Investment and Trading Predictive Model Proposal

Recommend Stock Tickers

Kelsey Odenthal
May 31st, 2017

An important aspect of stock trading and investment is being able to predict, with historic stock price and performance data, the immediate future of trading. By applying a supervised learning algorithm to the data that is gathered, feasible stock options can be presented and pursued by the user. It's important to understand that there are many more underlying factors to stock trading beyond historical data and company performance, which is somewhat difficult to account for, but overall this model should give some approximately valid options for investments.

Domain Background

Stock predictions have been a staple of investment trading even prior to machine learning. There are careers to be made from crunching numbers and evaluating the path of a particular stock, those who study behavior finance know to not fight momentum, others believe that by averaging out a falling and climbing stock leads to an understand of what path it may be on, and other similar tactics (http://www.investopedia.com/articles/07/mean_reversion_martingale.asp). My original motivation in pursuing this project was to understand bitcoin markets and make a predictive model based on the available data, but unfortunately finding available data was a lot more difficult than I thought it would be (a project to pursue in the future) and instead I decided to understand overall stock forecasting.

Problem Statement

Now, we have machines who can compute these predictions for us, learning from massive amounts of data that would take us years to sort through, and all within a matter of minutes. Stocks change quickly and unless someone hires a stockbroker or studies extensively on their own, the average layman could not hope to jump into the race. Why not have a program that could guide you through financial forecasts? This is solely a classification problem, as comparisons will be run between individual years and will judge whether a stock improved between the allotted time or not. The model would predict the tickers of stocks that are likely to continue to perform well. The feature we will use to gain the insight needed for performance will be based on the "Adjusted Close" of the stock from two sets of data, Quandl and Yahoo! Finance.

Datasets and Inputs

The majority of data pulled for this project is to be scraped from Yahoo! Finance with some additional data pulled from Quandl between the years 2000 and 2013 with approximately 200 rows. The focus from these two groups will be stocks from S&P 500. The data from Yahoo! Includes information on Ticker Symbols, Debt/Equity Ratio, Trailing Price to Earning Ratio, Profit Margin, Market Cap, Enterprise Value, Gross Profit, Total Cash Per Share, Total Debt, and many more factors.

The Quandl dataset includes Open, High, Low, Close, Ex-Dividend, Adjusted Open, and more. Given the context of the problem, this data would be very appropriate and extensive to interpret for predicting. Yahoo! Is far more extensive but Quandl allows for easy further exploration of the stocks for the average users.

Solution Statement

The solution to interpreting, understanding, and making predictions on the stock data is to accumulate a large amount of stock data and do a compare and contrast between one particular year and it’s predecessor. I do not consider myself an expert on stock interpretation but my solution is to see how the price, debt, and more change year-by-year. The intent is to make a prediction on what stocks will continue to have an upward momentum of success. Using a classification model to determine those in the S&P 500 that perform well and those that are not.

Benchmark Model

The model I will pursue is to sort by the Adjusted Close of S&P 500 for the current year and the last, stock price for the current year and the last, and evaluate the difference between stock change and S&P 500 change. Further, the data will be tested with these comparisons to see if they underperform or overperform in their own respective timeline. Using a Naive Bayes Model as a baseline example for a simpler predictive model to evaluate and score initially.

Evaluation Metrics

I will go through a process of selecting a classification method based on the “score accuracy” in order to deduce what the best model is for this particular problem. The higher the accuracy score, the more plausible that model is, and the program will focus on that output.

Project Design

A theoretical workflow for approaching this solution is to scrape for data, perform a binary classification based on whether a stock has underperformed or outperformed its previous year Adjusted Close with all of the accumulated data specifically within the S&P 500, and assembling a list of potential stock candidates for investment, based on the learning algorithms that has the highest accuracy. The classified (underperform, outperform) data will be utilized for the train test split. I will run evaluations on mainly Decision Trees, Random Forests, and Support Vector Machines to decide on the predictive model.