WiDS '22 - '23 Final Documentation

Project UID 11
Machine Learning for Algorithmic Trading

Introduction to Problem Statement

Monte Carlo Simulation Prediction of Stock Prices.
Most time-series libraries for any preferred programming language give point predictions and prediction intervals when forecasting. This is of course very useful for making predictions about the future. But what if you want to know the probability that a future value is above some important threshold? An example would be that a stock price in the future is above or below the strike price of an option. So, we aim to propose a probability based model that is capable of doing the afore-mentioned.

Existing Resources

Introduction
https://www.investopedia.com/terms/m/montecarlosimulation.asp
https://www.ibm.com/in-en/cloud/learn/monte-carlo-simulation

Sample Application
https://medium.com/analytics-vidhya/monte-carlo-simulations-for-predicting-stock-prices-python-a64f53585662
https://www.interviewqs.com/blog/intro-monte-carlo

Research Paper
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9719349&tag=1
https://arxiv.org/pdf/2212.00197.pdf

Proposed Solution

Introductory Financial Terminology:

1. **STOCKS AKA EQUITY:** A stock, also known as equity, is a security that represents the ownership of a fraction of the issuing corporation. Units of stock are called "shares" which entitles the owner to a proportion of the corporation's assets and profits equal to how much stock they own.

A stock is a form of security that indicates the holder has proportionate ownership in the issuing corporation and is sold predominantly on stock exchanges.
Corporations issue stock to raise funds to operate their businesses.
There are two main types of stock: common and preferred.

2. **PERIODIC DAILY RETURN(PDR)**:is the periodic rate of change(the rate that the asset increased or decreased in value) for one day.
   Periodic returns can be added together to obtain weekly, monthly or yearly rates of return
   The closing prices of assets are considered in computing it.
   todays stock price = yesterdays stock price * e^r
   The periodic return from stocks calculates the stock's growth for a given period of time.

3. **DERIVATIVES:**
   The term derivative refers to a type of financial contract whose value is dependent on an underlying asset, group of assets, or benchmark. A derivative is set between two or more parties that can trade on an exchange or over-the-counter (OTC).

4. **VOLATILITY:**
   Volatility is a measure of the rate of fluctuations in the price of a security over time. It indicates the level of risk associated with the price changes of a security. Investors and traders calculate the volatility of a security to assess past variations in the prices to predict their future movements.
   Volatility is determined either by using the standard deviation or beta. Standard deviation measures the amount of dispersion in a security's prices. Beta determines a security's volatility relative to that of the overall market. Beta can be calculated using regression analysis.

5. **OPTIONS:**
   An option is a derivative, a contract that gives the buyer the right, but not the obligation, to buy or sell the underlying asset by a certain date (expiration date) at a specified price (strike price). There are two types of options: calls and puts.
   American-style options can be exercised at any time prior to their expiration.
   European-style options can only be exercised on the expiration date.
   To enter into an option contract, the buyer must pay an option premium. The two most common types of options are calls and puts:

   1**. Call options**
   Calls give the buyer the right, but not the obligation, to buy the underlying asset at the strike price specified in the option contract. Investors buy calls when they believe the price of the underlying asset will increase and sell calls if they believe it will decrease.

   2. **Put options**

Puts give the buyer the right, but not the obligation, to sell the underlying asset at the strike price specified in the contract. The writer (seller) of the put option is obligated to buy the asset if the put buyer exercises their option. Investors buy puts when they believe the price of the underlying asset will decrease and sell puts if they believe it will increase.

6. **DISPERSION:**
   This range of possible investment returns is called dispersion. In other words, dispersion refers to the range of potential outcomes of investments based on historical volatility or returns.
   Dispersion can be measured using alpha and beta, which calculate risk-adjusted returns and returns relative to a benchmark index, respectively.
   Generally speaking, the higher the dispersion, the riskier an investment is, and vice versa.
   Understandably, since the dispersion of possible returns on an asset provides insight about the volatility and risk associated with holding that asset. The more variable the return on an asset, the more risky or volatile it is.'
   Dispersion uses statistical ratios and measures such as alpha and beta, which, respectively, determines whether the investment outperforms the market, as well as its riskiness relative to a market index.
   The higher the positive alpha, the more the investment outperforms the market, while a negative alpha illustrates underperformance.
   Beta can range between 1, >1, or <1, where 1 implies the same risk as the market index, >1 equates to more risk than the market, and <1 indicates less riskiness than the market.

