Assignment 3

Reid Ginoza

2/23/2020

Simulation

The analytical problem below asks about the following: Let $W(\cdot)$ be one-dimensional Brownian motion. Then let

1. $W_{\text{shift}} := W(t+s) - W(s)$ for all $s \geq 0$
2. $W_{\text{scale}} := cW\left(\dfrac{t}{c^2}\right)$ for all $c > 0$

Later, we will show that both of the above results in Brownian motion. Before looking at the math, we will see what they look like when simulated. First, I will simulate shifted one-dimensional Brownian motion. All code is written in `python`.

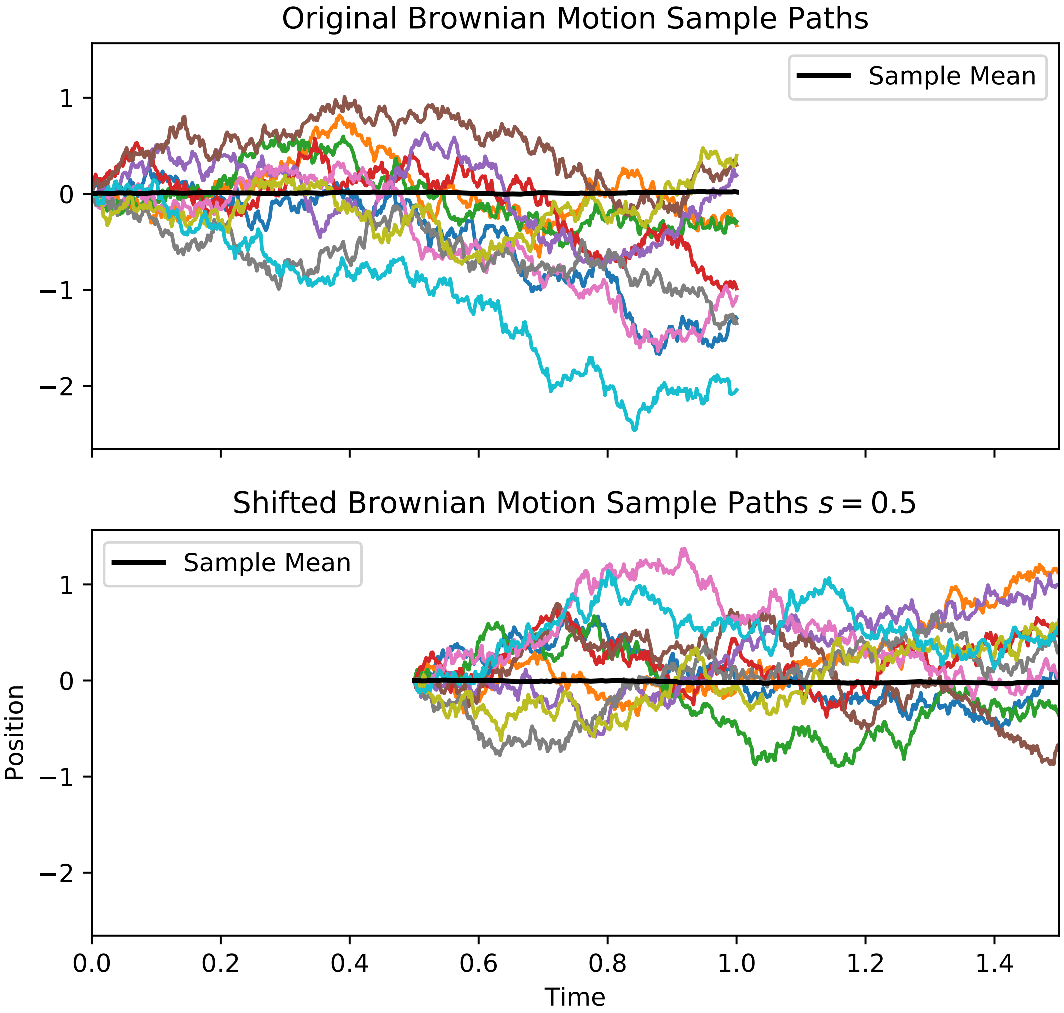
Shifted Brownian Motion

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


def plot_on_axis(ax, time, pos, cols, title, with_mean=False):
    ax.plot(time, pos[:, cols])
    ax.set_title(title)
    if with_mean:
        ax.plot(time, pos.mean(axis=1), color='black',
            label='Sample Mean', linewidth=2)
        ax.legend()


