Simulation of the geometric Brownian motion

March 20, 2022

1 Importing some basic libraries

[1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

2 Defining the parameters, *Sigma* is the volatility

[16]:
```
mu = 0.1
n = 100
T = 1
M = 100
S0 = 100
sigma = 0.3
```

3 Simulation of the geometric Brownian motion

[17]:
```
dt = T/n
St = np.exp((mu-sigma**2/2)*dt+sigma*np.transpose(np.random.normal(0,np.
 ↪sqrt(dt), size = (M,n))))

St = np.vstack([np.ones(M), St])

St = S0*St.cumprod(axis = 0)
```

4 Viewing one of the stacks

[18]:
```
St[1]
```

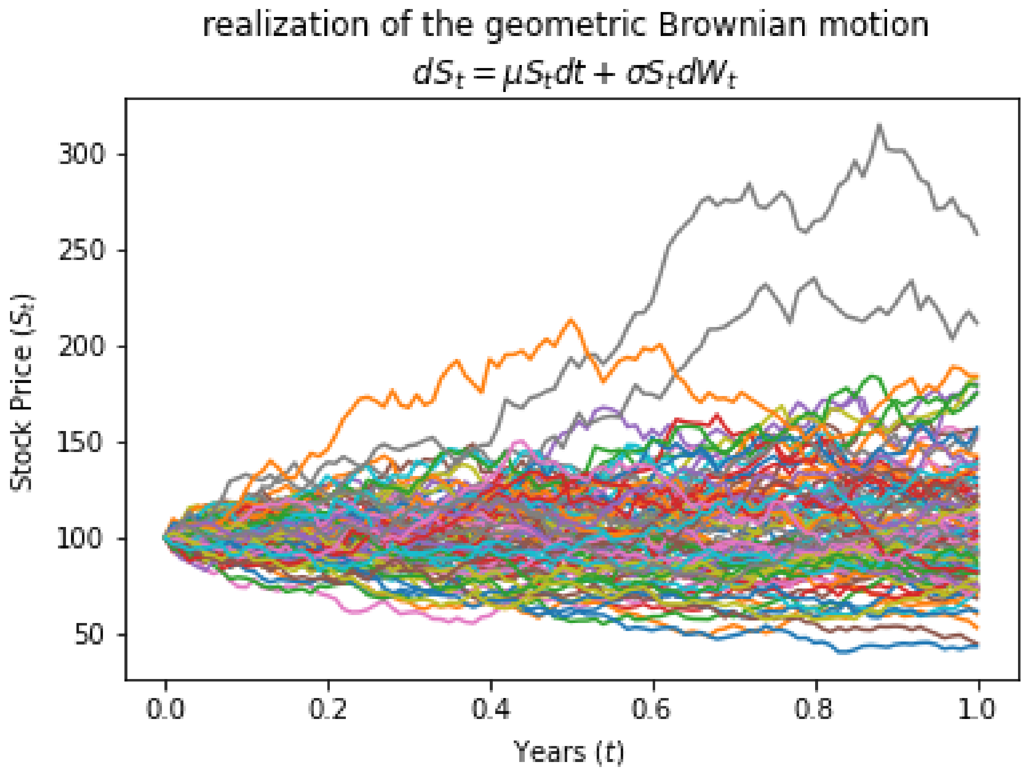
[18]:
```
array([102.04420083, 109.80626616,  98.99541203,  97.1118513 ,
        96.47168329, 100.61905122, 100.83590932, 101.3369178 ,
       104.8569994 , 102.5197927 ,  98.95257577,  99.26049979,
        97.85937488,  92.55197053,  99.44987543, 101.22146811,
        98.10633137, 100.15463116,  95.75956377,  98.03851726,
        99.66204828,  97.29506211,  97.14142841, 101.15790961,
```

```
        100.00487378,  99.35831172, 100.54026921, 100.2766185 ,
         99.56286846, 100.80857454,  98.50813018,  99.72674018,
         96.40209878, 102.35217527,  96.19300915,  99.85957292,
        101.98500138, 101.97272516, 100.81674954, 100.75452755,
         97.86216117, 102.09667759,  96.1109805 ,  99.80137208,
         99.22531014, 103.45569586,  99.55350729, 100.27390119,
        101.40905318, 102.99687868, 100.74523207, 100.49999279,
         99.30312908, 101.90076629,  99.87094821,  99.20456694,
         95.72725768,  98.57237388,  98.09482475, 103.63000336,
        102.71246968, 100.73714009,  95.87050297,  99.69294066,
         94.02464647, 100.39079069,  99.33674463,  97.58274558,
         97.57290311, 100.05504523, 101.00057507,  99.39657056,
         94.39484908,  99.88578207, 100.09119026,  96.93794753,
         98.58425622, 100.03116484,  99.47307778, 102.62004579,
        107.14331129, 100.10441481, 101.04890175,  99.77566073,
        102.47934761,  98.73213513,  99.77344284, 101.74784507,
        102.86310998,  99.84713645,  97.7353472 , 102.63860285,
         94.77715796,  97.8544595 , 104.57940974,  96.05937354,
         99.67370986, 100.97860494, 102.48682482,  96.49112272])
```

Time intervals in years

[19]:
```
time = np.linspace(0,T,n+1)
array = np.transpose(np.full((M,n+1), fill_value = time))
```

[20]:
```
plt.plot(array, St)
plt.xlabel("Years $(t)$")
plt.ylabel("Stock Price $(S_t)$")
plt.title("realization of the geometric Brownian motion \n $dS_t = \mu S_tdt␣
 ↪+\sigma S_t dW_t$")
plt.show()
```


realization of the geometric Brownian motion
$dS_t = \mu S_t dt + \sigma S_t dW_t$
Stock Price (S_t)
300
250
200
150
100
50
0.0
0.2
0.4
0.6
0.8
1.0
Years (t)

```
[ ]:
```