The simulation of equity returns properties using GBM, and URN models

Senyuan Liu [a*]

[a*]Dept. of Statistics and Applied Probability, University of California, Santa Barbara

Goleta, California, CA USA 93106

[*] Corresponding author: senyuan@ucsb.edu

Supervisor/Professor email: Alexander Shkolnik (shkolnik@ucsb.edu)

Abstract

We have been presented the properties of asset return by simulation within the empirical data. However, is it possible to illustrate properties by statistical analysis? Most currently existing models fail to reproduce all these statistical features. In this paper, we will elaborate the properties by applying different statistical models: Geometric Brownian Motion and Ehrenfest URN. We will focus on the following properties: distributional properties, tail properties and extreme fluctuations, path-wise regularity, linear and nonlinear dependence of returns in time and across stocks. In this project, I will use S&P500 index return as the data and apply it with the models to compare the results with empirical data.

1. Geometric Brownian Model (GBM)

Scottish Biologist Robert Brown discovered Brownian motion when he studied pollen particles floating in water through the microscope. Brownian motion has contributed a lot in thermodynamics foundations and it is also one of the key component of statistical physics.

Brownian motion is also widely used in finance when modeling random behavior that evolves over time.

Before applying the Geometric Brownian Model to the data, I first plot the graph of empirical data of sp500 index return, as shown in Figure 1.1. To easily compare the data in different periods, I also divide the time periods into several parts in a time interval of 10 years. The results are presented in two ways: Line chart and histogram, corresponding to Figure 1.2 and Figure 1.3

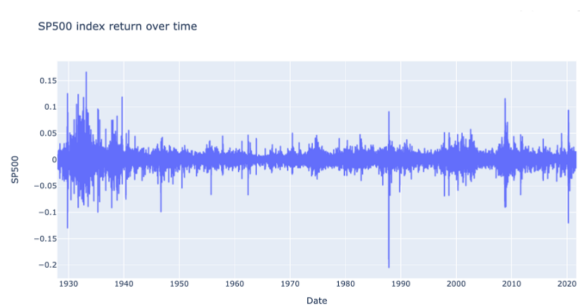
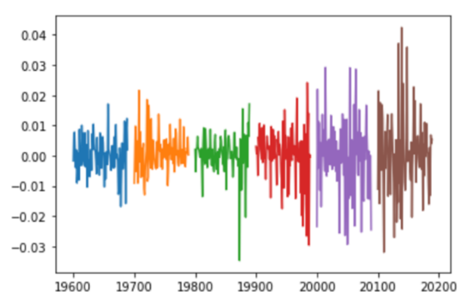


Figure 1.1: SP500 index return



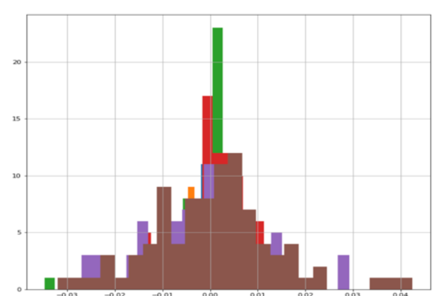


Figure 1.2

Figure 1.3

Definition 1.1

Geometric Brownian Motion: the formula is:

$$S_t = S_0 e^{((\mu-\sigma^2/2)t+\sigma W_t)}$$

Here, St represents the stock price at time t, and in the case of SP500, μ (return of stock) is roughly 9% annually and σ (volatility)is roughly 16% annually. In the lecture, we have proven that μ is the return of stock and σ is essentially the volatility. To test whether μ is roughly about 9%, we could apply the formula:

$$\mu = \frac{1}{n}\sum_{k=1}^{n}\left(\frac{S_{t_k} - S_{t_{k-1}}}{S_{t_{k-1}}}\right)$$

And in annually:

$$\sigma_{annually} = \left(1 + \sigma_{daily}\right)^{252} - 1$$

To test whether σ is roughly about 16%, we could apply the formula:

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{k=1}^{n}\left(\frac{S_{t_k} - S_{t_{k-1}}}{S_{t_{k-1}}} - \mu\right)^2}$$

And in annually:

$$\sigma_{annually} = \sqrt{252}\sigma_{daily}$$

I use python to test it and here are the results Figure 1.4 and Figure 1.5:

```python
sp500_list = []
for i in range(len(df['SP500'])):
    sp500_list.append(df['SP500'][i])
#sp500_list
miu = sum(sp500_list)/len(sp500_list)
miu_annually = ((1+miu)**252)-1
print("The annually return of SP500 since 1928 is: ", miu_annually)
```

```
The annually return of SP500 since 1928 is:  0.08058781300911733
```

Figure 1.4: Annually return test

```python
import math
value = 0
for i in range(len(sp500_list)):
    value += (sp500_list[i]-miu)**2
sigma_daily = math.sqrt(value/(len(sp500_list)-1))
sigma_annually = sigma_daily*math.sqrt(252)
print("The annually volatility of SP500 since 1928 is: ", sigma_annually)
```

```
The annually volatility of SP500 since 1928 is:  0.19001760315371574
```

Figure 1.5:Annually volatility rest

We can clearly see that the python test results are very close to the realistic value. It is reliable because we only select certain periods of time to prove and thus the bias of value is acceptable. In the following codes and explanation, I will keep the value of μ to 0.09 and σ to 0.16.

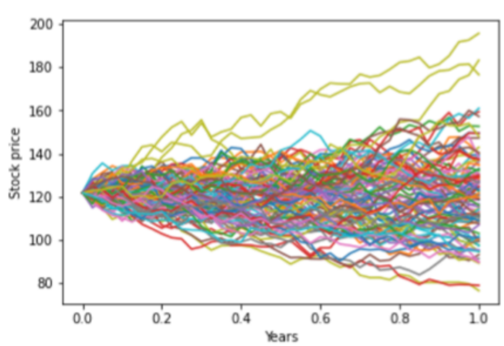
Now, we are good to simulate stock price using Geometric Brownian Motion. We apply the Brownian motion formula:

$$S_t = S_0 e^{((\mu - \sigma^2/2)t + \sigma W_t)}$$

And SDE:

$$dS_t = \mu S_t dt + \sigma S_t dW_t$$

I select the initial price S0 as 121.71, starting from 1981-11-20. To test the reliability, I first simulated 40days Stock Price with 100 simulations, and simulated 4000 days with one simulation, as shown below denoted as Figure 1.6 and Figure 1.7.



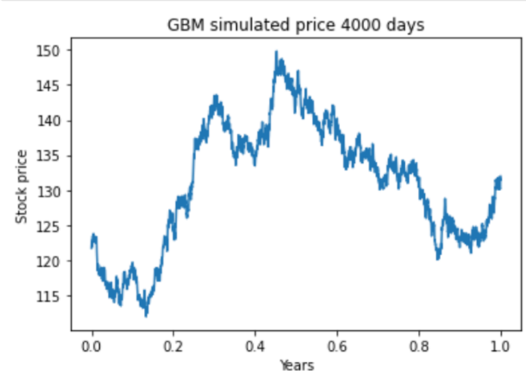


Figure 1.6: 40 days with 100 simulations

Figure 1.7: 4000 days with 1 simulation

Then, to compare the simulations of GBM and realistic price index daily return, I calculated the GBM price index daily return of the 4000 days, and 4000 days realistic daily price index return starting from 1981- 11-20. Here are the results:

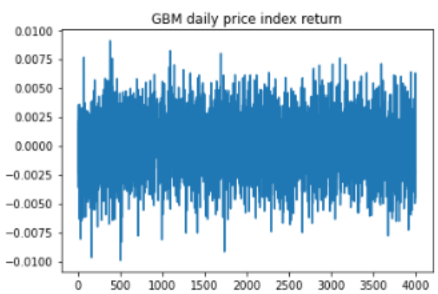


Figure 1.8 Simulated index return

Figure 1.9: Realistic index return

It is not hard to see that we can not directly see the difference from the line charts. To visualize the difference in the same plot, I apply the KDE plot and histogram, as followed in Figure1.10.

