Capstone Final Report

--Weekly Stock Price Prediction

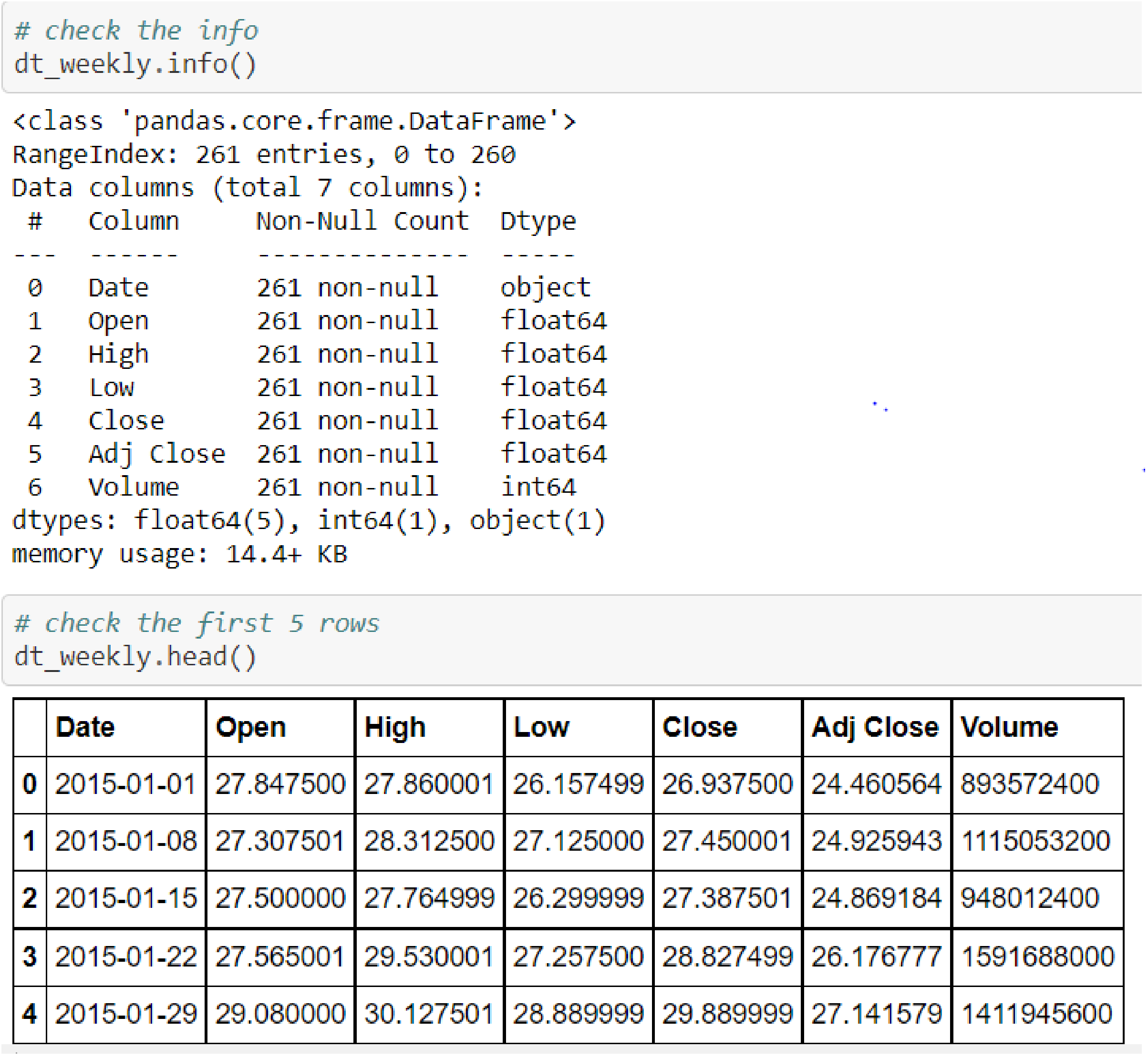
Problem Statement

For decades, stock market prediction has been one of the most challenging tasks that financial analysts eager to achieve. Although until now no one could accurately predict the daily target stock prices, weekly, bi-weekly stock prices prediction, or even only trending predictions with acceptable accuracy are also valuable for both analysts and investors.

For this capstone project, some timeseries data analysis and predictions are performed based on the stock trading record of Apple. Inc. from 2015 to 2019.

Data Wrangling

The raw data set of Apple weekly stock price record from 01/01/2015 to 12/26/2019 pulled from Yahoo Finance contained 261 rows and 7 columns. For instance, each row represents one week's Apple stock record, and 7 columns represents the date, open price, highest price, lowest price, closing price, adjusted closing price, and trading volume during the week. There is no missing value included.



