A Project Report

On

Stock-Price Modelling by the Geometric Fractional Brownian Motion

BY

SRIHARI SIRISIPALLI (18XJ1A353)

NIKITHA DONGSARWAR (18XJ1A517)

PARUCHURI JAHNAVI (18XJ1A522)

GADDAM SATWIK REDDY (18XJ1A520)

Under the supervision of

SANJUKTA DAS

SUBMITTED IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS OF

PR 402: PROJECT TYPE COURSE

ECOLE CENTRALE SCHOOL OF ENGINEERING

HYDERABAD

(DECEMBER 2021)

ACKNOWLEDGMENTS

We would like to express our deep gratitude to our project guide Dr.Sanjukta Das, Assistant Professor, Mathematics, Mahindra University, for his/her guidance with unsurpassed knowledge and immense encouragement. We are very much thankful to the vice chancellor and Management, MAHINDRA UNIVERSITY, Hyderabad, for their encouragement and cooperation to carry out this work. We thank all teaching faculty of Department of CSE, whose suggestions during reviews helped us in accomplishment of our project. We would like to thank our parents, friends, and classmates for their encouragement throughout our project period. At last, but not the least, we thank everyone for supporting us directly or indirectly in completing this project successfully.

SRIHARI SIRISIPALLI- 18XJ1A353

DONGSARWAR NIKITHA-18XJ1A517

PARUCHURI JAHNAVI-18XJ1A522

GADDAM SATWIK REDDY-18XJ1A520

Ecole Centrale School of Engineering

Hyderabad

Certificate

This is to certify that the project report entitled "Stock-Price Modelling by the Geometric Fractional Brownian Motion" Submitted by **Mr.Srihari** (HT No. 18353), **Ms. Nikitha Dongsarwar** (HT.No 18517), **Ms.Jahnavi** Paruchuri (HT No 18522), **Mr.Gaddam Satwik Reddy** (HT No 18520) in partial fulfillment of the requirements of the course PR 402, Project Course, embodies the work done by him/her under my supervision and guidance.

(SANJUKTA DAS & Signature)

Ecole Centrale School of Engineering, Hyderabad.

Date:

ABSTRACT

The geometric Brownian motion (GBM) model is a mathematical model that is used to model asset price paths. By incorporating Hurst parameter to GBM to characterize long-memory phenomenon, the geometric fractional Brownian motion (GFBM) model was introduced, which allows its disjoint increments to be correlated. This paper investigates the accuracy of GBM and GFBM in modelling coal India's price simulation, and to see display of persistent or anti-persistent behaviour across different periods with varying Hurst parameter. Results show that the GFBM model is more accurate than the GBM model in simulating future price path for the given data set.

Key words: geometric fractional Brownian motion, fractional Brownian motion, fractional Gaussian noise, Cholesky method, Hurst exponent.

CONTENTS

CHAPTER 1

1.INTRODUCTION:

1.1 Information about Stocks:

We all have heard the word stock one way or the other. Particularly stock is related with the associates and companies which are commercialized and are to settling in the world of marketization. People even term is as an investment plan and it's something people see as a long-term investment that secures and provides an abundant funds during the retirement age.

Buying a company stock is purchasing a small share of it. People invest on the same to get a long-term benefit which they think is less value for now but has to potential to grow with the time. It's an investment that provides the long-time run and deals with long time goals with the fair objectives. The value of share you invest today has to give you a yield of best tomorrow but it's not the same. Market is unpredictable so are the resources and the factors that are taken to drive it off or on the set. It's never been on the same level and the pattern of the same is still unpredictable till the time. Some closeness and prediction method had been derived and approximates values and the rough figures are generated hoping for thebest but all of the resource can’t be trusted and are still unpredictable in nature.

Stock is another way for company to collect revenue and boost up the production for the upper yield and to gain the most out of the business plan for the bigger pictures. This is found to be an effective way to invest and grow in the commercial field and a better alternative to tackle the financial crisis during the requirement. When an individual purchases a company stock then they’re referred as a shareholder and they will get a share out of the same as they have invested in theirprofit or the gain. An investor can sell and buy the stock as per their needs. They can share their stock to their respective or the other individuals where as there are many stock brokers available out in the firm playing with the same.

1.2 Geometric Brownian Motion:

In order to make financial investment decisions, simulated price paths of financial assets are often used to make predictions about the future price. The stochastic price movements of financial assets are often modelled by a GBM, using estimates of the drift and volatility.

A geometric Brownian motion (GBM) (also known as exponential Brownian motion) is a continuous-time_stochastic process in which the logarithm of the randomly varying quantity follows a Brownian motion (Also called a Wiener process) with drift.

Geometric Brownian motion is used to model stock prices in the Black–Scholes model and is the most widely used model of stock price behaviour. The expected returns of GBM are independent of the value of the process (stock price), which agrees with what we would expect in reality. A GBM process only assumes positive values, just like real stock prices. In real stock prices, volatility changes over time (possibly stochastically), but in GBM, volatility is assumed constant.

1.3 Geometrical Fractional Brownian Motion

To improve the accuracy of stock-price modelling, Mandelbrot and Van Ness introduced the definition of fractional Brownian motion or fractal Brownian motion (FBM) in 1968 generalizing the BM by considering the Hurst exponent. Therefore, a new stock-price model, the geometric fractional Brownian motion (GFBM) model was published as an extension of the GBM model. The GFBM model is more general than the GBM model. It can explain more situations about the change of stock prices.

The Hurst exponent indicates the intensity of long-range dependence in a time series, and it can be estimated heuristically using various methods. The GFBM model is more general than the GBM model, and it can explain more behaviours of stock price changes. Past studies have tried to analytically estimate the parameters of this model. The focus is on simulating and testing the model to assess the accuracy of the simulation with comparison to the actual prices by computing the mean absolute percentage error (MAPE) taken relative to the simulated prices, as well as to see display of persistent and anti-persistent behaviour of the given data set.

This thesis will give a detailed introduction to stock-price modelling by the GFBM model and thereby analyse its applications to example data from the COAL INDIA. To evaluate the GFBM stock-price model, simulation and parameter estimations for the GFBM is included.

2.PROBLEM DEFINITION

2.1

Stock is an unpredictable curve that had been in picture ever since its essence had been ever long living and indulging. It had grown its popularity with respect to time. Stock is unpredictable and it's been the same from the start. Its way of escalating and deescalating had been phenomenon and experiencing the same is the best integral part of it. It has its upper hand and flexibility with the changes that has the chances of uprising as well as crashing the whole market. Its easily defined in few words but makingan essence and understanding the same is way more hectic and time consuming.

