Methods For Simulating Stock Prices

Estimating Parameters

By Ryan Tavolilla

Math 552
Applications of Scientific Computing

April 30, 2024

Motivation And Goal For This Project

I am incredibly interested in the stock market and understanding the motion of stock prices.

The goal is to find a model to simulate (and even predict) the price of a stock.

I then use real stock data to analyze this model and understand it better.

Finally, I examine the numerical results and come to conclusions about the model.

Modeling Seemingly Random Processes

I don't have time to go in depth but my goal is to give the understanding you will need to follow the rest of this presentation.

- **Stochastic Processes**

  Generally, a series of random events indexed by time. Alternatively, it can also be thought of as a probability distribution of possible paths. Has no deterministic or fixed path. [1]

- **Brownian Motion (Wiener Process)**

  Specific type of stochastic process such that $\{W_t\}_{t \geq 0+}$. In other words. W_t is indexed on non-negative real numbers. [2]

  The more paths you take, eventually you converge towards the standard brownian motion.

Stochastic Differential Equations (SDEs)

The movement of most things in life can be described as some form of differential equation, so if we can solve “random differential equations”, we can model more complicated random processes.

- **Stochastic Differential Equations (SDEs)**

  $$dX = a(t, X)dt + b(t, X)dW_t$$

  The first term is known as the ‘drift’ term and the second is known as the ‘diffusion’ term, of which includes a standard wiener process (from last slide). [3]

  SDEs are given in differential form, not derivative form. This is because Brownian Motion and other stochastic processes are continuous but not differentiable.

The Cornerstone of Stochastic Calculus (Ito's Lemma)

Ito's Lemma provides a way to calculate the differential of a function of a stochastic process such as the one described in the last slide.

Suppose f is a C^2 function and B_t is a standard Brownian motion. Then for every t:

$$f(B_t) = f(B_0) + \int_0^t f'(B_s)dB_s + \frac{1}{2}\int_0^t f''(B_s)ds$$

In differential form, this is written as:

$$df(B_t) = f'(B_t)dB_t + \frac{1}{2}f''(B_t)dt$$

[4]

Forming Our Model For Stock Price

A typical model for a "time-evolution" of an asset price $S(t)$ is given by a **geometric Brownian motion** [5] , which is represented using a stochastic differential equation:

$$dS(t) = \mu S(t)dt + \sigma S(t)B(t)$$

[3]

- μ is the 'drift' of the asset
- σ is the 'volatility' of the asset

Solving Stochastic Differential Equations (SDEs)

Solving the SDE gives us a function that describes the prices of a stock at any time t.

1. Most SDEs require a **numerical methods** to solve
   - Strong Convergence of SDE Solvers.
   - Weak Convergence of SDE Solvers.
   - Populare Methods: Euler-Maruyama Method, Milstein Method, Runge-Kutta Method, Taylor Method.
2. **Exact Solutions**
   Luckily, the model I described for the stock price is one of the few SDEs that have an exact solution to them. Using Ito's Lemma, you can prove the following solution:

$$S(t) = S(0)\exp\left(\left(\mu - \frac{1}{2}\sigma^2\right)t + \sigma B(t)\right)$$

[3]

Estimating Parameters

This is the problem to answer! If you have exact parameters then you can in theory model the stock price exactly using this model.

These parameters can come in many forms:

μ	σ
$\mu = a$	$\sigma = b$
$\mu = \mu(t)$	$\sigma = \sigma(t)$
$\mu = \mu(t, S(t))$	$\sigma = \sigma(t, S(t))$
$\mu = \mu(t, S(t), \ldots)$	$\sigma = \sigma(t, S(t), \ldots)$

[6]

These parameters allow the model to appear so simple, but solving them can be difficult because of how they can be reliant on some many other processes.

Constant Parameters: $\mu = a, \sigma = b$

The Fixed Parameter Method is based on the theory that:

In general, μ and σ don't change incredibly much with time, so if you can get a good approximation of these parameters as constant values, they will give a you a good idea of the general trend and motion of a given stock.

I specifically using values of:

$$\mu = 0.08, \quad \sigma = 0.2$$

Because I am specifically choosing stock data from SP500, I know that the average yearly return is roughly 8-10% which is what helped me determine my μ. As for σ I just 'eyeballed' it based on what was close to the true stock price when testing.

Capital Asset Pricing Model (CAPM): $\mu(S(t)), \sigma(S(t))$

The Capital Asset Pricing Model (CAPM) is a financial model used to determine the expected return on an investment based on its risk. [7]

$$\mu = r_f + \beta(r_m - r_f)$$

- r_f is the risk free rate of return (often denoted by the rate of return of a US Treasure Bond).
- β is the beta of the stock against the market. It represents the systematic risk of the trade.
- r_m is the expected return of the market portfolio.

$$\sigma = \frac{s}{\sqrt{\tau}}$$

- s is the daily standard deviation
- τ is the time-step interval, often 1/252 (number of business trading days)

Assumed SP500 stocks only for simplicity.

Bootstrap Method $\mu(t, S(t)), \sigma(t, S(t))$

The Bootstrap Method is a resampling technique where multiple samples are drawn with replacement from the original dataset. [8]

$$R_t = \frac{S_t - S_{t-1}}{S_{t-1}}, \quad \mu = \frac{1}{M\delta t}\sum_{t=1}^{M} R_t$$

- R_t is the return of the stock at time t.
- S_t is the stock price at time t. S_{t-1} is the stock price at time $t-1$.

$$\sigma_1 = \sqrt{\frac{1}{(M-1)\delta t}\sum_{t=1}^{M}(R_t - \bar{R})^2}, \quad \sigma_2 = \sqrt{\frac{1}{(N-1)\delta t}\sum_{t=2}^{N}(\log(S_t) - log(S_{t-1}))^2}$$

- σ_1 = common volatility
- σ_2 = log volatility
- $\bar{R}$ = mean of stock return
- M = amount of stock return data
- N = amount of stock data [9]

Method of Moments

Method Of Moments is a statistical technique used for parameter estimation in probability distribution by equating sample moments to their theoretical counterparts. [10]

The steps are as followed:

1. Define the population of moments
2. Calculate sample moments from observed data
3. Equate the sample moments to the population moments
4. Solve equations to obtain the parameters

Other Parameter Estimation Methods

There are many other parameter estimation methods I planned on using, but was limited due to implementation details as well as hardware factors:

- Maximum Likelihood Expectation
- Bayesian Inference
- Stochastic Gradient Descent (SGD)

Some of these estimation methods require minimizing or maximizing some sort function. In the case of MLE, it is maximizing a likelihood function. I planned on using a built in implementation for a gradient descent algorithms, however it was giving me warnings as I do not have a built in GPU on my computer, so I decided to opt not to implement them.

Data Acquisition

Using **Yahoo Finance API**, I was able to use real stock data for modeling the problems, but also using it for testing my estimated solutions. [11] [12]

For example, using stock data up until the start of 2023 allows me to simulate data from 2023-2024 and compare the simulation to the real-world stock prices.

It is important to note since random processes are involved in this model, when I am simulating this code I ensure that each method uses the same random seed to ensure any variations in output are purely from the choice of model.

Model Validation

There are 3 main metrics commonly used for comparing methods like these to the true stock price: [7]

1. **Correlation Coefficient**

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

   -1 is perfect negative correlation, 0 is no correlation, and +1 is perfect positive correlation.

2. **Mean Absolute Percentage Error (MAPE)**

$$\text{MAPE} = \frac{\sum_{t=1}^{n} \left| \frac{A_t - F_t}{F_t} \right|}{n}$$

   $< 10\%$ is highly accurate, $10\% - 20\%$ are pretty good, $20\% - 50\%$ is reasonable, and anything more is an inaccurate forecast.

