages to sampling over marginal likelihood optimization.

However, there are areas, like Bayesian Optimization (see e.g. [28]) where sampling kernel hyperparameters has become really popular. Nonetheless, given also the previous reasons, we decided to adopt the empirical Bayes approach in our study, as you will see later in chapters 3 and 4.

Let us now see the two different approaches more in detail, the discussion will be general but focused on the specific treatment of Gaussian process models for regression. Here we refer to the GPR hyperparameters as the covariance function parameters, however, it is possible to incorporate and treat in the same way the parameters of the noise distribution (e.g. $\sigma^2_n$ in the regression model (1.1) where the additive noise has normal distribution, $\epsilon \sim \mathcal{N}(0, \sigma^2_n)$).

### 2.1.1. The fully Bayes approach

As we said before, in the fully Bayes approach it is common to use a hierarchical specification of priors. At the lowest level is the parameter $f$ (the prediction function can be seen as a parameter), and at the second level are the hyperparameters $\theta$ which control the distribution of $f$.

Inference takes place one level at a time, on the posterior distribution. At the bottom level, by Bayes’s rule (A.4) the posterior distribution is given by

$$p(f|Y, X, \theta) = \frac{p(Y|X, f)p(f|\theta)}{p(Y|X, \theta)},$$

where $p(f|\theta)$ is the prior distribution of $f$ and $p(Y|X, f)$ the likelihood function. The prior, as a probability distribution, represents the knowledge of the parameter $f$ that we have before seeing the data. The posterior combines information from the prior and the data (through the likelihood). The probability density $p(Y|X, \theta)$ is the normalizing constant and we see that it does not depend on the parameter $f$, and it is given by

$$p(Y|X, \theta) = \int p(Y|X, f)dp(f|\theta).$$
At the second level, in the same way, the posterior distribution of the hyperparameters \( \theta \) is given by

\[
p(\theta | Y, X) = \frac{p(Y | X, \theta) p(\theta)}{p(Y | X)},
\]

where \( p(\theta) \) is the prior distribution of the hyperparameters, often called hyperprior. The normalizing constant is given by

\[
p(Y | X) = \int p(Y | X, \theta) p(\theta) d\theta.
\]

(2.3)

(2.4)

We see that the use of a fully Bayes approach implies the implementation of several integrals. Depending on the model, some of them may not be analytically tractable and therefore, they could require analytical approximations or MCMC methods. Moreover, the behaviour of Gaussian process regression models is extremely sensitive to the choice of the kernel, and therefore, it is important to properly choose it and determine the hyperparameters. By contrast, overall prior distributions of the hyperparameters have little impact on the performance of GPR models, which implies that simple priors, like e.g. the uniform distribution in an appropriate interval, may be sufficient in GPR modelling in terms of predictability [13].

2.1.2. The empirical Bayes approach

Bayesian analysis, in addition to the likelihood function, depends on prior distributions for the model parameters. This prior, which in our case is the prior of the predictive function \( f \), can be parametric or nonparametric, depending on other parameters which can be drawn from other second-level prior. This sequence of parameters and prior distributions constitutes a hierarchical model. The hierarchy stops at some point, assuming all the left prior parameters known. The empirical Bayes approach, rather than assuming this, uses the observed data to estimate these final stage parameters, proceeding as if the prior distributions were known (for more details see [2]).

In particular, in Gaussian process regression, once a prior distribution for \( f \) has been chosen, and so its covariance function, the hyperparameters of this, \( \theta \), will not be given any prior distribution. The same will be done for the parameter of the additive noise \( \epsilon \), i.e. \( \sigma^2_n \), it will not be given any prior distribution. Thus, the estimation of the hyperparameters proceeds as follows. The common approach is to estimate them by means of maximum marginal likelihood, namely find the maximum of the marginal likelihood as function of the hyperparameters \( \theta \). Recall the model (1.1), and the assumptions we made in chapter 1,

\[
y_i = f(x_i) + \epsilon_i, \text{ for every } i = 1, \ldots, n,
\]

(2.5)

where the additive noise \( \epsilon \) is such that

\[
\epsilon_i \sim N(0, \sigma^2_n),
\]

(2.6)
and the joint distribution of \( \{f(x) : x \in \mathcal{X}\} \) is given by the law
\[
f(x) \sim \mathcal{N}(m(x), k(x, x')),
\]
for every \( x, x' \in \mathcal{X} \). Hence, if we then look at the vectors \( \epsilon \) and \( f \), and suppose that \( m(x) = 0 \) for every \( x \in \mathcal{X} \), we have
\[
\epsilon \sim \mathcal{N}(0, \sigma_n^2 I) \text{ and } f \sim \mathcal{N}(0, K),
\]
where \( I \) is the \( n \times n \) identity matrix and \( K \) the covariance function of the Gaussian process \( f \), such that each matrix entries is given by \( K_{ij} = k(x_i, x_j) \), for every \( x_i, x_j \in \mathcal{X} \). Therefore, from (2.5) and (2.8), the distribution of the outputs \( Y \) is given by
\[
p(Y | X, \theta) \sim \mathcal{N}(0, \Sigma_{\theta}),
\]
where \( \Sigma_{\theta} \) is the covariance function depending on the unknown hypeparameters \( \theta \) and given by \( \Sigma_{\theta} = K + \sigma_n^2 I \). Thus, the log marginal likelihood is given by,
\[
\log \mathcal{L}(\theta) = -\frac{n}{2} \log 2\pi - \frac{1}{2} \log |\Sigma_{\theta}| - \frac{1}{2} Y^T \Sigma_{\theta}^{-1} Y,
\]
where we denote with \( \mathcal{L}(\cdot) \) the marginal likelihood as function of the hyperparameters \( \theta \). We can see three different terms in (2.10). The only one which depends on the observed targets \( Y \) is given by \( -\frac{1}{2} Y^T \Sigma_{\theta}^{-1} Y \), \( -\frac{1}{2} \log |\Sigma_{\theta}| \) is the complexity penalty depending only on the covariance function \( \Sigma_{\theta} \) and the inputs \( X \), and finally \( -\frac{n}{2} \log 2\pi \) is a normalization constant [1]. We then estimate the hyperparameters setting the derivatives of (2.10) with respect to \( \theta \).

Using (A.11) and (A.12), we get
\[
\frac{\partial}{\partial \theta_i} \log \mathcal{L}(\theta) = \frac{1}{2} Y^T K^{-1} \frac{\partial}{\partial \theta_i} K^{-1} Y - \frac{1}{2} \text{Tr} \left( K^{-1} \frac{\partial}{\partial \theta_i} K \right).
\]
19

The complexity of computing the marginal likelihood (2.10) is mainly due to the inverse matrix \( K^{-1} \). Standard methods for matrix inversion of positive definitive symmetric matrices, as \( K \), require time \( \mathcal{O}(n^3) \) for a \( n \times n \) matrix. Once \( K^{-1} \) is known, the computation of the derivatives in (2.11) requires time \( \mathcal{O}(n^2) \) per hyperparameter (see [1]).

If the marginal likelihood is not analytically tractable, then estimating the hyperparameters can become difficult. Perhaps the most successful approach has been to "approximate the marginal likelihood by the use of e.g. Laplace, EP or variational approximations and then optimize the approximate marginal likelihood with respect to the hyperparameters" [15]. Another way could be the implementation of the Expectation Maximization (EM) algorithm, which is an ascent method for maximizing the likelihood, but it is only guaranteed to converge to a stationary point of the likelihood function (see [2, 20]).
2.2. Existence of multiple local maxima

The marginal likelihood can be multimodal as a function of the hyperparameters. In particular, the choice of the covariance function can influence the convexity of the marginal likelihood as well. Thus, there are cases where the likelihood is not convex with respect to the hyperparameters and therefore the optimisation algorithm may converge to a local optimum whereas the global one could provide better results [16]. In particular, the optimized hyperparameters may not be the global optima [15, 17]. In fact there is no guarantee that the likelihood function does not suffer of multiple local optima. A possible way to tackle this issue then could be the use of multiple starting points randomly selected from a specific prior distribution and after convergence, choose the maximum points with higher marginal likelihood function value as the estimates [13], or, for example in data with periodic structure, all the parameters which were part of the previous kernel are initialized to their previous values [16]. Anyway, care should be taken not to end up with a bad local maximum.

Therefore, we have seen that in general while estimating the GPR hyperparameters through optimisation of the marginal likelihood, the existence of local optima represents a possible major problem which actually received very little attention in the literature. Thus, what we are going to do next is to consider a special case of Gaussian process regression model, where a covariance function which depends only on one parameter is taken. We will show that even in this very simple case the problem of local optima can occur, and our final goal is to find a systematic method to find all the local optima of the marginal likelihood and in particular the global one.

Finally, we can see, figure 2.1, taken from [1], shows an example of the marginal likelihood as function of the hyperparameters with two local maxima. Every local maximum corresponds to a particular interpretation of the data [1]. One optimum corresponds to a relatively complicated model with low noise, whereas the other corresponds to a much simpler model with more noise. Figure (a) shows the marginal likelihood as a function of the hyperparameters \( l \), the characteristic length-scale, and \( \sigma^2_n \), the variance of the additive noise \( \epsilon \), in the GPR model with SE kernel given by (1.10), for a dataset of 7 observations. We see that there are two local optima indicated by ‘+’. The global optimum corresponds to low noise and short length-scale, while the local optimum has high noise and long length-scale. In (b) and (c) the inferred underlying functions (with 95% confidence bands) are shown for each of the two solutions [1]. We can think of the first figure in (b) as a overfitting model, indeed we only have 7 observations and 2 hyperparameters. Here residuals are almost zero, explaining also the noise. While in (c) there seems to be misspecification. The residuals are large, not explaining enough.
Figure 2.1.
2.3. Asymptotic behaviour of the empirical Bayes methods

After having described the empirical Bayes and fully Bayes approaches, a natural question that comes along is whether nonparametric Bayesian procedures "work". In our case, we are now interested in looking at the asymptotic behaviour of the empirical Bayes methods. The discussion will be general, but the adaptation to our case easily follows.

Let us suppose we have observations $X_n$ that are distributed accordingly to $P^n_{\theta}$, conditionally to some parameter $\theta \in \Theta$ (in the GPR model the parameter would be the predictive function $f$). Let $\{\Pi(\cdot|\lambda), \lambda \in \Lambda\}$ be a family of prior distributions on $\Theta$, with $\lambda$ hyperparameter (in the GPR model the family of prior distributions is the Gaussian process and $\lambda$ are the GPR hyperparameters). We adopt the "frequentist" view, namely we assume that the observations are generated accordingly to a true parameter, say $\theta_0$, and we want to see if increasing the number of observations indefinitely, the posterior distribution is able to recover the parameter. We then need two definitions, i.e. *posterior consistency* and *rate of contraction* (both taken from [11]).

**Definition 2.1.** Let $(\Theta, d)$ be a semimetric space. The posterior distribution $\Pi(\cdot|X_n, \lambda)$ is said to be (weakly) consistent at $\theta_0 \in \Theta$ if $\Pi(\theta : d(\theta, \theta_0) > \epsilon|X_n) \to 0$ in $P_{\theta_0}$-probability, as $n \to \infty$, for every $\epsilon > 0$. The posterior is said to be strongly consistent at $\theta_0 \in \Theta$ if the convergence is in the almost sure sense.

