Predicting Stock Volatility with Time-Series Analysis

Spring 2019

Springboard Capstone Project 2

Milestone Report 1

Brian J Zamkotowicz

Capstone Project #2
Milestone Report #1

Problem Description

The purpose of the following project is to attempt to create a model that make useful predictions about the movement of Microsoft stock. In particular the focus will be on using time-series analysis in an attempt to predict the 10 day historical volatility of the stock, which is a measurement of how much the stock tends to move over a period of 10 days. From a business perspective this could be useful in a number of way. If volatility can be predicted, trading strategies can be structured around those predictions. In particular buying or selling options would be ways to profit from periods of predicted high or low volatility respectively. Another different but related use of this information would for portfolio construction. Portfolio managers could use the predictions to increase or decrease market exposure based on expected volatility. Additionally, while this paper will focus specifically on Microsoft stock, the techniques used here could be applied to any individual stock or even to indexes or futures, and therefore could be used to choose stocks when constructing a diverse portfolio.

Data Acquisition and Wrangling

The first step in preparing for the analysis of volatility in Microsoft stock was to find appropriate data. Since statistical analysis of stocks is an area of high interest, there are a wealth of resources on the internet detailing the movement of stock prices. While it would have been possible to calculate historical volatility by using the closing prices of the stock, there were some other factors I wanted to examine in relation to historical volatility, specifically the “implied volatility” of the stock based on the price of the stock’s options. Implied volatility is a measure of the expected movement of the stock. Options become more expensive when the stock is expected to have greater movements in the same way that an insurance policy is more expensive when the event it insures against is considered more likely to happen. Option premiums are also affected by the date of maturity of the option. For instance an option that expires in 30 days is more expensive that an option that expires in 10 days because there is more opportunity for the price of the underlying stock to move in the additional 10 days.

I was able to find a source of data that provided not only pricing data about Microsoft stock that included closing price, high, low, daily volume and several others, but also provided some data on options. Quandl.com contains free data on many stocks traded on exchanges spanning the globe. They also provide a premium service that provides a good deal of information on the options of many stocks. Option data on Microsoft stock was provided for free as an example of the premiums service and therefore should be available to anyone who like to reproduce any of the research provided herein.

The first step in working with the Quandl data was to write into their API. Quandl provides documentation and an ID for anyone requesting it, and the unique identifier is entered from the Python command line (and thus not included in the Jupyter notebook). Once this step was completed the stock data could be pulled into a notebook by way of a csv file. After converting the csv into a pandas dataframe, I checked the newly created dataframe for NaN values. After a quick check revealed no NaN values I was ready to move on to the option data.

In working with the option data it quickly became obvious that there were a number of NaN values and a decision had to be made about how to handle them. Since I knew I planned to focus on shorter dated measures of volatility, I immediately dropped some of the longer dated measures (over 60 days) to see if this relieved the issue. Since I still showed quite a bit of invalid data, I next attempted to drop all the columns containing NaN's, but since this would have left only 6 remaining columns from the original 55, I deemed this solution unacceptable. I conducted a little more exploration of the dataset, which revealed that 3 records (rows) were causing all the NaN values. Since I was already looking for a way to limit my data set which ranged from 1986 through 2018, and the records in question were all from before 2006, I just chose to limit my exploration to the years of 2013 through 2017. This eliminated all of the NaN related issues.

