CS 171 Final Project White Paper

Alan Viollier, Jerry Zhao,
Victor La

December 1st, 2022
CS 171 Section 4
Introduction to Machine Learning
Group J

Contents

Business Background

Today's global market heavily emphasizes consumer sales and corporate gains as the means of global success. As a company rises in value, so does its potential capital gain, which is extremely attractive for investors. This directly introduces an essential economic specialization that focuses on stock analytics so that potential investors can make critical decisions to maximize profits and capital success. However, stock advisors are often an investment on their own and might not have consistent accuracy regarding their advice.

Objective

Our objective is to use machine learning to produce a model that can analyze a company's historic stock values to predict and forecast its future value accurately. We decided to choose the most popular stock APPL for our objective.

Data

The data we used to build, train, and test our forecasting model was taken from Yahoo Finances, which publicly broadcasts the historic stock values of all publicly traded companies. There is a Yahoo Finances Python module that allows us to do this.

```
import yfinance as yf
```

```
# Pull the data using yf.download which takes in arguments like tickers, start, end, and interval
stocknames = ["AAPL"]
df2y = yf.download(tickers=stocknames, start=start2year, end=yesterday, interval="1wk")
df3y = yf.download(tickers=stocknames, start=start3year, end=yesterday, interval="1wk")
df4y = yf.download(tickers=stocknames, start=start4year, end=yesterday, interval="1wk")
df5y = yf.download(tickers=stocknames, start=start5year, end=yesterday, interval="1wk")
```

The data format includes factors such as the value when the market opened, when the market closed, the local maxima and minima values given a weekly period, an adjusted closing value based on the company's dividends, and the volume of stock exchanged.

Our model focuses on the stock values of Apple Inc and uses historical data from up to 5 years ago. We decided to have different time periods to figure which time period created the most accurate model. We chose to use the “Close” variable for our predictions.

Variable Selection

We used the Cointegration Test to figure out which other variable is closely related to the “Close” variable. We found that the “Open” variable was the most significant and chose to use that as our exogenous variable later on.

Note: An exogenous variable should be outside of the system of the main variable, but this was outside the scope of this project as we did not have the resources or time. Therefore we resorted to using “Open” although it wouldn’t serve us much purpose.

```
2 year data type & shape:

Open          float64
High          float64
Low           float64
Close         float64
Adj Close     float64
Volume        float64
dtype: object

(105, 6)
```

```
3 years data type & shape:

Open          float64
High          float64
Low           float64
Close         float64
Adj Close     float64
Volume        float64
dtype: object

(157, 6)
```

```
4 year data type & shape:

Open          float64
High          float64
Low           float64
Close         float64
Adj Close     float64
Volume        float64
dtype: object

(210, 6)
```

```
5 years data type & shape:

Open          float64
High          float64
Low           float64
Close         float64
Adj Close     float64
Volume        float64
dtype: object

(262, 6)
```

```
Column Name     >  Test Stat > C(95%)    =>   Signif
--------------------------------------------------------
Close           >  116.79    > 83.9383   =>   True
Open            >  69.58     > 60.0627   =>   True
High            >  44.36     > 40.1749   =>   True
Low             >  24.06     > 24.2761   =>   False
Adj Close       >  10.89     > 12.3212   =>   False
Volume          >  0.53      > 4.1296    =>   False
--------------------------------------------------------
Column Name     >  Test Stat > C(95%)    =>   Signif
--------------------------------------------------------
Close           >  114.64    > 83.9383   =>   True
Open            >  70.0      > 60.0627   =>   True
High            >  36.65     > 40.1749   =>   False
Low             >  19.92     > 24.2761   =>   False
Adj Close       >  8.35      > 12.3212   =>   False
Volume          >  1.73      > 4.1296    =>   False
--------------------------------------------------------
Column Name     >  Test Stat > C(95%)    =>   Signif
--------------------------------------------------------
Close           >  147.62    > 83.9383   =>   True
Open            >  89.79     > 60.0627   =>   True
High            >  46.6      > 40.1749   =>   True
Low             >  27.42     > 24.2761   =>   True
Adj Close       >  12.75     > 12.3212   =>   True
Volume          >  2.22      > 4.1296    =>   False
--------------------------------------------------------
Column Name     >  Test Stat > C(95%)    =>   Signif
--------------------------------------------------------
Close           >  168.04    > 83.9383   =>   True
Open            >  101.59    > 60.0627   =>   True
High            >  52.57     > 40.1749   =>   True
Low             >  29.77     > 24.2761   =>   True
Adj Close       >  15.04     > 12.3212   =>   True
Volume          >  0.48      > 4.1296    =>   False
```

```
df2y
```