7. **DIVIDEND:**
   A dividend is a distribution of a portion of a company's earnings. Companies can choose to regularly reward their shareholders by paying dividends, usually in cash, although sometimes in stock. Companies that consistently generate more profits than management and can efficiently reinvest in the business often choose to start paying dividends.

8. **RANDOM WALK THEORY:**
   suggests that changes in stock prices have the same distribution and are independent of each other.
   Random walk theory infers that the past movement or trend of a stock price or market cannot be used to predict its future movement.
   Random walk theory believes it's impossible to outperform the market without assuming additional risk.
   Random walk theory considers technical analysis undependable because it results in chartists only buying or selling a security after a move has occurred.

Random walk theory considers fundamental analysis undependable due to the often-poor quality of information collected and its ability to be misinterpreted. Random walk theory claims that investment advisors add little or no value to an investor's portfolio.

9. **DRIFT:**
   Drift simply means steady, gradual movement of some measure, usually towards an equilibrium. The alternative would be a jump, such as where a stock price moves suddenly to an equilibrium price in reaction to something (news, other economic factors, etc.). A drift is more gradual, with the equilibrium being reached over a longer period of time.

10. **Paper Trade:**
    A paper trade is a simulated trade that allows an investor to practice buying and selling without risking real money. The term dates back to a time when (before the proliferation of online trading platforms) aspiring traders would practice on paper before risking money in live markets. While learning, a paper trader records all trades by hand to keep track of hypothetical trading positions, portfolios, and profits or losses.

What Is a Monte Carlo Simulation?

A Monte Carlo simulation is used to model the probability of different outcomes in a process that cannot easily be predicted due to the intervention of random variables. It is a technique used to understand the impact of risk and uncertainty. A Monte Carlo simulation takes the uncertain variable and assigns it a random value. The model is then run, and a result is provided. This process is repeated repeatedly while assigning many different values to the variable. Once the simulation is complete, the results are averaged to arrive at an estimate.

How to use Monte Carlo methods in general?

Regardless of what tool you use, Monte Carlo techniques involve three basic steps:

1. Set up the predictive model, identifying the dependent variable to be predicted and the independent variables (also known as the input, risk, or predictor variables) that will drive the prediction.
2. Specify probability distributions of the independent variables. Use historical data and/or the analyst's subjective judgment to define possible values and assign probability weights for each.
3. Run simulations repeatedly, generating random values of the independent variables. Do this until enough results are gathered to make up a representative sample of the infinite number of possible combinations.

How Is the Monte Carlo Simulation Used in Finance Applications?

The Monte Carlo simulation estimates the probability of a certain income. As such, it is widely used by investors and financial analysts to evaluate the probable success of investments they're considering. Some common uses include

- **Pricing stock options**. The potential price movements of the underlying asset are tracked given every possible variable. The results are averaged and then discounted to the asset's current price. This is intended to indicate the probable payoff of the options.
- **Portfolio valuation**. The Monte Carlo simulation can test several alternative portfolios to measure their comparative risk.
- **Fixed income investments.** The short rate is the random variable here. The simulation calculates the probable impact of movements in the short rate on fixed-rate investments.

Limitations

•It only gives statistical measures, not exact results.
•It is a complex process and requires other metrics to evaluate results

Formulation

$$P_t = P_{t-1} * e^r \qquad (1)$$

r = return Pt = price at time t

The return r is random in behavior and follows a stochastic process which can be modeled as Brownian motion, which has two components named drift and volatility.
Drift — the direction that rates of returns have had in the past. That is the expected return of the stock.
Volatility — the historical volatility multiplied by a random, standard normal variable.

$$\text{drift} = \mu - \tfrac{1}{2}\sigma^2$$
$$\text{volatility} = \sigma Z[Rand(0,1)]$$

Hence equation (1) can be rewritten as

$$P_t = P_{t-1} * e^{(\mu - \frac{1}{2}\sigma^2) + \sigma Z[Rand(0,1)]} \quad (2)$$

Python code for the above formulation

```
stdev = log_return.std()
days = 50

#using 10000 trails of monte carlo
trials = 10000

#generating random numbers for 50 days
random_nos = np.random.rand(days, trials)#days, trials

#ppf gives z-score given probability
Z = norm.ppf(random_nos)

daily_returns = np.exp(drift + stdev * Z)
```

- For applying predictive models to time series data, **few assumptions** are to be satisfied like data should be free from autocorrelation, should follow stationarity assumptions. Let's delve into these topics and become familiar with them.

