# 2. Question 2

## 2.1 Preparing the dataset (Dimension Reduction)

The original dataset consists of 4392 observations and 224 variables. In the original dataset high variance and overfitting are major concerns. The current aim is to reduce the dimension of the dataset by using some logical approaches without incurring much loss of information. As a result, we expected a dataset with minimum missing values and important variables only.

### 2.1.1 Delete variables dominated by NA's

We started cleaning our data by gathering all the variables that include more 10% of NAs. There were 54 variables found that include more than 10% of NAs.

### 2.1.2 Delete variables with singular values

By observing the variables from the *str()* function we spot that the variable *operatingProfitMargin* was an integer. Using the summary() function to observe the variable *operatingProfitMargin* we found out that its minimum and maximum values was 1 (the whole column was filled by 1s and some NA's). Variable *operatingProfitMargin* was removed because it was filled with singular values.

### 2.1.3 Delete highly correlated variables

Before moving into the part of removing highly correlated variables we deleted rows with missing values. A correlation matrix was used as a guide with a cutoff point 0.8 to see the highly correlated variables. Function *findCorrelation()* was used to print out the names of variables that should be removed.

The absolute values of pair-wise correlations are considered. If two variables have a high correlation, the function looks at the mean absolute correlation of each variable and prints the variable with the largest mean absolute correlation.

There were 77 variables removed after that process.

### 2.1.4 Determine the most important variables (tree based algorithm)

Next was we chose to use tree based algorithm r-part to determine the most important variables in the dataset. Top 5, most important variables were chosen. The ones on the y-axis of the figure below.

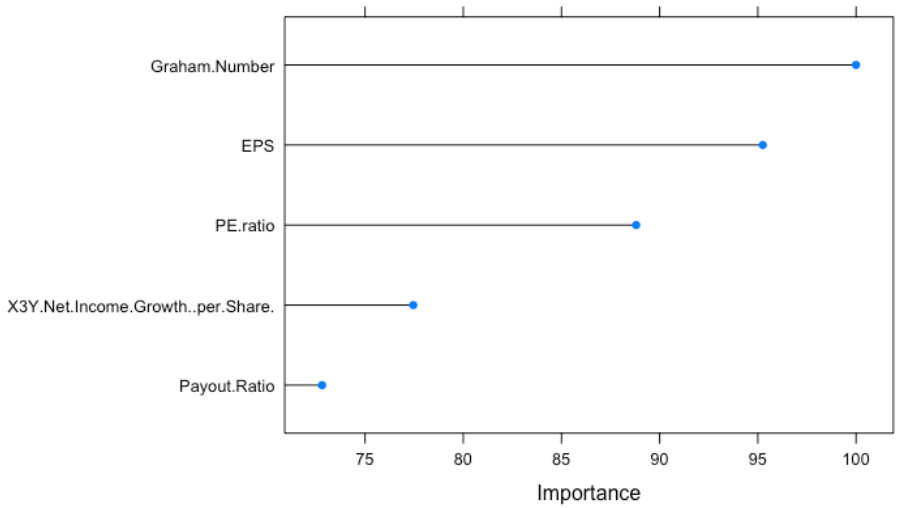


*Figure 2.1: Variable Importance Top 5*

### 2.1.5 Final Data check

From dimension 4392x224 we are not to 2979x8. Our variables remaining variables are;
X, Sector, Graham.Number, EPS, PE.ratio, NIG/share, Payout.ratio and Class. Class is our response, if the percent price variation of the stock for the year is positive then we have a value 1 otherwise 0. Last check before proceeding to exploratory analysis is to confirm that our numerical variables are not correlated. This can be seen in the figure below.

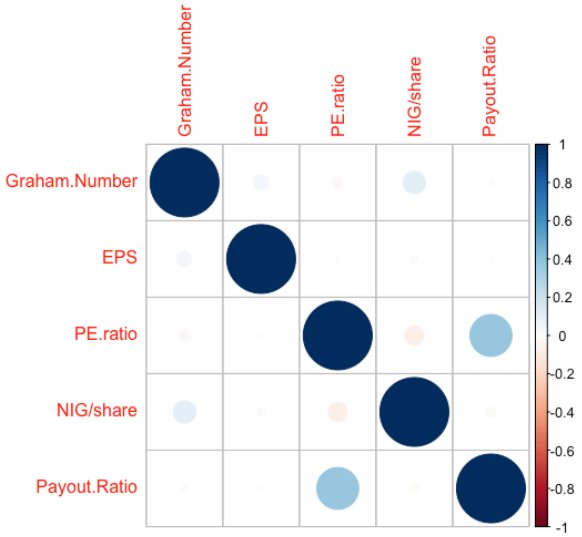


*Figure2.2: Visual Representation of the correlation matrix*

## 2.2 Exploratory Analysis

In this section we will provide some plot of each variable individual related to our response variable Class. Good to note that 71.5% of the stocks had a positive percent price variation for the year.

