11

Parameter Estimation in SDE Models

An issue that often arises in the context of practical modeling with SDEs is the problem of parameter estimation. In that context, we might know the parametric form of the SDE, but the parameters of the SDE have unknown values. However, we might have a set of experimental data that we wish to use for determining the values of the parameters. The aim of this chapter is to give an overview of solutions to these kinds of problems. We specifically consider statistical likelihood-based inference methods, but we also give pointers to other types of methods.

The problem of parameter estimation in SDE models has a long history, and overviews of parameter estimation methods in different types of models can be found, for example, in the articles of Nielsen et al. (2000), Sørensen (2004), and Särkkä et al. (2015b) and in the theses of Jeisman (2005) and Mbalawata (2014) as well as in the books of Rao (1999) and Iacus (2008). In the case of partially observed systems, the problem is closely related to parameter estimation in (discrete-time) state-space models (or hidden Markov models), and parameter estimation methods for these kinds of models have been summarized in the books of Särkkä (2013) and Cappé et al. (2005).

11.1 Overview of Parameter Estimation Methods

In the typical setting that we consider here, we have an SDE with a vector of (unknown) parameters $\boldsymbol{\theta} = \big(\theta_1, \theta_2, \ldots, \theta_d\big)$:

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t; \boldsymbol{\theta})\, \mathrm{d}t + \mathbf{L}(\mathbf{x}, t; \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\beta}, \quad \mathbf{x}(t_0) = \mathbf{x}_0. \tag{11.1}$$

The diffusion matrix of the Brownian motion $\mathbf{Q}(\boldsymbol{\theta})$ might also depend on the parameters. Additionally, we have a set of observations of the SDE. For example, we might have a set of known values of the state $\mathbf{x}(t)$ at a certain finite number of time points. Alternatively, we might only have partial observations of the state, and these observations might also be corrupted by noise.

Example 11.1 (Parameters in the Ornstein–Uhlenbeck model)**.** *Consider, for example, the Ornstein–Uhlenbeck process*

$$\mathrm{d}x = -\lambda\, x\, \mathrm{d}t + \mathrm{d}\beta, \quad x(0) = x_0. \tag{11.2}$$

For the sake of parameter estimation, we can assume that the parameters λ and q are unknown, that is, we have the unknown parameter vector $\boldsymbol{\theta} = (\lambda, q)$. Furthermore, we could assume that we know the values of the SDE at times $x(\Delta t), x(2\,\Delta t), \ldots, x(T\,\Delta t)$ which forms the data that are used for estimating the parameters.

In the case that we observe a finite number of values of the SDE, say, $\mathbf{x}(t_1), \mathbf{x}(t_2), \ldots, \mathbf{x}(t_T)$, a classical method for SDE parameter estimation is the maximum likelihood (ML) method. Due to the Markov properties of SDEs (cf. Section 5.4), we can write down the likelihood of the observed values given the parameters as follows:

$$p(\mathbf{x}(t_1), \ldots, \mathbf{x}(t_T) \mid \boldsymbol{\theta}) = \prod_{k=0}^{T-1} p(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta}), \tag{11.3}$$

where $p(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta})$ are the transition densities of the SDE that we discussed in Section 5.4. In the ML method, we wish to maximize the preceding likelihood expression or, equivalently, minimize the negative log-likelihood:

$$\begin{aligned}\ell(\boldsymbol{\theta}) &= -\log p(\mathbf{x}(t_1), \ldots, \mathbf{x}(t_T) \mid \boldsymbol{\theta}) \\ &= -\sum_{k=0}^{T} \log p(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta}).\end{aligned} \tag{11.4}$$

Alternately, given the likelihood we can use Bayesian methods that directly perform inference on the posterior distribution,

$$\begin{aligned}p(\boldsymbol{\theta} \mid \mathbf{x}(t_1), \ldots, \mathbf{x}(t_T)) &= \frac{p(\mathbf{x}(t_1), \ldots, \mathbf{x}(t_T) \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}{p(\mathbf{x}(t_1), \ldots, \mathbf{x}(t_T))} \\ &\propto p(\boldsymbol{\theta}) \prod_{k=0}^{T-1} p(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta}),\end{aligned} \tag{11.5}$$

where $p(\boldsymbol{\theta})$ is the prior distribution and $\propto$ denotes a constant proportionality. Typical Bayesian methods include maximum a posteriori (MAP) estimation, Laplace approximations, Markov chain Monte Carlo, and other Monte Carlo methods.

In order to use ML methods or Bayesian methods, we need to be able to evaluate the likelihood, which in general is hard. This is because the likelihood depends on the transition densities, which are solutions to the Fokker–Planck–Kolmogorov (FPK) equation (see Section 5.4) and are thus hard to compute. If we know the transition densities, then we can explicitly evaluate the likelihood, which is indeed the case for linear SDEs. Parameter estimation in these kinds of models is considered in Section 11.3.

In the case of general multivariate nonlinear SDEs, we cannot solve the FPK and thus the transition density is unknown. In that case, the typical approach is to replace the SDE or its transition density in the likelihood with a tractable approximation. We can, for example, use the various SDE discretization methods that we have already covered (Itô–Taylor, stochastic Runge–Kutta, linearization) for forming a discrete-time SDE approximation whose transition density we can evaluate. Another way is to directly approximate the transition density of the SDE using Gaussian approximations, Hermite expansions, or other approximations that we saw in the previous chapters.

