Introduction to Gaussian Processes

Rui-Yang Zhang

Contents

Contents 1

Preface 2

1 Introduction to Gaussian Processes 3
- 1.1 From Gaussian Distribution To Gaussian Process 3
- 1.2 Gaussian Properties and Linear Algebra 5
- 1.3 Gaussian Process Regression 7
  - 1.3.1 Linear Regression 7
  - 1.3.2 Regression using GPs 9
- 1.4 Covariance Function 11
  - 1.4.1 Definitions and General Properties 11
  - 1.4.2 Examples of Covariance Functions 11
  - 1.4.3 Composing Kernels 12
  - 1.4.4 Hyperparameters and Model Selection 12
- 1.5 Spectral Representation of Stationary Kernels 12
  - 1.5.1 Spectral Densities Examples 13
- 1.6 Differentiability of Gaussian Processes 14

2 Gaussian Processes for Vector Fields 16
- 2.1 Basics of Vector-Output GP 16
- 2.2 Velocity Decomposition 18
- 2.3 Helmholtz Decomposition 19
- 2.4 Modelling Ocean Currents 20

3 Building Kernel Spectra using Mixtures 22
- 3.1 Spectral Mixture 22
  - 3.1.1 Gaussian Mixtures 23
  - 3.1.2 Block Mixtures 26
- 3.2 Vector-Output Spectral Mixture 27
  - 3.2.1 Spectra of Vector-Output Kernels 27
  - 3.2.2 Gaussian Mixtures 28
  - 3.2.3 Block Mixtures 29

A Mathematical Background 31
- A.1 Linear Algebra 31
- A.2 Vector Calculus 32
- A.3 Fourier Analysis 33
  - A.3.1 Properties of Fourier Transform 33

Reference 34

Preface

This notes provides a short introduction to Gaussian processes, and describe some of its fundamental properties. We will provide two advanced chapters. Chapter 2 will discuss the construction of a GP that models a vector field, and Chapter 3 will discuss the spectral mixture kernels, which leverage the spectral representation of a kernel and use that to build more expressive kernels. Some mathematical backgrounds are provided in the appendix.

Chapter 1

Introduction to Gaussian Processes

In this chapter, we will introduce the basics of a Gaussian process (GP) and describe some of its fundamental properties. In Section 1.1, we motivate the construction of a Gaussian process by viewing it as a infinite-dimensional extension to a multivariate Gaussian distribution. We state some basic linear algebra and Gaussian distribution properties in Section 1.2, which will be helpful in later sections, such as the derivation of GP regression in Section 1.3. Some discussions of the covariance function of a GP, which is a key design choice, will be stated in Section 1.4 and some extensions regarding its spectral properties will be mentioned in Section 1.5. Finally, we will conclude with a brief discussion on the differentiablity and continuity of a GP related to those properties of a GP's kernel in Section 1.6.

Most of the material of this chapter are based on Williams and Rasmussen (2006).

1.1 From Gaussian Distribution To Gaussian Process

A **stochastic process** is a sequence of random variables $\{X_t\}_t$ indexed by t. The index usually represents the time steps of the process, but it is not always the case. In the case of the **Gaussian process** (GP), and in particular the applications of GP that we consider here, the index is not related to time but to space. This change will become clear soon.

A univariate Gaussian distribution $N(\mu, \sigma^2)$ with mean μ and variance σ^2 is a common object of interest. It is one-dimensional, and we can generalise it to make it finite-dimensional, which we denote by $N_d(\mu, \Sigma)$ and call it a multivariate Gaussian, where the **mean vector** μ here is now a d-vector and **covariance matrix** Σ is a $d \times d$ matrix that is symmetric and positive semi-definite. A matrix Σ is symmetric if $\Sigma^T = \Sigma$, and it is positive semi-definite if $x^T \Sigma x \geq 0$ for all d-vectors x.

For simplicity, we will let the means and the mean vectors we consider here zero unless stated otherwise. Consider a multivariate Gaussian with highly correlated dimensions. Normally, when we plot a sample from a d-dimensional Gaussian, we will plot it in the d-dimensional space. Here, we will do something different, and plot the values of each dimension in the same plot, sequentially. In the following plots, we can see how a sample might look like in this format with $d = 3, 10, 100$.

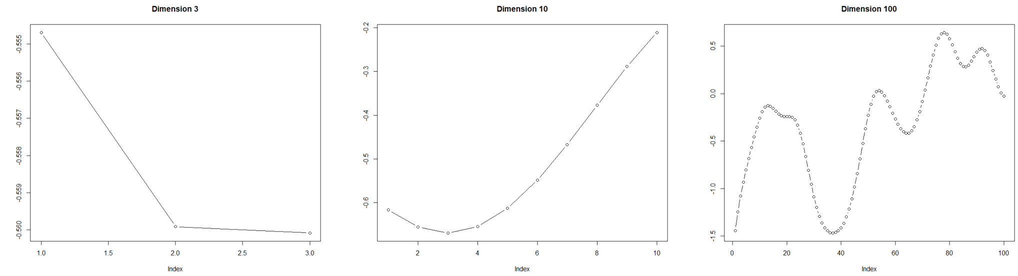

Figure 1: Sequentially plotted samples from d-dimensional Gaussian with $d = 3, 10, 100$.

As shown in Figure 1, the highly correlated Gaussian plotted sequentially makes it look very much like a (smooth) function. This inspires us to consider the infinite-dimensional extension of a multivariate Gaussian and use that to approximate a function. This is the idea of **Gaussian process**.

Formally, the **Gaussian process** (GP) $y = \{y(x)\}_{x \in \mathcal{X}}$ is a stochastic process indexed by space $x \in \mathcal{X}$ with a mean function $\mu(\cdot)$ and a covariance function $k(\cdot, \cdot)$ such that $\mathbb{E}[y(x)] = \mu(x)$ and $\operatorname{Cov}(y(x), y(x')) = k(x, x')$. We will denote such a process as

$$y(\cdot) \sim GP(\mu(\cdot), k(\cdot, \cdot)).$$

A key feature of a GP is that any finite-dimensional segment (i.e. isolating finitely many points from the full process) is a multivariate Gaussian distribution.

The covariance function needs to satisfy some requirements for it to be a suitable covariance for a GP. The requirement is essentially an extension of the requirement of covariance matrices for a multivariate Gaussian. The covariance function k of a GP needs to be **symmetric** (so $k(x, y) = k(y, x)$ for any x, y) and **positive semi-definite**, in the sense that for any $x_1, x_2, \ldots, x_n$, the matrix K formed by setting $K_{ij} = k(x_i, x_j)$ needs to be a positive semi-definite matrix. The covariance function is a way to ensure the dependency/similarity of points across indices.

The two degrees of freedom of a GP are its mean function and its covariance function, and we have assumed the mean is zero. It is not hard to assume that the covariance function characterises the GP to a very large extent. The detailed ways of how properties of the covariance function imply the properties of the GP, as well as some common covariance functions, will be discussed in Section 1.4.

A Gaussian process defined on the real line $\mathbb{R}$ will have (uncountably) infinite points, meaning that if we wish to plot it numerically, we would need to generate all infinitely many points, which is not feasible. Instead, the common practice in such situations is to do a simple discretisation. Assume that we are interested in a small region of the real line, say $[a, b]$ with $a < b$, we will break the interval down into smaller equal-length pieces and use the endpoints of those pieces to approximate the full trajectories. To be more precise, if we wish to break the interval down into n pieces, we would have

$$a = x_1 < x_2 < \cdots < x_n = b$$

where $x_{i+1} - x_i = (b - a)/(n - 1)$ for $i = 1, 2, \ldots, n - 1$. Then, using the property of GP, the distribution of any finite points will follow a multivariate Gaussian distribution. If we denote the covariance matrix of the points $x = (x_1, x_2, \ldots, x_n)$ by K where $K_{ij} = k(x_i, x_j)$ and k is the covariance function of the GP, then we have

$$y(x) = \begin{bmatrix} y(x_1) \\ y(x_2) \\ \vdots \\ y(x_n) \end{bmatrix} \sim N_n(\mu(x), K)$$

so generating those points would be straightforward. The gaps between the points will be extrapolated by a straight line. We can also the variance of the multivariate Gaussian to draw a confidence region of the generated process.

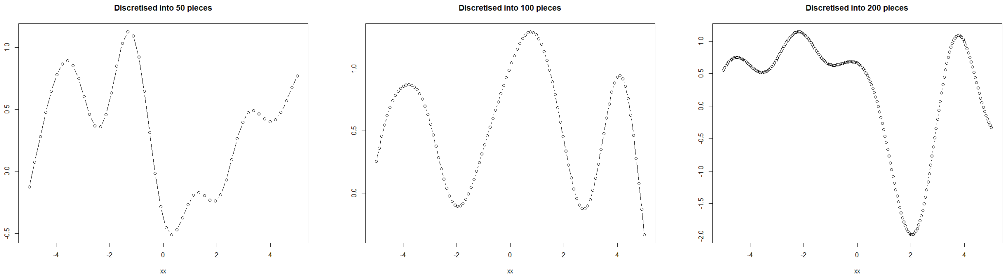

Figure 2: Gaussian Processes generated with different number of segments.

1.2 Gaussian Properties and Linear Algebra

In this section, we will look at several key properties of Gaussian random variables and Gaussian processes that will play a fundamental role in the rest of this note. Some derivations will be included.

Consider we have a multivariate Gaussian distribution $N_d(m, \Sigma)$ with mean vector m and covariance matrix Σ. The probability density function is given by

$$p(x; m, \Sigma) = (2\pi)^{-d/2}|\Sigma|^{-1/2} \exp\left[-\frac{1}{2}(x-m)^T\Sigma^{-1}(x-m)\right].$$

Next, consider two multivariate Gaussian $X \sim N_{d_1}(m_x, A)$ and $Y \sim N_{d_2}(m_y, B)$ where d_1 and d_2 may not be the same. The joint distribution of X and Y is given by

$$\begin{bmatrix} X \\ Y \end{bmatrix} \sim N_{d_1+d_2}\left(\begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} A & C \\ C^T & B \end{bmatrix}\right)$$

where C represents the covariance matrix between X and Y. If X and Y are independent, then C is zero. With this joint distribution, we would be interested in the marginal distributions and the conditional distributions. We have, for marginals,

$$\begin{aligned} p(x) &= \int p(x,y)dy, \qquad X \sim N_{d_1}(m_x, A) \\ p(y) &= \int p(x,y)dx, \qquad Y \sim N_{d_2}(m_y, B). \end{aligned}$$

For conditionals, we have

$$\begin{aligned} p(x|y) &= p(x,y)/p(y), \qquad X|Y=y \sim N(m_x + CB^{-1}(y-m_y), A - CB^{-1}C^T) \\ p(y|x) &= p(x,y)/p(x), \qquad Y|X=x \sim N(m_y + C^TA^{-1}(x-m_x), B - C^TA^{-1}C). \end{aligned}$$

The derivation of the marginal distributions is easy to see, which we omit. For conditionals, it is more complicated and we will establish it below. One identity that we will use is the formula for the inverse of the block matrix, which is easy to verify.

Proposition 1.1 (Inverse of Block Matrix)**.** *For block matrix*

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

where A, B, C, D are matrices on their own, the inverse

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} (A-BD^{-1}C)^{-1} & -(A-BD^{-1}C)^{-1}BD^{-1} \\ -D^{-1}C(A-BD^{-1}C)^{-1} & D^{-1} + D^{-1}C(A-BD^{-1}C)^{-1}BD^{-1} \end{bmatrix}.$$

Using the above result, we can derive the conditional distribution. We will just derive the case of $X|Y = y$, and the distribution of $Y|X = x$ is similar.

Proposition 1.2. *For joint distribution of X_1 and X_2 where X_1 is of dimension n_1, X_2 is of dimension n_2 and the joint is of dimension $n = n_1 + n_2$, it is given by*

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim N(\mu, \Sigma) = N\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right),$$

we have the conditional distribution

$$X_1 | X_2 = x_2 \sim N(\mu_{1|2}, \Sigma_{1|2}).$$

where $\mu_{1|2} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(X_2 - \mu_2)$ and $\Sigma_{1|2} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$.