**Definition 2.2.** Let $(\Theta, d)$ be a semimetric space. The posterior distribution $\Pi(\cdot|X_n, \lambda)$ is said to contract at rate $\epsilon_n \to 0$ at $\theta_0 \in \Theta$ if $\Pi(\theta : d(\theta, \theta_0) > M_n\epsilon_n|X_n) \to 0$ in $P_{\theta_0}$-probability, for every $M_n \to \infty$ as $n \to \infty$.

This last definition roughly means that the posterior distribution concentrates around $\theta_0$ on balls of radius of the order $\epsilon_n$, and the contraction rate can be seen as a natural refinement of consistency [11].

If the model is parametric, it has been shown that the maximum marginal likelihood estimator (MMLE) converges asymptotically to some oracle value $\lambda_0$ which maximizes in $\lambda$ the prior density calculated at the true value $\theta_0$ of the parameter, $\pi(\theta_0|\lambda_0) = \sup\{\pi(\theta_0|\lambda) : \lambda \in \Lambda\}$, where the density is with respect to Lebesgue measure [30]. However, it is not possible to extend this to nonparametric model, since for instance, the prior distributions $\Pi(\cdot|\lambda)$, $\lambda \in \Lambda$ typically are not absolutely continuous with respect to a fixed measure [29]. In [22] a special case has been studied. They focus on a particular case of the Gaussian white noise model and they consider an infinite dimensional Gaussian prior with fixed regularity parameter $\alpha$ and scaling hyperparameter $\tau$. They find out that for a given base prior and a given true regularity level, there exists an optimal scaling rate. However, EB methods fail to follow the true value if the regularity of the true parameter exceeds an even lower band. On the other side, it comes out that EB approach "works adequately if the base prior does not undersmooth the true parameter" [22]. More general studies are conducted in [29]. They study the asymptotic behaviour of the MMLE and they derive posterior contraction rates. Previously [31] provided sufficient conditions for deriving general EB posterior contraction rates when it is known...
that the MMLE belongs to a chosen subset $\Lambda_0$ of $\Lambda$. Their result was given by basically controlling

$$\sup_{\lambda \in \Lambda_0} \Pi(d(\theta, \theta_0) > \epsilon_n | X_n, \lambda).$$

Then in [29], two further important results for the asymptotic properties of the EB methods are given:

- they characterize $\Lambda_0$ as

$$\Lambda_0 = \{ \lambda : \epsilon_n(\lambda) \leq M_n \epsilon_n,0 \}$$

for any sequence $M_n$ going to $\infty$, with $\epsilon_{n,0} = \inf\{\epsilon_n(\lambda) : \lambda \in \Lambda\}$ and $\epsilon_n$ satisfying

$$\Pi(\|\theta - \theta_0\| \leq K \epsilon_n(\lambda) | \lambda) = e^{-n \epsilon_n^2(\lambda)},$$

with $(\Theta, \| \cdot \|)$ Banach space and for some large enough constant $K$;

- they reveal the exact posterior contraction rates for every $\theta_0 \in \Theta$. Namely, they prove that the concentration rate of the MMLE empirical Bayes posterior distribution is of order $O(M_n \epsilon_n,0)$, and they show that the posterior contraction rate is bounded from below by $\delta_n \epsilon_n,0$, for arbitrary $\delta_n = o(1)$.

Finally, and interestingly, comparing the fully Bayes approach that we have presented at the beginning of this chapter and the EB approach, they showed that empirical Bayes and fully Bayes methods behave similarly. Indeed, fully Bayes posterior has the same upper and lower bounds on the contraction rate for every $\theta_0 \in \Theta$.

Let us now go back to the Gaussian process regression model we have discussed in chapter 1 and the problem of the existence of multiple local maxima. Here the family of prior distributions is given by the Gaussian processes with a specific covariance function and the hyperparameters are now given by $\theta$ instead of $\lambda$. Then, we see that under certain conditions, for instance $\theta \in \Lambda_0$ for specific $M_n$ and $\epsilon_{n,0}$, as $n$ number of the observations grows indefinitely, the posterior distribution will concentrate close around the global maximum, with contraction rates of the order $O(M_n \epsilon_n,0)$, and bounded from below by $\delta_n \epsilon_n,0$, for arbitrary $\delta_n = o(1)$. In particular, as in the case of figure 2.1, if the marginal likelihood, as function of the GPR hyperparameters, admits two maxima (one local and one global maximum), we have that as $n$ goes to $\infty$, there will still exist two local maxima but the posterior distribution will concentrate around the global maximum, and as we can see the local maximum brings to a misspecified model while the global maximum to a overfitted model.
3. Regression with a centered Brownian motion prior

So far we have introduced the general Gaussian Process regression model, how it can be characterized and how it depends on the hyperparameters. In particular, we have seen that the Gaussian process predictor $f(x)$ relies on the covariance function, which encodes our assumptions about the function we wish to learn and which depends on free parameters, called GPR hyperparameters, that we want to estimate. We then have looked at different approaches with which we can estimate them, and finally we have introduced one of the major possible issues, the main focus of this thesis, the existence of multiple local maxima in the EB maximization problem.

We now restrict our attention to a specific model, with a specific covariance function and we are going to study this more in detail. Indeed, the kernel we chose is exactly the kernel of a centered Brownian motion. The reason why we did it is to make the Gaussian process regression model as simplest as possible (this kernel does not depend on any other parameters). Then, the chapter will be focused on the theoretical, and so analytical, study of the posterior density as function of the hyperparameters. There is going to be a first introductory part, where many things will be repeated from chapter 1, followed by the main results, and their proofs.

3.1. Main results

Let us consider the following model,

$$ Y = \mu + f(X) + \epsilon, $$

where $X$ is the input vector, $\mu$ is a constant, $f$ is the function value and $Y$ is the vector of the observed target values. We have assumed that the observed values $Y$ differ from the function values $f(X)$ by a constant $\mu$ and additive noise $\epsilon$, and we will further assume that this noise follows an independent, identically distributed Gaussian distribution, with zero mean and variance $\sigma^2$

$$ \epsilon_i \sim \mathcal{N}(0, \sigma^2), \text{ for } i = 1, \ldots, n. $$

Moreover, we need a restriction on $f$, namely $\mathbb{E}f(x) = 0$ for every $x \in X$. Hence, we assume that $f$ has Brownian motion prior distribution, with scale parameter $\alpha$, given by

$$ f \sim \mathcal{N}(0, e^{\alpha}H), $$

24
where \( f := f(X) \) and \( H \) is the covariance kernel matrix of a centered Brownian motion.

The noise assumption (3.2) together with the model (3.1) gives rise to the probability density function, which is factored over cases in the training set (because of the independence assumption) to give

\[
p(Y|X,f) = \prod_{i=1}^{n} p(y_i|x_i, f(x_i)) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi e^\beta}} \exp\left( -\frac{(y_i - (f(x_i) + \mu))^2}{2e^\beta} \right)
\]

(3.4)


Furthermore, by Bayes’ rule (A.4) we have

\[
p(f|X,Y) = \frac{p(Y|f,X)p(f)}{p(Y|X)},
\]

(3.6)

where the normalizing constant is independent of \( f \) and given by

\[
p(Y|X) = \int p(Y|X,f)dp(f).
\]

In particular we have

\[
p(f|X,Y) \propto p(Y|f,X)p(f).
\]

(3.7)

Hence, writing only the terms which depend on \( f \), and ”completing” the square, we obtain from (3.7)

\[
p(f|X,Y) \propto \exp\left( -\frac{1}{2e^\beta} (Y - (\mu + f))^T (Y - (\mu + f)) \right) \exp\left( -\frac{1}{e^\alpha} f^T H^{-1} f \right)
\]

\[
\propto \exp\left( -\frac{1}{2} (f - \tilde{f})^T \left( \frac{1}{e^\beta} I + \frac{1}{e^\alpha} H^{-1} \right) (f - \tilde{f}) \right),
\]

where \( \tilde{f} = e^{-\beta}(e^{-\beta}I + e^{-\alpha}H)^{-1}(Y - \mu) \), and we recognize the form of the posterior distribution as Gaussian with mean \( \tilde{f} \) and covariance matrix \( A^{-1} \)

\[
p(f|X,Y) = \mathcal{N}(\tilde{f} = \frac{1}{e^\beta}A^{-1}(Y - \mu), A^{-1}),
\]

(3.8)

where \( A = e^{-\beta}I + e^{-\alpha}H^{-1} \).

We see that (3.8) depends on the parameters \( \alpha, \beta \) and \( \mu \). We can estimate them by maximizing the marginal log-likelihood function, namely with their maximum likelihood estimators (MLE). If for \( \mu \) it is quite easy to find it (it is given by \( \tilde{Y} \)), for the other two parameters it gets more difficult.

Let us suppose to have a dataset of \( n \) observations, \( D = \{(x_i, y_i) : i = 1, \ldots, n\} \). We consider again the following model

\[
y_i = \mu + f(x_i) + \epsilon_i,
\]

(3.9)
with same assumptions for \( f \) and \( \epsilon \) that we made before. Thus, we have that

\[
(y_1, ..., y_n) \sim \mathcal{N}(\mu, K_{h,\alpha,\beta})
\]  

(3.10)

has a multivariate normal distribution with mean \( \mu \) and covariance matrix given by \( K_{h,\alpha,\beta} = e^\alpha H + e^\beta I \), where \( H \) is the covariance kernel matrix of a centered Brownian motion, \( I \) is the identity matrix and \( \alpha, \beta \) are scalar parameters. Let us replace \( \mu \) by its MLE \( \hat{\mu} \), namely \( \hat{\mu} = \bar{Y} = \frac{1}{n} \sum_{i=1}^{n} y_i \). What we are going to do next is considering the profile likelihood

\[
\mathcal{L}(\alpha, \beta) := \mathcal{L}(\alpha, \beta, \hat{\mu})
\]

as a function of \( \alpha \) and \( \beta \), and now the main goal is to find the local maxima of the profile log-likelihood, \( \log \mathcal{L}(\alpha, \beta) \).

If we consider as kernel \( h \) of the centered Brownian motion,

\[
h(x, x') = -\frac{1}{2} \left( \| x - x' \| - \frac{1}{n} \sum_j \| x - x_j \| - \frac{1}{n} \sum_i \| x_i - x' \| + \frac{1}{n^2} \sum_i \sum_j \| x_i - x_j \| \right)
\]  

(3.11)

we get the following plot of the profile log-likelihood \( \log \mathcal{L}(\alpha, \beta) \).

![Profile log-likelihood plot](image)

Figure 3.1.: Random log-likelihood for \( n=5 \)

As we can see, figure 3.1 shows the function having two ridges and it looks like the tops have straight line asymptotes. It would be then interesting for our main purpose to obtain their equations. Note that we are going to use this kernel (3.11) for chapters 3 and 4.

