**Project Overview:**

The project aims to use the 20 variables over the course of 30 days to predict the return of shares over the course of 30 days. This will allow us to create a tool to facilitate users in getting better entry points for trades. The 20 variables are as follows:

1. 'Book\_Value\_Per\_Share\_\*\_USD',
2. 'Cap\_Spending\_USD\_Mil',
3. 'Earnings\_Per\_Share\_USD',
4. 'Free\_Cash\_Flow\_Per\_Share\_\*\_USD',
5. 'Free\_Cash\_Flow\_USD\_Mil',
6. 'Gross\_Margin\_%',
7. 'Net\_Income\_USD\_Mil',
8. 'Operating\_Cash\_Flow\_USD\_Mil',
9. 'Operating\_Income\_USD\_Mil',
10. 'Operating\_Margin\_%',
11. 'Payout\_Ratio\_%\_\*',
12. 'Revenue\_USD\_Mil',
13. 'Shares\_Mil',
14. 'Working\_Capital\_USD\_Mil',
15. 'close',
16. 'dividend\_amount',
17. 'high', ‘
18. ‘low',
19. 'open',
20. 'volume'

Given that we have chosen 131 tickers which made up the Dow Jones U.S. Technology Index, we expect them to be helpful in explaining the moves of one another. For example, an increase in Apple’s earnings per share will have a predictive impact on Google’s returns.

For our input data, I used 2 data source specifically and spliced it together to come up with the complete data set. The first data set is from AlphaVantage that provides me with the data on the equity prices such as high, low, open, share splits, volume and dividend amount. The second set of data is picked up from Morningstar. The data consist of the key ratios such as earnings per share, operating income, payout ratios, revenue and working capital. The companies report these data on a quarterly basis but Morningstar complies them on annual basis which is useful as it removes the seasonality factor of the data. Furthermore, some of the data by the companies have yet been reported and thus, I decided to forward fill the information i.e. filling forward the information such that the most recent readings will be used for the yet-reported data.

Initially, there was a plan to utilise Twitter data but I failed to properly classify the sentiment used and hence decided that it will be much more useful to utilise clean data rather than having a data point that is flawed. Furthermore, I am fortunate enough to obtain a lot more information from Morningstar and thus decided on removing the Twitter dependency but introduce more variables that I have sourced from Morningstar.

**Problem Statement:**

I will look to build a stock price predictor that takes daily trading data over a certain date range as input and output buy or sell signals for the given query date.

The best proxy for the future is the recent past is the basis of the strategy to solve this problem. I believe that all shares exhibit similar behaviours which allows us to utilise what we have learned from the share price movements in Apple to predict the future movements of Facebook or Google. On this basis it will be incredibly useful for each factor to learn from one another and thus help us to maximise the dataset we have. Furthermore, a rise in Apple revenue also could drive other related companies like Micron or Intel and thus, we can utilise such cross company information to help refine the expected return of the shares.

One has to note that a 30 day return model is used as well and this is fundamental because it helps us avoid volatility which is intrinsic in the market. The expected solution will be one where we provide a complete set of information of all the related data and will be able to confidently tell us if the share price of our target company will rise or fall. This expected solution will be a dense neural network comprising of 3 layers. Between each of these layers, I look to drop out 20% of the point so as to ensure that we do not overfit the model to the data we have. It is extremely important to do that given that a lot of the information we have are correlated and thus resulting in us failing to see the wood for the trees.

**Metrics:**

Originally the proposed metric to use the area under the Receiver Operating Characteristic Curve where the curve is plotted by using the 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. However, this was scrapped after I ran the model once as I realised that 30 days return is often highly skewed. For example, the last 30 values of the 30 days return for AMD is all positive and there is only one class present in the y true value making it impossible for us to calculate True Positive and False Positive Rates and thus unable to utilise the ROC AUC score. Thus I decided to change to utilise accuracy score which is a simple singular metrics to use which make the model output and that of the benchmark model easily comparable.

<https://stackoverflow.com/questions/332289/how-do-you-change-the-size-of-figures-drawn-with-matplotlib>

<https://stackoverflow.com/questions/18770504/resize-ipython-notebook-output-window>

<https://www.somebits.com/~nelson/pandas-multiindex-slice-demo.html>

<https://www.shanelynn.ie/select-pandas-dataframe-rows-and-columns-using-iloc-loc-and-ix/>