Stock Price Forecasting Using Information from Yahoo Finance and Google Trend

Selene Yue Xu (UC Berkeley)

Abstract:

Stock price forecasting is a popular and important topic in financial and academic studies. Time series analysis is the most common and fundamental method used to perform this task. This paper aims to combine the conventional time series analysis technique with information from the Google trend website and the Yahoo finance website to predict weekly changes in stock price. Important news/events related to a selected stock over a five-year span are recorded and the weekly Google trend index values on this stock are used to provide a measure of the magnitude of these events. The result of this experiment shows significant correlation between the changes in weekly stock prices and the values of important news/events computed from the Google trend website. The algorithm proposed in this paper can potentially outperform the conventional time series analysis in stock price forecasting.

Introduction:

There are two main schools of thought in the financial markets, technical analysis and fundamental analysis. Fundamental analysis attempts to determine a stock's value by focusing on underlying factors that affect a company's actual business and its future prospects. Fundamental analysis can be performed on industries or the economy as a whole. Technical analysis, on the other hand, looks at the price movement of a stock and uses this data to predict its future price movements.

In this paper, both fundamental and technical data on a selected stock are collected from the Internet. Our selected company is Apple Inc. (aapl). We choose this stock mainly because it is popular and there is a large amount of information online that is relevant to our research and can facilitate us in evaluating ambiguous news. Our fundamental data is in the form of news articles and analyst opinions, whereas our technical data is in the form of historical stock prices. Scholars and researchers have developed many techniques to evaluate online news over the recent years. The most popular technique is text mining. But this method is complicated and subject to language biases. Hence we attempt to use information from the Yahoo finance website and the Google trend website to simplify the evaluation process of online news information.

In this paper, we first apply the conventional ARMA time series analysis on the historical weekly stock prices of aapl and obtain forecasting results. Then we propose an algorithm to evaluate news/events related to aapl stock using information from the Yahoo finance website and the Google trend website. We then regress the changes in weekly stock prices on the values of the news at the beginning of the week. We aim to use this regression result to study the relationship between news and stock price changes and improve the performance of the conventional stock price forecasting process.

Literature review:

The basic theory regarding stock price forecasting is the Efficient Market Hypothesis (EMH), which asserts that the price of a stock reflects all information available and everyone has some degree of access to the information. The implication of EMH is that the market reacts instantaneously to news and no one can outperform the market in the long run. However the degree of market efficiency is controversial and many believe that one can beat the market in a short period of time[1].

Time series analysis covers a large number of forecasting methods. Researchers have developed numerous modifications to the basic ARIMA model and found considerable success in these methods. The modifications include clustering time series from ARMA models with clipped data[2], fuzzy neural network approach[3] and support vector machines model[4]. Almost all these studies suggest that additional factors should be taken into account on top of the basic or unmodified model. The most common and important one of such factors is the online news information related to the stock.

Many researchers attempt to use textual information in public media to evaluate news. To perform this task, various mechanics are developed, such as the AZFin text system[5], a matrix form text mining system[6] and named entities representation scheme[7]. All of these processes require complex algorithm that performs text extraction and evaluation from online sources.

Data:

Weekly stock prices of aapl from the first week of September 2007 to the last week of August 2012 are extracted from the Yahoo finance website. This data set contains the open, high, low, close and adjusted close prices of aapl stock on every Monday throughout these five years. It also contains trading volume values on these days. To achieve consistency, the close prices are used as a general measure of stock price of aapl over the past five years.

We use the Key Developments feature under the Events tab on the Yahoo finance website to extract important events and news that are related to aapl stock over the past five years. The Key Developments of aapl from the first week of August 2007 to the last week of August 2012 are recorded. Most of these news comes from the Reuters news website. Reuters is an international news agency and a major provider of financial market data. Each piece of news is examined in greater details in order to determine whether the news should have positive or negative influence on the stock price. The news is then assigned a value of +1 or -1 accordingly. If the influence of the news is highly controversial or ambiguous, then the news is assigned a zero value. The starting point of the Yahoo finance news data is set one month earlier than the starting point of the stock price data because we eventually want to study the relationship between news at one time and stock price at a later time.

Google Trend is a public web facility of Google Inc. It shows how often a specific search term is entered relative to the total search volume on Google Search. It is possible to refine the request by geographical region and time domain. Google Trend provides a very rough measure of how much people talk about a certain topic at any point of time. Online search data has gained increasing relevance and attention in recent years. This is especially true in economic studies. Google Search Insights, a more sophisticated and advanced service displaying search trends data has been used to predict several economic metrics including initial claims for unemployment, automobile demand, and vacation destinations[8]. Search data can also be used for measuring consumer sentiment[9]. In this paper, weekly (every Sunday) search index of the term "aapl" on a global scale from the first week of August 2007 to the last week of August 2012 are extracted from the Google trend website. The starting point of the Google trend data is set one month earlier than the starting point of the stock price data because we eventually want to study the relationship between news at one time and stock price at a later time. In addition, this helps to align the news data from the Yahoo finance website with the data from the Google trend website.

Assumptions:

Several assumptions regarding the Google trend data, the Yahoo finance news information and the Yahoo finance stock price data are made:

1. The global Google trend index of the search term "aapl" shows how much people around the world search up aapl stock at a time and hence gives a rough measure of the impact of the news/events at that time. Note that here we acknowledge the "roughness" of this measure. After all, the objective of this paper is to simplify the process of news evaluation.
2. The major source of news information is the key development function on the Yahoo finance website. We assume that this function gives a list of significant news and important events related to Apple Inc. throughout the

past five years. In other words, we assume the criterion used by the Yahoo finance website when judging if certain news is important enough to be included in our analysis. Note that this list might not be a comprehensive list of the important news related to Apple Inc. over the past five years. But again, this paper attempts to simplify news selection process. Hence a simple criterion from the Yahoo finance website is used.

