Gaussian Processes for Machine Learning

Matthias Seeger*
Department of EECS
University of California at Berkeley
485 Soda Hall, Berkeley CA 94720-1776, USA
mseeger@cs.berkeley.edu

February 24, 2004

Abstract

Gaussian processes (GPs) are natural generalisations of multivariate Gaussian random variables to infinite (countably or continuous) index sets. GPs have been applied in a large number of fields to a diverse range of ends, and very many deep theoretical analyses of various properties are available. This paper gives an introduction to Gaussian processes on a fairly elementary level with special emphasis on characteristics relevant in machine learning. It draws explicit connections to branches such as spline smoothing models and support vector machines in which similar ideas have been investigated.

Gaussian process models are routinely used to solve hard machine learning problems. They are attractive because of their flexible non-parametric nature and computational simplicity. Treated within a Bayesian framework, very powerful statistical methods can be implemented which offer valid estimates of uncertainties in our predictions and generic model selection procedures cast as nonlinear optimization problems. Their main drawback of heavy computational scaling has recently been alleviated by the introduction of generic sparse approximations [13, 78, 31]. The mathematical literature on GPs is large and often uses deep concepts which are not required to fully understand most machine learning applications. In this tutorial paper, we aim to present characteristics of GPs relevant to machine learning and to show up precise connections to other "kernel machines" popular in the community. Our focus is on a simple presentation, but references to more detailed sources are provided.

1 Introduction and Overview: Gaussian Processes in a Nutshell

In this section, we introduce the basic reasoning behind non-parametric random field and Gaussian process models. Readers who have been exposed to these concepts may jump to the end of the section where an overview of the remaining sections is given.

In most machine learning problems, we aim to generalise from a finite set of observed data, in the sense that our ability to predict uncertain aspects of a problem improves after making

*Previously at: Institute for Adaptive and Neural Computation, University of Edinburgh, UK.

the observations. This is possible only if we postulate *a priori* a relationship between the variables we will observe and the ones we wish to predict. This relationship is uncertain itself, making generalisation a non-trivial problem. For example, in spatial statistics we observe the values of a function at certain locations and want to predict them at other ones. In temporal statistics, we might want to predict future values of a time series from its past. In the situations we are interested in here, the postulated relationship can be represented by an ensemble (or a distribution) of functions. It is helpful to imagine the observed data being "generated" by picking a function from the ensemble which gives rise to the sample (typically, observations themselves are imperfect or "noisy"). It is important to stress that this generative view can well be a crude abstraction of the mechanism we really hold capable of simulating the phenomenon, as long as its probabilistic inversion leads to satisfying predictions. This inversion is obtained by conditioning the generative ensemble on the observed data, which leads to a new *adapted* ensemble pinned down at observation points but still variable elsewhere.

In *parametric* statistics, we agree on a function class indexed by a finite number of parameters. A distribution over these parameters induces an ensemble over functions. Learning from observations means to modify this distribution so to adapt the ensemble to the data. If our *a priori* postulate is a very informed one (e.g. if the function class is motivated by a physical theory of the phenomenon), the parametric approach is the method of choice, but if many aspects of the phenomenon are unknown or hard to describe explicitly, *non-parametric* modelling can be more versatile and powerful. It is important to stress that our aim is solely to obtain accurate predictions together with valid estimates of uncertainty, *not* to "explain" the inner workings of the true generative process. In the latter case, non-parametric modelling is less applicable.

In *non-parametric* statistics, regularities of the relationship are postulated without requiring the ensemble to be concentrated on a easily describable class. For example, we may assume the ensemble to be stationary or isotropic (see Section 2), which allows us to infer properties of the generative ensemble even though our observations come from a single realisation thereof. We might also postulate smoothness so that nearby points (in space or time) have similar values with high probability, periodicity, boundary conditions, *etc.* In contrast to the parametric case, it is less clear how we can represent such a generative ensemble explicitly.

A random field is a mapping from an input space to real-valued random variables[1], a natural generalisation of a joint distribution to an infinite index set. Like a joint distribution, we can try to describe the field by its low-order cumulants such as mean and covariance function, the latter being a bivariate form satisfying a positive semidefiniteness property akin to a covariance matrix of a joint distribution. If all cumulants above second order vanish, the random field is Gaussian: a Gaussian process. Importantly, properties such as stationarity, isotropy, smoothness, periodicity, *etc.* can be enforced via the choice of the covariance function. Furthermore, all finite-dimensional marginal distributions of the field are jointly Gaussian, and inference and prediction require little more than numerical linear algebra.

With this brief introduction, we hope to have motivated the reader to browse through the more detailed sections to follow. Section 2 defines Gaussian processes, introduces the important subclasses of stationary and isotropic GPs and develops two different views on GPs

[1]The extension to complex-valued random fields is straightforward. Since most machine learning applications require real-valued fields only, we concentrate on this case for simplicity.

prominent in machine learning. Some elementary GP models are introduced in Section 3. Approximate inference techniques for such models are discussed in Section 4 using a generic framework. Theoretical aspects of GPs can be understood by associating them with reproducing kernel Hilbert spaces (RKHS), as shown in Section 5. Traditionally, GP models have been used in the context of penalised maximum likelihood and spline smoothing which are motivated in Section 6. A variant of spline smoothing, the support vector machine has gained large popularity in the machine learning community, its relationship to Bayesian GP techniques is given in Section 7. GP models have been used extensively in spatial statistics, using an estimation procedure called kriging, as described in Section 8. The final Section 9 deals with the choice of the covariance function which is of central importance in GP modelling. We describe classes of standard kernels and their properties, show how kernels can be constructed from elementary parts, discuss methods for learning aspects of the kernel and finally illustrate classes of covariance functions over discrete index sets.

Readers more interested in practical machine learning aspects may want to skip over Sections 5 and 6 which contain more theoretical material not required to understand GP applications. We use notational conventions familiar to probability theorists which is introduced in Section A.1, but are careful to motivate the formalism in the more applied sections.

2 Gaussian Processes: The Process and the Weight Space View

Gaussian process (GP) models are constructed from classical statistical models by replacing latent functions of parametric form (e.g. linear functions, truncated Fourier or Wavelet expansions, multi-layer perceptrons) by random processes with Gaussian prior. In this section, we will introduce GPs and highlight some aspects which are relevant to machine learning. We develop two simple views on GPs, pointing out similarities and key differences to distributions induced by parametric models. We follow [2], Chap. 1,2. A good introduction into the concepts required to study GP prediction is given in [74], Chap. 2. For concepts and vocabulary from general probability theory, we refer to [19, 6, 9].

Let $\mathcal{X}$ be an non-empty index set. For the main parts of this paper, $\mathcal{X}$ can be arbitrary, but here we assume that $\mathcal{X}$ is at least a group[2] (and sometimes we assume it to be $\mathbb{R}^g$). In a nutshell, a *random process* $\mathcal{X} \to \mathbb{R}$ is a collection of random variables (one for each $\boldsymbol{x} \in \mathcal{X}$) over a common probability space. The measure-theoretic definition is awkward, but basically the same as for a single variable. It can also be viewed as a function from the probability space and $\mathcal{X}$ into the reals. The functions $\mathcal{X} \to \mathbb{R}$ obtained for a fixed atomic event are called *sample paths*, and a random process can also be seen as the corresponding distribution over sample paths. If $X \subset \mathcal{X}$ is finite, we obtain a random variable $\in \mathbb{R}^{|X|}$ by evaluating the process at the points X, its distribution is called *finite-dimensional distribution (f.d.d.)*. If we assume that a random process exists and consider the system of all f.d.d.'s, it is clear that it has to be *symmetric* and *consistent*: a permutation of the components of X must result in the distribution of an equally permuted random vector, and if $X_1 \cap X_2 \neq \emptyset$, the marginal distributions on the intersection starting from the ones for X_1 and X_2 must be identical. Formally, for every $n \in \mathbb{N}_{>0}$, $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n \in \mathcal{X}$, Borel sets $B_1, \dots, B_n$ and every

[2] Has an addition $+$, an origin $\mathbf{0}$ and a negation $-$.

permutation π of $\{1,\dots,n\}$ we must have

$$\mu_{\boldsymbol{x}_{\pi(1)},\dots,\boldsymbol{x}_{\pi(n)}}(B_{\pi(1)}\times\cdots\times B_{\pi(n)}) = \mu_{\boldsymbol{x}_1,\dots,\boldsymbol{x}_n}(B_1\times\cdots\times B_n) \quad \text{and}$$
$$\mu_{\boldsymbol{x}_1,\dots,\boldsymbol{x}_n}(B_1\times\cdots\times B_{n-1}\times\mathbb{R}) = \mu_{\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1}}(B_1\times\cdots\times B_{n-1}).$$

Importantly, Kolmogorov [28] proved that symmetry and consistency are also *sufficient* conditions for such a specification to guarantee the existence of a random process (in the concrete measure-theoretic sense) with these f.d.d.'s. The question about uniqueness of random processes is tricky, because two processes can be *equivalent* ($u(\boldsymbol{x}) = v(\boldsymbol{x})$ almost surely for every $\boldsymbol{x}$; equivalent processes are called *versions* of each other), yet differ significantly w.r.t. almost sure properties of their sample paths. For example, one can construct a version of a smooth process whose sample paths are not differentiable at a finite number of points almost surely. In the context of machine learning applications we are interested here, sample path properties such as differentiability are of lesser importance, and we will focus on m.s. properties (to be introduced shortly) which can be characterised more directly and are invariant under change of version. In other words, we will in general identify a process with the equivalence class of all its versions or with a particularly "nice" member of this class,[3] and the simple nature of the applications we are interested in here guarantees the admissability of this practice. We will see that global sample path properties of a process (in this sense) such as smoothness and average variability are directly related to corresponding m.s. properties. See Adler [2] for methods of studying sample path properties.

Let $\{X_n\}$ be a sequence of real-valued random variables, and recall that $X_n \to X$ $(n\to\infty)$ *in quadratic mean* (or *in mean square (m.s.)*) if $\mathrm{E}[|X_n - X|^2] \to 0$. M.s. convergence is weaker than *almost sure (a.s.)* convergence, but turns out to be the most useful mode for discussing GP aspects we require here. In general, X and Y are *m.s. equivalent* if $\mathrm{E}[|X-Y|^2] = 0$. In a nutshell, for a property which is traditionally defined in terms of limits (such as continuity, differentiability, etc.) within $\mathbb{R}$ we can typically define the corresponding m.s. property for scalar random variables by substituting normal for m.s. convergence.

Suppose that $u(\boldsymbol{x})$ is a random process. The first and second-order statistics of $u(\boldsymbol{x})$ are its *mean function* $m(\boldsymbol{x}) = \mathrm{E}[u(\boldsymbol{x})]$ and *covariance function*

$$K(\boldsymbol{x},\boldsymbol{x}') = \mathrm{E}\left[(u(\boldsymbol{x}) - m(\boldsymbol{x}))(u(\boldsymbol{x}') - m(\boldsymbol{x}'))\right].$$

Obviously, both depend on the f.d.d.'s of the process only. The covariance function is central to studying characteristics of the process in the mean square sense. It is a *positive semidefinite*[4] function in the sense that for every $n\in\mathbb{N}$, $\boldsymbol{x}_1,\dots,\boldsymbol{x}_n\in\mathcal{X}$, $z_1,\dots,z_n\in\mathbb{R}$:

$$\sum_{i,j=1}^{n} z_i z_j K(\boldsymbol{x}_i,\boldsymbol{x}_j) \geq 0. \tag{1}$$

This is clear because for $X = \sum_i z_i(u(\boldsymbol{x}_i) - m(\boldsymbol{x}_i))$ we have $\mathrm{E}[|X|^2] \geq 0$. Positive semidefiniteness means that for every finite set $X \subset \mathcal{X}$ the symmetric matrix $K(X,X)\in\mathbb{R}^{|X|,|X|}$ obtained by evaluating K on $X\times X$ is positive semidefinite. Note that this implies that $K(\boldsymbol{x},\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x}$. K is called *positive definite* if (1) holds with $>$ whenever $\boldsymbol{z}\neq\mathbf{0}$.

[3] As an example, a Wiener process (see Section 2.3) always has a version with continuous sample paths.

[4] This term is not uniquely used in the literature, it is sometimes replaced by *non-negative definite* or even *positive definite* (which has a different meaning here).

The positive semidefiniteness of K leads to an important spectral decomposition which is discussed in Section 5. A positive semidefinite K will also be referred to as *kernel*, pointing out its role as kernel for a linear integral operator (see Section 5).

2.1 Stationary Processes

In many situations, the behaviour of the process does not depend on the location of the observer, and under this restriction a rich theory can be developed, linking local m.s. properties of the process to the behaviour of K close to the origin. A process is called *strictly homogeneous* (or *strictly stationary*) if its f.d.d.'s are invariant under simultaneous translation of their variables. This implies that $m(\boldsymbol{x})$ is constant and $K(\boldsymbol{x}, \boldsymbol{x}')$ is a function of $\boldsymbol{x} - \boldsymbol{x}'$ (we write $K(\boldsymbol{x}, \boldsymbol{x}') = K(\boldsymbol{x} - \boldsymbol{x}')$ in this case). A process fulfilling the latter two conditions is called *(weakly) homogeneous* (or *(weakly) stationary*). For a stationary process, the choice of the origin is not reflected in the statistics up to second order. If $K(\mathbf{0}) > 0$,

$$\rho(\boldsymbol{x}) = \frac{K(\boldsymbol{x})}{K(\mathbf{0})}$$

is called *correlation function.* A stationary process has a spectral representation as a stochastic Fourier integral (e.g., [2], Chap. 2; [19], Chap. 9; [88]), based on *Bochner's theorem* which (for $\mathcal{X} = \mathbb{R}^g$) asserts that $\rho(\boldsymbol{x})$ is positive semidefinite, furthermore uniformly continuous with $\rho(\mathbf{0}) = 1$, $|\rho(\boldsymbol{x})| \le 1$ iff it is the characteristic function of a variable $\boldsymbol{\omega}$, *i.e.*

$$\rho(\boldsymbol{x}) = \int e^{i\,\boldsymbol{x}^T\boldsymbol{\omega}}\, dF(\boldsymbol{\omega}) \tag{2}$$

for a probability distribution function $F(\boldsymbol{\omega})$. If $F(\boldsymbol{\omega})$ has a density $f(\boldsymbol{\omega})$ (w.r.t. Lebesgue measure), f is called *spectral density.* This theorem allows to prove positive semidefiniteness of K by computing its Fourier transform and checking that it is non-negative. If so, it must be proportional to the spectral density. Note that since $\rho(\boldsymbol{x})$ is an even function, the spectral distribution is symmetric around $\mathbf{0}$, and if $f(\boldsymbol{\omega})$ exists it is even as well.

The f.d.d.'s of a process determine its mean square properties, while this is not true in general for almost sure properties (such as continuity or differentiability of sample paths). Even stronger, for a zero-mean process, m.s. properties are usually determined *entirely* by the covariance function $K(\boldsymbol{x}, \boldsymbol{x}')$. For stationary processes, it is merely the behaviour of $K(\boldsymbol{x})$ at the origin which counts: the m.s. derivative[5] $D_{\boldsymbol{x}} u(\boldsymbol{x})$ exists everywhere iff $D_{\boldsymbol{x}} D_{\boldsymbol{x}} K(\boldsymbol{x})$ exists at $\boldsymbol{x} = \mathbf{0}$. Thus, the smoothness of the process in the m.s. sense grows with the degree of differentiability at $\mathbf{0}$. For example, a process with the RBF (Gaussian) covariance function K (27) is m.s. analytic, because K is analytic (differentiable up to any order) at $\mathbf{0}$.

2.2 Isotropic Processes

A stationary process is called *isotropic* if its covariance function $K(\boldsymbol{x})$ depends on $\|\boldsymbol{x}\|$ only. In this case, the spectral distribution F is invariant under isotropic isomorphisms (e.g., rotations). Loosely speaking, second-order characteristics of an isotropic process are

[5]Here, $D_{\boldsymbol{x}}$ denotes a differential functional, such as $\partial^2/(\partial x_1 \partial x_2)$.

the same from whatever position and direction they are observed. It is much simpler to characterise isotropic correlation functions than stationary ones in general. Let $\rho(\tau) = \rho(\boldsymbol{x})$ for $\tau = \|\boldsymbol{x}\|$. The spectral decomposition (2) simplifies to

$$\rho(\tau) = \int \Lambda_{g/2-1}(\tau\,\omega) dF(\omega) \tag{3}$$

where $F(\omega) = \int \mathrm{I}_{\{\|\boldsymbol{\omega}\| \le \omega\}}\, dF(\boldsymbol{\omega})$ is a distribution function for $\omega \ge 0$ and

$$\Lambda_\nu(z) = \frac{\Gamma(\nu+1)}{(z/2)^\nu} J_\nu(z),$$

where $J_\nu(z)$ is a Bessel function of the first kind (see [2], Sect. 2.5). Recall that g is the dimensionality of the input space $\mathcal{X} = \mathbb{R}^g$. The right hand side in (3) is the *Hankel transform* of order $g/2-1$ of F (see [74], Sect. 2.10). Alternatively, if the spectral density $f(\boldsymbol{\omega})$ exists and $f(\omega) = f(\boldsymbol{\omega})$ for $\omega = \|\boldsymbol{\omega}\|$, then $dF(\omega) = A_{g-1}\omega^{g-1} f(\omega)\, d\omega$,[6] so we can easily convert to the spectral representation in terms of $f(\omega)$. Denote the set of $\rho(\tau)$ corresponding to isotropic correlation functions in $\mathbb{R}^g$ by $\mathcal{D}^g$. Note that (3) characterises $\mathcal{D}^g$ (by Bochner's theorem). It is clear that $\mathcal{D}^{g+1} \subset \mathcal{D}^g$, since an isotropic correlation function in $\mathbb{R}^{g+1}$ restricted to a g-dimensional subspace is in $\mathcal{D}^g$. Beware that both $F(\omega)$ and $f(\omega)$ depend on the dimension g for which $\rho(\tau)$ is used to induce a correlation function (see (3)). Let $\mathcal{D}^\infty = \bigcap_{g\ge 1} \mathcal{D}^g$. Since

$$\Lambda_{g/2-1}\left((2g)^{1/2} x\right) \to e^{-x^2} \; (g \to \infty),$$

one can show that $\rho(\tau) \in \mathcal{D}^\infty$ iff $\rho(\tau) = \int \exp(-\tau^2\omega^2)\, dF(\omega)$ (this result is due to Schoenberg).

Note that the assumption of isotropy puts strong constraints on the correlation function, especially for large g. For example, $\rho(\tau) \ge \inf_x \Lambda_{g/2-1}(x) \ge -1/g$ so large negative correlations are ruled out. If $\rho(\tau) \in \mathcal{D}^\infty$, it must be non-negative. Furthermore, for large g $\rho(\tau)$ is smooth on $(0,\infty)$ while it may have a jump at 0 (additive white noise). If $\rho(\tau) \in \mathcal{D}^g$ and $\boldsymbol{B} \in \mathbb{R}^{g,g}$ is nonsingular, then

$$\rho_{\boldsymbol{B}}(\boldsymbol{x}) = \rho(\|\boldsymbol{B}\boldsymbol{x}\|)$$

is a correlation function as well, called *anisotropic*. Examples of (an)isotropic covariance functions are given in Section 9.

2.3 Two Views on Gaussian Processes

A *Gaussian process (GP)* is a process whose f.d.d.'s are Gaussian. Since a Gaussian is determined by its first and second-order cumulants and these involve pairwise interactions only, its f.d.d.'s are completely determined by mean and covariance function. This means that for GPs, strong and weak stationarity are the same concept. GPs are by far the most accessible and well-understood processes (on uncountable index sets). It is clear that for every positive semidefinite function K there exists a zero-mean GP with K as covariance function (by Kolmogorov's theorem), so GPs as modelling tool are very flexible. Importantly, by choosing K properly we can encode properties of the function distribution implicitly as we desired in Section 1.

[6] $A_{g-1} = 2\pi^{g/2}/\Gamma(g/2)$ is the surface area of the unit sphere in $\mathbb{R}^g$.

In conjunction with latent variable modelling techniques, a wide variety of non-parametric models can be constructed (see Section 3). The fact that all f.d.d.'s are Gaussian with covariance matrices induced by $K(\boldsymbol{x}, \boldsymbol{x}')$ can be used to obtain approximations to Bayesian inference fairly straightforwardly (see Section 4), and these approximations often turn out to be much more accurate than for parametric models of equal flexibility (such as multi-layer perceptrons). It is interesting to note that m.s. derivatives $D_{\boldsymbol{x}} u(\boldsymbol{x})$ of a GP are GPs again (if they exist), and

$$\mathrm{E}\left[D_{\boldsymbol{x}}^{(1)} u(\boldsymbol{x}) D_{\boldsymbol{x}'}^{(2)} u(\boldsymbol{x}')\right] = D_{\boldsymbol{x}}^{(1)} D_{\boldsymbol{x}'}^{(2)} K(\boldsymbol{x}, \boldsymbol{x}'),$$

thus derivative observations can be incorporated into a model in the same way as function value observations (for applications, see [46, 71]). Characteristics such as m.s. differentiability up to a given order can be controlled via the covariance function (see Section 2.1), an example is given in Section 9.

One of the most thoroughly studied GPs is the *Wiener process* (or *Brownian motion*, or *continuous random walk*) with covariance function $K(x, x') = \sigma^2 \min\{x, x'\}$ (here, $\mathcal{X} = \mathbb{R}_{\geq 0}$; for multivariate generalisations to Brownian sheets, see [2], Chap. 8). It is characterised by $u(0) = 0$ a.s., $\mathrm{E}[|u(x+h) - u(x)|^2] = \sigma^2 h,\ h \geq 0$, and by having orthogonal[7] increments: $\mathrm{E}[(u(x_1) - u(x_2))(u(x_3) - u(x_4))] = 0,\ x_1 \leq x_2 \leq x_3 \leq x_4$. Note that $u(x)$ is not stationary, a stationary version with orthogonal increments is the Ornstein-Uhlenbeck process (see Section 9.1). The Wiener process is an example for a diffusion process. It has a large number of applications in mathematics, physics and mathematical finance. The property of orthogonal increments allows to define *stochastic integrals* (e.g., [19], Chap. 13) with a Wiener process as (random) measure. $u(x)$ is m.s. continuous everywhere, but not m.s. differentiable at any point. In fact, a version of the Wiener process can be constructed which has continuous sample paths, but for every version sample paths are nowhere differentiable with probability 1. The Wiener process can be used to explicitly construct other GPs by means of stochastic integrals, the procedure is sketched in Section 6.

We now develop two elementary views on Gaussian processes, the *process* and the *weight space view*. While the former is usually much simpler to work with, the latter allows us to relate GP models to parametric linear models rather directly. We follow [85].[8] The process view on a zero-mean GP $u(\boldsymbol{x})$ with covariance function K is in the spirit of the GP definition given above. $u(\boldsymbol{x})$ is defined implicitly, in that for any finite subset $X \subset \mathcal{X}$ it induces a f.d.d. $N(\mathbf{0}, \boldsymbol{K}(X))$ over the vector $\boldsymbol{u} = u(X)$ of process values at the points X. Here, $\boldsymbol{K}(X) = \boldsymbol{K}(X, X) = (K(\boldsymbol{x}_i, \boldsymbol{x}_j))_{i,j}$ where $X = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\}$. Kolmogorov's theorem guarantees the existence of a GP with this family of f.d.d.'s.[9] In practice, many modelling problems involving an unknown functional relationship $u(\boldsymbol{x})$ can be formulated such that only ever a finite number of linear characteristics of $u(\boldsymbol{x})$ (e.g., evaluations or derivatives of $u(\boldsymbol{x})$) are linked to observations or predictive queries, and in such cases the process view boils down to dealing with the "projection" of the GP onto a multivariate Gaussian distribution, thus to simple linear algebra of quadratic forms.[10]

[7] Orthogonality implies independence since the process is Gaussian.

[8] We use the term "process view" instead of "function space view" employed in [85]. The relationship between GPs and associated spaces of smooth functions is a bit subtle and introduced only below in Section 5.

[9] If K is continuous everywhere, a version exists with continuous sample paths, but we do not require this here.

[10] In practice, some knowledge of numerical mathematics is required to avoid numerically instable proce-

GPs can also be seen from a *weight space viewpoint*, relating them to the linear model. In the Bayesian context this view was first suggested by O'Hagan [45] as a "localised regression model" (the weight space is finite-dimensional there) while the generalisation to arbitrary GP priors developed there uses the process view. This paper is among the first to address GP regression in a rigorous Bayesian context, while the equivalence between spline smoothing and Bayesian estimation of processes was noticed earlier by Kimeldorf and Wahba [27] (see Section 6). Recall the linear model

$$y = \Phi(\boldsymbol{x})^T \boldsymbol{\beta} + \varepsilon, \tag{4}$$

where $\Phi(\boldsymbol{x})$ is a feature map from the covariate $\boldsymbol{x}$ and ε is independent Gaussian noise. Every GP whose covariance function satisfies weak constraints can be written as (4), albeit with possibly infinite-dimensional weight space. To develop this view, we use some facts which are discussed in detail below in Section 5. Under mild conditions on the covariance function $K(\boldsymbol{x}, \boldsymbol{x}')$ of $u(\boldsymbol{x})$, we can construct a sequence

$$\sum_{\nu=1}^{k} \beta_\nu \lambda_\nu^{1/2} \phi_\nu(\boldsymbol{x}),$$

which converges to $u(\boldsymbol{x})$ in quadratic mean ($k \to \infty$).[11] Here, β_ν are i.i.d. $N(0,1)$ variables. ϕ_ν are orthonormal eigenfunctions of the operator induced by K with corresponding eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$, $\sum_{\nu \geq 1} \lambda_\nu^2 < \infty$, in a sense made precise in Section 5. Thus, if $\boldsymbol{\beta} = (\beta_\nu)_\nu$ and $\Phi(\boldsymbol{x}) = (\lambda_\nu^{1/2} \phi_\nu(\boldsymbol{x}))_\nu$, then $u(\boldsymbol{x}) = \Phi(\boldsymbol{x})^T \boldsymbol{\beta}$ in quadratic mean, and $\Phi(\boldsymbol{x})^T \Phi(\boldsymbol{x}') = K(\boldsymbol{x}, \boldsymbol{x}')$. This is the *weight space view* on GPs and allows to view a non-parametric regression model

$$y = u(\boldsymbol{x}) + \varepsilon$$

as direct infinite-dimensional generalisation of the linear model (4) with spherical Gaussian prior on $\boldsymbol{\beta}$. We say that $\Phi(\boldsymbol{x})$ maps into a *feature space* which is typically (countably) infinite-dimensional. It is important to note that in this construction of the feature map $\Phi(\boldsymbol{x})$ the individual components $\lambda_\nu^{1/2} \phi_\nu(\boldsymbol{x})$ do not have the same scaling, in the sense that their norm in $\mathcal{L}_2(\mu)$ (the Hilbert space they are drawn from and that K operates on) is $\lambda_\nu^{1/2} \to 0\,(\nu \to \infty)$. They are comparable in a different (RKHS) norm which scales with the "roughness" of a function. Intuitively, as $\nu \to \infty$, the graph of ϕ_ν becomes rougher and increasingly complicated, see Section 5 for details.

For all inference purposes which are concerned with f.d.d.'s of $u(\boldsymbol{x})$ and its derivatives (or other linear functionals) only, the process and the weight space view are equivalent: they lead to identical results. However, we feel that often the process view is much simpler to work with, avoiding spurious infinities[12] and relying on familiar Gaussian manipulations only. On the other hand, the weight space view is more frequently used at least in the machine learning literature, and its peculiarities may be a reason behind the perception that GP models are difficult to interpret. There is also the danger that false intuitions or conclusions

dures. Since most matrices to be dealt with are positive semidefinite, this is not too hard. Some reliable techniques are mentioned in Section 4.

[11] We only need pointwise m.s. convergence, although much stronger statements are possible under mild assumptions, e.g. [2], Sect. 3.3.

[12] Which seem to cancel out almost "magically" in the end from the weight space viewpoint, while infinities do not occur in the process view in the first place.

are developed from interpolating geometrical arguments from low-dimensional Euclidean space to the feature space.[13] We should also note that a weight space representation of a GP in terms of a feature map Φ is of course not unique. The route via eigenfunctions of the covariance operator is only one way to establish such.[14] About the only invariant is that we always have $\Phi(\boldsymbol{x})^T\Phi(\boldsymbol{x}') = K(\boldsymbol{x}, \boldsymbol{x}')$.

2.4 Gaussian Processes as Limit Priors of Parametric Models

We conclude this section by mentioning that one of the prime reasons for focusing current machine learning interest on GP models was a highly original different way of establishing a weight space view proposed in [42]. Consider a model

$$f(\boldsymbol{x}) = \sum_{j=1}^{H} v_j h(\boldsymbol{x}; \boldsymbol{u}^{(j)})$$

which could be a multi-layer perceptron (MLP) with hidden layer functions h, weights $\boldsymbol{u}^{(j)}$ and output layer weights $\boldsymbol{v}$. Suppose that $\boldsymbol{u}^{(j)}$ have independent identical priors s.t. the resulting $h(\boldsymbol{x}; \boldsymbol{u}^{(j)})$ are bounded almost surely over a compact region of interest. Also, $v_j \sim N(0, \omega^2/H)$ independently. Then, for $H \to \infty$, $f(\boldsymbol{x})$ converges in quadratic mean to a zero-mean GP with covariance function $\omega^2 \mathrm{E}_{\boldsymbol{u}}[h(\boldsymbol{x}; \boldsymbol{u}) h(\boldsymbol{x}'; \boldsymbol{u})]$. Stronger conditions would assure almost sure convergence uniformly over a compact region. The bottom line is that if we take a conventional parametric model which linearly combines the outputs of a large number of feature detectors, and if we scale the outputs s.t. each of them in isolation has only a negligible contribution to the response, we might just as well use the corresponding Gaussian process model. Neal [42] also shows that if a non-zero number of the non-Gaussian feature outputs have a significant impact on the response with non-zero probability, then the limit process is typically not Gaussian.

To conclude, the weight space view seems to relate non-parametric GP models with parametric linear models fairly directly. However, there are important differences in general. Neal showed that GPs are obtained as limit distributions of large linear combinations of features if each feature's contribution becomes negligible, while the output distributions of architectures which fit at least a few strong feature detectors are typically not Gaussian. Predictions from a GP model are smoothed versions of the data (in a sense made concrete in Section 6), *i.e.* interpolate by minimising general smoothness constraints encoded in the GP prior, as opposed to parametric models which predict by focusing on these functions (within the family) which are most consistent with the data. O'Hagan [45] discusses differences w.r.t. optimal design.

---

[13] Steinwart [75] gives the following example. For a universal covariance function (most kernels discussed here have this property), any two finite disjoint subsets of $\mathcal{X}$ can be separated by a hyperplane in feature space, and the distances of all points to the plane can be made to lie in an interval of arbitrarily small size. Steinwart concludes that "any finite dimensional interpretation of the geometric situation in a feature space of a universal kernel must fail". We strongly agree.

[14] For example, in Section 5 we discuss K's role as reproducing kernel, in the sense that $K(\boldsymbol{x}, \boldsymbol{x}') = (K(\cdot, \boldsymbol{x}), K(\cdot, \boldsymbol{x}'))_K$ in some Hilbert space with inner product $(\cdot, \cdot)_K$. We could define Φ to map $\boldsymbol{x} \mapsto K(\cdot, \boldsymbol{x})$ and use the Hilbert space as weight space.

3 Some Gaussian Process Models

The simplest Gaussian process model is useful for regression estimation:

$$y = u + \varepsilon,$$

where $u = u(\boldsymbol{x})$ is *a priori* a zero-mean Gaussian process with covariance function K and ε is independent $N(0, \sigma^2)$ noise. Inference for this model is simple and analytically tractable, because the observation process $y(\boldsymbol{x})$ is zero-mean Gaussian with covariance $K(\boldsymbol{x}, \boldsymbol{x}') + \sigma^2 \delta_{\boldsymbol{x},\boldsymbol{x}'}$.[15] Given some i.i.d. data $S = \{(\boldsymbol{x}_i, y_i) \mid i = 1, \dots, n\}$, let $\boldsymbol{K} = (K(\boldsymbol{x}_i, \boldsymbol{x}_j))_{i,j}$. Then, $P(\boldsymbol{u}) = N(\boldsymbol{0}, \boldsymbol{K})$ and

$$P(\boldsymbol{u}|S) = N\left(\boldsymbol{K}(\sigma^2\boldsymbol{I} + \boldsymbol{K})^{-1}\boldsymbol{y}, \sigma^2(\sigma^2\boldsymbol{I} + \boldsymbol{K})^{-1}\boldsymbol{K}\right), \tag{5}$$

where $\boldsymbol{u} = (u(\boldsymbol{x}_i))_i$. For some test point $\boldsymbol{x}_*$ distinct from the training points, $u_* = u(\boldsymbol{x}_*) \perp \boldsymbol{y} \mid \boldsymbol{u}$, so that

$$\begin{aligned} P(u_*|\boldsymbol{x}_*, S) &= \int P(u_*|\boldsymbol{x}_*, \boldsymbol{u}) P(\boldsymbol{u}|S)\, d\boldsymbol{u} \\ &= N\left(u_* | \boldsymbol{k}(\boldsymbol{x}_*)^T(\sigma^2\boldsymbol{I} + \boldsymbol{K})^{-1}\boldsymbol{y}, K(\boldsymbol{x}_*, \boldsymbol{x}_*) - \boldsymbol{k}(\boldsymbol{x}_*)^T(\sigma^2\boldsymbol{I} + \boldsymbol{K})^{-1}\boldsymbol{k}(\boldsymbol{x}_*)\right). \end{aligned}$$

Here, $\boldsymbol{k}(\boldsymbol{x}_*) = (K(\boldsymbol{x}_i, \boldsymbol{x}_*))_i$. We see that for this model, the *posterior predictive process* $u(\boldsymbol{x})$ given S is Gaussian with mean function $\boldsymbol{y}^T(\sigma^2\boldsymbol{I} + \boldsymbol{K})^{-1}\boldsymbol{k}(\boldsymbol{x})$ and covariance function

$$K(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{k}(\boldsymbol{x})^T(\sigma^2\boldsymbol{I} + \boldsymbol{K})^{-1}\boldsymbol{k}(\boldsymbol{x}').$$

Note that the mean function used for prediction is linear in the targets $\boldsymbol{y}$ for every fixed $\boldsymbol{x}_*$. Furthermore, the posterior covariance function does not depend on the targets at all.

In practice, if only posterior mean predictions are required, the prediction vector $\boldsymbol{\xi} = (\sigma^2\boldsymbol{I} + \boldsymbol{K})^{-1}\boldsymbol{y}$ can be computed using a linear conjugate gradients solver which runs in $O(n^2)$ if the eigenvalue spectrum of $\boldsymbol{K}$ shows a fast decay. If predictive variances for many test points are required, the Cholesky decomposition[16] $\sigma^2\boldsymbol{I}+\boldsymbol{K} = \boldsymbol{L}\boldsymbol{L}^T$ should be computed, after which each variance computation requires a single back-substitution.

The pointwise predictive variance is never larger than the corresponding prior variance, but the shrinkage decreases with increasing noise level σ^2. The same result can be derived in the weight space view with $u(\boldsymbol{x}) = \Phi(\boldsymbol{x})^T\boldsymbol{\beta}$, applying the standard derivation of Bayesian linear regression (e.g., [85]). Note that just as in parametric linear regression, the smoothed prediction $\mathrm{E}[\boldsymbol{u}|S]$ is a linear function of the observations $\boldsymbol{y}$, as is the mean function of the predictive process $\mathrm{E}[u(\boldsymbol{x})|S]$ (see also Section 8). Note also that if $K(\boldsymbol{x}, \boldsymbol{x}') \to 0$ as $\|\boldsymbol{x} - \boldsymbol{x}'\|$ gets big, predictive mean and variance for points $\boldsymbol{x}$ far from all data tend to prior mean 0 and prior variance $K(\boldsymbol{x}, \boldsymbol{x})$. Second-level inference problems such as selecting values for hyperparameters (parameters of K and σ^2) or integrating them out are not analytically

[15] In the context of this model, it is interesting to note that if K' is stationary and continuous everywhere except at $\boldsymbol{0}$, it is the sum of a continuous (stationary) covariance K and a white noise covariance $\propto \delta_{\boldsymbol{x},\boldsymbol{x}'}$. Furthermore, Schönberg conjectured that if K' is an isotropic bounded covariance function, it must be continuous except possibly at $\boldsymbol{0}$.

[16] A symmetric matrix is positive definite iff it has a (unique) *Cholesky decomposition* $\boldsymbol{L}\boldsymbol{L}^T$, where $\boldsymbol{L}$ is lower triangular with positive diagonal elements.

tractable and approximations have to be applied. Approximate model selection is discussed in Section 4.

We can generalise this model by allowing for an arbitrary "noise distribution" $P(y|u)$, retaining the GP prior on $u(\boldsymbol{x})$. The generative view is to sample the process $u(\cdot)$ from the prior, then $y_i \sim P(y_i|u(\boldsymbol{x}_i))$ independent from each other given $u(\cdot)$.[17] The likelihood function factors as a product of univariate terms:

$$P(\boldsymbol{y}|\boldsymbol{X}, u(\cdot)) = P(\boldsymbol{y}|\boldsymbol{u}) = \prod_{i=1}^{n} P(y_i|u_i). \tag{6}$$

Since the likelihood depends on $u(\cdot)$ only via the finite set $\boldsymbol{u}$, the predictive posterior process can be written as

$$dP(u(\cdot)|S) = \frac{P(\boldsymbol{u}|S)}{P(\boldsymbol{u})} dP(u(\cdot)), \tag{7}$$

i.e. $P(u(X)|S) = (P(\boldsymbol{u}|S)/P(\boldsymbol{u}))P(u(X))$ for any finite $X \subset \mathcal{X}$. The prior measure is "shifted" by multiplication with $P(\boldsymbol{u}|S)/P(\boldsymbol{u})$ depending on the process values $\boldsymbol{u}$ at the training points only. The predictive process is *not* Gaussian in general, but its mean and covariance function can be obtained from knowledge of the posterior mean and covariance matrix of $P(\boldsymbol{u}|S)$ as discussed in Section 4. For a test point $\boldsymbol{x}_*$,

$$P(y_*|\boldsymbol{x}_*, S) = \mathrm{E}\left[P(y_*|u_*)\right]$$

where the expectation is over the predictive distribution of $u_* = u(\boldsymbol{x}_*)$. In this general model, first-level inference is not analytically tractable. In Section 4 a general approximate inference framework is discussed. Markov Chain Monte Carlo (MCMC) methods can be applied fairly straightforwardly, for example by Gibbs sampling from the latent variables $\boldsymbol{u}$ [43]. Such methods are attractive because the marginalisation over hyperparameters can be dealt with in the same framework. However, naive realisations may have a prohibitive running time due to the large number of correlated latent variables, and more advanced techniques can be difficult to handle in practice. While MCMC is maybe the most advanced and widely used class of approximate inference techniques, it is not discussed in any further detail here (see [41] for a review).

3.1 Generalised Linear Models. Binary Classification

A large class of models of this kind is obtained by starting from *generalised linear models (GLMs)* [44, 37] and replacing the parametric linear function $\boldsymbol{x}^T\boldsymbol{\beta}$ by a process $u(\boldsymbol{x})$ with a GP prior. This can be seen as direct infinite-dimensional generalisation of GLMs by employing the weight space view (see Section 2). In the spline smoothing context, this framework is presented in detail in [18]. It employs noise distributions

$$P(y|u) = \exp\left(\phi^{-1}(y\,u - Z(u)) + c(y, \phi)\right),$$

i.e. $P(y|u)$ is in an exponential family with natural parameter u, sufficient statistics y/ϕ and log partition function $\phi^{-1}Z(u)$. Here, $\phi > 0$ is a scale hyperparameter. The linear model is

[17] This is generalised easily to allow for bounded linear functionals of the latent process $u(\cdot)$ instead of the evaluation functional $\delta_{\boldsymbol{x}_i}$, as discussed in Section 5.

a special case with $\phi = \sigma^2$, $u = \mu = \mathrm{E}_u[y]$ and $Z(u) = (1/2)u^2$. A technically attractive feature of this framework is that $\log P(y|u)$ is strictly concave in u, leading to a strictly log-concave, unimodal posterior $P(\boldsymbol{u}|S)$. For binary classification and $y \in \{-1, +1\}$, the GLM for the binomial noise distribution is *logistic regression* with the *logit* noise

$$P(y|u) = \sigma(y\,(u+b)),\ \sigma(t) = \frac{1}{1+e^{-t}}. \tag{8}$$

Here, $\phi = 2$ and $Z(u) = 2\log\cosh((u+b)/2)$. Another frequently used binary classification noise model is *probit* noise

$$P(y|u) = \Phi(y\,(u+b)) = \mathrm{E}_{\tau\sim N(0,1)}\left[\mathrm{I}_{\{y(u+b)+\tau>0\}}\right] \tag{9}$$

which can be seen as noisy Heaviside step and is not in the exponential family. Both noise models (8), (9) are strictly log-concave.

3.2 Models with C Latent Processes

We can also allow for a fixed number $C \geq 1$ of latent variables for each case $(\boldsymbol{x}, \boldsymbol{y})$, i.e. C processes $u_c(\boldsymbol{x})$. The likelihood factors as

$$\prod_{i=1}^{n} P(\boldsymbol{y}_i|\boldsymbol{u}^{(i)}), \quad \boldsymbol{u}^{(i)} = (u_c(\boldsymbol{x}_i))_c.$$

$u_c(\boldsymbol{x})$ is zero-mean Gaussian *a priori* with covariance function $K^{(c)}$. While it is theoretically possible to use cross-covariance functions for prior covariances between u_c for different c, it may be hard to come up with a suitable class of such functions.[18] Furthermore, the assumption that the processes u_c are independent *a priori* leads to large computational savings, since the joint covariance matrix over the data assumes block-diagonal structure. Note that in this structure, we separate w.r.t. different c, while in block-diagonal structures coming from the factorised likelihood we separate w.r.t. cases i.

An important example using C latent processes is C-class classification. The likelihood comes from a multinomial GLM (or multiple logistic regression). It is convenient to use a binary encoding for the class labels, i.e. $\boldsymbol{y} = \boldsymbol{\delta}_c$ for class $c \in \{1, \dots, C\}$.[19] The noise is multinomial with

$$\boldsymbol{\mu} = \mathrm{E}[\boldsymbol{y} \mid \boldsymbol{u}] = \mathrm{softmax}(\boldsymbol{u}) = \left(\boldsymbol{1}^T \exp(\boldsymbol{u})\right)^{-1} \exp(\boldsymbol{u}).$$

$\boldsymbol{u} \mapsto \boldsymbol{\mu}$ is sometimes called *softmax* mapping. Note that this mapping is not invertible, since we can add $\alpha\boldsymbol{1}$ to $\boldsymbol{u}$ for any α without changing $\boldsymbol{\mu}$. In other words, the parameterisation of the multinomial by $\boldsymbol{u}$ is overcomplete, due to the linear constraint $\boldsymbol{y}^T\boldsymbol{1} = 1$ on $\boldsymbol{y}$, and the corresponding GLM log partition function

$$Z(\boldsymbol{u}) = \log \boldsymbol{1}^T \exp(\boldsymbol{u})$$

is not strictly convex. The usual remedy is to constrain $\boldsymbol{u}$ by for example fixing $u_C = 0$. This is fine in the context of fitting parameters by maximum likelihood, but may be problematic

[18] Hyperparameters may be shared between the prior processes, making them marginally dependent.

[19] We use vector notation for $\boldsymbol{u}$, $\boldsymbol{y} \in \mathbb{R}^C$ associated with a single case. This should not be confused with the vector notation $\boldsymbol{u}$, $\boldsymbol{y} \in \mathbb{R}^n$ used above to group variables for all cases.

for Bayesian inference. As mentioned above, we typically use priors which are i.i.d. over the u_c, so if we fix $u_C = 0$, the induced prior on $\boldsymbol{\mu}$ is not an exchangeable distribution (i.e. component permutations of $\boldsymbol{u}$ can have different distributions) and μ_C is singled out for no other than technical reasons. We think it is preferable in the Bayesian context to retain symmetry and accept that $\boldsymbol{u} \mapsto \boldsymbol{\mu}$ is not 1-to-1. Dealing with this non-identifiability during inference approximations is not too hard since softmax is invertible on any plane orthogonal to $\mathbf{1}$ and $Z(\boldsymbol{u})$ is strictly convex on such. Anyway, this detail together with the two different blocking structures mentioned above renders implementations of approximate inference for the C-class model somewhat more involved than the binary case (see [86] for an example). Other examples for C-process models are ordinal regression ("ranking") models (see [37] for likelihood suggestions) or multivariate regression.

3.3 Robust Regression

GP regression with Gaussian noise can lead to poor results if the data is prone to outliers, due to the light tails of the noise distribution. A robust GP regression model can be obtained by using a heavy-tailed noise distribution $P(y|u)$ such as a Laplace or even Student-t distribution. An interesting idea is to use the fact that the latter is obtained by starting with $N(0, \tau^{-1})$ and to integrate out the precision τ over a Gamma distribution (e.g., [42]). Thus, a robust model can be written as

$$y = u + \varepsilon, \quad \varepsilon \sim N(0, \tau^{-1}),$$

where τ is drawn i.i.d. from a Gamma distribution (whose parameters are hyperparameters). The posterior $P(\boldsymbol{u}|S, \boldsymbol{\tau})$ conditioned on the precision values τ_i is Gaussian and is computed in the same way as for the case $\tau_i = \sigma^{-2}$ above. $\boldsymbol{\tau}$ can be sampled by MCMC, or may be chosen to maximise the posterior $P(\boldsymbol{\tau}|S)$. The marginal likelihood $P(\boldsymbol{y}|\boldsymbol{\tau})$ is Gaussian and can be computed easily. However, note that in the latter case the number of hyperparameters grows as n which might invalidate the usual justification of marginal likelihood maximisation (see Section 4).

4 Approximate Inference and Learning

We have seen in the previous section that the posterior process for a likelihood of the general form (6) can be written as "shifted" version (7) of the prior. About the only processes (in this context) which can be dealt with feasibly are Gaussian ones, and a general way of obtaining a GP approximation to the posterior process is to approximate $P(\boldsymbol{u}|S)$ by a Gaussian $Q(\boldsymbol{u})$,[20] leading to the process

$$dQ(u(\cdot)) = \frac{Q(\boldsymbol{u})}{P(\boldsymbol{u})} dP(u(\cdot)) \tag{10}$$

which is Gaussian (recall from Section A.1 that this is a concise way of writing that $Q(u(X)) = (Q(\boldsymbol{u})/P(\boldsymbol{u}))P(u(X))$ for every finite $X \subset \mathcal{X}$). An optimal way of choosing Q would be to minimise the relative entropy (Definition 1)

$$\mathrm{D}[P(u(\cdot)|S) \,\|\, Q(u(\cdot))] = \mathrm{D}[P(\boldsymbol{u}|S) \,\|\, Q(\boldsymbol{u})]. \tag{11}$$

[20] The conditioning on S in $Q(\cdot)$ is omitted for notational simplicity.

The equality is intuitively clear, since $Q(u(\cdot))$, $P(u(\cdot)|S)$ and $P(u(\cdot))$ are the same conditional on $\boldsymbol{u}$. Formally, it follows from the fact that if $dP(u(\cdot)|S) \ll dQ(u(\cdot))$, then

$$dP(u(\cdot)|S) = \frac{P(\boldsymbol{u}|S)}{Q(\boldsymbol{u})}\, dQ(u(\cdot)),$$

and otherwise $\mathrm{D}[P(\boldsymbol{u}|S) \,\|\, Q(\boldsymbol{u})] = \infty$ (recall our notation from Section A.1). At the minimum point (unique w.r.t. f.d.d.'s of Q) Q and $P(\cdot|S)$ have the same mean and covariance function. This is equivalent to moment matching and requires us to find mean and covariance matrix of $P(\boldsymbol{u}|S)$. Unfortunately, this is intractable in general for large datasets and non-Gaussian noise. Any other Gaussian approximation $Q(\boldsymbol{u})$ leads to a GP posterior approximation $Q(u(\cdot))$, and the intractable (11) can nevertheless be valuable as guideline.

Here, we are primarily interested in approximate inference methods for GP models which employ GP approximations (10) to posterior processes via

$$Q(\boldsymbol{u}) = N(\boldsymbol{u} \,|\, \boldsymbol{K}\boldsymbol{\xi}, \boldsymbol{A}). \tag{12}$$

Here, $\boldsymbol{\xi}$, $\boldsymbol{A}$ can depend on the data S, the covariance function K (often via the kernel matrix $\boldsymbol{K}$) and on other hyperparameters. This class contains a variety of methods proposed in the literature. Virtually all of these have a reduced $O(n)$ parameterisation, since $\boldsymbol{A}$ has the restricted form

$$\boldsymbol{A} = \left(\boldsymbol{K}^{-1} + \boldsymbol{I}_{\cdot,I}\boldsymbol{D}\boldsymbol{I}_{I,\cdot}\right)^{-1} \tag{13}$$

with $\boldsymbol{D} \in \mathbb{R}^{d,d}$ diagonal with positive entries and $I \subset \{1, \dots, n\}$, $|I| = d$. For the methods mentioned below in this section, $d = n$ and $\boldsymbol{I}_{\cdot,I} = \boldsymbol{I}$, but for sparse GP approximations (e.g., [13, 78, 31]) we have $d \ll n$. In the latter case, $\boldsymbol{\xi}_{\setminus I} = \boldsymbol{0}$ and we use $\boldsymbol{\xi} \in \mathbb{R}^d$ for simplicity, replacing $\boldsymbol{\xi}$ in (12) by $\boldsymbol{I}_{\cdot,I}\boldsymbol{\xi}$.

From (10), the (approximate) predictive posterior distribution of $u_* = u(\boldsymbol{x}_*)$ at a test point $\boldsymbol{x}_*$ is determined easily as $Q(u_*|\boldsymbol{x}_*, S) = N(u_*|\mu(\boldsymbol{x}_*), \sigma^2(\boldsymbol{x}_*))$, where

$$\begin{aligned} \mu(\boldsymbol{x}_*) &= \boldsymbol{k}_I(\boldsymbol{x}_*)^T\boldsymbol{\xi}, \\ \sigma^2(\boldsymbol{x}_*) &= K(\boldsymbol{x}_*, \boldsymbol{x}_*) - \boldsymbol{k}_I(\boldsymbol{x}_*)^T\boldsymbol{D}^{1/2}\boldsymbol{B}^{-1}\boldsymbol{D}^{1/2}\boldsymbol{k}_I(\boldsymbol{x}_*), \\ \boldsymbol{B} &= \boldsymbol{I} + \boldsymbol{D}^{1/2}\boldsymbol{K}_I\boldsymbol{D}^{1/2}. \end{aligned} \tag{14}$$

Here, $\boldsymbol{k}_I(\boldsymbol{x}_*) = (K(\boldsymbol{x}_i, \boldsymbol{x}_*))_{i \in I}$. More generally, the GP posterior approximation has mean function $\mu(\boldsymbol{x})$ and covariance function

$$K(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{k}_I(\boldsymbol{x})^T\boldsymbol{D}^{1/2}\boldsymbol{B}^{-1}\boldsymbol{D}^{1/2}\boldsymbol{k}_I(\boldsymbol{x}').$$

The predictive distribution $P(y_*|\boldsymbol{x}_*, S)$ is obtained by averaging $P(y_*|u_*)$ over $N(u_*|\mu(\boldsymbol{x}_*), \sigma^2(\boldsymbol{x}_*))$. If this expectation is not analytically tractable, it can be done by Gaussian quadrature (e.g., [54], Sect. 4.5) if $P(y_*|u_*)$ is smooth and does not grow faster than polynomial.

A simple and numerically stable way to determine the predictive variances is to compute the Cholesky decomposition $\boldsymbol{B} = \boldsymbol{L}\boldsymbol{L}^T$ after which each variance requires one back-substitution with $\boldsymbol{L}$. It is important to stress that while inference approximation in GP models often boils down to simple linear algebra, it is crucial in practice to choose representations and procedures which are numerically stable. In the presence of positive definite matrices, techniques based on the Cholesky factorisation are known to be most stable.[21] Furthermore, in our representation $\boldsymbol{B}$ is well-conditioned since all its eigenvalues are ≥ 1.

We will refer to $\boldsymbol{\xi}$ as *prediction vector*. More generally, as mentioned in Section 2, we can use derivative information or other bounded linear functionals of the latent process $u(\boldsymbol{x})$ in the likelihood and/or for the variables to be predicted, using the fact that the corresponding finite set of scalar variables is multivariate Gaussian with prior covariance matrix derived from the covariance function K (as discussed in more detail in Section 5).

A generalisation to the multi-process models of Section 3 is also straightforward in principle. Here, $\boldsymbol{u}$ has dimension $C\,n$. Again $\boldsymbol{A}$ is restricted to the form (13), although $\boldsymbol{D}$ is merely block-diagonal with n $(C \times C)$ blocks on the diagonal. Moreover, if the processes are *a priori* independent, both $\boldsymbol{K}$ and $\boldsymbol{K}^{-1}$ consist of C $(n \times n)$ blocks on the diagonal. The general formulae for prediction (14) have to be modified for efficiency. The details are more involved and may depend on the concrete approximation method, C-process models are not discussed in further detail here.

4.1 Some Examples

A simple and efficient way of obtaining a Gaussian approximation $Q(\boldsymbol{u}|S)$ is via Laplace's method (also called *saddle-point approximation*), as proposed in [86] for binary classification with logit noise (8). To this end, we have to find the posterior mode $\hat{\boldsymbol{u}}$ which can be done by a variant of Newton-Raphson (or Fisher scoring, see [37]). Each iteration consists of a weighted regression problem, i.e. requires the solution of an $n \times n$ positive definite linear system. This can be done approximately in $O(n^2)$ using a conjugate gradients solver. At the mode, we have

$$\boldsymbol{\xi} = \boldsymbol{Y}\sigma(-\boldsymbol{Y}\hat{\boldsymbol{u}}), \quad \boldsymbol{D} = (\operatorname{diag}\sigma(-\boldsymbol{Y}\hat{\boldsymbol{u}}))(\operatorname{diag}\sigma(\boldsymbol{Y}\hat{\boldsymbol{u}})), \tag{15}$$

where σ is the logistic function (8) and $\boldsymbol{Y} = \operatorname{diag}\boldsymbol{y}$. All n diagonal elements of $\boldsymbol{D}$ are positive. Recall that the Laplace approximation replaces the log posterior by a quadratic fitted to the local curvature at the mode $\hat{\boldsymbol{u}}$. For the logit noise the log posterior is strictly concave and dominated by the Gaussian prior far out, so in general a Gaussian approximation should be fairly accurate. On the other hand, the true posterior is significantly skewed, meaning that the mode can be quite distant from the mean (which would be optimal) and the covariance approximation via local curvature around the mode can be poor.

The expectation propagation (EP) algorithm [39] for GP models can significantly outperform the Laplace GP approximation in terms of prediction accuracy, but is also more costly.[22] It is also somewhat harder to ensure numerical stability. On the other hand, EP is more general and can for example deal with discontinuous or non-differentiable log likelihoods. In fact, the special case of EP for Gaussian fields has been given earlier by Opper and Winther [48] under the name ADATAP, and EP can be seen as an iterative generalization of older Bayesian online learning techniques.

[21] Matrix inversion is often recommended in the GP machine learning literature. It is well known in numerical mathematics that inversion should be avoided whenever possible for reasons of stability, and in the context of our GP framework using a Cholesky decomposition is even more efficient.

[22] Partly due to its more complex iterative structure, but also because its elementary steps are smaller than for the Laplace technique and cannot be vectorised as efficiently.

A range of different variational approximations have been suggested in [16, 65, 24]. Note that for the variational method where $Q(\boldsymbol{u}|S)$ is chosen to minimise $\mathrm{D}[\cdot \,\|\, P(\boldsymbol{u}|S)]$, it is easy to see that the best Gaussian variational distribution has a covariance matrix of the form (13) (e.g., [64], Sect. 5.2.1).

Sparse approximations to GP inference are developed in [12, 13, 31]. While the original application was online learning, they are understood easier as "sparsifications" of EP (or ADATAP). While the approximations mentioned so far have training time scaling of $O(n^3)$, sparse inference approximations reduce this scaling to $O(n\,d^2)$ with adjustable $d \ll n$. For many problems, sparse approximations attain sufficient accuracy in essentially linear time in n which allows the application in data-rich settings. The idea is to concentrate on a subset $I \subset \{1,\dots,n\}$, $|I| = d$ of the training data which we call the active set, then to approximate the true likelihood $P(\boldsymbol{y}|\boldsymbol{u})$ of the model by a *likelihood approximation* $Q(\boldsymbol{u}_I)$ which is a function of the components $\boldsymbol{u}_I$ only. With this replacement, inference becomes linear in n (as can be seen from the formulae in this section which allow the use of an active set). The challenge is how to choose I and the form for $Q(\boldsymbol{u}_I)$ in a way to best approximate the moments of the true posterior $P(\boldsymbol{u}|\boldsymbol{y})$, while staying within the resource limitations of $O(n\,d^2)$ time and $O(n\,d)$ memory.[23] Also, if $P(\boldsymbol{y}|\boldsymbol{u})$ is not Gaussian, the sparse technique has to be embedded in an inference approximation of the kind discussed in this section. Details on some sparse schemes can be found in [78, 13, 31], some generic schemes based on the EP algorithm and information-theoretic selection heuristics for I are described in [63]. Free Matlab software has been released by Lehel Csató.[24]

4.2 Model Selection

So far we have only been concerned with first-level inference conditioned on fixed hyperparameters. A useful general method has to provide some means to select good values for these parameters or to marginalise over them (see Section 3). The latter is the correct way to proceed in a strict Bayesian sense and can be approximated by MCMC techniques, but often *model selection* is computationally more attractive. A frequently used general *empirical Bayesian* method for marginalising over nuisance hyperparameters is *marginal likelihood maximisation* or *maximum likelihood II* (also called *evidence maximisation*). This technique can be applied to the generic GP approximation described in this section, leading to a powerful generic way of adjusting hyperparameters via nonlinear optimization which scales linearly in the number of parameters. It is important to point out that such automatic model selection techniques are a strong advantage of Bayesian GP methods over other kernel machines such as SVMs (see Section 7) for which we do not know of selection strategies of similar power and generality.

If we denote the hyperparameters by $\boldsymbol{\alpha}$, the marginal likelihood is $P(S|\boldsymbol{\alpha}) = P(\boldsymbol{y}|\boldsymbol{\alpha})$, where the latent "primary" parameters $\boldsymbol{u}$ have been integrated out. If S is sufficiently large and $\boldsymbol{\alpha}$ of rather small fixed dimension, the hyperposterior $P(\boldsymbol{\alpha}|S)$ frequently is highly concentrated around a mode $\hat{\boldsymbol{\alpha}}$. Instead of using $P(\boldsymbol{\alpha}|S)$ to marginalise over $\boldsymbol{\alpha}$, we replace the posterior by $\delta_{\hat{\boldsymbol{\alpha}}}(\boldsymbol{\alpha})$, thus simply plug in $\hat{\boldsymbol{\alpha}}$ for $\boldsymbol{\alpha}$. This is an example of a *maximum a pos-*

[23] Choosing I completely at random is possible, but performs poorly in situations such as classification where the influence of patterns on the posterior can be very different.

[24] See *http://www.kyb.tuebingen.mpg.de/bs/people/csatol/ogp/index.html.*

teriori (MAP) approximation.[25] Finding $\hat{\boldsymbol{\alpha}}$ basically amounts to maximising the marginal likelihood, because the hyperprior $P(\boldsymbol{\alpha})$ is of a simple form. Conditions under which the hyperposterior is sufficiently peaked are hard to come by in general and will usually be overrestrictive for realistic models.[26] Thus, while marginal likelihood maximisation does not solve the model selection problem in general, it has been shown to work well in many empirical studies featuring very different models, and its description as "plug-in" approximation to Bayesian marginalisation may lead to successful extensions in cases where the simple method fails.

Some readers might worry at this point that we propose to select $\boldsymbol{\alpha}$ by maximising the likelihood $P(\boldsymbol{y}|\boldsymbol{\alpha})$, and maximum likelihood techniques are prone to overfitting. The key difference is that in the marginal likelihood, the primary "parameter" $u(\cdot)$ has been integrated out. While choosing primary parameters so as to maximise the likelihood often leads to overcomplicated fits that generalise badly, this is not true in general for marginal likelihood maximisation. A simple argument (yet not a proof) is that a value $\boldsymbol{\alpha}^{(1)}$ leading to very complicated $u(\cdot)$ needs to assign mass $P(u(\cdot)|\boldsymbol{\alpha})$ to many more functions than a value $\boldsymbol{\alpha}^{(2)}$ leading to simple $u(\cdot)$ (e.g. linear or low-order polynomial), so even if the likelihood of $\boldsymbol{y}$ is much higher for some of the complicated $u(\cdot)$, in the process of marginalisation the complicated functions are downweighted stronger in the integral for $P(\boldsymbol{y}|\boldsymbol{\alpha}^{(1)})$ than are the simpler functions in the integral for $P(\boldsymbol{y}|\boldsymbol{\alpha}^{(2)})$. This "Occam razor" effect has been analysed by MacKay [33]. However it is obviously possible to create situations in which marginal likelihood maximisation still leads to overfitting.[27] As a general rule of thumb, the dimensionality of the hyperparameters $\boldsymbol{\alpha}$ should not scale with n,[28] and the Occam razor argument just given should intuitively apply to the situation (once more, we do not know of a definite test separating admissable from non-admissable cases in general).

We will focus on marginal likelihood maximisation as general model selection technique. The log marginal likelihood $\log P(\boldsymbol{y}|\boldsymbol{\alpha})$ is as difficult to compute as the posterior $P(\boldsymbol{u}|S,\boldsymbol{\alpha})$ and has to be approximated in general.[29] It is easy to see that the variational lower bound

$$\begin{aligned}\log P(\boldsymbol{y}|\boldsymbol{\alpha}) &\geq \mathrm{E}_Q\left[\log P(\boldsymbol{y}|\boldsymbol{u},\boldsymbol{\alpha}) + \log P(\boldsymbol{u}|\boldsymbol{\alpha})\right] + \mathrm{H}[Q(\boldsymbol{u})] \\ &= \mathrm{E}_Q\left[\log P(\boldsymbol{y}|\boldsymbol{u},\boldsymbol{\alpha})\right] - \mathrm{D}[Q(\boldsymbol{u})\,\|\,P(\boldsymbol{u}|\boldsymbol{\alpha})].\end{aligned} \tag{16}$$

holds for any distribution $Q(\boldsymbol{u})$ (recall relative and differential entropy from Section A.2). The slack in the bound is the relative entropy $\mathrm{D}[Q(\boldsymbol{u})\,\|\,P(\boldsymbol{u}|S,\boldsymbol{\alpha})]$. Note that the posterior approximation $Q(\boldsymbol{u})$ depends on $\boldsymbol{\alpha}$ as well, but it is not feasible in general to obtain its exact gradient w.r.t. $\boldsymbol{\alpha}$. Variational EM, an important special case of a lower bound maximisation algorithm is iterative, in turn freezing one of Q, $\boldsymbol{\alpha}$ and maximising the lower bound w.r.t. the other (here, Q can be chosen from a family of variational distributions). Alternatively,

[25] Multimodality in the hyperposterior can arise from non-identifiability of the model though symmetries in $\boldsymbol{\alpha}$, i.e. there exist different $\boldsymbol{\alpha}^{(1)}$, $\boldsymbol{\alpha}^{(2)}$ s.t. $P(\boldsymbol{y}|\{\boldsymbol{x}_i\},\boldsymbol{\alpha}^{(1)}) \approx P(\boldsymbol{y}|\{\boldsymbol{x}_i\},\boldsymbol{\alpha}^{(2)})$ for datasets of interest. In this case, we can just pick any of the dominant modes $\hat{\boldsymbol{\alpha}}$ in the hyperposterior to arrive at the same predictions as if we had chosen a peak train featuring all equivalent modes.

[26] Since we integrate out a variable $\boldsymbol{u}$ of the same dimension of the training sample and the latter is independent only conditional on the process $u(\cdot)$ (which is not in general a finite-dimensional variable), we cannot use the central limit theorem directly to assert Gaussianity of $P(\boldsymbol{y}|\boldsymbol{\alpha})$ as n gets large.

[27] For example, one could maliciously set $\boldsymbol{\alpha} = u(\cdot)$.

[28] Although in special situations the technique may still be applicable, see [78] or Section 3.3.

[29] It is analytically tractable for a Gaussian likelihood, for example in the case of GP regression with Gaussian noise discussed above it is $\log N(\boldsymbol{y}|\boldsymbol{0},\boldsymbol{K}+\sigma^2\boldsymbol{I})$.

Q can be chosen in a different way as approximation of the posterior $P(\boldsymbol{u}|S)$ (for example using the EP algorithm or sparse approximations). The deviation from the variational choice of Q (i.e. the one which maximises the lower bound over a family of candidates) can be criticised on the ground that other choices of Q can lead to decreases in the lower bound, so the overall algorithm does not increase its criterion strictly monotonically. On the other hand, Q chosen in a different way may lie outside families over which the lower bound can be maximised efficiently, thus may even result in a larger value than the variational maximiser within the family.[30] Furthermore, the lower bound criterion can be motivated by the fact that its gradient

$$\mathrm{E}_{Q(\boldsymbol{u})}\left[\nabla_{\boldsymbol{\alpha}} \log P(\boldsymbol{y}, \boldsymbol{u}|\boldsymbol{\alpha})\right]$$

(ignoring the dependence of Q on $\boldsymbol{\alpha}$) approximates the true gradient

$$\nabla_{\boldsymbol{\alpha}} \log P(\boldsymbol{y}|\boldsymbol{\alpha}) = \mathrm{E}_{P(\boldsymbol{u}|S)}\left[\nabla_{\boldsymbol{\alpha}} \log P(\boldsymbol{y}, \boldsymbol{u}|\boldsymbol{\alpha})\right]$$

at every point $\boldsymbol{\alpha}$.

We close by mentioning an interesting point in which lower bound maximisation for GP models might deviate from the usual practice with parametric architectures. For the latter, it is customary to maximise the lower bound w.r.t. $\boldsymbol{\alpha}$ while keeping Q *completely* fixed (the gradient of Q w.r.t. $\boldsymbol{\alpha}$ is ignored). This makes sense as long as Q is independent of the prior distribution in the model, but in the context of approximate GP inference methods, the dependence of $Q(\boldsymbol{u})$ on the GP prior (thus on $\boldsymbol{\alpha}$) is quite explicit (for example, the covariance of Q is $(\boldsymbol{K}^{-1} + \boldsymbol{D})^{-1}$ which depends strongly on the kernel matrix $\boldsymbol{K}$, since $\boldsymbol{D}$ is merely a diagonal matrix). We argue that instead of keeping all of Q fixed during the maximisation for $\boldsymbol{\alpha}$, we should merely ignore the dependence of the essential parameters $\boldsymbol{\xi}$, $\boldsymbol{D}$ on $\boldsymbol{\alpha}$.[31] This typically leads to a more involved gradient computation which is potentially closer to the true gradient. Alternatively, if this computation is beyond resource limits, further indirect dependencies on $\boldsymbol{\alpha}$ may be ignored. We remark that the optimisation problem is slightly non-standard due to the lack of strict monotonicity, and given optimisers have to be modified to take this into account. Details can be found in [63], Sect. 4.5.3.

5 Reproducing Kernel Hilbert Spaces

The theory of *reproducing kernel Hilbert spaces (RKHS)* can be used to characterise the space of random variables obtained as bounded linear functionals of a GP on which any method of prediction from finite information must be based. Apart from that, RKHS provide a unification of ideas from a wide area of mathematics, most of which will not be mentioned here. The interested reader may consult [3]. Our exposition is taken from [80]. This section can be skipped by readers interested primarily in practical applications.

A *reproducing kernel Hilbert space (RKHS)* $\mathcal{H}$ is a Hilbert space of functions $\mathcal{X} \to \mathbb{R}$ for which all evaluation functionals $\delta_{\boldsymbol{x}}$ are bounded. This implies that there exists a kernel

[30] For example, even though the bound maximiser over all Gaussians has a covariance matrix of the form (13), finding it is prohibitively costly in practice and proposed variational schemes [16, 65, 24] use restricted subfamilies.

[31] There is no simple analytic formula for this dependence, so we cannot do better than ignoring it.

$K(\boldsymbol{x}, \boldsymbol{x}')$ s.t. $K(\cdot, \boldsymbol{x}) \in \mathcal{H}$ for all $\boldsymbol{x} \in \mathcal{X}$ and

$$f(\boldsymbol{x}) = \delta_{\boldsymbol{x}} f = (K(\cdot, \boldsymbol{x}), f) \tag{17}$$

for all $f \in \mathcal{H}$, where $(\cdot, \cdot)$ is the inner product in $\mathcal{H}$. To be specific, a Hilbert space is a vector space with an inner product which is complete, in the sense that each Cauchy sequence converges to an element of the space. For example, a Hilbert space $\mathcal{H}$ can be generated from an inner product space of functions $\mathcal{X} \to \mathbb{R}$ by adjoining the limits of all Cauchy sequences to $\mathcal{H}$. Note that this is a rather abstract operation and the adjoined objects need not be functions in the usual sense. For example, $\mathcal{L}_2(\mu)$ is obtained by completing the vector space of functions for which

$$\int f(\boldsymbol{x})^2 \, d\mu(\boldsymbol{x}) < \infty \tag{18}$$

and can be shown to contain "functions" which are not defined pointwise.[32] For an RKHS $\mathcal{H}$ such anomalies cannot occur, since the functionals $\delta_{\boldsymbol{x}}$ are bounded:[33]

$$|f(\boldsymbol{x})| = |\delta_{\boldsymbol{x}} f| \le C_{\boldsymbol{x}} \|f\|.$$

By the Riesz representation theorem (e.g., [20]) there exists a unique *representer* $K_{\boldsymbol{x}} \in \mathcal{H}$ such that (17) holds with $K(\cdot, \boldsymbol{x}) = K_{\boldsymbol{x}}$. It is easy to see that the kernel K is positive semidefinite. K is called *reproducing kernel (RK)* of $\mathcal{H}$, note that

$$\big(K_{\boldsymbol{x}}, K_{\boldsymbol{x}'}\big) = \big(K(\cdot, \boldsymbol{x}), K(\cdot, \boldsymbol{x}')\big) = K(\boldsymbol{x}, \boldsymbol{x}').$$

It is important to note that in a RKHS, (norm) convergence implies pointwise convergence to a pointwise defined function, since

$$|f_m(\boldsymbol{x}) - f(\boldsymbol{x})| = |(K_{\boldsymbol{x}}, f_m - f)| \le C_{\boldsymbol{x}} \|f_m - f\|.$$

On the other hand, for any positive semidefinite K there exists a unique RKHS $\mathcal{H}$ with RK K. Namely, the set of finite linear combinations of $K(\cdot, \boldsymbol{x}_i)$, $\boldsymbol{x}_i \in \mathcal{X}$ with

$$\left(\sum_i a_i K(\cdot, \boldsymbol{x}_i), \sum_j b_j K(\cdot, \boldsymbol{x}'_j) \right) = \sum_{i,j} a_i b_j K(\boldsymbol{x}_i, \boldsymbol{x}'_j)$$

is an inner product space which is extended to a Hilbert space $\mathcal{H}$ by adjoining all limits of Cauchy sequences. Since norm convergence implies pointwise convergence in the inner product space, all adjoined limits are pointwise defined functions and $\mathcal{H}$ is an RKHS with RK K. To conclude, a RKHS has properties which make it much "nicer" to work with than a general Hilbert space. All functions are pointwise defined, and the representer of the evaluation functional $\delta_{\boldsymbol{x}}$ is explicitly given by $K(\cdot, \boldsymbol{x})$.

[32] The existence of such functions in $\mathcal{L}_2(\mu)$ means that expressions such as (18) have to be interpreted with some care. Each element $f \in \mathcal{L}_2(\mu)$ can be defined as the set of all equivalent Cauchy sequences which define f (two Cauchy sequences are equivalent if the sequence obtained by interleaving them is Cauchy as well). An expression $E(f, g)$ should then be understood as the limit $\lim_{n\to\infty} E(f_n, g_n)$ where $f_n \to f$, $g_n \to g$, etc. The existence of the limit has to be established independently. In the sequel, we will always use this convention.

[33] Bounded functionals are also called *continuous*.

5.1 RKHS by Mercer Eigendecomposition. Karhunen-Loeve Expansion

We have already mentioned that $\mathcal{L}_2(\mu)$ is not a RKHS in general, but for many kernels K it contains a (unique) RKHS as subspace. Recall that $\mathcal{L}_2(\mu)$ contains all functions $f : \mathcal{X} \to \mathbb{R}$ for which (18) holds. The standard inner product is

$$(f, g) = \int f(\boldsymbol{x})g(\boldsymbol{x})\, d\mu(\boldsymbol{x}).$$

Often, μ is taken as indicator function of a compact set such as the unit hypercube. A positive semidefinite $K(\boldsymbol{x}, \boldsymbol{x}')$ can be regarded as kernel (or representer) of a positive semidefinite linear operator $\mathcal{K}$ in the sense

$$(\mathcal{K}f)(\boldsymbol{x}) = (K(\cdot, \boldsymbol{x}), f).$$

ϕ is an eigenfunction of K with eigenvalue $\lambda \neq 0$ if

$$(\mathcal{K}\phi)(\boldsymbol{x}) = (K(\cdot, \boldsymbol{x}), \phi) = \lambda\, \phi(\boldsymbol{x}).$$

For K, all eigenvalues are real and non-negative. Furthermore, suppose K is continuous and

$$\int K(\boldsymbol{x}, \boldsymbol{x}')^2\, d\mu(\boldsymbol{x})d\mu(\boldsymbol{x}') < \infty.$$

Then, by the Mercer-Hilbert-Schmidt theorems there exists a countable orthonormal sequence of continuous eigenfunctions $\phi_\nu \in \mathcal{L}_2(\mu)$ with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$, and K can be expanded in terms of these:

$$K(\boldsymbol{x}, \boldsymbol{x}') = \sum_{\nu \geq 1} \lambda_\nu \phi_\nu(\boldsymbol{x})\phi_\nu(\boldsymbol{x}'), \tag{19}$$

and $\sum_{\nu\geq 1} \lambda_\nu^2 < \infty$, thus $\lambda_\nu \to 0 (\nu \to \infty)$. This can be seen as generalisation of the eigendecomposition of a positive semidefinite Hermitian matrix. Indeed, the reproducing property of positive semidefinite kernels was recognised and used by E. H. Moore [40] to develop the notion of general "positive Hermitian matrices". In this case, we can characterise the RKHS embedded in $\mathcal{L}_2(\mu)$ explicitly. For $f \in \mathcal{L}_2(\mu)$, define the Fourier coefficients

$$f_\nu = (f, \phi_\nu).$$

Consider the subspace $\mathcal{H}_K$ of all $f \in \mathcal{L}_2(\mu)$ with $\sum_{\nu\geq 1} \lambda_\nu^{-1} f_\nu^2 < \infty$. Then, $\mathcal{H}_K$ is a Hilbert space with inner product

$$(f, g)_K = \sum_{\nu \geq 1} \frac{f_\nu g_\nu}{\lambda_\nu},$$

moreover the Fourier series $\sum_{\nu\geq 1} f_\nu \phi_\nu$ converges pointwise to f.[34] Since $\{\lambda_\nu \phi_\nu(\boldsymbol{x})\}$ are the Fourier coefficients of $K(\cdot, \boldsymbol{x})$ (using Equation 19), we have

$$(f, K(\cdot, \boldsymbol{x}))_K = \sum_{\nu \geq 1} f_\nu \phi_\nu(\boldsymbol{x}) = f(\boldsymbol{x}),$$

[34] In particular, f is defined pointwise.

thus K is the RK of $\mathcal{H}_K$. It is important to distinguish clearly between the inner products $(\cdot,\cdot)$ in $\mathcal{L}_2(\mu)$ and $(\cdot,\cdot)_K$ in $\mathcal{H}_K$ (see [89] for more details about the relationship of these inner products). While $\|\cdot\|$ measures "expected squared distance" from 0 (w.r.t. $d\mu$), $\|\cdot\|_K$ is a measure of the "roughness" of a function. For example, the eigenfunctions have $\|\phi_\nu\| = 1$, but $\|\phi_\nu\|_K = \lambda_\nu^{-1/2}$ thus becoming increasingly rough.[35]

The spectral decomposition of K leads to an important representation of a zero-mean GP $u(\boldsymbol{x})$ with covariance function K: the *Karhunen-Loeve expansion*. Namely, the sequence

$$u_k(\boldsymbol{x}) = \sum_{\nu=1}^{k} u_\nu \phi_\nu(\boldsymbol{x}), \tag{20}$$

where u_ν are independent $N(0,\lambda_\nu)$ variables, converges to $u(\boldsymbol{x})$ in quadratic mean (a stronger statement under additional conditions can be found in [2]). Moreover,

$$u_\nu = \int u(\boldsymbol{x})\phi_\nu(\boldsymbol{x})\, d\mu(\boldsymbol{x})$$

which is well defined in quadratic mean. We have already used this expansion in Section 2 to introduce the "weight space view". Note that since the variances λ_ν decay to 0, the GP can be approximated by finite partial sums of the expansion (see [89]).

5.2 Duality between RKHS and Gaussian Process

If $u(\boldsymbol{x})$ is a zero-mean GP with covariance function K, what is the exact relationship between $u(\boldsymbol{x})$ and the RKHS with RK K? One might think that $u(\boldsymbol{x})$ can be seen as distribution over $\mathcal{H}_K$, but this is wrong (as pointed out in [80], Sect. 1.1). In fact, for any version of $u(\boldsymbol{x})$ sample functions from the process are *not* in $\mathcal{H}_K$ with probability 1! This can be seen by noting that for the partial sums (20) we have

$$\mathrm{E}\left[\|u_k\|_K^2\right] = \mathrm{E}\left[\sum_{\nu=1}^{k} \frac{u_\nu^2}{\lambda_\nu}\right] = k \to \infty \ (k \to \infty).$$

Roughly speaking, $\mathcal{H}_K$ contains "smooth", non-erratic functions from $\mathcal{L}_2(\mu)$, characteristics we cannot expect from sample paths of a random process. A better intuition about $\mathcal{H}_K$ is that it will turn out to contain expected values of $u(\boldsymbol{x})$ conditioned on a finite amount of information, thus the posterior mean functions we are interested in.

The following duality between $\mathcal{H}_K$ and a Hilbert space based on $u(\boldsymbol{x})$ was noticed in [27] and is important in the context of theoretical analyses. Namely, construct a Hilbert space $\mathcal{H}_{GP}$ in the same way as above starting from positive semidefinite K, but replace $K(\cdot,\boldsymbol{x}_i)$ by $u(\boldsymbol{x}_i)$ and use the inner product

$$(A,B)_{GP} = \mathrm{E}[AB],$$

thus

$$\left(\sum_i a_i u(\boldsymbol{x}_i), \sum_j b_j u(\boldsymbol{x}'_j)\right)_{GP} = \sum_{i,j} a_i b_j K(\boldsymbol{x}_i,\boldsymbol{x}'_j).$$

[35] In the same sense as high-frequency components in the usual Fourier transform.

$\mathcal{H}_{GP}$ is a space of random variables, not functions, but it is isometrically isomorphic to $\mathcal{H}_K$ under the mapping $u(\boldsymbol{x}_i) \mapsto K(\cdot, \boldsymbol{x}_i)$, with

$$(u(\boldsymbol{x}), u(\boldsymbol{x}'))_{GP} = \mathrm{E}[u(\boldsymbol{x})u(\boldsymbol{x}')] = K(\boldsymbol{x}, \boldsymbol{x}') = (K(\cdot, \boldsymbol{x}), K(\cdot, \boldsymbol{x}'))_K.$$

For most purposes, we can regard $\mathcal{H}_{GP}$ as RKHS with RK K. The space $\mathcal{H}_{GP}$ is important in the context of inference on GP models we are interested in, because it contains exactly the random variables we condition on or would like to predict in situations where only a finite amount of information is available (from observations which are linear functionals of the process).

If L is a bounded linear functional on $\mathcal{H}_K$, it has a representer $\nu \in \mathcal{H}_K$ with $\nu(\boldsymbol{x}) = LK_{\boldsymbol{x}}$. The isometry maps ν to a random variable $Z \in \mathcal{H}_{GP}$ which we formally denote by $Lu(\cdot)$. Note that

$$\mathrm{E}\left[(Lu(\cdot))u(\boldsymbol{x})\right] = (\nu, K_{\boldsymbol{x}})_K = \nu(\boldsymbol{x}) = LK_{\boldsymbol{x}}.$$

More generally, if $L^{(1)}, L^{(2)}$ are functionals with representers $\nu^{(1)}$, $\nu^{(2)}$ s.t. $\boldsymbol{x} \mapsto L^{(j)}K_{\boldsymbol{x}}$ are in $\mathcal{H}_K$, then

$$\mathrm{E}\left[(L^{(1)}u(\cdot))(L^{(2)}u(\cdot))\right] = (\nu^{(1)}, \nu^{(2)})_K = L^{(1)}_{\boldsymbol{x}}(K(\cdot, \boldsymbol{x}), \nu^{(2)})_K = L^{(1)}_{\boldsymbol{x}} L^{(2)}_{\boldsymbol{y}} K(\boldsymbol{x}, \boldsymbol{y}).$$

Again, it is clear that $Lu(\cdot)$ is (in general) very different from the process obtained by applying L to sample paths of $u(\boldsymbol{x})$. In fact, since the latter are almost surely not in $\mathcal{H}_K$, L does not even apply to them in general. The correct interpretation is in quadratic mean, using the isometry between $\mathcal{H}_{GP}$ and $\mathcal{H}_K$. As an example, suppose that $\mathcal{X} = \mathbb{R}^g$ and $L = D_{\boldsymbol{x}}$ is a differential functional evaluated at $\boldsymbol{x}$. Then, we retrieve the observations in Section 2 about derivatives of a GP.

6 Penalised Likelihood. Spline Smoothing

The GP models we are interested in here have their origin in spline smoothing techniques and penalised likelihood estimation, and for low-dimensional input spaces spline kernels are widely used due to the favourable approximation properties of splines and computational advantages. A comprehensive account of spline smoothing and relations to Bayesian estimation in GP models is [80] which our exposition is mainly based on. Spline smoothing is a special case of penalised likelihood methods, giving another view on the reproducing kernel via the Green's function of a penalisation (or regularisation) operator which will be introduced below. This section can be skipped by readers interested primarily in practical applications.

In Section 5 we have discussed the duality between a Gaussian process and the RKHS of its covariance function. Apart from the Bayesian viewpoint using GP models, a different and more direct approach to estimation in non-parametric models is the penalised likelihood approach, the oldest and most widely used incarnations of which are spline smoothing methods. We will introduce the basic ideas for the one-dimensional model which leads to the general notion of regularisation operators, penalty functionals and their connections to RKHS. We omit all details, (important) computational issues and multidimensional generalisations, see [80] for details. A more elementary account is [18].

We will only sketch the ideas, for rigorous details see [80, 27]. Interpolation and smoothing by splines originates from the work of Schönberg [61]. A *natural spline* $s(x)$ of order m on $[0,1]$ is defined based on *knots* $0 < x_1 < \cdots < x_n < 1$. If π^k denotes the set of polynomials of order $\leq k$, then $s(x) \in \pi^{2m-1}$ on $[x_i, x_{i+1}]$, $s(x) \in \pi^{m-1}$ on $[0, x_1]$ and on $[x_n, 1]$, and $s \in C^{2m-2}$ overall. *Natural cubic splines* are obtained for $m = 2$. Define the *roughness penalty*

$$J_m(f) = \int_0^1 \left(f^{(m)}(x)\right)^2 \, dx.$$

$J_m(f)$ penalises large derivatives of order m by a large value, for example J_2 is large for functions of large curvature. Then, for some fixed function values the interpolant minimising $J_m(f)$ over all f for which the latter is defined is a spline of order m. More precisely, $f \in \mathcal{W}_m[0,1]$, a so-called *Sobolev space* of all $f \in C^{m-1}[0,1]$ s.t. $f^{(m-1)}$ is absolutely continuous on $[0,1]$. If we consider the related *smoothing* problem of minimising the penalised empirical risk

$$\sum_{i=1}^n (y_i - f(x_i))^2 + \alpha J_m(f), \quad f \in \mathcal{W}_m[0,1], \tag{21}$$

it is clear that the minimiser is again a natural spline $s(x)$ of order m (any other $f \in \mathcal{W}_m[0,1]$ can be replaced by the spline with the same values at the knots, this does not change the risk term and cannot increase J_m). Now, from Taylor's theorem:

$$f(x) = \sum_{\nu=0}^{m-1} \frac{x^\nu}{\nu!} f^{(\nu)}(0) + \int_0^1 G_m(x,t) f^{(m)}(t) \, dt$$

with $G_m(x,t) = (x-t)_+^{m-1}/(m-1)!$ (here, $u_+ = u\mathrm{I}_{\{u \geq 0\}}$). If $f^{(\nu)}(0) = 0$, $\nu = 0, \dots, m-1$ then $(G_m(x,\cdot), D^m f) = f(x)$, thus $G_m(x,t)$ is the *Green's function* for the boundary value problem $D^m f = g$. These functions f form a Hilbert space with inner product

$$(f,g)_K = \int_0^1 f^{(m)}(t) g^{(m)}(t) \, dt$$

which is a RKHS with RK

$$K(x,x') = \int_0^1 G_m(x,t) G_m(x',t) \, dt. \tag{22}$$

It is interesting to note that a zero-mean GP with covariance function K can be obtained as $(m-1)$-fold integrated Wiener process (introduced in Section 2.3). Let $W(x)$ be a Wiener process on $[0,1]$ with $W(0) = 0$ a.s. and $\mathrm{E}[W(1)^2] = 1$ (its covariance function is $\min\{x, x'\}$). It is possible to define a stochastic integral against a process with independent increments.[36] The process $u(x)$ defined via the stochastic integral

$$u(x) = \int G_m(x,t) \, dW(t)$$

[36]See [19], Sect. 9.4 for an easy derivation. It is important to note that the stochastic integral is *not* the random variable arising from integrating over sample paths of the process, the latter integrals do not exist in many cases in which the stochastic integral can be constructed.

is a zero-mean GP with covariance function K. If W is chosen s.t. its sample paths are continuous, $u(x)$ is in $\mathcal{W}_m[0,1]$ and $u^{(\nu)}(0) = 0$ for $\nu < m$. Since $dG_m/dx = G_{m-1}$ and $G_1(x,t) = \mathrm{I}_{\{x>t\}}$, $u^{(m-1)}$ and W are m.s. equivalent. Note that $u(x)$ can be written as

$$u(x) = \int_0^x dW(t) \int_t^x dx_1 \ldots \int_{x_{m-2}}^x dx_{m-1}$$

for $m > 1$.

The boundary values can be satisfied by taking the direct sum of the space with π^{m-1}. The latter is trivially an RKHS w.r.t. an inner product of choice: choose an orthonormal basis and define the kernel to be the sum of outer products of the basis functions. The kernel for the direct sum is the sum of K and the finite-dimensional kernel. Note that $\|\cdot\|_K$ is only a seminorm on the full space because $\|p\|_K = 0$ for $p \in \pi^{m-1}$.

We only sketch the general case, see [80, 53] for details. We make use of the following duality between a RKHS and a *regularisation (pseudodifferential) operator* $\mathcal{P}$ on $\mathcal{L}_2(\mu)$. Let $\mathcal{H}$ be the Hilbert space of f s.t. $\mathcal{P}f \in \mathcal{L}_2(\mu)$. For $\mathcal{P}$, consider the operator[37] $\mathcal{P}^*\mathcal{P}$. If this has a null space (such as π^{m-1} in the example above), we restrict $\mathcal{H}$ to the orthogonal complement. Now, the operator is positive definite and has an inverse (its *Green's function*) whose kernel K is the RK of $\mathcal{H}$.[38] The inner product is

$$(f, g)_K = (\mathcal{P}f, \mathcal{P}g)$$

and the penalty functional is simply the squared RKHS norm. If $G(\boldsymbol{t}, \boldsymbol{u})$ exists s.t. $(G(\boldsymbol{t}, \cdot), \mathcal{P}f) = f(\boldsymbol{t})$ for all $f \in \mathcal{H}$, the RK is given by

$$K(\boldsymbol{s}, \boldsymbol{t}) = (G(\boldsymbol{s}, \cdot), G(\boldsymbol{t}, \cdot)).$$

On the other hand, we can start from an RKHS with RK K and derive the corresponding regularisation operator $\mathcal{P}$. This can give additional insight into the meaning of a covariance function (see [53, 70]). In fact, if K is stationary and continuous, we can use Bochner's theorem (2). Namely, if $f(\boldsymbol{\omega})$ is the spectral density of K, we can take $f(\boldsymbol{\omega})^{-1/2}$ as spectrum of $\mathcal{P}$.[39] The one-dimensional example above is readily generalised to splines on the unit sphere or to *thin plate splines* in $\mathcal{X} = \mathbb{R}^g$, but the details get quite involved (see [80], Chap. 2).

Kimeldorf and Wahba [27] generalised this setup to a general variational problem in an RKHS, allowing for general bounded linear functionals $L_i f$ instead of $f(x_i)$ in (21). The minimiser is determined by $n+M$ coefficients, where M is the dimension of the null space of the differential operator $\mathcal{P}$ associated with K ($M = m+1$ in the spline case above). These can be computed by direct formulae given in [80], Sect. 1.3. In the more general *penalised likelihood approach* [80, 18], n function values or linear functionals of f are used as latent variables in a likelihood (see Section 3), to obtain for example non-parametric extensions of GLMs [18]. The penalised likelihood is obtained by adding the penalty functional to the likelihood, and just as above the minimiser is determined by $n + M$ coefficients only (this *representer theorem* can be proved using the same argument as in the spline case above). In general, iterative methods are required to find values for these coefficients.

[37] $\mathcal{P}^*$ is the adjoint of $\mathcal{P}$, i.e. $(f, \mathcal{P}g) = (\mathcal{P}^* f, g)$.

[38] This construction via Green's functions is different from the one above involving $G_m(x,t)$. Without going into details, it may help to consider the analogue of the finite-dimensional case (vectors and matrices instead of functions and operators): $\boldsymbol{K} = (\boldsymbol{P}^T \boldsymbol{P})^{-1} = \boldsymbol{G}\boldsymbol{G}^T$ where $\boldsymbol{G} = \boldsymbol{P}^{-1}$.

[39] $\mathcal{P}$ is not uniquely defined, but only $\mathcal{P}^*\mathcal{P}$ (which has spectrum $f(\boldsymbol{\omega})^{-1}$).

6.1 Bayesian View on Spline Smoothing

We close this section by reviewing the equivalence between spline smoothing and Bayesian estimation for a GP model pointed out by Kimeldorf and Wahba [27]. Given a positive semidefinite kernel K corresponding to a pseudodifferential operator with M-dimensional null space, we can construct an RKHS $\mathcal{H}$ as follows. If $\mathcal{H}_0$ is the null space represented by an orthonormal basis p_ν and $\mathcal{H}_1$ the RKHS for K, let $\mathcal{H}$ be their direct sum. Consider the model

$$F(\boldsymbol{x}) = \sum_{\nu=1}^{M} \theta_\nu p_\nu(\boldsymbol{x}) + b^{1/2} u(\boldsymbol{x}), \quad y_i = F(\boldsymbol{x}_i) + \varepsilon_i,$$

where $u(\boldsymbol{x})$ is a zero-mean GP with covariance function K and ε_i are independent $N(0, \sigma^2)$. Furthermore, $\boldsymbol{\theta} \sim N(\mathbf{0}, a\boldsymbol{I})$ *a priori*. On the other hand, let f_λ be the minimiser in $\mathcal{H}$ of the regularised risk functional

$$\frac{1}{n} \sum_{i=1}^{n} (y_i - f(\boldsymbol{x}_i))^2 + \lambda \|\mathcal{P}_1 f\|^2_{\mathcal{H}_1},$$

where $\mathcal{P}_1$ is the orthogonal projection onto $\mathcal{H}_1$. Kimeldorf and Wahba [27] show that f_λ lies in the span of $\{p_\nu \,|\, \nu = 1, \dots, M\} \cup \{K(\cdot, \boldsymbol{x}_i) \,|\, i = 1, \dots, n\}$ and give a numerical procedure for computing the coefficients. If we define $\hat{F}_a(\boldsymbol{x}) = \mathrm{E}[F(\boldsymbol{x}) \,|\, y_1, \dots, y_n]$, then they show that

$$\lim_{a \to \infty} \hat{F}_a(\boldsymbol{x}) = f_\lambda(\boldsymbol{x}), \quad \lambda = \frac{\sigma^2}{n\, b}$$

for every fixed $\boldsymbol{x}$. The proof (see [80], Chap. 1) is a straightforward application of the duality between the RKHS $\mathcal{H}_1$ and the Hilbert space based on $u(\boldsymbol{x})$, as described in Section 5. The procedure of dealing with $\mathcal{H}_0$ and the improper prior on $\boldsymbol{\theta}$ is awkward but is not necessary if the RKHS $\mathcal{H}_1$ induced by K is rich enough.[40]

Finally, we note that a parametric extension of a non-parametric GP model can be sensible even if $\mathcal{H}_1$ is rich enough in principle, leading to *semiparametric models* (or *partial splines*). For details about such models, we refer to [18], Chap. 4 and [80], Chap. 6.

7 Maximum Entropy Discrimination. Large Margin Classifiers

We regard GPs as building blocks for statistical models in much the same way as a parametric family of distributions (see Section 3 for examples). Statistical methods to estimate unknown parameters in such models follow different paradigms, and in machine learning the following have been among the most popular.

1. Probabilistic Bayesian paradigm:
   This has been introduced in Section 3. As noted in Section 4, the (intractable) posterior process is typically approximated by a GP itself.

[40] This is not the case for spline kernels, for which $f \in \mathcal{H}_1$ is constrained by the boundary conditions.

2. Large margin (discriminative) paradigm:
   Here, a "posterior" process is obtained by associating margin constraints with observed data, then searching for a process which fulfils these (soft) constraints and at the same time is close to the prior GP, in a sense made concrete in this section. Since the constraints are linear in the latent outputs, the "posterior" process is always a GP with the same covariance as the prior.

The relationship between Bayesian methods and penalised likelihood or generalised spline smoothing methods has been discussed in Section 6. Large margin methods are special cases of spline smoothing models with a particular loss function which does not correspond to a probabilistic noise model (e.g., [81, 65, 73]). Several attempts have been made to express large margin discrimination methods as approximations to Bayesian inference (e.g., [73, 65, 64]), but the paradigm separation suggested in [25] seems somewhat more convincing.

The connection between these two paradigms has been formulated in [25], this section is based on their exposition. The large margin paradigm has been made popular by the empirical success of the *support vector machine (SVM)* (see [59, 8] for background material). In the Bayesian GP setting (see Section 3), the likelihood $P(\boldsymbol{y}|\boldsymbol{u})$ of the observed data $\boldsymbol{y}$ can be seen to impose "soft constraints" on the predictive distribution, in the sense that functions of significant probability under the posterior must not violate many of them strongly. In the large margin paradigm whose probabilistic view has been called *minimum relative entropy discrimination (MRED)* [25], such constraints are enforced more explicitly.[41] We introduce a set of latent *margin variables* $\boldsymbol{\gamma} = (\gamma_i)_i \in \mathbb{R}^n$, one for each datapoint. Along with the GP prior $P(u(\cdot))$ on the latent function, we choose a prior $P(\boldsymbol{\gamma})$ over $\boldsymbol{\gamma}$. The margin prior encourages large margins γ_i, as is discussed in detail below. The minimum relative entropy distribution $dQ(u(\cdot), \boldsymbol{\gamma})$ is defined as minimiser of $\mathrm{D}[Q \,\|\, P]$, subject to the *soft margin constraints*

$$E_{(u(\cdot),\boldsymbol{\gamma})\sim Q}\left[y_i u(\boldsymbol{x}_i) - \gamma_i\right] \geq 0, \; i = 1, \dots, n. \tag{23}$$

Just as in the case of a likelihood function, these constraints depend on the values $\boldsymbol{u} = (u(\boldsymbol{x}_i))_i$ of the random process $u(\cdot)$ only. It is well known in information theory (e.g., [22], Sect. 3.1) that the solution to this constrained problem is given by

$$dQ(u(\cdot), \boldsymbol{\gamma}) = Z(\boldsymbol{\lambda})^{-1} \exp\left(\sum_{i=1}^{n} \lambda_i \left(y_i u_i - \gamma_i\right)\right) dP(u(\cdot), \boldsymbol{\gamma}), \tag{24}$$

where

$$Z(\boldsymbol{\lambda}) = \mathrm{E}_{(u(\cdot),\boldsymbol{\gamma})\sim P}\left[\exp\left(\sum_{i=1}^{n} \lambda_i \left(y_i u_i - \gamma_i\right)\right)\right].$$

The value for the Lagrange multipliers $\boldsymbol{\lambda}$ is obtained by minimising the convex function $\log Z(\boldsymbol{\lambda})$ (sometimes called the *dual criterion*) under the constraints $\boldsymbol{\lambda} \geq \mathbf{0}$. Since the right hand side of (24) factorises between $u(\cdot)$ and $\boldsymbol{\gamma}$ and the same holds for the prior P, we see that Q must factorise in the same way. Furthermore, it is immediate from (24) that $Q(u(\cdot))$ is again a Gaussian process with the same covariance kernel K as $P(u(\cdot))$ and with mean

[41] For notational simplicity, we do not use a bias term b here. The modifications to do so are straightforward. In the original SVM formulation, b can be seen to have a uniform (improper) prior.

function $\mu(\boldsymbol{x}_*) = \boldsymbol{k}(\boldsymbol{x}_*)^T \boldsymbol{Y}\boldsymbol{\lambda}$, where $\boldsymbol{Y} = \mathrm{diag}(y_i)_i$. Due to the factorised form, we also have $Z(\boldsymbol{\lambda}) = Z_{u(\cdot)}(\boldsymbol{\lambda}) Z_{\boldsymbol{\gamma}}(\boldsymbol{\lambda})$ and

$$Z_{u(\cdot)}(\boldsymbol{\lambda}) = \mathrm{E}_{\boldsymbol{u}\sim P}\left[e^{\boldsymbol{\lambda}^T \boldsymbol{Y}\boldsymbol{u}}\right] = e^{\frac{1}{2}\boldsymbol{\lambda}^T \boldsymbol{Y}\boldsymbol{K}\boldsymbol{Y}\boldsymbol{\lambda}}.$$

The form of $Z_{\boldsymbol{\gamma}}$ depends on the choice of the prior $P(\boldsymbol{\gamma})$ on the margin variables. Jaakkola et. al. [25] give some examples of such priors which encourage large margins. For example, if $P(\boldsymbol{\gamma}) = \prod_i P(\gamma_i)$, then $P(\gamma_i)$ should drop quickly for $\gamma_i < 1$ in order to penalise small and especially negative margins (empirical errors). In order for (23) to be a "soft constraint" only w.r.t. margin violations and also to mimic the SVM situation, we have to use $P(\gamma_i) = 0$ for $\gamma_i > 1$.[42] If $P(\gamma_i) \propto e^{-c(1-\gamma_i)} \mathrm{I}_{\{\gamma_i \le 1\}}$, then

$$Z_{\boldsymbol{\gamma}}(\boldsymbol{\lambda}) \propto \prod_{i=1}^{n} \frac{e^{-\lambda_i}}{1 - \lambda_i/c},$$

and the complete dual criterion is

$$\log Z(\boldsymbol{\lambda}) = -\sum_{i=1}^{n} (\lambda_i + \log(1 - \lambda_i/c)) + \frac{1}{2}\boldsymbol{\lambda}^T \boldsymbol{Y}\boldsymbol{K}\boldsymbol{Y}\boldsymbol{\lambda}, \quad \boldsymbol{\lambda} \ge \boldsymbol{0}. \tag{25}$$

Except for the potential term $\log(1 - \lambda_i/c)$, this is identical to the SVM dual objective (see below).[43] The so-called *hard margin SVM* for which margin constraints are enforced without allowing for violations, is obtained for $c \to \infty$. It converges only if the training data is indeed separable and is prone to over-complicated solutions. The effect of the potential term on the solution is limited (see [25]). It keeps λ_i from saturating to c exactly (which happens in SVM for misclassified patterns). The dual criterion can be optimised using efficient algorithms such as SMO [52], although the nonlinear potential term introduces minor complications.[44] Just like in SVM, sparsity in $\boldsymbol{\lambda}$ is encouraged and can be observed in practice.

To conclude, MRED gives a complete probabilistic interpretation of the SVM, or at least of a close approximation thereof. Note that SVM classification cannot be seen as MAP approximation to Bayesian inference for a probabilistic model, because its loss function does not correspond to a proper negative log likelihood [65, 47, 73]. Interestingly, the MRED view points out limitations of this framework as opposed to a Bayesian treatment of a Gaussian process model with a proper likelihood. Recall from above that the margin constraints are linear in the latent outputs $\boldsymbol{u}$, leading to the fact that the MRED "posterior" process $Q(u(\cdot))$ has the *same* covariance kernel K as the prior. While the constraints enforce the predictive mean to move from 0 *a priori* to $\mu(\boldsymbol{x})$, the "predictive variances" are simply the prior ones, independent of the data. This suggests that if predictive variances (or error bars) are to be estimated besides simply performing a discrimination, then SVMs or other large margin discriminative methods may be less appropriate than probabilistic GP models. For more details on this argument, see [63], Sect. 4.7.2.

More important is the lack of practical methods for model selection with SVM. For Bayesian GP methods, a general model selection strategy is detailed in Section 4. Alternatively,

[42] As in the SVM setup, the choice of 1 as margin width is arbitrary, because the distance can be re-scaled in terms of the prior variance.

[43] The potential term acts like a logarithmic barrier to enforce the constraints $\lambda_i < c$ (e.g., [7]).

[44] SMO makes use of the fact that the SVM criterion is quadratic with linear constraints.

hyperparameters can be marginalised over approximately using MCMC techniques [43]. In contrast, model selection for SVM is typically done using variants of cross validation, which severely limits the number of free parameters that can be adapted.

While it is often claimed that learning-theoretical foundations count as distinctive advantage of SVM, similar or even superior guarantees can be given for approximate Bayesian GP techniques as well [67].

8 Kriging

An important and early application of Gaussian random field models has been termed *kriging* [35] after a South-African mining engineer D. Krige who developed methods for predicting spatial ore-grade distributions from sampled ore grades [30]. Optimal spatial linear prediction has its roots in earlier work by Wiener and Kolmogorov ("closeness in space" may have to be replaced by "closeness in time", since they were mainly concerned with time series). These fundamental ideas have been further developed in the fields of geostatistics [35] as kriging and in meteorology under the name *objective analysis* (see [11], Chap. 3 for references).

We will not go into any details, but refer to [11], Chap. 3 and [74] (we follow the latter here). The basic model is the same as for semiparametric smoothing:

$$z(\boldsymbol{x}) = \boldsymbol{m}(\boldsymbol{x})^T \boldsymbol{\beta} + \varepsilon(\boldsymbol{x})$$

where $\boldsymbol{m}(\boldsymbol{x})$ is a known feature map and $\varepsilon(\boldsymbol{x})$ is a zero-mean random field with covariance function K. In a nutshell, kriging is a minimum mean squared error prediction method for linear functionals of $z(\boldsymbol{x})$ given observations $\boldsymbol{z} = (z(\boldsymbol{x}_1), \dots, z(\boldsymbol{x}_n))^T$ at spatial locations $\boldsymbol{x}_i \in \mathbb{R}^g$. For example, if $z(\boldsymbol{x})$ measures ore grade at $\boldsymbol{x}$ one might be interested in predicting

$$\int_{\mathcal{B}} z(\boldsymbol{x})\, d\boldsymbol{x}$$

over some area $\mathcal{B} \subset \mathbb{R}^g$. Since they focus on m.s. error and m.s. properties of $z(\boldsymbol{x})$ in general, kriging methods typically depend on second-order properties of the process only, and $\varepsilon(\boldsymbol{x})$ is often assumed to be a Gaussian field. Furthermore, we restrict ourselves to linear predictors $\lambda_0 + \boldsymbol{\lambda}^T \boldsymbol{z}$. The optimal predictor of $z(\boldsymbol{x}_*)$ in the m.s. error sense is the conditional expectation which is linear in $\boldsymbol{z}$ if $\varepsilon(\boldsymbol{x})$ is Gaussian and $\boldsymbol{\beta}$ is known:

$$\boldsymbol{K}\boldsymbol{\lambda} = \boldsymbol{k}, \quad \lambda_0 = \left(\boldsymbol{m}(\boldsymbol{x}_*) - \boldsymbol{M}^T \boldsymbol{\lambda}\right)^T \boldsymbol{\beta}$$

where $\boldsymbol{K} = (K(\boldsymbol{x}_i, \boldsymbol{x}_j))_{i,j}$, $\boldsymbol{k} = (K(\boldsymbol{x}_i, \boldsymbol{x}_*))_i$ and $\boldsymbol{M} = (\boldsymbol{m}(\boldsymbol{x}_1), \dots, \boldsymbol{m}(\boldsymbol{x}_n))^T$. If $\boldsymbol{\beta}$ is unknown, a simple procedure is to plug in the generalised least squares estimate

$$\hat{\boldsymbol{\beta}} = \left(\boldsymbol{M}^T \boldsymbol{K}^{-1} \boldsymbol{M}\right)^{-1} \boldsymbol{M}^T \boldsymbol{K}^{-1} \boldsymbol{z}$$

for $\hat{\boldsymbol{\beta}}$. This procedure can be motivated from several angles. If we restrict our attention to linear predictors of $z(\boldsymbol{x}_*)$ which are *unbiased* in the sense

$$\mathrm{E}\left[\lambda_0 + \boldsymbol{\lambda}^T \boldsymbol{z}\right] = \lambda_0 + \boldsymbol{\lambda}^T \boldsymbol{M} \boldsymbol{\beta} = \mathrm{E}[z(\boldsymbol{x}_*)] = \boldsymbol{m}(\boldsymbol{x}_*)^T \boldsymbol{\beta}$$

for any $\boldsymbol{\beta}$, the suggested approach minimises the m.s. error over these unbiased predictors. It is therefore called *best linear unbiased predictor (BLUP)*. A Bayesian motivation can be constructed in the same way as mentioned in Section 6. Namely, $\boldsymbol{\beta}$ is given a Gaussian prior whose covariance matrix scales with $a > 0$ and $\varepsilon(\boldsymbol{x})$ is *a priori* Gaussian. Then, the posterior mean for $z(\boldsymbol{x}_*)$ converges to the BLUP as $a \to \infty$ (i.e. as the $\boldsymbol{\beta}$ prior becomes uninformative).

The equations behind the BLUP have been known long before and have been rediscovered in many areas of statistics. In practice, kriging methods are more concerned about inducing an appropriate covariance function (under the stationarity assumption) from observed data as well. The empirical *semivariogram* is a frequently used method for estimating the covariance function close to the origin. On the theoretical side, Stein [74] advocates the usefulness of fixed-domain asymptotics (a growing number of observations located within a fixed compact region) to understand the relationship between covariance model and behaviour of kriging predictors.[45] By Bochner's theorem (2) a stationary covariance function is characterised by its spectral distribution $F(\boldsymbol{\omega})$. Stein points out that fixed-domain asymptotics depend most strongly on the spectral masses for large $\|\boldsymbol{\omega}\|$, i.e. the high frequency components, much less so on the low frequency ones or the mean function $\boldsymbol{m}(\boldsymbol{x})^T\boldsymbol{\beta}$ (if $\boldsymbol{m}(\boldsymbol{x})$ is smooth itself, e.g. polynomials). Let $f(\boldsymbol{\omega})$ be the spectral density, i.e. the Fourier transform of $K(\boldsymbol{x})$. In general, the lighter the tails of $f(\boldsymbol{\omega})$ the smoother $\varepsilon(\boldsymbol{x})$ is in the m.s. sense. Stein advocates this expected smoothness as a central parameter of the GP prior and condemns the uncritical use of smooth (analytic) covariance functions such as the RBF (Gaussian) kernel (see Section 9). Another important concept highlighted by Stein (see also [80], Chap. 3) is the one of equivalence and orthogonality of GPs.[46] Essentially, GPs with covariance functions of different form can be equivalent in which case it is not possible to unambiguously decide for one of them even if an infinite amount of observations in a fixed region are given. On this basis, one can argue that for a parametric family of covariance functions inducing equivalent GPs the parameters can just as well be fixed *a priori* since their consistent estimation is not possible. On the other hand, parameters s.t. different values lead to orthogonal GPs should be learned from data and *not* be fixed *a priori*.

Note that kriging models are more generally concerned with *intrinsic random functions (IRF)* [36], generalisations of stationary processes which are also frequently used in the spline smoothing context. In a nutshell, a k-IRF $u(\boldsymbol{x})$ is a non-stationary random field based on a "spectral density" whose integral diverges on any neighborhood of the origin (e.g., has infinite pointwise variance). However, if $\boldsymbol{c} \in \mathbb{R}^n$ is a generalised divided difference (g.d.d.) for $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n$ in the sense that $\sum_i c_i p(\boldsymbol{x}_i) = 0$ for all polynomials p of total degree $\leq k$, then the variance of $\sum_i c_i u(\boldsymbol{x}_i)$ is finite and serves to define an "covariance function" $K(\boldsymbol{x})$ which is *k-conditionally positive semidefinite*, namely

$$\sum_{i,j=1}^{n} c_i c_j K(\boldsymbol{x}_i - \boldsymbol{x}_j) \geq 0$$

[45] Stein restricts his analysis to "interpolation", i.e. to situations where predictions are required only at locations which are in principle supported by observations (in contrast to "extrapolation" often studied in the time series context). This should not be confused with the distinction between interpolation and smoothing used in Section 6. All non-trivial kriging techniques are smoothing methods.

[46] Two probability measures are equivalent if they have the same null sets, i.e. are mutually absolutely continuous (see Section A.1). They are orthogonal if there is a null set of one of them which has mass 1 under the other. Gaussian measures are either orthogonal or equivalent.

for all g.d.d.'s $\boldsymbol{c}$. In practice, one uses semi-parametric models where the latent process of interest is the sum of a k-IRF and a polynomial of total degree $\leq k$ whose coefficients are parametric latent variables.[47]

In fact, IRFs do not add more generality w.r.t. high-frequency behaviour of the process since $f(\boldsymbol{\omega})$ must be integrable on the complement of any $\mathbf{0}$-neighborhood, so the IRF can be written as the uncorrelated sum of a stationary and a non-stationary part, the latter with $f(\boldsymbol{\omega}) = 0$ outside a $\mathbf{0}$-neighborhood (thus very smooth). IRFs are not discussed in any further detail here (see [36, 74]).

9 Choice of Kernel. Kernel Design

There is a tendency in the machine learning community to treat kernel methods as "black box" techniques, in the sense that covariance functions are chosen from a small set of candidates over and over again. If a family of kernels is used, it typically comes with a very small number of free parameters so that model selection techniques such as cross-validation can be applied. Even though such approaches work surprisingly well for many problems of interest in machine learning, experience almost invariably has shown that much can be gained by choosing or designing covariance functions carefully depending on known characteristics of a problem (for an example, see [59], Sect. 11.4).

Establishing a clear link between kernel functions and consequences for predictions is very non-trivial and theoretical results are typically asymptotic arguments. As opposed to finite-dimensional parametric models, the process prior affects predictions from a non-parametric model even in fixed-domain asymptotic situations (see Section 8). The sole aim of this section is to introduce a range of frequently used kernel functions and some of their characteristics, to give some methods for constructing covariance functions from simpler elements, and to show some techniques which can be used to obtain insight into the behaviour of the corresponding GP. Yaglom [88] gives extensive material, an accessible review is [1]. In the final part, we discuss some kernel methods over discrete spaces $\mathcal{X}$.

It should be noted that positive definiteness of an arbitrary symmetric form or function is hard to establish in general. For example, the sensible approach of constructing a distance $d(\boldsymbol{x}, \boldsymbol{x}')$ between patterns depending on prior knowledge, then proposing

$$K(\boldsymbol{x}, \boldsymbol{x}') = e^{-w\, d(\boldsymbol{x}, \boldsymbol{x}')^2} \tag{26}$$

as covariance function does not work in general because K need not be positive semidefinite, moreover there is no simple general criterion to prove that K is a covariance function.[48] If $d(\boldsymbol{x}, \boldsymbol{x}')$ can be represented in an Euclidean space, K is a kernel as we will see below. Note that if $K(\boldsymbol{x}, \boldsymbol{x}')$ of the form (26) is a kernel, so must be $K(\boldsymbol{x}, \boldsymbol{x}')^t$ for any $t > 0$.[49] Kernels with this property are called *infinitely divisible*. Schönberg [60] managed to characterise infinitely divisible kernels (26) by a property on $d(\boldsymbol{x}, \boldsymbol{x}')$ which unfortunately is just as hard to handle as positive semidefiniteness.[50]

[47] In fact, $\boldsymbol{m}(\boldsymbol{x})$ maps to a basis of π^k. As mentioned above, the BLUP is obtained as posterior expectation under an uninformative prior on the parametric coefficients.

[48] If $d(\boldsymbol{x}, \boldsymbol{x}')$ is stationary, one can try to compute the spectral density, but this will not be analytically tractable in general.

[49] This is true in general only for $t \in \mathbb{N}_{>0}$, see below.

[50] $-d(\boldsymbol{x}, \boldsymbol{x}')^2$ must be conditionally positive semidefinite of degree 0 (see Section 8).

9.1 Some Standard Kernels

In the following, we provide a list of frequently used "standard kernels". Most of these will have a *variance (scaling) parameter* $C > 0$ in practice, sometimes an *offset parameter* $v_b > 0$, thus instead of K one uses $C\,K$ or $C\,K + v_b$. C scales the variance of the process, while a $v_b > 0$ comes from the uncertainty of a bias parameter added to the process.[51] In applications where the kernel matrix $\boldsymbol{K}$ is used directly in linear systems, it is advised to add a *jitter term*[52] $\rho\delta_{\boldsymbol{x},\boldsymbol{x}'}$ to the kernel to improve the condition number of $\boldsymbol{K}$. This amounts to a small amount of additive white noise on $u(\boldsymbol{x})$ (ρ can be chosen quite small), but should not be confused with measurement noise which is modelled separately (see Section 3). These modifications are omitted in the sequel for simplicity.

The *Gaussian (RBF)* covariance function

$$K(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{w}{2}\|\boldsymbol{x} - \boldsymbol{x}'\|^2\right) \tag{27}$$

is isotropic for each $\mathcal{X} = \mathbb{R}^g$ (i.e. $\mathcal{D}^\infty$). $w > 0$ is an inverse squared length scale parameter, in the sense that $w^{-1/2}$ determines a scale on which $u(\boldsymbol{x})$ is expected to change significantly. $K(\boldsymbol{x})$ is analytic at $\boldsymbol{0}$, so $u(\boldsymbol{x})$ is m.s. analytic. Stein [74] points out that

$$\sum_{j=0}^{k} u^{(j)}(0)\frac{x^j}{j!} \to u(x)$$

in quadratic mean for every x (a similar formula holds for $\mathcal{X} = \mathbb{R}^g$), so that u can be predicted perfectly by knowing all its derivatives at $\boldsymbol{0}$ (which depend on u on an neighborhood of $\boldsymbol{0}$ only). He criticises the wide-spread use of the Gaussian covariance function because its strong smoothness assumptions are unrealistic for many physical processes, in particular predictive variances are often unreasonably small given data. The spectral density (in $\mathbb{R}$) is $f(\omega) = (2\pi w)^{-1/2}\exp(-\omega^2/(2w))$ with very light tails. On the other hand, Smola et. al. [70] recommend the use of the Gaussian covariance function for high-dimensional kernel classification methods because of the high degree of smoothness. It is interesting to note that in the context of using GPs for time series prediction, Girard et. al. [17] report problems with unreasonably small predictive variances using the Gaussian covariance function (although they do not consider other kernels in comparison). Figure 1 shows smoothed plots of some sample paths. Note the effect of the length scale $w^{-1/2}$ and the high degree of smoothness.

We can consider the anisotropic version, called *squared-exponential* covariance function:

$$K(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}')^T\boldsymbol{W}(\boldsymbol{x} - \boldsymbol{x}')\right). \tag{28}$$

Here, $\boldsymbol{W}$ is positive definite. Typically, $\boldsymbol{W}$ is a diagonal matrix with an inverse squared length scale parameter w_j for each dimension. Full matrices $\boldsymbol{W}$ have been considered in [53, 79], and factor analysis-type matrices $\boldsymbol{W}$ are a useful intermediate (e.g., [4, 65]). An important application of the additional d.o.f.'s in (28) as compared to the Gaussian kernel is automatic relevance determination (ARD), as discussed below. Note that the squared-exponential covariance function for diagonal $\boldsymbol{W}$ can be seen as product of g one-dimensional

[51] For reasons of numerical stability, v_b must not become too large.

[52] In the context of kriging (see Section 8), adding $\rho\delta_{\boldsymbol{x},\boldsymbol{x}'}$ has been proposed by Mathéron to model the so-called "nugget effect" (see [11], Sect. 2.3.1), but other authors have criticised this practice.

