Nicole Lind

12/15/2017

**Predicting Positive Returns for the Student Managed Investment Fund:**

**The Question/Problem:**

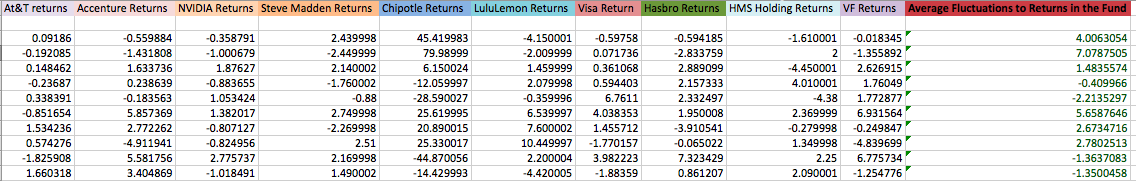
* This semester I was a part of a class that taught students how to analyze companies and invest money in the stock and bond markets. The end-goal of this two-part class is that this spring we will actually create a hedge fund of all of our stocks and attempt to beat the market (S&P 500). We learned how to identify an alpha and value stocks from a quantitative standpoint with relative valuations, geometric averages, fundamental analysis, and absolute valuations. However, our reports all had to include qualitative information about the company that “told a story” of why we think it was a good company to invest in. This was the hardest part of the report for most of us because it ended up being very subjective. I wanted to find a way to better predict which qualitative factors most influenced a positive return for the fund as a whole. That is how I landed on this project idea.
* How do you quantify a company’s corporate governance? Management is a crucial part of any successful corporation but there are a ton of factors that relate to “good” or “bad” management and they are often subjective by nature. Certain management styles work great for some companies and fail miserably with others. Factors like innovation, interest in research, community involvement, an inspirational CEO, and overall strength of mission/vision statement are all-important factors that can also lead to a positive stock return.
* Another thing I wanted to test was Fischer’s 15 points. He brings up great things to look for in a successful company that will in theory bring positive returns. I wanted to test whether or not these factors actually produced such positive results in our fund.

**The Data Collection Phase:**

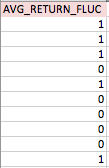
* The way our fund is set up is that everyone in the class picks a stock in the beginning of the semester that meets certain thresholds set by the fund. Once these stocks are chosen, each student is responsible for a 50-100 page report filled with calculations of valuations at different discount rates, P/E relative valuations against competitors, a fundamental analysis relative to competitors, external analysis, internal analysis, information from the annual report, yearly hypothetical projections based on historical data, and tons of qualitative analysis regarding press releases and any competitive advantages. Each student essentially becomes an expert on his or her particular stock.
* For this reason I thought it was a good idea to perform primary research and collect data from my peers. I gave each person in the fund a survey to fill out about his or her stock. The survey included qualitative and quantitative information that is not easily correlated to positive or negative returns. The survey also included scales for how well they think their stock fit the criteria for Phillip Fischer’s 15 points. Everyone was willing to fill out the survey and there were no missing values.
* Problems with my data include: I should have collected more data because my excel sheet is very wide and not very long, which often times leads to over fitting. Also, It would have been better to make the surveys shorter as people complained about the length.

**Coding/Cleaning up the Data:**

* The data was coded by analyst with the names kept in the excel file but deleted from the data frame in the python code.
* The returns were obtained from yahoo finance. I typed in each stock ticker and then went to the historical data tab and filtered it by 5 years, monthly returns. Some of the stock in the fund did not have historical data past 2014, so I ended up using stock prices from May 2014 to December 2017.
* Then for each of the stocks in the fund, the adjusted stock price was taken for that time period and calculated into monthly returns. I took the arithmetic average of all of the stock returns for the fund to find the fund’s average monthly return, assuming we invested the same amount of shares for each stock in the fund. This amount of data was too long of a column so it was necessary to average every three together and use the average 3-month returns for the fund. Below is a photo of how I calculated the average returns.



* Once the average 3-month returns for the fund were calculated, they were placed into two groups 1 was a positive return and 0 was negative return. This is how they are coded in my excel sheet.



* The explanatory variables include: Corporate governance, niche company, community involvement, financial slack, research and innovation focus, pace of growth, number of CEO awards in last year, # of press releases in past 3 months, perceived risk of the company, # of close competitors, strength of mission/vision statement, # of countries the company operates in, and a question for each of Phillip Fischer’s 15 points.
* The response variable: positive stock return for the fund
* As a result of doing primary research I had the liberty of coding the data how I wanted and placed my returns as the first column, since that is my response variable

**Importation and First Look:**

* I imported the data using the panda’s pd.read function that works for csv flies.
* Then, to make sure that all of the data was imported I created a print statement underneath to verify.
* The dataset imported very nicely.

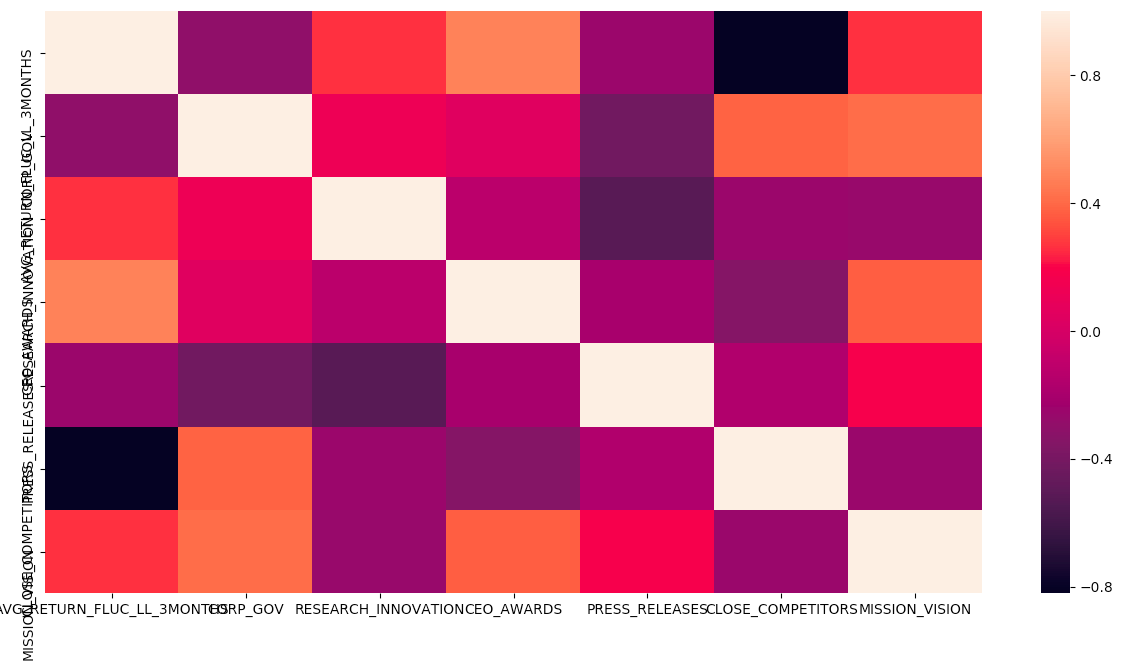
**Transformation and Cleaning up the Data:**

* To clean up the dataset and make sure my code ran I had to do a couple of transformations. The first change was to do a one-hot encoding for the non-numeric values to turn them into zeros and ones. To accomplish this I needed to use Panda’s get\_dummies method, which converts categorical variables into dummy variables and the cat\_features method gives you a list of categorical features for the dataframe. This deals nicely with our categorical variables so we can use them cleanly when running our analysis.
* Then I removed the response variable, which was average stock return for a period of 3 months, from the dataset.
* This way all other features called “returns” can be used as explanatory variables and labeled as data\_x.
* The response variable is now outside of the returns features and can be labeled data\_y.
* Then to make sure the data was coded and transformed correctly I put a print statement. Everything seemed fine.
* Important Note: I created a training set that was false values and only used average 3-month stock returns for the same period for Lulu Lemon’s stock. This was the data that I trained on.

**Data Visualization:**

* I used a seaborne heat map to visualize the data and infer potential correlations. I only used a map for the variables that I had a hunch would highly correlate to a positive stock return for our fund. This is a really cool way to visualize the data.