In this chapter, we also discuss partially and noisily observed SDE models. In those models, we do not observe the values $\mathbf{x}(t_1), \mathbf{x}(t_2), \ldots, \mathbf{x}(t_T)$ directly, but instead we only observe noisy versions of them, such as $\mathbf{y}_k = \mathbf{H}\,\mathbf{x}(t_k) + \mathbf{r}_k$, where $\mathbf{H}$ is some (possibly singular) matrix and $\mathbf{r}_k \sim \mathrm{N}(0, \mathbf{R})$ is a Gaussian noise. More generally, as in the previous chapter, the observations might come from a conditional distribution $p(\mathbf{y}_k \mid \mathbf{x}(t_k))$. In these models, we cannot use the previous likelihood expressions, but instead we need to compute the marginal likelihood of the measurements given the parameters $p(\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_T \mid \boldsymbol{\theta})$ or the corresponding posterior distributions $p(\boldsymbol{\theta} \mid \mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_T)$ at least up to an unknown constant factor. In order to do parameter estimation in these models, we need to use the filtering and smoothing methods from the previous chapter as parts of the parameter estimation methods for evaluating the marginal likelihood or posterior distribution.

11.2 Computational Methods for Parameter Estimation

As discussed in the previous section, provided that we can evaluate the likelihood $p(\mathbf{x}(t_1), \mathbf{x}(t_2), \ldots, \mathbf{x}(t_T) \mid \boldsymbol{\theta})$ or its negative logarithm $\ell(\boldsymbol{\theta}) =$

$-\log p(\mathbf{x}(t_1), \mathbf{x}(t_2), \ldots, \mathbf{x}(t_T) \mid \boldsymbol{\theta})$, we have a wide range of off-the-shelf computational methods that we can use for estimating the parameters $\boldsymbol{\theta}$. The aim of this section is to discuss a couple of them, optimization-based methods and Markov chain Monte Carlo (MCMC) methods. For more details and more advanced computational methods, the reader is referred to the books of Luenberger and Ye (2008), Liu (2001), Brooks et al. (2011), and Gelman et al. (2013), along with Särkkä (2013).

The simplest approach to estimate parameters is to maximize the likelihood of the measured values with respect to the parameter values. This can be seen as a method that finds the parameters that best fit the data. The method can be written in algorithmic form as follows.

Algorithm 11.2 (ML estimate)**.** *The maximum likelihood estimate of SDE parameters can be obtained by finding the vector of parameters* $\boldsymbol{\theta}_{\mathrm{ML}}$ *that minimizes the negative log-likelihood* $\ell(\boldsymbol{\theta})$ *defined in Equation* (11.4)*:*

$$\boldsymbol{\theta}_{\mathrm{ML}} = \arg\min_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta}). \tag{11.6}$$

In practice, the minimum can be computed either analytically by setting derivatives to zero or by using numerical optimization methods (see, e.g., Luenberger and Ye, 2008).

In the Bayesian setting, we might also have a prior distribution $p(\boldsymbol{\theta})$, which restricts or weights the possible parameter values. Then we can define the unnormalized negative log-posterior by using the following:

$$\ell_p(\boldsymbol{\theta}) = \ell(\boldsymbol{\theta}) - \log p(\boldsymbol{\theta}). \tag{11.7}$$

With this prior information, the ML estimate is generalized to the maximum a posteriori estimate shown in Algorithm 11.3.

Algorithm 11.3 (MAP estimate)**.** *The MAP estimate* $\boldsymbol{\theta}_{\mathrm{MAP}}$ *of the SDE parameters can be found by minimizing the unnormalized negative log-posterior defined in Equation* (11.7)*:*

$$\boldsymbol{\theta}_{\mathrm{MAP}} = \arg\min_{\boldsymbol{\theta}} \ell_p(\boldsymbol{\theta}). \tag{11.8}$$

A completely different class of algorithms for likelihood-based inference are MCMC methods. These kinds of methods are particularly commonly used in Bayesian analysis (Gelman et al., 2013; Särkkä, 2013), but MCMC methods have a multitude of other applications as well (Liu, 2001; Brooks et al., 2011).

The aim of MCMC in the SDE context is to generate samples from the posterior distribution (11.5). That is, instead of summarizing the posterior distribution via its single maximum (the MAP estimate), we generate a set of samples from the distribution. These samples can then be used for computing the best parameter estimates (e.g., the posterior mean) as well as uncertainty in the parameter estimates (e.g., the posterior covariance).

The most common MCMC method is the Metropolis–Hastings algorithm in Algorithm 11.4. In order to implement it, we only need to be able to evaluate the unnormalized negative log-posterior (11.7).

Algorithm 11.4 (Metropolis–Hastings)**.** *The Metropolis–Hastings (MH) algorithm for generating samples from a distribution* $p(\boldsymbol{\theta}) \propto \exp(-\ell_p(\boldsymbol{\theta}))$ *is the following.*

1. *Draw the starting point,* $\boldsymbol{\theta}^{(0)}$ *from an arbitrary initial distribution.*
2. *For each iteration* $i = 1, 2, \dots, N$*, do the following steps:*

   a. *Sample a candidate point* $\boldsymbol{\theta}^*$ *from the proposal distribution:*

$$\boldsymbol{\theta}^* \sim q(\boldsymbol{\theta}^* \mid \boldsymbol{\theta}^{(i-1)}). \tag{11.9}$$

   b. *Evaluate the acceptance probability:*

$$\alpha_i = \min\left\{1, \exp(\ell_p(\boldsymbol{\theta}^{(i-1)}) - \ell_p(\boldsymbol{\theta}^*))\, \frac{q(\boldsymbol{\theta}^{(i-1)} \mid \boldsymbol{\theta}^*)}{q(\boldsymbol{\theta}^* \mid \boldsymbol{\theta}^{(i-1)})}\right\}. \tag{11.10}$$

   c. *Generate a uniform random variable* $u \sim \mathrm{U}(0,1)$ *and set the following:*

$$\boldsymbol{\theta}^{(i)} = \begin{cases} \boldsymbol{\theta}^*, & \text{if } u \le \alpha_i, \\ \boldsymbol{\theta}^{(i-1)}, & \text{otherwise.} \end{cases} \tag{11.11}$$

The Metropolis algorithm is a commonly used special case of MH, where the proposal distribution is symmetric, $q(\boldsymbol{\theta}^{(i-1)} \mid \boldsymbol{\theta}^{(i)}) = q(\boldsymbol{\theta}^{(i)} \mid \boldsymbol{\theta}^{(i-1)})$. In this case, the acceptance probability reduces to the following:

$$\alpha_i = \min\left\{1, \exp(\ell_p(\boldsymbol{\theta}^{(i-1)}) - \ell_p(\boldsymbol{\theta}^*))\right\}. \tag{11.12}$$

The MH algorithm basically has a single design parameter, the proposal distribution $q(\boldsymbol{\theta}^* \mid \boldsymbol{\theta})$. However, this distribution almost completely defines the algorithm operation, and by selecting it in specific ways we get different brands of MCMC methods. For details, the reader is referred to Brooks et al. (2011).

