



THE UNIVERSITY OF
SYDNEY

THE UNIVERSITY OF SYDNEY

QBUS2820 Assignment 1

QBUS2820 Predictive Analytics

SID: 500490424

April 11, 2023

Table of Contents

Business context and problem formulation	2
Data processing	2
Main problem with year and ID	3
Exploratory data analysis (EDA)	3
Variable selection	7
Linear regression	7
KNN regression	7
Random forest	7
Methodology and modelling	8
Linear regression	8
KNN regression	8
Random forest	9
Best predictive model on single variable	9
Analysis and conclusion	9
Analysis	9
Problem with MAE as an error function	11
Conclusion	11
Appendix	12

1 Business context and problem formulation

Credit risk analysis is vital to ensure that the debtors are willing and capable of paying back its debts. This is especially important in a bank's decision whether or not to lend money to the entity as well as investor's decision to invest into the entity. Any improvements to the prediction of credit rating can lead to a massive profit within the industry.

In this report three different models are considered, linear regression, KNN regression and random forest. To train each of these model the data set **Credit Risk Rating Data set** was given, this data set was split into the a **training set** and a **testing set** in the ratio of 2:1. The models will be trained on the training set and also tuned on the training set if they have any hyper parameters. Their performance will be tested on the **testing set**. MAE will be used to as the metric to minimise during training as well as to measure the performance of each of the models and the model with the best MAE will be chosen as the final model.

My hypothesis is that linear regression will perform the worst in terms of MAE but will be the most explainable and random forest will produce the best MAE but will be mostly unexplainable.

2 Data processing

In the data processing stage the missing values in the data set were found. The features with missing values are shown in the figure below:

Assets_Total	0
Cash	9
Debt_in_Current_Liabilities_Total	0
Long_Term_Debt_Total	0
Earnings_Before_Interest	0
Gross_Profit_(Loss)	0
Liabilities_Total	0
Retained_Earnings	1
Total_debt/total_asset	0
total_asset/total_liabilities	0
EBTI/total_asset	0
gross_profit/rev	0
EBTI/REV	0
Dividends_per_Share_Pay_Date_Calendar	4
Sales/Turnover_(Net)	0
Stockholders_Equity_Total	0
Interest_and_Related_Expense_Total	1
Market_Value_Total_Fiscal	92
Book_Value_Per_Share	1
Common_Equity_Liquidation_Value	0
Comprehensive_Income_Parent	1
Employees	2
Inventories_Total	4
Earnings_Per_Share_from_Operations	2
Revenue_Total	0
Operating_Activities_Net_Cash_Flow	1
Financing_Activities_Net_Cash_Flow	1
Net_Cash_Flow	0
ID	0
Year	0
Rating	0
dtype: int64	

Figure 1: Number of missing values in each column

Then to fill in the missing values `iterativeImputer` in `sklearn` was used to model the feature with the

missing values as a function of two other features which are the most correlated with the current feature. Then that function was used to impute the missing values. This step was repeated in a loop for all the features with missing values. This method was chosen to impute the missing values instead of just replacing the missing values with the mean or median was because mean and median imputation does not take into account the underlying structure of the data whilst this model-based imputation takes into account the relationships between variables and usually provides a more accurate imputation.

The model based method to impute missing values does introduce multicollinearity into the data, however multicollinearity is not a problem for prediction and since we are prioritising the prediction accuracy of our model this is fine.

2.1 Main problem with year and ID

The basic assumption of predictive analytics is that the future will continue to be like the past. Therefore the features `year` and `ID` are dropped as they contribute nothing to the prediction of future credit ratings. For `year` since there are only two values of year in the data 2014 and 2015, it is not feasible to predict future credit rating if the report is made in any year past 2015 or before 2014 as their `year` variable would not be in the `training data`. For `ID`, this variable is just the index of each of the firms, the index of each firm in the dataframe does not contribute to their respective credit ratings.

3 Exploratory data analysis (EDA)

In EDA the distribution of each of the features were graphed, with the graphs we can see some severe outliers in almost all the features. However most of the outliers are unclear as to whether they are an legitimate entry or a mistake. However there are some clear mistakes in the entries most notably in the features `book_values_per_share`, `total_debt/total_asset`, `total_asset/total_liabilities` and `EBTI/total_assets`. The distributions of the following can be seen in a histogram below:

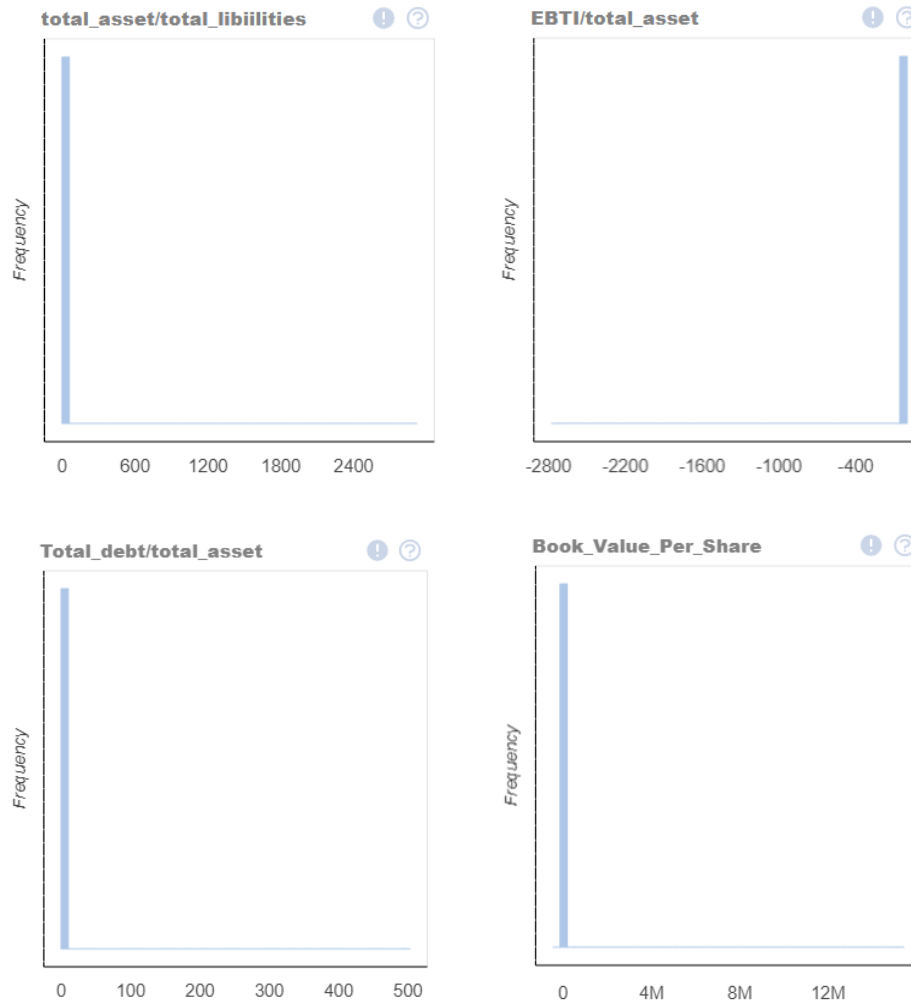


Figure 2: Histogram for features with outliers

We can see that there are some extreme outliers in these distributions, in particular entries 150, 881, 431, 1332, 515 in the excel sheet.

- Entry 150: For this entry the **book_value_per_share** of 15 million is clearly a mistake, the total asset of the firm is only 36762.7400 and the stockholder equity is only 15656.7610 which does not warrant a 15 million **book_value_per_share**.
- Entry: 881: This entry has outliers in **total_debt/total_assets**, **total_asset/total_liabilities** and **EBTI/total_assets**. For the **total_debt/total_assets** it is an mistake because if the total liabilities of the firm is added up it would equal to 10493.7810 which is less then the total assets of 20515.1390 which means that we cannot get the value of 502.8870. **total_asset/total_liabilities** is also an outlier at 2918.4420 because the the total assets is less than **total_liabilities**. Again for **EBTI/total_assets** is also an outlier and the result does not match up. Since there are so many errors in this entry, it was remove from the data.
- Entries 431 and 1132 all have outliers in **book_value_per_share**, and it is obvious its a mistake because **book_value_per_share** of all these entries are greater than the stockholder equity.
- Entries 515 is another mistake, the **book_value_per_share** is very negative at -451900 but the assets are greater than its liabilities.

The `book_value_per_share` of entries 150, 431, 1132 and 515 were set to nan so that it can be imputed based on the same method used in the data processing stage. This method to impute the outliers rather than just removing the whole row retains more information in the data set as all the other features of that entry can still be used. After removing the entry and imputing the rest, the 4 distribution looks like the following:

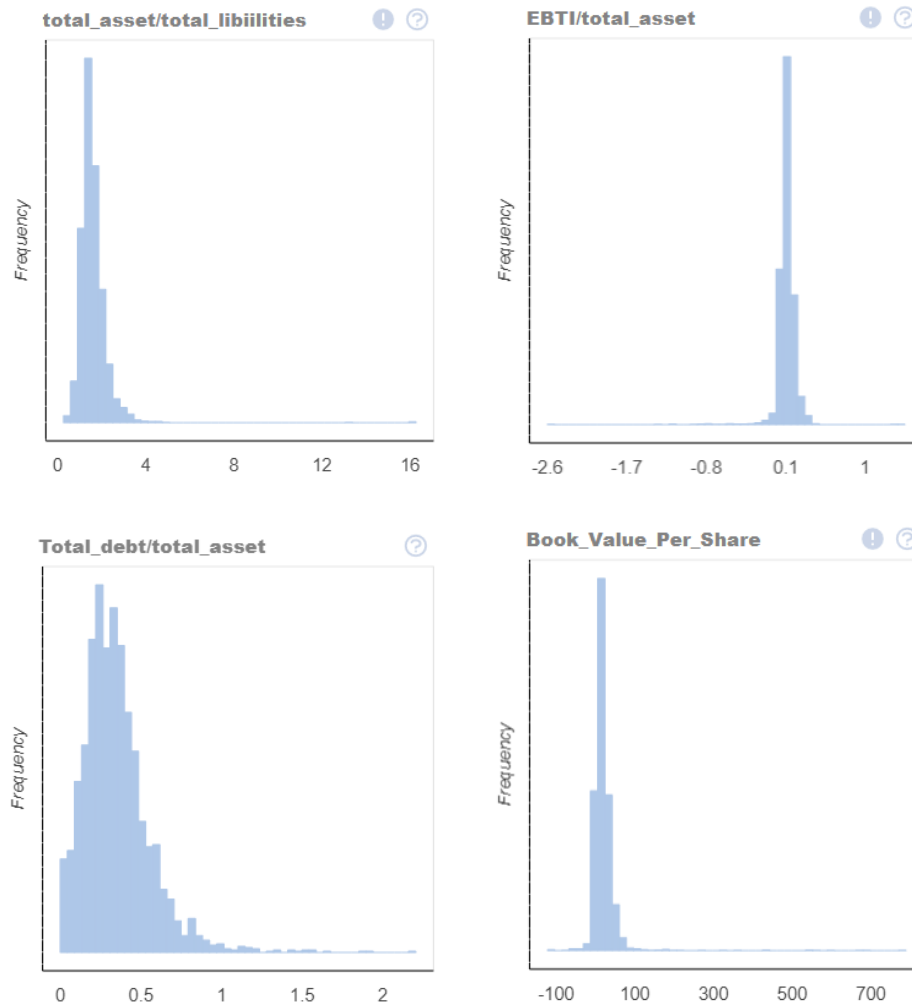


Figure 3: Histogram for features with outliers removed and imputed

The MAE of the models after removing and imputing the outliers improved significantly. The distributions are all skewed including features not shown in the figure, to make it more normal the yeo-johnson transformation could be applied however since we are only looking at the basic linear regression model with no transformations, KNN regression which does not require normal distribution and the third model random forest which does not require manual transformations either so no transformations were applied.

In EDA the correlation between the individual variable and the response (`Rating`) was examined and the 5 most correlated variables were displayed as boxplots below:

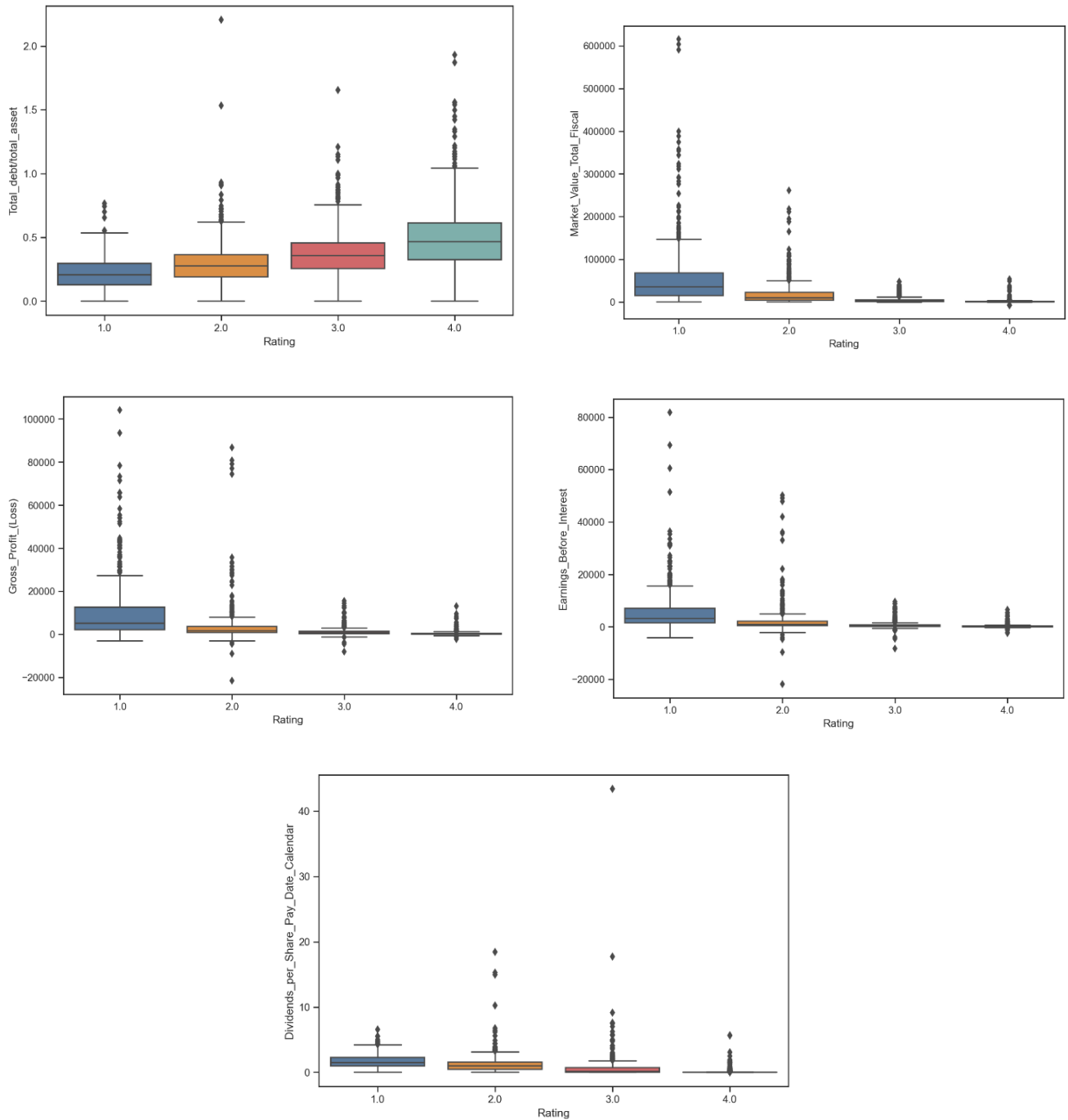


Figure 4: Box plot for most correlated feature to Rating

We can see that with higher rating there looks to be higher `total_debt/total_asset`, lower `Market_Values_Total_Fiscal`, lower `Gross_profit_(loss)`, lower `Earning_Before_Interest` and lower `Dividends_per_Share_Pay_Date_Calendar`. Just by examining the graphs we can see the general trend that firms with less debt, more market value, more profit, more earnings and more dividends paid will have a better credit rating. This intuitively makes sense as firms with better financial performance has a better credit rating. There is no perfect collinearity as seen in figure 7, and since we're optimising for predictive performance first multicollinearity is not examined.

4 Variable selection

Note: All cross-validation preform in this section and the sections onwards will use negative mean absolute error, so higher the cross-validation score the better the model is.

To perform variable selection and model selection later in the report, the data was split into two sections randomly in the ratio of 2:1, the larger portion of data is used to perform variable selection and training of the model, we will call this the **training data**, the smaller portion of the data is used to test the accuracy and the MAE of our model which we will call the **testing data**.

The variable selection was done individually for each model.

4.1 Linear regression

For linear regression a forward selection was used, firstly a base/best model was acquired by fitting the first predictor **Asset_total**. Then for each of the remaining predictors, we fit another model on the list of predictors plus this new predictor, if the model MAE on the **training set** is better than the base model then this model becomes the best model, and the predictor list gets updated with this new predictor. This repeats as more and more predictors are added to the model until the MAE no longer improves as more predictors are added. This is where the forward selection terminates. The variables selected from this method are as follows: [Assets_Total, Cash, Debt_in_Current_Liabilities_Total, Earnings_Before_Interest, Long_Term_Debt_Total, Gross_Profit_(Loss), Liabilities_Total, Total_debt/total_asset, Retained_Earnings, total_asset/total_liabilities, EBTI/total_asset, gross_profit/rev, Dividends_per_Share_Pay_Date_Calendar, EBTI/REV, Interest_and_Related_Expense_Total, Market_Value_Total_Fiscal, Stockholders_Equity_Total, Comprehensive_Income_Parent, Operating_Activities_Net_Cash_Flow].

4.2 KNN regression

For KNN regression forward selection was also used to determine which variables were going to be in the model. In the KNN regression forward selection, all predictors were looped through one at a time, for each predictor a 10 fold cross-validation (**neg_mean_absolute_error** as the error function) on the **training data** was ran on a KNN regression model from neighbour 1 to neighbour 20. If the mean cross-validation score in any of the 20 neighbours were better than the previous best then that predictor was added to a list of chosen predictors. If its not better then we continue with the loop. This looping process is then repeated until no predictors are chosen when we loop through the remaining predictors. This algorithm chooses the number of neighbours along with the variables, the best cross-validation score will determine the best neighbour and the chosen variables. The chosen neighbour is 1 and the chosen predictors for the KNN regression are as follows: [Assets_Total, Debt_in_Current_Liabilities_Total, Long_Term_Debt_Total, Gross_Profit_(Loss), Cash, Earnings_Before_Interest, Retained_Earnings, Sales/Turnover_(Net), Stockholders_Equity_Total, Market_Value_Total_Fiscal, Inventories_Total, Common_Equity_Liquidation_Value, Comprehensive_Income_Parent].

