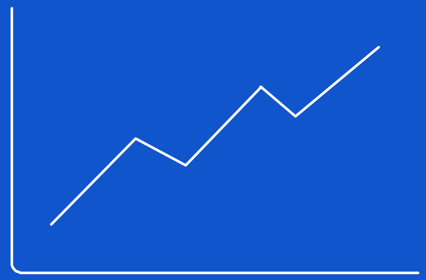
PREDICTING APPLE STOCK

CS 171 Final Project

Group J:

Alan Viollier, Jerry Zhao, Victor La

Business Background & Objective



- As a company rises in value, so does its potential capital gain, which is extremely attractive for investors.
- Stock advisors are often an investment on their own and also might not have consistent accuracy.
- Our contracting company has asked us to use machine learning to produce a model that can analyze a company's historic stock values to accurately predict and forecast its future value.
- We chose the most popular stock APPL.

Data

- Yahoo Finances Python module
- yf.download() downloads: Open, Close, High, Low, Adj Close, & Volume for any stock on yahoo.
- Can adjust time interval, start date, and end date.
- 2, 3, 4, & 5 year time periods.
- We chose these intervals because we wanted to see what our predictions would be based on which time frame used.

```
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")
```

Variable Selection

- Using Cointegration Test

```
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)
```

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
...	...	...	...	...	...	...

4 Year

```
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
----------------------------------------------------
```

5 Year

```
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
```

Data Cleaning

- Only kept Open and Close, got rid of the rest.
- Made sure there were no null variables.

```
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)
```

```python
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

- We split our data into 90% train and 10% test for every time frame

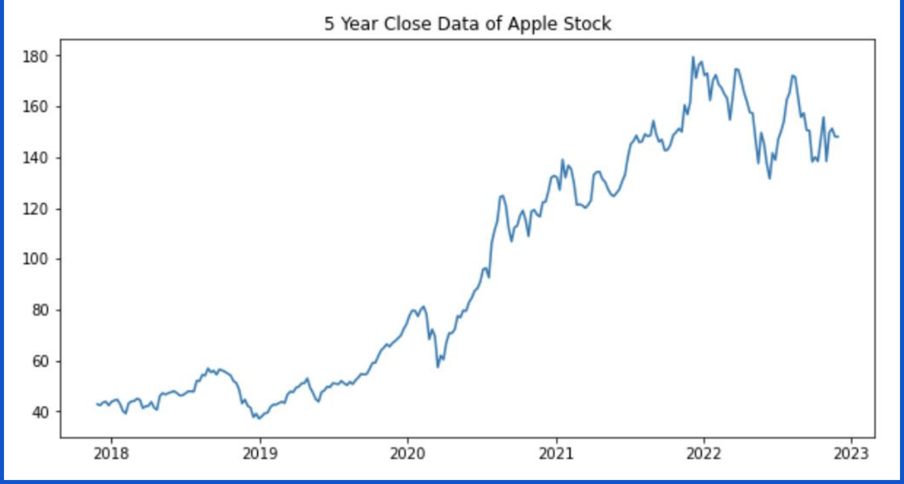
```
(94,)
(11,)
(141,)
(16,)
(189,)
(21,)
(235,)
(27,)
```

Splitting Data into Train & Test

