# Joint introduction to Gaussian Processes and Relevance Vector Machines with Connections to Kalman filtering and other Kernel Smoothers

Luca Martino* and Jesse Read†

* Dept. of Statistical Signal Processing, Universidad Rey Juan Carlos, Madrid (Spain)

† LIX, Ecole Polytechnique, Institut Polytechnique de Paris, France.

**Abstract**

The expressive power of Bayesian kernel-based methods has led them to become an important tool across many different facets of artificial intelligence, and useful to a plethora of modern application domains, providing both power and interpretability via uncertainty analysis. This article introduces and discusses two methods which straddle the areas of probabilistic Bayesian schemes and kernel methods for regression: Gaussian Processes and Relevance Vector Machines. Our focus is on developing a common framework with which to view these methods, via intermediate methods a probabilistic version of the well-known kernel ridge regression, and drawing connections among them, via dual formulations, and discussion of their application in the context of major tasks: regression, smoothing, interpolation, and filtering. Overall, we provide understanding of the mathematical concepts behind these models, and we summarize and discuss in depth different interpretations and highlight the relationship to other methods, such as linear kernel smoothers, Kalman filtering and Fourier approximations. Throughout, we provide numerous figures to promote understanding, and we make numerous recommendations to practitioners. Benefits and drawbacks of the different techniques are highlighted. To our knowledge, this is the most in-depth study of its kind to date focused on these two methods, and will be relevant to theoretical understanding and practitioners throughout the domains of data-science, signal processing, machine learning, and artificial intelligence in general.

**Keywords:** Gaussian processes, Relevance Vector Machines, Bayesian Learning, Bayesian Ridge, Kernel Smoothing, Kalman Filtering

## 1 Introduction

This work details and discusses techniques and methods lying on the intersection of two areas: probabilistic Bayesian schemes and kernel methods; in a regression framework. Such techniques have become increasingly popular in statistics, signal processing, and machine learning [1, 2, 3, 4, 5]. The expressive power of these methods increases with the number of data points

observed, and they can be effective for dealing with structured (non-tabular) sources such as sequential data. Indeed, despite the soaring popularity of deep neural network architectures in recent decades, the methods we approach in this work are still relevant to a plethora of modern application domains; and an understanding of the mathematical concepts behind them is still of fundamental importance across science and mathematics under the general umbrella of modern artificial intelligence.

More specifically, we look at Gaussian Processes (GPs) [6, 3] and Relevance Vector Machines (RVMs) for regression [7, 8] – both Bayesian nonparametric approaches which have yielded convincing results in recent years and attracted a correspondingly significant interest [9, 10, 11, 12, 13, 14]. The main scope of this manuscript encompasses a unified introduction to GPs, RVMs. We link these two methods via Kernel Ridge Regression – which we refer to as quasi GP in the probabilistic sense (often known elsewhere as Bayesian Ridge Regression); covering important material presented in the literature, which deserves to be properly highlighted [15, 16, 7, 17, 18]. We allow a correct comparison between these techniques and perform a detailed analysis of uncertainty estimation. Moreover, we leverage the opportunity to connect these methods to a number of other important methods in neighboring areas, including splines, kernel smoothers, k-Nearest Neighbor (kNN) schemes and a Fourier interpolation [1, 19], and particularly exploring the connection between GPs and Kalman filtering which has not been completely elaborated in the literature (although recent work remarked upon this link [20, 21, 5] we propose a gentle explanation with examples in discrete time, that is not specifically discussed in those works). We considering different possible scenarios in the framework of regression models, including prediction, filtering, smoothing and interpolation, providing the specific solutions in each one of these cases.

We will see that the main benefit of the RVM approach is the flexibility in the choice of the basis functions, whereas the main advantage of the GP approach is the good behavior of the predictive variance. And we will discuss the implication thereof. We will also discuss the interpretability of the chosen bases/kernel functions, the uncertainty analysis with each techniques and the generation of random functions from (direct or induced) priors and/or posteriors over the underlying function. Furthermore, several related concepts well-known in signal processing (e.g., the Fourier upsampling in Section 9.2.4, and the linear digital filters in Section 9.2.5) are described and connected to the rest of techniques. In this sense, this work builds bridges among different concepts in statistics, machine learning and signal processing.

The paper is structured as follows:

- In Section 2, we introduce the problem statement, the notation and provides a joint introduction of the RVM and GP methods (considering joint formulas and properties).
- The derivation of the RVM solution is given in Section 3.
- The probabilistic version of KRR is described in Section 4.
- The GP derivation is provided in Section 5.
- Section 6 describes the dual representation of RVM as a GP.

- An initial summary with important considerations and remarks is provided in Section 7; then
- Section 8 provides a discussion regarding the uncertainty analysis with GPs.
- Section 9 shows that RVM and GP can be seen as linear kernel smoothers and describes other well-known examples in the literature.
- In Section 10, we describes the connections between Kalman filtering (and smoothing) with the GP solution.
- A final discussion and concluding summary is provided in Section 11.

# 2 Problem Statement and Common Framework

In this section, we introduce the main notation and the problem statement. Moreover, we provide a joint introduction of GPs and RVMs in the form of a common framework, elaborating all the equations shared by both models. Namely, we introduce the analytic form of the regression function $\widehat{f}(\mathbf{x})$, the observation model and the likelihood function (all shared by both methods), as well as the design matrix and the interpolation case (where both schemes provide the same solution). The main notation of the work is summarized in Table 1.

Table 1: Main notation of the work.

| | |
|---|---|
| $\mathbf{x} \in \mathbb{R}^{d_X}$ | a $d_X$-dimensional input observation |
| $y \in \mathbb{R}$ | a scalar output |
| $\boldsymbol{y}$ | $\boldsymbol{y} = [y_1, \dots, y_N]^\top$, vector of outputs/observations |
| $e$, $\boldsymbol{e}$ | Gaussian noise perturbation $e \sim \mathcal{N}(\mu_e, \sigma_e^2)$, $\boldsymbol{e} \sim \mathcal{N}(\boldsymbol{\mu}_e, \boldsymbol{\Sigma}_e)$ |
| $\mathcal{D}$ | Dataset: $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N$ or $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$, of $N$ points |
| $f(\mathbf{x})$ | underlying/hidden function (unknown) $f$ evaluated at $\mathbf{x}$ |
| $\widehat{f}(\mathbf{x})$ | regression function $\widehat{f}$, est./approx. of $f$, evaluated at $\mathbf{x}$ |
| $\mathbf{f}$ or $\mathbf{f}(\mathbf{x})$ | vector $[f_1(\mathbf{x}), \dots, f(\mathbf{x})_N]^\top$, $\equiv f(\mathbf{X})$ |
| $\widehat{\mathbf{f}}$ or $\widehat{\mathbf{f}}(\mathbf{x})$ | vector $[\widehat{f}(\mathbf{x})_1, \dots, \widehat{f}(\mathbf{x})_N]^\top$, $\equiv \widehat{f}(\mathbf{X})$ |
| $\boldsymbol{\theta}$ | vector of (hyper-)parameters of the model $\boldsymbol{\theta} = [\theta_1, \dots, \theta_{d_\theta}]^\top$ |
| $\psi_i(\mathbf{x}, \mathbf{x}_i)$ | nonlinear basis, localized around $\mathbf{x}_i$ |
| $\boldsymbol{\varphi}(\mathbf{x})$ | $\boldsymbol{\varphi} = [\psi_1(\mathbf{x}, \mathbf{x}_1), ..., \psi_N(\mathbf{x}, \mathbf{x}_N)]^\top$, the $N \times 1$ design vector. |
| $\boldsymbol{\Psi}$ | $= [\boldsymbol{\psi}_1(\mathbf{x}_1), ..., \boldsymbol{\psi}_N(\mathbf{x}_N)]^\top$, the $N \times N$ design matrix. |
| $\widehat{\boldsymbol{\rho}}$ | vector of estimated coefficients, or $\widehat{\boldsymbol{\rho}}(\mathbf{y})$ if determined by $\mathbf{y}$ |
| $\boldsymbol{\varphi}(\mathbf{x})$ | vector of smoothing kernels, $\boldsymbol{\varphi} = [\varphi_1(\mathbf{x}\|\mathbf{x}_1), \dots, \varphi_N(\mathbf{x}\|\mathbf{x}_N)]$ |

**Observation model.** We have a dataset consisting of $N$ data points, $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N$, where

each $i$-th *input* $\mathbf{x}_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,d_X}]^\top \in \mathcal{X} \subseteq \mathbb{R}^{d_X}$ is associated with *scalar output* $y_i \in \mathbb{R}$.[1] To simplify notation we consider $\mathbb{E}[y_i] = 0$ for all $i = 1, \ldots, N$, without loss of generality. This assumption can be easily relaxed adding a bias to the probabilistic models. Namely, the goal is to approximate an unknown underlying function $f(\mathbf{x}) : \mathbb{R}^{d_X} \to \mathbb{R}$, which we assume has generated our training points, in the form

$$y_i = f(\mathbf{x}_i) + e_i, \tag{1}$$

where $e_i$ is a Gaussian perturbation with zero mean and variance $\sigma_e^2$, i.e., $e_i \sim \mathcal{N}(e|0, \sigma_e^2)$. That is to say, ideally, we want to learn some model $\widehat{f}$ (let us call it the *regression function*) such that $\widehat{f} \approx f$. We emphasise that this sample pair $\mathbf{x}, y$ do *not necessarily* belong to the training set, which means that in fact $y$ may not be observed at all. In prediction, we want to *generalize* to new test points. Note that, $\mathbf{y}$, $f(\mathbf{x})$ are random variables, whereas the inputs $\mathbf{x}$ play the role of (non-random) parameters. More specifically, for each $\mathbf{x} \in \mathcal{X}$, then $f(\mathbf{x})$ represents a different random variable.

**Regression function.** In this work, we describe theory and properties of different methods where the regression function can be expressed as a linear combination of $N$ non-linear functions, i.e.,

$$\widehat{f}(\mathbf{x}) = \sum_{i=1}^{N} \widehat{\rho}_i \psi_i(\mathbf{x}, \mathbf{x}_i), \tag{2}$$

where the non-linear functions $\psi_i(\mathbf{x}, \mathbf{z}) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ have been selected in advance by the user, according to the problem domain, i.e., encoding some prior knowledge about the underlying function $f(\mathbf{x})$. The coefficients $\widehat{\rho}_n \in \mathbb{R}$, $n = 1, \ldots, N$, are analytically obtained according to the chosen probabilistic derivation, e.g., either RVM or GP as covered in this article. Note that non-linearity $\psi_i$ is indexed by $i$, since, in RVM, these functions can differ among inputs $i$. In a GP model, we will consider the simpler notation $\psi_i(\mathbf{x}, \mathbf{x}_i) = \psi(\mathbf{x}, \mathbf{x}_i)$, precisely as we need to impose that the analytical form of $\psi$ does not vary with $i$.

**Remark 1.** *The number of components in Eq.* (2) *is exactly the number of data, i.e., $N$. Therefore, the flexibility of the model increases as the number of data $N$ grows. The regression methods, represented by Eq.* (2)*, are non-parametric models.*

**Remark 2.** *The RVM and GP solutions differ for the choice of the coefficients $\widehat{\rho}_n$. This different choice is due to the probabilistic approach employed by each method.*

Defining also the $N \times 1$ *design* vector $\boldsymbol{\psi}(\mathbf{x}) = [\psi_1(\mathbf{x}, \mathbf{x}_1), \ldots, \psi_N(\mathbf{x}, \mathbf{x}_N)]^\top$, the approximating function (of all the methods derived in this work) can be also written in a vectorial form,

$$\widehat{f}(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top \widehat{\boldsymbol{\rho}}. \tag{3}$$

Note that the the coefficients will be determined considering the vector of outputs $\mathbf{y} = [y_1, \ldots, y_N]^\top$, hence a more complete notation will be $\widehat{\boldsymbol{\rho}} = \widehat{\boldsymbol{\rho}}(\mathbf{y})$. Let us also define the $N \times N$

[1] In this work, we consider single-output regression problems. For multi-output approaches, see [22, 23].

*design matrix* $\boldsymbol{\Psi} = [\boldsymbol{\psi}(\mathbf{x}_1), \ldots, \boldsymbol{\psi}(\mathbf{x}_N)]^\top$, i.e.,

$$\boldsymbol{\Psi} = \begin{bmatrix} \psi_1(\mathbf{x}_1, \mathbf{x}_1) & \psi_1(\mathbf{x}_1, \mathbf{x}_2) & \ldots & \psi_1(\mathbf{x}_1, \mathbf{x}_N) \\ \psi_2(\mathbf{x}_2, \mathbf{x}_1) & \psi_2(\mathbf{x}_2, \mathbf{x}_2) & \ldots & \psi_2(\mathbf{x}_2, \mathbf{x}_N) \\ \vdots & & & \\ \psi_N(\mathbf{x}_N, \mathbf{x}_1) & \psi_N(\mathbf{x}_N, \mathbf{x}_2) & \ldots & \psi_N(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix}. \tag{4}$$

In the GP case, we will require that $\boldsymbol{\Psi}$ be symmetric, but for RVMs it could be a non-symmetric matrix. We will show below that the vectors of coefficients for RVMs and GPs are given by the formulas

$$\begin{aligned} \text{RVM:} \quad \widehat{\boldsymbol{\rho}} &= \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top \left( \boldsymbol{\Psi} \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top + \sigma_e^2 \mathbf{I}_N \right)^{-1} \mathbf{y}, \\ \text{GP:} \quad \widehat{\boldsymbol{\rho}} &= \left( \boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N \right)^{-1} \mathbf{y}, \end{aligned} \tag{5}$$

where $\boldsymbol{\Sigma}_\rho$ is a $N \times N$ matrix decided by the user. By substituting expressions (5) into Eq. (3), we obtain the following regression functions:

$$\begin{aligned} \text{RVM:} \quad \widehat{f}(\mathbf{x}) &= \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top \left( \boldsymbol{\Psi} \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top + \sigma_e^2 \mathbf{I}_N \right)^{-1} \mathbf{y}, \\ \text{GP:} \quad \widehat{f}(\mathbf{x}) &= \boldsymbol{\psi}(\mathbf{x})^\top \left( \boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N \right)^{-1} \mathbf{y}. \end{aligned} \tag{6}$$

The relationships among of the solution $\widehat{f}(\mathbf{x})$ of the main methods described in this work, RVM, GP and Quasi-GP (Q-GP) is graphically summarized in Figure 1. Furthermore, Figure 2 provides some examples of the solution $\widehat{f}(\mathbf{x})$ with $N = 3$ data points.

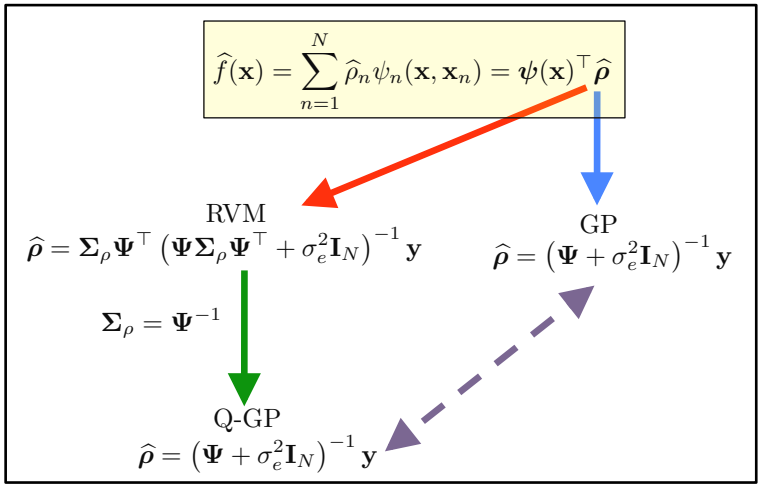
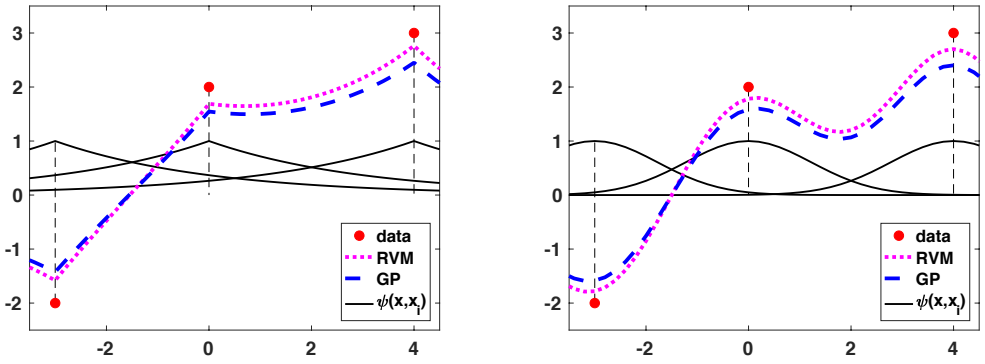



Figure 1: Graphical representation of the relationships among of the solution $\widehat{f}(\mathbf{x})$ of the main methods described in this work, RVM, GP and Quasi-GP (Q-GP). Note that the estimated function $\widehat{f}(\mathbf{x})$ coincides in GP and Q-GP.




(a) (b)

Figure 2: Examples of RVM and GP solutions $\widehat{f}(\mathbf{x})$ with **(a)** Laplacian and **(b)** Gaussian bases (depicted with solid lines). The $N = 3$ data points as shown with red dots. The GP solutions are depicted with dashed lines, whereas the RVM solutions are shown with dotted lines.

**Remark 3.** *RVMs and GPs provide a complete description of the posterior-predictive distribution over the underlying function $f(\mathbf{x})$. In both case, this posterior density is Gaussian. The expected value of the posterior-predictive distribution is $\widehat{f}(\mathbf{x})$ in Eq.* (2)*.*

**Localized nonlinearities.**

In this work, we denote the nonlinearities *localized "around" the inputs* with the notation $\psi_n(\mathbf{x}, \mathbf{x}_n)$, with $n = 1, \ldots, N$ (since they are localized around $\mathbf{x}_n$, we need exactly $N$ functions $\psi_n$). In some scenarios, we have the same functions translated in different regions of the space, i.e., $\psi_n(\mathbf{x}, \mathbf{x}_n) = \psi(\mathbf{x}, \mathbf{x}_n)$. As an example of localized function, consider for instance the constant basis

$$\psi(\mathbf{x}, \mathbf{x}_n) = \begin{cases} 1 & ||\mathbf{x} - \mathbf{x}_n||_p \leq \epsilon, \\ 0 & ||\mathbf{x} - \mathbf{x}_n||_p > \epsilon, \end{cases} \tag{7}$$

where $\epsilon > 0$ and $||\mathbf{z}||_p = \left(\sum_{i=1}^{d_X} |z_i|^p\right)^{1/p}$ represents the $L_p$ vector norm. Other example of localized function is the following radial exponential function,

$$\psi(\mathbf{x}, \mathbf{x}_n) = \exp\left(-\frac{||\mathbf{x} - \mathbf{x}_n||_p}{\lambda}\right). \tag{8}$$

For simplicity, we have removed the subindex $n$ in $\psi$, however in RVM we can employ different types of nonlinear functions, for instance, combining constant basis at some inputs and radial exponential functions at other inputs. The bases in Eqs. (7)-(8) are also *isotropic* (or *homogeneous*) since they depend only on the $L_p$ distance $r = ||\mathbf{x} - \mathbf{x}_n||_p$, that is a scalar value [6, Chapter 4]. They are also *stationary* kernels/bases. A kernel function is stationary if satisfies the condition $\psi(\mathbf{x}, \mathbf{x}_n) = \psi(\mathbf{x} - \mathbf{x}_n)$, i.e., it depends only on the difference vector $\mathbf{d} = \mathbf{x} - \mathbf{x}_n$, but not on

the values of the inputs, $\mathbf{x}$ and $\mathbf{x}_n$, themselves. Generally, a stationary kernel is an *anisotropic* kernel, since it depends on both the direction and the length of the difference vector $\mathbf{d}$. Clearly, an isotropic kernel is always a stationary kernel.

**Remark 4.** *The spline models are special cases of GPs, where the support of the bases is bounded (with a support smaller of the domain $\mathcal{X}$). In this scenario, the matrix $\mathbf{\Psi}$ is sparse and, in some scenarios, is a band matrix [6, Chapter 6], [24].*

## 2.1 Posterior-predictive distribution

We consider two different probabilistic approaches which provide different regression models. In the standard Bayesian derivation, the nonlinearities $\psi_n(\mathbf{x}, \mathbf{x}_n)$ play the role of basis functions. In the Gaussian process (GP) approach, the nonlinearities $\psi_n(\mathbf{x}, \mathbf{x}_n)$ play the role of kernel functions specifying the correlation among different pairs of inputs. A prior density over the underlying function $p(f(\mathbf{x}))$ is assumed (explicitly or implicitly) in both cases. Thus, in both cases, we obtain a complete description of a Gaussian posterior distribution of the hidden function in a generic test input $\mathbf{x}$, i.e.,

$$p(f(\mathbf{x})|\mathbf{y}) = \frac{1}{p(\mathbf{y})} p(\mathbf{y}|f(\mathbf{x}))p(f(\mathbf{x})), \tag{9}$$

where $p(\mathbf{y}|f(\mathbf{x}))$ is the *likelihood function* (which is induced by Eq. (1)), $p(f(\mathbf{x}))$ represents the *prior density* over $f(\mathbf{x})$ (which is given by the specific probabilistic approach), and $p(\mathbf{y})$ is the so-called *marginal likelihood*, useful for model selection (e.g., hyperparameter tuning). It is given by the expression $p(\mathbf{y}) = \int_{\mathcal{X}} p(\mathbf{y}|f(\mathbf{x}))p(f(\mathbf{x}))df(\mathbf{x})$. Below, we will derive the marginal likelihood for the different techniques (see also Section 7.4).

In the RVM and GP schemes, the posterior density is in both cases Gaussian, i.e.,

$$p(f(\mathbf{x})|\mathbf{y}) = \mathcal{N}(\mathbf{x}|\widehat{f}(\mathbf{x}), \sigma^2_{f|y}(\mathbf{x})). \tag{10}$$

where the mean is the function $\widehat{f}(\mathbf{x})$, i.e., $\mu_{f|y}(\mathbf{x}) = \widehat{f}(\mathbf{x})$ in Eqs. (2) and (6). The final expressions of the coefficient vector $\widehat{\boldsymbol{\rho}}$ and of the variance $\sigma^2(\mathbf{x})$ depend on the probabilistic derivation employed.[2] We will derive the variance for both techniques, obtaining

$$\begin{aligned} \text{RVM:} \quad \sigma^2_{f|y}(\mathbf{x}) &= \boldsymbol{\psi}(\mathbf{x})^\top \left( \frac{1}{\sigma_e^2} \mathbf{\Psi}^\top \mathbf{\Psi} + \mathbf{\Sigma}_\rho^{-1} \right)^{-1} \boldsymbol{\psi}(\mathbf{x}), \\ \text{GP:} \quad \sigma^2_{f|y}(\mathbf{x}) &= \psi(\mathbf{x}, \mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top (\mathbf{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \boldsymbol{\psi}(\mathbf{x}). \end{aligned} \tag{11}$$

**Remark 5.** *The RVM and GP models consider the same likelihood function, since they assume the same observation model in Eq.* (1).