Proof. Using the definition of density, we have

$$\begin{aligned} p(x_1|x_2) = \frac{p(x_1, x_2)}{p(x_2)} &= \frac{(2\pi)^{-n/2}|\Sigma|^{-1/2}\exp\left[-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right]}{(2\pi)^{-n_2/2}|\Sigma_{22}|^{-1/2}\exp\left[-\frac{1}{2}(x_2-\mu_2)^T\Sigma_{22}^{-1}(x_2-\mu_2)\right]} \\ &= (2\pi)^{-n_1/2}|\Sigma|^{-1/2}|\Sigma_{22}|^{1/2}\exp\left[-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu) + \frac{1}{2}(x_2-\mu_2)^T\Sigma_{22}^{-1}(x_2-\mu_2)\right]. \end{aligned}$$

Following the inverse of block matrix formula of Proposition 1.1, we have the following if we focus just on the exponential

$$\begin{aligned} & -\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu) + \frac{1}{2}(x_2-\mu_2)^T\Sigma_{22}^{-1}(x_2-\mu_2) \\ = & -\frac{1}{2}(x-\mu)^T \\ & \begin{bmatrix} (\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1} & -(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1}\Sigma_{12}\Sigma_{22}^{-1} \\ -\Sigma_{22}\Sigma_{21}(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1} & \Sigma_{22}^{-1} + \Sigma_{22}^{-1}\Sigma_{21}(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1}\Sigma_{12}\Sigma_{22}^{-1} \end{bmatrix} \\ & (x-\mu) + \frac{1}{2}(x_2-\mu_2)^T\Sigma_{22}^{-1}(x_2-\mu_2) \\ = & -\frac{1}{2}(x_1-\mu_1)^T(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1}(x_1-\mu_1) \\ & + (x_1-\mu_1)^T(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1}\Sigma_{12}\Sigma_{22}^{-1}(x_2-\mu_2) \\ & - \frac{1}{2}(x_2-\mu_2)^T[\Sigma_{22}^{-1} + \Sigma_{22}^{-1}\Sigma_{21}(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1}\Sigma_{12}\Sigma_{22}^{-1}](x_2-\mu_2) \\ & + \frac{1}{2}(x_2-\mu_2)^T\Sigma_{22}^{-1}(x_2-\mu_2) \\ = & -\frac{1}{2}(x_1-\mu_1)^T(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1}(x_1-\mu_1) \\ & + (x_1-\mu_1)^T(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1}\Sigma_{12}\Sigma_{22}^{-1}(x_2-\mu_2) \\ & - \frac{1}{2}(x_2-\mu_2)^T\Sigma_{22}^{-1}\Sigma_{21}(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1}\Sigma_{12}\Sigma_{22}^{-1}(x_2-\mu_2). \end{aligned}$$

If we rearrange them, the above term would become

$$-\frac{1}{2}[x_1 - \mu_1 - \Sigma_{12}\Sigma_{22}^{-1}(x_2-\mu_2)]^T(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1}[x_1 - \mu_1 - \Sigma_{12}\Sigma_{22}^{-1}(x_2-\mu_2)]$$

which yields the mean and covariance of the conditional distribution, as desired. □

Another nice property of Gaussian distributions is that a linear combination of Gaussian is still Gaussian. For example, for multivariate X, $AX + B$ is a multivariate distribution too where A is a matrix and B is a vector with the right dimension. This can be formally established using the moment-generating function, which we will omit.

Proposition 1.3. *For multivariate Gaussian $X \sim N(\mu, \Sigma)$, given matrix A and vector B with the right dimension, we have*

$$AX + B \sim N(A\mu + B, B\Sigma B^T).$$

A consequence of the above result is that the sum of two multivariate Gaussian with the same dimension is also a multivariate Gaussian.

Proposition 1.4. *For independent multivariate Gaussian $X \sim N(\mu_1, \Sigma_1)$ and $Y \sim N(\mu_2, \Sigma_2)$ with the same dimension, we have*

$$X + Y \sim N(\mu_1 + \mu_2, \Sigma_1 + \Sigma_2)$$

The above result can be obtained by first getting the joint distribution of X and Y, then considering the linear transformation with $A = I$ and $B = 0$.

Another result that we will state but not derive is the product of two Gaussian densities is proportionate to a Gaussian density.

Proposition 1.5. *Let $X \sim N(a, A)$ with density $p(x)$ and $Y \sim N(b, B)$ with density $p(y)$, we have*

$$p(x)p(y) \propto p(z)$$

where $p(z)$ is the density of $Z \sim N(c, C)$ and

$$C = (A^{-1} + B^{-1})^{-1}, \qquad c = C(A^{-1}a + B^{-1}b).$$

1.3 Gaussian Process Regression

The results mentioned in the previous section will play an important role in this section where we consider doing the task of **regressions** using Gaussian processes. In the field of **supervised learning** (where we have both data and their label, as opposed to **unsupervised learning** where we only have data and not labels), the two main tasks are regression and **classification**. They are essentially the same problem, but regression deals with continuous labels, and classification deals with discrete labels. One should also know that Gaussian process regression is used in many areas, and it is often called **kriging** in the spatial statistics literature.

1.3.1 Linear Regression

The problem of regression studies the relationship between covariates and the response using data. For example, in many scientific disciplines, we are interested in understanding the relationship between various factors (called **covariates**) and some key metric (called **response**), e.g. the relationship between biological measurements (such as weight, height, and blood pressure) and the hazard rate of a disease (such as diabetes). This has been heavily studied throughout the history of statistics and machine learning. One of the most basic types of regression, which is the topic here, is linear regression where we assume a linear relationship between the covariates and the response. We will also take a Bayesian approach.

If we denote the covariates by a vector x and the response by y (which we assume to be one-dimensional here, although it can be generalised), the linear regression assumes the following model:

$$f(x) = x^T w, \qquad y = f(x) + \varepsilon \tag{1}$$

where w is the weight vector for each of the covariates, $f(x)$ is the (latent) true function representing the relationship between x and y, while ε represents the noise vector, and we assume it is a Gaussian random variable here with mean zero and variance σ_n^2, i.e. $\varepsilon \sim N(0, \sigma_n^2)$ where n is the size of the data set. Normally, one of the covariates in x will represent the bias of the model, and that element in x is usually put as 1.

The Equation (1) represents the general formula, and for each data point, indexed by i, with response y_i and covariates x_i. The likelihood of the model for observing a data point (x_i, y_i) is then

$$p(y_i|w, x_i) = \frac{1}{\sqrt{2\pi}\sigma_n} \exp\left[-\frac{(y_i - x_i^T w)^2}{2\sigma_n^2}\right] \tag{2}$$

due to the Gaussianity of the noise ε.

We assume the variance of the noise is known, which can usually be estimated using an empirical variance estimator. This assumption means that the only unknowns of our linear regression problem are the values of the weight vector w, which we aim to infer using data. In the Frequentist setting, the likelihood described in Equation (2) is all we need - we just multiply the likelihoods for all the data points, and find the w values that maximise the multiplied quantity. In the Bayesian setting, we need more than that. First, we need to impose a prior distribution to the weight vector w, and then compute the posterior by multiplying the prior and the likelihood, then normalise, using the Bayes formula. For **prior** of weight vector w, we will use

$$w \sim N(0, \Sigma_p). \tag{3}$$

Some further comments on the prior choices will be stated later on.

Combining the prior of Equation (3) and the likelihood of Equation (2), we have the **posterior** distribution as follows using Proposition 1.5

$$\begin{aligned} p(w|x, y) &\propto p(w)p(y|w, x) \\ &\propto \exp\left[-\frac{1}{2\sigma_n^2}(y - x^T w)^T(y - x^T w)\right] \exp\left[-\frac{1}{2}w^T\Sigma_p^{-1}w\right] \\ &\propto \exp\left[-\frac{1}{2}(w - \bar{w})^T\left(\frac{1}{\sigma_n^2}xx^T - \Sigma_p^{-1}\right)(w - \bar{w})\right] \end{aligned}$$

where $\bar{w} = \sigma_n^{-2}(\sigma_n^{-2}xx^T + \Sigma_p^{-1})^{-1}xy$, making the posterior the density of

$$w|x, y \sim N(\bar{w}, A^{-1}), \qquad A := \sigma_n^{-2}xx^T + \Sigma_p^{-1} \tag{4}$$

which is a very nice conjugacy. An estimator for the weights given to the data will be some summary statistics of the posterior distribution, such as the mean or the mode. In this case, due to the geometry of the Gaussian distribution, the mean and the mode are identical and are identical to the maximum likelihood estimator when we formulate the problem as a **ridge regression** in the frequentist literature. If we use a different prior for w, such as the slab-and-spike prior, we could recover the **lasso regression** maximum likelihood estimator.

If we are fed with a new data point without the response to the model, we are essentially hoping to make a prediction using the weight vector $w|x, y$ we have inferred and the observed covariates x_*. The **prediction** distribution f_* will have the distribution

$$p(f_*|x_*, x, y) = \int p(f_*|x_*, w)p(w|x, y)dw, \qquad f_* \sim N\left(\frac{1}{\sigma_n^2}x_*^T A^{-1}xy, x_*^T A^{-1}x_*\right) \tag{5}$$

using again the result of Proposition 1.5.

In general, there are two main goals of regressions: (1) understanding the relationship between covariates and response, (2) predicting the response for new data points. The first point is to make analyses and comments based on our updated knowledge of weights w, while the second point is to exploit the predictive distribution of Equation (5). In our case, the relationship between the covariates and the response is forced to be linear by construction, which is like ‘fitting a straight line’. In general, as we expand the class of functions for the possible relationships $f(x)$ (such as in the case of GLM), we would be ‘fitting a curve’. In the next part, we will study how one can do curve-fitting using GPs.

1.3.2 Regression using GPs

Linear regression is fitting a linear function $f(x) = x^T w$ using the observed covariates and the response variable. A Gaussian process, as we have described in Section 1.1, can be used to approximate a lot of functions. This inspires us to use a GP to model the relationship $f(x)$. An important assumption that we will make here is to assume the noise ε is always Gaussian, which allows us to leverage the nice conjugacy of Gaussian random variables. This will become clear soon.

First, we will consider the simple problem of regression with exact observation, meaning that the observed data points $(x_i, y_i)_i$ contain no noise, so we get what we see. In this case, we will use x to denote all the observed covariates and f to denote all the observed responses. In this case, extending what we have described at the end of Section 1.1, we can have the joint distribution of f and f^* where f^* is the points that we use to approximate the GP trajectory discretised at points x^* as

$$\begin{bmatrix} f \\ f^* \end{bmatrix} \sim N\left(0, \begin{bmatrix} K(x,x) & K(x,x^*) \\ K(x^*,x) & K(x^*,x^*) \end{bmatrix}\right)$$

where K represents the covariance matrix constructed by measuring the covariance between any two points using the covariance function k of the GP. As our observations do not have noise, the covariance matrix is exactly as it is. Consequently, using Proposition 1.2, we know that

$$f^*|x, x^*, f \sim N(K(x^*,x)K(x,x)^{-1}f, K(x^*,x^*) - K(x^*,x)K(x,x)^{-1}K(x,x^*)$$

which essentially collapses the process at the observed points x, f due to the absence of noise in observations. This gives us the regression curve after observing the data (x, f).

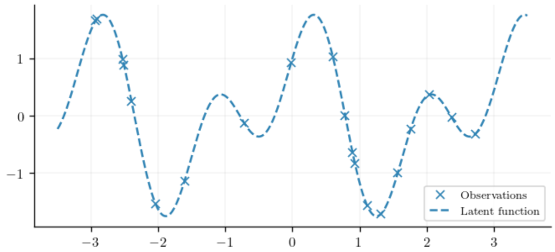

Figure 3: Latent Function of Gaussian Process Regression with 20 data points and no noise, using GPJax package of Pinder and Dodd (2022).

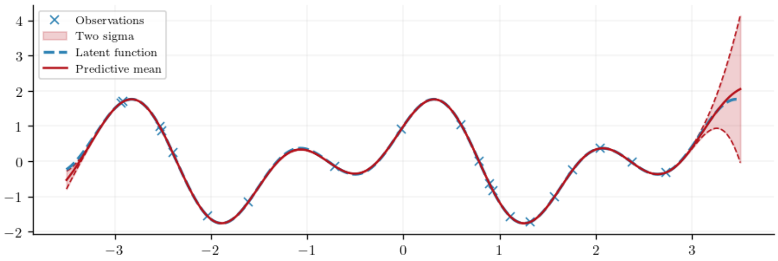

Figure 4: Posterior of Gaussian Process Regression with 20 data points and no noise, using GPJax package of Pinder and Dodd (2022).

