1

Background

This book describes some recent developments in *scalable Monte Carlo* algorithms and their applications within Bayesian learning: what exactly does this mean?

Monte Carlo methods are a class of computational methods that involve repeated sampling to numerically approximate quantities of interest. We specifically focus on Monte Carlo integration methods, which are sampling-based methods for evaluating or approximating the value of integrals. Such methods are widely used across science and engineering, but our motivation comes particularly from Bayesian statistics. One of the key quantities in Bayesian statistics is the posterior distribution, which encapsulates our belief regarding unknown parameters, of a model given our prior belief and an observed dataset. We can then obtain estimates of the parameters, or quantify our uncertainty about the parameters, in terms of expectations with respect to the posterior distribution. For example, a common estimate of a parameter is the posterior expectation of that parameter; the predictive probability of future observations is the expectation of the density/mass function of the future observation taken with respect to the posterior distribution. Calculating these expectations involves evaluating an integral, and the idea of Monte Carlo is to use samples from the posterior to estimate such integrals.

The main challenge with using Monte Carlo in Bayesian statistics is often in deriving an efficient algorithm to sample from the posterior distribution. Markov chain Monte Carlo (MCMC) is a general and widely used class of methods for sampling from a distribution, based on simulating a Markov process that has the posterior distribution as its stationary distribution.

In recent years, there has been interest in applying MCMC to ever-increasingly complex and challenging problems. For example, the dimension, d, say, of the parameter space of the models we wish to fit to data, or the number of data points, N, say, in our dataset can be large. As either

d or N increases, the efficiency of MCMC methods may reduce. For example, as d increases, we may need to have more iterations of our MCMC algorithm to achieve the required level of accuracy, while as N increases, the computational cost per iteration of a standard algorithm will increase. *Scalable MCMC* methods are specifically those methods that can scale well as either or both d and N increase.

The remainder of this introductory chapter will cover background relevant to scalable MCMC. Section 1.1 will introduce Monte Carlo methods, explain why Monte Carlo integration is widely used, and explain how it is relevant to Bayesian statistics. This will be followed by an introduction to some of the statistical models and applications that will be used to demonstrate the methods in this book, as well as an informal and brief introduction to some of the concepts from stochastic processes that will be used in later chapters. Finally, the chapter ends with a short introduction to kernel methods in preparation for a deeper exposition in Chapter 6.

1.1 Monte Carlo Methods

1.1.1 What Is Monte Carlo Integration?

Assume we have a distribution of interest. For simplicity of presentation, here and for the remainder of this chapter, we assume that the distribution is continuous on $\mathbb{R}^d$. Let $\mathbf{X}$ denote a random variable with this distribution, and let $\pi(\mathbf{x})$ denote the corresponding probability density function for $\mathbf{X}$; we will also use π to refer to the distribution itself when that is necessary. Then the expectation of some function h of $\mathbf{X}$ is an integral

$$I = \mathbb{E}\left[h(\mathbf{X})\right] = \int h(\mathbf{x})\pi(\mathbf{x})\,\mathrm{d}\mathbf{x}.$$

This expectation is *well-defined*; that is, h is integrable with respect to π if $\int |h(\mathbf{x})|\pi(\mathbf{x})\,\mathrm{d}\mathbf{x} < \infty$. We abbreviate this to $h \in \mathcal{L}^1(\pi)$, and throughout this section, we assume that this holds true. If we can sample from $\pi(\cdot)$, then we can estimate this expectation/integral by (i) drawing n independent realisations, $\mathbf{x}_1, \ldots, \mathbf{x}_n$, from $\pi(\cdot)$ and (ii) calculating the sample average of the values $h(\mathbf{x}_1), \ldots, h(\mathbf{x}_n)$. This gives an estimate of I, namely

$$\hat{I} = \frac{1}{n}\sum_{k=1}^{n} h(\mathbf{x}_k).$$

This is called a *Monte Carlo* estimate of I, as it is obtained from independent, random samples from $\pi(\cdot)$.

The Monte Carlo estimator can be interpreted as being based on n independent random variables $\mathbf{X}_1, \ldots, \mathbf{X}_n$, of which $\mathbf{x}_1, \ldots, \mathbf{x}_n$ are realisations. Is it a good estimator? This is impossible to answer in generality, but we can at least describe some good properties that the estimator can admit. First, since the $\mathbf{X}_i$ are i.i.d., $\mathbb{E}\left[\hat{I}\right] = \mathbb{E}\left[h(\mathbf{X}_1)\right] = I$, so the estimator is *unbiased*. Second, and more importantly, the strong law of large numbers tells us that we can make our estimate arbitrarily accurate, with high probability, if we choose n large enough. Formally, provided I is well-defined, that is, $h \in \mathcal{L}^1(\pi)$, and our samples from $\pi(\cdot)$ are independent, then as $n \to \infty$,

$$\frac{1}{n}\sum_{k=1}^{n} h(\mathbf{X}_k) \to I \quad \text{almost surely.} \tag{1.1}$$

Almost sure convergence means that the collection of outcomes where the convergence does not occur has a combined probability of 0.

Thus, with high probability, our Monte Carlo estimator will be accurate if we choose n large enough, but the result does not tell us how large n needs to be, nor how accurate the estimator will be for a given value of n. However, provided that $\int h(\mathbf{x})^2\pi(\mathbf{x})\,\mathrm{d}\mathbf{x} < \infty$, which we abbreviate to $h \in \mathcal{L}^2(\pi)$, we can use the central limit theorem to answer these questions. Again assume that our samples from $\pi(\cdot)$ are independent, and define

$$V = \int \{h(\mathbf{x}) - I\}^2\pi(\mathbf{x})\,\mathrm{d}\mathbf{x}.$$

Then, the central limit theorem states that

$$\sqrt{n}\left(\frac{\frac{1}{n}\sum_{k=1}^{n} h(\mathbf{X}_k) - I}{\sqrt{V}}\right) \xrightarrow{\mathsf{D}} \mathsf{N}(0,1),$$

as $n \to \infty$. Here the convergence is in distribution, and we have convergence to a standard normal distribution in the limit.

One way of interpreting this result is that, for large enough n, approximately,

$$\frac{1}{n}\sum_{k=1}^{n} h(\mathbf{X}_k) \sim \mathsf{N}\left(I, \frac{V}{n}\right).$$

That is our estimator will be approximately normally distributed, with mean equal to the integral, I, and a variance that is V/n. This shows that the quantity V governs how easy it is to estimate I via Monte Carlo integration, and the accuracy depends on both V and n. The order of the error of a Monte Carlo estimator is $\sqrt{V/n}$, and thus, the Monte Carlo error decays with sample size at a rate of $n^{-1/2}$.

1.1.2 Importance Sampling

What if we are interested in calculating or approximating a more general integral, $I = \int_\Omega g(\mathbf{x})\mathrm{d}\mathbf{x}$, of some function g over a region Ω? We can use Monte Carlo sampling to estimate this integral by rewriting the integral as an expectation with respect to some density function $q(\cdot)$ defined on Ω as follows,

$$I = \int_\Omega \frac{g(\mathbf{x})}{q(\mathbf{x})} q(\mathbf{x})\ \mathrm{d}\mathbf{x} = \mathbb{E}\left[h(\mathbf{X})\right],$$

where $h(\mathbf{x}) = g(\mathbf{x})/q(\mathbf{x})$. If I is well-defined, that is, $\int |g(\mathbf{x})|\mathrm{d}\mathbf{x} < \infty$, then $h \in \mathcal{L}^1(q)$, and I can be estimated using Monte Carlo integration as mentioned above, based on independent realised samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ from $q(\cdot)$ by calculating the arithmetic mean of $h(\mathbf{x}_1), \ldots, h(\mathbf{x}_n)$. This process is called *importance sampling*, and $q(\cdot)$ is known as the proposal distribution.

For this Monte Carlo estimator to be feasible, we have two constraints on q. First, we need $q(\mathbf{x}) > 0$ whenever $g(\mathbf{x}) > 0$, in order for $h(\mathbf{x})$ to be well-defined. Second, we need to be able to easily sample from $q(\cdot)$. The choice of $q(\cdot)$ will affect the accuracy of the estimator, with the variance of our estimator for a Monte Carlo sample of size n being V/n, where

$$V = \int \left(\frac{g(\mathbf{x})}{q(\mathbf{x})} - I\right)^2 q(\mathbf{x})\ \mathrm{d}\mathbf{x}.$$

This variance will be small if $g(\mathbf{x})/q(\mathbf{x})$ is roughly constant, and one can show that the optimal choice of $q(\cdot)$ in terms of minimising V is $q(\mathbf{x}) \propto |g(\mathbf{x})|$. If $g(\mathbf{x})$ is non-negative everywhere (or non-positive everywhere), then such a choice of q will give an estimator that has zero-variance, that is, an exact estimator. More generally, the variance V will be large if there are values of $\mathbf{x}$ for which $g(\mathbf{x})/q(\mathbf{x})$ is large. This leads to a rule of thumb that, if Ω is unbounded, one wants $q(\mathbf{x})$ to have heavier tails than $|g(\mathbf{x})|$ to avoid this ratio blowing up as $\|\mathbf{x}\| \to \infty$.

1.1.3 Monte Carlo or Quadrature?

It is natural to ask why one should use Monte Carlo integration when there are alternative numerical integration methods, such as quadrature. To see the potential benefits of Monte Carlo methods, consider estimating an integral on the unit hypercube $[0, 1]^d$. We can then compare quadrature with Monte Carlo integration based on samples from a uniform distribution on $[0, 1]^d$.

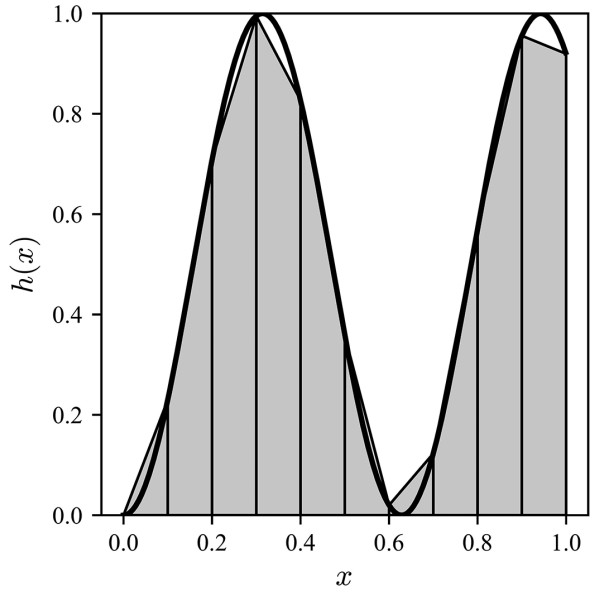

Figure 1.1 Example of trapezoid rule. We can estimate the integral by (i) setting $x_1, \ldots, x_n$ to be evenly spaced points on $[0, 1]$, (ii) creating $n - 1$ trapezoids based on joining up the points $(x_k, h(x_k))$ (shaded in regions), and (iii) estimating the integral by the sum of the areas of the trapezoids.

First, consider $d = 1$. In this case, quadrature methods tend to be much more accurate than the Monte Carlo methods. We have seen that the Monte Carlo variance, if we have n Monte Carlo samples, is $O(1/n)$, which means that the error of our Monte Carlo estimator will be $O_p(n^{-1/2})$.

By comparison, a simple numerical method is the trapezoidal rule. This involves evaluating the integrand, $h(x)$, at a set of equally spaced points, $x_1, \ldots, x_n$, on $[0, 1]$, and approximating the integral using the total area of the trapezoids formed by joining up the points $(x_k, h(x_k))$ for $k = 1, \ldots, n$ (Figure 1.1). Assuming our integrand has a bounded second derivative $|h''(x)| < L$ for some L, then we can bound the error in the estimate of the integral as $L\delta^2/12$, where $\delta = 1/(n-1)$ is the width of each trapezoid. This gives an error that decays like $O(1/n^2)$, which is much better than the Monte Carlo method. Furthermore, higher-order quadrature methods, such as Simpson's rule, can obtain even faster decay of the approximate error with n, if the integrand is sufficiently smooth.

So, quadrature methods can be more accurate for one-dimensional integrals, at least for functions whose second derivatives are bounded. How-

ever, now consider higher-dimensional integrals involving functions $h(\mathbf{x})$, the only information about which we have is that the second-order (partial) derivatives are bounded. Then we can apply a cubature rule based on a grid of $m + 1$ equally spaced points in each dimension. The spacing of these points will be $\delta = 1/m$, and there will be $n = (m + 1)^d$ points in total. If we have a cubature whose error decays like δ^r, for some power r, for example, $r = 2$ for the trapezoidal rule, then the error decays at a rate of $m^{-r} \approx n^{-r/d}$. For large d, this convergence will be slower than the $n^{-1/2}$ rate of Monte Carlo integration, explaining why Monte Carlo is often the default method for numerically approximating high-dimensional integrals. To overcome this *curse of dimension* in cubature, it is usually necessary to identify a sense in which the integrand $h(\mathbf{x})$ is effectively low-dimensional, which can be difficult or impossible depending on the applied context.

1.1.4 Control Variates

Let us return to the problem of estimating the expectation of some function of a random variable,

$$I = \mathbb{E}\left[h(\mathbf{X})\right] = \int h(\mathbf{x})\pi(\mathbf{x})\,\mathrm{d}\mathbf{x},$$

where $\pi(\mathbf{x})$ is the density of $\mathbf{X}$. We have seen how we can estimate this using a sample from $\pi(\cdot)$, and that the accuracy of this estimator is proportional to

$$V = \int \left\{h(\mathbf{x}) - I\right\}^2 \pi(\mathbf{x})\,\mathrm{d}\mathbf{x} = \int h(\mathbf{x})^2\pi(\mathbf{x})\,\mathrm{d}\mathbf{x} - I^2.$$

The latter expression is just the standard expression for the variance of $h(\mathbf{X})$. This shows that it is easier to estimate expectations of functions that vary less when evaluated at $\mathbf{X}$.

Assume that we know the expectation of a set of random variables $g_1(\mathbf{X}), \ldots, g_J(\mathbf{X})$, each a transformation of $\mathbf{X}$. Without loss of generality, we can assume that these random variables have mean zero, that is,

$$\mathbb{E}\left[g_j(\mathbf{X})\right] = 0, \quad \text{for } j = 1, \ldots, J,$$

as, if this is not the case, we can define new random variables equal to the old random variables minus their expectations. Then, for any constants $\gamma_1, \ldots, \gamma_n$,

$$I = \mathbb{E}\left[h(\mathbf{X})\right] - \sum_{j=1}^{J} \gamma_j \mathbb{E}\left[g_j(\mathbf{X})\right] = \mathbb{E}\left[h(\mathbf{X}) - \sum_{j=1}^{J} \gamma_j g_J(\mathbf{X})\right]. \tag{1.2}$$

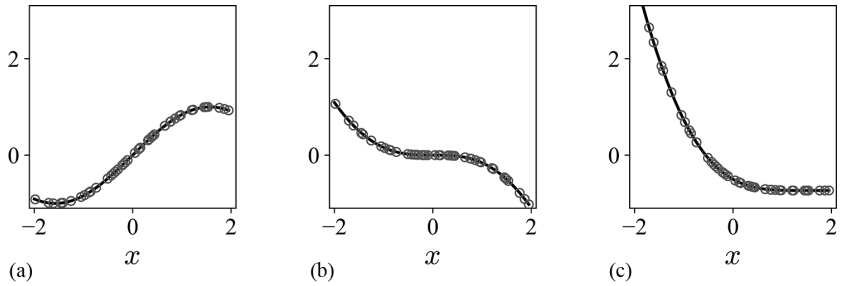

Figure 1.2 Example of control variates for estimating $\mathbb{E}[\sin(X)]$, where X has a standard normal distribution $\mathsf{N}(0, 1)$. Each plot shows the function whose expectation is being estimated and 50 values used in the Monte Carlo estimate (dots). The functions are as follows: (a) $h(x) = \sin(x)$, (b) $h(x) = \sin(x) - x$, and (c) $h(x) = \sin(x) - \pi x/2 + (x^2 - 1)/2$. The expectation of each function is constructed to be the same. The effect of introducing control variates in the middle and right-hand plot is to flatten out the function we are integrating – in the middle plot, this happens for $x \approx 0$ and for the right-hand plot for $x \approx \pi/2$. The variability of the function values, that is, the dots, is smallest for the middle plot and largest for the right-hand plot.

By suitable choice of the constants $\gamma_1, \ldots, \gamma_J$, the variability of the random variable $h(\mathbf{X}) - \sum_{j=1}^{J} \gamma_j g_j(\mathbf{X})$ can be made smaller than that of $h(\mathbf{X})$, and thus a Monte Carlo estimate of I based on (1.2) will have smaller Monte Carlo variance than the basic Monte Carlo estimator. We call $\sum_{j=1}^{J} \gamma_j g_j(\mathbf{X})$ a *control variate* for $h(\mathbf{X})$. Heuristically, we want to choose $\gamma_1, \ldots, \gamma_J$ so that $h(\mathbf{X}) \approx \gamma_0 + \sum_{j=1}^{J} \gamma_j g_j(\mathbf{X})$, which means that $h(\mathbf{X}) - \sum_{j=1}^{J} \gamma_j g_j(\mathbf{X})$ is approximately constant.

As a simple example, consider estimating the expectation of $\sin(X)$, where X has a standard normal distribution $\mathsf{N}(0, 1)$. We know that this expectation is 0 as the distribution of X is symmetric about 0 and $\sin(-x) = -\sin(x)$. We will compare the simple Monte Carlo estimator of the expectation with estimates using control variates with the functions $g_1(x) = x$ and $g_2(x) = x^2 - 1$. By using a Taylor expansion of $\sin(x)$ at $x = 0$, we have $\sin(x) \approx x$ for small x, and thus a simple choice of control variate is $g_1(x)$.

We show pictorially the benefit of using this control variate in Figure 1.2, where we see that $\sin(x) - x \approx 0$ for most x values sampled from the standard normal distribution. This reduces the Monte Carlo variance of the estimate of $\mathbb{E}[h(X)]$ by close to a factor of 2.

Care must be taken with control variates, however. For example, if we perform a Taylor expansion of $\sin(x)$ at $x = \pi/2$, we get $\sin(x) \approx 1 - (x - \pi/2)^2/2$, which suggests using $-g_2(x)/2 + \pi g_1(x)/2$ as a control variate. However, this choice leads to an increase in the Monte Carlo variance by over a factor of 3. Figure 1.2 shows that the function $\sin(x) - \pi x/2 + (x^2 - 1)/2$ is roughly constant for $x \approx \pi/2$, but overall it is more variable across the range $x \in [-2, 2]$, where most of the probability mass of $\mathsf{N}(0, 1)$ lies.

This example shows that the choice of $\gamma_1, \ldots, \gamma_J$ is important when using control variates. In some situations, there may be a natural way of choosing these – for example, based on a Taylor expansion of the function of interest around the mode of the distribution of $\mathbf{X}$. However, it is also possible to choose these values based on simulation. Ideally, we would choose $\gamma_1, \ldots, \gamma_J$ to minimise the Monte Carlo variance

$$\int \left\{ h(\mathbf{x}) - \sum_{j=1}^{J} \gamma_j g_j(\mathbf{x}) \right\}^2 \pi(\mathbf{x}) \, \mathrm{d}\mathbf{x} - I^2,$$

and we can obtain a Monte Carlo estimate of this. If $\mathbf{x}_1, \ldots, \mathbf{x}_n$ are realised samples from $\pi(\cdot)$, then we can choose $\gamma_1, \ldots, \gamma_J$ to minimise

$$\sum_{k=1}^{n} \left(h(\mathbf{x}_k) - \sum_{j=1}^{J} \gamma_j g_j(\mathbf{x}_k) \right)^2,$$

which just involves minimising a sum of squares criterion. If we let $\mathbf{h}$ be the $n \times 1$ vector whose ith entry is $h(\mathbf{x}_i)$, $\boldsymbol{\gamma}$ be the $J \times 1$ vector whose ith entry is γ_i, and $\mathbf{Z}$ be the $n \times J$ matrix whose (i, j)th entry is $g_j(\mathbf{x}_i)$, then, assuming $\mathbf{Z}$ is of full rank, the least-squares estimate of $\boldsymbol{\gamma}$ is

$$\hat{\boldsymbol{\gamma}} = (\mathbf{Z}^\top \mathbf{Z})^{-1} \mathbf{Z}^\top \mathbf{h}.$$

These estimates $\hat{\boldsymbol{\gamma}}$ depend on the Monte Carlo samples, and thus for the Monte Carlo estimate of I to be unbiased, we need to use a new set of Monte Carlo samples from $\mathbf{X}$ for estimating I using the $\hat{\boldsymbol{\gamma}}$.

While we have presented the idea of control variates for estimating expectations of functions, similar ideas can be used with importance sampling for estimating general integrals.

1.1.5 Monte Carlo Integration and Bayesian Statistics

One of the most important applications of Monte Carlo methods occurs within Bayesian statistics. To explain why, consider the problem of making inferences, from data, about the parameter of a statistical model. We will

use the notation $\mathcal{D}$ to denote data in general. In some situations, we will need to distinguish individual data points, and in those settings, we will assume $\mathcal{D} = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$, with $\mathbf{y}_i$ being the ith data point and N being the size of our dataset.

We further assume that we have a model for the data. Let the model depend on a parameter $\boldsymbol{\theta}$, and denote the likelihood of the data under our model by $L(\boldsymbol{\theta}; \mathcal{D})$. The likelihood is the probability, or probability density, of observing data $\mathcal{D}$ under our model if the parameter is $\boldsymbol{\theta}$. In Bayesian statistics, we represent beliefs, or uncertainty, about the parameter, $\boldsymbol{\theta}$, through probability distributions. Our beliefs about $\boldsymbol{\theta}$ before seeing the data are given by a prior, $\pi_0(\boldsymbol{\theta})$, and, once we observe data, Bayes' theorem provides the update to the posterior distribution

$$\pi(\boldsymbol{\theta} \mid \mathcal{D}) \propto \pi_0(\boldsymbol{\theta})\, L(\boldsymbol{\theta}; \mathcal{D}). \tag{1.3}$$

Where it will not cause confusion, we may drop the explicit conditioning on the data in the posterior and write $\pi(\boldsymbol{\theta})$ rather than $\pi(\boldsymbol{\theta} \mid \mathcal{D})$.

Assuming the correctness of our model, the posterior distribution contains all information about the parameter, $\boldsymbol{\theta}$, that can be logically deduced from our prior belief and the dataset. From it, we can then obtain a point estimate for $\boldsymbol{\theta}$, such as its posterior mean, and quantify uncertainty in terms of the posterior probability of $\boldsymbol{\theta}$ lying in a given set of values. However, in most applications, the posterior distribution is intractable, meaning that it cannot be explicitly calculated. The central challenge is that the posterior density $\pi(\boldsymbol{\theta} \mid \mathcal{D})$ is known, via Bayes' theorem, only up to a normalising constant.

The intractability of the posterior distribution is a key motivator for Monte Carlo methods. If we can draw samples from $\pi(\boldsymbol{\theta} \mid \mathcal{D})$, then we can obtain simple, and often accurate, Monte Carlo estimates of posterior quantities of interest. Given realisations $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n$ sampled from $\pi(\boldsymbol{\theta} \mid \mathcal{D})$ and a function $h(\boldsymbol{\theta})$ whose expectation

$$I := \mathbb{E}_\pi[h(\boldsymbol{\theta})] = \int h(\boldsymbol{\theta})\pi(\boldsymbol{\theta} \mid \mathcal{D})\, \mathrm{d}\boldsymbol{\theta}$$

is of interest, define

$$\widehat{I}_n(h) := \frac{1}{n}\sum_{k=1}^{n} h(\boldsymbol{\theta}_k). \tag{1.4}$$

As mentioned earlier, for any function $h \in \mathcal{L}^1(\pi)$, the strong law of large numbers (1.1) tells us that we can estimate $\mathbb{E}_\pi[h(\boldsymbol{\theta})]$ as accurately as

we desire using Monte Carlo integration, and provided enough samples are taken: $\widehat{I}_n(h) \to \mathbb{E}_\pi[h(\boldsymbol{\theta})]$ almost surely as $n \to \infty$. Moreover, if $h \in \mathcal{L}^2(\pi)$, then the central limit theorem states that the Monte Carlo error, $\widehat{I}_n(h) - \mathbb{E}_\pi[h(\boldsymbol{\theta})]$, is $O_p(n^{-1/2})$.

For example, the vector of posterior means can be estimated by

$$\hat{\boldsymbol{\theta}} = \frac{1}{n}\sum_{k=1}^{n} \boldsymbol{\theta}_k,$$

and the posterior probability of $\boldsymbol{\theta} \in \mathcal{B}$ for some set $\mathcal{B}$ can be estimated by the proportion of Monte Carlo samples in $\mathcal{B}$

$$\hat{\mathbb{P}}(\boldsymbol{\theta} \in \mathcal{B}) = \frac{1}{n}\sum_{k=1}^{n} \mathbb{I}\{\boldsymbol{\theta}_k \in \mathcal{B}\},$$

where $\mathbb{I}\{\boldsymbol{\theta}_k \in \mathcal{B}\}$ is the indicator function of the event $\boldsymbol{\theta}_k \in \mathcal{B}$.

The challenge with this Monte Carlo approach to Bayesian statistics is the difficulty in sampling from $\pi(\boldsymbol{\theta})$, particularly if $\boldsymbol{\theta}$ is high-dimensional. Of the Monte Carlo integration methods we have mentioned so far, importance sampling offers an alternative when we are unable to sample from π directly. Consider estimating the posterior expectation for some function $h(\boldsymbol{\theta})$, so $h(\boldsymbol{\theta}) = \boldsymbol{\theta}$ would give us the posterior mean of $\boldsymbol{\theta}$ and $h(\boldsymbol{\theta}) = \mathbb{I}\{\boldsymbol{\theta} \in \mathcal{B}\}$ would give us the posterior probability of $\boldsymbol{\theta} \in \mathcal{B}$. Let $q(\boldsymbol{\theta})$ be a proposal distribution with the same support as the posterior. Then we have

$$\mathbb{E}[h(\boldsymbol{\theta}) \mid \mathcal{D}] = \int h(\boldsymbol{\theta})\pi(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta} = \int \frac{h(\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{q(\boldsymbol{\theta})} q(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta}.$$

It is common to define *weights* $w(\boldsymbol{\theta}) := \pi(\boldsymbol{\theta})/q(\boldsymbol{\theta})$. Then given an independent sample $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n$ from $q(\boldsymbol{\theta})$, we can estimate the posterior expectation by the importance sampling estimator

$$\frac{1}{n}\sum_{k=1}^{n} w(\boldsymbol{\theta}_k) h(\boldsymbol{\theta}_k).$$

There are two issues with this estimator. The first is that as we only know the posterior up to a constant of proportionality, we only know the weights up to a constant of proportionality. However, the constant of proportionality can be estimated by setting $h(\boldsymbol{\theta}) = 1$, whence $\mathbb{E}[h(\boldsymbol{\theta})] = 1$ as the expectation of a constant is the constant. Thus we can use the unnormalised posterior density in the definition of the weights and estimate the

normalising constant as $(1/n)\sum_{k=1}^{n} w(\boldsymbol{\theta}_k)$. The posterior expectation of $h(\boldsymbol{\theta})$ is then estimated as

$$\sum_{k=1}^{n} \frac{w(\boldsymbol{\theta}_k)}{\sum_{j=1}^{n} w(\boldsymbol{\theta}_j)} h(\boldsymbol{\theta}_k),$$

which requires knowing the posterior density only up to a constant of proportionality. Often we define normalised weights, $w^*(\boldsymbol{\theta}_k) = w(\boldsymbol{\theta}_k)/\sum_{j=1}^{n} w(\boldsymbol{\theta}_j)$, and we can then view the weighted samples $(\boldsymbol{\theta}_k, w^*(\boldsymbol{\theta}_k))$, for $k = 1, \ldots, n$, as a discrete approximation to the posterior.

The second issue is that the Monte Carlo variances of our estimators of posterior expectations depend on the variability of the weights. Often this will be large if $\boldsymbol{\theta}$ is high-dimensional. To see this, consider a toy example where the posterior has independent components. Assume each component is normal with mean 0 and variance σ^2, and the importance-sampling proposal distribution is also independent over components, but with a standard normal distribution, that is, with mean zero and unit variance, for each component. The importance sampling weight for a realisation $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_d)$ is

$$w(\boldsymbol{\theta}) = \sigma^{-d} \exp\left\{ \frac{\sigma^2 - 1}{2\sigma^2} \sum_{i=1}^{d} \theta_i^2 \right\}.$$

Now $\sum_{i=1}^{d} \theta_i^2$ has a χ_d^2 distribution under the proposal, and using the moment generating function of a χ_d^2 distribution, we obtain the Monte Carlo variance of the weight:

$$\mathrm{var}\{w(\boldsymbol{\theta})\} = \sigma^{-d} \left(2 - \sigma^2\right)^{-d/2} - 1.$$

Writing $\sigma^2 = 1 + \epsilon$, for some $\epsilon > 0$, this variance is $(1/\sqrt{1-\epsilon^2})^d - 1$, which increases exponentially with d. The focus of MCMC methods that we introduce in Chapter 2 is to produce sampling algorithms that avoid this exponential curse of dimensionality.

1.2 Example Applications

In later chapters, we will demonstrate the Monte Carlo methods on some example models that we now introduce. Whilst these models are somewhat simple to describe, their posteriors exhibit many of the features of more challenging posterior distributions, in particular, with respect to scalable sampling.

1.2.1 Logistic Regression

Logistic regression models the relationship between a binary response and a set of covariates. Denote the responses by $y_1, \ldots, y_N$ and the covariates by d-dimensional vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$. Then, logistic regression models the data (the responses) as conditionally independent, given a d-dimensional parameter $\boldsymbol{\theta}$ and the covariates, and that

$$\mathbb{P}\left(Y = y_j | \mathbf{x}_j, \boldsymbol{\theta}\right) = \frac{\exp\{y_j \mathbf{x}_j^\top \boldsymbol{\theta}\}}{1 + \exp\{\mathbf{x}_j^\top \boldsymbol{\theta}\}}.$$

An intercept term can be included in the model by setting the first coordinate of each of $\mathbf{x}_1, \ldots, \mathbf{x}_N$ to be 1.

Our interest will be in sampling from the posterior distribution of $\boldsymbol{\theta}$. To define the posterior, we need to specify a prior $\pi_0(\boldsymbol{\theta})$, and we will assume that our prior is Gaussian with mean $\mathbf{0}$ and variance $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$. This leads to a posterior distribution, $\pi(\boldsymbol{\theta}|\mathcal{D})$, which can be written succinctly up to a multiplicative constant as

$$\pi(\boldsymbol{\theta}|\mathcal{D}) \propto \exp\left\{-\frac{1}{2}\boldsymbol{\theta}^\top \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{\theta}\right\} \prod_{j=1}^{N} \frac{\exp\{y_j \mathbf{x}_j^\top \boldsymbol{\theta}\}}{1 + \exp\{\mathbf{x}_j^\top \boldsymbol{\theta}\}}.$$

This is a canonical, albeit relatively simple, test problem for sampling methodologies. When we consider sampling methods for this model, we will drop the explicit conditioning on data $\mathcal{D}$ and use $\pi(\boldsymbol{\theta})$ to denote the target distribution of the sampler. The samplers we consider will often use gradient information about their target distribution, and we have

$$\frac{\partial \log \pi(\boldsymbol{\theta})}{\partial \theta_i} = -\left[\boldsymbol{\theta}^\top \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}\right]_i + \sum_{j=1}^{N} x_j^{(i)} \left\{ y_j - \frac{\exp\{\mathbf{x}_j^\top \boldsymbol{\theta}\}}{1 + \exp\{\mathbf{x}_j^\top \boldsymbol{\theta}\}} \right\}, \tag{1.5}$$

where $x_j^{(i)}$ indicates the ith component of $\mathbf{x}_j$.

1.2.2 Bayesian Matrix Factorisation

Bayesian matrix factorisation attempts to find a representation of a high-dimensional matrix as the product of two lower-dimensional matrices. Consider an $n \times m$ matrix $\mathbf{Y}$, and let $\boldsymbol{\theta} = \{\mathbf{U}, \mathbf{V}\}$, where $\mathbf{U}$ and $\mathbf{V}$ are $n \times d$ and $d \times m$ matrices, respectively. Then the approximation is $\mathbf{Y} \approx \mathbf{UV}$. If $d \ll \min\{m, n\}$, then this can lead to a substantial reduction in dimension, and the model can be viewed as attempting to find low-dimensional structure in $\mathbf{Y}$.

The interpretation of this model is that each row of $\mathbf{V}$ is a *factor*, and we aim to approximate each row of $\mathbf{Y}$ as a linear combination of these factors. The entries in $\mathbf{U}$ are called *factor loadings* and give the relative weight of each factor for each row of $\mathbf{Y}$.

One common approach to fitting these models is to use a Gaussian working model; thus, up to additive constants, the log-likelihood is

$$L(\boldsymbol{\theta}; \mathcal{D}) = -nm \log \sigma - \frac{1}{2\sigma^2} \left\{ \sum_{i=1}^{n} \sum_{j=1}^{m} \left(Y_{i,j} - \sum_{k=1}^{d} U_{i,k} V_{k,j} \right)^2 \right\},$$

where σ^2 is the variance of the difference between entries of $\mathbf{Y}$ and $\mathbf{UV}$. In Bayesian matrix factorisation, we then introduce a prior on the parameters $\mathbf{U}$ and $\mathbf{V}$. Often, the prior for each entry is Gaussian, or is a mixture of a Gaussian and a point mass at zero, as this encourages sparsity in the factors, which potentially aids the interpretation of $\mathbf{U}$ and $\mathbf{V}$. It is also possible to introduce a prior over the number of factors, d, with the priors for the entries of $\mathbf{U}$ and $\mathbf{V}$ potentially depending on d.

1.2.3 Bayesian Neural Networks for Classification

Artificial neural networks are a flexible and popular class of models used in machine learning for solving supervised learning problems, such as regression and classification tasks. In the case of classification, assume that $y_1, y_2, \ldots, y_N$ are observed data, where each y_j represents one of G classes, that is, $y_j \in \{1, 2, \ldots, G\}$. Assuming d-dimensional vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ for the covariates, then under a simple two-layer neural network model, the probability of a particular class y_j is

$$\mathbb{P}(Y = y_j | \mathbf{x}_j, \boldsymbol{\theta}) \propto \exp(\mathbf{A}_{y_j}^\top \sigma(\mathbf{B}^\top \mathbf{x}_j + \mathbf{b}) + a_{y_j}), \tag{1.6}$$

where $\mathbf{b}$ is a d_h-dimensional vector and $\mathbf{B}$ is a $d \times d_h$ matrix, with d_h the dimension of the variables in the hidden layer. The function $\sigma : \mathbb{R}^{d_h} \to (0,1)^{d_h}$ is a vector softmax function with $\sigma(\mathbf{z})_i = \exp(z_i)/\{\sum_{j=1}^{d_h} \exp(z_j)\}$ for $i = 1, \ldots, d_h$. The notation $\mathbf{A}_i$ refers to the ith column of the $d_h \times G$ matrix $\mathbf{A}$. The parameters of the model $\boldsymbol{\theta} = \text{vec}(\mathbf{a}, \mathbf{A}, \mathbf{b}, \mathbf{B})$ are represented by vectors $\mathbf{a}$ and $\mathbf{b}$, commonly referred to as *biases*, and matrices $\mathbf{A}$ and $\mathbf{B}$, commonly referred to as *weights*.

Taking a Bayesian approach to parameter estimation, we can place independent Gaussian priors on each of the elements of the biases and weights in $\boldsymbol{\theta}$. Monte Carlo algorithms can be used to sample from the posterior,

$$\pi(\boldsymbol{\theta}|\mathcal{D}) \propto \pi_0(\boldsymbol{\theta}) \prod_{j=1}^{N} \mathbb{P}(y_j|\mathbf{x}_j, \boldsymbol{\theta}). \tag{1.7}$$

For Bayesian neural network models, the dataset sizes tend to be very large, and approximating the posterior distribution requires Monte Carlo methods, which are scalable to large datasets. In Chapter 3, we will use stochastic gradient MCMC algorithms to approximate the posterior distribution of the Bayesian neural network model.

1.3 Markov Chains

This section describes discrete-time Markov chains, focusing on the concepts that will be required to understand the MCMC method and its efficiency: the stationary distribution, reversibility, convergence to the stationary distribution, ergodic averages, integrated auto-correlation time, and effective sample size.

Definition 1.1. A *discrete-time Markov chain* on a state space $\mathcal{X}$ is a collection of random variables $\{X_k\}_{k=0}^{\infty}$ with each $X_k \in \mathcal{X}$, such that for any $\mathcal{A} \subseteq \mathcal{X}$,

$$\mathbb{P}\left(X_{k+1} \in \mathcal{A} \mid X_k = x_k, \ldots, X_0 = x_0\right) = \mathbb{P}\left(X_{k+1} \in \mathcal{A} \mid X_k = x_k\right); \tag{1.8}$$

conditional on the current state, the distribution of the next state is independent of all previous states.

In this chapter, we will only consider *homogeneous* Markov chains, where the distribution of the next state given the current state does not depend on the value of k. Such a chain has a stationary distribution, ν, if $X_k \sim \nu \implies X_{k+1} \sim \nu$. If the chain also has a unique limiting distribution, then this must be ν, since, by repeated induction, if $X_j \sim \nu$, then $X_k \sim \nu$ for all $k > j$, including as $k \to \infty$.

The following two examples of Markov chains on the vertices of an m-sided polygon illustrate different ways that a chain can be stationary. We label the vertices of the polygon from 0 to $m-1$, increasing in a clockwise direction; thus, $\mathcal{X} = \{0, 1, \ldots, m-1\}$.

Example 1.2. (See Figure 1.3a.) Let $\{X_k\}_{k=0}^{\infty}$ be a Markov chain on the vertices of an m-sided polygon where the state at time $k+1$ is obtained from the state at time k by moving to the next vertex in a clockwise direction. If at time k the chain is equally likely to be at each of the vertices, then this is

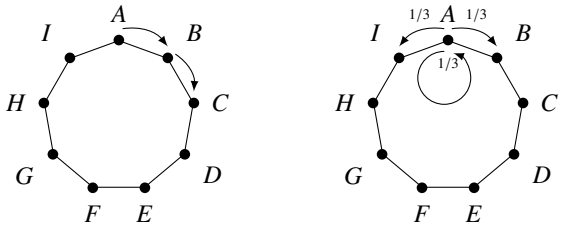

Figure 1.3 Nine-sided polygon where the Markov chain only moves clockwise (a), as in Example 1.2, or moves either a clockwise or anti-clockwise direction with probability $1/3$ (b), as in Example 1.3.

still the case at time $k+1$. The stationary distribution has $\mathbb{P}(X_k = x) = 1/m$ for $x \in \mathcal{X}$.

Example 1.3. (See Figure 1.3b.) Let $\{X_k\}_{k=0}^{\infty}$ be a Markov chain on the vertices of an n-sided polygon where the state at time $k+1$ is obtained from the state at time k by performing one of the following moves, each of which has a probability of $1/3$: move to the next vertex in an anti-clockwise direction; do not move; move to the next vertex in a clockwise direction. As with Example 1.2, the stationary distribution has $\mathbb{P}(X_k = x) = 1/m$ for $x \in \mathcal{X}$.

1.3.1 Reversible Markov Chains

Example 1.2 has a clear flow in a clockwise direction and, because of this, is an example of a non-reversible Markov chain; these will be discussed in detail in Chapter 4. By contrast, in Example 1.3, consider any two adjacent vertices: at stationarity, the probability of being at the first and moving to the second is the same as the probability of being at the second and moving to the first. Indeed, this is true of any pair of vertices, with the probability being 0 if they are not adjacent. This is an example of a reversible Markov chain.

Definition 1.4. A Markov chain $\{X_k\}_{k=1}^{\infty}$ with a state space of $\mathcal{X}$ is *reversible* with respect to a distribution ν when, for any two sets $\mathcal{B}, \mathcal{C} \subseteq \mathcal{X}$, if $X_k \sim \nu$, then $\mathbb{P}(X_k \in \mathcal{B}, X_{k+1} \in \mathcal{C}) = \mathbb{P}(X_k \in \mathcal{C}, X_{k+1} \in \mathcal{B})$.

