Stock Price Forecasting

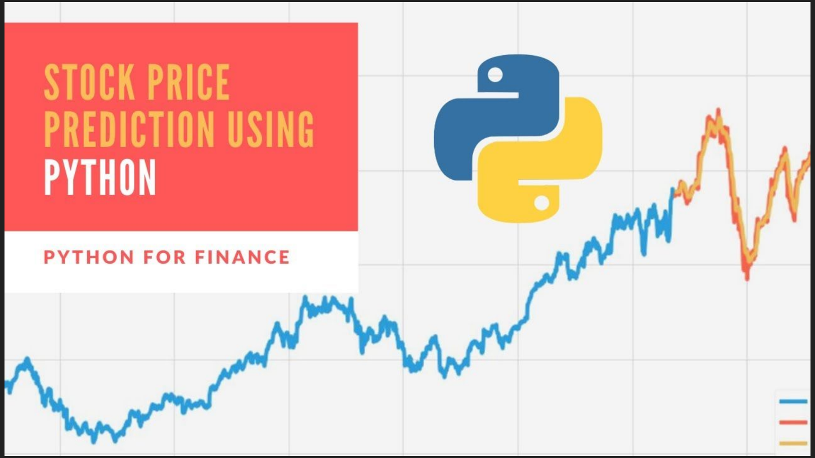
Michael Dyer

The Problem

Source: The Motley Fool

- The stock market plays a vital role in our economy.
- For years, industries, companies, and individuals have looked to capitalize on the stock market.

The Solution



Source: YouTube - Computer Science

- Use Machine Learning to create a model that can accurately forecast future prices or trends.

Why does this matter?

- Currently there are no legitimate ways to forecast prices or trends, only speculations
- Being able to accurately predict future prices and trends based upon the previous entries would be beneficial to anyone looking to profit from the market

The Data

AAPL_2006-01-01_to_2018-01-01.csv (153.62 KiB)

Detail Compact Column

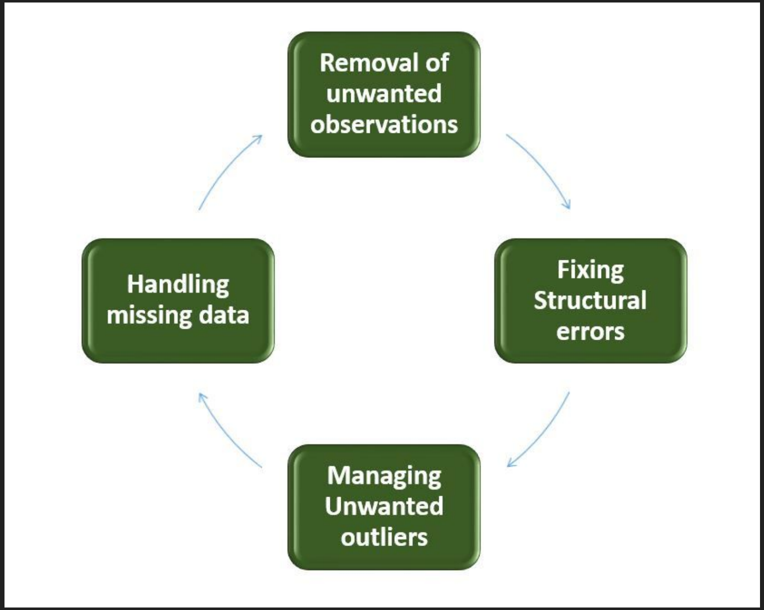
Date	Open	High	Low	Close	Volume	Name
2006-01-03	10.34	10.68	10.32	10.68	201853036	AAPL
2006-01-04	10.73	10.85	10.64	10.71	155225609	AAPL
2006-01-05	10.69	10.7	10.54	10.63	112396081	AAPL
2006-01-06	10.75	10.96	10.65	10.9	176139334	AAPL
2006-01-09	10.96	11.03	10.82	10.86	168861224	AAPL
2006-01-10	10.89	11.7	10.83	11.55	570088246	AAPL
2006-01-11	11.98	12.11	11.8	11.99	373548882	AAPL
2006-01-12	12.14	12.34	11.95	12.04	320201966	AAPL
2006-01-13	12.14	12.29	12.09	12.23	194153393	AAPL
2006-01-17	12.24	12.34	11.98	12.1	209215265	AAPL