11.3 Parameter Estimation in Linear SDE Models

In this section, the aim is to consider ML and Bayesian inference in linear SDEs. What makes linear SDEs special is that their transition densities are Gaussian and hence can be efficiently evaluated. In certain simple cases, we can also compute the ML estimates (or MAP estimates) of the parameters in closed form. However, more generally we need to resort to the computational methods outlined in the previous section.

Let us start by a considering the Ornstein–Uhlenbeck process which we already saw in Example 11.1:

$$\mathrm{d}x = -\lambda\, x\, \mathrm{d}t + \mathrm{d}\beta, \quad x(0) = x_0, \tag{11.13}$$

where λ is unknown and β has an unknown diffusion constant q. The vector of unknown parameters is thus $\boldsymbol{\theta} = (\lambda, q)$, and we assume that we have observed the SDE trajectory at $x(\Delta t), x(2\,\Delta t), \ldots, x(T\,\Delta t)$.

The transition density of the SDE is now given as (recall Example 6.2)

$$\begin{aligned} &p(x(t+\Delta t) \mid x(t)) \\ &= \mathrm{N}\left(x(t+\Delta t) \mid \exp(-\lambda\,\Delta t)\, x(t), \frac{q}{2\,\lambda}\,[1-\exp(-2\,\lambda\,\Delta t)]\right). \end{aligned} \tag{11.14}$$

Thus the negative log-likelihood can be written as

$$\begin{aligned} \ell(\lambda, q) = \sum_{k=0}^{T-1} \Bigg[&\frac{1}{2} \log\left(2\pi\, \frac{q}{2\,\lambda}\,[1-\exp(-2\,\lambda\,\Delta t)]\right) \\ &+ \frac{\lambda}{q\,[1-\exp(-2\,\lambda\,\Delta t)]}\,(x(t_{k+1}) - \exp(-\lambda\,\Delta t)\, x(t_k))^2 \Bigg]. \end{aligned} \tag{11.15}$$

However, for practical computation of the ML estimate it is more convenient to reparametrize the negative log-likelihood in terms of

$$\begin{aligned} a &= \exp(-\lambda\,\Delta t), \\ \Sigma &= \frac{q}{2\,\lambda}\,[1-\exp(-2\,\lambda\,\Delta t)], \end{aligned} \tag{11.16}$$

which thus gives

$$\ell(a, \Sigma) = \sum_{k=0}^{T-1} \left[\frac{1}{2}\, \log(2\pi\,\Sigma) + \frac{1}{2\Sigma}\,(x(t_{k+1}) - a\, x(t_k))^2 \right]. \tag{11.17}$$

Setting derivatives with respect to a and Σ to zero then gives

$$\begin{aligned} a_{\mathrm{ML}} &= \frac{\sum_{k=0}^{T-1} x(t_k)\, x(t_{k+1})}{\sum_{k=0}^{T-1} x(t_k)\, x(t_k)}, \\ \Sigma_{\mathrm{ML}} &= \frac{1}{T} \sum_{k=0}^{T-1} (x(t_{k+1}) - a_{\mathrm{ML}}\, x(t_k))^2, \end{aligned} \tag{11.18}$$

which in terms of original λ and q gives the final ML estimates of the parameters:

$$\begin{aligned} \lambda_{\mathrm{ML}} &= -\frac{1}{\Delta t} \log \left[\frac{\sum_{k=0}^{T-1} x(t_k)\, x(t_{k+1})}{\sum_{k=0}^{T-1} x(t_k)\, x(t_k)} \right], \\ q_{\mathrm{ML}} &= \frac{1}{T} \left(\frac{2\, \lambda_{\mathrm{ML}}}{1 - \exp(-2\, \lambda_{\mathrm{ML}}\, \Delta t)} \right) \sum_{k=0}^{T-1} (x(t_{k+1}) - \exp(-\lambda_{\mathrm{ML}}\, \Delta t)\, x(t_k))^2. \end{aligned} \tag{11.19}$$

We can also, in principle, do the similar inference for a more general LTI SDE

$$\mathrm{d}\mathbf{x} = \mathbf{F}(\boldsymbol{\theta})\, \mathbf{x}\, \mathrm{d}t + \mathbf{L}(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\beta}, \quad \mathbf{x}(0) = \mathbf{x}_0, \tag{11.20}$$

where the vector of Brownian motions $\boldsymbol{\beta}$ has the diffusion matrix $\mathbf{Q}(\boldsymbol{\theta})$. With sampling at $\mathbf{x}(\Delta t), \mathbf{x}(2\,\Delta t), \dots, \mathbf{x}(T\,\Delta t)$, we get the negative log-likelihood

$$\begin{aligned} \ell(\mathbf{A}, \boldsymbol{\Sigma}) = \sum_{k=0}^{T-1} \Bigg[& \frac{1}{2} \log |2\pi\, \boldsymbol{\Sigma}| \\ & + \frac{1}{2} (\mathbf{x}(t_{k+1}) - \mathbf{A}\, \mathbf{x}(t_k))^{\mathsf{T}}\, \boldsymbol{\Sigma}^{-1}\, (\mathbf{x}(t_{k+1}) - \mathbf{A}\, \mathbf{x}(t_k)) \Bigg], \end{aligned} \tag{11.21}$$

which we have already written in terms of

$$\begin{aligned} \mathbf{A} &= \exp(\mathbf{F}(\boldsymbol{\theta})\, \Delta t), \\ \boldsymbol{\Sigma} &= \int_0^{\Delta t} \exp(\mathbf{F}(\boldsymbol{\theta})\, (\Delta t - \tau))\, \mathbf{L}(\boldsymbol{\theta})\, \mathbf{Q}(\boldsymbol{\theta})\, \mathbf{L}^{\mathsf{T}}(\boldsymbol{\theta})\, \exp(\mathbf{F}(\boldsymbol{\theta})\, (\Delta t - \tau))^{\mathsf{T}}\, \mathrm{d}\tau. \end{aligned} \tag{11.22}$$