Date	Open	High	Low	Close	Adj Close	Volume
2020-11-30	116.970001	123.779999	116.809998	122.250000	120.799881	543370600.0
2020-12-07	122.309998	125.949997	120.150002	122.410004	120.957977	452278700.0
2020-12-14	122.599998	129.580002	121.540001	126.660004	125.157562	621538100.0
2020-12-21	125.019997	134.410004	123.449997	131.970001	130.404587	433310200.0
2020-12-28	133.990005	138.789993	131.720001	132.690002	131.116043	441102200.0
...	...	...	...	...	...	...

Data Cleaning

For data cleaning we decided to get rid of everything but the “Close” and “Open” variables. We also checked to see if there were any null variables, and there were not.

```
2 year data type & shape:

Open          float64
High          float64
Low           float64
Close         float64
Adj Close     float64
Volume        float64
dtype: object

(105, 6)
```

```
2 year data type & shape:

Open     float64
Close    float64
dtype: object

(105, 2)
```

```
print(df2y.isnull().sum())
print("-"*100)
print(df3y.isnull().sum())
print("-"*100)
print(df4y.isnull().sum())
print("-"*100)
print(df5y.isnull().sum())
```

```
Open     0
Close    0
dtype: int64
```

Splitting Data

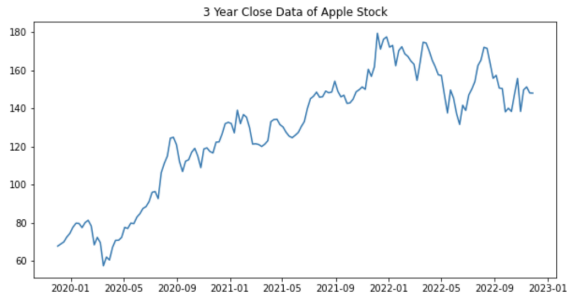
This data is split 90-10, where 90% of the data is used for model training, and the remaining 10% is used for model testing. We did this for all four of our different time frames.

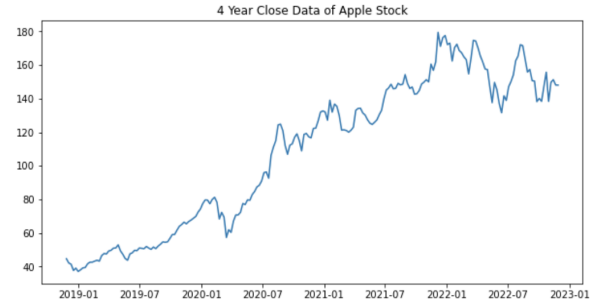
```
: # Here we will be splitting the data into 90% train - 10% test.

  # 2 years of data
  split2 = df2y.Close

  train2 = split2.iloc[:-split2.size//10]
  test2 = split2.iloc[-split2.size//10:]

  print(train2.shape)
  print(test2.shape)

  # 3 years of data
  split3 = df3y.Close

  train3 = split3.iloc[:-split3.size//10]
  test3 = split3.iloc[-split3.size//10:]

  print(train3.shape)
  print(test3.shape)

  # 4 years of data
  split4 = df4y.Close

  train4 = split4.iloc[:-split4.size//10]
  test4 = split4.iloc[-split4.size//10:]

  print(train4.shape)
  print(test4.shape)

  # 5 years of data
  split5 = df5y.Close

  train5 = split5.iloc[:-split5.size//10]
  test5 = split5.iloc[-split5.size//10:]

  print(train5.shape)
  print(test5.shape)
```

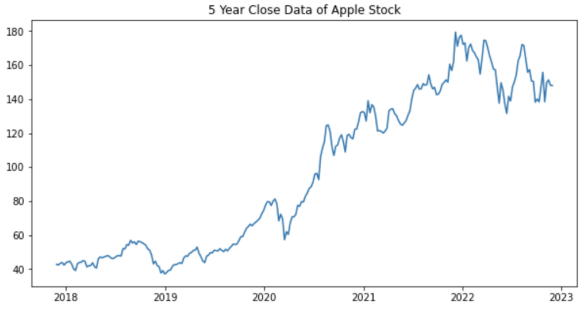
```
(94,)
(11,)
(141,)
(16,)
(189,)
(21,)
(235,)
(27,)
```

Data Check

To make sure everything looked good before we fit the models we graphed all the data. Everything looked great!







