Ito Calculus

Chiara Iannicelli

November 2023

1 Introduction

Ito integration and calculus are fundamental concepts in stochastic calculus, a branch of mathematics that extends traditional calculus to deal with stochastic processes.

Stochastic calculus is particularly important in the field of mathematical finance, where it provides a rigorous framework for modeling and analyzing financial instruments in uncertain and dynamic environments.

2 Stochastic Differential Equation (SDE):

Ito calculus begins with the formulation of stochastic differential equations (SDEs). An SDE is an equation that involves both deterministic differentials and stochastic differentials.

A standard form of an SDE is given by:

$$dX_t = \mu(t, X_t)dt + \sigma(t, X_t)dW_t$$

where:

- X_t is the stochastic process,
- $\mu(t, X_t)$ is the drift term,
- $\sigma(t, X_t)$ is the diffusion term,
- dt represents the deterministic part,
- dW_t is the stochastic differential associated with a Wiener process.

3 Ito Integral

The Ito integral is a stochastic generalization of the Riemann-Stieltjes integral. The integral is used to solve stochastic differential equations and represents the accumulation of a stochastic process with respect to time.

The Ito integral of a process Y_t with respect to a Wiener process W_t is denoted as:

$$\int_0^t Y_s dW_s$$

4 Ito's Lemma

Ito's Lemma is a fundamental result in stochastic calculus that provides a formula for differentiating a function of a stochastic process.

It is a stochastic version of the chain rule and is crucial for solving SDEs and understanding the dynamics of stochastic processes.

5 Didatical Simulation

Here follows a simulation of a simple example of an Ito integral using Python. We'll simulate a geometric Brownian motion and calculate the Ito integral of the process.

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

# Parameters
mu = 0.1
sigma = 0.2
initial_value = 50.0
num_steps = 1000
total_time = 1.0
dt = total_time / num_steps

# Simulate Geometric Brownian Motion
t_values = np.linspace(0, total_time, num_steps + 1)
W_t = np.random.randn(num_steps) * np.sqrt(dt)
X_t = np.zeros(num_steps + 1)
X_t[0] = initial_value

for i in range(1, num_steps + 1):
    dX = mu * X_t[i - 1] * dt + sigma * X_t[i - 1] * W_t[i - 1]
    X_t[i] = X_t[i - 1] + dX
```

```python

# Calculate Ito Integral
Ito_integral = np.cumsum(X_t[:-1] * W_t) * np.sqrt(dt)

# Plot the results
plt.figure(figsize=(12, 6))

plt.subplot(1, 2, 1)
plt.plot(t_values, X_t, label='Geometric Brownian Motion')
plt.title('Geometric Brownian Motion')
plt.xlabel('Time')
plt.ylabel('Value')
plt.legend()

plt.subplot(1, 2, 2)
plt.plot(t_values[:-1], Ito_integral, label='Ito Integral')
plt.title('Ito Integral of Geometric Brownian Motion')
plt.xlabel('Time')
plt.ylabel('Value')
plt.legend()

plt.tight_layout()
plt.show()

```

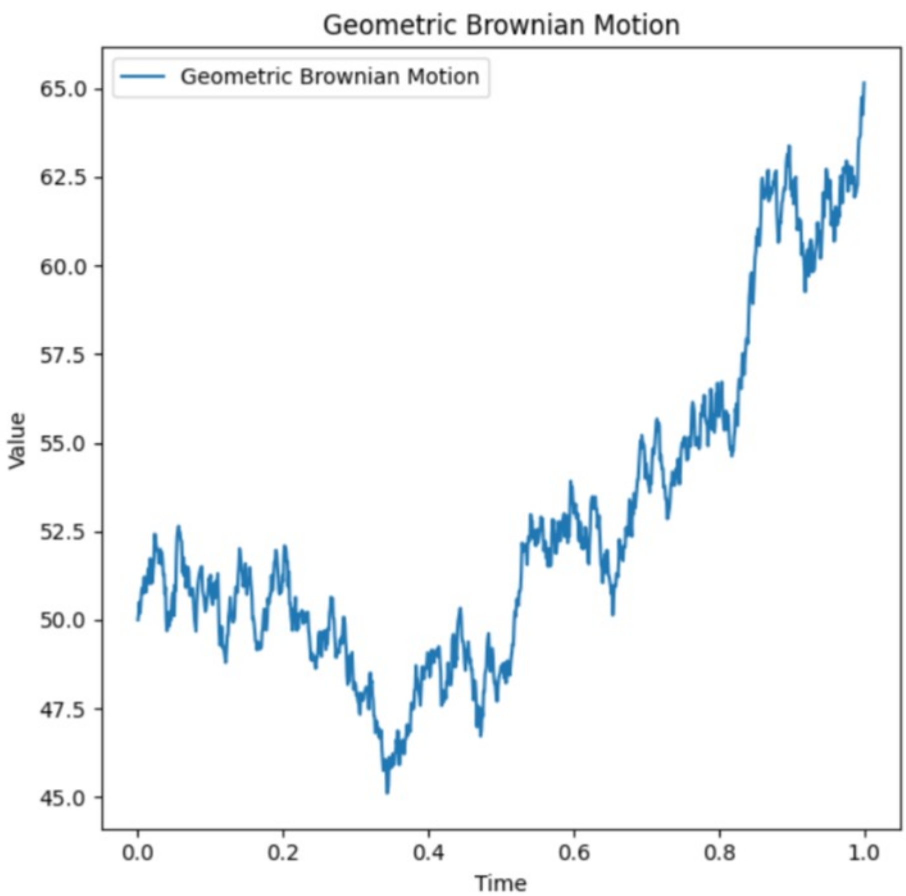

Figure 1: Geometric Brownian Motion

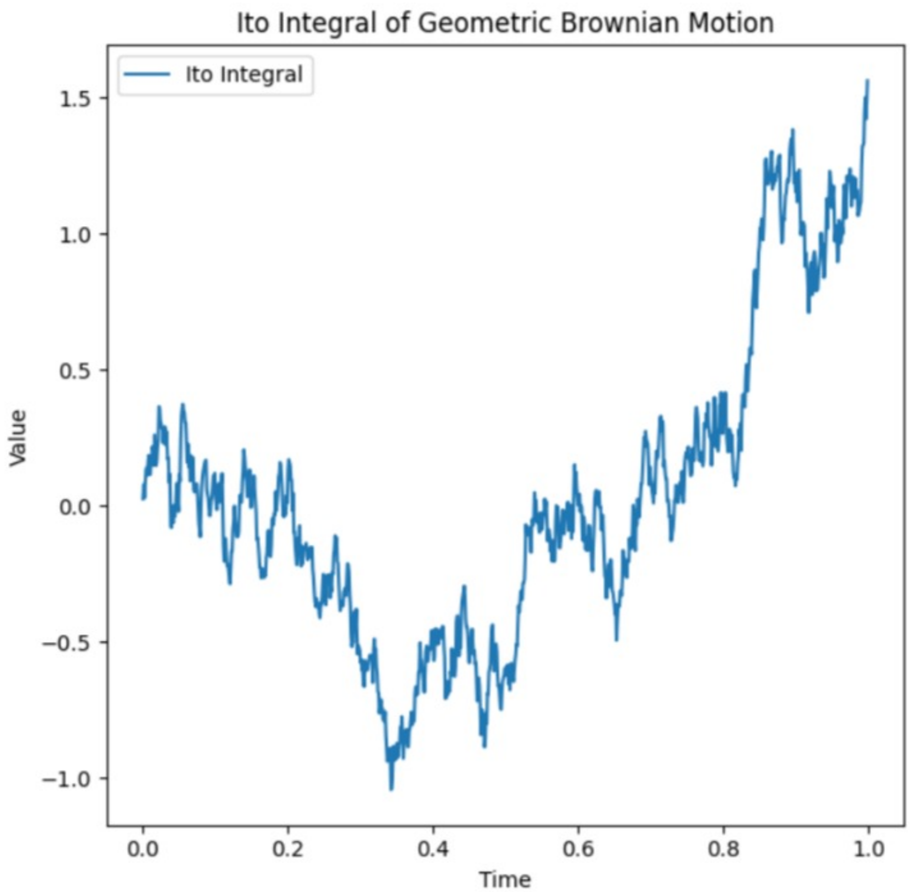

Figure 2: Ito Integral of GBM