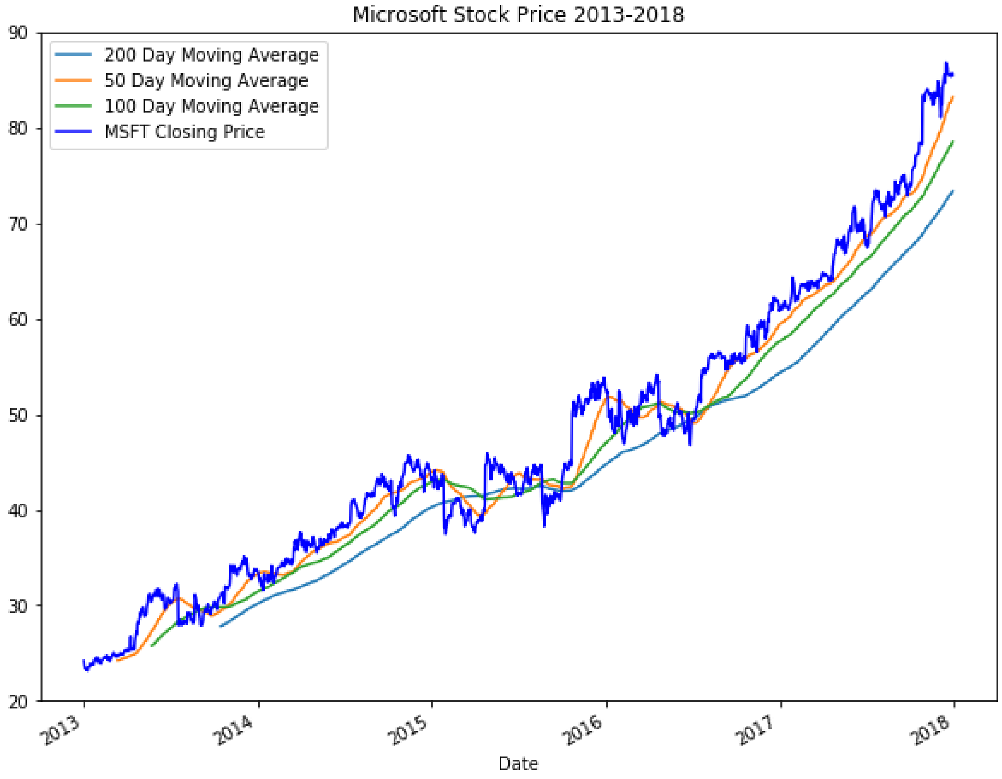
I also noticed a small issue when plotting the price of the stock. The closing price did not include stock splits, so when graphed, it appeared that the stock had a number of huge drops in price. After spending a few minutes looking over the split dates and how they were handled, I realized that the data set included an “Adj Close” column that already adjusted for the split price (as well as high low and volume issues). I resolved to just use the “Adj” columns going forward. I then used a relatively simple join to create a single dataframe that included both stock and option data. I pickled that dataframe for later use, and was ready to move forward to the next phase of the project.

Data Exploration and Storytelling

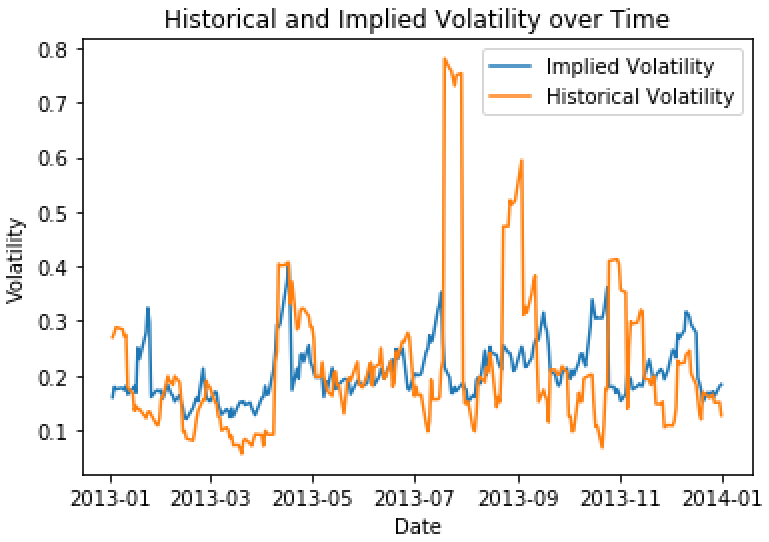
The next step upon completion of the data cleaning and wrangling was to begin to explore the dataset through visualisations. I hoped to find noticeable patterns in the historical volatility of

Microsoft stock that could be concisely visualized. I also hoped to find out more about the relationship of historical volatility to other features within the dataset.