Starting out first with some important terminology:

Summary statistic:

Summary statistics summarize and provide information about your sample data. It tells you something about the values in your data set. This includes where the mean lies and whether your data is skewed. Summary statistics fall into three main categories:

- Measures of location (also called central tendency).
- Measures of spread.
- Graphs/charts

Time series:

A time series is a sequence of data points that occur in successive order over some period of time. In investing, a time series tracks the movement of the chosen data points, such as a security's price, over a specified period of time with data points recorded at regular intervals.

Unit root process:

A unit root (also called a unit root process or a difference stationary process) is a stochastic trend in a time series, sometimes called a "random walk with drift"; If a time series has a unit root, it shows a systematic pattern that is unpredictable.

Reason behind the name
The reason why it's called a unit root is because of the mathematics behind the process. At a basic level, a process can be written as a series of monomials (expressions with a single term). Each monomial corresponds to a root. If one of these roots is equal to 1, then that's a unit root.

Statistical power (aka sensitivity):

The statistical power of a study (sometimes called sensitivity) is how likely the study is to distinguish an actual effect from one of chance. It's the likelihood that the test is correctly rejecting the null hypothesis (i.e. "proving" your hypothesis).

Order of integration:

"Order of integration" is a summary statistic used to describe a unit root process in time series analysis. Specifically, it tells you the minimum number of differences needed to get a stationary series.
A series with a unit root (a random walk) is said to be integrated of order one, or I(1) – A stationary series without a trend is said to be integrated of order 0, or I(0) – An I(1) series is differenced once to be I(0) – In general, we say that a series is I(d) if its dth difference is stationary.

Autocorrelation

It is a type of series independence. It is the degree of similarity between the current time series and the lagged version of itself. The autocorrelation between the successive terms (u2 and u1), (u3 and u2)....(u(n) and u(n-1)) gives the autocorrelation of order one. Autocorrelation should also be checked for the residuals.Autocorrelation is a mathematical representation of the degree of similarity between a given time series and a lagged version of itself over successive time intervals. It's conceptually similar to the correlation between two different time

series, but autocorrelation uses the same time series twice: once in its original form and once lagged one or more time periods.

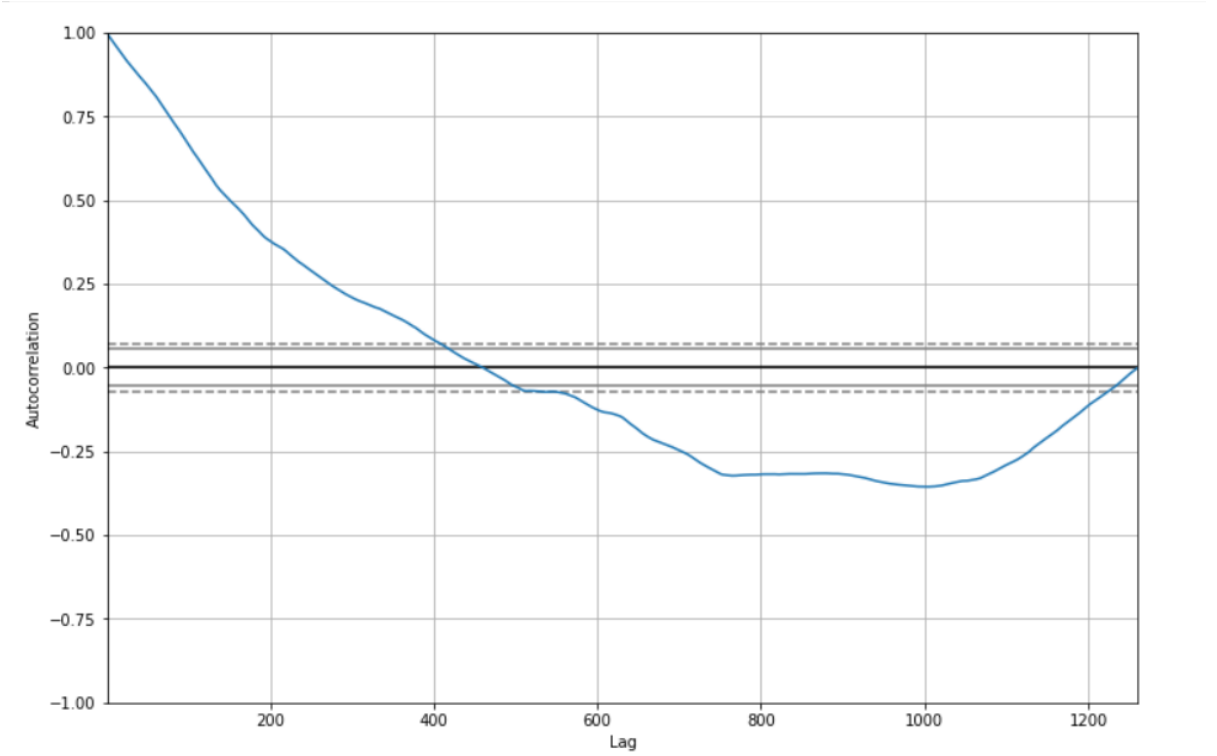
- Autocorrelation measures the relationship between a variable's current value and its past values.
- An autocorrelation of +1 represents a perfect positive correlation, while an autocorrelation of negative 1 represents a perfect negative correlation.
- Technical analysts can use autocorrelation to measure how much influence past prices for a security have on its future price.
- Autocorrelation can also be referred to as lagged correlation or serial correlation, as it measures the relationship between a variable's current value and its past values.