When we set the derivatives with respect to $\mathbf{A}$ and $\boldsymbol{\Sigma}$ to zero, we get

$$\begin{aligned} \boldsymbol{\Sigma}_{\mathrm{ML}} &= \frac{1}{T} \sum_{k=0}^{T-1} (\mathbf{x}(t_{k+1}) - \mathbf{A}\,\mathbf{x}(t_k))\,(\mathbf{x}(t_{k+1}) - \mathbf{A}\,\mathbf{x}(t_k))^{\mathsf{T}}, \\ \mathbf{A}_{\mathrm{ML}} &= \left(\sum_{k=0}^{T-1} \mathbf{x}(t_{k+1})\,\mathbf{x}^{\mathsf{T}}(t_k) \right) \left(\sum_{k=0}^{T-1} \mathbf{x}(t_k)\,\mathbf{x}^{\mathsf{T}}(t_k) \right)^{-1}, \end{aligned} \tag{11.23}$$

after which we still need to solve $\boldsymbol{\theta}$ from Equation (11.23). Unfortunately, this solution is rarely possible and seldom even exists when the parameters theta appear nontrivially in $\mathbf{A}$ and $\mathbf{Q}$.

In the preceding derivation for general $\mathbf{A}$ and $\mathbf{Q}$, we have failed to take into account that $\boldsymbol{\theta}$ might have significantly lower dimensionality than $\mathbf{A}$ and $\mathbf{Q}$ – thus it does not lead to the correct ML estimate unless we can uniquely solve the parameters given $\mathbf{A}$ and $\mathbf{Q}$. Furthermore, we often do not have a constant sampling period Δt, and the SDE might be time-varying – and we might also have an unknown offset function in the SDE.

For general linear SDEs of the general form

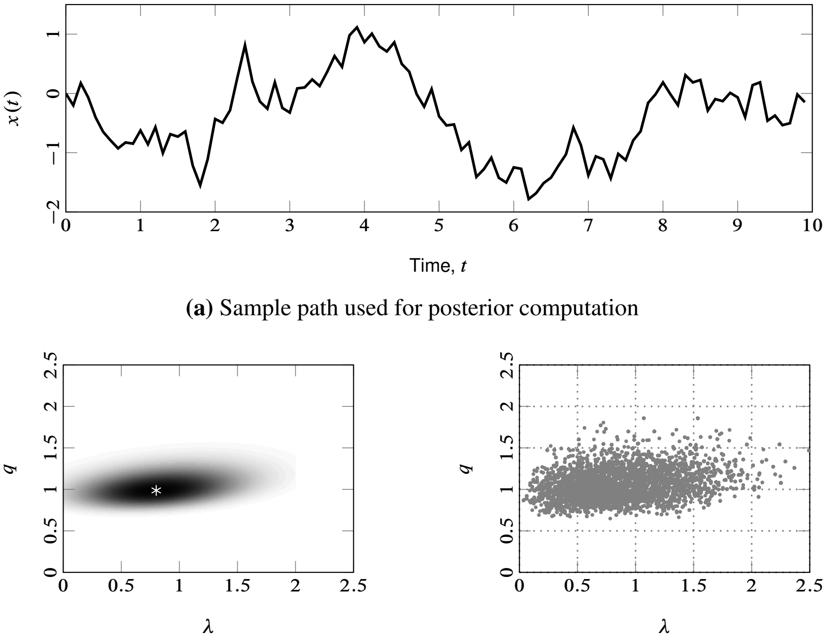
$$\mathrm{d}\mathbf{x} = \mathbf{F}(t;\boldsymbol{\theta})\,\mathbf{x}\,\mathrm{d}t + \mathbf{u}(t;\boldsymbol{\theta})\,\mathrm{d}t + \mathbf{L}(t;\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\beta}, \tag{11.24}$$

it is advisable to directly consider the negative log-likelihood, which is more generally given as

$$\begin{aligned} \ell(\boldsymbol{\theta}) = \sum_{k=0}^{T-1} \Bigg[&\frac{1}{2} \log |2\pi\, \boldsymbol{\Sigma}_k(\boldsymbol{\theta})| \\ &+ \frac{1}{2} (\mathbf{x}(t_{k+1}) - \mathbf{A}_k(\boldsymbol{\theta})\,\mathbf{x}(t_k) - \mathbf{u}_k(\boldsymbol{\theta}))^{\mathsf{T}}\, \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{\theta}) \\ &\times (\mathbf{x}(t_{k+1}) - \mathbf{A}_k(\boldsymbol{\theta})\,\mathbf{x}(t_k) - \mathbf{u}_k(\boldsymbol{\theta})) \Bigg], \end{aligned} \tag{11.25}$$

where $\mathbf{A}_k(\boldsymbol{\theta})$, $\mathbf{u}_k(\boldsymbol{\theta})$, and $\boldsymbol{\Sigma}_k(\boldsymbol{\theta})$ are given by Equations (6.9), (6.10), and (6.11), respectively. We can now numerically find the minimum of the negative log-likelihood by using numerical optimization methods (e.g., Luenberger and Ye, 2008). In order to do that, we also need to compute the derivatives with respect to the parameters, where the difficulty arises from deriving the partial derivatives $\partial \mathbf{A}_k(\boldsymbol{\theta})/\partial \theta_i$, $\partial \mathbf{u}_k(\boldsymbol{\theta})/\partial \theta_i$, and $\partial \boldsymbol{\Sigma}_k(\boldsymbol{\theta})/\partial \theta_i$. However, this can be done by using the matrix fraction decomposition (see, e.g., Mbalawata et al., 2013).