Let us consider again the model (3.9). Suppose we have \( n \) observations. Hence, we get

\[
\mathcal{L}(\alpha, \beta) = \frac{1}{\sqrt{(2\pi)^n |K_{h,\alpha,\beta}|}} \exp \left( -\frac{1}{2} (Y - \bar{Y})^T (K_{h,\alpha,\beta})^{-1} (Y - \bar{Y}) \right)
\]  

(3.12)
and, using equations (A.6), (A.8) and (A.10),

\[
\log \mathcal{L}(\alpha, \beta) = -\frac{n}{2} \log 2\pi - \frac{1}{2} \log |K^{-1}_{h,\alpha,\beta}| - \frac{1}{2} y^T K^{-1}_{h,\alpha,\beta} y^* \tag{3.13}
\]

\[
= -\frac{n}{2} \log 2\pi - \frac{1}{2} \sum_i \log(e^\alpha \lambda_i + e^\beta) - \frac{1}{2} \sum_i \left(\frac{a_i^T y^*}{e^\alpha \lambda_i + e^\beta}\right) \tag{3.14}
\]

where again \(K_{h,\alpha,\beta} = e^{\alpha H} + e^{\beta I}\), \(Y = (y_1, \ldots, y_n)\), \(y^*\) is the vector defined as \(y^* = (y_1 - \bar{Y}, \ldots, y_n - \bar{Y})\), \(\lambda_i\) and \(a_i\) are the eigenvalues and eigenvectors, respectively, of \(H\), (see B).

As we said before, the main goal is to find, if existing, the local maxima and the global maximum of (3.13). Thus, the first thing we should check is the existence of stationary points. Let us consider the general case \(n > 2\) (we will see later the case \(n = 2\) ). By (A.11) and (A.12), we get

\[
\frac{\partial}{\partial \alpha} \log \mathcal{L}(\alpha, \beta) = -\frac{1}{2} \frac{\partial}{\partial \alpha} \left(|K_{h,\alpha,\beta}|\right) - \frac{1}{2} \frac{\partial}{\partial \alpha} \left(y^T K^{-1}_{h,\alpha,\beta} y^*\right)
\]

\[
= -\frac{1}{2} \text{Tr} \left(K^{-1}_{h,\alpha,\beta} \frac{\partial}{\partial \alpha} K_{h,\alpha,\beta}\right) - \frac{1}{2} y^* T \left(-K^{-1}_{h,\alpha,\beta} \frac{\partial K_{h,\alpha,\beta}}{\partial \alpha} K^{-1}_{h,\alpha,\beta}\right) y^* \tag{3.15}
\]

and

\[
\frac{\partial}{\partial \beta} \log \mathcal{L}(\alpha, \beta) = -\frac{1}{2} \frac{\partial}{\partial \beta} \left(|K_{h,\alpha,\beta}|\right) - \frac{1}{2} \frac{\partial}{\partial \beta} \left(y^T K^{-1}_{h,\alpha,\beta} y^*\right)
\]

\[
= -\frac{1}{2} \text{Tr} \left(K^{-1}_{h,\alpha,\beta} \frac{\partial}{\partial \beta} K_{h,\alpha,\beta}\right) - \frac{1}{2} y^* T \left(-K^{-1}_{h,\alpha,\beta} \frac{\partial K_{h,\alpha,\beta}}{\partial \beta} K^{-1}_{h,\alpha,\beta}\right) y^*.
\]

Furthermore, we have

\[
\frac{\partial}{\partial \alpha} K_{h,\alpha,\beta} = \frac{\partial}{\partial \alpha} (e^{\alpha H} + e^{\beta I}) = e^{\alpha H} = \sum_i e^{\alpha \lambda_i} a_i a_i^T,
\]

and in particular

\[
K^{-1}_{h,\alpha,\beta} \frac{\partial}{\partial \alpha} K_{h,\alpha,\beta} = \sum_i \frac{e^{\alpha \lambda_i}}{e^{\alpha \lambda_i} + e^\beta}.
\]

Thus,

\[
\text{Tr} \left(K^{-1}_{h,\alpha,\beta} \frac{\partial}{\partial \alpha} K_{h,\alpha,\beta}\right) = \sum_i \left(\frac{e^{\alpha \lambda_i}}{e^{\alpha \lambda_i} + e^\beta}\right) \tag{3.15}
\]

and

\[
-K^{-1}_{h,\alpha,\beta} \frac{\partial K_{h,\alpha,\beta}}{\partial \alpha} K^{-1}_{h,\alpha,\beta} = -\sum_i \frac{e^{\alpha \lambda_i}}{(e^{\alpha \lambda_i} + e^\beta)^2} a_i a_i^T. \tag{3.16}
\]
Therefore, from (3.15) and (3.16) we get

\[
\frac{\partial}{\partial \alpha} \log L(\alpha, \beta) = -\frac{1}{2} Tr \left( (e^\alpha H + e^\beta I)^{-1} e^\alpha H \right) \\
- \frac{1}{2} y^T \left( - (e^\alpha H + e^\beta I)^{-1} e^\alpha H (e^\alpha H + e^\beta I)^{-1} \right) y^* \\
= -e^{\frac{1}{2}} \frac{1}{2} Tr \left( (e^\alpha H + e^\beta I)^{-1} \right) \\
- e^{\frac{1}{2}} \frac{1}{2} y^T \left( - (e^\alpha H + e^\beta I)^{-1} H (e^\alpha H + e^\beta I)^{-1} \right) y^*
\]

hence, in particular

\[
\frac{\partial}{\partial \alpha} \log L(\alpha, \beta) = -\frac{1}{2} \sum_i \left( \frac{e^\alpha \lambda_i}{(e^\alpha \lambda_i + e^\beta)} \right) + \frac{1}{2} y^T \sum_i \frac{e^\alpha \lambda_i}{(e^\alpha \lambda_i + e^\beta)^2} a_i a_i^T y^*. \tag{3.17}
\]

Similarly for \( \beta \), we get

\[
\frac{\partial}{\partial \beta} \log L(\alpha, \beta) = -\frac{1}{2} Tr \left( (e^\alpha H + e^\beta I)^{-1} e^\beta I \right) \\
- \frac{1}{2} y^T \left( - (e^\alpha H + e^\beta I)^{-1} e^\beta I (e^\alpha H + e^\beta I)^{-1} \right) y^* \\
= -e^{\frac{1}{2}} \frac{1}{2} Tr \left( (e^\alpha H + e^\beta I)^{-1} \right) \\
- e^{\frac{1}{2}} \frac{1}{2} y^T \left( - (e^\alpha H + e^\beta I)^{-1} H (e^\alpha H + e^\beta I)^{-1} \right) y^*
\]

and in particular

\[
\frac{\partial}{\partial \beta} \log L(\alpha, \beta) = -\frac{1}{2} \sum_i \left( \frac{e^\beta}{(e^\alpha \lambda_i + e^\beta)} \right) + \frac{1}{2} y^T \sum_i \frac{e^\beta}{(e^\alpha \lambda_i + e^\beta)^2} a_i a_i^T y^*. \tag{3.18}
\]

If we now set both (3.17) and (3.18) equal to zero, we get for \( \alpha \) and \( \beta \) respectively,

\[
\sum_i \frac{e^\alpha \lambda_i}{e^\alpha \lambda_i + e^\beta} = \sum_i \frac{e^\alpha \lambda_i}{(e^\alpha \lambda_i + e^\beta)^2} (a_i^T y^*)^2 \tag{3.19}
\]

\[
\sum_i \frac{e^\beta}{(e^\alpha \lambda_i + e^\beta)} = \sum_i \frac{e^\beta}{(e^\alpha \lambda_i + e^\beta)^2} (a_i^T y^*)^2. \tag{3.20}
\]

Investigating whether (3.19) and (3.20) hold gets complex, specifically for big \( n \), and establishing whether there are stationary points seems analytically not possible. Therefore, as the second derivatives are even more complicated, as well as checking the negative/positive definiteness of the Hessian matrix, in order to understand better the behaviour of (3.13) we can investigate the asymptotic behaviour of the function. Let us give then the following proposition.
Proposition 1. Let us consider the regression model given by (3.9), and in particular the profile log-likelihood function (3.13). Then

i) for \( n = 2 \), with probability one the function (3.13) goes to \(-\infty\) in all directions, except for

\[
\alpha \to -\infty, \beta \text{ fixed, } \log \mathcal{L}(\alpha, \beta) \to -\frac{n}{2} \log 2\pi - \frac{n}{2} \beta - \frac{1}{2} \sum_{i=1}^{n} \frac{(a_i^T y^*)^2}{e^{\beta^2}},
\]

\[
\beta \to -\infty, \alpha \text{ fixed, } \log \mathcal{L}(\alpha, \beta) \to +\infty
\]

and if we look at \( \log \mathcal{L}(u\gamma + c, v\gamma + d) \) for \( u, v, c, d, \gamma \in \mathbb{R} \), with \( u, v, c, d \) fixed,

if \((n-1)u+v < 0\), \( \log \mathcal{L}(u\gamma + c, v\gamma + d) \to \infty \)

if \((n-1)u+v = 0\), \( \log \mathcal{L}(u\gamma + c, v\gamma + d) \to 0 \)

ii) for \( n > 2 \) with probability one, the function (3.13) goes to \(-\infty\) in all directions, except for

as \( \alpha \to -\infty, \beta \text{ fixed, } \log \mathcal{L}(\alpha, \beta) \to -\frac{n}{2} \log 2\pi - \frac{n}{2} \beta - \frac{1}{2} \sum_{i=1}^{n} \frac{(a_i^T y^*)^2}{e^{\beta^2}} \) \hspace{1cm} (3.21)

as \( \beta \to -\infty, \alpha \text{ fixed, } \log \mathcal{L}(\alpha, \beta) \to -\frac{n}{2} \log 2\pi - \frac{n}{2} \alpha - \frac{1}{2} \sum_{i=1}^{n} \frac{(a_i^T y^*)^2}{e^{\alpha \lambda_i}} + \frac{1}{2} \sum_{i=1}^{n} \log \lambda_i. \) \hspace{1cm} (3.22)

Moreover, (3.13) admits two asymptotes, whose equations are given by

\[
\{(\alpha, \beta^*) : \alpha \in \mathbb{R}\} \hspace{1cm} (3.23)
\]

\[
\{\alpha^*, \beta) : \beta \in \mathbb{R}\} \hspace{1cm} (3.24)
\]

where \( \beta^* \) and \( \alpha^* \) are given by

\[
\beta^* = \log \left( \frac{1}{n} \sum_{i=1}^{n} (a_i^T y^*)^2 \right) \hspace{1cm} (3.25)
\]

\[
\alpha^* = \log \left( \frac{1}{n} \sum_{i=1}^{n} \frac{(a_i^T y^*)^2}{\lambda_i} \right). \hspace{1cm} (3.26)
\]

What we still want to investigate is whether and how many local maxima the function (3.13) has. From the above proposition we see that with probability 1, when \( n > 2 \), (3.13) goes to \(-\infty\) in almost all directions, hence, since it is continuous, it has at least one local maximum, probably on the ridge. In fact, one interesting case is given for \( n = 6 \). Plotting the function (figure 3.2) we see that it admits at least one stationary point around the cusp. Therefore, if we consider proposition 1 and what we have said before, with probability bigger than 0 the function will have at least two local maxima.

