

## § Statistics of SDE solution

In the previous section, we have shown how to simulate paths from SDEs. In reality, we may also be interested in the statistical quantities of SDEs instead of their trajectories. In this section, we show how to compute a number of statistical quantities as follows,

1.  $\mathbb{E}[\phi(X(t)) \mid X(s)]$ ,  $t > s$ , for any test function of interests. For instance, we can choose  $\phi$  to be  $x \mapsto x$  and  $x \mapsto x^2$  to obtain the conditional mean and ~~variance~~ variance of  $X(t)$ . We can use such conditional mean and covariance to, for example, predict the future COVID-19 cases based on the current.
2.  $p(x, t)$ , the time-marginal probability density function of  $X(t)$  for  $t \in [0, \infty)$ . We can view ~~SDEs as SDEs~~ as means that transform the probability measure of the initial  $X(0)$  to another at  $t$ , continuously. Based on this view, we show the celebrated Kolmogorov forward equation (also known as Fokker-Planck) that governs the time evolution of  $p(x, t)$ , as well as a stationary solution to the equation.

### 3. Parameter estimation with maximum likelihood.

Suppose that we have an SDE with unknown parameters, and that we have access to its trajectories. It is possible to identify the unknown parameters by maximum likelihood estimation.

### 4 Computing conditional expectations

Recall our SDE

$$dX(t) = a(X(t))dt + b(X(t))dW(t), \quad X(0) = X_0$$

where  $X \in \mathbb{R}^d$ ,  $a: \mathbb{R}^d \rightarrow \mathbb{R}^d$ ,  $b: \mathbb{R}^{d \times d} \rightarrow \mathbb{R}^d$ ,  $W \in \mathbb{R}^n$ , and we are now interested in computing

$$\mathbb{E}[\phi(X(t)) | X(s)], \quad t > s.$$

For simplicity, let us for now assume that  $\phi: \mathbb{R}^d \rightarrow \mathbb{R}$  is a scalar-valued function. Computing such conditional expectations is not hard if we know the explicit solution to the SDE, for instance, geometric Brownian motion. However, in reality, we rarely know the SDE evolution, and we have to approximate the solution, hence

As an example, by the Euler-Maruyama approximation

$$\mathbb{E}[\phi(x(t)) | x(s)] \approx \mathbb{E}\left[\phi\left(x(s) + a(x(s))(t-s) + b(x(s))\Delta W\right) | x(s)\right].$$

The conditional expectation on the RHS is usually not analytically tractable, unless  $\phi$  is, for instance, a polynomial.

For example,

$$\mathbb{E}[x(t) | x(s)] \approx x(s) + a(x(s))(t-s),$$

$$\mathbb{E}\left[(x(t) - \mathbb{E}[x(t) | x(s)])(x(t) - \mathbb{E}[x(t) | x(s)])^\top\right] \approx b(x(s)) b(x(s))^\top (t-s),$$

But even for polynomial  $\phi$  we can write down the approximation explicitly, the number of terms in such approximation grows in a double factorial speed in the number of  $x$ 's dimension and the polynomial degree (see, Isserlis' theorem), although there are tricks to reduce the computation cost (see, Kan, 2008).

Moreover, such an approximation works well only if  $t-s$  is sufficiently small. In reality, we need to discretise the time interval  $[s, t]$  with multiple intermediate times to get a good approximation to  $x(t)$ . But in this way, there is ~~almost no chance to write down the approximation of~~  $\mathbb{E}[\phi(x(t)) | x(s)]$  ~~in closed-form~~. We then need to approximate because  $x(t)$  is ~~no longer Gaussian-~~ approximated.

it by Monte Carlo at the cost of additional errors which dependent on phi. Essentially, let  $X^{(1)}(t)$ ,  $X^{(2)}(t)$ , ...,  $X^{(N)}(t)$  be ~~not~~ independent samples of the SDE at  $t$  starting from  $X(s)$  at  $s$ , then

$$\mathbb{E}[\phi(X(t)) | X(s)] \approx \frac{1}{N} \sum_{i=1}^N \phi(X^{(i)}(t)).$$

Algorithm<sup>b</sup> Approximate  $\mathbb{E}[\phi(X(t)) | X(s)]$  by discretisation.

Input:  $\phi$ ,  $t$ ,  $s$ , and  $N$

Output: Approximate  $\mathbb{E}[\phi(X(t)) | X(s)]$ .