```
# 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)
```

Data Check

- Graphed the data to see what it looks like. Looks good!



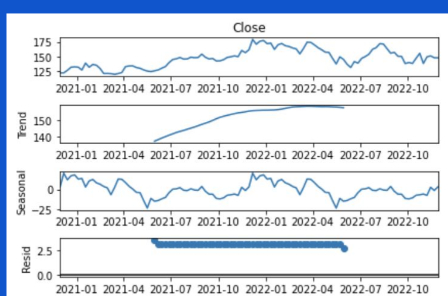
Methodology

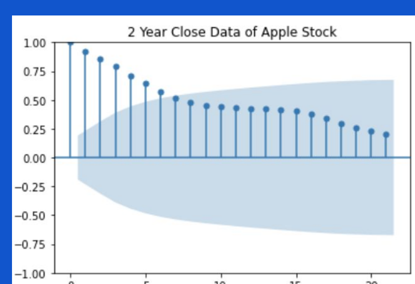
```
Results of Dickey-Fuller Test for column: 2 Year Close Data of Apple Stock
Test Statistic                  -2.262888
p-value                          0.184217
No Lags Used                     0.000000
Number of Observations Used    104.000000
Critical Value (1%)             -3.494850
Critical Value (5%)             -2.889758
Critical Value (10%)            -2.581822
dtype: float64
Conclusion:====>
Fail to reject the null hypothesis
Data is non-stationary
```

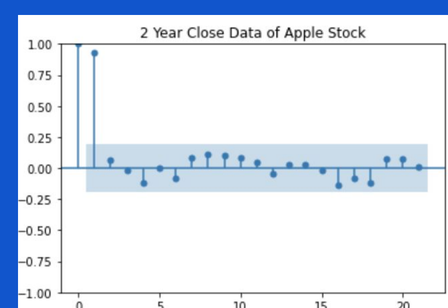
Time Series Forecasting

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

These models can use the data we collected to make predictions





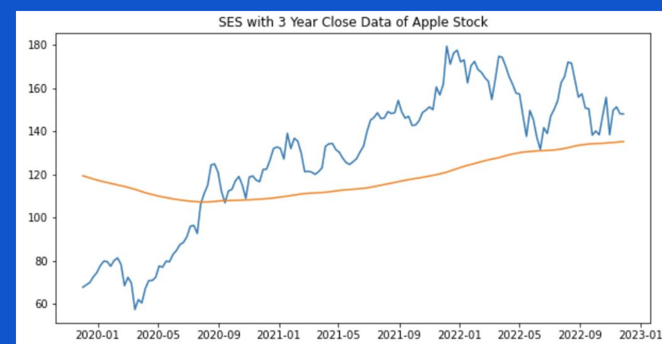


