# Training Data Analysis for Gaussian Process State Space Models

submitted
BACHELOR'S THESIS

cand. ing. Patricia Ferreiro Alonso

born on the 10.08.1992
living in:
Zugspitzstr. 12
81541 Munich
Tel.: 174 - 8529210

Chair of
INFORMATION-ORIENTED CONTROL
Technical University of Munich

Univ.-Prof. Dr.-Ing. Sandra Hirche

| | |
|---|---|
| Supervisor: | M.Sc. Thomas Beckers |
| Start: | 19.05.2017 |
| Intermediate Report: | 20.07.2017 |
| Delivery: | 31.08.2017 |

**Abstract**

Gaussian Process State Space Models aim at constructing models of nonlinear dynamical systems capable of quantifying the uncertainty in their predictions. By means of sampling in a noisy environment and covariance functions, Gaussian Process regression techniques aim to infer an estimate of the underlying function as well as a probabilistic confidence interval.

Optimally choosing sample points is crucial for system identification and control as it conforms, together with the prior knowledge, all the information available to approach the inference problem. The error between the real system and the estimation, as well as its probabilistic confidence interval, directly depend on a measure of the true function complexity, the maximum information gain and the number and distribution of the training data for a given kernel. However, a closed-form solution for the aforementioned parameters hasn't been presented in past literature.

In this work, we show proof of exact information confidence bounds for the Linear Kernel and derive a connection between its parameters and the most informative subset of sample points. We derive closed forms for the information maximization problem, thus avoiding a non-linear optimization problem and significantly reducing the computational load. We also compute the true function's norm in its associated Reproducing Kernel Hilbert Space and use it as a measure of complexity of the true function. Finally, we obtain a unique sample point distribution that ensures both minimal sample variance and maximum information gain for the Linear Kernel. Additionally, a similar intuition is developed for the Gaussian Kernel, computing the true function norm in terms of its Fourier transform and deriving a similar connection between the sample point distribution and the tightest confidence bounds.

# Contents

# Chapter 1

# Introduction

System identification aims to build models of dynamical systems based on empirical measures. While numerous methods and algorithms of machine learning and statistics have been developed for the identification of linear dynamic systems, such as ARX or ARMAX models [SP14], treating nonlinear systems requires using more sophisticated approaches. Volterra series [Fla63], neural networks[ZLB13] [YO10] or NARMAX models [POTO01] [Bil13] are some of the most popular ones and present, however, strong limitations arising from the difficulty of parameterizing the complex physics laws that govern them. Nonparametric regression methods, such as Gaussian Process Regression (GP-R), are not as restrictive as parametric models and offer a more flexible framework for accurately estimating unknown nonlinearities.

Nowadays, Gaussian Process State Space Model (GP-SSM) emerges as an upcoming model identification technique for complex nonlinear time invariant systems, such as human motion[WFH08] or gas-liquid separation[LK07]. The most important advantage of GPP-SM is the prediction of the variance, which contains information about the uncertainty of the identification and allows the computation of confidence bounds. Smoothness assumptions and previous knowledge about the true function is encoded in a covariance function and imposed on the inference model by using a Gaussian Process as prior. By means of Bayesian optimization, a posterior distribution of the objective function is computed, with its probabilistic nature revealing the uncertainty of the estimation, thus providing useful feedback for an iterative model fitting. This offers a powerful tool for nonlinear function regression in scenarios where little previous knowledge is available, and makes the GP-SSM a strong candidate for the analysis of different control applications based on system models, such as predictive and adaptive control.

Nevertheless, the fundamental system dynamics of GP-SSM are sparsely research, with recent contributions on GP-SSM stability [BH16b] and equilibrium distributions [BH16a], as most publications focus on computational issues, such as the derivation of more efficient sampling methods, like Particle Markov Chain Monte

Carlo [ADH10] [FLSR13] [TDR10] [ENDH] or the design of experimental design criteria for near-optimal sensor placement [KSGW05] and sensor calibration [GYCW15].

Inference techniques always rely on obtaining information from a real system. As the true function is to be inferred from the empirically measured data, an optimal design of the experiments is crucial in order to obtain the most informative sets of available data in an efficient manner. However, the measured data is always inaccurate in some way, due to the stochastic nature of variables and the presence of noise. Thus, the selection of an optimal subset from a given interval of possible sampling locations is a crucial task to be addressed in all model identification scenarios. In this work, we aim to analyze how the true function estimation error is affected by the number and distribution of the training points, as well as to analyze the accuracy of various confidence bounds for different prior functions.

## 1.1 Problem Statement

The problem of efficiently distributing sample points is crucial for system identification and control as it conforms, together with the prior knowledge, all the information available to approach the inference problem .

In particular, we want to extract information from as few samples as possible by using mutual information maximization as optimization criteria, that is, seeking to find training points distributions that are the most informative about unsensed locations. This criteria directly measures the effect of sample distribution on the posterior uncertainty of the Gaussian Process and allows us to analyze the system behavior depending on the number and the position of the training points. Consequently, the error between the real system and the GP regression, as well as its probabilistic confidence interval directly depends on the training data.

Our goal is thus to establish and characterize the connection between the model error and the amount and the distribution of the training data. For this, we perform the identification of linear and nonlinear systems by use of appropriate kernel methods, assess the quality of its estimation by comparing the properties of the true system and the GP-SSM. More specifically, we want to measure the certainty of our model's estimation by probabilistically bounding the difference of the estimate mean and the true function.

In particular, we work with the information related expression derived by Srinivas [SKKS12]:

$$Pr\{\forall N, \forall x \in \mathcal{D}, |\mu_N(x) - f(x)| \leq \beta_{N+1}^{1/2}\sigma_N(x)\} \geq 1 - \delta$$

where $\beta$ and $\sigma_N(x)$ are the confidence parameter and sample point variance, depending exclusively on the kernel choice and true function complexity measure.

In order to obtain the confidence parameter $\beta$, we provide closed forms for the determinant maximization problem needed, thus avoiding a non-linear optimization problem and significantly reducing the computational load. We also compute the true function's norm in its kernel associated Reproducing Kernel Hilbert Space (RKHS), to be used as a measure of complexity of $f(\mathbf{x})$. Similarly, we are interested in globally minimizing $\sigma_N(x)$, as well as in deriving a connection of that minimal point's variance and point distribution for the maximum information gain for different common kernels.

## 1.2 Related Work

In inference scenarios, there are two main challenges to be approached: the estimation of an unknown function $f(\mathbf{x})$ from samples drawn from a noisy environment, and the optimization of the obtained estimate function over some high-dimensional input space. For the former, much progress has been made through the study of kernel methods [HSS08] [Cap08] and GP-R [KMSRL03] [RW05]. More recent contributions on Bayesian optimization address one of the main issues in GP-R, the tuning of kernel hyper-parameters for complex models and algorithms in machine learning, robotics, and computer vision [CSPD14] [WA13]. Even to this date, the choice of an appropriate kernel and the adjustment of its hyper-parameters for each inference scenario remains an open problem in control engineering and statistics, and isn't considered in this work.

GP-SSM models aim at constructing models of nonlinear dynamical systems capable of quantifying the uncertainty in their predictions. For its successful application, the study of fundamental system properties for control such as equilibrium distributions and stability properties [BH16b] [BH16a] has been carried out in recent works.

How to optimally distribute sampling points has also been widely investigated during the last decade and different methods such as the minimal energy principle or the Fisher information matrix [GKS05]. Predominant approaches to this problem include the multi-armed bandit paradigm [BS92], where the goal is to maximize cumulative reward by optimally balancing exploration and exploitation [Aue03], and experimental design [CV95], where the function is to be explored globally with as

few evaluations as possible, for example by maximizing information.

The probabilistic terms of its estimation being one of the greatest advantages of using GP-SSM models, it is just natural that confidence bounds have become a field of major study. Multiple algorithms for an optimal sampling in terms of information and accuracy have been studied, most notably the information-related bounds presented by Srinivias [SKKS12] which based on an K-armed Bandit setting have served as basis for deriving sharper regret bounds [dFSZ12b] [CBRV13] [dFSZ12a] [DKB14].

However, a closed-form solution for any of the different parameters that model the confidence bound: sample point distribution for the maximum mutual information gain, true function norm in RKHS and minimum sample points covariance, hasn't been presented in past literature. Instead, the main focus has remained the optimization of numerical procedures and the design of exploration-exploitation algorithms and experiment layouts that lead to computationally expensive sub-optimal solutions.

In this work, we present proof of exact information confidence bounds for the Linear Kernel (LK) and derive a connection between its parameters and the most informative subset of sample points. We derive closed forms for the information maximization problem, thus avoiding a non-linear optimization problem and significantly reducing the computational load. We also compute the true function's norm in its associated RKHS and use it as a measure of complexity of the true function. Finally, we obtain a unique sample point distribution that ensures both minimal sample variance and maximum information gain for the LK. Additionally, a similar intuition is developed for the Gaussian Kernel (GK), computing the true function norm in terms of its Fourier transform and deriving similar sample point distribution effect on the confidence bounds.

# Chapter 2

# Regression with Gaussian Processes

Gaussian Process Regression Models are nonparametric kernel-based probabilistic models. Given any inference scenario where a true function $\mathbf{f} = f(\mathbf{x})$ is to be estimated, and for an arbitrary number of sample points $N$ we can write:

$$\mathcal{D} = \{(\boldsymbol{x}_n, y_n) |\ n = \{1, \ldots, N\}\ , \boldsymbol{x}_n \in \mathcal{X} \subseteq \mathbb{R}^n, \mathbf{y}_n \in \mathbb{R}\} \tag{2.1}$$

where $\mathbf{x}_n$ is an input vector of dimension $D$ and $y_n$ is the corresponding target value containing an evaluation of $\mathbf{f}$ corrupted by noise. The column vector inputs for all $n$ cases are aggregated in the $D \times n$ design matrix $\mathbf{X}$ so we can rewrite equation 2.1 as $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$.

In order to ultimately be able to infer the distribution over all possible functions $f : \mathcal{X} \to \mathbb{R}$, the first step is to predict responses $\mathbf{y}_*$ for any new inputs $\mathbf{x}_*$. By using a Bayesian approach, when performing regression with Gaussian Processes, we treat the true function $\mathbf{f}$ as a random function and introduce our knowledge about its expected behavior by placing a prior over it in terms of the covariance function or kernel. By means of sampling, we obtain a set of data points which we then use to update the prior distribution by placing a Gaussian posterior over the true function.

Hence, GP-R uses too main tools for inference: the kernel function or covariance $k(\mathbf{x}, \mathbf{x}')$, which encodes previous knowledge or assumptions about the true function behavior, and the sample points $\mathbf{x}_n$, which are used to condition the prior and enhance the fitting of the underlying distribution by means of a likelihood function.

## 2.1 Gaussian Process Regression Models

A Gaussian Process is a set of random variables, any finite number of which have a joint Gaussian distribution. Hence, for an arbitrary input vector $\mathbf{x} \in \mathbb{R}^n$, a GP is completely specified by its mean and covariance functions, which are in turn parametrized by a set of kernel parameters known as hyper-parameters.

We define the mean function $m(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$ of any real GP $f(\mathbf{x})$ as follows:

$$m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})] \tag{2.2}$$

$$k(\mathbf{x}, \mathbf{x}') = cov(f(\mathbf{x}), f(\mathbf{x}')) = \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}')] \tag{2.3}$$

and express the GP as:

$$f(\mathbf{x}) \sim \mathcal{GP}((m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \tag{2.4}$$

If $f(\mathbf{x})$ is a GP and given $N$ observations $\{x_1, x_2, ..., x_N\}$, the joint distribution of the random variables is also Gaussian:

$$[f(x_1), f(x_2), \cdots, f(x_n)]^\top \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}) \tag{2.5}$$

where the mean $N$-column vector $\boldsymbol{\mu}$ has entries of the form $\mu_n = m(x_n)$ and the $N \times N$ covariance matrix $\mathbf{K}$ has entries of the form $K_{nm} = k(x_n, x_m)$. The embedding of important properties of the true function, such as differentiability or periodicity, are determined by the choice of the kernel function, which is briefly discussed in section 2.4.

Let us consider the training set $\mathbf{x}_n \in \mathbb{R}^n$ with $n = \{1, 2, \cdots, N\}$ corresponding to samples drawn from an unknown objective function $\mathbf{f}$. A GP-R model aims to predict the value of the response variable $\mathbf{y}_*$ given a new input vector $\mathbf{x}_*$ by making use of the previously obtained training data. Note that our goal is the characterization of the relationship between inputs $\mathbf{x}_n$ and targets $\mathbf{y}_n$, that is, we want to obtain the conditional distribution of the targets given the inputs, rather than in modeling the input distribution itself [RW05].

The Bayesian analysis of the standard linear regression model corrupted with Gaussian noise is modeled as:

$$\mathbf{y} = \mathbf{x}^\top \boldsymbol{\beta} + \varepsilon \tag{2.6}$$

with $\boldsymbol{\beta} = [\beta_1, \beta_2, \cdots, \beta_p] \in \mathbb{R}^p$ and where we have assumed that the observed values $\mathbf{y}$ differ from the function values $\mathbf{f} = \mathbf{x}^\top \boldsymbol{\beta}$ by an additive noise that in turn follows an independent Gaussian distribution with zero mean and variance $\sigma_n^2$ such that

$\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$. The error variance $\sigma_n^2$ characterizing the noise and the real coefficients $\boldsymbol{\beta}$ are both estimated though the data obtained by sampling.

A GP-R model characterizes the response by drawing samples from the GP and defining explicit basis functions $\boldsymbol{\phi} = \{\phi(x_1), \phi(x_1), \cdots, \phi(x_P)\} : \mathcal{X} \to \mathbb{R}^p$ such that:

$$f(x) = \sum_{i=0}^{P} \beta_i \phi_i(x) \tag{2.7}$$

The covariance function captures the smoothness of the response and the basis functions project the inputs $\mathbf{x}_n$ into a higher dimension feature space.