The likelihood function is described below.

[2] Note that a complete notation should be $p(f(\mathbf{x})|\mathbf{y}, \mathbf{x}_{1:N}, \mathcal{M})$, i.e., we consider all the training input points $\mathbf{x}_{1:N} = \{\mathbf{x}_n\}_{n=1}^N$ given and fixed, and with $\mathcal{M}$ we denote the bases $\psi_n$ and all the parameters of the model. In the rest of the work, for simplicity, we keep the simpler notation $p(f(\mathbf{x})|\mathbf{y}) = p(f(\mathbf{x})|\mathbf{y}, \mathbf{x}_{1:N}, \mathcal{M})$.

### 2.1.1 Likelihood function

Given the observation model in Eq. (1), the induced likelihood function is given by

$$p(y_i|f(\mathbf{x}_i)) = \mathcal{N}(y_i|f(\mathbf{x}_i), \sigma_e^2). \tag{12}$$

Furthermore, defining $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]^\top$ and considering conditional independence for the observations $y_i$, we also have

$$p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^{N} p(y_i|f(\mathbf{x}_i)) = \mathcal{N}(\mathbf{y}|\mathbf{f}, \sigma_e^2\mathbf{I}_N), \tag{13}$$

where $\mathbf{I}_N$ is an $N \times N$ identity matrix. Depending on the employed probabilistic approach (see below), one can assume that $f(\mathbf{x})$ has exactly the form in Eq. (2) or Eq. (3), i.e., $f(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x}_i)^\top\boldsymbol{\rho}$, so that the observation model can be written as

$$y_i = \boldsymbol{\psi}(\mathbf{x}_i)^\top\boldsymbol{\rho} + e_i. \tag{14}$$

Considering that the nonlinearities are known and chosen by the user, the likelihood of a single observations with respect to the weights is

$$p(y_i|\boldsymbol{\rho}) = \mathcal{N}(y_i|\boldsymbol{\psi}(\mathbf{x}_i)^\top\boldsymbol{\rho}, \sigma_e^2). \tag{15}$$

The complete likelihood function with respect to the coefficients is

$$p(\mathbf{y}|\boldsymbol{\rho}) = \prod_{i=1}^{N} p(y_i|\boldsymbol{\rho}) = \mathcal{N}(\mathbf{y}|\boldsymbol{\Psi}\boldsymbol{\rho}, \sigma_e^2\mathbf{I}_N), \tag{16}$$

and can be obtained from Eq. (13) setting $\mathbf{f} = \boldsymbol{\Psi}\boldsymbol{\rho}$.

## 2.2 Smoothing and prediction

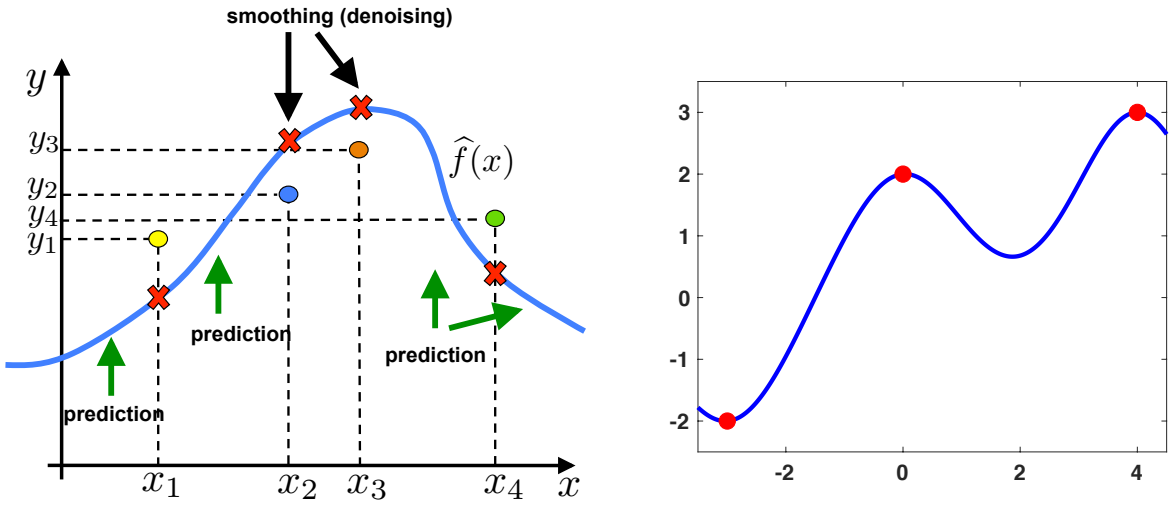
**Smoothing.** We refer to smoothing problem when one is interested only to obtain the estimation values $\widehat{f}(\mathbf{x}_1), \ldots, \widehat{f}(\mathbf{x}_N)$, i.e., to know the estimations only at the training inputs, $\mathbf{x}_{1:N} = \{\mathbf{x}_n\}_{n=1}^N$. In this scenario, Eq. (1) can be expressed in the following vectorial form

$$\mathbf{y} = \mathbf{f} + \mathbf{e} = \boldsymbol{\Psi}\boldsymbol{\rho} + \mathbf{e}, \tag{17}$$

where $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]^\top$ and $\mathbf{e} = [e_1, \ldots, e_N]^\top \sim \mathcal{N}(\mathbf{0}, \sigma_e^2\mathbf{I}_N)$ with $\mathbf{I}_N$ is an $N \times N$ unit matrix. The goal of the smoothing problem is to obtain a vector $\widehat{\mathbf{f}}$ that approximates $\mathbf{f}$. This is also known as *denoising*.

**Prediction.** We refer to a prediction problem if we consider the approximation of the underlying function $f(\mathbf{x})$ at some $\mathbf{x}$ which is not contained in $\{\mathbf{x}_n\}_{n=1}^N$. Namely, the goal in prediction is to infer the value $f(\mathbf{x}^*)$ at some test point $\mathbf{x}^*$ that is not contained in the training set, i.e., $\mathbf{x}^* \notin \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$. This is also referred as *extrapolation*. Some regression methods can differ in prediction but provide the same results in smoothing, for instance. We show some examples below. Figure 3-(a) provides a graphical representation of the differences between smoothing and prediction problems.

**Remark 6.** *The regression problem can be considered the union of the two important sub-problems, smoothing and prediction.*




(a) (b)

Figure 3: **(a)** Graphical representation of the differences between smoothing and prediction problems. The red crosses represent the solutions of the smoothing problem. The solid blue line represents the prediction at different inputs $\mathbf{x}^*$ which may not belong to the training set. Prediction and smoothing jointly form a complete regression problem. **(b)** Examples of RVM and GP solutions $\widehat{f}(\mathbf{x})$ for interpolation (with $N = 3$ data points).

## 2.3 Interpolation

If we force the conditions $\widehat{f}(\mathbf{x}_n) = y_n$ as shown in Figure 3-(b) (i.e., perfect fitting with the data, a.k.a., interpolation), RVMs and GPs provide the same solution in terms of mean of the posterior $\widehat{f}(\mathbf{x})$ (but different predictive variances). Let us consider an interpolating function of the form

$$\widehat{f}(\mathbf{x}) = \sum_{i=1}^{N} \widehat{\rho}_i \psi_n(\mathbf{x}, \mathbf{x}_i) = \boldsymbol{\psi}(\mathbf{x})^\top \widehat{\boldsymbol{\rho}}, \tag{18}$$

i.e., a linear combination of the nonlinearities $\psi_i(\mathbf{x}, \mathbf{x}_i)$. We would like that $\widehat{f}(\mathbf{x}_n) = y_n$ for all $n = 1, \ldots, N$. Therefore, in order to obtain the proper coefficients $\widehat{\rho}_i$, we can write a $N \times N$ linear system of $N$ conditions of passing through the points $(\mathbf{x}_n, y_n)$,

$$\begin{cases} \widehat{\rho}_1\psi_1(\mathbf{x}_1, \mathbf{x}_1) + \widehat{\rho}_2\psi_2(\mathbf{x}_1, \mathbf{x}_2) + \ldots. + \widehat{\rho}_N\psi_N(\mathbf{x}_1, \mathbf{x}_N) = y_1, \\ \widehat{\rho}_1\psi_1(\mathbf{x}_2, \mathbf{x}_1) + \widehat{\rho}_2\psi_2(\mathbf{x}_2, \mathbf{x}_2) + \ldots. + \widehat{\rho}_N\psi_N(\mathbf{x}_2, \mathbf{x}_N) = y_2, \\ \vdots \\ \widehat{\rho}_1\psi_1(\mathbf{x}_N, \mathbf{x}_1) + \widehat{\rho}_2\psi_2(\mathbf{x}_N, \mathbf{x}_2) + \ldots. + \widehat{\rho}_N\psi_N(\mathbf{x}_N, \mathbf{x}_N) = y_N, \end{cases} \tag{19}$$

i.e., in matrix form $\boldsymbol{\Psi}\widehat{\boldsymbol{\rho}} = \mathbf{y}$. If $\boldsymbol{\Psi}$ is invertible, then we get

$$\widehat{\boldsymbol{\rho}} = [\rho_1, \ldots, \rho_N]^\top = \boldsymbol{\Psi}^{-1}\mathbf{y}. \tag{20}$$

Thus, the interpolative function of both methods can be expressed as

$$\widehat{f}(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top \widehat{\boldsymbol{\rho}} = \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Psi}^{-1}\mathbf{y}. \tag{21}$$

Therefore, by definition we have $\widehat{f}(\mathbf{x}_n) = y_n$ (i.e., $\widehat{f}(\mathbf{x})$ is an interpolator).

# 3 Relevance Vector Machine (RVM)

Following a standard Bayesian approach, we consider a Gaussian prior density over the weights $\boldsymbol{\rho} = [\rho_1, \ldots, \rho_N]^\top$, i.e.,

$$p(\boldsymbol{\rho}) = \mathcal{N}(\boldsymbol{\rho}|\mathbf{0}, \boldsymbol{\Sigma}_\rho), \tag{22}$$

where $\boldsymbol{\Sigma}_\rho$ is an $N \times N$ matrix. Thus, observing Eq. (17), i.e., $\mathbf{y} = \boldsymbol{\Psi}\boldsymbol{\rho} + \mathbf{e}$, we can see that the vector $\mathbf{y}$ is the sum of two independent multivariate Gaussian variables, one with zero mean and covariance matrix $\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top$ and the other one with zero mean and covariance matrix $\sigma_e^2\mathbf{I}_N$. The sum of two independent Gaussian variables is itself a Gaussian variable with mean the sum of the means, and covariance matrix the sum of the covariance matrices, i.e., the marginal likelihood is

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\mathbf{0}, \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2\mathbf{I}_N). \tag{23}$$

## 3.1 Posterior and induced prior distributions of RVM

As we have done with the likelihood functions in Section 2.1.1, in this section we describe posterior distributions of $\boldsymbol{\rho}$ and $f(\mathbf{x})$. Moreover, we derive the induced prior density over $f(\mathbf{x})$.

### 3.1.1 Posterior of the weights $\boldsymbol{\rho}$

Recalling that the likelihood $p(\mathbf{y}|\boldsymbol{\rho}) = \mathcal{N}(\mathbf{y}|\boldsymbol{\Psi}\boldsymbol{\rho}, \sigma_e^2\mathbf{I}_N)$ is Gaussian, the posterior density of the weights is thus proportional to the product of two Gaussians, $p(\boldsymbol{\rho}|\mathbf{y}) \propto p(\mathbf{y}|\boldsymbol{\rho})p(\boldsymbol{\rho})$, and therefore it is also Gaussian:

$$p(\boldsymbol{\rho}|\mathbf{y}) = \frac{1}{p(\mathbf{y})}p(\mathbf{y}|\boldsymbol{\rho})p(\boldsymbol{\rho}) = \mathcal{N}(\boldsymbol{\rho}|\boldsymbol{\mu}_{\rho|y}, \boldsymbol{\Sigma}_{\rho|y}), \tag{24}$$

After some algebra, the mean of the posterior $\boldsymbol{\mu}_{\rho|y} = \widehat{\boldsymbol{\rho}}$ can be expressed in different ways,

$$\begin{aligned}
\boldsymbol{\mu}_{\rho|y} = \widehat{\boldsymbol{\rho}} &= \frac{1}{\sigma_e^2}\left(\frac{1}{\sigma_e^2}\boldsymbol{\Psi}^\top\boldsymbol{\Psi} + \boldsymbol{\Sigma}_\rho^{-1}\right)^{-1}\boldsymbol{\Psi}^\top\mathbf{y}, && (25)\\
&= \left(\boldsymbol{\Psi}^\top\boldsymbol{\Psi} + \sigma_e^2\boldsymbol{\Sigma}_\rho^{-1}\right)^{-1}\boldsymbol{\Psi}^\top\mathbf{y}, && (26)\\
&= \boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2\mathbf{I}_N\right)^{-1}\mathbf{y}, && (27)
\end{aligned}$$

and likewise, the covariance matrix can be written variously as

$$\begin{aligned}\boldsymbol{\Sigma}_{\rho|y} &= \left(\frac{1}{\sigma_e^2}\boldsymbol{\Psi}^\top\boldsymbol{\Psi} + \boldsymbol{\Sigma}_\rho^{-1}\right)^{-1}, &(28)\\ &= \sigma_e^2\left(\boldsymbol{\Psi}^\top\boldsymbol{\Psi} + \sigma_e^2\boldsymbol{\Sigma}_\rho^{-1}\right)^{-1}, &(29)\\ &= \boldsymbol{\Sigma}_\rho - \boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2\mathbf{I}_N\right)^{-1}\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho. &(30)\end{aligned}$$

See [6, 1] and Appendix A for additional details regarding the last equality. Note also that we may substitute $\mathbf{S}^{-1} = \sigma_e^2\boldsymbol{\Sigma}_\rho^{-1}$ into (26). where $\mathbf{S}$ can be interpreted as an inverse of a "signal-to-noise ratio" (SNR), where $\sigma_e^2$ is the noise power and the covariance of the prior $\boldsymbol{\Sigma}_\rho$ plays the role of "power of the signal".

### 3.1.2 Posterior of the function: predictive distribution

The posterior of $f(\mathbf{x})$ in a generic $\mathbf{x} \in \mathcal{X}$, is also Gaussian,

$$p(f(\mathbf{x})|\mathbf{y}) = \mathcal{N}\left(f(\mathbf{x}) \mid \mu_{f|y}(\mathbf{x}), \sigma_{f|y}^2(\mathbf{x})\right),$$

with

$$\begin{aligned}\mu_{f|y}(\mathbf{x}) = \widehat{f}(\mathbf{x}) &= \boldsymbol{\psi}(\mathbf{x})^\top\widehat{\boldsymbol{\rho}}\\ &= \boldsymbol{\psi}(\mathbf{x})^\top\left(\boldsymbol{\Psi}^\top\boldsymbol{\Psi} + \sigma_e^2\boldsymbol{\Sigma}_\rho^{-1}\right)^{-1}\boldsymbol{\Psi}^\top\mathbf{y}, &(31)\\ &= \boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2\mathbf{I}_N\right)^{-1}\mathbf{y},\end{aligned}$$

where have replaced the two possible expressions of $\widehat{\boldsymbol{\rho}}$ in Eqs. (26)–(27), and

$$\begin{aligned}\sigma_{f|y}^2(\mathbf{x}) &= \boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\Sigma}_{\rho|y}\boldsymbol{\psi}(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top\left(\frac{1}{\sigma_e^2}\boldsymbol{\Psi}^\top\boldsymbol{\Psi} + \boldsymbol{\Sigma}_\rho^{-1}\right)^{-1}\boldsymbol{\psi}(\mathbf{x}), &(32)\\ &= \boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\Sigma}_\rho\boldsymbol{\psi}(\mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2\mathbf{I}_N\right)^{-1}\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\psi}(\mathbf{x}).\end{aligned}$$

where we recall that $\boldsymbol{\psi}(\mathbf{x})$ is an $N \times 1$ dimensional vector. For the last equality, see Appendix A.

### 3.1.3 Interpolation with RVM

Considering (26), if we have noisy-free observations $\sigma_e^2 = 0$, we obtain the following expression

$$\begin{aligned}\widehat{\boldsymbol{\rho}} &= \left(\boldsymbol{\Psi}^\top\boldsymbol{\Psi}\right)^{-1}\boldsymbol{\Psi}^\top\mathbf{y},\\ &= \boldsymbol{\Psi}^{-1}\mathbf{y}, &(33)\end{aligned}$$

and the resulting mean function

$$\widehat{f}(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top\widehat{\boldsymbol{\rho}} = \boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\Psi}^{-1}\mathbf{y},$$

is an interpolant, satisfying the passing conditions $\widehat{f}(\mathbf{x}_n) = y_n$, for all $n = 1, \ldots, N$, as described in Section 2.3 and the linear system given in Eqs. (19). Let us begin with the simple case $\boldsymbol{\Sigma}_\rho = \sigma_\rho^2\mathbf{I}_N$. Then, we can define $\text{SNR} = \frac{\sigma_\rho^2}{\sigma_e^2}$ and $\mathbf{S} = \text{SNR}\cdot\mathbf{I}_N$. Note that if $\text{SNR} = \infty$, i.e., if we have noise-free observations $\sigma_e^2 = 0$ or an uninformative prior $\sigma_\rho^2 = \infty$, in both cases we obtain Eq. (33).

**Remark 7.** *With RVM, we can obtain the interpolative solution* $\widehat{\boldsymbol{\rho}} = \boldsymbol{\Psi}^{-1}\mathbf{y}$ *in Section 2.3, either with* $\sigma_e^2 = 0$ *(and a finite* $\sigma_\rho^2$*) or using an uninformative prior over the weights,* $\sigma_\rho^2 = \infty$ *(and* $\sigma_e^2 \neq 0$*).*
In the opposite scenario, with SNR = 0 (if $\sigma_e^2 = \infty$ or $\sigma_\rho^2 = 0$), we have $\widehat{\boldsymbol{\rho}} = \mathbf{0}$. Regarding the predictive variance $\sigma_{f|y}^2(\mathbf{x})$, when $\sigma_e^2 = 0$ we obtain

$$\begin{aligned}\sigma_{f|y}^2(\mathbf{x}) &= \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top \left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\right)^{-1} \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\psi}(\mathbf{x}). \\ &= \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{x}) = 0, \end{aligned} \tag{34}$$

where we have used $\left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\right)^{-1} = \left(\boldsymbol{\Psi}^\top\right)^{-1}\left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\right)^{-1}$. Namely, in the interpolation scenario, the predictive variance of RVM is zero, i.e., $\sigma_{f|y}^2(\mathbf{x}) = 0$, for all $\mathbf{x} \in \mathcal{X}$.

### 3.1.4 Posterior density for the smoothing problem

Considering the smoothing problem, the posterior of the vector $\mathbf{f} = \boldsymbol{\Psi}\boldsymbol{\rho}$ is a multivariate Gaussian pdf, $p(\mathbf{f}|\mathbf{y}) = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}_{f|y}, \boldsymbol{\Sigma}_{f|y})$, where the mean vector is

$$\begin{aligned}\boldsymbol{\mu}_{f|y} &= \widehat{\mathbf{f}} = \boldsymbol{\Psi}\boldsymbol{\mu}_{\rho|y} \\ &= \boldsymbol{\Psi}\left(\boldsymbol{\Psi}^\top\boldsymbol{\Psi} + \sigma_e^2\boldsymbol{\Sigma}_\rho^{-1}\right)^{-1}\boldsymbol{\Psi}^\top\mathbf{y} \\ &= \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2\mathbf{I}_N\right)^{-1}\mathbf{y}, \end{aligned} \tag{35}$$

and covariance matrix is given by

$$\begin{aligned}\boldsymbol{\Sigma}_{f|y} &= \boldsymbol{\Psi}\boldsymbol{\Sigma}_{\rho|y}\boldsymbol{\Psi}^\top, \\ &= \boldsymbol{\Psi}\left(\frac{1}{\sigma_e^2}\boldsymbol{\Psi}^\top\boldsymbol{\Psi} + \boldsymbol{\Sigma}_\rho^{-1}\right)^{-1}\boldsymbol{\Psi}^\top, \\ &= \left[\left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\right)^{-1} + \left(\sigma_e^2\mathbf{I}_N\right)^{-1}\right]^{-1}, \\ &= \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top - \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\left(\sigma_e^2\mathbf{I}_N + \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\right)^{-1}\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top. \end{aligned} \tag{36}$$

For more details, see Appendix A.

### 3.1.5 Induced prior density over the underlying function

Given a test input $\mathbf{x}$ and the vector $\boldsymbol{\psi}(\mathbf{x})$ (choosing and fixing the bases) and considering the random variable $f(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\rho}$ (a scalar value), we can observe that

$$p(f(\mathbf{x})) = \mathcal{N}(f(\mathbf{x})|\mu_f(\mathbf{x}), \sigma_f^2(\mathbf{x})), \quad \text{with} \quad \mu_f(\mathbf{x}) = 0, \quad \sigma_f^2(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\Sigma}_\rho\boldsymbol{\psi}(\mathbf{x}). \tag{37}$$

Therefore, in this probabilistic approach, we directly impose a prior over the weights $\boldsymbol{\rho}$, and we also induce a prior density over the function $f(\mathbf{x})$. Clearly, if we consider the $N \times 1$ vector $\mathbf{f} = \boldsymbol{\Psi}\boldsymbol{\rho}$, we have that

$$p(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}_f, \boldsymbol{\Sigma}_f), \quad \text{with} \quad \boldsymbol{\mu}_f = \mathbf{0}, \quad \boldsymbol{\Sigma}_f = \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top. \tag{38}$$

### 3.1.6 Why it is called a Releveance Vector Machine

Let us consider a prior covariance matrix over the weights of type

$$\boldsymbol{\Sigma}_\rho = \begin{bmatrix} 1/\alpha_1 & 0 & 0 & & 0 \\ 0 & 1/\alpha_2 & 0 & \dots & 0 \\ 0 & 0 & 1/\alpha_3 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1/\alpha_N \end{bmatrix} \tag{39}$$

i.e., $\boldsymbol{\Sigma}_\rho$ is diagonal with elements $[1/\alpha_1, 1/\alpha_2, \dots, 1/\alpha_N]$ in its diagonal. The idea is to use a hierarchical approach considering that also the hyper-parameters $\alpha_i$ are unknown coefficients to be learned. As an example of learning procedure, we can maximize the marginal likelihood with respect to these hyper-parameters. It is possible to show that a significant proportion of the $\{\alpha_i\}$ diverge to infinity. As a consequence, the mean of the posterior in Eq. (26) of the weights $\{\rho_i\}$ corresponding to these "divergent" $\{\alpha_i\}$ is close to zero (with negligible variance). Hence, the basis functions $\psi_i$ associated with these weights $\rho_i$ are virtually pruned out, and the function $\widehat{f}(\mathbf{x})$ depends only on a few bases. Then, the result is a *sparse* model. As the bases are localized around particular training inputs $\mathbf{x}_i$, this learning procedure can be also interpreted as a way of selecting *relevant* inputs. Therefore, RVM can be considered a Bayesian sparse kernel technique.