### *2.2.1 Factor Variables*

From the figure below we can visualize how stocks are split into sector categories. For example, most stocks in our dataset are in Financial Services, Healthcare, Technology and Industrials.

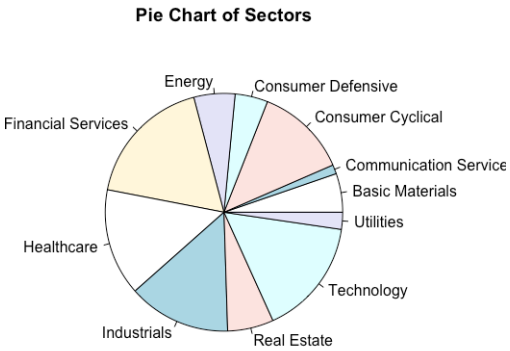


*Figure 2.3: Pie Chart of Sectors*

The following plot presents the inner split of our response variable Class in the Sectors. We can see that Communication services and Energy stocks include the largest percentage of stocks that have negative percent price variation for the year. Financial Services, Real Estate and Utilities are 3 of the best performing sectors, with almost all their stocks having positive percent price variation for the year.

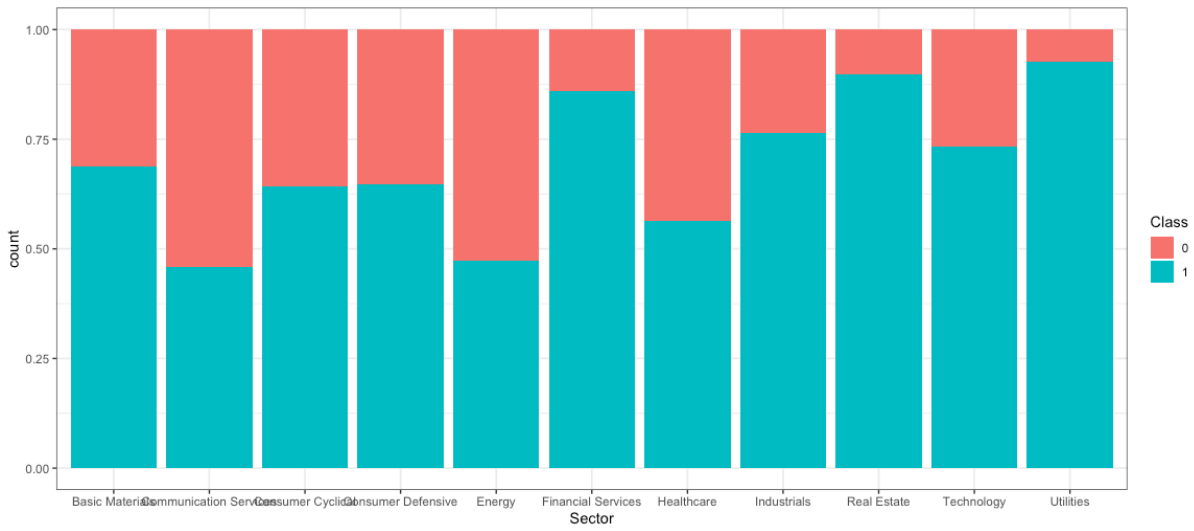


*Figure 2.4: Sector split by Class percentages*

### *2.2.2 Numeric Variables*

The following figure consists of violin plots. Violin plots are like boxplot with the extra added feature of density plot. The wider, the more frequent a number.

