Simulation Stock Prices Using Geometric Brownian Motion

Juman Ahmed

Arun Kakani

Xu Yan

Abstract

This study uses the Geometric Brownian Motion (GBM) method to simulate the stock price, and test whether the simulated stock prices align with actual tock returns. The sample for this study was based on nine companies listed in some key industries in the U.S. Daily stock price data is obtained from Yahoo! Finance, from October 3, 2016 to March 29, 2019. By applying GBM, we simulate the stock prices after 1000 iterations, and from the results the test of mean forecast error (MFE) remains small before 250 days. By separating these nine companies into three portfolios via volatility, findings of further correlation are showed in this research.

Keywords: Geometric Brownian Motion, Stock Price, Simulation, Mean Forecast Error (MFE), Volatility

1.1. Introduction:

As a Financial Mathematics student, we keep learning knowledge of stock prices, option and different types of derivatives, while pricing is always the key topic of this discipline. In order to make most precise forecast of a stock price, not only should we study it from the outline and basic financial background, but also, we have to analyze it based on history data. While constructing the bridge, or professional called, the path of history data and future price remains the most important part in the eyes of financial managers, investors and related researchers.

In this study, we focus on one of the most useful method in real financial world to get the forecast prices and build portfolios by simulating the price of stock. By selecting nine stocks in unique industries, collecting history data to do the simulation, we finial using Geometric Brownian Motion (hereafter GBM) method to test how well the simulated stock prices align with actual stock returns. We try to compare the volitivity of each selected stock and then build portfolios to have insight look at real financial world.

The remainder of this paper is set out as follows: section II describes how to get history data and how do we do the basic analysis. Section III focuses on the GBM method using on estimating stock prices. Section IV presents hypothesis tests we use while Section V compare actual and simulated prices. Thus, Section VI gives the results and finding coding by MATLAB and Section VII shows how to construct portfolios using these stocks. Besides, Section IX moves to future research and related references.

2.1. The validity of geometric Brownian motion

Brownian motion is often used to explain the movement of time series variables, and in corporate finance the movement of asset prices. Brownian motion dates back to the nineteenth century when it was discovered by biologist Robert Brown examining pollen particles floating in water under the microscope (Ermogenous, 2005). Brown observed that the pollen particles exhibited a jittery motion, and concluded that the particles were ‘alive’. This hypothesis was later confirmed by Albert Einstein in 1905 who observed that under the right conditions, the molecules of water moved at random. A common assumption for stock markets is that they follow Brownian motion, where asset prices are constantly changing often by random amounts (Ermogenous, 2005). This concept has led to the development of a number of models based on radically different theories.

Two common approaches to predicting stock prices are those based on the theory of technical analysis and those based on the theory of fundamental analysis (Fama, 1995). Technical theorists assume that history repeats itself, that is, past patterns of price behavior tend to recur in the future. The fundamental analysis approach assumes that at any point in time an individual security has an intrinsic value that depends on the earning potential of the security, meaning some stocks are overpriced or underpriced (Fama, 1995). Many believe in an entirely different approach; the theory that stock market prices exhibit random walk. The random walk theory is the idea that stocks take a random and unpredictable path, making it near impossible to outperform the market without assuming additional risk. This theory casts serious doubts on the other methods of describing and predicting stock price behavior. The GBM model incorporates this idea of random walks in stock prices through its uncertain component, along with the idea that stocks maintain price trends over time as the certain component. Brewer, Feng and Kwan (2012) describe the uncertain component to the GBM model as the product of the stock's volatility and a stochastic process called Weiner process, which incorporates random volatility and a time interval.

Sengupta (2004) claims that for GBM model to be effective one must imply that:

- The company is a going concern, and its stock prices are continuous in time and value.
- Stocks follow a Markov process, meaning only the current stock price is relevant for predicting future prices.
- The proportional return of a stock is log-normally distributed.
- The continuously compounded return for a stock is normally distributed.
  As discussed in section 3, each of these assumptions has an effect on the GBM model and its inputs.

GBM has two components; a certain component and an uncertain component. The certain component represents the return that the stock will earn over a short period of time, also referred to as the drift of the stock. The uncertain component is a stochastic process including the stocks volatility and an element of random volatility (Sengupta, 2004). Brewer, Fend and Kwan (2012) show that only the volatility parameter is present in the Black-Scholes (BS) model, but the drift parameter is not, as the BS model is derived based on the idea of arbitrage-free pricing. For Brownian motion simulations both the drift and volatility parameter are required, and a higher drift value tends to result in higher simulated prices over the period being analyzed (Brewer, Feng and Kwan, 2012).

Although the GBM process is well-supported, there is a growing amount of literature that focus on testing the validity of the model and accuracy of forecasts using Brownian motion. For example, Abidin and Jaffar (2014) use GBM to forecast future closing prices of small sized companies in Bursa Malaysia. The study focuses on small sized companies because the asset prices are lower and more affordable for individual investors. The study looks into the accuracy of forecasts made using the model over different horizons, and also at the time horizon needed for data inputs into the model, that is, past stock prices. According to Abidin and Jaffar (2014), GBM can be used to forecast a maximum of two-week closing prices. It was also found that one week's data was enough to forecast the share prices using GBM.

Marathe and Ryan (2005) discuss the process for checking whether a given time series follows the GBM process. They also look at methods to remove seasonal variation from a time series, which they claim is important because the GBM process does not include cyclical or seasonal effects. They found that of the four industries they studied, the time series for usage of established services met the criteria for a GBM process; while the data form growth of emergent services did not.