A straight-forward and probably necessary extension is to consider the case where we make not exact but noisy observations where the noise is

$$\varepsilon \sim N(0, \sigma_n^2).$$

In this case, if we observe (x_p, y_p) and (x_q, y_q), their covariance will be

$$\mathrm{Cov}(y_p, y_q) = k(x_p, x_q) + \sigma_n^2 \delta_{pq}$$

where $\delta_{ab} = 1$ when $a = b$ and is zero otherwise. We can then generalise it to get

$$\mathrm{Cov}(y) = K(x, x) + \sigma_n^2 I$$

and

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

This then leads to the following conditional distribution, which is the regression curve after the observations

$$\begin{aligned} f^*|x, x^*, f &\sim N(\bar{f}^*, \mathrm{Cov}(f^*)) \\ \bar{f}^* &:= K(x^*, x)[K(x,x) + \sigma_n^2 I]^{-1} f, \\ \mathrm{Cov}(f^*) &:= K(x^*, x^*) - K(x^*, x)[K(x,x) + \sigma_n^2 I]^{-1} K(x, x^*). \end{aligned}$$

Another quantity of interest is the **marginal likelihood** of observing the data $p(y|x)$ given the model. Notice that here we are not specifying the dependencies of parameters in any of our expressions explicitly, we can certainly imagine the existence of some hyperparameters in our kernel function, as well as the variance of the observation noise being unknown. Knowing the marginal likelihood $p(y|x)$ is helpful for estimating these quantities of interest, and an expression for the marginal likelihood is

$$\begin{aligned} f|X \sim N(0, K), &\quad y|f \sim N(f, \sigma_n^2 I) \\ p(y|x) &= \int p(y|x, f) p(f|y) df \\ \log p(y|x) &= -\frac{1}{2} y^T (K + \sigma_n^2 I)^{-1} y - \frac{1}{2} \log |K + \sigma_n^2 I| - \frac{n}{2} \log 2\pi \end{aligned} \tag{6}$$

when we set a Gaussian prior to $f|X$ as this leads to $y \sim N(0, K + \sigma_n^2 I)$.

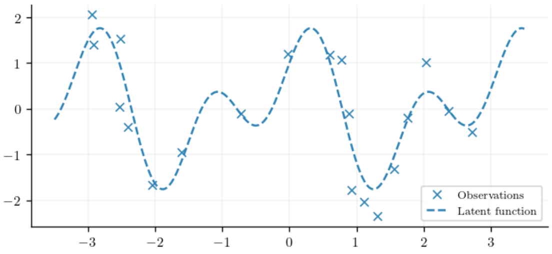

Figure 5: Latent Function of Gaussian Process Regression with 20 data points and noise, using GPJax package of Pinder and Dodd (2022).

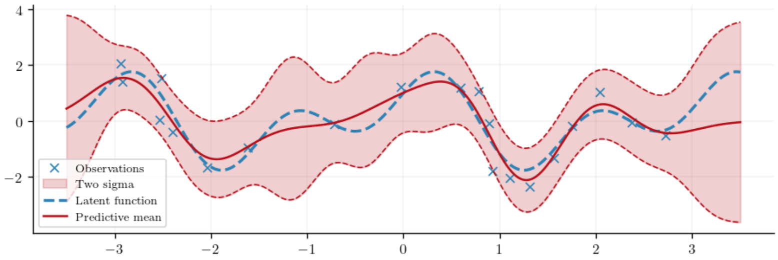

Figure 6: Posterior of Gaussian Process Regression with 20 data points and noise, using GPJax package of Pinder and Dodd (2022).

1.4 Covariance Function

In this section, we will explore more concretely the covariance functions of a Gaussian process. The two degrees of freedom of a Gaussian process are the mean function and the covariance function, and we often set the mean function to be zero, so the only real variability of a GP is the covariance function. Different choices of the covariance function will certainly lead to very different GPs. In this section, we will first outline some general properties and definitions related to covariance functions, then study a few commonly used covariance functions in detail.

1.4.1 Definitions and General Properties

As mentioned in Section 1.1, a GP $y(\cdot)$ is a stochastic process with the mean function $\mu(\cdot)$ and the covariance function $k(\cdot, \cdot)$, such that any finite points of the GP will form a multivariate Gaussian distribution. Because of this requirement, the covariance function needs to be (1) **symmetric**, i.e. $k(a, b) = k(b, a)$ (2) **positive semi-definite**, i.e.

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

for all $f \in L_2(X, \mu)$ where μ is some base measure and X is the support of the GP. Such a covariance function k will also be called a **kernel**, due to its link with the theory of integral operators.

For a set of points $x = \{x_i\}_{i=1}^n$, we can compute its **Gram matrix** K using the kernel k such that $K \in \mathbb{R}^{n\times n}$ and $K_{ij} = k(x_i, x_j)$. The Gram matrix in the context of GP will be used as the covariance matrix for the joint distribution of the points $x = \{x_i\}_i$.

One can view the kernel as a way to measure the similarity between two points. Since we would wish two points x, x' close to each other to be very similar - so highly dependent - in order to achieve some degrees of smoothness and the regularities of the overall GP, the quantities $x - x'$ and $\|x - x'\|$ would be of major importance. A covariance function that can be defined as a function of $x - x'$ is called **stationary (in the wide sense)**, or wide-sense stationarity (WSS), as it will be invariant to translations in the input space/support. A covariance function that can be defined as a function of $\|x - x'\|$ is called **isotropic** as it will be invariant under all rigid motions.

1.4.2 Examples of Covariance Functions

Two examples of covariance functions will be introduced here - the **squared exponential** (SE) covariance function and the **Matérn class** covariance function.

The **squared exponential** (SE) covariance function k_{SE} is defined by

$$k_{\text{SE}}(x, x') = \exp\left[-\frac{\|x - x'\|^2}{2l^2}\right] =: \exp\left[-\frac{r^2}{2l^2}\right] = k_{\text{SE}}(r)$$

where we define $r := \|x - x'\|$ and $l > 0$ is the **length-scale** of the kernel. From the definition, it is straightforward to notice that the SE kernel is stationary and isotropic, and the value of the length-scale characterises the degree of similarity between two nearby points - the higher the l, the more dependent two nearby points become. One should realise by the expression of k_{SE} that it is more of an exponentiated quadratic than a squared exponential, therefore some authors will denote the same kernel as the **exponentiated quadratic** kernel.

Since the SE kernel is defined by an exponential function, it is therefore infinitely differentiable (or smooth). This property will become useful when we discuss the differentiability of a GP in Section 1.6.

The **Matérn** class kernels is defined by

$$\begin{aligned} k_{\text{Matérn}}(x,x') &= \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}\|x-x'\|}{l}\right)^{\nu} K_\nu\left(\frac{\sqrt{2\nu}\|x-x'\|}{l}\right) \\ &=: \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}r}{l}\right)^{\nu} K_\nu\left(\frac{\sqrt{2\nu}}{l}r\right) = k_{\text{Matérn}}(r) \end{aligned}$$

where we define $r := \|x-x'\|$, $l > 0$ is the **length-scale** of the kernel, Γ is the Gamma function, $\nu > 0$ is the smoothness parameter of the kernel, and K_ν is the modified Bessel function of the second kind. The smoothness parameter ν usually is chosen to be half integers, as $\nu = p + 1/2$ for non-negative integer p. For example, we have the following three examples of ν:

$$\begin{aligned} k^{\nu=1/2}_{\text{Matérn}}(r) &= \exp\left(-\frac{r}{l}\right) \\ k^{\nu=3/2}_{\text{Matérn}}(r) &= \left[1 + \frac{\sqrt{3}r}{l}\right]\exp\left(-\frac{\sqrt{3}r}{l}\right) \\ k^{\nu=5/2}_{\text{Matérn}}(r) &= \left[1 + \frac{\sqrt{5}r}{l} + \frac{5r^2}{3l^2}\right]\exp\left(-\frac{\sqrt{5}r}{l}\right) \end{aligned}$$

and we can also show that using the definition, as $\nu \to \infty$, $k_{\text{Matérn}}(r) \to k_{\text{SE}}(r)$.

1.4.3 Composing Kernels

sum of kernels is a kernel. product of kernels is a kernel.

1.4.4 Hyperparameters and Model Selection

In Section 1.3, we have focused on how one can do curve fitting using GPs with a fixed, predetermined kernel. However, as we have seen in earlier parts of this section, there are multiple choices for kernel, and each kernel also depends on tuning hyperparameters that will influence the GP. Therefore, in practice, one should really consider the problem of selecting kernels and tuning hyperparameters as part of the curve-fitting process.

Model selection is a key and well-studied problem in statistics, especially for regression. ...

Assuming that we have fixed the choice of kernel, we then need to worry about how to tune the hyperparameters. We will treat the hyperparameters as additional parameters of the overall GP model while fitting the GP during regression. If one wishes to do regression using maximum likelihood, then it is quite straightforward - just pick the values for the hyperparameters (and the weight vector for covariates) that maximise the joint likelihood function using all the data. If one wishes to do regression using a Bayesian approach, then one would need to pose some prior on the weight vector, as well as the hyperparameters of the kernel, then compute the posterior distribution of all the parameters of interest and do the estimation using some summary statistics of the posterior, computed/estimated using conjugacy or Monte Carlo methods (such as MCMC).

1.5 Spectral Representation of Stationary Kernels

A very key result of stationary covariance functions is the Bochner Theorem, which states that all positive semi-definite functions have a unique spectral representation.

Theorem 1.6 (Bochner Theorem)**.** *A complex-valued function k on $\mathbb{R}^d$ is the covariance function of a wide-sense stationary, continuous in mean square complex-valued random process on $\mathbb{R}^d$ if and only if it can be represented as*

$$k(\tau) = \int_{\mathbb{R}^d} \exp[2\pi i s \cdot \tau] d\mu(s)$$

*where μ is a non-negative **finite** probability measure.*

A proof of the above result can be found in Section 1.4.3 of Rudin (2017). Note that the original result does not require k to be related to a stochastic process, and is more general to any positive semi-definite functions $k(\tau)$. The key message of the above theorem is the equivalence between the class of positive semi-definite functions and the class of their spectral representations.

If μ admits a density $S(s)$, then the above equation becomes

$$k(\tau) = \int_{\mathbb{R}^d} \exp[2\pi i s \cdot \tau] S(s) ds$$

which is the Fourier transform of k. This relationship inspires people to call the density S as the **spectral density** or the **power spectrum** for covariance function k. We can recover the covariance function from the spectral density using the inverse Fourier formula, and we get the following pairs of identities

$$k(\tau) = \int S(s) \exp[2\pi i s \cdot \tau] ds, \qquad S(s) = \int k(\tau) \exp[-2\pi i s \cdot \tau] d\tau.$$

This result is, in fact, a key result called the Wiener-Khintchine theorem, which is formally stated below.

Theorem 1.7 (Wiener-Khintchine Theorem)**.** *A real-valued function $k(\tau)$ on $\mathbb{R}^d$ is a covariance function if and only if it can be represented in the form*

$$k(\tau) = \int_{\mathbb{R}^d} \exp[2\pi i s \cdot \tau] dF(s)$$

where $F(s)$ is a distribution function on $\mathbb{R}^n$, and is often called the ***spectral distribution*** *function. When F admits a density S, we have*

$$k(\tau) = \int_{\mathbb{R}^d} \exp[2\pi i s \cdot \tau] S(s) ds.$$

Referring back to the Fourier transform, what we have been doing here is highlighting the transformation of the covariance function k in the time domain to the spectral density S in the frequency domain. A key advantage of this equivalence is that the requirement for S for the corresponding k to be positive semi-definite is merely the fact that S is non-negative for any s, which is much easier to verify.

Another attractive property of the spectral density is that we may be interested in understanding the characteristics of the stochastic process better in the frequency domain rather than the time domain.

1.5.1 Spectral Densities Examples

Here we will just state some examples of spectral densities of common kernels.

The squared exponential (SE) kernel on D dimension

$$k_{\text{SE}}(r) = \exp\left[-\frac{r^2}{2l}\right]$$

is stationary, and its spectral density is given by

$$S_{\text{SE}}(s) = (2\pi l^2)^{D/2} \exp(-2\pi^2 l^2 s^2).$$

The Matérn class kernel on D dimension

$$k_{\text{Matérn}}(r) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu} r}{l}\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}}{l} r\right)$$

has its spectral density as

$$S(s) = \frac{2^D \pi^{D/2} \Gamma(\nu + D/2)(2\nu)^{\nu}}{\Gamma(\nu) l^{2\nu}} \left(\frac{2\nu}{l^2} + 4\pi^2 s^2\right)^{-\nu - D/2}.$$

1.6 Differentiability of Gaussian Processes

A Gaussian process is a special case of a **random field**, which is any stochastic process supported on some domain, such as $\mathbb{R}^d$. An interesting mathematical question to ask is what is the meaning of continuous and differentiable in this context. As we can see in Figure 2, the GP realisations are very smooth, to the point that we can almost certainly call them smooth. However, those are only one realisation of the stochastic process. One way to do so is to define continuity and differentiability in the mean square. Consider a converging sequence $x_1, x_2, \ldots$ in the support of the random field that converges to x^*. A random field $f(x)$ is said to be **continuous in mean square** at x^* if

$$\mathbb{E}\|f(x_k) - f(x^*)\|^2 \to 0$$

as $k \to \infty$. If f is continuous in a mean square for all $x^* \in A$, then f is continuous in mean square over A. An equivalent way to check for the continuity in mean square of a random field f is to look at the continuity of its covariance function. Assuming that the mean function of f is continuous, then the random field is continuous in mean square at t if and only if its covariance function $k(s, s')$ is continuous at $s = s' = t$. Therefore, if the covariance function k is continuous at all diagonal points $s = s'$, the overall random field f will be continuous too (Adler, 2010). If the random field is stationary, so the covariance function only depends on $\tau = s - s'$, then f will be continuous if and only if $k(\tau)$ is continuous at $\tau = 0$.

Similarly, for f to be **differentiable in mean square** for the i-th coordinate at x, we must have

$$\frac{\partial f(x)}{\partial x_i} = \overline{\lim}_{h \to 0} \frac{f(x + he_i) - f(x)}{h} < \infty$$

where e_i is the unit vector in the i-th coordinate and $\overline{\lim}$ is the limit in mean square.

For a Gaussian process $y \sim GP(0, k)$ with scalar output and one-dimensional support (assumed for simplicity, can be generalised) to be differentiable in mean square, it suffices to show that

$$\mathbb{E}\left[\left\|\frac{y(x + h_1) - y(x)}{h_1} - \frac{y(x + h_2) - y(x)}{h_2}\right\|^2\right] \to 0$$

as $h_1, h_2 \to 0$ using the Cauchy criterion of convergence. We can see that, the quantity of interest can be decomposed into three terms by opening up the square. For example, we have

$$\mathbb{E}[[y(x + h_1) - y(x)][y(x + h_2) - y(x)]] = k(x + h_1, x + h_2) - k(x + h_1, x) - k(x, x + h_2) + k(x, x)$$

which becomes $\partial_x \partial_x k(x, x)$, if it exists, as $h_1, h_2 \to 0$ by definition. The same limit result holds for the two other terms of the square

$$\mathbb{E}[[y(x + h_1) - y(x)][y(x + h_1) - y(x)]], \quad \mathbb{E}[[y(x + h_2) - y(x)][y(x + h_2) - y(x)]].$$

Therefore, as long as $\partial_x \partial_y k(x, y)$ exists, the GP $y \sim GP(0, k)$ will be differentiable in mean square. As a corollary, if k is smooth (i.e. infinitely differentiable), then the GP is smooth too. Using a similar logic, as a corollary, we have that for $y \sim GP(m, k)$ such that both m, k are differentiable, we have

$$y' \sim GP(m', k')$$

and *the derivative of a GP is still a GP*.

Due to the linearity of the covariance function, we can also know that the covariance of derivatives is the derivatives of the covariance. For example, in the case of GP with differentiable kernels, we can have the following identities:

$$\mathrm{Cov}[\partial_x y(x_1), y(x_2)] = \partial_{x_1} \mathrm{Cov}[y(x_1), y(x_2)].$$

This will become very helpful when we are interested in vector fields and derivative observations.

Another interesting consequence of the above derivative properties is related to how the derivative of the time-domain (stationary) kernel is passed down to the frequency-domain. Consider we have a stationary kernel $k(\tau)$ in the time domain, and its frequency-domain counterpart $S(s)$ is provided by

$$k(\tau) = \frac{1}{2\pi} \int S(s) \exp[is \cdot \tau] ds, \qquad S(s) = \int k(\tau) \exp[-is \cdot \tau] d\tau.$$

where we do a slight change-of-variable to allow simple notations later on. It is not hard to see that the above formulation is equivalent to the formulation in Theorem 1.7. Since we have certain smoothness conditions on the functions that allow us to interchange derivatives and integrals, we have

$$\begin{aligned}
\partial^2_{tt} k(\tau) &= \partial^2_{tt} \frac{1}{2\pi} \int S(s) \exp[is \cdot \tau] ds \\
&= \frac{1}{2\pi} \int S(s) \partial^2_{tt} \exp[is \cdot \tau] ds \\
&= \frac{1}{2\pi} \int S(s)(is)^2 \exp[is \cdot \tau] ds \\
&= \frac{1}{2\pi} \int [-s^2 S(s)] \exp[is \cdot \tau] ds
\end{aligned}$$

which implies the frequency-domain counterpart of $\partial^2_{tt} k(\tau)$ is $-s^2 S(s)$.

Chapter 2

Gaussian Processes for Vector Fields

In Chapter 1, we have looked at Gaussian processes with scalar output. In this chapter, we will look at Gaussian processes with vector output, and in particular GPs with 2-dimensional vector field output.

One way to think about this problem is that we now have two output streams instead of one, so we could model two GPs (independently, perhaps) using the same training data, and produce a prediction for each of the two outputs separately. This does assume, however, the independence between the two outputs. A slightly improved way of doing such modelling is by doing a joint modelling of the two streams while imposing relatively weak and restrictive notions of dependency between the two streams of outputs. This will be discovered more in Section 2.2.

Another approach is to leverage some physical knowledge of the system and the vector field and do some smarter decomposition that enables us to express more types of dependency. One such approach is done via the **Helmholtz decomposition** of the vector field and will be discussed in Section 2.3. This approach, however, is more restrictive as it assumes the object of interest is a vector field of a physical system. We will showcase an example of modelling the ocean currents using a GP in Section 2.4.

The problem of modelling multiple outputs using GPs has a diverse range of applications. In some settings, we will call the built GPs **multi-task** or **multi-output** GPs, as each output stream represents one task that we are interested in. In such scenarios, we may have missing data issues (i.e. we know some of the outputs for some sets of data, while the other outputs are missing). This field of work is often called **transfer learning**, where we move our knowledge about one task to another (similar) task, via the dependency among tasks. This line of work will not be discussed here, and interested readers are referred to Bonilla et al. (2007).

2.1 Basics of Vector-Output GP

This section will look at the notations and basics of regression with vector-output GP. Apart from some tedious notations and more involved manipulations with matrices, it is the same as scalar-output GP regression discussed in Section 1.3. The notations are based on Alvarez et al. (2012).

We will still have the regression setup of

$$y = f(x) + \varepsilon$$

where x is the input, y is the output, and ε is the noise. The relationship f is to be modelled using a Gaussian process. In Chapter 1, we allow x to be a vector, but almost always state that

y needs to be a scalar quantity. We have also assumed that ε is a Gaussian noise. Here, we will relax the assumption on y and allow it to be a vector too.

A consequence of the relaxation is that we need more notations and more thoughts are needed to consider various cases. We will denote the dimension of the output y as D, so $y \in \mathbb{R}^D$. For a vector field output, $D = 2$. We also denote the dimension of the input for the d-th output as N_d, where $d = 1, 2, \ldots, D$. This allows the dimensionalities of input for different outputs to be different. This will not be necessary for the case of vector field modelling but could be relevant to more general multi-task GPs where some but not all data are shared among the tasks. Therefore, the data for each of the d output will be $S_d = \{(x_{d,i}, y_{d,i})\}_{i=1}^{N_d}$ and the full data set will be $S = \bigcup_{d=1}^{D} S_d$.

If N_d is the same for all d, and $x_{d,i}$ is the same for fixed i and varying d (so the input data for each stream/coordinate of the output is identical), then the overall model is called **isotopic**. The model is called **heterotopic** otherwise. In our case, the model is isotopic as the input data are the geographical location and the output data is the vector at that location of the vector field. Therefore, to simplify the notation, we will not be bothered with distinguishing between N_d for different d, but instead denote the shared dimension as N so the input becomes $\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\}$, the output becomes $\boldsymbol{y} = \{y_1, y_2, \ldots, y_D\}$ and the relationship function $\boldsymbol{f} = (f_1, f_2, \ldots, f_D)$ is a vector-valued function mapping from $\mathbb{R}^N$ to $\mathbb{R}^D$.

With this setup, the next thing we would like to know is how exactly we can model f using a GP, in particular, if we want to have

$$f \sim GP(m, k),$$

what should m and k be? For simplicity, we can still assume $m = 0$, although in this case m is no longer a scalar zero, but a D-vector zero. For kernel k, it is slightly more involved.

Here, k is no longer a positive semi-definite function with scalar output as in the case of Chapter 1, but rather a function with matrix output that is symmetric and positive semi-definite to some extent. Let X denote the input space, we have

$$k : X \times X \to \mathbb{R}^{D \times D}$$

and it needs to be symmetric (i.e. $k(x, x') = k(x', x)$ for any $x, x' \in X$) and for each x, x', the output matrix $k(x, x')$ needs to be positive semi-definite. Since the output is a matrix, we can define k entry-wise such that

$$k_{d,d'}(x, x') = [k(x, x')]_{d,d'} =: R\left((x, d), (x', d')\right)$$

for some scalar kernel R defined on the space $X \times \{1, 2, \ldots, D\}$, and the above quantity represents the covariance between output $f_d(x)$ and $f_{d'}(x')$. Furthermore, if we have a set of inputs $\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\}$, the kernel $k(\boldsymbol{x}, \boldsymbol{x})$ will be a matrix of matrices

$$k(\boldsymbol{x}, \boldsymbol{x}) = \begin{bmatrix} k_{1,1}(\boldsymbol{x}, \boldsymbol{x}) & k_{1,2}(\boldsymbol{x}, \boldsymbol{x}) & \ldots & k_{1,D}(\boldsymbol{x}, \boldsymbol{x}) \\ k_{2,1}(\boldsymbol{x}, \boldsymbol{x}) & k_{2,2}(\boldsymbol{x}, \boldsymbol{x}) & \ldots & k_{2,D}(\boldsymbol{x}, \boldsymbol{x}) \\ \vdots & \vdots & \ldots & \vdots \\ k_{D,1}(\boldsymbol{x}, \boldsymbol{x}) & k_{D,2}(\boldsymbol{x}, \boldsymbol{x}) & \ldots & k_{D,D}(\boldsymbol{x}, \boldsymbol{x}) \end{bmatrix} \in \mathbb{R}^{ND \times ND}$$

where each $k_{i,j}(\boldsymbol{x}, \boldsymbol{x})$ is a $\mathbb{R}^{N \times N}$ matrix on its own.

With these ready, we can write down the distributions for GP regression with vector output. For simplicity, we will only consider the case where the observations are noiseless and exact. The case of noisy observations can be obtained similarly by adding some additive noise matrices at the right places, not too different from the case in Section 1.3.

Given a set of inputs $\boldsymbol{x}$, the GP $f \sim GP(0, k)$ is given by

$$f(\boldsymbol{x}) \sim N(0, k(\boldsymbol{x}, \boldsymbol{x})).$$

Assuming we have a set of corresponding outputs $\boldsymbol{y}$ and we assume the variance of the noises for the d-th coordinate of the output (same noise variance for each input across outputs) is σ_d^2 and let $\Sigma \in \mathbb{R}^{D\times D}$ denote the diagonal matrix with σ_d^2 at d, d-th entry. This gives us the likelihood

$$\boldsymbol{y}|\boldsymbol{f}, \boldsymbol{x}, \Sigma \sim N(f(x), \Sigma)$$

and the predictive distribution for a new set of inputs $\boldsymbol{x}_*$ becomes

$$f(\boldsymbol{x}_*)|\boldsymbol{f}, \boldsymbol{x}_*, \boldsymbol{x}, \boldsymbol{y}, \phi \sim N(\boldsymbol{f}_*(\boldsymbol{x}_*), k_*(\boldsymbol{x}_*, \boldsymbol{x}_*))$$

where

$$\begin{aligned}
\boldsymbol{f}_*(\boldsymbol{x}_*) &= \boldsymbol{k}_{\boldsymbol{x}_*}(k(\boldsymbol{x}, \boldsymbol{x}) + \boldsymbol{\Sigma})^{-1}\overline{\boldsymbol{y}} \\
k_*(\boldsymbol{x}_*, \boldsymbol{x}_*) &= k(\boldsymbol{x}_*, \boldsymbol{x}_*) - \boldsymbol{k}_{\boldsymbol{x}_*}(k(\boldsymbol{x}, \boldsymbol{x}) + \boldsymbol{\Sigma})^{-1}\boldsymbol{k}_{\boldsymbol{x}_*}^T \\
\boldsymbol{\Sigma} &= \Sigma \otimes I_N
\end{aligned}$$

with $\overline{\boldsymbol{y}} \in \mathbb{R}^{ND}$ being the concatenation of D-vectors y for each of the N inputs, $\boldsymbol{k}_{\boldsymbol{x}_*} \in \mathbb{R}^{D\times ND}$ being the concatenation of $D \times D$ matrix $k(\boldsymbol{x}_*, x_j)$ where $j = 1, 2, \ldots, N$, and ϕ is the hyperparameter choices of the kernel k. The operation $\otimes$ is the Kronecker product, and more information is provided in Section A.1. The above equations are direct extensions to the regression outputs of Section 1.3, and can also be found in Alvarez et al. (2012).

2.2 Velocity Decomposition

In this section, we will look at a direct decomposition of multi-output GP by considering each coordinate of the output separately, with potentially some correlation among coordinates. The methods outlined in this section could be applied to more general multi-output GP problems, although the focus here will be on the particular case of modelling a vector field using GPs.