The case \( n = 2 \) is different. The function can reach \( \infty \). Let us now have a look at this special case. Here things get easier and it is possible to directly compute what we did so far for the general case.
3.1.1. The case $n=2$

Let us consider the case with $n=2$. Hence for $i=1,2$ we have the following

$$(y_1, y_2) \sim \mathcal{N}_2(\mu, K_{h,\alpha,\beta}).$$  \hfill (3.27)

Again, let us consider the profile likelihood $\mathcal{L}(\alpha, \beta) := \mathcal{L}(\alpha, \beta, \hat{\mu})$, where $\hat{\mu}$ is the MLE of $\mu$, namely $\hat{\mu} = \bar{Y} = \frac{1}{2} \sum_{i=1}^2 y_i$. If we plot the profile log-likelihood $\log \mathcal{L}(\alpha, \beta)$ as a function of $\alpha$ and $\beta$, we get

$$\log \mathcal{L}(\alpha, \beta) = -\log 2\pi - \frac{1}{2} \log |K_{h,\alpha,\beta}| - \frac{1}{2} (y_1 - \bar{Y}, y_2 - \bar{Y}) K_{h,\alpha,\beta}^{-1} (y_1 - \bar{Y}, y_2 - \bar{Y})^T$$  \hfill (3.28)

In this case it is easy to directly compute the profile log-likelihood, indeed we have

$$\log \mathcal{L}(\alpha, \beta) = -e^\beta + \log (e^\alpha + 4e^\beta) + 2 \log \pi - \frac{(y^*_1 - y^*_2)^2}{e^\alpha + 4e^\beta}$$  \hfill (3.29)

where $y^*_i = y_i - \bar{y}$ for $i = 1, 2$, and analytically the function is easier to study. Let us give the following proposition.
Proposition 2. Let
\[ y_i = \mu + f(x_i) + \epsilon_i, \]
for \( i = 1, 2 \), where \( \mu \) is a constant, \( f \) is a realization of a centered Brownian motion with scale parameter \( e^\alpha \) and \( \epsilon_i \sim \mathcal{N}(0, e^\beta) \). The function \( \log L(\alpha, \beta) \) given by (3.29) admits no stationary point. Furthermore, we have

i) (3.29) has two ridges, whose equations are given by
\[
(\alpha, \beta, \alpha, \log L(\alpha, \beta)) \quad \text{and} \quad (\alpha, \beta, \beta, \log L(\alpha, \beta)),
\]
where
\[
\beta = \log \left( \frac{1}{16} \left( -3e^\alpha \pm 2 \sqrt{e^{2\alpha} - 12e^\alpha (y_1^* - y_2^*)^2 + 4(y_1^* - y_2^*)^4 + 2(y_1^* - y_2^*)^2} \right) \right)
\]
\[
\alpha = \log \left( 2(y_1^* - y_2^*)^2 - 4e^\beta \right)
\]
with \( y_i^* = y_i - \bar{Y} \) for \( i = 1, 2, \).

ii) The above ridges have asymptotes, whose equations are given by
\[
\beta = \log \left( \frac{(y_1^* - y_2^*)^2}{4} \right), \text{ for the ridge } \beta
\]
\[
\beta = \log \left( \frac{(y_1^* - y_2^*)^2}{2} \right) \text{ and } \alpha = \log(2(y_1^* - y_2^*)^2), \text{ for the ridge } \alpha.
\]

iii) (3.29) is concave for the values of \( \alpha \) and \( \beta \) such that
\[
e^\alpha + 4e^\beta > 2(y_1^* - y_2^*)^2. \quad (3.30)
Figure 3.4.: Random log-likelihood for $n = 2$, the blue region is where the function is concave, $\alpha, \beta \in [-10, 10]$

### 3.2. Proofs

**Proof of proposition 1.** We will start showing $ii)$ and then it will follow $i)$. We have that (see B)

$$
\log \mathcal{L}(\alpha, \beta) = -\frac{n}{2} \log 2\pi - \frac{1}{2} \sum_i \log (e^\alpha \lambda_i + e^\beta) - \frac{1}{2} \sum_i \left( \frac{(a_i^T y^*)^2}{e^\alpha \lambda_i + e^\beta} \right).
$$

Critical points are given when $e^\alpha \lambda_i + e^\beta \leq 0$, as the logarithm is not well defined. Since $\lambda_i$ for $i = 1, ..., n$ are eigenvalues of the covariance kernel matrix of a centered Brownian motion, we know that they all are nonnegative. Hence we can assume that $e^\alpha \lambda_i + e^\beta > 0$. We have

- as $\alpha \to \infty$, $\beta$ fixed, $\log \mathcal{L}(\alpha, \beta) \to -\infty$;
- as $\alpha \to -\infty$, $\beta$ fixed,

$$
\log \mathcal{L}(\alpha, \beta) \to -\frac{n}{2} \log 2\pi - \frac{n}{2} \beta - \frac{1}{2} \sum_i \frac{(a_i^T y^*)^2}{e^\beta} := t^* > -\infty; \quad (3.31)
$$

- as $\beta \to \infty$, $\alpha$ fixed, $\log \mathcal{L}(\alpha, \beta) \to -\infty$;
- as $\beta \to -\infty$, $\alpha$ fixed,

$$
\log \mathcal{L}(\alpha, \beta) \to -\frac{n}{2} \log 2\pi - \frac{n}{2} \alpha - \frac{1}{2} \sum_i \log \lambda_i - \frac{1}{2} \sum_i \frac{(a_i^T y^*)^2}{e^\alpha \lambda_i} := t^{**} > -\infty. \quad (3.32)
$$
Now let us have a look at $\log L(u\gamma + c, v\gamma + d)$ where $\gamma, u, v, c, d \in \mathbb{R}$ and $u, v, c, d$ are fixed. Hence

$$\log L(u\gamma + c, v\gamma + d) = -\frac{n}{2} \log 2\pi - \frac{1}{2} \sum \log(e^{u\gamma + c} + e^{v\gamma + d}) - \frac{1}{2} \sum \frac{(a_i^T y^*)^2}{e^{u\gamma + c} + e^{v\gamma + d}}.$$ 

Therefore,

- suppose $u, v > 0$ fixed, then
  $$\lim_{\gamma \to -\infty} \log L(u\gamma + c, v\gamma + d) = -\infty,$$
  $$\lim_{\gamma \to -\infty} \log L(u\gamma + c, v\gamma + d) = -\infty;$$

- suppose $u > 0$ and $v < 0$ fixed, then
  $$\lim_{\gamma \to -\infty} \log L(u\gamma + c, v\gamma + d) = -\infty,$$
  $$\lim_{\gamma \to -\infty} \log L(u\gamma + c, v\gamma + d) = -\infty.$$

Thus, we see that $\log L(\alpha, \beta)$ goes to $-\infty$ in all directions, but (3.32) and (3.31). Now, let us suppose that there exists an $i$ such that $\lambda_i = 0$, and without loss of generality we can assume that $\lambda_1 = 0$. Then,

$$\log L(\alpha, \beta) = -\frac{n}{2} \log 2\pi - \frac{1}{2} \sum_{i \neq 1} \log(e^{\alpha \lambda_i} + e^{\beta}) - \frac{\beta}{2} - \frac{1}{2} \sum_{i \neq 1} \frac{(a_i^T y^*)^2}{e^{\alpha \lambda_i} + e^{\beta}} - \frac{1}{2} (a_1^T y^*)^2.$$ 

Thus,

- as $\alpha \to \infty$, $\beta$ fixed, $\log L(\alpha, \beta) \to -\infty$;
- as $\alpha \to -\infty$, $\beta$ fixed, $\log L(\alpha, \beta) \to -\frac{n}{2} \log 2\pi - \frac{\beta}{2} - \frac{1}{2} \sum (a_i^T y^*)^2 > -\infty$;
- as $\beta \to \infty$, $\alpha$ fixed, $\log L(\alpha, \beta) \to -\infty$;
- as $\beta \to -\infty$, $\alpha$ fixed,
  $$\begin{cases} 
  \text{if } (a_1^T y^*)^2 = 0, & \log L(\alpha, \beta) \to +\infty \\
  \text{if } (a_1^T y^*)^2 \neq 0, & \log L(\alpha, \beta) \to -\infty 
  \end{cases}$$

Again, if we look at $\log L(u\gamma + c, v\gamma + d)$, we get

- if $u, v > 0$ as $\gamma \to \infty \log L(u\gamma + c, v\gamma + d) \to -\infty$;
- if $u, v < 0$ as $\gamma \to \infty \log L(u\gamma + c, v\gamma + d) \to -\infty$;
- if $u < 0, v > 0$ as $\gamma \to \infty \log L(u\gamma + c, v\gamma + d) \to -\infty$;
• if $u > 0, v < 0$ and $(a_i^T y)^2 \neq 0$, as $\gamma \to \infty \log L(u\gamma + c, v\gamma + d) \to -\infty$

• if $u > 0, v < 0$ and $(a_i^T y)^2 = 0$, as $\gamma \to \infty$
  
  if $(n-1)u + v > 0$ \( \log L(u\gamma + c, v\gamma + d) \to -\infty \)
  if $(n-1)u + v < 0$ \( \log L(u\gamma + c, v\gamma + d) \to \infty \)
  if $(n-1)u + v = 0$ \( \log L(u\gamma + c, v\gamma + d) \to 0. \)

Therefore, we see what changes in the asymptotic behaviour of $\log L(\alpha, \beta)$ if at least one eigenvalue of $H$ is zero. In particular, if $n > 2$ then $P(a_i^T y^* = 0) = 0$ by continuity of $a_i^T y^*$ for every $i = 1, \ldots, n$. Thus for $n > 2$ the function goes to $-\infty$ in all directions, except for (3.31) and (3.32), yielding $ii)$. While, we see for $n = 2$, by equation (3.29), with probability bigger than zero there exists $i$ such that $a_i^T y^* = 0$, yielding $i$).

To complete the proof we only need to find the equations of the asymptotes. Looking at (3.31) and (3.32), we can maximize them with respect to $\beta$ and $\alpha$, respectively. We get

\[
\frac{\partial}{\partial \beta} l^* = -\frac{n}{2} + \frac{1}{2} \sum_{i=1}^{n} \frac{(a_i^T y^*)^2}{e^\beta}
\]

\[
\frac{\partial}{\partial \alpha} l^{**} = -\frac{n}{2} + \frac{1}{2} \sum_{i=1}^{n} \frac{(a_i^T y^*)^2}{\lambda_i e^\alpha}
\]

which are equal to zero for, respectively,

\[
\beta^* = \log \left( \frac{1}{n} \sum_{i=1}^{n} (a_i^T y^*)^2 \right) \quad (3.33)
\]

\[
\alpha^* = \log \left( \frac{1}{n} \sum_{i=1}^{n} \frac{(a_i^T y^*)^2}{\lambda_i} \right). \quad (3.34)
\]

These two define in particular the asymptotes of the function $\log L(\alpha, \beta)$, whose equations are given by

\[
\{ (\alpha, \beta^*) : \alpha \in \mathbb{R} \} \quad (3.35)
\]

and

\[
\{ (\alpha^*, \beta) : \beta \in \mathbb{R} \}, \quad (3.36)
\]

which finishes our proof.

**Proof of proposition 2.** Firstly, we need to show that (3.29) has no stationary points. For $n = 2$, we can directly compute the derivatives, yielding

\[
\frac{\partial}{\partial \alpha} \log L(\alpha, \beta) = -\frac{e^\alpha (e^\alpha + 4e^\beta - 2(y_1^* - y_2^*)^2)}{2(e^\alpha + 4e^\beta)^2}
\]

\[
\frac{\partial}{\partial \beta} \log L(\alpha, \beta) = \frac{8e^\beta (y_1^* - y_2^*)^2 - (e^\alpha + 4e^\beta)(e^\alpha + 8e^\beta)}{2(e^\alpha + 4e^\beta)^2}.
\]

34
If we set both the derivatives with respect to $\alpha$ and $\beta$ equal to zero, it is easy to see that there is no solution.