def multiple_brownian_motion(end_time=1., num_tsteps=500, n_trials=1000):
    dt = (end_time - 0) / num_tsteps
    dw = np.random.normal(scale=np.sqrt(dt), size=(num_tsteps+1, n_trials))
    # Brownian motion must start at time 0 with value 0
    dw[0] = np.zeros_like(dw[0])
    w = dw.cumsum(axis=0)
    t = np.linspace(0, end_time, num=num_tsteps+1)  # not used in calculations
    assert w.shape[0] == t.shape[0], f'time and position arrays are not the same length. w.shape[0] - t
    assert w.shape == dw.shape, f'position and velocity arrays are not the same shape: w.shape: {w.shap
    return t, w, dw


def shift_brownian_motion(time_length=1., num_tsteps=500, n_trials=1000, shift=0.5):
    """
    This function calculates the shift with time starting at 0
    and the first value at shift
    """
    dt = time_length / num_tsteps
    end_time = shift + time_length
    assert (end_time / dt).is_integer(), 'adjust end_time and shift so that (end_time / dt).is_integer(
    total_steps = int(end_time // dt) + 1
    dw = np.random.normal(scale=np.sqrt(dt), size=(total_steps, n_trials))

    # Brownian motion must start at time 0 with value 0
    dw[0] = np.zeros_like(dw[0])
    w = dw.cumsum(axis=0)
    t = np.linspace(0, end_time, num=total_steps)

    # shift occurs here
    w = w[t >= shift]
    w -= w[0]
    t = t[t >= shift]

    assert w.shape[0] == t.shape[0], f'time and position arrays are not the same length. w.shape[0] - t
    return t, w
```

```
END_TIME = 1.
NUM_TSTEPS = 500
N_TRIALS = 1000
PLOTS = True
N_PATHS = 10
```

Although the Brownian Motion is shifted, it still looks like Brownian motion.

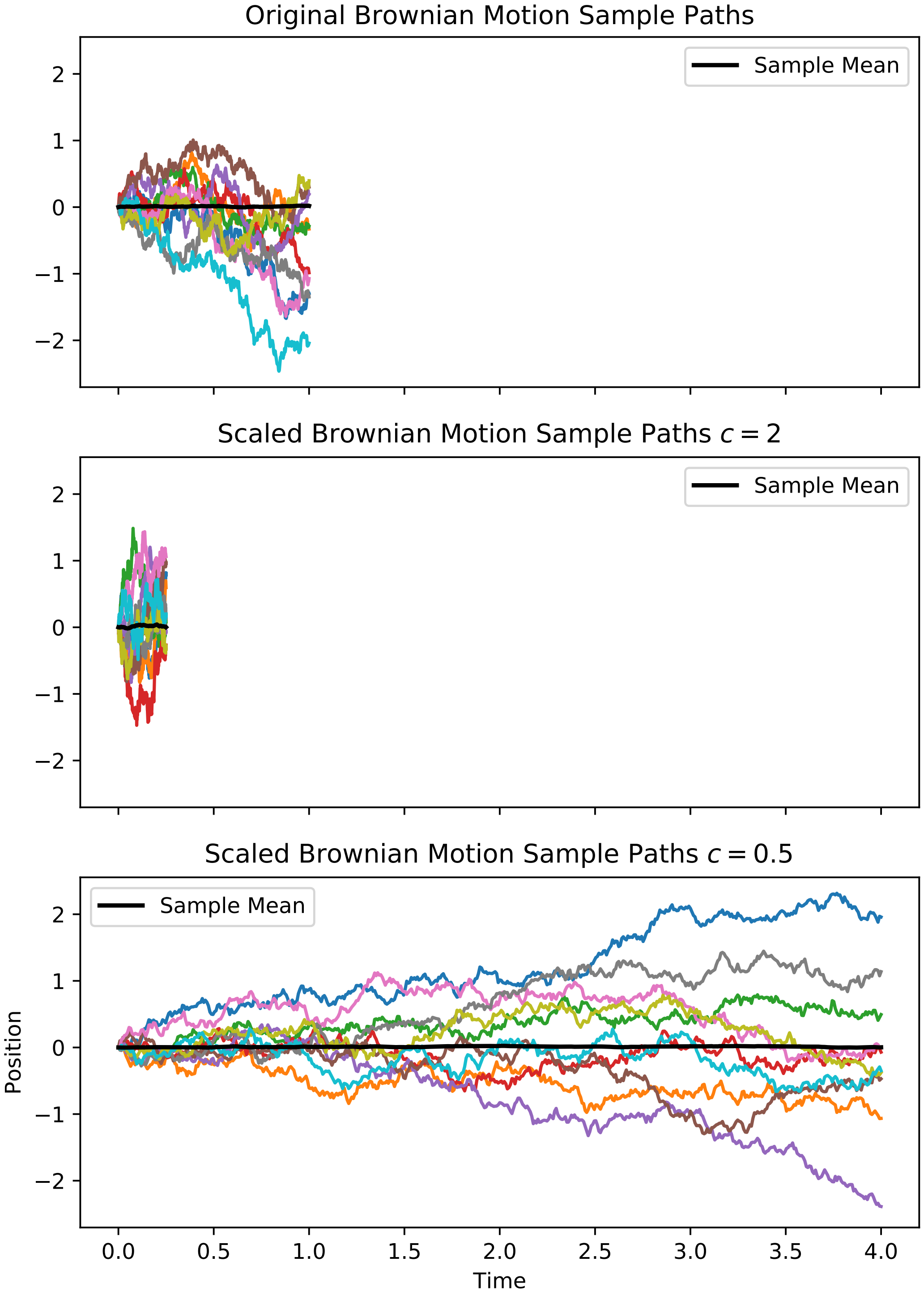
```
orig_time, orig_position, orig_velocity = multiple_brownian_motion(
    end_time=1., num_tsteps=NUM_TSTEPS, n_trials=N_TRIALS)
shift_time, shift_position = shift_brownian_motion(
    time_length=1., num_tsteps=NUM_TSTEPS, n_trials=N_TRIALS, shift=0.5)