Simple Exponential Smoothing (SES)

- Adaptation of simple moving average.
- Weighted average of past values.
- We did not expect this to be the best or work efficiently, but wanted to use it to compare to other models.

Simple Exponential Smoothing (SES)

- Only signal we can observe with SES is trend. It’s simple.
- We can observe that as we get a larger and larger time frame 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.

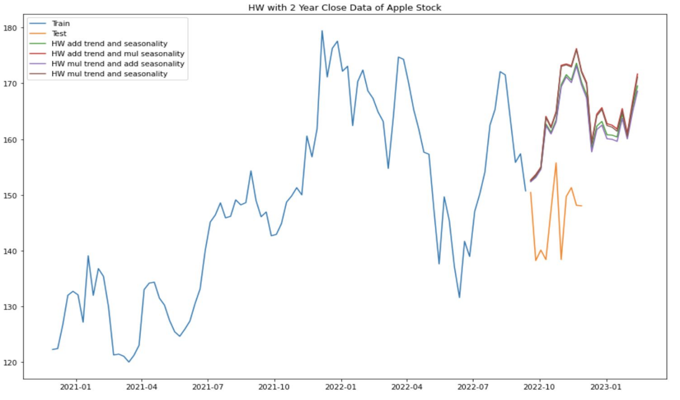


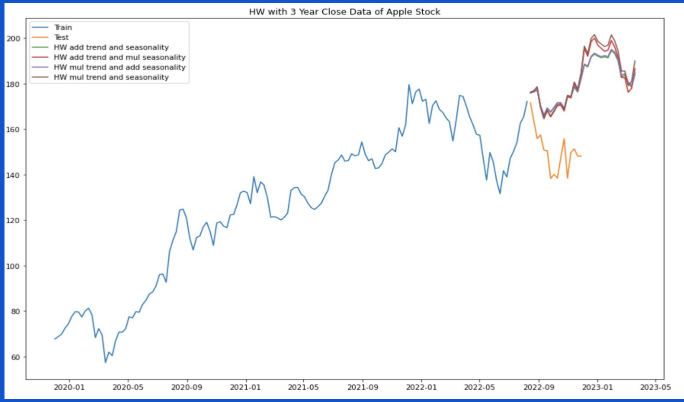
Triple Exponential Smoothing (Holt-Winters)

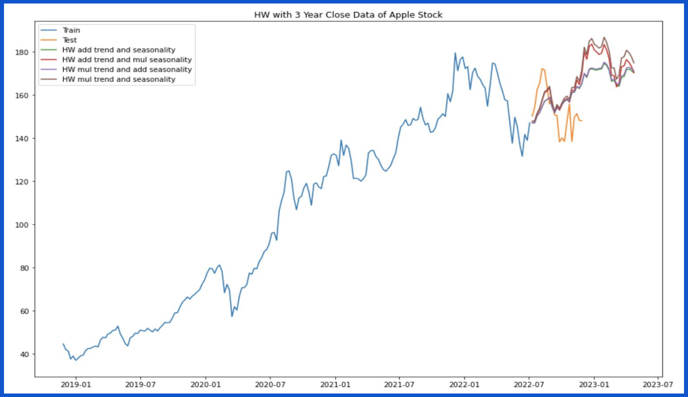
- Holt-Winters' method is based on three smoothing equations one for the level, one for trend, and one for seasonality.
- We expected this to work better than SES but will probably ultimately be worse than SARIMA.

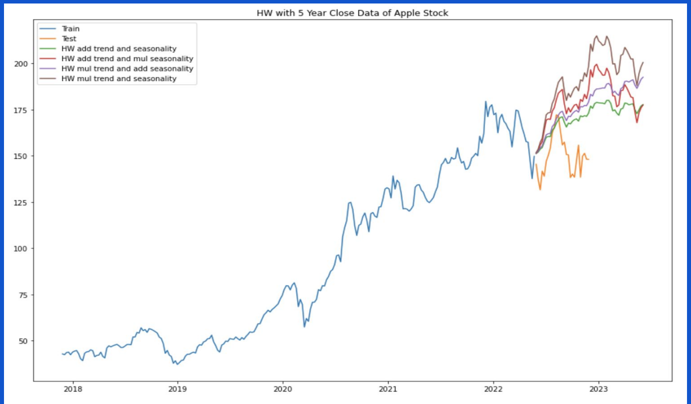
Triple Exponential Smoothing (Holt-Winters)

- 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 constant increase.