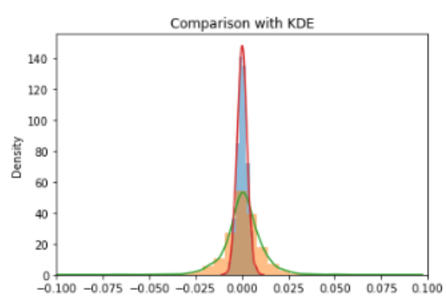
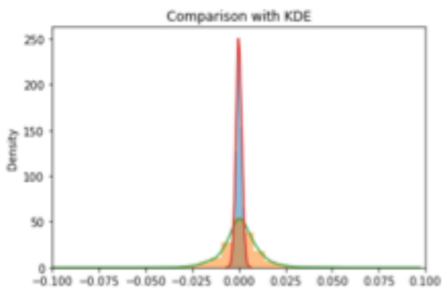


Figure 1.10

It is very clear that the GBM return is not very accurate. The range is smaller compared to the realistic return, which shows that there are many other factors influencing the stock price.

The value of parameters will also affect the accuracy of simulation. For example, with the increase of volatility from 0.10 to 0.25, the GBM will be a better fit, as shown in Figure 11 and 12.



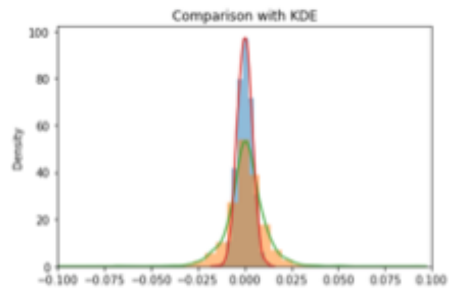


Figure 1.11: sigma = 0.1

Figure 2: sigma = 0.25

2. Modified Ehrenfest Urn Model

Before moving to Modified URN model, I would like to introduce and implement the original URN model. Ehrenfest Urn model is named for Paul Ehrenfest, which is a discrete models for the exchange of gas molecules between two containers. However, they can be formulated as simple ball and urn models; the balls correspond to the molecules and the urns to the two containers.

Suppose there are two urns A and B. Urn A contains N marbles and Urn B contains none. The marbles are labelled 1,2,...N. In each step of the algorithm, a number between 1 and N is chosen randomly, with all values having equal probability. The marble corresponding to that value is moved to the opposite urn. Hence the first step of the algorithm will always involve moving a marble from A to B. The state of the system at time n is the number of balls in urn 1, which we will denote by X_n. The first step analysis is as followed:

$$W_{n-1,n} = \frac{n}{N}, \qquad n = 1, \dots, N$$

$$W_{n+1,n} = \frac{N-n}{N} \qquad n = 0,1, \dots, N-1$$

I realized the model in python and plot the graph, which I simulated 1000 steps with 1000 balls.

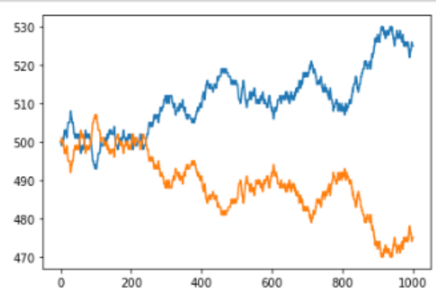


Figure 2.1: Urn model with 1000 steps and 1000 balls

We can further generalize the dynamics abandoning the independence of the creation probability and introducing the dependence on the occupation num- bers, in order to obtain more general equilibrium distributions. Therefore, we add alpha and beta to the scenario.