**Evaluate Different Modeling Approaches:**

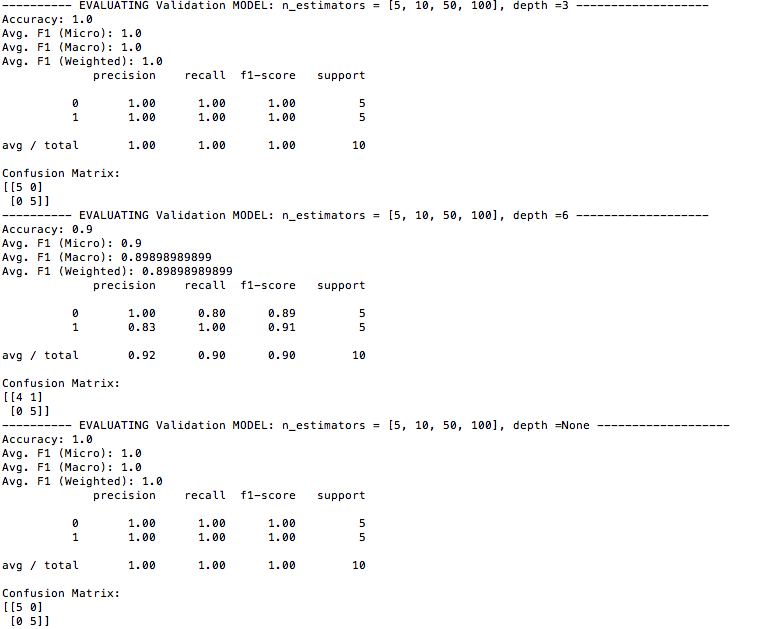
* This was the hardest part of the project because the data did not fit well into any of the models. First, the Lasso regression gave negative R^2 values, which means the data was not linear. After this realization, some alteration to the data coding was needed (I changed the returns from values to positive or negative and then to a simple 0 and 1).
* With the new data, a support vector machine was run but it did not perform well with an accuracy of .333, Avg. F1 Micro of .333, Avg. F1 macro of .25, Avg. F1 weighted of .167, average precision of .11, average recall of .33,and average f1-score of 17. The confusion matrix was weighted very heavy on the right side meaning that it was not a very accurate model.
* These are bad results because accuracy predicts the total % correctly classified by calculating the total correct predictions divided by the total number of predictions. Therefore, an accuracy of .333 is very low.
* Accuracy is the not the whole picture, so we need to look at precision. Precision reveals the percent predicted positive that are in fact positive. This is calculated by taking the amount of true positives and dividing them by the sum of the true positives and false positives. An average precision of .11 is horrible because it entails that the model is not predicting values correctly.
* Recall is the percent predicted positive out of all the positives and is calculated by taking the true positive and dividing it by the summation of the true positives and the false negatives. The average recall for the first model was .33 so it missed a lot of true positives. The F1-score combines the precision and recall into one measure and was also low.
* This means that our data is not a good fit for this type of classification method so we need to choose a different one.
* The next model that was run was the decision tree model. With entropy criterion there was an accuracy of .333, avg. F1–micro score of .333, av. F1-macro of .25, avg. F1 weighted of .166, precision of .11, recall of .33, and f1-score of .17. These again were very bad error metric scores.
* Then I ran a random forest classifier and it came up with some acceptable results. The accuracy score was .667, the avg. F1-micro was .667, the avg. F1-macro was .667, the avg. F1-weighted score was .667, the precision was .83, and the recall was .67. The confusion matrix was okay but not great. This is the model I chose to use.
* An important note is that our models are only going to perform as well as the data given to them and since my data was taken from such a small sample, all models are not going to give us great results.

**Model Ensemble (Optimize):**

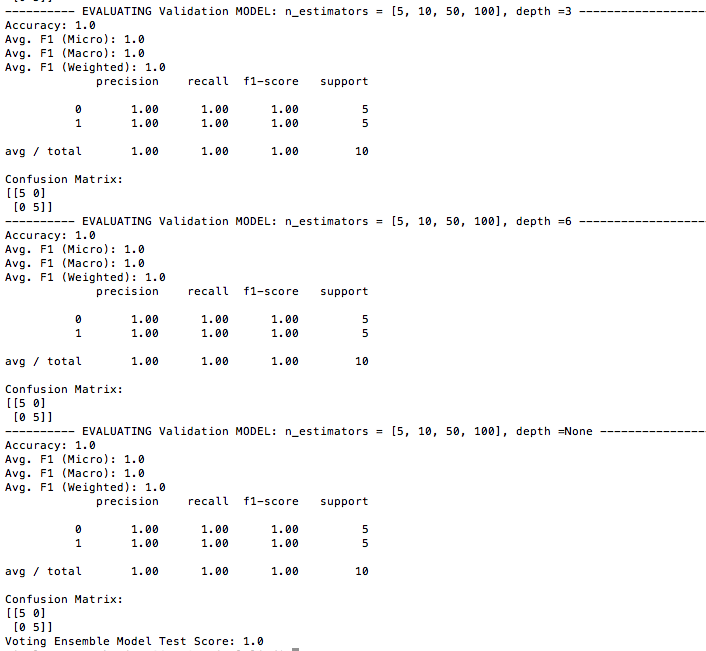
* To optimize our model and get the best results we usually use model ensembling. This is a method of combining multiple different models to offset the problems with any one model. Often time’s models that give good results alone are then combined to create an even better model.
* I used a voting classifier with a grid search to optimize my model. It came up with a panda’s error because my sample size was too small to split the data effectively but it still ran and gave me results. The results for my training data were definitely a product of over fitting because my error metrics were almost all 1’s.

**Validation Without optimization:**

* When I ran the classification on my validation set it came up with great measures, which trended towards over fitting. Below are the error metrics for the random forest classifier model alone on the validation set.



**Validation with Optimization:**



* Above is a picture of the results when I ran the optimized model using a voting classifier with CV and grid search. All of the error metrics were one and there was a perfect confusion matrix. This means it predicted the actual values perfectly and probably was a result of over fitting. I think this occurred because my sample was too small.

**Lesson learned:**

* I did not do enough design and planning and chose too little data to really be able to answer my research question. Next time I will take more time to plan out where I am getting my data from and how it will fit in a model. This project has made me realize that if you put bad data in then you will receive bad results, regardless of how well your model was coded.