- The 3-4 year timeframe looks like the prediction closest to test. Still wouldn’t make predictions as SARIMA is probably better. This data says will have an upcoming bump followed by a drop but still an overall upward trend.







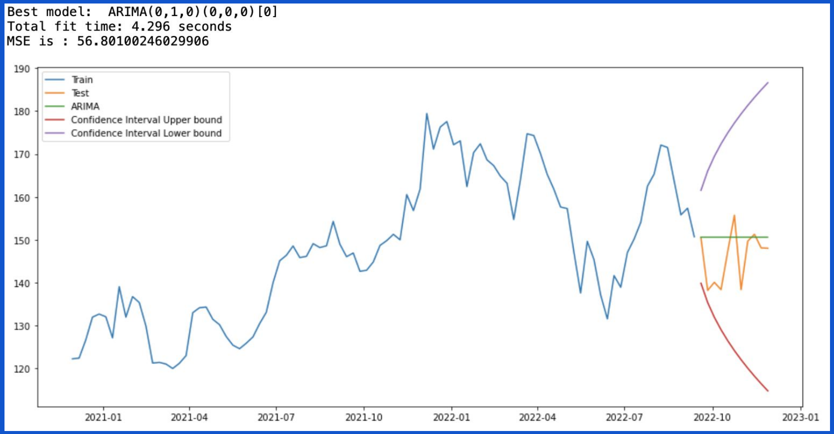


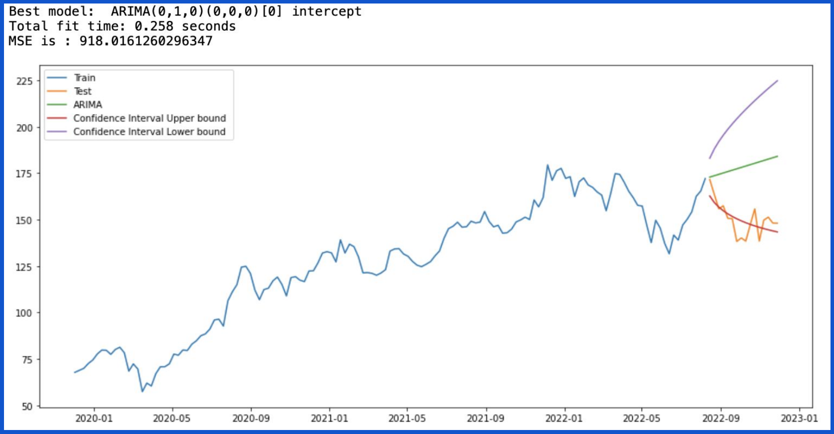
Autoregressive Integrated Moving Average (ARIMA)

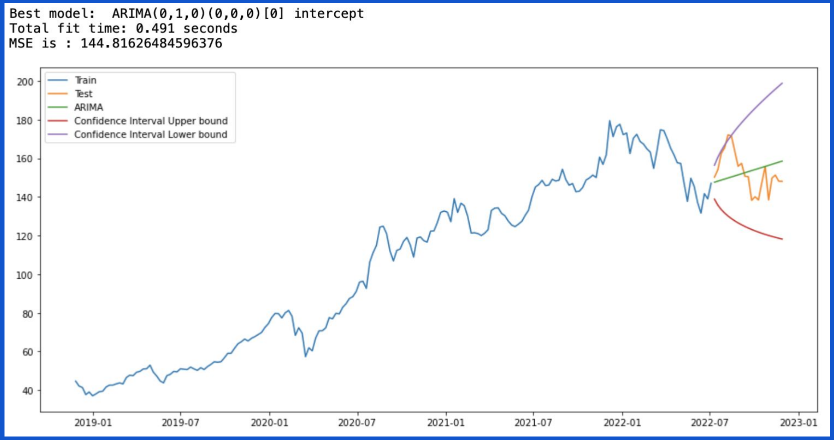
- The ARIMA family of models is a set of smaller models that can be combined.
- Autoregression (AR) regression model that explains a variable’s future value using its past values.
- Moving average (MA) uses past values to predict the current value of the variable. Uses the prediction error in previous time steps to predict the future.
- Autoregressive moving average (ARMA) combines the two previous building blocks into one model.
- Autoregressive integrated moving average (ARIMA) adds automatic differencing to the ARMA model. Can also set to the number of times that the time series needs to be differenced.

Autoregressive Integrated Moving Average (ARIMA)

- ARIMA model is ok but lacks precision due to 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 long run 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.

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

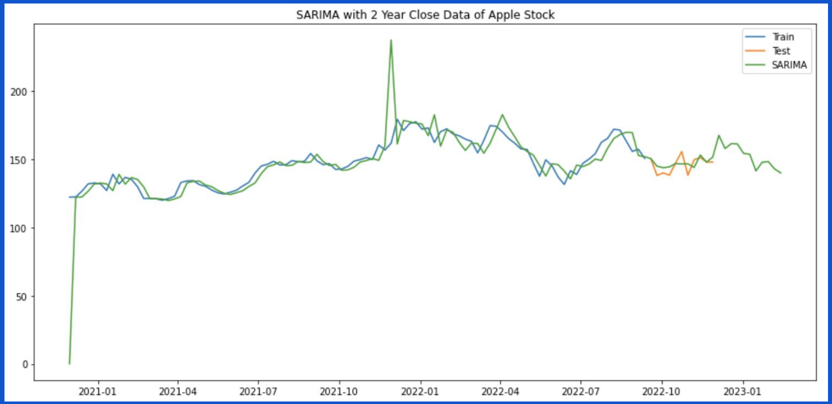
For 3 years of Apple stock data seasonality = 52
ARIMA(0,1,1)(0,1,1)[52] AIC=594.970 seems to be the best because it has the lowest AIC
MSE = 614.2349896911138

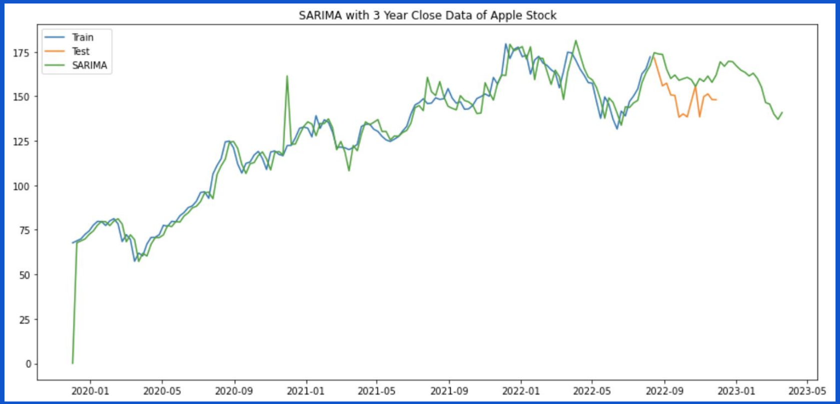
For 4 years of Apple stock data seasonality = 52
ARIMA(0,1,0)(2,1,0)[52] AIC=871.052 seems to be the best because it has the lowest AIC
MSE = 170.36248521972735

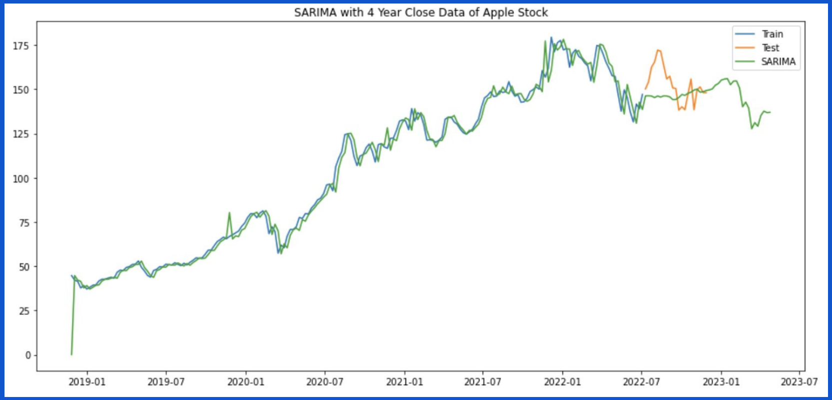
For 5 years of Apple stock data seasonality = 52
ARIMA(0,1,0)(0,1,2)[52] AIC=1103.524 seems to be the best because it has the lowest AIC
MSE = 475.2424274429505

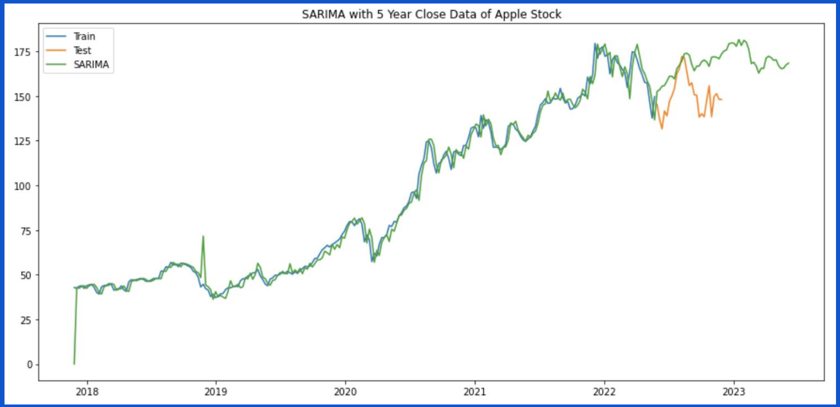
Seasonal ARIMA (SARIMA)

- 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.









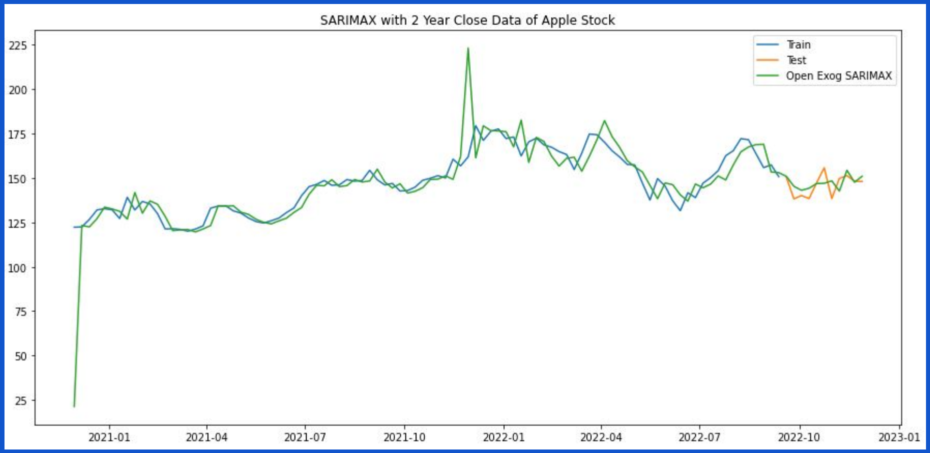
This took a very long time for my computer to process! :)

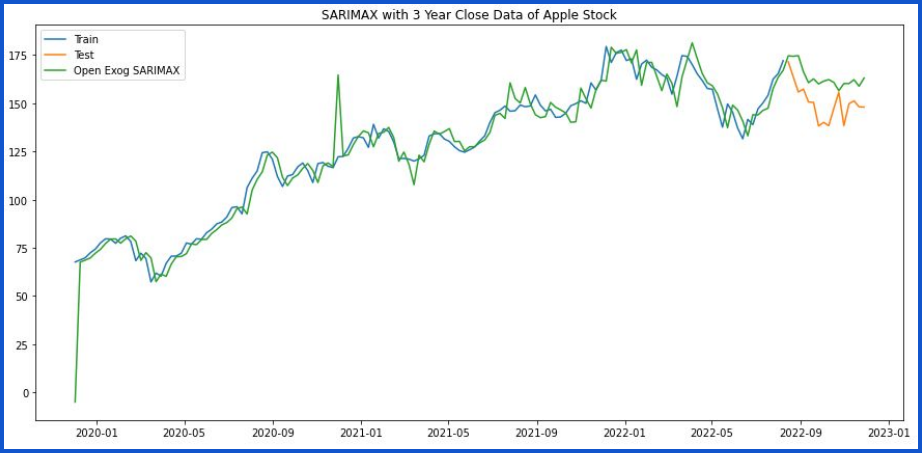
SARIMA with Exogenous Regressors (SARIMAX)

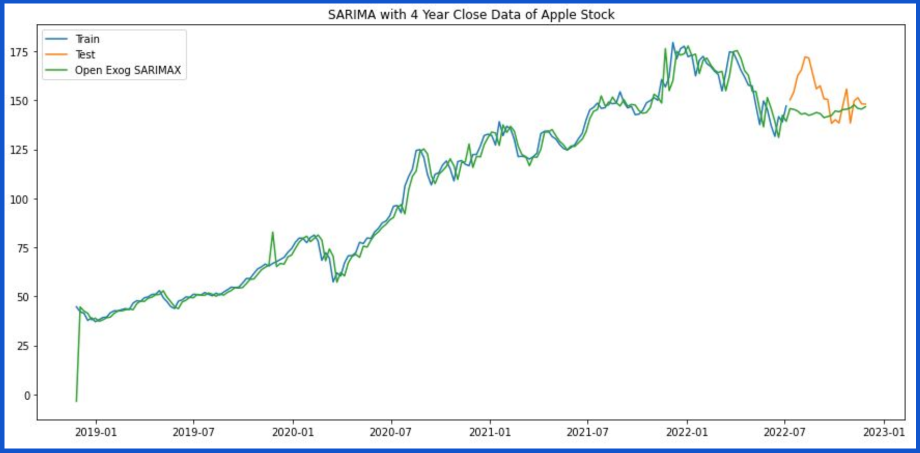
- The most complex variant is the SARIMAX model.
- It regroups AR, MA, differencing, and seasonal effects.
- On top of that, it adds the X: external variables.
- If you have any variables that could help your model to improve, you could add them with SARIMAX.

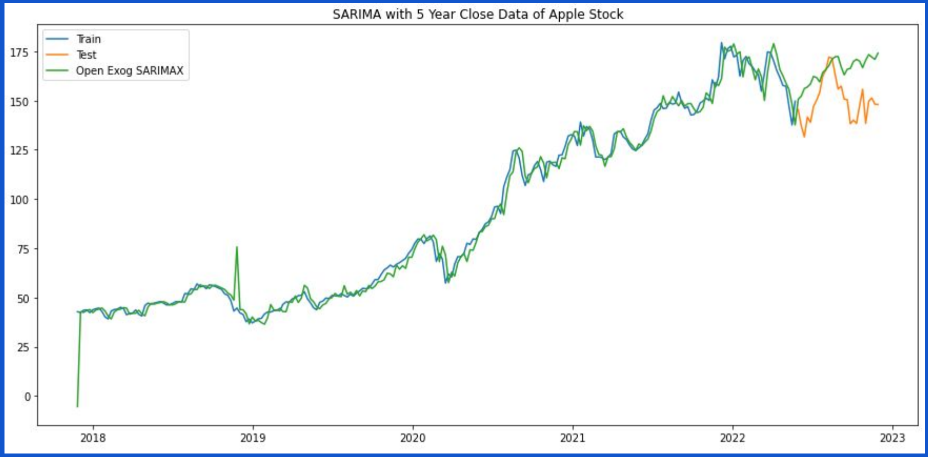
- **We used Open for our exogenous variable but it did not change anything from SARIMA! The Exogenous variable did not help us in this case.**

SARIMA with Exogenous Regressors (SARIMAX)









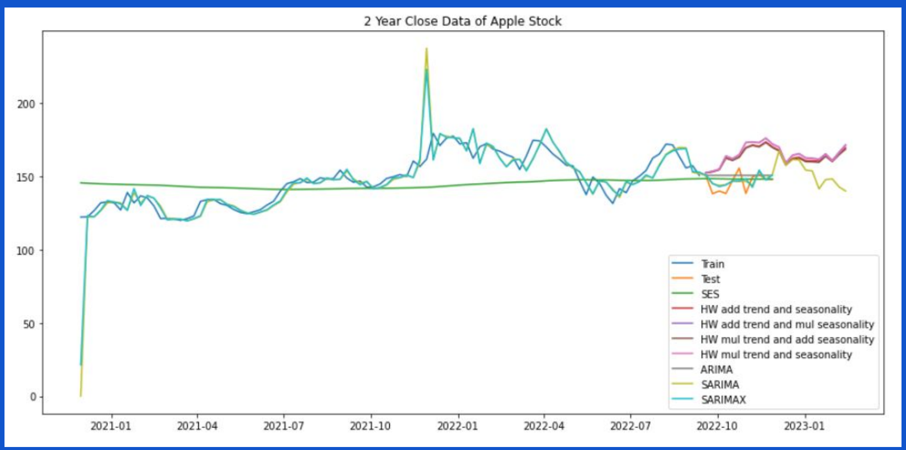