## 3.2 Random functions according to RVM models

Note that (also in this approach) random functions can be generated from the prior and posterior densities. Indeed, we show different generating procedures in order to draw from the prior and posterior pdf $f(\mathbf{x})$.

**Draw functions from the prior.** Let us consider the following procedure for generating $S$ random functions from the prior pdf:
For $s = 1, \dots, S$:

1. Draw a vector $\boldsymbol{\rho}^{(s)} = [\rho_1^{(s)}, \dots, \rho_N^{(s)}]^\top \sim \mathcal{N}(\boldsymbol{\rho}|\mathbf{0}, \boldsymbol{\Sigma}_\rho)$.
2. Then, set

$$f^{(s)}(\mathbf{x}) = \sum_{n=1}^{N} \rho_n^{(s)} \psi_n(\mathbf{x}, \mathbf{x}_n), \qquad \forall \mathbf{x} \in \mathcal{X}. \tag{40}$$

Note that the procedure above takes into account the correlation (among different $\mathbf{x}$) induced by the model assumptions. See below and Section 6.1 for further details regarding the induced correlation.

**Draw functions from the posterior.** In order to draw functions from the posterior pdf, we can use the following steps:
For $s = 1, \dots, S$:

1. Draw a vector $\boldsymbol{\rho}^{(s)} = [\rho_1^{(s)}, \ldots, \rho_N^{(s)}]^\top \sim \mathcal{N}(\boldsymbol{\rho}|\boldsymbol{\mu}_{\rho|y}, \boldsymbol{\Sigma}_{\rho|y})$ where mean and variance are given in Eqs. (26)–(28).

2. Then, set

$$f^{(s)}(\mathbf{x}) = \sum_{n=1}^{N} \rho_n^{(s)} \psi_n(\mathbf{x}, \mathbf{x}_n), \qquad \forall \mathbf{x} \in \mathcal{X}. \tag{41}$$

Thus, the generation of random function from the prior and posterior density is also possible in RVMs. Hence, this possibility is not only a prerogative of the Gaussian Process (GP) approach in Section 5, as is often hinted in the literature. See Figure 4 for some example of random functions drawn from a RVM model.

**Alternative sampling schemes.** We describe two alternative procedures from drawing from the RVM prior and posterior equivalent to the procedures above. For instance, in order to draw from the RVM prior, we can consider the Eqs. (37)-(38). Let us consider $P$ test points $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(P)}$. Then, the following procedure generates $S$ random functions from a RVM prior:

1. Compute the $N \times P$ matrix $\mathbf{V} = [\boldsymbol{\psi}(\mathbf{x}^{(1)}), \ldots, \boldsymbol{\psi}(\mathbf{x}^{(P)})]$. Recall that $\boldsymbol{\psi}(\mathbf{x}) = [\psi_1(\mathbf{x}, \mathbf{x}_1), \ldots, \psi_N(\mathbf{x}, \mathbf{x}_N)]^\top$ is the $N \times 1$ design vector.

2. Compute the $P \times P$ covariance matrix of the vector $\mathbf{f}_P = [f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(P)})]^\top$,

$$\mathbf{C} = \mathbf{V}\boldsymbol{\Sigma}_\rho \mathbf{V}^\top.$$

3. Draw $S$ vectors $\mathbf{f}_P^{(s)} = [f^{(s)}(\mathbf{x}^{(1)}), \ldots, f^{(s)}(\mathbf{x}^{(P)})]^\top$ from a multivariate Gaussian, i.e.,

$$\mathbf{f}_P^{(s)} \sim \mathcal{N}(\mathbf{f}_P|\mathbf{0}, \mathbf{C}), \quad s = 1, \ldots, S,$$

   where $\mathbf{0}$ is a $P \times 1$ null vector and $\mathbf{C}$ is given above.

In the same fashion, considering again $P$ test inputs $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(P)}$ and the Eqs. (31)–(32) and Eqs. (35)–(36), we can consider the following procedure for sampling from the RVM posterior:

1. Compute the $N \times P$ matrix $\mathbf{V} = [\boldsymbol{\psi}(\mathbf{x}^{(1)}), \ldots, \boldsymbol{\psi}(\mathbf{x}^{(P)})]$. Recall that $\boldsymbol{\psi}(\mathbf{x}) = [\psi_1(\mathbf{x}, \mathbf{x}_1), \ldots, \psi_N(\mathbf{x}, \mathbf{x}_N)]^\top$ is the $N \times 1$ design vector.

2. Compute the $P \times P$ covariance matrix of the vector $\mathbf{f}_P = [f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(P)})]^\top$,

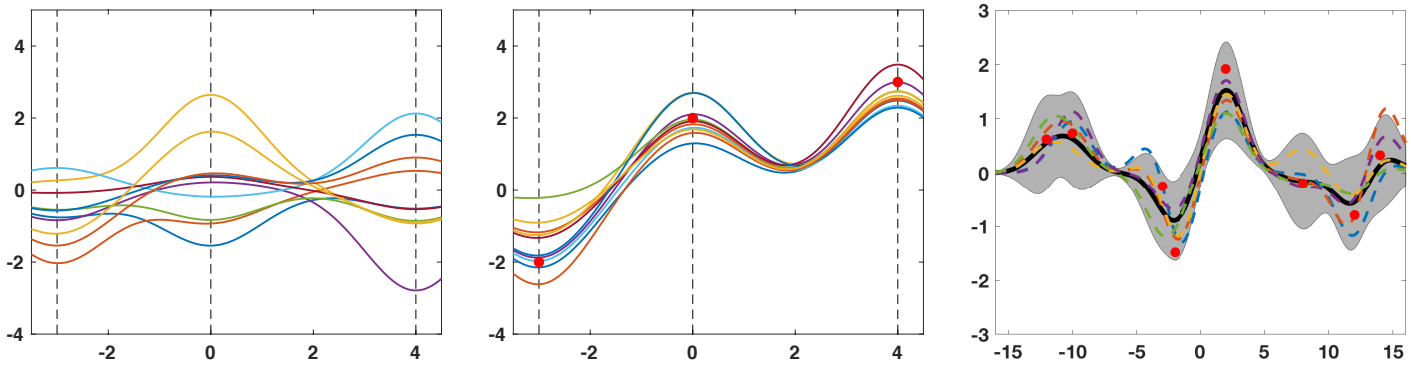
$$\mathbf{C} = \mathbf{V}\boldsymbol{\Sigma}_\rho \mathbf{V}^\top.$$

3. Draw $S$ vectors $\mathbf{f}_P^{(s)} = [f^{(s)}(\mathbf{x}^{(1)}), \ldots, f^{(s)}(\mathbf{x}^{(P)})]^\top$ from a multivariate Gaussian, i.e.,

$$\mathbf{f}_P^{(s)} \sim \mathcal{N}(\mathbf{f}_P|\boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad s = 1, \ldots, S,$$

   where

$$\begin{aligned} \boldsymbol{\mu} &= \mathbf{V}\boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top \left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top + \sigma_e^2 \mathbf{I}_N\right)^{-1} \mathbf{y}, && \text{is a } P \times 1 \text{ vector and,} \\ \boldsymbol{\Sigma} &= \mathbf{C} - \mathbf{V}\boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top \left(\sigma_e^2 \mathbf{I}_N + \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top\right)^{-1} \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho \mathbf{V}^\top, && \text{is a } P \times P \text{ covariance matrix.} \end{aligned}$$



(a) (b) (c)

Figure 4: Random functions $\widehat{f}^{(s)}(\mathbf{x})$ with $s = 1, \ldots, 10$ **(a)** from a RVM prior over $f(\mathbf{x})$ and **(b)** from a RVM posterior (after knowing the $N = 3$ data points), with $N = 3$ Gaussian bases with the mean location depicted by the dashed lines (and bandwidth $\lambda = 2$). We have considered a diagonal covariance matrix $\boldsymbol{\Sigma}_\rho$ with all the elements in the diagonal equal to 2.25. **(c)** Example of RVM mean and variance with $N = 8$ data points and $S = 5$ random functions from the posterior depicted with dashed lines ($\sigma_e = 0.5$, $\lambda = 4$, $\boldsymbol{\Sigma}_\rho$ diagonal all the elements equal to 1). The black solid line shows the mean $\mu_{f|y}(\mathbf{x}) = \widehat{f}(\mathbf{x})$ and the boundary of the grey area corresponds to $\widehat{f}(\mathbf{x}) \pm 2\sigma^2_{f|y}(\mathbf{x})$ (i.e., $\approx 95\%$ of the probability).

# 4 Probabilistic Kernel Ridge Regression: the Quasi GP model

Quasi Gaussian Process (Q-GP) model is an intermediate model between RVM and GP, which can be also useful for achieving a better understanding of both. The Q-GP model represents the probabilistic version of the so-called *Kernel Ridge Regression* [1, 19]. Therefore, Q-GP is a probabilistic version of the Kernel Ridge regression, sometimes called *Bayesian Ridge Regression*. Q-GP is also related to the so-called *regularization networks* in the literature [18]. We highlight in advance the connections with RVM for helping the reader.

**Remark 8.** *Q-GP is a special case of RVM with a specific choice of* $\boldsymbol{\Sigma}_\rho = \boldsymbol{\Psi}^{-1}$ *(i.e., the covariance prior over* $\boldsymbol{\rho}$*). Note that, we need* $\boldsymbol{\Psi} = \boldsymbol{\Psi}^\top$*, unlike in RVM.*

**Remark 9.** *In the smoothing problem, Q-GP can be also derived with a specific choice of* $\boldsymbol{\Psi}$ *as covariance prior over* $\mathbf{f}$*.*

## 4.1 Q-GP solution for regression

We consider the same classical Bayesian approach used for RVMs. Namely, the observation model is again $y_i = \boldsymbol{\psi}(\mathbf{x}_i)^\top \boldsymbol{\rho} + e_i$, as in Eq. (14). However, in this case, we assume the following prior

over the vector of weights,

$$p(\boldsymbol{\rho}) = \mathcal{N}(\boldsymbol{\rho}|\mathbf{0}, \boldsymbol{\Psi}^{-1}) \propto \exp(-\boldsymbol{\rho}^\top \boldsymbol{\Psi} \boldsymbol{\rho}), \tag{42}$$

i.e., $\boldsymbol{\Sigma}_\rho = \boldsymbol{\Psi}^{-1}$. This is possible just if $\boldsymbol{\Psi}^{-1}$ is a covariance matrix, so that it must be positive definite and symmetric, hence $\boldsymbol{\Psi} = \boldsymbol{\Psi}^\top$. The rest of formulas can be obtained replacing $\boldsymbol{\Sigma}_\rho = \boldsymbol{\Psi}^{-1}$ and $\boldsymbol{\Psi} = \boldsymbol{\Psi}^\top$, in the RVM expressions. For instance, replacing $\boldsymbol{\Sigma}_\rho = \boldsymbol{\Psi}^{-1}$ in Eq.(27), we have $p(\boldsymbol{\rho}|\mathbf{y}) = \mathcal{N}(\boldsymbol{\rho}|\boldsymbol{\mu}_{\rho|y}, \boldsymbol{\Sigma}_{\rho|y})$, where

$$\begin{aligned} \widehat{\boldsymbol{\rho}} = \boldsymbol{\mu}_{\rho|y} &= \left(\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N\right)^{-1} \mathbf{y} \\ \boldsymbol{\Sigma}_{\rho|y} &= \boldsymbol{\Psi}^{-1} - \left(\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N\right)^{-1}, \end{aligned} \tag{43}$$

The marginal likelihood is $p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\mathbf{0}, \boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)$, and the posterior of $f(\mathbf{x})$ in a generic $\mathbf{x} \in \mathcal{X}$, is also Gaussian,

$$p(f(\mathbf{x})|\mathbf{y}) = \mathcal{N}\left(f(\mathbf{x})|\mu_{f|y}(\mathbf{x}), \sigma^2_{f|y}(\mathbf{x})\right),$$

with

$$\mu_{f|y}(\mathbf{x}) = \widehat{f}(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top \left(\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N\right)^{-1} \mathbf{y}, \tag{44}$$

and

$$\sigma^2_{f|y}(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Psi}^{-1} \boldsymbol{\psi}(\mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top \left(\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N\right)^{-1} \boldsymbol{\psi}(\mathbf{x}). \tag{45}$$

**Remark 10.** *We will see, that the mean solution $\widehat{f}(\mathbf{x})$ of Q-GP coincides perfectly with the standard GP solution, that we will describe later. However, Q-GP and GP differ for the expression of variance $\sigma^2_{f|y}(\mathbf{x})$, for a generic $\mathbf{x}$ (as we will show later). For further details, see below and [25].*

## 4.2 Q-GP for smoothing

For obtaining the expressions for the smoothing scenario, we can replace the vector $\boldsymbol{\psi}(\mathbf{x})^\top$ with the matrix $\boldsymbol{\Psi}$ in the formulas above. However, in the smoothing case, Q-GP can be directly derived assuming a particular prior over $f(\mathbf{x})$. Although the formulas of the Q-GP for smoothing can be obtained as particular case of the expressions above, we repeat the derivation since it can be useful for understanding the classical GP derivation (in the next section). We will use again the previous standard Bayesian approach, but now we focus on removing noise of the observations $y_n$ at the inputs $\mathbf{x}_n$ obtaining $\widehat{\mathbf{f}}$, and we will consider a specific covariance prior over $\mathbf{f}$. More specifically, we consider a Gaussian prior over the vector $\mathbf{f}$ (i.e., a prior over the *hidden function*),

$$\mathbf{f} \sim \mathcal{N}(\mathbf{f}|\mathbf{0}, \boldsymbol{\Psi}) \propto \exp(-\mathbf{f}^\top \boldsymbol{\Psi}^{-1} \mathbf{f}), \tag{46}$$

where $\boldsymbol{\Psi}$ is exactly the design matrix in Eq. (4). This is possible only if $\boldsymbol{\Psi}$ can represent a covariance matrix, then $\boldsymbol{\Psi}$ must be positive definite and symmetric, $\boldsymbol{\Psi} = \boldsymbol{\Psi}^\top$. Therefore, we need that $\psi_n(\mathbf{x}|\mathbf{z}) = \psi_n(\mathbf{z}|\mathbf{x})$. Recall that the observation model has the form

$$\mathbf{y} = \mathbf{f} + \mathbf{e}.$$

We are interesting in inferring $\mathbf{f}$ given $\mathbf{y}$ under the assumption $\mathbf{e} \sim \mathcal{N}(\mathbf{e}|\mathbf{0}, \sigma_e^2 \mathbf{I}_N)$. Then, the likelihood is

$$p(\mathbf{y}|\mathbf{f}) = \mathcal{N}(\mathbf{e}|\mathbf{f}, \sigma_e^2 \mathbf{I}_N). \tag{47}$$

Hence, the marginal likelihood is

$$p(\mathbf{y}) = \int_{\mathbb{R}^N} p(\mathbf{y}|\mathbf{f}) p(\mathbf{f}) d\mathbf{f} = \mathcal{N}(\mathbf{y}|\mathbf{0}, \boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N). \tag{48}$$

The posterior of the vector $\mathbf{f}$ is

$$\begin{aligned} p(\mathbf{f}|\mathbf{y}) &= \frac{1}{p(\mathbf{y})} p(\mathbf{y}|\mathbf{f}) p(\mathbf{f}) \propto p(\mathbf{y}|\mathbf{f}) p(\mathbf{f}) & (49) \\ &= \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}_{f|y}, \boldsymbol{\Sigma}_{f|y}), & (50) \end{aligned}$$

where the vector mean $\boldsymbol{\mu}_{f|y} = \widehat{\mathbf{f}}$ and covariance matrix are

$$\begin{aligned} \boldsymbol{\mu}_{f|y} &= \widehat{\mathbf{f}} = \boldsymbol{\Psi}(\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \mathbf{y}, \quad \text{and} & (51) \\ \boldsymbol{\Sigma}_{f|y} &= \left[ (\boldsymbol{\Psi})^{-1} + \left(\sigma_e^2 \mathbf{I}_N\right)^{-1} \right]^{-1} \\ &= \boldsymbol{\Psi} - \boldsymbol{\Psi} \left(\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N\right)^{-1} \boldsymbol{\Psi}. & (52) \end{aligned}$$

**Remark 11.** *The solution above of Q-GP for smoothing, represented by Eqs.* (51)–(52)*, coincides perfectly with the standard GP solution for smoothing, that we will describe below. See Eqs.* (67)–(68)*.*

# 5 Gaussian Processes (GPs)

## 5.1 Definition

For simplicity, let us consider $\psi_n(\mathbf{x}, \mathbf{x}_n) = \psi(\mathbf{x}, \mathbf{x}_n)$. Moreover, let us assume that the nonlinearity $\psi$ is chosen such that (a) $\psi(\mathbf{x}, \mathbf{x}) > 0$, (b) the design matrix $\boldsymbol{\Psi} = \boldsymbol{\Psi}^\top$ (symmetric) and (c) $\boldsymbol{\Psi}$ is a positive-definite matrix. In this case, we can interpret the design matrix $\boldsymbol{\Psi}$ as a covariance matrix (as we have assumed in Section 4.1). As in Section 4.2, we can consider a Gaussian prior over the vector $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]^\top$, e.g.,

$$\mathbf{f} \sim \mathcal{N}(\mathbf{f}|\mathbf{0}, \boldsymbol{\Psi}) \propto \exp(-\mathbf{f}^\top \boldsymbol{\Psi}^{-1} \mathbf{f}), \tag{53}$$

assuming a zero mean vector for the sake of simplicity. We can generalize the idea above, for two generic inputs $\mathbf{x}$ and $\mathbf{z}$, which are not (necessarily) in the input dataset $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$. Let consider that the function $\psi$ is symmetric, i.e., $\psi(\mathbf{x}, \mathbf{z}) = \psi(\mathbf{z}, \mathbf{x})$ for all $\mathbf{x}, \mathbf{z} \in \mathcal{X}$ and represents a *covariance function* [6, 1] (see below).

**Remark 12.** *We are assuming that function $\psi(\mathbf{x}, \mathbf{z})$ represents the covariance function between the random variables $f(\mathbf{x})$ and $f(\mathbf{z})$, at two generic inputs $\mathbf{x}$ and $\mathbf{z}$. Namely,*

$$\psi(\mathbf{x}, \mathbf{z}) = E[(f(\mathbf{x}) - \mu(\mathbf{x}))(f(\mathbf{z}) - \mu(\mathbf{z}))], \tag{54}$$

*where we have assumed for simplicity* $\mu(\mathbf{x}) = 0$, $\mu(\mathbf{z}) = 0$. *Thus,* $\psi(\mathbf{x}, \mathbf{x}) = E[(f(\mathbf{x}) - \mu(\mathbf{x}))^2]$ *is the variance the random variable* $f(\mathbf{x})$, *i.e,* $p(f(\mathbf{x})) \sim \mathcal{N}(0, \psi(\mathbf{x}, \mathbf{x}))$.

Thus, we can assume that the two random variables $f(\mathbf{x})$ and $f(\mathbf{z})$ are jointly Gaussian, with mean $[0, 0]^\top$ and $2 \times 2$ covariance matrix

$$\mathbf{C}(f(\mathbf{x}), f(\mathbf{z})) = \begin{bmatrix} \psi(\mathbf{x}, \mathbf{x}) & \psi(\mathbf{z}, \mathbf{x}) \\ \psi(\mathbf{x}, \mathbf{z}) & \psi(\mathbf{z}, \mathbf{z}) \end{bmatrix}. \tag{55}$$

Considering 3 generic inputs $\mathbf{x}, \mathbf{z}, \mathbf{t} \in \mathcal{X}$, we have the following covariance matrix

$$\mathbf{C}(f(\mathbf{x}), f(\mathbf{z}), f(\mathbf{t})) = \begin{bmatrix} \psi(\mathbf{x}, \mathbf{x}) & \psi(\mathbf{z}, \mathbf{x}) & \psi(\mathbf{t}, \mathbf{x}) \\ \psi(\mathbf{x}, \mathbf{z}) & \psi(\mathbf{z}, \mathbf{z}) & \psi(\mathbf{t}, \mathbf{z}) \\ \psi(\mathbf{x}, \mathbf{t}) & \psi(\mathbf{z}, \mathbf{t}) & \psi(\mathbf{t}, \mathbf{t}) \end{bmatrix}. \tag{56}$$

Moreover, considering all the input dataset $\{\mathbf{x}_n\}_{n=1}^N$, we have that

$$\mathbf{C}(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N) = \mathbf{\Psi}.$$

**Marginal Likelihood.** Recalling that $\mathbf{y} = \mathbf{f} + \mathbf{e}$, then the marginal likelihood is again

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\mathbf{0}, \mathbf{\Psi} + \sigma_e^2 \mathbf{I}_N). \tag{57}$$

where we have the sum of two independent multivariate Gaussian random variables $\mathbf{f} \sim \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{\Psi})$ in Eq. (53) and $\mathbf{e} \sim \mathcal{N}(\mathbf{e}|\mathbf{0}, \sigma_e^2 \mathbf{I}_N)$.

**Remark 13.** *If the kernel function is stationary* $\psi(\mathbf{x}, \mathbf{z}) = \psi(||\mathbf{x} - \mathbf{z}||)$, *we are converting the distance between the inputs* $\mathbf{x}$ *and* $\mathbf{z}$, *into an priori covariance/correlation information. Often, we associate small correlation to high distances and high correlation to small distances.*

**Definition.** "A Gaussian process (GP) is a collection of random variables, any finite number of which have a joint Gaussian distribution" [6]. A GP is completely specified by its mean function $m(\mathbf{x}) : \mathcal{X} \to \mathbb{R}$ (that we have assumed $m(\mathbf{x}) = 0$, for simplicity) and its covariance function $\psi(\mathbf{x}, \mathbf{z}) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

A graphical representation of a GP prior considering a vector of dimension $N = 4$ is given in Figure 5.