Let us now prove $i)$ and $ii)$. Firstly, let us define

$$\beta_\alpha := \arg \max_{\beta} \log L(\alpha, \beta)$$
$$= \arg \max_{\beta} \left(- \frac{(e^\alpha + 4e^\beta) (\log(e^{\beta}(e^\alpha + 4e^\beta)) + 2\log(\pi)) + 2y_1^2 - 4y_2^2y_1 + 2y_2^2}{2(e^\alpha + 4e^\beta)}\right).$$

If we compute the derivative with respect to $\beta$ and we set this equal to zero, we get

$$\beta_{\alpha_1} = \log \left(\frac{1}{16}(-3e^\alpha + \sqrt{e^{2\alpha} - 12e^\alpha(y_1^*-y_2^*)^2 + 4(y_1^*-y_2^*)^4 + 2(y_1^*-y_2^*)^2})\right)$$
$$\beta_{\alpha_2} = \log \left(\frac{1}{16}(-3e^\alpha - \sqrt{e^{2\alpha} - 12e^\alpha(y_1^*-y_2^*)^2 + 4(y_1^*-y_2^*)^4 + 2(y_1^*-y_2^*)^2})\right)$$

Both are well defined for $\alpha \leq \log\left[\frac{6}{4\sqrt{2}}(y_1^*-y_2^*)^2\right]$. If we let $\alpha \to -\infty$, we can see that $\beta_{\alpha_1}$ has an horizontal asymptote given by $\beta = \log\left(\frac{y_1^*-y_2^*}{2}\right)$.

Secondly, let us define

$$\alpha_\beta := \arg \max_{\alpha} \log L(\alpha, \beta)$$
$$= \arg \max_{\alpha} \left(- \frac{(e^\alpha + 4e^\beta) (\log(e^{\beta}(e^\alpha + 4e^\beta)) + 2\log(\pi)) + 2y_1^2 - 4y_2^2y_1 + 2y_2^2}{2(e^\alpha + 4e^\beta)}\right).$$

If we compute the derivative with respect to $\alpha$ and we set this equal to zero, we get

$$\alpha_{\beta} = \log \left(2(y_1^*-y_2^*)^2 - 4e^\beta\right),$$

which is well defined for $\beta < \log\left(\frac{1}{2}(y_1^*-y_2^*)^2\right)$ and it has a vertical asymptote and a horizontal asymptote, whose equations are given, respectively, by

$$\beta = \log\left(\frac{1}{2}(y_1^*-y_2^*)^2\right)$$
$$\alpha = \log\left(2(y_1^*-y_2^*)^2\right)$$

if we let $\beta \to -\infty$.

Finally, let us prove the last point, $iii)$. A function is concave when the Hessian matrix $H$ is semi-definite negative, i.e. if for every vector $x \in \mathbb{R}^n$ $xHx \leq 0$, or equivalently if all the eigenvalues of $H$ are nonpositive. In this case, when $n = 2$, it is possible to directly
compute the eigenvalues of the Hessian matrix $\phi_i$, for $i = 1, 2$. Indeed we have

$$
\phi_1 = \frac{(y_1^*-y_2^*)^2(e^\alpha-4e^\beta)^2}{2(e^\alpha + 4e^\beta)^3} \left( (y_1^*-y_2^*)^2(224e^{2(\alpha+\beta)} + e^{4\alpha} + 256e^{4\beta}) - 128(y_1^*-y_2^*)^2e^{2(\alpha+\beta)}(e^\alpha + 4e^\beta) \right) \\
+ \frac{16e^{2(\alpha+\beta)}(e^\alpha + 4e^\beta)^2)^{1/2}}{2(e^\alpha + 4e^\beta)^3} - \frac{4e^{\alpha+\beta}(e^\alpha + 4e^\beta)}{2(e^\alpha + 4e^\beta)^3} \\
\phi_2 = \frac{(y_1^*-y_2^*)^2(-e^\alpha - 4e^\beta)^2}{2(e^\alpha + 4e^\beta)^3} \\
+ \frac{(y_1^*-y_2^*)^2(224e^{2(\alpha+\beta)} + e^{4\alpha} + 256e^{4\beta}) - 128(y_1^*-y_2^*)^2e^{2(\alpha+\beta)}(e^\alpha + 4e^\beta) \right) \\
+ \frac{16e^{2(\alpha+\beta)}(e^\alpha + 4e^\beta)^2)^{1/2}}{2(e^\alpha + 4e^\beta)^3} - \frac{4e^{\alpha+\beta}(e^\alpha + 4e^\beta)}{2(e^\alpha + 4e^\beta)^3}.
$$

We see that $\phi_1$ is negative, while $\phi_2$ can be negative only for specific values of $\alpha$ and $\beta$. In particular, we get that $\phi_2 < 0$ if and only if $e^\alpha + 4e^\beta > 2(y_1^*-y_2^*)^2$. Indeed we see that the first term of $\phi_2$ is negative, the third one as well and the second one is positive. Thus $\phi_2$ can be negative if and only if the sum of the two negative terms is in absolute value bigger than the third one, namely if and only if

$$
(y_1^*-y_2^*)^4(e^\alpha-4e^\beta)^2 + 8(y_1^*-y_2^*)^2(e^\alpha-4e^\beta)^2e^{\alpha+\beta}(e^\alpha + 4e^\beta) \\
+ 16e^{2\alpha+2\beta}(e^\alpha + 4e^\beta)^2 > 0
$$

and only if

$$
(y_1^*-y_2^*)^4(224e^{2\alpha+2\beta} + e^{4\alpha} + 256e^{4\beta}) - 128(y_1^*-y_2^*)^2e^{2\alpha+2\beta}(e^\alpha + 4e^\beta) \\
+ 16e^{2\alpha+2\beta}(e^\alpha + 4e^\beta)^2 > 0
$$

if and only if

$$
(y_1^*-y_2^*)^4(e^\alpha-4e^\beta)^2 + 8(y_1^*-y_2^*)^2(e^\alpha-4e^\beta)^2e^{\alpha+\beta}(e^\alpha + 4e^\beta) > 0
$$

if and only if

$$
(y_1^*-y_2^*)^4(224e^{2\alpha+2\beta} + e^{4\alpha} + 256e^{4\beta}) - 128(y_1^*-y_2^*)^2e^{2\alpha+2\beta}(e^\alpha + 4e^\beta) > 0
$$

if and only if

$$
-16(y_1^*-y_2^*)^4e^{\alpha+\beta}(e^\alpha + 4e^\beta)^2 + 8(y_1^*-y_2^*)^2e^{\alpha+\beta}(e^\alpha + 4e^\beta)^3 > 0
$$

if and only if

$$
8(y_1^*-y_2^*)^2e^{\alpha+\beta}(e^\alpha + 4e^\beta)^2(-2(y_1^*-y_2^*)^2 + e^\alpha + 4e^\beta) > 0
$$

if and only if

$$
e^{\alpha} + 4e^\beta > 2(y_1^*-y_2^*)^2.
$$

Therefore, we see that for $e^\alpha + 4e^\beta > 2(y_1^*-y_2^*)^2$ the Hessian matrix admits two negative eigenvalues, hence it is negative definite and in particular the function $\log \mathcal{L}(\alpha, \beta)$ is concave, and this concludes our proof.
4. Experimental Results

In the previous chapter we have obtained theoretical results, but it is still not possible to solve our main problem, whether the function $L(\alpha, \beta)$, given by (3.13), has one or multiple local maxima. We could not even determine the existence of stationary points for $n > 2$. Therefore, since this local problem of optimization can not be solved using the classical solution given by mathematical analysis, which involves the use of the gradient and Hessian matrix (namely analytical procedures), we then need numerical analysis.

Our goal in this chapter is to verify the existence of multiple local maxima and global maximum of the function $\log L(\alpha, \beta)$, and, most of all, to find a systematic method that can detect the local optima and the global optimum of the function, if possible. Moreover, we want to simulate randomly and estimate the probability of $\log L(\alpha, \beta)$ having one, or more local maxima. In order to do that, we use the software Wolfram Mathematica 11.2, which implements several maximization methods.

4.1. Methods

The Wolfram Language has a collection of commands that perform unconstrained optimization, like e.g. FindMaximum or NMaximize. What we used the most is the command FindMaximum, which tries, implementing numerical methods, to find the local maxima of the objective function $f$. It does so by doing a search starting at some initial value, let us call it $x_0$, generating a sequence of iterates $\{x_k\}_{k=0}^{\infty}$ that terminate when a solution point has been approximated with sufficient accuracy or when it seems that no more progress can be made. In deciding how to move from $x_k$ to $x_{k+1}$, the algorithm uses information from the previous iterates and $f_k := f(x_k)$, such that every step increases the height of the function. In contrast, the function NMaximize tries harder to find the global maximum, but it has much more work to do and thus, it generally takes much more time than FindMaximum, which is often preferred to the other one for global optimization problems as well.

When using FindMaximum, if we do not specify any method, Mathematica uses the Quasi-Newton method, unless the problem is structurally a sum of squares, in which case the Levenberg-Marquardt variant of the Gauss-Newton method is used, or, if we give an initial point, the Principal Axis method is used. However, Mathematica has four essentially different methods (we are excluding the Levenberg-Marquardt since we do not have a function sum of squares), namely

- **Newton Method** It uses "the exact Hessian matrix or a finite difference approx-
imation if the symbolic derivative cannot be computed” [6, 9].

- **Quasi Newton Method** It uses "the quasi-Newton BFGS approximation to the Hessian built up by updates based on past steps” [5, 6, 9].

- **Conjugate Gradient** It is "a non-linear version of the conjugate gradient method for solving linear systems” [6, 9].

- **Principal Axis** It "works without using any derivatives, not even the gradient, but by keeping track of the values from past steps. It requires two starting conditions in each variable” [7, 9].

Thus, we chose the method which could work for all n, with different initial points. It came out that the only method working every time is the nonlinear Conjugate Gradient method, or Gradient method which simply is the older version of the first one (see B for a complete description of the algorithm). Note that as a numerical algorithm converges, it is necessary to keep track of the convergence and try to understand if the maximum point has been approached closely enough. Since we are dealing with bivariate functions, this depends on the sequence of steps taken and the values of the function, its gradient and the Hessian at these points.

In general there are two fundamental methods for moving from the current point $x_k$ to the new one $x_{k+1}$. The *line search* method and the *trust region* method. In the following part we are going to consider algorithms which try to minimize the objective function. In our case we want to maximize, but the adaption easily follows. The following parts are taken from [6].

### 4.1.1. The line search method

The *line search* method chooses a direction $p_k$ and searches along this direction from the current iterate $x_k$ to a new iterate $x_{k+1}$ with a lower function value. The distance $\alpha$ to move along $p_k$ can be found by approximately solving the following minimization problem:

$$
\min_{\alpha > 0} f(x_k + \alpha p_k).
$$

Solving exactly (4.1) is usually not needed. Therefore, the line search algorithm generates a limited number of trial step lengths until it finds the one that approximates the minimum of (4.1). At $x_{k+1}$ a new search direction $p_{k+1}$ and step length $\alpha_{k+1}$ are computed, and the process is repeated.