2.2. Other methods of forecasting stock prices

Overtime a number of models have been developed with the objective of forecasting stock prices and pricing options. Some of these models are summarized by Granger (1992), with a particular emphasis on non-linear models. Higgins (2011) demonstrate a simple model to forecast stock prices using analyst earnings forecasts based on the residual income model (RIM). Higgins shows how to implement the RIM and explains how to adjust for auto-correlation to improve forecast accuracy. The RIM uses a combination of fundamental accounting data and mechanical analysis of trends in time series data to derive a valuation of the firm.

Hadavandi, Shavandi and Ghanbari (2010) present an integrated approach based on genetic fuzzy systems (GFS) and artificial neural networks (ANN) for constructing a stock price forecasting expert system. To evaluate the capability of their proposed approach they apply the model to stock price data gathered from IT and Airlines sectors, and compare the outcomes with previous stock price forecasting methods using mean absolute percentage error (MAPE). The results they obtained show that the proposed approach outperforms all previous methods, and so it can be considered a suitable tool for stock price forecasting. However, a drawback to the MAPE approach is that it can only be used to predict the next day closing prices, making it less relevant to managers making strategic decisions.

Hsu, Liu, Yeh and Hung (2009) used a combination of the grey model (GM), Fourier series and Markov state transition matrices to produce a new integration prediction method called the Markov-Fourier grey model (MFGM). The hybrid model was used to predict the turning time of Taiwan weighted stock index (TAIEX) in order to improve forecasting accuracy. According to Hse et al. (2009), MFGM method can predict accurately but it is only suitable for long-term operation.

2.3. Contribution of this paper

Despite an abundance of literature on the application and modifications to the GBM model, there is an apparent lack of research on the accuracy of forecasts made with the model, and thus the validity of the GBM assumption. Therefore, this study contributes to the literature in the following ways:

- The sample chosen is large Australian stocks, representing a market with very little research on GBM.
- The paper focusses on the validity of the GBM assumption over a range of holding periods, namely, one week, two weeks, one month, six months and twelve months. This tests the GBM model on its validity in the long-term as well as the short-term, while most literature only tests the accuracy of forecasts over the short-term.
- Sample is subdivided into portfolios to analyze the effect of stock volatility, expected returns and industry on the accuracy of the model. Other research tends to focus on individual stocks and to the best of our knowledge no one to date has tested the validity of GBM on portfolios.
- Finally, a variety of methods are used to compare actual and simulated prices, specifically, the correlation coefficient, percentage of correct directional predictions, and mean absolute percentage error techniques. Although these methods have been applied by other researchers, most of them focus on only one method.

- Based on the above, following hypotheses are proposed:

H1**:** There is no significant difference between the actual stock prices and the simulated prices using GBM over the sample period.

H2**:** There is no significant relationship between stock volatility and the difference between actual and simulated stock prices.

3.1. Data Collection and Basic Analysis

In this part, we show how do we collect related data and then use some methods to do basic analysis thus to prepare for the GBM simulation.

First, we focus on the diversity of companies that be chosen to guarantee the strictness of our research and lower the average risk of portfolios. Considering key industries in the U.S., we select Amazon on behalf of Specialty Retail, Boeing as the representative of Aerospace & Defense, Chipotle for Restaurants, Duke Energy to present Utilities, Johnson & Johnson as Drug Manufacturers, also, EXXON, Gap, Walmart, Wells Fargo as the representative of Oil & Gas Integrated, Apparel Stores, Discount Stores and Banks.

Information of each stock could be founded on Yahoo! Finance and we take October 3, 2016 as the starting date and March 29, 2019 as the ending data. By downloading the open price and close price for following period, we gather all the data we need for GBM.

3.2. Application of GBM

Equation 1 below shows the formula for the proportional return of a stock:

$$\frac{\Delta S}{S} = \mu\Delta t + \sigma\varepsilon\sqrt{\Delta t} \qquad (1)$$

First component shows the expected rate of return μ that a stock will earn over a short period of time Δt, this component is often referred to as the certain component. The second component follows a random process where σ is the expected volatility of the stock and $\varepsilon\sqrt{\Delta t}$ represents the random volatility which magnifies as the period of time increases.
GBM assumes that stock prices are log-normally distributed with a mean of the certain component and a standard deviation of the uncertain component, shown in equation 2 below:

$$\ln\frac{S_T}{S_0} \sim \phi\left[\left(\mu - \frac{\sigma^2}{2}\right)T, \sigma\varepsilon\sqrt{T}\right] \qquad (2)$$

Where S0 is the stock price now and ST is the price at time T. Notice that μ has been replaced with (μ-σ2/2) to superimpose an uncertainty component to generate a fluctuating stock price. As T represents any time interval it is possible to simulate the price of a stock at time t+Δt given its price at t, where Δt is a short time interval, using the lognormal distribution as shown in the following equation:

$$\ln\left(\frac{S_{t+\Delta t}}{S_t}\right) = \left(\mu - \frac{\sigma^2}{2}\right)\Delta t + \sigma\varepsilon\sqrt{\Delta t} \qquad (3)$$

Finally, rearranging equation 3 results in the final equation I used in our stock price simulations, shown below:

$$S_{t+\Delta t} = S_t \exp\left[\left(\mu - \frac{\sigma^2}{2}\right)\Delta t + \sigma\varepsilon\sqrt{\Delta t}\right] \qquad (4)$$

To recap, St is the stock price at time t, Δt is the time interval for prediction, μ is the expected annual rate of return, σ is the expected annual volatility, and ε is a randomly drawn number from a normal distribution with a mean of zero and a standard deviation of one, representing random volatility. The time interval for prediction we use is one day, as we are interested in predicting daily prices over the period October 3, 2016 to March 29, 2019.