An autocorrelation of +1 represents a perfect positive correlation (an increase seen in one time series leads to a proportionate increase in the other time series).

On the other hand, an autocorrelation of -1 represents a perfect negative correlation (an increase seen in one time series results in a proportionate decrease in the other time series). Autocorrelation measures linear relationships. Even if the autocorrelation is minuscule, there can still be a nonlinear relationship between a time series and a lagged version of itself

Some of the possible reasons for the introduction of autocorrelation in the data are as follows:
1. Carryover of effect, at least in part, is an important source of autocorrelation. For example, the monthly data on household expenditure is influenced by the expenditure of the preceding month. In time-series data, time is the factor that produces autocorrelation. Whenever some ordering of sampling units is present, the autocorrelation may arise.
2. Another source of autocorrelation is the effect of deletion of some variables. In regression modeling, it is not possible to include all the variables in the model. There can be various reasons for this, e.g., some variables may be qualitative, sometimes direct observations may not be available on the variable etc. The joint effect of such deleted variables gives rise to autocorrelation in the data.

Tests for autocorrelation:
We used an autocorrelation plot available in the pandas library, following a graph we obtained for different values of lags.



It shows that for lag=0, the data is fully correlated and uncorrelated for the lag in the range of 450-470.

Stationarity

Data is said to be stationary if the various statistical measures of data such as mean, median, variance etc. does not change with time. It means that data is without trend, constant variance over time, a constant autocorrelation structure over time and no periodic fluctuations (seasonality). A data series is said to be stationary if its mean and variance are constant (non-changing) over time and the value of covariance between two time periods depends only on the distance or lag between the two time periods and not on the actual time at which the covariance is computed.

IMPORTANCE OF STATIONARITY: Most forecasting methods assume that a distribution has stationarity. For example, autocovariance and autocorrelations rely on the assumption of stationarity. An absence of stationarity can cause unexpected or bizarre behaviors, like t-ratios not following a t-distribution or high r-squared values assigned to variables that aren't correlated at all. A stationary time series is one whose statistical properties such as mean, variance, autocorrelation, etc. are all constant over time. A stationarized series is relatively easy to predict --you simply predict that its statistical properties will be the same in the future as they have been in the past!

If the time series is not stationary, we can often transform it to stationarity with one of the following techniques.

1. We can differentiate the data. That is, given the series Z(t), we can create new series as Y(t) = Z(t) - Z(t-1). The differenced data will contain one less point than the original data. Although we can differentiate the data more than once, one difference is usually sufficient.
2. If the data contain a trend, we can fit some type of curve to the data and then model the residuals from that fit. Since the fit's purpose is to remove the long-term trend, a simple fit, such as a straight line, is typically used.
3. For non-constant variance, taking the logarithm or square root of the series may stabilize the variance. For negative data, we can add a suitable constant to make all the data positive before applying the transformation. This constant can then be subtracted from the model to obtain predicted (i.e., the fitted) values and forecasts for future points.