When choosing the direction $p_k$ in the line search approach, the steepest descent direction $-\nabla f_k$ is the most obvious one, as moving from $x_k$ it is the one along which $f$ decreases most rapidly [6], yielding to the *steepest descent method* (but slow on difficult problems). Another important search direction is the *Newton* direction, which is derived from the second-order Taylor series approximation to $f(x_k + p)$,

$$
f(x_k + p) \approx f_k + p^T \nabla f_k + \frac{1}{2} p^T \nabla^2 f_k p =: m_k(p).
$$

38
Assuming that $\nabla^2 f_k$ is positive definite, we obtain the Newton direction by finding the vector $p$ which minimizes $m_k(p)$, namely

$$p_k = -(\nabla^2 f_k)^{-1}\nabla f_k.$$  \hfill (4.3)

Usually methods that use the Newton direction have a quadratic rate of convergence. An alternative is given by the Quasi-Newton search direction, which does not require the computation of the Hessian matrix [6].

4.1.2. The trust region method

The second method, known as trust region, is such that the information about the function $f$ is used to build a model function $m_k$ of $f$, such that they behave similarly near the current point $x_k$. In order to do that, the algorithm restricts the search for a minimizer of $m_k$ to some region around $x_k$. In other words, the candidate step $p$ is found by approximately solving the following sub-problem:

$$\min_p m_k(x_k + p), \text{ where } x_k + p \text{ lies inside the trust region.}$$  \hfill (4.4)

If the candidate solution does not sufficiently decrease $f$, the trust region is too large and a new solution of (4.4) is found. Usually the trust region is a ball defined by $||p||_2 \leq \Delta$, where the scalar $\Delta > 0$ is called trust-region radius. The model $m_k$ in (4.4) is usually defined by the following quadratic function

$$m_k(x_k + p) = f_k + p^T \nabla f_k + \frac{1}{2} p^T B_k p,$$  \hfill (4.5)

where $f_k$, $\nabla f_k$, and $B_k$ are a scalar, vector, and matrix, respectively. The matrix $B_k$ is either the Hessian $\nabla^2 f_k$ or some approximation. If we set $B_k = 0$ in (4.5) and define the trust region using the Euclidean norm, the trust region problem (4.4) becomes

$$\min_p f_k + p^T \nabla f_k \text{ subject to } ||p||_2 \leq \Delta,$$

and the solution is given by

$$p_k = -\frac{\Delta_k \nabla f_k}{||\nabla f_k||}.$$ 

This is simply the steepest descent step in which the step length is determined by the trust-region radius. If we choose $B_k$ to be the exact Hessian matrix then we obtain the trust-region Newton method, which has been shown to be highly effective in practice [6].

Therefore, we see that the line search and trust region strategies differ in the order in which they choose the direction and distance of the move to the next iterate.
4.2. Setting the problem

Let us consider again the model (3.1) given in chapter 3, and the profile log-likelihood obtained in (3.12),

\[ L(\alpha, \beta) = \frac{1}{\sqrt{(2\pi)^n |K_{h,\alpha,\beta}|}} \exp \left( -\frac{1}{2}(Y - \bar{Y})^T (K_{h,\alpha,\beta})^{-1} (Y - \bar{Y}) \right). \]

By propositions 1 and 2 we have

- for \( n = 2 \) with probability 1 the function reaches \( \infty \), and it has two ridges, whose equations are found
- for \( n > 2 \) with probability 1 the function goes to \( -\infty \) in all directions, and hence we can suppose it has one and probably more local maxima.

Our goal is to find a systematic method such that it can detect all the local maxima and the global maximum of the function, taking random or real data as input. Thus, the first thing to decide, once the method is chosen, is which initial point we should give to the optimisation algorithm. It would make sense to give a point where the Hessian matrix is definite negative, hence the marginal likelihood concave, and relatively close to the point of maximum. Therefore, as we are supposing that the local maxima are on the two ridges and/or around the cusp, we give the following three intial points:

- \((\alpha^*, \beta^*)\), the intersection of the two asymptotes of \( \log L(\alpha, \beta) \) found in the previous chapter (Proposition 1),

\[
(\alpha^*, \beta^*) = \left( \log \left( \frac{1}{n} \sum_{i=1}^{n} \frac{(a_i^T y^*)^2}{\lambda_i} \right), \log \left( \frac{1}{n} \sum_{i=1}^{n} (a_i^T y^*)^2 \right) \right). \tag{4.6}
\]

- \((\alpha^*, \beta_{\alpha^*}^*)\), where \( \beta_{\alpha^*}^* \) is defined such that

\[
\beta_{\alpha^*}^* := \arg \max_{\beta} \log L(\alpha^*, \beta) \tag{4.7}
\]

- \((\alpha_{\beta^*}^*, \beta^*)\), where \( \alpha_{\beta^*}^* \) is defined such that

\[
\alpha_{\beta^*}^* := \arg \max_{\alpha} \log L(\alpha, \beta^*) \tag{4.8}
\]

We see that \((\alpha^*, \beta^*)\) can be found analytically, while it is not possible for the other two points, \((\alpha^*, \beta_{\alpha^*}^*)\) and \((\alpha_{\beta^*}^*, \beta^*)\). Thus, again, in this situation we are going to use the command FindMaximum.

For \( i = 1, \ldots, n \), we simulate randomly, \( y_i \) as real values between 0 and 1, and \( x_i \) such that \( x_i = \frac{\text{range}[n] - 0.5}{n} \), where in Mathematica \text{range}[n] generates a list of integer values between 1 and \( n \). Let us have a look firstly at the plots (with some possible variations) of...
As you can see in figure 4.1 the evaluation of the function $\log L(\alpha^*, \beta)$ is not stable after a certain value of $\beta$. Thus, FindMaximum is not always able to find exactly the point of maximum, and it will often give back a value with a warning (e.g. "machine precision is insufficient to achieve the requested accuracy or precision"). In figure 4.2 the situation is different. If we look at the first one 4.2a, we do not have a maximum, the function is just increasing until it seems it remains constant after a certain value of $\alpha$. Thus, FindMaximum gives back different values if we use different methods, depending on the convergence of these. In the second one, 4.2b, we see the function having a maximum, whose value will be exactly given.

4.3. Results: number of local maxima and location of the global maximum

We ran 150 simulations for each $n = 3, \ldots, 10$ observation. As said before, we expected the function having at most three local maxima, two on the ridges and one around the cusp. Let us call the $\alpha$-ridge, the ridge where $\beta = \beta^*$ and $\alpha$ is varying, and the $\beta$-ridge defined in the analogous way, where $\alpha = \alpha^*$ and $\beta$ is varying. Surprisingly, the function shows at most two local maxima, one around the cusp and one on or around the $\beta$-ridge.
Moreover, the global maximum is always on the $\beta$-ridge or around it. Here we compare the results obtained for different numbers of observations $n$.

<table>
<thead>
<tr>
<th>$n$</th>
<th>Estimated probability of having at least 1 maximum</th>
<th>Estimated probability of having at least 2 local maxima</th>
<th>Estimated probability of having the global maximum on the $\beta$-ridge</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0.76</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0.06</td>
<td>0.527</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0.12</td>
<td>0.427</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>0.22</td>
<td>0.36</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>0.367</td>
<td>0.367</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>0.293</td>
<td>0.24</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>0.187</td>
<td>0.427</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>0.193</td>
<td>0.293</td>
</tr>
</tbody>
</table>

Figure 4.3.  
Figure 4.4.

Note that the estimated probabilities in the above tables are computed as fractions of the 150 runs.

We firstly see from the tables, and then from the graphs below, that the probability of $\log L(\alpha, \beta)$ having at least one maximum is always one. This result should not surprise us, in fact, already in the previous chapter, we have shown that for $n \geq 3$ the function goes to $-\infty$ in almost all directions, and so it has at least one maximum with probability one (since it is continuous). More interesting is that the probability of having at least two local maxima is not always the same. For $n = 3, 4$, the probability is actually zero, while for $n > 4$ this probability increases, obtaining the biggest value for $n = 10$. It seems then that as $n$ gets bigger, the probability does so, even if for $n = 7$ we have an exception, as the value here is bigger than the value for $n = 8, 9$, but it is probably due to the fact of having 150 simulations, so the probability estimates may not be that exact.
Regarding the location of the two maxima, we have the global maximum always on or around the $\beta$-ridge, usually it would be given by $(\alpha^*, \beta_0^*)$, while the local optimum is around the cusp. However, it is not always possible to determine the location with enough accuracy and precision.

Overall we can observe the following three different behaviours of $\log L(\alpha, \beta)$. The function having at least two local maxima, one is visible and around the cusp, while the other one found by Mathematica is on or around the $\beta$-ridge (figure 4.6, where 4.6a is the plot of $\log L(\alpha, \beta)$ for a certain domain and 4.6b is still the plot of the same function but just zoomed in). Then, in figure 4.7, where again the second picture is the same function just zoomed in, we do not have any maximum around the cusp, the function proceeds smoothly and actually it seems to decrease until it becomes flat on the $\alpha$-ridge. Thus it seems that this time the function has only one point of maximum. This is finally different than the behaviour of the function that we have in figure 4.8, where even if the function does not show any maximum around the cusp, it increases passing from the cusp to the $\alpha$-ridge, but still the function shows having only one maximum.

It is interesting is to compare our results with what we obtained in the previous chapter. In particular we can look at $\log L(-\infty, \beta^*)$ and $\log L(\alpha^*, -\infty)$ given by (3.21) and (3.22), respectively. We computed them in every simulation, and what we obtained is that $\log L(\alpha^*, -\infty)$ is always smaller than the maximum found by the program. This let us suppose that what we found seems to be a global maximum, as the function then decreases until it becomes constant for fixed $\alpha$ and as $\beta$ goes to $-\infty$. On the contrast, $\log L(-\infty, \beta^*)$ has the same value as the value the function has on the $\alpha$-ridge. It seems then that on the $\alpha$-ridge there is a very quick convergence to a constant value and probably the maximum is $\log L(-\infty, \beta^*)$, even though we cannot numerically see this (it looks completely flat).

Overall, we can conclude that there is no evidence of a third local maximum, in particular not on the $\alpha$-ridge, as we thought at the beginning. Indeed, the function shows at most two local maxima, of which the global maximum is on (or around) the $\beta$-ridge.

### 4.4. Comments and problems

It is not always possible to get a uniform interpretation of a result, especially when it does not lead to a unique final conclusion.

We have asked ourselves "how many (local and global) maxima are there in the marginal likelihood for Brownian motion regression with independent identical distributed normal errors?". The results suggest the following. The function does not always have a unique point of maximum, but it happens to have at most two local maxima, with a probability that increases as the number of observations $n$ increases.

It is noted that our study is limited by several practical considerations. First of all
we only ran 150 simulations, thus in order to obtain more accurated values of the estimated probabilities, it would be better to increase the number of runs to at least 1000 for each $n = 3, \ldots, 10$, number of observations. Furthermore, the number of observations $n$ should be increased as well, in order to obtain broader results.

Finally, while implementing the algorithm that brought us to these results, we had numerical problems. In particular, we had two different numerical problems in the evaluation of the function.