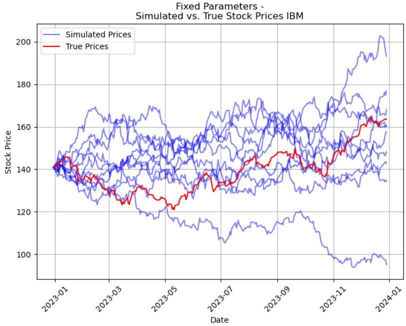
3. **Percentage Of Correct Predictions (Within Threshold)** returns the percentage of 'inliers' for points on our simulation within 10% of the true value of the stock price.

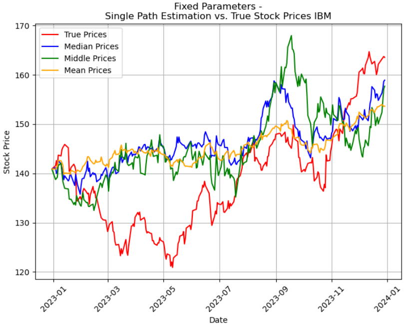
Analysis and Interpretation

I am going to simulate a number of SP500 stocks from the dates "2023-01-01" to "2024-01-01". I will point out any interesting results and observations and then attempt to accumulate all the data collectively.

Showing all of the results from all of the stock data is unrealistic on these slides, so I will instead opt to show the outputs of one stock ('IBM') as well as demonstrate the collective numerical analysis values such as Correlation Coefficient, MAPE, and % Inliers.

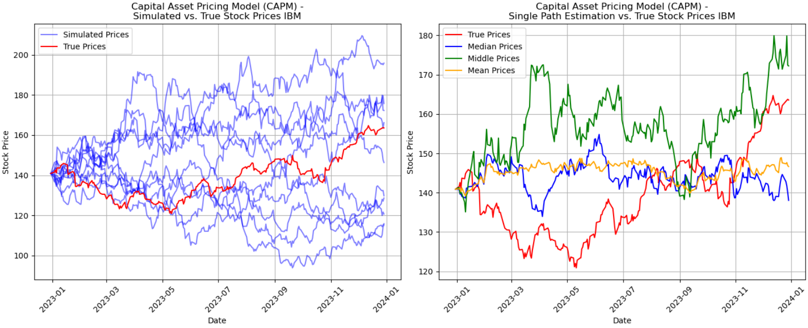
'IBM' Fixed Parameters





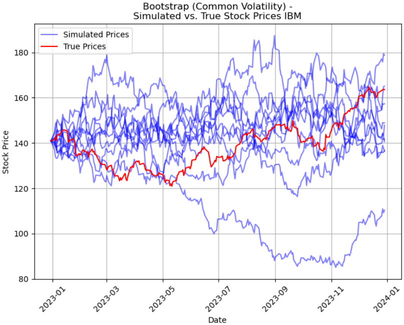
Analysis Group	Correlation Coeff	MAPE	% Inliers
Multiple Paths	0.299209	0.106161	53.9841
Average of Single Paths	0.630222	0.0690384	73.9708

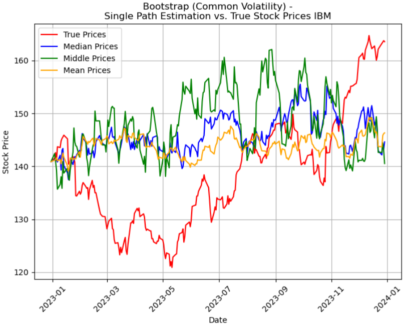
'IBM' Capital Asset Pricing Model (CAPM)



Analysis Group	Correlation Coeff	MAPE	% Inliers
Multiple Paths	-0.106906	0.152252	40.4382
Average of Single Paths	-0.128619	0.101011	56.1753

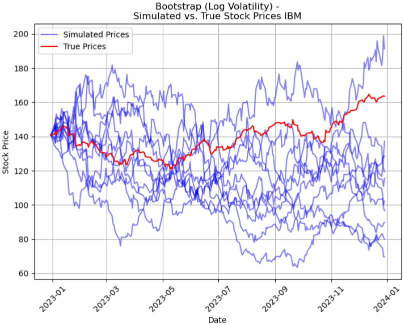
'IBM' Boostrap Method (Common Volatility)

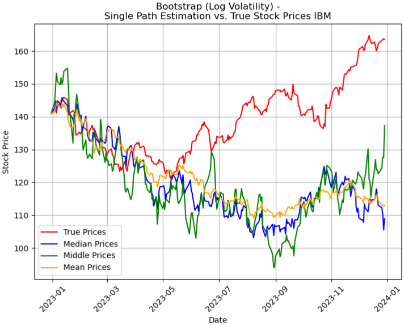




Analysis Group	Correlation Coeff	MAPE	% Inliers
Multiple Paths	0.0577446	0.11127	54.2629
Average of Single Paths	0.151154	0.0828053	60.6906

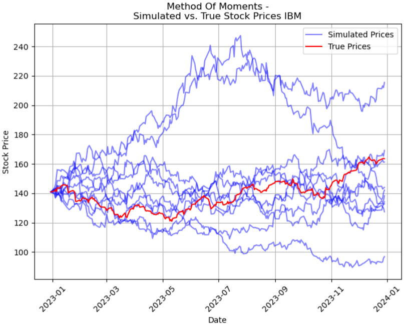
'IBM' Boostrap Method (Log Volatility)

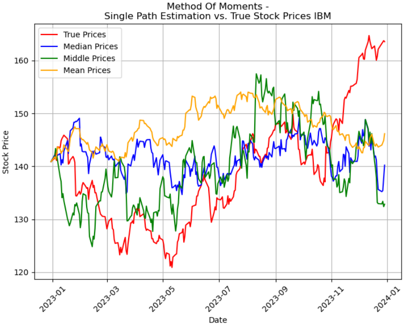




Analysis Group	Correlation Coeff	MAPE	% Inliers
Multiple Paths	-0.190193	0.184571	32.8287
Average of Single Paths	-0.260495	0.141545	41.9655

‘IBM’ Method Of Moments





Analysis Group	Correlation Coeff	MAPE	% Inliers
Multiple Paths	0.0589796	0.158986	51.8327
Average of Single Paths	0.24243	0.0708245	71.1819

Analysis Results

Below is the output of the code in the terminal. It is comprised of the tables I listed on the above slides.

Analysis For: IBM

Method Name	Analysis Group	Correlation Coefficient	MAPE	Percentage Inliers
Fixed Parameters	Multiple Paths	0.299209	0.106161	53.9841
Fixed Parameters	Average of Single Paths	0.630222	0.0690384	73.9708
Capital Asset Pricing Model (CAPM)	Multiple Paths	-0.106906	0.152252	40.4382
Capital Asset Pricing Model (CAPM)	Average of Single Paths	-0.128619	0.101011	56.1753
Bootstrap (Common Volatility)	Multiple Paths	0.0577446	0.11127	54.2629
Bootstrap (Common Volatility)	Average of Single Paths	0.151154	0.0828053	60.6906
Bootstrap (Log Volatility)	Multiple Paths	-0.190193	0.184571	32.8287
Bootstrap (Log Volatility)	Average of Single Paths	-0.260495	0.141545	41.9655
Method Of Moments	Multiple Paths	0.0589796	0.158986	51.8327
Method Of Moments	Average of Single Paths	0.24243	0.0708245	71.1819

**I can also specify the table to show each single path individual statistics.*

General Analysis Over Multiple Stocks

Stocks I used: AAPL, DG, DIS, IBM, INTC, NVDA, PG, SHW, TXN

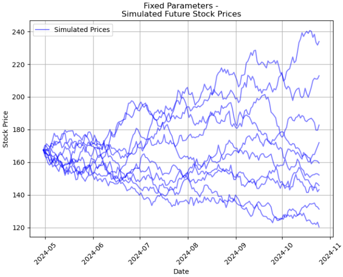
Averaging the values of each comparison metric for all the stocks listed above of each method:

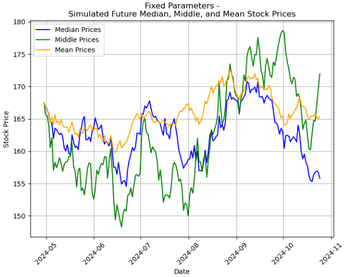
Method Name	Correlation Coeff	MAPE	% Inliers
Fixed Parameter	0.211876	0.2253658556	44.54773778
CAPM	-0.1464773333	0.2304961667	39.73735444
Bootstrap (Common)	**0.2383370343**	**0.1960659778**	40.62269667
Bootstrap (Log)	-0.1258122889	0.2947922222	25.01106556
Method of Moments	0.16705	0.2122124667	**50.67136556**

In general, all the methods performed quite well on this set of stock data. Correlation Coefficient was quite low, indicating there wasn't much of a correlation between these sets of data, but in terms of MAPE and % inliers, the methods all performed quite well. 20% – 30% is considered to be a reasonable prediction for the stock price.

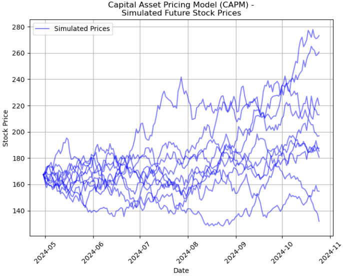
'AAPL' Simulating Future Stock Prices I

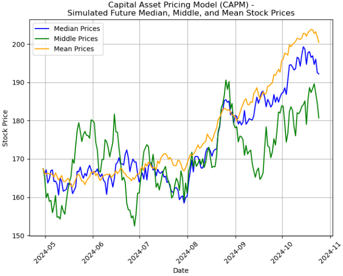
For fun, let's simulate the future stock price of 'AAPL' using this model:

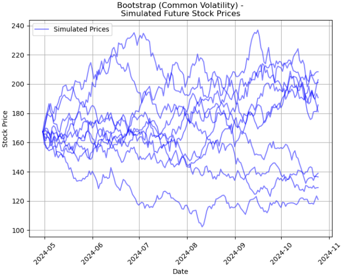


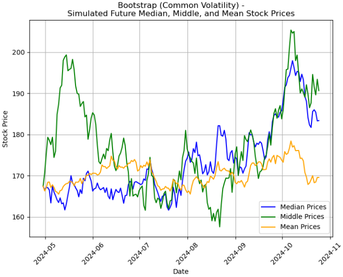


‘AAPL’ Simulating Future Stock Prices II

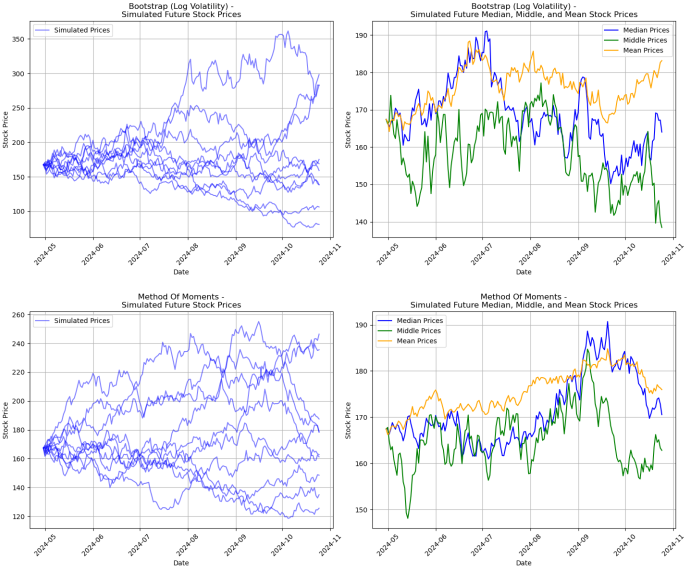








'AAPL' Simulating Future Stock Prices III



Try It For Yourself

You can find the approximately 1500 lines of python code at the Github repository below:

https://github.com/RyanTavol/Brownian-Motion-Stock-Model

Please feel free to take a look at the repository. I have provided very descriptive instructions on the Github readme as well as comments in the code to direct you on how to run it and make changes.

I have provided functionality for simulating (and comparing) any individual parameter evaluation method and comparing it to the true stock price over that interval as well as doing that for all of the methods at once. You can also simulate future stock prices for using individual methods or all of them together.

References I

MIT, "Stochastic processees 1," *Topics in Mathematics with Applications in Finance*, 2013. [Online]. Available: https://ocw.mit.edu/courses/18-s096-topics-in-mathematics-with-applications-in-finance-fall-2013/f5784e4facf3de690210d17c97358eba_MIT18_S096F13_lecnote5.pdf

U. of Chicago, "Brownian motion." [Online]. Available: https://galton.uchicago.edu/~lalley/Courses/313/WienerProcess.pdf

T. Sauer, "Numerical solutions of stochastic differential equations in finance," *George Mason University*. [Online]. Available: https://math.gmu.edu/~tsauer/pre/sde.pdf

Y. Yoo, "Stochastic calculus and black-scholes model," *UChicago*. [Online]. Available: https://math.uchicago.edu/~may/REU2017/REUPapers/Yoo.pdf

K. Siegrist, "Geometric brownian motino," *LibreTexts Statistics*, 2022. [Online]. Available: https://stats.libretexts.org/Bookshelves/Probability_Theory/Probability_Mathematical_Statistics_and_Stochastic_Processes_(Siegrist)/18%3A_Brownian_Motion/18.04%3A_Geometric_Brownian_Motion

References II

QuantStart, “Geometric brownian motion.” [Online]. Available: https://www.quantstart.com/articles/Geometric-Brownian-Motion/

K. Reddy and V. Clinton, “Simulating stock prices using geometric brownian motion: Evidence from australian companies,” *AABFJ*, 2016.

P. S. University, “Bootstrapping methods,” *Applied Statistics*. [Online]. Available: https://online.stat.psu.edu/stat500/lesson/11/11.2/11.2.1

W. F. A. et al, “Stock price prediction using geometric brownian motion,” *Journal Of Physics: Conference*, 2018. [Online]. Available: https://iopscience.iop.org/article/10.1088/1742-6596/974/1/012047/pdf

P. S. University, “Method of moments,” *Introduction to Mathematical Statistics*. [Online]. Available: https://online.stat.psu.edu/stat415/lesson/1/1.4

“Exploring finance apis,” *Python Invest*, 2020. [Online]. Available: https://pythoninvest.com/long-read/exploring-finance-apis

References III

A. Neto, "Python stock price api's," *Medium*, 2023. [Online]. Available: https://medium.com/@aguimarneto/python-stock-price-apis-e67d5310f6e3

T. Areerak, "Mathematical model of stock prices via a fractional brownian motion model with adaptive parameters," *Hindawi*, 2014. [Online]. Available: https://www.hindawi.com/journals/isrn/2014/791418/

J. Lidén, "Stock price predicitons using a geometric brownian motion," *Uppsala University*, 2018. [Online]. Available: https://www.diva-portal.org/smash/get/diva2:1218088/FULLTEXT01.pdf

QuantStart, "Stochastic differential equations." [Online]. Available: https://www.quantstart.com/articles/Stochastic-Differential-Equations/