1. Partition the time interval  $[s, t]$  by  $T$  times  
 $s < t_1 < t_2 < \dots < t_T = t$

2. For  $i = 1, 2, \dots, N$  do  
use e.g., Euler–Maruyama to simulate the SDE at the times  $t_1, t_2, \dots, t_T$  then select the end value to get  $X^{(i)}(t)$ .

Return  $\frac{1}{N} \sum_{i=1}^N \phi(X^{(i)}(t))$

one downside of the MC method is the computation. In order to get a qualified estimation, we need to use a large number of MC samples and the discrete times. Moreover, the variance of the MC estimator heavily depends on the function  $\phi$ , although it is scaled by  $N^{\frac{1}{2}}$ . As an example, suppose that  $X$  is unidimensional, and that we are interested in computing the moments:

$$\mathbb{E}\left[\frac{1}{N} \sum_{i=1}^N (X(t_i))^\alpha\right] = \mathbb{E}[X(t)^\alpha], \text{ for } n=1, 2, \dots$$

$$\text{Var}\left[\frac{1}{N} \sum_{i=1}^N (X(t_i))^\alpha\right] = \frac{1}{N} \text{Var}[X(t)^\alpha].$$

The estimator is unbiased indeed, but the variance will significantly increase as  $n$  goes large, hence we would need more samples to compensate the error.

Example 17. Recall the geometric Brownian motion SDE

$$dX(t) = a X(t) dt + b X(t) dW(t), \quad X(0) = X_0,$$

the solution of which is

$$X(t) = X_0 e^{(a - \frac{b^2}{2})t + b W(t)}.$$

We would like to compute its moments  $\mathbb{E}[X(t)^\alpha | X(0)]$  for

$n=1, 2, \dots$ . Since we know that  $X(t)$  is log-Normal distributed, we can compute such quantities exactly. We then use the exact results to compare to the other approximate methods.

See 'lec4\_moments\_estimation\_COMPARE.ipynb'.

Another downside of the MC estimation is the biasedness. It is hard to analyse if the estimator is biased before running the ~~estimation~~, except for a few isolated cases, for example, central moments. Recall what is the unbiased estimator of the variance?

↳ Analytical computation of conditional expectations.

what are the assumptions precisely?

Under a few assumptions, it is possible to compute  $\mathbb{E}[\phi(X(t)) | X(s)]$  analytically without relying on the discretisation scheme nor MC. The idea is to apply Itô's formula iteratively.

To see this, suppose that the function of interest  $\phi$  is twice continuously differentiable. The Itô's formula shows that

$$\begin{aligned}
\phi(x(t)) &= \phi(x(s)) + \int_s^t \nabla_x \phi(a(x(z)))^\top a(x(z)) + \frac{1}{2} \text{tr}(\Gamma(x(z)) H_x \phi(x(z))) dz \\
&\quad + \int_s^t \nabla_x \phi(a(x(z)))^\top b(x(z)) dW(z) \\
&:= \phi(x(s)) + \int_s^t A\phi(x(z)) dz + \int_s^t \nabla_x \phi(a(x(z)))^\top b(x(z)) dW(z), \quad (1)
\end{aligned}$$

where we denote

$$\begin{aligned}
A\phi(x) &:= \nabla_x \phi(x)^\top a(x) + \frac{1}{2} \text{tr}(\Gamma(x) H_x \phi(x)) \\
&:= \sum_{i=1}^d \frac{\partial \phi}{\partial x_i} a_i(x) + \frac{1}{2} \sum_{i,j=1}^d \Gamma_{ij}(x) \frac{\partial^2 \phi}{\partial x_i \partial x_j}
\end{aligned}$$

the infinitesimal generator of the SDE. Now apply

$\mathbb{E}[ \cdot | x(s) ]$  on both side of (1), we get:

$$\begin{aligned}
\mathbb{E}[\phi(x(t)) | x(s)] &= \phi(x(s)) + \int_s^t \mathbb{E}[A\phi(x(z)) | x(s)] dz \\
&\quad + \underline{0}. \quad \cancel{dW}.
\end{aligned}$$

The expectation of the Itô integral is zero because we assume that it is a martingale.

↳ This not ~~is general~~ always true. because the Itô integral can also be a local martingale, see, Kuo, 2006, ch.5.

Hence, we can see that  $\mathbb{E}[\phi(X(t)) | X(s)]$  solves an ODE:

$$\frac{d\mathbb{E}[\phi(X(t)) | X(s)]}{dt} = \mathbb{E}[A\phi(X(t)) | X(s)], \quad t > s$$

Starting from the initial value  $\phi(X(s))$  at  $s$ . This result is intuitive, since we essentially project the SDE into an ODE.