The first step transition probabilities are:

$$W_{n-1,n} = \frac{n}{N}\frac{\beta+N-n}{\alpha+\beta+N-1}, \qquad n = 1, \dots, N$$

$$W_{n,n} = \frac{n}{N}\frac{\alpha+N-1}{\alpha+\beta+N-1} + \frac{N-n}{N}\frac{\beta+N-n-1}{\alpha+\beta+N-1} \qquad n = 1, \dots, N$$

$$W_{n+1,n} = \frac{N-n}{N}\frac{\alpha+N-n}{\alpha+\beta+N-1}, \qquad n = 1, \dots, N$$

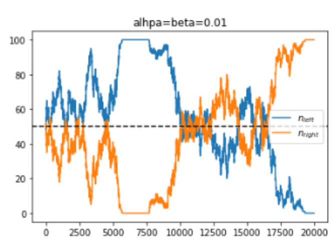
There are three cases of alpha and beta when we discuss it in different situations:

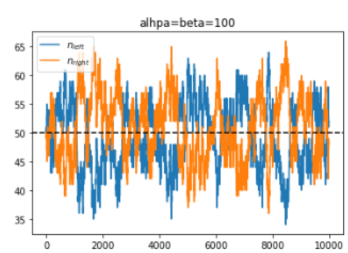
Case1: $0 < \alpha, \beta < 1$: Long time spent around the barriers, with rapid fluctuation across the central region;

Case2: $\alpha, \beta >> 1$: Short time spent around the barriers, with long fluctuation across the central region. If $\alpha, \beta \rightarrow \infty$ the behavior is Bernoullian.

Case3: $\alpha, \beta < 0$, $|\alpha| \geq n$, $|\beta| \geq N - n$: Very short time spent around the barriers, with long fluctuation across the central region;

I plot each case with corresponding alpha and beta values, Figure2.2, 2.3, and 2.4 correspond respectively to case 1, 2 and 3.





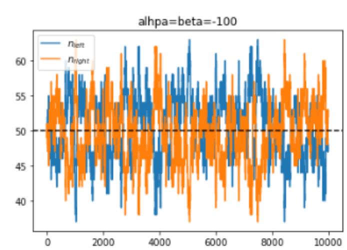
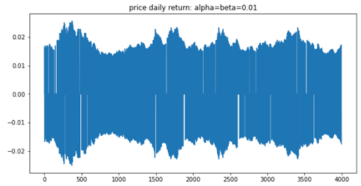


Figure2.2: case 1 *Figure 2.3: case 2* *Figure 2.4: case 3*

When alpha = beta = 0.01, there is a horizontal line from steps 5000 to 7500, and it is unnormal from the other steps interval. When there are 99 balls in the left urn and 1 ball in the right urn, alpha and beta doesn't have much influence on first formula because they are much less than N. In this case, the first formula is fully dependent on N. So, the chance of replacing 1 ball from left to right is very light. There- fore, there are multiple steps that no ball is chosen and replaced from right urn which means that the number of balls in left and right will remain the same. And that's why the line is horizontal from steps 5000 to 7500.

When it comes to case 2, alpha = beta = 100, the fluctuation is rapidly floating from the central line. And that is because alpha and beta now are much greater than N which means they dominate the first step probability formula. Therefore, the probability is more tending to 0.5. In this case, balls are replaced many times between two urns in a balance motion.
To summarize, we pick the particle with probability n / N (in proportion to how many there are in the urn), but we move it with prob roughly (N-n) / N (in proportion to how many there are in the other urn). Therefore, if we are at large n, we tend to stay at large n.

After analyzing the situation of add-up values (alpha and beta), we connect the model to two traders: bull and bear. A bull market is the condition of a financial market in which prices are rising or are expected to rise. The term "bull market" is most often used to refer to the stock market but can be applied to anything that is traded, such as bonds, real estate, currencies, and commodities. Bull market is when investors get very bullish and start buying a lot of stocks driving up their prices and often causing bubbles. Because prices of securities rise and fall essentially continuously during trading, the term "bull market" is typically reserved for extended periods in which a large portion of security prices are rising. Bull markets tend to last for months or even years. In short, a bull trader takes a lot of risk, and a bear trader is very risk averse. I plot the daily returns in 3 cases, with 4000 days and 100 trade, as shown in Figure 2.5, 2.6 and 2.7.



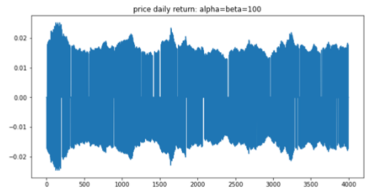
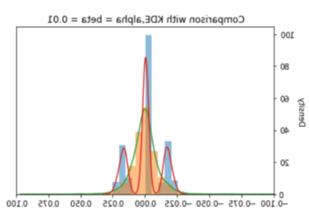
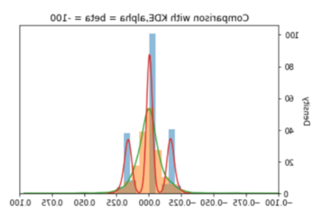


Figure 2.5: case 1 *Figure 2.6: case 2*

Figure 2.7: case 3

To compare with empirical data, I, again, used sp500 datasets to plot the realistic return starting from 1981.11.20, and use KDE plot to compare the difference. The plots are as followed:





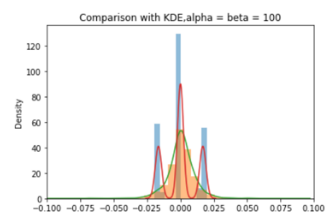


Figure 2.8: case 1 *Figure 2.9: case 2* *Figure 2.10: case 3*

To compare the difference between different alpha and beta values, I also plot the three cases in one plot to directly see the results.

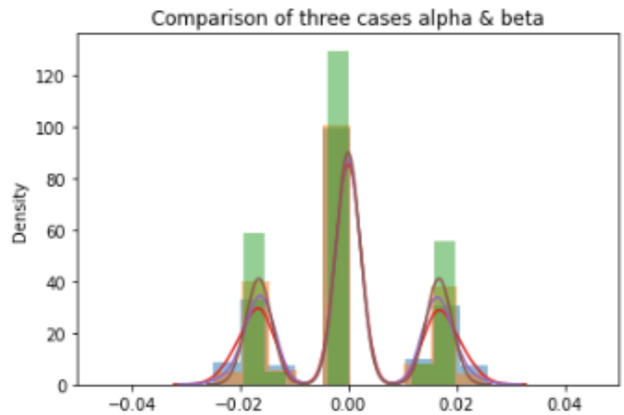


Figure 2.11

We can see from the plots that Modified Urn Model is still not a reliable model. The range is bigger, and the fluctuation is huge. There are many other factors which will affect the performance of the model. For example, we could also have a neutral trader who do not trade in certain situations. This leads to new cases and scenarios.

So far, we have been setting the price change as unit price increment 1. However, in the real stock market, bull or bear traders won't keep the increment as 1. Therefore, the prices change is N(1, sigma) when a bear becomes a bull. I revised the code and add use the method of numpy.random.normal (miu, sigma) to change the price increment. Here are the results of all three cases under the new scenario.

Figure 2.12: Normalized Case 1

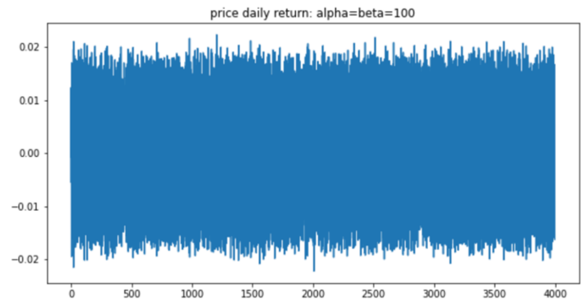


Figure 2.13: Normalized Case2

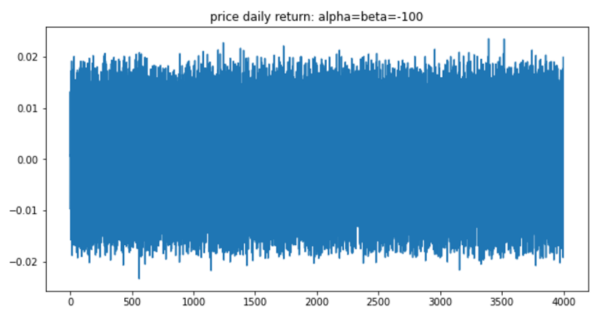
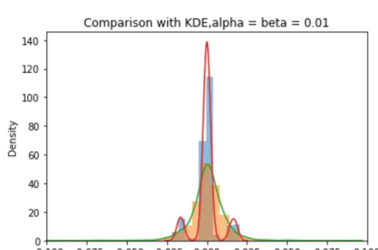
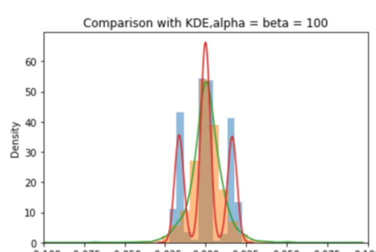


Figure 2.14: Normalized Case3

We can see from the Figure 2.12, 2.13 and 2.14 that the curves are smoother. It becomes more realistic and more similar when it comes to compare with realistic price increment. I also use KDE plots to process the comparison, as showed below in Figure 2.15, 2.16 and 2.17.





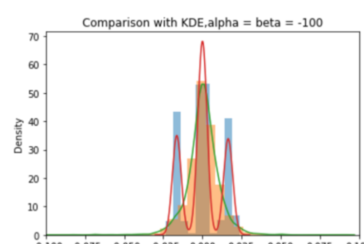


Figure 2.15

Figure 2.16

Figure 2.17

It is clear that the error is smaller but not significantly decreasing. There is still some gap between them because there are so many other factors/randomness which can affect the results, such as the value of alpha and beta, the value of sigma, the population of agents/traders and the simulations time steps. There even may exist a third trader who do not do any action under certain circumstances, which we call a neutral trader. Therefore, the simulation of modified URN model is not very precise to the real data given so many randomness.

3. Conclusion

In this paper, I simulated Geometric Brownian motion and modified Ehrenfest URN model using SP500 dataset. By the results, we can see that both models perform but do not accurately fit the data because there include many affect factors. My next goal is to look at Fractional Brownian motion model.

Reference

"400 Bad Request." 400 Bad Request. Accessed June 17, 2022. https://iopscience.iop.org/article/10.1088/1469-7688/1/2/304/meta.

"Ehrenfest Urn Problem with Applications." Arcane Knowledge. Accessed June 17, 2022. https://www.arcaneknowledge.org/science/ehrenfest.html.

"Generalization of the Ehrenfest Urn Model." Physical Review Journals Archive. Last modified July 1, 1956. https://journals.aps.org/pr/abstract/10.1103/PhysRev.103.17.

"Ehrenfest Urn Model Generalized: an Exact Approach for Market Participation Models." Google Scholar. Accessed June 17, 2022. https://scholar.google.com/citations?view_op=view_citation&hl=en&user=orv15y8AAAAJ&citation_for_view=orv15y8AAAAJ:Tyk-4Ss8FVUC.

Senyuan (Russell) Liu is a senior from Beijing, China, who is majoring in Financial Mathematics and Statistics at the UC Santa Barbara. He is expected to graduate in 2023. He currently is one of the research scholars in SEEDS program (Student Engagement in Data Science) and in Dynamo Lab(prof.Ambuj Singh). Afterwards, he hopes to pursue a master degree and work in Financial Engineering or Data science area.