By combining this noise assumption with the linear regression model, and as the probability density of the Gaussian observations also follows a Gaussian distribution, the following likelihood function can be derived:

$$p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}) = \prod_{n=1}^{N} p(y_n|\mathbf{x}_n, \boldsymbol{\beta}) = \prod_{n=1}^{N} \frac{1}{\sqrt{2\pi}\sigma_n} \exp\left(-\frac{\left(y_n - \mathbf{x}_n^\top \boldsymbol{\beta}\right)^2}{2\sigma_n^2}\right) \tag{2.8}$$

$$= \left(\frac{1}{\sqrt{2\pi}\sigma_n}\right)^n \exp\left(-\frac{\left|\mathbf{y} - \mathbf{X}^\top \boldsymbol{\beta}\right|^2}{2\sigma_n^2}\right) = \mathcal{N}(\mathbf{X}^\top \boldsymbol{\beta}, \sigma_n^2 \mathbf{I}) \tag{2.9}$$

As we are suing a Bayesian approach, we need to specify a prior over the parameters, thus expressing our beliefs about their behavior before taking into consideration any observations. We place a zero mean Gaussian prior with covariance matrix $\Sigma_p$ on the weights $\boldsymbol{\beta}$:

$$\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_p) \tag{2.10}$$

In the Bayesian linear model, inference is based on the posterior distribution over the weights, which can be computed using Bayes rule:

$$p(\boldsymbol{\beta}|\mathbf{y}, \mathbf{X}) = \frac{p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}) p(\boldsymbol{\beta})}{p(\mathbf{y}|\mathbf{X})} \tag{2.11}$$

where the normalizing constant, also known as the marginal likelihood, is independent of the weights and given by:

$$p(\mathbf{y}|\mathbf{X}) = \int p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}), p(\boldsymbol{\beta}) \mathrm{d}\boldsymbol{\beta} \tag{2.12}$$

where the posterior $p(\boldsymbol{\beta}|\mathbf{y}, \mathbf{X})$ combines the likelihood and the prior, capturing knowledge about the parameters.

By considering only the weight-dependent terms from the likelihood and prior we obtain:

$$p(\mathbf{y}|\mathbf{X},\boldsymbol{\beta}) \propto \exp\Big(-\frac{1}{2\sigma_n^2}(\mathbf{y}-\mathbf{X}^\top\boldsymbol{\beta})^\top(\mathbf{y}-\mathbf{X}^\top\boldsymbol{\beta})\Big)\exp(-\frac{1}{2}\boldsymbol{\beta})^\top\boldsymbol{\Sigma}_p^{-1}\boldsymbol{\beta}) \tag{2.13}$$

$$\propto \exp\left(\left(-\frac{1}{2}(\boldsymbol{\beta}-\overline{\mathbf{w}})^\top\right)\left(\frac{1}{\sigma_n^2}\mathbf{X}\mathbf{X}^\top+\boldsymbol{\Sigma}_p^{-1}\right)\left(-\frac{1}{2}(\boldsymbol{\beta}-\overline{\mathbf{w}})\right)\right) \tag{2.14}$$

where $\overline{\mathbf{w}} = \sigma_n^{-2}(\sigma_n^{-2}\mathbf{X}\mathbf{X}^\top + \Sigma_p^{-1})^{-1}\mathbf{X}\mathbf{y}$.

Note the posterior distribution is also a Gaussian with mean $\overline{\mathbf{w}}$ and covariance matrix $\mathbf{C}_{\mathbf{ov}}^{-\mathbf{1}}$ such that:

$$p(\mathbf{y}|\mathbf{X},\boldsymbol{\beta}) \sim \mathcal{N}(\overline{\mathbf{w}}, \mathbf{C}_{\mathbf{ov}}^{-\mathbf{1}}) \tag{2.15}$$

where $C_{ov} = \sigma_n^{-2}\mathbf{X}\mathbf{X}^\top + \Sigma_p^{-1}$ so that $\overline{\mathbf{w}} = \sigma_n^2\mathbf{C}_{\mathbf{ov}}^{-\mathbf{1}}\mathbf{X}\mathbf{y}$.

To make predictions for a specific input point $\mathbf{x}_*$ we average over all possible parameter predictive distribution values weighted by their posterior probability.

The predictive distribution for $f_* \triangleq f(\mathbf{x}_*)$ is given by the average of all possible linear model outputs with respect to the Gaussian posterior:

$$p(f_*|\mathbf{x}_*,\mathbf{X},\mathbf{y}) = \int p(f_*|\mathbf{x}_*,\boldsymbol{\beta})p(\mathbf{w}|\mathbf{X},\mathbf{y})\mathrm{d}\boldsymbol{\beta} \tag{2.16}$$

$$= \mathcal{N}(\sigma_n^2\mathbf{x}_*^\top\mathbf{C}_{\mathbf{ov}}^{-\mathbf{1}}\mathbf{X}\mathbf{y}, \mathbf{x}_*^\top\mathbf{C}_{\mathbf{ov}}^{-\mathbf{1}}\mathbf{x}_*) \tag{2.17}$$

Thus, the predicted distribution is also Gaussian and of mean given by the posterior mean of the weights $\boldsymbol{\beta}$ multiplied by the input point $\mathbf{x}_*$.

### 2.1.1 Inference in the projected Feature Space

The Bayesian linear model is too simple to express complex functions. A very simple and common approach to overcome this issue is to project the input data into a higher dimension, more tractable feature space in which the linear model can be applied. A projection into this feature space can be defined by using an appropriate set of basis functions and as long as the projections are independent of the weights $\boldsymbol{\beta}$, the model keeps its linearity and thus is still analytically tractable.

We assume such feature functions are given as $\phi(\mathbf{x})$, which map a $D$-dimensional input vector $\mathbf{x}$ into an $N$ dimensional feature space.

Let the matrix $\phi(\mathbf{X})$ be the aggregation of columns $\phi(\mathbf{x})$ for all cases in the training set. Now the model can be written as:

$$f(\mathbf{x}) = \phi(\mathbf{x}) \tag{2.18}$$

where the vector of parameters now has length $N$.

The analysis for this model is analogous to the standard linear model, except that everywhere $\Phi(\mathbf{X})$ is substituted for $\mathbf{X}$. Thus, the predictive distribution becomes:

$$f_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y} \sim \mathcal{N}\left(\frac{1}{\sigma_n^{-2}}\phi(\mathbf{x}_*)^\top \mathbf{C_{ov}^{-1}}\Phi(\mathbf{X})\mathbf{y}, \phi(\mathbf{x})_*^\top \mathbf{C_{ov}^{-1}}\phi(\mathbf{x}_*)\right) \tag{2.19}$$

with $C_{ov} = \sigma_n^{-2}\Phi(\mathbf{X})\Phi(\mathbf{X})^\top + \mathbf{\Sigma_p^{-1}}$.

In order to make predictions using equation 2.19, we need to invert the $\mathbf{C_{ov}}$ matrix of size $N \times N$ which may not be convenient if $N$, the dimension of the feature space, is large. However, by defining $K = \Phi^\top \mathbf{\Sigma}_p \Phi$, where $\Phi = \Phi(\mathbf{X})$ predictions can be computed as follows:

$$f_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y} \sim \mathcal{N}(\phi_*^\top \mathbf{\Sigma}_p \Phi(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y},\ \phi_*^\top \mathbf{\Sigma}_p \phi_* - \phi_*^\top \mathbf{\Sigma}_p \Phi(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\Phi^\top \mathbf{\Sigma}_p \phi_*) \tag{2.20}$$

where we have simplified the notation with $\phi_*(\mathbf{x}) = \phi_*$.

Let us define $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \mathbf{\Sigma}_p \phi(\mathbf{x}')$ with $k(\cdot, \cdot)$ being a covariance function or kernel and $\mathbf{x}$, $\mathbf{x}'$ being either the training or tests input points.

Because $\phi(\mathbf{x})^\top \mathbf{\Sigma}_p \phi(\mathbf{x}')$ is an inner product and $\mathbf{\Sigma}_p$ is a positive definite matrix, we can write $\mathbf{\Sigma}_p = (\mathbf{\Sigma}_p^{1/2})^2$. By defining $\psi(\mathbf{x}) = \mathbf{\Sigma}_p^{1/2}\phi(\mathbf{x})$ we can observe that a dot product of the form $k(\mathbf{x}, \mathbf{x}') = \psi(\mathbf{x})\psi(\mathbf{x}')$ results from directly evaluating the kernel. This will allow us to avoid complex computations of dot products, specially when working with large datasets, as they can be performed in the higher dimension input space as simple kernel evaluations.

More specifically, if an algorithm is defined solely in terms of inner products in its input space, then it can be expanded into a feature space by the explicit computation of the required inner products by $k(\mathbf{x}, \mathbf{x}')$. This method is also known as the "kernel-trick".

### 2.1.2 Inference in Function Space

We can also obtain the results shown in section 2.1.1 by directly performing inference in the function space.

### Prediction with Noise-free Observations

Initially, we consider all observations to be noise free. The joint prior distribution of the training outputs $\mathbf{f}$ and the test outputs $\mathbf{f}_*$ according to the prior is:

$$\begin{bmatrix}\mathbf{f}\\ \mathbf{f}_*\end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix}K(\mathbf{X},\mathbf{X}) & K(\mathbf{X},\mathbf{X}_*)\\ K(\mathbf{X}_*,\mathbf{X}) & K(\mathbf{X}_*,\mathbf{X}_*)\end{bmatrix}\right) \tag{2.21}$$

where for $n$ training points and $n_*$ test points, $K(\mathbf{X},\mathbf{X}_*)$ is the $n \times n_*$ matrix of the covariances for all pairs of training and test points.

In order to obtain the posterior distribution over all possible underlying functions, we impose the sample points over the joint prior distribution, thus filtering out all candidate functions that do not contain the empirically observed points. Mathematically, this is equivalent to conditioning the joint Gaussian prior distribution on the observations:

$$\begin{aligned}\mathbf{f}_*|\mathbf{X}_*,\mathbf{X},\mathbf{f} \ \sim \mathcal{N}\big(&K(\mathbf{X}_*,\mathbf{X})K(\mathbf{X},\mathbf{X})^{-1}\mathbf{f},\\ &K(\mathbf{X}_*,\mathbf{X}_*) - K(\mathbf{X}_*,\mathbf{X})K(\mathbf{X},\mathbf{X})^{-1}K(\mathbf{X},\mathbf{X}_*)\big)\end{aligned} \tag{2.22}$$

Thus, the objective function evaluations $\mathbf{f}_*$ associated with inputs $\mathbf{X}_*$ can then be sampled from the joint posterior distribution by simply evaluating the mean and covariance matrix.

### Prediction using Noisy Observations

We consider the more realistic setting where only noisy samples are available, i.e. $\mathbf{y} = \mathbf{f} + \varepsilon$. Assuming additive and independent Gaussian noise of the form $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$, the prior on the noisy observations becomes:

$$cov(\mathbf{y}) = \mathbf{K}(\mathbf{X}^\top\mathbf{X} + \sigma_n^2\mathbf{I}) \tag{2.23}$$

By adding the noise $\sigma_n^2\mathbf{I}$ in equation 2.21, we obtain the joint distribution of the observed and true function values at the input points $\mathbf{X}_*$:

$$\begin{bmatrix}\mathbf{y}\\ \mathbf{f}_*\end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix}\mathbf{K}(\mathbf{X},\mathbf{X}) + \sigma_n^2\mathbf{I} & \mathbf{K}(\mathbf{X},\mathbf{X}_*)\\ \mathbf{K}(\mathbf{X}_*,\mathbf{X}) & \mathbf{K}(\mathbf{X}_*,\mathbf{X}_*)\end{bmatrix}\right) \tag{2.24}$$

Deriving the conditional distribution we obtain the predictive equations for Gaussian Process regression:

$$\mathbf{f}_*|\mathbf{X},\mathbf{y},\mathbf{X}_* \sim \mathcal{N}(\bar{\mathbf{f}}_*, var(\mathbf{f}_*)) \tag{2.25}$$

where the true function can be expressed as:

$$\overline{\mathbf{f}}_* \triangleq \mathbb{E}[\mathbf{f}_*|\mathbf{X}, \mathbf{y}, \mathbf{X}_*] = \mathbf{K}(\mathbf{X}_*, \mathbf{X})[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_n^2\mathbf{I}]^{-1}\mathbf{y} \tag{2.26}$$

and has a covariance of the form:

$$cov(\mathbf{f}_*) = \mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) - \mathbf{K}(\mathbf{X}, \mathbf{X}_*)[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_n^2\mathbf{I}]^{-1}\mathbf{K}(\mathbf{X}, \mathbf{X}_*) \tag{2.27}$$

Now for any set of basis functions $\boldsymbol{\phi}$ we can compute the corresponding covariance function as $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)\boldsymbol{\Sigma}_p\phi(\mathbf{x}_j)$. Analogously, every positive definite covariance function $k$can be expressed in terms of a set of basis functions.

Using a compact notation and for a single test point $x_*$:

$$\overline{\mathbf{f}}_* = \mathbf{k}_*^\top(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y} \tag{2.28}$$

The mean prediction is a linear combination of observations $\mathbf{y}$ or, similarly, a linear combination of $n$ kernel functions, each one evaluated on a training point. This can be explicitly expressed by writing:

$$\overline{f}(\mathbf{x}_*) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}_*) \tag{2.29}$$

where $\alpha = (\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y}$. The fact that the mean prediction for $\mathbf{f}_*$ can be written as a linear combination of basis functions evaluations is a consequence of the representer theorem, discussed in section 1.

## 2.2 Gaussian Process State Space Models

GPS-SM are characterized by a $n$-dimensional state vector $\mathbf{x}_k \in \mathcal{X}$:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k), \quad k \in N \tag{2.30}$$
$$\mathbf{y}_k = \mathbf{x}_k + \epsilon_k \tag{2.31}$$
$$f(\mathbf{x}_k) \sim \mathcal{GP}(m(\mathbf{x}_k), \mathbf{k}(\mathbf{x}_k, \mathbf{x'}_k)) \tag{2.32}$$
$$(\mathbf{x}_k, \mathbf{x'}_k)) \tag{2.33}$$
$$\sigma_k \sim \mathcal{N}(\mathbf{0}, diag(\sigma_{1,n}^2, \cdots, \sigma_{n,n}^2)) \tag{2.34}$$

A $n$-dimensional system must be modeled by $n$ GPs, so the vector valued function $\mathbf{m}(\cdot) = [m_1(\cdot), \cdots, \mathbf{m}_n(\cdot)]^\top$ contains the mean estimates for each component of the vector state $\mathbf{x}_{k+1}$. The Gaussian Process for each state is given by:

$$f(\mathbf{x}_k) = \begin{cases} f_1(\mathbf{x}_k) \sim \mathcal{GP}(m(\mathbf{x}_k), \mathbf{k}_{\varphi_1}(\mathbf{x}_k, \mathbf{x'}_k)) \\ \vdots \quad \vdots \\ f_n(\mathbf{x}_k) \sim \mathcal{GP}(m_n(\mathbf{x}_k), \mathbf{k}_{\varphi_n}(\mathbf{x}_k, \mathbf{x'}_k)) \end{cases} \tag{2.35}$$

with $\varphi_i$ being the set of hyper-parameters.

The training data is $\mathcal{D} = X, Y$ where $X = [\tilde{x}_1, \cdots, \tilde{x}_m]$ contains the $m$ inputs and $Y = [\tilde{y}_1, \cdots, \tilde{y}_m]$ contains the $m$ outputs.

The prediction for each component of the next state $\mathbf{x}_{i,k+1}$ is computed as a Gaussian random variable of mean $\mu(\mathbf{x}_{i,k+1}|\mathbf{x}_{i,k,\mathcal{D}})$. The joint distribution of the $i$-th component of the predicted next step is:

$$\begin{bmatrix} \mathbf{Y}_{:,\mathbf{i}} \\ \mathbf{x}_{i,k+1} \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K_{\varphi_i}(X,X) & \mathbf{k}_{\varphi_i}(\mathbf{x}_k, X) \\ \mathbf{k}_{\varphi_i}(\mathbf{x}_k, X)^\top & k_{\varphi_i}(\mathbf{x}_k, \mathbf{x}_k) \end{bmatrix}\right). \tag{2.36}$$

where $Y_{:,i}$ is the $i$-th column of the matrix $Y$, $K_{\varphi_i}(X,X)$ the covariance matrix and $\mathbf{k}_{\varphi_i}(\mathbf{x}_k, X)$ the vector-valued extended covariance function with hyper-parameters $\varphi_i$.

$$K_{\varphi_i}(X,X) : \mathcal{X}^m \times \mathcal{X}^m \to \mathbb{R}^{m \times m} \tag{2.37}$$
$$K_{j',j} = k_{\varphi_i}(\mathbf{X}_{:,j'}, \mathbf{X}_{:,j}) \tag{2.38}$$
$$\mathbf{k}_{\varphi i}(\mathbf{x}_k, X) : \mathcal{X} \times \mathcal{X}^m \to \mathbb{R}^m, k_j = k_{\varphi i}(\mathbf{x}_k, X_{:,j}) \tag{2.39}$$
$$\forall j', j \in 1, \cdots, m, i \in \{1, \cdots, n\} \tag{2.40}$$

Assuming all GP functions are of mean zero, a prediction for the $i$-th component of $\S_{k+1}$ is derived from the joint distribution as a Gaussian conditional probability distribution of mean:

$$\mu_i(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathcal{D}) = \mathbf{k}_{\varphi i}(\mathbf{x}_k, X)^\top \mathbf{h}(i) \tag{2.41}$$

where $\mathbf{h}(i) = (K_{\varphi i} + \mathbf{I}\sigma^2_{n,i})^{-1} Y_{:,i}$.

The variance of the prediction is of the form:

$$var_i(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathcal{D}) = k_{\varphi i}(\mathbf{x}_k, \mathbf{x}_k)) - \mathbf{k}_{\varphi i}(\mathbf{x}_k, X)^\top (K_{\varphi i}) + \mathbf{I}\sigma^2_{n,i})^{-1} \mathbf{k}_{\varphi i}(\mathbf{x}_k, X) \tag{2.42}$$

where $\sigma^2_{n,i} \in \mathbb{R}$ is the standard deviation of the noise of the input data for all $i \in \{1, \cdots, m\}$. The set of hyper-parameters $\varphi_i$ are optimized by means of the likelihood function, thus by maximizing the probability of:

$$\varphi_i^* = \arg\max_{\varphi_i} log\ P(Y_{:,i}|X, \varphi_i) \tag{2.43}$$

The $n$ normally distributed components of $x_{i,k+1}|\mathbf{x}_k, \mathcal{D}$ are combined in a multivariable Gaussian distribution:

$$\mathbf{x}_{k+1}|\mathbf{x}_k, \mathcal{D} \sim \mathcal{N}(\boldsymbol{\mu}(\cdot), \epsilon(\cdot)) \tag{2.44}$$
$$\mu_i(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathcal{D}) = [\mu_1(\cdot), \cdots, \mu_n(\cdot)]^\top \tag{2.45}$$
$$\epsilon(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathcal{D}) = diag(var_1(\cdot), \cdots, var_n(\cdot)) \tag{2.46}$$

Hence, the whole system can be rewritten as an affine stochastic system with state dependent noise:

$$\mathbf{x}_{k+1} = \mu_i(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathcal{D}) + \epsilon(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathcal{D})\boldsymbol{\eta} \tag{2.47}$$

with $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, I)$.

## 2.3 The Reproducing Kernel Hilbert Space

In order to avoid overfitting, we want to choose $f$ to be as smooth as possible. The norm of the RKHS, which define spaces of functions, is extremely useful to encode this criterion. In addition, the use of RKHS offers many other advantages, as under some restrictions, results and algorithms for linear models in Euclidean spaces can be generalized and applied [Ros10]. The RKHS main three properties of interest are:

- The reproducing property:

$$\forall x \in \mathcal{X}, \forall f \in \mathcal{H}_k,\ f(x) = \langle f(\cdot), k(\cdot, x)\rangle_k, \tag{2.48}$$

  where $\langle \cdot, \cdot \rangle_k$ is the inner product defined in space $\mathcal{H}_k$, the associated RKHS to the kernel $k(\cdot, \cdot)$.

- Functions in a RKHS can be expressed as a linear combination of the kernel $k(\cdot, \cdot)$ evaluated at given points:

$$f(x) = \sum_i \alpha_i k(x_i, x) \tag{2.49}$$

- The squared norm in a RKHS can be written as:

$$||f||_k^2 = \sum_{i,j} \alpha_i \alpha_j k(x_i, x_j) \tag{2.50}$$

  and can be viewed as a measure of complexity or smoothness of function $f$.

Let $\mathcal{X}$ be an arbitrary set and $\mathcal{H}_k$ a Hilbert space of real-valued functions on $\mathcal{X}$. The evaluation functional over the Hilbert space of functions $\mathcal{H}_k$ is a linear functional that evaluates each function at a point $x$:

$$L_x : f \to f(x),\ \forall f \in \mathcal{H}_k \tag{2.51}$$

We say that $\mathcal{H}_k$ is a RKHS if $\forall x \in \mathcal{X}$, $L_x$ is continuous at any $f \in \mathcal{H}_k$.

A RKHS is a Hilbert space $\mathcal{H}_k$ of functions defined by a symmetric, positive-definite function $k : X \times X \to \mathbb{R}$ called the reproducing kernel, such that $k(x, \cdot) \in \mathcal{H}_k$ for

$\forall x \in \mathcal{X}$.

A more intuitive definition of the RKHS can be obtained by observing that this property guarantees that the evaluation functional can be represented by taking the inner product of $\mathbf{f}$ with a function $K_x$ in $\mathcal{H}_k$. More formally, the Riesz representation theorem[Goo70] implies that for $\forall x \in \mathcal{X}$ there a unique element $K_x$ of $\mathcal{H}_k$ exists and holds:

$$f(x) = L_x(f) = \langle f, K_x \rangle, \ \forall f \in \mathcal{H}_k \tag{2.52}$$

which is known as the reproducing property.

Since $K_x$ is itself a function in $\mathcal{H}_k$, it holds that for $\forall y \in \mathcal{X}$ there exist a $K_y \in \mathcal{H}_k$ such that:

$$K_x(y) = \langle K_x, K_y \rangle \tag{2.53}$$

This allows us to define the reproducing kernel $\mathcal{H}_k$ as a function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$:

$$K(x, y) = \langle K_x, K_y \rangle \tag{2.54}$$

From this definition it is easy to see $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is both symmetric and positive definite.

### 2.3.1 RKHS norm and smoothness

Some kernels of interest, such as the Gaussian, are invariant. This implies we can evaluate them with a single argument as:

$$K(x, y) = K(x - y) \tag{2.55}$$

We can define its Fourier series representation:

$$K(x) = \sum_{\omega=-\infty}^{\infty} \hat{K}_\omega exp(i\omega x) \tag{2.56}$$

with $K$ and its Fourier transform $\hat{K}_\omega$ being real and symmetric.

Define $\mathcal{H}$ to be the space of functions with an possibly infinite feature space representation of the form:

$$f(\cdot) = \begin{bmatrix} \cdots & \frac{\hat{f}_\omega}{\sqrt{\hat{K}_\omega}} & \cdots \end{bmatrix}^\top \tag{2.57}$$

The space $\mathcal{H}$ has an inner product:

$$\langle f, g\rangle_{\mathcal{H}} = \sum_{\omega=-\infty}^{\infty} \frac{\overline{\hat{f}}_\omega \hat{g}_\omega}{\left(\sqrt{\hat{K}_\omega}\right)\left(\sqrt{\hat{K}_\omega}\right)} = \sum_{\omega=-\infty}^{\infty} \frac{\overline{\hat{f}}_\omega \hat{g}_\omega}{\left(\hat{K}_\omega\right)} \tag{2.58}$$

So the feature map can be defined as:

$$K(\cdot, x) = \phi(x) = \left[\cdots \sqrt{\hat{K}_\omega exp(-i\omega x)} \cdots\right]^\top \tag{2.59}$$

We can check that the reproducing property holds:

$$\langle f(\cdot), K(\cdot, x)\rangle_{\mathcal{H}} = \sum_{\omega=-\infty}^{\infty} \frac{\hat{f}_\omega \overline{\sqrt{\hat{K}_\omega} exp(-i\omega x)}}{\sqrt{\hat{K}_\omega}} = \sum_{\omega=-\infty}^{\infty} \hat{f}_\omega exp(i\omega x) = f(x) \tag{2.60}$$

By using the inner product definition and for a given kernel $K$, we can write the squared norm of a function $f$ in its associated RKHS $\mathcal{H}_k$ as:

$$\|f\|^2_{\mathcal{H}} = \langle f, f\rangle_{\mathcal{H}} = \sum_{i=-\infty}^{\infty} \frac{\hat{f}_i^2}{\hat{k}_l}. \tag{2.61}$$

where $\hat{f}$ is the Fourier transform of $f$. If $||f||_{\mathcal{H}} < \infty$, then $f$ belongs to the Hilbert space $\mathcal{H}_k$. We can see that $\hat{f}_i^2$ has to decrease quicker than $\hat{k}_i$ for the sum to converge so that $||f||^2_{\mathcal{H}_k} < \infty$.

### 2.3.2 Mercer's theorem

Mercer's theorem [Mer09] derives a strong connection between the eigenvalues and eigenfunctions associated with the kernel. More specifically, it provides a representation of any symmetric positive-definite function $K(\cdot, \cdot) \in L_2(X \times X)$ as a sum of a convergent sequence of product functions.

**Theorem 1** (Mercer's Theorem)**.**
*Let $X \in \mathbb{R}^n$ be a finite set $\{x\}$ and $K : X \times X \to \mathbb{R}$ be a symmetric, non-negative definite, continuous function. A countable sequence of eigenfunctions $\{\phi\}$, i.e. $K\phi = \lambda\phi$, and a sequence of associated eigenvalues $\lambda \in \mathbb{R}^+$ exist such that $K(\cdot, \cdot)$ can be expressed as:*

$$K(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^{\infty} \lambda_i \phi_i(\mathbf{x}) \phi_i(\mathbf{x}')$$

Thus, Mercer kernels [Her02] can be decomposed in terms of the set of basis functions corresponding to the non-zero eigenvalues. This ensures that every zero mean GP is defined by an RKHS such that orthonormal eigenfunctions $\{\phi_i\}$ with associated eigenvalues $\{\lambda_i\}$ exist and are given by:

$$k(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^{\infty} \lambda_i \phi_i(\mathbf{x}) \phi_i(\mathbf{x}') \tag{2.62}$$

Mercer's theorem itself is a generalization of the result that any positive semidefinite matrix is the Gramian matrix of a set of vectors and ensures the existence of a feature map $\phi : X \to K \in l_2$. Thus, expanding the previous expression:

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') &= \sum_{i=1}^{\infty} \lambda_i \psi_i(\mathbf{x}) \psi_i(\mathbf{x}') \\ &= (\sqrt{\lambda_1}\psi_1(\mathbf{x}), \sqrt{\lambda_2}\psi_2(\mathbf{x}), \cdots)^\top (\sqrt{\lambda_1}\psi_1(\mathbf{x}'), \sqrt{\lambda_2}\psi_2(\mathbf{x}'), \cdots) \end{aligned} \tag{2.63}$$

where $\psi(\mathbf{x}) = \{\psi_1(\mathbf{x}), \psi_2(\mathbf{x}), \cdots\}$ is known as the Mercer map. With this, for any given kernel that satisfies Mercer's conditions, we can define a feature space $\mathcal{K}$ and a feature map $\phi : X \to \mathcal{K} \in l_2$, with $\phi(\mathbf{x}) = (\sqrt{\lambda_1}\psi_1(\mathbf{x}), \sqrt{\lambda_2}\psi_2(\mathbf{x}), \cdots)^\top$.

Note that because $\mathcal{H}_k$ is a function space, we note that it may be infinite-dimensional. While this is not a problem in theory, it does pose a computational problem.[Ros10]

## 2.4 Kernel selection for GP Regression

In order to build accurate models we want to encode some of the underlying true function main characteristics, such as symmetry and periodicity. We can achieve that by defining a suitable positive definite covariance kernel, an operator that determines the similarity between all pairs of points in the given domain.

Hence, choosing the kernel that best reflects the prior knowledge and the assumptions about the true function is by no means trivial, and greatly determines the performance properties of a GP model as well as the quality of its estimation and confidence bounds [Kri14]. Ultimately, the choice of the kernel and its regularization parameters can be automated to a certain degree by combining Bayesian inference with cross-validation techniques.

### 2.4.1 Hyper-parameter optimization

In Bayesian statistics, a hyper-parameter is a parameter of a prior distribution. This term has been coined to distinguish the characterization of the prior from the parameters of the model for inferring the underlying system. Altogether with the

covariance function, they encode prior information about the underlying system and its selection can dramatically impact the accuracy of the inference algorithm.

Once an appropriate kernel has been chosen, the next step involves the tuning of its hyper-parameters, which generally implies solving a constrained optimization problem. Their optimal value for a particular data set can be automatically estimated by maximizing the log marginal likelihood using standard optimization methods.

The posterior over the parameters is given by Bayes' rule:

$$p(\boldsymbol{\beta}|\mathbf{y}, X, \sigma, \mathcal{H}_i) = \frac{p(\mathbf{y}|X, \boldsymbol{\beta}, \mathcal{H}_i)p(\boldsymbol{\beta}|\sigma, \mathcal{H}_i)}{p(\mathbf{y}|X, \sigma, \mathcal{H}_i)} \tag{2.64}$$

where $p(\mathbf{y}|X, \boldsymbol{\beta}, \mathcal{H}_i)$ is the likelihood and $p(\boldsymbol{\beta}|\sigma, \mathcal{H}_i)$ is the prior. The prior encodes as a probability distribution the knowledge about the parameters before seeing the data. For example, if we only have vague prior information about the parameters, a broad prior distribution should be chosen. The posterior combines the information from the prior and the data through the likelihood.