4.3 Random forest

For random forest no variable selection was done as this particular model has its own inbuilt variable selection.

5 Methodology and modelling

5.1 Linear regression

This model was required by the assignment to be included, however normally linear regression would not be used for this kind of classification tasks.

Since linear regression is not used for classification the prediction values it gives will most likely have decimals and have negative values, however our response variable can only take the values of 1, 2, 3 or 4 so a rounding function was made so that if the predictions were less than 1 it would be rounded up to 1 and if the predictions were larger than 4 they would be rounded down to 4 and if the predictions were between 1 and 4 they would be rounded to the nearest integer.

Assumptions for linear regression:

- **Linearity:** $Y = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p + \epsilon$ is the true underlying population model. This assumption is most likely not satisfied as the response variable is categorical in nature and the residuals looks to have a pattern shown in figure 6. This non linearity can potentially be fixed through manual data transformation of the variables such as a log transformation.
- **Exogeneity:** $E(\epsilon|X_1, \dots, X_p) = 0$. This is most likely not satisfied as the mean of the residuals is nowhere close to 0 as shown in figure 6. This assumption can be fixed through manual data transformation to make the residuals have a mean close to 0.
- **Independence:** The data are iid. Unknown whether or not this assumption is satisfied.
- **4th moments exist for all predictors and response variable:** The 4th moment exists for the response variable because it is confined between 1 and 4. However it is unknown whether the 4th moment of all the predictors exist, but it is unlikely they do since there are huge outliers within predictors such as `Assets_total`. This assumption is unfixable given the nature of the data, there could very well be `Assets_total` that can keep growing and is unbounded.
- **No perfect collinearity:** This is satisfied as shown in figure 7

The `sklearn LinearRegression()` was used as the linear model. The linear regression model was fitted on the `training data` using the variables obtained from the previous section, since there are no hyper parameter for linear regression no hyper parameter tuning was done. The performance was then measured by predicting the `test data` and calculating the MAE.

5.2 KNN regression

This was another model that was required for this assignment, this model was better than the linear model because it was actually designed for both classification and regression and the results show this as well with its MAE being significantly better than that of linear regression. There are no assumptions to use the KNN regression model as it is a non parametric model which means it does not assume an underlying population model like linear regression. The distance metric chosen for this model is the mahalanobis distance because in practise it usually works better than eculidean distance. Sklearn's `KNeighborsRegressor()` was used in this report as the KNN model.

The data was also mix-max scaled to be between $[0,1]$ for KNN regression in order to prevent one feature value which is very large like `Assets_total` from dominating features with small values like `total_asset/total_liabilities`.


KNN regression has a main hyper parameter that needs tuning which is the number of neighbours it uses to predict new data. This hyper parameter was tuned using cross-validation with 10 folds together in one function with forward selection as explained in the variable selection section. The best number of neighbours which minimised the cross-validation score was 1. After finding the best KNN model it was fitted on the **training data** with neighbour equal to 1 and the performance was then measured by predicting the **test data** and calculating the MAE.

5.3 Random forest

Since random forests is a non parameteric model so it does not assume an underlying distribution and can handle skewed and categorical data. This model was chosen specifically because of its performance right out of the box as it can handle a wide range data types, does not assume anything, is specifically built for classification tasks and usually performs better than single classifier models such as KNN as it is an ensemble. The random state of the random forest is fixed to 1 so that each time it is ran it would produce the same output.

The number of estimators or trees in the random forest was tuned using cross-validation from 1 tree to 500 trees. Then the random forest with the highest cross-validation score was chosen and the number of trees chosen was 483. This best model was then fitted on the **training data** with 483 estimators and the performance was then measured by predicting the **test data** and calculating the MAE.

5.4 Best predictive model on single variable

To find the best model on a single variable all the predictors were looped through one by one and for each predictor each of the three candidate models found previously were tested using cross-validation with 10 folds using `neg_mean_absolute_error` on the **training data** and the predictor with the highest cross-validation score were selected. Then the KNN and random forest model were tuned again using 10 fold cross-validation for their hyper parameters. The results can be seen in table 2. 


6 Analysis and conclusion

6.1 Analysis

The MAE of all 3 models fitted with their respective chosen variables and predicted on the **testing data** is shown in the table below:

Model	MAE
Linear regression	0.5034
KNN regression (k = 1)	0.2556
Random forest (trees = 483)	0.1928

Table 1: Model MAE

It can be clearly seen in the table above that the best model is random forest which has the best MAE, significantly better than the other two models. This is due to the fact that random forest is an ensemble learning method for classification. Ensembles generally produce a better result than single contribution model and ensembles have better robustness than single models because they reduce the spread or dispersion of the predictions. 

However a downside with the ensemble is that they are are less interpretable than a simple model such as linear regression and in an industry where business decisions are made according to the models,

interpretability may be very important. However in the context of our problem this is not an issue, this will be discussed further in the conclusion which will justify the trade offs between explanations and better MAE.

Below a graph of the individual importance of the features is shown, where the y axis is the mean accuracy decrease if the feature is removed from the model, so the higher the mean accuracy decrease the more important the feature is. From the graph it can be seen that the 3 most important driving factors for the credit rating is **Retained_Earnings**, **Dividends_per_share_Pay_Date_Calendar** and **Market_Value_Total_Fiscal**.

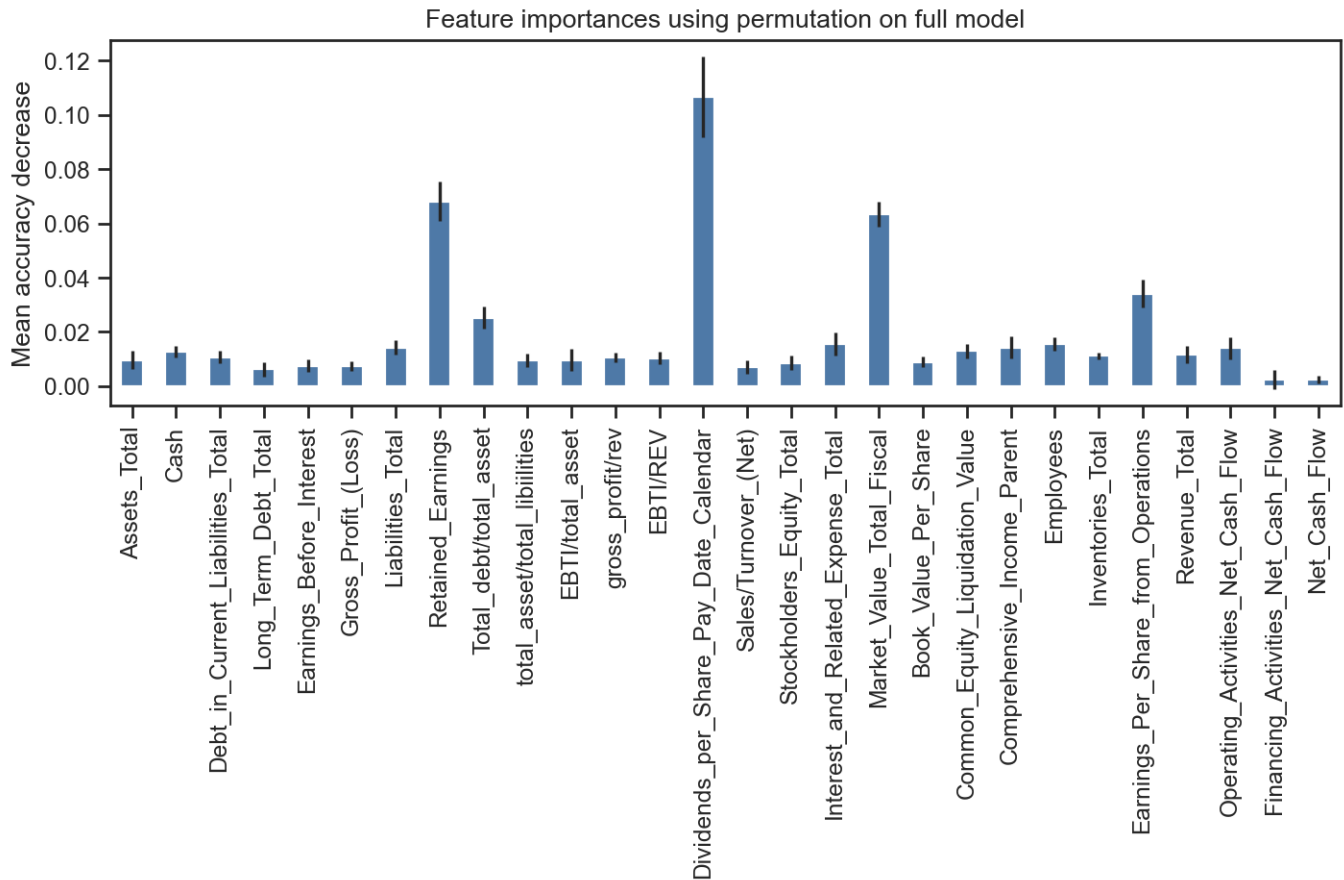


Figure 5: Driving factor for rating

The best MAE for the 3 model classes on single variables are shown in the table below:

Model	Variable	MAE
Linear regression	Dividends_per_Share_Pay_Date_Calendar	0.7351
KNN regression (k = 19)	Retained_Earnings	0.5310
Random forest (trees = 8)	Dividends_per_Share_Pay_Date_Calendar	0.6726

Table 2: Single variable MAE

We can see that the best single variable model is the KNN regression and the worst model is linear regression.

6.2 Problem with MAE as an error function

Note: Under predicting here means predicting a smaller credit rating than the true value (i.e. model predict 1, but true value is 4).

MAE measure the average magnitude of the error in the predictions without taking into consideration whether the predictions are over or under the actual credit ratings. This can create issues where under predicting the credit rating can result in a bank or investors lending money to a firm with a lower capability or willingness to pay back debt which could cause the bank or investors to loose money. It could also over predict which may make investor hesitant to invest in a firm that has a good credit rating. Therefore to combat this the error metrics mean percentage error (MPE) is used so that its clear whether the model is over predicting or under predicting the true values on average. MPE also punishes extremely wrong predictions as it calculates the error as a percentage of the true value. Below is a table which shows the MPE of the three candidate models:

Model	MPE
Linear regression	-9.6526
KNN regression	-3.4846
Random forest	-4.4656

Table 3: Model MPE

We can see from the table above that KNN regression is the best in regards to using MPE as the error function and we can see that on average all three of our models over predicting on credit rating.

6.3 Conclusion

The accurate estimation of prediction error is shown in table 1 with MAE as the loss function.

The best overall model based MAE is random forest with 483 trees with a MAE of 0.1928, the main driving factors are `Retained_Earnings`, `Dividends_per_share`, `Pay_Date_Calendar` and `Market_Value_Total_Fiscal`. We cannot fully explain how these factors impact rating but it is known that they contribute the most towards the model accuracy. The linear model is the most explainable out of the three with its many assumptions, but we can see that out of the three models it has the worst performance on unseen data. In the context of the question we are predicting the credit rating of firms, it does not matter if we understand how each variable influence the credit rating, as long as we get the credit rating correct the investors will make more accurate judgement and increase the probability of making money. Therefore this trade off between accuracy and explanations is necessary.

The best single variable model is KNN regression with 19 neighbours, on the variable `Retained_Earnings` with a MAE of 0.5310.

The hypothesis was correct, random forest performed the best and linear regression performed the worst in terms of MAE.

7 Appendix

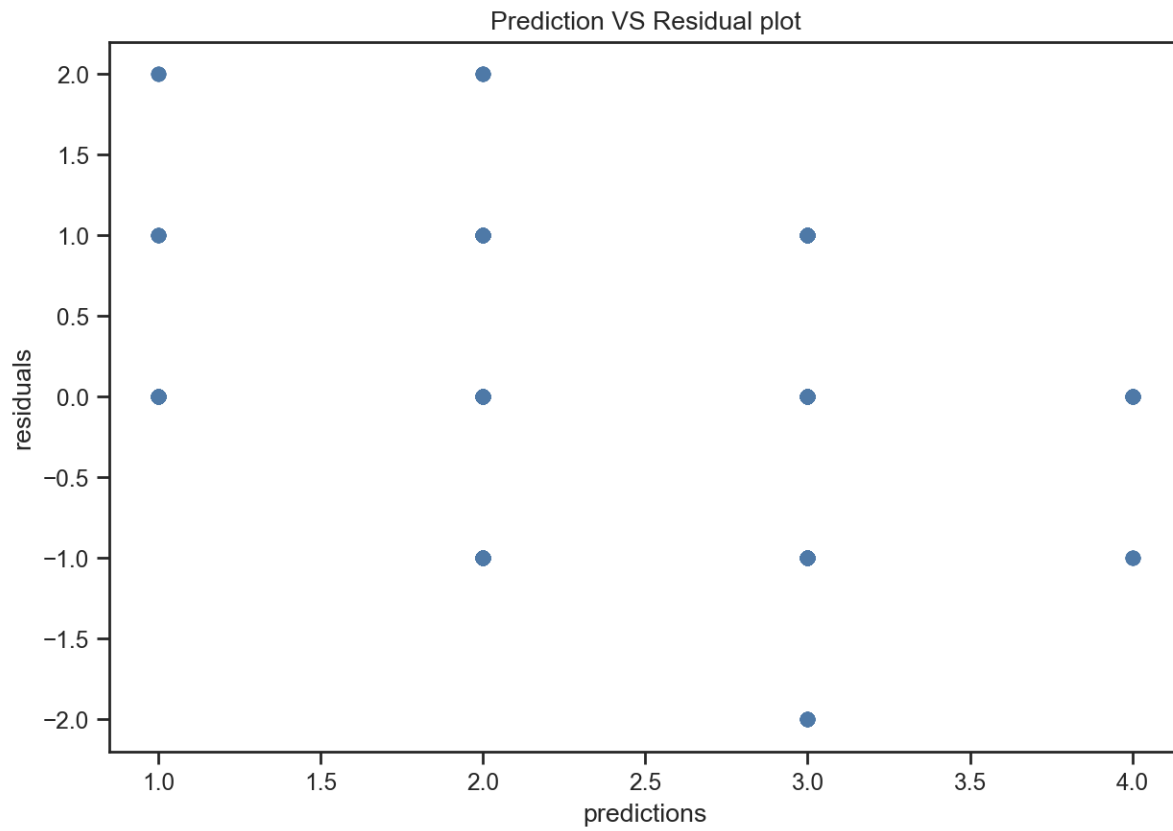


Figure 6: Residual plot for linear regression

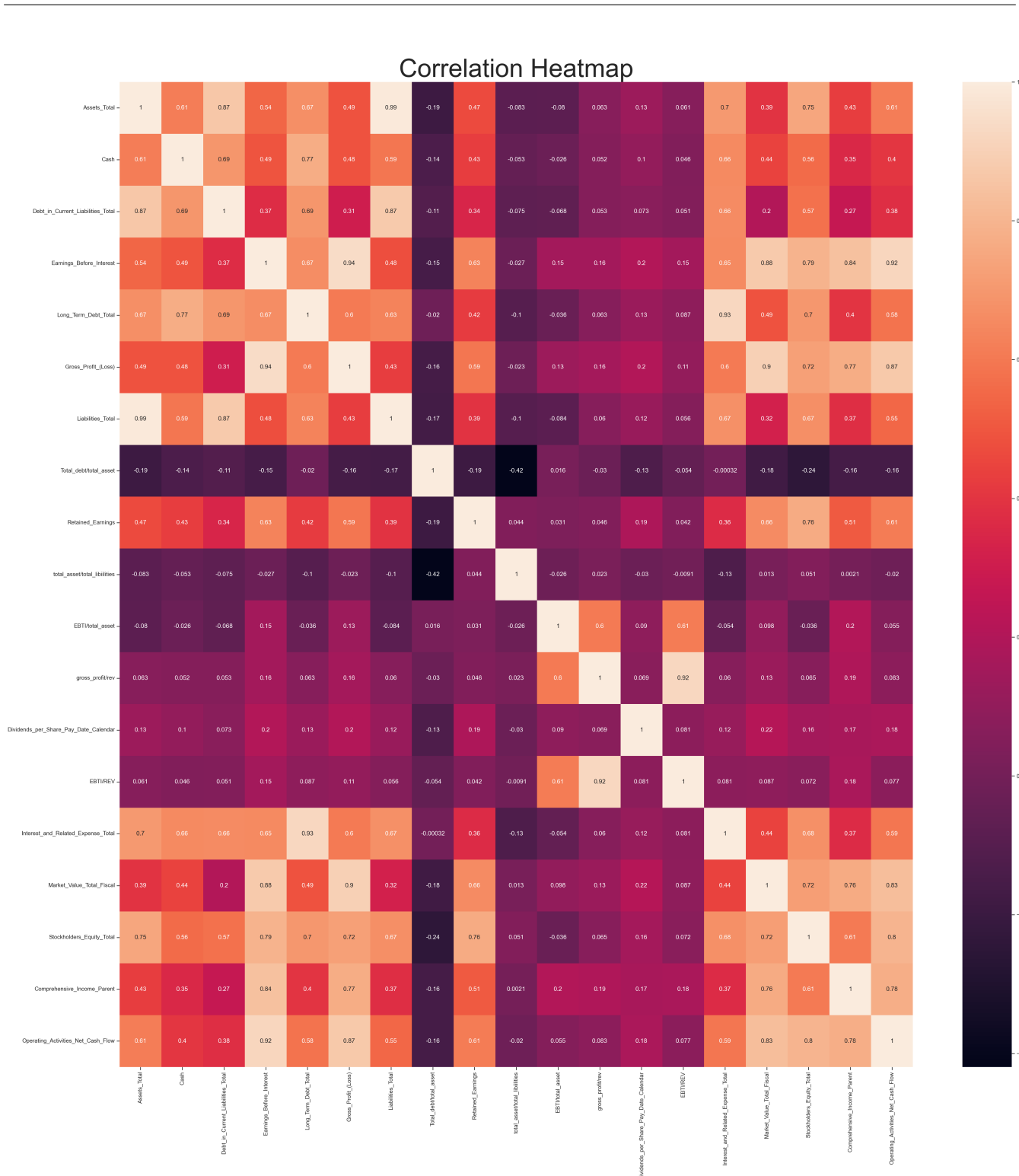


Figure 7: Correlation heatmap for chosen variables in linear regression