The negative log-likelihood expression in Equation (11.25) also allows for the use of Bayesian methods for parameter estimation. This is because



(a) Sample path used for posterior computation

(b) Posterior distribution

(c) MCMC samples from posterior

Figure 11.1 Illustration of the posterior computation for the Ornstein–Uhlenbeck model in Example 11.5. Subfigure (a) shows the simulated data (with $\lambda = 1/2$, $q = 1$), (b) shows the posterior distribution along with the ML/MAP estimates $\lambda_{\text{MAP}} = 0.80$ and $q_{\text{MAP}} = 0.99$, and (c) shows MCMC samples from the posterior generated with Metropolis–Hastings.

by Equation (11.7), the posterior distribution of the parameters can be written as

$$\begin{aligned} p(\boldsymbol{\theta} \mid \mathbf{x}(t_1), \mathbf{x}(t_2), \ldots, \mathbf{x}(t_T)) &\propto p(\boldsymbol{\theta}) \exp(-\ell(\boldsymbol{\theta})) \\ &= \exp(-\ell_p(\boldsymbol{\theta})), \end{aligned} \tag{11.26}$$

where $p(\boldsymbol{\theta})$ is the prior distribution. This unnormalized posterior distribution can now be plugged into various MCMC methods (see, e.g., Brooks et al., 2011), or we can compute MAP estimates (or Laplace approximations; see Gelman et al., 2013) by minimizing the negative logarithm of it by numerical optimization (e.g., Luenberger and Ye, 2008).

Example 11.5 (Exact parameter estimation in the Ornstein–Uhlenbeck model)**.** *The posterior distribution of the parameters λ and q of the*

Ornstein–Uhlenbeck model

$$\mathrm{d}x = -\lambda\, x\, \mathrm{d}t + \mathrm{d}\beta, \quad x(0) = 0, \tag{11.27}$$

using the exact negative log-likelihood (11.15) *and a uniform prior, is shown in Figure 11.1. The data used for the posterior distribution are shown as well. The data were generated using the parameter values* $\lambda = 1/2$, $q = 1$, *and the sampling period in the data was* $\Delta t = 1/10$ *with a total of* 100 *points.*

11.4 Approximated-Likelihood Methods

With nonlinear SDEs of the generic form given in Equation (11.1), we have an additional challenge that the evaluation of the transition density is intractable. This also makes parameter estimation harder, because we cannot evaluate the likelihood term in Equation (11.3) nor its negative logarithm in Equation (11.4).

In approximated-likelihood methods, we replace the likelihood, or more specifically the transition densities used for computing the likelihood, with approximations. One approach is to approximate the SDE with a continuous- or discrete-time system whose transition density we can evaluate. For that purpose, we can use the various SDE simulation and discretization methods that we discussed in Chapter 8.

For example, recall that one step of the Euler–Maruyama method in Algorithm 8.1 is

$$\hat{\mathbf{x}}(t_{k+1}) = \hat{\mathbf{x}}(t_k) + \mathbf{f}(\hat{\mathbf{x}}(t_k), t_k; \boldsymbol{\theta})\, \Delta t + \mathbf{L}(\hat{\mathbf{x}}(t_k), t_k; \boldsymbol{\theta})\, \Delta\boldsymbol{\beta}_k, \tag{11.28}$$

for which $\Delta\boldsymbol{\beta}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{Q}(\boldsymbol{\theta})\, \Delta t)$. It has a Gaussian transition density and hence leads to the approximation

$$\begin{aligned} p(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta}) & \\ \approx \mathrm{N}(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k) &+ \mathbf{f}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta})\, \Delta t, \\ &\mathbf{L}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta})\, \mathbf{Q}(\boldsymbol{\theta})\, \mathbf{L}^\mathsf{T}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta})\, \Delta t). \end{aligned} \tag{11.29}$$

The approximation to the likelihood in Equation (11.3) is then given as

$$\begin{aligned} p(\mathbf{x}(t_1), \ldots, \mathbf{x}(t_T) \mid \boldsymbol{\theta}) & \\ = \prod_{k=0}^{T-1} \mathrm{N}(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k) &+ \mathbf{f}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta})\, \Delta t, \\ &\mathbf{L}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta})\, \mathbf{Q}(\boldsymbol{\theta})\, \mathbf{L}^\mathsf{T}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta})\, \Delta t), \end{aligned} \tag{11.30}$$

and the approximation to the negative log-likelihood (11.4) will be

$$\begin{aligned}\ell(\boldsymbol{\theta}) = \sum_{k=0}^{T-1} \Bigg[&\frac{1}{2} \log |2\pi \, \mathbf{L}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta}) \, \mathbf{Q}(\boldsymbol{\theta}) \, \mathbf{L}^{\mathsf{T}}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta}) \, \Delta t| \\ &+ \frac{1}{2} (\mathbf{x}(t_{k+1}) - \mathbf{x}(t_k) - \mathbf{f}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta}) \, \Delta t)^{\mathsf{T}} \\ &\times (\mathbf{L}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta}) \, \mathbf{Q}(\boldsymbol{\theta}) \, \mathbf{L}^{\mathsf{T}}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta}) \, \Delta t)^{-1} \\ &\times (\mathbf{x}(t_{k+1}) - \mathbf{x}(t_k) - \mathbf{f}(\mathbf{x}(t_k), t_k; \boldsymbol{\theta}) \, \Delta t) \Bigg], \end{aligned} \tag{11.31}$$

which is going to be as (in)accurate as the Euler–Maruyama method is – thus we expect this approximation to work only with small Δt. Similarly, we can approximate the SDE with a step of the Milstein method or higher-order Itô–Taylor expansion-based methods. We can also use strong or weak Runge–Kutta and related methods. The only limitation is that we need to be able to evaluate the transition density corresponding to the discrete-time approximation. In algorithm form, we have the following.