3.3. How the hypotheses were tested (Need to work on)

This paper involves Two tests of the GBM assumption, comparing simulated and actual stock prices for individual stocks, portfolios based on volatility. Each test corresponds to the hypotheses set in section 2 and were carried out as follows:

3.3.1. Test One: Individual simulations were conducted for each constituent stock of nine different industries. Index over the period 1 October 2016 to 31 March 2019. This test involves an analysis of the forecast error over a period of 7 days, 14 days, 30 days, 126 days, and 252 days. After the Analysis, a forecast of 1000 simulation was created for next one year.

3.3.2. Text Two: Stocks were ranked in terms of their total volatility over the chosen period and grouped into quintile portfolios, portfolio 1 containing low volatility stocks and portfolio 3 containing high volatility stocks. The portfolios were value-weighted based on an investment of $1000 in each stock.

3.4. Comparing Actual and Simulated Prices

We have used two methods to compare simulated and actual prices. First, the correlation coefficient (r) is considered to measure the linear correlation between simulated and actual prices. The correlation coefficient produces a value between negative one and positive one, where negative one is perfect negative correlation, zero is no correlation, and positive one is perfect positive correlation. The following formula was used where x and y are different variables (simulated vs actual prices) and n is the number of observations:

$$r = \frac{n(\sum xy)-(\sum x)(\sum y)}{\sqrt{[n(\sum x^2)-(\sum x)^2][n(\sum y^2)-(\sum y)^2]}} \tag{5}$$

Second, we calculated the Forecast Error which is the residual error of the prediction. We calculated the forecast error of the mean of 7 days, 14 days, 30 days, 126 days, and 252 days. Equation 6 was used to calculate the forecast error:

$$e(t) = y(t) - \hat{y}(t|t-1) \tag{6}$$

where,

$y(t)$ = observation

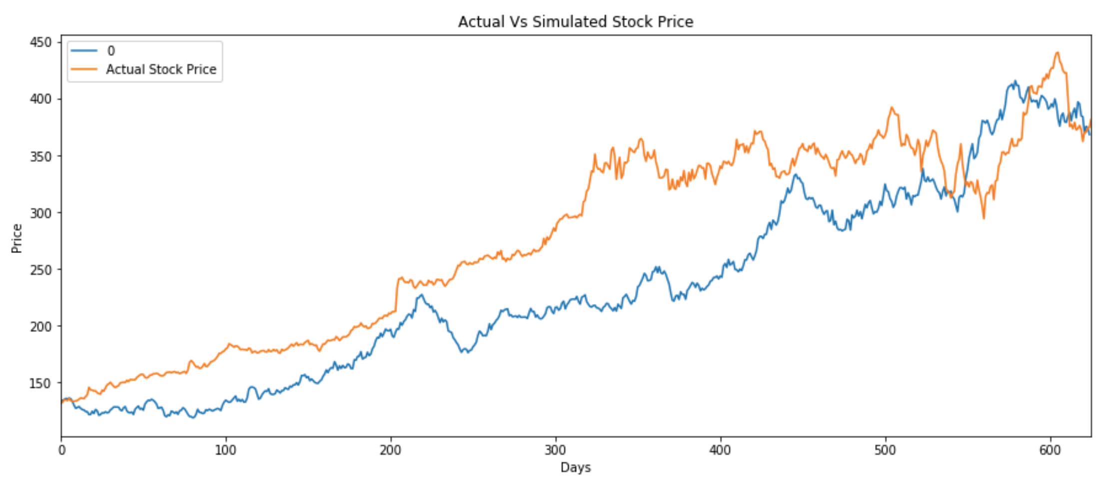
$\hat{y}(t|t-1)$ = denote the forecast of $y(t)$ based on all previous observations

4. Results

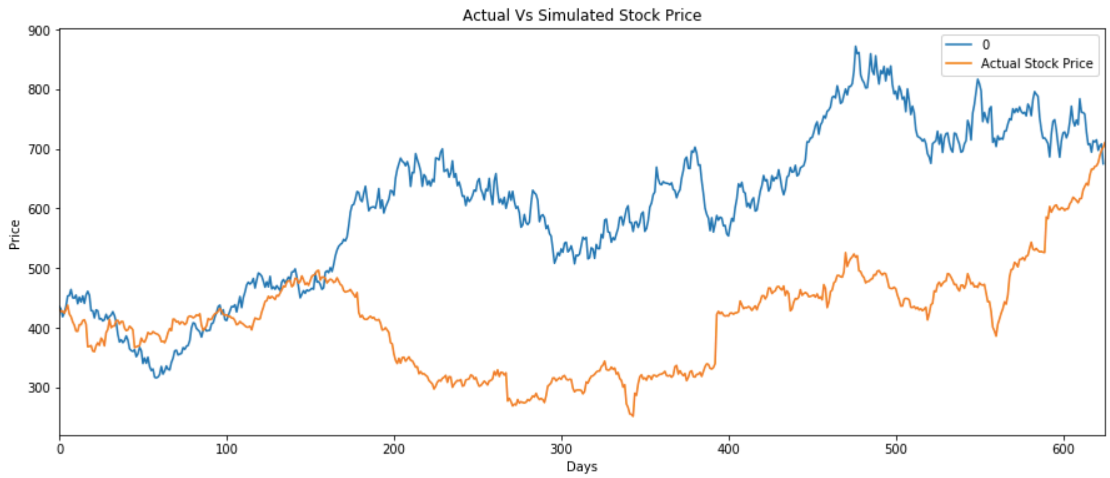
4.1. Test One – Individual Stocks

Below are the graphs we have gotten that shows 'Actual Vs Simulated Stock Price.' As you can see from the graph and the table below, the correlation between the actual and simulated stock price is that for short periods of time, the correlation is positive and the longer it goes, the correlation decreases. This shows that for a short term period, geometric Brownian motion is a relatively good model to simulate stock prices, but as the time goes longer, geometric Brownian motion deviate quite a bit from the actual stock price making it less useful when predicting the stock price and year or two from now.

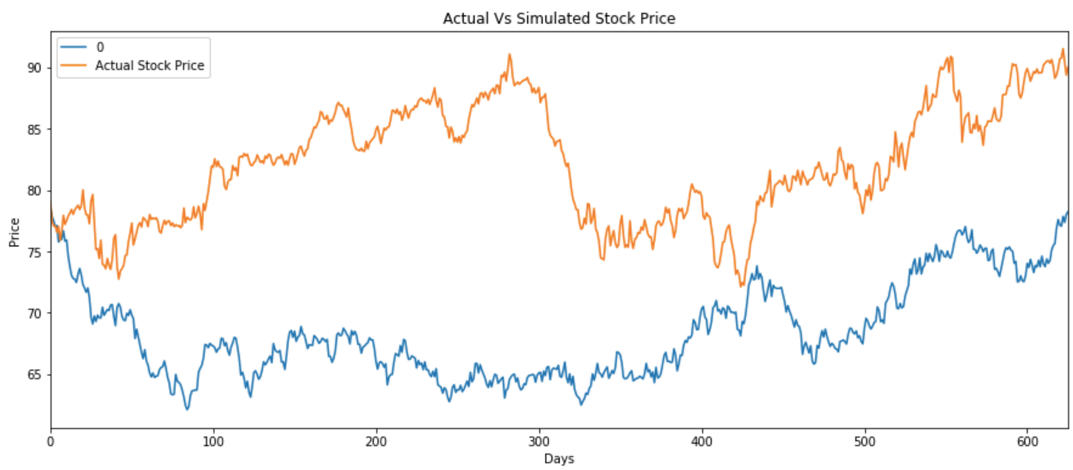
Amazon



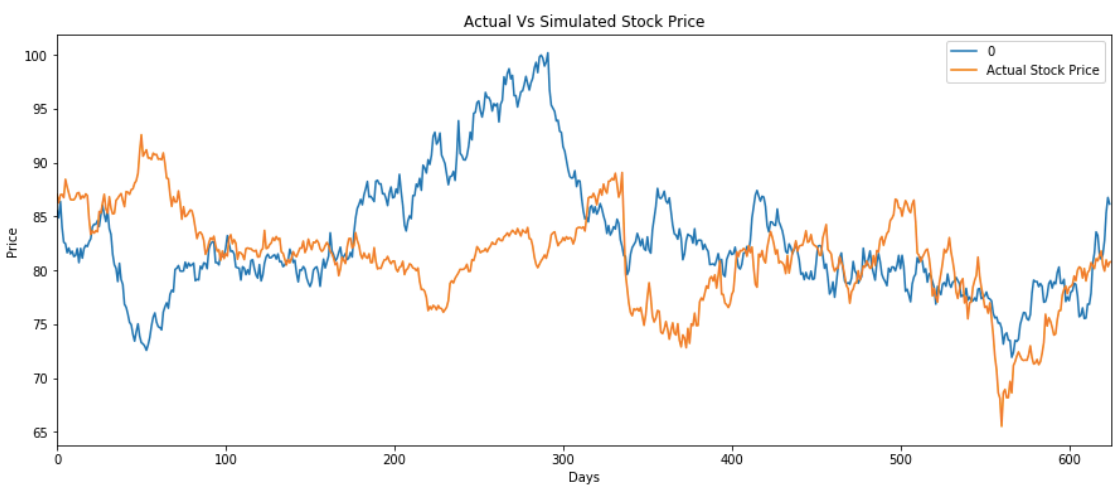
Boeing



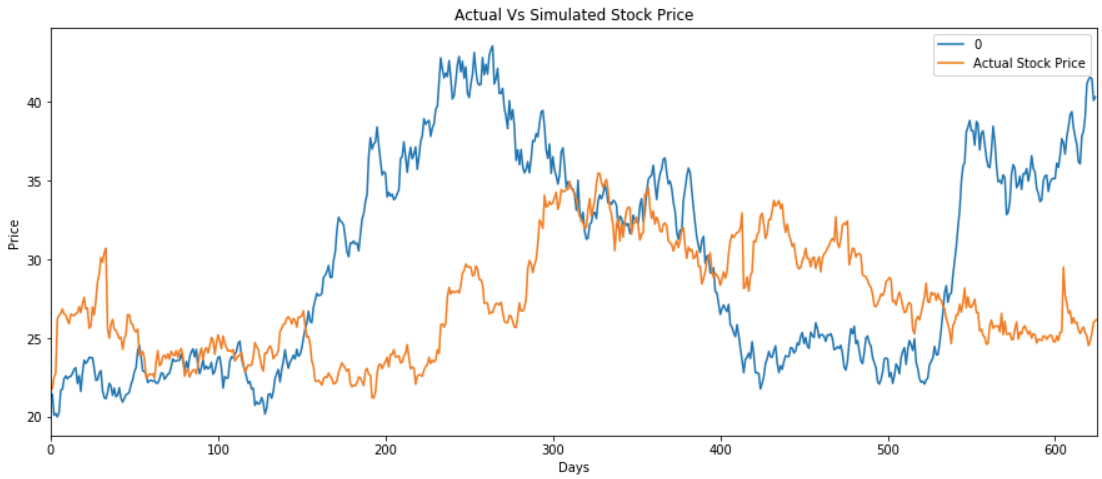
Chipotle



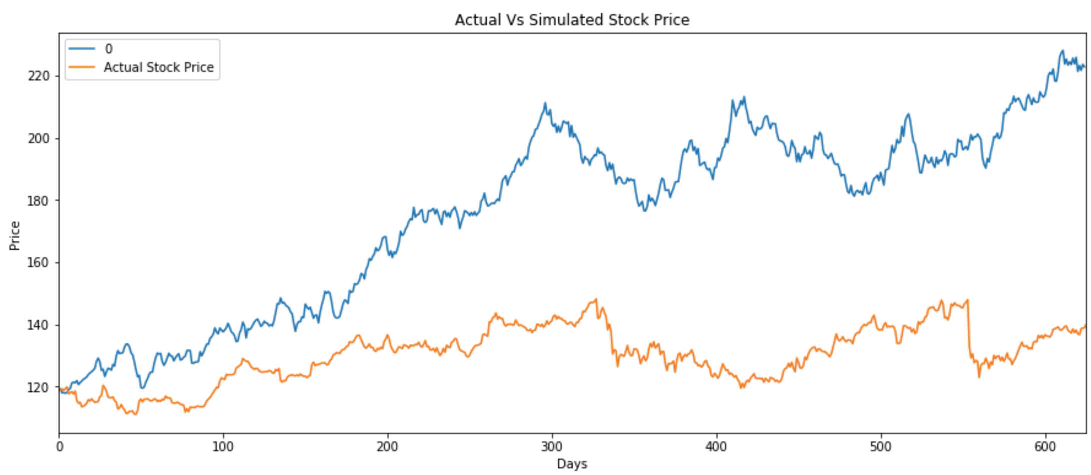
Duke Energy



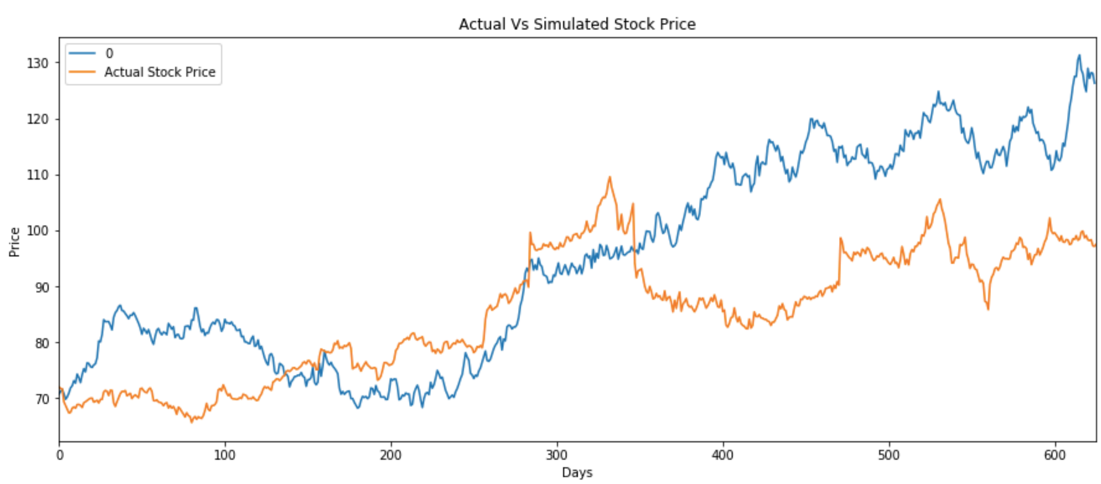
Exxon



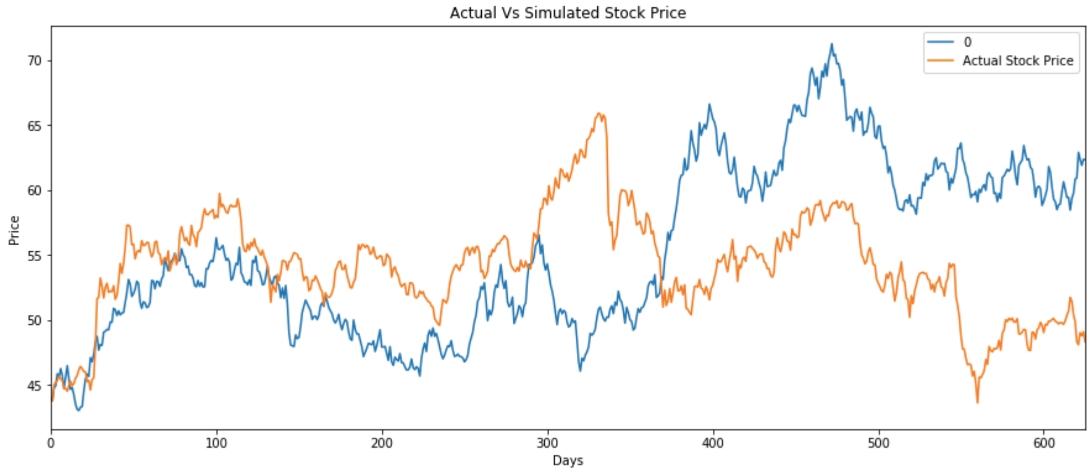
Gap



Johnson and Johnson

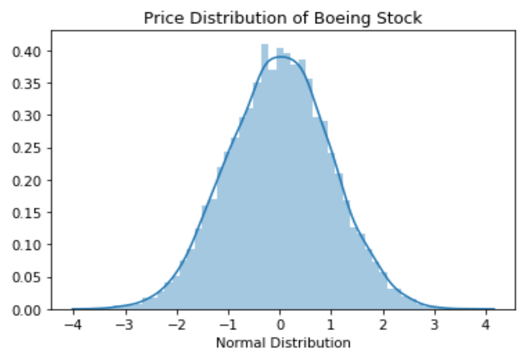


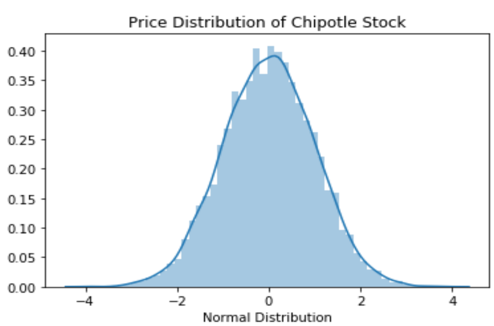
Walmart

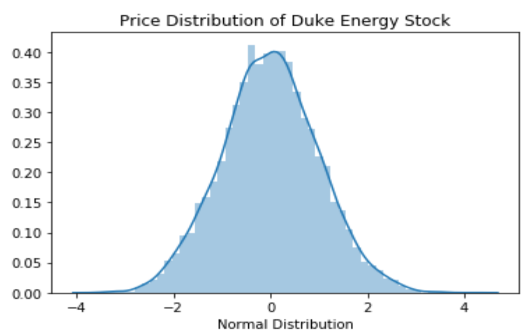

Actual Vs Simulated Stock Price
0
Actual Stock Price
Price
Days