3. The historical weekly close prices of aapl reflect changes in the real values of aapl stock during this period of time.
4. Keeping other factors constant, positive news and negative news have equal impact on stock prices. Both positive news and negative news are assigned an initial magnitude of 1, which is later adjusted by the Google trend index values. The only difference is their sign: positive news is assigned to +1 while negative news is assigned to -1.
5. This paper assumes that the impact of news on stock prices follows an exponential function with a parameter of 1/7. The specific algorithm to calculate the value of news at a certain time after it is first released is shown in the next section. Essentially, we expect the news to die out after about two weeks.

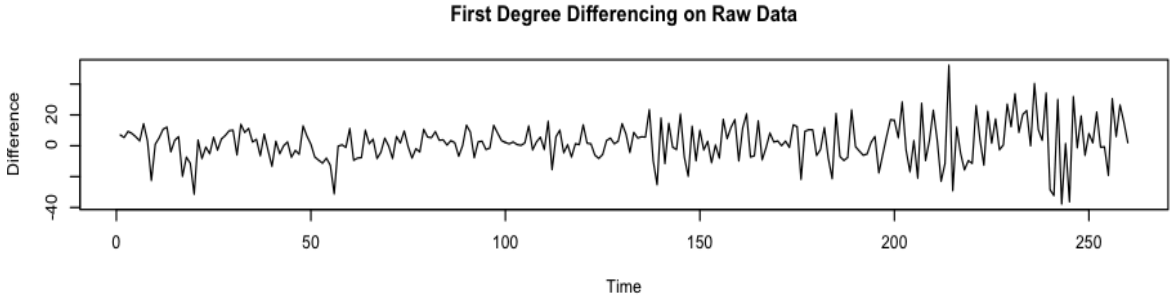
Analysis of Data:

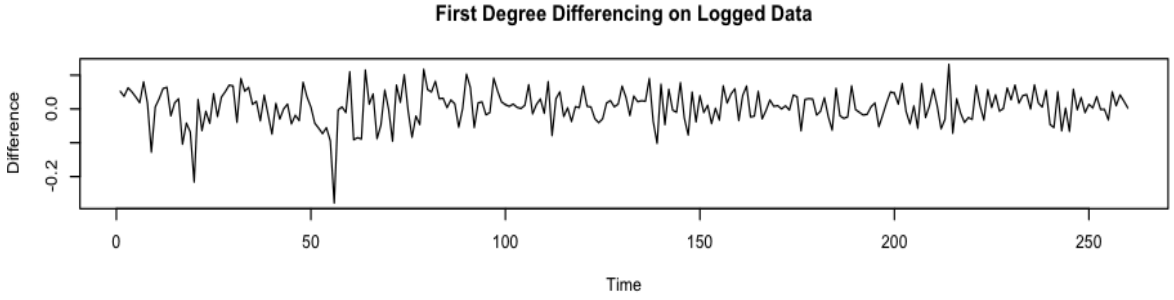
1. The basic ARIMA model analysis of the historical stock prices:

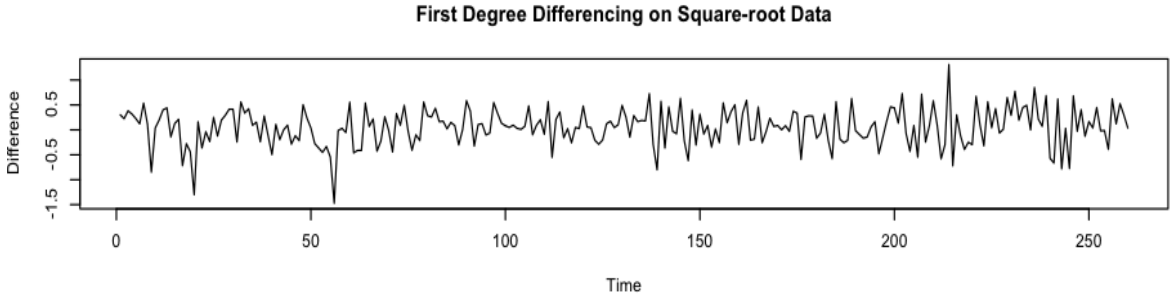
To perform the basic ARIMA time series analysis on the historical stock prices, we first make a plot of the raw data, i.e. the weekly close prices of aapl over time. The plot is shown below:

This plot shows that the close price of aapl increases in general over the past five years. However, there is no apparent pattern in the movement of the stock price. The variance of the stock price seems to increase slightly with time. The stock price is especially volatile near the end. These observations imply that a log or square-root transformation of the raw data might be appropriate in order to stabilize variance.

Since there is no linear or other discernable mathematical pattern in the data, we perform first degree differencing on the raw data and on the transformed data. We plot the results and compare them:

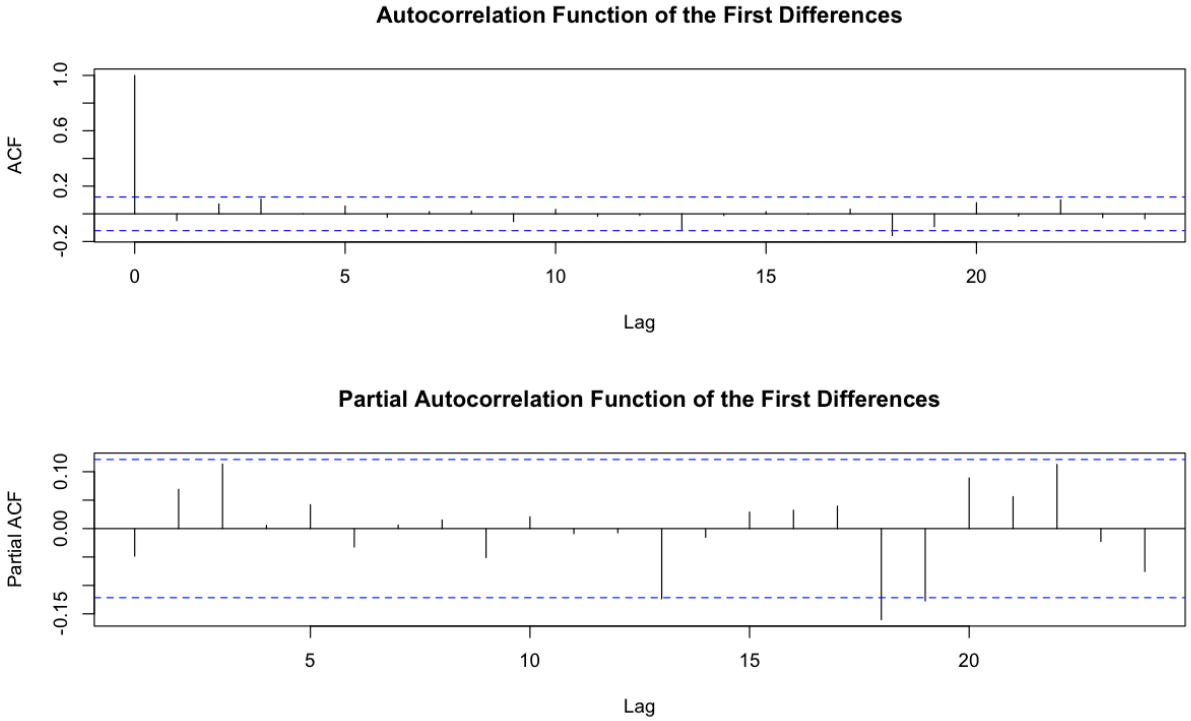






Comparing the three plots, it is clear that the first differences of the raw data show increasing variance over time whereas the first differences of the transformed data show relatively stable variance over time. Hence we confirm that it is indeed desirable to perform transformation on the raw data. Comparing the second and the third plot, it seems that both the logarithm and the square-root transformations do a good job stabilizing the variance and the distributions of the first differences in both plots look random. We decide to use a square-root transformation because it seems to do a better job tuning down some of the extreme values such as the ones near the 50^{th} data point.

Next we plot the autocorrelation function and the partial autocorrelation function of the first differences of the transformed data:



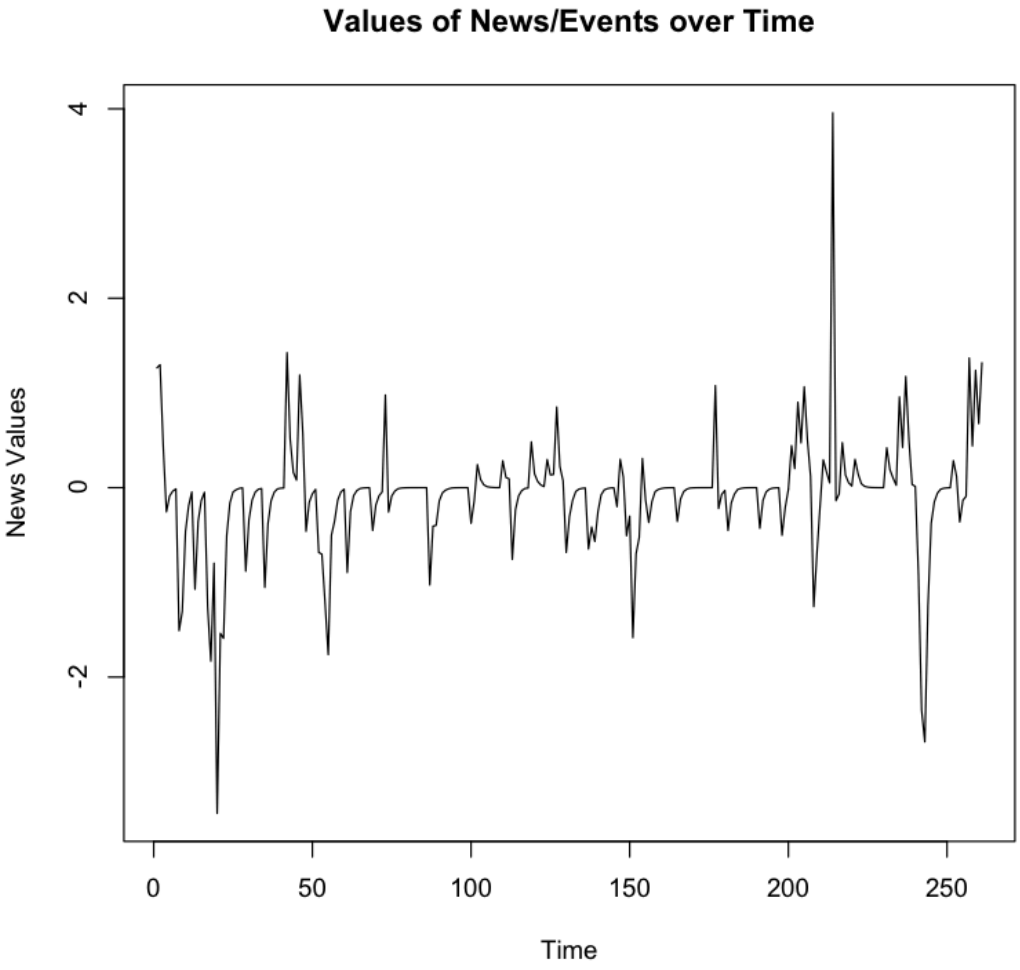
From these plots, it seems that the first differences of the transformed data are random. There doesn't seem to be an ARMA relationship in the first differences with time. The transformed stock prices essentially follow an ARIMA(0,1,0) process. This is essentially a random walk process. The random walk model in stock price forecasting has been commonly used and studied throughout history[10]. The random walk model has similar implications as the efficient market hypothesis as they both suggest that one cannot outperform the market by analyzing historical prices of a certain stock.

