Machine Learning Engineer Nanodegree

Capstone Proposal

Yee Jie Thay

19 May 2018

Domain Background

The main aim of the project is to create a tool in facilitating indivudals to get better entry points for trades and know when will be a good time to realise profits/ losses of trades so as to maximise returns on investments.

Utilising daily trading information avaliable, we have a rich set of time series data which can be used to understand behavioural patterns of investors. Such behavioural patterns such as bouts of euphoria or signs of panics will allow investors to better understand the environment they are deploying their capital in.

Below are several pieces of academic paper that has explored this concept from various different angles.

http://revistaie.ase.ro/content/57/16%20-%20Moldovan,%20Moca,%20Nitchi.pdf (http://revistaie.ase.ro/content/57/16%20-%20Moldovan,%20Moca,%20Nitchi.pdf)

https://www.worldscientific.com/doi/abs/10.1142/S0219024906003512 (https://www.worldscientific.com/doi/abs/10.1142/S0219024906003512)

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1344230 (https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1344230)

Problem Statement

For this project, I will look to build a stock price predictor that takes daily trading data over a certain date range as input, and outputs buy/sell signals for given query dates.

The problem that investors often face with a stock is that thay know which companies are the industry leaders and who the winners are. However, investors do not know the opportune time to invest in it such as to maximise their profits. This results a significant reduction returns. This is especially so for stock whose prices that exhibits high volatility or for investors with a shorter time horizon.

One has to be mindful that different shares exhibit different behaviour. To reduce the universe of shares to investigate in this project, we will aim to have a model that specifically looks at the behaviour on the components of Dow Jones Technology (DJUSTC) Index.

https://www.investing.com/indices/dj-technology (https://www.investing.com/indices/dj-technology)

I will approach the problem as a classification problem such that we get binary outputs of actions (i.e. buy or sell). The binary action will be decided by a logistic function defined by a threshold of 0.5 (will adjust threshold for the area under receiver operating characteristic curve calculations)

Datasets and Inputs

I plan to use a magnitude of data coming from several sources.

1) Bloomberg API (https://www.bloomberg.com/professional/support/api-library/)

   a) I plan to get several different types of price information such as

      1) Opening Price
      2) Closing Price
      3) Volume traded that day
      4) Intraday high of price
      5) Intraday low of price

   b) I also plan to get data such as

      1) Earnings Reporting Date
      2) Earnings
      3) Sales
      4) Expenses
      5) Net Income
      6) Dividends

   so as to be able to get align them with the share price moves and enrich our price data.

   c) I will look to classify all the shares into one of the following 9 categories. However, given the magnitude of the data, it might be wise to focus on a single sector. The sector chosen will be the Technology sector and the index that we will be comparing to will be Dow Jones Technology (DJUSTC) which has 131 components and will form the basis of our analysis. This means that the current scope of data will focus solely on Technology stocks with an eye to the future of expanding it out to other sectors such as

      1) Basic Materials
      2) Consumer Goods
      3) Consumer Services
      4) Healthcare
      5) Financials
      6) Industrials
      7) Oil and Gas
      8) Utilities

2) Twitter

   a) This mainly involves getting tweets which has the ticket name in it so as use it as proxy of the sentiments associated with the stock. This is done by running it through a word processor so as to get the words with the highest frequency associated with the word as a proxy of the sentiment of the stock. I look to have it digest down to 2-3 words and a variable with positive, negative or neutral will be assigned to it.

This will give us 13 variables on each stock at each point (although last variable will be the same i.e. Technology). Thus we will be looking at a total of 131 * 12 = 1572 data points. If we use the data for 500 trading days (approximately 2 years worth of data), we will be looking at a data set that has 786,0000 data points which will give us a sufficiently large data set for various analysis.

I will look at data split of 70:30 split where the initial 70% of the data for training/cross validation and the lat 30% for testing and reporting of the performance of my model. Given that this is a time series data, I will be sorting the data by chronological order and taking the first 70% for training/ cross-validation and next 30% for

Solution Statement

The solution to the problem is such that we perform better than just random guessing. If a buy signal is generated, we will look at the 30 days return of the share against our benchmark index. If the return is positive, we will classify it as a true positive. If the return is negative, we will classify it as a false positive. Similarly, if a sell signal is generated, we will use a similar method to classify true negatives adn false negatives. The solution will be defined as the area under the receiver operating characteristic curve (ROC curve) between the predicted probability and observed target.

