# Exercise 1

A computational introduction to stochastic differential equations
FTN0332 TN22H006

## How to pass this exercise

This exercise round is concerned with Lectures 2 - 3. To pass this exercise, score $\geq$ **12** points and finish the assignment(s) marked with $\star$. Please submit your assignments in an email sent to `zheng.zhao@it.uu.se` before 13:15, 2 Nov, 2022.

## Assignment 1 (1 point)

Let $W$ be a standard Brownian motion, and $f\colon \mathbb{R} \to \mathbb{R}$ be a continuously differentiable function. Show that

$$\int_0^t f(W(s))\,\mathrm{d}W(s) = \int_0^{W(t)} f(s)\,\mathrm{d}s - \frac{1}{2}\int_0^t f'(W(s))\,\mathrm{d}s,$$

where $f'$ denotes the derivative of $f$. (Hint: $F(u) \coloneqq \int_0^u f(s)\,\mathrm{d}s$ and Itô's formula.)

## Assignment 2 (1 point)

Let $W$ be a standard Brownian motion, and define

$$X(t) \coloneqq \sin\big(\alpha\, W(t)\big).$$

Show that $X$ satisfies the SDE

$$\mathrm{d}X(t) = -\frac{\alpha^2}{2}\,X(t)\,\mathrm{d}t + \alpha\sqrt{1 - X(t)^2}\,\mathrm{d}W(t), \quad X(0) = 0.$$

(Hint: Itô's formula)

## Assignment 3 (1 point)

Consider an SDE

$$\mathrm{d}X(t) = a(X(t))\,\mathrm{d}t + b\,\mathrm{d}W(t),$$

where $X\colon [0,\infty) \to \mathbb{R}^d$, $a\colon \mathbb{R}^d \to \mathbb{R}^d$, $b \in \mathbb{R}^d$, and $W$ is a standard Brownian motion. Suppose that the drift function $a$ is smooth and is the gradient of a smooth mapping $\phi\colon \mathbb{R}^d \to \mathbb{R}$, viz.,

$$a(X(t)) \coloneqq \nabla\phi(X(t)).$$

Show that

$$\mathrm{d}\phi(X(t)) = \left(\|a(X(t))\|_2^2 + \frac{1}{2}\,\mathrm{tr}\Big(b\,b^{\mathsf{T}}\,\mathrm{J}_X a(X(t))\Big)\right)\mathrm{d}t + a(X(t))^{\mathsf{T}}\,b\,\mathrm{d}W(t),$$

where $\|\cdot\|_2$ and $\mathrm{J}_X a$ denote the Euclidean norm and Jacobian of $a$, respectively. (Hint: Itô's formula.)

# Assignment 4 (2 points)

Let $W$ be a standard Brownian motion, and define $X$ by

$$X(t) \coloneqq X_0 \exp\left(\left(a - \frac{b^2}{2}\right)t + b\,W(t)\right).$$

Verify that $X$ satisfies the SDE

$$\mathrm{d}X(t) = a\,X(t)\,\mathrm{d}t + b\,X(t)\,\mathrm{d}W(t), \quad X(0) = X_0,$$

This process is also known as the geometric Brownian motion. (Hint: Itô's formula.)

# Assignment 5 (5 points)

Let $X_{1:T} \coloneqq \begin{bmatrix} X_1 & X_2 & \cdots & X_T \end{bmatrix} \in \mathbb{R}^T$ be a vector of zero-mean joint Normal random variables, distributed according to $X_{1:T} \sim \mathrm{N}(x_{1:T} \mid 0, C_{1:T})$ with a covariance matrix $C_{1:T} \in \mathbb{R}^{T\times T}$ whose $i,j$-th element is

$$(C_{1:T})_{ij} \coloneqq \mathrm{e}^{-\Delta\,|i-j|},$$

for $1 \leq i,j \leq T$. Set $T = 100$ and $\Delta = 0.1$.

- Compute the matrix $C_{1:T}$ and implement the PDF $\mathrm{N}(x_{1:T} \mid 0, C_{1:T})$ as a function of vector $x_{1:T}$. It is allowed to use e.g., `scipy.stats.multivariate_normal`.

The random variables $X_{1:T}$ are in fact can be generated from an SDE (which we will see in a later lecture). By the Markov property of the SDE, we can factorise the joint PDF in terms of its transition densities as follows.

$$\begin{aligned}
p(x_{1:T}) &\coloneqq p(x_1)\prod_{k=2}^{T} p(x_k \mid x_{k-1}),\\
p(x_1) &\coloneqq \mathrm{N}(x_1 \mid 0, 1),\\
p(x_k \mid x_{k-1}) &\coloneqq \mathrm{N}\big(x_k \mid \mathrm{e}^{-\Delta}\,x_{k-1}, 1 - \mathrm{e}^{-2\,\Delta}\big).
\end{aligned}$$

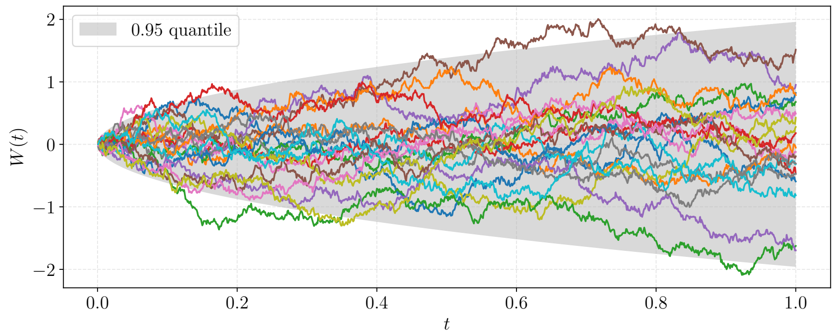


Figure 1: Trajectories of Brownian motion

- Implement $p(x_{1:T})$, and numerically verify that $p(x_{1:T}) = \mathrm{N}(x_{1:T} \mid 0, C_{1:T})$ for any input $x_{1:T}$. Compare the speed of the two PDF implementations and see which is faster, in particular, try let $T$ be some huge values.

**Note**: you may want to implement the PDFs in log scale for better numerical representation.

- (**Bonus** +2 **points**) We can also verify that the samples drawn from the joint density and the Markov-factorised density are the same. 1) Draw a sample $X_{1:T}^{(1)} \sim \mathrm{N}(x_{1:T} \mid 0, C_{1:T})$ from this multivariate Normal distribution. 2) Sequentially draw $X_1^{(2)} \sim p(x_1)$ then draw $X_2^{(2)} \mid X_1^{(2)} \sim p(x_2 \mid x_1)$ and so on until $X_T^{(2)} \mid X_{T-1}^{(2)} \sim p(x_T \mid x_{T-1})$. Collect these samples in a vector $X_{1:T}^{(2)}$. 3) Verify that $X_{1:T}^{(1)} = X_{1:T}^{(2)}$, and compare which sampling method is faster (recall to let the two sampling routines share the same randomness).

# ⋆ Assignment 6 (1 point)

*This assignment is mandatory, as you have to simulate a Brownian motion in order to simulate any SDE.* Simulate and plot 20 independent paths from a standard Brownian motion on the time interval $[0, 1]$. In the same figure, plot the 0.95 quantile/interval of the Brownian motion at the times, and verify that the trajectories are mostly within the interval. You should get a similar result as in Figure 1. Note: since $\mathrm{Var}[W(t)] = t$, the 0.95 quantile is a function $t \mapsto 1.96\, t$.

# Assignment 7 (2 point)

Using the Euler–Maruyama scheme to simulate 100 independent trajectories from the double-well SDE

$$\mathrm{d}X(t) = 4\,(X(t) - X(t)^3)\,\mathrm{d}t + \mathrm{d}W(t), \quad X(0) = 1,$$

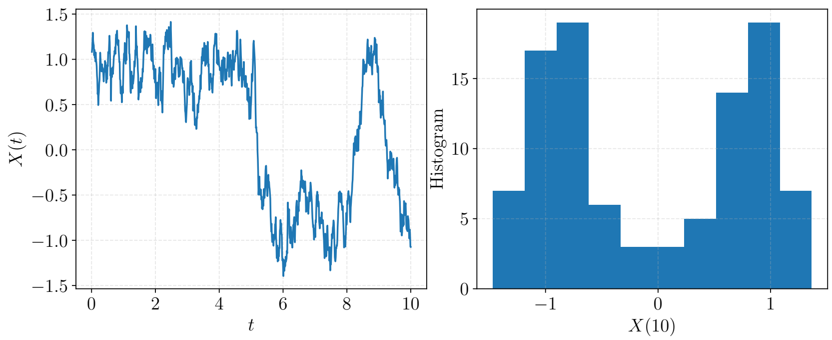


Figure 2: Left: a trajectory from the double-well SDE. Right: the histogram of the trajectories at $t = 10$.

at evenly placed times $t_1 = 0.01, t_2 = 0.02, \ldots, t_{1000} = 10$. Demonstrate one trajectory, and plot the histogram of the 100 trajectories at the terminal time $t_{1000}$. You should get a similar plot as in Figure 2.

# Assignment 8 (2 points)

Consider a Cox–Ingersoll–Ross (CIR) process given by

$$\mathrm{d}X(t) = a\left(b - X(t)\right)\mathrm{d}t + \sigma\sqrt{X(t)}\,\mathrm{d}W(t), \quad X(0) = 0.1,$$

where we let $a = 2$, $b = 0.5$, and $\sigma = 1.5$. Remark that this CIR process is by definition non-negative (i.e., $X(t) \geq 0$ for all $t \geq 0$). Now consider evenly placed times $t_1 = 0.001, t_2 = 0.002, \ldots, t_{10000} = 10$.

- Use Euler–Maruyama to simulate multiple independent trajectories of this SDE at the times. Check if any of the simulations numerically fails, and explain why.
- Use Milstein's method or any other higher order method to simulate the trajectory, and compare to that of Euler–Maruyama.

# Assignment 9 (3 points)

Recall the SDE in Assignment 4:

$$\mathrm{d}X(t) = a\,X(t)\,\mathrm{d}t + b\,X(t)\,\mathrm{d}W(t),$$

and we have shown that

$$X(t) \coloneqq X_0 \exp\left(\left(a - \frac{b^2}{2}\right)t + b\,W(t)\right)$$

solves the SDE. Now set $a = -1$, $b = 1$, $X_0 = 1$, and consider times $t_1 = \Delta, t_2 = 2\,\Delta, \ldots, t_T = T\,\Delta$. Let $T = 10$ and $\Delta = 0.1$.

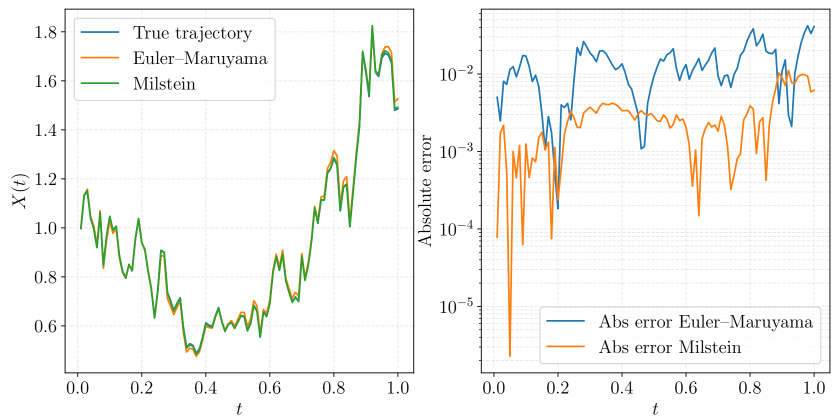


Figure 3: Left: trajectories of the SDE in Assignment 9. Right: absolute errors compared to the true trajectory.

- Simulate a trajectory of $X$ at the times by using the explicit solution.
- Use Euler–Maruyama to simulate a path of the SDE at the times, then compare to the true path. (recall to use the same realisation of the Brownian motion path to control the randomness)
- Keep $T\Delta = 1$ and increase the number of times $T$, for example, $(T = 100, \Delta = 0.01), (T = 1000, \Delta = 0.001), \ldots$ Check if the Euler–Maruyama approximation gets better as the discretisation gets finer.
- (**bonus** +2 **point**) use Milstein's method to simulate $X$ at the times, then compare to that of Euler–Maruyama (e.g., plot the absolute differences w.r.t. the true trajectory in log scale). As an example, Figure 3 compares the errors when $T = 100$ and $\Delta = 0.01$.

# Assignment 10 (2 points)

Consider a three-dimensional stochastic Lorenz model

$$\mathrm{d}\begin{bmatrix} X(t) \\ Y(t) \\ Z(t) \end{bmatrix} = \begin{bmatrix} \eta\,\big(Y(t) - X(t)\big) \\ X(t)\,\big(\rho - Z(t)\big) - Y(t) \\ X(t)\,Y(t) - \beta\,Z(t) \end{bmatrix} \mathrm{d}t + \sigma\,\mathrm{d}W(t),$$

$$\begin{bmatrix} X(0) \\ Y(0) \\ Z(0) \end{bmatrix} \sim \mathrm{N}(0, I_3),$$

where $W$ is a three-dimensional Brownian motion (and its components are mutually independent). Let $\eta = 10, \rho = 28,\ \beta = 8\,/\,3$, and $\sigma = 2$. Simulate a trajectory from this SDE by using Euler–Maruyama. The discretisation times are up to you.

(**bonus** +4 **points**) Use the order 1.5 strong Taylor scheme to simulate a trajectory, and compare to that of Euler–Maruyama. Recall the Gaussian-increment based simulation in the lecture note, so that you can compare them under the same randomness. The following figure is an example.

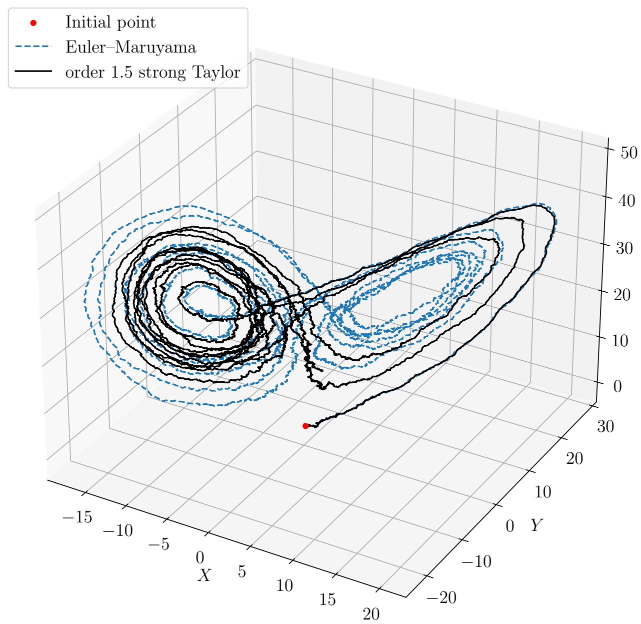


Figure 4: Euler–Maruyama and order 1.5 strong Taylor for simulating the Lorenz model ($T = 10^4, \Delta = 10^{-3}$). Due to the non-linearity of the model, two methods deviate significantly as $t$ increases.