A class of matrix output kernels is the **separable** kernels, and it is the class that we will pick our kernels from in this section. Consider a matrix kernel k, it is **separable** if we can write it in the form of

$$(k(x, x'))_{d,d'} = k_S(x, x')k_T(d, d') \qquad\qquad k = S \otimes T$$

where $S \in \mathbb{R}^{N\times N}, T \in \mathbb{R}^{D\times D}$ are symmetric, positive semi-definite matrices. The reason why such kernels are called separable is obvious, as we can separate the contribution from inputs and outputs in the kernel value. Since the sum of kernels is also a kernel, the sum of separable kernels is also a kernel. In the case where $T = I_D$, we are treating the D streams of output independently, as there is no correlation assumed between them.

Building separable kernels is then easy, as we can pick a kernel for input and output separately, and combine them using the Kronecker product. We could also add up a few independent separable kernels to get more expressive **sum of separable kernels**. This is known as the **linear model of coregionalisation** in the geostatistics literature (Journel and Huijbregts, 1978).

Consider a two-dimensional vector field that is sufficiently regular (at least twice continuously differentiable), and we denote it as F, so $F : \mathbb{R}^2 \to \mathbb{R}^2$. We further denote the component of F as u and v so $F(x) = (u(x), v(x))^T$.

In the case of velocity decomposition, we will model u, v as two Gaussian processes with mean zero and kernel k_u, k_v respectively, and the two processes are independent. So, the velocity kernel becomes

$$k_{\text{vel}}(x, x') = \begin{bmatrix} k_u(x, x') & 0 \\ 0 & k_v(x, x') \end{bmatrix}.$$

Then, k_u, k_v can be picked separately. Naturally, we can see that this decomposition ignores any potential dependency between the data for two directions of the vector field.

2.3 Helmholtz Decomposition

Consider a two-dimensional vector field that is sufficiently regular (at least twice continuously differentiable), and we denote it as F, so $F : \mathbb{R}^2 \to \mathbb{R}^2$. We further denote the component of F as u and v so $F(x) = (u(x), v(x))^T$.

According to the Helmholtz decomposition in 2D (see Definition A.3 for more information), we can find a pair of scalar functions Φ, Ψ such that

$$F = \begin{bmatrix} u \\ v \end{bmatrix} = \text{grad } \Phi + \text{rot } \Psi.$$

In the case of Helmholtz decomposition of Berlinghieri et al. (2023), we will give a kernel to each of Φ, Ψ, denoted by k_Φ, k_Ψ respectively.

Obviously, if we let u, v be Gaussian processes, the joint vector field is also a Gaussian process and the joint distribution of Gaussian is still Gaussian. Here, as we are letting Φ, Ψ be Gaussian processes, will the vector field F be a Gaussian process? The answer is yes, and it is explained in the following result.

Proposition 2.1 (Prop 3.1 of Berlinghieri et al. (2023))**.** *Consider a twice continuously differentiable vector field $F : \mathbb{R}^2 \to \mathbb{R}^2$ with Helmholtz decomposition $F = \text{grad } \Phi + \text{rot } \Psi$. If we have, independently,*

$$\Phi \sim GP(0, k_\Phi), \quad \Psi \sim GP(0, k_\Psi)$$

where k_Φ, k_Ψ are kernels such that Φ, Ψ have sample paths that are continuously differentiable almost surely, then

$$F \sim GP(0, k_{Helm})$$

where for $x, x' \in \mathbb{R}^2$, we have

$$k_{Helm}(x, x') = \begin{bmatrix} \partial^2_{x_1 x_1'} k_\Phi(x, x') - \partial^2_{x_2 x_2'} k_\Psi(x, x') & \partial^2_{x_1 x_2'} k_\Phi(x, x') + \partial^2_{x_2 x_1'} k_\Psi(x, x') \\ \partial^2_{x_2 x_1'} k_\Phi(x, x') + \partial^2_{x_1 x_2'} k_\Psi(x, x') & \partial^2_{x_2 x_2'} k_\Phi(x, x') - \partial^2_{x_1 x_1'} k_\Psi(x, x') \end{bmatrix}.$$

Proof. According to the result in Section 1.6, we know that

$$\text{Cov}[\partial_x k, \partial_y k] = \partial^2_{xy} \text{Cov}[k, k]$$

since Cov is linear. Therefore, denoting $x = (x_1, x_2)^T$ and $x' = (x_1', x_2')^T$, we have

$$\begin{aligned} \text{Cov}[\text{grad } \Phi(x), \text{grad } \Phi(x')] &= \text{Cov}\left[\begin{bmatrix} \partial_{x_1} k_\Phi(x) \\ \partial_{x_2} k_\Phi(x) \end{bmatrix}, \begin{bmatrix} \partial_{x_1'} k_\Phi(x') \\ \partial_{x_2'} k_\Phi(x') \end{bmatrix}\right] \\ &= \begin{bmatrix} \partial^2_{x_1 x_1'} k_\Phi(x, x') & \partial^2_{x_1 x_2'} k_\Phi(x, x') \\ \partial^2_{x_2 x_1'} k_\Phi(x, x') & \partial^2_{x_2 x_2'} k_\Phi(x, x') \end{bmatrix} \end{aligned}$$

and

$$\begin{aligned} \text{Cov}[\text{rot } \Psi(x), \text{rot } \Psi(x')] &= \text{Cov}\left[\begin{bmatrix} -\partial_{x_2} k_\Psi(x) \\ \partial_{x_1} k_\Psi(x) \end{bmatrix}, \begin{bmatrix} -\partial_{x_2'} k_\Psi(x') \\ \partial_{x_1'} k_\Psi(x') \end{bmatrix}\right] \\ &= \begin{bmatrix} -\partial^2_{x_2 x_2'} k_\Psi(x, x') & \partial^2_{x_2 x_1'} k_\Psi(x, x') \\ \partial^2_{x_1 x_2'} k_\Psi(x, x') & -\partial^2_{x_1 x_1'} k_\Psi(x, x') \end{bmatrix} \end{aligned}$$

which sums up to the desired covariance function, as desired. □

Another interesting derivation is on the spectral densities of the kernels, which can be obtained easily using the properties derived at the end of Section 1.6 on how the frequency-domain function is changed by differentiations in the time-domain function.

Proposition 2.2. *Using the same setup as Proposition 2.1, we further assume that the kernels k_Φ, k_Ψ are stationary, so S_Φ, S_Ψ are their frequency-domain counterparts following Theorem 1.7. Assuming S_Φ, S_Ψ are twice continuously differentiable, we have*

$$S_{Helm}(\omega, \omega') = \begin{bmatrix} -\omega_1\omega_1' S_\Phi(\omega,\omega') + \omega_2\omega_2' S_\Psi(\omega,\omega') & -\omega_1\omega_2' S_\Phi(\omega,\omega') - \omega_2\omega_1' S_\Psi(\omega,\omega') \\ -\omega_2\omega_1' S_\Phi(\omega,\omega') - \omega_1\omega_2' S_\Psi(\omega,\omega') & -\omega_2\omega_2' S_\Phi(\omega,\omega') + \omega_1\omega_1' S_\Psi(\omega,\omega') \end{bmatrix}.$$

Notice that the setup in Proposition 2.1 and Proposition 2.2 assumes independence between Φ and Ψ, which can be changed. If correlations between the two functions are included, the Helmholtz kernel will just have a few more terms in its entries, and one could obtain those terms easily by following the computations in the proofs. Also, detailed expressions of the terms are listed as Equations (4-9) of Ponte et al. (2024) using a slightly different definition of the curl operator.

2.4 Modelling Ocean Currents

In this section, we will consider a toy example of the ocean currents model using the gulf (of Mexico) drifters open dataset of Lilly and Pérez-Brunius (2021), and apply the two GP decompositions of Section 2.2 and Section 2.3 to compare their performance.

The Gulf Drifters data set provides the ground truth of ocean currents at 34 × 16 grid points, equally spaced over the longitude-latitude region of $[-90.8, -83.8] \times [24.0 \times 27.5]$. The ground truth of ocean currents is computed by the average velocity of the velocities within the grid.

In addition to the ground truth, the data set also provides training data, which are the velocity observations obtained by various drifters placed in that region. One should note that the temporal factor of the model and the data is ignored in this toy example, and should be considered in practice.

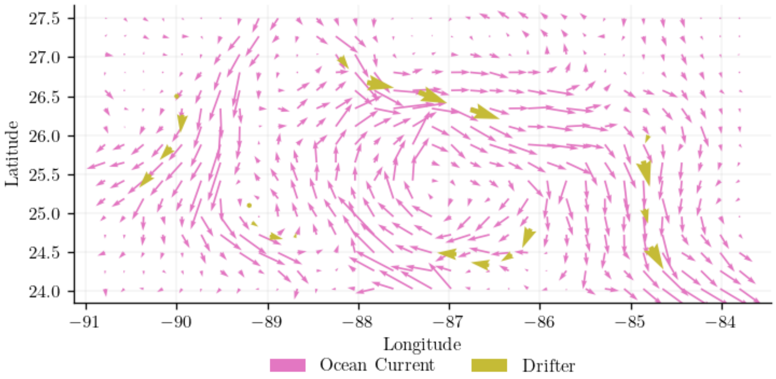

Figure 7: Ground Truth of the Current and Drifter Trajectory of the Gulf Drifters Dataset of Lilly and Pérez-Brunius (2021).

Next, we use both the velocity and Helmholtz decomposition as the GP model to fit the drifter data. The kernel of choice for all these two decompositions is always the squared exponential kernel, and the hyperparameters are fitted using the data as well. This is just to simplify the codes, and get a rough comparison. A more complicated, and more realistic kernel such as the Matérn kernel can be used too. The velocity decomposition fit is listed in Figure 8, and the Helmholtz decomposition fit is listed in Figure 9.

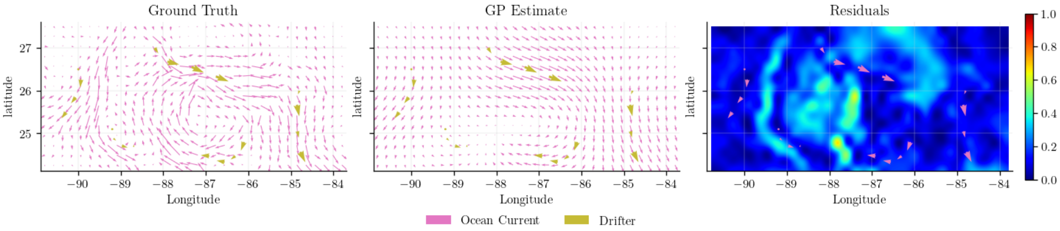

Figure 8: Vector Field Estimation using the Velocity Decomposition with SE kernels.

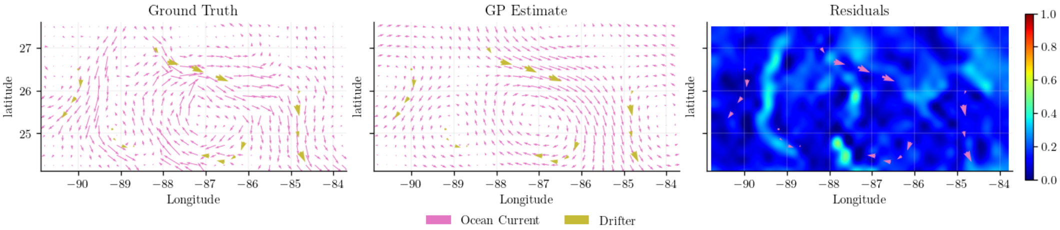

Figure 9: Vector Field Estimation using the Helmholtz Decomposition with SE kernels.

The right-most plots of both figures are the residuals between the fitted vector field and the ground truth. One can notice from the residual plots that the Helmholtz decomposition provides a much better fit, especially at regions where there is little data - such as the bottom two grids of the $[-88, -87]$ longitude strip. This is due to the fact that the Helmholtz decomposition better captures the physics of the system by imposing GP priors on Φ, Ψ, which allows more accurate extrapolations of the full vector field.

Chapter 3

Building Kernel Spectra using Mixtures

Due to the Bochner Theorem of Theorem 1.6, the covariance function k of a stationary Gaussian process admits a Fourier transform, known as the spectral density S, which is a positive finite measure on the frequency domain. The equivalence of k and S can also be established using the Wiener-Khintchine Theorem of Theorem 1.7, allowing us to easily build the kernel in one space and move to another. In this chapter, we will investigate existing methods of building kernels in the frequency domain, i.e. the spectral density, using a mixture of simple distributions. Both the scalar-output GPs and vector-output GPs will be considered, as they impose some challenges to this spectral mixture approach.

Here, we will use the term '**spectral mixture**' kernels to refer to all kernels built using a mixture distribution at the spectrum. The respective names of each spectral mixture will be denoted descriptively.

3.1 Spectral Mixture

In this section, we will look at the various ways one could build a stationary kernel with scalar output using a mixture distribution for the spectrum. The two mixture ingredients are Gaussians and blocks.

Recall that the Bochner Theorem of Theorem 1.6 states that, a complex-valued function k on $\mathbb{R}^d$ that is positive (semi-)definite if and only if it admits the Fourier transformation

$$k(\tau) = \int_{\mathbb{R}^d} \exp[2\pi i s^T \tau] \mu(ds)$$

where μ is a **non-negative finite** probability measure. This result draws the **equivalence** of the class of positive (semi-)definite functions and the class of their corresponding spectral representations.