2. Algorithm for computing the value of news at a certain time:

As mentioned earlier, important news and events related to Apple Inc. over the past five years are recorded and assessed to be either +1 or -1. An exponential algorithm is used to simulate the change of impact of a piece of news over time. The details of the algorithm is as follow:

a. To start with, each day in the past five years is assigned a value of +1, -1 or 0 depending on if there is important news/event on that day and if the news is positive, negative or neutral respectively.
b. The impact of the news decreases exponentially such that n days later, the absolute value of the news/event becomes exp(-n/7) and the sign still follows the original sign of the news/event. This exponential form means that on the day that certain news/event occurs, the absolute value of that news/event is always exp(0)=1. One day later, the absolute value becomes exp(-1/7)=86.69% of the original absolute value. A week later, the absolute value becomes exp(-7/7)=36.79% of the original absolute value. Two weeks later, the absolute value becomes exp(-14/7)=13.53% of the original absolute value. We design the algorithm in a way that news/event almost dies off after two weeks.
c. On any day within the past five years, the value of news/event on that date is the sum of all news/events values on and before that date, which are calculated using the aforementioned algorithm.
d. After computing the value of news/event for every single day in the past five years, the days corresponding to the Google trend dates (every Sunday in the past five years) are selected. The news/events value computed from the key developments feature on the Yahoo finance website gives the sign of news/event at a particular time and the Google trend index data gives the magnitude of the news/event at that time. They are multiplied together to give a measure of the final value of news/event at that point of time.

We plot the final values of news/events over time to get a rough picture of our news vector:



From the plot, we see that a lot of the time the news values are close to zero. This means that there is little news to influence the public at those times. There are also some large spikes in the plot. For example, there is a big spike in the negative direction around the 20th data point; this means that there is some very bad and significant news about Apple Inc. around this time. There is a big spike in the positive direction around the 220th data point; this means that there is some very good and significant news about Apple Inc. around this time.

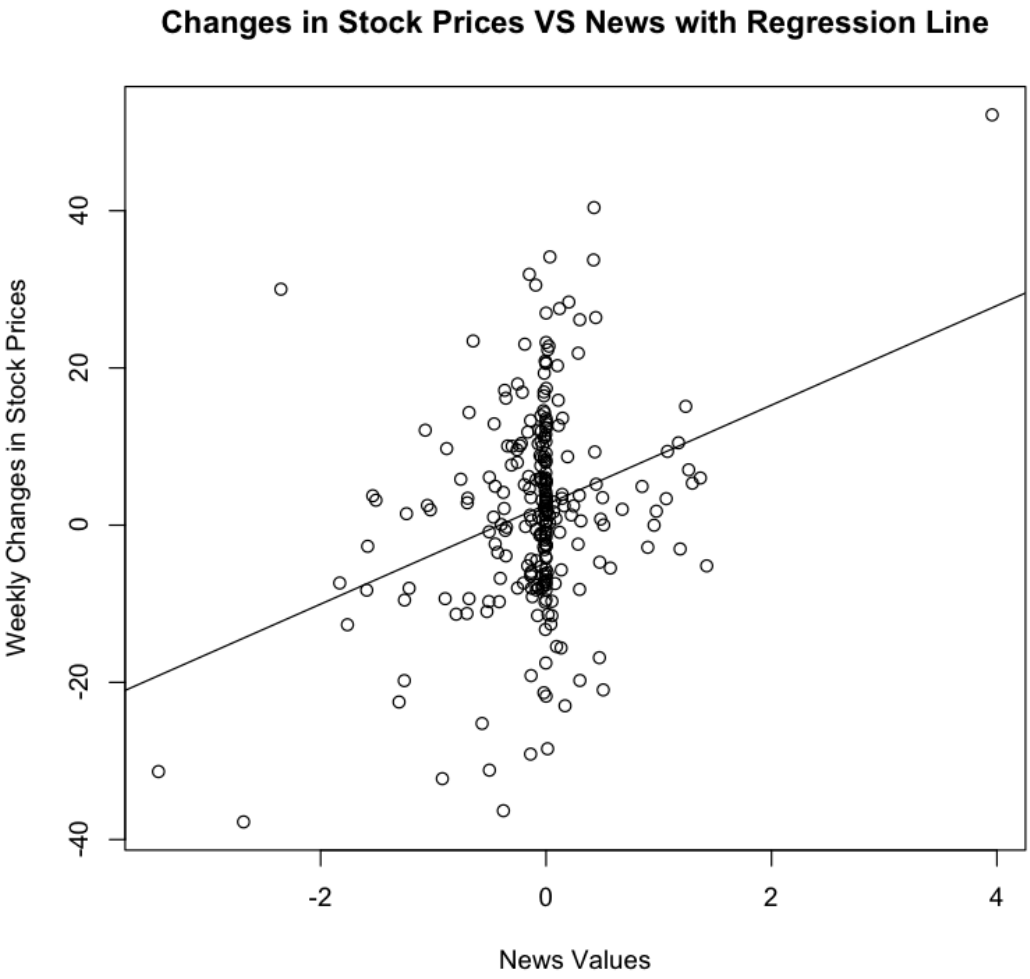
3. Regression of weekly stock price changes on the news values at the beginning of each week:

In order to study the relationship between news/event at certain time and stock price at a later time, we regress the weekly stock price changes on the news values from the previous week. The summary of the regression is shown below:

```
> summary(mod)

Call:
lm(formula = d1 ~ inf2, data = d)

Residuals:
    Min      1Q  Median      3Q     Max
-36.547  -7.656  -0.669   6.981  42.297

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.6130     0.7694   3.396 0.000791 ***
inf2          6.3290     1.2347   5.126 5.81e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.28 on 258 degrees of freedom
Multiple R-squared: 0.09243,  Adjusted R-squared: 0.08892
F-statistic: 26.28 on 1 and 258 DF,  p-value: 5.808e-07
```