However, the RHS of the ODE above is not tractable, as there is an expectation  $\mathbb{E}[A\phi(X(t)) | X(s)]$  which are in general not ~~extra~~ tractable. But we can again apply Ifb's formula on  $A\phi$ , then we arrive at

$$A\phi(X(t)) = A\phi(X(s)) + \int_s^t A^2\phi(X(\tau)) d\tau + \int_s^t \dots \text{omit} \dots dW(\tau).$$

In fact

$$A^r\phi(X(t)) = A^r\phi(X(s)) + \int_s^t A^{r+1}\phi(X(\tau)) d\tau + \int_s^t \dots \text{omit} \dots dW(\tau)$$

for all  $r = 0, 1, 2, \dots$ . Hence, by applying this identity multiples times iteratively, we have

$$\begin{aligned}
 \mathbb{E}[\phi(x(t)) | x(s_1)] &= \phi(x(s_1)) + \int_s^t \mathbb{E}[A\phi(x(z)) | x(s_1)] dz \\
 &= \phi(x(s_1)) + \int_s^t \left( A\phi(x(s_1)) + \int_s^z A^2\phi(x(z)) dz \right) dz \quad \text{missing } \mathbb{E}[ \cdot | x(s_1) ] \\
 &= \phi(x(s_1)) + A\phi(x(s_1))(t-s) + \int_s^t \int_s^z \left( A^2\phi(x(s_1)) + \int_s^h A^3\phi(x(h)) dh \right) dz dh \\
 &= \phi(x(s_1)) + A\phi(x(s_1))(t-s) + \frac{1}{2} A^2\phi(x(s_1))(t-s)^2 + \dots
 \end{aligned}$$

We have Based on this observation, we have the following theorem:

**Theorem 18 (Taylor moment expansion).** Let  $M \geq 0$  be an integer and  $X$  be the solution to the SDE. Suppose that the SDE drift  $a: \mathbb{R}^d \rightarrow \mathbb{R}$  and dispersion  $b: \mathbb{R}^{dxw} \rightarrow \mathbb{R}^d$  are  $M$  times continuously differentiable, and that  $\phi: \mathbb{R}^d \rightarrow \mathbb{R}$  is  $2(M+1)$  times continuously differentiable, then

$$\mathbb{E}[\phi(x(t)) | x(s_1)] = \sum_{r=0}^M \frac{1}{r!} A^r \phi(x(s_1)) (t-s)^r + R_{M,\phi}(x(s_1), t-s),$$

where the remainder reads

$$R_{M,\phi}(x(s_1), t-s) := \int_s^t \int_s^z \dots \int_s^{z_m} \mathbb{E}[A^{M+1}\phi(x(z)) | x(s_1)] dz \dots dz_m.$$

Therefore we can analytically approximate the conditional expectations by using such expansions and discarding their remainders. However, please remark that the convergence of such expansions specifically depend on the time interval  $t-s$  and the SDE coefficients. It is possible that by fixing a  $t-s$ , the expansion does not converge as  $M \rightarrow \infty$ .

Example 19. Recall the modified Duffing van der Pol SDE-

$$d \begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix} = \begin{bmatrix} X_2(t) \\ \cancel{\frac{d}{dt}(X_1(t)^2)} - X_2(t) \end{bmatrix} dt + \begin{bmatrix} 0 \\ X_1(t) \end{bmatrix} dW(t),$$

$$X(0) = X_0.$$

and we are interested in computing  $\mathbb{E}[X_1(t) | X(s)]$ . We define  $\phi(x) = x$ , that select the first element for any  $x \in \mathbb{R}^2$ , then applying  $A$  we get:

$$\begin{aligned} A\phi(x) &= \nabla_x \phi(x)^T a(x) + \frac{1}{2} \text{tr}(T(x) H x \phi(x)) \\ &= a_1(x) + 0 \end{aligned}$$

$$A^2\phi(x) = \nabla_x a_1(x)^T a(x) + \frac{1}{2} \text{tr}(T(x) H x a_1(x))$$

$$A^3\phi(x) = \dots$$

$$\begin{aligned} \mathbb{E}[X_1(t) | X(s)] &\approx \cancel{X_1(s)} + \sum_{r=0}^{\infty} \frac{1}{r!} A^r \phi(X(s))_{k=1} = X_1(s) + X_2(s)(t-s) * \\ &\quad + (X_1(s)(\alpha - X_1(s)^2) - X_2(s)) \frac{(t-s)^2}{2} + \dots \end{aligned}$$

Back in time, it is difficult to implement the Taylor moment expansion, ~~but~~ due to the iterative derivatives. But now with the help of automatic differentiation and symbolic computation, it is possible to implement it efficiently. See the python package 'tme'.

