2 BACKGROUND

2.1 PROBABILISTIC MACHINE LEARNING

Probabilistic models have become pillars of modern ML. They are at the core of powerful frameworks that can uncover hidden structures, learn useful representations, and efficiently utilise them to make accurate predictions or generate realistic samples. Through the formalism of probability theory and Bayesian inference, probabilistic models provide a coherent framework for systematically reasoning about the unknown. Such a framework possesses notable advantages: it can quantify uncertainty in predictions, naturally handle missing data, and avoid over-fitting to spurious patterns. The probabilistic approach to ML is deeply embedded in many of its most impactful applications today.

In a probabilistic model, all quantities are treated as random variables – the data is treated as *observed*, or, *known*, variables, which are assumed to be governed by some underlying *hidden*, *latent*, or, *unknown* variables. Let $\mathcal{D}$ be the set of observed variables and $\mathcal{H}$ the set of hidden variables, with the joint density

variables, observed

variables, latent

joint density

$$p(\mathcal{D}, \mathcal{H}) = p(\mathcal{D} \mid \mathcal{H}) p(\mathcal{H}).$$

Notably, the distribution of the observed variables is assumed to be governed by the hidden variables. In particular, a *prior* density $p(\mathcal{H})$ is placed on the hidden variables $\mathcal{H}$, reflecting the beliefs about its plausible values, and to rule out absurd values that should not be entertained. Its relationship to the observed variables $\mathcal{D}$ is then defined through the *likelihood function*, or, simply, *likelihood*, $p(\mathcal{D} \mid \mathcal{H})$. Note this conditional is sometimes also referred to as the *observational model*. Now, the problem of inference in Bayesian models amounts to computing the *posterior* density $p(\mathcal{H} \mid \mathcal{D})$, the conditional probability of the hidden factors given the observed data. By Bayes' theorem,

prior

likelihood

observational model

posterior

Bayes' theorem

$$p(\mathcal{H} \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \mathcal{H}) p(\mathcal{H})}{p(\mathcal{D})}. \tag{2.1}$$

The posterior can be seen as a refinement of the prior beliefs in light of observed data. In the Bayesian learning framework, the posterior can be updated iteratively as more data or new evidence becomes available.

To compute the conditional in Equation (2.1) exactly, one must compute the denominator $p(\mathcal{D})$, often referred to as the model *evidence*, It is also known as the *marginal likelihood*, since it is obtained by

evidence

marginal likelihood

marginalising out the hidden variables from the joint density,

$$p(\mathcal{D}) = \int p(\mathcal{D} \mid \mathcal{H}) p(\mathcal{H}) \, \mathrm{d}\mathcal{H}. \tag{2.2}$$

The posterior distribution in Equation (2.1) may be useful in and of itself, but is most commonly used downstream in a number of ways, e. g., for decision-making, or as the new prior as additional data arrives, or to make predictions on unseen data $\mathcal{D}_*$,

posterior predictive

$$p(\mathcal{D}_* \mid \mathcal{D}) = \int p(\mathcal{D}_* \mid \mathcal{H}) p(\mathcal{H} \mid \mathcal{D}) \, \mathrm{d}\mathcal{H}.$$

Despite its conceptual simplicity, exact Bayesian inference is often fraught with intractabilities. Specifically, computing the evidence integral in Equation (2.2) proves to be a frequent source of difficulties for many model families. This computation can exhibit exponential time complexity, rendering it *computationally* intractable. Even with the advanced hardware available today, an unassuming polynomial time complexity is still considered computationally intractable when dealing with sufficiently large datasets. For example, as of the current writing, algorithms with a cost of $\mathcal{O}(N^3)$ are typically deemed prohibitively slow when N is on the modest order of thousands [99, 277]. Moreover, in many cases, this integral doesn't even have a closed-form expression (e. g., due to non-conjugacy), rendering it *analytically* intractable. Consequently, the accurate and efficient evaluation of the evidence integral stands as a paramount challenge when performing Bayesian inference for the vast array of complex models that dominate modern probabilistic ML.

intractability, computational

intractability, analytical

When it is not feasible to carry out exact inference, one must instead resort to approximate inference techniques. Some dominant forms of approximate inference include the Laplace approximation [155], expectation propagation (EP) [173], sampling-based approaches such as Markov chain Monte Carlo (MCMC) [187], or optimisation-based approaches such as VI [118, 276]. In this thesis, we shall focus on VI, which turns out to be a common thread that weaves together a number of seemingly disaparate research topics.

approximate inference

2.2 VARIATIONAL INFERENCE

The basic idea of variational inference (VI) is to cast inference as an optimisation problem [17]. We first specify a family $\mathcal{Q}$ of densities over the latent variables. Each member $q \in \mathcal{Q}$ is a candidate approximation to the exact posterior $p(\mathcal{H} \mid \mathcal{D})$. We then optimise over this family to find that member that minimises the Kullback–Leibler (KL) divergence to the exact posterior,

Kullback–Leibler divergence

$$q^*(\mathcal{H}) = \arg\min_{q \in \mathcal{Q}} \mathrm{KL}\left[q(\mathcal{H}) \parallel p(\mathcal{H} \mid \mathcal{D})\right]. \tag{2.3}$$

Having found the optimal approximate density $q^*(\mathcal{H})$, it can then be used as a subtitute for the exact posterior density. However, a difficulty remains – explicitly spelling out the KL divergence in Equation (2.3) reveals its dependence on $p(\mathcal{D})$, the model evidence from Equation (2.2),

$$\begin{aligned}\mathrm{KL}\left[q(\mathcal{H}) \parallel p(\mathcal{H} \mid \mathcal{D})\right] &\triangleq \mathbb{E}_{q(\mathcal{H})}\left[\log \frac{q(\mathcal{H})}{p(\mathcal{H} \mid \mathcal{D})}\right] = \mathbb{E}_{q(\mathcal{H})}\left[\log \frac{p(\mathcal{D})q(\mathcal{H})}{p(\mathcal{D}, \mathcal{H})}\right] \\ &= \log p(\mathcal{D}) + \mathbb{E}_{q(\mathcal{H})}\left[\log \frac{q(\mathcal{H})}{p(\mathcal{D}, \mathcal{H})}\right] \qquad (2.4)\end{aligned}$$

However, let's not forget that the intractability of the evidence is the *raison d'être* of approximate inference in the first place. Clearly, directly minimising the KL is infeasible, prompting the need to consider an alternative strategy.

2.2.1 *Evidence Lower Bound*

evidence lower bound

This brings us to the well-known evidence lower bound (ELBO) objective, which is defined as

$$\mathrm{ELBO}(q) \triangleq \mathbb{E}_{q(\mathcal{H})}\left[\log \frac{p(\mathcal{D}, \mathcal{H})}{q(\mathcal{H})}\right]. \qquad (2.5)$$

Crucially, as the name suggests, the ELBO is a lower bound on the model evidence. In particular, adding $\mathrm{ELBO}(q)$ to both sides of Equation (2.4), we get

$$\log p(\mathcal{D}) = \mathrm{ELBO}(q) + \mathrm{KL}\left[q(\mathcal{H}) \parallel p(\mathcal{H} \mid \mathcal{D})\right].$$

Hence, the ELBO consists of the negative KL divergence and the log marginal likelihood, which is a constant wrt $q(\mathcal{H})$. Thus seen, maximising the ELBO is equivalent to minimising the KL divergence in Equation (2.3). Moreover, since the KL divergence is nonnegative, $\mathrm{KL}\left[\cdot \parallel \cdot\right] \geq 0$, it further follows that the ELBO is a lower bound on the log marginal likelihood, $\log p(\mathcal{D}) \geq \mathrm{ELBO}(q)$, for any $q \in \mathcal{Q}$. This bound can also be derived using Jensen's inequality, as originally shown by Jordan et al. [118].

We can expand the ELBO as

$$\mathrm{ELBO}(q) = \mathbb{E}_{q(\mathcal{H})}[\log p(\mathcal{D} \mid \mathcal{H})] - \mathrm{KL}\left[q(\mathcal{H}) \parallel p(\mathcal{H})\right]. \qquad (2.6)$$

The first term in Equation (2.6) is commonly referred to as the expected log-likelihood (ELL), while the second term is the negative KL between the approximate posterior $q(\mathcal{H})$ and prior $p(\mathcal{H})$. The ELL term encourages the approximate density to place its mass on configurations of the latent variables that explain the observed data, while the negative KL divergence term encourages densities that resemble the prior. Combined, these terms constitute the ELBO and reflect the

expected log-likelihood

usual balance between the likelihood and prior – and between data fit and regularisation.

Under benign conditions, the solution $q^*(\mathcal{H})$ to the optimisation problem outlined in Equation (2.3) can be derived analytically. An illustrative example of this is found in GPs, a widely used family of models that we shall formally introduce in Section 2.4. Specifically, in the sparse GP regression (SGPR) framework discussed in Section 2.4.2.3, the optimal $q^*(\mathcal{H})$ has a closed-form expression. However, in most cases, $q^*(\mathcal{H})$ is obtained through a hill-climbing optimisation procedure, specifically, gradient ascent, applied to an analytical form of the ELBO. This approach is employed in the more general sparse variational GP (SVGP) framework, presented in Section 2.4.2.1, where the likelihood is not (necessarily) Gaussian. If the likelihood factorises, the use of mini-batch training for stochastic optimisation [105], as explained in Section 2.4.2.2, allows for scaling to massive datasets.

More generally, in other scenarios, such as VI with blackbox likelihoods [209], discrete hidden variables [114, 158], or implicit distributions [110, 267], one or more components of the ELBO may lack analytical tractability and thus necessitate further approximations. Chapter 3 of this thesis focuses on improving inference in the SVGP framework through the use of NN basis functions, Chapter 4 examines a new kind of VI scheme designed to handle implicit distributions, and Appendix A explores the efficient posterior sampling of GPs and their sparse variational approximations.

For a complete resource on the foundations of VI, we refer the interested reader to the review article of Blei, Kucukelbir, and McAuliffe [17], now a contemporary classic.

2.3 STATISTICAL DIVERGENCES AND DENSITY-RATIO ESTIMATION

Statistical divergences quantify the dissimilarity between probability distributions and are essential in probabilistic ML. In the preceding section on variational inference, we saw a prime example of one such divergence, namely, the well-known KL divergence. In fact, the KL divergence is just one of many divergences that belong to a larger family of statistical divergences known as the f-divergences [48, 146], also known as the Ali-Silvey distances [2]. For a convex, lower-semicontinuous function $f : \mathbb{R}_+ \to \mathbb{R}$ satisfying $f(1) = 0$, the f-divergence between two distributions with probability densities $p(\mathbf{x})$ and $q(\mathbf{x})$ is defined as

f-divergence

$$\mathcal{D}_f\left[p(\mathbf{x}) \parallel q(\mathbf{x})\right] \triangleq \mathbb{E}_{q(\mathbf{x})}\left[f\left(\frac{p(\mathbf{x})}{q(\mathbf{x})}\right)\right]. \tag{2.7}$$

For instance, the familiar KL divergence that appears extensively in VI – more precisely, the *reverse*[1] KL divergence $\mathrm{KL}\left[q \parallel p\right]$ – is obtained as a special case of Equation (2.7) under the setting $f : u \mapsto -\log u$. At the heart of Equation (2.7) is the fraction, or, *ratio*,

$$r(\mathbf{x}) \triangleq \frac{p(\mathbf{x})}{q(\mathbf{x})} \tag{2.8}$$

with density $p(\mathbf{x})$ as the numerator and $q(\mathbf{x})$ as the denominator. This crucial quantity is referred to as the *density-ratio* of $p(\mathbf{x})$ and $q(\mathbf{x})$. The density-ratio is known variously in other parts of the literature as the "likelihood ratio," or the "importance weight". Clearly, when either one or both of the densities are unavailable in analytical form, either due to intractabilities or intentional modelling choices, the f-divergence from Equation (2.7) will also be analytically intractable. Perhaps the most notable case of such intractabilities is in the framework of generative adversarial networks (GANs) [86], in which the underlying goal is to minimise some f-divergence between two distributions where neither admits a tractable density, and one must therefore rely solely on their samples [177, 191].

density-ratio

More broadly, the problem of DRE is concerned with approximating density-ratios when no information is available from distributions p or q other than their samples. The DRE problem is pervasive throughout ML and arises in a impressively diverse range of contexts, e. g., in covariate shift adaptation [15, 250, 268], energy-based models (EBMS) [90, 92, 269], VI [110, 172], likelihood-free inference [64, 257, 267], mutual information estimation [11], bias-correction for generative models [39, 91], and Bayesian experimental design (BED) [129, 130]. Chapter 5 of this thesis demonstrates how DRE arises in the context of BO [241, 259], a close cousin of BED. Furthermore, as alluded to earlier, Chapter 4 discusses a novel VI approach that relies heavily on DRE to deal with implicit distributions.

The most obvious but naïve approach to tackling the DRE problem is to separately estimate the densities $p(\mathbf{x})$ and $q(\mathbf{x})$ using, e. g., kernel density estimation (KDE) [233], and then to use their ratio as an approximation to the unknown true density-ratio. Not surprisingly, this approach suffers from a large host of issues, most of which are well-documented by Sugiyama, Suzuki, and Kanamori [251]. We discuss these at further length in Chapter 5, with an added emphasis on the drawbacks that most impact applications in global optimisation.

Not surprisingly, there is a substantial body of existing works on DRE [251]. Recognising the deficiencies of the naïve KDE approach, a myriad alternatives have since been proposed, including KL importance estimation procedure (KLIEP) [250], kernel mean matching (KMM) [88], unconstrained least-squares importance fitting (ULSIF) [122], and relative ULSIF (RULSIF) [292]. In this thesis, we shall primarily focus

1 the KL divergence is asymmetric

on CPE, introduced in Section 2.3.2, an effective and versatile approach that has found widespread adoption in a diverse range of contexts such as those mentioned above.

2.3.1 *Variational Divergence Estimation*

The problem of estimating statistical divergences and, by extension, density-ratios, using only samples [188, 191] can be effectively tackled by leveraging the framework of convex analysis [213]. Convex analysis is a vast topic in its own right. For a light and intuitive introduction to convex duality (albeit applied in a different context), the reader is envouraged to to consult the self-contained section from the text of Bishop [16, §10.5]. Now, every convex, lower-semicontinuous function f has a convex dual $f^\star$, also known as the Fenchel conjugate [213]. More precisely, function f and its convex dual $f^\star$ are related as follows,

convex dual
Fenchel conjugate

$$f(u) = \max_t \{ut - f^\star(t)\}, \qquad f^\star(t) = \max_u \{ut - f(u)\}. \tag{2.9}$$

The convex dual is *involutory*, meaning that the convex dual of $f^\star$ is simply $f^{\star\star} = f$. Since f is convex, its first derivative f' is strictly nondecreasing. Therefore, we can reparameterise the variational formulation of $f(u)$ from Equation (2.9) by substituting t with $f'(s)$ (for some s in the domain of f'),

$$f(u) = \max_s \left\{ uf'(s) - f^\star(f'(s)) \right\}.$$

Substituting this into the f-divergence from Equation (2.7) and invoking Jensen's inequality gives the lower bound

$$\mathcal{D}_f \left[p(\mathbf{x}) \parallel q(\mathbf{x}) \right] \geq \max_{\boldsymbol{\theta}} \left\{ \mathbb{E}_{p(\mathbf{x})}[f'(r_{\boldsymbol{\theta}}(\mathbf{x}))] - \mathbb{E}_{q(\mathbf{x})}[f^\star(f'(r_{\boldsymbol{\theta}}(\mathbf{x})))] \right\}, \tag{2.10}$$

where $r_{\boldsymbol{\theta}} : \mathcal{X} \to \mathbb{R}_+$ is some mapping with parameters $\boldsymbol{\theta}$. This is a powerful bound with far-reaching implications. Firstly, observe that this lower bound objective does not strictly rely on the densities $p(\mathbf{x})$ and $q(\mathbf{x})$ – to efficiently maximise this objective in practice, e. g., using stochastic gradients with the reparameterisation trick, we need only be able to draw samples from $p(\mathbf{x})$ and $q(\mathbf{x})$. Secondly, some straightforward calculus of variations shows that the bound is tighest when $r_{\boldsymbol{\theta}}(\mathbf{x}) = r(\mathbf{x})$, i. e., when the parameterised mapping is precisely the density-ratio introduced in Equation (2.8). In other words, optimising the objective in Equation (2.10) to obtain a tight lower bound directly goes hand-in-hand with obtaining an accurate estimate of the density-ratio.

2.3.2 *Class-Probability Estimation*

We've just discussed a general framework for simulatenously estimating divergences and addressing the DRE problem. Let's now consider

a prominent special case of this known as density-ratio estimation by class-probability estimation (CPE) [15, 37, 170, 203, 251]. Let $\pi_{\boldsymbol{\theta}}$ be a probabilistic classifier: a mapping $\pi_{\boldsymbol{\theta}} : \mathcal{X} \to [0,1]$ parameterised by $\boldsymbol{\theta}$. Recall the well-known binary cross-entropy (BCE) loss, also known as the log loss, prevalent in binary classification,

binary cross-entropy

$$\mathcal{L}(\boldsymbol{\theta}) \triangleq -\mathbb{E}_{p(\mathbf{x})}[\log \pi_{\boldsymbol{\theta}}(\mathbf{x})] - \mathbb{E}_{q(\mathbf{x})}[\log(1 - \pi_{\boldsymbol{\theta}}(\mathbf{x}))]. \tag{2.11}$$

Interestingly, there is a lower bound on the BCE loss [86] that can be expressed in terms of an f-divergence, namely, the Jensen-Shannon (JS) divergence $\mathcal{D}_{\text{JS}}[p \parallel q]$, which is a symmetrised variant of the KL divergence,

Jensen-Shannon divergence

$$\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) \geq -2\left(\mathcal{D}_{\text{JS}}[p(\mathbf{x}) \parallel q(\mathbf{x})] - \log 2\right).$$

To see this, let's first parameterise the classifier as

$$\pi_{\boldsymbol{\theta}}(\mathbf{x}) \triangleq \sigma(\log r_{\boldsymbol{\theta}}(\mathbf{x})), \tag{2.12}$$

where σ denotes the logistic sigmoid function and $r_{\boldsymbol{\theta}}$ is some function parameterised by $\boldsymbol{\theta}$. The intermediate (pre-activation) output $\log r_{\boldsymbol{\theta}}(\mathbf{x})$ is known as the *logits*, or *log-odds*.

logits, log-odds

In the special case of

$$f_{\text{BCE}}(u) \triangleq u \log u - (u+1)\log(u+1) \tag{2.13}$$

in Equation (2.10), we get

$$\begin{aligned}
2\left(\mathcal{D}_{\text{JS}}[p(\mathbf{x}) \parallel q(\mathbf{x})] - \log 2\right) &= \mathcal{D}_{f_{\text{BCE}}}[p(\mathbf{x}) \parallel q(\mathbf{x})] \\
&\geq \max_{\boldsymbol{\theta}} \left\{\mathbb{E}_{p(\mathbf{x})}[\log \sigma(\log r_{\boldsymbol{\theta}}(\mathbf{x}))] + \mathbb{E}_{q(\mathbf{x})}[\log(1 - \sigma(\log r_{\boldsymbol{\theta}}(\mathbf{x})))]\right\} \\
&= \max_{\boldsymbol{\theta}} \{-\mathcal{L}(\boldsymbol{\theta})\} = -\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}),
\end{aligned}$$

and negating both sides gives the desired bound. Like in Equation (2.10), the BCE loss is minimised when $r_{\boldsymbol{\theta}}(\mathbf{x}) = r(\mathbf{x})$, or equivalently when

$$\pi_{\boldsymbol{\theta}}(\mathbf{x}) = \sigma(\log r(\mathbf{x})) = \frac{p(\mathbf{x})}{p(\mathbf{x}) + q(\mathbf{x})},$$

where $r(\mathbf{x})$ is the true density-ratio defined in Equation (2.8). Importantly, this provides a straightforward means of recovering a density-ratio estimator from a probabilistic classifier

$$r_{\boldsymbol{\theta}}(\mathbf{x}) = \exp \sigma^{-1}(\pi_{\boldsymbol{\theta}}(\mathbf{x})) = \frac{\pi_{\boldsymbol{\theta}}(\mathbf{x})}{1 - \pi_{\boldsymbol{\theta}}(\mathbf{x})},$$

and vice versa. Thus, we've obtained a direct way of casting the problem of DRE as the well-studied problem of CPE. Furthermore, this general approach is not restricted only to the BCE loss but extends to any other proper scoring rule [85] that produce well-calibrated probabilistic predictions, such as the hinge loss [218].

proper scoring rule

The CPE approach described here constitutes the predominant approach to DRE. It's not difficult to imagine why, considering the veritable cornucopia of user-friendly, off-the-shelf software frameworks that are available for supervised learning. Notable examples include scikit-learn [199] a versatile library covering a wide range of different paradigms, as well as specialised libraries like XGBoost [34] for decision tree ensembles with extreme gradient-boosting (XGBOOST), and PyTorch [197]/Lightning and TensorFlow [1]/Keras [40] for deep neural networks (DNNS), to name just a few. These frameworks have made it easier than ever to train powerful classifiers, driving the widespread adoption of the CPE approach to tackling the problem of DRE.

TOY 1D EXAMPLE. Consider the following toy example where the densities $\ell(x)$ and $g(x)$ are *known* and given exactly by the following (mixture of) Gaussians,

$$\ell(x) \triangleq 0.3\mathcal{N}(2, 1^2) + 0.7\mathcal{N}(-3, 0.5^2), \qquad \text{and} \qquad g(x) \triangleq \mathcal{N}(0, 2^2),$$

as illustrated by the *solid red* and *blue* lines in Figure 2.1, respectively.

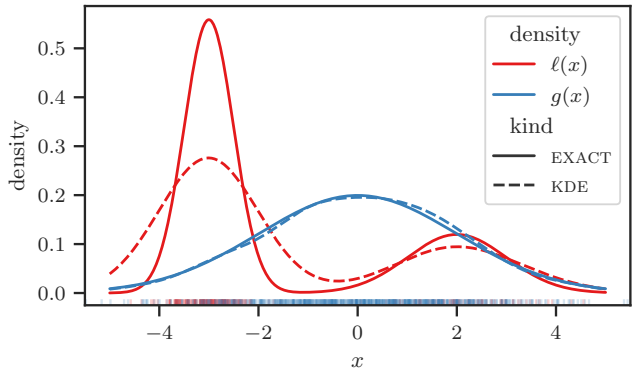
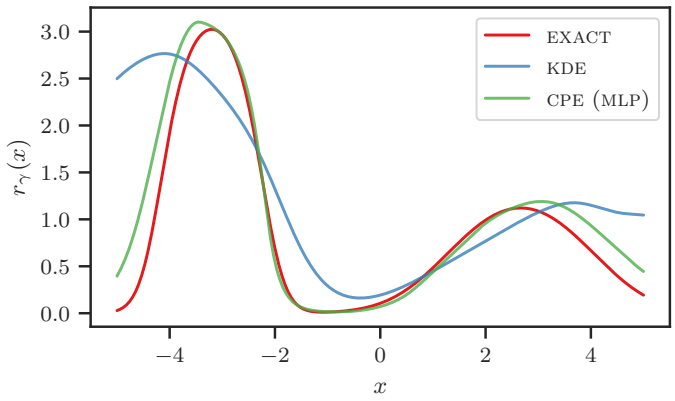


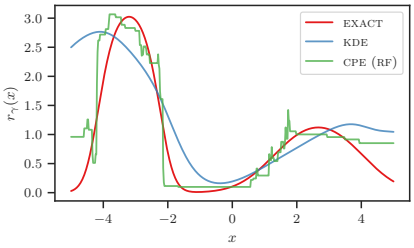
Figure 2.1: Densities $\ell(x)$ and $g(x)$ and their (kernel density) estimates.

We draw a total of $N = 1,000$ samples from these distributions, with a fraction $\gamma = 1/4$ drawn from $\ell(x)$ and the remainder from $g(x)$. These are represented by the vertical markers along the bottom of the x-axis (a so-called "rug plot"). Then, two KDEs, shown with *dashed* lines, are fit on these respective sample sets, with kernel bandwidths selected according to the "normal reference" rule-of-thumb. We see that, for both densities, the modes are recovered well, while for $\ell(x)$, the variances are overestimated in both of its mixture components. As we shall see, this has deleterious effects on the resulting density-ratio estimate.

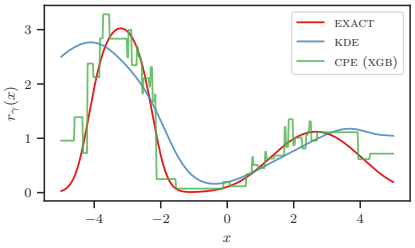
In Figure 2.2a, we represent the true *relative* density-ratio with the *red* line. Note that the relative density-ratio, as we shall see in Section 5.2.1, is a generalisation of the *ordinary* density-ratio we introduced at the beginning of this section. For the purposes of the present discussion,



(a) The relative density-ratio, estimated with an MLP classifier.



(b) The relative density-ratio, estimated with a RF classifier.



(c) The relative density-ratio, estimated with an XGBOOST classifier.

Figure 2.2: Synthetic toy example with (mixtures of) Gaussians.

its precise definition is immaterial as the same analysis applies both to relative and ordinary density-ratios. The estimate resulting from taking the ratio of the KDEs is shown in *blue*, while that of the CPE method described in this section is shown in *green*. In this subfigure, the probabilistic classifier consists of a simple MLP with 3 hidden layers, each with and 32 units and `elu` activations. In Figures 2.2b and 2.2c, we show the same, but with RF and XGBOOST classifiers.

The CPE methods appear, at least visually, to recover the exact density ratios well, whereas the KDE method does so quite poorly. Perhaps the more important quality to focus on, particularly if used in the context of global optimisation as in Chapter 5, is the *mode* of the density-ratio functions. In the case of the KDE method, we can see that this deviates significantly from that of the true density-ratio. In this instance, although KDE fit $g(x)$ well and recovered the modes of $\ell(x)$ accurately, even a slight overestimation of the variance in the latter led to a significant shift in the maximiser of the resulting density-ratio functions.

2.4 GAUSSIAN PROCESSES

We now shift gears and turn our focus to Gaussian processes (GPs), a class of nonparametric Bayesian models that provide a powerful framework for reasoning about unknown functions. GPs are ubiquitous in probabilistic ML [156]. They exhibit remarkable data efficiency, achieving high accuracy even with limited data. Moreoever, they inherently possess mechanisms that help to mitigate over-fitting, and can flexibly encode prior beliefs and assumptions through their covariance function. Last but not least, by virtue of their ability to faithfully capture predictive uncertainty, they form the backbone of many sequential decision-making procedures that require reliable uncertainty estimates to appropriately balance important trade-offs such as that of *exploration* and *exploitation*, for instance, in active learning [108], reinforcement learning [55], Bayesian optimisation [20, 75, 228] (covered in-depth separately in Section 2.5), probabilistic numerics [95], and more.

2.4.1 *Gaussian Process Regression*

More formally, GPs are a flexible class of distributions over functions. A random function $f : \mathcal{X} \to \mathbb{R}$ on some domain $\mathcal{X} \subseteq \mathbb{R}^D$ is distributed according to a GP if, at any finite collection of input locations $\mathbf{X}_* \subseteq \mathcal{X}$, its values $\mathbf{f}_* = f(\mathbf{X}_*)$ follow a Gaussian distribution. A GP is fully determined by its covariance function $k(\mathbf{x}, \mathbf{x}')$ and mean function, which can be assumed without loss of generality to be constant (e. g., zero).

covariance function

Consider a supervised learning problem in which we have a dataset $\{\mathbf{x}_n, y_n\}_{n=1}^N$ consisting of scalar outputs y_n, which are related to $f_n \triangleq f(\mathbf{x}_n)$, the value of some unknown function $f(\cdot)$ at input $\mathbf{x}_n \in \mathcal{X}$, through the likelihood $p(y_n \mid f_n)$. A powerful modelling approach consists of specifying a GP prior on the latent function $f(\cdot)$,

$$f(\mathbf{x}) \sim \mathcal{GP}\left(0, k(\mathbf{x}, \mathbf{x}')\right). \tag{2.14}$$

Let $\mathbf{X}$ denote the inputs, $\mathbf{f}$ the corresponding latent function values, and $\mathbf{y}$ the outputs. In the regression setting, the outputs $\mathbf{y}$ are assumed to be noisy observations of the latent values $\mathbf{f}$, typically related through a Gaussian likelihood

$$p(\mathbf{y} \mid \mathbf{f}, \beta) = \mathcal{N}(\mathbf{y} \mid \mathbf{f}, \beta^{-1}\mathbf{I}), \tag{2.15}$$

for some *precision* $\beta > 0$.

precision

Under this likelihood, the posterior predictive density $p(\mathbf{f}_* \mid \mathbf{y})$ at test inputs is has the closed-form expression

$$p(\mathbf{f}_* \mid \mathbf{y}) = \mathcal{N}\left(\mathbf{K}_{*\mathbf{f}}(\mathbf{K}_{\mathbf{ff}} + \beta^{-1}\mathbf{I})^{-1}\mathbf{y}, \mathbf{K}_{**} - \mathbf{K}_{*\mathbf{f}}(\mathbf{K}_{\mathbf{ff}} + \beta^{-1}\mathbf{I})^{-1}\mathbf{K}_{\mathbf{f}*}\right). \tag{2.16}$$

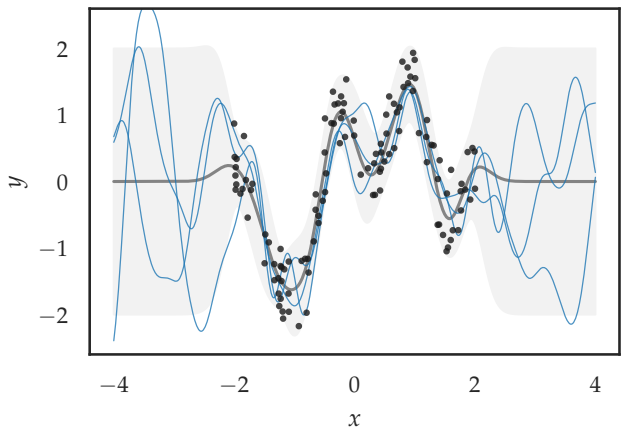


Figure 2.3: Gaussian process (GP) posterior predictive density on the synthetic one-dimensional SNELSON1D dataset [235]; three random functions sampled from this density are indicated by the blue curves.

Clearly, evaluating this density has a time complexity of $\mathcal{O}(N^3)$, which stems from the costs associated with calculating the matrix inverse of $\mathbf{K_{ff}} + \beta^{-1}\mathbf{I}$. Furthermore, for other (i. e., non-Gaussian) likelihoods the closed-form expression for $p(\mathbf{f}_* \mid \mathbf{y})$ is generally unavailable.

2.4.1.1 *Covariance Functions*

The covariance function holds a pivotal role in GP models, as it encapsulates prior beliefs and assumptions about the latent function of interest. It provides a means to encode various characteristics such as periodicity, roughness, and smoothness (or, to be more precise, *orders of differentiability*), etc. Specifically, let us examine the family of *stationary* covariance functions, which are translation invariant in the input space. In other words, $k_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{x}')$ only depends on the difference $\mathbf{x} - \mathbf{x}'$ between the input locations $\mathbf{x}$ and $\mathbf{x}'$. This can be expressed mathematically as

stationarity

$$k_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{x}') = \kappa_{\boldsymbol{\theta}}(\mathbf{x} - \mathbf{x}'),$$

for some function $\kappa_{\boldsymbol{\theta}}$, where $\boldsymbol{\theta}$ consists of some collection of parameters. The use of a stationary covariance function reflects the assumption that the relationship between $f(\mathbf{x})$ and $f(\mathbf{x}')$ is fully characterised by the difference between $\mathbf{x}$ and $\mathbf{x}'$. In particular, consider the squared exponential (SE) kernel, or, the exponentiated quadratic kernel, which can be expressed in terms of function $\kappa_{\boldsymbol{\theta}}$ of the difference $t \triangleq x - x'$,

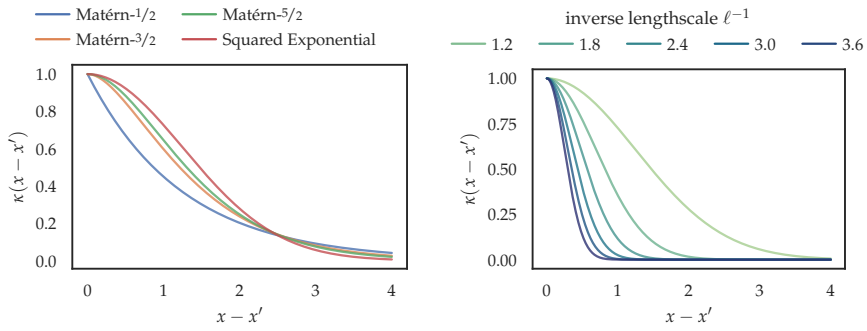
squared exponential kernel

$$\kappa_{\boldsymbol{\theta}}(t) = \sigma_f^2 \exp\left(-\frac{t^2}{2\ell^2}\right), \tag{2.17}$$

where the parameters $\boldsymbol{\theta} \triangleq \{\ell, \sigma_f^2\}$ are made up of the *characteristic lengthscale* ℓ and the variance, or, *amplitude*, σ_f^2. Generalising the

characteristic lengthscale

amplitude



(a) Profile of various covariance functions with characteristic lengthscale $\ell = 5/4$.

(b) The squared exponential (SE) covariance function with varying characteristic lengthscales.

Figure 2.4: Several widely-used stationary covariance functions.

squared exponential (SE) kernel to D dimensions, we have

$$\kappa_{\boldsymbol{\theta}}(\mathbf{t}) = \sigma_f^2 \exp\left(-\frac{1}{2}\mathbf{t}^\top \boldsymbol{\Lambda}^{-1}\mathbf{t}\right), \tag{2.18}$$

for some nonsingular matrix $\boldsymbol{\Lambda}$. The most common and arguably useful choice for $\boldsymbol{\Lambda}$ is the diagonal matrix,

$$\boldsymbol{\Lambda} \triangleq \operatorname{diag}(\ell_1^2, \ldots, \ell_D^2),$$

consisting of the characteristic lengthscales $\ell_1, \ldots, \ell_D$, each associated with an input dimension. Intuitively, each lengthscale dictates how close the input location needs to be (along the associated dimension) for the function values to exhibit high correlation. This effectively implements the functionality known as automatic relevance determination (ARD) [185], because the relevance of an input dimension is inversely proportional to the corresponding lengthscale – that is, an input dimension with a large associated lengthscale will have virtually no influence on the covariance, effectively disregarding its variations during inference [286]. The SE covariance function is infinitely differentiable, implying that the latent function $f(\mathbf{x})$ will have derivatives of all orders. However, assuming such a degree of smoothness is often unreasonable for many applications. For this reason, many practitioners appeal to the Matérn family of covariance functions [247], which were originally named after Matérn [164]. This family of functions offers greater flexibility in modelling various degrees of smoothness, which can be adjusted by specifying a smoothness parameter ν.

automatic relevance determination

Matérn kernel

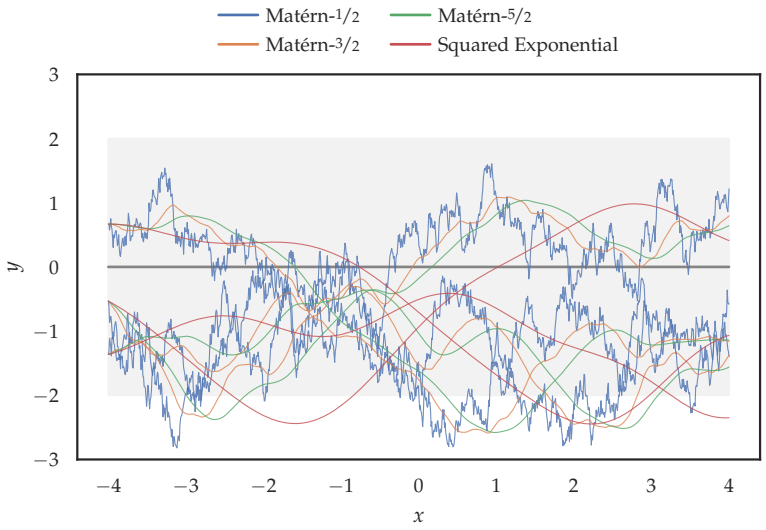


Figure 2.5: Gaussian process (GP) prior samples resulting from different stationary covariance functions with characteristic lengthscale $\ell = 5/4$; three random samples are drawn for each covariance function.

Specifically, in the case when ν is half-integer, i. e., $\nu = \rho + 1/2$ for some nonnegative integer ρ, the Matérn-ν covariance function can be expressed as

$$\begin{aligned}\kappa_{\boldsymbol{\theta}}^{(\nu)}(\mathbf{t}) =& \sigma_f^2 \exp\left(-\sqrt{2\nu\mathbf{t}^\top\mathbf{M}^{-1}\mathbf{t}}\right)\frac{\Gamma(\rho+1)}{\Gamma(2\rho+1)} \\ &\times \sum_{i=0}^{\rho}\frac{(\rho+i)!}{i!(\rho-i)!}\left(\sqrt{8\nu\mathbf{t}^\top\mathbf{M}^{-1}\mathbf{t}}\right)^{\rho-i}.\end{aligned} \tag{2.19}$$

There are a few properties worth noting here. First, the latent function $f(\mathbf{x})$ will have derivatives up to order ρ. This is consistent with the fact that we obtain the SE kernel in the limit as $\nu \to \infty$. The most interesting cases are $\nu = 1/2, 3/2, 5/2$, with the last perhaps being the most widely-used in practice. The choice of $\nu = 5/2$ signifies a prior belief that the latent function $f(\mathbf{x})$ is twice differentiable (since $\rho = 2$), which has been advocated as a helpful assumption, e. g., in the context of global optimisation [236].

2.4.1.2 *Hyperparameter Estimation*

We've already discussed how to obtain the posterior predictive density at test inputs for a given set of hyperparameters, such as $\{\boldsymbol{\theta}, \beta\}$ for noise precision β and kernel parameters $\boldsymbol{\theta}$. As one can imagine from our earlier discussions, these hyperparameters exert a large influence on the behaviour of the GP and its predictions. However, determining the appropriate values for these hyperparameters is often challenging and impractical to do manually. In most cases, when the ideal fully-

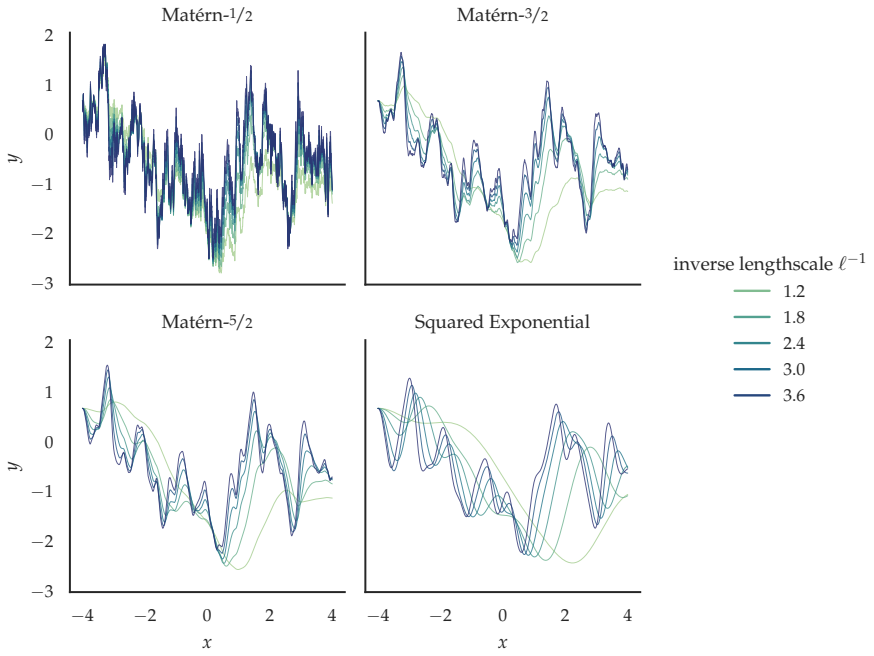


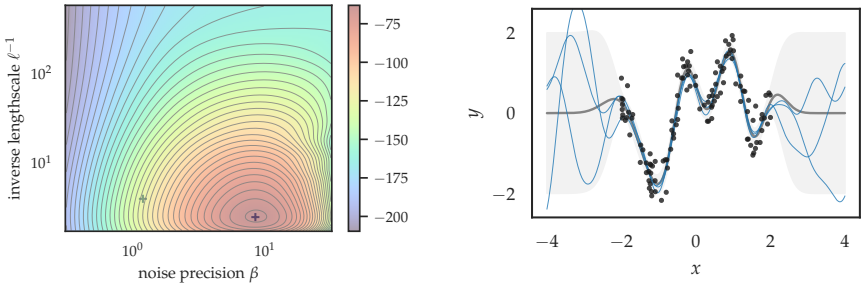
Figure 2.6: Gaussian process (GP) prior samples resulting from different stationary covariance functions with varying lengthscales; one sample is drawn for each combination.

Bayesian treatment of these hyperparemeters proves too unwieldy, it is common practice to adopt a configuration that maximises the marginal likelihood, known as type-II maximum likelihood estimation (MLE). In the regression setting we have been discussing, the marginal likelihood has the closed-form expression

$$\log p(\mathbf{y} \mid \boldsymbol{\theta}, \beta) = \log \int p(\mathbf{y} \mid \mathbf{f}, \beta) p(\mathbf{f} \mid \boldsymbol{\theta}) \, \mathrm{d}\mathbf{f} = \log \mathcal{N}(\mathbf{y} \mid \mathbf{0}, \mathbf{K}_{\mathbf{ff}} + \beta^{-1}\mathbf{I})$$
$$= -\frac{1}{2}\mathbf{y}^\top (\mathbf{K}_{\mathbf{ff}} + \beta^{-1}\mathbf{I})^{-1}\mathbf{y} - \frac{1}{2}\log|\mathbf{K}_{\mathbf{ff}} + \beta^{-1}\mathbf{I}| - \frac{N}{2}\log 2\pi. \quad (2.20)$$

The first term is a quadratic in the observations $\mathbf{y}$, which encourages a precise fit to the data. On the other hand, the second term acts a regulariser that discourages overly complex models. Consequently, optimising the hyperparameters wrt to the marginal likelihood automatically strikes a balance between data fit and model complexity, ultimately seeking the simplest model that best explains the data. It is due to this mechanism that GP models are often characterised as being inherently robust against over-fitting. However, it is important to note that the ability to mitigate over-fitting is more accurately attributed to the marginal likelihood, which is essentially what distinguishes the Bayesian inference approach from other approaches based purely on optimisation [286, §5.2].

As with the predictive density in Equation (2.16), the time complexity of evaluating the marginal likelihood is dominated by the $\mathcal{O}(N^3)$ cost of computing the matrix inverse and determinant. Furthermore,



(a) Marginal likelihood of a GP regression model with a SE covariance function and amplitude $\sigma_f = 1$ on the SNELSON1D dataset.

(b) Posterior predictive density with *optimal* hyperparameters.

Figure 2.7: Hyperparameter estimation in a GP regression model and its effects; the two '+' markers correspond to optimal and reasonably-good-but-not-quite-optimal settings of the hyperparameters (ℓ, β). The resulting posterior predictive densities, visualised in Figures 2.3 and 2.7b, respectively, reveal a clear contrast in predictive uncertainty, with the optimal hyperparameters delivering finely tuned confidence intervals.

apart from the case of the GP regression model with Gaussian noise from Equation (2.15), which serves as an exception that proves the rule, the marginal likelihood is generally analytically intractable for arguably the majority of interesting models in probabilistic ML. These two intractabilities have long been recognised as the most significant challenges in establishing the practicality and widespread adoption of GPS.

2.4.2 *Sparse Gaussian Processes*

sparse Gaussian process

A range of sparse GP methods have been developed over the years to mitigate these limitations [46, 204, 226, 234]. Broadly speaking, in sparse GPS, one summarises $f(\cdot)$ succinctly in terms of *inducing variables*, which are values $\mathbf{u} \triangleq f(\mathbf{Z})$ taken at a collection of M locations $\mathbf{Z} = [\mathbf{z}_1 \cdots \mathbf{z}_M]^\top$, where $\mathbf{z}_m \in \mathcal{X}$. Not least among these approaches is SVGP/SGPR, first proposed by [262], which casts sparse GPS within the framework of VI, which we described earlier in Section 2.2. In this section, we examine this framework in detail and discuss some of the extensions for blackbox likelihoods and large-scale inference with mini-batch training [57, 98, 99].

Specifically, the joint distribution of the model augmented by inducing variables $\mathbf{u}$ is $p(\mathbf{f}, \mathbf{u}, \mathbf{y}) = p(\mathbf{y} \mid \mathbf{f}) p(\mathbf{f}, \mathbf{u})$, where the joint over $(\mathbf{f}, \mathbf{u})$ factorises as $p(\mathbf{f}, \mathbf{u}) = p(\mathbf{f} \mid \mathbf{u}) p(\mathbf{u})$. The prior $p(\mathbf{u})$ is

$$p(\mathbf{u}) = \mathcal{N}(\mathbf{0}, \mathbf{K}_{\mathbf{uu}}), \tag{2.21}$$

and the conditional $p(\mathbf{f} \mid \mathbf{u})$ is

$$p(\mathbf{f} \mid \mathbf{u}) = \mathcal{N}(\mathbf{f} \mid \mathbf{Q}_{\mathbf{fu}}\mathbf{u}, \mathbf{K}_{\mathbf{ff}} - \mathbf{Q}_{\mathbf{ff}}), \tag{2.22}$$

where $\mathbf{Q}_{\mathbf{ff}} \triangleq \mathbf{Q}_{\mathbf{fu}}\mathbf{K}_{\mathbf{uu}}\mathbf{Q}_{\mathbf{uf}}$ and $\mathbf{Q}_{\mathbf{fu}} \triangleq \mathbf{K}_{\mathbf{fu}}\mathbf{K}_{\mathbf{uu}}^{-1}$. The joint variational distribution is defined as $q(\mathbf{f}, \mathbf{u}) \triangleq q(\mathbf{f} \mid \mathbf{u})q(\mathbf{u})$ where

$$q(\mathbf{u}) \triangleq \mathcal{N}(\mathbf{m}_{\mathbf{u}}, \mathbf{C}_{\mathbf{u}}) \tag{2.23}$$

for variational parameters $\mathbf{m}_{\mathbf{u}} \in \mathbb{R}^M$ and $\mathbf{C}_{\mathbf{u}} \in \mathbb{R}^{M \times M}$ s.t. $\mathbf{C}_{\mathbf{u}} \succeq 0$. Commonly, for convenience, one simply defines $q(\mathbf{f} \mid \mathbf{u}) \triangleq p(\mathbf{f} \mid \mathbf{u})$. At unseen points $\mathbf{f}_* \triangleq f(\mathbf{X}_*)$, integrating out $\mathbf{u}$ leads to the test predictive density

$$q(\mathbf{f}_*) = \mathcal{N}(\mathbf{f}_* \mid \mathbf{Q}_{*\mathbf{u}}\mathbf{m}_{\mathbf{u}}, \mathbf{K}_{**} - \mathbf{Q}_{*\mathbf{u}}(\mathbf{K}_{\mathbf{uu}} - \mathbf{C}_{\mathbf{u}})\mathbf{Q}_{\mathbf{u}*}), \tag{2.24}$$

where parameters $\mathbf{m}_{\mathbf{u}}$ and $\mathbf{C}_{\mathbf{u}}$ are learned by minimising the KL divergence between the approximate and exact posteriors, $\mathrm{KL}\left[q(\mathbf{f}^*, \mathbf{f}, \mathbf{u}) \parallel p(\mathbf{f}^*, \mathbf{f}, \mathbf{u} \mid \mathbf{y})\right]$. Conveniently, since the posteriors factorise as

$$q(\mathbf{f}^*, \mathbf{f}, \mathbf{u}) = p(\mathbf{f}^* \mid \mathbf{f}, \mathbf{u})q(\mathbf{f}, \mathbf{u}),$$

and

$$p(\mathbf{f}^*, \mathbf{f}, \mathbf{u} \mid \mathbf{y}) = p(\mathbf{f}^* \mid \mathbf{f}, \mathbf{u}, \cancel{\mathbf{y}})p(\mathbf{f}, \mathbf{u} \mid \mathbf{y}),$$

the common factor $p(\mathbf{f}^* \mid \mathbf{f}, \mathbf{u})$ cancels each other to simplify the KL,

$$\mathrm{KL}\left[q(\mathbf{f}^*, \mathbf{f}, \mathbf{u}) \parallel p(\mathbf{f}^*, \mathbf{f}, \mathbf{u} \mid \mathbf{y})\right] = \mathrm{KL}\left[q(\mathbf{f}, \mathbf{u}) \parallel p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})\right].$$

Refer to Section 2.A for details. Now, by Bayes' rule, we have

$$\begin{aligned} &\mathrm{KL}\left[q(\mathbf{f}, \mathbf{u}) \parallel p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})\right] && (2.25)\\ &= \iint q(\mathbf{f}, \mathbf{u}) \log \frac{q(\mathbf{f}, \mathbf{u})}{p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})} \, \mathrm{d}\mathbf{f}\mathrm{d}\mathbf{u} \\ &= \log p(\mathbf{y}) - \iint q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{f}, \mathbf{u}, \mathbf{y})}{q(\mathbf{f}, \mathbf{u})} \, \mathrm{d}\mathbf{f}\mathrm{d}\mathbf{u}. && (2.26) \end{aligned}$$

The astute reader might find this familiar, as it an instance of the general expression we examined in Section 2.2. Indeed, if we define the ELBO as

$$\mathrm{ELBO}(q) \triangleq \iint q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{f}, \mathbf{u}, \mathbf{y})}{q(\mathbf{f}, \mathbf{u})} \, \mathrm{d}\mathbf{f}\mathrm{d}\mathbf{u},$$

then, upon re-arranging Equation (2.26), we get

$$\log p(\mathbf{y}) = \mathrm{ELBO}(q) + \mathrm{KL}\left[q(\mathbf{f}, \mathbf{u}) \parallel p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})\right].$$

Now, since $p(\mathbf{f}, \mathbf{u}, \mathbf{y})$ factorises as

$$p(\mathbf{f}, \mathbf{u}, \mathbf{y}) = p(\mathbf{y} \mid \mathbf{f}, \cancel{\mathbf{u}})p(\mathbf{f}, \mathbf{u}) = p(\mathbf{y} \mid \mathbf{f})p(\mathbf{f} \mid \mathbf{u})p(\mathbf{u}),$$

we can simplify the ELBO to

$$\begin{aligned}\text{ELBO}(q) &= \iint p(\mathbf{f} \mid \mathbf{u}) q(\mathbf{u}) \log \frac{p(\mathbf{y} \mid \mathbf{f}) \cancel{p(\mathbf{f} \mid \mathbf{u})} p(\mathbf{u})}{\cancel{p(\mathbf{f} \mid \mathbf{u})} q(\mathbf{u})} \, \mathrm{d}\mathbf{f}\mathrm{d}\mathbf{u} \\ &= \int q(\mathbf{u}) \log \frac{F(\mathbf{y}, \mathbf{u}) p(\mathbf{u})}{q(\mathbf{u})} \mathrm{d}\mathbf{u}, \end{aligned} \tag{2.27}$$

where we have defined

$$F(\mathbf{y}, \mathbf{u}) \triangleq \exp\left(\int p(\mathbf{f} \mid \mathbf{u}) \log p(\mathbf{y} \mid \mathbf{f}) \, \mathrm{d}\mathbf{f} \right). \tag{2.28}$$

We can re-arrange the ELBO of Equation (2.27) into the usual composition made up of ELL and KL divergence terms,

$$\text{ELBO}(q) = \int q(\mathbf{u}) \log F(\mathbf{y}, \mathbf{u}) \mathrm{d}\mathbf{u} - \text{KL}\left[q(\mathbf{u}) \parallel p(\mathbf{u}) \right]. \tag{2.29}$$

Interestingly, $\log\left[F(\mathbf{y}, \mathbf{u})\right]$ is a lower bound on the log conditional probability $\log p(\mathbf{y} \mid \mathbf{u})$ – quite simply, by Jensen's inequality, we have

$$\begin{aligned}\log p(\mathbf{y} \mid \mathbf{u}) &= \log \mathbb{E}_{p(\mathbf{f} \mid \mathbf{u})}[p(\mathbf{y} \mid \mathbf{f})] \\ &\geq \mathbb{E}_{p(\mathbf{f} \mid \mathbf{u})}[\log p(\mathbf{y} \mid \mathbf{f})] = \log F(\mathbf{y}, \mathbf{u}).\end{aligned}$$

Refer to the manuscript of Hensman, Matthews, and Ghahramani [99, Equation 1] for a discussion of the role that this "intermediate" lower bound plays in various contexts.

It is worth mentioning that there are some nuanced technical concerns over whether maximising the ELBO in Equation (2.29) truly minimises the KL divergence between the prior and posterior stochastic processes. We shall not delve further into this issue here except to note that these were largely resolved by Matthews et al. [165].

OPTIMAL VARIATIONAL DISTRIBUTION. From the ELBO as expressed in Equation (2.27), it's evident that the maximising variational distribution takes the form $q^*(\mathbf{u}) \propto F(\mathbf{y}, \mathbf{u}) p(\mathbf{u})$,

$$q^*(\mathbf{u}) = \frac{F(\mathbf{y}, \mathbf{u}) p(\mathbf{u})}{\int F(\mathbf{y}, \mathbf{u}) p(\mathbf{u}) \, \mathrm{d}\mathbf{u}}. \tag{2.30}$$

This can also be verified through the use of calculus of variations, as shown in Section 2.B, or by applying Jensen's inequality, but in the opposite direction.

COLLAPSED LOWER BOUND. If we now substitute q^* back into the ELBO, we get the so-called *collapsed* lower bound,

$$\text{ELBO}(q^*) = \log\left(\int p(\mathbf{u}) F(\mathbf{y}, \mathbf{u}) \, \mathrm{d}\mathbf{u} \right). \tag{2.31}$$

This bound is "collapsed" in the sense that it is no longer a function of q (it is already optimal wrt q) but implicitly remains a function of other (hyper)parameters such as the kernel parameters $\boldsymbol{\theta}$ and the inducing input locations $\mathbf{Z}$.

2.4.2.1 *General Likelihoods*

black-box likelihood

When we make no assumptions about the explicit form of the likelihood $p(\mathbf{y} \mid \mathbf{f})$ nor of its structure or behaviour, it is characterised as "black-box". The integral that constitutes the ELL term in Equation (2.29) is generally intractable for black-box likelihoods. However, if we marginalise out $\mathbf{u}$ to rewrite the ELL as

$$\begin{aligned}\int q(\mathbf{u}) \log F(\mathbf{y}, \mathbf{u}) \, \mathrm{d}\mathbf{u} &= \int \left(\int q(\mathbf{u}) p(\mathbf{f} \mid \mathbf{u}) \, \mathrm{d}\mathbf{u} \right) \log p(\mathbf{y} \mid \mathbf{f}) \, \mathrm{d}\mathbf{f} \\ &= \int q(\mathbf{f}) \log p(\mathbf{y} \mid \mathbf{f}) \, \mathrm{d}\mathbf{f},\end{aligned}$$

we can approximate it efficiently using numerical integration methods such as Monte Carlo (MC) estimation or quadrature rules, by virtue of the fact that the marginal $q(\mathbf{f})$ is available in the analytical form of Equation (2.24) and can thus be sampled easily,

$$\int q(\mathbf{f}) \log p(\mathbf{y} \mid \mathbf{f}) \, \mathrm{d}\mathbf{f} \approx \frac{1}{S} \sum_{s=1}^{S} \log p(\mathbf{y} \mid \mathbf{f}^{(s)}), \quad \mathbf{f}^{(s)} \sim q(\mathbf{f})$$

Moreover, because $q(\mathbf{f})$ is Gaussian, we can utilise simple and effective rules like Gauss-Hermite quadrature, described further in Appendix A for a different application.

2.4.2.2 *Factorised Likelihoods (for Scalability)*

Further, suppose the likelihood factorises, i. e., the observations depend point-wise on the latent functions,

$$p(\mathbf{y} \mid \mathbf{f}) = \prod_{n=1}^{N} p(y_n \mid f_n),$$

we then have

$$\int q(\mathbf{f}) \log p(\mathbf{y} \mid \mathbf{f}) \mathrm{d}\mathbf{f} = \sum_{n=1}^{N} \int q(f_n) \log p(y_n \mid f_n) \, \mathrm{d}f_n.$$

Therefore, the ELBO can be written as

$$\text{ELBO}(q) = \sum_{n=1}^{N} \mathbb{E}_{q(f_n)}[\log p(y_n \mid f_n)] - \text{KL}\left[q(\mathbf{u}) \parallel p(\mathbf{u})\right].$$

Importantly, it's clear that this objective is amenable to mini-batch training for stochastic optimisation [98].

2.4.2.3 *Gaussian Likelihood (for Regression)*

Now suppose the problem at hand is regression, for which the likelihood of choice is typically the Gaussian from Equation (2.15). We can show that

$$F(\mathbf{y}, \mathbf{u}) = \mathcal{N}(\mathbf{y} \mid \mathbf{Q}_{\mathbf{fu}}\mathbf{u}, \beta^{-1}\mathbf{I}) \times \exp\left(-\frac{\beta}{2} \operatorname{tr}(\mathbf{K}_{\mathbf{ff}} - \mathbf{Q}_{\mathbf{ff}})\right). \tag{2.32}$$

Refer to Section 2.C for detailed derivations. This framework, first studied in the landmark paper by Titsias [262], is often referred to as sparse GP regression (SGPR).

OPTIMAL VARIATIONAL DISTRIBUTION. Since the likelihood is Gaussian, by Equation (2.32), the maximiser of the ELBO in Equation (2.30) is the product of two exponentiated-quadratic functions of $\mathbf{u}$. When normalised, this becomes

$$q^*(\mathbf{u}) = \mathcal{N}(\mathbf{u} \mid \beta \mathbf{K_{uu}} \mathbf{M}^{-1} \mathbf{K_{uf}} \mathbf{y}, \mathbf{K_{uu}} \mathbf{M}^{-1} \mathbf{K_{uu}}), \tag{2.33}$$

where $\mathbf{M} \triangleq \mathbf{K_{uu}} + \beta \mathbf{K_{fu}} \mathbf{K_{uf}}$. Refer to Section 2.D for details.

COLLAPSED LOWER BOUND. The optimal lower bound wrt q from Equation (2.31) now becomes

$$\text{ELBO}(q^*) = \log \mathcal{N}(\mathbf{y} \mid \mathbf{0}, \mathbf{Q_{ff}} + \beta^{-1}\mathbf{I}) - \frac{\beta}{2} \operatorname{tr}(\mathbf{K_{ff}} - \mathbf{Q_{ff}}).$$

Refer to Section 2.E for further details. It's instructive at this point to compare this with the log marginal likelihood $\log p(\mathbf{y})$ of the exact GP regression setting from Equation (2.20). We readily see that $\log p(\mathbf{y}) = \text{ELBO}(q^*)$ when $\mathbf{Q_{ff}} = \mathbf{K_{ff}}$. Furthermore, evaluating $\log p(\mathbf{y})$ has a computational complexity of $\mathcal{O}(N^3)$, whereas calculating the ELBO has a complexity of $\mathcal{O}(NM^2 + M^3)$.

TEST PREDICTIVE DISTRIBUTION. Finally, we can obtain the posterior predictive density at test inputs $\mathbf{X}_*$ by substituting the mean and covariance from Equation (2.33) into $\mathbf{m_u}$ and $\mathbf{C_u}$ from Equation (2.24),

$$q(\mathbf{f}_*) = \mathcal{N}\left(\mathbf{f}_* \mid \beta \mathbf{K}_{*\mathbf{u}} \mathbf{M}^{-1} \mathbf{K_{uf}} \mathbf{y}, \mathbf{K}_{**} - \mathbf{K}_{*\mathbf{u}} (\mathbf{K_{uu}}^{-1} - \mathbf{M}^{-1}) \mathbf{K}_{\mathbf{u}*}\right). \tag{2.34}$$

All told, we see that SGPR has time complexity $\mathcal{O}(M^3)$ at prediction time and $\mathcal{O}(NM^2 + M^3)$ during training, with a space complexity of $\mathcal{O}(NM + M^2)$, which offers a substantial speedup over exact inference when $M \ll N$.

Finding a comprehensive, self-contained resource that provides derivations of the equations summarised in this section is surprisingly difficult. Consequently, the pieces necessary to construct the contents of this section and its derivations are collected variously from the unpublished technical report of Titsias [263], the technical notes of Bui and Turner, the paper of Hensman, Matthews, and Ghahramani [99], as well as the PhD theses of Bui [23], Matthews [168], and Van der Wilk [272].

For the newcomer to Bayesian statistics, it is instructive to derive these independently, as they invoke nearly all the essential tools of the trade, such as identities relating to conditioning, marginalisation, and affine transformations of Gaussians, the Woodbury matrix identity,

Jensen's inequality, calculus of variations, "completing the square", including less standard ones such as the "inner-product as outer-product-trace" identity. We reiterate only a few of these here, as most can be found in the well-known texts of Deisenroth, Faisal, and Ong [56, p. MML], Murphy [181, MLaPP], Bishop [16, PRML], and Williams and Rasmussen [286, GPML].

Deriving the quantities from this section not only provides an ideal exercise regimen for hands-on practice with these vital tools, taking the journey from exact GP regression to SGPR, SVGP, and its stochastic variant offers a prime example of a model family that effectively spans the spectrum of exactness and approximation often present in Bayesian modelling and VI that we previously alluded to in Section 2.2. This progression leads us from an exact posterior to an closed-form optimal variational posterior, followed by a variational posterior optimised wrt an exact deterministic ELBO and, ultimately, to one optimised wrt a stochastic ELBO.

In this section, we have examined sparse GPs through the lens of VI. This framework, which we and others have referred to as SVGP, also goes by the name of the variational free energy (VFE) framework, owing to the ELBO's interpretation from the perspective of statistical thermodynamics. The framework known as *stochastic* variational GP [98] shares the same acronym as SVGP, but specifically pertains to the scalable mini-batch variant of the VFE framework. Other prominent sparse GP methods, such as the deterministic training conditional (DTC) [226] and the fully independent training conditional (FITC) [234], are beyond the scope of this thesis. Nonetheless, the topic of their connection to VFE is fascinating, and we direct the interested reader to the manuscript by Bui, Yan, and Turner [22] and the thesis of Bui [23] for a unifying framework under the umbrella of EP. For further insights and practical implications of their connections, we recommend the manuscript by Bauer, Wilk, and Rasmussen [9] and the thesis of Van der Wilk [272].

2.4.3 *Random Fourier Features*

In the previous section, we examined a kind of GP approximation that effectively approximates the GP *posterior* predictive density. Now let's examine a different approximation – one that approximates the covariance function itself, and, therefore, the *prior*.

Bayesian linear model

Consider the Bayesian linear regression (BLR) model with weights $\mathbf{w} \in \mathbb{R}^L$,

$$f(\mathbf{x}) = \sum_{i=1}^{L} w_i \phi_i(\mathbf{x}) = \boldsymbol{\phi}^\top(\mathbf{x})\mathbf{w}, \tag{2.35}$$

basis functions

for some set of L *basis functions*, or, *features*, $\boldsymbol{\phi}(\mathbf{x}) = [\phi_1(\mathbf{x}) \dots \phi_L(\mathbf{x})]^\top \in \mathbb{R}^L$. As before, in Equation (2.15), the observed targets y are assumed

to be function values corrupted by additive noise ε, which are further assumed to be iid Gaussian with zero mean and precision $\beta > 0$,

$$y = f(\mathbf{x}) + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \beta^{-1}). \tag{2.36}$$

This implies the likelihood $p(\mathbf{y} \mid \mathbf{w}) = \mathcal{N}(\mathbf{y} \mid \mathbf{\Phi}\mathbf{w}, \beta^{-1}\mathbf{I})$, where $\mathbf{\Phi} \triangleq \boldsymbol{\phi}(\mathbf{X}) \in \mathbb{R}^{N \times L}$. Suppose we have a Gaussian prior over the weights $p(\mathbf{w}) = \mathcal{N}(\mathbf{w} \mid \mathbf{0}, \mathbf{\Sigma}_\mathbf{w})$. When f is evaluated at a finite collection of T locations $\mathbf{X}_*$, the vector $\mathbf{f}_* = f(\mathbf{X}_*) \in \mathbb{R}^T$ follows the Gaussian distribution $\mathcal{N}(\mathbf{f}_* \mid \mathbf{0}, \mathbf{\Phi}_* \mathbf{\Sigma}_\mathbf{w} \mathbf{\Phi}_*^\top)$ where $\mathbf{\Phi}_* \triangleq \boldsymbol{\phi}(\mathbf{X}_*) \in \mathbb{R}^{T \times L}$. In other words f is by definition a GP with the covariance function $k(\mathbf{x}, \mathbf{x}') = \boldsymbol{\phi}(\mathbf{x})^\top \mathbf{\Sigma}_\mathbf{w} \boldsymbol{\phi}(\mathbf{x}')$. This is known as the weight-space perspective of GPs [286]. Sampling random functions $f(\cdot)$ from the prior amounts to sampling $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_\mathbf{w})$. Therefore, if $\mathbf{\Sigma}_\mathbf{w}$ is diagonal, as is often the case in practice, $f(\cdot)$ can be sampled cheaply at a cost of $\mathcal{O}(L)$. Additionally, for a given realisation of $\mathbf{w}$, the corresponding sample $f(\mathbf{x})$ is a deterministic function – importantly, one that is differentiable wrt $\mathbf{x}$. Consequently, the weight-space approximation is relied upon in Thompson sampling [258] to address sequential decision-making problems that require balancing exploration and exploitation, as we will discuss further in Section 2.5.2.4.

weight-space approximation

Now, the posterior weight density is

$$p(\mathbf{w} \mid \mathbf{y}) = \mathcal{N}\left(\beta(\mathbf{\Sigma}_\mathbf{w}^{-1} + \beta\mathbf{\Phi}^\top\mathbf{\Phi})^{-1}\mathbf{\Phi}^\top\mathbf{y}, (\mathbf{\Sigma}_\mathbf{w}^{-1} + \beta\mathbf{\Phi}^\top\mathbf{\Phi})^{-1}\right) \tag{2.37}$$

Assuming $\mathbf{\Sigma}_\mathbf{w} = \mathbf{I}$, the covariance function is becomes $k(\mathbf{x}, \mathbf{x}') = \boldsymbol{\phi}(\mathbf{x})^\top \boldsymbol{\phi}(\mathbf{x}')$ and the posterior density simplifies to

$$p(\mathbf{w} \mid \mathbf{y}) = \mathcal{N}\left((\mathbf{\Phi}^\top\mathbf{\Phi} + \beta^{-1}\mathbf{I})^{-1}\mathbf{\Phi}^\top\mathbf{y}, \beta^{-1}(\mathbf{\Phi}^\top\mathbf{\Phi} + \beta^{-1}\mathbf{I})^{-1}\right).$$

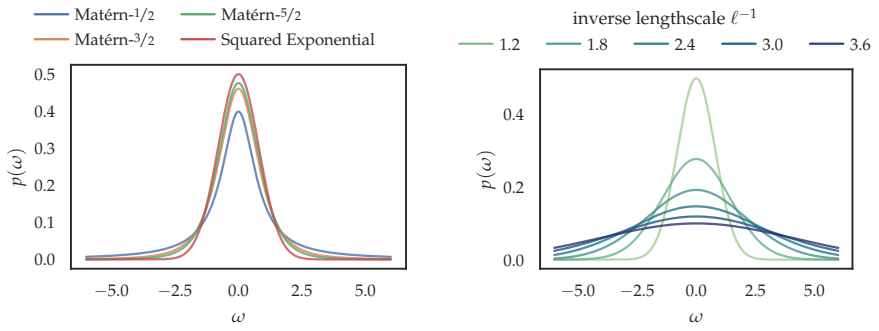
Thus seen, the computational complexity of evaluating this density is dominated by the cost associated with inverting the matrix $\mathbf{\Phi}^\top\mathbf{\Phi} + \beta^{-1}\mathbf{I}$. Through judicious application of the Woodbury matrix identity, this cost is $\mathcal{O}(\min\{L, N\}^3)$.

Now, by the *kernel trick*, a kernel k can be seen as an inner product in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ equipped with a feature map $\varphi : \mathcal{X} \to \mathcal{H}$. For separable $\mathcal{H}$, we can approximate this inner product as

kernel trick

$$k(\mathbf{x}, \mathbf{x}') = \langle \varphi(\mathbf{x}), \varphi(\mathbf{x}') \rangle_\mathcal{H} \approx \boldsymbol{\phi}(\mathbf{x})^\top \boldsymbol{\phi}(\mathbf{x}'), \tag{2.38}$$

for some finite-dimensional feature map $\boldsymbol{\phi} : \mathcal{X} \to \mathbb{R}^L$. In particular, let us focus on the stationary covariance functions, which possess properties that can be leveraged to construct efficient approximations. Extensions beyond stationary covariance functions are possible through the application of Mercer's theorem and the Karhunen–Loève expansion [30, 73]. In Chapter 3, we discuss an example of this in the spherical harmonics for *zonal* covariance functions.



(a) Spectral density of various covariance functions with characteristic lengthscale $\ell = 5/4$.

(b) Spectral density of the squared exponential (SE) covariance function with varying characteristic lengthscales.

Figure 2.8: Spectral densities of the stationary covariance functions from Section 2.4.1.1.

Kernel	$\kappa(\mathbf{t})$	$p(\boldsymbol{\omega})$
SE	$\exp\left(-\frac{1}{2}\mathbf{t}^\top\mathbf{M}^{-1}\mathbf{t}\right)$	$\mathcal{N}\left(\mathbf{0}, \mathbf{M}^{-1}\right)$
Matérn-3/2	$\left(1+\sqrt{3\mathbf{t}^\top\mathbf{M}^{-1}\mathbf{t}}\right)\exp\left(-\sqrt{3\mathbf{t}^\top\mathbf{M}^{-1}\mathbf{t}}\right)$	$t_3\left(\mathbf{0}, \mathbf{M}^{-1}\right)$
Matérn-5/2	$\left(1+\sqrt{5\mathbf{t}^\top\mathbf{M}^{-1}\mathbf{t}}+\frac{5}{3}\mathbf{t}^\top\mathbf{M}^{-1}\mathbf{t}\right)\exp\left(-\sqrt{5\mathbf{t}^\top\mathbf{M}^{-1}\mathbf{t}}\right)$	$t_5\left(\mathbf{0}, \mathbf{M}^{-1}\right)$

Table 2.1: Fourier transform pairs of stationary covariance function $\kappa(\mathbf{t})$ and their spectral density $p(\boldsymbol{\omega})$, with $\mathbf{t} \triangleq \mathbf{x} - \mathbf{x}'$ and $\mathbf{M} \triangleq \operatorname{diag}(\ell_1^2, \ldots, \ell_D^2)$.

Theorem 2.4.1 (Bochner's theorem)**.** *A continuous, translation invariant kernel $k(\mathbf{x}, \mathbf{x}') = \kappa(\mathbf{x} - \mathbf{x}')$ is positive definite if and only if it is the Fourier transform of a nonnegative, finite measure μ,*

$$\kappa(\mathbf{x} - \mathbf{x}') = \int e^{-i\boldsymbol{\omega}^\top(\mathbf{x}-\mathbf{x}')}\,\mathrm{d}\mu(\boldsymbol{\omega}).$$

spectral density

If measure μ has a density $p(\boldsymbol{\omega})$, it is referred to as the *spectral density*, or, *power spectrum*, associated with kernel k. We have the following Fourier transform pair,

$$\kappa(\mathbf{t}) = \int p(\boldsymbol{\omega}) e^{-i\boldsymbol{\omega}^\top\mathbf{t}}\,\mathrm{d}\boldsymbol{\omega}, \qquad \text{and} \qquad p(\boldsymbol{\omega}) = \frac{1}{2\pi}\int \kappa(\mathbf{t}) e^{i\boldsymbol{\omega}^\top\mathbf{t}}\,\mathrm{d}\mathbf{t}. \tag{2.39}$$

For example, for the 1D SE kernel from Equation (2.17), we can calculate its corresponding spectral density using Equation (2.39) to obtain

$$p(\omega) = \mathcal{N}\left(\omega \,|\, 0, \ell^{-2}\right). \tag{2.40}$$

Refer to Section 2.F for details. More generally, for the D-dimensional SE kernel from Equation (2.18), we have

$$p(\boldsymbol{\omega}) = \mathcal{N}\left(\mathbf{0}, \mathbf{M}^{-1}\right),$$

and, for the Matérn-ν kernel from Equation (2.19), we have

$$p(\boldsymbol{\omega}) = t_{2\nu}\left(\mathbf{0}, \mathbf{M}^{-1}\right),$$

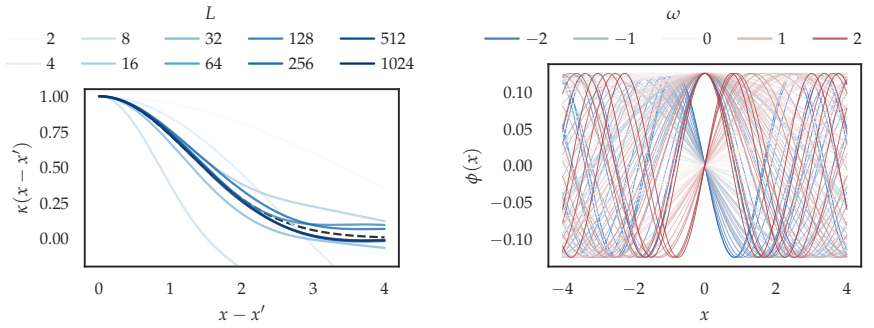
where $t_{2\nu}$ denotes the Student's t-distribution with 2ν degrees of freedom. See Table 2.1 for a summary of popular stationary kernels and their spectral densities. Now, assuming $p(\boldsymbol{\omega})$ is *even symmetric*, $\kappa(\mathbf{t})$ from Equation (2.39) is real-valued and simplifies further to the Fourier *cosine* transform,

$$\begin{aligned}\kappa(\mathbf{x} - \mathbf{x}') &= \int p(\boldsymbol{\omega}) \cos\left(\boldsymbol{\omega}^\top(\mathbf{x} - \mathbf{x}')\right) \mathrm{d}\boldsymbol{\omega} \\ &= \mathbb{E}_{p(\boldsymbol{\omega})}[\cos\left(\boldsymbol{\omega}^\top(\mathbf{x} - \mathbf{x}')\right)]. \end{aligned} \tag{2.41}$$

Let $\psi_{\boldsymbol{\omega}} : \mathbb{R}^D \to \mathbb{R}^2$ denote the projection in some random direction $\boldsymbol{\omega} \sim p(\boldsymbol{\omega})$, mapped to the unit circle,

$$\psi_{\boldsymbol{\omega}}(\mathbf{x}) \triangleq \begin{bmatrix}\cos \boldsymbol{\omega}^\top \mathbf{x} \\ \sin \boldsymbol{\omega}^\top \mathbf{x}\end{bmatrix}. \tag{2.42}$$

Using elementary trigonometric identities, we can show that the inner



(a) Covariance function approximations $\boldsymbol{\phi}(\mathbf{x})^\top \boldsymbol{\phi}(\mathbf{x}') \approx k(\mathbf{x}, \mathbf{x}')$ for $L = 2^i$ and increasing values of $i = 1, \ldots, 10$.