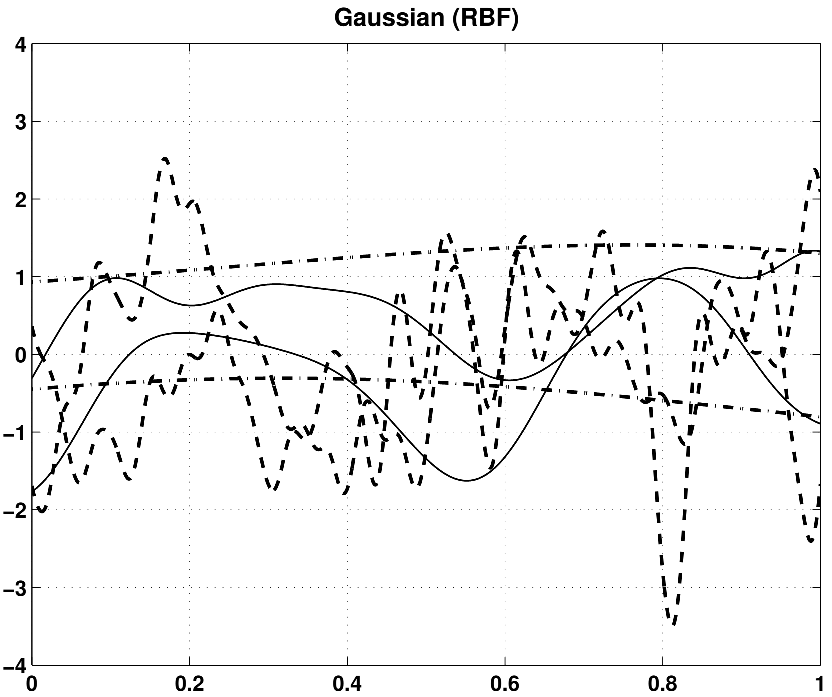

Figure 1: Smoothed sample paths from GP with Gaussian covariance function. All have variance $C = 1$. Dash-dotted: $w = 1$. Solid: $w = 10^2$. Dashed: $w = 50^2$.

Gaussian kernels with different length scales, so the corresponding RKHS is a tensor product space built from RKHS's for one-dimensional functions (see Section 5).

The *Matérn class* of covariance functions (also called *modified Bessel* covariance functions) is given by

$$K(\tau) = \frac{\pi^{1/2}}{2^{\nu-1}\Gamma(\nu+1/2)\alpha^{2\nu}}(\alpha\tau)^{\nu}K_{\nu}(\alpha\tau), \quad \tau = \|\boldsymbol{x} - \boldsymbol{x}'\|, \tag{29}$$

where $\nu > 0$, $\alpha > 0$ and $K_\nu(x)$ is a modified Bessel function (e.g., [74], Sect. 2.7). One can show that $z^\nu K_\nu(z) \to 2^{\nu-1}\Gamma(\nu)$ for $z \to 0$, so

$$K(0) = \frac{\pi^{1/2}\Gamma(\nu)}{\Gamma(\nu+1/2)\alpha^{2\nu}}.$$

K is isotropic for each $\mathcal{X} = \mathbb{R}^g$. An important feature of this class is that the m.s. smoothness of $u(\boldsymbol{x})$ can be regulated directly via ν. For example, $u(\boldsymbol{x})$ is m times m.s. differentiable iff $\nu > m$. The spectral density in $\mathbb{R}$ is $f(\omega) = (\alpha^2 + \omega^2)^{-\nu-1/2}$. For $\nu = 1/2 + m$ we obtain a process with rational spectral density, a continuous time analogue of an AR time series model. For $\nu = 1/2$, $K(\tau) \propto e^{-\alpha\tau}$ defines an *Ornstein-Uhlenbeck process*, a stationary analogue to the Wiener process which also has independent increments. In general, for $\nu = 1/2 + m$ we have $K(\tau) \propto e^{-\alpha\tau}p(\alpha\tau)$, where $p(x)$ is a polynomial of order m (e.g., [74], Sect. 2.7). Note that if $\alpha = (w(2\nu+1))^{1/2}$, then

$$\alpha^{2\nu+1}f(\omega) \to e^{-\omega^2/(2w)} \ (\nu \to \infty),$$

thus $K(\tau)$ converges to the Gaussian covariance function after appropriate re-scaling.

The Matérn class can be generalised to an anisotropic family in the same way as the Gaussian kernel. Figure 2 show some sample function plots for values $\nu = 1/2, 3/2, 5/2, 10$. Note the effect of ν on the roughness of the sample paths. For $\nu = 1/2$ the paths are erratic even though the length scale is 1, i.e. the same as the horizontal region shown. For $\nu = 3/2$, the process is m.s. differentiable, for $\nu = 5/2$ twice so.

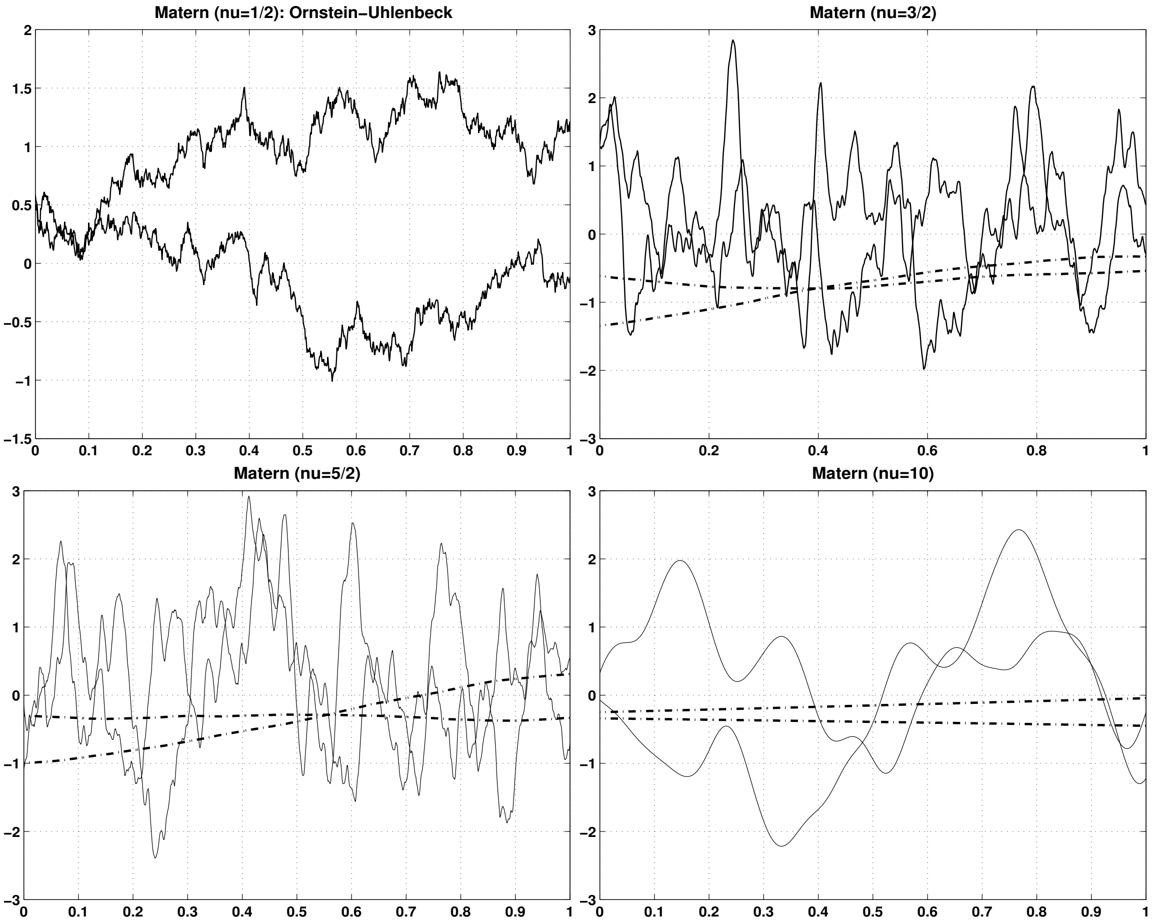

Figure 2: Smoothed sample paths from GP with Matérn covariance function. All have variance $C = 1$. Upper left: Ornstein-Uhlenbeck (Matérn, $\nu = 1/2$), $\alpha = 1$. Upper right: Matérn, $\nu = 3/2$, $\alpha = 1$ (dash-dotted), $\alpha = 10^2$ (solid). Lower left: Matérn, $\nu = 5/2$, $\alpha = 1$ (dash-dotted), $\alpha = 10^2$ (solid). Lower right: Matérn, $\nu = 10$, $\alpha = 1$ (dash-dotted), $\alpha = 10^2$ (solid).

The *exponential class* of covariance functions is given by

$$K(\tau) = e^{-\alpha\tau^\delta}, \ \delta \in (0, 2].$$

The positive definiteness can be proved using the Matérn class (see [74], Sect. 2.7). For $\delta = 1$, we have the Ornstein-Uhlenbeck covariance function, for $\delta = 2$ the Gaussian one. Although it seems that the kernel varies smoothly in δ, the processes have quite different properties in the regimes $\delta \in (0, 1)$, $\delta = 1$, $\delta \in (1, 2)$ and $\delta = 2$. Continuous sample paths can be ensured for any $\delta \in (0, 2]$, but differentiable sample paths can only be obtained for

$\delta = 2$ (in which case they are analytic).[53] $K(\tau)$ is not positive definite for $\delta > 2$. Figure 3 shows some sample path plots.

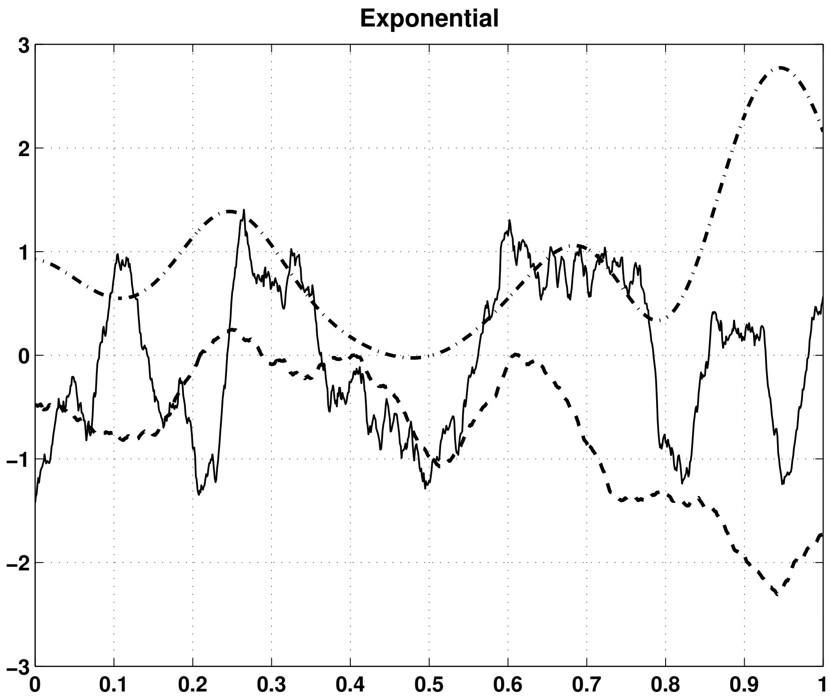

Figure 3: Smoothed sample paths from GP with exponential covariance function. All have variance $C = 1$ and $\alpha = 10^2$. Solid: $\delta = 1.5$. Dashed: $\delta = 1.9$. Dash-dotted: $\delta = 2$ (Gaussian).

We have derived the *spline* covariance function on $[0, 1]$ (22) from first principles above. This kernel is of interest because posterior mean functions in GP models (or minimisers of the variational problem over the RKHS) are splines of order m, i.e. piecewise polynomials in C^{2m-2} (see Section 6) and associated computations are $O(n)$ (where n is the number of training points, or "knots") only. On the other hand, technical complications arise because spline kernels are RKs for subspaces of $\mathcal{W}_m[0, 1]$ only, namely of the functions which satisfy the boundary conditions (see Section 6). The operator induced by a spline kernel has a null space spanned by polynomials, and in practice it is necessary to adjoin the corresponding (finite-dimensional) space. The spline kernels are not stationary (they are supported on $[0, 1]$), but we can obtain spline kernels on the circle by imposing periodic boundary conditions on $\mathcal{W}_m[0, 1]$, leading to the stationary kernel

$$K(x, x') = \sum_{\nu \geq 1} \frac{2}{(2\pi\nu)^{2m}} \cos(2\pi\nu(x - x')).$$

From this representation, it follows that the spectral density is

$$f(\omega) = \sum_{\nu \geq 1} \frac{1}{(2\pi\nu)^{2m}} \delta_{2\pi\nu}(|\omega|)$$

which is discrete. Note that sample functions from $u(x)$ are periodic with probability 1. In Wahba [80], Chap. 2 it is shown how to construct splines on the sphere by using the

[53] All these statements hold with probability 1, as usual.

iterated Laplacian, but this becomes quite involved. An equivalent to splines (in a sense) can be defined in $\mathbb{R}^g$ using *thin-plate spline* conditionally positive definite functions (see Section 8), see [80, 18] for details.

For kernel discrimination methods, *polynomial* covariance functions

$$K(\boldsymbol{x}, \boldsymbol{x}') = \frac{(\boldsymbol{x}^T\boldsymbol{x}' + \alpha)^m}{((\|\boldsymbol{x}\|^2 + \alpha)(\|\boldsymbol{x}'\|^2 + \alpha))^{m/2}}, \quad \alpha \geq 0, m \in \mathbb{N}$$

are popular although they seem unsuitable for regression problems. The denominator normalises the kernel to $K(\boldsymbol{x}, \boldsymbol{x}) = 1$. Although this normalisation is not done in some applications, it seems to be recommended in general. Polynomial kernels without the normalising denominator can be seen to induce a finite-dimensional feature space of polynomials of total degree $\leq m$ (if $\alpha > 0$).[54] It is interesting to note that this is exactly the RKHS we have to adjoin to one for a conditionally positive definite kernel of order m such as the thin-plate spline covariance function. On the other hand, in the spline case these polynomial parts are usually not regularised at all. By the Karhunen-Loeve expansion (see Section 5), we can write $u(\boldsymbol{x})$ as expansion in all monomials of total degree $\leq m$ with Gaussian random coefficients. The regularisation operator (see Section 6) for polynomial kernels is worked out in [70]. Note that $K(\boldsymbol{x}, \boldsymbol{x}')$ is not a covariance function for $m \notin \mathbb{N}$, thus the kernel is not infinitely divisible. Figure 4 shows some sample path plots. These are polynomials and therefore analytic.

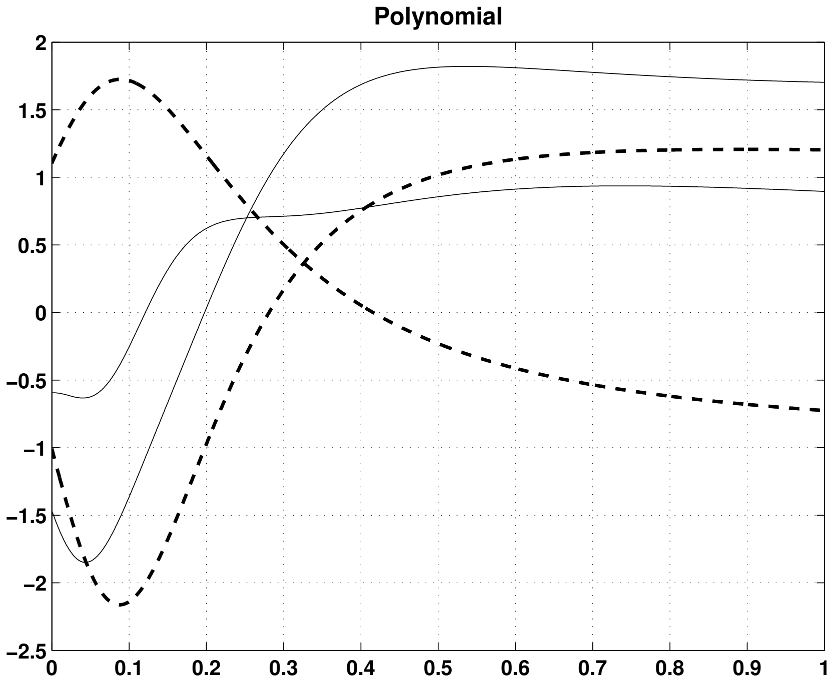

Figure 4: Sample paths from GP with polynomial covariance function. All have variance $C = 1$ and $\alpha = 0.05$. Solid: $m = 10$. Dashed: $m = 5$.

The Euclidean inner product $\boldsymbol{x}^T\boldsymbol{\Sigma}\boldsymbol{x}'$ is sometimes referred to as "linear kernel" in the

[54]The feature space of the normalised polynomial kernel consists of polynomials of total degree $\leq m$ divided by $(\|\boldsymbol{x}\|^2 + \alpha)^{m/2}$.

machine learning literature. GP models based on this kernel are nothing else than straightforward linear models (linear regression, logistic regression, etc.). It is clear from the weight space view (see Section 2) that a linear model can always be regarded as a GP model (or kernel technique), but this makes sense only if $n < g$, where n is the number of training points.[55] Furthermore, the SVM with linear kernel is a variant of the perceptron method [55] with "maximal stability" [57] studied in statistical physics.

Finally, let us give an example of a function which is *not* a covariance function, the so-called "sigmoid kernel"

$$K(\boldsymbol{x}, \boldsymbol{x}') = \tanh\left(a\boldsymbol{x}^T\boldsymbol{x}' + b\right).$$

K is not positive semidefinite for any a, b (see [69]), it is nevertheless shipped in most SVM packages we know of. It springs from the desire to make kernel expansions look like restricted one-layer neural networks. The correct link between MLPs and GP models has been given by Neal (see Section 2), which involves taking the limit of infinitely large networks. A covariance function corresponding to a one-layer MLP in the limit has been given by Williams [84]. In practice, it is of course possible to fit expansions of kernels to data which are not covariance functions. However, the whole underlying theory of minimisation in a RKHS (see Sections 5 and 6) breaks down, as does the view as inference in a GP model. On the practical side, flawed results such as negative predictive variances can pop up when least expected. Even worse, most optimisation techniques (including SVM algorithms) rely on the positive semidefiniteness of matrices and may break down otherwise. In fact, the SVM optimisation problem is not convex and has local minima for general K.

9.2 Constructing Kernels from Elementary Parts

We can construct complicated covariance functions from simple restricted ones which are easier to characterise (e.g. stationary or (an)isotropic covariance functions, see Section 2). A large number of families of elementary covariance functions are known (e.g., [88]), some of which are reviewed in Section 9.1.

A generalisation of stationary kernels to conditionally positive semidefinite ones (stationary fields to IRFs) is frequently used in geostatistical models (see Section 8) but will not be discussed here. The class of positive semidefinite forms has formidable closure properties. It is closed under positive linear (so-called conic) combinations, pointwise product and pointwise limit. If $K(\boldsymbol{v}, \boldsymbol{v}')$ is a covariance function, so is

$$\tilde{K}(\boldsymbol{x}, \boldsymbol{x}') = \int h(\boldsymbol{x}; \boldsymbol{v}) h(\boldsymbol{x}'; \boldsymbol{v}') K(\boldsymbol{v}, \boldsymbol{v}')\, d\boldsymbol{v}\, d\boldsymbol{v}' \tag{30}$$

(if $\tilde{K}$ is finite everywhere). An important special case is $\tilde{K}(\boldsymbol{x}, \boldsymbol{x}') = a(\boldsymbol{x})K(\boldsymbol{x}, \boldsymbol{x}')\, a(\boldsymbol{x}')$. For example, a given kernel (with positive variance everywhere) can always be modified to be constant on the diagonal by choosing $a(\boldsymbol{x}) = K(\boldsymbol{x}, \boldsymbol{x})^{-1/2}$, this normalisation has been discussed in the context of the polynomial kernel above. Note that O'Hagan's "localised regression model" (Section 2) is also a special case of (30). A general way of creating a non-stationary covariance function $\tilde{K}(\boldsymbol{y}, \boldsymbol{y}')$ from a parametric model $h(\boldsymbol{y}; \boldsymbol{\theta})$ linear in $\boldsymbol{\theta}$ is to assume a GP prior on $\boldsymbol{\theta}$, then to integrate out the parameters (see [62] for details). Furthermore, suppose we do so with a sequence of models and priors to obtain a sequence

[55] Otherwise, running a kernel algorithm is wasteful and awkward due to a singular kernel matrix.

of kernels. If the priors are appropriately scaled, the pointwise limit exists and is a kernel again. Many standard kernels can be obtained in this way (e.g., [84]). Neal showed that if the model size goes to infinity and the prior variances tend to 0 accordingly, layered models with non-Gaussian priors will also tend to a GP (due to the central limit theorem; see Section 2).

Another important modification is embedding. If $K(\boldsymbol{h},\boldsymbol{h}')$ is a covariance function and $\boldsymbol{h}(\boldsymbol{x})$ is an arbitrary map, then

$$\tilde{K}(\boldsymbol{x},\boldsymbol{x}') = K(\boldsymbol{h}(\boldsymbol{x}),\boldsymbol{h}(\boldsymbol{x}')) \tag{31}$$

is a covariance function as well (this is a special case of (30)). For example, if we have $d(\boldsymbol{x},\boldsymbol{x}') = \|\boldsymbol{h}(\boldsymbol{x}) - \boldsymbol{h}(\boldsymbol{x}')\|$ in some Euclidean space, then (26) is a valid kernel induced from the Gaussian (RBF) kernel (27). The Fisher kernel [23] and mutual information kernels [66] are examples. Embedding can be used to put rigid constraints on the GP. For example, if K is stationary in (31) and $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{h}(\boldsymbol{x}')$, then $u(\boldsymbol{x}) = u(\boldsymbol{x}')$ almost surely.[56] For $\boldsymbol{h}(x) = (\cos((2\pi/\nu)x), \sin((2\pi/\nu)x))^T$, sample paths of $u(x)$ are ν-periodic functions.

Embedding can be used to create non-stationary kernels from elementary stationary ones. A more powerful mechanism starts from viewing (30) in a different way. Let K be the squared-exponential kernel (28), but suppose the input $\boldsymbol{x}$ is subject to noise:

$$\boldsymbol{x} = \boldsymbol{t} + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim N(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{t})).$$

Here, $\boldsymbol{\varepsilon}$ at different observed locations $\boldsymbol{t}$ are independent, and all noise variables are independent of the process $u(\cdot)$. The process $v(\boldsymbol{t}) = u(\boldsymbol{x}) = u(\boldsymbol{t}+\boldsymbol{\varepsilon})$ is not Gaussian, but its mean and covariance function are determined easily: $\mathrm{E}[v(\boldsymbol{t})] = 0$ and

$$\mathrm{E}[v(\boldsymbol{t})v(\boldsymbol{t}')] \propto \mathrm{E}\left[N(\boldsymbol{t}+\boldsymbol{\varepsilon}\,|\,\boldsymbol{t}'+\boldsymbol{\varepsilon}', \boldsymbol{W}^{-1})\right] = N(\boldsymbol{t}\,|\,\boldsymbol{t}', \boldsymbol{W}^{-1} + \boldsymbol{S}(\boldsymbol{t}) + \boldsymbol{S}(\boldsymbol{t}'))$$

which has the form of a squared-exponential kernel with covariance matrix which depends on $\boldsymbol{t}$, $\boldsymbol{t}'$. A similar construction was used in [16] to create non-stationary covariance functions. This idea can be generalised considerably as shown in [49]. Define

$$Q(\boldsymbol{t},\boldsymbol{t}') = \sqrt{(\boldsymbol{t}-\boldsymbol{t}')^T \left(\frac{1}{2}(\boldsymbol{S}(\boldsymbol{t}) + \boldsymbol{S}(\boldsymbol{t}'))\right)^{-1} (\boldsymbol{t}-\boldsymbol{t}')}.$$

Note that Q is not a Mahalanobis distance, because the covariance matrix depends on $\boldsymbol{t}$, $\boldsymbol{t}'$. Now, if $\rho(\tau)$ is an isotropic correlation function in $\mathcal{D}^\infty$ (recall Section 2.2), it is shown in [49] that

$$\rho_Q(\boldsymbol{t},\boldsymbol{t}') = |\boldsymbol{S}(\boldsymbol{t})|^{1/4} |\boldsymbol{S}(\boldsymbol{t}')|^{1/4} \left|\frac{1}{2}(\boldsymbol{S}(\boldsymbol{t}) + \boldsymbol{S}(\boldsymbol{t}'))\right|^{-1/2} \rho(Q(\boldsymbol{t},\boldsymbol{t}')) \tag{32}$$

is a valid correlation function. The proof uses the characterisation

$$\rho(\tau) = \int e^{-\tau^2\omega^2}\, dF(\omega)$$

[56] This is because the correlation $\rho(u(\boldsymbol{x}), u(\boldsymbol{x}'))$ is 1, thus $u(\boldsymbol{x}) = a\, u(\boldsymbol{x}') + b$ for fixed a, b, then $a = 1, b = 0$ because both variables have the same mean and variance.

of $\mathcal{D}^\infty$ (see Section 2.2), thus

$$\rho(Q(\boldsymbol{t},\boldsymbol{t}')) = \left|\frac{1}{2}(S(\boldsymbol{t})+S(\boldsymbol{t}'))\right|^{1/2} \left(\frac{\pi}{2\omega^2}\right)^{d/2} \int N\left(\boldsymbol{t}\,\middle|\,\boldsymbol{t}', \frac{1}{4\omega^2}(S(\boldsymbol{t})+S(\boldsymbol{t}'))\right)\, dF(\omega).$$

The integral can now be written as

$$\propto \int\int N(\boldsymbol{r}\,|\,\boldsymbol{t}, \tilde{\boldsymbol{S}}(\boldsymbol{t},\omega)) N(\boldsymbol{r}\,|\,\boldsymbol{t}', \tilde{\boldsymbol{S}}(\boldsymbol{t}',\omega))\, d\boldsymbol{r}\, dF(\omega)$$

which is positive semi-definite as a special case of (30).[57] Equation (32) can be used to create many new families of non-stationary kernels from isotropic ones. Note that now there are two fields to estimate, $u(\cdot)$ and $\boldsymbol{t} \mapsto S(\boldsymbol{t})$. In principle, the latter one can be specified via GPs as well (see [49]), but inference becomes very costly. On the other hand, simpler parametric models may be sufficient. If unlabelled data is abundant, it is possible to learn the second field from this source only (see [62]). It is interesting to note that if $\boldsymbol{t} \mapsto S(\boldsymbol{t})$ is smooth, then m.s. properties of $u(\cdot)$ deducible from $\rho(\tau)$ are transferred to the GP with correlation function ρ_Q (32).

9.3 Guidelines for Kernel Choice

Choosing a good kernel for a task depends on intuition and experience. On high-dimensional tasks where no suitable prior knowledge is available, the best option may be to explore simple combinations of the standard kernels listed above. If invariances are known, they may be encoded using the methods described in [59], Sect. 11.4. With approximate Bayesian GP inference, one can in principle use combinations of different kernels with a lot of free (hyper)parameters which can be adapted automatically.

For low-dimensional $\mathcal{X}$, one can obtain further insight. Stein [74] points out the usefulness of studying fixed-domain asymptotics (see Section 8). In this respect, the tail behaviour of the spectral density (see Section 2) is important. The m.s. degree of differentiability (degree of smoothness) of the process depends on the rate of decay of $f(\boldsymbol{\omega})$. Stein recommends kernel families such as the Matérn class (29) which come with a degree of smoothness parameter ν. He also stresses the importance of the concept of equivalence and orthogonality of GPs (see Section 8). His arguments are of asymptotic nature, for example it is not clear whether ν in the Matérn class can be learned accurately enough from a limited amount of data. Also, predictions from equivalent processes with different kernels can be different.[58]

There are ways of "getting a feeling" for the behaviour of a process by visualisation, which is an option if $\mathcal{X} = \mathbb{R}^g$ is low-dimensional, $g = 1, 2$. We can draw "samples" from the process and plot them as follows (the plots in this section have been produced in this way). Let $X \subset \mathcal{X}$ be a fine grid[59] over a domain of interest, $n = |X|$ and $\boldsymbol{u} = u(X) \sim$

[57] Namely, (30) applies especially to "diagonal kernels" $K(\boldsymbol{v},\boldsymbol{v}') = f(\boldsymbol{v})\delta_{\boldsymbol{v}}(\boldsymbol{v}')$ where f is positive. In our case, $\boldsymbol{v} = (\boldsymbol{r}^T, \omega)^T$.

[58] Stein argues (citing Jeffreys) that such differences cannot be important since they do not lead to consistency in the large data limit (in a fixed domain).

[59] For fine grids and smooth kernels such as the Gaussian one, the Cholesky technique described here fails due to round-off errors. The singular value decomposition (SVD) should be used in this case, concentrating on the leading eigendirections which can be determined reliably.

$N(\mathbf{0}, \boldsymbol{K}(X))$. We can sample $\boldsymbol{u}$ as $\boldsymbol{u} = \boldsymbol{L}\boldsymbol{v}$, $\boldsymbol{v} \sim N(0, \boldsymbol{I})$, where $\boldsymbol{K}(X) = \boldsymbol{L}\boldsymbol{L}^T$ is the Cholesky decomposition. If X is too large, $\boldsymbol{u}$ can be approximated using an incomplete Cholesky factorisation of $\boldsymbol{K}(X)$ (see [87]). If $g = 1$, the process is isotropic and the grid is regularly spaced, $\boldsymbol{K}(X)$ has *Toeplitz* structure[60] and its Cholesky decomposition can be computed in $O(n^2)$ (see [14]). Repeatedly sampling $\boldsymbol{u}$ and plotting $(X, \boldsymbol{u})$ can give an idea about degree of smoothness, average length scales (Euclidean distance in $\mathcal{X}$ over which $u(\boldsymbol{x})$ is expected to vary significantly) or other special features of K.

9.4 Learning the Kernel

One promising approach for choosing a covariance function is to learn it from data and/or prior knowledge. For example, given a parametric family of covariance functions, how can we choose[61] the parameters in order for the corresponding process to model the observed data well?

Model selection from a fixed family can be done by the empirical Bayesian method of marginal likelihood maximisation, a generic approximation of which in the case of GP models is given in Section 4. Since this procedure typically scales linearly in the number of hyperparameters, elaborate and heavily parameterised families can be employed. An important special case has been termed *automatic relevance determination (ARD)* by MacKay [34] and Neal [42]. The idea is to introduce a hyperparameter which determines the scale of variability of a related variable of interesting (with prior mean 0). For example, we might set up a linear model (4) by throwing in a host of different features (components in $\Phi(\boldsymbol{x})$), then place a $N(\boldsymbol{\beta}|\mathbf{0}, \boldsymbol{D})$ on the weights $\boldsymbol{\beta}$ where $\boldsymbol{D}$ is a diagonal matrix of positive hyperparameters. If we place a hyperprior on $\operatorname{diag} \boldsymbol{D}$ which encourages small values, there is an *a priori* incentive for $d_i = \boldsymbol{D}_{i,i}$ to become very small, inducing a variance of β_i close to 0 which effectively switches off the effect of $\beta_i \phi_i(\boldsymbol{x})$ on predictions. This is balanced against the need to use at least some of the components of the model to fit the data well, leading to an automatic discrimination between relevant and irrelevant components. In the context of covariance functions, we can implement ARD with any anisotropic kernel (see Section 2) of the form

$$K(\boldsymbol{x}, \boldsymbol{x}') = \tilde{K}((\boldsymbol{x} - \boldsymbol{x}')^T \boldsymbol{W} (\boldsymbol{x} - \boldsymbol{x}')),$$

where $\tilde{K}$ is isotropic and $\boldsymbol{W}$ is diagonal and positive definite. An example is the squared-exponential covariance function (28). Here, w_i determines the scale of variability of the (prior) field as $\boldsymbol{x}$ moves along the i-th coordinate axis. If we imagine the field being restricted to a line parallel to this axis, $w_i^{-1/2}$ is the length scale of this restriction, i.e. a distance for which the expected change of the process is significant. If $w_i \approx 0$, this length scale is very large, thus the field will be almost constant along this direction (in regions of interest). Thus, via ARD we can discriminate relevant from irrelevant dimensions in the input variable $\boldsymbol{x}$ automatically, and predictions will not be influenced significantly by the latter.

In spatial statistics, semivariogram techniques (see [11], Sect. 2.3.1) are frequently used. For a stationary process, the (semi)variogram is $\gamma(\boldsymbol{x} - \boldsymbol{x}') = (1/2)\mathrm{Var}[u(\boldsymbol{x}) - u(\boldsymbol{x}')]$. It is estimated by averaged squared distances over groups of datapoints which are roughly the

[60] A matrix is Toeplitz if all its diagonals (main and off-diagonals) are constant.

[61] The proper Bayesian solution would be to integrate out the parameters, but even if this can be approximated with MCMC techniques, the outcome is a mixture of covariance functions leading to expensive predictors.

same distance apart and fitted to parametric families by maximum likelihood. Stein [74] criticises the use of the empirical semivariogram as single input for choosing a covariance function and suggests a range of other techniques, including the empirical Bayesian approach mentioned above.

For classification models, the idea of local invariance w.r.t. certain groups of transformations is important. For example, the recognition of handwritten digits should not be influenced by translations or small-angle rotations of the bitmap.[62] If a process is used as latent function in a classification problem, e.g. representing the log probability ratio between classes (see Section 3), then starting from some $\boldsymbol{x}$ and applying small transformations from a group w.r.t. which discrimination should remain invariant should not lead to significant changes in the process output (e.g. in the m.s. sense). To relate this notion to ARD above, varying $\boldsymbol{x}$ along such invariant directions should induce a coordinate of $\boldsymbol{x}$ (non-linear in general) which is irrelevant for prediction. Chapter 11 in [59] gives a number of methods for modifying a covariance function in order to incorporate invariance knowledge to some degree, also reviewing work in that direction which we omit here.

Finally, Minka [38] pointed out that instances of the "learning how to learn" or "prior learning" paradigm can be seen as learning a GP prior from multi-task data (see his paper for references). In fact, the setup is the one of a standard hierarchical model frequently used in Bayesian statistics to implement realistic prior distributions. We have access to *several* noisy samples and make the assumption that these have been sampled from different realisations of the latent process which in turn have been sampled i.i.d. from the process prior. Data of this sort is very valuable for inferring aspects of the underlying covariance function. In a simple multi-task scenario a multi-layer perceptron is fit to several samples by penalised maximum likelihood, sharing the same input-to-hidden weights but using different sets of hidden-to-output weights for each sample. The idea is that the hidden units might discover features which are important in general, while the combination in the uppermost layer is specific. If we place Gaussian priors on the hidden-to-output weights, this becomes a GP model with a covariance function determined by the hidden units. More generally, we can start from any parametric family of covariance functions and learn hyperparameters or even the hyperposterior from multi-task data using marginal likelihood maximisation together with the hierarchical sampling model. An approximate implementation of this idea has been reported in [51].

9.5 Kernels for Discrete Objects

As mentioned in Section 2, in principle the input space $\mathcal{X}$ is not restricted to be $\mathbb{R}^g$ or even a group. For example, Gaussian processes over lattices are important in vision applications (in the form of a Gaussian Markov random field with sparse structured inverse covariance matrix). For Gaussian likelihoods, the posterior mean can be determined most efficiently using a conjugate gradients solver[63] and the embedded trees algorithm of Wainwright, Sudderth and Willsky [82] can be used to compute the marginal variances as well. Kernel methods, i.e. methods which use covariance matrices over variables determined from the "spatial" relationship of these (or associated covariates) have been proposed for a number

[62] Although a 180-degree rotation of a 6 results in a 9.

[63] Loopy belief propagation renders the correct mean as well if it converges [83], but is much slower and often numerically unstable.

of problems involving discrete spaces $\mathcal{X}$ (finite or countably infinite). Our aim in this section is no more than to give a few selected examples.

Kernels can be defined on the set of finite-length strings from a finite alphabet. Many *string kernels* have been proposed recently, but we will not try to review any of this work. Important applications of string kernels (or distance measures between sequences) arise from problems in DNA or RNA biology where statistical models have to be built for nucleotide sequences. Many proposed string kernels are special cases of *convolution kernels* introduced by Haussler [21]. Maybe the most interesting case discussed there is the extension of a hidden Markov random field (HMRF). The latter is a Markov random field (MRF) with observed variables $\boldsymbol{x}$, latent variables $\boldsymbol{u}$ and clique potentials $C_d(x_d, u_d)$ where x_d, u_d are subsets of components of $\boldsymbol{x}$, $\boldsymbol{u}$, and $\boldsymbol{u}$ is marginalised over. If we replace the clique potential by positive definite kernels $K_d((x_d, u_d), (x'_d, u'_d))$ and marginalise over $\boldsymbol{u}$, $\boldsymbol{u}'$, the result is a covariance kernel which can also be seen as unnormalised joint generative distribution for $(\boldsymbol{x}, \boldsymbol{x}')$. If the original MRF has a structure which allows for tractable computation, the same algorithm can be used to evaluate the covariance function efficiently. For example, a hidden Markov model (HMM) for sequences can be extended to a pair-HMM in this way, emitting two observed sequences sharing the same latent sequence, and many string kernels arise as special cases of this construction.

In practice, string kernels (and more generally kernels obtained from joint probabilities under pair-HMRFs) often suffer from the "ridge problem": $K(\boldsymbol{x}, \boldsymbol{x})$ is much larger than $K(\boldsymbol{x}, \boldsymbol{x}')$ for many $\boldsymbol{x}'$ for which *a priori* we would like to attain a significant correlation, especially if rather long sequences are compared. For example, in models involving DNA sequences we would like sequences to correlate strongly if they are homologous, *i.e.* encode for proteins of very similar function. In a standard string kernel, two sequences are strongly correlated if both can be obtained from a common "ancestor" latent sequence by few operations such as insertions and substitutions (this ancestor model is motivated by the evolution of genes and gives a good example for the pair-HMM setup). However, often homologous sequences differ quite substantially in regions on which the structure of the functional part of the protein does not depend strongly. Such "remote" homologies are the really interesting ones, since very close homologies can often be detected using simpler statistical techniques than process models based on string kernels. On the other hand, it may be possible to spot such homologies by going beyond string kernels and pair-HMRF constructions, for example building on the general framework given in [10] where kernels are obtained from finite transducers.

A conceptually simple way to obtain a kernel on $\mathcal{X}$ is to embed $\mathcal{X}$ in some Euclidean space $\mathbb{R}^g$, then to concatenate the embedding with any of the known $\mathbb{R}^g$ kernels, for example the Gaussian one (27). An example is the Fisher kernel [23] which maps datapoints to their "Fisher scores" under a parametric model. There has been a surge of interest recently in automatic methods for parameterising low-dimensional non-linear manifolds (e.g., [77, 56]) by local Euclidean coordinates. Although these methods are non-parametric, they can be used to fit conventional parametric mixture models in order to obtain a parametric embedding which could then be used to obtain a kernel.

Recently, Kondor and Lafferty [29] proposed kernels on discrete objects using concepts from spectral graph theory (diffusion on graphs). If $\mathcal{X}$ is finite, a covariance function on $\mathcal{X}$ is simply a positive semidefinite matrix. If $\boldsymbol{H}$ is a symmetric *generator matrix*, the corresponding *exponential kernel* is defined as

$$\boldsymbol{K}^{(\beta)} = \exp(\beta \boldsymbol{H}) = \sum_{j \geq 1} \frac{\beta^j}{j!} \boldsymbol{H}^j, \quad \beta \geq 0. \tag{33}$$

We define $K_\beta(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{K}^{(\beta)}_{\boldsymbol{x},\boldsymbol{x}'}$, where we use elements of $\mathcal{X}$ as indices into the matrix $\boldsymbol{K}^{(\beta)}$. $\boldsymbol{K}^{(\beta)}$ is positive definite. In fact, it has the same eigenvectors as $\boldsymbol{H}$, but the eigenspectrum is transformed via $\lambda \to \exp(\beta\lambda)$. In practice, general exponential kernels cannot be computed feasibly if $\mathcal{X}$ is large, in particular there is no general efficient way of computing kernel matrices of K_β over points of interest. It might be possible to approximate marginalisations of $\boldsymbol{K}^{(\beta)}$ by sampling. The kernel and generator matrices are linked by the *heat equation*

$$\frac{\partial}{\partial \beta} \boldsymbol{K}^{(\beta)} = \boldsymbol{H} \boldsymbol{K}^{(\beta)}.$$

It is interesting to note that every infinitely divisible covariance function K_β with scale parameter β on $\mathcal{X}$ has the form (33). Namely, if $\boldsymbol{K}$ is the covariance matrix for K_β, then $\boldsymbol{H} = \partial \boldsymbol{K} / \partial \beta$ at $\beta = 0$. Kondor and Lafferty are interested in *diffusion kernels* on graphs as special cases of exponential kernels. Here, the generator is the negative of the so-called graph Laplacian. The construction can be seen as stationary Markov chain (random walk) in continuous time on the vertices of the graph. The kernel $K_\beta(\boldsymbol{x}, \boldsymbol{x}')$ is the probability of being at $\boldsymbol{x}'$ at time β, given that the state at time 0 was $\boldsymbol{x}$. This interpretation requires that $\boldsymbol{H}\boldsymbol{1} = \boldsymbol{0}$ which is true for the negative graph Laplacian and which implies that $\boldsymbol{K}^{(\beta)}$ is (doubly) stochastic. The same equation describes heat flow or diffusion from an initial distribution. The idea is to describe the structure of $\mathcal{X}$ (in the sense of "closeness", *i.e.* close points should be highly correlated under the covariance function) in terms of local neighbourhood association which induce an (weighted or unweighted) undirected graph. Then, the correlation at some $\boldsymbol{x}$ with all other points is proportional to the distribution of a random walk started at $\boldsymbol{x}$ after time β. Similar ideas have been used very effectively for non-parametric clustering or classification with partially labelled data [76]. Kondor and Lafferty give examples for graphs of special regular structures for which the diffusion kernel can be determined efficiently. These include certain special cases of string kernels (here, $\mathcal{X}$ is infinite and the analogue to Markov chains has to be treated more carefully). In situations where K_β cannot be determined by known simple recursive formulae, one could represent $\mathcal{X}$ by a representative sample including the training set (but also unlabelled data). If the generator matrix of the underlying graph (projected onto the representative sample in a sensible way) is sparse, its leading eigenvectors and eigenvalues could be approximated by sparse eigensolvers which would lead to an approximation of $\boldsymbol{K}^{(\beta)}$ which is low-rank optimal w.r.t. the Frobenius norm. Kondor and Lafferty also note that on the graph given by a regular grid in $\mathbb{R}^g$, the generator matrix converges towards the usual Laplacian operator and K_β towards the Gaussian kernel (27) as the mesh size approaches 0.

9.6 How Useful are Uncertainty Estimates?

In this section we have highlighted a number of powerful techniques of encoding prior knowledge in a covariance function or learning an appropriate kernel. For many problems in machine learning (especially in classification) one does not observe a big difference in generalisation error over a range of different common kernels, while significant differences arise in the uncertainty estimates (predictive variances) for Bayesian GP techniques. Moreover, the discussion in Section 7 suggests that much of the additional complexity in Bayesian GP methods as compared to SVM arise exactly because such uncertainty estimates are desired as well. It is therefore important to ask how useful these estimates are in practice.

Strictly speaking, both frequentist confidence intervals and Bayesian uncertainty estimates are tied to assumptions which are likely to be violated in non-trivial real world situations. The former are conditioned on a null hypothesis which is certainly violated at some scale, the latter require the data to be generated by the model. In a Bayesian setting, different priors and models can be compared either to conclude that the predictions enjoy a certain robustness or to detect mismatches which should trigger a refinement.

In the case of GP models, the choice of the covariance function can have a significant effect on the uncertainty estimates. We demonstrate this fact using a simple one-dimensional regression task. Note that in GP regression with Gaussian noise, the error bars do not depend on the targets (this is different for non-Gaussian likelihoods, e.g. in classification). Data was sampled from a noisy sine wave around $\pi/2$, $(3/2)\pi$, a single point at π, the noise standard deviation was $\sigma = 0.05$. We compare the RBF covariance function (27) with $w = 4$ against the Matérn kernel with different ν and $\alpha = (w(2\nu + 1))^{1/2}$, the process variance was $C = 1$ in all cases. Recall that for the Matérn kernel, ν controls the degree of m.s. differentiability of the process, while the RBF process is m.s. analytic. Figure 5 shows mean predictions and one standard deviation error bars (the noise level was set to the true value).

As expected, for the Ornstein-Uhlenbeck prior ($\nu = 1/2$) the mean prediction interpolates the data, the error bars grow to the maximum value 1 very rapidly away from the data. A Brownian motion process is not suitable as prior for a smoothing technique. The tendency to interpolate rather than smooth the data diminishes with growing ν, as does the speed with which the error bars grow to 1 away from data. Note also the very slim error bars for the RBF prediction in the data-rich regions, expressing the strong (prior) belief that the underlying function is smooth, thus close to the smooth mean prediction there. Stein [74] notes that predictions using the RBF covariance function often come with unrealistically small error bars.

In many situations, the uncertainty estimates themselves are of less importance than the quality of the *decisions* based on them. In the Bayesian context, decisions are made by substituting the predictive distribution inferred from data for the unknown truth. Utility values can be computed as expectations over the predictive distribution and "Bayesian optimal" decisions be made by comparing these for different alternatives. A simple example arises in binary classification if the task allows us to *reject* a certain fraction of the test patterns. The Bayesian optimal decision is to reject patterns for which the target predictive distribution $P(y_*|\boldsymbol{x}_*, D)$ is most uncertain (has highest entropy). A similar setting is treated heuristically with SVM discriminants rejecting those patterns for which the discriminant value is closest to zero. Note that in both cases, we are interested in the order relations of the scores over a test set rather than their numerical values. A study comparing both practices (the GP technique is a sparse IVM approximation [31] using the same amount of running time) has been done in [63], Sect. 4.7.2. It concludes that on the example considered the SVM reject strategy shows significant weaknesses compared to the approximate Bayesian IVM setup and that the additional work for obtaining uncertainty estimates can pay off.[64]

[64] It is shown that large wrong predictive means are often accompanied by large predictive variances,

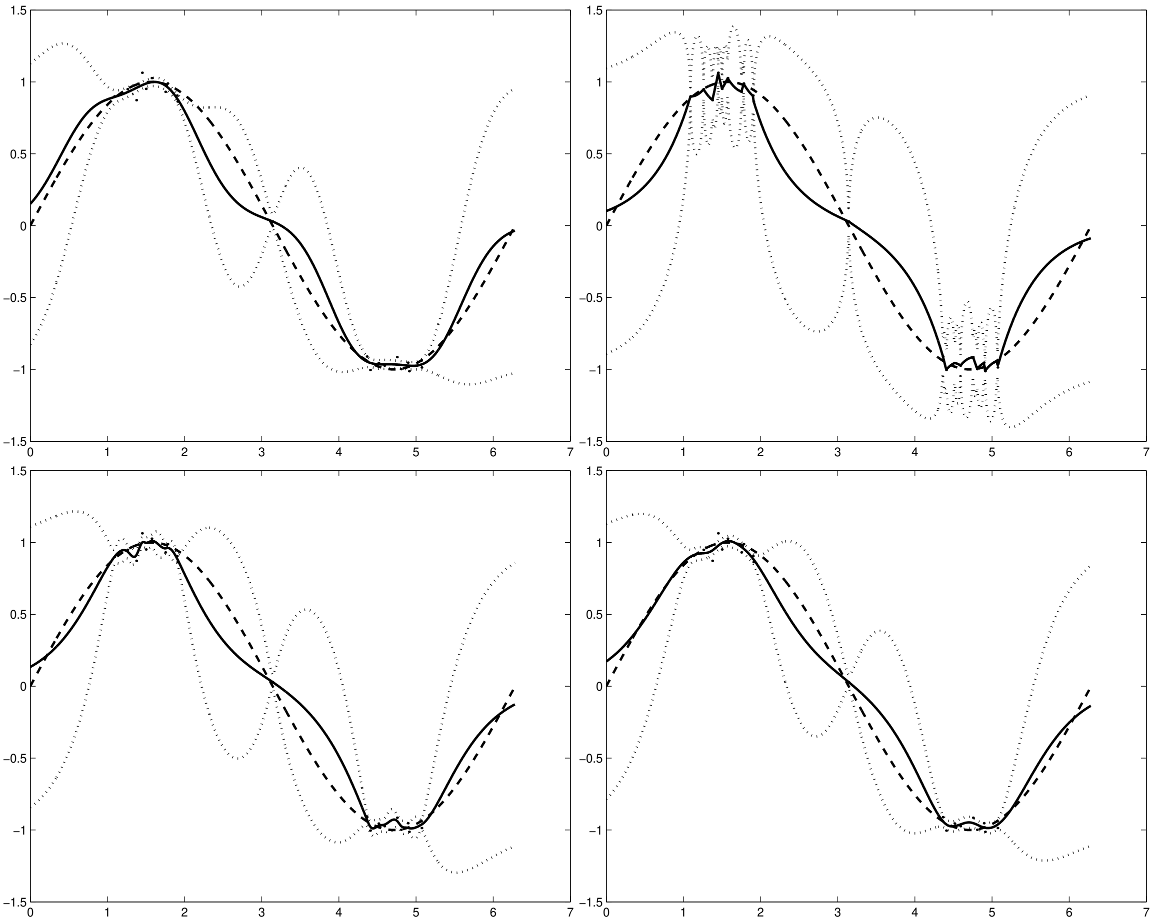

Figure 5: Error bars for noisy sine regression task for different covariance functions. Mean prediction (solid), errors bars (dotted), true curve (dashed), data (dots). Upper left: RBF, $w = 4$. Upper right: Ornstein-Uhlenbeck (Matérn, $\nu = 1/2$), $\alpha = 2.8284$. Lower left: Matérn, $\nu = 3/2,\ \alpha = 4$. Lower right: Matérn, $\nu = 3/2,\ \alpha = 4.899$.

Note that these shortcomings of SVM cannot be alleviated by posthoc transformations of the discriminant output (as suggested by [50]) because these leave order relations invariant.

10 Summary

In this paper, we described central properties of Gaussian processes and statistical models based on GPs together with efficient generic ways of approximate inference and model selection. The focus is less on giving algorithmic descriptions of concrete inference approximations and their variational optimisation problems, which may be found in the references provided. Instead we hope to have conveyed the basic concepts of latent variables and Gaussian random fields required to understand these non-parametric algorithms and to have highlighted some of the essential differences to parametric statistical models. By the

explaining the superior performance of the Bayesian score which combines these two quantities.

evolution of ever more powerful computers and the development of fast sparse inference approximations, we feel that GP models will become applicable to large-data problems which were previously restricted to parametric models. GP models are more powerful and flexible than simple linear parametric models and easier to handle than complicated ones such as multi-layer perceptrons, and the availability of fast algorithms should remove remaining obstacles of them becoming part of the standard toolbox of machine learning practitioners.

Acknowledgements

We thank Chris Williams for many discussions and important comments on early drafts, David Barber and Bernhard Schölkopf for corrections and improvements, Bernhard Schölkopf and the MPI Tübingen for their hospitality in September 2003, furthermore Neil Lawrence, Ralf Herbrich, Lehel Csató, Manfred Opper, Carl Rasmussen, Amos Storkey and Michael Tipping for discussions and comments. The author gratefully acknowledges support through a research studentship from *Microsoft Research Ltd.* during his postgraduate studies.

A Appendix

In section A.1, we describe the notational conventions used in this paper and some concepts from probability theory. In Section A.2 we collect some definitions.

A.1 Notation

Vectors $\boldsymbol{a} = (a_i)_i = (a_1 \dots a_n)^T$ (column by default) and matrices $\boldsymbol{A} = (a_{i,j})_{i,j}$ are written in bold-face. If $\boldsymbol{A} \in \mathbb{R}^{m,n}$, $I \subset \{1, \dots, m\}$, $J \subset \{1, \dots, n\}$ are index sets,[65] then $\boldsymbol{A}_{I,J}$ denotes the $|I| \times |J|$ sub-matrix formed from $\boldsymbol{A}$ by selecting the corresponding entries (i, j), $i \in I$, $j \in J$.

Some special vectors and matrices are defined as follows: $\mathbf{0} = (0)_i$ and $\mathbf{1} = (1)_i$ the vectors of all zero and all ones, $\boldsymbol{\delta}_j = (\delta_{i,j})_i$ the j-th standard unit vector. Here, $\delta_{i,j} = 1$ if $i = j$, and 0 otherwise (Kronecker symbol). Furthermore, $\boldsymbol{I} = (\delta_{i,j})_{i,j}$ is the identity matrix.

The superscript T denotes transposition. $\operatorname{diag} \boldsymbol{a}$ is the matrix with diagonal $\boldsymbol{a}$ and 0 elsewhere. $\operatorname{diag} \boldsymbol{A}$ is the vector containing the diagonal of $\boldsymbol{A}$. $\operatorname{tr} \boldsymbol{A}$ is the sum of the diagonal elements of $\boldsymbol{A}$, $\operatorname{tr} \boldsymbol{A} = \mathbf{1}^T(\operatorname{diag} \boldsymbol{A})$. $|\boldsymbol{A}|$ denotes the determinant of the square matrix $\boldsymbol{A}$. For $p > 1$, $\|\boldsymbol{a}\|_p$ denotes the p-norm of the vector $\boldsymbol{a}$, $\|\boldsymbol{a}\|_p = (\sum_i |a_i|^p)^{1/p}$. If nothing else is said, $\|\cdot\| = \|\cdot\|_2$, the Euclidean norm. Relations are vectorised in Matlab style, as are scalar functions: $\boldsymbol{a} \ge \boldsymbol{b}$ means that $a_i \ge b_i$ for all i, and $f(\boldsymbol{a}) = (f(a_i))_i$.

We do not distinguish notationally between a random variable and its possible values. Vector or matrix random variables are written in the same way as vectors or matrices. If a distribution has a density, we generally use the same notation for the distribution and its density function. If $\boldsymbol{x}$ is a random variable, then $\mathrm{E}[\boldsymbol{x}]$ denotes the expectation (or expected value) of $\boldsymbol{x}$. If A is an event, then $\Pr\{A\}$ denotes its probability. The probability space will

[65] All index sets and sets of data points are assumed to be ordered, although we use a notation known from unordered sets.

usually be clear from the context, but for clarity we often use an additional subscript, e.g. $\Pr_S\{A\}$ or $\mathrm{E}_P[\boldsymbol{x}]$ (meaning that $\boldsymbol{x} \sim P$). By I_A, we denote the indicator function of an event A, i.e. $\mathrm{I}_A = 1$ if A is true, $\mathrm{I}_A = 0$ otherwise. Note that $\Pr\{A\} = \mathrm{E}[\mathrm{I}_A]$. The delta distribution $\delta_{\boldsymbol{x}}$ places mass 1 onto the point $\boldsymbol{x}$ and no mass elsewhere, $\delta_{\boldsymbol{x}}(B) = \mathrm{I}_{\{\boldsymbol{x} \in B\}}$. Let $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$ be sets of random variables, $\mathcal{X}$, $\mathcal{Y}$ non-empty. We write $\mathcal{X} \perp \mathcal{Y} \,|\, \mathcal{Z}$ to denote the conditional independence of $\mathcal{X}$ and $\mathcal{Y}$ given $\mathcal{Z}$: the conditional distribution of $\mathcal{X}$ given $\mathcal{Y}$, $\mathcal{Z}$ does not depend on $\mathcal{Y}$.

log denotes the logarithm to Euler's base e. The notation $f(\boldsymbol{x}) \propto g(\boldsymbol{x})$ means that $f(\boldsymbol{x}) = c g(\boldsymbol{x})$ for $c \neq 0$ constant w.r.t. $\boldsymbol{x}$. We often use this notation with the left hand side being a density. By $\operatorname{sgn} x$, we denote the sign of x, i.e. $\operatorname{sgn} x = +1$ for $x > 0$, $\operatorname{sgn} x = -1$ for $x < 0$, and $\operatorname{sgn} 0 = 0$. The Landau O-notation is defined as $g(n) = O(f(n))$ iff there exists a constant $c \geq 0$ such that $g(n) \leq c\, f(n)$ for almost all n.

We use some probability-theoretic concepts and notation which might be unfamiliar to the reader. A measure is denoted by $d\mu(\boldsymbol{x})$, the Lebesgue measure in $\mathbb{R}^g$ is denoted by $d\boldsymbol{x}$. If A is a measurable set ("event"), $\mu(A) = \int \mathrm{I}_{\{\boldsymbol{x} \in A\}} d\mu(\boldsymbol{x})$ denotes its mass under μ. A measure is finite if the mass of the whole space is finite, and a probability measure if this mass is 1. If $d\mu$ is a probability measure, we denote its distribution by μ. The events A of mass 0 are called *null sets*.[66] For example, in $\mathbb{R}^g$ with Lebesgue measure (the usual "volume") all affine spaces of dimension $< g$ are null sets. A property is *almost surely (a.s.)* true if the event of it being false is a null set. $d\mu_1$ is called *absolutely continuous* w.r.t. $d\mu_2$ if all null sets of $d\mu_1$ are null sets of $d\mu_2$ (the notation is $d\mu_1 \ll d\mu_2$). The theorem of Radon and Nikodym states that $d\mu_1$ has a *density* $f(\boldsymbol{x})$ w.r.t. $d\mu_2$, i.e.

$$\mu_1(A) = \int \mathrm{I}_{\{\boldsymbol{x} \in A\}} f(\boldsymbol{x})\, d\mu_2(\boldsymbol{x})$$

for all measurable A, iff $d\mu_1 \ll d\mu_2$. In this case,

$$f(\boldsymbol{x}) = \frac{d\mu_1(\boldsymbol{x})}{d\mu_2(\boldsymbol{x})}$$

is called *Radon-Nikodym derivative* or simply density w.r.t. $d\mu_2$.

A.2 Definitions

Definition 1 (Relative Entropy) *Let P, Q be two probability measures on the same space with $Q \ll P$, such that the density dQ/dP exists almost everywhere. The* relative entropy *is defined as*

$$\mathrm{D}[Q \,\|\, P] = \mathrm{E}_Q\left[\log \frac{dQ}{dP}\right] = \int \left(\log \frac{dQ}{dP}\right) dQ.$$

If Q is not absolutely continuous w.r.t. P, we set $\mathrm{D}[Q \,\|\, P] = \infty$. It is always non-negative, and equal to 0 iff $Q = P$. The function $(Q, P) \mapsto \mathrm{D}[Q \,\|\, P]$ is strictly convex.

[66] In order not to run into trouble, we always assume that our probability space is *complete*, meaning that its sigma-algebra contains all subsets of null sets.

If both Q and P have a density w.r.t. Lebesgue measure $d\boldsymbol{w}$, then $dQ/dP = Q(\boldsymbol{w})/P(\boldsymbol{w})$, the ratio of the densities.

If we fix a base measure P_0 (finite, need not be a probability), the *entropy* can be defined as $\mathrm{H}[Q] = -\mathrm{D}[Q \,\|\, P_0]$. For continuous distributions over $\mathbb{R}^g$, the uniform (Lebesgue) measure is not finite. The usual remedy is to subtract off an infinite part of the entropy which does not depend on the argument Q, ending up with the *differential entropy*

$$\mathrm{H}[Q] = -\int Q(\boldsymbol{w}) \, \log Q(\boldsymbol{w}) \, d\boldsymbol{w}. \tag{34}$$

Both entropy and differential entropy are concave functions (being the negative of convex ones).

References

[1] P. Abrahamsen. A review of Gaussian random fields and correlation functions. Technical report, Norwegian Computing Centre, 1997.

[2] R. J. Adler. *Geometry of Random Fields.* John Wiley & Sons, 1981.

[3] N. Aronszajn. Theory of reproducing kernels. *Trans. Amer. Math. Soc.*, 68(3):337–404, 1950.

[4] D. Barber and C. Bishop. Ensemble learning for multi-layer networks. In M. Jordan, M. Kearns, and S. Solla, editors, *Advances in Neural Information Processing Systems 10*, pages 395–401. MIT Press, 1998.

[5] S. Becker, S. Thrun, and K. Obermayer, editors. *Advances in Neural Information Processing Systems 15.* MIT Press, 2003. To appear.

[6] P. Billingsley. *Probability and Measure.* John Wiley & Sons, 3rd edition, 1995.

[7] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2002. Available online at `www.stanford.edu/~boyd/cvxbook.html`.

[8] Christopher Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[9] K. L. Chung. *A Course in Probability Theory.* Academic Press, 2nd edition, 1974.

[10] C. Cortes, P. Haffner, and M. Mohri. Rational kernels. In Becker et al. [5]. To appear.

[11] N. Cressie. *Statistics for Spatial Data.* John Wiley & Sons, 2nd edition, 1993.

[12] Lehel Csató, Ernest Fokoué, Manfred Opper, Bernhard Schottky, and Ole Winther. Efficient approaches to Gaussian process classification. In Solla et al. [72], pages 251–257.

[13] Lehel Csató and Manfred Opper. Sparse on-line Gaussian processes. *Neural Computation*, 14:641–668, 2002.

[14] G. Cybenko and M. Berry. Hyperbolic Householder algorithms for factoring structured matrices. *SIAM J. Matrix Anal. Appl.*, 11:499–520, 1990.

[15] T. Dietterich, S. Becker, and Z. Ghahramani, editors. *Advances in Neural Information Processing Systems 14*. MIT Press, 2002.

[16] Mark N. Gibbs. *Bayesian Gaussian Processes for Regression and Classification*. PhD thesis, University of Cambridge, 1997.

[17] A. Girard, C. Rasmussen, and R. Murray-Smith. Gaussian process priors with uncertain inputs — application to multiple-step ahead time series forecasting. In Becker et al. [5]. To appear.

[18] P.J. Green and Bernhard Silverman. *Nonparametric Regression and Generalized Linear Models*. Monographs on Statistics and Probability. Chapman & Hall, 1994.

[19] Geoffrey Grimmett and David Stirzaker. *Probability and Random Processes*. Oxford University Press, 3rd edition, 2001.

[20] P. Halmos. *Introduction to Hilbert Space and the Theory of Spectral Multiplicity*. Chelsea, New York, 1957.

[21] David Haussler. Convolution kernels on discrete structures. Technical Report UCSC-CRL-99-10, University of California, Santa Cruz, July 1999. See `http://www.cse.ucsc.edu/~haussler/pubs.html`.

[22] Shunsuke Ihara. *Information Theory for Continuous Systems*. World Scientific, 1st edition, 1993.

[23] T. S. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In Kearns et al. [26], pages 487–493.

[24] Tommi Jaakkola and David Haussler. Probabilistic kernel regression models. In D. Heckerman and J. Whittaker, editors, *Workshop on Artificial Intelligence and Statistics 7*. Morgan Kaufmann, 1999.

[25] Tommi Jaakkola, Marina Meila, and Tony Jebara. Maximum entropy discrimination. In Solla et al. [72], pages 470–476.

[26] M. Kearns, S. Solla, and D. Cohn, editors. *Advances in Neural Information Processing Systems 11*. MIT Press, 1999.

[27] G. Kimeldorf and G. Wahba. A correspondence between Bayesian estimation of stochastic processes and smoothing by splines. *Annals of Mathematical Statistics*, 41:495–502, 1970.

[28] A. N. Kolmogorov. *Foundations of the Theory of Probability*. Chelsea, New York, 2nd edition, 1933. Trans. N. Morrison (1956).

[29] R. I. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete input spaces. In C. Sammut and A. Hofmann, editors, *International Conference on Machine Learning 19*. Morgan Kaufmann, 2002.

[30] D. Krige. A statistical approach to some basic mine valuation problems on the witwatersrand. *Journal of the Chemical, Metallurgical and Mining Society of South Africa*, 52:119–139, 1951.

[31] Neil D. Lawrence, Matthias Seeger, and Ralf Herbrich. Fast sparse Gaussian process methods: The informative vector machine. In Becker et al. [5]. See `www.cs.berkeley.edu/~mseeger`.

[32] T. Leen, T. Dietterich, and V. Tresp, editors. *Advances in Neural Information Processing Systems 13*. MIT Press, 2001.

[33] D. MacKay. *Bayesian Methods for Adaptive Models*. PhD thesis, California Institute of Technology, 1991.

[34] D. MacKay. Bayesian non-linear modeling for the energy prediction competition. In *ASHRAE Transactions*, volume 100, pages 1053–1062, 1994.

[35] G. Matheron. Principles of geostatistics. *Economic Geology*, 58:1246–1266, 1963.

[36] G. Matheron. The intrinsic random functions and their applications. *Journal for Applied Probability*, 5:439–468, 1973.

[37] P. McCullach and J.A. Nelder. *Generalized Linear Models*. Number 37 in Monographs on Statistics and Applied Probability. Chapman & Hall, 1st edition, 1983.

[38] T. Minka and R. Picard. Learning how to learn is learning with point sets. Unpublished manuscript. Available at `http://wwwwhite.media.mit.edu/~tpminka/papers/learning.html`., 1997.

[39] Thomas Minka. *A Family of Algorithms for Approximate Bayesian Inference*. PhD thesis, Massachusetts Institute of Technology, January 2001.

[40] E. H. Moore. On properly positive hermitian matrices. *Bull. Amer. Math. Soc.*, 23:59, 1916.

[41] R. M. Neal. Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG-TR-93-1, University of Toronto, 1993. See `www.cs.toronto.edu/~radford`.

[42] R. M. Neal. *Bayesian Learning for Neural Networks*. Number 118 in Lecture Notes in Statistics. Springer, 1996.

[43] Radford M. Neal. Monte Carlo implementation of Gaussian process models for Bayesian classification and regression. Technical Report 9702, Department of Statistics, University of Toronto, January 1997.

[44] J. Nelder and R. Wedderburn. Generalized linear models. *Journal of Roy. Stat. Soc. B*, 135:370–384, 1972.

[45] A. O'Hagan. Curve fitting and optimal design. *Journal of Roy. Stat. Soc. B*, 40(1):1–42, 1978.

[46] A. O'Hagan. Some Bayesian numerical analysis. In J. Bernardo, J. Berger, A. Dawid, and A. Smith, editors, *Bayesian Statistics 4*, pages 345–363. Oxford University Press, 1992.

[47] M. Opper and O. Winther. Gaussian process classification and SVM: Mean field results and leave-one-out estimator. In Smola et al. [68].

[48] Manfred Opper and Ole Winther. Gaussian processes for classification: Mean field algorithms. *Neural Computation*, 12(11):2655–2684, 2000.

[49] C. Paciorek. *Nonstationary Gaussian Processes for Regression and Spatial Modelling.* PhD thesis, Carnegie Mellon University, Pittsburg, 2003.

[50] J. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In Smola et al. [68].

[51] J. Platt, C. Burges, S. Swenson, C. Weare, and A. Zheng. Learning a Gaussian process prior for automatically generating music playlists. In Dietterich et al. [15], pages 1425–1432.

[52] John C. Platt. Fast training of support vector machines using sequential minimal optimization. In Schölkopf et al. [58], pages 185–208.

[53] T. Poggio and F. Girosi. Networks for approximation and learning. *Proceedings of IEEE*, 78(9):1481–1497, 1990.

[54] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C.* Cambridge University Press, 2nd edition, 1992.

[55] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[56] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000.

[57] P. Rujan. A fast method for calculating the perceptron with maximal stability. *Journal de Physique I*, 3:277–290, 1993.

[58] B. Schölkopf, C. Burges, and A. Smola, editors. *Advances in Kernel Methods: Support Vector Learning.* MIT Press, 1998.

[59] Bernhard Schölkopf and Alexander Smola. *Learning with Kernels.* MIT Press, 1st edition, 2002.

[60] I. J. Schönberg. Metric spaces and completely monotone functions. In *Proc. Nat. Acad. Sci.*, volume 39, pages 811–841, 1938.

[61] I. J. Schönberg. Spline functions and the problem of graduation. *Annals of Mathematicals*, 52:947–950, 1964.

[62] M. Seeger. Covariance kernels from Bayesian generative models. In Dietterich et al. [15], pages 905–912.

[63] M. Seeger. *Bayesian Gaussian Process Models: PAC-Bayesian Generalisation Error Bounds and Sparse Approximations.* PhD thesis, University of Edinburgh, July 2003. See `www.cs.berkeley.edu/~mseeger`.

[64] Matthias Seeger. Bayesian methods for support vector machines and Gaussian processes. Master's thesis, University of Karlsruhe, Germany, 1999. See `www.cs.berkeley.edu/~mseeger`.

[65] Matthias Seeger. Bayesian model selection for support vector machines, Gaussian processes and other kernel classifiers. In Solla et al. [72], pages 603–609.

[66] Matthias Seeger. Covariance kernels from Bayesian generative models. Technical report, Institute for ANC, Edinburgh, UK, 2000. See `www.cs.berkeley.edu/~mseeger`.

[67] Matthias Seeger. PAC-Bayesian generalization error bounds for Gaussian process classification. *Journal of Machine Learning Research*, 3:233–269, October 2002.

[68] A. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, editors. *Advances in Large Margin Classifiers.* MIT Press, 1999.

[69] A. Smola, Z. Óvári, and R. Williamson. Regularization with dot-product kernels. In Leen et al. [32], pages 308–314.

[70] A. Smola, B. Schölkopf, and K.-R. Müller. The connection between regularization operators and support vector kernels. *Neural Networks*, 11:637–649, 1998.

[71] E. Solak, R. Murray-Smith, W. Leithead, and C. Rasmussen. Derivative observations in Gaussian process models of dynamic systems. In Becker et al. [5]. To appear.

[72] S. Solla, T. Leen, and K.-R. Müller, editors. *Advances in Neural Information Processing Systems 12.* MIT Press, 2000.

[73] Peter Sollich. Probabilistic methods for support vector machines. In Solla et al. [72], pages 349–355.

[74] M. Stein. *Interpolation of Spatial Data: Some Theory for Kriging.* Springer, 1999.

[75] I. Steinwart. On the influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, 2:67–93, 2001.

[76] Martin Szummer and Tommi Jaakkola. Partially labeled classification with Markov random walks. In Dietterich et al. [15], pages 945–952.

[77] J. Tenenbaum, A. de Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, 2000.

[78] Michael Tipping. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

[79] F. Vivarelli and C. K. I. Williams. Discovering hidden features with Gaussian process regression. In Kearns et al. [26].

[80] Grace Wahba. *Spline Models for Observational Data.* CBMS-NSF Regional Conference Series. SIAM Society for Industrial and Applied Mathematics, 1990.

[81] Grace Wahba. Support vector machines, reproducing kernel Hilbert spaces and the randomized GACV. In Schölkopf et al. [58], pages 69–88.

[82] M. Wainwright, E. Sudderth, and A. Willsky. Tree-based modeling and estimation of Gaussian processes on graphs with cycles. In Leen et al. [32], pages 661–667.

[83] Y. Weiss and W. Freeman. Correctness of belief propagation in Gaussian graphical models of arbitrary topology. In Solla et al. [72], pages 673–679.

[84] C. Williams. Computation with infinite neural networks. *Neural Computation*, 10(5):1203–1216, 1998.

[85] Christopher K. I. Williams. Prediction with Gaussian processes: From linear regression to linear prediction and beyond. In M. I. Jordan, editor, *Learning in Graphical Models*. Kluwer, 1997.

[86] Christopher K. I. Williams and David Barber. Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1342–1351, 1998.

[87] S. Wright. Modified Cholesky factorizations in interior-point algorithms for linear programming. *SIAM Journal on Optimization*, 9(4):1159–1191, 1999.

[88] A. Yaglom. *Correlation Theory of Stationary and Related Random Functions*, volume I. Springer, 1987.

[89] H. Zhu, C. K. I. Williams, R. Rohwer, and M. Morciniec. Gaussian regression and optimal finite dimensional linear models. In C. Bishop, editor, *Neural Networks and Machine Learning*, volume 168 of *NATO ASI series*. Springer, 1998.