Consider the decomposition

$$\mathbb{P}(X_k \in \mathcal{B}, X_{k+1} \in C) = \mathbb{P}(X_k \in \mathcal{B})\,\mathbb{P}(X_{k+1} \in C | X_k \in \mathcal{B}).$$

The first term on the right-hand side is the amount of probability mass in $\mathcal{B}$ at time k and the second term is the fraction of that mass that moves to C at time $k+1$, so the product is the amount of probability mass moving from $\mathcal{B}$ to C. If the chain is reversible with respect to ν and $X_k \sim \nu$, then this is also the amount of mass moving from C to $\mathcal{B}$. Given this balance, referred to as *detailed balance*, we would expect the total amount of probability mass in any set to remain constant. Indeed, setting $C = \mathcal{X}$ in Definition 1.4, we see that reversibility implies that if $X_k \sim \nu$, $\mathbb{P}(X_k \in \mathcal{B}) = \mathbb{P}(\mathcal{X}_{k+1} \in \mathcal{B})$. Since this is also true for all $\mathcal{B}$, $X_{k+1} \sim \nu$.

1.3.2 Convergence, Averages, and Variances

In Example 1.3, whatever the value or distribution of X_0, as $k \to \infty$, the distribution of X_k converges to the stationary distribution. For simplicity of presentation, we show this when $m = 2m' + 1$ is odd. For all $x_0, x \in \mathcal{X}$, $\mathbb{P}(X_{m'} = x | X_0 = x_0) \geq 1/3^{m'}$ since it takes at most m' moves in a single direction to reach x, and if the chain arrives earlier, we include the probability of it staying at x until time m'. Thus, the transition probability after m' steps can be written as a mixture

$$\mathbb{P}(X_{m'} = x | X_0 = x_0) = \delta\nu(x) + (1 - \delta)q(x|x_0), \tag{1.9}$$

for some conditional probability mass function q and with $\delta = m/3^{m'}$. The distribution at the start of a given iteration can always be thought of as a mixture of ν and some other distribution, where the mixture probability for ν could be 0. We can imagine that there is a hidden coin, and if it is showing 'heads', then the distribution of the chain is ν. Since ν is the stationary distribution, if the coin is currently showing 'heads', it will still be showing heads after a further m' moves. If the coin is showing 'tails', then (1.9) tells us that there is a probability of at least δ that it will be showing heads after the next m' moves. Equivalently, the mixture probability of the component that is not ν has been multiplied by $1 - \delta$ or less. After km' iterations, it is, therefore, at most $(1 - \delta)^k \to 0$ as $k \to \infty$.

However, convergence to a stationary distribution does not occur for all Markov chains. The chain in Example 1.2 is deterministic: if $X_0 = 0$, then $X_{km} = 0$ for all integers, k. The following examples illustrate two further cases.

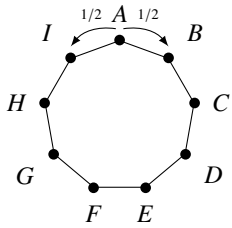

Figure 1.4 Nine-sided polygon with Markov transitions described in Example 1.5.

Example 1.5. (See Figure 1.4.) Alter Example 1.3 so that the chain cannot remain at its current vertex but must move either clockwise or anti-clockwise by a single vertex, each with a probability of $1/2$. As with the Examples 1.2 and 1.3, the stationary distribution has $\mathbb{P}(X_k = x) = 1/m$ for $x \in \mathcal{X}$.

If n is an even number and X_0 is even, then the chain in Example 1.5 only visits even-numbered states at even-numbered times and odd-numbered states at odd-numbered times. Such chains are termed *periodic* and clearly do not converge to their stationary distribution.

Example 1.6. Consider a Markov chain of the form in Example 1.3, but on two separate m-sided polygons with no movement between the two. A chain with separate regions between which there can be no movement is termed *reducible* because it can be reduced to simpler component parts.

A reducible chain does not even have a single stationary distribution. In Example 1.6 for any $\beta \in [0, 1]$, the distribution with probabilities β/m for each vertex on the first polygon and $(1-\beta)/m$ for each vertex on the second polygon is stationary.

A chain which is not reducible is termed *irreducible*, and a chain which is not periodic is termed *aperiodic*.

Ergodic Averages

The *ergodic theorem* for a Markov chain on a general state space, $\mathcal{X}$, states that provided the chain satisfies natural generalisations of irreducibility and aperiodicity and has a proper stationary distribution, ν, then as $k \to \infty$, the distribution of X_k converges to that stationary distribution. Furthermore, subject to the same conditions, for any $h \in \mathcal{L}^1(\nu)$, samples from the Markov chain satisfy a strong law of large numbers

$$\widehat{I}_n(h) := \frac{1}{n}\sum_{k=1}^{n} h(X_k) \to \mathbb{E}_\nu\left[h(X)\right], \tag{1.10}$$

almost surely as $n \to \infty$.

Integrated Auto-Correlation Time and Effective Sample Size

Let us assume that X_0 is, in fact, drawn from the stationary distribution. Define $\sigma_h^2 := \mathrm{Var}_\nu\left[h(X)\right]$ and assume $\sigma_h^2 < \infty$. For $k \in \{0, 1, 2, \dots\}$, the lag-k auto-correlation is $\rho_k := \mathrm{Cor}\left[h(X_0), h(X_k)\right] = \mathrm{Cor}\left[h(X_j), h(X_{j+k})\right]$ since the Markov chain is time-homogeneous. If the X_j were independent samples from ν, then $n\mathrm{Var}\left[\widehat{I}_n(h)\right] = \sigma_h^2$. For a stationary Markov chain with

$$\sum_{k=1}^{\infty} |\rho_k| < \infty, \tag{1.11}$$

it holds that

$$\lim_{n\to\infty} n\mathrm{Var}\left[\widehat{I}_n(h)\right] = \sigma_h^2\, \mathsf{IACT}_h, \tag{1.12}$$

where

$$\mathsf{IACT}_h := 1 + 2\sum_{k=1}^{\infty} \rho_k \tag{1.13}$$

is the *integrated auto-correlation time*. To see why this is the case, first, without loss of generality, take h to have $\mathbb{E}_\nu\left[h(X)\right] = 0$; if this is not true initially, we subtract off the expectation: the variance properties are unchanged. Then

$$n\mathrm{Var}\left[\widehat{I}_n(h)\right] = \frac{1}{n}\mathbb{E}\left[\sum_{k=1}^{n}\sum_{j=1}^{n} h(X_j)h(X_k)\right] = \frac{\sigma_h^2}{n}\sum_{k=1}^{n}\sum_{j=1}^{n}\rho_{|k-j|}. \tag{1.14}$$

But $\sum_{k=1}^{n}\sum_{j=1}^{n}\rho_{|k-j|} = n\rho_0 + 2\sum_{k=1}^{n}(n-k)\rho_k$, so

$$\frac{n}{\sigma_h^2}\mathrm{Var}\left[\widehat{I}_n(h)\right] = 1 + 2\sum_{k=1}^{n}\left(1 - \frac{k}{n}\right)\rho_k = 1 + 2\sum_{k=0}^{\infty}\max\left(0, 1 - \frac{k}{n}\right)\rho_k.$$

Given (1.11), the dominated converge theorem permits us to exchange the ordering of the limit as $n \to \infty$ and the sum over k, which provides the limit (1.12).

The practical consequence of (1.12) is that, for finite n,

$$\mathrm{Var}\left[\widehat{I}_n(h)\right] \approx \frac{\sigma_h^2}{n/\mathsf{IACT}_h}, \tag{1.15}$$

the same as the variance if n/IACT_h i.i.d. samples from ν had been used. The quantity n/IACT_h is, therefore, known as the *effective sample size*. Since they relate directly to the inverse variance of $\widehat{I}_n(h)$, effective sample size and the inverse of the integrated auto-correlation time are useful measures of the efficiency of a Markov chain for estimating $\mathbb{E}_\nu[h(X)]$.

1.4 Stochastic Differential Equations

The Langevin stochastic differential equation (SDE) is the basis for the Metropolis-Adjusted Langevin algorithm (Section 2.1.4) and for stochastic gradient Langevin methods (Chapter 3). It is also key to understanding the efficiency of various Metropolis–Hastings algorithms when the dimension, d, is high (see Chapter 2). We start with a heuristic introduction to SDEs before considering a special case of the Langevin diffusion known as the Ornstein–Uhlenbeck (OU) process and then moving on to the general Langevin diffusion.

Consider a differential equation of the form

$$\frac{\mathrm{d}x_t}{\mathrm{d}t} = a(x_t, t),$$

with a known initial value for x_0. Discretising time leads to the simple Euler approximation

$$\delta x_t \approx a(x_t, t)\delta t,$$

where $\delta x_t = x_{t+\delta t} - x_t$. Setting $\delta t = T/m$, starting with x_0, and recursively applying the Euler update m times leads to an approximation $\widehat{x}_T$, which approaches the true value x_T as $m \to \infty$.

Instead of deterministic updates, we might wish to allow for the addition of random noise with a scale proportional to $b(x_t, t)$. The initial value, X_0, may now be random, and setting $\delta X_t := X_{t+\delta t} - X_t$ leads to one possible update

$$\delta X_t \approx a(X_t, t)\delta t + b(X_t, t)\epsilon_t, \qquad \epsilon_t \sim \mathsf{N}(0, \delta t),$$

where the Gaussian noise terms ϵ_t are independent of all previous randomness, and X_t has become a random variable. A noise distribution of the form $\mathsf{N}(0, \delta t)$ is chosen because it is self-consistent. For example, with

$a(X_t, t) = a$ and $b(x_t, t) = b$, after two time steps initialised at $X_0 = x_0$, we have

$$X_{2\delta t} \approx x_0 + a\delta t + b\epsilon_{\delta t} + a\delta t + b\epsilon_{2\delta t} = x_0 + 2a\delta t + b\tilde{\epsilon}_{2\delta t},$$

where $\tilde{\epsilon}_{2\delta t} \sim \mathsf{N}(0, 2\delta t)$, since the two noise terms $\epsilon_{\delta t}$ and $\epsilon_{2\delta t}$ are independent. However, the right-hand side of this expression is exactly of the same form we would get from a single time step of size $2\delta t$ to obtain $X_{2\delta t}$ from X_0.

The process with $a = 0$, $b = 1$ and $X_0 = 0$ consists of a sequence of mean-zero Gaussian increments, each with a variance of δt. This is a discretisation of a process known as *Brownian motion*, which is often denoted by W_t. In particular, we have that

$$\delta W_t = W_{t+\delta t} - W_t \sim \mathsf{N}(0, \delta t),$$

and $W_t \sim \mathsf{N}(0, t)$. From the definition of W_t, we may rewrite the noisy update as

$$\delta X_t \approx a(X_t, t)\delta t + b(X_t, t)\delta W_t. \tag{1.16}$$

Consider this process on some interval $[0, T]$, with $\delta t = T/m$ and $X_0 = x_0$, for some initial value x_0. Subject to some regularity conditions, the limit as $m \to \infty$ exists and is written

$$\mathrm{d}X_t = a(X_t, t)\mathrm{d}t + b(X_t, t)\mathrm{d}W_t.$$

This is known as a SDE, and (1.16) is the Euler–Maruyama approximation to it. Subject to the initial condition, the *solution* to this SDE is the stochastic process $\{X_t\}_{t\in[0,T]}$ obtained from the limit $\delta t \to 0$ of the discrete-time process defined through (1.16).

The above-mentioned heuristic describes a one-dimensional SDE and its Euler–Maruyama discretisation; however, it is straightforward to extend these to higher dimensions with $\mathbf{X}_t \in \mathbb{R}^d$, $\mathbf{a}\colon \mathbb{R}^d \times [0, \infty) \to \mathbb{R}^d$, $\mathbf{W}_t \in \mathbb{R}^k$ and the $d \times k$ matrix $\mathbf{b}\colon \mathbb{R}^d \times [0, \infty) \to \mathbb{R}^{dk}$.

A stochastic process that satisfies an SDE is called a *diffusion*. For the most part, we will deal with time-homogeneous diffusions, where a and b have no explicit time dependence; however, time-inhomogeneous diffusions will be used in Chapter 3.

1.4.1 The Ornstein–Uhlenbeck Process

Consider the SDE

$$\mathrm{d}X_t = -\frac{1}{2\sigma^2}b^2 X_t \mathrm{d}t + b\mathrm{d}W_t.$$

The Euler–Maruyama discretisation gives

$$X_{t+\delta t} \approx X_t + \delta X_t = \left(1 - \frac{b^2}{2\sigma^2}\delta t\right) X_t + b\delta W_t.$$

Since δW_t is Gaussian distributed and independent of X_t, if X_t is Gaussian so is $X_{t+\delta t}$. Moreover, if $\mathbb{E}[X_t] = 0$, then $\mathbb{E}[X_{t+\delta t}] = 0$. Finally, if $\mathrm{Var}[X_t] = \sigma^2$, then

$$\mathrm{Var}[X_{t+\delta t}] = \left(1 - \frac{b^2}{2\sigma^2}\delta t\right)^2 \sigma^2 + b^2\delta t = \sigma^2 + \frac{1}{4\sigma^4} b^4 \delta t^2.$$

In the limit $m \to \infty$, as the number of increments is increased, with $\delta t = T/m \downarrow 0$, the term in δt^2 becomes irrelevant: the variance does not change. Thus, if $X_0 \sim \mathsf{N}(0, \sigma^2)$, then $X_t \sim \mathsf{N}(0, \sigma^2)$ for all $t > 0$; the SDE is stationary. Shifting the coordinate system by m, we see that the slightly more general SDE

$$\mathrm{d}X_t = -\frac{1}{2\sigma^2} b^2 (X_t - m)\mathrm{d}t + b\mathrm{d}W_t \tag{1.17}$$

has a stationary distribution of $\mathsf{N}(m, \sigma^2)$. The process arising from the SDE (1.17) is known as the OU process. Substituting $s = b^2 t$, the SDE becomes $\mathrm{d}X_s = -(X_s - m)/(2\sigma^2)\mathrm{d}s + \mathrm{d}W_s$, which explains why b^2 is termed the *speed* of the diffusion. Figure 1.5 presents three realisations of OU processes with stationary distribution $\mathsf{N}(m, 1)$ and started from the corresponding $m/2$. Each diffusion has a different speed, and the effect of this on the convergence to, and mixing within, the stationary distribution is clearly visible.