(b) An example realisation of the basis functions $\boldsymbol{\phi} : \mathcal{X} \to \mathbb{R}^L$ for $L = 2^6$.

Figure 2.9: A example random Fourier features (RFF) decomposition of the SE covariance function with characteristic lengthscale $\ell = 5/4$. The exact values of the covariance function are indicated by the dashed black line.

product of $\psi_{\boldsymbol{\omega}}$ evaluated at inputs $\mathbf{x}$ and $\mathbf{x}'$ is

$$\psi_{\boldsymbol{\omega}}(\mathbf{x})^\top \psi_{\boldsymbol{\omega}}(\mathbf{x}') = \cos\left(\boldsymbol{\omega}^\top(\mathbf{x} - \mathbf{x}')\right). \tag{2.43}$$

Refer to Section 2.G for details. Finally, by Equation (2.41), we recover the kernel k by taking the expectation of Equation (2.43) on both sides,

$$\begin{aligned}\mathbb{E}_{p(\boldsymbol{\omega})}[\psi_{\boldsymbol{\omega}}(\mathbf{x})^\top \psi_{\boldsymbol{\omega}}(\mathbf{x}')] &= \mathbb{E}_{p(\boldsymbol{\omega})}[\cos\left(\boldsymbol{\omega}^\top(\mathbf{x} - \mathbf{x}')\right)] \\ &= k(\mathbf{x}, \mathbf{x}'). \end{aligned} \tag{2.44}$$

This shows that the inner product of Equation (2.43) is an unbiased estimator of $k(\mathbf{x}, \mathbf{x}')$. In other words, evaluating the kernel amounts to computing the expectation in the LHS of Equation (2.44). Hence, in order to approximate the kernel, we can leverage techniques of numerical integration [51] to construct a set of basis functions, or, features, $\boldsymbol{\phi} : \mathcal{X} \to \mathbb{R}^L$, such that

$$\boldsymbol{\phi}(\mathbf{x})^\top \boldsymbol{\phi}(\mathbf{x}') = \sum_{i=1}^{L} \phi_i(\mathbf{x})^\top \phi_i(\mathbf{x}') \approx \mathbb{E}_{p(\boldsymbol{\omega})}[\psi_{\boldsymbol{\omega}}(\mathbf{x})^\top \psi_{\boldsymbol{\omega}}(\mathbf{x}')] = k(\mathbf{x}, \mathbf{x}').$$

Fourier feature decomposition

We refer to this as the *Fourier feature decomposition*. Perhaps the most well-known example of this is the (award-winning) random Fourier features (RFF) decomposition of Rahimi and Recht [206], in which $\phi_i : \mathbf{x} \mapsto \sqrt{2/L}\cos(\boldsymbol{\omega}^{(i)} \cdot \mathbf{x} + b^{(i)})$ for $\boldsymbol{\omega}^{(i)} \sim p(\boldsymbol{\omega})$ and $b^{(i)} \sim \mathcal{U}[0, 2\pi]$. This feature decomposition is based on the relatively straightforward application of MC estimation in combination with a few trigonometric identities.

In Appendix A, we provide a detailed derivation of the random Fourier features (RFF) decomposition, in addition to alternative feature decompositions based on various numerical integration schemes. For further details on the weight-space approximation and generalisations beyond stationary covariance functions, the interested reader may refer to the manuscript of Wilson et al. [291] upon which our treatment of this topic is based.

2.5 BAYESIAN OPTIMISATION

Bayesian optimisation (BO) is a powerful framework for efficiently locating the global optima of expensive black-box functions [20, 75, 228]. It can be seen as a sequential algorithm for decision-making amidst the uncertainties inherent in the problem of global optimisation.

global optimisation

Formally, for a real-valued blackbox function $f : \mathcal{X} \to \mathbb{R}$, the goal of global optimisation is to locate an input $\mathbf{x} \in \mathcal{X}$ at which it is minimised,

$$\mathbf{x}^* = \arg\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}).$$

Throughout our presentation, we shall focus on the *minimisation* problem without loss of generality, as any maximisation problem can be translated into a minimisation problem, and vice versa, simply by negating the function of interest. In contrast with classical mathematical optimisation, which frequently rely upon a number of simplifying assumptions, BO is particularly well-equipped to address problems with the following general properties:

OPAQUE. The functions are largely inscrutable, lacking a well-defined functional form or useful closed-form expression (hence, characterised as "black boxes"). Additionally, these functions do not

Algorithm 1: A generic sequential decision-making procedure for optimisation.

Input: blackbox function $f : \mathcal{X} \to \mathbb{R}$, initial dataset $\mathcal{D}_0$.
repeat
  $\mathbf{x}_N \leftarrow \textsc{policy}(\mathcal{D}_{N-1})$ // suggest next candidate location
  $y_N \leftarrow \textsc{evaluate}(\mathbf{x}_N)$ // evaluate f at the suggested location
  $\mathcal{D}_N \leftarrow \mathcal{D}_{N-1} \cup \{(\mathbf{x}_N, y_N)\}$ // update dataset
  $N \leftarrow N + 1$
until *termination condition satisfied*

provide helpful "hints" or "clues" typically exploited by traditional optimisation methods, such as first-order gradients, let alone higher-order derivatives. Lastly, the function is assumed to be nonconvex, which is to say that a local optimum is not automatically considered a globally optimal solution.

EXPENSIVE. The functions are assumed to be costly to evaluate. Since evaluations require substantial resources like time and money, the function cannot be trivially optimised by exhaustive evaluation.

IMPRECISE. The mechanism by which the function is evaluated is assumed to be imperfect, involving randomness, low-fidelity simulation, or indirect observations through noisy measurements.

Simply stated, BO only requires a way to obtain noisy observations of an objective function at suggested locations. It should go without saying that these characteristics are not *preconditions* for BO, but rather represent the complex problem scenarios where BO demonstrates its strength and versatility.

sequential decision-making

Every optimisation procedure boils down to making a series of decisions. In each iteration, we are tasked with deciding which candidate location is the most promising to evaluate next. These decisions must be made in the face of uncertainty, as we cannot know the outcome of an evaluation beforehand, even with access to past observations. Further, the sequential nature of the optimisation process exacerbates the impact of this uncertainty. Any sound optimisation framework must be equipped manage this uncertainty. In light of these considerations, it is helpful to approach BO from the perspective of *Bayesian decision theory* [12, 54], which views it as a principled framework that provides a systematic approach to decision-making under uncertainty tailored for global optimisation. Thus, our remaining treatment of this topic will follow the decision-theoretic introduction provided by Garnett [75].

Bayesian decision theory

The procedure in Algorithm 1 formalises a generic approach to global optimisation. The procedure is initialised with a dataset $\mathcal{D}_0$, which typically consists of a small handful of existing observations

made at randomly-selected locations. For notational simplicity, suppose $\mathcal{D}_0 = \emptyset$. Then, in iteration N, the dataset consists of past observations $\mathcal{D}_N = \{\mathbf{x}_n, y_n\}_{n=1}^N$, where $y_n = f(\mathbf{x}_n) + \varepsilon$ for some additive noise ε. In other words, output y_n is the (inexact) function value at input $\mathbf{x}_n$, assumed to be corrupted by some noise, typically Gaussian distributed $\varepsilon \sim \mathcal{N}(0, \beta^{-1})$, with some precision $\beta > 0$. This effectively leads to the observation model previously introduced in Equations (2.15) and (2.36).

state
optimisation policy
action
outcome

Now, the observed dataset $\mathcal{D}_N$ (which can be viewed as the *state*) is mapped, through an *optimisation policy*, to the candidate location $\mathbf{x}$ to be evaluated next (which can be viewed as an *action*). This location $\mathbf{x}$ is in turn mapped, through evaluation of the blackbox function, to a corresponding value y (which can be viewed as the *outcome*). Finally, the state is updated by appending the new observation $(\mathbf{x}, y)$ to the dataset, and the process is repeated until the termination criteria are met.

The optimisation policy varies along two principal axes. They are either: (1) *deterministic* or *stochastic*, and (2) *adaptive* or *non-adaptive*. Non-adaptive policies disregard the data, exemplified by methods such as grid search and random search [13], which are in turn representative of deterministic and stochastic policies, respectively. On the other hand, BO methods are driven by adaptive optimisation policies that leverage past data to make informed future decisions.

surrogate model
utility function
acquisition function

Accordingly, a hallmark of BO methods is that they maintain a probabilistic model known as the *surrogate model*, which encapsulates our knowledge and beliefs about the unknown function. These beliefs are continuously updated as new data is acquired, allowing the algorithm to adapt its behaviour to make optimal decisions based on the evolving information. In addition to the surrogate model, often a *utility function* $U(y)$ is specified to encode our preferences for the kinds of observations that are considered useful. These preferences are connected to the posterior beliefs, through the surrogate model's posterior predictive density $p(y \mid \mathbf{x}, \mathcal{D}_N)$, to form the *acquisition function* $\alpha(\mathbf{x}; \mathcal{D}_N)$, which serves as a criterion or score for candidate locations, indicating the benefit they bring to the optimisation procedure. Ultimately, the optimisation policy produces the maximiser of the acquisition function,

$$\textsc{policy} : \mathcal{D}_N \mapsto \arg\max_{\mathbf{x} \in \mathcal{X}} \alpha(\mathbf{x}; \mathcal{D}_N).$$

The reason that this approach works at all (namely, optimising a function by optimising yet another function) is that the acquisition function is designed to be more manageable than the unknown function $f(\mathbf{x})$. Specifically, the acquisition function is usually relatively inexpensive to evaluate, possesses closed-form expressions, and offers analytically tractable gradients. As a result, they can be optimised efficiently using conventional, readily available mathematical optimisation methods.

All acquisition functions try to negotiate between the opposing forces of exploration and exploitation. In the context of optimisation, exploitation is the tendency to favour locations where the function value is expected to be low (assuming the goal is minimisation), while exploration is the tendency to favour locations where there is a high degree of uncertainty concerning the function value, enabling the acquisition of more data to improve the model and make more informed decisions in the future. The key to an effective optimisation approach lies in striking a balance within the acquisition function, ensuring that neither force overpowers the other.

In the remainder of this section, we provide an overview of the key components we have introduced, namely, the surrogate model and acquisition function. In particular, we examine the main considerations for their design and discuss several proven approaches.

Before moving on, a quick word on notation: throughout the earlier chapters we have used $p(f_* \mid \mathbf{y})$ to denote the posterior predictive density. This is itself a shorthand for $p(f_* \mid \mathbf{x}_*, \mathbf{X}, \mathbf{y})$, which, considering that $\mathcal{D}_N$ is another way to denote $(\mathbf{X}, \mathbf{y})$, is not too dissimilar to the $p(y \mid \mathbf{x}, \mathcal{D}_N)$ notation used here. In the present context, the asterisks are no longer required for the purpose of distinguishing unseen test points as the observations are instead disambiguated by indexed subscripts (i. e., $\mathbf{x}_n, y_n$).

2.5.1 *Surrogate Models*

From the high-level description of BO we have presented above, it shouldn't be difficult to appreciate the importance of having a consistent framework for systematically reasoning about unknown functions. Therefore, it is not surprising that GPs have emerged as the predominant model family in BO. Indeed, BO is often regarded as the "killer application" for GPs.

GPs possess several compelling characteristics that make them an ideal choice as surrogate models for BO. First and foremost, GP models offer reliable and well-calibrated predictive uncertainty estimates, which has proven to be of crucial importance in practice [228]. Second, to be specific, the GP regression model with Gaussian noise (i. e., the "textbook" version described in Section 2.4) stands out as a rare example of a highly-flexible model that retains its analytical tractability. Notably, both the posterior predictive density and the marginal likelihood can be computed analytically – see Equations (2.16) and (2.20). This tractability is crucial, as eliminating the need for approximate inference implies not having to compromise on the quality and accuracy of uncertainty quantification and hyperparameter estimation for Bayesian model selection. Despite their favorable tractability properties, GPs remain highly expressive, incorporating only a limited set of assumptions related to smoothness, stationarity, and characteristic

lengthscales. These assumptions are generally mild and do not impose significant restrictions in most problems. On the contrary, such assumptions often prove beneficial in many real-world optimisation problems.

With this being said, it's always possible to find counterexamples of problem scenarios in BO for which GPs are ill-suited. When the unknown function is believed to exhibit nonstationary behaviour, augmenting a stationary covariance function by warping the inputs through a nonlinear mapping can create a more expressive nonstationary covariance function. Notable examples of such warping functions include using cdfs that are flexible yet succinctly parameterised [238] or employing deep neural networks (DNNs) [28, 288]. Similarly, when the measurement error is believed to be heteroscedastic, extensions can applied to the observation model [89, 159]. In more complex scenarios involving discrete (ordered and unordered) inputs [77], sequential inputs [178], or structured inputs with conditional dependencies [116], it can be challenging to devise useful covariance functions. It goes without saying that even the most promising approaches introduce a significant footprint to the framework, not least in terms of computational overhead or additional parameters to contend with. Moreover, none of this makes mention of the fact that there is no straightforward workaround for the more fundamental limitation of exact GP regression, which has a computational cost that scales cubically with the number of observations. This limitation precludes running BO for extended horizons on problems that require numerous evaluations to reach a global optimum. While the sparse GP approximations described in Section 2.4.2 can be readily applied, it is essential to allocate the inducing points properly [179]. Neglecting this careful allocation often leads to impractical solutions with degraded performance due to poorly calibrated uncertainty estimates [228].

nonstationarity
input warping

heteroscedasticity

If resorting to approximations becomes inevitable, it stands to reason that leveraging alternative estimators that are explicitly designed to address these specific problem scenarios could potentially provide greater advantages. For example, when dealing with functions involving discrete or structured inputs or high-dimensionalities, ensembles of decision tree regressors such as extreme gradient-boosting (XGBOOST) [34] and random forests (RFS) [19] offer attractive alternatives. In particular, RFS underpin the popular sequential model-based algorithm configuration (SMAC) method [111]. In a similar vein, the tree-structured Parzen estimator (TPE) method [14], on which we expand further in Chapter 5, has also enjoyed considerable success. These approaches can handle complex input structures and have proven effective in various applications, particularly in hyperparameter optimisation (HPO) for automated machine learning (AUTOML).

Similarly, for modelling nonstationarity, capturing nonlinear behaviour, or handling multi-output functions in settings like multi-

task [255], multi-fidelity [124], or multi-objective [101] optimisation, BNNs provide an attractive choice [200, 237, 243, 281]. The prominent approaches are Bayesian to varying extents. For instance, Snoek et al. [237] consider a Bayesian treatment of only the final layer of weights in a posthoc manner, effectively leading to the BLR model described in Section 2.4.3 with neural network (NN) basis functions. In contrast, Springenberg et al. [243] adopt a more thoroughly Bayesian approach that encompasses all the NN weights, and utilise sampling-based inference, specifically, stochastic gradient Hamiltonian Monte Carlo (SGHMC) [33], to approximate the posterior predictive density. Recent efforts to enhance the performance of BNNs in BO have focused on leveraging the latest advancements in Bayesian deep learning [131, 145].

Thus seen, ensuring tractability of the posterior predictive density often necessitates making compromises in the form of simplifications and crude approximations. Unfortunately, these compromises can often inhibit the expressive power and the range of benefits offered by these alternative surrogate model families. Consequently, there is no model family that can perfectly address all problem scenarios and provide an ideal solution without incurring some trade-offs.

In Chapter 5, we explore an alternative paradigm for BO that circumvents the need for an explicit model of the unknown function, instead focusing on directly approximating the acquisition function. This reframing effectively sidesteps the tractability requirements and opens the door to powerful model families that would otherwise render the predictive density unwieldy or simply intractable to compute.

2.5.2 *Acquisition Functions*

Almost without exception, acquisition functions rely on the predictive density to represent posterior beliefs about the unknown function in order to score the potential benefit of a candidate location. In certain cases, this score incorporates a preference for outcomes specified through a *utility function*. This thesis is primarily concerned with acquisition functions of this nature, so-called the *improvement-based* acquisition functions, such as the well-established probability of improvement (PI) [117] and expected improvement (EI) [176]. Despite the emergence of numerous new and sophisticated acquisition functions like knowledge gradient (KG) [225], entropy search (ES) [96], predictive ES (PES) [102], and their variants [279], the improvement-based acquisition functions remain widely used. Such functions can generally be expressed as an *expectation* of the *utility* function,

improvement-based acquisition function

expected utility

$$\alpha(\mathbf{x}; \mathcal{D}_N, \tau) \triangleq \mathbb{E}_{p(y \mid \mathbf{x}, \mathcal{D}_N)}[U(y; \tau)], \tag{2.45}$$

where τ denotes a parameter representing some threshold and $U(y; \tau)$ denotes a utility function that typically depends on the difference be-

tween τ and y (i. e., the "improvement"). By convention, τ is set to the *incumbent*, the lowest function value observed so far, $\tau = \min_n y_n$ [290].

2.5.2.1 *Probability of Improvement*

In the classical PI acquisition function [117], the utility function simply indicates whether y improves upon some threshold τ,

$$U_{\text{PI}}(y, \tau) \triangleq \mathbb{I}(\tau - y > 0). \tag{2.46}$$

Suppose the posterior predictive density takes the form of a Gaussian

$$p(y \mid \mathbf{x}, \mathcal{D}_N) = \mathcal{N}(y \mid \mu(\mathbf{x}), \sigma^2(\mathbf{x})). \tag{2.47}$$

Then, Equation (2.45) leads to

$$\alpha_{\text{PI}}(\mathbf{x}; \mathcal{D}_N, \tau) = p(y \leq \tau \mid \mathbf{x}, \mathcal{D}_N) = \Psi(Z_\tau(\mathbf{x})), \tag{2.48}$$

where

$$Z_\tau(\mathbf{x}) \triangleq \frac{\tau - \mu(\mathbf{x})}{\sigma(\mathbf{x})},$$

and Ψ denotes the cdf of the standard normal distribution

2.5.2.2 *Expected Improvement (EI)*

In EI [176], the utility function quantifies the nonnegative amount by which y improves upon threshold τ,

$$U_{\text{EI}}(y, \tau) \triangleq \max(\tau - y, 0). \tag{2.49}$$

This is known as the *improvement* utility function. When the predictive density is the Gaussian from Equation (2.47), the expectation from Equation (2.45) is of the improvement utility function (hence the name), and evaluates to

$$\alpha_{\text{EI}}(\mathbf{x}; \mathcal{D}_N, \tau) = \sigma(\mathbf{x}) \cdot \left[Z_\tau(\mathbf{x}) \cdot \Psi(Z_\tau(\mathbf{x})) + \psi(Z_\tau(\mathbf{x})) \right], \tag{2.50}$$

where ψ denotes the pdf of the standard normal distribution.

In Figure 2.10, we plot the EI/PI criteria as functions of the posterior predictive mean $\mu(\mathbf{x})$ and variance $\sigma^2(\mathbf{x})$. We see that the value to which PI assigns $\mathbf{x}$ depends primarily on whether the predictive mean $\mu(\mathbf{x})$ exceeds the threshold, in this example $\tau = 0$, and less so on the predictive variance $\sigma^2(\mathbf{x})$. Furthermore, particularly when the predictive variance is close to zero, the function is essentially piecewise constant with a discontinuity at τ. In other words, and as can be expected from simply looking at its analytical expression alone, PI either rewards a high or low value depending on whether or not $\mu(\mathbf{x})$ exceeds the threshold, but is indifferent to the amount by which it does. In practice, this can lead to the optimisation procedure getting stuck in local optima and inadequately exploring the search space [75]. In

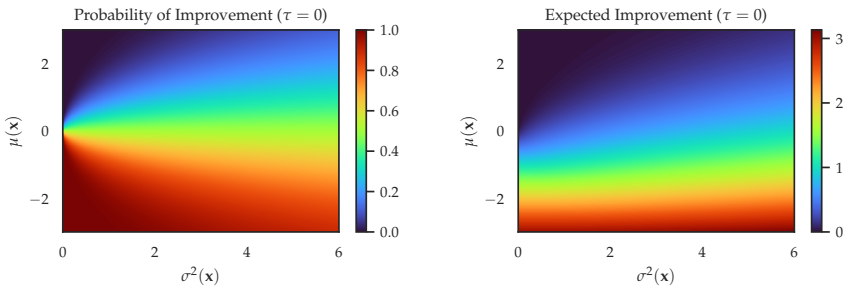


Figure 2.10: Values of improvement-based acquisition functions plotted in terms of the posterior predictive mean $\mu(\mathbf{x})$ and variance $\sigma^2(\mathbf{x})$. PI (*left*) heavily favours exploitation while EI (*right*) strikes a slightly better balance between exploitation and exploration.

contrast, the EI criterion does take into account the amount by which a candidate location in expectation improves upon the threshold. Furthermore, broadly speaking, for any given fixed value of $\mu(\mathbf{x})$, the reward assigned by EI increases as the uncertainty, or, more precisely, the variance $\sigma^2(\mathbf{x})$, increases. Thus seen, EI is less prone than PI to exploit too aggressively to its own detriment.

While the exact expressions of Equations (2.48) and (2.50) are both easy to evaluate and optimise, the conditions necessary to satisfy Equation (2.47) can often come at the expense of flexibility and expressiveness. In Chapter 5, we will consider an altogether different way to express PI/EI themselves.

2.5.2.3 *Upper/Lower Confidence Bound*

The upper confidence bound (UCB) [244] function is another popular criterion. UCB has its roots in the multi-armed bandits literature [135] and come with favorable theoretical properties and provable regret bounds. To maintain consistency with our running context of function *minimisation*, we shall discuss the LCB. Like PI/EI the LCB criterion can also be expressed in terms of the predictive mean and variance μ and σ^2,

$$\alpha_{\text{LCB}}(\mathbf{x}; \mathcal{D}_N, \lambda) \triangleq -\mu(\mathbf{x}) + \sqrt{\lambda} \cdot \sigma(\mathbf{x}),$$

where, similar to τ in the improvement-based criteria, λ is a parameter that controls the tendency to explore. Interestingly, UCB/LCB cannot be expressed in terms of the expected utility from Equation (2.45). UCB/LCB is known as an *optimistic* acquisition function, since, by design, it behaves optimistically in the presence of uncertainty. Indeed, from Figure 2.11, we readily see that it assigns greater value to locations $\mathbf{x}$ where the level of uncertainty, or, more precisely, the predictive variance $\sigma^2(\mathbf{x})$, is high.

optimistic acquisition function

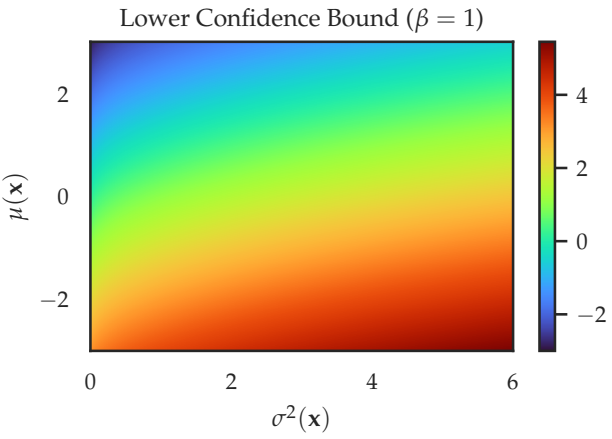


Figure 2.11: Values of the lower confidence bound (LCB) criterion with $\lambda = 1$ plotted in terms of the posterior predictive mean $\mu(\mathbf{x})$ and variance $\sigma^2(\mathbf{x})$. LCB is said to favour exploration, since it behaves optimistically in the face of uncertainty – a higher value is assigned to regions where the variance $\sigma^2(\mathbf{x})$ is large.

2.5.2.4 *Thompson Sampling*

Thompson sampling, a widely-used optimisation policy in BO, was adapted for continuous optimisation from a policy originally proposed for the multi-armed bandit problem almost a century ago [258].

Unlike the acquisition functions discussed earlier, which represented adaptive, *deterministic* policies, Thompson sampling is an adaptive, *stochastic* policy. Like previous acquisition functions, it still depends on the posterior predictive distribution, but does not explicitly involve the predictive mean and variance. Instead, Thompson sampling involves realisations of the unknown objective function randomly sampled from the predictive distribution itself,

$$\alpha_{\mathrm{TS}}(\mathbf{x}; \mathcal{D}_N) \triangleq f(\mathbf{x}), \qquad f \sim p(f_* \mid \mathbf{x}_*, \mathbf{X}, \mathbf{y}).$$

In other words, while the improvement-based polices from Sections 2.5.2.1 and 2.5.2.2 select the best candidate solution *in expectation* by averaging over the objective functions, Thompson sampling determines the best candidate solution according to a *randomly sampled* objective function. Thus seen, this approach balances exploration and exploitation by sampling observations proportional to their probability of optimality, effectively encouraging exploitation, while the stochasticity inherent in random sampling ensures exploration [75].

In practice, sampling random functions from a GP that can be evaluated at arbitrary points, let alone efficiently optimised, poses a considerable challenge. Consequently, a dominant approach adopts the weight-space perspective of GPs, leveraging its spectral decomposition to obtain a posterior weight density. Weights $\mathbf{w}$ can be sampled efficiently from this posterior and used to construct random functions

$f(\mathbf{x}) \triangleq \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x})$ that are (approximately) equal in distribution to GP posterior samples, yet are easy to manipulate and optimise [102, 228, 289, 291]. In Appendix A, we explore the use of various numerical integration methods to further improve the computational efficiency of sampling random functions from GP posteriors.

2.6 SUMMARY

This chapter laid the essential groundwork for our thesis by introducing fundamental concepts in probabilistic modelling, Bayesian statistics, and variational inference. We highlighted the role of statistical divergences and density-ratio estimation in approximate inference, establishing a foundation for advanced topics in probabilistic ML. Our discussion also included Gaussian processes and their sparse approximations based on VI, concluding with the basic concepts behind Bayesian optimisation.

Our discussion of Gaussian processes and variational inference set the stage for our subsequent exploration of orthogonally-decoupled sparse Gaussian processes with spherical neural network activation features. This forms the focus of Chapter 3, representing a unique integration neural networks with Gaussian processes. Similarly, our examination of variational inference, the variational estimation of f-divergences, and density-ratio estimation, laid the groundwork for a new derivation of CYCLEGANS from the perspective of approximate Bayesian inference, which we examine in Chapter 4. Lastly, the basic concepts of density-ratio estimation and Bayesian optimisation introduced here forms the basis for our model-agnostic approach to Bayesian optimisation based on binary classification, which we discuss in Chapter 5.

In summary, this chapter provides the the necessary foundation for the advanced methodologies described in the subsequent chapters, bridging fundamental principles with new perspectives in probabilistic ML.

ADDENDUM

2.A KL DIVERGENCE SIMPLIFICATION

The KL divergence simplifies as follows:

$$\begin{aligned}
&\mathrm{KL}\left[q(\mathbf{f}^*, \mathbf{f}, \mathbf{u}) \parallel p(\mathbf{f}^*, \mathbf{f}, \mathbf{u} \mid \mathbf{y})\right] \\
&\quad = \iiint p(\mathbf{f}^* \mid \mathbf{f}, \mathbf{u}) q(\mathbf{f}, \mathbf{u}) \log \frac{\cancel{p(\mathbf{f}^* \mid \mathbf{f}, \mathbf{u})} q(\mathbf{f}, \mathbf{u})}{\cancel{p(\mathbf{f}^* \mid \mathbf{f}, \mathbf{u})} p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})} \, \mathrm{d}\mathbf{f}^* \mathrm{d}\mathbf{f} \mathrm{d}\mathbf{u} \\
&\quad = \iint q(\mathbf{f}, \mathbf{u}) \log \frac{q(\mathbf{f}, \mathbf{u})}{p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})} \, \mathrm{d}\mathbf{f} \mathrm{d}\mathbf{u} = \mathrm{KL}\left[q(\mathbf{f}, \mathbf{u}) \parallel p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})\right].
\end{aligned}$$

2.B OPTIMAL VARIATIONAL DISTRIBUTION FOR GENERAL LIKELIHOODS

We have

$$\begin{aligned}
\mathrm{ELBO}(q) &= \iint p(\mathbf{f} \mid \mathbf{u}) q(\mathbf{u}) \log p(\mathbf{y} \mid \mathbf{f}) \mathrm{d}\mathbf{f} \mathrm{d}\mathbf{u} + \iint p(\mathbf{f} \mid \mathbf{u}) q(\mathbf{u}) \log \frac{p(\mathbf{u})}{q(\mathbf{u})} \mathrm{d}\mathbf{f} \mathrm{d}\mathbf{u} \\
&= \int q(\mathbf{u}) \left(\int p(\mathbf{f} \mid \mathbf{u}) \log p(\mathbf{y} \mid \mathbf{f}) \mathrm{d}\mathbf{f} \right) \mathrm{d}\mathbf{u} + \int q(\mathbf{u}) \log \frac{p(\mathbf{u})}{q(\mathbf{u})} \mathrm{d}\mathbf{u} \\
&= \int q(\mathbf{u}) \log F(\mathbf{y}, \mathbf{u}) \mathrm{d}\mathbf{u} + \int q(\mathbf{u}) \log \frac{p(\mathbf{u})}{q(\mathbf{u})} \mathrm{d}\mathbf{u} \\
&= \int q(\mathbf{u}) \log \frac{F(\mathbf{y}, \mathbf{u}) p(\mathbf{u})}{q(\mathbf{u})} \mathrm{d}\mathbf{u}.
\end{aligned}$$

Taking the functional derivative of the ELBO wrt to $q(\mathbf{u})$, we get

$$\begin{aligned}
\frac{\partial}{\partial q(\mathbf{u})} \mathrm{ELBO}(q) &= \frac{\partial}{\partial q(\mathbf{u})} \left(\int \log \frac{F(\mathbf{y}, \mathbf{u}) p(\mathbf{u})}{q(\mathbf{u})} q(\mathbf{u}) \, \mathrm{d}\mathbf{u} \right) \\
&= \int \frac{\partial}{\partial q(\mathbf{u})} \left(\log \frac{F(\mathbf{y}, \mathbf{u}) p(\mathbf{u})}{q(\mathbf{u})} q(\mathbf{u}) \right) \mathrm{d}\mathbf{u} \\
&= \int \log \frac{F(\mathbf{y}, \mathbf{u}) p(\mathbf{u})}{q(\mathbf{u})} \left(\frac{\partial}{\partial q(\mathbf{u})} q(\mathbf{u}) \right) + \\
&\qquad q(\mathbf{u}) \left(\frac{\partial}{\partial q(\mathbf{u})} \log \frac{F(\mathbf{y}, \mathbf{u}) p(\mathbf{u})}{q(\mathbf{u})} \right) \mathrm{d}\mathbf{u} \\
&= \int \log \frac{F(\mathbf{y}, \mathbf{u}) p(\mathbf{u})}{q(\mathbf{u})} + q(\mathbf{u}) \left(-\frac{1}{q(\mathbf{u})} \right) \mathrm{d}\mathbf{u} \\
&= \int \log F(\mathbf{y}, \mathbf{u}) + \log p(\mathbf{u}) - \log q(\mathbf{u}) - 1 \, \mathrm{d}\mathbf{u}.
\end{aligned}$$

Setting this expression to zero, we obtain

$$\begin{aligned}
&\log q^*(\mathbf{u}) = \log F(\mathbf{y}, \mathbf{u}) + \log p(\mathbf{u}) - 1 \\
\Rightarrow \qquad &q^*(\mathbf{u}) \propto F(\mathbf{y}, \mathbf{u}) p(\mathbf{u}).
\end{aligned}$$

2.C INTERMEDIATE LOWER BOUND FOR GAUSSIAN LIKELIHOODS

To carry out this derivation, we will need to recall the following two straightforward identities. First, we can express the inner product between two vectors as the trace of their outer product,

$$\mathbf{a}^\top \mathbf{b} = \operatorname{tr}(\mathbf{a}\mathbf{b}^\top).$$

Second, we have the following relationship between the covariance matrix $\operatorname{Cov}[\mathbf{a}]$ and the auto-correlation matrix $\mathbb{E}[\mathbf{a}\mathbf{a}^\top]$,

$$\begin{aligned} & \operatorname{Cov}[\mathbf{a}] = \mathbb{E}[\mathbf{a}\mathbf{a}^\top] - \mathbb{E}[\mathbf{a}]\,\mathbb{E}[\mathbf{a}]^\top \\ \Leftrightarrow \quad & \mathbb{E}[\mathbf{a}\mathbf{a}^\top] = \operatorname{Cov}[\mathbf{a}] + \mathbb{E}[\mathbf{a}]\,\mathbb{E}[\mathbf{a}]^\top \end{aligned}$$

Additionally, let's denote the mean and covariance of the prior conditional $p(\mathbf{f} \mid \mathbf{u})$ in Equation (2.22) as

$$\mathbf{b} \triangleq \mathbf{Q}_{\mathbf{fu}}\mathbf{u}, \qquad \text{and} \qquad \mathbf{S}_{\mathbf{ff}} \triangleq \mathbf{K}_{\mathbf{ff}} - \mathbf{Q}_{\mathbf{ff}},$$

respectively. Together, these allow us to write

$$\begin{aligned} \log F(\mathbf{y}, \mathbf{u}) &= \int \log \mathcal{N}(\mathbf{y}|\mathbf{f}, \beta^{-1}\mathbf{I}) \mathcal{N}(\mathbf{f} \mid \mathbf{b}, \mathbf{S}_{\mathbf{ff}})\, \mathrm{d}\mathbf{f} \\ &= -\frac{\beta}{2} \int (\mathbf{y}-\mathbf{f})^\top (\mathbf{y}-\mathbf{f}) \mathcal{N}(\mathbf{f} \mid \mathbf{b}, \mathbf{S}_{\mathbf{ff}})\, \mathrm{d}\mathbf{f} - \frac{N}{2}\log(2\pi\beta^{-1}) \\ &= -\frac{\beta}{2} \int \operatorname{tr}\left(\mathbf{y}\mathbf{y}^\top - 2\mathbf{y}\mathbf{f}^\top + \mathbf{f}\mathbf{f}^\top\right) \mathcal{N}(\mathbf{f} \mid \mathbf{b}, \mathbf{S}_{\mathbf{ff}})\, \mathrm{d}\mathbf{f} - \frac{N}{2}\log(2\pi\beta^{-1}) \\ &= -\frac{\beta}{2} \operatorname{tr}\left(\mathbf{y}\mathbf{y}^\top - 2\mathbf{y}\mathbf{b}^\top + \mathbf{S}_{\mathbf{ff}} + \mathbf{b}\mathbf{b}^\top\right) - \frac{N}{2}\log(2\pi\beta^{-1}) \\ &= -\frac{\beta}{2}(\mathbf{y}-\mathbf{b})^\top(\mathbf{y}-\mathbf{b}) - \frac{N}{2}\log(2\pi\beta^{-1}) - \frac{\beta}{2}\operatorname{tr}(\mathbf{S}_{\mathbf{ff}}) \\ &= \log \mathcal{N}(\mathbf{y} \mid \mathbf{b}, \beta^{-1}\mathbf{I}) - \frac{\beta}{2}\operatorname{tr}(\mathbf{S}_{\mathbf{ff}}). \end{aligned}$$

Therefore, we have

$$F(\mathbf{y}, \mathbf{u}) = \mathcal{N}(\mathbf{y} \mid \mathbf{b}, \beta^{-1}\mathbf{I}) \times \exp\left(-\frac{\beta}{2}\operatorname{tr}(\mathbf{S}_{\mathbf{ff}})\right). \tag{2.51}$$

as required.

2.D OPTIMAL VARIATIONAL DISTRIBUTION FOR GAUSSIAN LIKELIHOODS

Firstly, the optimal variational distribution can be found in closed-form as

$$\begin{aligned} q^*(\mathbf{u}) &\propto F(\mathbf{y}, \mathbf{u}) p(\mathbf{u}) \\ &\propto \mathcal{N}(\mathbf{y} \mid \mathbf{Q}_{\mathbf{fu}}\mathbf{u}, \beta^{-1}\mathbf{I}) \mathcal{N}(\mathbf{u} \mid \mathbf{0}, \mathbf{K}_{\mathbf{uu}}) \\ &\propto \exp\left(-\frac{\beta}{2}(\mathbf{y}-\mathbf{Q}_{\mathbf{fu}}\mathbf{u})^\top(\mathbf{y}-\mathbf{Q}_{\mathbf{fu}}\mathbf{u}) - \frac{1}{2}\mathbf{u}^\top \mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}\right) \\ &\propto \exp\left(-\frac{1}{2}\left(\mathbf{u}^\top \boldsymbol{\Lambda}\mathbf{u} - 2\beta(\mathbf{Q}_{\mathbf{uf}}\mathbf{y})^\top \mathbf{u}\right)\right), \end{aligned}$$

where

$$\mathbf{\Lambda} \triangleq \mathbf{K}_{\mathbf{uu}}^{-1} + \beta \mathbf{Q}_{\mathbf{uf}} \mathbf{Q}_{\mathbf{fu}} = \mathbf{K}_{\mathbf{uu}}^{-1} (\mathbf{K}_{\mathbf{uu}} + \beta \mathbf{K}_{\mathbf{uf}} \mathbf{K}_{\mathbf{fu}}) \mathbf{K}_{\mathbf{uu}}^{-1}.$$

By completing the square, we get

$$\begin{aligned} q^*(\mathbf{u}) &\propto \exp\left(-\frac{1}{2}(\mathbf{u} - \beta \mathbf{\Lambda}^{-1} \mathbf{Q}_{\mathbf{uf}} \mathbf{y})^\top \mathbf{\Lambda} (\mathbf{u} - \beta \mathbf{\Lambda}^{-1} \mathbf{Q}_{\mathbf{uf}} \mathbf{y})\right) \\ &\propto \mathcal{N}(\mathbf{u} \,|\, \beta \mathbf{\Lambda}^{-1} \mathbf{Q}_{\mathbf{uf}} \mathbf{y}, \mathbf{\Lambda}^{-1}). \end{aligned}$$

If we define

$$\mathbf{M} \triangleq \mathbf{K}_{\mathbf{uu}} + \beta \mathbf{K}_{\mathbf{uf}} \mathbf{K}_{\mathbf{fu}}$$

so that

$$\mathbf{\Lambda} = \mathbf{K}_{\mathbf{uu}}^{-1} \mathbf{M} \mathbf{K}_{\mathbf{uu}}^{-1},$$

we finally get

$$q^*(\mathbf{u}) = \mathcal{N}(\mathbf{u} \,|\, \beta \mathbf{K}_{\mathbf{uu}} \mathbf{M}^{-1} \mathbf{K}_{\mathbf{uf}} \mathbf{y}, \mathbf{K}_{\mathbf{uu}} \mathbf{M}^{-1} \mathbf{K}_{\mathbf{uu}}),$$

as required.

2.E COLLAPSED LOWER BOUND FOR GAUSSIAN LIKELIHOODS

We have

$$\begin{aligned} \text{ELBO}(q^*) &= \log\left(\int p(\mathbf{u}) F(\mathbf{y}, \mathbf{u}) \, \mathrm{d}\mathbf{u}\right) \\ &= \log\left[\exp\left(-\frac{\beta}{2} \operatorname{tr}(\mathbf{S}_{\mathbf{ff}})\right) \int \mathcal{N}(\mathbf{y} \,|\, \mathbf{Q}_{\mathbf{fu}} \mathbf{u}, \beta^{-1} \mathbf{I}) p(\mathbf{u}) \, \mathrm{d}\mathbf{u}\right] \\ &= \log \int \mathcal{N}(\mathbf{y} \,|\, \mathbf{Q}_{\mathbf{fu}} \mathbf{u}, \beta^{-1} \mathbf{I}) \mathcal{N}(\mathbf{u} \,|\, \mathbf{0}, \mathbf{K}_{\mathbf{uu}}) \, \mathrm{d}\mathbf{u} - \frac{\beta}{2} \operatorname{tr}(\mathbf{S}_{\mathbf{ff}}) \\ &= \log \mathcal{N}(\mathbf{y} \,|\, \mathbf{0}, \beta^{-1} \mathbf{I} + \mathbf{Q}_{\mathbf{fu}} \mathbf{K}_{\mathbf{uu}} \mathbf{Q}_{\mathbf{uf}}) - \frac{\beta}{2} \operatorname{tr}(\mathbf{S}_{\mathbf{ff}}) \\ &= \log \mathcal{N}(\mathbf{y} \,|\, \mathbf{0}, \mathbf{Q}_{\mathbf{ff}} + \beta^{-1} \mathbf{I}) - \frac{\beta}{2} \operatorname{tr}(\mathbf{S}_{\mathbf{ff}}). \end{aligned}$$

2.F SPECTRAL DENSITY OF THE SQUARED EXPONENTIAL KERNEL

We calculate the spectral density for the SE kernel in 1D from Equation (2.17). Using Equation (2.39), we have

$$\begin{aligned} p(\omega) &= \frac{1}{2\pi} \int k(t, 0) e^{i\omega t} \, \mathrm{d}t \\ &= \frac{\ell}{\sqrt{2\pi}} \int \mathcal{N}(t \,|\, 0, \ell^2) e^{i\omega t} \, \mathrm{d}t \\ &= \frac{\ell}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} \ell^2 \omega^2\right) = \mathcal{N}\left(\omega \,|\, 0, \ell^{-2}\right), \end{aligned}$$

as required.

2.G COSINE DIFFERENCE AS INNER PRODUCT

Firstly, recall the *angle sum-and-difference* trigonometric identity,

$$\cos \alpha \pm \beta = \cos \alpha \cos \beta \mp \sin \alpha \sin \beta. \tag{2.52}$$

Taking the inner product of ψ_ω evaluated at inputs $\mathbf{x}$ and $\mathbf{x}'$, we obtain

$$\begin{aligned} \psi_\omega(\mathbf{x})^\top \psi_\omega(\mathbf{x}') &= \cos\left(\boldsymbol{\omega}^\top \mathbf{x}\right) \cos\left(\boldsymbol{\omega}^\top \mathbf{x}'\right) + \sin\left(\boldsymbol{\omega}^\top \mathbf{x}\right) \sin\left(\boldsymbol{\omega}^\top \mathbf{x}'\right) \\ &= \cos\left(\boldsymbol{\omega}^\top (\mathbf{x} - \mathbf{x}')\right), \end{aligned} \tag{2.53}$$

as required.