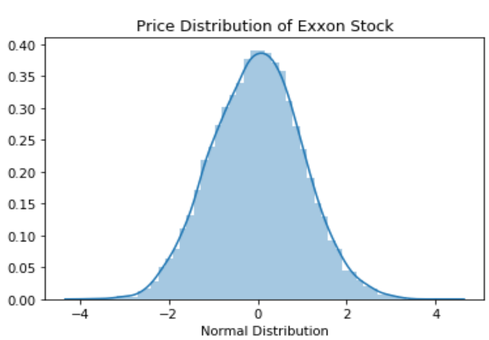
Wells Fargo

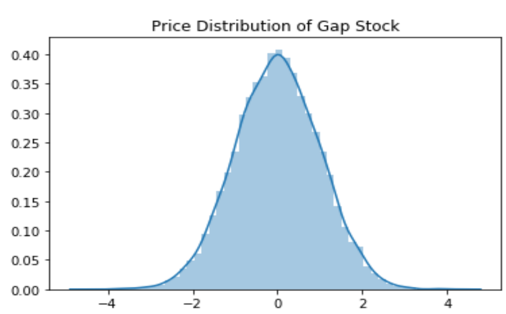

Price Distribution of Amazon Stock
Normal Distribution

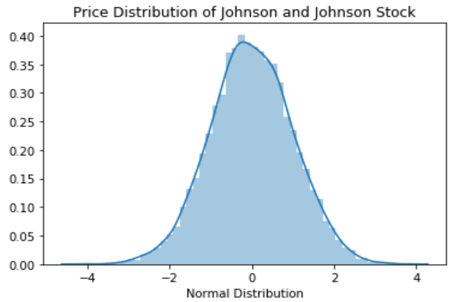

Price Distribution of Boeing Stock
Normal Distribution

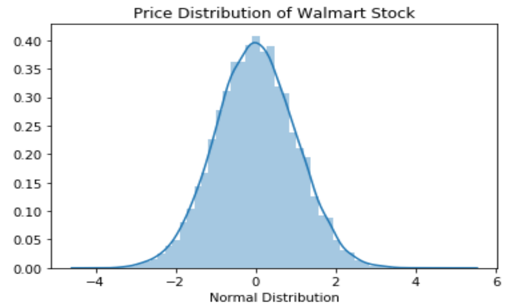

Price Distribution of Chipotle Stock
Normal Distribution

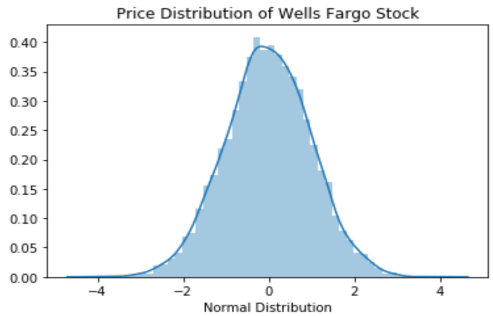

Price Distribution of Duke Energy Stock
Normal Distribution


Price Distribution of Exxon Stock
Normal Distribution


Price Distribution of Gap Stock
Normal Distribution







	Sim (7Days)	Actual (7Days)
Sim (7Days)	1.000000	0.631208
Actual (7Days)	0.631208	1.000000

	Sim (14Days)	Actual (14Days)
Sim (14Days)	1.000000	0.681687
Actual (14Days)	0.681687	1.000000

	Sim (30Days)	Actual (30Days)
Sim (30Days)	1.000000	0.682026
Actual (30Days)	0.682026	1.000000

	Sim (126Days)	Actual (126Days)
Sim (126Days)	1.000000	0.455148
Actual (126Days)	0.455148	1.000000

	Sim (252Days)	Actual (252Days)
Sim (252Days)	1.000000	0.446718
Actual (252Days)	0.446718	1.000000