1.4.2 The Infinitesimal Generator

The *infinitesimal generator* (or, simply, *generator*) of a continuous-time stochastic process acts on a function h of the process

$$(\mathcal{L}h)(\mathbf{x}) := \frac{\partial}{\partial t}\mathbb{E}[h(\mathbf{X}_t)|\mathbf{X}_0 = \mathbf{x}]\bigg|_{t=0} = \lim_{\delta t \downarrow 0} \frac{\mathbb{E}[h(\mathbf{X}_{\delta t})] - h(\mathbf{x})}{\delta t}. \tag{1.18}$$

The set of functions for which the limit exists for all $\mathbf{x}$ is called the *domain* of the infinitesimal generator. Subject to regularity conditions, this includes the set of compactly supported functions with a second derivative that is continuous, denoted C_0^2.

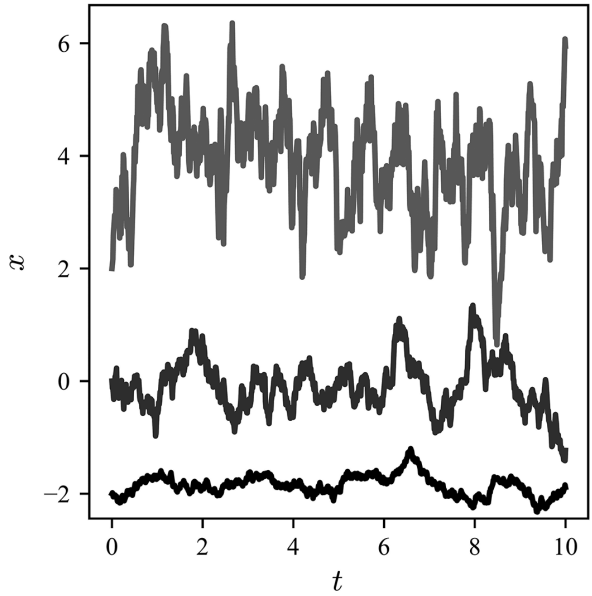

Figure 1.5 Three realisations of the OU processes, all with $\sigma = 1$, and on the time interval $[0, 10]$. Other parameter settings are $x_0 = 2$, $m = 4$, and $b = 3$; $x_0 = m = 0$ and $b = 1$; $x_0 = -2$, $m = -4$, and $b = 1/3$.

For processes defined by an SDE, we can gain some insight into their generator by considering a Taylor expansion. For simplicity of presentation, we consider $x \in \mathbb{R}$:

$$\frac{1}{\delta t}\mathbb{E}\left[h(X_{\delta t}) - h(x)\right] = \frac{1}{\delta t}\mathbb{E}\left[(X_{\delta t} - x)h'(x) + \frac{1}{2}(X_{\delta t} - x)^2 h''(x) + \ldots\right].$$

The Euler–Maruyama approximation of the SDE is $X_{\delta t} - x \approx a(x)\delta t + b(x)\delta W_t$. Thus, $\mathbb{E}\left[X_{\delta t} - x\right] \approx a(x)\delta t$ and $\mathbb{E}\left[(X_{\delta t} - x)^2\right] \approx b(x)^2\delta t + a(x)^2[\delta t]^2$, with all higher-order expectations at most $o(\delta t)$. Thus, we might expect that

$$(\mathcal{L}h)(x) = a(x)\frac{\mathrm{d}h}{\mathrm{d}x} + \frac{1}{2}b(x)^2\frac{\mathrm{d}^2h}{\mathrm{d}x^2},$$

and this is indeed the case. For a multivariate diffusion, the generator is

$$(\mathcal{L}h)(\mathbf{x}) = \sum_{i=1}^{d} a_i \left.\frac{\partial h}{\partial x_i}\right|_{\mathbf{x}} + \frac{1}{2}\sum_{i=1}^{d}\sum_{j=1}^{d} (bb^\top)_{i,j} \left.\frac{\partial^2 h}{\partial x_i \partial x_j}\right|_{\mathbf{x}}. \tag{1.19}$$

Generators of diffusion processes are used in the next subsection to derive the stationary density of two classes of diffusion that appear repeatedly in Chapters 2 and 3. Generators of diffusions are also used in Chapter 6 for the assessment and improvement of algorithms. Finally, Chapter 5 employs the generators of another class of continuous-time stochastic processes to determine the processes' stationary distributions.

1.4.3 Langevin Diffusions

We now describe two classes of diffusion, the overdamped and underdamped Langevin diffusions, where the stationary density forms an explicit part of the SDE formulation.

The Overdamped Langevin Diffusion

Consider a positive, differentiable density function $f(\mathbf{x})$ for $\mathbf{x} \in \mathbb{R}^d$, and the following SDE:

$$\mathrm{d}\mathbf{X}_t = \frac{1}{2}\nabla \log f(\mathbf{X}_t) b^2 \mathrm{d}t + b\mathrm{d}\mathbf{W}_t. \tag{1.20}$$

A solution to this SDE is known as an *overdamped Langevin diffusion*. The OU process (1.17) with $f(x) = \mathsf{N}(x; m, 1/a)$ is a special case of this class of diffusions, and in this case, as seen in Section 1.4.1, f is the density of the stationary process. In fact, this is true in general: the stationary density of the overdamped Langevin diffusion (1.20) is f. To see this in one dimension, consider the infinitesimal generator of the diffusion

$$(\mathcal{L}h)(x) = \frac{1}{2}b^2 \frac{f'(x)}{f(x)} h'(x) + \frac{1}{2}b^2 h''(x).$$

This is the rate of change of the expectation of $h(X_t)$ at $t = 0$, when started from $X_0 = x$. Suppose instead that X_0 has a density of f. Then, the rate of change of the expectation of X_t at $t = 0$ can be calculated by taking expectations with respect to X_0. This is

$$\frac{1}{2}b^2 \int \left\{ \frac{f'(x)}{f(x)} h'(x) + h''(x) \right\} f(x)\, \mathrm{d}x = \frac{1}{2}b^2 \int \{f(x)h'(x)\}'\, \mathrm{d}x = 0$$

for all sufficiently smooth h with compact support. Thus, if $X_0 \sim f$, $\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}\left[h(X_t)\right]\big|_{t=0} = 0$. This is true for *all* $h \in C_0^2$, and so the distribution of X_t does not change as t increases from 0. The distribution at time 0 must, therefore, be the stationary distribution of the Langevin diffusion (1.20), and f is the corresponding stationary density.

When Langevin diffusions are employed in a Bayesian setting, $f(\mathbf{x})$ is often a posterior density whose normalising constant is, typically, intractable.

The fact that the calculation of $\nabla \log f(\mathbf{X}_t)$ does not require this normalising constant is crucial to the practical use of these diffusions.

The Underdamped Langevin Diffusion

The *underdamped Langevin diffusion* extends the state space to include a velocity component, $\mathbf{P}_t$:

$$\mathrm{d}\mathbf{X}_t = \mathbf{P}_t \mathrm{d}t, \tag{1.21}$$

$$\mathrm{d}\mathbf{P}_t = -\gamma \mathbf{P}_t \mathrm{d}t + c\nabla \log f(\mathbf{X}_t)\mathrm{d}t + \sqrt{2\gamma c}\, \mathrm{d}\mathbf{W}_t. \tag{1.22}$$

Intuitively, dividing (1.22) through by γ and taking the limit as $\gamma \to \infty$ and $c \to \infty$ with $c/\gamma = b^2/2$ fixed, we obtain the overdamped Langevin diffusion, so the latter is a limiting case of the underdamped diffusion.

The underdamped Langevin diffusion targets $f(\mathbf{x})g(\mathbf{p})$, where

$$g(\mathbf{p}) = \frac{1}{\sqrt{2\pi c}} \exp\left(-\frac{1}{2c}\|\mathbf{p}\|^2\right).$$

To see this, we, again, restrict ourselves to the one-dimensional case to simplify the presentation, and, again, we start from the generator

$$(\mathcal{L}h)(x,p) = ph_x(x,p) - \gamma p h_p(x,p) + c\frac{f'(x)}{f(x)} h_p(x,p) + \gamma c h_{p,p}(x,p),$$

where we have used subscripts to denote the differentiation of h with respect to x or p. The quantity $(\mathcal{L}h)(x,p)$ is the rate of change of the expectation of $h(X_t, P_t)$ at $t = 0$, when started at $X_0 = x$ and $P_0 = p$. Thus if X_0 and P_0 have respective densities of $f(x)$ and $g(p)$, then the rate of change of $\mathbb{E}\left[h(X_t, P_t)\right]$ at $t = 0$ is

$$\iint \left\{ph_x - \gamma p h_p + c\frac{f'(x)}{f(x)} h_p + \gamma c h_{p,p}\right\} f(x)g(p)\mathrm{d}p\mathrm{d}x.$$

In the manipulations that follow, we will twice use the fact that $g'(p) = -pg(p)/c$. First integration by parts gives

$$\int \gamma c h_{p,p} g(p)\, \mathrm{d}p = \int p\gamma h_p g(p)\, \mathrm{d}p,$$

so the second and fourth terms cancel. Second two integrations by parts, first with respect to p and then with respect to x, give

$$\begin{aligned}\iint c f'(x) h_p g(p)\, \mathrm{d}p\mathrm{d}x &= \iint f'(x) hpg(p)\, \mathrm{d}p\mathrm{d}x \\ &= -\iint f(x) h_x p g(p)\, \mathrm{d}p\mathrm{d}x,\end{aligned}$$

so the first and third terms cancel. The argument is completed analogously to that for the overdamped Langevin diffusion.

In Chapter 3, we explore further the overdamped and underdamped Langevin diffusions as practical algorithms for scalable Monte Carlo inference in the large-data setting and show that the discretisation of these diffusion processes leads to important special cases of the general framework for stochastic gradient MCMC algorithms.

1.5 The Kernel Trick

Chapter 6 introduces the *kernel Stein discrepancy* and uses it to measure the discrepancy between a sample of points and a distribution of interest. Practical use of the methodology is made feasible by the ability to reduce what appears to be an infinite amount of computation – maximising a quantity over an uncountably infinite set of possible functions – to only a finite number of arithmetic operations. The key mechanism for this simplification is often called *the kernel trick*, and the setting for its use is a *reproducing kernel Hilbert space*.

This section first explains the kernel trick in the more familiar setting of a finite-dimensional inner-product space, before extending to the more general setting required for Chapter 6. Whilst many of the concepts introduced are much more general, our presentation focuses on the specific setting of relevance: the vectors of our inner-product space are functions, the associated field is $\mathbb{R}$, and the inner product is an integral with respect to a probability distribution.

Throughout, $f(\cdot)$, $g(\cdot)$, etc., are functions from $\mathcal{X} \to \mathbb{R}$, where $\mathcal{X}$ is $\mathbb{R}^d$ or some closed or open subset of $\mathbb{R}^d$; $f(\mathbf{x})$, $g(\mathbf{x})$ *etc* denote the function evaluated at $\mathbf{x} \in \mathcal{X}$. The probability distribution ν is assumed to have a density $\nu(\mathbf{x})$ on $\mathcal{X}$.

1.5.1 Finite-Dimensional Inner-Product Space

Let $0(\cdot)$ be the function such that $0(\mathbf{x}) = 0$ for all $\mathbf{x} \in \mathcal{X}$. A set, $\mathbb{V}$, of functions from $\mathcal{X} \to \mathbb{R}$ is a *vector space* over $\mathbb{R}$ if the following axioms are satisfied:

1. $0(\cdot) \in \mathbb{V}$.
2. $f(\cdot) \in \mathbb{V} \implies -f(\cdot) \in \mathbb{V}$.
3. $f(\cdot), g(\cdot) \in \mathbb{V} \implies f(\cdot) + g(\cdot) \in \mathbb{V}$.
4. $f(\cdot) \in \mathbb{V}$ and $a \in \mathbb{R} \implies af(\cdot) \in \mathbb{V}$.

Aside: The associativity, commutativity, and distributativity axioms of a general vector space are satisfied automatically when the elements are functions and from $\mathcal{X}$ to $\mathbb{R}$ and the field is $\mathbb{R}$.

Every finite-dimensional vector space has a dimension, n, such that there is a set of n vectors $\{b_1(\cdot), \ldots, b_n(\cdot)\}$, which satisfy the following two properties:

1. **Linear independence**: If there are $a_1, \ldots, a_n \in \mathbb{R}$ such that $\sum_{i=1}^n a_i b_i(\cdot) = 0$, then $a_i = 0$ for all $i \in \{1, \ldots, n\}$.
2. **Spanning** $\mathbb{V}$: for each $f(\cdot) \in \mathbb{V}$, there are $a_1, \ldots, a_n \in \mathbb{R}$ such that $f(\cdot) = \sum_{i=1}^n a_i b_i(\cdot)$.

The set $\{b_1(\cdot), \ldots, b_n(\cdot)\}$ is called a *basis*.

Example 1.7. It is straightforward to check that the set

$$\begin{aligned}\mathbb{V} &= \{f(\cdot) : f(x) = c\sin(x+\theta) : c \in \mathbb{R}, \theta \in [0, 2\pi)\} \\ &= \{f(\cdot) : f(x) = a\sin x + b\cos x; a, b \in \mathbb{R}\}\end{aligned}$$

satisfies Axioms 1–4, whatever the domain, $\mathcal{X} \subseteq \mathbb{R}$. We may take $b_1(\cdot) = \sin(\cdot)$ and $b_2(\cdot) = \cos(\cdot)$. However, we may also take $b_1(\cdot) = \sin(\cdot)+3\cos(\cdot)$ and $b_2(\cdot) = \cos(\cdot)$, for example.

For any vector space $\mathbb{V}$ of functions from $\mathcal{X} \to \mathbb{R}$ and any distribution ν with a probability density function on $\mathcal{X}$ of $\nu(x)$, we define the *inner product*

$$\langle f(\cdot), g(\cdot)\rangle_\nu = \int f(\mathbf{x})g(\mathbf{x})\nu(\mathbf{x})\,\mathrm{d}\mathbf{x}, \tag{1.23}$$

where here and throughout this section, if the integral range is not specified, then it is $\mathcal{X}$. We refer to this inner product as $\langle\cdot,\cdot\rangle_\nu$.

The inner product defined by (1.23) clearly satisfies two of the three defining properties of an inner product: $\langle f(\cdot), g(\cdot)\rangle = \langle g(\cdot), f(\cdot)\rangle$ and $\langle f(\cdot) + g(\cdot), h(\cdot)\rangle = \langle f(\cdot), h(\cdot)\rangle + \langle g(\cdot), h(\cdot)\rangle$. However, we have only that $\langle f(\cdot), f(\cdot)\rangle = 0 \Leftrightarrow f(\mathbf{x}) = 0(\mathbf{x})$ ν-almost everywhere, rather than $\langle f(\cdot), f(\cdot)\rangle = 0 \Leftrightarrow f(\cdot) = 0(\cdot)$. Each f belongs to an equivalence class of functions that are equal ν-almost everywhere. This set of equivalence classes forms a vector space, and (1.23) defines an inner product on this space, not on the space of functions, $\mathbb{V}$. To keep the presentation in this section as straightforward as possible, our wording ignores this distinction, but the more rigorous reader may wish to replace any vector space of functions and inner product between these functions with the corresponding vector space of equivalence classes of functions and inner products between these equivalence classes.

The inner product provides a norm, called the *induced norm*, the square of which is

$$\|f(\cdot)\|_\nu^2 = \langle f(\cdot), f(\cdot)\rangle_\nu = \int f(\mathbf{x})^2 \nu(\mathbf{x})\, d\mathbf{x}.$$

Example 1.8 (Example 1.7 continued). Let $\mathcal{X} = [0, 2\pi]$, and let ν be the uniform distribution on $[0, 2\pi]$. For any $f(\cdot), g(\cdot) \in \mathbb{V}$,

$$\langle f(\cdot), g(\cdot)\rangle_\nu = \frac{1}{2\pi}\int_0^{2\pi} f(x)g(x)\, dx \quad \text{and} \quad \|f(\cdot)\|_\nu^2 = \frac{1}{2\pi}\int_0^{2\pi} f(x)^2\, dx.$$

Example 1.9. For a general vector space $\mathbb{V}$ of functions of the form $\mathcal{X} \to \mathbb{R}$, let $\mathbb{V}_\nu$ be the elements of $\mathbb{V}$ that have a finite norm induced by ν:

$$\mathbb{V}_\nu = \left\{ f(\cdot) \in \mathbb{V} : \int f(\mathbf{x})^2 \nu(\mathbf{x})\, d\mathbf{x} < \infty \right\}.$$

Then $\mathbb{V}_\nu$ is also a vector space, since Axioms 1, 2 and 4 are satisfied trivially, and Axiom 3 is satisfied since for any $f(\cdot), g(\cdot) \in \mathbb{V}_\nu$,

$$\begin{aligned}\|f(\cdot) + g(\cdot)\|_\nu^2 &= \langle f(\cdot) + g(\cdot), f(\cdot) + g(\cdot)\rangle_\nu \\ &= \|f(\cdot)\|_\nu^2 + 2\langle f(\cdot), g(\cdot)\rangle_\nu + \|g(\cdot)\|_\nu^2 \\ &\le \|f(\cdot)\|_\nu^2 + 2\|f(\cdot)\|_\nu\|g(\cdot)\|_\nu + \|g(\cdot)\|_\nu^2 < \infty,\end{aligned}$$

where the third line uses the Cauchy–Schwarz inequality, which, in this case, is the familiar inequality $\mathbb{E}\left[f(\mathbf{X})g(\mathbf{X})\right]^2 \le \mathbb{E}[f(\mathbf{X})^2]\mathbb{E}\left[g(\mathbf{X})^2\right]$, where $\mathbf{X}$ has a density ν on $\mathcal{X}$.

Henceforth, for narrative simplicity, we will assume that $\mathbb{V}$ is a finite-dimensional vector space with dimension n. Section 1.5.4 extends the narrative to potentially infinite-dimensional spaces.

When considering the inner product $\langle \cdot, \cdot \rangle_\nu$, two vectors $f(\cdot), g(\cdot) \in \mathbb{V}$ are said to be *orthogonal* if $\langle f(\cdot), g(\cdot)\rangle_\nu = 0$, and the basis vectors, $e_1(\cdot), \ldots, e_n(\cdot)$, are said to be *orthonormal* if they are orthogonal and each has a norm of 1: for each $j, k \in \{1, \ldots, n\}$,

$$\|e_j(\cdot)\|_\nu = 1 \quad \text{and} \quad \langle e_j(\cdot), e_k(\cdot)\rangle_\nu = 0,$$

whenever $j \ne k$. We will reserve the symbols $\{e_k(\cdot)\}_{k=1}^n$ for any set of n orthonormal basis functions.

The representation of $f(\cdot)$ in terms of an orthonormal basis

$$f(\cdot) = \sum_{j=1}^n f_j e_j(\cdot)$$

is termed an *orthonormal decomposition* of $f(\cdot)$. Since the $e_i(\cdot)$ are orthonormal, the projection of $f(\cdot)$ onto $e_k(\cdot)$ is f_k:

$$\begin{aligned}\langle f(\cdot), e_k(\cdot)\rangle_\nu &= \left\langle \sum_{j=1}^n f_j e_j(\cdot), e_k(\cdot)\right\rangle_\nu = \sum_{j=1}^n f_j \langle e_j(\cdot), e_k(\cdot)\rangle_\nu \\ &= f_k. \end{aligned} \tag{1.24}$$

Furthermore, the squared norm of $f(\cdot)$ is the sum of the squares of the orthonormal projections

$$\begin{aligned}\|f(\cdot)\|^2 &= \left\langle \sum_{j=1}^n f_j e_j(\cdot), \sum_{k=1}^n f_k e_k(\cdot)\right\rangle = \sum_{j=1}^n \sum_{k=1}^n f_j f_k \langle e_j(\cdot), e_k(\cdot)\rangle \\ &= \sum_{j=1}^n f_j^2. \end{aligned} \tag{1.25}$$

Example 1.10. In Example 1.7, since $\int_0^{2\pi} \sin^2 x \, \mathrm{d}x = \int_0^{2\pi} \cos^2 x \, \mathrm{d}x = \pi$ and $\int_0^{2\pi} \sin x \cos x \, \mathrm{d}x = 0$,

$$e_1(\cdot) = \sqrt{2}\sin(\cdot) \quad \text{and} \quad e_2(\cdot) = \sqrt{2}\cos(\cdot)$$

form an orthonormal basis for $\mathbb{V}$ when ν is the uniform distribution on $[0, 2\pi]$. Any function $f(\cdot) \in \mathbb{V}$ can be written as $f(\cdot) = f_1 e_1(\cdot) + f_2 e_2(\cdot)$. For example set

$$f(x) = \sin(x + \pi/6) = \frac{\sqrt{3}}{2}\sin x + \frac{1}{2}\cos x. \tag{1.26}$$

So $f_1 = \sqrt{3}/(2\sqrt{2})$ and $f_2 = 1/(2\sqrt{2})$. Also,

$$\|f(\cdot)\|_\nu^2 = f_1^2 + f_2^2 = \frac{3}{8} + \frac{1}{8} = \frac{1}{2} = \frac{1}{2\pi}\int_0^{2\pi} \sin^2(x + \pi/6) \, \mathrm{d}x.$$

1.5.2 Kernels in a Finite-Dimensional Inner-Product Space

As in the previous subsection, let $\mathbb{V}$ be an n-dimensional vector space of functions from $\mathcal{X}$ to $\mathbb{R}$, and let ν be a probability distribution on $\mathcal{X}$ with a probability density of $\nu(\mathbf{x})$, $\mathbf{x} \in \mathcal{X}$. Finally, let $\{e_k(\cdot)\}_{k=1}^n$ be a set of basis functions that is orthonormal with respect to the inner product (1.23).

Let $\lambda_1, \ldots, \lambda_n$ be a set of non-negative scalars, and consider the following real-valued function on $\mathcal{X} \times \mathcal{X}$:

$$\mathsf{k}(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^n \lambda_j e_j(\mathbf{x}) e_j(\mathbf{y}). \tag{1.27}$$

Clearly, $\mathsf{k}(\cdot,\cdot)$ is *symmetric*: $\mathsf{k}(\mathbf{y},\mathbf{x}) = \mathsf{k}(\mathbf{x},\mathbf{y})$. Moreover, $\mathsf{k}(\cdot,\cdot)$ is *positive semidefinite*: for any finite $J < \infty$, $c_1, \ldots, c_J \in \mathbb{R}$ and $\mathbf{x}_1, \ldots \mathbf{x}_J \in \mathbb{R}^d$,

$$\begin{aligned}\sum_{j=1}^{J}\sum_{k=1}^{J} c_j c_k \mathsf{k}(\mathbf{x}_j,\mathbf{x}_k) &= \sum_{j=1}^{J}\sum_{k=1}^{J} c_j c_k \sum_{l=1}^{n} \lambda_l e_l(\mathbf{x}_j) e_l(\mathbf{x}_k) \\ &= \sum_{l=1}^{n} \lambda_l \sum_{j=1}^{J}\sum_{k=1}^{J} c_j c_k e_l(\mathbf{x}_j) e_l(\mathbf{x}_k) \\ &= \sum_{l=1}^{n} \lambda_l \left\{\sum_{j=1}^{J} c_j e_l(\mathbf{x}_j)\right\}^2 \geq 0.\end{aligned}$$

Any function $\mathsf{k}(\cdot,\cdot) : \mathcal{X}\times\mathcal{X} \to \mathbb{R}$, which is both symmetric and positive semidefinite, is called a *kernel*.

Example 1.11. Continuing Example 1.7, let $\mathsf{k}\colon [0,2\pi]\times[0,2\pi] \to \mathbb{R}$ be

$$\begin{aligned}\mathsf{k}(x,y) &= \frac{1}{2} e_1(x) e_1(y) + \frac{3}{2} e_2(x) e_2(y) = \sin x \sin y + 3\cos x \cos y \\ &= 2\cos(y-x) + \cos(y+x).\end{aligned}$$

This is symmetric and positive definite by construction.

Given the definition of $\mathsf{k}(\cdot,\cdot)$ in (1.27), define

$$\mathsf{k}(\mathbf{x},\cdot) = \sum_{j=1}^{n} \lambda_j e_j(\mathbf{x}) e_j(\cdot), \tag{1.28}$$

and $\mathsf{k}(\cdot,\mathbf{x}) = \mathsf{k}(\mathbf{x},\cdot)$. Since $e_j(\mathbf{x}) \in \mathbb{R}$, $\mathsf{k}(\mathbf{x},\cdot) \in \mathbb{V}$. Furthermore, for $f(\cdot) \in \mathbb{V}$, define the operator T_{k} via

$$T_{\mathsf{k}} f(\cdot) = \int \mathsf{k}(\cdot,\mathbf{y}) f(\mathbf{y}) \nu(\mathbf{y})\, d\mathbf{y}. \tag{1.29}$$

Then T_{k} is a *linear operator*, since for any $a, b \in \mathbb{R}$ and $f(\cdot), g(\cdot) \in \mathbb{V}$,

$$T_{\mathsf{k}}\{a f(\cdot) + b g(\cdot)\} = a T_{\mathsf{k}} f(\cdot) + b T_{\mathsf{k}} g(\cdot).$$

Now, writing $f(\cdot) = \sum_{k=1}^{n} f_k e_k(\cdot)$,

$$\begin{aligned}(T_{\mathsf{k}} f(\cdot))(\mathbf{x}) &= \int \mathsf{k}(\mathbf{x}, \mathbf{y}) f(\mathbf{y}) \nu(\mathbf{y}) \mathrm{d}\mathbf{y} \\ &= \langle \mathsf{k}(\mathbf{x}, \cdot), f(\cdot) \rangle_{\nu} = \left\langle \sum_{j=1}^{n} \lambda_j e_j(\mathbf{x}) e_j(\cdot), \sum_{k=1}^{n} f_k e_k(\cdot) \right\rangle_{\nu} \\ &= \sum_{j=1}^{n} \sum_{k=1}^{n} \lambda_j e_j(\mathbf{x}) f_k \langle e_j(\cdot), e_k(\cdot) \rangle_{\nu} = \sum_{k=1}^{n} \lambda_k f_k e_k(\mathbf{x}).\end{aligned}$$

So $T_{\mathsf{k}} f(\cdot) = \sum_{k=1}^{n} \lambda_k f_k e_k(\cdot)$ and, hence, $T_{\mathsf{k}} f(\cdot) \in \mathbb{V}$, too. Moreover, considering $f(\cdot) = e_j(\cdot)$, we see that $T_{\mathsf{k}} e_j(\cdot) = \lambda_j e_j(\cdot)$; each $e_j(\cdot)$ is an eigenfunction of T_{k} with a corresponding eigenvalue of λ_j.

Example 1.12. Continuing Example 1.7, with the kernel from Example 1.11,

$$\mathsf{k}(x, \cdot) = \sin x \sin(\cdot) + 3 \cos x \cos(\cdot) = 2 \cos(\cdot - x) + \cos(\cdot + x).$$

Let $f(\cdot)$ be as defined in (1.26). Then, using the definite integrals at the start of Example 1.10,

$$\begin{aligned}T_{\mathsf{k}} f(\cdot) &= \frac{1}{2\pi} \int_0^{2\pi} \{\sin(\cdot) \sin y + 3 \cos(\cdot) \cos y\} \left\{ \frac{\sqrt{3}}{2} \sin y + \frac{1}{2} \cos y \right\} \mathrm{d}y \\ &= \frac{1}{4\pi} \int_0^{2\pi} \sqrt{3} \sin(\cdot) \sin^2 y + 3 \cos(\cdot) \cos^2 y \, \mathrm{d}y \\ &= \frac{1}{4} \left\{ \sqrt{3} \sin(\cdot) + 3 \cos(\cdot) \right\}.\end{aligned}$$

Since

$$e_1(\cdot) = \sqrt{2} \sin(\cdot), \; e_2(\cdot) = \sqrt{2} \cos(\cdot), \; f_1 = \frac{\sqrt{3}}{2\sqrt{2}}, \; f_2 = \frac{1}{2\sqrt{2}},$$

$\lambda_1 = 1/2$ and $\lambda_2 = 3/2$, $T_{\mathsf{k}} f(\cdot)$ is, therefore,

$$\lambda_1 f_1 e_1(\cdot) + \lambda_2 f_2 e_2(\cdot),$$

as we would hope.

1.5.3 A New Inner Product and the Kernel Trick in Finite Dimensions

Let $\{e_j(\cdot)\}_{j=1}^n$ be an orthonormal basis for $\mathbb{V}$, and let k be defined through (1.27) with respect to this basis. For $f(\cdot), g(\cdot) \in \mathbb{V}$ with

$$f(\cdot) = \sum_{j=1}^{n} f_j e_j(\cdot) \quad \text{and} \quad g(\cdot) = \sum_{j=1}^{n} g_j e_j(\cdot), \tag{1.30}$$

the inner product with respect to ν is the sum of the products of the orthogonal projections

$$\begin{aligned}\langle f(\cdot), g(\cdot)\rangle_\nu &= \left\langle \sum_{j=1}^{n} f_j e_j(\cdot), \sum_{k=1}^{n} g_k e_k(\cdot) \right\rangle_\nu = \sum_{j=1}^{n}\sum_{k=1}^{n} f_j g_k \langle e_j(\cdot), e_k(\cdot)\rangle_\nu \\ &= \sum_{j=1}^{n} f_j g_j.\end{aligned}$$

We now define a new inner product

$$\langle f(\cdot), g(\cdot)\rangle_{\mathsf{k}} = \sum_{j=1}^{n} \frac{f_j g_j}{\lambda_j}, \tag{1.31}$$

where the $\{\lambda_j\}_{j=1}^n$ are exactly those from the definition of k and are the eigenvalues of the operator T_{k}.

This inner product may be rephrased in terms of a set of eigenfunctions that are orthonormal with respect to $\langle\cdot,\cdot\rangle_{\mathsf{k}}$: $\{e'_j(\cdot)\}_{j=1}^n$ with $e'_j(\cdot) = \sqrt{\lambda_j} e_j(\cdot)$. With respect to this basis, the vector

$$f(\cdot) = \sum_{j=1}^{n} f'_j e'_j(\cdot),$$

with $f'_j = f_j/\sqrt{\lambda_j}$. Using an analogous decomposition for $g(\cdot)$,

$$\langle f(\cdot), g(\cdot)\rangle_{\mathsf{k}} = \sum_{j=1}^{n} f'_j g'_j,$$

as expected. Finally,

$$\mathsf{k}(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^{n} e'_j(\mathbf{x}) e'_j(\mathbf{y}).$$

Example 1.13. In Example 1.11, $e_1'(\cdot) = \sin(\cdot)$ and $e_2'(\cdot) = \sqrt{3}\cos(\cdot)$. Clearly,

$$\mathsf{k}(x, y) = \sin x \sin y + 3 \cos x \cos y = e_1'(x)e_1'(y) + e_2'(x)e_2'(y).$$

For $f(\cdot)$, as in Example 1.26,

$$f(\cdot) = \frac{\sqrt{3}}{2}\sin(\cdot) + \frac{1}{2}\cos(\cdot) = \frac{\sqrt{3}}{2}\sin(\cdot) + \frac{\sqrt{3}}{6}\sqrt{3}\cos(\cdot),$$

so $f_1' = \sqrt{3}/2$ and $f_2' = \sqrt{3}/6$. Thus,

$$\|f(\cdot)\|_{\mathsf{k}}^2 = \frac{3}{4} + \frac{3}{36} = \frac{5}{6}.$$

The Kernel Trick

From the definition (1.28) and with $f(\cdot)$ decomposed as in (1.30),

$$\begin{aligned}\langle \mathsf{k}(\mathbf{x}, \cdot), f(\cdot)\rangle_{\mathsf{k}} &= \left\langle \sum_{j=1}^{n} \lambda_j e_j(\mathbf{x}) e_j(\cdot), \sum_{k=1}^{n} f_k e_k(\cdot) \right\rangle_{\mathsf{k}} = \sum_{j=1}^{n} \frac{\lambda_j e_j(\mathbf{x}) f_j}{\lambda_j} \\ &= f(\mathbf{x}). \end{aligned} \tag{1.32}$$

Moreover, choosing $f(\cdot)$ to be $\mathsf{k}(\mathbf{y}, \cdot)$, $f_j = \lambda_j e_j(\mathbf{y})$ from (1.28), and, hence,

$$\langle \mathsf{k}(\mathbf{x}, \cdot), \mathsf{k}(\mathbf{y}, \cdot)\rangle_{\mathsf{k}} = \sum_{j=1}^{n} \lambda_j e_j(\mathbf{x}) e_j(\mathbf{y}) = \mathsf{k}(\mathbf{x}, \mathbf{y}). \tag{1.33}$$

Together, (1.32) and (1.33) enable the evaluation of inner products in $\langle \cdot, \cdot \rangle_{\mathsf{k}}$ *without needing to know the original basis functions* $e_1(\cdot), \ldots, e_n(\cdot)$ *nor the associated values* $\lambda_1, \ldots, \lambda_n$. Indeed, we do not even need to know ν. This is known as *the kernel trick*, and we will exemplify its use in Section 1.5.5. First, we generalise to a much broader class of kernels.

1.5.4 General Kernels

In Section 1.5.2, we created a kernel via (1.27) using a known orthonormal basis for the inner-product space $\mathbb{V}$, with the inner product specified by (1.23) according to the density ν. However, a kernel is *any* positive-definite symmetric function and we are interested in kernels $\mathsf{k}\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

Example 1.14. The Gaussian kernel is

$$\mathsf{k}(\mathbf{x}, \mathbf{y}) = \exp\left(-\|\mathbf{y} - \mathbf{x}\|^2\right),$$

where $\|\cdot\|$ represents the standard Euclidean norm. This is clearly symmetric. To see that k is also positive semidefinite on $\mathcal{X} = \mathbb{R}^d$, note that

$$\mathbf{Z} \sim \mathsf{N}_d\left(\mathbf{x}, \frac{1}{4}\mathbf{I}_d\right) \quad \text{and} \quad \mathbf{Y}|\mathbf{Z} \sim \mathsf{N}_d\left(\mathbf{Z}, \frac{1}{4}\mathbf{I}_d\right) \implies \mathbf{Y} \sim \mathsf{N}_d\left(\mathbf{x}, \frac{1}{2}\mathbf{I}_d\right),$$

from which

$$\exp\left(-\|\mathbf{y}-\mathbf{x}\|^2\right) = \gamma \int \exp\left(-2\|\mathbf{y}-\mathbf{z}\|^2\right) \exp\left(-2\|\mathbf{x}-\mathbf{z}\|^2\right) \mathrm{d}\mathbf{z},$$

where $\gamma = 2^d/\pi^{d/2}$. Hence, $\sum_{j=1}^J \sum_{k=1}^J c_j c_k \mathsf{k}(\mathbf{x}_j, \mathbf{x}_k)$ is

$$\begin{aligned}
&\gamma \sum_{j=1}^J \sum_{k=1}^J c_j c_k \int \exp\left(-2\|\mathbf{x}_j-\mathbf{z}\|^2\right) \exp\left(-2\|\mathbf{x}_k-\mathbf{z}\|^2\right) \mathrm{d}\mathbf{z} \\
&\qquad = \gamma \int \sum_{j=1}^J \sum_{k=1}^J c_j c_k \exp\left(-2\|\mathbf{x}_j-\mathbf{z}\|^2\right) \exp\left(-2\|\mathbf{x}_k-\mathbf{z}\|^2\right) \mathrm{d}\mathbf{z} \\
&\qquad = \gamma \int \left\{\sum_{j=1}^J c_j \exp\left(-2\|\mathbf{x}_j-\mathbf{z}\|^2\right)\right\}^2 \mathrm{d}\mathbf{z} \\
&\qquad \geq 0.
\end{aligned}$$

When specifying k in Example 1.14, we have not specified a vector space, nor a density ν, nor an associated inner product. However, since k is a kernel, we might hope that if we do specify ν and the inner product $\langle\cdot,\cdot\rangle_\nu$ in (1.23), then there might be a vector space with a basis that is orthonormal with respect to $\langle\cdot,\cdot\rangle_\nu$ such that k has the decomposition (1.27). If this were the case, then we would know that there was a new inner product $\langle\cdot,\cdot\rangle_{\mathsf{k}}$ such that (1.32) and (1.33) held. Hence, we could evaluate inner products with respect to k without knowing the basis itself nor the eigenvalues of T_{k}, nor, even, the details about ν.

The decomposition in (1.23) does not hold in general, but Mercer's theorem and generalisations of it tell us that an analogous decomposition, but with n potentially infinite, holds widely.

Specifically, let $\mathcal{X}$ be $\mathbb{R}^d$ or a closed or open subset of $\mathbb{R}^d$, $\mathsf{k}(\cdot,\cdot)$: $\mathcal{X}\times\mathcal{X} \to \mathbb{R}$ be a kernel and $\nu(\mathbf{x})$, $\mathbf{x} \in \mathcal{X}$, be a probability density on $\mathcal{X}$. Then, provided $\mathsf{k}(\mathbf{x},\mathbf{y})$ is a continuous function of $\mathbf{x}$ and $\mathbf{y}$, and

$$\int \mathsf{k}(\mathbf{x},\mathbf{y})^2 \nu(\mathbf{y})\, \mathrm{d}\mathbf{y} < \infty \quad \text{for all } \mathbf{x} \in \mathcal{X}, \tag{1.34}$$

the linear operator T_{k} defined in (1.29) has at most countably many positive (and no negative) eigenvalues $\lambda_1, \lambda_2, \ldots$ with corresponding eigenfunctions

$e_1(\cdot), e_2(\cdot), \ldots$, which are orthonormal with respect to the inner product $\langle \cdot, \cdot \rangle_\nu$ defined in (1.23). Furthermore, k can be decomposed as

$$\mathsf{k}(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^{\infty} \lambda_k e_j(\mathbf{x}) e_j(\mathbf{y}),$$

and the set $\{\sqrt{\lambda_j} e_j(\cdot)\}_{j=1}^{\infty}$ forms an orthonormal basis with respect to the inner product

$$\langle f(\cdot), g(\cdot) \rangle_{\mathsf{k}} = \sum_{j=1}^{\infty} \frac{f_j g_j}{\lambda_j},$$

where

$$f(\cdot) = \sum_{j=1}^{\infty} f_j e_j(\cdot) \quad \text{and} \quad g(\cdot) = \sum_{j=1}^{\infty} g_j e_j(\cdot). \tag{1.35}$$

The space in which $e_1(\cdot), e_2(\cdot), \ldots$ lie is a generalisation of the vector space of Example 1.9 to the *Hilbert space*, $\mathcal{H}_\nu$, of functions $f(\cdot) : \mathcal{X} \to \mathbb{R}$ with the inner product $\langle \cdot, \cdot \rangle_\nu$ and such that $\|f(\cdot)\|_\nu^2 = \int f(\mathbf{x})^2 \nu(\mathbf{x}) \, \mathrm{d}\mathbf{x} < \infty$. Likewise, the orthonormal basis $\{\sqrt{\lambda_j} e_j(\cdot)\}_{j=1}^{\infty}$ lies in the *reproducing kernel Hilbert space*, $\mathcal{H}_\mathsf{k}$, of functions with the inner product $\langle \cdot, \cdot \rangle_{\mathsf{k}}$ and such that $\|f(\cdot)\|_{\mathsf{k}} < \infty$. A *Hilbert space* $\mathcal{H}$ is an inner-product space with a potentially infinite set of basis vectors that is *complete*; informally, it contains no 'holes', so that for any sequence $f_1, f_2, \ldots$ with $\sum_{j=1}^{\infty} f_j^2 < \infty$ then (e.g. considering $\mathcal{H}_\mathsf{k}$) $f(\cdot) = \lim_{n \to \infty} \sum_{j=1}^{n} f_j e'_j(\cdot)$ exists, with distance measured through the norm induced by the inner product, and is in $\mathcal{H}_\mathsf{k}$.

Thus, the simplifications of the inner products in (1.32) and (1.33) continue to hold; in general, the intermediate steps must replace n with ∞.

Example 1.15. For the Gaussian kernel of Example 1.14, $\mathsf{k}(\mathbf{x}, \cdot) = \exp(-\|\mathbf{x} - \cdot\|^2)$ and

$$\left\langle \exp(-\|\mathbf{x} - \cdot\|^2), \exp(-\|\mathbf{y} - \cdot\|^2) \right\rangle_{\mathsf{k}} = \exp(-\|\mathbf{y} - \mathbf{x}\|^2).$$

Also, for any $f(\cdot) \in \mathcal{H}_\mathsf{k}$,

$$\left\langle \exp(-\|\mathbf{x} - \cdot\|^2), f(\cdot) \right\rangle_{\mathsf{k}} = f(\mathbf{x}).$$

Trace-Class Kernels

A kernel where $\int \mathsf{k}(\mathbf{x}, \mathbf{x}) \nu(\mathbf{x}) \, \mathrm{d}\mathbf{x} = c < \infty$ is referred to as *trace class*. This property has important consequences for the set of eigenvalues, $\lambda_1, \lambda_2, \ldots$, of T_k, since

$$\int \mathsf{k}(\mathbf{x},\mathbf{x})\nu(\mathbf{x})\,\mathrm{d}\mathbf{x} = \int \sum_{k=1}^{\infty} \lambda_k e_k(\mathbf{x}) e_k(\mathbf{x}) \nu(\mathbf{x})\,\mathrm{d}\mathbf{x}$$
$$= \sum_{k=1}^{\infty} \lambda_k \int e_k(\mathbf{x})^2 \nu(\mathbf{x})\,\mathrm{d}\mathbf{x} = \sum_{k=1}^{\infty} \lambda_k.$$

Thus, $\sum_{k=1}^{\infty} \lambda_k = c$. Since each $\lambda_k \geq 0$, we have $\lim_{k\to\infty} \lambda_k = 0$.

The Gaussian kernel of Example 1.14 is a trace class with $c = 1$ since $\mathsf{k}(\mathbf{x},\mathbf{x}) = 1$ for all $\mathbf{x} \in \mathbb{R}^d$. The kernel we will meet in in Chapter 6 is also of trace class, following a similar reasoning.

Without loss of generality, we label the eigenvalues $\lambda_1, \lambda_2 \ldots$ in order of decreasing size (choosing any one of the possibilities if some of the λ_j are not unique). With the decomposition of $f(\cdot)$ in (1.35),

$$\|f(\cdot)\|_{\mathsf{k}}^2 = \sum_{j=1}^{\infty} \frac{f_j^2}{\lambda_j} \geq \frac{1}{\lambda_1} \sum_{j=1}^{\infty} f_j^2 = \frac{1}{\lambda_1} \|f(\cdot)\|_{\nu}^2.$$

Thus, $\|f(\cdot)\|_{\mathsf{k}} < \infty \implies \|f(\cdot)\|_{\nu} < \infty$ and hence $\mathcal{H}_{\mathsf{k}} \subseteq \mathcal{H}_{\nu}$. In general, $\mathcal{H}_{\mathsf{k}}$ is strictly smaller than $\mathcal{H}_{\nu}$, and the more quickly the eigenvalues of T_{k} decay, the smaller the space $\mathcal{H}_{\mathsf{k}}$.

1.5.5 The Power of the Kernel Trick

Suppose we have values $\mathbf{x}_1, \ldots, \mathbf{x}_m \in \mathcal{X}$, and we are interested in

$$\mathbb{V}^* = \left\{ g(\cdot) : g(\cdot) = \sum_{j=1}^{m} g_j \mathsf{k}(\mathbf{x}_j, \cdot),\ g_1, \ldots, g_m \in \mathbb{R} \right\}.$$

First for any $g(\cdot) = \sum_{j=1}^{m} g_j \mathsf{k}(\mathbf{x}_j, \cdot)$,

$$\|g(\cdot)\|_{\mathsf{k}}^2 = \left\langle \sum_{j=1}^{m} g_j \mathsf{k}(\mathbf{x}_j, \cdot), \sum_{k=1}^{m} g_k \mathsf{k}(\mathbf{x}_k, \cdot) \right\rangle_{\mathsf{k}} = \sum_{j=1}^{m} \sum_{k=1}^{m} g_j \left\langle \mathsf{k}(\mathbf{x}_j, \cdot), \mathsf{k}(\mathbf{x}_k, \cdot) \right\rangle g_k$$
$$= \sum_{j=1}^{m} \sum_{k=1}^{m} g_j \mathsf{k}(\mathbf{x}_j, \mathbf{x}_k) g_k < \infty. \tag{1.36}$$

So, $\mathbb{V}^* \subseteq \mathcal{H}_{\mathsf{k}}$. Second for any $f(\cdot) \in \mathcal{H}_{\mathsf{k}}$,

$$\langle f(\cdot), g(\cdot) \rangle_{\mathsf{k}} = \sum_{j=1}^{m} g_j \left\langle f(\cdot), \mathsf{k}(\mathbf{x}_j, \cdot) \right\rangle_{\mathsf{k}} = \sum_{j=1}^{m} g_j f(\mathbf{x}_j). \tag{1.37}$$

Suppose there is a particular function of interest, $f(\cdot) \in \mathcal{H}_{\mathsf{k}}$, and we would like to construct the function $g(\cdot) \in \mathbb{V}^*$ that most closely resembles

$f(\cdot)$ in shape. We could find the unit vector in $\mathbb{V}^*$ that has the largest component in the $f(\cdot)$ direction

$$\widehat{g}(\cdot) = \underset{g(\cdot)\in\mathbb{V}^*:\|g(\cdot)\|=1}{\arg\max} \langle f(\cdot), g(\cdot)\rangle_{\mathsf{k}}\,.$$

The size of the inner product, $\langle f(\cdot), \widehat{g}(\cdot)\rangle_{\mathsf{k}}$, is a measure of the ability of $\mathbb{V}^*$ to represent $f(\cdot)$.

Define $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_m)]^\top$ and $\mathbf{g} = [g_1, \ldots g_m]^\top$, and let $\mathbf{K}$ be the matrix with elements $K_{i,j} = \mathsf{k}(\mathbf{x}_i, \mathbf{x}_j)$. Then, (1.36) and (1.37) become

$$\langle f(\cdot), g(\cdot)\rangle_{\mathsf{k}} = \mathbf{g}^\top\mathbf{f} \quad \text{and} \quad \|g(\cdot)\|_{\mathsf{k}}^2 = \mathbf{g}^\top\mathbf{K}\mathbf{g}.$$

To find $\widehat{g}(\cdot)$, we must find the vector $\widehat{\mathbf{g}}$ that maximises $\mathbf{g}^\top\mathbf{f}$ subject to $\mathbf{g}^\top\mathbf{K}\mathbf{g} = 1$.

Let $\mathbf{A}$ be a square matrix such that $\mathbf{A}\mathbf{A}^\top = \mathbf{K}$, and set $\mathbf{h} = \mathbf{A}^\top\mathbf{g}$. Then, equivalently, we wish to maximise $\mathbf{h}^\top\mathbf{A}^{-1}\mathbf{f}$ such that $\|\mathbf{h}\| = 1$. We must find the unit m-vector with the largest component in the $\mathbf{A}^{-1}\mathbf{f}$ direction, which is

$$\widehat{\mathbf{h}} = \frac{\mathbf{A}^{-1}\mathbf{f}}{\sqrt{(\mathbf{A}^{-1}\mathbf{f})^\top\mathbf{A}^{-1}\mathbf{f}}} \implies \widehat{\mathbf{g}} = \frac{\mathbf{A}^{-\top}\mathbf{A}^{-1}\mathbf{f}}{\sqrt{\mathbf{f}^\top\mathbf{A}^{-\top}\mathbf{A}^{-1}\mathbf{f}}} = \frac{\mathbf{K}^{-1}\mathbf{f}}{\sqrt{\mathbf{f}^\top\mathbf{K}^{-1}\mathbf{f}}},$$

since $\widehat{\mathbf{g}} = \mathbf{A}^{-\top}\widehat{\mathbf{h}}$. The inner product $\widehat{\mathbf{g}}^\top\mathbf{f}$ is

$$\frac{\mathbf{f}^\top\mathbf{K}^{-1}\mathbf{f}}{\sqrt{\mathbf{f}^\top\mathbf{K}^{-1}\mathbf{f}}} = \sqrt{\mathbf{f}^\top\mathbf{K}^{-1}\mathbf{f}}.$$

This calculation *only* requires us to be able to evaluate $f(\mathbf{x}_j)$ and $\mathsf{k}(\mathbf{x}_j, \mathbf{x}_k)$ for $j, k = 1, \ldots, m$. We do not need to know the eigenfunctions $e_1(\cdot), \ldots$ nor eigenvalues $\lambda_1, \ldots$ of T_{k}. Indeed, we do not even need to know ν; only that (1.34) is satisfied.

Example 1.16. Let $\mathcal{X} = \mathbb{R}$ and let k be the one-dimensional case of the Gaussian kernel in Example 1.14. We find the approximations to the function

$$f(x) = \frac{1}{1+x^2},$$

using gradually more and more kernel functions $\mathsf{k}(x_j, x)$. For points, $x_1, \ldots, x_J$, $\mathbf{K}$ is the matrix with elements $K_{i,j} = \exp[-(x_i - x_j)^2]$, and $\mathbf{f}$ is the vector with $f_j = f(x_j)$. We set $(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = (-3, \ldots, 3)$ and approximate $f(x)$ using just x_1 then $x_1, \ldots, x_3$, then $x_1, \ldots, x_5$ and finally $x_1, \ldots, x_7$. Figure 1.6 compares the four approximations with the truth. Each time new points are added to the set, the approximation improves, but

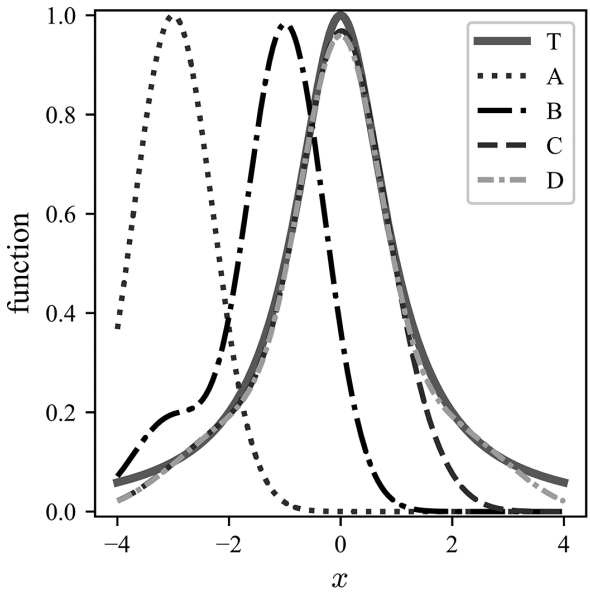

Figure 1.6 The function $f(x) = 1/(1+x^2)$ (T) and kernel-based approximations to $f(x)$ from Example 1.16. Curves use A: $x = -3$, B: $x = -3, -2, -1$, C: $x = -3, \ldots, 1$, and D: $x = -3, \ldots, 3$.

it matters where the points are added; some basis vectors are more helpful than others.

1.6 Chapter Notes

There are many texts that cover the introductory material from this chapter in more depth and rigour than we have allowed; we suggest a few on each topic.

Basic Monte Carlo and importance sampling are covered in Ripley (2009) and Rubinstein and Kroese (2008). For an introduction to Bayesian statistics and the use of Monte Carlo methods for Bayesian analysis, see Bernardo and Smith (2009), Robert (2007) and Robert and Casella (1999).

Norris (1998) provides a gentle introduction to Markov chains on discrete state spaces, while Meyn and Tweedie (2012) give a thorough treatment on general state spaces; a less thorough but more readily accessible treatment for general state spaces is given by Roberts and Rosenthal (2004). Geyer

(1992) describes methods for estimating the integrated auto-correlation time from a sample of the chain when the Markov chain is reversible; for the non-reversible chains of Chapter 4, the integrated auto-correlation can be estimated by fitting an auto-regressive process to the time series $\{h(X_k)\}_{k=1}^n$ or by estimating the spectral density of the series at a frequency of 0 (e.g. Heidelberger & Welch, 1981).

Diffusions and SDEs are the subject of Oksendal (2013), Rogers and Williams (2000a) and Rogers and Williams (2000b). An alternative to simple Monte Carlo, which attempts to obtain better convergence rates with the Monte Carlo sample size n, is quasi-Monte Carlo. See, for example, Caflisch (1998) for an introduction and L'Ecuyer and Lemieux (2002) for work on randomised quasi-Monte Carlo.

Chapter 1 of Conway (2010) introduces Hilbert spaces in general, and kernels and reproducing kernel Hilbert spaces are covered in Chapter 6 of Rasmussen and Williams (2005). Mercer's theorem is usually stated for a compact $\mathcal{X}$; we have used the generalisation to non-compact spaces in Sun (2005).