First of all, it is not possible to detect and locate the maximum on the $\alpha$-ridge. Mathematica finds it, but it is not a point of maximum, the ridge in that point it actually remains flat. Second of all, it is difficult to detect and locate the maximum on the $\beta$-ridge, namely the global maximum. Especially, for $\beta < -70$, the function starts being more and more bumpy and, as a consequence, FindMaximum is not always able to intercept a unique point of maximum (figure 4.5). Therefore, it often happened to get more than one point of maximum around the $\beta$-ridge with really high values of the order of $10^{10}$ or even bigger. It is important to notice that we did not consider these as maxima and this is also the reason why it is not possible to give a more exact location of the local maxima.

Nonetheless, we obtained interesting result and most of all, given the initial points mentioned in the previous section, it is always possible to find the local optima of the function. Therefore, we found what seems to be an automatic method to find a number of (or possibly all) the local maxima of the marginal likelihood. Finally, we were not able to establish the existence of a third local maximum, as it has never been possible to intercept it and there were not enough evidences to show it.
Figure 4.6.: Example of log $\mathcal{L}(\alpha, \beta)$ with $n = 10$ having at least two local maxima

Figure 4.7.: Example of log $\mathcal{L}(\alpha, \beta)$ with $n = 6$ having at least one maximum

Figure 4.8.: Example of log $\mathcal{L}(\alpha, \beta)$ with $n = 6$
5. Conclusion

This thesis concerns the theoretical and numerical study of the marginal likelihood, as function of the GPR hyperparameters. In particular, the main focus is the analysis of a potential major problem which received little attention in the literature, namely, the existence of multiple local maxima. This work can be easily divided into two different but related parts. Let us see each of them.

The first part made by chapters 1 and 2, served as the main purpose to give a more general and theoretical framework to Gaussian process regression and the estimation of GPR hyperparameters. Starting from the background required to deal with Gaussian process regression, we have seen how it is structured, describing all its components, its scope, and how it works, focusing especially on Gaussian processes, covariance function, its importance and the related hyperparameters. Especially these last two elements are necessary for the second chapter, where the problem of estimating GPR hyperparameters is described. Two approaches are explained, the fully Bayes and empirical Bayes methods. In particular, we successively adopted the second one, where the hyperparameters are estimated by means of maximum marginal likelihood. Hence, a major potential problem is introduced, the existence of multiple local maxima, and therefore the risk of finding the local optima instead of the global maxima. Finally, some asymptotic properties of the EB methods are presented.

The second part of this thesis is given by chapters 3 and 4. Here the attention is restricted to a specific GPR model, which depends only on two hyperparameters, the error variance and the scale parameter of the Gaussian process. This part could again be divided into two different but related parts.
A theoretical study of the marginal likelihood is done in chapter 3. We looked at the possible stationary points of the function and we analysed its asymptotic behaviour. Interesting results have then been obtained. We first analysed an easier case, when we have \( n = 2 \) observations. Here, the function has no stationary points and moreover it goes to \( \infty \) with probability one. However, the situation is different if \( n > 2 \). Indeed, it was not possible to analytically detect any stationary points of the function, but, more importantly and differently from the previous case, the function goes to \( -\infty \) in all directions, and so, since the function is continuous, it admits at least one point of maximum with probability one. Moreover, important properties of the ridges (whose equations were found for the case \( n = 2 \)) and asymptotes of the functions are found.

Issues which were theoretically intractable are analysed in chapter 4. Using the software Wolfram Mathematica 11.2, we implemented the nonlinear Conjugate Gradient optimisation algorithm to find the local maxima of the marginal likelihood and we run
simulations in order to estimate the probability of having one or more local maxima. Results showed that for \( n > 2 \) the function has with probability one a point of maximum, already obtained previously, but confirmed numerically, and, more interestingly, it seems that as \( n \) gets bigger, the estimated probability of having at least two local maxima gets bigger. Furthermore, the global maximum was always found on the \( \beta \)-ridge of the function or around it.

Finally and most importantly, we found what seems to be an automatic method to detect a number of (or possibly all) the local maxima of the marginal likelihood. Indeed, given as initial points for the optimisation algorithm (in this case the Conjugate Gradient algorithm) \( (\alpha^*, \beta^*), (\alpha^*, \beta_{\alpha^*}^*) \) and \( (\alpha_{\beta^*}^*, \beta^*) \), found in the previous chapter, we are able to find (possibly all of) the maxima of the function located around the cusp and on or around the \( \beta \)-ridge.

Overall, even in this very simple GPR model, this work shows that the marginal likelihood is affected by the problem of the existence of multiple local maxima. However, if at the beginning we expected the function to have at most three local maxima, we actually found no evidence of the existence of a third maximum, and in fact, the function showed at most two local maxima and only for \( n > 4 \) observations. This together with the increase of the estimated probability of having 2 local maxima as \( n \) gets bigger, suggests that increasing the number of observations \( n \), we will still have two local maxima. However, the posterior, as we saw at the end of chapter 2, as \( n \) grows indefinitely will concentrate around the global maximum. In particular this will be on or around the \( \beta \)-ridge.

In conclusion, this study could provide help to researchers and programmers to find the global maximum of the marginal likelihood in order to retrieve the most useful information from the data. However, note that the study in this thesis is far from comprehensive. Future work can look at a wider range of covariance functions and related hyperparameters which need to be considered, as well as more complex and real data, and a larger number of \( n \) observations. Moreover, we faced several numerical problems, hence other softwares and more efficient algorithms and methods could be implemented (see e.g. [21] or EM algorithm in [20]).
A. Appendix A

A.1. Bayes’s rule

Let us suppose we have data $x$ modelled as a realisation of a random variable $X$. A statistical model is the set of all possible probability distributions $P_\theta$ of $X$, indexed by a parameter $\theta$ taking values in the parameter space $\Theta$. In a Bayesian framework $\theta$ is considered to be a realisation itself of a random variable, say $\vartheta$. Then we can formally define $X$ and $\vartheta$ as measurable maps on a probability space, with values in measurable spaces, $(X, \mathcal{F})$ and $(\Theta, \mathcal{H})$. The distribution $P_\theta$ is then considered a conditional distribution of $X$ given $\vartheta$ and $(X, \vartheta)$ has joint probability distribution given by

$$P(X \in A, \vartheta \in B) = \int_B P_\theta(A) d\Pi(\theta),$$

(A.1)

for every $A \in \mathcal{F}, B \in \mathcal{H}$ and $\Pi$ is the marginal distribution of $\vartheta$, which is called the prior distribution of $\vartheta$ in this setup. Finally, we can define the posterior distribution of $\vartheta$, which is the conditional distribution of $\vartheta$ given $X = x$,

$$\Pi(B|x) = P(\vartheta \in B | X = x),$$

(A.2)

for $B \in \mathcal{H}$ $P^{\Pi}$-a.s.

We can now present a version of Bayes’s formula, as it is given in Ghosal S. and Van der Vaart A. 2017 [27]. For a dominated collection of probability distributions $\{P_\theta : \theta \in \Theta\}$, the distributions $P_\theta$ of the statistical model are defined such that they allow the existence of jointly measurable densities relative to a $\sigma$-finite measure $\mu$ in the sample space $(X, \mathcal{F})$. We assume that there exists a map $(x, \theta) \rightarrow p_\theta(x)$ that is measurable in $(\mathcal{F}, \mathcal{H})$, such that $P_\theta(A) = \int_A p_\theta(x) d\mu(x)$, for every $A \in \mathcal{H}$. In general, the following holds:

$$\int_A \Pi(B|X) dP^{\Pi} = \int_B P_\theta(A) d\Pi(\theta),$$

(A.3)

where $\theta \mapsto P_\theta(A)$ must be measurable for all $A$. Moreover, whenever $\{P_\theta : \theta \in \Theta\}$ is dominated, a version of the posterior distribution (A.2) is a solution of (A.3) and it is given by:

$$\Pi(B|x) = \frac{\int_B p_\theta(x) d\Pi(\theta)}{\int p_\theta(x) d\Pi(\theta)},$$

(A.4)

which holds when the denominator $\int p_\theta(x) d\Pi(\theta)$ is not zero. If there exists a $x$ such that the denominator happens to be zero, then the posterior distribution (A.2) is equal to $Q(B)$ for any arbitrary probability measure $Q$ on $(\Theta, \mathcal{H})$. 

48
Proposition 3. (Bayes’s rule) For a dominated collection of probability distributions \( \{ P_\theta : \theta \in \Theta \} \), if there exists a \( \sigma \)-finite measure \( \mu \) on the sample space \((X, \mathcal{F})\) and jointly measurable maps \((x, \theta) \rightarrow p_\theta(x)\) such that \( P_\theta(A) = \int_A p_\theta(x) d\mu(x) \), for every \( A \in \mathcal{H} \), then (A.4) gives an expression for the posterior distribution.

A detailed proof of this proposition can be e.g. found in [11]. The denominator \( \int p_\theta(x)d\Pi(\theta) \) in (A.4) acts as the norming constant for the density, such that the quotient (A.4) is 1 for \( B = \Theta \). Therefore, we can rewrite Bayes’s formula as,

\[
d\Pi(\theta) \propto p_\theta(x)d\Pi(\theta),
\]

or in words,

posterior \( \propto \) likelihood \( \times \) prior,

where \( \propto \) means ‘proportional to’.

A.2. Eigendecomposition of a matrix

By eigendecomposition of a matrix we mean the factorization of a matrix into a canonical form, whereby the matrix is represented in terms of its eigenvalues and eigenvectors. Only diagonalizable matrices can be factorized in this way.

Let A be a square diagonalizable \( n \times n \) matrix, with \( n \) linearly independent eigenvectors \( q_i \) and eigenvalues \( \lambda_i \) for \( i = 1, \ldots, n \). Then A can be factorized as

\[
A = Q \Lambda Q^{-1}
\]

where \( Q \) is the square \( n \times n \) matrix whose \( i^{th} \) column is the eigenvector \( q_i \) of \( A \) and \( \Lambda \) is the diagonal matrix whose diagonal elements are the corresponding eigenvalues, namely \( \Lambda_{ii} = \lambda_i \). In particular, a matrix \( A \) which admits the above eigendecomposition can be written as

\[
A = \sum_{i=1}^{n} \lambda_i q_i,
\]

the determinant of \( A \) is the product of the eigenvalues

\[
|A| = \prod_{i=1}^{N_A} \lambda_i^{n_i},
\]

and the trace of \( A \) is the sum of the eigenvalues

\[
\text{Tr}(A) = \sum_{i=1}^{N_A} n_i \lambda_i,
\]

where both in (A.8) and (A.9) \( n_i \) is the algebraic multiplicity.
Finally, if a matrix admits inverse matrix, namely all its eigenvalues are different from zero, \( A \) is \textit{nonsingular} and its inverse \( A^{-1} \) has the following eigendecomposition

\[
A^{-1} = QA^{-1}Q^{-1}
\]  

(A.10)

where \( \Lambda^{-1}_{ii} = \frac{1}{\lambda_i} \).