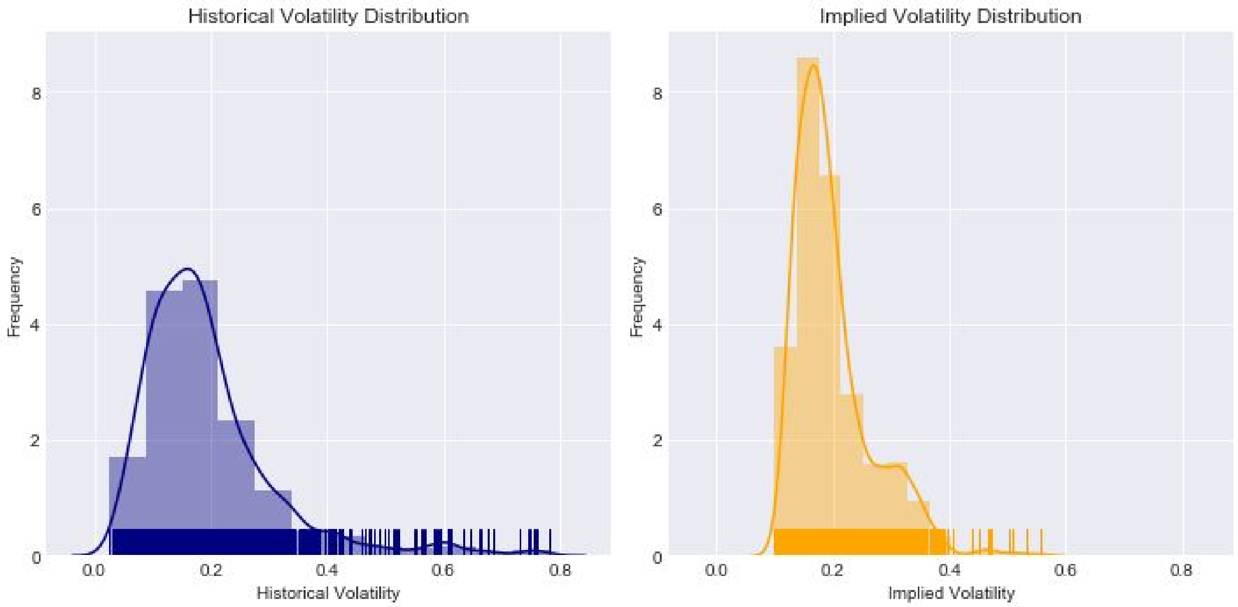
I began this journey by unpacking the previously pickled data, and making sure it appeared to be in order. After this step was completed I began to visually explore. I began by plotting the stock's closing price as well as its range for a fixed period (March 2016) to get and idea of the daily ranges of the stock. With this completed, I moved on to building a chart of Microsoft's daily closing price for the period being examined. I also included the 50 day, 100 day and 200 day moving averages of the stock price, knowing that these 'lag variables' are often used in technical analysis and can be used to identify patterns in price. In particular crossing moving averages often indicate a change in the stocks current trend.



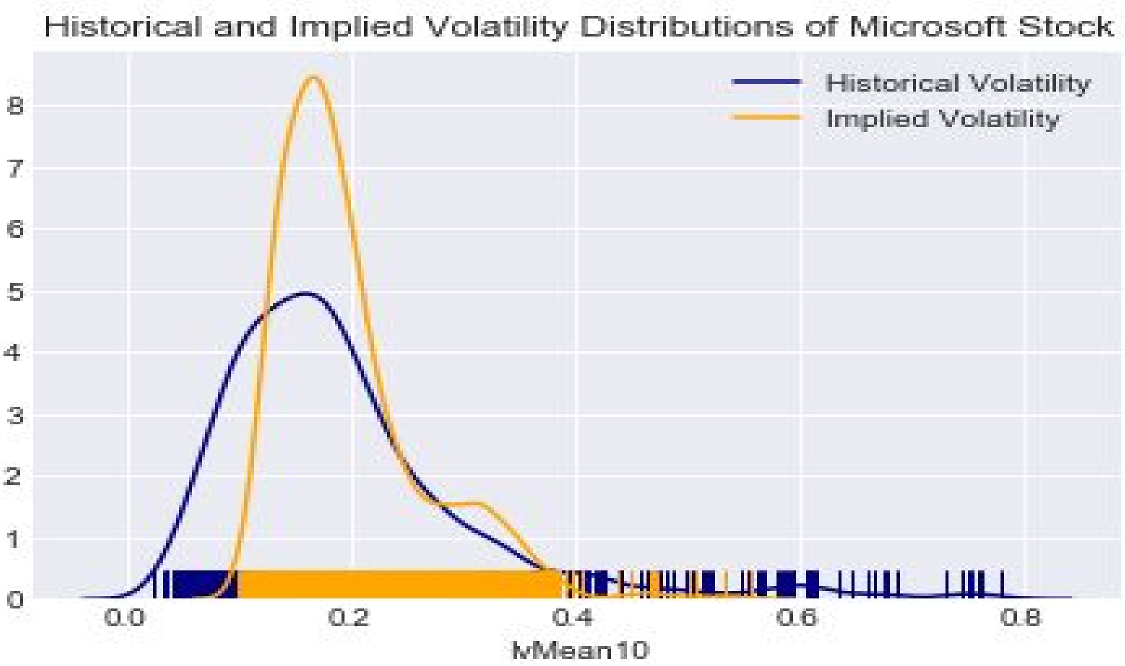
After examining the chart of the stocks price, I decided that an exploration of the Microsoft's historical volatility, specifically in respect to the implied volatility of the options was in order. After initially charting the entire period I decided to focus on a more condensed period. The next figure shows both the historical and implied volatility in the year 2013. I could be seen that at several times throughout the year historical volatility spiked well above the implied volatility.



