Direct Integration

November 26, 2022

[1]:
```
# numerical libraries
import numpy as np
import scipy
import matplotlib.pyplot as plt
%matplotlib inline
```

Path-sampling approximation requires us to simulate:

$$I_{(1,2)}[1]_{0,1} = \int_0^1 W_s^1 dW_s^2$$

[3]:
```
np.random.seed(10)
nmc = 10000
mean2 = 0
for _ in range(nmc):
    # first Wiener process
    dt = 0.001
    Delta = 1
    t = np.arange(0, Delta+dt, dt)
    Nt = len(t)
    dW1 = np.sqrt(dt) * np.random.randn(Nt)
    W1 = np.cumsum(dW1)
    # second Wiener process
    dW2 = np.sqrt(dt) * np.random.randn(Nt)
    # integration
    I_12 = np.sum(W1 * dW2)
    # final result
    mean2 += W1[-1] * I_12/nmc
mean2
```

[3]:
```
0.006803603192876089
```

We see that both methods approximate the true mean 0.