Taken from https://www.kaggle.com/datasets/szrlee/stock-time-series-20050101-to-20171231?select=AAPL_2006-01-01_to_2018-01-01.csv

Data Wrangling

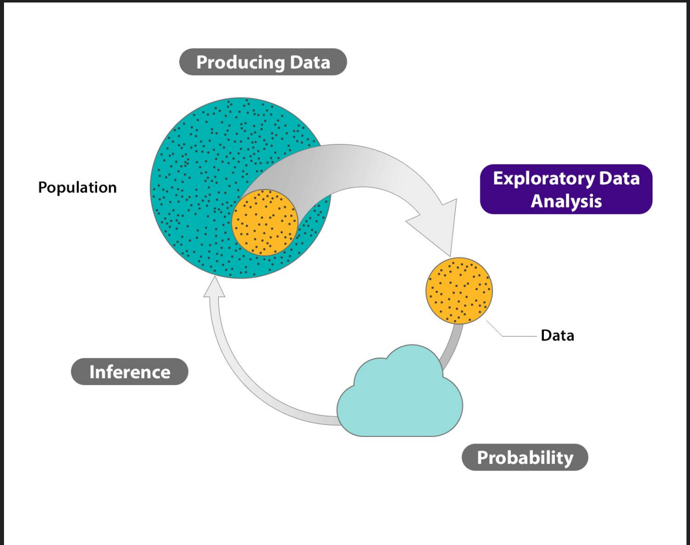
- Chose 5 stocks csv files
  - aapl, amzn, googl, nke, vz
- Each file contained:
  - Date, Open, Close, High, Low, Volume, Name
- Each file has a little over 3000 rows of information
- Converted to DataFrame
- All work done on Jupyter Notebook

Data Cleaning

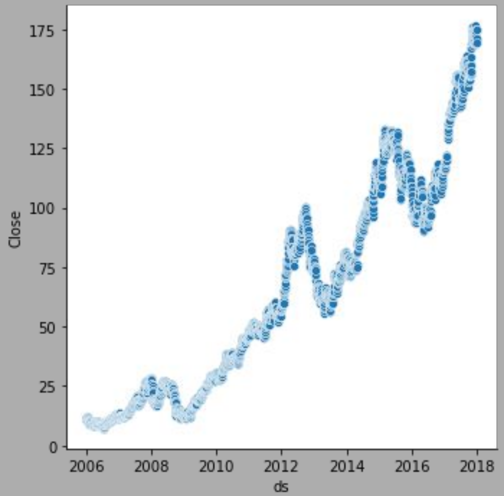


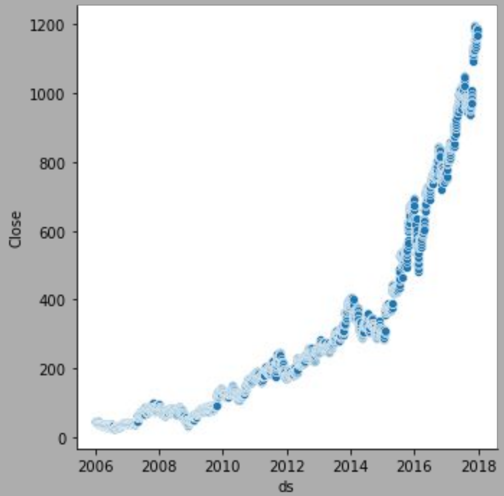
- There was only 1 missing value in all 5 csv files
  - Row was just dropped as there was ample data
- No “Incorrect” data values
- Created DateTime object for the date and deleted included date column