Note that the class of covariance functions of weakly stationary, mean square continuous, complex-valued stochastic processes on $\mathbb{R}^d$ is equivalent to the class of positive (semi-)definite functions on $\mathbb{R}^d$, we can therefore draw the further equivalence of the kernel of such a stochastic process admits a spectral representation μ. If the non-negative finite probability measure μ further admits a density S, then we can have the Wiener-Khintchine Theorem of Theorem 1.7, which states that the covariance function k has the Fourier dual S

$$k(\tau) = \int_{\mathbb{R}^d} S(s) \exp[2\pi i s^T \tau] ds, \qquad S(s) = \int_{\mathbb{R}^d} k(\tau) \exp[-2\pi i s^T \tau] d\tau.$$

Here, S is denoted as the **(power) spectral density** of the covariance function k, and the goal of this chapter is to figure out how to set the spectral density S as a mixture distribution, and its consequences to the modelling power of the Gaussian process with that kernel.

One key property of the spectral density of a real-valued, stationary stochastic process is that it is **symmetric**, i.e. $S(s) = S(-s)$ for all s. To see this, we first recall that a kernel is symmetric, i.e. $k(x, x') = k(x', x)$. So, for stationary kernel $k(x, x') = k(\tau)$ with $\tau = x - x'$, we have $k(x, x') = k(\tau) = k(-\tau) = k(x', x)$. Next, using this result, we have, for stationary kernels supported on $\mathbb{R}$ (for simplicity only, can be easily generalised to higher dimensions)

$$\begin{aligned}
S(-s) &= \int_{-\infty}^{\infty} k(\tau) \exp[-2\pi i(-s)^T \tau] d\tau \\
&= \int_{\infty}^{-\infty} k(-\tau) \exp[-2\pi i(-s)^T (-\tau)] d(-\tau) \\
&= -\int_{\infty}^{-\infty} k(\tau) \exp[-2\pi i s^T \tau] d\tau \\
&= \int_{-\infty}^{\infty} k(\tau) \exp[-2\pi i s^T \tau] d\tau = S(s)
\end{aligned}$$

as desired.

3.1.1 Gaussian Mixtures

Here, we are going to introduce the Gaussian spectral mixture kernel of Wilson and Adams (2013).

Consider a one-dimensional Gaussian random variable $N(\mu, \sigma^2)$, its density is given by

$$g(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$$

where the mean and the variance of the random variable are included explicitly above. We can then consider the following symmetric mixture

$$S(s) = \frac{1}{2} g(s; \mu, \sigma^2) + \frac{1}{2} g(-s; \mu, \sigma^2)$$

which is symmetric by construction.

Given the context, it is not hard to imagine that the next step is to find the stationary kernel corresponding to S. First, S is clearly non-negative and finite by construction, so the stationary kernel that we are looking for does exist. Then, we will do the Fourier transform. Since Fourier transformation is a linear operator, we can consider it term by term. We have

$$\begin{aligned}
f_1(\tau) &= \int_{-\infty}^{\infty} g(s; \mu, \sigma^2) \exp[-2\pi i s \tau] ds \\
&= \int \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2\sigma^2}(s-\mu)^2\right] \exp[-2\pi i s \tau] ds \\
&= \frac{1}{\sqrt{2\pi}\sigma} \int \exp\left[-\frac{1}{2\sigma^2}(s-\mu)^2 - 2\pi i s \tau\right] ds \\
&= \frac{1}{\sqrt{2\pi}\sigma} \int \exp\left[-\frac{1}{2\sigma^2} s^2 + \frac{\mu}{\sigma^2} s - 2\pi i \tau s - \frac{\mu^2}{2\sigma^2}\right] ds.
\end{aligned}$$

We will try to complete the square for the insides of the exponential above with respect to s to allow nice integration tricks. We could have

$$\begin{aligned}
& -\frac{1}{2\sigma^2}s^2 + \frac{\mu}{\sigma^2}s - 2\pi i\tau s - \frac{\mu^2}{2\sigma^2} \\
&= -\frac{1}{2\sigma^2}\left[s^2 - \left(2\mu - 4\pi i\tau\sigma^2\right)s\right] - \frac{\mu^2}{2\sigma^2} \\
&= -\frac{1}{2\sigma^2}\left[(s - (\mu - 2\pi i\tau\sigma^2))^2 - \mu^2 + 4\pi i\tau\sigma^2\mu - 4\pi^2 i^2\tau^2\sigma^4\right] - \frac{\mu^2}{2\sigma^2} \\
&= -\frac{1}{2\sigma^2}\left[(s - (\mu - 2\pi i\tau\sigma^2))^2\right] + \frac{\mu^2}{2\sigma^2} - 2\pi i\tau\mu - 2\pi^2\tau^2\sigma^2 - \frac{\mu^2}{2\sigma^2} \\
&= -\frac{1}{2\sigma^2}\left[(s - (\mu - 2\pi i\tau\sigma^2))^2\right] - 2\pi i\tau\mu - 2\pi^2\tau^2\sigma^2.
\end{aligned}$$

Therefore, plugging the above reformulation back into the integral, we have

$$\begin{aligned}
f_1(\tau) &= \frac{1}{\sqrt{2\pi}\sigma}\int \exp\left[-\frac{1}{2\sigma^2}s^2 + \frac{\mu}{\sigma^2}s - 2\pi i\tau s - \frac{\mu^2}{2\sigma^2}\right] ds \\
&= \frac{1}{\sqrt{2\pi}\sigma}\int \exp\left[-\frac{1}{2\sigma^2}\left[(s - (\mu - 2\pi i\tau\sigma^2))^2\right] - 2\pi i\tau\mu - 2\pi^2\tau^2\sigma^2\right] ds \\
&= \exp\left[-2\pi i\tau\mu - 2\pi^2\tau^2\sigma^2\right]\frac{1}{\sqrt{2\pi}\sigma}\int \exp\left[-\frac{1}{2\sigma^2}(s - (\mu - 2\pi i\tau\sigma^2))^2\right] ds \\
&= \exp\left[-2\pi i\tau\mu - 2\pi^2\tau^2\sigma^2\right]
\end{aligned}$$

where the last step is by realising the integrated corresponds to the (unnormalised) density function of a Gaussian random variable. The Fourier transform of $g(-s;\mu,\sigma^2)$ can be derived similarly and easily by realising $g(-s;\mu,\sigma^2) = g(s;-\mu,\sigma^2)$. We will omit the derivations and just state the result below:

$$f_2(\tau) = \exp\left[2\pi i\tau\mu - 2\pi^2\tau^2\sigma^2\right]$$

Thus, we have the full Fourier transform of the symmetric Gaussian mixture $S(s)$:

$$\begin{aligned}
k(\tau) &= \int_{-\infty}^{\infty} S(s)\exp[-2\pi i s\tau] ds \\
&= \int_{-\infty}^{\infty}\left[\frac{1}{2}g(s;\mu,\sigma^2) + \frac{1}{2}g(-s;\mu,\sigma^2)\right]\exp[-2\pi i s\tau] ds \\
&= \frac{1}{2}f_1(\tau) + \frac{1}{2}f_2(\tau) \\
&= \exp[-2\pi^2\tau^2\sigma^2]\left[\frac{1}{2}\exp[-2\pi i\tau\mu] + \frac{1}{2}\exp[2\pi i\tau\mu]\right] \\
&= \exp[-2\pi^2\tau^2\sigma^2]\cos(2\pi\tau\mu).
\end{aligned}$$

where the last step uses the trigonometric identity.

To summarise the above derivations, we have the following proposition.

Proposition 3.1. *Consider a one-dimensional Gaussian random variable with mean μ and variance σ^2, where we denote its density as $g(\cdot;\mu,\sigma^2)$. The symmetric mixture S of g, defined by*

$$S(s) = \frac{1}{2}g(s;\mu,\sigma^2) + \frac{1}{2}g(-s;\mu,\sigma^2)$$

is the spectrum of a stationary kernel k, defined by

$$k(\tau) = \exp[-2\pi^2\tau^2\sigma^2]\cos(2\pi\tau\mu).$$

Due to the linearity of the Fourier transform, we can extend the above proposition to the case where more than one Gaussian is used to build the mixture.

Proposition 3.2. *Consider a sequence of one-dimensional Gaussian random variables with mean μ_q and variance σ_q^2 for $q = 1, 2, \ldots, Q$, where we denote each density as $g_q(\cdot; \mu_q, \sigma_q^2)$. The symmetric mixture S of $\{g_q\}_{q=1}^Q$ with normalised weights $\{A_q\}_{q=1}^Q$, defined by*

$$S(s) = \frac{1}{2} \sum_{q=1}^{Q} A_q \left[g_q(s; \mu_q, \sigma_q^2) + g_q(-s; \mu_q, \sigma_q^2) \right]$$

is the spectrum of a stationary kernel k, defined by

$$k(\tau) = \sum_{q=1}^{Q} A_q \exp[-2\pi^2 \tau^2 \sigma_q^2] \cos(2\pi\tau\mu_q).$$

We can also easily extend the above result to the case of general dimensional input space.

Proposition 3.3. *Consider a sequence of n-dimensional Gaussian random vector with mean vector μ_q and* ***diagonal*** *covariance matrix Σ_q where the d-th coordinate of that random vector has mean $\mu_q^{(d)}$ and variance $(\sigma_q^{(d)})^2$ for $q = 1, 2, \ldots, Q$, where we denote each density as $g_q(\cdot; \mu_q, \Sigma_q)$. The symmetric mixture S of $\{g_q\}_{q=1}^Q$ with normalised weights $\{A_q\}_{q=1}^Q$, defined by*

$$S(s) = \frac{1}{2} \sum_{q=1}^{Q} A_q \left[g_q(s; \mu_q, \Sigma_q) + g_q(-s; \mu_q, \Sigma_q) \right]$$

is the spectrum of a stationary kernel k, defined by

$$k(\tau) = \sum_{q=1}^{Q} A_q \prod_{d=1}^{n} \exp[-2\pi^2 \tau_d^2 (\sigma_q^{(d)})^2] \cos(2\pi\tau_d\mu_q^{(d)})$$

where τ_d denotes the distance in the d coordinate between the two compared points of the kernel.

A very desirable property of Gaussian mixtures is that mixtures of Gaussian distribution are dense in the set of probability distributions (w.r.t. the weak topology). This means, intuitively, that we can approximate any probability distributions to any arbitrary precision using a Gaussian mixture - although the number of Gaussians needed for such a mixture might be very large.

Proposition 3.4. *Mixtures of Gaussians are weak-* dense in the space of probability distributions.*

Sketch of Proof. First, we should convince ourselves that we can approximate any constant random variable (i.e. a random variable that takes a constant value C with probability 1) using Gaussian mixtures. Simply consider a sequence of Gaussians $\{X_n\}_n$ with mean C and variance $1/n$, and we could obtain a sufficiently good approximate of the constant random variable for large enough n.

Next, we should realise that any random variable can be approximated, in distribution, by a mixture of constants. This means that for any random variable X of interest, its distribution function can be approximated with arbitrary precision by a linear combination of step functions (note that the distribution function of a constant random variable is the Heaviside step function).

Combining the two, we can therefore convince ourselves that the desired statement is correct. □

One takeaway from the proof sketch above is that we are not limited to Gaussians as the building blocks for our mixtures. As long as we can show that the new building block allows us to construct a dense set in the space of probability measures, we would get the same theoretical justification for establishing spectral mixtures with those blocks. An alternative choice of the building blocks is the topic of the next subsection.

The main issue with such Gaussian spectral mixture kernels (and spectral mixture kernels in general) is that the number of components Q and the parameters $\{(\mu_q, \sigma_q^2)\}_{q=1}^Q$ could be hard to infer.

The parameters may not be identifiable, and a full inference for the number of components Q is a model selection problem which could be very computationally costly to investigate properly. In practice, people often pick a nice, easy value of Q arbitrarily, and estimate the parameters consequently. A more theoretically justified analysis of the model selection and parameter estimations of spectral mixture kernels is a direction of future work in this area.

3.1.2 Block Mixtures

The block mixture of Tobar (2019), in essence, replaces the Gaussian components of the Gaussian spectral mixture with a rectangle function $r(s)$ given by

$$r(s) = \begin{cases} 1 & |s| < 1/2 \\ 1/2 & |s| = 1/2 \\ 0 & \text{elsewhere.} \end{cases}$$

Note that first, this is (almost everywhere) a uniform distribution $\text{Unif}(-0.5, 0.5)$, which is a probability distribution. Next, we also realise that shifting the location and width of the rectangle function $r(s)$ is very straightforward: a rectangle function with mean ξ is $r(s-\xi)$, while a rectangle function with width Δ is $r(s/\Delta)/\Delta$. Finally, the Fourier transform of $r(s)$ is known from the signal processing literature, and we will derive it below.

For simplicity, we will do the Fourier transform to the basic rectangle function $r(\xi)$ with location 0 and width 1. Any shifting and scaling will impact the Fourier transform in a very simple way, thus they are omitted here.

We have

$$\begin{aligned} \int_{-\infty}^{\infty} r(s) \exp[-2\pi i s\tau] ds &= \int_{-1/2}^{1/2} \exp[-2\pi i s\tau] ds \\ &= \frac{1}{-2\pi i\tau} \left[\exp[-2\pi i s\tau]\right]_{-1/2}^{1/2} \\ &= \frac{-2i\sin(\pi\tau)}{-2\pi i\tau} =: \text{sinc}(\tau) \end{aligned}$$

where $\text{sinc}(x) := \sin(\pi x)/(\pi x)$ is the normalised sinc function, and consequently

$$\int_{-\infty}^{\infty} \frac{1}{\Delta} r\left(\frac{s-\xi}{\Delta}\right) \exp[-2\pi i s\tau] ds = e^{-2\pi i\tau\xi}\, \text{sinc}(\Delta\tau).$$

Therefore if we construct a symmetric mixture

$$S(s) = \frac{1}{2\Delta} r\left(\frac{s-\xi}{\Delta}\right) + \frac{1}{2\Delta} r\left(\frac{s+\xi}{\Delta}\right),$$

we would have

$$k(\tau) = \int_{-\infty}^{\infty} S(s) \exp[-2\pi i s\tau] ds = \frac{\text{sinc}(\Delta\tau)}{2}\left[e^{-2\pi i\tau\xi} + e^{2\pi i\tau\xi}\right] = \text{sinc}(\Delta\tau)\cos(2\pi\tau\xi).$$

The above kernel is called the **sinc kernel** in Tobar (2019). A direct property by construction of the sinc kernel is the support of its spectrum is bounded (and compact), meaning that, in signal processing terms, the kernel is band-limited, and only has a bounded range of frequencies, rather than the full range of frequencies in the case of Gaussian mixture kernels as well as most commonly used kernels like the SE kernels and Matérn kernels. This property could be beneficial on its own, as it could be a nice modelling assumption that fits certain applications. But even more importantly, this allows us to borrow a lot of existing results in the signal processing literature regarding Shannon sampling and interpolation theory (Marks, 1990) such as the Nyquist-Shannon sampling theorem and the Nyquist frequency.

The Nyquist-Shannon sampling theorem states that, for a band-limited (i.e. bounded spectrum support) signal with the **bandlimit** (i.e. largest frequency) being f_B, we can reconstruct the signal perfectly using the observations of the signals obtained at the **Nyquist frequency** $f_B/2$ by doing the **Shannon interpolation** given by

$$x(t) = \sum_{n=-\infty}^{\infty} x_n \operatorname{sinc}\left(\frac{t - nf_B}{f_B}\right)$$

where $x(t)$ is the reconstructed signal, $\{x_n\}$ is the observations at the Nyquist frequency, and f_B is the bandlimit (Marks, 1990). One caveat of this seemingly nice result is that the reconstruction requires infinite observations (at a certain frequency) which is not realistic.

In the case of GP, the introduction of sampling and interpolation theory could guide our intuition for picking inducing points (i.e. representative subsamples of the observations). Such discussions can be found in Section 3 of Tobar (2019), and we will omit those here.

A linear combination of rectangle functions for spectral mixture can then be transformed into a covariance function using a very similar argument as in the case of Gaussian mixtures. This will bypass the issue of the sinc kernel taking constant value over a range of frequencies. The author of Tobar (2019) also proposed a **generalised sinc kernel** by leveraging the convolution property of the Fourier transform to build more expressive kernels in general, but a further discretisation is needed as there does not always exist a closed form, easy to compute convolution. This kernel is thus omitted as the class of generalised sinc kernels after discretisation is a subset of the class of block mixture kernels.

Proposition 3.5. *Consider a sequence of one-dimensional rectangle functions with location ξ_q and width Δ_q for $q = 1, 2, \ldots, Q$, where we denote each density as $r_q(s) := r((s - \xi_q)/\Delta_q)$. The symmetric mixture S of $\{r_q\}_{q=1}^{Q}$ with normalised weights $\{A_q\}_{q=1}^{Q}$, defined by*

$$S(s) = \frac{1}{2}\sum_{q=1}^{Q} A_q \left[r_q(s) + r_q(-s)\right]$$

is the spectrum of a stationary kernel k, defined by

$$k(\tau) = \sum_{q=1}^{Q} A_q \operatorname{sinc}(\Delta_q \tau) \cos(2\pi\tau\xi_q).$$

Additionally, we can convince ourselves easily, just like in the case of using Gaussian mixtures, that the rectangle functions are **dense** in the space of probability measures. To see this, we can use the same argument as in the proof sketch of Proposition 3.4. We can realise that if we have a sequence of rectangle functions $\{r_n\}$ with mean C and width $1/n$, then as $n \to \infty$, the sequence of rectangle function will converge to the constant random variable at C. The rest of the proof then follows exactly.

3.2 Vector-Output Spectral Mixture

In the previous section, we have looked at spectral mixture for scalar-output GPs. In this section, we will look at the extensions of the above models to vector-output GPs. Many of the ideas used will be based heavily on the concepts in Chapter 2.

3.2.1 Spectra of Vector-Output Kernels

Recall that the key result we rely on for building spectral kernels for scalar output GP is the Bochner theorem of Theorem 1.6. To study the spectrum of vector output GP, it will be nice to have an extension of the Bochner theorem to guide our constructions. Luckily, such a theorem exists, which we will state below.

Theorem 3.6 (Cramér Theorem, Cramér (1940))**.** *A family $\{k_{i,j}(\tau)\}_{i,j=1}^{m}$ of integrable functions are the covariance functions of a weakly stationary, multivariate stochastic process if and only if we have*

- *the spectral representation*

$$k_{ij}(\tau) = \int_{\mathbb{R}^d} \exp[\sqrt{-1}\omega^T \tau] S_{ij}(\omega) d\omega, \qquad \forall i, j \in \{1, 2, \ldots, m\}$$

where each $S_{ij} : \mathbb{R}^d \to \mathbb{C}$ is integrable, which we denote as the ***spectral density*** *associated to the covariance function $k_{ij}(\tau)$.*

- *the spectral densities are positive definite, in the sense that*

$$\sum_{i,j=1}^{m} \overline{z_i} z_j S_{ij}(\omega) \geq 0 \qquad \textit{for any } z_i, z_j \in \mathbb{C}, \omega \in \mathbb{R}^d$$

where $\overline{z}$ is the complex conjugate of z.

To summarise, the Cramér theorem states that for vector-output spectrum, not only does each term of the spectrum matrix need to be the exact Fourier dual, but the spectrum matrix itself should also be positive definite. So we should consider one more thing when designing spectral kernels for vector-output GPs.

3.2.2 Gaussian Mixtures

Given the Cramér theorem, we are now ready to define the Gaussian mixtures for vector-output GPs. The goal is to make sure that each coordinate of the output, marginally, is a scalar-valued Gaussian mixture kernel. Consequently, this leads to certain (relatively) limited choices of the dependency between coordinates.

First, we recall the Cholesky decomposition states that, for a positive definite matrix S, we can find the decomposition

$$S = R^H R$$

where R^H is the Hermitian of R. This means, in order to build the desired spectrum S, we just need to find some matrix R and compute $R^H R$ to make it positive definite, as required for the Cramémer theorem.

Then, we will model each R_i where $i = 1, 2, \ldots, m$ with m being the number of dimensions of the vector output of our GP. We will set each R_i as a SE kernel here, as

$$R_i(\omega) = \omega_i \exp\left[-\frac{1}{4}(\omega - \mu_{ij})^T \Sigma_i^{-1}(\omega - \mu_{ij})\right]$$

which implies that after $R^H R$, we have the spectral densities

$$S_{ij}(\omega) = \omega_{ij} \exp\left[-\frac{1}{2}(\omega - \mu_{ij})^T \Sigma_{ij}^{-1}(\omega - \mu_{ij})\right].$$

Note that we have used the following parameters above:

- (covariance) $\Sigma_{ij} = 2\Sigma_i(\Sigma_i + \Sigma_j)^{-1}\Sigma_j$
- (mean) $\mu_{ij} = (\Sigma_i + \Sigma_j)^{-1}(\Sigma_i \mu_j + \Sigma_j \mu_i)$
- (magnitude) $\omega_{ij} = \omega_i \omega_j \exp\left[-\frac{1}{4}(\mu_i - \mu_j)^T (\Sigma_i + \Sigma_j)^{-1}(\mu_i - \mu_j)\right]$.

We can also model a delay (between the spectrum of the various coordinates) and a phase parameter into our model, as outlined in Parra and Tobar (2017) and Ulrich et al. (2015). These are simple tweaks that do not affect the main idea behind the kernel design and, thus are omitted here for simplicity of presentation.

One should easily realise that the S_{ij} constructed above is (proportionate to) the multivariate Gaussian density, as desired. Next, we can use a similar calculation as the scalar-output case, and derive both the symmetric Gaussian mixture for the spectral-domain kernel and its time-domain representation, which we state below:

$$\begin{aligned}
S_{ij}(\omega) &= \frac{\omega_{ij}}{2}\exp\left[-\frac{1}{2}(\omega-\mu_{ij})^T\Sigma_{ij}^{-1}(\omega-\mu_{ij})\right]+\frac{\omega_{ij}}{2}\exp\left[-\frac{1}{2}(\omega+\mu_{ij})^T\Sigma_{ij}^{-1}(\omega+\mu_{ij})\right]\\
k_{ij}(\tau) &= \alpha_{ij}\exp\left[-\frac{1}{2}\tau^T\Sigma_{ij}\tau\right]\cos\left(\tau^T\mu_{ij}\right)
\end{aligned}$$

where $\alpha_{ij} := \omega_{ij}(2\pi)^{n/2}|\Sigma_{ij}|^{1/2}$.

Naturally, we can do a linear combination of q copies of the above spectral mixtures with different parameters as before, which gives us the full version of the vector-output Gaussian spectral mixture kernel proposed in Parra and Tobar (2017). This is also a more general kernel than the spectral mixture kernel proposed in Ulrich et al. (2015) using the linear model of coregionalisation idea to extend the scalar-output spectral mixture to the vector-output case.

To simplify the notations, we consider the special case of a mixture of two Gaussians with equal weights for a two-vector output Gaussian spectral mixture kernel. Let u, v denote the two coordinates of the output, the spectrum of the first coordinate is

$$S_{uu} = \frac{1}{2}\left[\frac{1}{2}(G_{u_1}^+ + G_{u_1}^-) + \frac{1}{2}(G_{u_2}^+ + G_{u_2}^-)\right]$$

where G_{u_i} with $i = 1, 2$ are the Gaussian components for u and the ± superscripts denote the symmetric pairs of the Gaussians. The spectrum of the second coordinate, similarly, is

$$S_{vv} = \frac{1}{2}\left[\frac{1}{2}(G_{v_1}^+ + G_{v_1}^-) + \frac{1}{2}(G_{v_2}^+ + G_{v_2}^-)\right].$$

The cross-spectrum (i.e. the spectrum of the cross variance), consequently, is given by

$$S_{uv} = S_{vu} = \sqrt{G_{u_1}^+G_{v_1}^+} + \sqrt{G_{u_1}^-G_{v_1}^-} + \sqrt{G_{u_2}^+G_{v_2}^+} + \sqrt{G_{u_2}^-G_{v_2}^-}.$$

Given this cross-spectrum, the correlation of the vector-output Gaussian spectral kernel $\rho(\omega)$ becomes

$$\begin{aligned}
\rho(\omega) &= \frac{S_{uv}(\omega)}{\sqrt{S_{uu}(\omega)S_{vv}(\omega)}}\\
&= \frac{\sqrt{G_{u_1}^+G_{v_1}^+} + \sqrt{G_{u_1}^-G_{v_1}^-} + \sqrt{G_{u_2}^+G_{v_2}^+} + \sqrt{G_{u_2}^-G_{v_2}^-}}{\sqrt{\frac{1}{2}\left[\frac{1}{2}(G_{u_1}^+ + G_{u_1}^-) + \frac{1}{2}(G_{u_2}^+ + G_{u_2}^-)\right] + \frac{1}{2}\left[\frac{1}{2}(G_{v_1}^+ + G_{v_1}^-) + \frac{1}{2}(G_{v_2}^+ + G_{v_2}^-)\right]}}\\
&= \frac{\sqrt{G_{u_1}^+G_{v_1}^+} + \sqrt{G_{u_1}^-G_{v_1}^-} + \sqrt{G_{u_2}^+G_{v_2}^+} + \sqrt{G_{u_2}^-G_{v_2}^-}}{\frac{1}{2}\sqrt{\left[(G_{u_1}^+ + G_{u_1}^-) + (G_{u_2}^+ + G_{u_2}^-)\right] + \left[(G_{v_1}^+ + G_{v_1}^-) + (G_{v_2}^+ + G_{v_2}^-)\right]}}.
\end{aligned}$$

Notice that the value of the correlation $\rho(\omega)$ could only take 1 if we could establish the Cauchy-Schwartz inequality's equality condition. If we wish to model a perfectly correlated two-vector output GP using Gaussian spectral kernel, we could only pick our Gaussian components such that S_{uu} and S_{vv} are of the same shape completely (for Cauchy-Schwartz to take equality), which is extremely limiting. Therefore, this serves as a counterexample of the statement: 'vector-output Gaussian spectral mixture kernels are dense in the space of stationary vector-output kernels'.