The normalizing constant in the denominator, $p(\mathbf{y}|X, \sigma, \mathcal{H}_i)$ is the marginal likelihood, which is independent of the parameters and of the form:

$$p(\mathbf{y}|X, \sigma, \mathcal{H}_i) = \int p(\mathbf{y}|X, \boldsymbol{\beta}, \mathcal{H}_i)p(\boldsymbol{\beta}|\sigma, \mathcal{H}_i)d\beta \tag{2.65}$$

At the next level, we analogously express the posterior over the hyper-parameters, where the marginal likelihood from the first level plays the role of the level 2 inference likelihood:

$$p(\sigma|X, \mathcal{H}_i) = \frac{p(\mathbf{y}|X, \sigma, \mathcal{H}_i)p(\sigma|\mathcal{H}_i)}{p(\mathbf{y}|X, \sigma, \mathcal{H}_i)}, \tag{2.66}$$

where $p(\mathbf{y}|X, \sigma, \mathcal{H}_i)$ is the prior for the hyper-parameters with a normalizing constant of:

$$p(\mathbf{y}|X, \mathcal{H}_i) = \int p(\mathbf{y}|X, \sigma, \mathcal{H}_i)p(\sigma|\mathcal{H}_i)d\sigma \tag{2.67}$$

However, due to computational expensiveness, a good approximation is the maximization of the first level marginal likelihood over the hyper-parameters $\sigma$. For a Gaussian prior as in GP-R:

$$log\ p(\mathbf{y}|X, \boldsymbol{\sigma}) = -\frac{1}{2}\mathbf{y}^\top K_y^{-1}\mathbf{y} - \frac{1}{2}log(|K_y|) - \frac{d}{2}log(2\pi) \tag{2.68}$$

where $K_y = K_f + \sigma_n^2\mathbf{I}$ is the covariance matrix for the noisy targets $\mathbf{y}$ and $K_f$ is the covariance matrix for the noise-free latent $\mathbf{f}$. The only term involving the observed targets is the data-fit $\frac{1}{2}\mathbf{y}^\top K_y^{-1}\mathbf{y}$; $\frac{1}{2}log(|K_y|)$ is the complexity penalty depending

only on the covariance function and the inputs and $\frac{d}{2}log(2\pi)$ is a normalization constant.

Bayesian optimization starts on assuming a general prior over a set of possible functions, which combined with observations allow the derivation of hyper-parameters.

Usually, hyper-parameters are conditioned to optimizing a certain measure of the algorithm's performance. This restrictions, plus marginal likelihood computations and cross-validation techniques are used iteratively to obtain generally sub-optimal estimates of the kernel function [RW05]. Hyper-parameter optimization remains an open problem that, due to its complexity and extension, is excluded from the scope of this work.

Taking into account that kernels encode our prior information about the true function, some general guidelines can be followed regarding its choice. However, hyper-parameter optimization must still be addressed for optimal results [RW05].

In the following sections, some commonly used basic kernels, together with which kind of behavior from the true function they expect are presented.

We established earlier that if a space of functions can be represented as an RKHS, it has useful properties. More specifically, the inner product and the ability for each function to be evaluated continuously at any arbitrary point. In the following section we describe specific examples of RKHS and examine how their different norms provide different forms of regularization.

### 2.4.2 Linear Kernel

The Linear Kernel captures strongly linear trends and can be expressed as:

$$k_L(\mathbf{x}, \mathbf{x}') = x^\top x', x, x' \in \mathbb{R}^n \tag{2.69}$$

Its associated RKHS is the dual space consisting of functions $f(\mathbf{x}) = \langle x, w \rangle$ of norm:

$$\|f\|_k^2 = \|w\|^2 \tag{2.70}$$

so that our measure of complexity is the slope of the line.

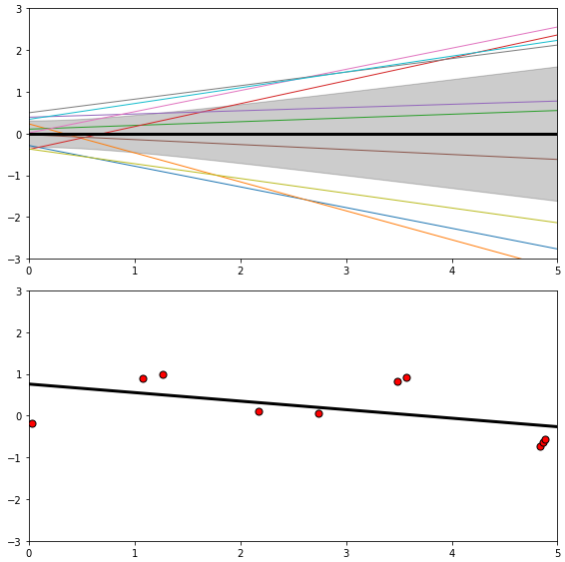

Figure 2.1: Prior and posterior GP Regression using a Linear Kernel

### 2.4.3 Polynomial Kernel

$$k_P(\mathbf{x}, \mathbf{x}') = (x^\top x' + \alpha)^\beta, x, x' \in \mathbb{R}^n, \beta \in \mathbb{R}^+ \tag{2.71}$$

To aid in a proper fitting, the Linear Kernel is commonly combined with exponentiation in order to conform a Polynomial Kernel. For example, for $\alpha = 0$ ans $\beta = 2$:

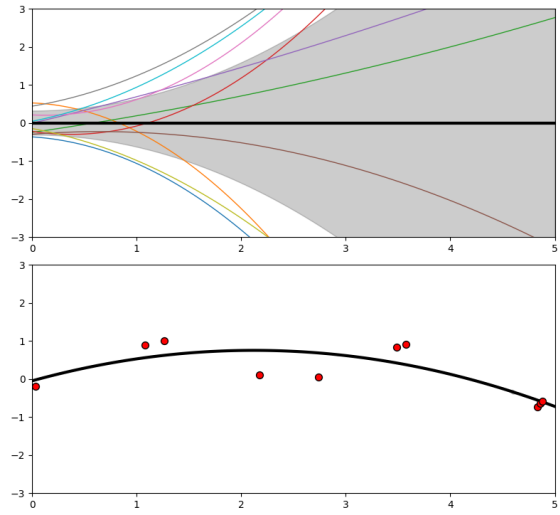

Figure 2.2: Prior and posterior GP Regression using a Squared Polynomial Kernel

### 2.4.4 Gaussian Kernel

Also known as Radial Basis Function (RBF) or Squared Exponential kernel, inputs are weighted such that closer variables are highly correlated, whilst those far away are uncorrelated. It is of the form:

$$k_G(\mathbf{x}, \mathbf{x}') = exp\left( - \frac{||(\mathbf{x} - \mathbf{x}')||^2}{2\ell^2} \right), x, x' \in \mathbb{R}^n, \ell \in \mathbb{R}^+ \tag{2.72}$$

where $\ell \in \mathbb{R}^+$ is the lengthscale hyper-parameter.

In more detail, $\ell$ determines the degree of smoothness in the function, that is, it is a measure of how quickly the Gaussian Process varies when the input $\mathbf{x}$ changes. Hence, a bigger $\ell$ would prioritize low complexity over prediction accuracy, resulting in a very smooth function which may overlook the shape of the underlying true function, thus incurring in underfitting. On the contrary, a smaller choice of $\ell$ would strongly prioritize the fitting of small fluctuations in the data, possibly caused by noise. Consequently, the inferred function would present a too noisy outline due to overfitting.

Recalling that the definition of the RKHS norm in terms of $f$'s Fourier transform $\hat{f}$ is:

$$\|f\|^2_{\mathcal{H}} = \langle f, f \rangle_{\mathcal{H}} = \sum_{n=-\infty}^{\infty} \frac{\hat{f}_n^2}{\hat{k}_n} \tag{2.73}$$

where $\hat{k}$ is in turn the Fourier transform of the kernel associated with the RKHS.

Since a Gaussian function has another Gaussian as Fourier transform, we can ensure $f \in \mathcal{H}_k$. Hence, a Gaussian Kernel has a RKHS norm of the form:

$$||f||^2_k = \frac{1}{2\pi^d} \int |F(\omega)|^2 \exp \frac{\sigma^2 \omega^2}{2} \, d\omega \tag{2.74}$$

which penalizes harshly high-frequency components and where $F(\omega)$ is now the Fourier transform of $f$ [Ros10].

More specifically, $|F(\omega)|$ decreases exponentially with an increase of $|\omega|$ so for an arbitrary function $f$ to be contained in this space its bounded Fourier transform has to decrease even faster than that of the kernel [Moo28]. By analyzing this behavior in the frequency domain, we can conclude that $f$ has most of its spectrum consisting of high amplitude, low frequency components, in contrast with weak or non-present higher frequency contributions. Low frequency in $|F(\omega)|$ implies smoothness in $|F(\omega)|^{-1}$. Hence, a Gaussian kernel covariance is always be best used for the inference of smooth functions, even though this complexity can be tuned to a certain

degree by accordingly modifying the lengthscale hyper-parameter.

Thus, the Gaussian Kernel presents very useful properties, such as being stationary and having very smooth sample functions, which makes them infinitely differentiable and results in posterior distributions of the following kind:

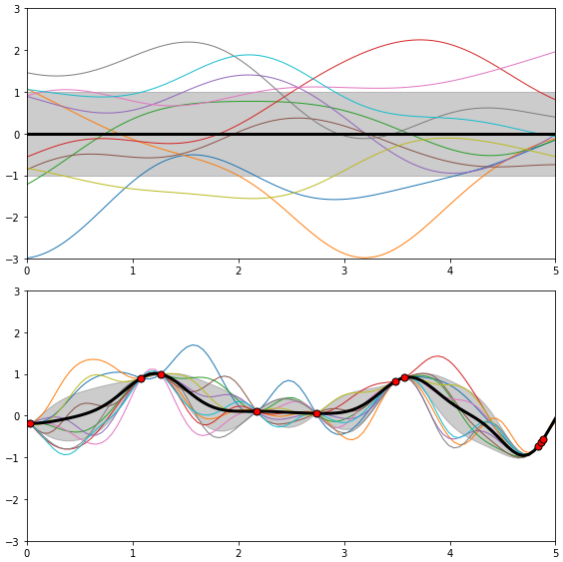

Figure 2.3: Prior and posterior GP Regression using a Gaussian Kernel

### 2.4.5 Rational Quadratic Kernel

This kernel is equivalent to adding together multiple SE kernels with different lengthscales. Hence, such as prior would expect true functions that vary smoothly across various lengthscales. It is of the form:

$$k_{RQ}(\mathbf{x}, \mathbf{x}') = \sigma^2 exp\left( - \frac{(\mathbf{x} - \mathbf{x}')^2}{2\alpha\ell^2} \right)^{-\alpha} \tag{2.75}$$

where the parameter $\alpha$ determines the relative weighting of large and small-scale variations. It can be easily observed, that when $\alpha \to \infty$, the RQ corresponds to the Squared Exponential kernel and thus its estimates also have a smooth behavior:

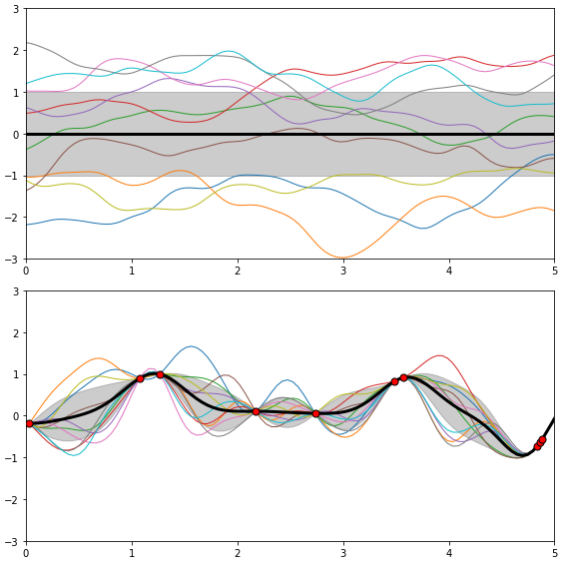

Figure 2.4: Prior and posterior GP Regression using a Rational Quadratic Kernel

### 2.4.6 Sine Exponential Periodic Kernel

Allows capturing periodic behavior. This kernel corresponds to the space of band limited functions $f \in L^2(\mathbb{R})$ with bandwidth $2a$.

$$k_P(\mathbf{x}, \mathbf{x}') = \sigma^2 exp\Big\{ -\frac{1}{2} \sum_i \frac{sin\ a(x_i - x_i')^2}{\ell_i} \Big\} \tag{2.76}$$

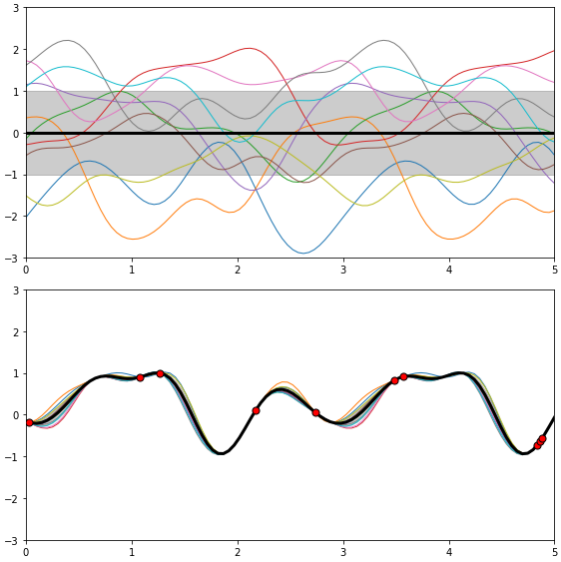

Figure 2.5: Prior and posterior GP Regression using a Sine Exponential Kernel

# Chapter 3

# Confidence bounds for GP Regression

## 3.1 Optimal Bayesian experimental design

One of the key points of statistical inference is the fact that there is a probability associated to the estimation. In particular, this probability bears information about the accuracy of the obtained estimation and is given by the confidence interval. In other words, it can be thought of as a measure of how good our system model is, and is thus used as feedback for its iterative improvement by means of an increase or alternative distribution of the sample points or a better kernel hyper-parameter tuning.

We are concerned with GP optimization where $f$ is sampled from a GP distribution or has low "complexity" measured in terms of its RKHS norm under some kernel. Thus, we study the regret bounds in this nonparametric setting and how the distribution of the sample points affects the estimation error.

Our goal is to understand how the choice and distribution of the sampling points influences the confidence intervals of a non-linear Gaussian regression setting. In order to study this behavior, we use the Information-related bound presented on "Information-Theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting" [SKKS12].