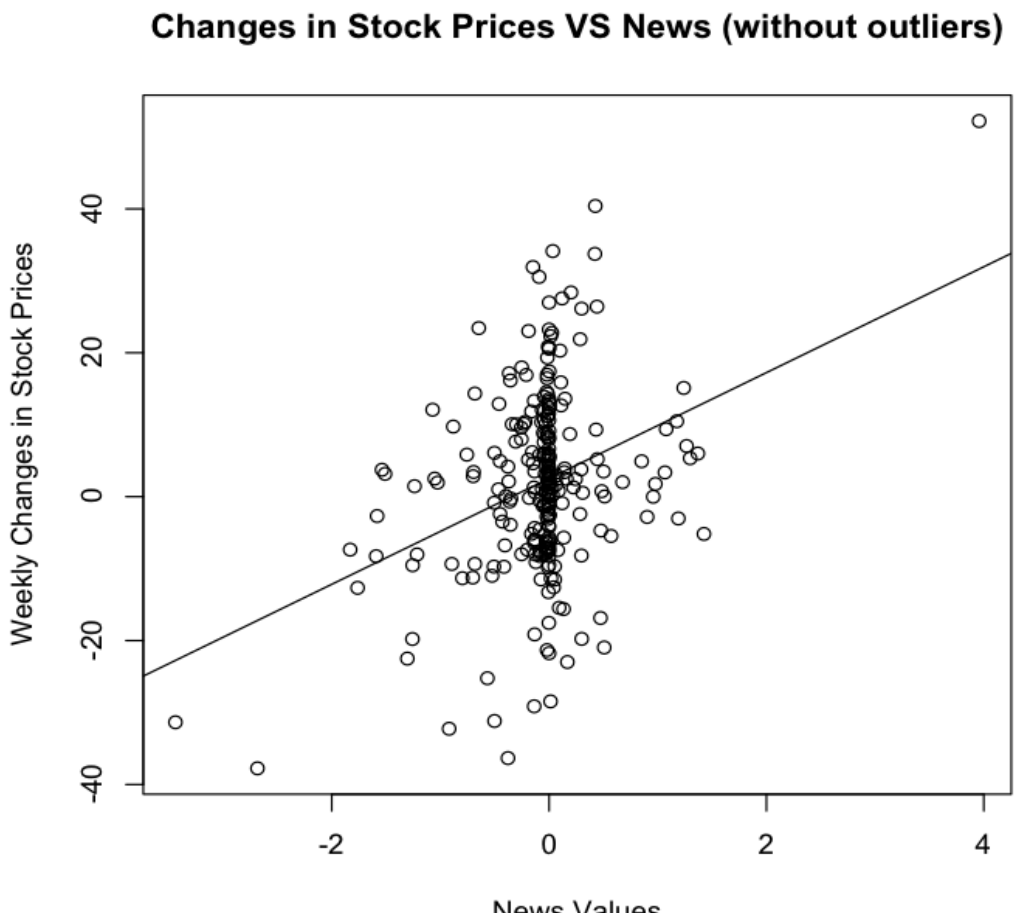
The regression result shows that there seems to be a very significant and positive correlation (6.3290) between the weekly changes in stock prices and the news values at the beginning of each week. To put this into words, one unit increase in news values should result in roughly 6.33 unit increase in stock price change by the end of the week. However we notice that the R-squared value, which shows the proportion of the variability in the weekly stock price changes explained by the model, is very small (0.09243). This means that the model is not doing a good job predicting the changes in stock prices.

In order to further examine the regression data, we plot the weekly stock price changes against the news values from the previous week with the regression line drawn:



From the plot, we notice that there are potential outliers in the data such as the point well above the regression line near news value -2. An outlier is a data point that is markedly distant from the rest of the data and does not fit the current model. Identifying outliers helps to distinguish between truly unusual points and residuals that are large but not exceptional. We use the Jacknife residuals and the Bonferroni inequality method to come up with a conservative test at a level of 10% to filter out the outliers in our data. This procedure picks out the data point we just mentioned as the sole outlier in our data set. We exclude this data point for the rest of our analysis. The regression result and the plot of the new data set without the outlier are shown on the next page:

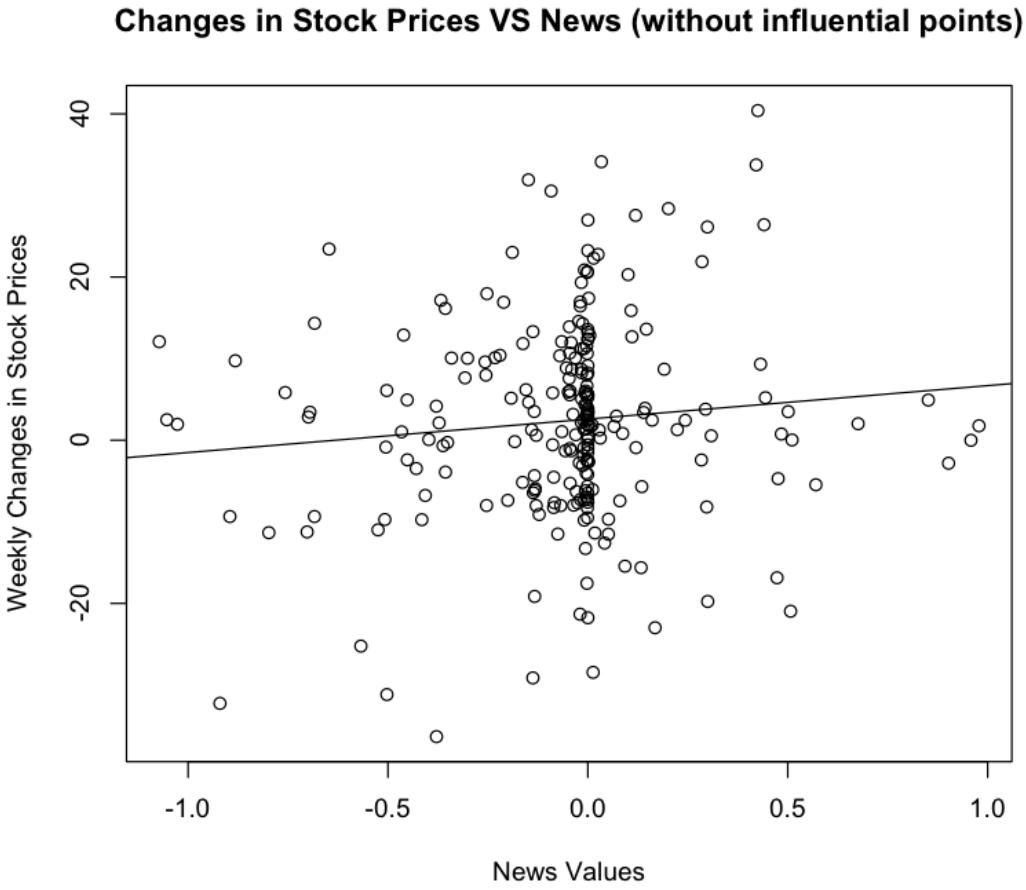
```
> summary(mod3)

Call:
lm(formula = d1 ~ inf2, data = d3)

Residuals:
    Min      1Q  Median      3Q     Max
-36.077  -7.609  -0.626   7.550  34.741

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.5317     0.7522   3.366  0.00088 ***
inf2          7.3542     1.2392   5.934  9.5e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12 on 257 degrees of freedom
Multiple R-squared: 0.1205,   Adjusted R-squared: 0.1171
F-statistic: 35.22 on 1 and 257 DF,  p-value: 9.498e-09
```



The regression results in a similar estimate of the coefficient on the news variable (7.3542), which is also significant at level 0.001. This shows that the estimate for the correlation coefficient between weekly stock price changes and news values at the beginning of each week is relatively stable. There is slight improvement in the R-squared value (0.1205). A little over 10% of the variability in the weekly stock price changes can be explained by the model. Although the R-squared value shows slight increase, it is still very small, implying that the model is still not doing a great job predicting the changes in stock prices.

We further examine the model by analyzing data leverages. If leverage is large, the data point will pull the fitted value toward itself. Such observations often have small residuals and do not show up as outliers on the residual plots. The fit looks good, but is dominated by a single observation. If the leverage of a data point is substantially greater than k/n, where k is the number of parameters being estimated and n is the sample size, then we say that it has high leverage. A rule of thumb says that if leverage is bigger than 2k/n, then this data point is a candidate for further consideration. When checking for high leverage points in our data set using R, we find 24 influential points. If we take out these points and perform regression on the remaining data. The regression summary and plot are shown here:

```
> summary(mod2)

Call:
lm(formula = d1 ~ inf2, data = d2)

Residuals:
    Min      1Q  Median      3Q     Max
-37.360  -8.038  -0.710   6.825  36.070

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.5845     0.8018   3.224  0.00145 **
inf2          4.1054     2.5734   1.595  0.11200
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.11 on 234 degrees of freedom
Multiple R-squared: 0.01076,     Adjusted R-squared: 0.006531
F-statistic: 2.545 on 1 and 234 DF,  p-value: 0.112
```



The estimate of the correlation coefficient between weekly stock price changes and news values is no longer significant. In fact, now the distribution of weekly stock price changes over news values seems rather random. Hence we suspect that the previous regression result is dominated by a number of highly influential data points in our sample and does not represent general situations very well.

4. Regression of weekly stock price changes on dummy variables representing different intervals of news values:

In addition to studying the relationship between stock price changes and specific news values, we would like to examine the general behavior of stock price changes when the news value falls within a certain range. In order to do this, we divide the news values into different intervals, represent them by dummy variables and regress the weekly changes in stock prices onto these dummy variables (without an intercept term). When we perform a regression like this, the estimated coefficient for each dummy variable in the regression output is the average of the weekly stock price changes that correspond to news values within that particular interval.

First we divide the news values into two general categories: negative and positive. The regression result is shown below:

```
> summary(m)

Call:
lm(formula = d1 ~ pdum + ndum - 1)

Residuals:
    Min      1Q  Median      3Q     Max
-38.722  -7.444  -0.337   7.688  47.583

Coefficients:
     Estimate Std. Error t value Pr(>|t|)
pdum   4.6173     1.4565   3.170  0.00171 **
ndum   0.9723     0.9448   1.029  0.30435
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.78 on 258 degrees of freedom
Multiple R-squared: 0.04128,     Adjusted R-squared: 0.03385
F-statistic: 5.555 on 2 and 258 DF,  p-value: 0.004347
```

From the regression summary, we see that the mean of weekly stock price changes for positive news is 4.6173. It is significant at a level of 0.01. This makes sense since we would expect positive news to be followed by a positive change in stock price.

The mean of weekly stock price changes for negative news is 0.9723. It is not significant at any level smaller than 0.1. This contradicts to what we would have expected since normally negative news should be followed by a negative change in stock price.