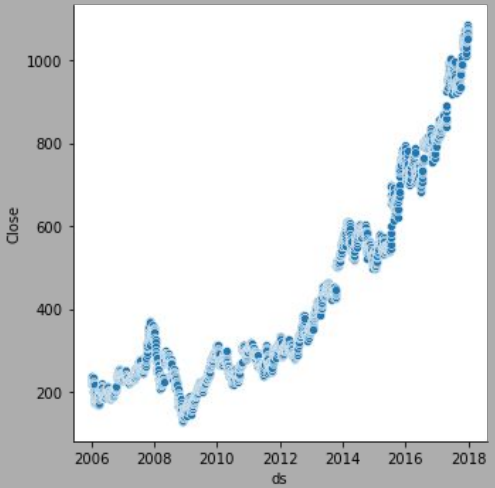
Exploratory Data Analysis

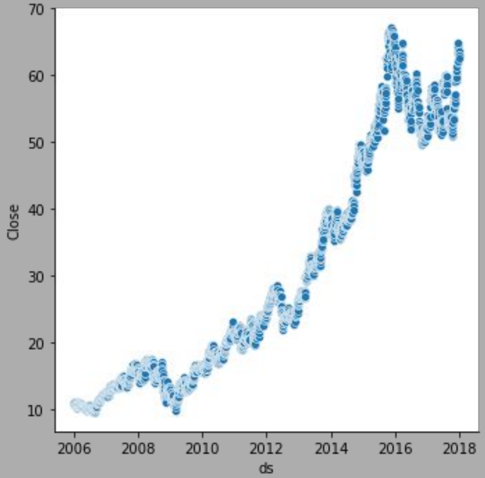


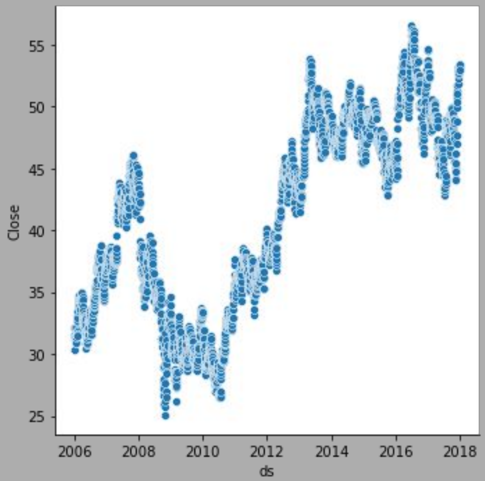
- Still checking integrity of Data
- Looked for correlations amongst features
- Need to go down to 2 features, with Date being one of them
- After checking many plots and graphs, Close value was chosen for the 2nd feature

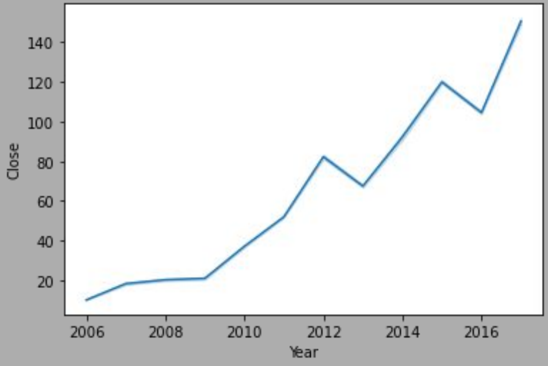

Close
ds
0
25
50
75
100
125
150
175
2006
2008
2010
2012
2014
2016
2018

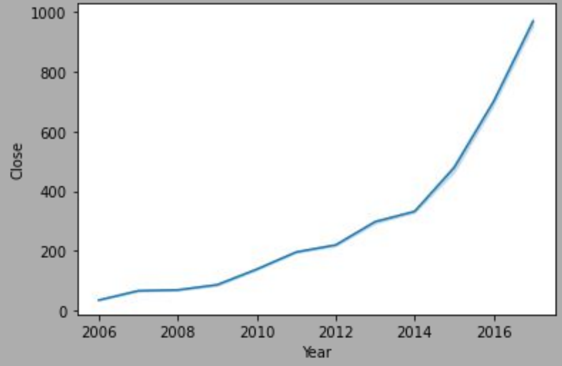

Close
ds
0
200
400
600
800
1000
1200
2006
2008
2010
2012
2014
2016
2018

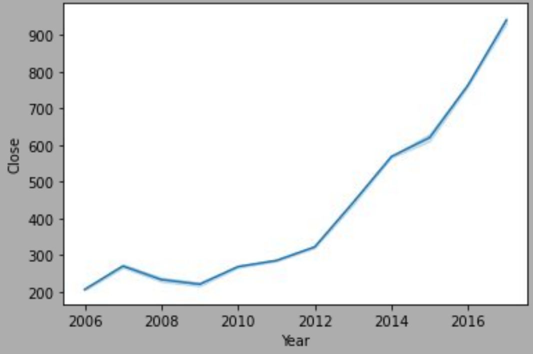

Close
ds
200
400
600
800
1000
2006
2008
2010
2012
2014
2016
2018

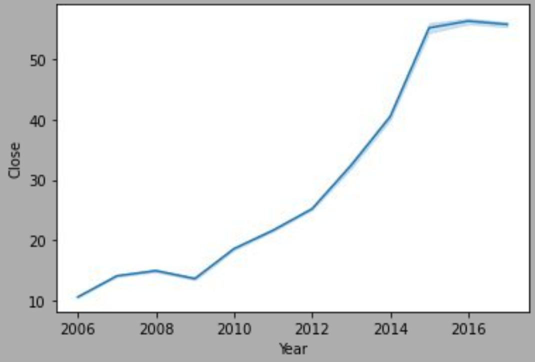

Close
ds
10
20
30
40
50
60
70
2006
2008
2010
2012
2014
2016
2018

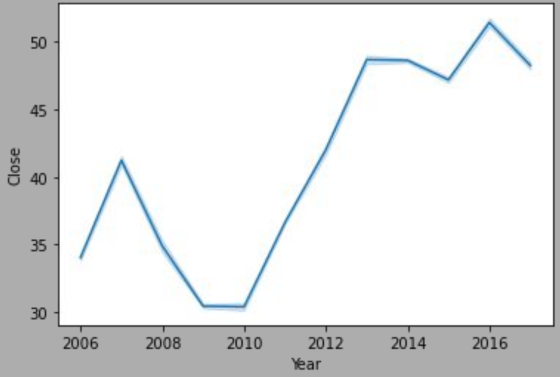

Close
ds
25
30
35
40
45
50
55
2006
2008
2010
2012
2014
2016
2018


Close
Year
20
40
60
80
100
120
140
2006
2008
2010
2012
2014
2016

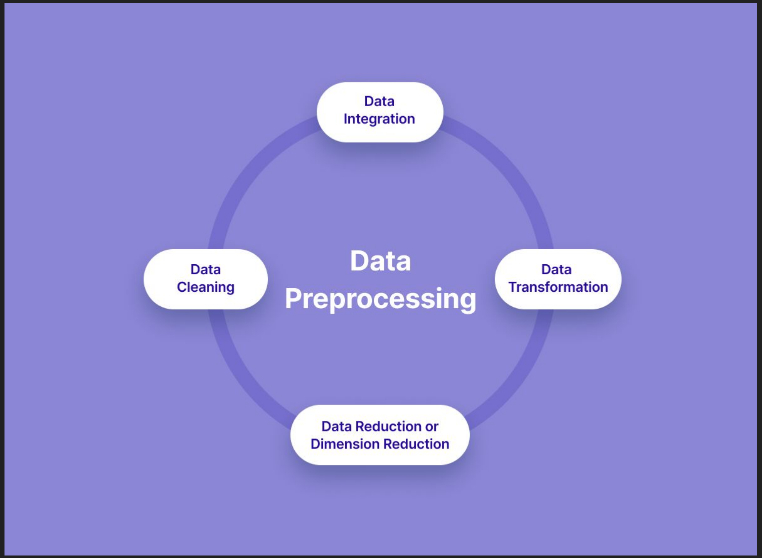

Close
Year
0
200
400
600
800
1000
2006
2008
2010
2012
2014
2016


Close
Year
200
300
400
500
600
700
800
900
2006
2008
2010
2012
2014
2016


Close
Year
10
20
30
40
50
2006
2008
2010
2012
2014
2016

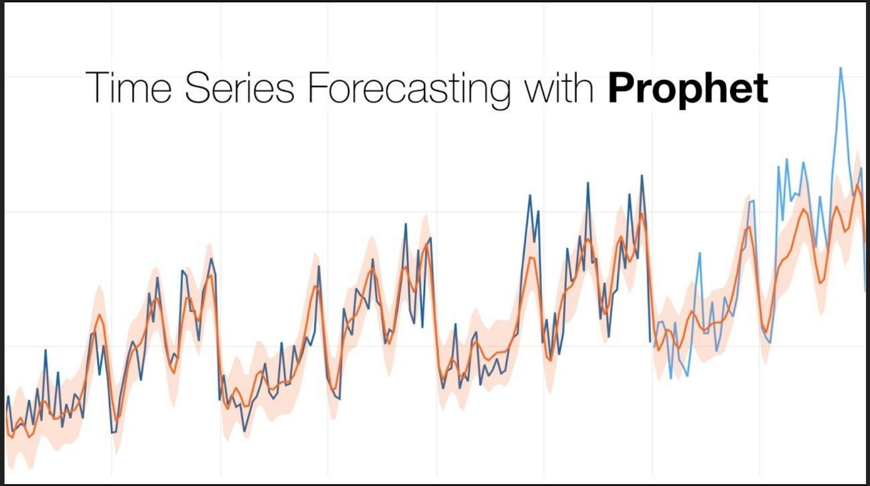

Close
Year
30
35
40
45
50
2006
2008
2010
2012
2014
2016

Pre-Processing



- There were no categorical values, so I did not need to create dummy variables
- There was no need to standardize any metrics
- For the FBProphet Model I needed two columns:
  - 1 marked ‘ds’, this was the DateTime column
  - 1 marked ‘y’, this was the price point for each stock

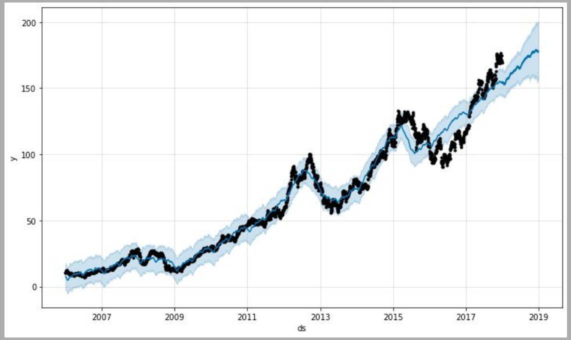
Modeling



Source: blog.exploratory.io

- I used 3 modeling methods
  - FBProphet
  - ARIMA
  - Exponential Smoothing
- I was most familiar with ARIMA and least familiar with Exp. Smoothing.
- Since I am using 5 stocks, and creating 3 models each, I will create a total of 15 models.
- I will compare the RMSE values of each model per stock, and the overall AVG RMSE of each Model

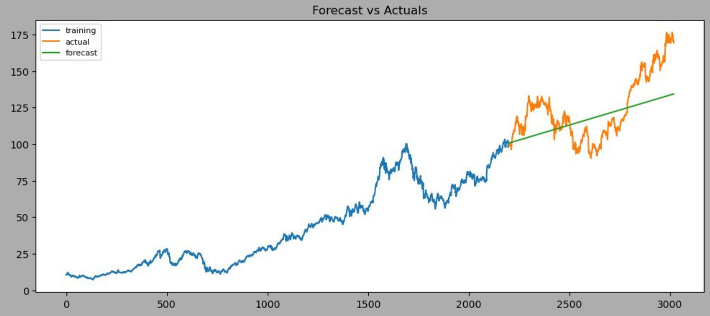
FBProphet



RMSE Values
aapl - 24.87553

AVG of all models = 25.00905

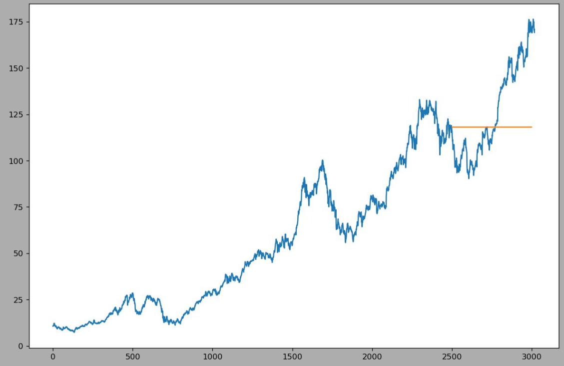
ARIMA



RMSE Values
aapl - 18.17807

AVG of all models = 49.90812

Exponential Smoothing



RMSE Values
aapl - 25.43805

AVG of all models = 73.75101

Arima did best for aapl stock, but FBProphet did best overall

Takeaways

RMSE SCORES BY MODEL

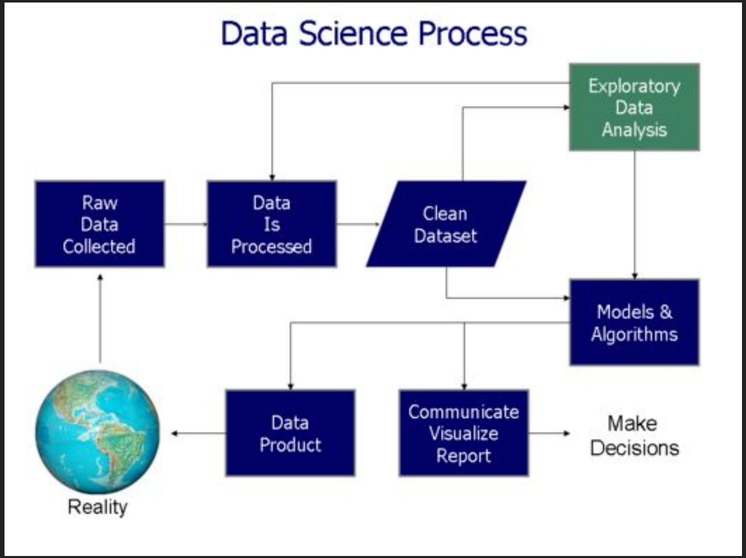
FBProphet - AVG ACROSS MODELS = 25.00905
aapl - 24.87553
amzn - 47.25800
googl - 41.67631
nke - 7.04094
vz - 4.19446

ARIMA - AVG ACROSS MODELS = 49.90812
aapl - 18.17807
amzn - 146.55111
googl - 66.06066
nke - 15.23596
vz - 3.51478

EXPONENTIAL SMOOTHING - AVG ACROSS MODELS = 73.75101
aapl - 25.43805
amzn - 208.51732
googl - 119.27863
nke - 10.85077
vz - 4.67031

- Each model had pro's and con's. The ARIMA Model was the most difficult to set up, while the FBProphet was probably the easiest
- FBProphet secured the best average RMSE score
- Overall, the RMSE scores **do not show** that any of the models predict in a way that can be considered accurate

Future Research



- I would love to be able to spend more time on the Models individually seeing if there are any other parameters that can be tuned.
- I would focus more on trend analysis and forecasting than price
- Collect more data