Algorithm 11.6 (Discretization-based approximated-likelihood estimation)**.** *SDE discretization-based approximated-likelihood parameter estimation can be done as follows.*

1. *Use an SDE discretization such as Euler–Maruyama, Itô–Taylor expansions, or stochastic Runge–Kutta to form a discrete-time approximation to the SDE.*
2. *Let $\hat{p}(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta})$ be the transition density of the discrete-time approximation. Approximate the negative log-likelihood as*

$$\hat{\ell}(\boldsymbol{\theta}) = -\sum_{k=0}^{T-1} \log \hat{p}(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta}). \tag{11.32}$$

3. *Perform maximum likelihood estimation or any form of Bayesian estimation by replacing the exact likelihood ℓ with the preceding approximation $\hat{\ell}$.*

Instead of approximating the SDE as such, we can also approximate its transition density. One general approach to transition density approximation is to approximate is as Gaussian. For this, we can use, for example, the Gaussian and linearization approximations considered in Sections 9.1, 9.2, and 9.3 or alternatively using the Taylor series expansions for the first two moments as described in Section 9.4. Even the Euler–Maruyama happens to have this form, although higher-order Ito–Taylor based methods typically do not. Thus in the end we get a transition density approximation of the form

$$p(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta}) \approx \mathrm{N}(\mathbf{x}(t_{k+1}) \mid \boldsymbol{\mu}(\mathbf{x}(t_k), \Delta t; \boldsymbol{\theta}), \boldsymbol{\Sigma}(\mathbf{x}(t_k), \Delta t; \boldsymbol{\theta})), \tag{11.33}$$

which can be then further plugged into the likelihood expression. Thus we get the following algorithm.

Algorithm 11.7 (Gaussian approximated-likelihood parameter estimation)**.** *Gaussian approximation-based approximated-likelihood parameter estimation can be done as follows:*

1. *Use linearization, moment approximation, or any other methods to form the approximations to the mean $\boldsymbol{\mu}(\mathbf{x}(t_k), \Delta t; \boldsymbol{\theta})$ and covariance $\boldsymbol{\Sigma}(\mathbf{x}(t_k), \Delta t; \boldsymbol{\theta})$ of the Gaussian approximation on $p(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta})$.*
2. *Approximate the negative log likelihood as*

$$\begin{aligned}\hat{\ell}(\boldsymbol{\theta}) &= -\sum_{k=0}^{T-1} \log \mathrm{N}(\mathbf{x}(t_{k+1}) \mid \boldsymbol{\mu}(\mathbf{x}(t_k), \Delta t; \boldsymbol{\theta}), \boldsymbol{\Sigma}(\mathbf{x}(t_k), \Delta t; \boldsymbol{\theta})) \\ &= \sum_{k=0}^{T-1} \Bigg[\frac{1}{2} \log |2\pi \, \boldsymbol{\Sigma}(\mathbf{x}(t_k), \Delta t; \boldsymbol{\theta})| \\ &\quad + \frac{1}{2} (\mathbf{x}(t_{k+1}) - \boldsymbol{\mu}(\mathbf{x}(t_k), \Delta t; \boldsymbol{\theta}))^{\mathsf{T}} \, \boldsymbol{\Sigma}^{-1}(\mathbf{x}(t_k), \Delta t; \boldsymbol{\theta}) \\ &\quad \times (\mathbf{x}(t_{k+1}) - \boldsymbol{\mu}(\mathbf{x}(t_k), \Delta t; \boldsymbol{\theta})) \Bigg].\end{aligned} \tag{11.34}$$

3. *Perform maximum likelihood estimation or any form of Bayesian estimation by replacing the exact likelihood with the preceding approximation.*

Note that one way to approximate the mean and covariance of the preceding Gaussian approximation is as the conditional mean $\mathrm{E}[\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k)]$ and covariance $\mathrm{Cov}[\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k)]$ of the SDE, that is, by *moment matching*. However, this might not always lead to the best possible approximation on the likelihood as whole (cf. Archambeau and Opper, 2011; García-Fernández et al., 2017; Tronarp et al., 2018).

Not all transition densities can be approximated as Gaussian, for example, when they are multimodal. Then we can use non-Gaussian approximations such as the Hermite expansions considered in Section 9.5. We can also numerically solve the Fokker–Planck–Kolmogorov partial differential equation as described in Section 9.6 using methods such as finite-differences or Galerkin methods, which also leads to approximations to the transition density. The Taylor series expansion considered in Section 9.4 can also be used to approximate the moments of the transition density, and given the moments we can, for example, form maximum entropy approximation to the density (Cover and Thomas, 2006). The simulated likelihood method in Section 9.7 was also originally proposed exactly for this purpose.

We get the following algorithm.