It has its whole different sets of dependencies and integration from different agents which fluctuate the same in the market. Finding an accurate and getting the exact values out of the same is still unaligned and no particular model of the same is seen in the market value.

Finding the closest and getting an accurate proximate value out of such an unpredictability is a problem in itself. Merging of the data getting the best prediction to increase the efficiency alongside considering the different expects of the moderator is tough and we took the same in consideration and implemented with every aspect to generate the best out of the same and get a result that can be better interrupted andthe efficiency remains the same with the value of different aspects of creating an impactof reducing the risk and influencing the same over the time period to gain the most out of it.

This is totally based on Geometric Fractional Brownian Motion Algorithm to proceed and provide an effective result. Getting the data and processing it and generating a forecast for thousand days is the problem statement that we worked on.

2.2 PROJECT PURPOSE:

Stock market prediction is the act of trying to determine the future value of a company stock or other financial instrument traded on an exchange. The successful prediction of a stock's future price could yield significant profit.

Data is considered as the digital fuel that gives the possibilities of higher yearn and gives the upcoming terms. Knowledge is power and same holds correct with the stock. Stock is unpredictable and over-changing its dynamic in nature. The rise and fall ofthe same is uneven and can't be classified so easily. Dependencies of the same dealswith flexible resources and the agents behind it. The stock is tremendous and hectic in nature. The main theme of the project is to predict the turning curves and bring the predictability method and undergo the process and algorithms to conclude to a viable resource source.

This project helps in bridging the resources and empowering the people to know and trade the most out of stock and understand the generation and the vulnerabilities that has to be seen and predicted.

The enhancement of the same is done which makes a user or the customer to analyses the same and take the needs and important details before dealing and consider those things for the yield that the person is willing to invest on. Forecasting of the stock prediction is done by the available data source and the prediction is done for the upcoming thousand days. The predictability itself is a challenge and that's the main purpose of the report.

BACKGROUND AND RELATED WORK:

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 (Ermogemous, 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 (Ermogemous, 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 behaviour 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 under-priced

The GBM model incorporates the 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 in 2004 claimed 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.

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 analysed (Brewer, Feng and Kwan, 2012).

According to Abidin and Jaffar (2014) [1], 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. The study only focuses on small sized companies because the asset prices are lower and more affordable for individual investors.

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.

Simulating Stock Prices Using Geometric Brownian Motion: Evidence from Australian Companies, and Australian journal written by *Krishna Reddy* and *Vaughan Clinton.* [2]

This study uses the geometric Brownian motion (GBM) method to simulate stock price paths, and tests whether the simulated stock prices align with actual stock returns. The training sample set used for this study on the large listed Australian companies listed on the S&P/ASX 50 Index. Daily stock price data was obtained from the Thomson One database over the period 1 January 2013 to 31 December 2014. The findings are slightly encouraging as results show that over all time horizons the chances of a stock price simulated using GBM moving in the same direction as real stock prices was a little greater than 50 percent. However, the results improved slightly when portfolios were formed.

Stock-Price Modelling by the Geometric Fractional Brownian Motion, written by *Zijie Feng* is based on the Chinese financial markets. [3]

As an extension of the geometric Brownian motion, a geometric fractional Brownian motion (GFBM) is considered as a stock-price model. The modelled GFBM is compared with empirical Chinese stock prices. Comparisons are performed by considering logarithmic-return densities, autocovariance functions, spectral densities and trajectories. Built on the randomness of the BM, it evolves into the geometric Brownian motion (GBM) model, an important model frequently used for stock prices. In financial mathematics, the application of the GBM model is a start for stock-price modelling. It may need to be modified since the increments of real stocks are not mutually independent as in the assumption of the GBM model. To improve the accuracy of stock-price modelling, Mandelbrot and Van Ness introduced the definition of fractional Brownian motion or fractal Brownian motion (FBM) in 1968 generalizing the BM by considering the Hurst exponent. Therefore, a new stock-price model, the geometric fractional Brownian motion (GFBM) model was published as an extension of the GBM model.

Forecasting share prices of small size companies in Bursa Malaysia using geometric Brownian motion, written by *Abidin, S., & Jaffar, M* in 2004.

This paper proposes a way to forecast the future closing price of small sized companies by using geometric Brownian motion. Forecasting is restricted to short term investment because most of the investors aim to gain profit in short period of time. But, to choose the suitable counters to invest is difficult and with the uncertainty of market prices, it will lead to the decline of the investor's confidence level. Therefore, forecasting future closing price is essential. In this paper, we suggest that geometric Brownian motion which involves randomness, volatility and drift can be used to forecast a maximum of two-week investment closing prices. This method is accurately proven by the lower value of the Mean Absolute Percentage Error (MAPE). In addition, the uses of data are also investigated and found that one week data is enough to forecast the share prices using geometric Brownian motion.

CHAPTER 2: Simulation Steps and Theoretical Analysis

To evaluate a GFBM stock-price model, we must be able to simulate GFBM paths. After which, the empirical stock prices can be visually compared with simulated FBM paths. So, let us first deal with simulation of FBM from which we can simulate GFBM paths.

There are three steps involved in the process of simulation of GFBM model:

1. Simulate a finite sequence of FGN based on a given Hurst exponent and the autocovariance function of FGN.
2. Create a sequence of FBM by taking the cumulative sum of our simulated FGN.
3. Finally, the simulation of stock prices can be obtained from the GFBM model with the help of the created FBM.

Remark For convenience to perform analysis, we only simulate a series of daily stock prices. So, we assume $\Delta t = 1$.

2.1 Simulation of Fractional Gaussian Noise

We have two methods to do this, and they are, The Cholesky method and The Davies and Harte method. We will be discussing Cholesky method.

2.1.1 The Cholesky Method

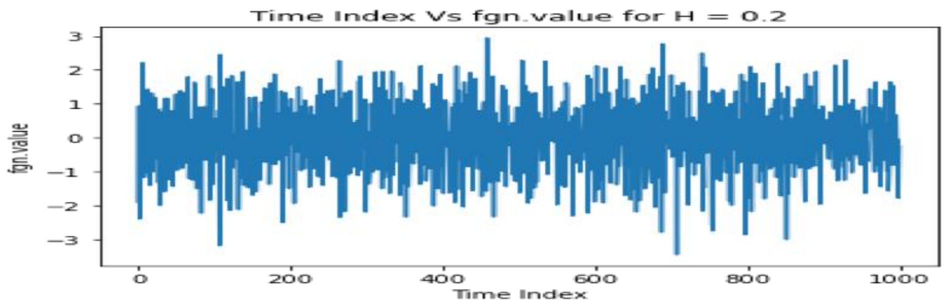
The Cholesky decomposition is an algorithm which can separate a positive- definite matrix into the product of a lower triangular matrix and its conjugate transpose. It means that L is a lower triangular matrix.

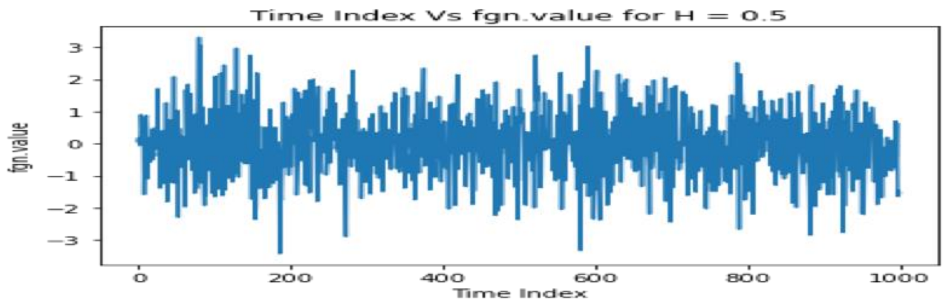
The following codes are used to simulate a series of FGN by Cholesky decom- position in Python:

```
def chelosky(N,H):
    V=np.random.normal(0,1,size=[N,1])
    sigma=np.zeros([N,N])
    dt=1
    for t in range(0,N):
        for s in range(0,N):
            ds=t-s
            c=(1/2)*(abs(ds+dt)**(2*H)-2*abs(ds)**(2*H)+abs(ds-dt)**(2*H))
            sigma[t][s]=c;
    L=np.linalg.cholesky(sigma)
    return np.dot(L,V)
```

With the help of this FGN simulation method, it is easy to visualize the relation among models with different Hurst exponents and the corresponding FGNs.

Figure 2.1 includes three example paths of FGNs with 1000 elements and different Hurst exponents H = 0.2, H = 0.5 and H = 0.8.





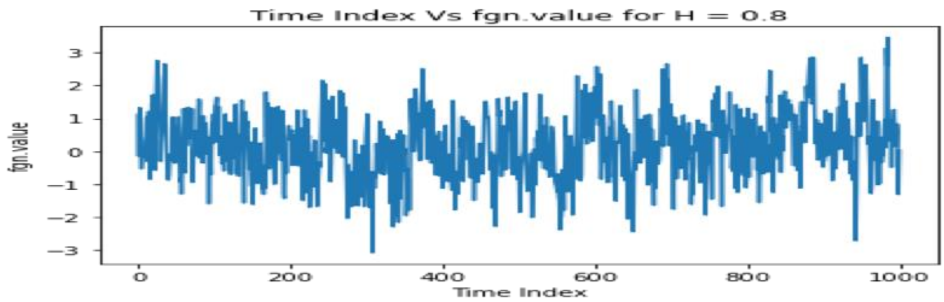


Fig 2.1: Simulated fractional Gaussian noises

The top path represents a simulated sequence of FGN with H = 0.2. It seems to be similar to the middle path, a simulated sequence of FGN with H = 0.5. However, they are actually different since the outcomes of the former path are negatively correlated. Such path flips up and down rapidly, which is in line with the discussion of the sign of $c_{H,\Delta t}(\tau)$. For FGN with H = 0.5, the outcomes are independent.

Different from the first two paths, for the bottom path which is a simulated sequence of FGN with H = 0.8, there obviously exists positive autocorrelation for adjacent time points and that is also in line with the discussion of the sign of $c_{H,\Delta t}(\tau)$.

2.2 Simulated Fractional Brownian motion

$$X_{H,\Delta t}(t) := B_H(t + \Delta t) - B_H(t), t \geq 0,$$

is the increment of BM, called fractional Gaussian noise (FGN) during the time span Δt.

For a given series of FGN, we can obtain a series of FBM at integer-time points by

$$B_H(t) = \sum_{k=1}^{t} X_H(k), \quad t = 1,2,3,...n,$$

where $X_H(k) := X_{H,1}(k)$ is the FGN, and $B_H(t)$ is the FBM. Figure 2.2 contains three paths of FBMs generated from the FGNs in Figure 2.1.

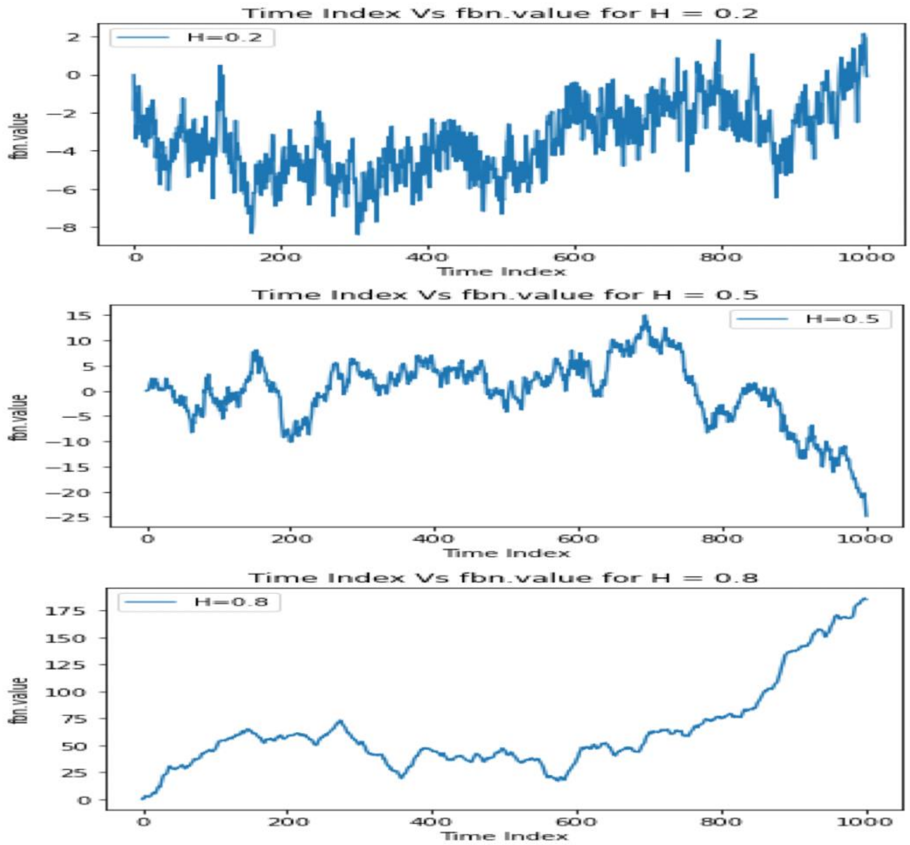


Fig 2.2: Simulated fractional Brownian motions

The first FBM (H = 0.2) has negatively correlated increments and it is much rough compared with the two other FBMs, especially compared with the FBM with H = 0.8. The range of a path is relatively small, here only from -8 to 2.

The bottom path, FBM with H = 0.8, nearly has highly dependent incre- ments, which helps the path to get a dramatically big range, which here is from 0 to 200.

FBM with H = 0.5 looks like an intermediate of FBMs with H = 0.2 and with H = 0.8. It not only fluctuates frequently, but also has quite large range. It is a type of a one-dimensional random walk and moves up and down with the same probabilities.

2.3 Simulated stock prices

The following code in python is used to predict stock price and helps in plotting:

```
H=[0.2,0.5,0.8]
n=1000
mu=0.0060
sigma= 0.027
for h in H:
    fgn = chelosky(n,h)
    plt.plot(fgn)
    plt.ylabel("fgn.value")
    plt.xlabel("Time Index")
    plt.title("Time Index Vs fgn.value for H = "+str(h))
    plt.show()
    B=[0]*(n+1)
    for i in range(1,n+1):
        B[i]=sum(fgn[0:i])[0]
    plt.plot(B)
    plt.ylabel("fbn.value")
    plt.xlabel("Time Index")
    plt.legend(["H="+str(h)])
    plt.title("Time Index Vs fbn.value for H = "+str(h))
    plt.show()
    S=[10]
    for i in range(1,n+1):
        S.append(S[0]*(math.exp(((mu-((sigma**2)/2))*i)+(sigma*B[i]))))
    plt.plot(S)
    plt.xlabel("Time Index")
    plt.ylabel("Stock Price")
    plt.title("Time Index Vs Stock Price H = "+str(h))
    plt.legend(["H="+str(h)])
    plt.show()

    print("-------------------------------------------------------------\n")
```

To simulate a series of stock prices, we use GFBM model. Assume for instance a drift $\mu = 0.0060$, a volatility $\sigma = 0.027$ and an initial stock price $S_0 = 10$. Figure 2.3 gives the output sequences based on those parameter values and the FBMs from Figure 2.2.

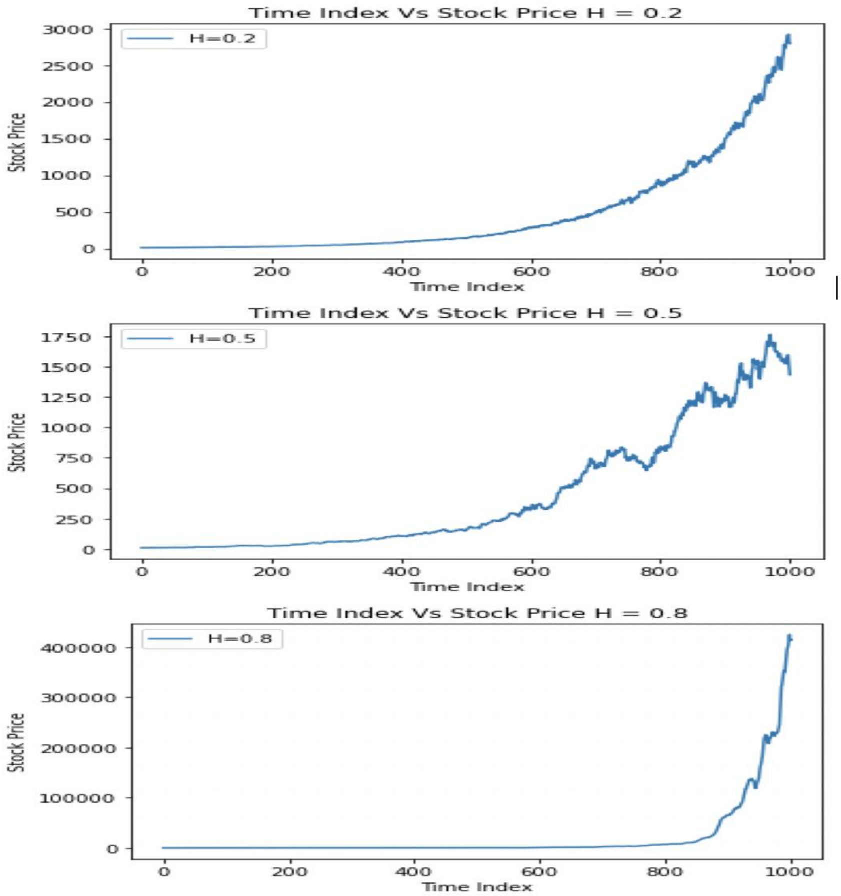


Fig 2.3: Simulated stock prices

Although the growth tendency of prices of the three example sequences of stocks in Figure 2.3 are similar, they are totally different with the development of n. Opposite to Figure 2.2, the simulated stock with H = 0.2 seems to at a large time scale grow more smoothly compared with the other stocks. That can be explained by the fact that the FGN for H = 0.2 fluctuates wildly around zero while its cumulative sum is more concentrated to zero than for H = 0.5 and H = 0.8 due to the negative correlations for H = 0.2. That makes the GFBM to be more exponential like with the randomness smoothed out.

On the contrary, the stock of H = 0.8 is an undulating path with relatively large periods of ups and downs. The cumulative sum of the FGN for H = 0.8 are at long periods increasing and decreasing due to the positive correlation of the FGN implying that the path looks to be more random at a large time scale.

Stock prices with H = 0.5 still seems like an intermediate between the first path and the third path. Its stock price seems to fluctuate more roughly than the stock price with H = 0.2 do at a large time scale, and have shorter periods of ups and downs than stock prices with H = 0.8.

CHAPTER 3 : Comparison between GBM and GFBM

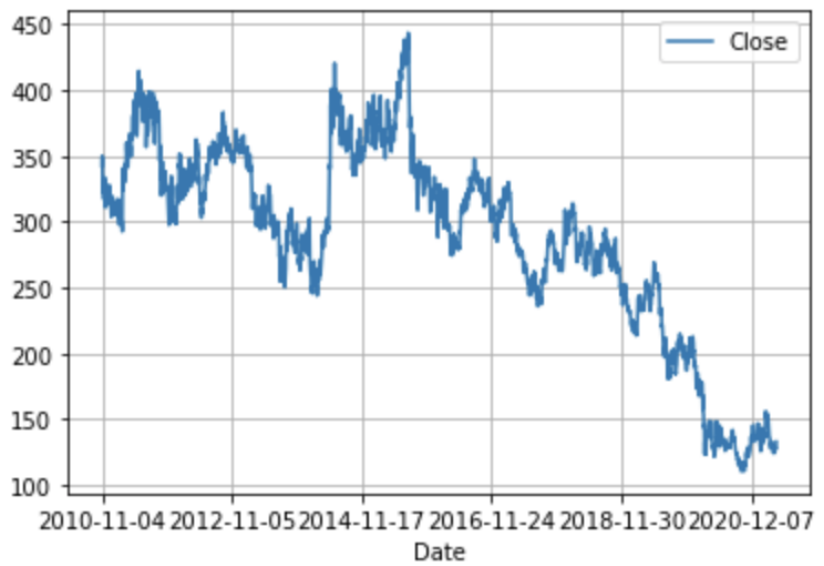
3.1 Random Walk Simulation Of Stock Prices Using Geometric Brownian Motion

To simulate the stock prices, we have taken CoalIndia data since 2010. You can download the data from here. The code for importing the libraries and price data is as follows:

```
import yfinance as yf
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

```
df = pd.read_csv("COALINDIA.csv")
df.index=df["Date"]
df["Close"].plot()
plt.legend()
plt.grid()
plt.show()
df.tail()
```

The output is:



Date	Date	Symbol	Series	Prev Close	Open	High	Low	Last	Close	VWAP	Volume	Turnover	Trades	Deliverable Volume	%Deliverble
2021-04-26	2021-04-26	COALINDIA	EQ	126.15	127.75	128.00	126.50	127.00	126.95	127.30	4967884	6.324104e+13	29393.0	1755583	0.3534
2021-04-27	2021-04-27	COALINDIA	EQ	126.95	127.00	127.90	126.60	127.55	127.50	127.47	3982954	5.077023e+13	28486.0	1521312	0.3820
2021-04-28	2021-04-28	COALINDIA	EQ	127.50	128.00	129.45	127.50	128.50	128.50	128.67	6206074	7.985049e+13	32352.0	1981995	0.3194
2021-04-29	2021-04-29	COALINDIA	EQ	128.50	129.75	130.05	127.65	127.95	128.05	128.46	8345584	1.072040e+14	37445.0	3185650	0.3817
2021-04-30	2021-04-30	COALINDIA	EQ	128.05	127.40	134.60	127.00	133.00	133.05	132.61	27396950	3.633107e+14	103147.0	6859319	0.2504

We will now simulate the prices for the past year and compare them with actual stock performance. First, we calculate the sigma and mu parameters from the previous equations.
Next, we create a dictionary to save all the simulations that we will be making. In this dictionary along with the simulations, we will save the actual stock prices for comparison.
Next, to simulate the prices we should begin all the simulations from the same starting price. In this case, we consider the adjusted close price of the stock that was one year ago.
Next, we simulate the next days' close price as per the GBM formula and append it to the simulation data.

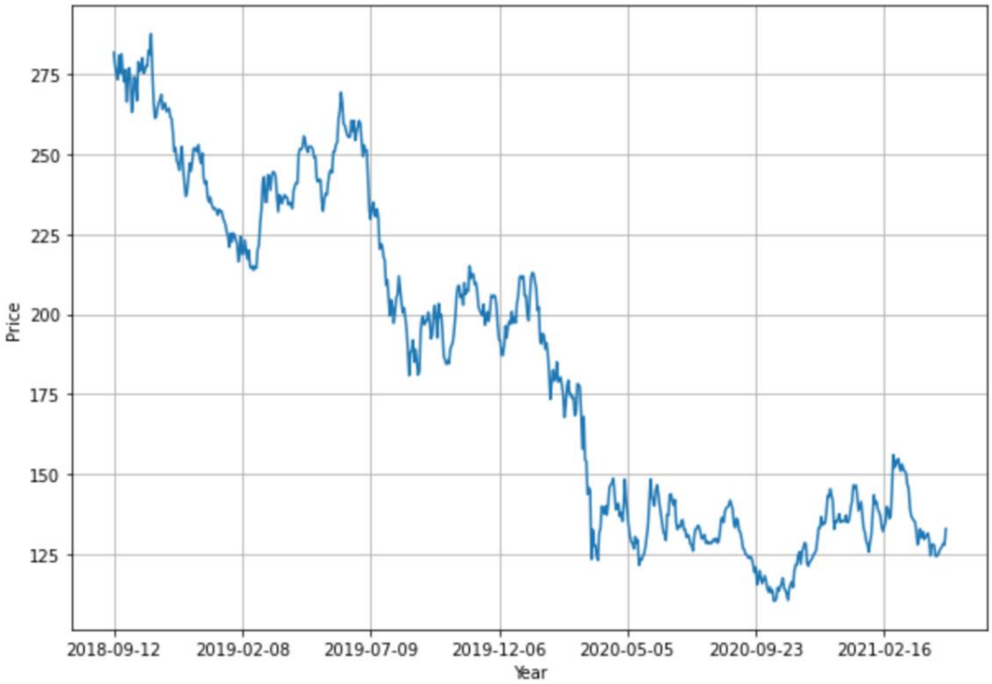
The python code for the above is as follows:

```
validation_size=int(len(df)*0.25)
# Calculate the daily percentage change, mean and sigma
df['daily_pct_change'] = df['Close'].pct_change()
mu = df['daily_pct_change'].iloc[:-validation_size].mean()
sigma = df['daily_pct_change'].iloc[:-validation_size].std()
# Creating the random walk simulation of the probable price path
simulation = {}
simulation['Actual'] = list(df['Close'].iloc[-validation_size:].values)
npaths=100
for sim in range(1,npaths): # Taking npaths
    simulation["Simulation_"+str(sim)] = [df['Close'].iloc[-validation_size]]
    for days in range(validation_size-1):
        next_day = simulation["Simulation_"+str(sim)][-1]*np.exp((mu-(sigma**2/2))+sigma*np.random.normal())
        simulation["Simulation_"+str(sim)].append(next_day)
```

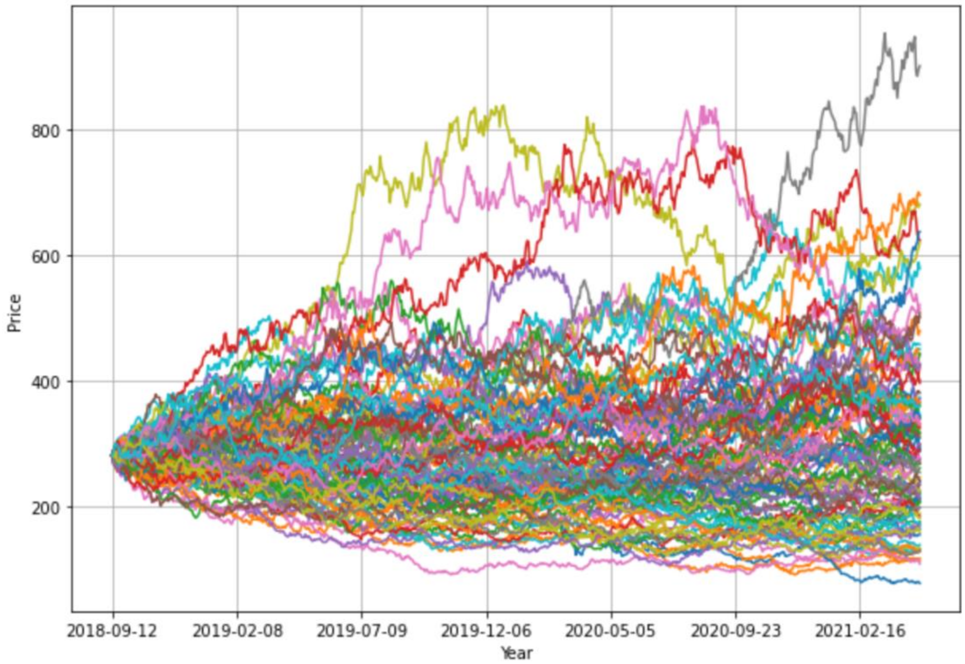
Finally, we will plot 100 simulation and actual price to visualize it.

```
# Plotting the actual Adjusted Close price of General Motors
df['Close'][-validation_size:].plot(figsize=(10,7),grid=True,legend=False)
plt.xlabel('Year')
plt.ylabel('Price')

# Plotting the simulation of random walk

simulation=pd.DataFrame(simulation)
simulation.index=df[-validation_size:].index
simulation.plot(figsize=(10,7),grid=True,legend=False)
plt.xlabel('Year')
plt.ylabel('Price')
```

The actual price plot is as shown below:



The random walk simulation is given by (for 100 simulations):



3.2 Finding Optimal simulation based on RMSE on validation set

Out of the 100 simulations, we find the optimal simulation using the root mean square error on validation set and the code for the same is as follows:

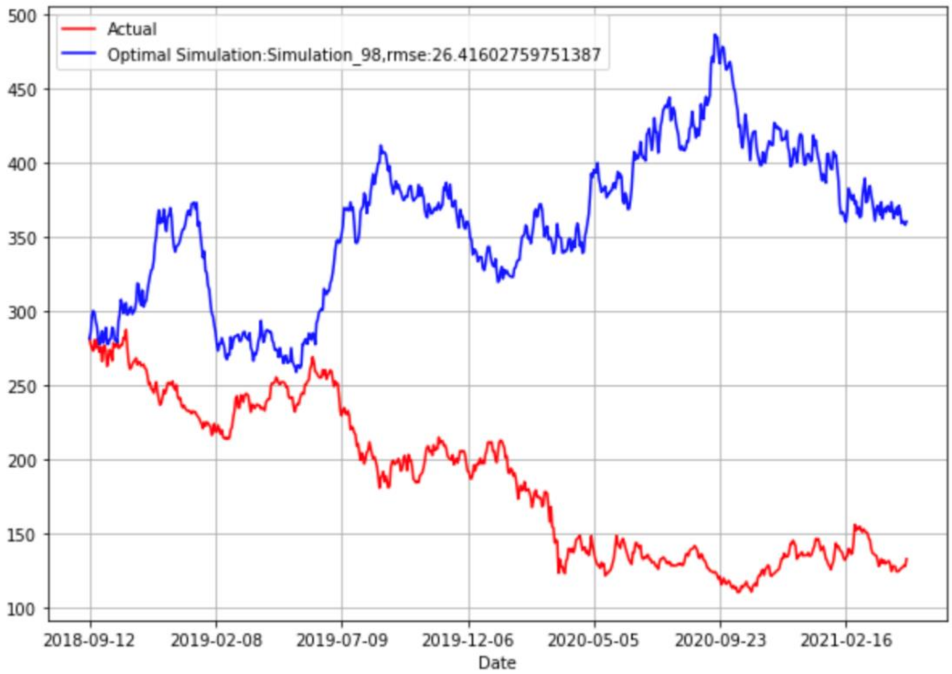
```
# To store errors for each simulation/path
Error_dict=dict()
for sim in range(1,npaths):
    key="Simulation_"+str(sim)
    Error_dict[key]=(((simulation['Actual']-simulation[key])**2).mean())**(1/2)
min_error=min(Error_dict.values())
min_sim=None
for key in Error_dict.keys():
    if Error_dict[key]==min_error:
        print("Optimal Simulation:",key,"with RMSE=",min_error)
        min_sim=key
```

```
Optimal Simulation: Simulation_98 with RMSE= 26.41602759751387
```

Code for plotting the actual and optimal simulation:

```
simulation['Actual'].plot(color='r',grid=True,figsize=(10,7))
plt.plot(simulation[key],color='b')
legend=['Actual','Optimal Simulation:'+str(min_sim)+',rmse:'+str(min_error)]
plt.legend(legend)
plt.show()
```

Output:



Date	Actual	Simulation_1	Simulation_2	Simulation_3	Simulation_4	Simulation_5	Simulation_6	Simulation_7	Simulation_8	Simulation_9	...	Simulation_90	Si
2018-09-12	281.55	281.550000	281.550000	281.550000	281.550000	281.550000	281.550000	281.550000	281.550000	281.550000	...	281.550000	
2018-09-14	277.35	277.566692	280.153863	276.227244	290.871273	274.100896	283.777794	277.622221	280.950134	287.194014	...	276.132435	
2018-09-17	274.95	269.630787	278.687821	282.526550	284.017823	268.490628	270.922445	281.077200	274.258247	290.732993	...	279.175635	
2018-09-18	273.20	271.499282	280.917027	286.568548	287.232614	275.085083	273.724332	287.884441	272.766537	302.197495	...	271.505869	
2018-09-19	280.65	267.265971	282.196417	278.106337	290.363806	269.983988	265.498958	279.906350	272.095498	303.397783	...	272.591436	
...	...	...	...	...	...	...	...	...	...	...	...	...	
2021-04-26	126.95	117.894007	261.615753	333.839750	421.538557	366.510058	168.523150	258.850878	321.587344	439.900660	...	608.074798	
2021-04-27	127.50	117.584349	254.656909	328.301165	415.634006	376.269481	169.424519	255.530263	322.696935	445.018820	...	615.586179	
2021-04-28	128.50	116.688228	255.117829	331.811047	412.316559	383.420279	171.389864	244.354756	318.884394	449.125524	...	620.420992	
2021-04-29	128.05	116.168580	258.197122	333.623130	404.037175	384.230587	174.316563	244.158828	318.699220	451.287729	...	635.427455	
2021-04-30	133.05	118.069173	261.888446	327.947165	401.171257	380.967748	172.655980	239.841807	310.178965	450.606753	...	637.277848	

3.3 Simulation of GFBM