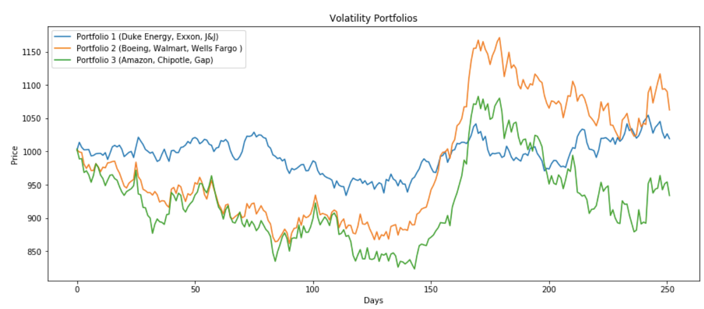
	Mean 7 Days	Mean 14 Days	Mean 30 Days	Mean 126 Days	Mean 252 Days
Mean forecast error of Actual vs Simulated	-0.0102	-0.0033	-0.0003	-0.0023	-0.0005

We have also found the mean forecast error of Actual vs Simulated. As you can see the error decreases as the time increases.

4.2. Test Two – Volatility Portfolios

The second test looks at portfolios formed on the basis of the stock's annual volatility to investigate the hypothesis that there is no significant relationship between stock volatility and the difference between actual and simulated stock prices. The graph below shows the simulated prices for each portfolio based on a $1000 investment in each stock. Portfolio 1 represents stocks with a low volatility and portfolio 3 contains stocks with the highest annual volatility. The volatility of a stock affects the uncertain component of the GBM model, and when volatility increases it effectively magnifies any variation caused by random volatility. The chart below shows that as volatility increases the forecasted stock prices tend to stray further from their mean value. The change in the price of the stock for Portfolio 1 is from 950 to 1040; a change of 90, whereas Portfolio 3 changes from 820 to 1060, change of 240. This shows that, due to their volatility, your investment of a $1000 dollar are very safe under Portfolio 1 but there is not much to gain. But with Portfolio 3, even though there is considerably less safety in your $1000 investment, as you can have a heavy loss, but also have the possibility of having the price of the stock to be very high.

	Volatility of the 9 Companies	
Portfolio 1	Duke Energy	8.64E-05
	J & J	0.000112
	Exxon	0.000118
Portfolio 2	Boeing	0.000259
	Walmart	0.00016
	Wells Fargo	0.00018
Portfolio 3	Chipotle	0.000559
	Gap	0.000593
	Amazon	0.0003378



5. Conclusion

This study explores the geometric Brownian motion model for simulating stock price paths and provides two methods to test the validity of the model. The first method calculates the correlation coefficient between simulated stock prices and actual stock prices. The second method is the forecast error. We have also divided the companies into three to show which Portfolio is the safest investment for $1000 only taking into account the volatility.

References

https://ro.uow.edu.au/cgi/viewcontent.cgi?article=1705&context=aabfj

Codes

- Importing important languages

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt import seaborn as sns
import sklearn
import math
from scipy.stats import norm %matplotlib inline
```

- Finding the Correlation between the companies

```
ticker = r'/Users/jumanahmed/Desktop/MA548 Project/Price.xls'
data=pd.read_excel(ticker,sheet_name='Closing Price')

corr=data.corr()

data.head()

data.tail()

corr.style.background_gradient(cmap='coolwarm')
```

- Finding Actual vs Simulated Stock Price (This is the code for Walmart)
- Similar code was used to simulate other companies stock prices.

```
ticker = r'/Users/jumanahmed/Desktop/MA548 Project/Price.xls'
data=pd.read_excel(ticker,
sheet_name='Walmart').drop(columns={'Open'})

 percent_change = data.Close.pct_change().iloc[:-1]
 percent_change.tail()

 data['Close'].plot(title = 'Walmart Closing
 Price',label='Actual: Startin plt.xlabel('Days')

plt.ylabel('Price')
plt.legend();


percent_change.plot(label='Actual: Starting from
10/03/2016',figsize=(15, plt.xlabel('Days')
```

```
plt.ylabel('Percent Change')
plt.legend();


T=100000
mu = data.Close.pct_change().iloc[:-1].mean() sigma =
data.Close.pct_change().iloc[:-1].var() stdev =
data.Close.pct_change().iloc[:-1].std() daily_drift = mu -
(sigma/2)
ini_price = data.Close.iloc[-625] simulatedprice_Walmart =
pd.DataFrame()

for i in range(1000):
daily_returns=np.exp(daily_drift+stdev*np.random.normal(0,1,625))
.T if i==1:

ini_Walmart = +pd.DataFrame(ini_price*daily_returns.cumprod())

simulatedprice_Walmart = ini_Walmart else:

simulatedprice_Walmart
simulatedprice_Walmart.plot(figsize=(15,6), label = 'Simulated
Stock Pric data['Close'].plot(figsize=(15,6), label='Actual Stock
Price') plt.xlabel('Days')
plt.ylabel('Price')
plt.title('Actual Vs Simulated Stock Price')
plt.legend()

sigma= data.Close.pct_change().iloc[:-1].var()

sigma

simulatedprice_Walmart;

pct_simmu1=simulatedprice_Walmart.pct_change().iloc[:-7].mean()
pct_simmu2=simulatedprice_Walmart.pct_change().iloc[:-14].mean()
pct_simmu3=simulatedprice_Walmart.pct_change().iloc[:-30].mean()
pct_simmu4=simulatedprice_Walmart.pct_change().iloc[:-126].mean()
pct_simmu5=simulatedprice_Walmart.pct_change().iloc[:-252].mean()
pct_simmu=pd.DataFrame({list:[pct_simmu1,pct_simmu2,pct_simmu3,pc
t_simmu4 pct_simmu
```