Test for stationarity

We used the Augmented Dickey-Fuller test, which can also be used to check stationarity and non-stationarity for large-sized sets of time series models. The Augmented Dickey-Fuller test is based on the following hypothesis.
Null hypothesis: A unit root exists in the time series and is non-stationary.
Alternate hypothesis: No unit root exists in the time series and is stationary or trend stationary.

For our data, we used the stats model library. We applied adfuller method to check the p-value, which was found to be more than 0.05, which implies the null hypothesis is accurate so we made data stationary using the first method and took differencing time series.

Model selection:

We used Auto Regressive Integrated Moving Average method (ARIMA models) to analyze the stock data because if the data exhibits no apparent deviations from stationarity and has a rapidly decreasing autocorrelation function, we shall seek a suitable ARMA (Auto Regressive Moving Average) process to represent the mean-correlated data. This can frequently be achieved by differencing, leading us to consider the class of ARIMA processes. ARIMA models are generally denoted as ARIMA (p,d,q) where p is the order of the autoregressive model, d is the degree of differencing, and q is the order of moving-average model. ARIMA models use differencing to convert a non-stationary time series into a stationary one, and then

predict future values from historical data. These models use “auto” correlations and moving averages over residual errors in the data to forecast future values.

ARIMA Model terminology:

- **AutoRegressive - AR(p)** is a regression model with lagged values of y, until p-th time in the past, as predictors. Here, p = the number of lagged observations in the model, ε is white noise at time t, c is a constant and φs are parameters.
- **Integrated I(d)** - The difference is taken d times until the original series becomes stationary. A stationary time series is one whose properties do not depend on the time at which the series is observed.
- **Moving average MA(q)** - A moving average model uses a regression-like model on past forecast errors. Here, ε is white noise at time t, c is a constant, and θs are parameters

Combining all of the three types of models above gives the resulting ARIMA(p,d,q) model.

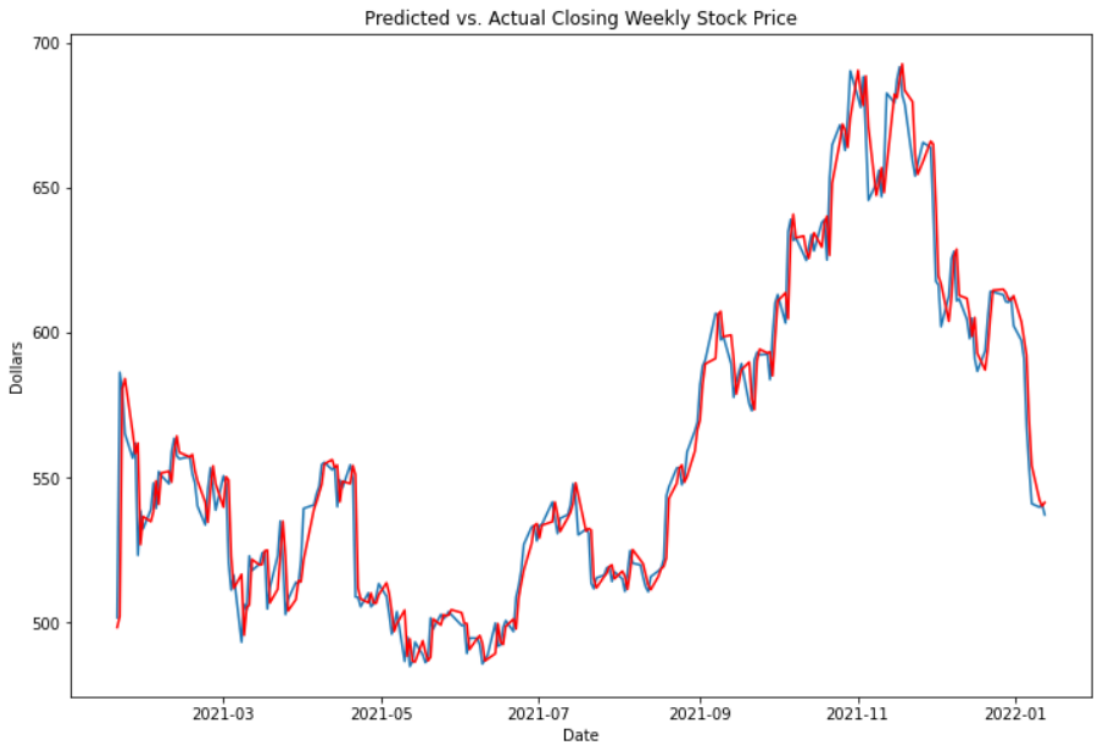
- A process X_t is said to be $ARIMA(p, d, q)$ if

$$\nabla^d X_t = (1 - B)^d X_t$$

  is $ARMA(p, q)$.
- In general, $ARIMA(p, d, q)$ model can be written as:

$$\phi(B)(1 - B)^d X_t = \theta(B) w_t$$

Results obtained from the ARIMA model on validation data



Methodology & Progress (Mention the work done week-wise)