## ↳ Kolmogorov forward equation

The Kolmogorov forward equation (KFE), also known as, Fokker-Planck equation in physics, is a partial differential equation (PDE) that governs the time-evolution of the time-marginal probability density function of SDE solution. More precisely, if  ~~$X: \mathbb{R}^d \rightarrow \mathbb{R}$~~   $X: [0, \infty) \rightarrow \mathbb{R}^d$  is a diffusion process that satisfies the SDE

$$dX(t) = a(X(t))dt + b(X(t))dW(t),$$

then the probability density function (PDF)  $P(X, t)$  of  $X(t)$

for every  $t \in [0, \infty)$  satisfies

$$\frac{\partial P(x, t)}{\partial t} = - \sum_{i=1}^d \frac{\partial}{\partial x_i} [a_i(x) P(x, t)] + \frac{1}{2} \sum_{i,j=1}^d \frac{\partial^2}{\partial x_i \partial x_j} [T_{ij}(x) P(x, t)],$$

starting from the initial  $P(x_0)$  of  $X(0)$ . If  $d=1$ , the PDE simplifies to

$$\frac{\partial P(x, t)}{\partial t} = - \frac{\partial}{\partial x} [a(x) P(x, t)] + \frac{1}{2} \frac{\partial^2}{\partial x^2} [T(x) P(x, t)].$$

We can informally prove the KFE by Ito's formula, and integration by parts. For simplicity, let us consider  $d=1$ . We also need to assume that  $P(x, t) a(x)$  and  $T(x) P(x, t)$  vanishes as  $x \rightarrow -\infty$  and  $x \rightarrow \infty$ .

By Ito's formula, for any  $\phi$ , this is loose!

$$\begin{aligned} \phi(X(t)) &= \phi(X(0)) + \int_0^t \frac{d\phi(X(s))}{dx} a(X(s)) + \frac{1}{2} \int_0^t \frac{d^2\phi(X(s))}{dx^2} T(X(s)) ds \\ &\quad + \int_0^t \dots \text{omit } \dots dW(s). \end{aligned}$$

Take expectations on both sides and rewrite the integral equation in the differential form, we get

$$\frac{d\mathbb{E}[\phi(x(t))]}{dt} = \mathbb{E}\left[\frac{d\phi(x(t))}{dx} a(x(t)) + \frac{1}{2} \frac{d^2\phi(x(t))}{dx^2} \Gamma(x)\right],$$

which writes in terms of  $P(x,t)$  as  
move  $\frac{dt}{dt}$  inside

$$\begin{aligned} \frac{d}{dt} \int \phi(x) P(x,t) dx &= \int \phi(x) \frac{\partial P(x,t)}{\partial t} dx \\ &= \underbrace{\int \frac{d\phi(x)}{dx} a(x) P(x,t) dx}_{(1)} + \underbrace{\frac{1}{2} \frac{d^2\phi(x)}{dx^2} \Gamma(x) P(x,t) dx}_{(2)} \end{aligned}$$

Apply integration by parts on (1) we have

$$\begin{aligned} \int \frac{d\phi(x)}{dx} a(x) P(x,t) dx &= \phi(x)a(x)P(x,t) \Big|_{-\infty}^{\infty} - \int \phi(x) d(a(x)P(x,t)) \\ &= - \int \phi(x) \frac{d}{dx} [a(x)P(x,t)] dx. \end{aligned}$$

Similar, apply integration by parts twice on (2). We finally have

$$\int \phi(x) \frac{\partial P(x,t)}{\partial t} dx = - \int \phi(x) \frac{d}{dx} [a(x)P(x,t)] dx + \frac{1}{2} \int \phi(x) \frac{d^2}{dx^2} [\Gamma(x)P(x,t)] dx.$$

Since the equation above must hold for all  $\phi$ , we conclude that  $\frac{dP(x,t)}{dt} = -\frac{d}{dx}[a(x)P(x,t)] + \frac{1}{2}\frac{d^2}{dx^2}[T(x)P(x,t)]$  which is the KFE. Why? Imagine an inner product defined by  $\langle f, g \rangle := \int f(x)g(x)dx$ . If we know that  $\langle \phi, f \rangle = \langle \phi, g \rangle$  for all  $\phi$ , then  $\langle \phi, fg \rangle = 0$  for all  $\phi$ . The only vector that is orthogonal to all vectors is the zero vector =  $f-g$ .

Why this derivation of KFE is informal? 1) the assumption on the SDE coefficients and the PDF. 2) the equality holds in terms of a specific set of test functions. 3) ---- homework!

In fact, the derivation of KFE is independent of SDE.

The KFE holds for all diffusion processes which are not necessarily solutions of SDEs. Diffusion processes and SDEs are different concepts, although they are connected. It is possible that a diffusion process does not satisfy any SDE.

We are now interested in how to numerically solve the KFE. For pedagogy, let us consider the simplest finite difference method for 1D-KFE. You could also try Galerkin's method as a homework.

The basic idea of the finite difference methods is to approximate the spatial derivatives (i.e.,  $\frac{\partial}{\partial x}$ ,  $\frac{\partial^2}{\partial x^2}$ , ...) by finite differences, ~~so that~~ at discrete spatial locations, so that the PDE ~~reduces to a system of ODEs in time.~~

More precisely, suppose that we are interested in the solution  $p(x, t)$  at spatial locations  $X_{1:N} := \{x_1, x_2, \dots, x_N\}$ , and times  $t_{1:T} := \{t_1, t_2, \dots, t_T\}$ , we essentially would like to compute a matrix

$$\bar{P}(X_{1:N}, t_{1:T}) = \begin{bmatrix} p(x_1, t_1) & p(x_1, t_2) & \dots & p(x_1, t_T) \\ p(x_2, t_1) & \ddots & & \\ \vdots & & \ddots & \\ p(x_N, t_1) & & & p(x_N, t_T) \end{bmatrix} \in \mathbb{R}^{N \times T}$$

To obtain the vectors  $\bar{p}(X_{1:N}, t_1), \bar{p}(X_{1:N}, t_2), \dots, \bar{p}(X_{1:N}, t_T)$  in time, let us approximate the spatial derivatives at  $X_{1:N}$  as follows:

$$\frac{\partial}{\partial x} [a(x_i) p(x_i, t)] \approx \frac{a(x_{i-1}) p(x_{i-1}, t) - a(x_{i+1}) p(x_{i+1}, t)}{2\epsilon},$$

$$\epsilon = x_{i+1} - x_i$$

$$\frac{\partial^2}{\partial x^2} [\Gamma(x_i) p(x_i, t)] \approx \frac{\Gamma(x_{i-1}) p(x_{i-1}, t) - 2\Gamma(x_i) p(x_i, t) + \Gamma(x_{i+1}) p(x_{i+1}, t)}{\epsilon^2}$$

Hence

$$\frac{\partial \bar{P}(x_{1:N}, t)}{\partial t} \Rightarrow \begin{bmatrix} p(x_0, t) \\ p(x_1, t) \\ p(x_2, t) \\ \vdots \\ p(x_N, t) \\ p(x_{N+1}, t) \end{bmatrix} \approx \begin{bmatrix} 1 & 0 & \cdots & \cdots \\ a(x_0) & 0 & -a(x_2) & \cdots \\ a(x_1) & 0 & -a(x_3) & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ p(x_N, t) & p(x_{N+1}, t) \end{bmatrix}$$

divided by  $\frac{1}{\Delta x}$  missing

$$+ \frac{1}{2\Delta x^2} \begin{bmatrix} 0 & 0 & \cdots & \cdots \\ T(x_0) & -2T(x_1) & T(x_2) & \cdots \\ & T(x_1) & -2T(x_2) & T(x_3) & \cdots \\ & & & \ddots & \ddots \\ 0 & & & & \end{bmatrix} \begin{bmatrix} p(x_0, t) \\ p(x_1, t) \\ p(x_2, t) \\ \vdots \\ p(x_N, t) \\ p(x_{N+1}, t) \end{bmatrix}$$

How to deal  
with the boundaries  
 $p(x_0, t)$  and  $p(x_{N+1}, t)$ ?

We clearly see that the equation above is a linear system  
of ODEs, and we can solve the linear ODEs explicitly.