```
# check the info
dt_weekly.info()
```

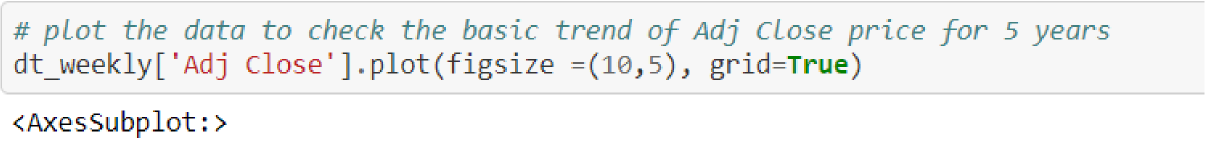
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 261 entries, 0 to 260
Data columns (total 7 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   Date       261 non-null    object 
 1   Open       261 non-null    float64
 2   High       261 non-null    float64
 3   Low        261 non-null    float64
 4   Close      261 non-null    float64
 5   Adj Close  261 non-null    float64
 6   Volume     261 non-null    int64  
dtypes: float64(5), int64(1), object(1)
memory usage: 14.4+ KB
```

```
# check the first 5 rows
dt_weekly.head()
```

	Date	Open	High	Low	Close	Adj Close	Volume
0	2015-01-01	27.847500	27.860001	26.157499	26.937500	24.460564	893572400
1	2015-01-08	27.307501	28.312500	27.125000	27.450001	24.925943	1115053200
2	2015-01-15	27.500000	27.764999	26.299999	27.387501	24.869184	948012400
3	2015-01-22	27.565001	29.530001	27.257500	28.827499	26.176777	1591688000
4	2015-01-29	29.080000	30.127501	28.889999	29.889999	27.141579	1411945600

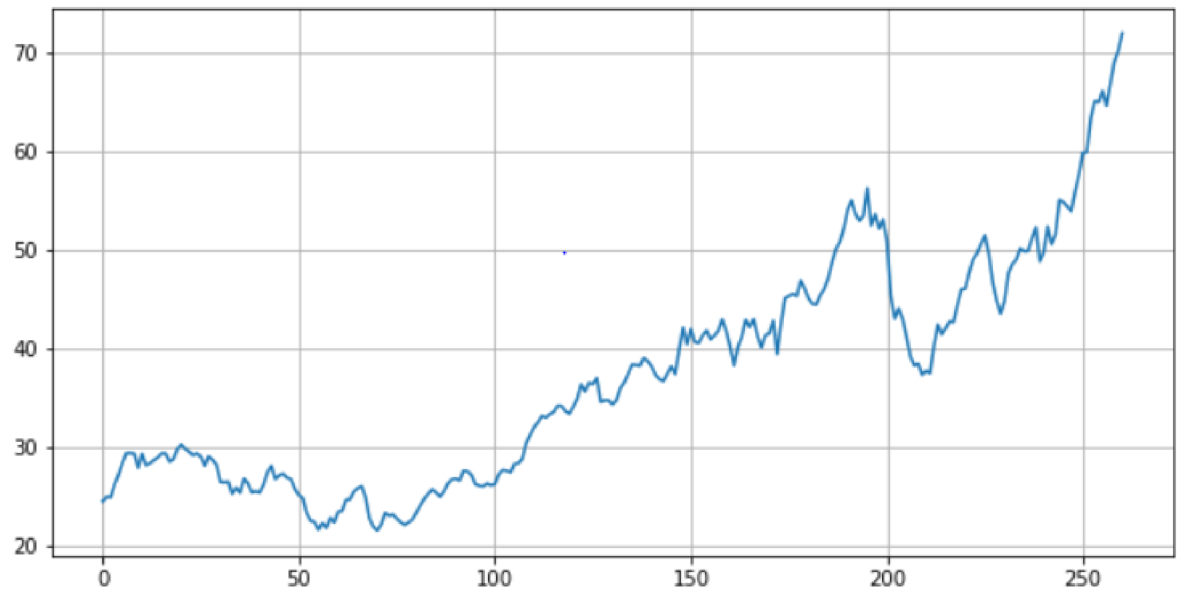
Exploratory Data Analysis and initial findings

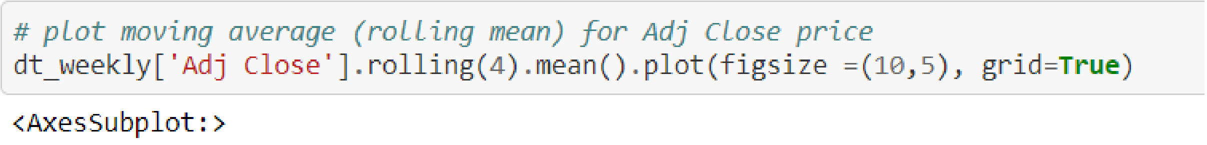
a) Plots of run chats for weekly and four-week (monthly) smoothing adjusted closing price to get a basic understanding of how it looks



```
# plot the data to check the basic trend of Adj Close price for 5 years
dt_weekly['Adj Close'].plot(figsize =(10,5), grid=True)
```

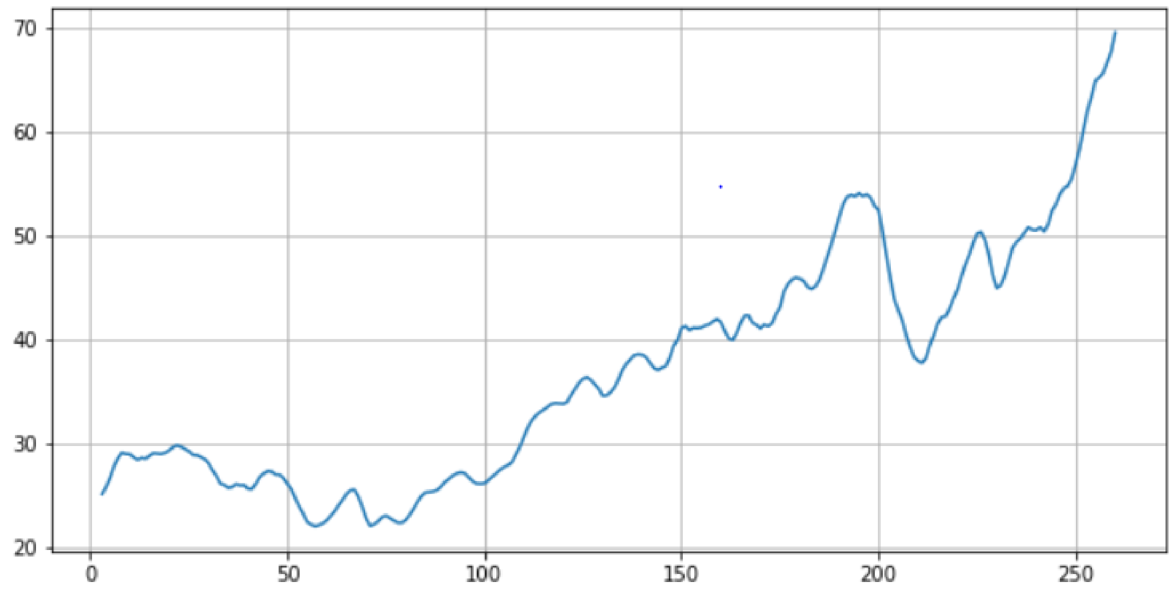
```
<AxesSubplot:>
```





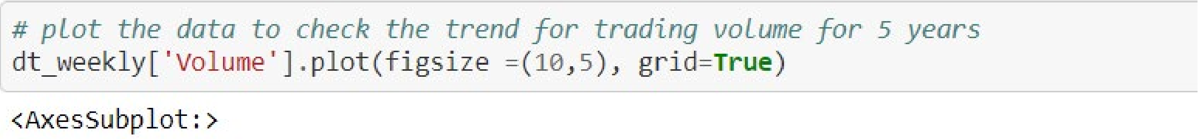
```
# plot moving average (rolling mean) for Adj Close price
dt_weekly['Adj Close'].rolling(4).mean().plot(figsize =(10,5), grid=True)
```

```
<AxesSubplot:>
```



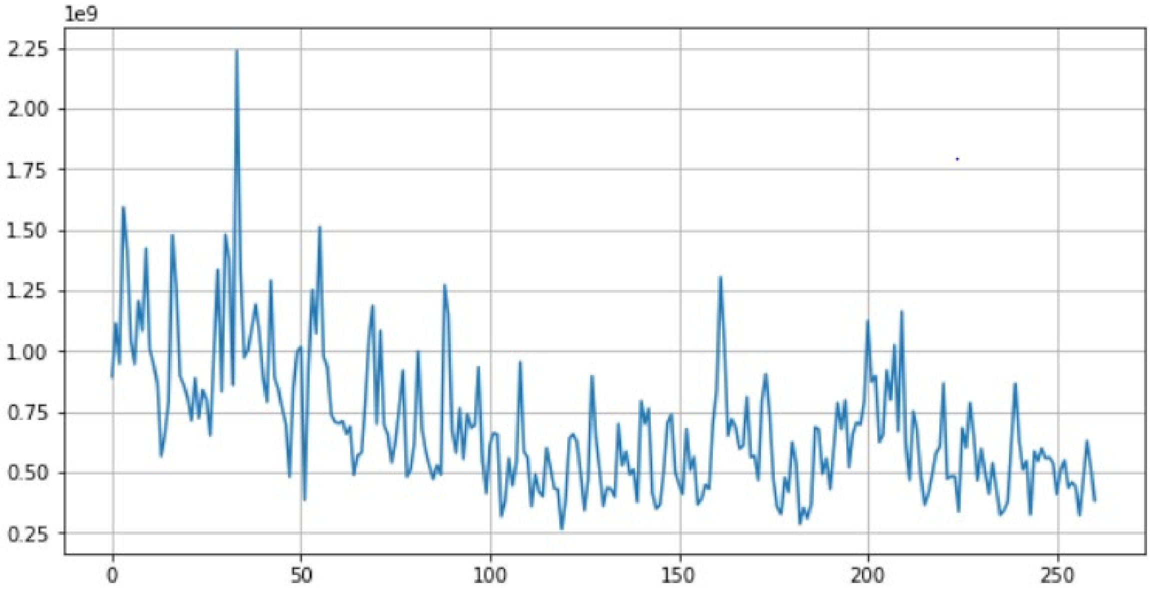
Findings: weekly price shows a clearly overall up trend during the 5-year period, but no noticeable seasonal patterns.

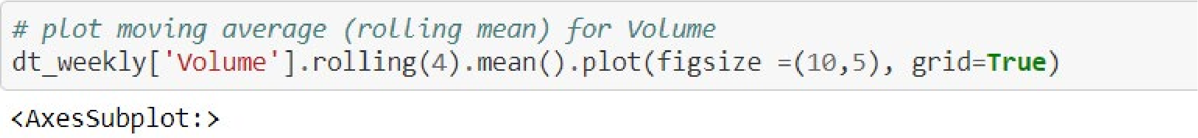
b) Plots of run chats for weekly and four-week (monthly) smoothing trading volume



```
# plot the data to check the trend for trading volume for 5 years
dt_weekly['Volume'].plot(figsize =(10,5), grid=True)
```

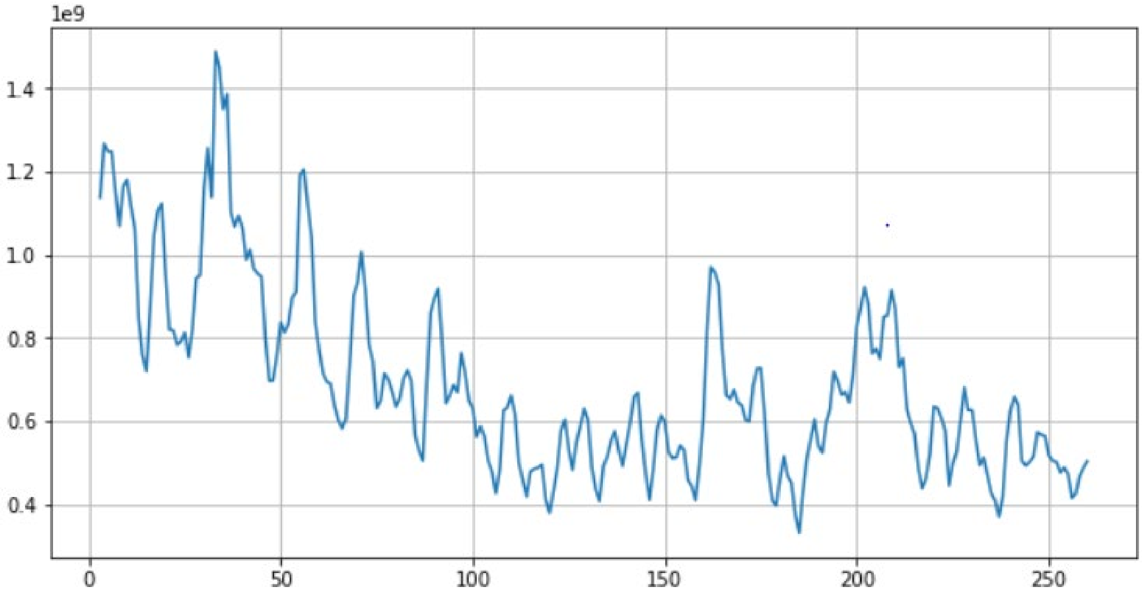
```
<AxesSubplot:>
```





```
# plot moving average (rolling mean) for Volume
dt_weekly['Volume'].rolling(4).mean().plot(figsize =(10,5), grid=True)
```

```
<AxesSubplot:>
```



Findings: weekly volume shows an overall slightly down trend during the 5-year period, with up and downs but no noticeable seasonal patterns.

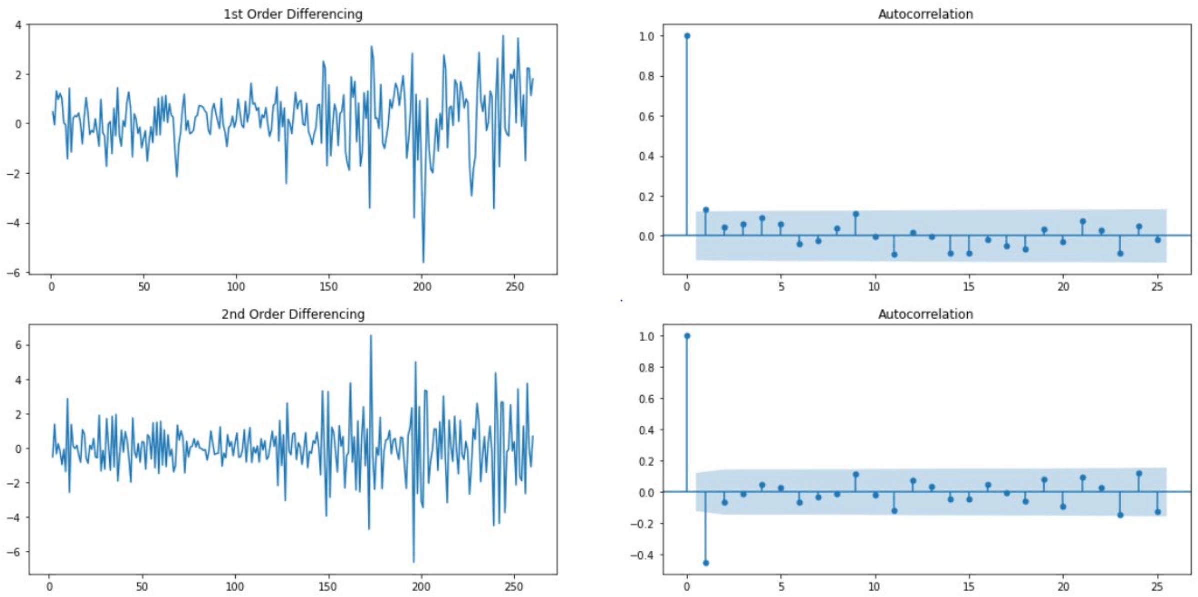
c) Kernel Density plots for weekly price and volume, pair plot for all features, and Heatmap plot are drawn but no significant findings are shown.

Preprocessing

First, I dropped all other columns and kept only the adjusted closing price representing the weekly stock price.

For a timeseries analysis, confirming that the data set is stationary not a random walk is of essential. For instance, I performed augmented Dickey-Fuller test on the weekly price. The P value is 0.9958835435458218, much bigger than the critical value 0.05, meaning that we can't reject that the weekly price is random walk.

As a possible solution, I differenced the series and see how the autocorrelation plot looks like.



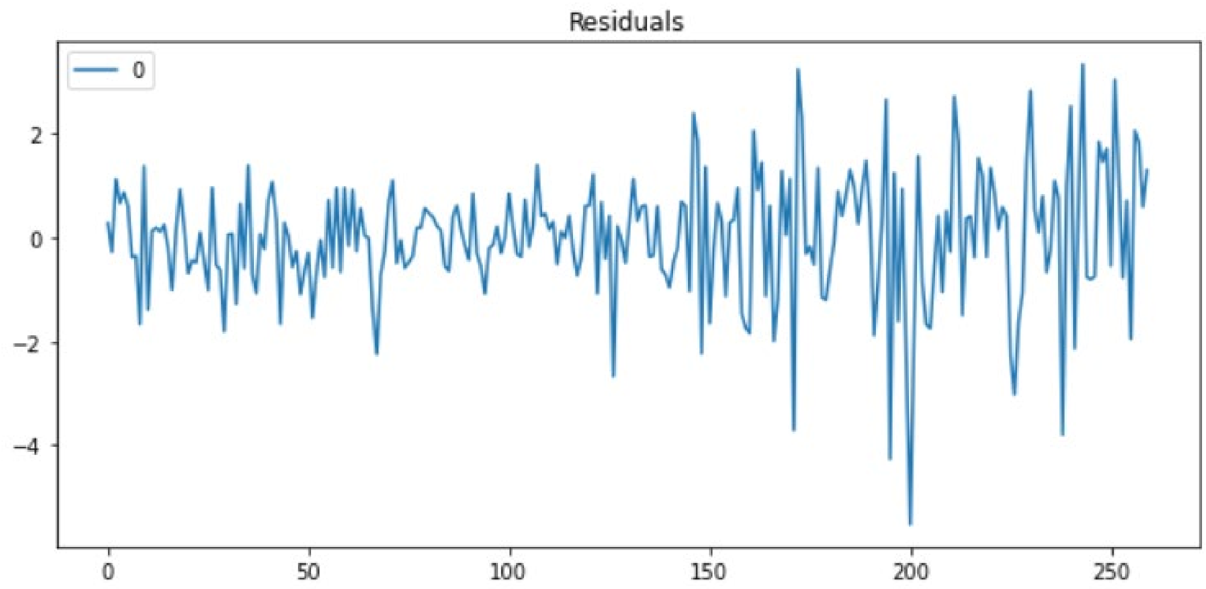
The lag of the autocorrelation plot for the 2nd differencing goes into the far negative zone fairly quick, which indicates that the series might have been over differenced. So, I chose the order of differencing as 1 even though the series might not perfectly stationary (weak stationarity).

P value of the augmented Dickey-Fuller test for the 1st differencing is 4.315100826772732e-26, much smaller than the critical value 0.05, meaning that the first difference of adjusted close price is not random walk and reaches stationarity.

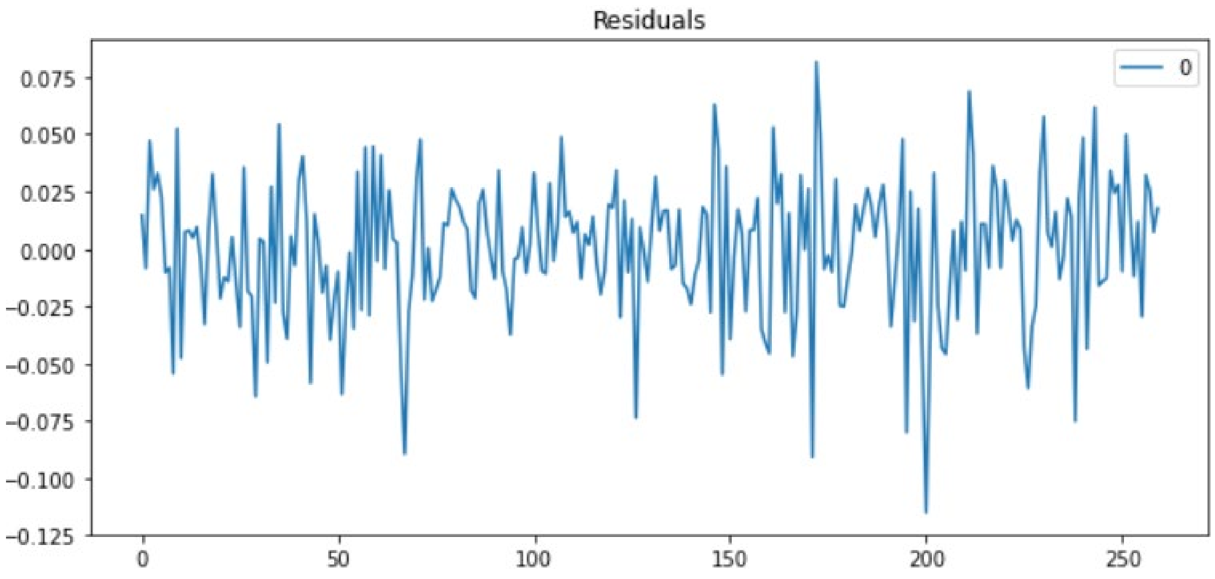
Modeling

I selected 3 models, Auto Regressive Integrated Moving Average (ARIMA), TBATS, and Holt-Winters Exponential Smoothing (HWES), for this timeseries capstone project.

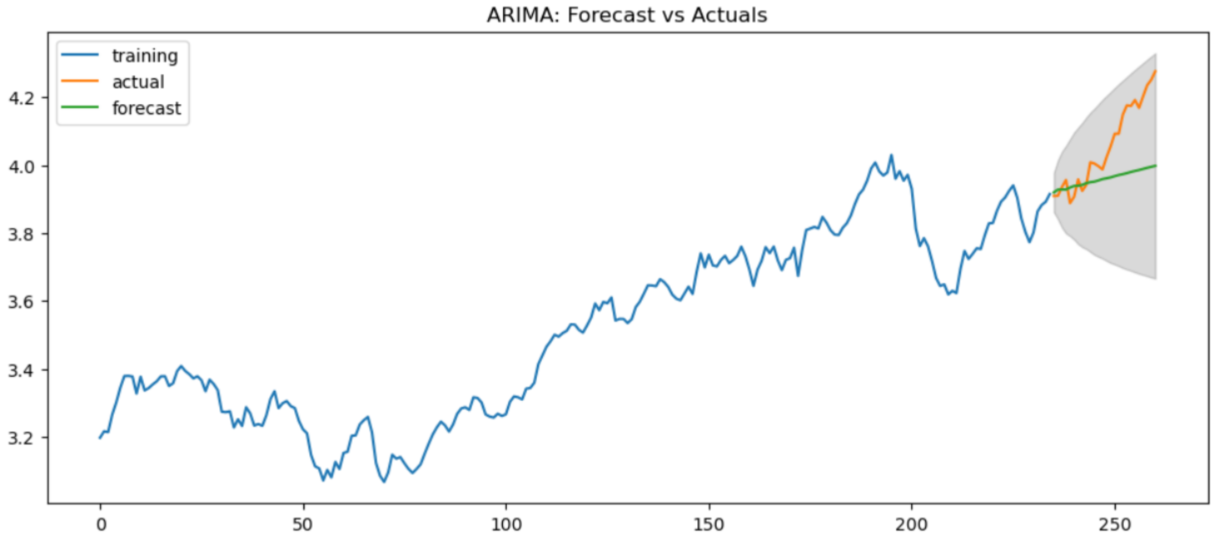
Before making predictions when working on ARIMA model, I found that the variance of residual error is spreading over time as shown below, which is commonly caused by heteroskedastic bias.

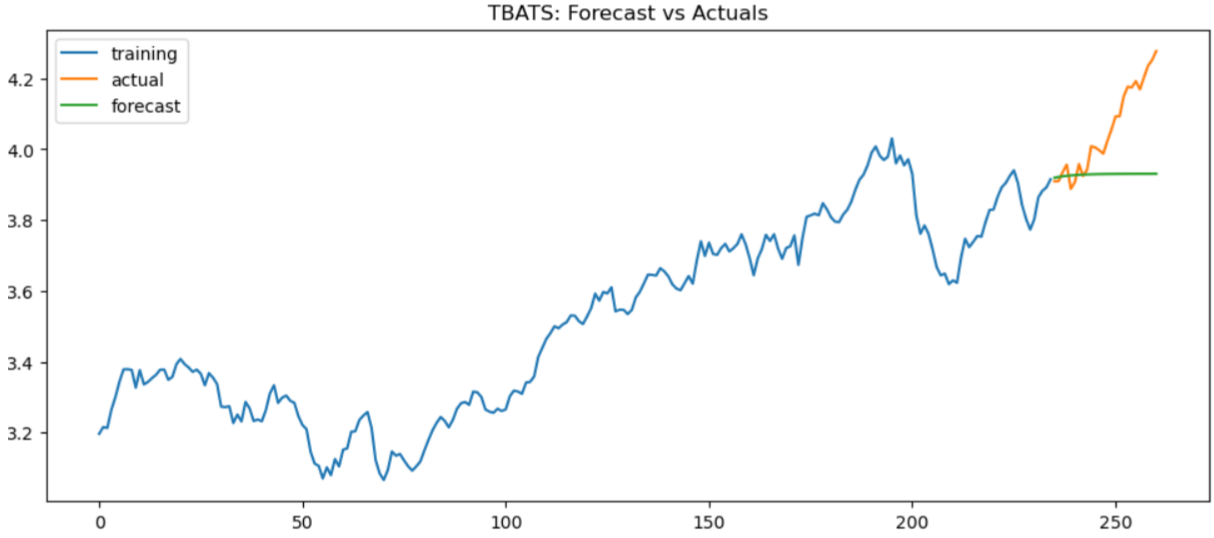


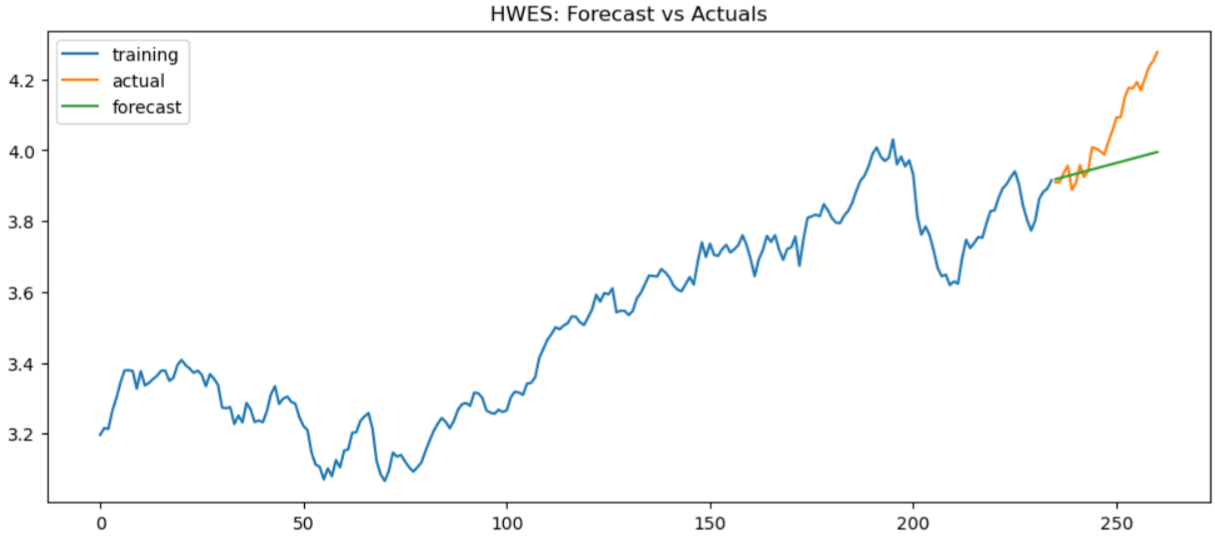
After confirmed with het_white test for heteroskedasticity of the residual errors, I decided to log the original data to mitigate or eliminate the heteroskedastic bias. For instance, the residual error plot and het_white test of the logged data shows a satisfying result as below.



I split the data into train and test set in 9:1 with total 260 observations to optimize the models using the out-of-time cross validation. Prediction plots are shown as below:



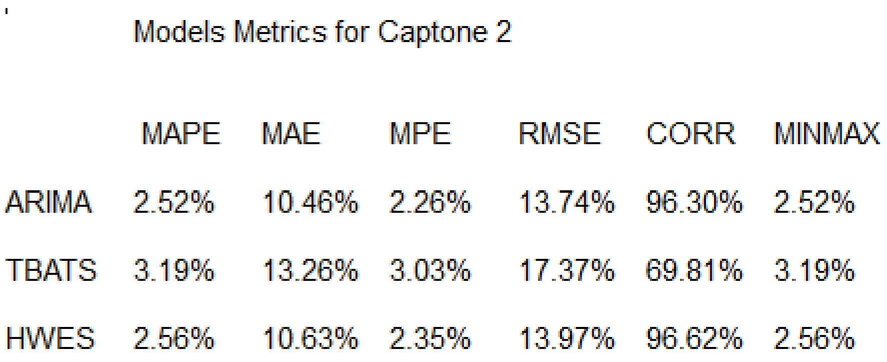




Metrics used for evaluation are:

1. Mean Absolute Percentage Error (MAPE)
2. Mean Absolute Error (MAE)
3. Mean Percentage Error (MPE)
4. Root Mean Squared Error (RMSE)
5. Correlation between the Actual and the Forecast (CORR)
6. Min-Max Error (minmax)

I checked all of them, and focus on MAPE, MPE, and CORR for final decision of optimal model. Results of metrics are shown as below:



Models Metrics for Captone 2

	MAPE	MAE	MPE	RMSE	CORR	MINMAX
ARIMA	2.52%	10.46%	2.26%	13.74%	96.30%	2.52%
TBATS	3.19%	13.26%	3.03%	17.37%	69.81%	3.19%
HWES	2.56%	10.63%	2.35%	13.97%	96.62%	2.56%

Takeaways

For ARIMA model, around 2.52% MAPE and 2.26% MPE implies the model is highly accurate and low biased in predicting the next 26 observations. Also, score of 96.30% for CORR implies the predicted values are highly correlated to the actual values which is a good sign.

For TBATS model, around 3.19% MAPE and 3.03% MPE implies the model is also highly accurate and low biased in predicting the next 26 observations. Nevertheless, score of 69.81% for CORR implies the predicted values are less correlated to the actual values (compare to the 96.30% from ARIMA model) which is not good enough.

For HWES model, around 2.56% MAPE and 2.35% MPE implies the model is also highly accurate and low biased in predicting the next 26 observations. Also, score of 96.62% for CORR implies the predicted values are highly correlated to the actual values which is pretty good.

Based on the results of the metrics used, ARIMA and HWES performed better than TBATS does. For instance, we will choose ARIMA for overall better performance over HWES.

Future Research

This capstone project gave me a lot to think about as far as ways to improve stock price predictions. For instance, I would like to expand these models with more features about the stock and stock markets. Also, trying more models and tuning more on the model parameters would be always useful for optimizing the results.