Using news to predict stock market trends

Data Wraggling

This project uses data from Kaggle resources to predict stock market data from news data. First, two data sets (one for the stock market data and one for the news data) were downloaded from the Kaggle kernel using urlretrieve. A summary of each data set is as follows:

Market Data

- Dataset contains 4,072,956 rows and 17 columns.
- Data contents include date/time stamps for the market data, a company name and identification code, and data pertaining to stock market measurements (e.g. opening and closing prices, volume, returns over 10 day period, etc.)
- There are null values in several columns. In the returnsClosePrevMktres1 and returnsOpenPrevMktres1 columns there are 15,980 null values in each column. In the returnsClosePrevMktres10 and returnsOpenPrevMktres10 there are 93,054 null values in each column. These values are not necessarily needed to model the system and therefore there is little need to change the null values.

News Data

- Dataset contains 9,328,750 rows and 36 columns.
- Data contents include date/time stamps for news segment, the company name and identification code that the news segment is referring to, and details of the news segment (e.g. word count, headline, type of segment, measurement if the segment was positive, negative or neutral toward the company, etc.)
- There are null values in two columns. The headline column has 73,960 null values. The headlineTag column has 6,341,993 null values. Both of these columns will not play a role in the mathematical modeling and there is no logical way of filling them in with alternative values. Hence, the null values are kept in the dataset.

The data sets from Kaggle were already very clean and therefore there was little data cleaning necessary. After downloading and saving the data in csv files these were the steps preformed to clean and join the data sets:

- Each dataset was loaded into python using pd.read_csv(file_name, parse_dates =[1]) so that the date/time was read as a datetime object.
- The null values (see description of each data set above) were all in columns that will not be used in the mathematical analysis. Also, some of the null values (such as headlines) cannot logically be replaced with anything therefore these values were kept null.
- The news data had outliers that are deemed to be statistically realistic and therefore there is no need to adjust for these outliers. The market data had several hundred outliers in several columns that seemed to be a result of miscalculations or incomplete information in calculations. The market data had columns such as opening and closing stock data that had no outliers. It is noted that the outliers are in columns such as returnsOpenNextMktres10 or returnsClosePrevMktres10 which are columns that are calculated from opening price data from the date of the row under observation and data from a date before or after the observation row date. Sometimes a company will appear

or disappear from the data set over time; reasons could include bankruptcy, the make or drop from the ranking of companies on the list or the company is newly formed. This means that if a company appears or disappears from the list of companies in the data set there will be an error in calculated some of these values. Hence, the outliers in such cases are not realistic values but rather are calculation errors. It is found that there are 248 outliers in the returnOpenNextMktres10 column; this column is a key value that will be used when modeling. There are two ways one could deal with these outliers. One, the outlier can be replaced with a mean value for that specific company. Two, the entire row can be eliminated completely from the data set. Since there are over 4 million rows of data and only a few hundred outliers it seems reasonable that eliminating the outlier rows would not alter analysis results significantly. Also, there is no basis for changing the value of the outlier therefore it is more logical to remove the data. Hence, the outlier rows were eliminated from the data set.

- The last step of data wraggling was to join the two data sets. The logical way to join the data sets was to join them along the company name and the date/time. The challenge is that the news data and the stock data have timestamps at different times during the day. Hence, they will never exactly line up and therefore the data cannot be directly joined using the date/times provided. To join the data effectively we first assume that the data we want to join is only day-to-day and the time of day is irrelevant. Therefore, the datetime variables in the market data and the news data are converted to datetimes with the date only and no timestamp. Then the market and news data were merged along the company names and the date. This resulting dataframe was then saved the a csv file to be used further.