Ensemble Technique

- 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

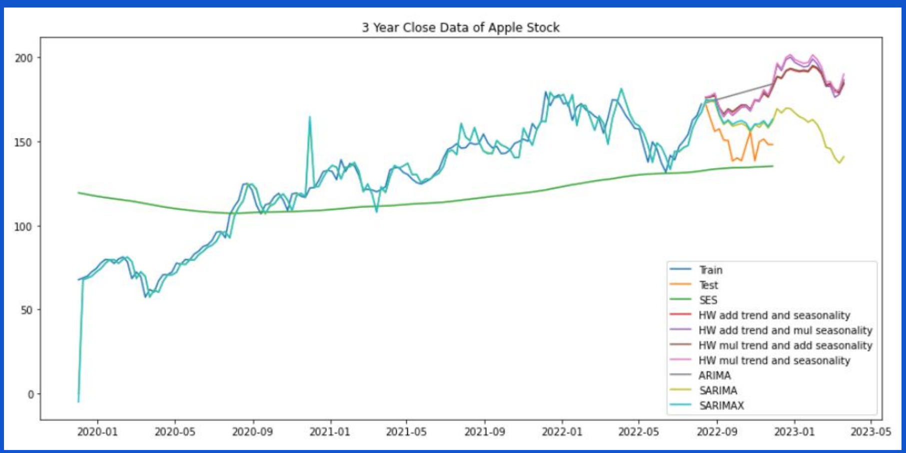
Monitoring

- 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.
- We will check back on out predictions in a couple months!

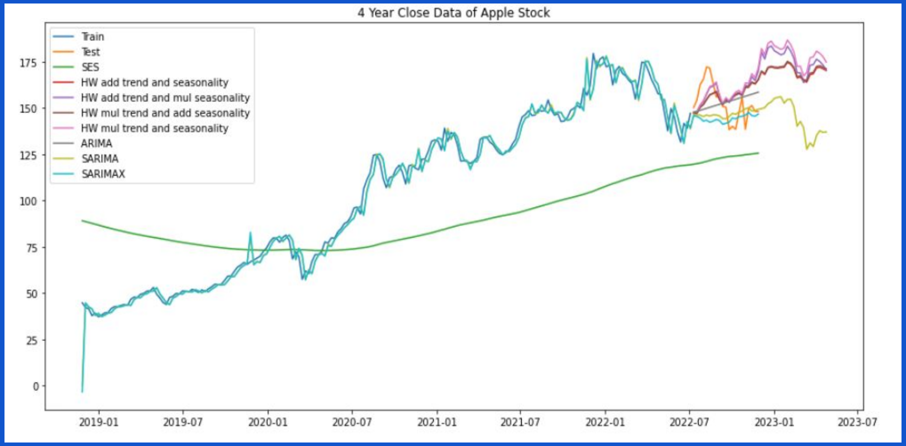
Conclusion



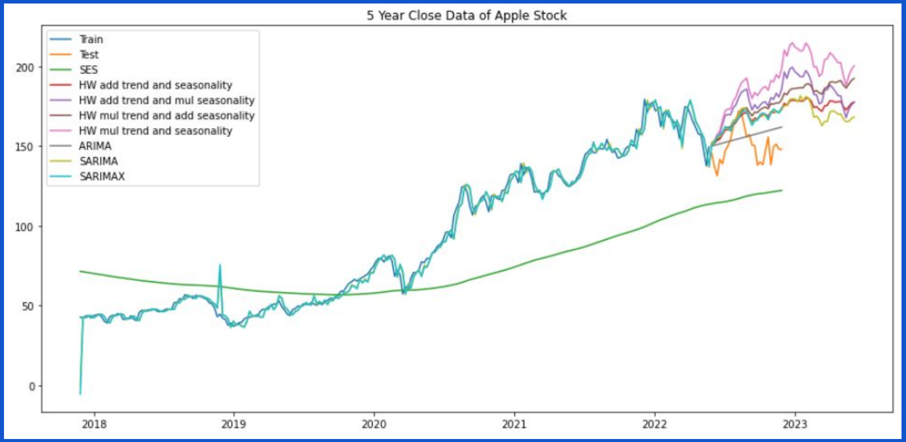
2 Year Close Data of Apple Stock
200
150
100
50
0
2021-01
2021-04
2021-07
2021-10
2022-01
2022-04
2022-07
2022-10
2023-01
Train
Test
SES
HW add trend and seasonality
HW add trend and mul seasonality
HW mul trend and add seasonality
HW mul trend and seasonality
ARIMA
SARIMA
SARIMAX



3 Year Close Data of Apple Stock
200
150
100
50
0
2020-01
2020-05
2020-09
2021-01
2021-05
2021-09
2022-01
2022-05
2022-09
2023-01
2023-05
Train
Test
SES
HW add trend and seasonality
HW add trend and mul seasonality
HW mul trend and add seasonality
HW mul trend and seasonality
ARIMA
SARIMA
SARIMAX