In particular, we work with Theorem 6:

**Theorem 2** (Theorem 6, Srinivas)**.** *Let us choose $\delta \in (0, 1)$ and assume that the noise variables $\epsilon_t$ are uniformly bounded by $\sigma$, i.e. $\forall t \in \mathcal{D},\ y_t = f(x_t) + \epsilon_t \leq f(x_t) + \sigma$. We can define probabilistic confidence intervals as:*

$$Pr\{\forall N, \forall x \in \mathcal{D}, |\mu_N(x) - f(x)| \leq \beta_{N+1}^{1/2} \sigma_N(x)\} \geq 1 - \delta \tag{3.1}$$

*where $\beta$ is a confidence parameter of form:*

$$\beta_t = 2||f||_k^2 + 300\gamma_n ln^3\Big(\frac{t}{\delta}\Big) \tag{3.2}$$

*where $||f||_k$ is the true function's norm in the kernel's associated RKHS and $\gamma_N$ is the maximum information gain obtained by maximizing the mutual information $I(y_A; f_A)$, which can be computed as:*

$$\gamma_N = \max_{A\in\mathcal{D}:|A|=N} I(y_A; f_A) = \max_{A\in\mathcal{D}:|A|=N} \left(\frac{1}{2} log|\mathbf{I} + \sigma^{-2}\mathbf{K}_N|\right) \tag{3.3}$$

### 3.1.1 D-Optimality: Mutual Information as criteria

To quantify how informative an individual sampling is, we use the criterion of mutual information, which for an unknown function $f$ and a set of sample points $A \in \mathcal{D}$ of form $\mathbf{y}_A = \mathbf{f}_A + \epsilon_A$:

$$I(\mathbf{y}_A; f) = H(\mathbf{y}_A) - H(\mathbf{y}_A|f) \tag{3.4}$$

quantifying the reduction in uncertainty about $f$ from revealing $\mathbf{y}_A$, with $\mathbf{f}_A = [f(\mathbf{x})]_{\mathbf{x}\in A}$ and $\epsilon_A \sim N(\mathbf{0}, \sigma^2\mathbf{I})$.

For a Gaussian distribution, the entropy can be expressed as:

$$H(N(\mu, \mathbf{\Sigma})) = |\frac{1}{2} log\mathbf{I} + \sigma^{-2}\mathbf{K}_N| \tag{3.5}$$

which allows us to rewrite the information gain for our setting as:

$$I(\mathbf{y}_A; f) = I(\mathbf{y}_A; \mathbf{f}_A) = \frac{1}{2} log|\mathbf{I} + \sigma^{-2}\mathbf{K}_N| \tag{3.6}$$

where $\mathbf{K}_N = [k(\mathbf{x}, \mathbf{x}')]_{\mathbf{x},\mathbf{x}'\in A}$.

This criterion expresses the expected reduction of global entropy for all locations where we didn't sample, after taking into account the information given by the selected sample points.

While finding the information gain maximizer among $A \in \mathcal{D}$, $|A| \leq T$ is NP-hard [KLQ95], we aim to find closed expressions for particular kernel choices. A general sub-optimal result can be attained by an efficient greedy algorithm. If $F(A) = I(\mathbf{y}_A; f)$, the algorithm chooses $\mathbf{x}_t = argmax_{x\in\mathcal{D}} F(A_{t-1} \cup \{\mathbf{x}\})$ in round $t$, that is:

$$\mathbf{x}_t = argmax_{\mathbf{x}\,\in\,\mathcal{D}}\ \sigma_{t-1}(\mathbf{x}) \tag{3.7}$$

where $A_{t-1} = \{\mathbf{x}_1, \cdots, \mathbf{x}_{t-1}\}$. It is guaranteed to find a near-optimal solution [SKKS12].

Besides avoiding the finite-dimensional analysis, we must handle confidence issues, which are more delicate for nonlinear random functions. Importantly, note that the information gain is a problem dependent quantity - properties of both the kernel and the input space determines the growth of regret. This theorem shows that, with high probability over samples from the GP, the cumulative regret is bounded in terms of the maximum information gain, thus directly connecting GP optimization and experimental design. Moreover, the sub-modularity of $I(\mathbf{y}_A; \mathbf{f}_A)$ allows the derivation of sharp a priori bounds, also depending on the kernel type and hyper-parameter estimation.

The smoothness assumption on $k(\mathbf{x}, \mathbf{x}')$ disqualifies GPs with highly erratic sample paths. It holds for stationary kernels which are four times differentiable.

Note that in our case, since we train the GP offline, there are no rounds but a fixed number of available sample points $N$.

We define $\delta$ in order to obtain the desired probability, e.g. $\delta = 0.1$ for a 90% probability. Then, for $\forall x \in \mathcal{D}$ and with our given $N$ sample points, we need to compute the following:

- Information gain $\gamma_N$, which depends on $N$ and on the kernel choice (in particular, on the kernel's eigenvalues)
- $||f||_k^2$ bound in the kernel's associated RKHS, as the function generator is not known

Note $|\mu_N(\mathbf{x}) - f(\mathbf{x})|$, where $\mu_T(\mathbf{x})$ is the posterior mean function. This value is obtained by simulation once the confidence bounds for the specific inference setting are computed.

In our study we use the squared-exponential kernel:

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\tfrac{||\mathbf{x}-\mathbf{x}'||^2}{(2\ell^2)})$$

Then, for $\mathcal{D} \in \mathbb{R}^n, n \in \mathbb{N}$ and assuming $k(\mathbf{x}, \mathbf{x}') \leq 1$:

$$\gamma_N = \mathcal{O}((logT)^{d+1}))$$

Even though knowing how $\gamma_n$ behaves is useful for a general understanding of the algorithm, it is too loose to be used as an assumption. Hence, based on the definition of information gain:

$$\gamma_N := \max_{A \in \mathcal{D}:|A|=N} I(y_A; f_A)$$

where $I(\mathbf{y}_A; f_A) = I(\mathbf{y}_A; f)$ and $I(\mathbf{y}_A; f) = \frac{1}{2} log|\mathbf{I}+\sigma^{-2}\mathbf{K}_A|$, with $\mathbf{K}_n = [k(\mathbf{x}, \mathbf{x}')]_{\mathbf{x},\mathbf{x}' \in A}$ being the covariance matrix or kernel of $f_A = [f(\mathbf{x})]_{\mathbf{x} \in A}$ associated with the $N$ samples contained in $A$. Note that the properties of both the kernel and the input space influence how the regret grows.

### 3.1.2 Computation of the maximum information gain

We want to directly obtain:

$$\gamma_N = \underset{A \in \mathcal{D}:|A|=N}{max} \frac{1}{2} log|\mathbf{I} + \sigma^{-2}\mathbf{K}_N|$$

where $\sigma$ is the sampling noise variance from the Gaussian Process prior described by $\mathcal{N} \sim (0, \sigma^2)$.

Even though finding the information gain maximizer among $A \in \mathcal{D} : |A| \leq N$ is NP-hard [KLQ95], it can be approximated by an efficient greedy algorithm, as proposed by Srinivas [SKKS12]

However, we want to train the GP offline and directly compute the confidence bounds for a fixed amount of available sampling points, so for each specific case we can simply compute:

$$\gamma_N = \frac{1}{2} log|\mathbf{I} + \sigma^{-2}\mathbf{K}_N|$$

Hence, $\gamma_N$'s value is determined by the covariance matrix $\mathbf{K}_N$, which is in turn defined by the sample points and the type of kernel chosen, and by the variance $\sigma^2$. For a noisy sample $\mathbf{y}_n = \{y_1, \cdots, y_N\}^T$ at points $\mathbf{x}_n = \{x_1, \cdots, x_N\}$, so that $y_n = f(x_n) + \epsilon_n$ with $\epsilon_n \sim \mathcal{N}(0, \sigma^2)$ i.i.d. Gaussian noise, the posterior over $f$ is a GP distribution with mean $\mu_N(\mathbf{x})$, covariance $k_N(\mathbf{x}, \mathbf{x}')$ and variance $\sigma^2(\mathbf{x})$.
Having defined a fixed number of sample points $N \in \mathcal{D}$, we can compute the maximum information gain attainable. The problem can be tackled as a non-convex optimization, where $\gamma_N$ is the function to maximize for which a discrete subset $\mathbf{x}_n$ is chosen from a continuous bounded interval.

An iterative approach was implemented using the interior point method [Ren01], but the quality of the result depends greatly on the initial conditions, that is, on the set of sample points selected. Its choice is not trivial, and even under restrictions we can't guarantee that the absolute minimum is reached.

Hence, a more theoretical approach has been decided in order to find a closed form for the maximum information gain, for which we need to work directly with the determinant of the covariance matrix $\mathbf{K}_N$:

$$\gamma_N = \max_{A \in \mathcal{D}:|A|=N} \frac{1}{2} log|\mathbf{I} + \sigma^{-2}\mathbf{K}_N|$$

As this is a too generic approach, we particularize the computation of the confidence bounds for the linear and Gaussian kernels.

## 3.2 Computation of the confidence bounds

For a fixed $\delta \in (0,1)$ and $|\epsilon_n| \leq \sigma$, the following confidence parameter is defined:

$$\beta_n = 2||f||_k^2 + 300\gamma_n ln^3\left(\frac{n}{\delta}\right) \tag{3.8}$$

The main problem now is how to accurately compute the true function's norm in its associated RKHS, $||f||_k$.

In general, since $f$ is unknown, we cannot address this question directly, but aim to obtain an upper bound for the norm. For the time being, let us consider $||f||_k < B$.

$$\beta_n = 2||f||_k^2 + 300\gamma_n ln^3\left(\frac{n}{\delta}\right) \leq 2B^2 + 300\gamma_n ln^3\left(\frac{n}{\delta}\right) \tag{3.9}$$

Note that we are only considering functions with low complexity in RKHS, that is, smooth or with a small norm. This implies that, working within an ideal setting, the norm's contribution to the confidence parameter $\beta$ could be significantly smaller than the rest of expression, which is in turn tied with the parameters $\gamma_n$, $\delta$ and the number and distribution of the selected samples $N$.

As the focus of our study is the derivation of exact bounds, we assume the true function is known and attempt to compute its RKHS norm under the assumption of a good kernel choice. Let us discuss once again the Linear and Gaussian kernels.

### 3.2.1 Confidence bounds for the Linear Kernel

#### 3.2.1.1 Maximum Information Gain for the Linear Kernel

Starting with the generic known maximum information gain expression, we now particularize it for the Linear Kernel $k(x,y) = x^\top y$:

$$\gamma_{N_{LK}} = \max_{A \in \mathcal{D}:|A|=N} \frac{1}{2} log|\mathbf{I} + \sigma^{-2}(\mathbf{x}\mathbf{x}^\top)| \tag{3.10}$$

We can disregard the constant and the logarithm, as it is a monotonic increasing function that won't affect the optimization result. Hence, we work with:

$$\underset{A\in\mathcal{D}:|A|=N}{max}|\mathbf{I}+\sigma^{-2}(\mathbf{x}\mathbf{x}^\top)| =$$

$$\underset{A\in\mathcal{D}:|A|=N}{max}\left|\begin{pmatrix}1&0&\cdots&0&0\\0&1&\cdots&0&0\\\vdots&\vdots&\ddots&\vdots&\vdots\\0&0&\cdots&1&0\\0&0&\cdots&0&1\end{pmatrix}+\sigma^{-2}\begin{pmatrix}\|x_1\|^2&x_1x_2&\cdots&x_1x_{N-1}&x_1x_T\\x_2x_1&\|x_2\|^2&\cdots&x_2x_{N-1}&x_2x_T\\\vdots&\vdots&\ddots&\vdots&\vdots\\x_{N-1}x_1&x_{N-1}x_2&\cdots&\|x_{N-1}\|^2&x_{N-1}x_T\\x_Tx_1&x_Tx_2&\cdots&x_Tx_{N-1}&\|x_T\|^2\end{pmatrix}\right|$$

To help us gain intuition, we first consider to have only 2 samples so that $\mathbf{x} = [x_1, x_2]$. The matrix of which we want to obtain the determinant now looks like:

$$\left|\begin{pmatrix}1&0\\0&1\end{pmatrix}+\sigma^{-2}\begin{pmatrix}\|x_1\|^2&x_1x_2\\x_2x_1&\|x_2\|^2\end{pmatrix}\right| = \left|\begin{pmatrix}1+\sigma^{-2}\|x_1\|^2&\sigma^{-2}x_1x_2\\\sigma^{-2}x_2x_1&1+\sigma^{-2}\|x_2\|^2\end{pmatrix}\right|$$

$$= \big(1+\sigma^{-2}\|x_1\|^2)(1+\sigma^{-2}\|x_2\|^2)\big) - \sigma^{-4}\big((x_1x_2)(x_2x_1)\big) = 1+\sigma^{-2}(\|x_1\|^2+\|x_2\|^2)$$

It can be easily seen that this last expression is maximized when $x_1$ and $x_2$ have the biggest module, that is, when the sampling is done as far as possible from the origin of coordinates. This also implies that $x_1 = x_2$, which means we sample the same point twice.

Our aim is now gaining an intuitive understanding as to why sampling the same point repeatedly is the optimum approach, always in terms of maximizing the information gain. By using a Linear Kernel, our prior knowledge about the true function is that it behaves linearly. This strong bias influences our estimation the most so we already have information on how the function behaves. Under this circumstances we are much more interested in obtaining the gradient of the true function, which is greatly affected by noise. A small $\epsilon$ variation in $y_n = f(x_n) + \epsilon_n$ can dramatically affect our estimation's accuracy, and hence we concentrate sampling in the same point as an attempt to reduce noise as much as possible.

The next natural step would be increasing the number of samples. Let us increase the input vector by one in order to analyze the 3-dimensional case, i.e. now $\mathbf{x} = [x_1, x_2, x_3]$. We are again interested in distributing the sampling points so the information gain is maximized and, in particular, in finding a generalization for $n$-dimensions. Following the results from the 2 sample points case, we want to know if there are any limitations as when to cluster sampling points steadily increases $\gamma_N$.

Computing the determinant:

$$\left|\begin{pmatrix}1&0&0\\0&1&0\\0&0&1\end{pmatrix}+\sigma^{-2}\begin{pmatrix}\|x_1\|^2&x_1x_2&x_1x_3\\x_2x_1&\|x_2\|^2&x_2x_3\\x_3x_1&x_3x_2&\|x_3\|^2\end{pmatrix}\right| = 1+\sigma^{-2}(\|x_1\|^2+\|x_2\|^2+\|x_3\|^2)$$

Now that we have acquired some intuition, we attempt to generalize the result for an arbitrary number of samples. Determinant developments for $n \times n$ matrices with $n > 3$ are of exponentially increasing computational cost and becomes intractable in inference problems where the number of samples is high. However, given the special properties of the covariance matrix, we can simplify the problem in order to obtain a general result.

We first focus on analyzing the variance-covariance matrix of the samples, $\mathbf{K}_N$, which is squared, symmetric, and positive semi-definite. It can be seen that the elements of $\mathbf{K}_N$ compute the inner product of projections of all pairs of samples into a feature space, i.e. the kernel function is as a scalar product so a function $\phi(\mathbf{x})$ exists such that $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \phi(\mathbf{x})$. The matrix formed by the cross scalar products of the samples receives the name of Gram matrix, with a determinant of closed from $||x_1 \wedge ... \wedge x_n||$, where the $\wedge$ operator is the exterior product.