## 5.2 Posterior density of a GP in regression

Let us continue the derivation of the main GP formulas for regression, considering the joint probability $p(\mathbf{y}, f(\mathbf{x}))$ where $\mathbf{x} \in \mathcal{X}$ is a generic input and $\mathbf{y} = \mathbf{f} + \mathbf{e}$ (recall that $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]^\top$). By the GP definition,

$$p(\mathbf{y}, f(\mathbf{x})) = \mathcal{N}([\mathbf{y}, f(\mathbf{x})]^\top | \boldsymbol{\mu}_{\texttt{joint}}, \mathbf{C}_{\texttt{joint}}) \tag{58}$$

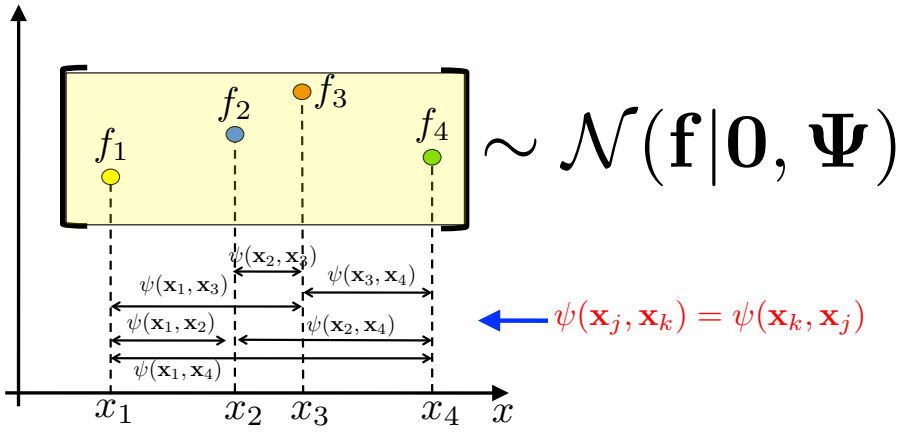



Figure 5: Graphical representation of a GP prior idea with $N = 4$ and $\mathbf{\Psi} = \mathbf{C}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4)$.

where $\boldsymbol{\mu}_{\texttt{joint}} = [0, \ldots, 0]^\top$ is a null vector of dimension $(N+1)$, and

$$\mathbf{C}_{\texttt{joint}} = \mathbf{C}(\mathbf{y}, f(\mathbf{x})) = \begin{bmatrix} \mathbf{\Psi} + \sigma_e^2 \mathbf{I}_N & \boldsymbol{\psi}(\mathbf{x}) \\ \boldsymbol{\psi}(\mathbf{x})^\top & \psi(\mathbf{x}, \mathbf{x}) \end{bmatrix}, \tag{59}$$

is a $(N+1) \times (N+1)$ covariance matrix. We recall that

$$\boldsymbol{\psi}(\mathbf{x}) = [\psi(\mathbf{x}, \mathbf{x}_1), \ldots, \psi(\mathbf{x}, \mathbf{x}_N)]^\top,$$

is the $N \times 1$ design vector. The first bock in the diagonal of $\mathbf{C}_{\texttt{joint}}$ is the covariance $N \times N$ matrix of $\mathbf{y}$ i.e., $\mathbf{C}(\mathbf{y}, \mathbf{y}) = \mathbf{\Psi} + \sigma_e^2 \mathbf{I}_N$, and the last element in the diagonal is $\text{var}[f(\mathbf{x})] = \psi(\mathbf{x}, \mathbf{x})$. The covariance of each element $y_n$ of the vector $\mathbf{y} = [y_1, \ldots, y_N]^\top$, and the random variable $f(\mathbf{x})$ is $\psi(\mathbf{x}_n, \mathbf{x}) = \psi(\mathbf{x}, \mathbf{x}_n)$. All those $N$ covariances are contained in the vector $\boldsymbol{\psi}(\mathbf{x})$. We will use the following property of the Gaussian distributions,

$$p(\mathbf{a}, \mathbf{b}) \sim \mathcal{N}\left( [\mathbf{a}, \mathbf{b}]^\top \middle| [\boldsymbol{\mu}_a, \boldsymbol{\mu}_b]^\top, \begin{bmatrix} \mathbf{C}_a & \mathbf{\Lambda} \\ \mathbf{\Lambda} & \mathbf{C}_b \end{bmatrix}, \right) \tag{60}$$

then the conditional pdf $p(\mathbf{b}|\mathbf{a}) = \mathcal{N}(\mathbf{b}|\boldsymbol{\mu}_{b|a}, \mathbf{C}_{b|a})$ has the following mean and variance,

$$\boldsymbol{\mu}_{b|a} = \boldsymbol{\mu}_b + \mathbf{\Lambda}^\top \mathbf{C}_a^{-1}(\mathbf{a} - \boldsymbol{\mu}_a), \qquad \mathbf{C}_{b|a} = \mathbf{C}_b - \mathbf{\Lambda}^\top \mathbf{C}_a^{-1} \mathbf{\Lambda}. \tag{61}$$

Hence, given the joint probability in (58), now we can obtain the mean and variance of posterior pdf

$$p(f(\mathbf{x})|\mathbf{y}) = \mathcal{N}\left(f(\mathbf{x})|\mu_{f|y}(\mathbf{x}), \sigma^2_{f|y}(\mathbf{x})\right).$$

Thus, in this case, we have $\mathbf{a} = \mathbf{y}$, $\boldsymbol{\mu}_a$, $\boldsymbol{\mu}_b$ are zero, $\mathbf{C}_a = \mathbf{\Psi} + \sigma_e^2 \mathbf{I}_N$, $\mathbf{\Lambda} = \boldsymbol{\psi}(\mathbf{x})$ and $\mathbf{C}_b = \psi(\mathbf{x}|\mathbf{x})$ (i.e., a scalar in this case),

$$\mu_{f|y}(\mathbf{x}) = \widehat{f}(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top (\mathbf{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \mathbf{y}, \tag{62}$$

$$\sigma^2_{f|y}(\mathbf{x}) = \psi(\mathbf{x}, \mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top (\mathbf{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \boldsymbol{\psi}(\mathbf{x}). \tag{63}$$

**Remark 14.** *Note that, also in this case,* $\widehat{f}(\mathbf{x})$ *can be expressed as Eq.* (3)*, i.e.,* $\widehat{f}(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top \widehat{\boldsymbol{\rho}}$ *where*

$$\widehat{\boldsymbol{\rho}} = (\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1}\mathbf{y}. \tag{64}$$

**Remark 15.** *Also in the GP formulation, with noise-free data* $\sigma_e^2 = 0$ *(interpolation), we come back to* $\widehat{\boldsymbol{\rho}} = \boldsymbol{\Psi}^{-1}\mathbf{y}$*, as expected.*

**Posterior of several test points.** The formulas (62)–(63) can be easily generalized when we consider the posterior distribution of the hidden function in $P$ different generic test points

$$f(\mathbf{x}^{(1)}), f(\mathbf{x}^{(2)}), \ldots, f(\mathbf{x}^{(P)}).$$

In this case, we have to replace above the $N \times 1$ vector $\boldsymbol{\psi}(\mathbf{x})$ with the $N \times P$ matrix $\mathbf{V} = [\boldsymbol{\psi}(\mathbf{x}^{(1)}), \ldots, \boldsymbol{\psi}(\mathbf{x}^{(P)})]$, and the scalar value $\psi(\mathbf{x}|\mathbf{x})$ with the $P \times P$ covariance matrix

$$\mathbf{C} = \mathbf{C}(f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(P)})) = \begin{bmatrix} \psi(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) & \cdots & \psi(\mathbf{x}^{(P)}, \mathbf{x}^{(1)}) \\ \psi(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) & \cdots & \psi(\mathbf{x}^{(P)}, \mathbf{x}^{(2)}) \\ \vdots & \cdots & \vdots \\ \psi(\mathbf{x}^{(1)}, \mathbf{x}^{(P)}) & \cdots & \psi(\mathbf{x}^{(P)}, \mathbf{x}^{(P)}) \end{bmatrix}.$$

Then, the posterior pdf is a multivariate Gaussian with the following $P \times 1$ mean vector and $P \times P$ covariance matrix

$$\boldsymbol{\mu}_{f|y} = \mathbf{V}^\top (\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1}\mathbf{y}, \tag{65}$$

$$\boldsymbol{\Sigma}_{f|y} = \mathbf{C} - \mathbf{V}^\top (\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1}\mathbf{V}. \tag{66}$$

where $\boldsymbol{\mu}_{f|y} = [\widehat{f}(\mathbf{x}^{(1)}), \ldots, \widehat{f}(\mathbf{x}^{(P)})]^\top$.

**Smoothing case.** If we consider the training inputs $\mathbf{x}_1, \ldots, \mathbf{x}_N$, then

$$\mathbf{C}(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)) = \boldsymbol{\Psi}.$$

The posterior of the vector $\mathbf{f}$ is

$$p(\mathbf{f}|\mathbf{y}) = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}_{f|y}, \boldsymbol{\Sigma}_{f|y}),$$

where

$$\boldsymbol{\mu}_{f|y} = \widehat{\mathbf{f}} = \boldsymbol{\Psi}(\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1}\mathbf{y}, \tag{67}$$

$$\boldsymbol{\Sigma}_{f|y} = \boldsymbol{\Psi} - \boldsymbol{\Psi}(\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1}\boldsymbol{\Psi}. \tag{68}$$

Note that we have used $\boldsymbol{\Psi} = \boldsymbol{\Psi}^\top$.

**Remark 16.** *The expressions* (67)–(68) *are exactly the same as in Eqs.* (51)–(52) *of the Q-GP.*

## 5.3 Generation of random functions according to GP models

**Drawing functions from the GP prior.** Let us consider that we desire to know (and then, e.g., to plot) a random function $f(\mathbf{x})$ in the input points $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(P)}$. Then, the following procedure generates $S$ random "functions" (represented as random vectors) from a GP prior with kernel function $\psi(\mathbf{x}, \mathbf{z})$:

1. Compute the $P \times P$ covariance matrix

$$\mathbf{C} = \mathbf{C}(f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(P)})) = \begin{bmatrix} \psi(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) & \vdots & \psi(\mathbf{x}^{(P)}, \mathbf{x}^{(1)}) \\ \vdots & \ddots & \vdots \\ \psi(\mathbf{x}^{(1)}, \mathbf{x}^{(P)}) & \cdots & \psi(\mathbf{x}^{(P)}, \mathbf{x}^{(P)}) \end{bmatrix}.$$

   Recall that, for simplicity, we are assuming $\boldsymbol{\mu} = [\mu(\mathbf{x}^{(1)}), \ldots, \mu(\mathbf{x}^{(P)})]^\top = [0, \ldots, 0]^\top$.

2. Draw $S$ vectors $\mathbf{f}_P^{(s)} = [f^{(s)}(\mathbf{x}^{(1)}), \ldots, f^{(s)}(\mathbf{x}^{(P)})]^\top$ from a multivariate Gaussian with zero mean and covariance matrix $\mathbf{C}$ above, i.e.,

$$\mathbf{f}_P^{(s)} \sim \mathcal{N}(\mathbf{f}_P | \mathbf{0}, \mathbf{C}), \quad s = 1, \ldots, S.$$

**Drawing functions from the GP posterior.** Let us consider the test points $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(P)}$. Then, the following procedure generates $S$ random "functions" (that actually are random vectors) from a GP prior with kernel function $\psi(\mathbf{x}, \mathbf{z})$:

1. Compute the $N \times P$ matrix $\mathbf{V} = [\boldsymbol{\psi}(\mathbf{x}^{(1)}), \ldots, \boldsymbol{\psi}(\mathbf{x}^{(P)})]$. Recall that $\boldsymbol{\psi}(\mathbf{x}) = [\psi_1(\mathbf{x}, \mathbf{x}_1), \ldots, \psi_N(\mathbf{x}, \mathbf{x}_N)]^\top$ is the $N \times 1$ design vector.

2. Compute the $P \times P$ covariance matrix

$$\mathbf{C} = \mathbf{C}(f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(P)})) = \begin{bmatrix} \psi(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) & \cdots & \psi(\mathbf{x}^{(P)}, \mathbf{x}^{(1)}) \\ \psi(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) & \cdots & \psi(\mathbf{x}^{(P)}, \mathbf{x}^{(2)}) \\ \vdots & \vdots & \vdots \\ \psi(\mathbf{x}^{(1)}, \mathbf{x}^{(P)}) & \cdots & \psi(\mathbf{x}^{(P)}, \mathbf{x}^{(P)}) \end{bmatrix}.$$

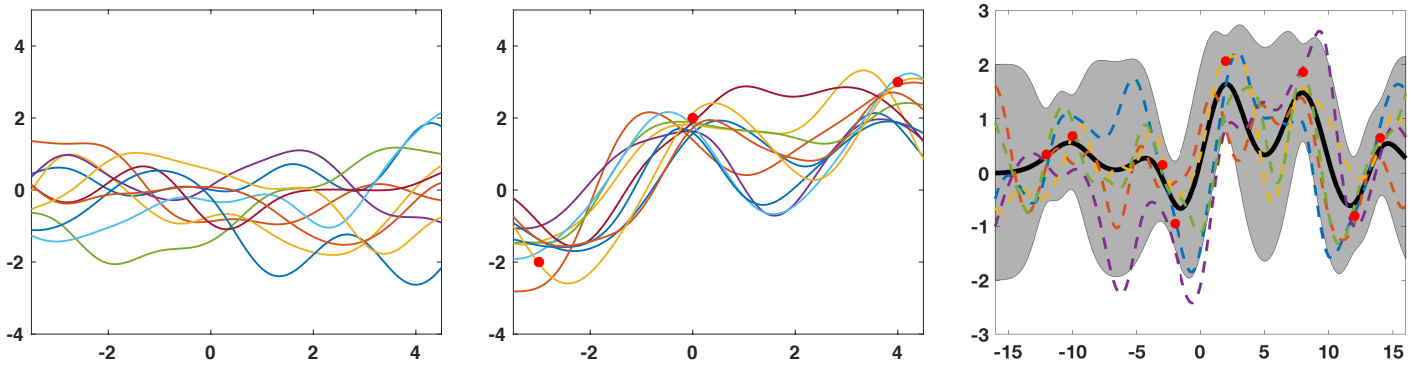
3. Draw $S$ vectors $\mathbf{f}_P^{(s)} = [f^{(s)}(\mathbf{x}^{(1)}), \ldots, f^{(s)}(\mathbf{x}^{(P)})]^\top$ from a multivariate Gaussian, i.e.,

$$\mathbf{f}_P^{(s)} \sim \mathcal{N}(\mathbf{f}_P | \boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad s = 1, \ldots, S,$$

   where

$$\begin{aligned} \boldsymbol{\mu} &= \mathbf{V}^\top (\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \mathbf{y} && \text{is a } P \times 1 \text{ mean vector, and} \\ \boldsymbol{\Sigma} &= \mathbf{C} - \mathbf{V}^\top (\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \mathbf{V} && \text{is a } P \times P \text{ covariance matrix.} \end{aligned}$$

Figure 6 shows some examples of random functions, predictive mean and variance of GP model with a Gaussian kernel function.



(a) (b) (c)

Figure 6: Random functions $\widehat{f}^{(s)}(\mathbf{x})$ with $\mathbf{x} = x \in \mathbb{R}$ scalar, and $s = 1, \ldots, 10$ **(a)** from a GP prior over $f(\mathbf{x})$ and **(b)** from a GP posterior (after knowing the $N = 3$ data points), with $N = 3$ Gaussian kernels and bandwidth $\lambda = 2$, $\sigma_e = 0.5$. **(c)** Example of GP mean and variance with $N = 8$ data points and $S = 5$ random functions from the posterior depicted with dashed lines ($\sigma_e = 0.5$, $\lambda = 4$). The black solid line shows the mean $\mu_{f|y}(\mathbf{x}) = \widehat{f}(\mathbf{x})$ and the boundary of the grey area corresponds to $\widehat{f}(\mathbf{x}) \pm 2\sigma^2_{f|y}(\mathbf{x})$ (i.e., $\approx 95\%$ of the probability).

## 5.4 Interpretation of the hyper-parameters

The kernel hyper-parameters can be learned from data, maximizing the marginal likelihood or by a cross-validation (CV) approach. In some cases, they are also interpretable in statistical terms [26]. As an example, let $x \in \mathbb{R}$ and consider the following exponential kernel function

$$k(x_i, x_j) = a \exp\left(-\frac{|x_i - x_j|^\beta}{\lambda}\right) + v_1 + v_2 \cdot \delta_{ij} \tag{69}$$

where $a, \lambda, v_1, v_2, \beta > 0$. Moreover, $\delta_{ij} = 1$ when $i = j$ and zero otherwise. The statistical interpretation of each parameter is:

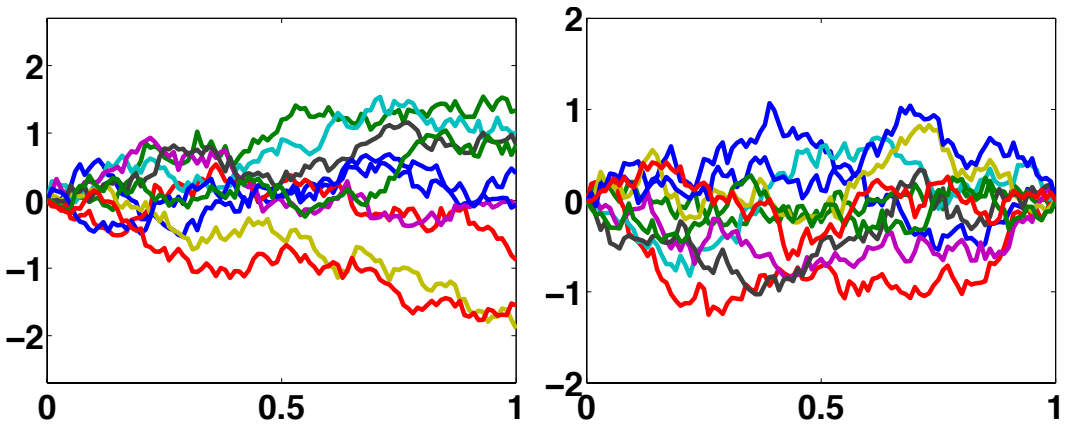
- $a$ : *a-priori signal variance,* i.e., the prior variance that the random function $f(x)$ has following the user's belief, without knowing any data. In regions where there are no data points, the posterior/predictive variance will be $a$.
- $\lambda$ : *lengthscale.* This parameter determines the oscillations that the solution has. The optimal $\lambda$ becomes usually smaller as the number of data points grows and becoming closer and closer.
- $\beta$ : *roughness.* This parameter determines the derivability and the smoothness of the resulting solution.
- $v_1$ : *variance of bias.*
- $v_2$ : *additional noise power.* Since we consider the parameter $\sigma_e^2$, we can avoid the use of $v_2$.

### 5.5 Relevant GP special cases

For simplicity, let us assume again a scalar input, $x \in \mathbb{R}$. Different well-known stochastic processes are GPs [27, Chapter 6]. Table 2 provides some examples, clarifying the specific choice of the mean $\mu(x)$ and covariance function $\psi(x, z)$ of the GP prior. Recall that, in this work, we have always considered $\mu(x) = 0$ for the sake of simplicity. Figure 7 depicts ten realizations of a Wiener process and of a standard Brownian bridge.

Table 2: Special cases of Gaussian processes.

| **Type** | **Mean** $\mu(x)$ | **Covariance function** $\psi(x, z)$ |
|---|---|---|
| *Wiener process* | 0 | $\min\{x, z\}$ |
| *Standard Brownian bridge* | 0 | $\min\{x, z\} - xz$ |
| *Ornstein-Uhlenbeck process* | $e^{-\theta x}\mu_0 + \nu(1 - e^{-\theta x})$ | $\frac{\sigma^2}{2\theta} e^{-\theta(x+z)} \left(e^{2\theta \min\{x,z\}} - 1\right)$ |




(a) (b)

Figure 7: Ten independent realizations **(a)** of a Wiener process and **(b)** of a a standard Brownian bridge.

# 6 Dual representation of RVM - Dual Gaussian Process

In this section, we show that RVM method can be seen also a GP model with a specific choice of the kernel function. Namely, RVM is also a GP. However, the fact of choosing *just indirectly* the kernel function can provides undesirable behavior in the predictive variance, as we discuss in Section 7. Below, we obtain the covariance function (i.e., the kernel function) of RVM and related vectors and matrices.

## 6.1 Covariance function of RVM

We can compute the covariance between the random variables $f = f(\mathbf{x})$ and $f' = f(\mathbf{x}')$. Indeed, recalling that $f(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\rho}$, we can write

$$\begin{aligned} k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}') &= E[f(\mathbf{x}) - \mu_f, f(\mathbf{x}') - \mu_f] \\ &= E[f(\mathbf{x}), f(\mathbf{x}')] \\ &= E[\boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\rho}, \boldsymbol{\psi}(\mathbf{x}')^\top \boldsymbol{\rho}] \\ &= \boldsymbol{\psi}(\mathbf{x})^\top E[\boldsymbol{\rho}, \boldsymbol{\rho}] \boldsymbol{\psi}(\mathbf{x}'), \\ &= \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{x}') : \mathcal{X} \times \mathcal{X} \to \mathbb{R}. \end{aligned} \tag{70}$$

where we have also used the fact that $E[\boldsymbol{\rho}] = 0$, hence $E[\boldsymbol{\rho}, \boldsymbol{\rho}] = \boldsymbol{\Sigma}_\rho$. The function $k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}')$ is called *dual kernel function*, and it is also symmetric, i.e.,

$$\begin{aligned} k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}') &= \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{x}') \\ &= \boldsymbol{\psi}(\mathbf{x}')^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{x}) = k_{\texttt{dual}}(\mathbf{x}', \mathbf{x}). \end{aligned}$$

Moreover, we define the vector

$$\begin{aligned} \mathbf{k}_{\texttt{dual}}(\mathbf{x}) &= [k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}_1), k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}_2), \ldots, k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}_N)]^\top : \mathcal{X} \to \mathbb{R}^{N \times 1} \\ &= \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top, \end{aligned} \tag{71}$$

where we have considered the training inputs $\{\mathbf{x}_n\}_{n=1}^N$, and finally we introduce the $N \times N$ matrix

$$\begin{aligned} \mathbf{K}_{\texttt{dual}} &= [\mathbf{k}_{\texttt{dual}}(\mathbf{x}_1), \mathbf{k}_{\texttt{dual}}(\mathbf{x}_2), \ldots, \mathbf{k}_{\texttt{dual}}(\mathbf{x}_N)]^\top, \\ &= \boldsymbol{\Psi} \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top. \end{aligned} \tag{72}$$

Therefore, the probabilistic model of RVM *indirectly* assumes *correlation* between two inputs $\mathbf{x}$ and $\mathbf{z}$,

$$\text{corr}(f(\mathbf{x}), f(\mathbf{z})) = \frac{\boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{z})}{\sqrt{\boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{x})} \sqrt{\boldsymbol{\psi}(\mathbf{z})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{z})}}. \tag{73}$$

where $|\text{corr}(f(\mathbf{x}), f(\mathbf{z}))| \leq 1$ (i.e., it is a normalized covariance).