Methodology

The methodology we used to analyze and forecast the stock value of Apple Inc primarily takes advantage of time-series forecasting methods such as

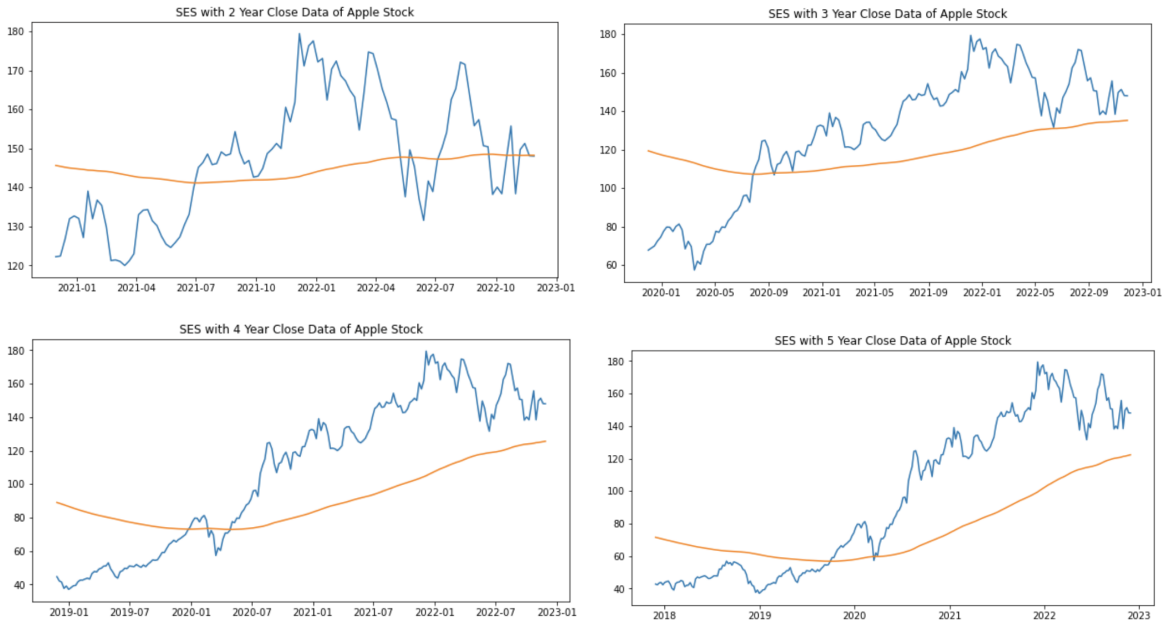
- Simple Exponential Smoothing (SES)
- Triple Exponential Smoothing (Holt-Winters)
    - Both Additive and Multiplicative Seasonality
- Autoregressive Integrated Moving Average (ARIMA)
- Seasonal ARIMA (SARIMA)
- SARIMA with Exogenous Regressors (SARIMAX)

Simple Exponential Smoothing (SES)

Exponential smoothing is an adaptation of simple moving average. Rather than taking the average, it a weighted average of past values. A value that is further back will count less and a more recent value will count more.

We do not expect this to be the best or work efficiently. When trends are present SES it does not work well, as the model cannot make the distinction between variation and trend correctly.

SES Models



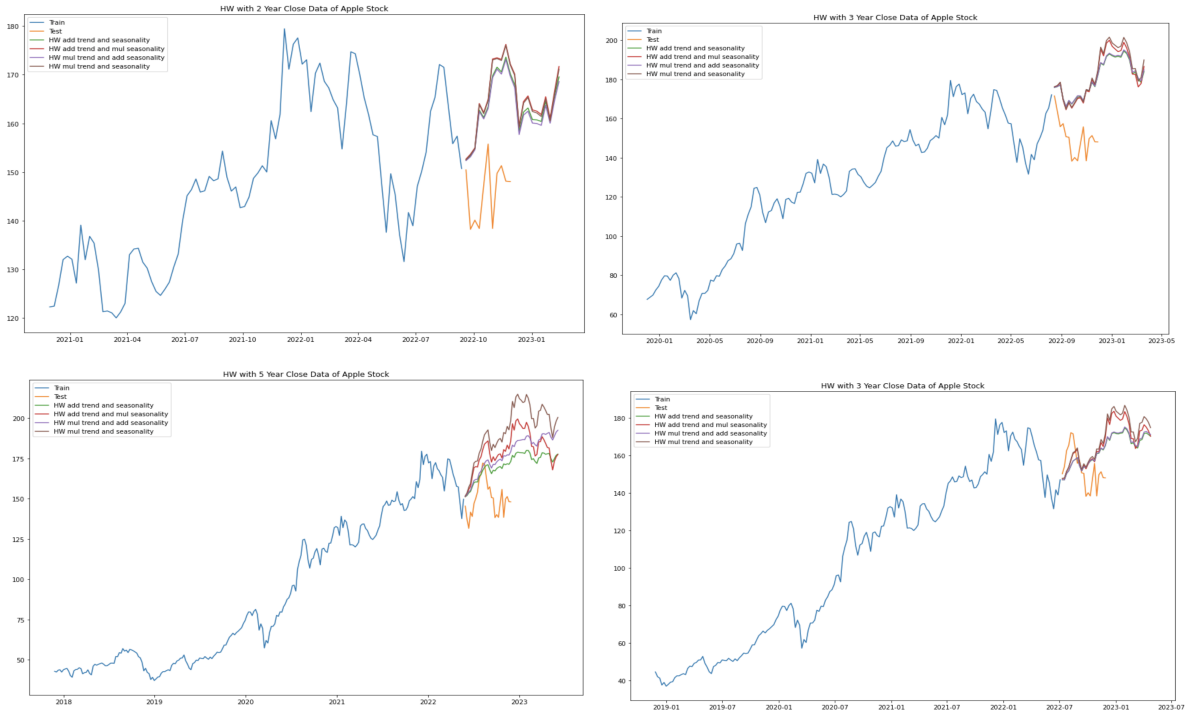
As we can see the only signal we can observe with SES is trend. We can observe that as we get a larger and larger timeframe the trend seems to be going higher. In every case however, it

seems that the trend is accelerating downwards even if its still going up. We wouldn't make any stock predictions based on this data.

Triple Exponential Smoothing (Holt-Winters)

Holt proposed a method for seasonal data using triple exponential smoothing. His method was studied by Winters, and so now it is usually known as "Holt-Winters' method". Holt-Winters' method is based on three smoothing equations one for the level, one for trend, and one for seasonality. This will definitely work better than SES but will probably ultimately be worse than SARIMA.

HW Models



These predictions are fine and look a lot better than than SES.

The stock market can be unpredictable so the Holt Winters couldn't really predict Apple stock slowing down from constantly rising.

The 3 & 5 year timeframe looks like the prediction closest to test. Based on this information we could guess that Apple stock will have an upcoming bump followed by a drop but still an overall upward trend.

Autoregressive Integrated Moving Average (ARIMA)

The ARIMA family of models is a set of smaller models that can be combined. Each part of the ARMIA model can be used as a stand-alone component, or the different building blocks can be combined.

These smaller models are:

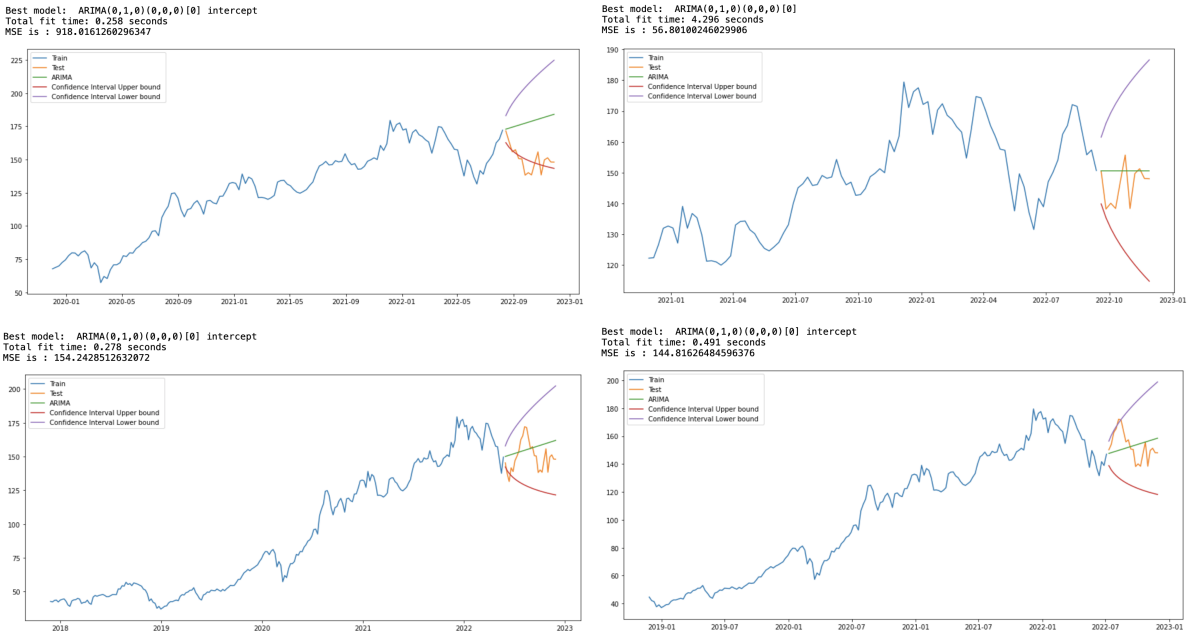
Autoregression (AR).
It's a regression model that explains a variable's future value using its past values.

Moving average (MA).
It also uses past values to predict the current value of the variable. However, the Moving Average uses the prediction error in previous time steps to predict the future.

Autoregressive moving average (ARMA).
This model combines the two previous building blocks into one model. ARMA can therefore use both the value and the prediction errors from the past.

Finally we have, Autoregressive integrated moving average (ARIMA).
The ARIMA model adds automatic differencing to the ARMA model. You can also set to the number of times that the time series needs to be differenced.

ARIMA Models



We can see here that the ARIMA model is fine but lacks precision due to it's simplicity and lack of seasonality. All of the timeframes performed ok except the 3 year timeframe which performed horribly. We would say this is due to stock market unpredictability.

We would not make any stock predictions based on this but overall it seems apple stock has an upward trend in the longrun but is pretty uninteresting in the short term.

Seasonal ARIMA (SARIMA)

SARIMA simply adds seasonal effects into the ARIMA model. If seasonality is present in your time series, it is very important to use it in your forecast.

SARIMA Models

For 2 years of Apple stock data seasonality = 52
ARIMA(0,1,1)(0,1,0)[52] AIC=292.715 seems to be the best because the AIC is the lowest
MSE = 167.51463805579658
For 3 years of Apple stock data seasonality = 52
ARIMA(0,1,1)(0,1,1)[52] AIC=594.970 seems to be the best because it has the lowest AIC
MSE = 614.2349896911138
For 4 years of Apple stock data seasonality = 52
ARIMA(0,1,0)(2,1,0)[52] AIC=871.052 seems to be the best because it has the lowest AIC
MSE = 170.36248521972735
For 5 years of Apple stock data seasonality = 52
ARIMA(0,1,0)(0,1,2)[52] AIC=1103.524 seems to be the best because it has the lowest AIC
MSE = 475.2424274429505

We can see here that the SARIMA model is pretty good with the additional seasonality. All of the timeframes performed pretty well.

We are still unsure as to if we'd make confident stock predictions with this information but this data is telling us that Apple might not do too well in the future! The stock market will always be unpredictable though.