However, we do not want only to compute a certain determinant, but to also select the most informative subset of $N$ samples from a continuous interval $\mathcal{D} = [0, a]$. We present a closed form for the maximum information gain for the Linear Kernel and derive the most informative distribution of sample points.

**Theorem 3.** *Given a Linear Kernel $k(\mathbf{x}, \mathbf{y}) = \mathbf{x}\mathbf{y}^\top$ and a bounded continuous sampling interval $\mathcal{D} = [0, a]$, by selecting an arbitrary number of sample points $N$ such that $\forall\{\mathbf{x}, \mathbf{y}\} \in \mathcal{D}$ the information gain $\gamma_N$ is maximized as:*

$$\gamma_{N_{LK}} = \frac{1}{2} log\big(1 + \sigma^{-2} N \left\|a\right\|^2\big) \tag{3.11}$$

*that is, when sampling from points with the maximum magnitude $\left\|a\right\|^2$.*

*Proof.* This result can be proved by using a particular case of the matrix determinant lemma [Har97]. Given an invertible $n \times n$ square matrix $\mathbf{A}$ and two column vectors $\mathbf{u}$, $\mathbf{v}$ of length $n$, we can write:

$$\det(\mathbf{A} + \mathbf{u}\mathbf{v}^\top) = \det(\mathbf{A})(1 + \mathbf{v}^\top(\mathbf{A}^{-1}\mathbf{u})) \tag{3.12}$$

In our case, $\mathbf{A}$ is the identity matrix of range $n$, $\mathbf{I}_n$ and $\mathbf{v} = \mathbf{u} = \mathbf{x}$ is the vector containing the $n$ sample points, thus:

$$\det(\mathbf{I} + \mathbf{x}\mathbf{x}^\top) = \det(\mathbf{I})(1 + \mathbf{x}^\top(\mathbf{I}^{-1}\mathbf{x})) = 1 + \mathbf{x}^\top\mathbf{x} \tag{3.13}$$

as $det(\mathbf{I}) = \prod(diag(\mathbf{I})) = 1$ and $\mathbf{I}^{-1} = \mathbf{I}$.

Applying this result to the maximum information gain expression and particularizing

for the Linear Kernel, we finally obtain:

$$\gamma_{N_{LK}} = \max_{A\in\mathcal{D}:|A|=N} \frac{1}{2} log|\mathbf{I} + \sigma^{-2}(\mathbf{x}^\top \mathbf{x})| \tag{3.14}$$

$$= \max_{A\in\mathcal{D}:|A|=N} \frac{1}{2} log\big(1 + \sigma^{-2} \sum_{n=1}^{\mathrm{N}} \|\mathbf{x}_n\|^2\big) \tag{3.15}$$

$$= \frac{1}{2} log\big(1 + \sigma^{-2} N \|a\|^2\big) \tag{3.16}$$

which is the result we wanted to prove. □

We conclude the information gain $\gamma_N$ is maximum for the Linear Kernel, when repeatedly sampling the point with the biggest magnitude.

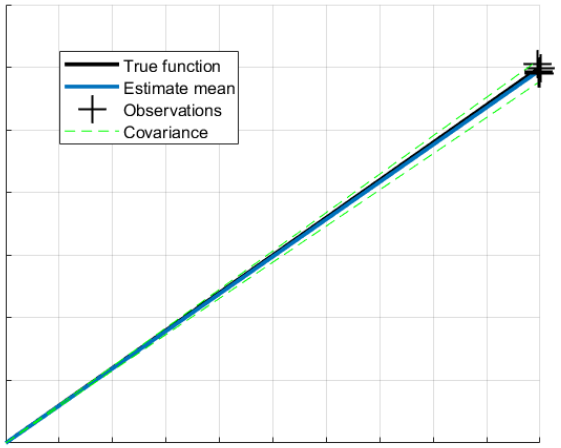


Figure 3.1: LK Most informative distribution $N = 5$ points, $\sigma_n = 0.1$, $\gamma_N = 0.01$

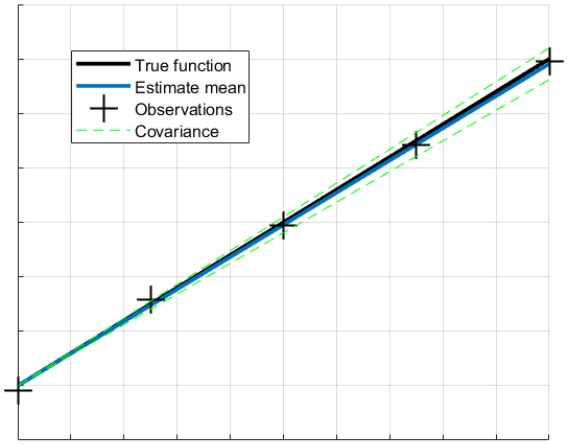


Figure 3.2: LK Uniform distribution $N = 5$ points, $\sigma_n = 0.1$, $\gamma_N = 0.004$

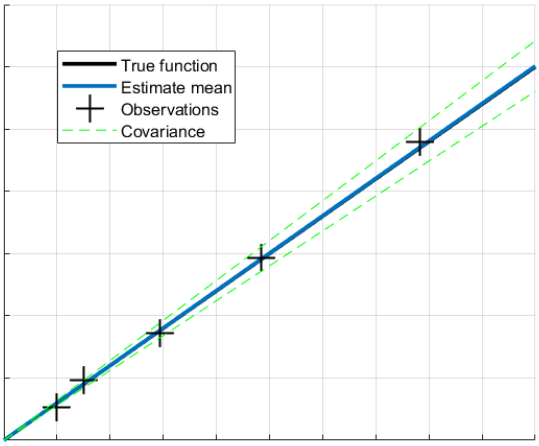


Figure 3.3: LK Random distribution $N = 5$ points, $\sigma_n = 0.1$, $\gamma_N = 0.002$

#### 3.2.1.2 RKHS norm for the Linear Kernel

For a true function with a strong linear treat, which can in turn be best inferred with a Linear Kernel, the RKHS norm is simply the slope:

$$\|\mathbf{f}\|_k^2 = \|\mathbf{w}\|^2 \tag{3.17}$$

hence, this simple parameter measures the "smoothness" or complexity of the true function. For our setting, we are assuming this measure to be low, even tough this assumption has no impact on the computation methodology.

We can now calculate the confidence bounds for the Linear Kernel for any arbitrary sample size $N$ and a selected $\delta \in (0, 1)$:

$$\beta_N = 2||\mathbf{w}||^2 + 300\gamma_N ln^3\left(\frac{N+1}{\delta}\right) \tag{3.18}$$

with $\gamma_N = \frac{1}{2}log\left(1 + \sigma^{-2}N\left\|a\right\|^2\right)$ and where $a \in \mathcal{D}$ is the sample point of greatest magnitude possible.

#### 3.2.1.3 Minimum variance for the Linear Kernel

We can now rewrite the confidence bounds expression for the Linear Kernel for $\forall x \in \mathcal{D}$ as:

$$Pr\{|\mu_N(x) - f(x)| \leq (2||w||^2 + 300\gamma_N ln^3\left(\frac{N+1}{\delta}\right))_{N+1}^{1/2}\sigma_N(x)\} \geq 1 - \delta \tag{3.19}$$

Now the only parameter left to determine is the variance of the sample points $\sigma_N^2$, which can be computed directly. We are interested in our confidence interval to

be as tight as possible, i.e. to achieve the highest model accuracy. By examining equation 3.19 we can observe that once the constant $\beta_N$ has been determined, the tightness of the interval is directly proportional to the variance of the sample points. Hence, we want to minimize it.

**Theorem 4.** *Given a Linear Kernel $k(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top \mathbf{y}$ and a compact continuous sampling interval $\mathcal{D} = [0, a]$, by selecting an arbitrary number of sample points $N$ such that $\forall\{\mathbf{x}, \mathbf{y}\} \in \mathcal{D}$ the sample variance $\sigma_N(x)$ is minimized when:*

$$\|x\|^2 = \|a\|^2, \ \forall \ \mathbf{x} \in \mathcal{D} \tag{3.20}$$

*that is, when the $N$ sample points are points of maximum magnitude $\|a\|^2$.*

*Proof.* The variance of the sample points squared can be written as a function of kernel evaluations:

$$\sigma_N^2(x) = K(x, x) - K(\mathbf{x}_N, x)^\top \left(K(\mathbf{x}_N, \mathbf{x}_N) + \sigma^2 \mathbf{I}\right)^{-1} K(\mathbf{x}_N, x) \tag{3.21}$$

where $x_N$ is the vector containing the $N$ sampled points, $K(\cdot, \cdot)$ is the kernel function and $\sigma^2$ is the noise of the Gaussian prior. Note that we evaluate the covariance for each $x$, so for each evaluation $K(\mathbf{x}_N, x)$ is a $N \times 1$ vector. It is then natural to compact the notation as:

- $K(\mathbf{x}_N, x) = \mathbf{k}_N(x)$
  A $N \times 1$ vector containing the covariance of the samples with the queried point.
- $K(x, x) = K_x$
  A scalar resulting of evaluating the kernel at the queried point.
- $K(\mathbf{x}_N, \mathbf{x}_N) = \mathbf{K}_N$
  The $N \times N$ covariance matrix of the samples.

Thus, we can rewrite the variance expression as:

$$\sigma_N^2(x) = K_x - \mathbf{k}_N(x)^\top \left(\mathbf{K}_N + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{k}_N(x) \tag{3.22}$$

Note that, in a spatial context, the minimum variance associated to a kernel would be equivalent to having the smallest correlation possible between all points within a fixed distance apart. AS $K_x$ is a constant, independent of $\mathbf{x}_N$, it can be disregarded so that minimizing $\sigma_N^2(x)$ is equivalent to maximizing the top right expression in 3.22. Hence, for $\forall x \in \mathcal{D}$:

$$\min_{\mathbf{x_N}}\{\sigma_N^2(x)\} = \max_{\mathbf{x_N}}\{\mathbf{k}_N(x)^\top \left(\mathbf{K}_N + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{k}_N(x)\} \tag{3.23}$$

More specifically, for the Linear Kernel $\mathbf{K}_N$ is the outer product of the training pints, which enables us to rewrite the previous expression as:

$$\min_{\mathbf{x_N}}\{\sigma_N^2(x)\} = \max_{\mathbf{x_N}}\{\mathbf{k}_N(x)^\top \left(\mathbf{x}_N\mathbf{x}_N^\top + \sigma^2\mathbf{I}\right)^{-1}\mathbf{k}_N(x)\} \tag{3.24}$$

We can compute the required inverse by applying the Sherman-Morrison formula[SM50]. Let an $\mathbf{A} \in \mathbb{R}^{n\times n}$ invertible square matrix and $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ be two column vectors, then we can write:

$$\left(\mathbf{u}\mathbf{v}^\top + \mathbf{A}\right)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^\top\mathbf{A}^{-1}}{1+\mathbf{v}^\top\mathbf{A}^{-1}\mathbf{u}} \tag{3.25}$$

and using that $(\sigma^2\mathbf{I})^{-1} = \frac{1}{\sigma^2}\mathbf{I}$ we obtain:

$$\left(\mathbf{x}_N\mathbf{x}_N^\top + \frac{1}{\sigma^2}\mathbf{I}\right)^{-1} = \frac{1}{\sigma^2}\mathbf{I} - \frac{\frac{1}{\sigma^2}\mathbf{I}\mathbf{x}_N\mathbf{x}_N^\top\frac{1}{\sigma^2}\mathbf{I}}{1+\mathbf{x}_N^\top\frac{1}{\sigma^2}\mathbf{I}\mathbf{x}_N} = \frac{1}{\sigma^2}\left(\mathbf{I} - \frac{\frac{1}{\sigma^2}\mathbf{x}_N\mathbf{x}_N^\top}{1+\frac{1}{\sigma^2}\mathbf{x}_N^\top\mathbf{x}_N}\right) \tag{3.26}$$

Substituting this result in the previous equation:

$$\min_{\mathbf{x_N}}\{\sigma_N^2(x)\} = \max_{\mathbf{x_N}}\left\{\mathbf{k}_N^\top\left(\mathbf{I} - \frac{\frac{1}{\sigma^2}\mathbf{x}_N\mathbf{x}_N^\top}{1+\frac{1}{\sigma^2}\mathbf{x}_N^\top\mathbf{x}_N}\right)\mathbf{k}_N\right\} = \max_{\mathbf{x_N}}\left\{\mathbf{k}_N^\top\mathbf{k}_N - \frac{\frac{1}{\sigma^2}\mathbf{k}_N^\top\mathbf{x}_N\mathbf{x}_N^\top\mathbf{k}_N}{1+\frac{1}{\sigma^2}\mathbf{x}_N^\top\mathbf{x}_N}\right\} \tag{3.27}$$

Taking into account that $\mathbf{k}_N = x\mathbf{x}_N$ and for an arbitrary value of the prior's noise $\sigma^2$, disregarding the constant $\frac{1}{\sigma^2}$, we obtain:

$$\min_{\mathbf{x_N}}\sigma_N^2(x) = \max_{\mathbf{x_N}}\left\{\mathbf{k}_N^\top\mathbf{k}_N - \frac{\frac{1}{\sigma^2}\mathbf{k}_N^\top\mathbf{x}_N\mathbf{x}_N^\top\mathbf{k}_N}{1+\frac{1}{\sigma^2}\mathbf{x}_N^\top\mathbf{x}_N}\right\} = \max_{\mathbf{x_N}}\left\{|x|^2\frac{\frac{1}{\sigma^2}\mathbf{x}_N\mathbf{x}_N^\top}{1+\frac{1}{\sigma^2}\mathbf{x}_N^\top\mathbf{x}_N}\right\} \tag{3.28}$$

The product we want to maximize behaves as follows:

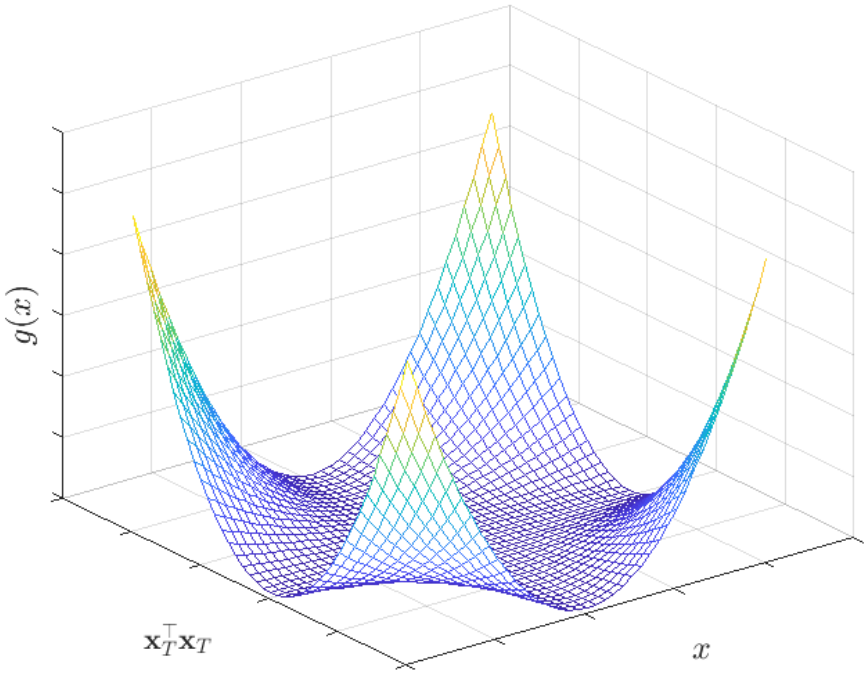


Figure 3.4: Variance of the sample points for the Linear Kernel

with $x^2$ increasing quadratically and $\mathbf{x}_N^\top \mathbf{x}_N$ being maximized by selecting the most informative subset of sample points, i.e. those with modulus $a$.

□

This conclusion aligns with the fact that, for the Linear Kernel, the most informative points are those found the farthest apart from the origin of axis. As we discussed in the information gain section and as a consequence of the linearity of the true function, a continuous sampling in that area allows us to better infer and discard the noise.

It is interesting to also note how the level of noise measured by $\sigma^2(x)$ influences the overall variance. Taking extreme values:

$$\sigma^2(x) \to \infty \Longrightarrow \lim_{\sigma^2(x)\to\infty} \min_{\mathbf{x}_N} \{\sigma_N^2(x)\} \approx \infty$$

$$\sigma^2(x) \to 0 \Longrightarrow \lim_{\sigma^2(x)\to 0} \min_{\mathbf{x}_N} \{\sigma_N^2(x)\} \approx 0$$

Such results can be easily interpreted: if there is no prior noise, i.e. $\sigma^2 \approx 0$, the variance equals zero and is indeed minimum, as the sample points contain exactly a value of the true function $f$. On the other hand, if the noise is very high, it totally

corrupts the information retrieved by sampling, making the estimation completely uncertain. This reasoning may aid with the understanding of how noise affects the sample variance but, as both limit scenarios are unlikely to occur in real settings, we still need to analyze its general expression.

By means of simulation, we have exactly computed the confidence bounds for the Linear covariance functions and three different sample points distributions: most informative, uniform and random.

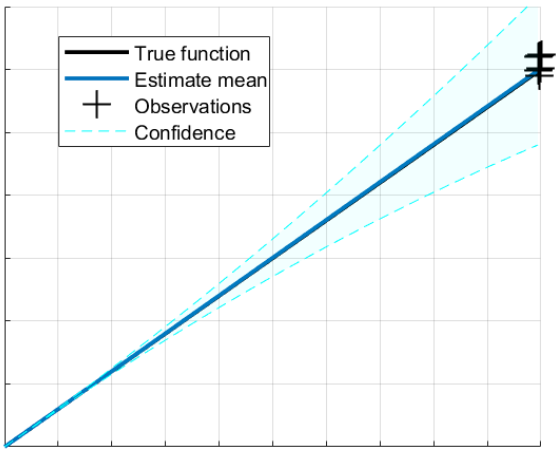


Figure 3.5: Most informative distribution of sample points for the LK with $N = 6$, $\sigma_n = 0.3$, $\gamma_N = 0.093$, $\delta = 0.15$

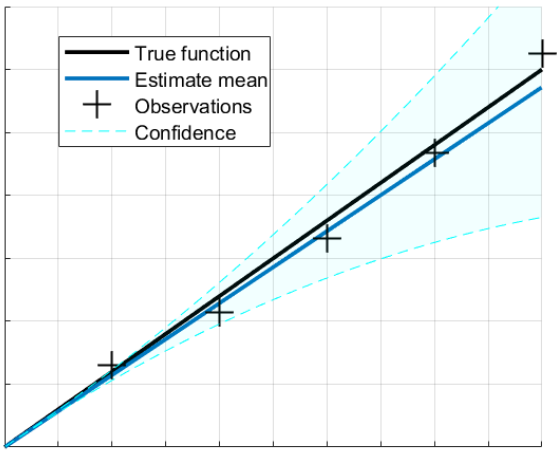


Figure 3.6: Uniform distribution of sample points for the LK with $N = 6$ points, $\sigma_n = 0.3$, $\gamma_N = 0.039$, $\delta = 0.15$

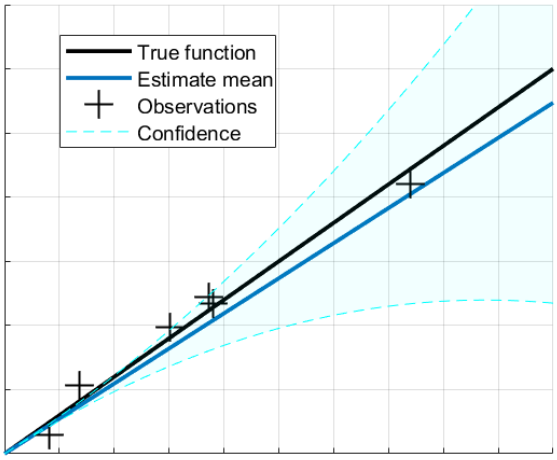


Figure 3.7: Random distribution of sample points for the LK with $N = 6$ points, $\sigma_n = 0.3$, $\gamma_N = 0.017$, $\delta = 0.15$

We can clearly observe how the most informative distribution yields the best results, both in estimation quality and tightness of bounds.

As stated in our theoretical results, a sample point distribution exists such that not only maximizes maximum information, but also guarantees a minimum variance of the sample points. This result may seem counter intuitive, as the bounds become looser when the information increases. However, we must note $\gamma_N$ precisely defines the information bound and requires a distribution that also minimizes the variance of the bound, a value that. Analogously, this implies the attainment of the tightest confidence bounds for the Linear Kernel, thus maximizing the information on the accuracy of the model's quality provided by Gaussian inference technique. In other words, sample points can be allocated such that they maximize model accuracy knowledge while obtaining optimal mean estimates at the same time.

### 3.2.2 Confidence bounds for the Gaussian Kernel

#### 3.2.2.1 Maximum Information Gain for the Gaussian Kernel

We now particularize the maximum information gain for the Gaussian Kernel:

$$\gamma_{N_{GK}} = \max_{A \in \mathcal{D}:|A|=N} \frac{1}{2} log \left| \mathbf{I} + \sigma^{-2} e^{-\frac{|\mathbf{x}-\mathbf{x}'|^2}{2\ell^2}} \right|$$

As we did with the linear case, we first analyze the covariance matrix alone. For the Gaussian kernel, its covariance matrix $\mathbf{K_N}$ it is of the form:

$$\mathbf{K}_N = \sigma^{-2} \begin{pmatrix} 1 & k(x_1, x_2) & \cdots & k(x_1, x_{N-1}) & k(x_1, x_N) \\ k(x_2, x_1) & 1 & \cdots & k(x_2, x_{N-1}) & k(x_2, x_N) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ k(x_{N-1}, x_1) & k(x_1, x_2) & \cdots & 1 & k(x_{N-1}, x_N) \\ k(x_T, x_1) & k(x_1, x_2) & \cdots & k(x_N, x_{N-1}) & 1 \end{pmatrix} \tag{3.29}$$

where $k(x_i, x_j) = e^{-\frac{\left|x_i - x_j\right|^2}{2\ell^2}}$.

We perform an analogous analysis to gain an intuition of the determinant behavior in relation with the sample points distribution. For $n = 2$, the determinant of $\mathbf{K}_N$ has the following form:

$$det(\mathbf{K}_N) = \sigma^{-2} \left| \begin{pmatrix} 1 & k(x_1, x_2) \\ k(x_2, x_1) & 1 \end{pmatrix} \right| = \sigma^{-2} \left| \begin{pmatrix} 1 & e^{-\frac{|x_1 - x_2|^2}{2\ell^2}} \\ e^{-\frac{|x_2 - x_1|^2}{2\ell^2}} & 1 \end{pmatrix} \right| \tag{3.30}$$

$$det(\mathbf{K}_N) = \sigma^{-2} \Big( 1 - e^{-\frac{|x_1 - x_2|^2}{\ell^2}} \Big) \tag{3.31}$$

which can be easily maximized by minimizing $e^{\frac{-|x_1 - x_2|^2}{\ell^2}}$, i.e. when the distance between $x_1$ and $x_2$ is maximal. For a sampling interval of $\mathcal{D} \in [0, a]$ this would imply the maximum information gain is attained when the sample points are distributed among the interval extremes by alternatively sampling at 0 and $a$.

Following the same procedure, for $n = 3$:

$$det(\mathbf{K}_N) = \sigma^{-2} \left| \begin{pmatrix} 1 & k(x_1, x_2) & k(x_1, x_3) \\ k(x_2, x_1) & 1 & k(x_2, x_3) \\ k(x_3, x_1) & k(x_3, x_2) & 1 \end{pmatrix} \right| \tag{3.32}$$

we can now express the determinant compactly as a function of distances:

$$det(\mathbf{K}_N) = \sigma^{-2}(1 + 2e^{(d_{1,2} + d_{2,3} + d_{1,3})} - e^{2d_{1,2}} - e^{2d_{2,3}} - e^{2d_{1,2}}) \tag{3.33}$$

where $d_{i,j} = |x_i - x_j|^2 / \ell^{-2}$. Setting $d_{1,3} = d_{1,2} + d_{2,3}$ without losing generality, we can treat the expression as a 3-D maximization problem, which leads to the following solution:

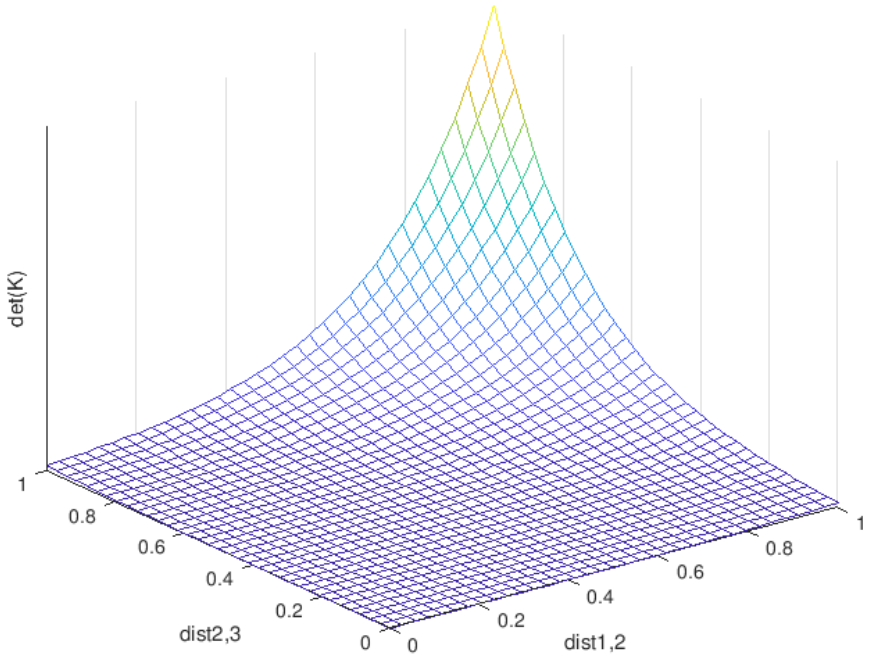


Figure 3.8: Determinant of the Gaussian kernel for N=3 points

which doesn't present any global maxima. Hence, the maximum is determined by restricting the candidate sampling values to a chosen sampling interval $[0, a]$ and placing pairs of points as far away from one another as possible, i.e. iteratively selecting the extremes: 0 and $a$. This result coincides with that obtained for $n = 2$.

Now, the full matrix of which we want to compute the maximum determinant is:

$$\begin{pmatrix} 1+\sigma^{-2} & k(x_1,x_2) & \cdots & k(x_1,x_{N-1}) & k(x_1,x_N) \\ k(x_2,x_1) & 1+\sigma^{-2} & \cdots & k(x_2,x_{N-1}) & k(x_2,x_N) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ k(x_{N-1},x_1) & k(x_1,x_2) & \cdots & 1+\sigma^{-2} & k(x_{N-1},x_N) \\ k(x_T,x_1) & k(x_1,x_2) & \cdots & k(x_N,x_{N-1}) & 1+\sigma^{-2} \end{pmatrix}$$

We observe it is a positive definite, symmetric, Toeplitz matrix, whose entries only depend on the euclidean distance between the points. This series of special properties may be used to derive a closed solution for the determinant maximization problem, such as the one obtained in the linear case.

#### 3.2.2.2 RKHS Norm for the Gaussian Kernel

We have seen that a Gaussian Kernel has a RKHS norm of the form:

$$||f||_k^2 = \frac{1}{2\pi^n} \int |F(\omega)|^2 \exp^{\frac{\sigma^2\omega^2}{2}} d\omega \tag{3.34}$$

where $F(\omega)$ is the Fourier transform of the true function $f(\mathbf{x}), \mathbf{x} \in \mathbb{R}^n$.

Note that, as the Gaussian transform spectra is bounded in frequency, i.e. has a limited bandwidth $B < \infty$, the norm can be calculated as a Fast Fourier Transform (FFT) within a given set of discrete points.

#### 3.2.2.3 Minimum variance for the Gaussian Kernel

Given the general information confidence bounds expression:

$$Pr\{\forall N, \forall x \in \mathcal{D}, |\mu_N(x) - f(x)| \leq \beta_{N+1}^{1/2} \sigma_N(x)\} \geq 1 - \delta \tag{3.35}$$

we now want to tighten the bound by minimizing the sample point variance for a Gaussian kernel.

Note that, in a spatial context, the minimum variance associated to a kernel would be equivalent to having the smallest correlation possible between all points within a fixed distance apart. However, the Gaussian covariance matrix measures the correlation precisely taking into account only the distance between each pair of points, while disregarding its magnitude.

Examining again the general expression of the sample points variance:

$$\sigma_N^2(x) = K(x,x) - K(\mathbf{x}_N, x)^\top \left(K(\mathbf{x}_N, \mathbf{x}_N) + \sigma^2\mathbf{I}\right)^{-1} K(\mathbf{x}_N, x) \tag{3.36}$$

we note that for the Gaussian kernel, $K(x,x) = 1, \forall x \in \mathcal{D}$ so that, using the more compact notation for the rest of kernel evaluations, we can rewrite $\sigma_N^2(x)$ as:

$$\sigma_N^2(x) = 1 - \mathbf{k}_N(x)^\top \left(\mathbf{K}_N + \sigma^2\mathbf{I}\right)^{-1} \mathbf{k}_N(x) \tag{3.37}$$

as we did in the Linear Kernel analysis, in order to maximize $\sigma_N^2(x)$ we maximize the second term of the expression.

