Mitchell Gil

# Final Report

## Problem Statement

Trading the stock market requires some foresight and analysis into what sort of patterns emerge from historical data. There are unlimited way to try and create models that could try to predict what the next day's moves and direction might be. I am studying the feasibility of exploiting timezone differences and measuring if one exchange in an earlier timezone overall can predict the movements of an exchange in a latter one.

Since there arent many stock that run in multiple exchanges, the best way I have found to keep a consistent estimator is to use each exchange's flagship index as an overall summary indicator of the general direction the market trended towards in a particular trading day.

The four indices used will be the NYSE Dow Jones, Euronext100, Tokyo Nikkei 225, and Hong Kong's Hang Seng.

## Data Wrangling

The data used was derived from Yahoo Finance. I imported 20 years of daily data, including the open, high, low, close and adjusted close. Because the data came from Yahoo finance, there wasnt much cleaning required. Missing values accounted for around 10% of all rows, therefore they were dropped entirely.

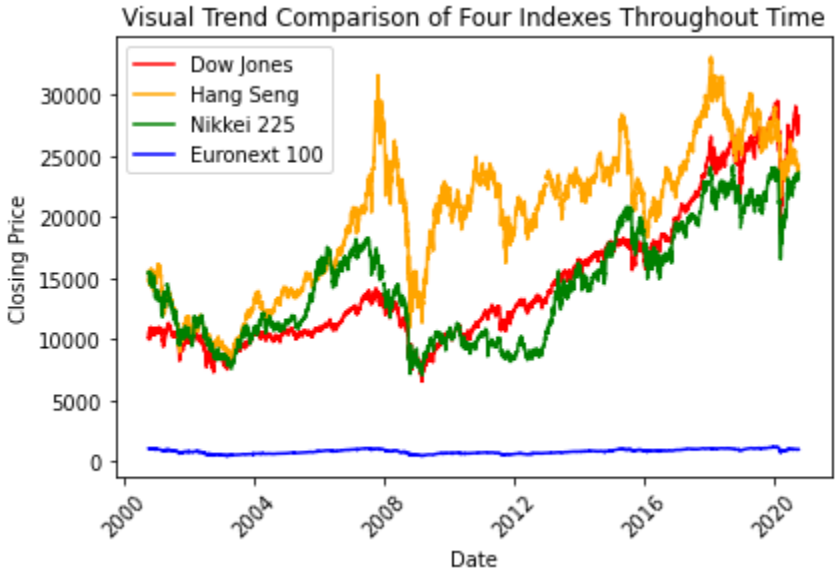
Furthermore, to maintain consistency, I only kept the data for the dates where all four exchanges had a trading day. Not all exchanges have trading days due to differing holidays and such. It was calculated that 90.49% of the data was for dates in which all four exchanges traded. Therefore, the others were discarded. This was saved into a separate csv.

## Exploratory Data Analysis

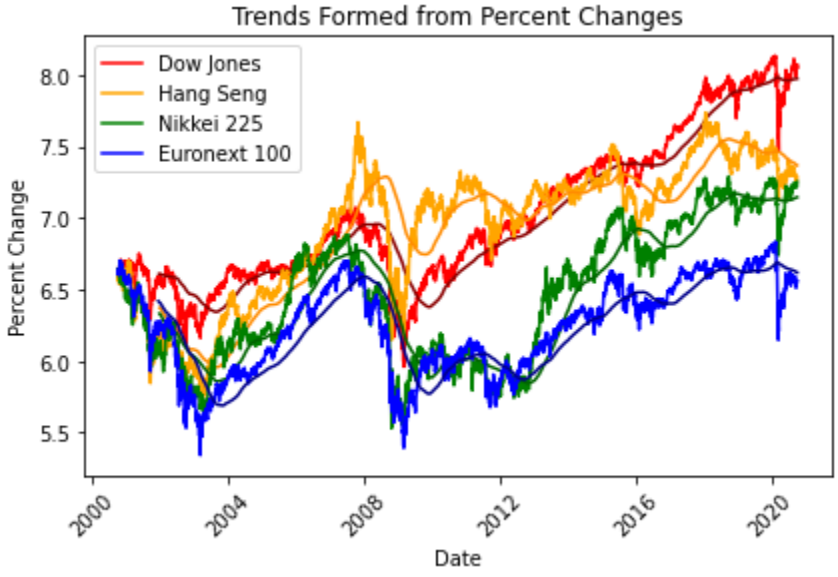
All four exchanges had closing prices in different ranges. This ranged from as high as $30,000 to as low as $400. Because the nominal value of the closing price was not consistent across exchanges, a new column was created with one that was. This new column calculated the percent change of the closing price relative to the closing price of the day prior.

The newly calculated percent change column provided a consistent way to track the performance of each index across multiple exchanges, and would provide the basis of our Y variable when running scikitlearn models.

**Nominal Price Trends by Index Unadjusted**

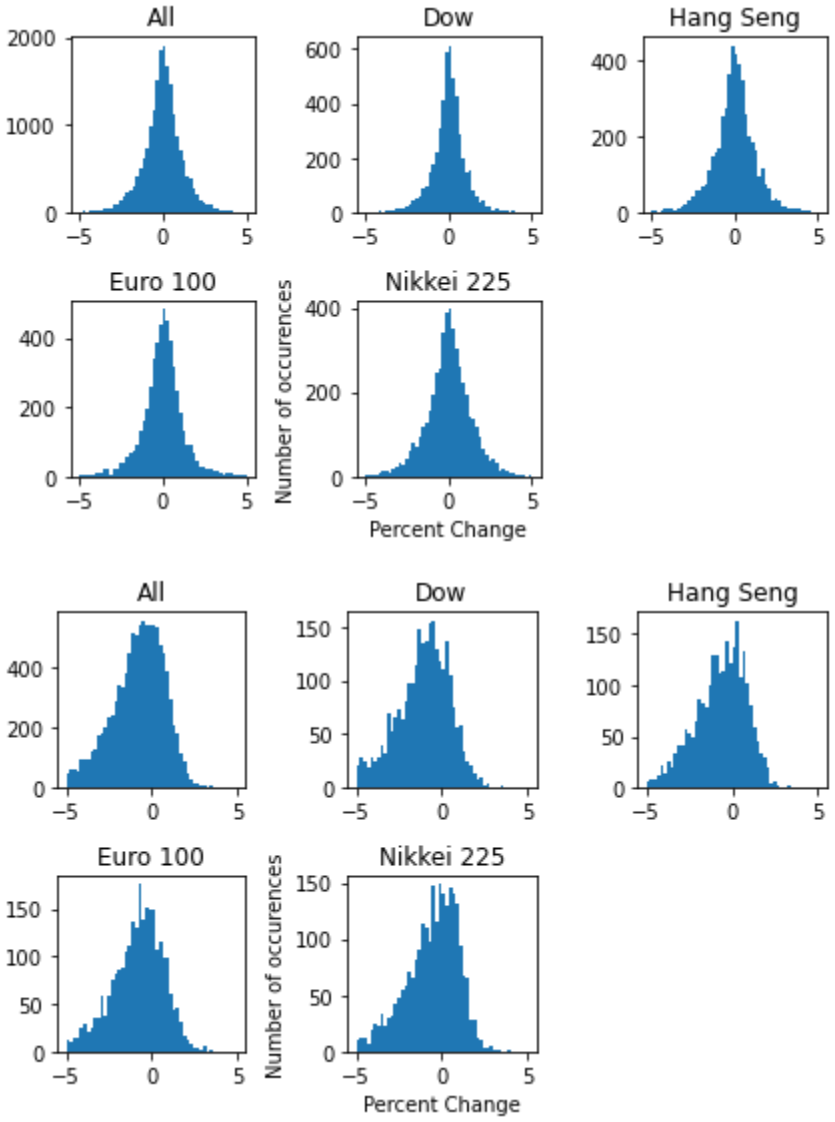



**Scaled Price Trends By Index Based on Daily Closing Price Percent Change Scaled by Log2**



When analyzing the new percent change variable, it was found that it was a double exponential distribution with a slight negative skew.

**Distribution Plots Unscaled and Log2 Scaled**



## Pre-processing

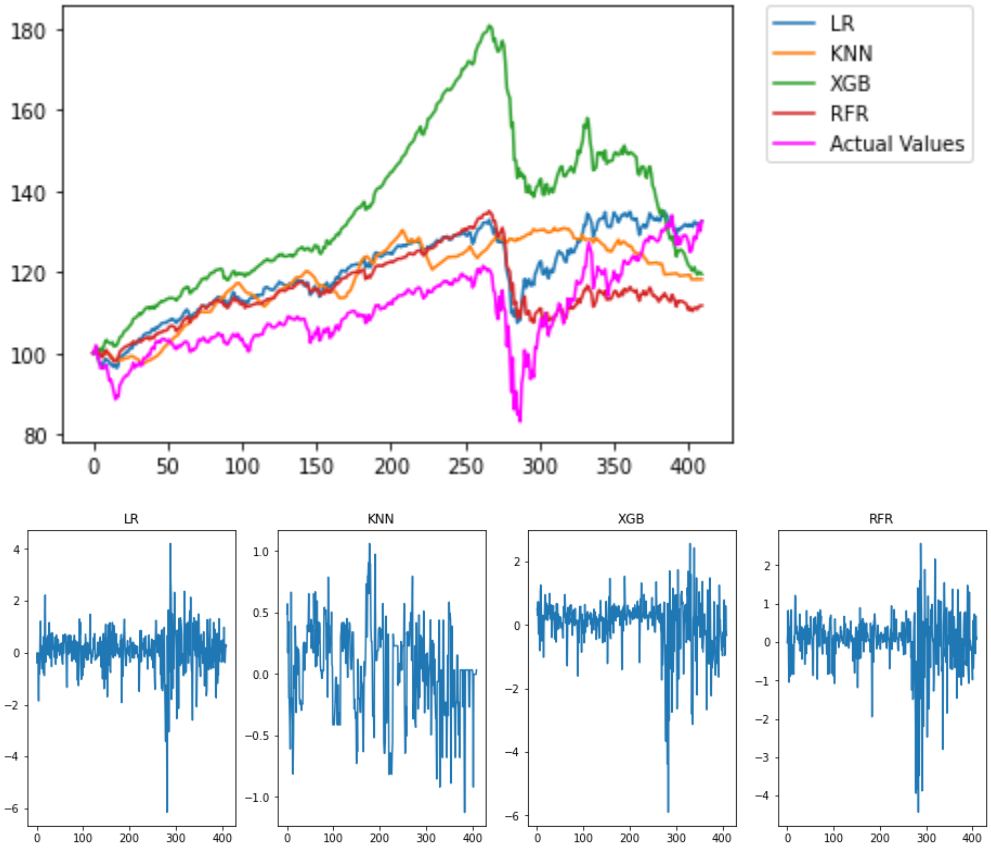
For preprocessing, I dropped all the string variable columns like date, and Index in preparation for test train split. For test train split, I used the unconventional TimeSeriesSplit() package provided by scikitlearn to maintain the ordered columns. I also split, and remerged the dataset by indices with a suffix at the end of each column to keep track of which column belongs to which index dataset. The single test dataset consisted entirely of the

To deal with the fact that data for the same closing day might be a latter date in Asian markets in accordance with GMT time, I also shifted the European and New York Datasets back by one day, as well as removing the final date of the asian Indices to keep the number of rows equal. I referred to this throughout the notebook as the “shifted” dataset.

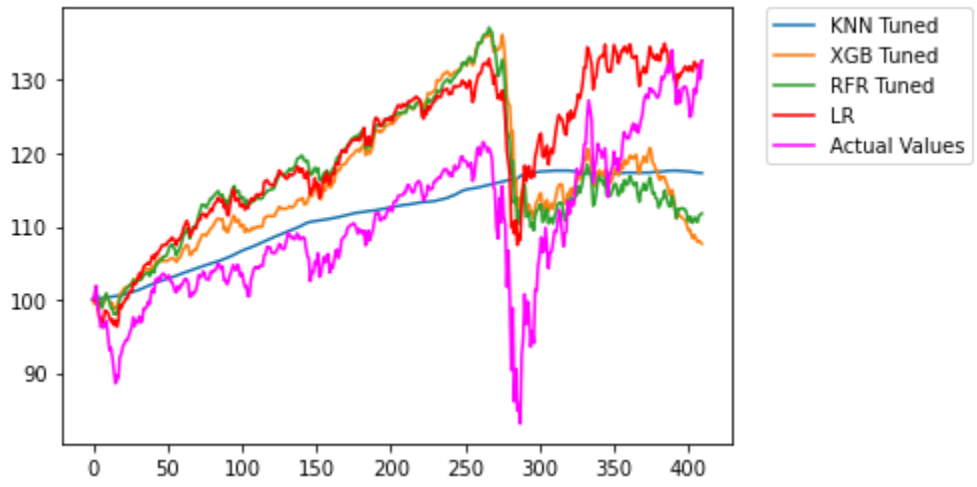
## Modeling

For modeling, I determined that this was a Regression problem. I therefore chose two models to run the data through. These are Linear Regression, KNearestRegressor, Random Forest Regressor, and XGBoostRegressor.

The performances of these initial models are shown in the following chart:

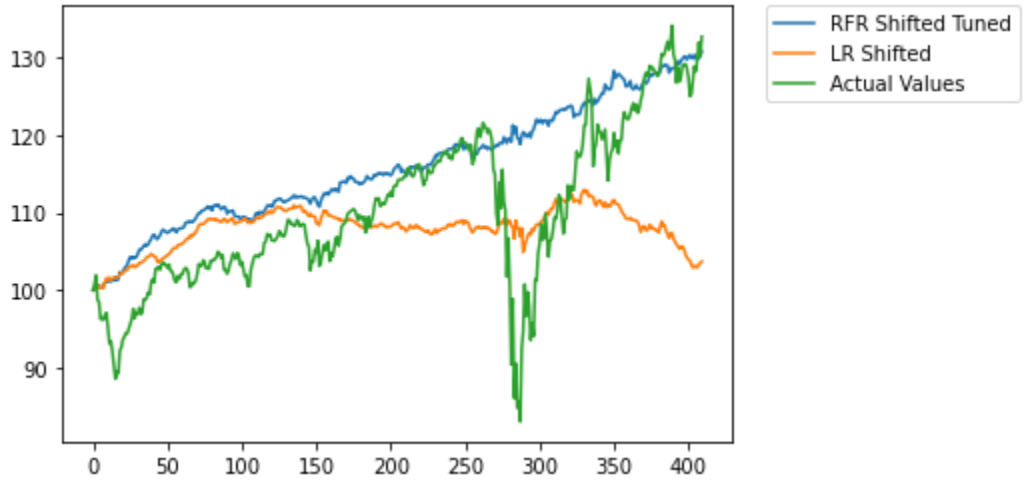


Initially it seemed Linear Regression, and Random Forests best baseline models. KNR completely missed the mark, and XGBoost seemed to follow the pattern, but was off on scale. I re-ran the same models with some hyperparameter tuning.
After running them through gridsearchcv, I came up with the following tuned chart:




Tuning XGBoost seemed to have brought it closer in line to Random Forest. However it took much longer to calculate the parameters for it, therefore in this particular implementation it is inferior to Random Forest.

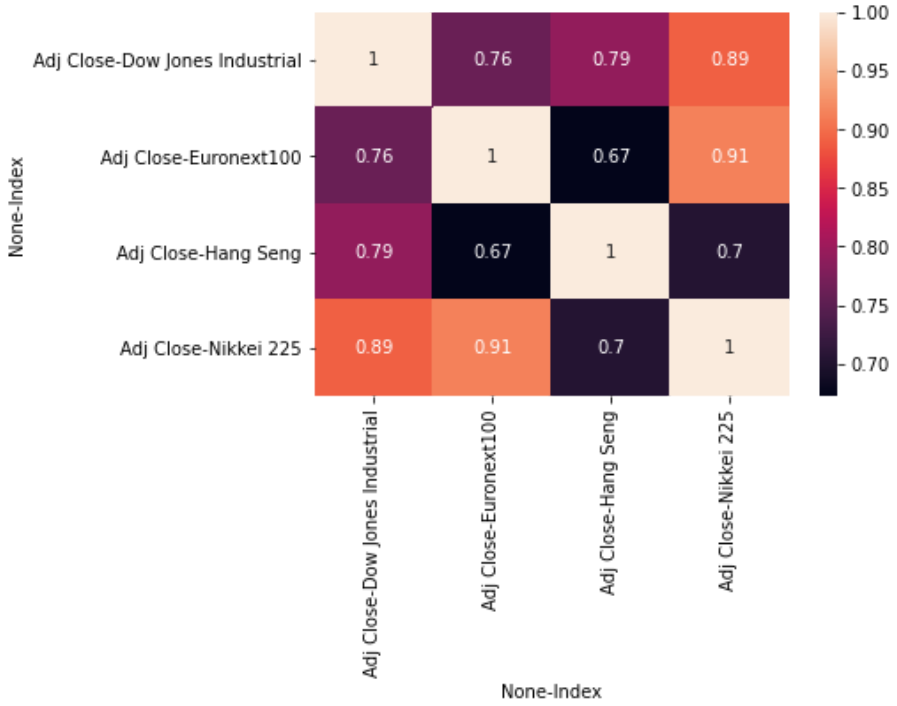
I ran Linear and Random Forest through the time shifted data and came up with the following:




Shifting the data by a day seemed to have thrown off the models.

## Findings and Conclusion

In short,using same day data for NYSE closing percent changes, one can predict using linear regression how much of a percentage change indices that close later like the Euronext100, or Hang Seng, to within a negative mean squared error of roughly -0.798. This is in line with the correlation heatmap generated during Exploratory Analysis, where the average of the correlation values was 0.7867:




It is also in line with the general visual trend presented by the chart plotting the percent change movement of the four indices: