Lab Number : **08**
Due Date : **Nov 04, 2020**
Student Details :

- Name : **AB Satyaprakash**
- Roll Number : **180123062**
- Department : **Mathematics and Computing**

Formulae used:
Apart from the ones needed for estimation of μ and σ, we have used the following formulae/algorithms in the code

1. $N(\mu, \sigma^2) = \mu + \sigma N(0,1)$
2. Box-Muller algorithm to generate N(0,1)
3. $Rj+1 = -log(U)/\lambda$
4. $X(0) = log(S(0))$
5. $X(\tau j+1) = X(\tau j) + (\mu - 1/2\sigma^2)Rj+1 + \sigma\sqrt{Rj+1}\, Zj+1 + log\ Yj$
6. $\tau j+1 = \tau j + Rj+1$

Question :

Explanation:

----------Previous Assignment Info ----------

First of all we used **pandas** (a python library) to read the csv file, i.e. the file having the daily adjusted closing stock prices for SBI for **July 1, 2020 to Sept 30, 2020**.

Then we moved on to the specific column of the data and stored it as a list. This list is then used to generate the **Ui**s given by natural logarithm of ratio of the current and previous day stock prices (more like previous entry, since days were missing in the csv accounting for non-working days).

Using the formulae given in the question pdf, corresponding to the gBm model, we obtained the **estimated** values of μ and σ.

After obtaining the values of μ and σ, we calculate values for 3 parameters (lists, each with 1000 values) namely,

a. Z using box muller algorithm
b. Y using the formula 1 and then taking exponential value
c. R using formula 3

Now having the values for all these parameters, we can proceed, and obtain the value of X recursively using formula 5 (with X(0) being calculated as per formula 4), while we keep updating time as per formula 6. Also note that, since we have used the jump-point method, we need to stop our loop when we have actually reached 1000 time points. Now for each case, we can correspondingly find the S (stock prices), using the fact that X=log(S), or S=exp(X).

Finally we plot the S list obtained vs the time-point list obtained and get our required graphs!

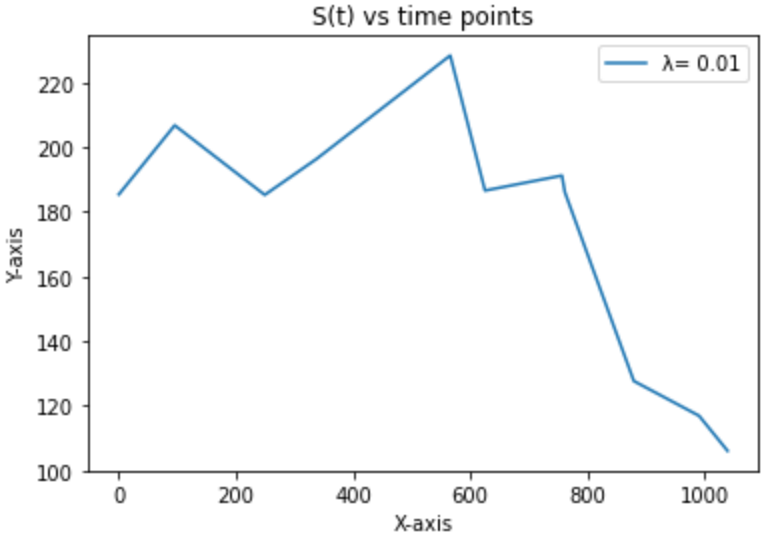
Output:

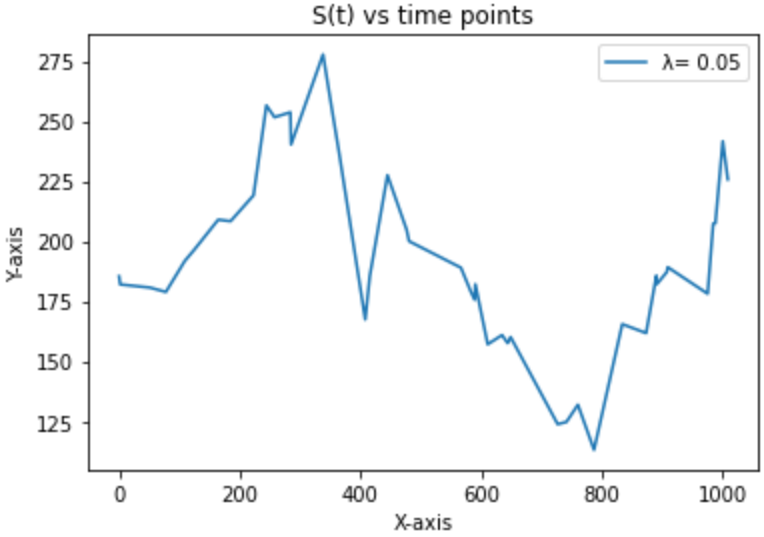
*Please make sure that the file '**SBIN.NS.csv**' is in the same directory as well for pandas to do its work.*
*Also, the following command will install **pandas** on the linux system. One can use pip if one doesn't use python3.*

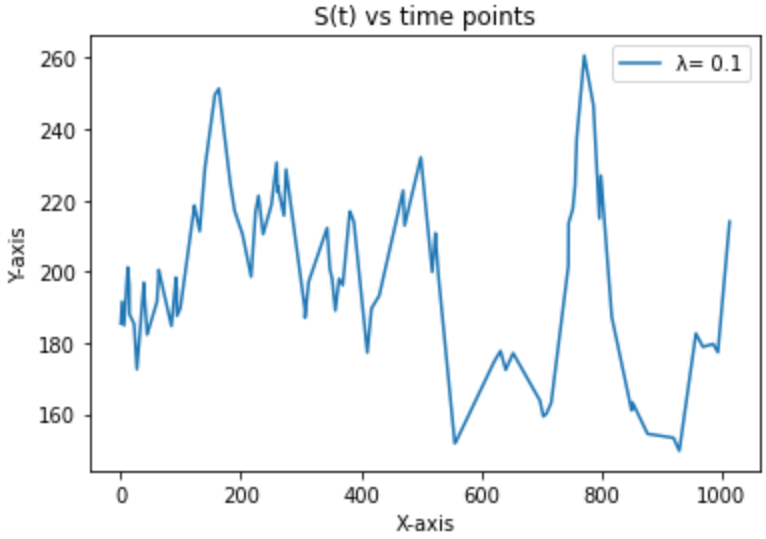
```
pip3 install pandas
```

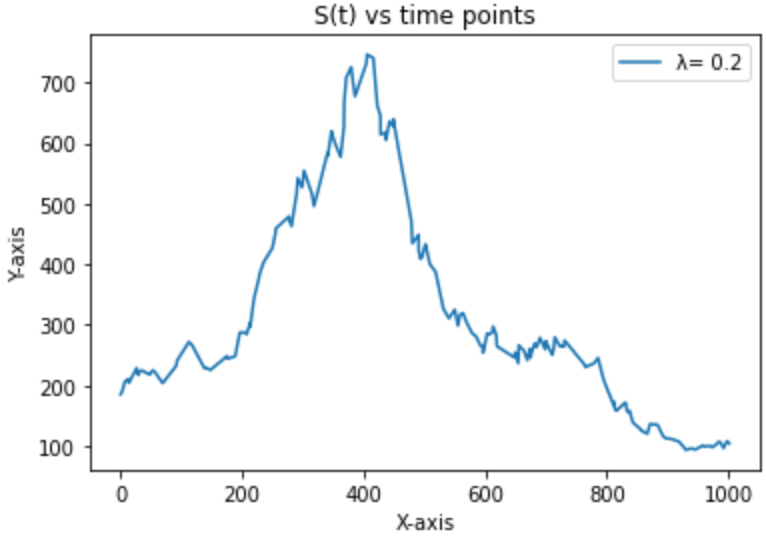
The following output is obtained in the terminal after running the program **180123062_AB.py. (also find the 4 attached plots)**

```
lord:Lab 7 imperial_lord$ python3 180123062_ABSatyaprakash.py
Estimated value of μ (mean) = 0.0002981060700200034
Estimated value of σ (standard deviation) = 0.022281727058705555
```


S(t) vs time points
λ= 0.01
Y-axis
X-axis
100
120
140
160
180
200
220
0
200
400
600
800
1000


S(t) vs time points
λ= 0.05
Y-axis
X-axis
125
150
175
200
225
250
275
0
200
400
600
800
1000





Also note that the graphs won't be exactly the same in each case we run the program, since the value of S(t) is actually governed by randomly generated values!