if PLOTS:
    # select random paths since showing all N_TRIALS would be visually crowded.
    columns = np.arange(orig_position.shape[1])
    np.random.shuffle(columns)
    plot_columns = columns[:N_PATHS]

    fig, axs = plt.subplots(2, sharex=True, sharey=True, figsize=(7, 6.5))
    plot_on_axis(axs[0], orig_time, orig_position,
                 plot_columns, 'Original Brownian Motion Sample Paths',
                 with_mean=True)
    plot_on_axis(axs[1], shift_time, shift_position,
                 plot_columns, r'Shifted Brownian Motion Sample Paths $s=0.5$',
                 with_mean=True)
    axs[1].set_xlim(0, shift_time.max())
    plt.xlabel('Time')
    plt.ylabel('Position')
    plt.show()
```


Original Brownian Motion Sample Paths
Sample Mean
1
0
−1
−2
Shifted Brownian Motion Sample Paths $s = 0.5$
Sample Mean
1
0
−1
−2
Position
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
Time

Scaled Brownian Motion

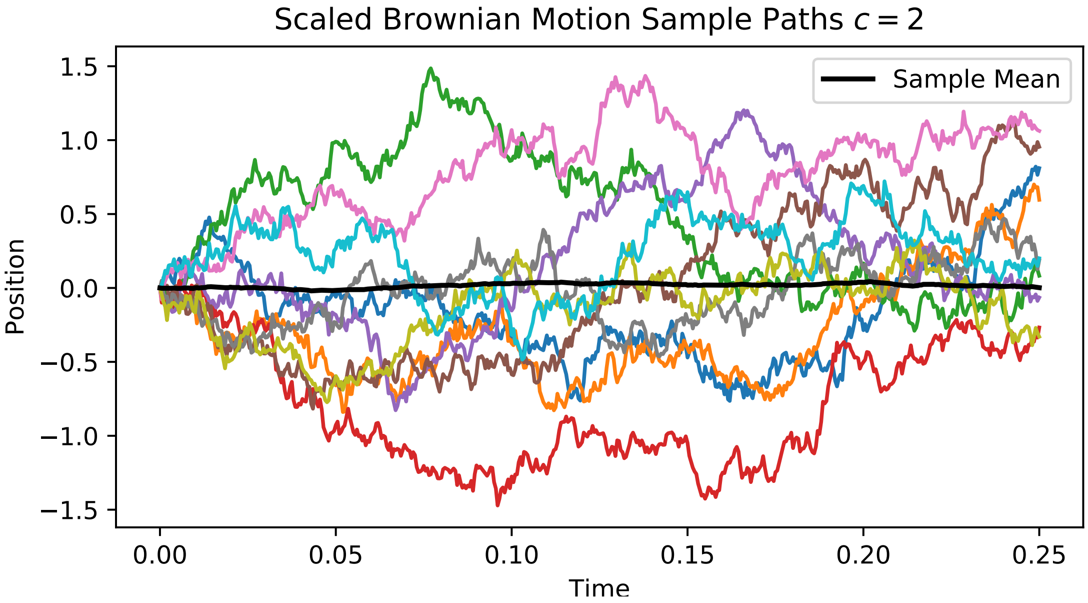
```
def scale_brownian_motion(end_time=1., num_tsteps=500, n_trials=1000, scale=0.5):
    """
    This function calculates the scaling
    """
    dt = (end_time - 0) / num_tsteps / scale**2
    dw = np.random.normal(scale=np.sqrt(dt), size=(num_tsteps + 1, n_trials))
    # Brownian motion must start at time 0 with value 0
    dw[0] = np.zeros_like(dw[0])
    w = dw.cumsum(axis=0) * scale
    t = np.linspace(0, end_time, num=num_tsteps + 1) / scale**2
    assert w.shape[0] == t.shape[0], f'time and position arrays are not the same length. w.shape[0] - t
    assert w.shape == dw.shape, f'position and velocity arrays are not the same shape: w.shape: {w.shape
    return t, w, dw
```