Next we divide the news values into finer intervals: smaller than -2, bigger than or equal to -2 and smaller than -1.5, bigger than or equal to -1.5 and smaller than -1, bigger than or equal to -1 and smaller than -0.5, bigger than or equal to -0.5 and smaller than 0, bigger than or equal to 0 and smaller than 0.5, bigger than or equal to 0.5 and smaller than 1, bigger than or equal to 1 and smaller than 1.5, and finally bigger than or equal to 1.5. The regression result over these nine dummy variables (without intercept term) is shown below:

```
> summary(m2)

Call:
lm(formula = d1 ~ dum1 + dum2 + dum3 + dum4 + dum5 + dum6 + dum7
+
    dum8 + dum9 - 1)

Residuals:
    Min      1Q  Median      3Q     Max
-38.805  -7.030  -0.465   7.573  43.047

Coefficients:
     Estimate Std. Error t value Pr(>|t|)
dum1  -13.027      7.057  -1.846  0.06606 .
dum2   -4.013      4.990  -0.804  0.42197
dum3   -5.230      4.321  -1.210  0.22730
dum4   -5.046      2.964  -1.702  0.08996 .
dum5    2.475      1.001   2.471  0.01412 *
dum6    4.687      1.605   2.920  0.00381 **
dum7   -1.890      4.074  -0.464  0.64312
dum8    5.390      4.074   1.323  0.18704
dum9   52.200     12.222   4.271 2.76e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.22 on 251 degrees of freedom
Multiple R-squared: 0.147, Adjusted R-squared: 0.1164
F-statistic: 4.807 on 9 and 251 DF,  p-value: 6.274e-06
```

From the regression summary, we see that the averages of weekly stock price changes for news values that fall into the first four intervals are negative. But only the averages in the first interval (<-2) and the fourth interval (-1∼-0.5) are significant, both at level 0.1. The other two are not significant at any level smaller than 0.1. Also we would have expected the mean of weekly stock price changes to be more negative in the second interval (-2∼-1.5) than those in the third (-1.5∼-1) and

fourth interval (-1∼-0.5). The average of weekly stock price changes in the fifth interval (-0.5∼0) is very interesting. Contrary to our expectation that it should be negative, it is actually positive here and is significantly so (at a level 0.05). We will discuss about this in the next section. The averages of weekly stock price changes for news values that fall into the last four intervals, except for the seventh interval, are positive, just as what we would have expected. The averages in the sixth interval (0∼0.5) and the last interval (>1.5) are significant at level 0.01 and level 0.001 respectively. The mean in the eighth interval (1∼1.5) is not significant at any level smaller than 0.1. The mean in the seventh interval (0.5∼1) is a negative value, which contradicts to our expectation. But it is not significant at any level smaller than 0.1.

All in all, when dividing news values into different intervals, the trend of average weekly stock price changes is similar to what we would have expected, with a couple of exceptions.

Discussion:

1. We assign the same initial magnitude of 1 to positive and negative news. This might not accurately reflect how people generally react to good and bad news. There are numerous studies showing that negative information has a much greater impact on individuals' attitudes than does positive information[11]. Hence it might be more reasonable to assign a slightly larger initial absolute value to negative news than to positive news.
2. A lot of the time, the judgment of whether a piece of news is good or bad is subjective. People might have different, even opposite, interpretations of the same information. Since the determination of the sign of news is done manually in our experiment, this process is subject to human biases. This is especially true when the news is ambiguous and not as significant as other news. For example, in the last section, we see that the stock price on average goes up when there is "slightly negative" news. It is possible that some of these news are deemed negative in our analysis but actually are considered goods news by popular opinion. It is hard to accurately and systematically determine whether news is good or bad in the opinion of the majority of the public when there is ambiguity.
3. We use an exponential function to simulate how fast the impact of news fades away. There might be better alternatives to simulate this process. Moreover, positive news and negative news might fade out in different manners. This is not accounted for in our analysis.
4. Our main news source is the key developments function on the Yahoo finance website, which mostly comes from the Reuters News website. This might not provide a comprehensive list of all the important news/events that took place over the past five years.
5. We only extract the Google trend index record for the term "aapl". Often when people search about a company, they would type in the name of the company or the news directly instead of using a stock symbol. The Google trend data for more relevant terms could be added to the ones we get for "aapl" in order to get a more accurate measure of how much people care about the news related to this stock at a particular time.

Conclusion:

Our initial analysis shows significant correlation between news values and weekly stock price changes. But we need to be careful when using this result since it is likely that the result is dominated by a number of influential observations and is not reflective of the general trend. In general the weekly stock price changes within different intervals of news values behave in the same way as what we expect. But we also find some interesting exceptions worth looking into. There are a number of limitations and shortcomings in our experiment as mentioned in the discussion section and potential improvements can be made to our data collection and analysis method. Further researches can be done with possible improvements such as more refined search data and more accurate algorithm to compute news values.

Reference:

1. Gili Yen and Cheng-Few Lee, “Efficient Market Hypothesis (EMH): Past, Present and Future” in Review of Pacific Basin Financial Markets and Policies, vol 11, issue 2, 2008

2. A. J. Bagnall and G. J. Janacek, “Clustering Time Series from ARMA Models with Clipped Data” in KDD, W. Kim, R. Kohavi, J. Gehrke, and W. DuMouchel, Eds. ACM, 2004

3. D. Marcek, “Stock Price forecasting: Statistical, Classical and Fuzzy Neural Network Approach” in MDAI, V. Torra and Y. Narukawa, Eds, vol. 3131. Springer, 2004

4. “A Hybrid ARIMA and Support Vector Machines Model in Stock Price Forecasting” in Omega, the International Journal of Management Science. vol. 33, no. 3, 2005.

5. Schumaker, Robert P., and Hsinchun Chen. "Textual Analysis of Stock Market Prediction using Breaking Financial News: The AZFin text system." *ACM Transactions on Information Systems (TOIS)* 27.2 (2009)

6. Deng, Shangkun, et al. "Combining Technical Analysis with Sentiment Analysis for Stock Price Prediction." *Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on*. IEEE, 2011

7. BABU, M. SURESH, DRN GEETHANJALI, and V. RATNA KUMARI. "TEXTUAL ANALYSIS OF STOCK MARKET PREDICTION USING FINANCIAL NEWS ARTICLES." *The Technology World Quarterly Journal* (2010)

8. Choi, Hyunyoung, and Hal Varian. "Predicting the present with google trends." *Economic Record* 88.s1 (2012): 2-9

9. Preis, Tobias, Daniel Reith, and H. Eugene Stanley. "Complex dynamics of our economic life on different scales: insights from search engine query data." *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 368.1933 (2010): 5707-5719

10. Fama, Eugene F. "Random walks in stock market prices." *Financial Analysts Journal* (1965): 55-59.

11. Soroka, Stuart N. "Good news and bad news: Asymmetric responses to economic information." *Journal of Politics* 68.2 (2006): 372-385.