Let $\mathbf{A}$ is a $n \times n$ positive, symmetric matrix with known $\mathbf{A}^{-1}$ and let $\mathbf{D}$ be a $n \times n$ positive, diagonal matrix of the same rank as $\mathbf{A}$. For our particular case, if we consider low noise, i.e. $\mathbf{D} = \sigma^2\mathbf{I}$ to be small and $\mathbf{A} = \mathbf{K}_N$, by making use of Woodbury Matrix identity [Woo49] we can obtain that:

$$(\mathbf{A} + \mathbf{D})^{-1} \sim \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{D}\mathbf{A}^{-1} \tag{3.38}$$

which applied to our maximization problem gives:

$$max \left\{ \mathbf{k}_N^\top \left( \mathbf{K}_N^{-1} - \sigma^2 \mathbf{K}_N^{-1} \mathbf{K}_N^{-1} \right) \mathbf{k}_N \right\} \tag{3.39}$$

However, by evaluating $\mathbf{k_N}$ at the sample points locations we can derive the solution for this special scenario.

$$max \left\{ \mathbf{I} - \mathbf{I}\mathbf{D}\mathbf{K}_N^{-1} \right\} = max \left\{ \mathbf{I}(1 - \mathbf{D}\mathbf{K}_N^{-1} \right\} = min \left\{ \mathbf{I}\mathbf{D}\mathbf{K}_N^{-1} \right\} = min \left\{ \mathbf{D}\mathbf{K}_N^{-1} \right\} \tag{3.40}$$

in particular:

$$min \left\{ \sigma^2 \mathbf{I}\mathbf{K}_N^{-1} \right\} = min \left\{ \mathbf{K}_N^{-1} \right\} \tag{3.41}$$

It is now worth noting that the problem of determining the sample point distribution for the tightest confidence bounds has been reduced to the computation of the inverse of the covariance matrix, also known as the concentration or precision matrix. Indeed, $\mathbf{K}_N^{-1}$ has as its elements the partial correlations of all sample points pairs, which can be understood as the correlation two random variables have with each other while disregarding the impact the rest of variables may have on them.

Particularly, when such variables are Gaussian, the partial correlation is zero only if they are independent. Hence, in order to obtain the minimum variance possible, or equivalently, to minimize the inverse of the covariance matrix, we would need to select points with the highest distance possible between them. Such a distribution of the sample points coincides with the one required for maximum information gain, the same connection found in the Linear Kernel analysis.

Given the fact that this approach requires the computation of an inverse covariance function, alternative methods are suggested to continue this work and attempt to derive a closed-form solution:

- Volume maximization of the parallelepiped spanned by the column vectors that form the covariance matrix
- Use of the Matrix determinant lemma for determinant computation, by finding the kernel's function in its feature space
- Use of symmetric Toeplitz and Euclidean distance matrices properties

## 3.3 Sample point distribution for optimal confidence bound

Given our analysis of the linear and Gaussian kernels, we have concluded that the sample point distribution for the maximum information gain coincides with that

required for a minimum variance of the points, i.e. the tightest confidence bound possible.

Taking a look at the confidence interval again, for $\forall N$ and $\forall x \in \mathcal{D}$:

$$Pr\{|\mu_N(x) - f(x)| \leq \beta_{N+1}^{1/2}\sigma_N(x)\} \geq p \tag{3.42}$$

we can see that $\sigma_N(x)$ shapes the bound for any $x$ choice and $\beta_{N+1}^{1/2}$, being a constant, regulates the global uncertainty or, more visually, "tightness" of the estimation model. In other words, it can be seen as a measure of how fitting our model is.

Hence, the first and more obvious step for the computation of good confidence bounds is ensuring the quality of the available prior knowledge. Particularly, the true function's complexity in terms of its RKHS norm, which influences $\beta$'s value, and the kernel choice and hyper-parameters tunning, which affects both $\beta$ and $\sigma_N(x)$.

The second step, which is usually less restricted in a system inference scenario, is the choice of sample points. This decision influences greatly the information gain $\gamma_N$, which in turn affects $\beta$ and the sample covariance $\sigma_N(x)$, both becoming the decision criteria for sample point distribution: maximum information gain and minimum sample covariance.

# Chapter 4

# Conclusion

We have addressed the more specific issue of defining a relationship between the selection of the training points and the estimation error. In particular, we have selected some of the contributions of Srinivas for information-related confidence bounds [SKKS12], in particular Theorem 6, and proceeded to analyze it for different common kernels.

Starting off with the simplest case, we have analyzed the linear kernel and, by means of well-known matrix algebra properties and identities, have been able to provide a closed form for the mutual maximization problem. Thus, we have determined the optimal distribution of the sampling points for this particular case: samples must be of maximum magnitude, and always the same. This result assumes a good tuning of the hyper-parameters has been attained. In order to compute the mutual information, the study of functions in the Reproducing Kernel Hilbert Spaces and in particular, the expression of its norm was crucial, as it is a useful measure of complexity under which we derive the smoothness of our estimation. For the Linear kernel, the RKHS norm is simply the slope of the true function, which provides a useful measure for kernel design.

Similarly, we performed an analogous analysis for the Gaussian kernel. Due to the increased complexity of the covariance matrix, we have only analyzed and obtained closed-forms of the most informative sample point distributions for numbers of points, but have provided with a strong intuition about its behavior, which has also been reinforced with simulations. We also have been able to compute its RKHS norm by using Fourier transform properties.

A natural question arose: would the sample points that maximize the mutual information gain correspond to those with minimum variance? If they are indeed the same, it would imply that the most informative distribution of sample points is the same that provides the tightest confidence intervals. We have been able to proof this idea for the Linear Kernel, as well as a low noise approximation for the Gaus-

sian kernel. Both results strengthen the well-known importance of an appropriate kernel choice, but also highlight that of an optimal sample points distribution by explicitly showing its influence on the quality of the confidence bounds in terms of its information gain.

By means of simulation, we have noted that information-related confidence bounds serve as a much more precise estimation certainty measure in comparison with classic covariance intervals. They take greatly into account the prior knowledge about the true function introduced by the kernel choice and hyper-parameter tunning, the mutual information of all pairs of sample points and the function's smoothness encoded by its norm in RKHS. It is also worth noting how, if all those parameters are set correctly, a lower number of sample points can provide good results, both in terms of mean estimation as well as probabilistic confidence.

Future work would include the expansion of this results for more complex covariance functions or custom built kernels, such as additive combinations of the ones presented in this work. In particular, given the strong results obtained for the Linear Kernel, next steps should focus on the derivation of a similar connection between the most informative sample subset and the one that guarantees the tightest confidence bounds.

# List of Figures

# Bibliography

[ADH10] Christophe Andrieu, Arnaud Doucet, and Roman Holenstein. Particle markov chain monte carlo methods, 2010.

[Aue03] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *J. Mach. Learn. Res.*, 3:397–422, March 2003. URL: `http://dl.acm.org/citation.cfm?id=944919.944941`.

[BH16a] T. Beckers and S. Hirche. Equilibrium distributions and stability analysis of gaussian process state space models. In *Proceedings of the 55th Conference on Decision and Control (CDC)*, Las Vegas, USA, 2016. (accepted).

[BH16b] T. Beckers and S. Hirche. Stability of gaussian process state space models. In *Proceedings of the European Control Conference (ECC)*, 2016.

[Bil13] Stephen A Billings. *Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains*. John Wiley & Sons, 2013.

[BS92] Jeffrey S Banks and Rangarajan K Sundaram. A class of bandit problems yielding myopic optimal strategies. *Journal of Applied Probability*, pages 625–632, 1992.

[Cap08] Enrico Capobianco. Kernel methods and flexible inference for complex stochastic dynamics. *Physica A: Statistical Mechanics and its Applications*, 387:4077–4098, 2008.

[CBRV13] Emile Contal, David Buffoni, Alexandre Robicquet, and Nicolas Vayatis. Parallel gaussian process optimization with upper confidence bound and pure exploration. *CoRR*, abs/1304.5350, 2013. URL: `http://arxiv.org/abs/1304.5350`.

[CSPD14] R. Calandra, A. Seyfarth, J. Peters, and MP. Deisenroth. An experimental comparison of bayesian optimization for bipedal locomotion. In *Proceedings of 2014 IEEE International Conference on Robotics and Automation*, pages 1951–1958. IEEE, 2014.

[CV95] Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995.

[dFSZ12a] Nando de Freitas, Alex Smola, and Masrour Zoghi. Exponential regret bounds for Gaussian process bandits with deterministic observations. In *International Conference on Machine Learning (ICML)*, 2012. URL: http://icml.cc/discuss/2012/853.html.

[dFSZ12b] Nando de Freitas, Alexander J. Smola, and Masrour Zoghi. Regret bounds for deterministic gaussian process bandits. *CoRR*, abs/1203.2177, 2012. URL: http://arxiv.org/abs/1203.2177.

[DKB14] Thomas Desautels, Andreas Krause, and Joel W. Burdick. Parallelizing exploration-exploitation tradeoffs in gaussian process bandit optimization. *J. Mach. Learn. Res.*, 15(1), 2014.

[ENDH] S. Eleftheriadis, T. F. W. Nicholson, M. P. Deisenroth, and J. Hensman. Identification of Gaussian Process State Space Models. *ArXiv e-prints*. arXiv:1705.10888.

[Fla63] R. H. Flake. Volterra series representation of nonlinear systems. *Transactions of the American Institute of Electrical Engineers, Part II: Applications and Industry*, 81(6):330–335, 1963. URL: https://doi.org/10.1109/tai.1963.6371765, doi:10.1109/tai.1963.6371765.

[FLSR13] R. Frigola, F. Lindsten, T. B. Schön, and C. E. Rasmussen. Bayesian Inference and Learning in Gaussian Process State-Space Models with Particle MCMC. *ArXiv e-prints*, 2013. arXiv:1306.2861.

[GKS05] Carlos Guestrin, Andreas Krause, and Ajit Paul Singh. Near-optimal sensor placements in gaussian processes. In *Proceedings of the 22Nd International Conference on Machine Learning*, ICML '05, pages 265–272, New York, NY, USA, 2005. ACM. URL: http://doi.acm.org/10.1145/1102351.1102385, doi:10.1145/1102351.1102385.

[Goo70] Robert Kent Goodrich. A riesz representation theorem. *Proceedings of the American Mathematical Society*, pages 629–636, 1970.

[GYCW15] Zongyu Geng, Feng Yang, Xi Chen, and Nianqiang Wu. Gaussian process based modeling and experimental design for sensor calibration in drifting environments. *Sensors and Actuators B: Chemical*, 216:321–331, sep 2015. URL: https://doi.org/10.1016/j.snb.2015.03.071, doi:10.1016/j.snb.2015.03.071.

[Har97] David A. Harville. *Matrix Algebra From a Statistician's Perspective*. Springer New York, 1997. URL: https://doi.org/10.1007/b98818, doi:10.1007/b98818.

[Her02] Ralf Herbrich. *Learning kernel classifiers: theory and algorithms*. Mit Press, 2002.

[HSS08] Thomas Hofmann, Bernhard Schölkopf, and Alexander J. Smola. Kernel methods in machine learning. *The Annals of Statistics*, 36(3):1171–1220, jun 2008. URL: https://doi.org/10.1214/009053607000000677, doi:10.1214/009053607000000677.

[KLQ95] Chun-Wa Ko, Jon Lee, and Maurice Queyranne. An exact algorithm for maximum entropy sampling. *Operations Research*, 43(4):684–691, 1995.

[KMSRL03] J. Kocijan, R. Murray-Smith, C.E. Rasmussen, and B. Likar. *Predictive control with Gaussian process models*, volume 1. IEEE, 2003.

[Kri14] David Kristjanson. *Automatic Model Construction with Gaussian Processes*. PhD thesis, Cambridge University, 2014.

[KSGW05] Andreas Krause, Ajit Singh, Carlos Guestrin, and Chris Williams. Near-optimal sensor placements in gaussian processes. In *In ICML*. ICML, 2005.

[LK07] Bojan Likar and Juš Kocijan. Predictive control of a gas–liquid separation plant based on a gaussian process model. *Computers & Chemical Engineering*, 31(3):142–152, jan 2007. URL: https://doi.org/10.1016/j.compchemeng.2006.05.011, doi:10.1016/j.compchemeng.2006.05.011.

[Mer09] James Mercer. Functions of positive and negative type, and their connection with the theory of integral equations. *Philosophical transactions of the royal society of London. Series A, containing papers of a mathematical or physical character*, 209:415–446, 1909.

[Moo28] C. N. Moore. Review: T. j. i'a. bromwich, an introduction to the theory of infinite series. *Bull. Amer. Math. Soc.*, 34(2):244, 03 1928. URL: https://projecteuclid.org:443/euclid.bams/1183492634.

[POTO01] H. Peng, T. Ozaki, Y. Toyoda, and K. Oda. Modeling and control of systems with signal dependent nonlinear danymics. In *2001 European Control Conference (ECC)*, pages 42–47, Sept 2001.

[Ren01] James Renegar. *A Mathematical View of Interior-point Methods in Convex Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001.

[Ros10] Lorenzo Rosasco. Lecture notes in reproducing kernel hilbert spaces, 2010.

[RW05] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[SKKS12] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias W Seeger. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012.

[SM50] Jack Sherman and Winifred J. Morrison. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *Ann. Math. Statist.*, 21(1):124–127, 03 1950. URL: `http://dx.doi.org/10.1214/aoms/1177729893`, `doi:10.1214/aoms/1177729893`.

[SP14] Johan Schoukens and Rik Pintelon. Identification of linear systems: a practical guideline to accurate modeling, 2014.

[TDR10] Ryan Turner, Marc Deisenroth, and Carl Rasmussen. State-space inference and learning with gaussian processes. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 868–875, 2010.

[WA13] Andrew Gordon Wilson and Ryan P. Adams. Gaussian process covariance kernels for pattern discovery and extrapolation. *CoRR*, abs/1302.4245, 2013.

[WFH08] J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):283–298, Feb 2008. `doi:10.1109/TPAMI.2007.1167`.

[Woo49] Max A Woodbury. The stability of out-input matrices. *Chicago, IL*, 9, 1949.

[YO10] S. Yilmaz and Y. Oysal. Fuzzy wavelet neural network models for prediction and identification of dynamical systems. *IEEE Transactions on Neural Networks*, 21(10):1599–1609, Oct 2010. `doi:10.1109/TNN.2010.2066285`.

[ZLB13] L. Zhang, K. Li, and E. W. Bai. A new extension of newton algorithm for nonlinear system modelling using rbf neural networks. *IEEE Transactions on Automatic Control*, 58(11):2929–2933, Nov 2013. `doi:10.1109/TAC.2013.2258782`.

# License