```
validation_size=int(len(df)*0.10)
# Calculate the daily percentage change, mean and sigma
df['daily_pct_change'] = df['Close'].pct_change()
mu = df['daily_pct_change'].iloc[:-validation_size].mean()
sigma = df['daily_pct_change'].iloc[:-validation_size].std()
n=validation_size
simulation = {}
simulation['Actual'] = list(df['Close'].iloc[-validation_size:].values)
S0=df["Close"][len(df)-validation_size]
error_dict=dict()
H=[0.1*i for i in range(1,10)]
for h in H:
    fgn = chelosky(n,h)
    B=[0]*(n+1)
    for i in range(1,n):
        B[i]=sum(fgn[0:i])[0]
    S=[S0]
    for i in range(1,n):
        S.append(S[0]*(math.exp(((mu-((sigma**2)/2))*i)+(sigma*B[i]))))
    simulation["simulation_H="+str(h)]=S
    error_dict["simulation_H="+str(h)]=(((simulation["Actual"]-np.array(S))**2).mean())**(1/2)
simulation=pd.DataFrame(simulation)
simulation.index=df[-validation_size:].index
min_err=min(error_dict.values())
for key in error_dict.keys():
    if error_dict[key]==min_err:
        print("Optimal Simulation:",key,"with RMSE=",error_dict[key])
```

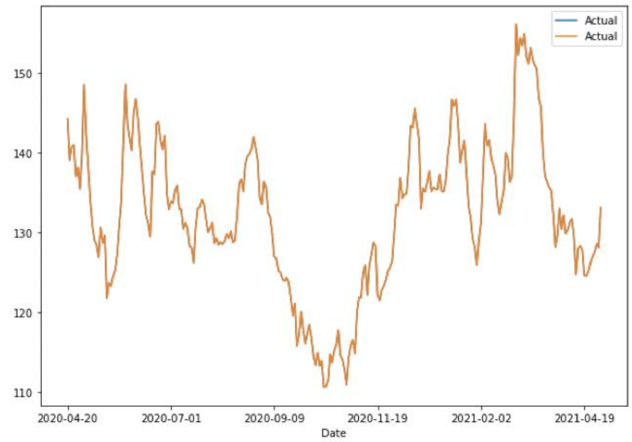
```
Optimal Simulation: simulation_H=0.30000000000000004 with RMSE= 12.121559941238417
```

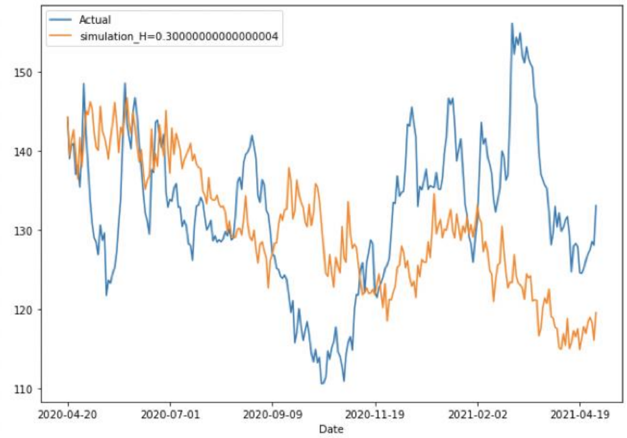
So, here we get the optimal solution with RMSE as 12.12155.

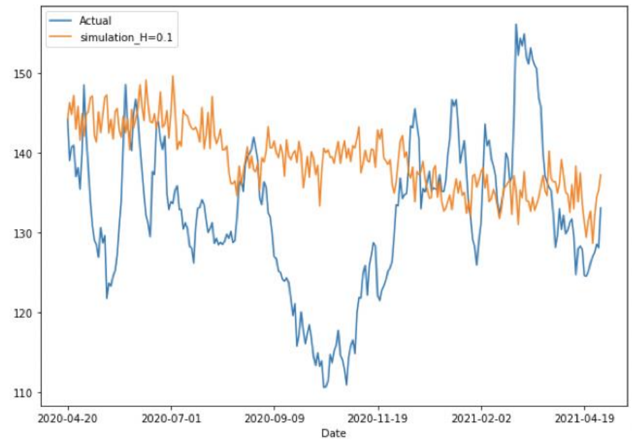
Code for plotting graphs for these simulations is as follows:

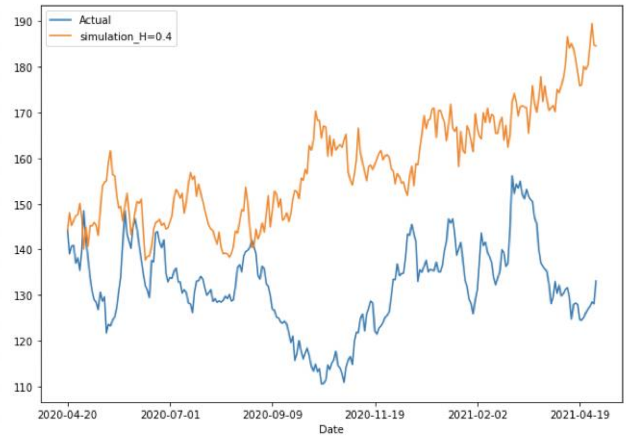
```
for ele in simulation.columns:
    simulation["Actual"].plot(figsize=(10,7),grid=True)
    simulation[ele].plot()
    plt.legend(["Actual",ele])
    plt.show()
```

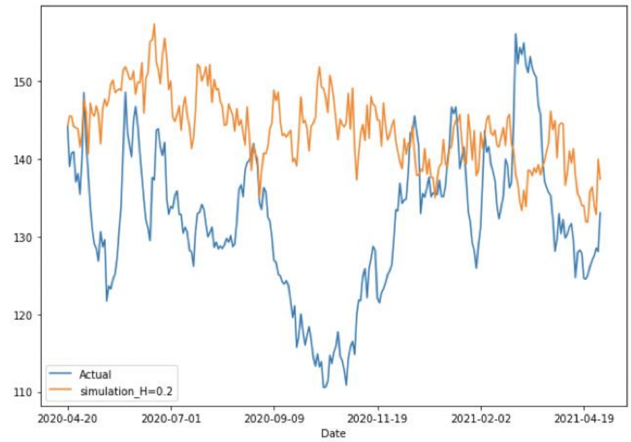
Plots:

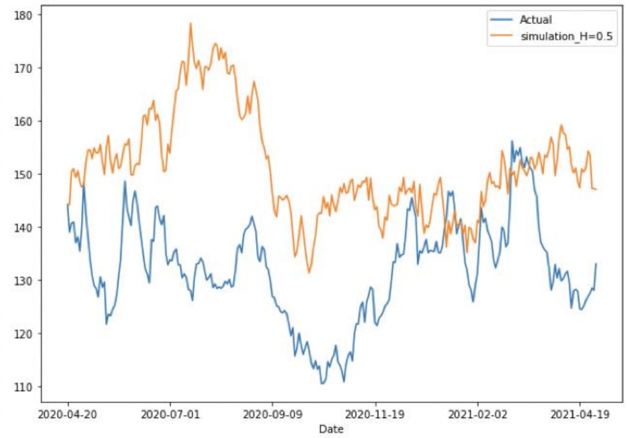

Actual
Actual
150
140
130
120
110
2020-04-20
2020-07-01
2020-09-09
2020-11-19
2021-02-02
2021-04-19
Date

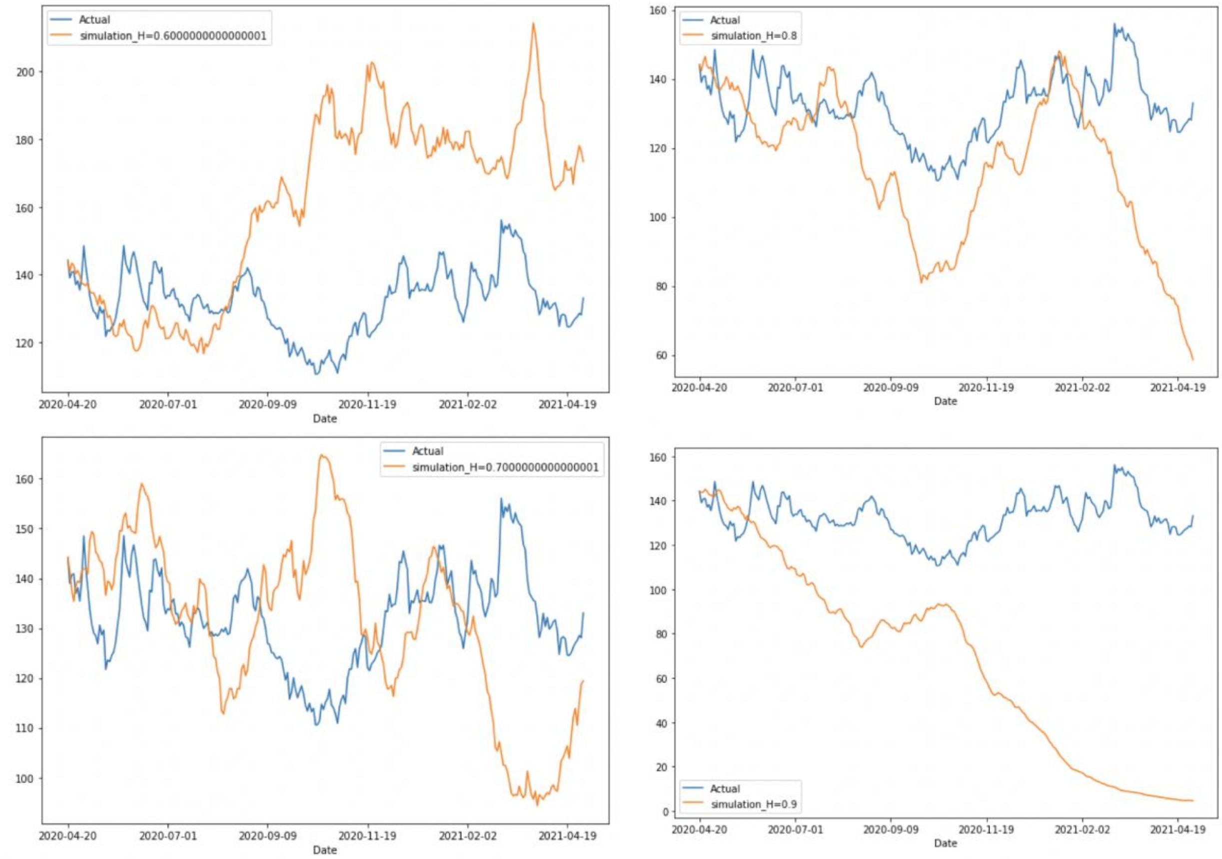

Actual
simulation_H=0.30000000000000004
150
140
130
120
110
2020-04-20
2020-07-01
2020-09-09
2020-11-19
2021-02-02
2021-04-19
Date


Actual
simulation_H=0.1
150
140
130
120
110
2020-04-20
2020-07-01
2020-09-09
2020-11-19
2021-02-02
2021-04-19
Date


Actual
simulation_H=0.4
190
180
170
160
150
140
130
120
110
2020-04-20
2020-07-01
2020-09-09
2020-11-19
2021-02-02
2021-04-19
Date


Actual
simulation_H=0.2
150
140
130
120
110
2020-04-20
2020-07-01
2020-09-09
2020-11-19
2021-02-02
2021-04-19
Date


Actual
simulation_H=0.5
180
170
160
150
140
130
120
110
2020-04-20
2020-07-01
2020-09-09
2020-11-19
2021-02-02
2021-04-19
Date



CONCLUSION

This study tests the accuracy of two mathematical models, geometric Brownian motion (GBM) and geometric fractional Brownian motion (GFBM) models in simulating the future closing price paths of CoalIndia. In order to use these models, unknown parameters need to be estimated using historical prices of the commodity. The accuracy of the simulations is determined by computing the root-mean square error (RMSE), where we conclude that the GFBM model is more accurate than the GBM models for three different Hurst estimators.

Future work can consider to simulate the fractional Gaussian noise using methods such as fast Fourier transform (FFT) which reduces the computation time taken by the Cholesky method, even though the Cholesky method is more straightforward to implement. Moreover, more factors can be incorporated to the GBM and GFBM models such as mean-reversion with jumps models, or seasonality to replicate the commodity market more closely.

REFERENCES

[1] Abidin, S., & Jaffar, M. (2014). *Forecasting share prices of small size companies in bursa Malaysia using geometric Brownian motion. Applied Mathematics & Information Sciences*

[2] Krishna Reddy and Vaughan Clinton (2016). *Simulating Stock Prices Using Geometric Brownian Motion: Evidence from Australian Companies*

[3] Zijie Feng (2018). *Stock-Price Modelling by the Geometric Fractional Brownian Motion based on the Chinese financial markets.*

[4] https://towardsdatascience.com/simulating-stock-prices-in-python-using-geometric-brownian-motion-8dfd6e8c6b18

[5] http://202.62.95.70:8080/jspui/bitstream/123456789/12663/1/1NH16CS134.pdf

[6] https://www.investopedia.com/articles/07/montecarlo.asp