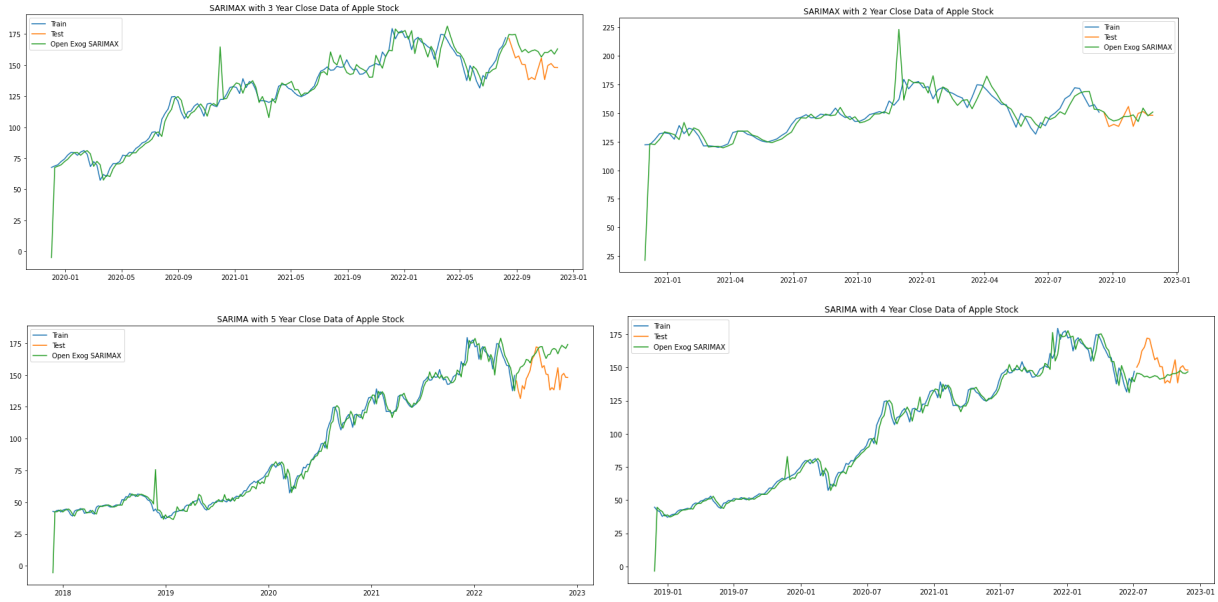
SARIMA with Exogenous Regressors (SARIMAX)

The most complex variant is the SARIMAX model. It regroups AR, MA, differencing, and seasonal effects. On top of that, it adds the X: external variables. If you have any variables that could help your model to improve, you could add them with SARIMAX.

We used “Open” for our exogenous variable but it did not change anything from SARIMA. The Exogenous variable did not help us in this case.

As Stated Before: An exogenous variable should be outside of the system of the main variable, but this was outside the scope of this project as we did not have the resources or time. Therefore we resorted to using “Open” although it wouldn’t serve us much purpose.

SARIMAX Models



Ensemble Technique Applied

We did not find the need to apply and ensemble technique to our time series. Our models performed well and as expected. It would be extra fluff to apply an ensemble technique.

Assumptions

In this paper, we make several assumptions about the state of the data we are using as well as the knowledge of the audience to which this paper is intended.

- The audience of this paper is familiar with the basics of stock market analysis and forecasting, as well as machine learning techniques for time series analysis.
- The historical data from Yahoo Finance used in this paper is complete and accurate.
- The machine learning techniques used in this paper (SES, Holt-Winters, ARIMA, SARIMA, and SARIMAX) have been tested and shown to produce accurate predictions on this data.
- The predictions made in this paper are based on the assumption that the future of Apple's stock will be similar to its past performance and that the trends and patterns identified in the historical data by our models will continue into the future.

Equations

Simple Exponential Smoothing (SES)

$$L_t = \sum_{i=0}^{t-1} \alpha(1-\alpha)^i Y_{t-i} + (1-\alpha)^t L_0$$

Triple Exponential Smoothing (Holt-Winters)

Additive Seasonality:

$$L_t = \alpha(Y_t - S_{t-m}) + (1-\alpha)(L_{t-1} + T_{t-1})$$
$$T_t = \beta((L_t - L_{t-1}) + (1-\beta)T_{t-1}$$
$$S_t = \gamma(Y_t - L_{t-1} - T_{t-1}) + (1-\gamma)S_{t-m}$$

Multiplicative Seasonality:

$$L_t = \alpha(Y_t / S_{t-m}) + (1-\alpha)(L_{t-1} + T_{t-1})$$
$$T_t = \beta((L_t - L_{t-1}) + (1-\beta)T_{t-1}$$
$$S_t = \gamma[Y_t / (L_{t-1} + T_{t-1})] + (1-\gamma)S_{t-m}$$