Benchmark Model

Most benchmark models that we can use for direct comparison are proprietary and hence will not be easy for us to use as reference for performance benchmark. However, the benchmark for the financial industry has always been indices. Thus, a good reference benchmark model will be the returns achieved by various indices over the course of 30 days.

We will eventually classify the shares into 9 different sectors which has 9 different benchmark indices from the Dow Jones Index family. But currently, we will specifically look at the Technology index so as to make sure that we are comparing equivalents. Dow Jones Technology (DJUSTC) Index has 131 components to it and it will be the sector we will be applying our model to.

However as a starting point, I will be employing an out-of-the-box random forest model as a benchmark. I hope to be able to outperform the random forest model and will look to perform a comparison against the random forest model to find where my model will be underperforming and thus will require more developmental work.

Eventually I will look to match the returns of achieved by the DJUSTC Index.

Evaluation Metrics

The model will be evaluated on area under the Receiver Operating Characteristic curve (ROC curve) between the predicted probability and the observed target.

We plot the curve by using the model to calculate True Positive Rate (True Positive divided by the sum of False Negative and True Positive)and False Positive Rate (False Positive divided by the sum of False Positive and True Negative) with different thresholds for the logistic regression.

An area under the curve of more than 0.5 would mean that we have performed better than randomly guessing.

Project Design

I envisage the workflow to be broken up to 5 main steps.

1) First and foremost, I look to splice the data set into a consistent set of data for data exploration. This will mainly be carried out in Python and will involve ensuring consistency in the data for interpretation at a later stage. I will look to download the data via the Bloomberg API.

a) However, one has to be mindful of various issues such as different listing dates (e.g. a constituenting company might have been listed for less than the 500 trading days window that we are looking at).

b) Also there are issues such as trading halts etc on the name pending news release or merger and acquisitions that might result in a particular ticker name changing etc.

All of these are important information will have result in unintended skewing of our model and thus introduce bias into the result.

2) This will be the stage of data exploration where I will be spending time to explore the structure of the data. This involves plotting the distributions of various variables that are of intuitive importance as well as looking at the various correlations of the variables in the data.

a) This stage will focus on visualisation of the underlying data set to see if there are any interesting relationship between the various variables. This step will also be incredibly useful in spotting results that are outliers given that it gives us perspective of where the data are with respect to other equivalent data points.

3) Next I will embark on processing the data. This means to refining the data and creating new variables that are normalised or creating data that describe the data better such as ratios. It is also important in cleaning out dirty data as well as removal of outliers that might lead to unintended behaviour in our model.

a) I will look to scale and normalise the data using the Min-Max Scaling but if there are unforeseen implications of it, I will revert to using the traditional method of subtracting the mean and dividing by the standard deviation.

b) The next step of data cleaning involve converting all the data into the most suitable format. This is especially so when we are looking at dates. "20/02/2018" will coming after "10/03/2018" if we sort the data when the date is recognised as a string. This poses a huge issue as we require the data to be in chronological order.

c) Lastly, with regards to outliers, I will look if it impacts our inference or relationship between other data points. This is important as outliers might be leading indicators that can help us to more efficienctly generate signals. If data point is missing, I will look to fill it in with the nearest neighbours or the mean to ensure that we minimise any unintentional skewing of the model.

4) I will look to built the neural network and different layers of it using Python Keras module and train the model. I will specifically set a subset of result

s aside for testing the model and will not use it for any training.

a) I would expect the relationship of various features to be non-linearly separable along the 13 dimentions thus I would expect to use at least a single layer of hidden layer. Tentatively, I will expect to create the first layer of dense layer which will output a layer of 156 variables which I will then use a max pooling layer of 32 filters with a pooling size of 2. This will half the number of variables and thereafter, I will look to add the convulation layer before applying Global Average Pooling as the next layer. To avoid overfitting, we have a layer of dropouts where 20% of the samples are randomly dropped so as to allow the model to be more robust. Lastly, I will have a Dense layer that will output the result of the result.

b) I will also look to use transfer learning. I will look at the creating a model using the first share and then adapt the neural network to a new different data set based off a second share. This is especially useful our case where we have large data set and the result are very similar. https://arxiv.org/pdf/1411.1792.pdf

In [ ]: