JONATHAN K SULLIVAN

http://i.dailymail.co.uk/i/pix/2012/03/26/article-2120416-125687D3000005DC-576_468x351.jpg

Capstone Project Proposal

Prepared for: Official Udacity Reviewer

Prepared by: Jonathan K Sullivan, Machine Learning Engineer Nanodegree Student

November 10, 2016

Proposal number: 1

JONATHAN K SULLIVAN

DOMAIN BACKGROUND

The domain background for this project is finance, the study of investments. More specifically we will focus corporate finance and stock prices. Today the affect of the stock markets are ubiquitous. Even If you are not a stock trader or a Company, the markets affect your retirement plans, your favorite businesses, or even that cool new product you want. Stocks or shares in equity while important are an old concept. Matter of fact the ancient Romans “issued shares called partes (for large cooperatives) and particulae which were small shares that acted like today's over-the-counter shares.”(https://en.wikipedia.org/wiki/Stock) These predecessor of the stock just like there modern counterpart had fluctuating prices. There is also evidence from the 1200’s that show shares where traded for milling and mining company in Sweden and France.

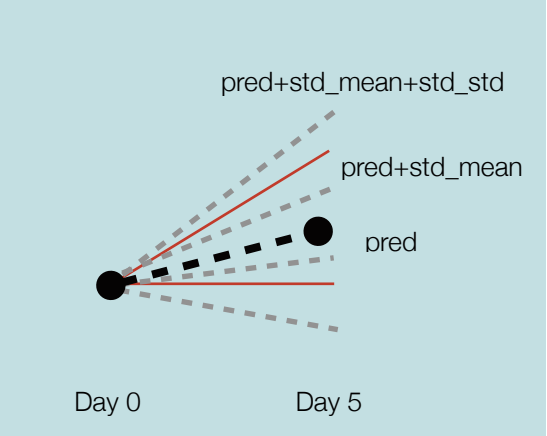
“The earliest recognized joint-stock company in modern times was the English East India Company, one of the most famous joint-stock companies. Soon afterwards, in 1602, the Dutch East India Company issued the first shares that were made tradeable on the Amsterdam Stock Exchange, an invention that enhanced the ability of joint-stock companies to attract capital from investors as they now easily could dispose of their shares”(https://en.wikipedia.org/wiki/Stock).The Amsterdam Stock Exchange is the predecessors Euronext Amsterdam, which functions similar to the New York Stock Exchange or Nasdaq Stock Exchange. Today these exchange host an unimaginable amount of trades 252 days a year.

In the Stock market fortunes can be made or loss. One of the most important factors when it come to making money using technical analysis on the market is the price of a company’s stock. If there were more innovative tools that hedge funds could use to help them make decisions in the market, then not only will it decrease volatility and better stabilize the market but lead to higher returns into the future.

PROBLEM STATEMENT

As more people use the same strategies in the market the the less effective they become. Similar to a new Knife if one person uses it on a day to day basis it will stay sharp longer than 100 or a 1000 people using it on a day to day basis. The problem here is we need a new knife. The proposed solution to this problem is to build a stock price predictor using supervised regression learning. That takes daily trading data and statistics over a certain date range as input and outputs projected estimates of adjusted close for given query dates. This estimate should include not only a price but also the uncertainty. I believe this estimate of the price can best be accomplished through a artificial neural network who’s input are the results of a an ensemble of regression learners. The input to the regression learners will be principle component of original data and calculated normalized indicator such as simple rolling average, and Bollinger bands ®. I also believe that the best measurement of uncertainty for the

predicted prices for a n-day forecast would be the mean of the standard deviation of the all possible consecutive n-day historical prices in the training data. By looking at uncertainty this way I believe we are saying the probability of company's instantaneous volatility being x can be modeled as the a Gaussian distribution do to the central limit theorem. When we combine this with the fact that the markets are normally less volatile at any point than they were in the past we can use this method to create a conservative range of most probable adjusted close ranges with a certain confidence. For example if we do a 5 day forecast we look at every 5 day period in our training data. We measure the volatility of each on of these periods. We take the mean and standard deviation of these measures and use them to construct our confidence intervals.



The probability of true price is within inner dotted grey lines is .16. The probability of true price is within solid red lines is .5. The probability of true price is within outer dotted grey lines is 0.84.

DATASETS AND INPUTS

There are several open sources for historical stock price data which could be chosen for this project. They are
Yahoo! Finance, Bloomberg API, and Quandl. After weighting the pros and cons of each api and I choose the Quandl. I found it most robust one for our application because of it's large collection of data available for free in virtually any format. In addition it has free and unlimited use and a python api.

The input of the data will contain the following daily stock values such as opening price (Open), highest price the stock traded at (High), Lowest price the stock traded at (Low), how many stocks were traded (Volume), closing price(Close) and closing price adjusted for stock splits and dividends (Adjusted Close). The Adjusted Close is our label or what we are trying to predict. We will be predicting the adjusted close so that stock splits and dividends do not confuse our model. In the adjusted close these to problems are not factors and the discontinuity that are present in the close feature are absent in the adjusted close feature. Where as the Open, High, Low, Volume, Close are our features. I will also explore using statistics in the feature space such as normalized momentum, normalized simple rolling average, and normalized Bollinger bands ®. The specific equities we will look at are as follows:

1. Apple (APPL)
2. Alphabet (GOOG)
3. Microsoft (MSFT)
4. Amazon (AMZN)
5. Facebook (FB)

6. Exxon Mobile (XOM)

SOLUTION STATEMENT

The solution to this problem is to use a machine learning technique, more specifically a artificial neural network where the inputs are outputs to a ensemble of parameterized and instance-based supervised regression learners. We will create a training interface that accepts a date range (start_date, end_date) and a list of ticker symbols (e.g. GOOG, AAPL), and builds a model of stock behavior. Secondly we will add a query interface that accepts a list of dates and a list of ticker symbols, and outputs the predicted stock prices for each of those stocks on the given dates.

BENCHMARK MODEL

“The most common form of ANN in use for stock market prediction is the feed forward network utilizing the backward propagation of errors algorithm to update the network weights. These networks are commonly referred to as Back-propagation networks. Another form of ANN that is more appropriate for stock prediction is the time recurrent neural network (RNN) or time delay neural network (TDNN).” (https://en.wikipedia.org/wiki/Stock_market_prediction)

As we mentioned earlier as a method becomes more common it become less useful because of so many intelligent actors. Even though I am unable to peek into Blackstone hedge fund apis and view their models, we can objectively measure the model that we create and compare it to the performance in the current market. I believe this will be a better metric of success because if it works well then we can be semi-certain that it is better than the most current models in use. I plan to use market performance as the benchmark. I will pick different close stock prices for different tickers and measure if they are within x% on average.

SET OF EVALUATION METRICS

I will evaluate the performance of our algorithm in two different manners. To ensure that the regressor is a good fit for our data I will use the coefficient R^2 and mean squared error. To prevent overfitting we will focus on improving the R^2 and mean squared error value within our testing set. However we will construct R^2 and mean squared error values for our training set just as a sanity check to make sure our algorithm has the potential to perform well. We will also employ rolling cross validation and grid search to tune our parameters and hyperparameters and improve performance.

OUTLINE OF THE PROJECT DESIGN

The first thing that I will focus on is building an interface that will accepts a list of symbol and returns a list of panda data frame with no-Nan Values. Second next I will use this data to compute momentum, simple rolling average, and Bollinger bands. Once we have good data and statistics we can normalize these values and place records in chronological order. Next I will set the size of the testing set and the training set. Since peeking into the future is not allowed we will make sure our testing set is always after the testing set. Also we will segment the data so that our features and labels are in their own separate data structures. Next we will perform principal component analysis on this data to reduce the dimensionality.

Next I will construct a assemble of learners using a approach called bagging. One learner will be composed of a k-means regressor and the other will be a polynomial regressor. The output of these regressors will be fed in to a artificial neural network where using backpropogation. This algorithm will be trained by starting at the some number of days, d, before the start training date stoping at the stop training date. We will construct a datapoint where d days before the start date will represent out input and the adjusted close on the day after the start date is out label. We will continue to iterate through this data until we get to the last day in the training set. We will then make predictions with the test set and measure our accuracy. We will make necessary changes and evaluate our model to see how we can improve it.

Also if we are making an n-day prediction we will look at all possible n-day sequence measure their standard deviation. Then make a note of the mean of standard deviations and standard deviation of the standard deviations. We can use these values to make a good measure of risk.