```python
scale_small_time, scale_small_position, scale_small_vel = scale_brownian_motion(
    end_time=1., num_tsteps=NUM_TSTEPS, n_trials=N_TRIALS, scale=2
)
scale_big_time, scale_big_position, scale_big_vel = scale_brownian_motion(
    end_time=1., num_tsteps=NUM_TSTEPS, n_trials=N_TRIALS, scale=0.5
)

if PLOTS:
    fig, axs = plt.subplots(3, sharex=True, sharey=True, figsize=(7, 10))
    plot_on_axis(axs[0], orig_time, orig_position,
                 plot_columns, 'Original Brownian Motion Sample Paths',
                 with_mean=True)
    plot_on_axis(axs[1], scale_small_time, scale_small_position,
                 plot_columns, r'Scaled Brownian Motion Sample Paths $c=2$',
                 with_mean=True)
    plot_on_axis(axs[2], scale_big_time, scale_big_position,
                 plot_columns, r'Scaled Brownian Motion Sample Paths $c=0.5$',
                 with_mean=True)

    plt.xlabel('Time')
    plt.ylabel('Position')
    plt.show()
```


Original Brownian Motion Sample Paths
Sample Mean
2
1
0
−1
−2
Scaled Brownian Motion Sample Paths $c = 2$
Sample Mean
2
1
0
−1
−2
Scaled Brownian Motion Sample Paths $c = 0.5$
Sample Mean
2
1
0
−1
−2
Position
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Time

And out of curiosity, I plotted the horizontally compressed Brownian motion on its own:

```python
if PLOTS:
    fig, ax = plt.subplots(1, sharex=True, sharey=True, figsize=(7, 3.5))
    plot_on_axis(ax, scale_small_time, scale_small_position,
                 plot_columns, r'Scaled Brownian Motion Sample Paths $c=2$',
                 with_mean=True)
    plt.xlabel('Time')
    plt.ylabel('Position')
    plt.show()
```



Analytical Problem

From (Evans 2012), Exercise (23.): Show that if $\mathbf{W}(\cdot)$ is an n-dimensional Brownian motion, then so are

1. $\mathbf{W}(t+s) - \mathbf{W}(s)$ for all $s \geq 0$
2. $c\mathbf{W}\left(\frac{t}{c^2}\right)$ for all $c > 0$

My Solution

I studied only the one-dimensional case for the above, guided by (Dunbar 2010). Each section will have three subsections corresponding to the three conditions in the definition of Brownian Motion. The three conditions to be satisfied for the definition of Brownian Motion $W(\cdot)$ are:

1. $W(0) = 0$,
2. $W(t) - W(s) \sim N(0, t-s)$ for all $t \geq s \geq 0$,
3. For all times $0 = t_0 < t_1 < t_2 < \ldots < t_n$, the random variables $W(t_1) - W(t_0), W(t_2) - W(t_1), \ldots, W(t_n) - W(t_{n-1})$ are independent.

One-dimensional Shifting

The first problem studies $W_{\text{shift}}(t) := W(t+s) - W(s)$ for a fixed $s \geq 0$.

Initial Condition

Fix $s \geq 0$. Then,

$$W_{\text{shift}}(0) = W(0+s) - W(s) = W(s) - W(s) = 0 \quad \text{almost surely.}$$

Increments are normally distributed

Fix $s \geq 0$. Then,

$$\begin{aligned}
W_{\text{shift}}(t_2) - W_{\text{shift}}(t_1) &= W(t_2+s) - W(s) - (W(t_1+s) - W(s)) && (1)\\
&= W(t_2+s) - W(t_1+s) && (2)\\
&\sim N(0, (t_2+s) - (t_1+s)) = N(0, t_2 - t_1) && (3)
\end{aligned}$$

Increments are Independent

Fix $s \geq 0$. Let there be $m+1$ time steps such that $t_0 + s < t_1 + s < t_2 + s, \ldots < t_m + s$. Then for any $i, j \in \{1, \ldots, m\}$ with $i < j$:

$$W_{\text{shift}}(t_j) - W_{\text{shift}}(t_{j-1}) = W(t_j+s) - W(s) - (W(t_{j-1}+s) - W(s)) = W(t_j+s) - W(t_{j-1}+s) \quad (4)$$
$$W_{\text{shift}}(t_i) - W_{\text{shift}}(t_{i-1}) = W(t_i+s) - W(s) - (W(t_{i-1}+s) - W(s)) = W(t_i+s) - W(t_{i-1}+s) \quad (5)$$

and by definition of $W(\cdot)$, $W(t_j+s) - W(t_{j-1}+s)$ and $W(t_i+s) - W(t_{i-1}+s)$ are independent.

One-dimensional Scaling

This problem studies $W_{\text{scale}}(t) := cW\left(\frac{t}{c^2}\right)$ for a fixed $c > 0$.

Initial Condition

Fix $c > 0$. Then,

$$W_{\text{scale}} = cW\left(\frac{0}{c^2}\right) = cW(0) = 0 \quad \text{almost surely.}$$

Increments are normally distributed

Fix $c > 0$. First, we know from the definition of $W(\cdot)$ that $W(t_2) - W(t_1)$ is normally distributed with mean 0 and variance $t_2 - t_1$. Thus, $W_{\text{shift}}(t_2) - W_{\text{shift}}(t_1)$ is a normally distributed random variable times a scalar.

$$\begin{aligned}
W_{\text{shift}}(t_2) - W_{\text{shift}}(t_1) &= cW\left(\frac{t_2}{c^2}\right) - cW\left(\frac{t_1}{c^2}\right) \tag{6}\\
&= c\left(W\left(\frac{t_2}{c^2}\right) - W\left(\frac{t_1}{c^2}\right)\right) \tag{7}\\
&\sim cN\left(0, \frac{(t_2 - t_1)}{c^2}\right) = N\left(0, c^2\frac{(t_2 - t_1)}{c^2}\right) = N(0, t_2 - t_1) \tag{8}
\end{aligned}$$

This last step is due to the fact that, in general, for a random variable X with finite mean and variance, $\text{E}(cX) = c\text{E}(X)$ and $\text{Var}(cX) = c^2\text{Var}(X)$.

Increments are independent

Fix $c > 0$. Let there be $m + 1$ time steps such that $\frac{t_0}{c^2} < \frac{t_1}{c^2} < \frac{t_2}{c^2}, \ldots < \frac{t_m}{c^2}$. Then for any $i, j \in \{1, \ldots, m\}$ with $i < j$:

$$\begin{aligned}
W_{\text{scale}}(t_j) - W_{\text{scale}}(t_{j-1}) &= c\left(W\left(\frac{t_j}{c^2}\right) - W\left(\frac{t_{j-1}}{c^2}\right)\right) \tag{9}\\
W_{\text{scale}}(t_i) - W_{\text{scale}}(t_{i-1}) &= c\left(W\left(\frac{t_i}{c^2}\right) - W\left(\frac{t_{i-1}}{c^2}\right)\right) \tag{10}\\
& \tag{11}
\end{aligned}$$

and by definition of $W(\cdot)$, $W\left(\frac{t_j}{c^2}\right) - W\left(\frac{t_{j-1}}{c^2}\right)$ and $W\left(\frac{t_i}{c^2}\right) - W\left(\frac{t_{i-1}}{c^2}\right)$ are independent, and independent random variables multiplied by a scalar are also independent.

References

Dunbar, Steve. 2010. “Math 489/Math 889 Stochastic Processes and Advanced Mathematical Finance: Transformations of Brownian Motion.” 2010. http://www.math.unl.edu/~sdunbar1/MathematicalFinance/Lessons/BrownianMotion/Transformations/transformations.pdf.

Evans, Lawrence C. 2012. *An Introduction to Stochastic Differential Equations*. Vol. 82. American Mathematical Soc.