## 6.2 GP formulation of RVM - Dual Gaussian Process

Using the results above, the RVM formulas in Eqs. (31)–(32) can be rewritten in some way such that they coincide with the corresponding GP solutions (dual GP formulation), when $k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}')$ in Eq. (70) is used as a kernel function (see Section 5). If we replace the expressions (70)–(71)–(72) within the predictive-posterior RVM distribution

$$p(f(\mathbf{x})|\mathbf{y}) = \mathcal{N}(f(\mathbf{x})|\mu_{f|y}, \sigma^2_{f|y}),$$

i.e., in Eqs. (31)–(32), we obtain the following expressions for the corresponding mean and variance functions,

$$\begin{aligned}\mu_{f|y} &= \widehat{f}(\mathbf{x}) = \mathbf{k}_{\texttt{dual}}(\mathbf{x}) \left(\mathbf{K}_{\texttt{dual}} + \sigma_e^2 \mathbf{I}_N\right)^{-1} \mathbf{y}, \\ \sigma^2_{f|y} &= k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}) - \mathbf{k}_{\texttt{dual}}(\mathbf{x}) \left(\mathbf{K}_{\texttt{dual}} + \sigma_e^2 \mathbf{I}_N\right)^{-1} \mathbf{k}_{\texttt{dual}}(\mathbf{x})^\top.\end{aligned} \tag{74}$$

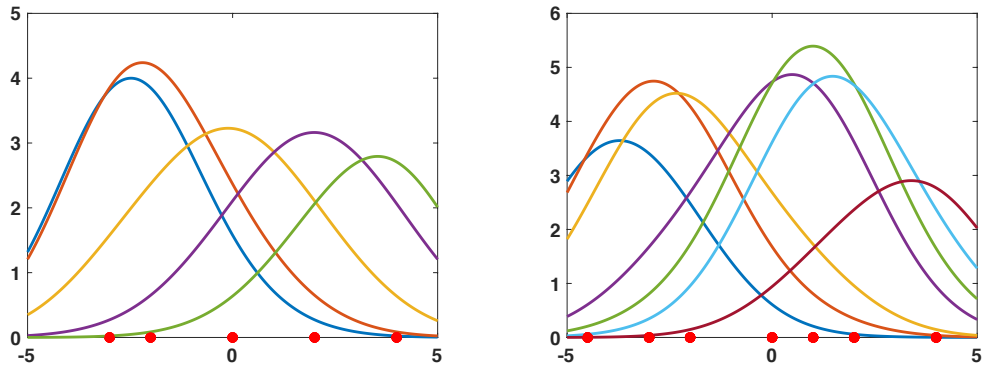
We can observe that these formulas coincide with the mathematical form the GP solutions in Eqs. (62)–(63). Replacing the expressions (70)-(71)-(72) within the smoothing solution, $p(\mathbf{f}|\mathbf{y}) = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}_{f|y}, \boldsymbol{\Sigma}_{f|y})$, i.e., in Eqs. (35)–(36), we obtain

$$\begin{aligned}\boldsymbol{\mu}_{f|y} &= \widehat{\mathbf{f}} = \mathbf{K}_{\texttt{dual}} \left(\mathbf{K}_{\texttt{dual}} + \sigma_e^2 \mathbf{I}_N\right)^{-1} \mathbf{y}, \\ \boldsymbol{\Sigma}_{f|y} &= \mathbf{K}_{\texttt{dual}} - \mathbf{K}_{\texttt{dual}}^\top \left(\mathbf{K}_{\texttt{dual}} + \sigma_e^2 \mathbf{I}_N\right)^{-1} \mathbf{K}_{\texttt{dual}}.\end{aligned} \tag{75}$$

Again, the formulas above coincide with the mathematical form of the GP solutions in Eqs. (67)–(68). Then, a RVM can be interpreted as a GP using $k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}')$ in Eq. (70) as kernel function. However, this implicit choice of $k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}')$ in general does not provide a good behavior of the predictive variance, as we remark in Section 7.

**Remark 17.** *Generally, the dual kernel function $k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}')$ is not stationary. The choice of the bases functions $\psi(\mathbf{x}, \mathbf{x}')$ such that $k_{\texttt{dual}}(\mathbf{x}, \mathbf{x}')$ be stationary is not straightforward.*

Some examples of dual kernel functions are provided in Figure 8.




(a) $N = 5$ (b) $N = 7$

Figure 8: Examples of dual kernels $k_{\texttt{dual}}(x, x_n)$ obtained as in Eq. (70), where **(a)** $x_n \in \{-3, -2, 0, 2, 4\}$, **(b)** $x_n \in \{-4.5, -3, -2, 0, 1, 2, 4\}$ (shown with dots), and the bases $\varphi(x, x_n)$ are Gaussian (with mean at $x_n$ and bandwidth $\lambda = 5$). We have considered a diagonal covariance matrix $\boldsymbol{\Sigma}_\rho$ with all the elements in the diagonal equal to 2.25.

# 7 Summary of Relationships among RVMs, Q-GPs, and GPs

In all the considered methods, we have a complete characterization of posterior of the function $f(\mathbf{x})$ for all $\mathbf{x}$. Moreover, in all the methods, we have shown the generation of random functions from (direct or induced) priors and/or posteriors. In this section, we recall the connections among all the methods. Below, we enumerate some important observations highlighted so far:

- Q-GP is a special case of RVM setting $\boldsymbol{\Sigma}_p = \boldsymbol{\Psi}^{-1}$.
- The posterior mean of Q-GP coincides with the posterior mean of a standard GP.
- In the smoothing scenario, i.e., only considering only the data inputs $\{\mathbf{x}_n\}_{n=1}^N$, Q-GP and GP are perfectly equivalent, i.e., they have the same posterior/predictive distributions (Gaussian with the same mean and variance).
- In Q-GP and GP, the design matrix $\boldsymbol{\Psi}$ must be symmetric, i.e., $\boldsymbol{\Psi} = \boldsymbol{\Psi}^\top$, and positive definite. This is not required in RVM.

These relationships among the methods are graphically represented in Figure 9. Summaries of the main formulas for regression, smoothing and interpolation are given also in Tables 3, 4 and 5, respectively.

Table 3: Regression formulas. N.B. In Q-GP and GP, we have $\boldsymbol{\Psi} = \boldsymbol{\Psi}^\top$.

| **Method** | **Mean** $\widehat{f}(\mathbf{x})$ | **Variance** $\sigma^2_{f\|y}(\mathbf{x})$ |
|---|---|---|
| RVM | $\boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top \left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2 \mathbf{I}_N\right)^{-1} \mathbf{y}$ | $\boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{x}) -$ $\boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top \left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2 \mathbf{I}_N\right)^{-1} \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho \boldsymbol{\psi}(\mathbf{x})$ |
| Q-GP | $\boldsymbol{\psi}(\mathbf{x})^\top (\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \mathbf{y}$ | $\boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Psi}^{-1} \boldsymbol{\psi}(\mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top \left(\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N\right)^{-1} \boldsymbol{\psi}(\mathbf{x})$ |
| GP | $\boldsymbol{\psi}(\mathbf{x})^\top (\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \mathbf{y}$ | $\psi(\mathbf{x}, \mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top (\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \boldsymbol{\psi}(\mathbf{x})$ |

## 7.1 Advantages and weaknesses

The advantage of RVM is that we have fewer restrictions in choosing the bases $\psi_n$, i.e., we have more flexibility in this choice. With Q-GP and GP, we need some function $\psi$ such that the matrix $\boldsymbol{\Psi}$ be invertible and positive definite, since $\boldsymbol{\Psi}$ must be interpreted as a covariance matrix. For instance, in RVM, we can employ directly $N$ different bases $\psi_n$, each one with a different analytical form and different parameters. In this scenario, with Q-GP and GP, first of all we should be check it the resulting matrix $\boldsymbol{\Psi}$ is positive definite, for any possible values of the entries and the parameters.

Table 4: Smoothing formulas. N.B. In Q-GP and GP, we have $\boldsymbol{\Psi} = \boldsymbol{\Psi}^\top$.

| **Method** | **Mean vector** $\widehat{\mathbf{f}} = \boldsymbol{\mu}_{f\vert y}$ | **Covariance matrix** $\boldsymbol{\Sigma}_{f\vert y}$ |
|---|---|---|
| RVM | $\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top \left(\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2\mathbf{I}_N\right)^{-1}\mathbf{y}$ | $\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top - \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top \left(\sigma_e^2\mathbf{I}_N + \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top\right)^{-1}\boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top$ |
| Q-GP | $\boldsymbol{\Psi}(\boldsymbol{\Psi} + \sigma_e^2\mathbf{I}_N)^{-1}\mathbf{y}$ | $\boldsymbol{\Psi} - \boldsymbol{\Psi}\left(\boldsymbol{\Psi} + \sigma_e^2\mathbf{I}_N\right)^{-1}\boldsymbol{\Psi}$ |
| GP | $\boldsymbol{\Psi}(\boldsymbol{\Psi} + \sigma_e^2\mathbf{I}_N)^{-1}\mathbf{y}$ | $\boldsymbol{\Psi} - \boldsymbol{\Psi}\left(\boldsymbol{\Psi} + \sigma_e^2\mathbf{I}_N\right)^{-1}\boldsymbol{\Psi}$ |

Table 5: Interpolation formulas. N.B. In Q-GP and GP, we have $\boldsymbol{\Psi} = \boldsymbol{\Psi}^\top$.

| **Method** | **Mean** $\widehat{f}(\mathbf{x})$ | **Variance** $\sigma^2_{f\vert y}(\mathbf{x})$ |
|---|---|---|
| RVM | $\boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\Psi}^{-1}\mathbf{y}$ | 0 for all $\mathbf{x}$ |
| Q-GP | $\boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\Psi}^{-1}\mathbf{y}$ | 0 for all $\mathbf{x}$ |
| GP | $\boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\Psi}^{-1}\mathbf{y}$ | $\psi(\mathbf{x},\mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top\boldsymbol{\Psi}^{-1}\boldsymbol{\psi}(\mathbf{x})$ |

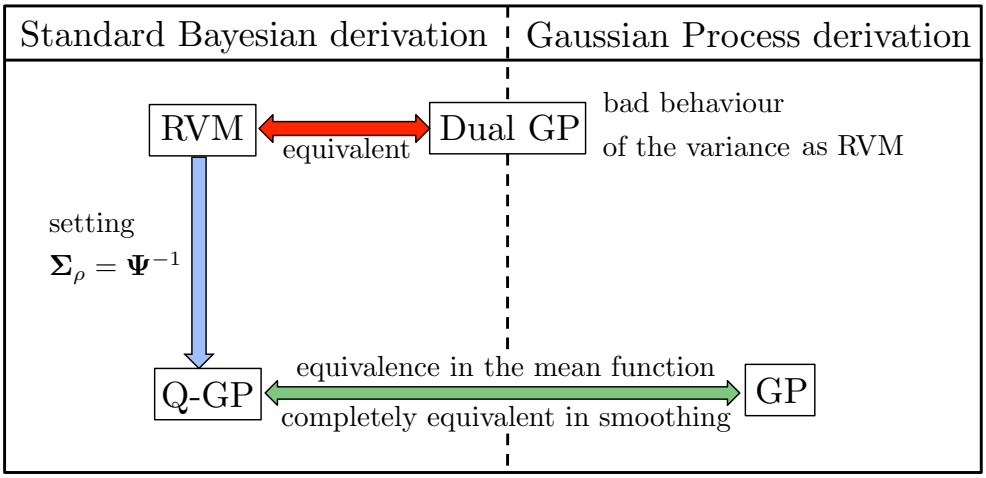



Figure 9: Graphical representation of the relationships among the different methods.

In all cases, RVM, Q-GP and GP, the mean solution can be expressed as linear combination of $N$ nonlinearities. Hence, in both cases, the flexibility of the solution grows with the number of data (i.e., $N$).

**Remark 18.** *The main benefit of the RVM approach is that we have fewer restrictions in the choice of the nonlinearities* $\phi_n$ *that can be also different for each data input* $\mathbf{x}_n$*, since we do not need that* $\boldsymbol{\Psi}$ *be symmetric.*

**Remark 19.** *The main advantage of GP with respect to the other methods, is that we directly decide the covariance function ensuring, for instance, stationary and other statistical properties (when required). This has also another important consequence: the behavior of GP predictive variance has a natural/intuitive behavior (as we shown below), unlike the predictive variances of RVM and Q-GP [16, 18, 15, 17].*

## 7.2 Variance behavior

A intuitive and natural behavior of the predictive variance is the following: it should be smaller at $\mathbf{x}$ close to the data inputs $\mathbf{x}_n$, and greater far away from the data points. This usually happens with a GP with a reasonable choice of the kernel function. With RVM (and Q-GP) and localized bases, the behavior is arguably non-intuitive: the predictive variance is greater close to $\mathbf{x}_n$, and smaller far away from the data inputs.

**Comparison of Q-GP and GP variances.** We know that Q-GP is a special case of RVM, which coincides with GP in the mean of te posterior/predictive function. In order to provide a comparison between Q-GP and GP, consider a one dimensional example, $x \in \mathbb{R}$, with the following basis/kernel

$$k(x,z) = a \exp\left(-\frac{(x-z)^2}{\lambda}\right), \tag{76}$$

with $a = 0.7$ and $\lambda = 2$. Given a set of data points, we have applied the Q-GP and GP methods, considering $\sigma_e = 0.5$ (in Figure 10). In Figure 10(a), we show the data points and the posterior means of Q-GP and GP. As expected, they perfectly coincide in both cases. Figure 10(b) depicts the mean of Q-GP $\widehat{f}(x)$ and $\widehat{f}(x) \pm 2\sqrt{\sigma^2_{f|y}(x)}$ with a shaded area. Figure 10(c) depicts the mean $\widehat{f}(x)$ of the standard GP and $\widehat{f}(x) \pm 2\sqrt{\sigma^2_{f|y}(x)}$ with a shaded area. The corresponding variances $\sigma^2_{f|y}(x)$ as function of $x$ are shown in Figure 10(d). We can observe that the the variance of the GP is smaller closer to the data points (as expected). The opposite occurs with Q-GP. The two variance functions coincide exactly in the data points $\{x_n\}_{n=1}^N$. Indeed, in the smoothing scenario, Q-GP and GP are perfectly equivalent. Furthermore, observe that the variance of GP is always greater than the variance of Q-GP. Finally, let us compare Figures 10(d) and 11, for instance. Note that, when $\sigma_e \to 0$, the variance of Q-GP vanishes to zero for all $x$, i.e., $\sigma^2_{f|y}(x) \to 0 \ \forall x$. Whereas, when $\sigma_e \to 0$, the variance of GP goes to zero only in $\{x_n\}_{n=1}^N$, i.e., in the data points $\sigma^2_{f|y}(x_n) = 0$ for all $n$. The interpolation case is given in Figure 11 (see also Table 5).

## 7.3 The legend of infinite bases

Several authors show that the squared exponential kernel function, defined (scalar $x \in \mathbb{R}$ for simplicity) as

$$\psi(x_i, x_j) = \exp\left(-\frac{(x_i - x_j)^2}{2\sqrt{2}\lambda}\right),$$

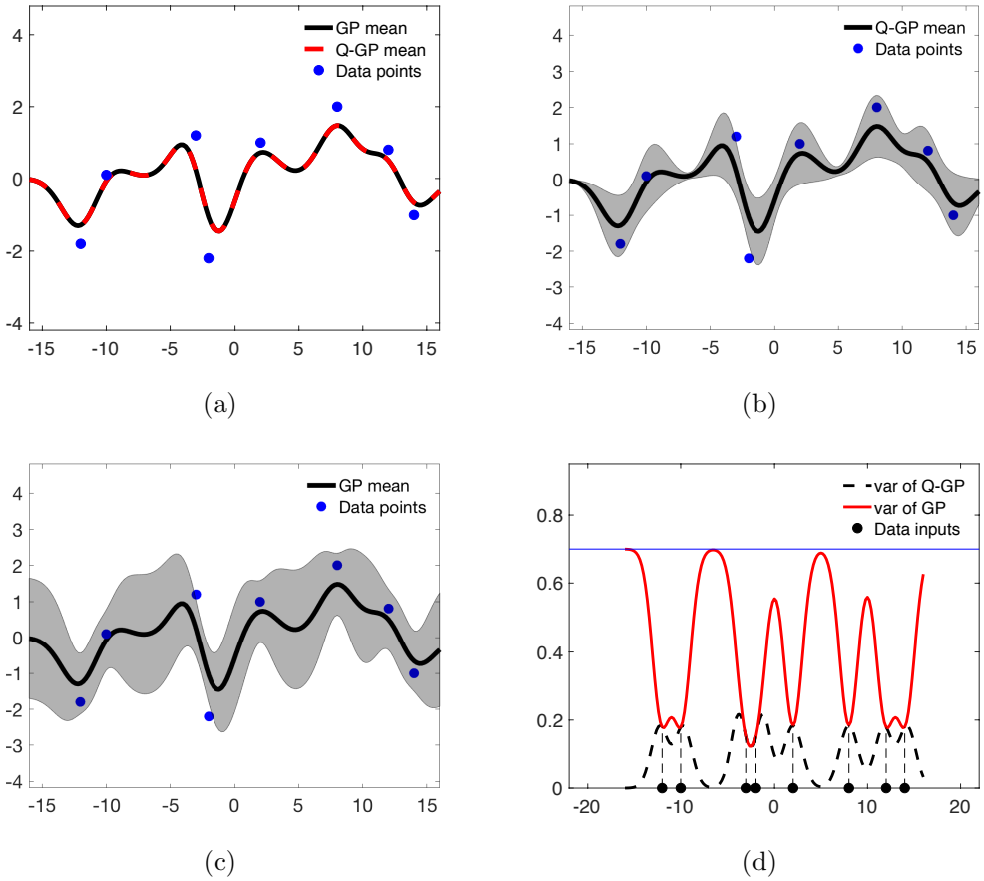
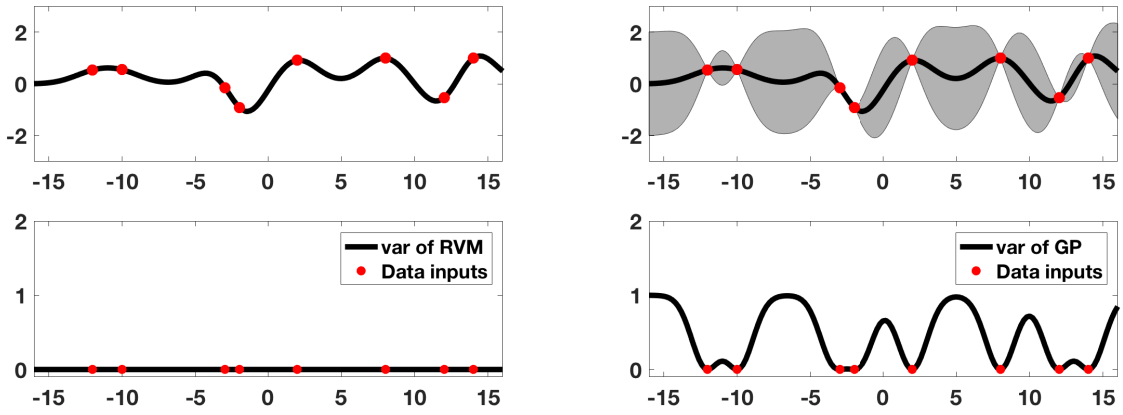



Figure 10: Variance Behaviour (with $\sigma_e = 0.5$). **(a)-(b)-(c)** show the posterior (predictive) mean $\widehat{f}(x)$ of GP and Q-GP. The shaded areas represents $\widehat{f}(x) \pm 2\sqrt{\sigma^2_{f|y}(x)}$. Note the posterior means coincides. **(d)** shows the corresponding variances $\sigma^2_{f|y}(x)$ as function of $x$. The horizontal blue line denotes the value of $a = 0.7$.

can also be obtained by expanding the input into a feature space represented by an *infinite* network defined by Gaussian-shaped basis functions $\phi_c(x) = \exp\left(-\frac{(x-c)^2}{2\lambda}\right)$, where $c$ denotes the centre of the basis function. Let us consider $M$ bases centered in different values $c$. Assuming the covariance matrix of the prior density as $\mathbf{\Sigma}_p = \sigma_p^2 \mathbf{I}_M$, the induced kernel can be written as

$$k_M(x_i, x_j) = \sigma_p^2 \sum_{c=1}^{M} \phi_c(x_i)\phi_c(x_j).$$

It is possible to show that $\lim_{M\to\infty} k_M(x_i, x_j) \propto \psi(x_i, x_j)$ (see, e.g., [6]).
This result can lead to misleading conclusions. For instance, one could state that *" to obtain the GP flexibility and performance, with a standard Bayesian formulation, we need infinite bases"*. This statement is not true, or only partially true. Indeed, regarding the posterior/predictive mean function, we know that we can obtain exactly the GP solution with a standard Bayesian




(a) RVM and Q-GP

(b) GP

Figure 11: Interpolation case, i.e., $\sigma_e = 0$ (and $a = 1$). **(a)** In this case, RVM and Q-GP provides the same solution with a null predictive variance, i.e., $\sigma^2_{f|y}(x) = 0$. The posterior/predictive mean $\widehat{f}(x)$ of RVM/Q-GP is also given. **(b)** Predictive mean and variance of a GP when $\sigma_e = 0$ (interpolation). The shaded area represents $\widehat{f}(x) \pm 2\sqrt{\sigma^2_{f|y}(x)}$. The corresponding variances $\sigma^2_{f|y}(x)$ as function of $x$ is also given below. Note that $\sigma^2_{f|y}(x)$ is zero only at the data inputs.

formulation setting $M = N$ (finite) and using a *suitable* covariance prior over the weights, $\boldsymbol{\Sigma}_p = \boldsymbol{\Psi}^{-1}$. On the other hand, we cannot obtain the same posterior/predictive variance, at least not with a finite number of bases.