Appendix (R code):

```
price=read.csv("/Users/selenexu/Desktop/Econ Honor/aapl
 table.csv",header=TRUE)
price=price[order(price$Date,decreasing=FALSE),]
trend=read.csv("/Users/selenexu/Desktop/Econ Honor/aapl
 trends.csv",header=TRUE)
plot(trend$aapl,type='l')
plot(price$Close,type='l',xlab='Time',ylab='Close
 Prices',main='Weekly Close Prices of aapl')


d1=diff(price$Close)

logd1=diff(log(price$Close))

sd1=diff(sqrt(price$Close))


par(mfrow=c(3,1))
plot(d1,type='l',xlab='Time',ylab='Difference',main='First Degree
 Differencing on Raw Data')
plot(logd1,type='l',xlab='Time',ylab='Difference',main='First
 Degree Differencing on Logged Data')
plot(sd1,type='l',xlab='Time',ylab='Difference',main='First
 Degree Differencing on Square-root Data')

par(mfrow=c(2,1))
acf(sd1,main='Autocorrelation Function of the First Differences')
pacf(sd1,main='Partial Autocorrelation Function of the First
 Differences')

sd2=diff(sd1)
par(mfrow=c(2,1))
acf(sd2,main='Autocorrelation Function of the Second
 Differences')
pacf(sd2,main='Partial Autocorrelation Function of the Second
 Differences')
arima(sqrt(price$Close),order=c(0,2,1))


inf2=info[1:(length(info)-1)]
d=data.frame(d1,inf2)
mod=lm(d1~inf2,data=d)
```

```
summary(mod)
plot(inf2,d1,xlab='News Values',ylab='Weekly Changes in Stock
 Prices',main='Changes in Stock Prices VS News with Regression
 Line')
abline(a=summary(mod)$coefficients[1],b=summary(mod)$coefficients
 [2])

jack=rstudent(mod)
plot(jack, ylab='jacknife residuals',main='jacknife residuals')
q=abs(qt(.1/(260*2),(260-1-2)))
jack[abs(jack)==max(abs(jack))]
which(abs(jack)>q)
d3=d[-which(abs(jack)>q),]
mod3=lm(d1~inf2,data=d3)
summary(mod3)
plot(d3$inf2,d3$d1,xlab='News Values',ylab='Weekly Changes in
 Stock Prices',main='Changes in Stock Prices VS News (without
 outliers)')
abline(a=summary(mod3)$coefficients[1],b=summary(mod3)$coefficien
 ts[2])

influence(mod)$h
dim(d)
which(influence(mod)$h > 2*2/260 )
influence(mod)$h[which(influence(mod)$h > 2*2/260 )]
d2=d[-which(influence(mod)$h > 2*2/260 ),]
plot(d2$inf2,d2$d1,xlab='News Values',ylab='Weekly Changes in
 Stock Prices',main='Changes in Stock Prices VS News (without
 influential points)')
abline(a=summary(mod2)$coefficients[1],b=summary(mod2)$coefficien
 ts[2])
mod2=lm(d1~inf2,data=d2)
summary(mod2)


pdum=(inf2>0)*1
ndum=(inf2<0)*1
m=lm(d1~pdum+ndum-1)
summary(m)

x=d[inf2>0,]
mean(x$d1)

min(inf2)
```

```
max(inf2)
dum1=(inf2<(-2))*1
dum2=(inf2>=(-2)&inf2<(-1.5))*1
dum3=(inf2>=(-1.5)&inf2<(-1))*1
dum4=(inf2>=(-1)&inf2<(-0.5))*1
dum5=(inf2>=(-0.5)&inf2<(0))*1
dum6=(inf2>=(0)&inf2<(0.5))*1
dum7=(inf2>=(0.5)&inf2<(1))*1
dum8=(inf2>=(1)&inf2<(1.5))*1
dum9=(inf2>=(1.5))*1
m2=lm(d1~dum1+dum2+dum3+dum4+dum5+dum6+dum7+dum8+dum9-1)
summary(m2)

#the following shows how to create the news vector
news=read.csv("/Users/selenexu/Desktop/Econ
 Honor/event.csv",header=TRUE)
news$Date=as.Date(news$Date,"%m/%d/%Y")
date=seq(from=as.Date("2007-08-29"), to=as.Date("2012-09-15"),
 by=1)
length(date)
match=match(news$Date, date)
fac=1:length(date)
for (i in fac) {if (i %in% match) {fac[i]=news[match(i,match),3]}
 else {fac[i]=0}}
mul=1:length(fac)
for (i in mul) {mul[i]=sum(exp(-((i-1):0)*1/7)*fac[1:i])}
plot(mul,type='l')

d=date[5:length(date)]
da=d[seq(1,length(d),7)]
tail(da)
weeklydate=da[1:(length(da)-2)]

m=mul[5:length(mul)]
m1=m[seq(1,length(m),7)]
m2=m1[1:(length(m1)-2)]

trend=read.csv("/Users/selenexu/Desktop/Econ Honor/aapl
 trends.csv",header=TRUE)
infor=trend$aapl[5:length(trend$aapl)]
info=infor*m2
plot(info,type='l', xlab='Time',ylab='News Values',main='Values
 of News/Events over Time')
```