Wiener Process and GMB

Chiara Iannicelli

November 2023

1 Introduction

The Wiener process, also known as Brownian motion, is a continuous-time stochastic process named after mathematician Norbert Wiener. It is a fundamental model in probability theory and serves as the foundation for stochastic calculus.

Geometric Brownian Motion is a stochastic process that extends the concept of Brownian motion to model the exponential growth or decay of a quantity over time. It is commonly used to model the dynamics of financial instruments such as stock prices.

2 Wiener Process

Originating from the study of random motion in physical systems, the Wiener process has found applications in diverse fields, ranging from finance and physics to biology and engineering.
Its unpredictable and continuous nature has made it a fundamental tool in understanding the randomness inherent in various natural phenomena.

The Wiener process not only serves as a basis for stochastic calculus but also provides a powerful framework for modeling and analyzing random processes in real-world systems.

2.1 Properties

The Wiener process has the following key properties:

- **Continuous Paths:** The Wiener process has continuous sample paths. This means that, for every fixed realization of the process, the path is a continuous function of time.

- **Independent Increments:** Increments of the process in disjoint time intervals are independent of each other. This property is a consequence of the randomness inherent in the Wiener process.

- **Gaussian Distribution:** At any fixed time, the value of the Wiener process follows a Gaussian (normal) distribution with mean zero and variance equal to the elapsed time.

- **Stationary Increments:** The distribution of the increments only depends on the length of the time interval, not on its starting point.

Mathematically, the Wiener process is often denoted as $W(t)$, and its increments ΔW are normally distributed with mean zero and variance proportional to the time interval Δt.

2.2 Simulation

Simulating a Wiener process in Python involves generating random samples from a normal distribution to represent increments over small time intervals.

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

def simulate_wiener_process(num_steps, total_time):
    dt = total_time / num_steps  # Time step
    t_values = np.linspace(0, total_time, num_steps + 1)  # Time values

    # Generate random increments from a normal distribution
    increments = np.random.normal(loc=0, scale=np.sqrt(dt), size=num_steps)

    # Calculate the cumulative sum to obtain the Wiener process
    wiener_process = np.cumsum(increments)

    # Add the initial value to the Wiener process
    wiener_process = np.insert(wiener_process, 0, 0.0)

    return t_values, wiener_process

# Set the parameters for the simulation
num_steps = 1000
total_time = 1.0

# Simulate the Wiener process
time, wiener_process = simulate_wiener_process(num_steps, total_time)
```

```python

# Plot the results
plt.plot(time, wiener_process, label='Wiener Process')
plt.title('Simulated Wiener Process')
plt.xlabel('Time')
plt.ylabel('Value')
plt.legend()
plt.show()

```

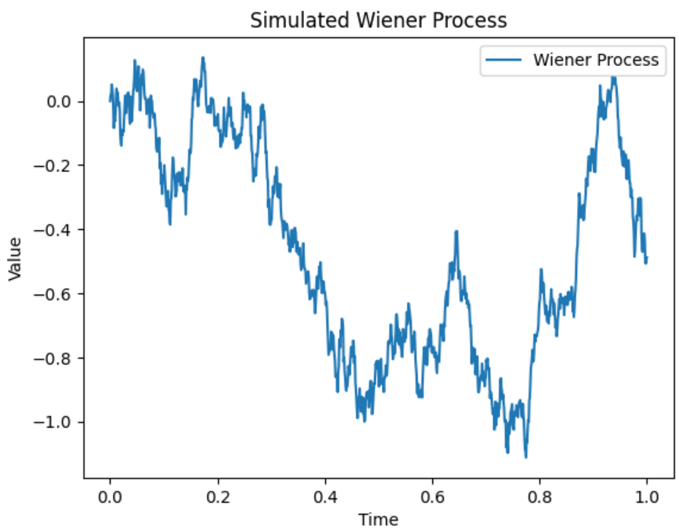

Figure 1: Wiener simulation

3 Geometric Brownian Motion (GBM)

The process is governed by a stochastic differential equation, capturing both a deterministic drift term representing the average growth rate (μ) and a stochastic term proportional to the volatility (σ).

GBM's application extends beyond finance to fields like physics, biology, and environmental science, where exponential growth or decay phenomena are prevalent. Its ability to reflect the inherent uncertainty in dynamic systems has established GBM as a valuable tool for understanding and simulating the complex and often unpredictable behavior of various processes in the natural world.

3.1 Properties

The GBM process is defined by the following stochastic differential equation (SDE):

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

where:

- X_t is the value of the process at time t,
- μ is the drift coefficient (average growth rate),
- σ is the diffusion coefficient (volatility),
- W_t is a Wiener process.

The solution to this SDE is given by:

$$X_t = X_0 e^{\mu - \frac{\sigma^2}{2} t + \sigma W_t}$$

Key features of GBM include exponential growth (or decay) and the fact that the logarithm of the process follows a Brownian motion. The parameters μ and σ govern the average growth rate and volatility of the process, respectively.

3.2 Simulation

Simulating a Geometric Brownian Motion (GBM) involves iteratively updating the value of the process based on its stochastic differential equation.

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

def simulate_gbm(num_steps, total_time, initial_value, mu, sigma):
    dt = total_time / num_steps  # Time step
    t_values = np.linspace(0, total_time, num_steps + 1)  # Time values

    # Generate random increments from a normal distribution
    increments = np.random.normal(loc=mu * dt, scale=sigma * np.sqrt(dt), size=num_steps)

    # Initialize an array to store GBM values
    gbm_values = np.zeros(num_steps + 1)
    gbm_values[0] = initial_value

    # Simulate the GBM process
    for i in range(1, num_steps + 1):
```

```python
        gbm_values[i] = gbm_values[i - 1] * np.exp(increments[i - 1])

    return t_values, gbm_values

# Set the parameters for the simulation
num_steps = 1000
total_time = 1.0
initial_value = 50.0
mu = 0.1
sigma = 0.2

# Simulate the Geometric Brownian Motion
time, gbm_values = simulate_gbm(num_steps, total_time, initial_value, mu, sigma)

# Plot the results
plt.plot(time, gbm_values, label='GBM')
plt.title('Simulated Geometric Brownian Motion')
plt.xlabel('Time')
plt.ylabel('Value')
plt.legend()
plt.show()


```

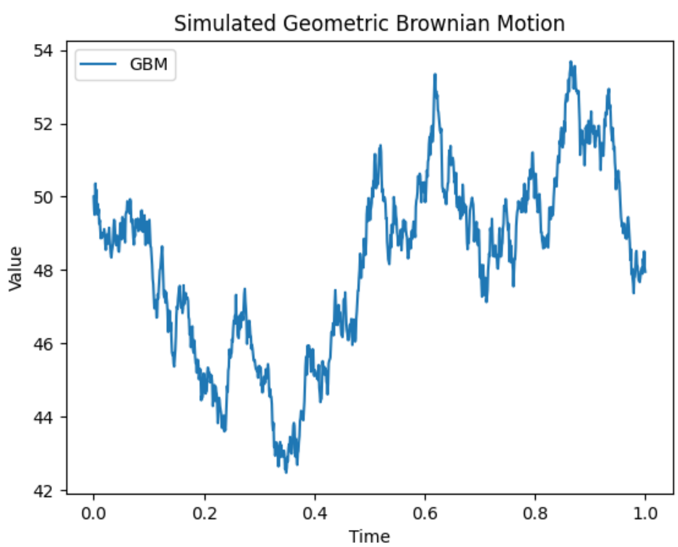

Figure 2: GBM Simulation