## 7.4 Marginal likelihood and parameter learning

The marginal likelihood (a.k.a., Bayesian evidence) is defined as

$$p(\mathbf{y}) = \int_{\mathbb{R}^N} p(\mathbf{y}|\mathbf{f})p(\mathbf{f})\mathbf{df}. \tag{77}$$

where $p(\mathbf{f})$ is the prior over the $N \times 1$ vector $\mathbf{f}$ given the considered probabilistic model. The marginal likelihood represents the probability of data given the model $\mathcal{M}$ and its parameters, $p(\mathbf{y}) = p(\mathbf{y}|\mathcal{M})$, hence it is useful for model selection purposes. For instance, it can be used in order to tune of the parameters and hyper-parameters of the model denoted as $\boldsymbol{\theta}$. In the vector $\boldsymbol{\theta} = [\boldsymbol{\lambda}, \sigma^2_e]$ we include all the parameters $\boldsymbol{\lambda}$ of the nonlinearities $\psi_n$, as well as the parameters need for defining the prior densities and the power of the noise perturbation $\sigma^2_e$. Therefore, in this case, a more complete notation is the following

$$p(\mathbf{y}|\boldsymbol{\theta}) = \int_{\mathbb{R}^N} p(\mathbf{y}|\mathbf{f}, \boldsymbol{\theta})p(\mathbf{f}|\boldsymbol{\theta})\mathbf{df}. \tag{78}$$

The marginal likelihoods of different methods studied so far can be computed analytically, and are given below

$$\begin{aligned} \text{RVM:} \quad p(\mathbf{y}|\boldsymbol{\theta}) &= \mathcal{N}(\mathbf{y}|\mathbf{0}, \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2\mathbf{I}_N), \\ \text{Q-GP, GP:} \quad p(\mathbf{y}|\boldsymbol{\theta}) &= \mathcal{N}(\mathbf{y}|\mathbf{0}, \boldsymbol{\Psi} + \sigma_e^2\mathbf{I}_N). \end{aligned}$$

**Remark 20.** *Since Q-GP and GP have the same marginal likelihood, then they have the same estimator $\widehat{\boldsymbol{\theta}}$ of the hyper-parameters (e.g., maximum likelihood, MAP, MMSE etc.).*

Note that, in all cases, we have a multivariate Gaussian density

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\mathbf{0}, \mathbf{C}_{yy}),$$

with $\mathbf{C}_{yy} = \boldsymbol{\Psi}\boldsymbol{\Sigma}_\rho\boldsymbol{\Psi}^\top + \sigma_e^2\mathbf{I}_N$ in RVM, and $\mathbf{C}_{yy} = \boldsymbol{\Psi} + \sigma_e^2\mathbf{I}_N$ in Q-GP and GP. Then, we can write the full negative log-marginal likelihood as

$$-\log p(\mathbf{y}|\boldsymbol{\theta}) = \frac{1}{2}\mathbf{y}^\top\mathbf{C}_{yy}^{-1}\mathbf{y} + \frac{1}{2}\log\left[\det\mathbf{C}_{yy}\right] + const. \tag{79}$$

The first term $\frac{1}{2}\mathbf{y}^\top\mathbf{C}_{yy}^{-1}\mathbf{y}$ in Eq. (79) can be considered a fitting term, the second one $\frac{1}{2}\log\left[\det\mathbf{C}_{yy}\right]$ plays the role of a regularizer, i.e., a penalty on the model complexity.
We can maximize $p(\mathbf{y}|\boldsymbol{\theta})$ obtaining a possible choice $\widehat{\boldsymbol{\theta}}$ (i.e., a possible estimator). This approach is also called *type-II maximum likelihood procedure* (a.k.a., *empirical Bayes*). Note that, in this way, we avoid the use of a cross-validation (CV) procedure. Alternatively, one can also consider a prior $p(\boldsymbol{\theta})$ over $\boldsymbol{\theta}$ and study the posterior $p(\boldsymbol{\theta}|\mathbf{y}) \propto p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})$ using for instance Monte Carlo methods [27, 28]. Different possible point estimators $\widehat{\boldsymbol{\theta}}$ are possible such as the maximum, the expected value or the median of the posterior $p(\boldsymbol{\theta}|\mathbf{y})$. Additionally, studying the posterior $p(\boldsymbol{\theta}|\mathbf{y})$, we can obtain credible intervals of each parameter and approximate its marginal distribution, for instance. An additional alternative is the use of a full Bayesian approach, described in the next section.

# 8 Uncertainty analysis with GPs

Although the predictive GP variance has a good intuitive behavior, its analytical form depends *explicitly* just on the inputs $\{\mathbf{x}_n\}_{n=1}^N$ (not on the outputs $\{y_n\}_{n=1}^N$), i.e., $\sigma_{f|y}^2(\mathbf{x}) = \psi(\mathbf{x},\mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top(\boldsymbol{\Psi} + \sigma_e^2\mathbf{I}_N)^{-1}\boldsymbol{\psi}(\mathbf{x})$.
Namely, fixing the hyper-parameters of the employed kernel, we could compute $\sigma_{f|y}^2(\mathbf{x})$ only knowing $\{\mathbf{x}_n\}_{n=1}^N$ and without any information of the signal values $\{y_n\}_{n=1}^N$. In this case, since we can compute $\sigma_{f|y}^2(\mathbf{x})$ before knowing the signal, it seems that $\sigma_{f|y}^2(\mathbf{x})$ can not provide relevant information. However, we will learn the hyper-parameters given the data $\mathbf{y} = [y_1, \ldots, y_N]^\top$ and the choice of the hyper-parameters affects (a) the value $\psi(\mathbf{x},\mathbf{x})$ (if we have a multiplicative parameter $a$ as in Eq. (69)), (b) the vector $\boldsymbol{\psi}(\mathbf{x})$ and (c) the matrix $\boldsymbol{\Psi}$. Hence, we can assert that the variance $\sigma_{f|y}^2(\mathbf{x})$ depends on $\{y_n\}_{n=1}^N$ through the hyper-parameters learning. More information can be obtained by performing a full Bayesian study.

**Full Bayesian solution.** A full Bayesian solution can provide more information for a proper uncertainty analysis, as we show below. Let us assume also a prior $p(\boldsymbol{\theta})$ over the hyper-parameters $\boldsymbol{\theta}$. A full Bayesian analysis considers the complete joint posterior, which can be expressed as

$$p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{f}, \boldsymbol{\theta}, \mathbf{y})}{p(\mathbf{y})} = \frac{p(\mathbf{y}|\mathbf{f}, \boldsymbol{\theta})p(\mathbf{f}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})}. \tag{80}$$

This is a more complete and proper approach from a Bayesian point of view. However, moments and other features of $p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{y})$ are not analytically available, so that the application of computational algorithms (such as Monte Carlo methods) is required [27, 28]. Indeed, so far we have considered a *conditional posterior*, i.e.,

$$p(\mathbf{f}|\mathbf{y}, \boldsymbol{\theta}) = \frac{p(\mathbf{f}, \boldsymbol{\theta}, \mathbf{y})}{p(\boldsymbol{\theta}, \mathbf{y})} = \frac{p(\mathbf{y}|\mathbf{f}, \boldsymbol{\theta})p(\mathbf{f}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})} = \frac{p(\mathbf{y}|\mathbf{f}, \boldsymbol{\theta})p(\mathbf{f}|\boldsymbol{\theta})}{p(\mathbf{y}|\boldsymbol{\theta})}, \tag{81}$$

where the conditional marginal likelihood is $p(\mathbf{y}|\boldsymbol{\theta}) = \int_{\mathbb{R}^N} p(\mathbf{y}|\mathbf{f}, \boldsymbol{\theta})p(\mathbf{f}|\boldsymbol{\theta})d\mathbf{f}$. One *marginal posterior* of $\boldsymbol{\theta}$ is given as

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})} \propto p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}), \tag{82}$$

where

$$p(\mathbf{y}) = \int_{\boldsymbol{\Theta}} p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}.$$

which can be useful for comparing GP models using different kernels, for instance. Note that the relationship among the full posterior in Eq. (80), the conditional posterior in Eq. (81), and the marginal posterior in Eq. (82), is give by

$$p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{y}) = p(\mathbf{f}|\mathbf{y}, \boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y}). \tag{83}$$

Furthermore, the other *marginal posterior* is

$$p(\mathbf{f}|\mathbf{y}) = \int_{\boldsymbol{\Theta}} p(\mathbf{f}|\mathbf{y}, \boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y})d\boldsymbol{\theta}. \tag{84}$$

In order to study the posterior $p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{y})$, we can use a Monte Carlo approximation. We can generate $N$ samples from the complete posterior $\{\mathbf{f}_s, \boldsymbol{\theta}_s\} \sim p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{y})$, where $\boldsymbol{\theta}_s \sim p(\boldsymbol{\theta}|\mathbf{y})$ and $\mathbf{f}_s \sim p(\mathbf{f}|\mathbf{y}, \boldsymbol{\theta}_s)$, with $s = 1, ..., S$. The difficult task is to draw from the marginal posterior $p(\boldsymbol{\theta}|\mathbf{y})$, whereas the conditional posterior $p(\mathbf{f}|\mathbf{y}, \boldsymbol{\theta}_s)$ is a Gaussian pdf with known mean and covariance matrix (for any possible value of $\boldsymbol{\theta}_s$). Thus, the Monte Carlo approximation of marginal posterior $p(\mathbf{f}|\mathbf{y})$ in Eq. (84), is a mixture of Gaussians,

$$p(\mathbf{f}|\mathbf{y}) \approx \frac{1}{S}\sum_{s=1}^{S} p(\mathbf{f}|\mathbf{y}, \boldsymbol{\theta}_s), \qquad \boldsymbol{\theta}_s \sim p(\boldsymbol{\theta}|\mathbf{y}). \tag{85}$$

**Dependence on the choice of the hyperparameters.** Here, we discuss a more complete study related to the uncertainty analysis of the solutions. Let us draw $S$ samples $\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_S$ from the marginal posterior $p(\boldsymbol{\theta}|\mathbf{y})$. Since the conditional posterior mean $\mu_{f|y}(\mathbf{x}|\boldsymbol{\theta}_s) = \widehat{f}(\mathbf{x}|\boldsymbol{\theta}_s)$ and variance $\sigma^2_{f|y}(\mathbf{x}|\boldsymbol{\theta}_s)$ depend on the hyper-parameters $\boldsymbol{\theta}_s$, we also have $S$ mean and variance values (for each $\mathbf{x}$), i.e., $\widehat{f}(\mathbf{x}|\boldsymbol{\theta}_1)$,....,$\widehat{f}(\mathbf{x}|\boldsymbol{\theta}_S)$ and $\sigma^2_{f|y}(\mathbf{x}|\boldsymbol{\theta}_1)$,....,$\sigma^2_{f|y}(\mathbf{x}|\boldsymbol{\theta}_S)$. Then we can calculate the approximate averaged solution as

$$\begin{aligned}
\bar{f}(\mathbf{x}) &= \frac{1}{S}\sum_{s=1}^{S} \widehat{f}(\mathbf{x}|\boldsymbol{\theta}_s) \approx \int_{\boldsymbol{\Theta}} \widehat{f}(\mathbf{x}|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{y}) d\boldsymbol{\theta} \\
\bar{\sigma}^2(\mathbf{x}) &= \frac{1}{S}\sum_{s=1}^{S} \sigma^2_{f|y}(\mathbf{x}|\boldsymbol{\theta}_s) \approx E_p\left[\sigma^2_{f|y}(\mathbf{x}|\boldsymbol{\theta})\right] = \int_{\boldsymbol{\Theta}} \sigma^2_{f|y}(\mathbf{x}|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{y}) d\boldsymbol{\theta}.
\end{aligned}$$

Note that $\bar{f}(\mathbf{x})$ is an approximation the expected value associated to marginal posterior $p(f(\mathbf{x})|\mathbf{y})$. Thus, we can also compute

$$V_f(\mathbf{x}) = \frac{1}{S}\sum_{s=1}^{S} \left(\widehat{f}(\mathbf{x}|\boldsymbol{\theta}_s) - \bar{f}(\mathbf{x})\right)^2 \approx \mathrm{Var}\left[\widehat{f}(\mathbf{x}|\boldsymbol{\theta})\right].$$

For the law of total variance, the variance associated to $p(f(\mathbf{x})|\mathbf{y})$ is

$$\begin{aligned}
\mathrm{Var}[\widehat{f}(\mathbf{x})] &= \mathrm{Var}\left[E_p[f(\mathbf{x}|\boldsymbol{\theta})]\right] + E_p\left[\mathrm{Var}[f(\mathbf{x}|\boldsymbol{\theta})]\right], &(86)\\
&= \mathrm{Var}\left[\widehat{f}(\mathbf{x}|\boldsymbol{\theta})\right] + E_p\left[\sigma^2_{f|y}(\mathbf{x}|\boldsymbol{\theta})\right], &(87)\\
&\approx V_f(\mathbf{x}) + \bar{\sigma}^2(\mathbf{x}). &(88)
\end{aligned}$$

Therefore, the complete variance is the sum of the two terms $\bar{\sigma}^2(\mathbf{x})$ and $V_f(\mathbf{x})$. Moreover, it is interesting to analyze the term $V_f(\mathbf{x})$ which provides the variation of the mean solution depending on the choice of the hyper-parameters $\boldsymbol{\theta}$ (i.e., a sensitivity analysis).

# 9 Linear kernel smoothers

RVMs and GPs belong to a more general class of regressors: the linear kernel smoothers. In this family of regression methods the prediction $\widehat{f}(\mathbf{x})$ at some input $\mathbf{x}$ is expressed as linear combination of the outputs $y_1, \ldots, y_N$. The weights of this combination vary with $\mathbf{x}$. We can interpret that is another way of *implicitly* modeling the correlation among the different outputs. Generally, the linear smoothers have not associated a probabilistic derivation (unlike RVMs and GPs), so that we focus on the approximation $\widehat{f}(\mathbf{x})$.

## 9.1 Definition and examples

A linear smoother is a regressor which combines linearly the observations $y_1, \ldots, y_N$ at each $\mathbf{x}$, i.e.,

$$\widehat{f}(\mathbf{x}) = \sum_{n=1}^{N} \varphi_n(\mathbf{x}, \mathbf{x}_n) y_n = \boldsymbol{\varphi}(\mathbf{x})^\top \mathbf{y}, \tag{89}$$

where $\boldsymbol{\varphi}(\mathbf{x}) = [\varphi_1(\mathbf{x}, \mathbf{x}_1), \ldots, \varphi_N(\mathbf{x}, \mathbf{x}_N)]^\top$ and $\varphi_n(\mathbf{x}, \mathbf{x}_n) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ plays the role as a weight. When the output of a regression technique can be expressed as in Eq. (89) then it is also called linear kernel smoother. Equation (89) shows that the estimator at any input $\mathbf{x}$, i.e., $\widehat{f}(\mathbf{x})$, can be expressed as linear combination of the $N$ outputs $y_1, \ldots, y_N$. We can observe that the coefficients of the linear combination $\varphi_n(\mathbf{x}, \mathbf{x}_n)$, with $n = 1, \ldots, N$, depend on $\mathbf{x}$.

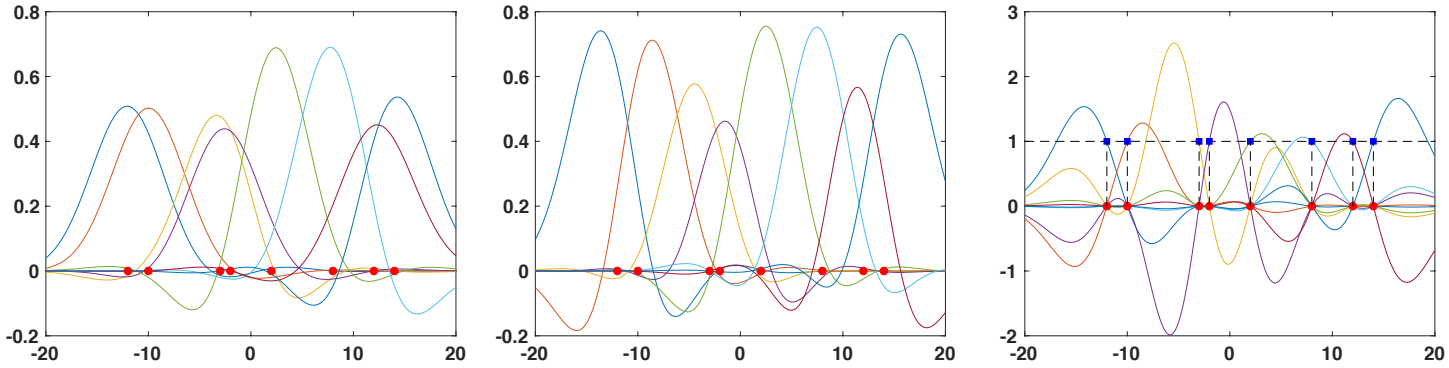
### 9.1.1 The case of RVM and GP

The RVM and Q-GP, GP are linear kernel smoothers since the mean function of the posterior can be expressed as in Eq. (89), setting

$$\text{RVM:} \quad \boldsymbol{\varphi}(\mathbf{x})^\top = \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top \left( \boldsymbol{\Psi} \boldsymbol{\Sigma}_\rho \boldsymbol{\Psi}^\top + \sigma_e^2 \mathbf{I}_N \right)^{-1}, \tag{90}$$

$$\text{Q-GP,GP:} \quad \boldsymbol{\varphi}(\mathbf{x})^\top = \boldsymbol{\psi}(\mathbf{x})^\top \left( \boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N \right)^{-1}. \tag{91}$$

Namely, the weighting functions $\varphi_n(\mathbf{x}, \mathbf{x}_n)$ depend also on the nonlinearities $\psi_n(\mathbf{x}, \mathbf{x}_n)$ as shown above. Figure 12 shows some examples of the weighting functions $\varphi_n(\mathbf{x}, \mathbf{x}_n)$ in RVM and GP cases.



(a) RVM ($\sigma_e = 0.5$) (b) GP ($\sigma_e = 0.5$) (c) RVM, GP ($\sigma_e = 0$)

Figure 12: An example of weighting functions $\varphi_n(x, x_n)$; **(a)** in a RVM and **(b)** in a GP. We have considered $N = 8$ data inputs $x_n$ (shown with dots), $\psi(\mathbf{x}, \mathbf{x}_n) = \exp\left(-\frac{(x-x_n)^2}{\lambda}\right)$ with $\lambda = 25$, and $\sigma_e = 0.5$. For RVM, we have also considered a diagonal covariance matrix $\boldsymbol{\Sigma}_\rho$ with all the elements in the diagonal equal to 1. **(c)** Weighting functions $\varphi_n(x, x_n)$ of RVM and GP for the interpolation case, $\sigma_e = 0$. In this scenario, at each data input $x_n$, all $\varphi_n(x, x_n)$ are zero except the $n$-th function where $\varphi_n(x_n, x_n) = 1$, i.e., $\varphi_n(x_j, x_n) = \delta_{jn}$.

### 9.1.2 Normalized weighed functions

Generally, the linear combination above in Eq. (89) is *not* a convex combination. Often, people considers linear smoothers defined as a convex combination where the weight function are positive and the sum is 1. For instance, consider when auxiliary weighting function $h_\lambda(\mathbf{x}, \mathbf{x}_n) \geq 0$ is

used to assign weights to $x_n$ based on its distance from $\mathbf{x}$. The parameter $\lambda \in \mathbb{R}$ indicates the bandwidth (the width of the neighborhood), determined from the training data. One example is *the Nadaraya-Watson estimator* where

$$\widehat{f}(\mathbf{x}) = \sum_{n=1}^{N} \frac{h_\lambda(\mathbf{x}, \mathbf{x}_n)}{\sum_{j=1}^{N} h_\lambda(\mathbf{x}, \mathbf{x}_j)} y_n = \sum_{n=1}^{N} \varphi_n(\mathbf{x}, \mathbf{x}_n) y_n, \tag{92}$$

where $\varphi_n(\mathbf{x}, \mathbf{x}_n) = \frac{h_\lambda(\mathbf{x}, \mathbf{x}_n)}{\sum_{j=1}^{N} h_\lambda(\mathbf{x}, \mathbf{x}_j)}$. Note that, with this definition,

$$\sum_{n=1}^{N} \varphi_n(\mathbf{x}, \mathbf{x}_n) = 1.$$

Figure 13 provides some examples of $\widehat{f}(x)$ (with $x \in \mathbb{R}$) when $h_\lambda(x, z) = \exp\left(-(x-z)^2/\lambda\right)$ and different values of $\lambda$. The form of this estimator above is quite general. For instance, it contains the k-nearest neighbors algorithm (kNN) for regression as a specific case (with a specific choice of $h_\lambda(\mathbf{x}, \mathbf{x}_n)$). See the next sections for further details.

### 9.1.3 Derivation of Nadaraya-Watson estimator

So far we have considered the outputs $y_n$ as random variables (affected by random perturbations), whereas the inputs $\mathbf{x}_n$ are considered as auxiliary deterministic information. Now, let us consider both $\mathbf{x}_n$ and $y_n$ as random variables with joint density $p(\mathbf{x}, y)$. Namely, we assume

$$[\mathbf{x}_n, y_n] \sim p(\mathbf{x}, y) \qquad n = 1, \ldots, N.$$

We can try to estimate $p(\mathbf{x}, y)$ via kernel density estimation,

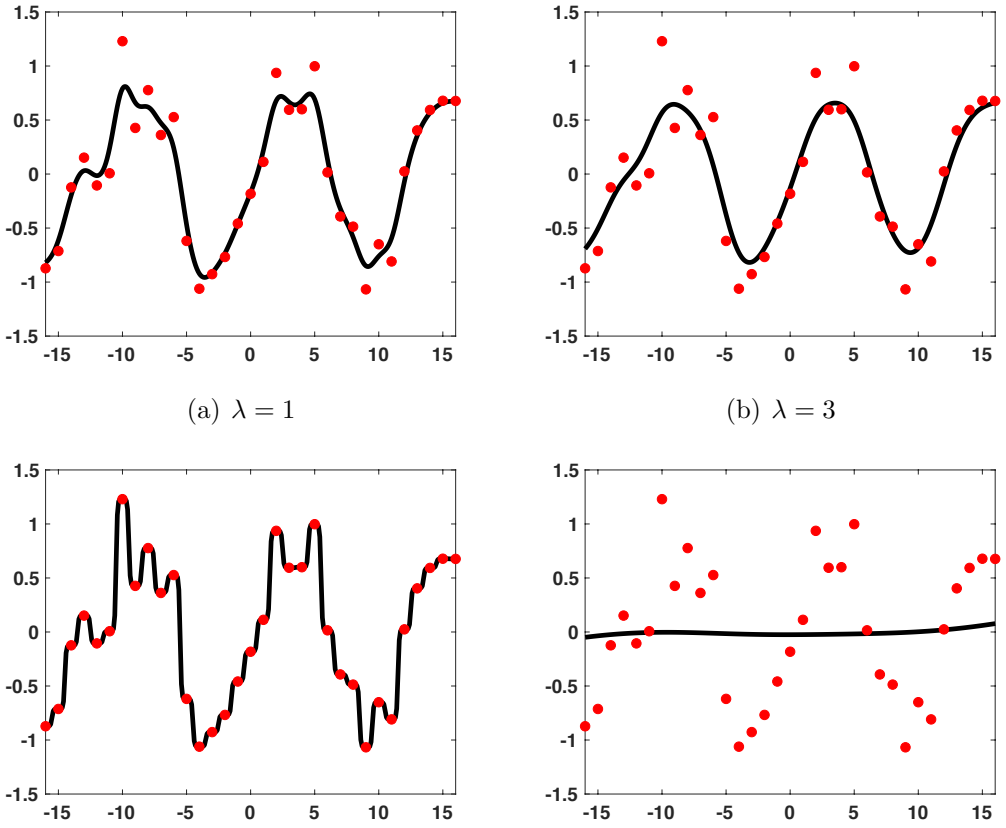
$$\widehat{p}(\mathbf{x}, y) = \frac{1}{N} \sum_{n=1}^{N} h_{\lambda_x}(\mathbf{x} - \mathbf{x}_n) h_{\lambda_y}(y - y_n). \tag{93}$$

where $\int_{\mathcal{X}} h_{\lambda_x}(\mathbf{x}) d\mathbf{x} = 1$ and $\int_{\mathbb{R}} h_{\lambda_y}(y) dy = 1$. Moreover, $\int_{\mathcal{X}} \mathbf{x} h_{\lambda_x}(\mathbf{x}) d\mathbf{x} = \mathbf{0}$ and $\int_{\mathbb{R}} y h_{\lambda_y}(y) dy = 0$. The regression function is defined as

$$\widehat{f}(\mathbf{x}) = E[y|\mathbf{x}] = \int_{\mathbb{R}} y p(y|\mathbf{x}) dy = \frac{\int_{\mathbb{R}} y p(\mathbf{x}, y) dy}{\int_{\mathbb{R}} p(\mathbf{x}, y) dy}. \tag{94}$$

Replacing $p(\mathbf{x}, y)$ with $\widehat{p}(\mathbf{x}, y)$, then we have

$$\begin{aligned} \int_{\mathbb{R}} y \widehat{p}(\mathbf{x}, y) dy &= \frac{1}{N} \int_{\mathbb{R}} y \sum_{n=1}^{N} h_{\lambda_x}(\mathbf{x} - \mathbf{x}_n) h_{\lambda_y}(y - y_n) dy, \\ &= \frac{1}{N} \sum_{n=1}^{N} h_{\lambda_x}(\mathbf{x} - \mathbf{x}_n) y_n, \end{aligned}$$



(a) $\lambda = 1$ (b) $\lambda = 3$

(c) $\lambda = 0.1$ (d) $\lambda = 100$

Figure 13: Examples of $\widehat{f}(x)$ (with $x \in \mathbb{R}$) when $h_\lambda(x,z) = \exp\left(-(x-z)^2/\lambda\right)$ and $\lambda \in \{0.1, 1, 3, 100\}$. The data points are shown with red dots.

where we have used $\int_{\mathbb{R}} y h_{\lambda_y}(y - y_n)dy = y_n$, since $\int_{\mathbb{R}} y h_{\lambda_y}(y)dy = 0$, i.e., the mean is zero by assumption and, hence the mean of $h_{\lambda_y}(y - y_n)$ is $y_n$ (it is just a translation of the pdf). Moreover,

$$\begin{aligned}\int_{\mathbb{R}} \widehat{p}(\mathbf{x}, y)dy &= \int_{\mathbb{R}} \sum_{n=1}^{N} h_{\lambda_x}(\mathbf{x} - \mathbf{x}_n) h_{\lambda_y}(y - y_n)dy, \\ &= \frac{1}{N} \sum_{n=1}^{N} h_{\lambda_x}(\mathbf{x} - \mathbf{x}_n).\end{aligned}$$

Thus, replacing the numerator and denominator of Eq. (94) with the two approximations above, finally we can write

$$\widehat{f}(\mathbf{x}) \approx \sum_{n=1}^{N} \frac{h_{\lambda_x}(\mathbf{x}, \mathbf{x}_n)}{\sum_{j=1}^{N} h_{\lambda_x}(\mathbf{x}, \mathbf{x}_j)} y_n. \tag{95}$$

This is the Nadaraya-Watson estimator with $\varphi_n(\mathbf{x}, \mathbf{x}_n) = \frac{h_{\lambda_x}(\mathbf{x}, \mathbf{x}_n)}{\sum_{j=1}^{N} h_{\lambda_x}(\mathbf{x}, \mathbf{x}_j)}$ [1].

## 9.2 Other examples of linear smoothers

In section, we describe some well-known linear smoothers that are encompassed in Eq. (89).

### 9.2.1 k-Nearest Neighbors (kNN)

In this section, we replace the real parameter $\lambda$ with an integer value $k \in \mathbb{N}^+$. In the kNN technique for regression, given an integer value $1 \leq k \leq N$, we have

$$h_k(\mathbf{x}, \mathbf{x}_n) = 1,$$

if $\mathbf{x}_n$ is one of the $k$ nearest inputs of $\mathbf{x}$ (within the $N$ possible inputs $\mathbf{x}_n$), otherwise

$$h_k(\mathbf{x}, \mathbf{x}_n) = 0,$$

if $\mathbf{x}_n$ does not belong to the set $k$ of nearest inputs of $\mathbf{x}$. Let us consider now the two extreme cases. If $k = 1$, only one function $h_k(\mathbf{x}, \mathbf{x}_{j^*})$ will be equal to 1 (where $\mathbf{x}_{j^*}$ represents to the closest input to $\mathbf{x}$). As a consequence, for all $\mathbf{x}$ such that $\mathbf{x}_{j^*}$ is the closest input then $\widehat{f}(\mathbf{x}) = y_{j^*}$, i.e., we obtain an interpolator. If $k = N$, all functions $h_k(\mathbf{x}, \mathbf{x}_{j^*}) = 1$ and, as a consequence,

$$\widehat{f}(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^{N} y_n, \qquad \forall \mathbf{x} \in \mathcal{X},$$

i.e., we obtain a constant approximation, equal to the arithmetic mean of the outputs $y_n$ [19].

### 9.2.2 Inverse distance weighting

Another example is the so-called *inverse distance weighting* method for multivariate interpolation with the choice

$$h_\lambda(\mathbf{x}, \mathbf{x}_n) = \frac{1}{d_\lambda(\mathbf{x}, \mathbf{x}_n)^p}, \quad \text{if} \quad \mathbf{x} \neq \mathbf{x}_n,$$

where $d_\lambda(\mathbf{x}, \mathbf{x}_n)$ is a distance (metric operator) and $p > 0$ is a positive real value [29]. When $\mathbf{x} = \mathbf{x}_n$, we directly set $\widehat{f}(\mathbf{x}_n) = y_n$ (i.e., we have an interpolator). Note that the weight $h_\lambda(\mathbf{x}, \mathbf{x}_n)$ decreases as $d_\lambda(\mathbf{x}, \mathbf{x}_n)$ grows.

### 9.2.3 Polynomial interpolation with Lagrange bases

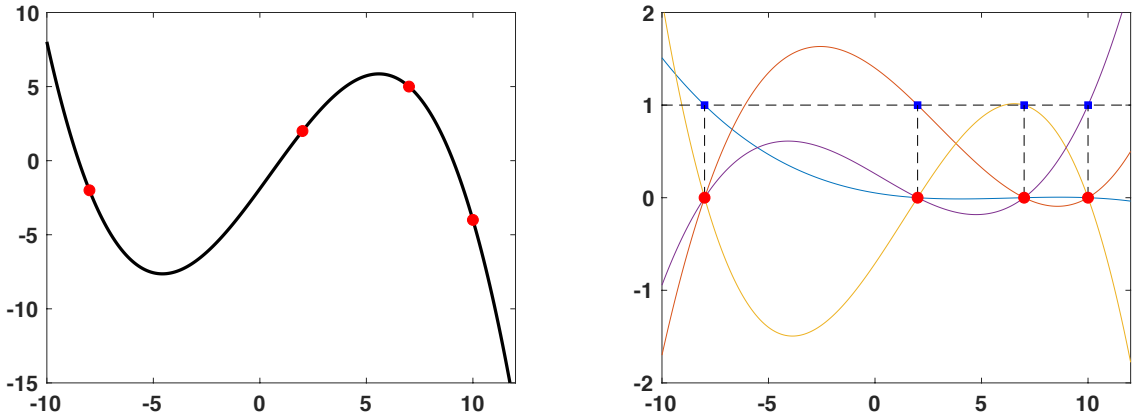
In this section, we focus on the interpolation problem as typycally addressed in the initial courses of numerical analysis [30]. Let us consider for simplicity the scalar input case, $x_i \in \mathbb{R}$. Consider the problem of obtaining the polynomial interpolation of order $N-1$ of $N$ data $\{x_i, y_i\}_{i=1}^N$. Let us also define the Lagrange polynomial weighting functions [30, 31],

$$\begin{aligned} \varphi_n(x, x_n) &= L_n(x) \\ &= \frac{(x - x_1) \cdots (x - x_{k-1})(x - x_{k+1}) \cdots (x - x_N)}{(x_n - x_1) \cdots (x_n - x_{k-1})(x_n - x_{n+1}) \cdots (x_n - x_N)}, \qquad (96) \\ &= \prod_{i=1, i \neq n}^{N} \frac{x - x_i}{x_n - x_i}, \qquad n = 1, \ldots, N. \qquad (97) \end{aligned}$$

Note that $L_n(x_n) = 1$ and $L_n(x_i) = 0$ for all $i \neq n$, i.e., we have $L_n(x_i) = \delta_{in}$ (as in Figure 12(c) for RVM and GP), which is exactly the condition for obtaining an interpolator, i.e., $\widehat{f}(x_n) = y_n$. Note that Lagrange functions $L_n(x)$ are also polynomials of order $N-1$. The polynomial interpolator $\widehat{f}(x)$ can be written as

$$\widehat{f}(x) = \sum_{n=1}^{N} L_n(x) y_n, \tag{98}$$

i.e., a linear combination of the outputs $y_n$ with $\varphi_n(x, x_n) = L_n(x)$ (or, equivalently, a linear combination of the Lagrange functions $L_n(x)$). Figure 14 gives an example of polynomial interpolation of $N = 4$ data points and the corresponding Lagrange functions $L_n(x)$.




(a) (b)

Figure 14: **(a)** Example of polynomial interpolation of $N = 4$ data points. **(b)** The corresponding Lagrange functions $\varphi_n(x, x_n) = L_n(x)$. Note that, at each data input $x_n$, all $\varphi_n(x, x_n)$ are zero except the $n$-th function where $\varphi_n(x_n, x_n) = 1$, i.e., $\varphi_n(x_j, x_n) = \delta_{jn}$, as in Figure 12(c).

### 9.2.4 Ideal Fourier interpolation

The interpolation idea is employed as upsampling in signal processing in a context of equidistant inputs [32, 33]. For simplicity, let us consider a scalar input $x \in \mathbb{R}$. Moreover, consider an infinite number of equidistant inputs, i.e.,

$$x_n = nT_0, \quad T_0 \in \mathbb{R}.$$

This is an important difference and limitation compared with the previous cases: here we consider equidistant inputs with a sampling period $T_0$. In this scenario, a well-known interpolator $\widehat{f}(x) = \sum_{n=1}^{N} \varphi_n(x, x_n) y_n$ is the ideal Fourier interpolator, where

$$\varphi_n(x, x_n) = \mathrm{sinc}(x, x_n) = T_0 \frac{\sin\left(\frac{\pi}{T_0}(x - nT_0)\right)}{\pi(x - nT_0)}. \tag{99}$$

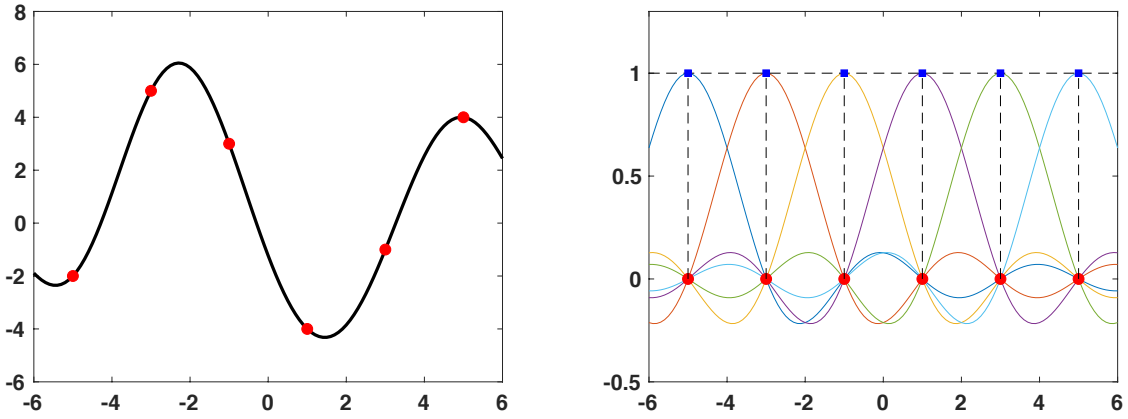
Note that $\varphi_i(x_n, x_n) = 1$, indeed

$$\text{sinc}(x_n, x_n) = T_0 \frac{\sin\left(\frac{\pi}{T_0}(nT_0 - nT_0)\right)}{\pi(nT_0 - nT_0)} = 1 \quad \text{(solving the indeterminate form)}$$

and $\varphi_n(x_i, x_n) = 0$, if $i \neq n$, indeed

$$\text{sinc}(x_i, x_n) = \frac{\sin(\pi(i-n))}{\pi(i-n)} = \frac{\sin(k\pi)}{k\pi} = 0, \qquad k = i - n \in \mathbb{Z}\backslash\{0\},$$

since $\sin(k\pi) = 0$ with $k \in \mathbb{Z}$. Then, we have again $\varphi_n(x_i, x_n) = \delta_{in}$ which is the condition to obtain $\widehat{f}(x_n) = y_n$ for all $n$. The choice of $\varphi_n(x, x_n) = \text{sinc}(x, x_n)$ is due to the Fourier transform of a sin function is an ideal rectangle filter (for more details see [32]). An example is shown in Figure 15.




(a) (b)

Figure 15: **(a)** Example of the ideal Fourier interpolation of $N = 6$ data points. **(b)** The corresponding weighting sinc functions $\varphi_n(x, x_n) = \text{sinc}(x, x_n)$. Note that, at each data input $x_n$, all $\varphi_n(x, x_n)$ are zero except the $n$-th function where $\varphi_n(x_n, x_n) = 1$, i.e., $\varphi_n(x_j, x_n) = \delta_{jn}$, as in Figure 12(c) and Figure 14(b).

### 9.2.5 Finite and infinite impulse response filters (FIR, IIR)

Here, we describe two classes of discrete filters connected to the previous techniques. Consider again a *scalar and discrete* input $x = t \in \mathbb{Z}$, representing a discrete time index, i.e., $t = \ldots, -2, -1, 0, 1, 2, 3, \ldots$. Moreover, we consider *consecutive* time instants, $t = 1, 2, \ldots, N$, we can use the simpler notation

$$\{t_n, y_n\}_{n=1}^N = \{t, y_t\}_{t=1}^N,$$

removing the sub-index $n$. In this context, the Finite Impulse Response (FIR) filters are defined

$$\begin{aligned}\widehat{f}(t) = \widehat{f_t} &= a_0 y_t + a_1 y_{t-1} + a_1 y_{t-2} \ldots + a_R y_{t-R}, \\ &= \sum_{r=0}^{R} a_r y_{t-r},\end{aligned} \tag{100}$$

where $a_r$ and $R$ are constant values decided by the user [32, 33]. The value $R$ is the *order* of the filter. Comparing Eq. (89) and the expression above, we can see that a FIR filter is also a linear smoother with coefficients $a_r$ for $r = 0, \ldots, R$, and the linear combination only considers the previous $R$ samples $y_{t-1}, \ldots, y_{t-R}$ and the current sample $y_t$. If $a_r = \frac{1}{R}$ for all $r$, we have a low-pass filter which compute the arithmetic mean of $R+1$ outputs $y_t, y_{t-1}, \ldots, y_{t-R}$. Therefore, we have a "sliding" memory window of length $R$. The FIR filters are similar to the kNN approach but considering $k = R$ nearest neighbors only in *the past*, i.e., $t' \leq t$ (not in the future, i.e., $t' > t$). In a FIR filter, the output sequence is a weighted sum of the most recent values. In terms of neural network architectures, this can be seen as a linear type of time delay neural network with fixed weights (a standard multi-layer perceptron taking a time window as input) [34].

A memory window of infinite length can be obtained considering a FIR filter of order $R = \infty$, i.e., with an infinite amount of coefficients. Thus, the sum in Eq. (100) would be a series. In that case, the FIR filter with $R = \infty$ is converted into an Infinite Impulse Response (IIR) filter [32, 33]. An equivalent way of expressing a IIR filter is incorporating an autoregressive part in Eq. (100), i.e.,

$$\widehat{f_t} = \sum_{\ell=1}^{L} b_\ell \widehat{f}_{t-\ell} + \sum_{r=0}^{R} a_r y_{t-r}, \tag{101}$$

where $b_\ell$ and $L$ are anther constant values, decided by the user. In the context of stochastic filtering, these filters are also called Autoregressive Moving Average (ARMA) models. In terms of neural network architectures, an IIR filter can be interpreted as a linear version of recurrent neural network.

# 10 GPs and Kalman filtering

In this section, we describe the connection, differences and similarities between the GP approach and the Kalman filtering (KF) approach, from the simplest case to the most general case, progressively [20, 21, 5]. The Kalman filter is a recursive MMSE estimator of the state of in a linear state-space model with Gaussian noise perturbations (see Eq. (105), below). Note that the MMSE and MAP estimators coincide in this context, i.e., under the assumptions of linearity and Gaussian noises. For more clarifications, see below.

## 10.1 Kalman filter in discrete time

For the sake of simplicity, we change the notation considering *scalar and discrete* inputs

$$x = t \in \mathbb{Z}, \tag{102}$$

representing a discrete time index. Later on, we will consider also a continuous time index. The dataset is then $\{t_n, y_n\}_{n=1}^N$. Moreover, if we consider *consecutive* time instants, $t = 1, 2, \ldots, N$, we can use the notation

$$\{t_n, y_n\}_{n=1}^N = \{t, y_t\}_{t=1}^N,$$

removing the sub-index $n$. The observation vector and the corresponding values of the hidden function is

$$\mathbf{y} = [y_1, \ldots, y_N]^\top, \text{ and } \mathbf{f} = [f_1, \ldots, f_N]^\top,$$

and the observation model is

$$y_t = f_t + e_t, \quad \text{and} \quad \mathbf{y} = \mathbf{f} + \mathbf{e}, \tag{103}$$

where $\mathbf{e} = [e_1, \ldots, e_N]^\top \sim \mathcal{N}(\mathbf{e}|\mathbf{0}, \sigma_e^2 \mathbf{I}_N)$ with $\mathbf{I}_N$ is an $N \times N$ unit matrix. The likelihood function is again $p(\mathbf{y}|\mathbf{f}) = \mathcal{N}(\mathbf{y}|\mathbf{f}, \sigma_e^2 \mathbf{I}_N)$. Therefore, the observation model (likelihood) is exactly the same.

**Remark 21.** *The index $t$ is playing the role of the input, the variable $f_t = f(t)$ is the hidden function at the instant $t$ and $y_t$ is the corresponding observation at the input $t$. Note that we are also considering a normalized uniform sampling case, i.e., $t_i - t_j = 1$ for all $i, j$.*

However, instead of assuming directly a covariance function $k(t, t')$ as prior information, we consider an autoregressive (AR) model over $f_t$, i.e.,

$$f_t = \gamma f_{t-1} + v_t, \quad \text{with} \quad |\gamma| < 1, \tag{104}$$

where $v_t \sim \mathcal{N}(v|0, \sigma_v^2)$, inducing a transition probability $p(f_t|f_{t-1})$. The complete *state-space model* is formed by the transition (prior) and observation (likelihood) equations (densities), i.e.,

$$\begin{cases} f_t = \gamma f_{t-1} + v_t, \\ y_t = f_t + e_t, \end{cases} \Longrightarrow \begin{cases} p(f_t|f_{t-1}), \\ p(y_t|f_t), \end{cases} \qquad t = 1, 2 \ldots, N. \tag{105}$$

Assuming also $p(f_1|f_0) = p(f_1) \sim \mathcal{N}(f_1|0, \sigma_v^2)$, the complete prior density and likelihood function are

$$p(\mathbf{f}) = \prod_{t=1}^N p(f_t|f_{t-1}) = \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{\Psi}), \tag{106}$$

$$p(\mathbf{y}|\mathbf{f}) = \prod_{t=1}^N p(y_t|f_t) = \mathcal{N}(\mathbf{y}|\mathbf{f}, \sigma_e^2 \mathbf{I}_N). \tag{107}$$

where the $N \times N$ covariance matrix $\mathbf{\Psi}$ is generated by the autoregressive process in Eq. (104) is given below.

**Remark 22.** *The recursion $f_t = \gamma f_{t-1} + v_t$ (equivalent to transition probability $p(f_t|f_{t-1})$) induces a prior over $\mathbf{f}$. Namely, the recursive equation plays the same role of a kernel function $k(t, t')$ in the GP derivation. Indeed, we see below that we can obtain an equivalent a kernel/covariance function $k(t, t')$.*

### 10.1.1 Equivalent kernel function of an AR model

For simplicity, let start the recursion in Eq. (104) with $f_0 = 0$. Then, we can write

$$E[f_t] = \gamma E[f_{t-1}] + E[v_t] = 0, \qquad \forall t. \tag{108}$$

Hence, $E[f_1, \ldots, f_N] = 0$. Moreover, regarding the variance, we have

$$\begin{aligned} \text{var}[f_t] &= \gamma^2 \text{var}[f_{t-1}] + \sigma_v^2 \\ \text{var}[f_t] &= \gamma^4 \text{var}[f_{t-2}] + (\gamma^2 + 1)\sigma_v^2 \\ &\vdots \\ \text{var}[f_t] &= \gamma^{2t} \text{var}[f_0] + \left( \sum_{i=0}^{t-1} \gamma^{2i} \right) \sigma_v^2. \end{aligned} \tag{109}$$

Since we start with $f_0 = 0$ then $\text{var}[f_0] = 0$. However, in any case, the variance of initial condition $\text{var}[f_0]$ goes to zero as $t$ approaches infinity since $|\gamma| < 1$. Therefore, we can write

$$\text{var}[f_t] = \frac{1 - \gamma^{2t}}{1 - \gamma^2} \sigma_v^2, \tag{110}$$

and as $t \to \infty$, the stationary variance is

$$\sigma_f^2 = \text{var}[f_\infty] = \frac{\sigma_v^2}{1 - \gamma^2}. \tag{111}$$

Recall that $|\gamma| < 1$, so that the expression above is finite and positive. The diagonal of $\mathbf{\Psi}_t$ is given by the values $\text{var}[f_t]$. Similarly, the stationary auto-covariance function

$$\begin{aligned} r(\tau) = \psi(f_t, f_{t+\tau}) = \psi(t, t+\tau) = & \\ &= E[(f_t - \mu_t)(f_{t+\tau} - \mu_{t-\tau})], \\ &= E[f_t f_{t+\tau}], \end{aligned}$$

depends only to the instants $t$ and $t - \tau$ (actually, only on the different $\tau$). It is possible to show that

$$r(\tau) = \gamma \cdot r(\tau - 1) \quad \text{where} \quad r(0) = \sigma_f^2. \tag{112}$$

Then, we have $r(1) = \gamma \sigma_f^2$, $r(2) = \gamma^2 \sigma_f^2$, so that $r(\tau) = \gamma^{|\tau|} \sigma_f^2$. Replacing $\sigma_f^2 = \frac{\sigma_v^2}{1-\gamma^2}$ in the previous expression, we obtain the formula of the stationary covariance function

$$r(\tau) = \psi(t, t \pm \tau) = \frac{\gamma^{|\tau|} \sigma_v^2}{1 - \gamma^2}, \quad \tau \in \mathbb{Z}. \tag{113}$$

So that, after a transient, we can write

$$[\mathbf{\Psi}]_{i,j} = \psi(i, j) = \frac{\gamma^{|i-j|} \sigma_v^2}{1 - \gamma^2}, \tag{114}$$

for all $i, j = 1, \ldots, N$.