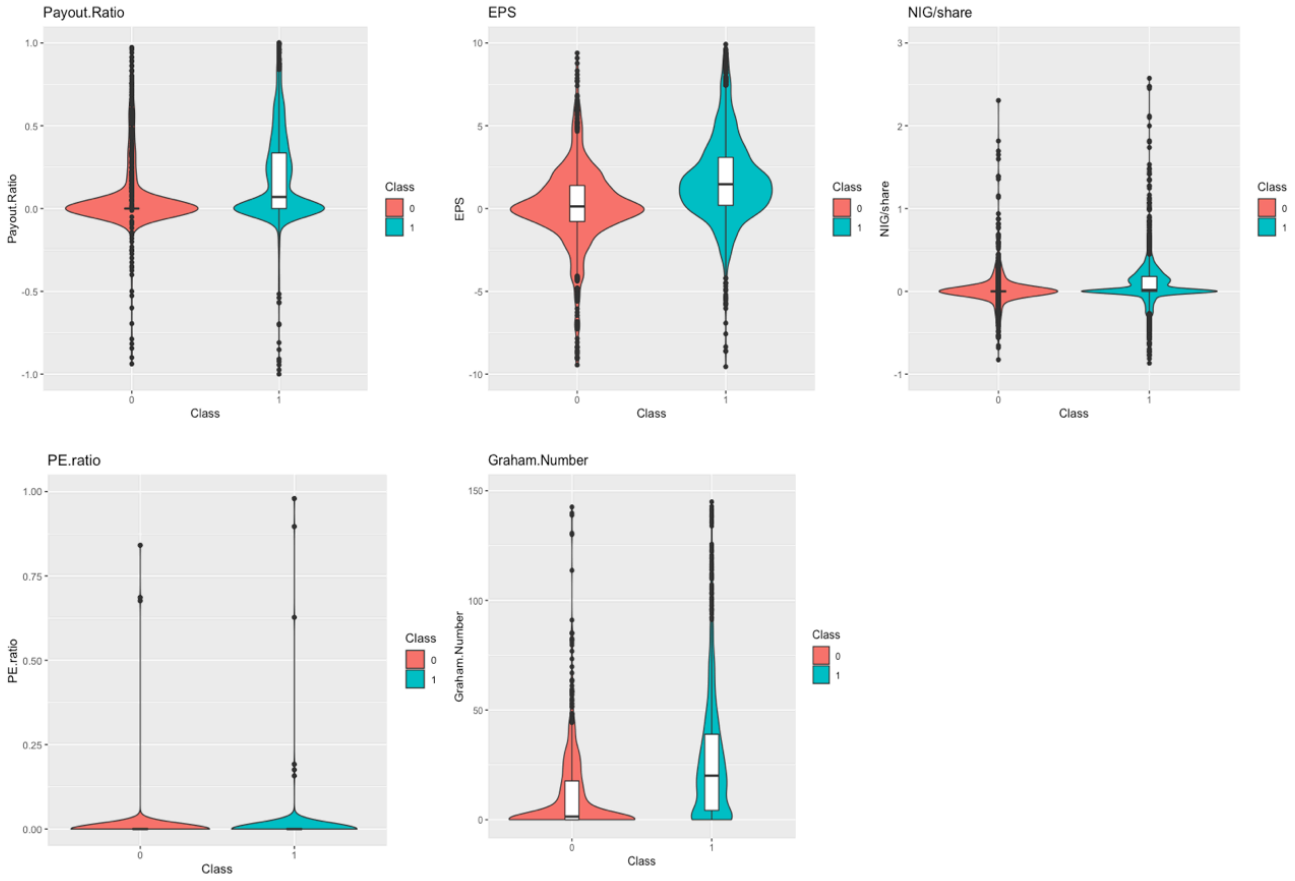


*Figure 2.5: Violin plots of numeric variables*

From the above figure we can see that stocks of class 1 tend to have significantly higher Payout.Ratio, EPS and Graham.Number. Their median and upper quantile on class 1 is relatively higher than on class 0. Variables PE.ratio and NIG/share have approximately the same violin shape and boxplot, but again we can see that the values of class 1 are slightly higher.

`

## *2.3 Comparison of Classification Methods*

Using the variables discussed in the previous sections, we used the train() function and its feature method to try different classification methods. We set controls to repeated 5-fold cross validation. We split the data into 60% train and 40% test. We firstly train our model and test it on the test data set. Methods used were KNN, LDA, QDA, Random Forest, Support Vector Machine and Neural Networks. We obtained the following results;

| **Method** | **Confusion Matrix** | | | **Accuracy** |
|---|---|---|---|---|
| KNN (K = 9) | | 0 | 1 | 0.710 |
| | 0 | 114 | 120 | |
| | 1 | 225 | 732 | |
| LDA | | 0 | 1 | 0.715 |
| | 0 | 42 | 43 | |
| | 1 | 297 | 809 | |
| QDA | | 0 | 1 | 0.715 |
| | 0 | 62 | 62 | |
| | 1 | 277 | 790 | |
| Random Forest | | 0 | 1 | 0.735 |
| | 0 | 106 | 83 | |
| | 1 | 233 | 769 | |
| SVM | | 0 | 1 | 0.721 |
| | 0 | 97 | 90 | |
| | 1 | 242 | 762 | |

Neural Network

| | 0 | 1 |
|---|---|---|
| 0 | 130 | 111 |
| 1 | 209 | 741 |

0.731

As we can see we get approximately the same accuracies from the the different methods used. From the results above we can see that the best performing methods are Neural Network and Random Forest. The worst performing method is KNN.

LDA performed better than any other method on classifying class 1 observations. The best performing method on classifying correctly class 0 observations was Neural Network who is one of the performing methods on the dataset.

In general, all of the method performed good on correctly classifying class 1 observations but not so good on classifying the class 0 observations.