Stat 544 Final Project: Applied Stochastic Processes

Lazizbek Sadullaev

April 2025

Question 1: Drift Diffusion Equation

We study specific stochastic differential equations (SDEs), here particularly the **drift-diffusion equation**.

For each SDE, we are asked to:

(i). Write down the SDE with its initial condition.

(ii). Define and describe all terms, variables, and parameters.

(iii). Solve the SDE using an appropriate method.

(iv). Choose specific parameters and plot several stochastic paths on the same graph.

(v). Study the expectation, variance, autocorrelation, and any other interesting properties.

(vi). Conduct a literature search and discuss one or two applications of the SDE.

Solution

(i) SDE with Initial Condition

The general form of an SDE is given by:

$$dX_t = \mu(X_t, t)\, dt + \sigma(X_t, t)\, dW_t, \tag{1}$$

where W_t is a standard Wiener process (also called Brownian motion). A **strong solution** satisfies:

$$X_t = X_0 + \int_0^t \mu(X_\tau, \tau)\, d\tau + \int_0^t \sigma(X_\tau, \tau)\, dW_\tau. \tag{2}$$

We consider linear SDEs with:

$$\mu(X_t, t) = a_1(t)X_t + a_2(t), \tag{3}$$
$$\sigma(X_t, t) = b_1(t)X_t + b_2(t), \tag{4}$$

leading to the **general linear SDE**:

$$dX_t = [a_1(t)X_t + a_2(t)]\, dt + [b_1(t)X_t + b_2(t)]\, dW_t. \tag{5}$$

(ii) Parameters for a Special Case: Drift-Diffusion SDE

To obtain the **drift-diffusion SDE**, we consider a special case with:

$$a_1(t) = 0, \quad b_1(t) = 0.$$

Thus the SDE reduces to:

$$dX_t = a_2(t)\,dt + b_2(t)\,dW_t, \tag{6}$$

with initial condition:

$$X_0 = x_0. \tag{7}$$

(iii) Solution of the SDE

Since $\mu(X_t, t) = a_2(t)$ and $\sigma(X_t, t) = b_2(t)$ depend only on time and not on X_t, the SDE is solved by **direct integration**:

$$X_t = X_0 + \int_0^t a_2(\tau)\,d\tau + \int_0^t b_2(\tau)\,dW_\tau. \tag{8}$$

(iv) Graphing the Solution

To visualize the stochastic dynamics of the drift-diffusion process, we proceed as follows:

- Choose specific functions for the drift and diffusion terms: $a_2(t) = \mu$ and $b_2(t) = \sigma$, where both μ and σ are constants.
- Use a numerical simulation technique — the Euler–Maruyama method or an exact solution — to generate sample paths of the process.
- Simulate a large number of stochastic trajectories (e.g., 100 paths) over a fixed time interval.
- Overlay these paths on a single graph to observe random fluctuations, trends, and ensemble behavior.
- Additionally, compute and plot the empirical expectation $\mathbb{E}[X_t]$ as a bold reference curve.

Figure 1 displays the result of simulating 100 sample paths of the drift-diffusion SDE using the Python code provided in the Appendix. The black curve represents the analytical mean trajectory $\mathbb{E}[X_t] = X_0 + \mu t$, while the red lines show individual realizations.

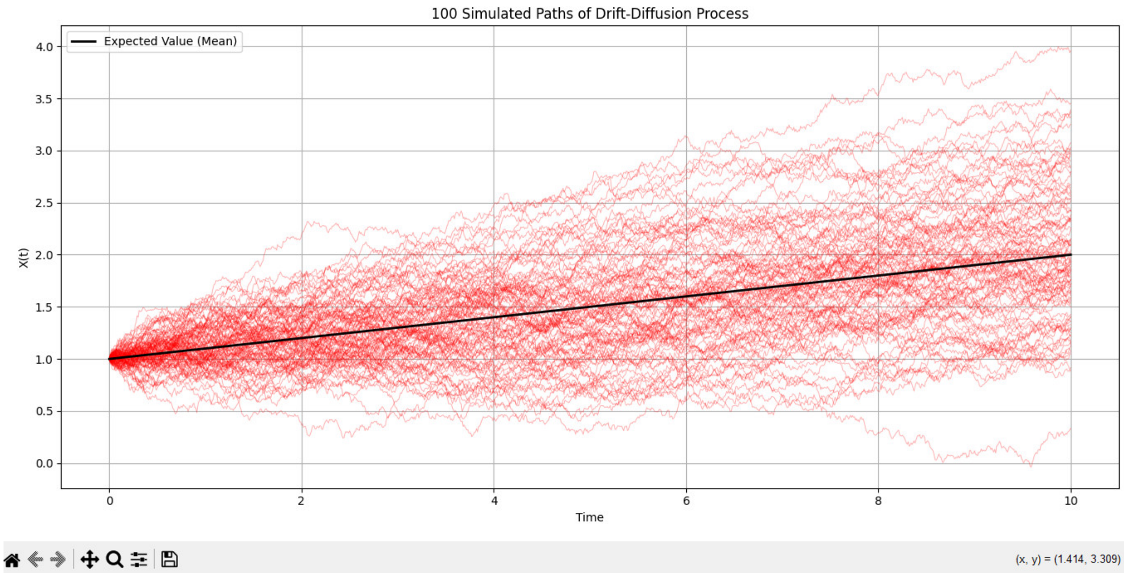

Figure 1: Simulated sample paths of the drift-diffusion SDE with mean overlay

(v) Statistics of the SDE

Expectation of X_t

Taking the expectation of (8) and using linearity of expectation:

$$\mathbb{E}[X_t] = \mathbb{E}[X_0] + \mathbb{E}\left[\int_0^t a_2(\tau)\,d\tau\right] + \mathbb{E}\left[\int_0^t b_2(\tau)\,dW_\tau\right] \tag{9}$$

$$= x_0 + \int_0^t a_2(\tau)\,d\tau, \tag{10}$$

because the stochastic integral has zero mean:

$$\mathbb{E}\left[\int_0^t b_2(\tau)\,dW_\tau\right] = 0.$$

Variance of X_t

The variance of X_t follows from:

$$\mathrm{Var}(X_t) = \mathrm{Var}\left(\int_0^t b_2(\tau)\,dW_\tau\right),$$

since the other terms are deterministic. By Itô isometry:

$$\operatorname{Var}(X_t) = \mathbb{E}\left[\left(\int_0^t b_2(\tau)\, dW_\tau\right)^2\right] - \left[\mathbb{E}\left(\int_0^t b_2(\tau)\, dW_\tau\right)\right]^2 = \tag{11}$$

$$= \mathbb{E}\left[\int_0^t (b_2(\tau))^2\ d\tau\right] - \left[\mathbb{E}\left(Ito\ Integral\right)\right]^2 = \tag{12}$$

$$= \int_0^t (b_2(\tau))^2\ d\tau - [0]^2 = \tag{13}$$

$$= \int_0^t b_2^2(\tau)\, d\tau. \tag{14}$$

by applying Itô isometry and $\mathbb{E}[number] = number$ to first Expectation, followed by $\mathbb{E}[Ito\ Integral] = 0$ to the second Expectation expression.

Covariance and Autocorrelation of X_t

Let $0 \leq s \leq t$. The covariance between X_s and X_t is given by:

$$\operatorname{Cov}(X_s, X_t) = \mathbb{E}[(X_s - \mathbb{E}[X_s])(X_t - \mathbb{E}[X_t])].$$

From the solution:

$$X_t - \mathbb{E}[X_t] = \int_0^t b_2(\tau)\, dW_\tau, \quad X_s - \mathbb{E}[X_s] = \int_0^s b_2(\tau)\, dW_\tau.$$

Using Itô isometry and the fact that increments of Brownian motion are independent, we get:

$$\operatorname{Cov}(X_s, X_t) = \mathbb{E}\left[\left(\int_0^s b_2(\tau)\, dW_\tau\right)\left(\int_0^t b_2(\lambda)\, dW_\lambda\right)\right] \tag{15}$$

$$= \int_0^{\min(s,t)} b_2^2(\tau)\, d\tau. \tag{16}$$

Therefore, the autocorrelation function is:

$$R(s,t) = \operatorname{Cov}(X_s, X_t) = \int_0^{\min(s,t)} b_2^2(\tau)\, d\tau.$$

(vi) Applications of the Drift-Diffusion SDE

Application 1: Neuroscience and Decision-Making Models

The Drift Diffusion Model (DDM) is widely used in neuroscience to model evidence accumulation in binary decision tasks. Here, X_t represents the cumulative evidence.

- Drift $a_2(t)$ reflects decision bias or signal strength. - Diffusion $b_2(t)$ models internal cognitive noise. - Decision occurs when X_t hits a boundary (threshold).

DDM successfully fits behavioral data in cognitive psychology. For example, Ratcliff and McKoon (2008) [1] demonstrated its utility in modeling reaction times and error rates.

Application 2: Particle Motion in Physics

In statistical physics, the drift-diffusion SDE describes the motion of particles suspended in a fluid (e.g., pollen grains in water), under both deterministic drift (gravity, electric fields) and random thermal fluctuations (diffusion).

This process is modeled by:

$$dX_t = \mu\, dt + \sigma\, dW_t,$$

which corresponds to the Langevin equation in physical literature. The solution's distribution evolves according to the Fokker–Planck equation.

Such models are foundational in thermodynamics, statistical mechanics, and modeling molecular dynamics.

References

[1] Ratcliff, R., & McKoon, G. (2008). *The Diffusion Decision Model: Theory and Data for Two-Choice Decision Tasks*. Neural Computation, 20(4), 873–922. https://doi.org/10.1162/neco.2008.12-06-420

Appendix: Python Code for Question 1 (Drift-Diffusion)

The following Python script was used to generate the simulation plot in Part (iv):

Listing 1: Drift-Diffusion Simulation (Q1)

```
import numpy as np
import matplotlib.pyplot as plt

# Parameters
X0 = 1.0          # Initial value
mu = 0.1          # Drift (a_2(t))
sigma = 0.2       # Diffusion (b_2(t))
T = 10            # Total time
N = 1000          # Number of time steps
dt = T / N        # Time step size
t = np.linspace(0, T, N + 1)  # Time grid
M = 100           # Number of sample paths

# Initialize solution matrix
X = np.zeros((M, N + 1))
X[:, 0] = X0

# Simulate paths
for i in range(M):
    dW = np.random.randn(N) * np.sqrt(dt)              # Brownian increments
    W = np.concatenate(([0], np.cumsum(dW)))           # Cumulative Brownian motio
    X[i, :] = X0 + mu * t + sigma * W                   # Drift-diffusion solution

# Analytical mean
X_mean = X0 + mu * t

# Plotting
plt.figure(figsize=(12, 6))
for i in range(M):
    plt.plot(t, X[i, :], color='red', alpha=0.3, linewidth=0.7)

plt.plot(t, X_mean, color='black', linewidth=2, label='Expected Value (Mean)'

# Formatting
plt.title('100 Simulated Paths of Drift-Diffusion Process')
plt.xlabel('Time')
plt.ylabel('X(t)')
plt.grid(True)
plt.legend()
plt.tight_layout()
```

```
plt.show()
```