3.2.3 Block Mixtures

To remedy the above problem of lack of denseness of vector-output Gaussian spectral mixture kernels, Simpson et al. (2021) proposed an alternative model with the same construction (via Cramér theorem) but different building block (using rectangle functions).

In Simpson et al. (2021), the authors argued that the lack of expressiveness of Gaussian components in the cross-spectrum, revealing itself in the form of a limited correlation range, is due to the fact that the Gaussian components overlap in tails as the overlapping in tails will induce unintended dependencies. To resolve this issue, a fixed bandwidth building block, the rectangle function introduced in Section 3.1.2, is used as it does not have the overlapping tail problem. The proposed model is summarised below.

Proposition 3.7 (Minecraft Kernel)**.** *Using rectangle function B_{μ^q,ω^q} with location μ^q and width ω^q for $q = 1, 2, \ldots, Q$ and weights A^q_{ij} for the i, j-th component of the vector-output kernel with D-dimensional input, we have the **Minecraft kernel** defined as*

$$S_{ij}(\nu) = \sum_{q=1}^{Q} \frac{1}{2} A^q_{ij} [B_{\mu^q,\omega^q}(\nu) + B_{-\mu^q,\omega^q}(\nu)]$$

$$K_{ij}(r) = \sum_{q=1}^{Q} A^q_{ij} \cos(r^T \mu^q) \prod_{d=1}^{D} \operatorname{sinc}(r_d \omega^q_d)$$

where the Q amplitude matrices A^q constructed using the weights A^q_{ij} are all positive definite.

It was shown in Simpson et al. (2021) as Theorem 3 that the above Minecraft kernel is L^1-dense in the space of vector-output stationary, real-valued kernels.

Appendix A

Mathematical Background

A.1 Linear Algebra

A helpful operation between matrices is the **Kronecker product** $\otimes$ which takes two matrices and outputs a block matrix. This is a special case of **tensor product**, and it is not the same as matrix multiplication.

Definition A.1 (Kronecker product)**.** *Consider a matrix $A \in \mathbb{R}^{m\times n}$ and a matrix $B \in \mathbb{R}^{p\times q}$, the* ***Kronecker product*** *$A \otimes B \in \mathbb{R}^{pm\times qn}$ is a block matrix defined as*

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}$$

which can be written more explicitly as

$$A \otimes B = \begin{bmatrix}
a_{11}b_{11} & \cdots a_{11}b_{12} & \cdots & a_{11}b_{1q} & \cdots & \cdots & a_{1n}b_{11} & a_{1n}b_{12} & \cdots & a_{1n}b_{1q} \\
a_{11}b_{21} & \cdots a_{11}b_{22} & \cdots & a_{11}b_{2q} & \cdots & \cdots & a_{1n}b_{21} & a_{1n}b_{22} & \cdots & a_{1n}b_{2q} \\
\vdots & \vdots & \ddots & \vdots & & & \vdots & \vdots & \ddots & \vdots \\
a_{11}b_{p1} & \cdots a_{11}b_{p2} & \cdots & a_{11}b_{pq} & \cdots & \cdots & a_{1n}b_{p1} & a_{1n}b_{p2} & \cdots & a_{1n}b_{pq} \\
\vdots & \vdots & & \vdots & \ddots & & \vdots & \vdots & & \vdots \\
\vdots & \vdots & & \vdots & & \ddots & \vdots & \vdots & & \vdots \\
a_{m1}b_{11} & \cdots a_{m1}b_{12} & \cdots & a_{m1}b_{1q} & \cdots & \cdots & a_{mn}b_{11} & a_{mn}b_{12} & \cdots & a_{mn}b_{1q} \\
a_{m1}b_{21} & \cdots a_{m1}b_{22} & \cdots & a_{m1}b_{2q} & \cdots & \cdots & a_{mn}b_{21} & a_{mn}b_{22} & \cdots & a_{mn}b_{2q} \\
\vdots & \vdots & \ddots & \vdots & & & \vdots & \vdots & \ddots & \vdots \\
a_{m1}b_{p1} & \cdots a_{m1}b_{p2} & \cdots & a_{m1}b_{pq} & \cdots & \cdots & a_{mn}b_{p1} & a_{mn}b_{p2} & \cdots & a_{mn}b_{pq}
\end{bmatrix}.$$

Some of the properties of the Kronecker product are listed below.

Proposition A.2. *Consider matrices A, B, C, a zero matrix 0, and a scalar $k \in \mathbb{R}$. We have*

- $A \otimes (B + C) = A \otimes B + A \otimes C$
- $(B + C) \otimes A = B \otimes A + C \otimes A$
- $(kA) \otimes B = A \otimes (kB) = k(A \otimes B)$
- $(A \otimes B) \otimes C = A \otimes (B \otimes C)$
- $A \otimes 0 = 0 \otimes A = 0$

A.2 Vector Calculus

Consider the scalar function $f : \mathbb{R}^2 \to \mathbb{R}$ that is differentiable, and the vector field $F : \mathbb{R}^2 \to \mathbb{R}^2$ that is differentiable in both of its coordinates.

The **gradient** of f is defined by

$$\operatorname{grad} f(x) = \nabla f(x) = \begin{bmatrix} \partial_{x_1} f(x) \\ \partial_{x_2} f(x) \end{bmatrix}$$

where $x = (x_1, x_2)$ and $\partial_{x_i} f$ is the partial derivative of f with respect to x_i where $i = 1, 2$ here. Notice that the scalar function f becomes a vector field after ∇. In particular, we can isolate the ∇ operator and it is defined as

$$\nabla = \begin{bmatrix} \partial_{x_1} \\ \partial_{x_2} \end{bmatrix}.$$

The **divergence** of F is defined by

$$\operatorname{div} F(x) = \nabla \cdot F(x) = (\nabla \cdot F)(x) = \begin{bmatrix} \partial_{x_1} F_1(x) \\ \partial_{x_2} F_2(x) \end{bmatrix}$$

where $F_i(x)$ represents the i-th coordinate of $F(x)$ where $i = 1, 2$ here.

The **curl** of F is defined by

$$\operatorname{curl} F(x) = \nabla \times F(x) = \begin{bmatrix} \partial_{x_1} \\ \partial_{x_2} \end{bmatrix} \times \begin{bmatrix} F_1(x) \\ F_2(x) \end{bmatrix} = \partial_{x_1} F_2(x) - \partial_{x_2} F_1(x)$$

where × represents the cross product of vectors. The curl, which makes much more intuitive sense in $\mathbb{R}^3$, captures the infinitesimal circulation of the motion.

The **rotation** of f is defined by

$$\operatorname{rot} f(x) = \boldsymbol{k} \times \nabla f = \begin{bmatrix} -\partial_{x_2} f \\ \partial_{x_1} f \end{bmatrix}$$

where $\boldsymbol{k}$ represents the unit vector in the third dimension that is orthogonal to the two unit vectors in the first and second dimensions that we use (by default) to set up the vector. Note that different groups of people use different versions of this operator. In the oceanography literature, the above definition is the commonly used version, which is what we will use here. In the physics literature, however, the minus sign is on the second coordinate rather than the first.

One should note that a vector field F is called **curl-free** if $\operatorname{curl} F = 0$, and it is called **incompressible**, or **divergence-free**, if $\operatorname{div} F = 0$. Two easy-to-show but relevant identities about curl, div, grad, and rot are below: for continuous scalar field f, we have

$$\operatorname{curl}(\operatorname{grad} f) = 0, \qquad \operatorname{div}(\operatorname{rot} f) = 0.$$

This motivates the **Helmholtz decomposition** of a vector field in 2D.

Definition A.3 (Helmholtz Decomposition in 2D)**.** *Consider a twice continuously differentiable and compactly supported vector field $F : \mathbb{R}^2 \to \mathbb{R}^2$. The* ***Helmholtz decomposition*** *indicates that there exists a scalar potential $\Phi : \mathbb{R}^2 \to \mathbb{R}$, called the* ***potential function****, and a scalar potential $\Psi : \mathbb{R}^2 \to \mathbb{R}$, called the* ***stream function****, such that we have*

$$F = \operatorname{grad}\ \Phi + \operatorname{rot}\ \Psi$$

where functions Φ, Ψ are not unique.

Notice that since $\operatorname{curl}(\operatorname{grad} f) = 0$ and $\operatorname{div}(\operatorname{rot} f) = 0$, the above decomposition essentially decomposes the vector field into a curl-free part $\operatorname{grad}\ \Phi$ and a divergence-free part $\operatorname{rot}\ \Psi$.

A.3 Fourier Analysis

A.3.1 Properties of Fourier Transform

Proposition A.4. *If we denote the Fourier transform operator as $\mathcal{F}$ and f, h be functions that admit Fourier transforms. We have*

- $\mathcal{F}[f(x - x_0)] = \exp[-2\pi i x_0 \tau] \cdot \mathcal{F}[f](\tau), \quad x_0 \in \mathbb{R}.$
- $\mathcal{F}[f(ax)] = |a|^{-1} \mathcal{F}[f](\tau/a), \quad a \neq 0.$
- $\mathcal{F}[af(x) + bh(x)] = a\mathcal{F}[f](x) + b\mathcal{F}[f](x), \quad a, b \in \mathbb{C}.$
- $\mathcal{F}[f * h] = \mathcal{F}[f] \cdot \mathcal{F}[g]$ *where* $*$ *is the convolution operator.*

Bibliography

Adler, R. J. (2010). *The Geometry of Random Fields*, SIAM.

Alvarez, M. A., Rosasco, L., Lawrence, N. D. et al. (2012). Kernels for vector-valued functions: A review, *Foundations and Trends® in Machine Learning* **4**(3): 195–266.

Berlinghieri, R., Trippe, B. L., Burt, D. R., Giordano, R. J., Srinivasan, K., Özgökmen, T., Xia, J. and Broderick, T. (2023). Gaussian processes at the Helm (holtz): A more fluid model for ocean currents, *International Conference on Machine Learning*, PMLR, pp. 2113–2163.

Bonilla, E. V., Chai, K. and Williams, C. (2007). Multi-task Gaussian process prediction, *Advances in Neural Information Processing Systems* **20**.

Cramér, H. (1940). On the theory of stationary random processes, *Annals of Mathematics* **41**(1): 215–230.

Journel, A. G. and Huijbregts, C. J. (1978). *Mining Geostatistics*, Academic Press, San Diego, CA.

Lilly, J. M. and Pérez-Brunius, P. (2021). GulfDrifters: A consolidated surface drifter dataset for the Gulf of Mexico.

Marks, R. (1990). *Introduction to Shannon sampling and Interpolation Theory*, Coastal and Estuarine Studies, Springer, New York, NY.

Parra, G. and Tobar, F. (2017). Spectral mixture kernels for multi-output Gaussian processes, *Advances in Neural Information Processing Systems* **30**.

Pinder, T. and Dodd, D. (2022). GPJax: A Gaussian Process Framework in JAX, *Journal of Open Source Software* **7**(75): 4455.

Ponte, A. L. S., Astfalck, L., Rayson, M., Zulberti, A. and Jones, N. (2024). Inferring flow energy, space and time scales: freely-drifting vs fixed point observations, *Nonlinear Processes in Geophysics Discussions* **2024**: 1–17.

Rudin, W. (2017). *Fourier Analysis on Groups*, Dover Publications, Mineola, NY.

Simpson, F., Boukouvalas, A., Cadek, V., Sarkans, E. and Durrande, N. (2021). The minecraft kernel: Modelling correlated gaussian processes in the fourier domain, *International Conference on Artificial Intelligence and Statistics*, PMLR, pp. 1945–1953.

Tobar, F. (2019). Band-limited Gaussian processes: The sinc kernel, *Advances in Neural Information Processing Systems* **32**.

Ulrich, K. R., Carlson, D. E., Dzirasa, K. and Carin, L. (2015). GP kernels for cross-spectrum analysis, *Advances in Neural Information Processing Systems* **28**.

Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian Processes for Machine Learning*, Vol. 2, MIT press Cambridge, MA.

Wilson, A. and Adams, R. (2013). Gaussian process kernels for pattern discovery and extrapolation, *International Conference on Machine Learning*, PMLR, pp. 1067–1075.