We can also use Euler approximation

$$\bar{P}(x_{1:N}, t_k) \approx \bar{P}(x_{1:N}, t_{k-1}) + [\text{finite-difference matrix}] \bar{p}(x_{1:N}, t_{k-1})(t_k - t_{k-1})$$

but unnecessary.

It is possible to approximate the spatial derivatives with  
higher-order finite difference schemes, you may do a homework  
by searching finite difference stencil/table.

In practice, when you implement the finite difference in Python, the function 'np.gradient' is your good friend. To approximate the second derivatives, apply np.gradient twice.

Example, 20. Again recall the geometric Brownian motion

$$dX(t) = a X(t) dt + b X(t) dW(t),$$

$$X(0) \sim N(M_0, V_0),$$

and the solution  $X(t) = X(0) e^{(a - \frac{b^2}{2})t + b W(t)}$ .

We can easily make samples of ~~X at~~  $X(t)$  at any time  $t$ , then verify the sample histogram against the (approximate) solution to the associated KFE.

see 'lec4\_kolmogorov\_forward\_equation.ipynb' to see how this is implemented.

It is worth remarking a special classes of SDEs of the form

$$dx(t) = -\frac{1}{2} \nabla_x \log P(x(t)) dt + dW(t).$$

This SDE is special because as  $t \rightarrow \infty$ , the density of  $x(t)$  will converge to the stationary ~~P(x)~~  $p(x)$  as defined in the SDE above. Indeed, ~~on~~ we can substitute the  $p(x)$  ~~and~~ to the associated KFE, and find that the RHS of the KFE is zero, so  $\frac{\partial p(x,t)}{\partial t} = 0$ .

This class of SDEs has a great impact in statistics. In statistics, people are interested in generating samples according to a distribution ~~P~~. We can then generate such samples by simulating the SDE with the drift function defined by  $-\frac{1}{2} \nabla_x \log P(x)$ . This is the key idea behind the Langevin MCMC and Metropolis-adjusted Langevin algorithm.

for long enough time

Example: Imagine  $P(x)$  is a Normal density function,  $\nabla_x \log P(x)$  will give a linear function.

## SDE parameter estimation

The Kolmogorov forward equation gives the time-marginal density  $p(x, t)$ . In many applications, we are also interested in the finite-dimensional distribution of the SDE solution, viz., the joint distribution of  $X(t_1), X(t_2), \dots, X(t_T)$  for any  $t_1 < t_2 < \dots < t_T$ . By the Markov property, the joint density factorizes as

$$p(x_0, x_1, x_2, \dots, x_T) = p(x_0) \prod_{k=1}^T p(x_k | x_{k-1}).$$

which is the transition density  $p(x_k | x_{k-1})$  is usually not tractable in closed-form, although it satisfies the Kolmogorov backward equation with the dirac delta as the initial. In practice, we can approximate the transition density by, for instance,

$$p(x_k | x_{k-1}) \approx N(x_k | \mathbb{E}[X(t_k) | X(t_{k-1})], \text{Cov}[X(t_k) | X(t_{k-1})]),$$

where we can further approximate the transition mean and

covariance by Euler-Maryama or Taylor moment expansion

We need to evaluate the joint PDF when we want to identify the unknown parameters of SDE by maximum likelihood estimation. More specifically, suppose that we have an SDE

$$dX(t) = a(X(t); \theta) dt + b(X(t); \theta) dW(t),$$

but the parameter  $\theta$  is unknown. If we have access to its samples ~~at~~, say,  $\{X_1, X_2, \dots, X_T\}$  at times  $t_1, t_2, \dots, t_T$ , we can then use the maximum likelihood estimation (MLE) to estimate  $\theta$ . The method consists in an optimisation

$$\hat{\theta} = \arg \min_{\theta} \left( -\log P(X_1; \theta) - \sum_{k=2}^T \log P(X_k | X_{k-1}; \theta) \right).$$

If we have any prior knowledge on the parameter  $\theta$ , we can also put a prior  $P(\theta)$  over it then compute the posterior distribution by, e.g., MCMC.

Example 21. Consider an SDE

$$dX(t) = \theta \sin(X(t)) dt + dW(t),$$

where  $\theta$  is a parameter. Set  $\theta$  to be some number, then generate a trajectory of the SDE  $X_1, X_2, \dots, X_T$  based on the  $\theta$ . Now apply MLE to estimate  $\theta$  from the generated trajectory then compare the estimate to the true  $\theta$ .

See 'lec4\_Parameter-estimation-mle.ipynb' for how this is implemented.