Week 1 - Getting accustomed to what Monte Carlo simulation means
Week 2 - Going through the sample application on various websites to study how the basic implementation works
Week 3 & 4 - Reading the research paper to get hang of the latest model of Monte Carlo and its advancements
Week 5 & 6 - Implementing the code based on the previous research paper and observing those trends

Results

Test data evaluated using root mean square value which found to be 6.485 in case of ARIMA model. There was negligible autocorrelation found in the data.

The link for the Github repo which contains the implemented code is as follows:

https://github.com/Aanuj55/Algorithmic-Trading

Summary of the Implementation Code

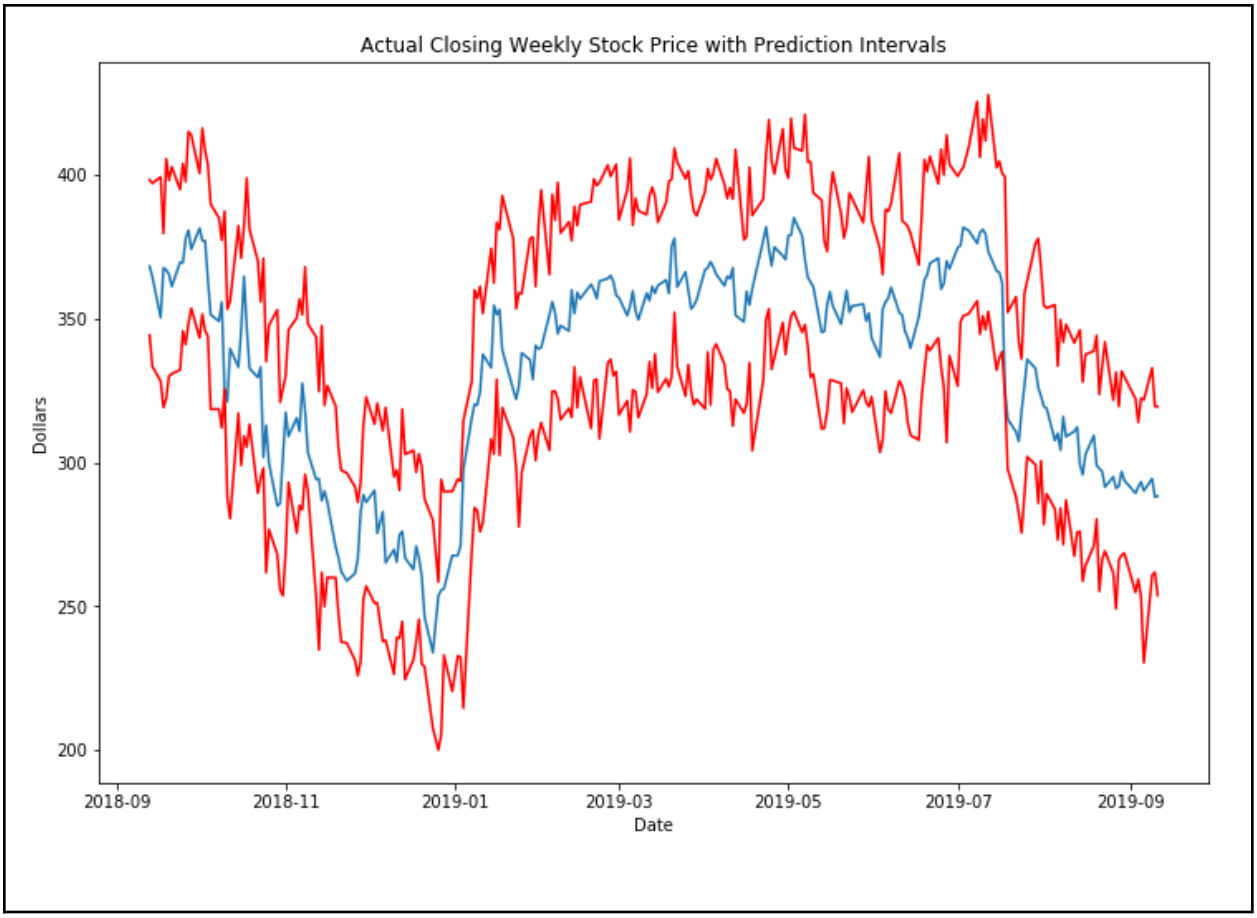
The script is a time series analysis of the historical closing stock prices of Netflix Inc. (NFLX) using Python programming language. The script uses various libraries such as DateTime, yfinance, pandas, NumPy, matplotlib, and statsmodels. The script is divided into several parts, each serving a specific purpose in analyzing the data.

The first part of the script imports the necessary libraries and sets the sample period as seven years back from 09/12/2019. The script then uses yfinance library to extract the closing price of NFLX for the past seven years. The data is stored in a pandas data frame and is renamed with lowercase column names. The script then splits the data into three subsets: my_stock_resid_distr_fitting, my_stock_train, and my_stock_test. my_stock_resid_distr_fitting is used for fitting a distribution for Monte Carlo simulation, my_stock_train for creating a training subset missing the final 250 observations, and my_stock_test for grabbing the final 250 observations for a test set. The script then plots the NFLX weekly closing price over the past seven years.
The second part of the script uses the augmented Dickey-Fuller Test to establish the order of integration of time series. The script performs ADF tests with no constant, a constant, and a constant plus linear trend on NFLX closing share prices and different NFLX closing share prices. The script prints the ADF statistic and p-value for each test.

The third part of the script uses the ARIMA model to fit the data. The script fits the ARIMA model of order (1, 1, 1) to the log-transformed my_stock_train data and prints the model summary. The script then plots the training data's residual errors and shows the residuals' distribution.

The fourth part of the script uses the rolling forecast to predict the next period's closing price of the NFLX stock. The script loops through the forecasted set's indexes and estimates an ARIMA model of order (1,1,1). The script then fits the model, forecasts the next period, and takes the predicted value. The script also prints the mean squared error of the model for the rolling forecast period. The script then plots the predicted and actual stock prices.

The fifth part of the script defines two functions: laplace_monte_carlo() and roll_forecast_nmc(). The first function, laplace_monte_carlo(), takes in three parameters: a mean, residuals, and the number of simulations to run. It then uses the NumPy function np.random.laplace() to generate an array of simulated values using the Laplace distribution. The Laplace distribution is a probability distribution where the random variable's probability density function is defined by its mean and standard deviation. The second function, roll_forecast_nmc(), takes in four parameters: train, test, std_dev, and n_sims. It then creates a new data frame with the log of the train data, creates an empty list to hold predictions, and loops through the indexes of the test set. Within the loop, it estimates an ARIMA model of order (1,1,1), fits the model, forecasts the next period, takes the predicted value, performs Monte Carlo simulation using the predicted price as the mean, the user-specified standard deviation, and several simulations, then appends the range of simulated prices to the list of predictions. Finally, it converts the predictions list to a pandas data frame with the same index as the actual values, converts all the estimated yhats in each column to one list per row and grabs only the column with all values in a list, and returns predictions. The last part of the code attaches the data withheld for investigating the forecast residuals back to the training data set, produces a rolling forecast with prediction intervals using 1000 MC sims, creates an empty list, loops through the rows in the testing data set, and appends true if the actual price is in the interval of predicted prices and false otherwise, and finally, prints the percentage of actual prices in the prediction intervals.



Learning Value

We proposed a probability based model which can be used for algorithmic trading which instead of solely predicting the future price of a stock or option, outputs the probability of the stock being in a particular range and thus is more comprehensive than the point based models.

Prediction intervals generated using Monte Carlo simulation can be used to predict the probability that an option is in the money or if a stock is a buy, hold, or sell over your chosen time horizon. For example, we calculated the probability that the stock price on 09/12/2019 is greater than or equal to 290 and got the probability of 36.7%.

Tech-stack Used

`Numpy, pandas, scikit learn, matplotlib, plotly, statsmodel, python`

Suggestions for others

Most financial professionals have some familiarity with the Monte Carlo method. Many learn of it as a tool financial engineers use for pricing derivatives.
Just try it out and you will love to see the results.

References and Citations

Weishi Wang - GAN-MC: a Variance Reduction Tool for Derivatives Pricing - Dec 2022