### 10.1.2 Covariance and precision matrices in the stationary regime

Let us now assume that the marginal distribution of $f_0$ is Gaussian with mean zero and variance $\frac{\sigma_v^2}{1-\gamma^2}$ (recall that $|\gamma| < 1$), which is simply the stationary distribution of this process. Therefore, considering as an example $N = 5$, the covariance $\mathbf{\Psi}$ and the precision $\mathbf{P} = \mathbf{\Psi}^{-1}$ matrices in the stationary regime is

$$\mathbf{\Psi} = \sigma_v^2 \begin{bmatrix} 1 & \gamma & \gamma^2 & \gamma^3 & \gamma^4 \\ \gamma & 1 & \gamma & \gamma^2 & \gamma^3 \\ \gamma^2 & \gamma & 1 & \gamma & \gamma^2 \\ \gamma^3 & \gamma^2 & \gamma & 1 & \gamma \\ \gamma^4 & \gamma^3 & \gamma^2 & \gamma & 1 \end{bmatrix}, \quad \mathbf{P} = \frac{1}{\sigma_v^2} \begin{bmatrix} 1 & -\gamma & 0 & 0 & 0 \\ -\gamma & 1+\gamma^2 & -\gamma & 0 & 0 \\ 0 & -\gamma & 1+\gamma^2 & -\gamma & 0 \\ 0 & 0 & -\gamma & 1+\gamma^2 & -\gamma \\ 0 & 0 & 0 & -\gamma & 1 \end{bmatrix}.$$

Note that the precision matrix $\mathbf{P}$ is the tridiagonal matrix, i.e., with zero entries outside the diagonal and first off-diagonals. The tridiagonal form is due to the fact that $f_i$ and $f_j$ are *conditionally independent* for $|i-j| > 1$ given the rest of variables. It is interesting to remark the entries in the covariance matrix $\mathbf{\Psi}$ only give direct information about the *marginal* dependence structure, not about the *conditional* dependence.

### 10.1.3 Filtering, smoothing, prediction

Given the state-space model in Eq. (105), at each time instant, we have an additional variable $f_t$ and an additional observation $y_t$. Several algorithms in the literature tackle different inference problems, corresponding to different posterior and/or predictive densities.

**Complete and partial smoothing.** As we already have seen, the complete smoothing problem consider the joint posterior density

$$p(f_1, \ldots, f_N | y_1, \ldots, y_N) = p(\mathbf{f}|\mathbf{y}).$$

Other partial smoothing densities can be considered in this scenario, for instance,

$$p(f_t | y_1, \ldots, y_N) = p(f_t|\mathbf{y}), \quad t < N.$$

or considering different time instants, for instance, $p(f_{t_1}, f_{t_2}|y_1, \ldots, y_T)$ with $t_1, t_2 < N$. More generally, people are often interested in studying the posterior

$$p(f_1, \ldots, f_t | y_1, \ldots, y_t), \quad t \leq N,$$

where we analyze the vector $[f_1, \ldots, f_t]^\top$ considering only the observations $[y_1, \ldots, y_t]^\top$ (assuming unknown the data $y_{t+1}, \ldots, y_N$).

**Filtering.** The filtering problem corresponds to the study of the following posterior densities

$$p(f_t | y_1, \ldots, y_t), \quad t \leq N, \tag{115}$$

where we have only the variable $f_t$ given all the measurements $[y_1, \ldots, y_t]^\top$ obtained so far, i.e., assuming unknown the future observations $y_{t+1}, \ldots, y_N$. Generally, the people consider the sequential problem considering the sequence of filtering posteriors

$$\begin{aligned} &p(f_1|y_1), \\ &p(f_2|y_1, y_2), \\ &\vdots \\ &p(f_t|y_1, \ldots, y_{t-1}, y_t), \end{aligned}$$

providing recursive solutions.

**Prediction at lag-$\tau$.** in time series analysis, one often consider the predictive density

$$p(f_{t+\tau}|y_1, \ldots, y_t), \quad \tau \geq 1,$$

where we are interesting in inferring the variable $f_{t+\tau}$ in the future instant $t' = t + \tau$, observing only the measurements until time $t$.

### 10.1.4 Discrete Kalman solution for filtering

**Remark 23.** *The standard discrete Kalman filter provides the recursive equations for computing the mean $\widehat{\mu}_{t|t}$ and variance $\widehat{\sigma}^2_{t|t}$ of the filtering posterior density, i.e.,*

$$p(f_t|y_1, \ldots, y_t) = \mathcal{N}(f_t|\widehat{\mu}_{t|t}, \widehat{\sigma}^2_{t|t}). \tag{116}$$

Let us also denote $\widehat{\mu}_{t|t-1}$ and $\widehat{\sigma}^2_{t|t-1}$ the mean and variance of the predictive density

$$p(f_t|y_1, \ldots, y_{t-1}) = \mathcal{N}(f_t|\widehat{\mu}_{t|t-1}, \widehat{\sigma}^2_{t|t-1}). \tag{117}$$

The Kalman equations provide recursively the means and variances of these two densities as new observations are obtained. From the instant $t-1$ to $t$, the sequential Kalman solution, for computing mean and variance of the predictive density $p(f_t|y_1, \ldots, y_{t-1})$, is then given by

$$\begin{cases} \widehat{\mu}_{t|t-1} = \gamma \widehat{\mu}_{t-1|t-1} \\ \widehat{\sigma}^2_{t|t-1} = \gamma^2 \widehat{\sigma}^2_{t-1|t-1} + \sigma_v^2 \end{cases} \tag{118}$$

and, for the filtering pdf $p(f_t|y_1, \ldots, y_t)$, we have

$$\begin{cases} \widehat{\mu}_{t|t} = \dfrac{\sigma_e^2}{\widehat{\sigma}^2_{t|t-1} + \sigma_e^2} \widehat{\mu}_{t|t-1} + \dfrac{\widehat{\sigma}^2_{t|t-1}}{\widehat{\sigma}^2_{t|t-1} + \sigma_e^2} y_t \\ \widehat{\sigma}^2_{t|t} = \dfrac{\widehat{\sigma}^2_{t|t-1} \sigma_e^2}{\widehat{\sigma}^2_{t|t-1} + \sigma_e^2}. \end{cases} \tag{119}$$

Defining the precision values as $\widehat{p}_{t|t-1} = \frac{1}{\widehat{\sigma}^2_{t|t-1}}$ and $p_e = \frac{1}{\sigma_e^2}$, we can rewrite the last two equations as

$$\begin{cases} \widehat{\mu}_{t|t} = \dfrac{\widehat{p}_{t|t-1}}{\widehat{p}_{t|t-1} + \bar{p}_e}\bar{\mu}_t + \dfrac{\bar{p}_e}{\widehat{p}_{t|t-1} + \bar{p}_e} y_t, \\ \widehat{p}_{t|t} = \widehat{p}_{t|t-1} + p_e. \end{cases} \tag{120}$$

**Remark 24.** *The standard Kalman filter focuses on the sequence of filtering densities, $p(f_t|y_1, \ldots, y_t) = \mathcal{N}(f_t|\widehat{\mu}_t, \widehat{\sigma}^2_t)$ for $t = 1, \ldots, N$. We can have a complete equivalence with the GP solution for smoothing, if we consider a Kalman approach for smoothing, i.e., considering the density $p(f_{1:N}|y_{1:N}) = p(\mathbf{f}|\mathbf{y})$.*

Note that we have considered scalar values $f_1, \ldots, f_N$ and $y_1, \ldots, y_N$ to be coherent to the rest of the paper, and facilitate the comparison with the other techniques. However, the Kalman filter can be directly generalized for multivariate/multioutput case, i.e., at each iteration we can have vectors $\mathbf{f}_t$ and $\mathbf{y}_t$ of the observations.

### 10.1.5 Backward filter for partial Kalman smoothing

Let us consider that we have already run the *forward* Kalman filter described above, obtaining $\widehat{\mu}_{t|t-1}$, $\widehat{\sigma}^2_{t|t-1}$, $\widehat{\mu}_{t|t}$ and $\widehat{\sigma}^2_{t|t}$ for all $t = 1, \ldots, N$. Now, we focus on the partial smoothing densities, i.e.,

$$p(f_t|y_1, \ldots, y_N) = p(f_t|\mathbf{y}) = \mathcal{N}(f_t|\widehat{\mu}_{t|N}, \widehat{\sigma}^2_{t|N}), \qquad \forall t < N. \tag{121}$$

Then, we can consider the following backward recursion (from $t = N-1$ to $t = 1$):

$$\begin{cases} \widehat{\mu}_{t|N} = \widehat{\mu}_{t|t} + \gamma \dfrac{\widehat{\sigma}^2_{t|t}}{\widehat{\sigma}^2_{t+1|t}} (\widehat{\mu}_{t+1|N} - \widehat{\mu}_{t+1|t}), \\ \widehat{\sigma}^2_{t|N} = \widehat{\sigma}^2_{t|t} + \left(\gamma \dfrac{\widehat{\sigma}^2_{t|t}}{\widehat{\sigma}^2_{t+1|t}}\right)^2 (\widehat{\sigma}_{t+1|N} - \widehat{\sigma}_{t+1|t}). \end{cases} \tag{122}$$

For the solution of the complete smoothing problem, see [35, 36, 37, 5]. Let us consider again the GP solution in Eqs. (62)–(63) computed in a training input $t = 1, 2, \ldots, N$, i.e.,

$$\begin{aligned} \mu_{f|y}(t) &= \widehat{f}(t) = \boldsymbol{\psi}(t)^\top (\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \mathbf{y}, \\ \sigma^2_{f|y}(t) &= \psi(t,t) - \boldsymbol{\psi}(t)^\top (\boldsymbol{\Psi} + \sigma_e^2 \mathbf{I}_N)^{-1} \boldsymbol{\psi}(t), \end{aligned}$$

where $\psi(t, t') = \frac{\gamma^{|t-t'|}\sigma_v^2}{1-\gamma^2}$ with $t, t' \in \mathbb{N}^+$, $\mathbf{y} = [y_1, \ldots, y_N]^\top$,

$$\boldsymbol{\psi}(t) = [\psi(t,1), \ldots, \psi(t,N)]^\top, \quad \text{and} \quad [\boldsymbol{\Psi}]_{i,j} = \psi(i,j) = \frac{\gamma^{|i-j|}\sigma_v^2}{1-\gamma^2}.$$

These mean and variance completely define the partial smoothing density

$$p(f_t|\mathbf{y}) = \mathcal{N}(f_t|\mu_{f|y}(t), \sigma^2_{f|y}(t)), \qquad \forall t < N. \tag{123}$$

**Remark 25.** *It is possible to show that $\mu_{f|y}(t) = \widehat{\mu}_{t|N}$ and $\sigma^2_{f|y}(t) = \widehat{\sigma}^2_{t|N}$, clearly considering the equivalent kernel function, induced by the propagation equation in the state-space model.*

## 10.2 Continous-time state-space models

In order to obtain a complete equivalence to GP models and a sequential Kalman solutions we have to consider a continuous input variable, $t \in \mathbb{R}$. In this scenario, the space-state model is formed by a linear differential equation with constant coefficients and an observation equation. The prior information is included by the linear differential equation which plays the same role of the kernel/covariance function in the GP models. A linear differential equation with constant coefficients of order $R$ with a Gaussian white noise input $v(t)$,

$$\frac{d^R f(t)}{dt^R} + a_{R-1}\frac{d^{R-1} f(t)}{dt^{R-1}} + \cdots + a_1 \frac{df(t)}{dt} + a_0 f(t) = v(t), \tag{124}$$

can be rewritten as a first order vector Markov process, i.e.,

$$\frac{d\mathbf{f}(t)}{dt} = \mathbf{A}\mathbf{f}(t) + \mathbf{b}v(t) \tag{125}$$

where $\mathbf{f}(t) = \left[\frac{d^{R-1}f(t)}{dt^{R-1}}, \ldots, \frac{df(t)}{dt}, f(t)\right]^\top$ is an $R \times 1$ vector,

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ -a_{R-1} & -a_{R-2} & \cdots & -a_1 & -a_0 \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix},$$

are a $R \times R$ matrix and an $R \times 1$ vector, respectively. Note also that

$$f(t) = \mathbf{b}^\top \mathbf{f}(t),$$

i.e., we can extract $f(t)$ from the vector $\mathbf{f}(t)$ by the multiplication above. It possible to compute the *power spectral density* of $\mathbf{f}(t)$ (a) replacing $f(t) = \mathbf{b}^\top \mathbf{f}(t)$ in Eq. (125), (b) taking the Fourier transform to both sides of Eq. (125), after replacing $f(t) = \mathbf{b}^\top \mathbf{f}(t)$. Moreover, since the noise $v(t)$ is white, we have the its power spectral density is $S_V(\omega) = c$ where $c > 0$. After some algebra and rearrangement, this procedure yields [20, 21]

$$S_F(\omega) = \mathbf{b}^\top (\mathbf{A} + j\omega\mathbf{I})^{-1}\mathbf{b} S_V(\omega) \mathbf{b}^\top \left[(\mathbf{A} + j\omega\mathbf{I})^{-1}\right]^\top \mathbf{b}, \tag{126}$$

$$= c\mathbf{b}^\top (\mathbf{A} + j\omega\mathbf{I})^{-1}\mathbf{b}\mathbf{b}^\top \left[(\mathbf{A} + j\omega\mathbf{I})^{-1}\right]^\top \mathbf{b}, \tag{127}$$

In the stationary state (i.e., when the process has run an infinite amount of time), the stationary covariance function $\psi(t, t') = \psi(\tau)$ of $f(t)$ (with $\tau = |t - t'|$) can be expressed as inverse Fourier transform of its spectral density $S_F(\omega)$, hence

$$\psi(\tau) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} S_F(\omega) e^{j\omega\tau} d\omega. \tag{128}$$

We have shown that a linear differential equation determines a covariance function over $f(t)$ [20, 21, 5].

# 11 Summary

In this work, we have provided a joint introduction to RVMs and GPs for regression, including within this framework the tasks of filtering, smoothing, and interpolation. The probabilistic derivation of both methods is given, along with several observations and recommendations for the use of these methods in practice. We have highlighted the connections between them and to related techniques such as kernel ridge regression, kernel smoothers, Fourier interpolators and Kalman filtering. We have also remarked the benefits and drawbacks of each schemes. RVMs allow the choice of more general basis functions whereas the behavior of the predictive variance is generally counterintuitive. GPs present a good behavior of the predictive variance but the choice of kernel functions is more restrictive. The Kalman smoothing method provides the same solution as that of a GP with a specific kernel function, which is implicitly induced by the considered propagation equation in the state-space model.

# References


[1] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[2] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer School on Machine Learning*. Springer, 2003, pp. 63–71.

[3] D. J. MacKay, "Introduction to Gaussian processes," *NATO ASI Series F Computer and Systems Sciences*, vol. 168, pp. 133–166, 1998.

[4] B. Schölkopf, A. J. Smola, F. Bach *et al.*, *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[5] S. Särkkä, *Bayesian Filtering and Smoothing*, ser. Institute of Mathematical Statistics Textbooks. Cambridge University Press, 2013.

[6] C. E. Rasmussen, "Gaussian processes for machine learning," in *the MIT Press*, 2006, pp. 1–245.

[7] M. E. Tipping, "Sparse Bayesian learning and the Relevance Vector Machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[8] J. Q. Candela, "Learning with uncertainty - Gaussian Processes and Relevance Vector Machines," *Technical University of Denmark*, pp. 1–152, 2004.

[9] U. B. Gewali, S. T. Monteiro, and E. Saber, "Gaussian Processes for vegetation parameter estimation from hyperspectral data with limited ground truth," *Remote Sensing*, vol. 11, no. 13, p. 1614, 2019.

[10] M. Alvarez, D. Luengo, and N. D. Lawrence, "Latent force models," in *Artificial Intelligence and Statistics*, 2009, pp. 9–16.

[11] J. L. Gómez-Dans, P. E. Lewis, and M. Disney, "Efficient emulation of radiative transfer codes using Gaussian processes and application to land surface parameter inferences," *Remote Sensing*, vol. 8, no. 2, p. 119, 2016.

[12] D. H. Svendsen, L. Martino, M. Campos-Taberner, F. J. García-Haro, and G. Camps-Valls, "Joint Gaussian processes for biophysical parameter retrieval," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 3, pp. 1718–1727, 2017.

[13] D. H. Svendsen, L. Martino, and G. Camps-Valls, "Active emulation of computer codes with Gaussian processes–application to remote sensing," *Pattern Recognition*, vol. 100, p. 107103, 2020.

[14] L. Martino, J. Vicent, and G. Camps-Valls, "Automatic emulation by adaptive Relevance Vector Machines," *Scandinavian Conference on image analysis (SCIA)*, vol. 100, pp. 1–11, 2017.

[15] B. W. Silverman, "Some aspects of the Spline Smoothing approach to non-parametric regression curve fitting," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 47, no. 1, pp. 1–52, 1985.

[16] R. Szeliski, "Regularization uses fractal priors," in *In Proceedings of the 6th National Conference on Artificial Intelligence (AAAI)*, 1987.

[17] C. E. Rasmussen and J. Quiñonero Candela, "Healing the Relevance Vector Machine through augmentation," in *Proceedings of the 22nd International Conference on Machine Learning (ICML-05)*. New York, NY, USA: Association for Computing Machinery, 2005, pp. 689–696.

[18] T. Poggio and F. Girosi, "Networks for approximation and learning," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1481–1497, 1990.

[19] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012.

[20] J. Hartikainen and S. Särkkä, "Kalman filtering and smoothing solutions to temporal Gaussian process regression models," in *2010 IEEE International Workshop on Machine Learning for Signal Processing*, 2010, pp. 379–384.

[21] S. Särkkä and J. Hartikainen, "Infinite-dimensional Kalman filtering approach to spatio-temporal Gaussian Process regression," in *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, vol. 22. PMLR, 21-23 Apr 2012, pp. 993–1001.

[22] M. A. Álvarez, L. Rosasco, and N. D. Lawrence, "Kernels for vector-valued functions: A review," *Found. Trends Mach. Learn.*, vol. 4, no. 3, p. 195266, 2012.

[23] J. Read and L. Martino, "Probabilistic regressor chains with Monte Carlo methods," *Neurocomputing*, vol. 413, pp. 471 – 486, 2020.

[24] G. Wahba, *Spline Models for Observational Data*. Society for Industrial and Applied Mathematics, Philadelphia, PA., 1990.

[25] G. Kimeldorf and G. Wahba, "A correspondence between bayesian estimation of stochastic processes and smoothing by Splines," *Annals of Mathematical Statistics*, vol. 41, pp. 495–502, 1970.

[26] Z. Ghahramani, "A tutorial on Gaussian Processes (or why i dont use SVMs)," *MLSS Workshop talk by Zoubin Ghahramani on Gaussian Processes (Slides).*, 2011.

[27] L. Martino, D. Luengo, and J. Miguez, *Independent Random Sampling Methods*. springer, 2018.

[28] C. Robert and G. Casella, *Monte Carlo statistical methods*. Springer, 2004.

[29] D. Shepard, "A two-dimensional interpolation function for irregularly-spaced data," in *Proceedings of the 1968 23rd ACM National Conference*, 1968, p. 517524.

[30] R. L. Burden and J. D. Faires, *Numerical Analysis*. Brooks Cole, 2000.

[31] B. F. Plybon, *An Introduction to Applied Numerical Analysis*. Boston, MA: PWS-Kent, 1992.

[32] E. W. Kamen and B. S. Heck, *Fundamental of signals and systems using the web and Matlab*. Prentice-Hall, Inc., 2000.

[33] J. G. Proakis, *Digital Communications (4th edition)*. Singapore: McGraw-Hill, 2000.

[34] K.-L. Du and M. N. Swamy, *Neural Networks and Statistical Learning*. Springer Publishing Company, Incorporated, 2013.

[35] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA Journal*, vol. 3, no. 8, pp. 1445–1450, 1965.

[36] G. A. Einicke, "Asymptotic optimality of the minimum-variance fixed-interval smoother," *IEEE Transactions on Signal Processing*, vol. 55, no. 4, pp. 1543–1547, 2007.

[37] G. A. Einicke, J. C. Ralston, C. O. Hargrave, D. C. Reid, and D. W. Hainsworth, "Longwall mining automation an application of minimum-variance smoothing [applications of control]," *IEEE Control Systems Magazine*, vol. 28, no. 6, pp. 28–37, 2008.

[38] W. W. Hager, "Updating the inverse of a matrix," *SIAM Review*, vol. 31, no. 2, pp. 221–239, 1989.

# A Alternative formulation of the RVM variance

In this section, we analyze the expression of the variance of RVM.
Let us consider the following generic matrices $\mathbf{Z}$ of size $M \times M$, $\mathbf{U}$ of size $N \times M$, $\mathbf{L}$ of size $N \times N$ and $\mathbf{V}$ of size $M \times N$, the following *matrix inversion lemma* [38], [6, Appendix A] is satisfied,

$$\left(\mathbf{Z} + \mathbf{U}\mathbf{L}\mathbf{V}^\top\right)^{-1} = \mathbf{Z}^{-1} - \mathbf{Z}^{-1}\mathbf{U}\left(\mathbf{L}^{-1} + \mathbf{V}^\top\mathbf{Z}^{-1}\mathbf{U}\right)^{-1}\mathbf{V}^\top\mathbf{Z}^{-1}. \tag{129}$$

Using this matrix inversion lemma with $Z^{-1} = \mathbf{\Sigma}_\rho$, $\mathbf{L}^{-1} = \sigma_e^2\mathbf{I}_N$ and $U = V = \mathbf{\Psi}^\top$ and considering the variance of RVM, we obtain

$$\left(\mathbf{\Sigma}_\rho^{-1} + \frac{1}{\sigma_e^2}\mathbf{\Psi}^\top\mathbf{\Psi}\right)^{-1} = \mathbf{\Sigma}_\rho - \mathbf{\Sigma}_\rho\mathbf{\Psi}^\top\left(\sigma_e^2\mathbf{I}_N + \mathbf{\Psi}\mathbf{\Sigma}_\rho\mathbf{\Psi}^\top\right)^{-1}\mathbf{\Psi}\mathbf{\Sigma}_\rho. \tag{130}$$

Replacing in Eq. (32), where we consider the matrix $\mathbf{\Psi}$ instead of $\mathbf{\Psi}$, we have

$$\begin{aligned}
\widehat{\sigma}(\mathbf{x}) &= \boldsymbol{\psi}(\mathbf{x})^\top\left(\mathbf{\Sigma}_\rho - \mathbf{\Sigma}_\rho\mathbf{\Psi}^\top\left(\sigma_e^2\mathbf{I}_N + \mathbf{\Psi}\mathbf{\Sigma}_\rho\mathbf{\Psi}^\top\right)^{-1}\mathbf{\Psi}\mathbf{\Sigma}_\rho\right)\boldsymbol{\psi}(\mathbf{x}), \\
&= \boldsymbol{\psi}(\mathbf{x})^\top\mathbf{\Sigma}_\rho\boldsymbol{\psi}(\mathbf{x}) - \boldsymbol{\psi}(\mathbf{x})^\top\mathbf{\Sigma}_\rho\mathbf{\Psi}^\top\left(\sigma_e^2\mathbf{I}_N + \mathbf{\Psi}\mathbf{\Sigma}_\rho\mathbf{\Psi}^\top\right)^{-1}\mathbf{\Psi}\mathbf{\Sigma}_\rho\boldsymbol{\psi}(\mathbf{x}).
\end{aligned}$$