```
pct_actmu1=data.Close.pct_change().iloc[:-7].mean()
pct_actmu2=data.Close.pct_change().iloc[:-14].mean()
pct_actmu3=data.Close.pct_change().iloc[:-30].mean()
pct_actmu4=data.Close.pct_change().iloc[:-126].mean()
pct_actmu5=data.Close.pct_change().iloc[:-252].mean()
pct_actmu=pd.DataFrame({list:[pct_actmu1,pct_actmu2,pct_actmu3,pc
t_actmu4 pct_actmu

expected = [pct_actmu1, pct_actmu2, pct_actmu3, pct_actmu4,
pct_actmu5] simulated = [pct_simmu1, pct_simmu2, pct_simmu3,
pct_simmu4, pct_simmu5] forecast_errors = [expected[i]-
simulated[i] for i in range(len(expected))
Error=pd.DataFrame({list:forecast_errors})

Error

mu = data.Close.pct_change().iloc[:-1].mean() sigma =
data.Close.pct_change().iloc[:-1].var() stdev =
data.Close.pct_change().iloc[:-1].std() daily_drift = mu -
(sigma/2)

ini_price = data.Close.iloc[-625]

norm.ppf(0.95)
t_intervals = 252
iterations = 1000
daily_returns = np.exp(daily_drift + stdev *
norm.ppf(np.random.rand(t_in S0 = ini_price

price_list = np.zeros_like(daily_returns) price_list[0]
price_list[0] = S0
for t in range(1, t_intervals):

price_list[t] = price_list[t - 1] * daily_returns[t]

plt.figure(figsize=(15,6)) plt.plot(price_list)
plt.xlabel('Days')
plt.ylabel('Price')
plt.title('Forecast Stock Price of Walmart')
```

- Simulating Stock Price Based on Volatility

```
ticker = r'/Users/jumanahmed/Desktop/MA548 Project/Price.xls'
data=pd.read_excel(ticker, sheet_name= 'Closing Price')

mu_p1 = data.P1.pct_change().iloc[:-1].mean() sigma_p1=
data.P1.pct_change().iloc[:-1].var()
stdev_p1=data.P1.pct_change().iloc[:-1].std()
daily_drift_p1=mu_p1-(sigma_p1/2)

ini_price_p1=data.P1.iloc[-626]

mu_p2 = data.P2.pct_change().iloc[:-1].mean() sigma_p2=
data.P2.pct_change().iloc[:-1].var()
stdev_p2=data.P2.pct_change().iloc[:-1].std()
daily_drift_p2=mu_p2-(sigma_p2/2)

ini_price_p2=data.P2.iloc[-626]

mu_p3 = data.P3.pct_change().iloc[:-1].mean() sigma_p3=
data.P3.pct_change().iloc[:-1].var()
stdev_p3=data.P3.pct_change().iloc[:-1].std()
daily_drift_p3=mu_p3-(sigma_p3/2) ini_price_p3=data.P3.iloc[-626]

corr=data.corr()

corr

T=1

mean=[mu_p1,mu_p2,mu_p3]

p1_p2=0.388384
p1_p3=0.221682
p2_p3=0.877550
cov=[[1,p1_p2,p1_p3],[p1_p2,1,p2_p3],[p1_p3,p2_p3,1]]

S0_p1=1000 simprice_p1=pd.DataFrame()

S0_p2=1000 simprice_p2=pd.DataFrame()

S0_p3=1000 simprice_p3=pd.DataFrame()
```

```
for i in range(10000):
p1,p2,p3=np.random.multivariate_normal(mean,cov,252).T
daily_returns_p1=np.exp(daily_drift_p1+stdev_p1*p1)
daily_returns_p2=np.exp(daily_drift_p2+stdev_p2*p2)
daily_returns_p3=np.exp(daily_drift_p3+stdev_p3*p3)

if i==1: sim_p1=+pd.DataFrame(S0_p1*daily_returns_p1.cumprod())
simprice_p1=sim_p1
sim_p2=+pd.DataFrame(S0_p2*daily_returns_p2.cumprod())
simprice_p2=sim_p2
sim_p3=+pd.DataFrame(S0_p3*daily_returns_p3.cumprod())
simprice_p3=sim_p3

else: simprice_p1

simprice_p2
simprice_p3

plt.figure(figsize=(15,6))
plt.plot(simprice_p1,label= 'Portfolio 1 (Duke Energy, Exxon,
J&J)') plt.plot(simprice_p2, label= 'Portfolio 2 (Boeing,
Walmart, Wells Fargo ) plt.plot(simprice_p3, label='Portfolio 3
(Amazon, Chipotle, Gap)') plt.xlabel('Days')
plt.ylabel('Price')
plt.title('Volatility Portfolios')
plt.legend()
```

- Correlation of the Mean

```
data1=pd.read_excel(r'/Users/jumanahmed/Desktop/MA548 Project/Price.
xls', sheet_name= 'Corr7d')
corr=data1.corr()
corr

data2=pd.read_excel(r'/Users/jumanahmed/Desktop/MA548 Project/Price.
xls', sheet_name= 'Corr14d')
corr=data2.corr()
corr

data3=pd.read_excel(r'/Users/jumanahmed/Desktop/MA548 Project/Price.
xls', sheet_name= 'Corr30d')
corr=data3.corr()
```

```
corr

data4=pd.read_excel(r'/Users/jumanahmed/Desktop/MA548 Project/Price.
xls', sheet_name= 'Corr126d')
corr=data4.corr()
corr

data5=pd.read_excel(r'/Users/jumanahmed/Desktop/MA548 Project/Price.
xls', sheet_name= 'Corr252d')
corr=data5.corr()
corr
```