This pattern of historical volatility jumping to levels well above those indicated by option prices deserved additional analysis. The next step was to plot the distributions of each type of volatility via histogram.



In this diagram it becomes obvious that not only does historical volatility tend to spike above the implied volatility in tail events, it also spends a reasonable amount below the implied volatility. While each time of volatility has a similar mean, the shape of the distribution curves are very different. When drawn on top of one another this relationship becomes even more obvious.

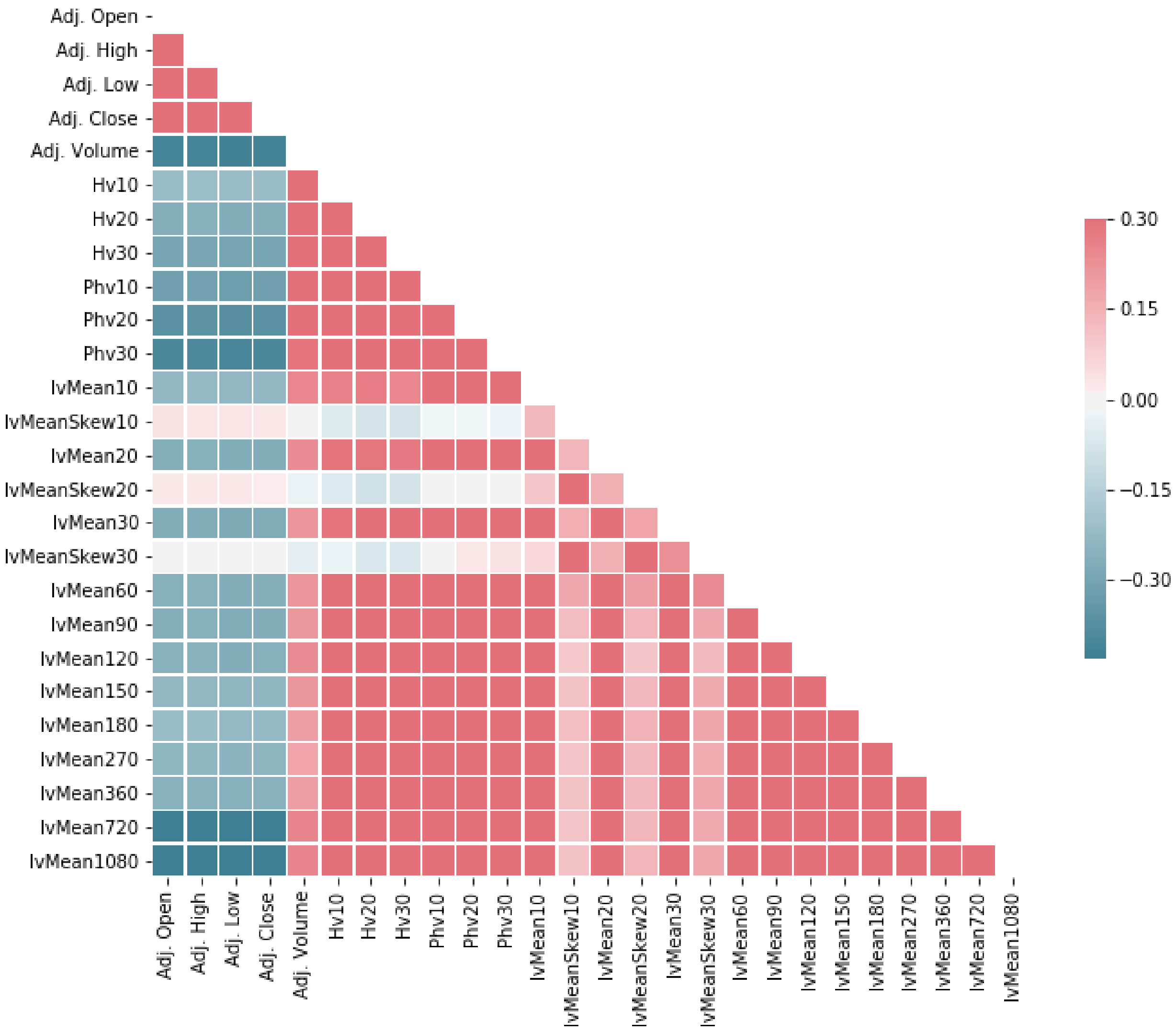


These visualizations helped to provide insight into patterns in Microsoft stock. Both trends and seasonal patterns were discovered in the closing prices of the stocks. Also a number of observations about the relationship between historical and implied volatility were made. With these realizations in mind it was time to begin a deeper dive into these statistical patterns.

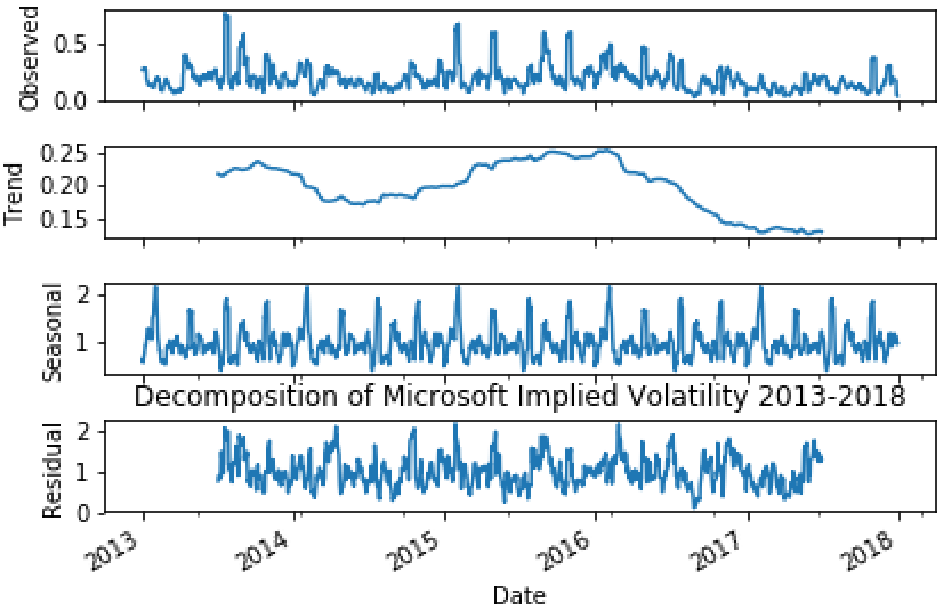
Applications of Inferential Statistics

Once the data had been explored visually, I began to perform statistical analysis on historical volatility of the stock, as well as how the historical volatility related to the other features contained in the data set. I was interested in knowing how each of the features might relate to one another, so I used a heatmap to visualize the Pearson's correlation coefficients of each one (pictured on next page). Unsurprisingly the historical volatilities showed the most correlation to historical volatilities of other lengths, and to Parkinson volatilities (another measure of historical volatility that uses highs and lows instead of just closing prices). There was also a positive correlation to volume (more of the stock traded in volatile periods) and skew (a measure of increased implied volatility on out of the money options). Higher prices, on the other hand seemed to indicate lower volatility.

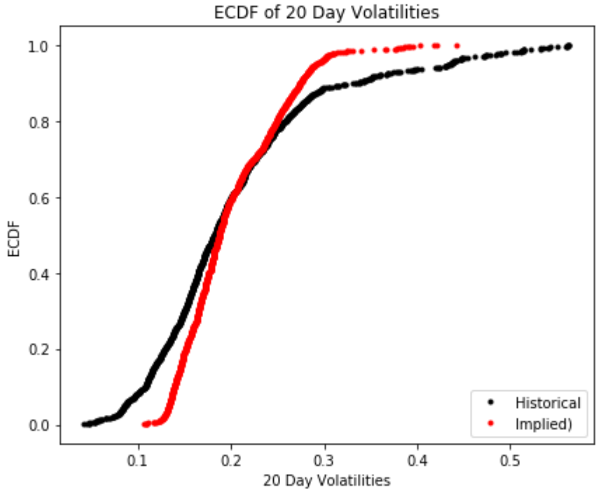
.

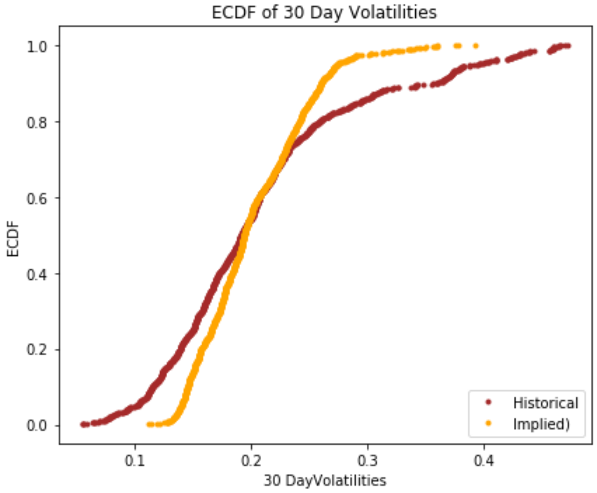


In the data visualization notebook I used the statsmodel seasonal decomposition function to examine Microsoft stock prices (not pictured). I once again applied this function, but this time to historical volatility. I found that the seasonal trend was very much inverse to that of the stock price. Low volatility could be found late in the year, a time when the stock traditionally rallies, and volatility increased in January, a period when prices often moved lower. While the decomposition also identified some semblance of a trend in volatility it was nowhere near as clear as the trend in the stock's price.

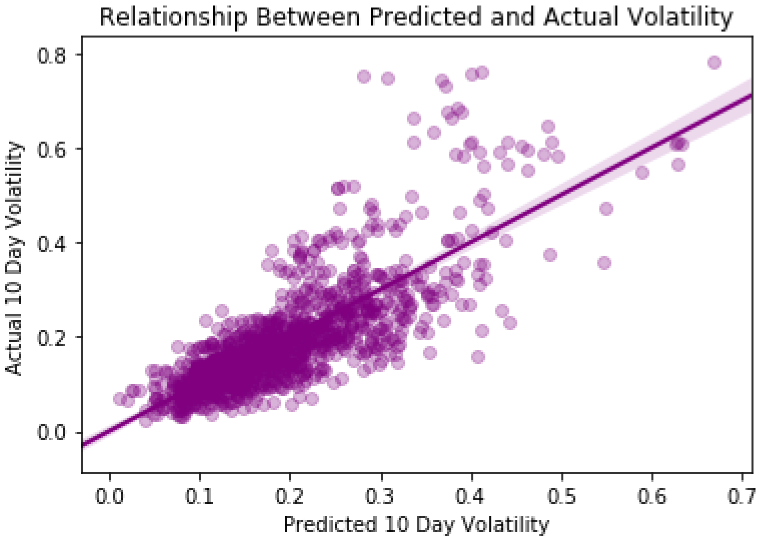


In an an attempt to further study the relationship between implied and historical volatility I ran empirical cumulative distribution functions (ECDF) on both.The results showed differences between implied and historical volatilities increased when looking at longer periods of time. This implies that the option markets assume historical volatility will revert to its mean over time.





I then used statsmodel's ols function to further breakdown the relationship between implied and historical volatility. I found an adjusted r-squared score of 68 % meaning that a reasonable amount of variance in historical volatility of Microsoft stock could be explained by a model using implied volatility of at-the-money options as a predictor. The t-score (P>|t|) of zero (likely just a very small number) indicated that there was a statistically significant relationship between the two variables. I also used a regression plot better visualize the relationship between those two variables.



The regression plot showed that when actual volatility went above 40% the model had difficulty predicting the actual volatility. Conversely there are are a reasonable amount of predictions in the 30-40% range that are higher than the actual volatility.

Next Steps

Throughout the first stages of this project I was able to pinpoint seasonality and trends within Microsoft's stock prices and its volatility. Quantifiable relationships between implied volatility and several others features were proven and further explored. The next step will be to determine if historical volatility can be predicted. First using simple time series analysis and then by adding features I will attempt to use machine learning algorithms to build a model that predicts historical volatility in a useful way.