4 Year Close Data of Apple Stock
Train
Test
SES
HW add trend and seasonality
HW add trend and mul seasonality
HW mul trend and add seasonality
HW mul trend and seasonality
ARIMA
SARIMA
SARIMAX
175
150
125
100
75
50
25
0
2019-01
2019-07
2020-01
2020-07
2021-01
2021-07
2022-01
2022-07
2023-01
2023-07



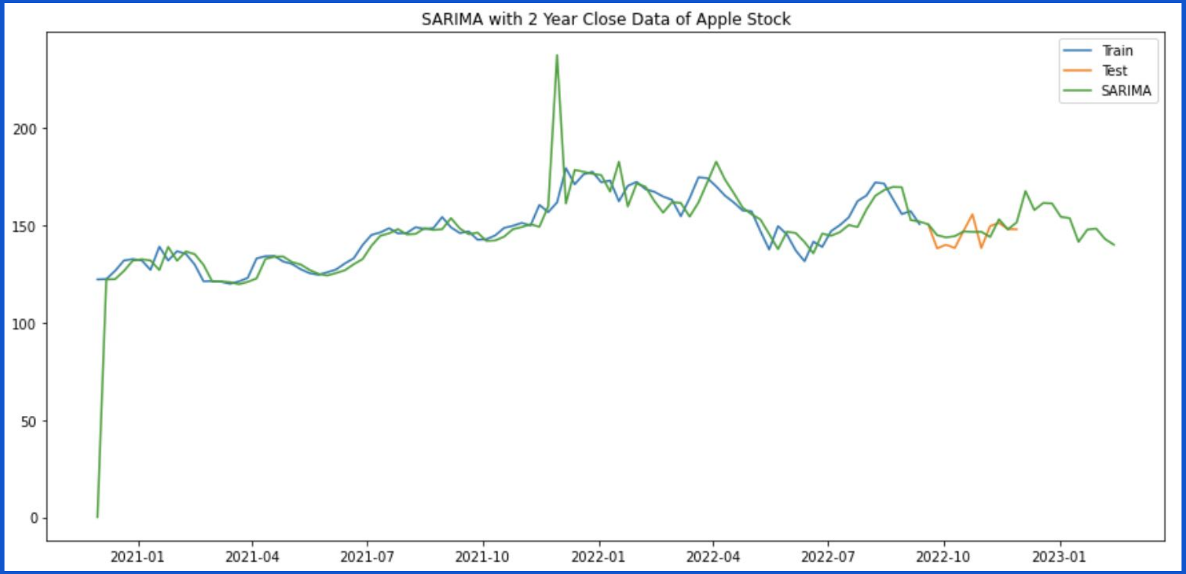
5 Year Close Data of Apple Stock
Train
Test
SES
HW add trend and seasonality
HW add trend and mul seasonality
HW mul trend and add seasonality
HW mul trend and seasonality
ARIMA
SARIMA
SARIMAX
200
150
100
50
0
2018
2019
2020
2021
2022
2023

Conclusion

- Overall, We think the SARIMA algorithm performed the best.
- The Holt Winters also performed ok.
- The SARIMA model for the 2 time frame 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 putting 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.

THANKS FOR LISTENING!

Any questions?



Group J:

Alan Viollier, Jerry Zhao, Victor La