Thesis 12

Anna Lucia Lamacchia
Statistics

January 12, 2024

Ito Integration and Calculus

1 Concept

Ito Integration, a fundamental concept in stochastic calculus, extends traditional calculus to handle stochastic processes. Unlike deterministic integrals, Ito Integration incorporates the role of Brownian motion, introducing stochastic differentials and Ito's Lemma. This extension is crucial for modeling dynamic systems subject to random fluctuations, particularly in finance, physics, and engineering. Key concepts in Stochastic Calculus are:

1. **Stochastic Differential Equations (SDEs)**: A Stochastic Differential Equation (SDE) involves both deterministic and stochastic components and is expressed as:

   $$dX(t) = \mu(t)dt + \sigma(t)dW(t)$$

   where:

   $$\begin{aligned}
   &X(t) \text{ is the stochastic process,}\\
   &\mu(t) \text{ is the drift term,}\\
   &\sigma(t) \text{ is the volatility term,}\\
   &dt \text{ represents the deterministic time increment,}\\
   &dW(t) \text{ is the differential of a Brownian motion.}
   \end{aligned}$$

2. **Ito's Lemma**: Ito's Lemma is a fundamental result in stochastic calculus, providing a method for differentiating functions of stochastic processes. For a function $f(t, X(t))$, Ito's Lemma is expressed as:

   $$df(t, X(t)) = \left(\frac{\partial f}{\partial t} + \mu(t)\frac{\partial f}{\partial X} + \frac{1}{2}\sigma^2(t)\frac{\partial^2 f}{\partial X^2}\right)dt + \sigma(t)\frac{\partial f}{\partial X}dW(t)$$

2 Didactical Simulation in Python

In this simulation, we generate Brownian motion and use it to simulate a stochastic process involving Ito Integration. The resulting plots illustrate the paths of both Brownian motion and the simulated stochastic process. This simple example provides a visual understanding of how stochastic processes evolve over time through Ito Integration.

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

# Parameters
T = 1.0  # Total time
N = 1000  # Number of time steps
dt = T / N  # Time step size

# Generate Brownian motion increments
dW = np.sqrt(dt) * np.random.normal(size=N)

# Compute the cumulative sum to get Brownian motion
W = np.cumsum(dW)

# Simulate a process involving Ito Integration
t_values = np.linspace(0, T, N)
X = np.cumsum(np.sin(t_values) * dW)

# Plot Brownian motion and the simulated process
plt.figure(figsize=(10, 6))

plt.subplot(2, 1, 1)
plt.plot(t_values, W, label='Brownian Motion')
plt.title('Simulation of Brownian Motion')

plt.subplot(2, 1, 2)
plt.plot(t_values, X, label='Stochastic Process')
plt.title('Simulation of a Stochastic Process with Ito Integration')

plt.tight_layout()
plt.show()
```

Figure 1: Code

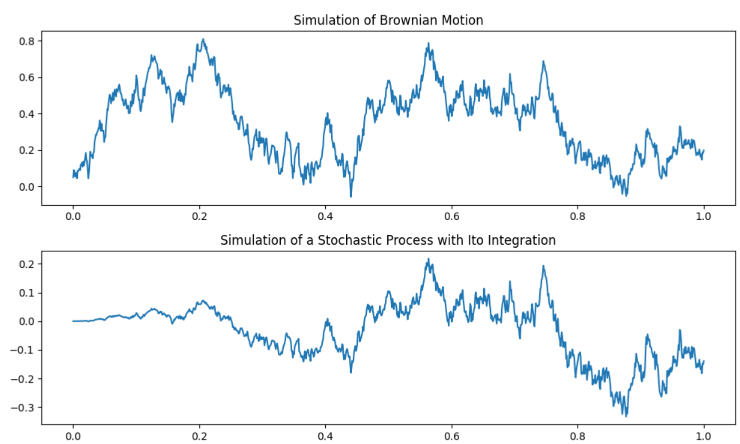

Figure 2: Chart