Autoregressive Integrated Moving Average (ARIMA)

$$(1-B)^d Y_t = \mu + \frac{\left(1 + \theta_1 B + \cdots + \theta_q B^q\right)\varepsilon_t}{\left(1 - \emptyset_1 B - \cdots - \emptyset_p B^p\right)}$$

Seasonal Autoregressive Integrated Moving Average (SARIMA)

$$(1-B)^d(1-B^s)^D Y_t = \mu + \frac{\left(1 + \theta_1 B + \cdots + \theta_q B^q\right)\left(1 + \theta_{s,1} B + \cdots + \theta_{s,Q} B^Q\right)\varepsilon_t}{\left(1 - \emptyset_1 B - \cdots - \emptyset_p B^p\right)\left(1 - \emptyset_{s,1} B - \cdots - \emptyset_{s,Q} B^Q\right)}$$

SARIMA with Exogenous Regressors (SARIMAX)

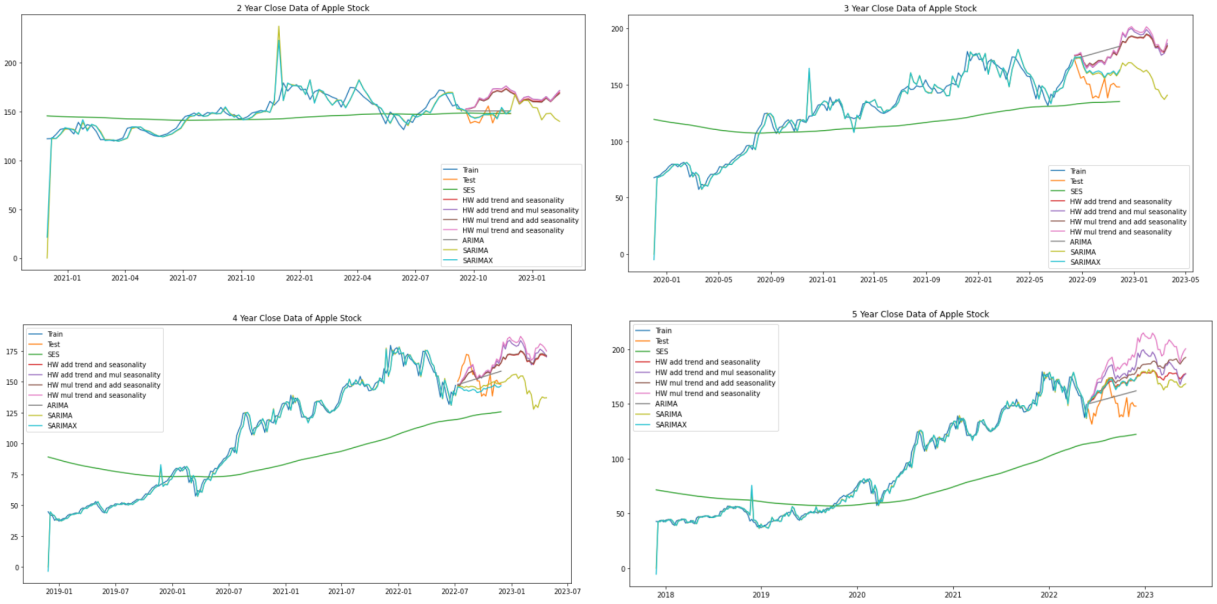
$$(1-B)^d(1-B^s)^D Y_t = \mu + \Psi(B)(1-B)^d(1-B^s)^D X_t + \frac{\theta(B)\theta_s(B^s)\varepsilon_t}{\emptyset(B)\emptyset_s(B^s)}$$

Monitoring

Model Monitoring is the process of performing data validation between expected or forecasted information with actual data measured from newly acquired data.

Due to the short-term context of the assignment, we have very limited monitoring opportunities to assess the performance of our model in future scenarios.

Conclusion



Overall, We think the SARIMA algorithm performed the best. The Holt-Winters also performed ok. The SARIMA model for the 2 year time frames performed the best with the lowest MSE and AIC. The other time frames performed ok but seemed to get worse the more we added time. We think the stock market took a sudden and unexpected downturn recently that these models had a hard time predicting.

If we were to make any predictions with all the data and put more consideration into the better-performing models, we would have to guess that in the long term, Apple will probably keep going up. In the short term, however, we would have to say it's not looking too hot, and the stock will probably decline a bit.

We do not consider any of this concrete financial advice as the stock market is unpredictable.