Algorithm 11.8 (Non-Gaussian approximated-likelihood estimation)**.** *Transition density approximation-based non-Gaussian approximated-likelihood parameter estimation can be done as follows:*

1. *Use a suitable method to form a parametric approximation* $\hat{p}(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta})$ *to the transition density using some of the previously discussed methods. Then approximate the negative log-likelihood as*

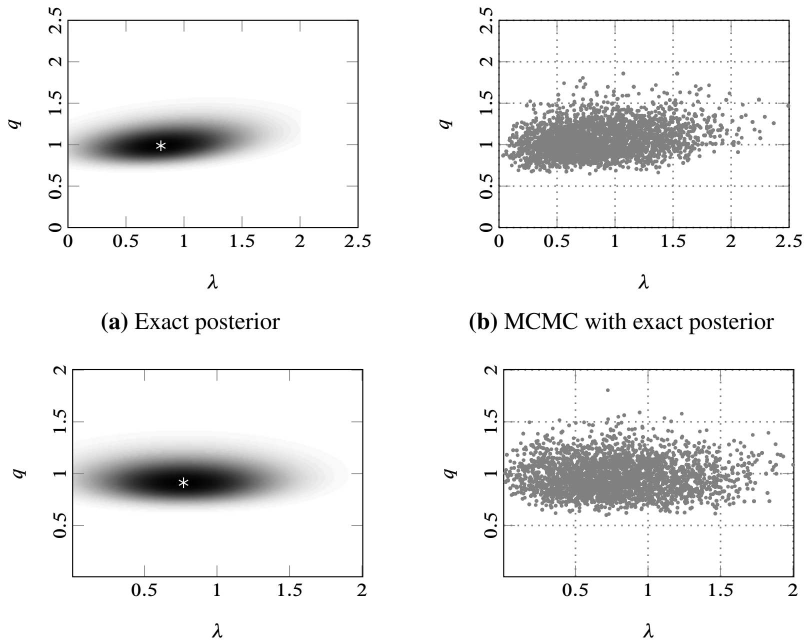
$$\hat{\ell}(\boldsymbol{\theta}) = -\sum_{k=0}^{T-1} \log \hat{p}(\mathbf{x}(t_{k+1}) \mid \mathbf{x}(t_k), \boldsymbol{\theta}). \tag{11.35}$$

2. *Perform maximum likelihood estimation or any form of Bayesian estimation by replacing the exact likelihood* ℓ *with the preceding approximation* $\hat{\ell}$*.*

Example 11.9 (Approximate parameter estimation in the Ornstein–Uhlenbeck model)**.** *The parameter estimation problem in Example 11.5 was repeated using an approximated likelihood method using the Euler–Maruyama transition density approximation. Figure 11.2 shows a comparison of the exact and approximate posteriors. As can be seen in the figure, the effect of the approximation is the loss of the correlation between the parameters.*

11.5 Likelihood Methods for Indirectly Observed SDEs

So far in this chapter, we have assumed that the states $\mathbf{x}(t_1), \mathbf{x}(t_2), \mathbf{x}(t_3), \ldots$ have been perfectly measured. However, as we saw in Chapter 10, in many real-world problems we do not directly observe the states, but we only get to see measurements $\mathbf{y}_1, \mathbf{y}_2, \ldots$, which are indirectly related to the state and contain noise. However, it turns out



(a) Exact posterior

(b) MCMC with exact posterior

(c) Approximate posterior

(d) MCMC with approximate posterior

Figure 11.2 Illustration of the effect of the Euler–Maruyama approximation to the posterior distribution of Ornstein–Uhlenbeck model parameters from Example 11.9. The exact posterior and its MCMC samples are shown in subfigures (a) and (b), and the approximations in (c) and (d), respectively. The point estimates of the parameters with Euler–Maryuama approximation are $\hat{\lambda}_{\text{MAP}} = 0.77$ and $\hat{q}_{\text{MAP}} = 0.91$ when with the exact posterior they were $\lambda_{\text{MAP}} = 0.80$ and $q_{\text{MAP}} = 0.99$.

that the filtering and smoothing methods can be combined with parameter estimation methods to allow for parameter estimation from such indirect, noisy measurements. These kinds of methods for SDE models have been presented, for example, in Singer (2002), Mbalawata et al. (2013), and Särkkä et al. (2015b) and their discrete-time analogs in Cappé et al. (2005) and Särkkä (2013). However, the idea itself dates back at least to Schweppe (1965) and Jazwinski (1970).

Let us assume that our model has the form

$$\begin{aligned} \mathrm{d}\mathbf{x} &= \mathbf{f}(\mathbf{x}, t; \boldsymbol{\theta})\, \mathrm{d}t + \mathbf{L}(\mathbf{x}, t; \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\beta}(t), \\ \mathbf{y}_k &\sim p(\mathbf{y}_k \mid \mathbf{x}(t_k); \boldsymbol{\theta}), \end{aligned} \tag{11.36}$$

where $\boldsymbol{\theta}$ is a vector of unknown parameters of the system. This model is a continuous-discrete model for which we developed two types of Bayesian filters in Section 10.5. The key to the solution is to notice that the normalization constant Z_k in Equation (10.54), which is a byproduct of the Bayesian filtering equations, is actually

$$\begin{aligned} Z_k(\boldsymbol{\theta}) &= \int p(\mathbf{y}_k \mid \mathbf{x}(t_k); \boldsymbol{\theta})\, p(\mathbf{x}(t_k) \mid \mathbf{y}_1, \dots, \mathbf{y}_{k-1}; \boldsymbol{\theta})\, \mathrm{d}\mathbf{x}(t_k) \\ &= p(\mathbf{y}_k \mid \mathbf{y}_1, \dots, \mathbf{y}_{k-1}; \boldsymbol{\theta}). \end{aligned} \tag{11.37}$$

We can now express the marginal likelihood by using the prediction error decomposition

$$p(\mathbf{y}_1, \dots, \mathbf{y}_T \mid \boldsymbol{\theta}) = \prod_{k=1}^{T} p(\mathbf{y}_k \mid \mathbf{y}_1, \dots, \mathbf{y}_{k-1}; \boldsymbol{\theta}) = \prod_{k=1}^{T} Z_k(\boldsymbol{\theta}). \tag{11.38}$$

Thus, provided that we can compute the terms $Z_k(\boldsymbol{\theta})$ or approximate them accurately, we can obtain the expression for the (negative logarithm of marginal) likelihood as

$$\ell(\boldsymbol{\theta}) = -\log p(\mathbf{y}_1, \dots, \mathbf{y}_T \mid \boldsymbol{\theta}) = -\sum_{k=1}^{T} \log Z_k(\boldsymbol{\theta}). \tag{11.39}$$

For linear Gaussian models, we can compute these likelihoods exactly with Kalman filters, whereas for nonlinear and non-Gaussian models we need to approximate them. Fortunately, the same approximations that can be used for approximate continuous-discrete filtering also provide approximations to the terms $Z_k(\boldsymbol{\theta})$.

The resulting likelihood expressions or approximations can be further optimized or sampled using methods such as MCMC. For more information on these kinds of methods, the reader is referred to Särkkä (2013), Mbalawata et al. (2013), and Särkkä et al. (2015b).

11.6 Expectation–Maximization, Variational Bayes, and Other Methods

There is also a wide range of other parameter estimation methods that we have not discussed here. We have not considered, for example, estimating function methods, generalized method of moments, nor methods based on approximating the continuous-time estimators. For more details about these methods, the reader is referred to the books of Rao (1999) and Iacus

(2008). We also have only considered parametric models, although nonparametric estimation of, for example, drift functions is possible as well (see Section 12.6 and Rao, 1999; Ruttor et al., 2013).

The indirectly observed case in the previous section has also spanned many other families of methods for parameter estimation in SDEs. For example, expectation–maximization methods can be used to approximate the maximum likelihood methods of discrete-time state-space models (e.g., Shumway and Stoffer, 1982; Särkkä, 2013; Kokkala et al., 2016). However, provided that we discretize the model first, these methods are directly applicable to continuous-discrete SDE models as well. So-called variational Bayes methods (e.g., Šmídl and Quinn, 2006) can also be used in the context of SDEs after the SDE has been discretized. Archambeau and Opper (2011) have also developed variational Bayes methods that can directly be used to estimate parameters in SDEs.

11.7 Exercises

11.1 Consider the following problem with unknown parameter θ:

$$x_k = \theta + r_k, \quad k = 1, 2, \dots, T, \tag{11.40}$$

where $r_k \sim \mathrm{N}(0, \sigma^2)$ are independent. Then accomplish the following:

(a) Derive the ML estimate of θ given the measurements $x_1, x_2, \dots, x_T$.

(b) Fix $\theta = 1, \sigma = 1$ and simulate a set of data from the preceding model. Then compute the ML estimate. How close is it to the truth?

(c) Plot the negative log-likelihood as function of the parameter θ. Is the maximum close to the true value?

11.2 Assume that in Equation (11.40) we have a Gaussian prior density $p(\theta) = \mathrm{N}(\theta \mid 0, \lambda^2)$. Then accomplish the following:

(a) Derive the MAP estimate of θ.

(b) How does the estimate behave as function of λ?

(c) Fix $\theta = 1, \sigma = 1, \lambda = 2$, and simulate a set of data from the preceding model. Then compute the MAP estimate. How close is it to the truth?

(d) Plot the posterior distribution of the parameter. Is the true parameter value well within the support of the distribution?

11.3 Estimate the parameter θ in Equation (11.40) using MCMC:

(a) Fix $\theta = 1, \sigma = 1, \lambda = 2$, and simulate a set of data from Equation 11.40 (with the prior).

(b) Implement an MH algorithm for sampling the parameter θ. Use a Gaussian proposal distribution $q(\theta^* \mid \theta) = \mathrm{N}(\theta^* \mid \theta, \gamma^2)$, and select the parameter γ such that about $1/4$ of the proposals are accepted.

(c) Plot the true posterior distribution and the histogram of samples. Do the results match?

11.4 Fill in the details in the derivation of the Ornstein–Uhlenbeck ML estimate:

(a) Derive Equations (11.18) from (11.17).
(b) Derive Equation (11.19).

11.5 Consider the parameter estimates (11.23) in the context of the Wiener velocity model. Assume that the only unknown is q:

(a) Recall the expressions for $\mathbf{A}(\Delta t)$ and $\boldsymbol{\Sigma}(\Delta t)$ from Example 6.3.
(b) Choose some Δt and q, and simulate some data from the exact discrete-time model in the example.
(c) Compute estimates $\mathbf{A}_{\mathrm{ML}}$ and $\boldsymbol{\Sigma}_{\mathrm{ML}}$ using Equations 11.23.
(d) Does the estimate of $\mathbf{A}$ match the correct one? Could you determine the parameter q from the estimate of $\boldsymbol{\Sigma}$?

11.6 Using the data from the preceding exercise, estimate q by numerically finding the ML estimate using the negative log-likelihood expression (11.25).

11.7 Still using the same data, estimate the posterior distribution of the parameter q by generating samples from the posterior distribution with uniform prior $p(q) \propto 1$ for $q > 0$ using the MH algorithm. Use a log transform $\theta = \log q$, which transforms the uniform prior to $p(\theta) \propto \exp(\theta)$.

11.8 Derive the derivatives of Equation (11.25) using the matrix fraction decomposition (see, e.g., Mbalawata et al., 2013). Check numerically using the Wiener velocity model that they are correct.

11.9 Consider the following model, where the parameters θ_1 and θ_2 are unknown:

$$\mathrm{d}x = \theta_1 \sin(x - \theta_2)\,\mathrm{d}t + \mathrm{d}\beta, \quad x(0) = 0,$$

where β is a standard Brownian motion.

(a) Simulate data from the model using the Euler–Maruyama method.
(b) Compute the maximum likelihood estimates of the parameters by using Euler–Maruyama approximation to the transition density.
(c) Compute the MAP estimates of the parameters with independent $\mathrm{N}(0, 1)$ priors on them both.
(d) Simulate samples from the posterior distribution of parameters using the MH algorithm.

11.10 Repeat Exercise 11.9 with the Itô–Taylor method in Algorithm 8.4.

11.11 Repeat Exercise 11.9 by replacing the transition density with a linearization approximation given in Algorithm 9.4.

11.12 Extend the Kalman filter equations in Example 10.19 so that they can be used to compute the marginal likelihood of parameters λ and q. Using simulated data, estimate the parameters from noisy observations by maximizing the marginal likelihood.