\section*{A.3. Matrix Derivatives}

Derivatives of the elements of an inverse matrix:

\[
\frac{\partial}{\partial \lambda} A^{-1} = -A^{-1} \frac{\partial A}{\partial \lambda} A^{-1}
\]  

(A.11)

where \( \frac{\partial A}{\partial \lambda} \) is a matrix of elementwise derivatives. For the log determinant of a positive definite symmetric matrix we have

\[
\frac{\partial}{\partial \lambda} \log |A| = \text{Tr} \left( A^{-1} \frac{\partial A}{\partial \lambda} \right).
\]

(A.12)

\section*{A.4. Gaussian Identities}

Let us suppose we have a multivariate Gaussian vector with mean vector \( \mu \in \mathbb{R}^N \) and \( N \times N \) covariance matrix \( \Sigma \), \( X \sim \mathcal{N}(\mu_x, \Sigma) \), which has joint probability density given by

\[
p(x) = \frac{1}{\sqrt{(2\pi)^N |\Sigma|}} \exp \left( -\frac{1}{2} (x - \mu_x)^T \Sigma^{-1} (x - \mu_x) \right) .
\]

(A.13)

Thus, let us consider \( Y \) and \( Z \) be jointly Gaussian random vectors,

\[
\begin{bmatrix} Y \\ Z \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} \mu_y \\ \mu_z \end{bmatrix}, \begin{bmatrix} A & C \\ C^T & B \end{bmatrix} \right),
\]

then the \textit{marginal} distribution of \( Y \) and the \textit{conditional} distribution of \( Y \) given \( Z \) are given by

\[
Y \sim \mathcal{N}(\mu_y, A),
\]

(A.14)

\[
Y | Z \sim \mathcal{N}(\mu_y + CB^{-1}(Z - \mu_z), A - CB^{-1}C^T),
\]

(A.15)

see e.g. [1].
B. Appendix B

The nonlinear conjugate gradient method is an adaptation of the linear conjugate gradient method to nonlinear convex function $f$. Therefore, let us briefly see the latter approach to then discover the former. It is taken from J. Nocedal and S. Wright 2006 [6].

B.1. The Linear Conjugate Gradient Method

The conjugate gradient method is an iterative method used to solve linear systems of equations

$$Ax = b,$$

where $A$ is an $n \times n$ symmetric positive definite matrix. The problem can be stated equivalently as a minimization problem for $\phi(x)$, where,

$$\phi(x) := \frac{1}{2}x^T Ax - b^T x,$$

and (B.1) and (B.2) have the same unique solution. We notice that

$$\nabla \phi(x) = Ax - b =: r(x),$$

hence, in particular, at $x = x_k$ we have

$$r_k = Ax_k - b.$$  

**Definition B.1.** A set of nonzero vectors $\{p_0, \ldots, p_d\}$ is said to be conjugate with respect to the symmetric definite positive matrix $A$ if

$$p_i^T Ap_j = 0, \text{ for all } i \neq j.$$  

If a set of vectors has the conjugacy property than they are also linear independent [6]. Then, given a starting point $x_0 \in \mathbb{R}^n$ and a set of conjugate vectors $\{p_0, \ldots, p_{n-1}\}$, let us generate the sequence $\{x_k\}$ by setting

$$x_{k+1} = x_k + \alpha_k p_k,$$

where $\alpha_k$ is the one-dimensional minimizer of $\phi(\cdot)$ along $x_k + \alpha_p$, given by

$$\alpha_k = -\frac{r_k^T p_k}{p_k^T Ap_k}.$$  

51
So, we have that a sequence of iterates generated as in (B.6) converges to the solution $x^*$ in at most $n$ steps [6].

The conjugate gradient method has the important property that in generating its sets of conjugate vectors, it can compute a new vector $p_k$ by using only the previous one $p_{k-1}$ and $p_k$ is automatically conjugate to $p_0, \ldots, p_{k-1}$. This means saving storage and little computation. In the conjugate gradient method $p_k$ is chosen such that

$$p_k = -r_k + \beta_k p_{k-1}, \quad (B.8)$$

where $b_k$ is a scalar to be determined by asking $p_k$ and $p_{k-1}$ to be conjugate with respect to $A$, namely

$$\beta_k = \frac{r_k^T A p_{k-1}}{p_{k-1}^T A p_{k-1}}. \quad (B.9)$$

Then, $p_0$ is chosen such that it is the steepest descent direction at the initial point $x_0$ and we perform successive one-dimensional minimizations along each of the search directions. If we replace $\alpha_k$ and $\beta_{k+1}$ with

$$\alpha_k = \frac{r_k^T r_k}{p_k^T A p_k} \quad (B.10)$$

$$\beta_{k+1} = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}, \quad (B.13)$$

we obtain the following standard and more efficient conjugate gradient algorithm:

**(CG) Algorithm**

Given $x_0$;
Set $r_0 \leftarrow Ax_0 - b, p_0 \leftarrow -r_0, k \leftarrow 0$;
while $r_k \neq 0$

$$\alpha_k \leftarrow \frac{r_k^T r_k}{p_k^T A p_k}; \quad (B.10)$$

$$x_{k+1} \leftarrow x_k + \alpha_k p_k; \quad (B.11)$$

$$r_{k+1} \leftarrow r_k + \alpha_k A p_k; \quad (B.12)$$

$$\beta_{k+1} \leftarrow \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}; \quad (B.13)$$

$$p_{k+1} \leftarrow -r_{k+1} + \beta_{k+1} p_k; \quad (B.14)$$

$$k \leftarrow k + 1 \quad (B.15)$$

end (while)
B.2. The Nonlinear Conjugate Gradient Method

The first nonlinear conjugate gradient method was proposed by Fletcher and Reeves 1964 [23] as follows. If we look at the previous algorithm, in place of $\alpha_k$ we need to perform a line search that identifies an approximate minimum of $f$ in the direction of $p_k$. Second, the residual $r$, which is simply the gradient of $\phi$ must be replaced by the gradient of the nonlinear objective function $f$. These modifications yield the following algorithm for nonlinear optimization.

**(FR) Algorithm**

Given $x_0$;
Evaluate $f_0 = f(x_0), \nabla f_0 = \nabla f(x_0)$;
Set $p_0 \leftarrow -\nabla f_0, k \leftarrow 0$;

while $\nabla f_k \neq 0$

Compute $\alpha_k$ and set $x_{k+1} = x_k + \alpha_k p_k$;
Evaluate $\nabla f_{k+1}$; 

$$\beta_{k+1}^{FR} \leftarrow \frac{\nabla f_{k+1}^T \nabla f_{k+1}}{\nabla f_k^T \nabla f_k};$$  \hspace{1cm} \text{(B.16)}

$$p_{k+1} \leftarrow -\nabla f_{k+1} + \beta_{k+1}^{FR} p_k;$$  \hspace{1cm} \text{(B.17)}

$$k \leftarrow k + 1$$  \hspace{1cm} \text{(B.18)}

end (while)

We need to be more precise about the choice of the line search parameter $\alpha_k$. Because of the second term in (B.16), the search direction $p_k$ may fail to be a descent direction unless $\alpha_k$ satisfies certain conditions. By taking the inner product of (B.16) with the gradient vector $\nabla f_k$, we obtain

$$\nabla f_k^T p_k = -||\nabla f_k||^2 + \beta_k^{FR} \nabla f_k^T p_{k-1}. \hspace{1cm} \text{(B.19)}$$

If the line search is exact, namely $\alpha_{k-1}$ is a local minimizer of $f$ along $p_{k-1}$, we have $\nabla f_k^T p_{k-1} = 0$, and in this case from (B.19) we have $\nabla f_k^T p_k < 0$, so that $p_k$ is indeed descent direction. If the line search is not exact, then we may have $\nabla f_k^T p_k > 0$, implying that $p_k$ is actually a ascent direction. We can avoid this imposing the Wolfe conditions [6],

$$f(x_k + \alpha_k p_k) \leq f(x_k) + c_1 \alpha_k \nabla f_k^T p_k, \hspace{1cm} \text{(B.20)}$$

$$|\nabla f(x_k + \alpha_k p_k)^T p_k| \leq -c_2 \nabla f_k^T p_k, \hspace{1cm} \text{(B.21)}$$

where $0 < c_1 < c_2 < \frac{1}{2}$. 

53
An alternative method, which often works better in practice (more efficient and robust) is that of Polak and Ribiere [6], which differs from the FR method in the choice of the parameter $\beta_k$, as follows

$$\beta_{k+1}^{PR} = \frac{\nabla f_{k+1}^T (\nabla f_{k+1} - \nabla f_k)}{||\nabla f_k||^2}.$$  \hspace{1cm} (B.22)

In Mathematica, the default conjugate gradient method is the Polak-Ribiere, but it can be chosen the Fletcher-Reeves as well. The advantage of the conjugate gradient algorithm is that it uses relatively little memory for large nonlinear optimization problems and requires no numerical linear algebra, but only the evaluation of the objective function and its gradient, and finally no matrix operations for the step computation. A drawback is that it typically converges much more slowly than Newton or quasi-Newton methods. However, back to our study, the Conjugate Gradient algorithm was the only one working for every $n$ number of observations, and for every function we needed to maximize.

Finally, one issue that can arise with nonlinear conjugate gradient methods is when to restart them. The usual procedure is to restart the iteration at every $n$ steps by setting $\beta_k = 0$ in (B.16), that is, by taking a steepest descent step. Restarting serves to periodically refresh the algorithm, erasing information that may not be beneficial. This leads to $n$-steps quadratic convergence [6], that is,

$$||x_{k+n} - x|| = O(||x_k - x^*||^2).$$  \hspace{1cm} (B.23)

An alternative, as the first one may be not practical for problems with large $n$, is to restart whenever two consecutive gradients are far from orthogonal, as measured by the test

$$\frac{|\nabla f_k^T \nabla f_{k-1}|}{||\nabla f_k||^2} \geq \nu,$$  \hspace{1cm} (B.24)

where a typical value of the threshold $\nu$ is 0.1.
Popular summary

Gaussian process regression is a useful and extremely powerful methodology which has been used in many fields, such as machine learning, classification, geostatistics, genetics and so on.

Let us consider the usual regression model used to make prediction and inference. In the Gaussian process regression framework, instead of choosing a specific predictive function $f$, such as a linear or polynomial function of the inputs, we assume that the predictive function is Gaussian distributed, and thus, we can specify a mean and a covariance function. In particular, the covariance function deeply characterizes and affects GPR performance, and hence, the predictive value, of the model. Moreover, the covariance function depends on free parameters, called hyperparameters, which need to be properly determined. The common approach is to estimate them from the training data by means of maximum marginal likelihood. A potential problem that one can encounter is the existence of multiple local maxima and therefore, the estimated hyperparameter may not be a global maximum.

This thesis, in particular, focuses on a specific and simple GPR model which depends only on two hyperparameters. This work combines theoretical and numerical study of the marginal likelihood, as function of the hyperparameters. Firstly, its asymptotic behaviour is studied and properties regarding the more general function behaviour are derived, such as its limits, equations of its asymptotes and, in a restricted case, of its ridges as well. Then, we numerically study the function, implementing optimisation algorithms in order to find the local maxima, and locate the global maximum of the function. Moreover, we run simulations in order to estimate the probability of having one, two or more local optima. Finally, the main and most important goal of this thesis is to find an automatic and systematic method to detect a number of (and possibly all) the local maxima of the marginal likelihood.
Bibliography


18. D. J. MacKay, Gaussian processes - a replacement for supervised neural networks?


