Boston University

Metropolitan College

CS 699 Fall 2018

WEKA Project

Ravi Shastri

Krishanu Ghosh

Table of Contents

[1. Overview 1](#_Toc529372895)

[2. Data Mining Goal 1](#_Toc529372896)

[3. Detailed description of the dataset 2](#_Toc529372897)

[4. Data Pre-processing 4](#_Toc529372898)

[4.1 Initial load and classification run in WEKA 4](#_Toc529372899)

[4.2 Observations 5](#_Toc529372900)

[4.3 Manual preprocessing actions 5](#_Toc529372901)

[4.4 Initial Classification runs 6](#_Toc529372902)

[4.5 Smoothing class attributes and revised run 7](#_Toc529372903)

[4.6 Revised classification runs 7](#_Toc529372904)

[5. Detailed description of data mining procedure 8](#_Toc529372905)

[5.1 Overview of Attribute selections 8](#_Toc529372906)

[5.2 Overview of Classification Algorithms run 9](#_Toc529372907)

[6. Interpretation of Data Mining Results and Evaluation 10](#_Toc529372908)

[6.1 InforGainAttributeSelection 10](#_Toc529372909)

[6.2 CorrelationAttributeSelection 11](#_Toc529372910)

[6.3 PrincipalComponentAnalysis 12](#_Toc529372911)

[6.4 ReleIf 13](#_Toc529372912)

[6.5 SymetricUncertain 14](#_Toc529372913)

[6.6 No Attribute Selection 15](#_Toc529372914)

[7. Classifier Performance Evaluation and model selection 16](#_Toc529372915)

[8. Discussion and Conclusion 17](#_Toc529372916)

[9. Individual Contribution 17](#_Toc529372917)

[10. Appendix 17](#_Toc529372918)

# Overview

This project is part of the CS99 Data Mining class of Fall 2018. The goal is to perform data mining task with a real-life dataset. The data mining task chosen in the course is Classification which is a Supervised Learning process.

WKA is chosen for Machine Learning tool for the project. It is also the data mining tool chosen by the current CS 699 program. Weka is a collection of machine learning algorithms for data mining tasks. It contains tools for data preparation, classification, regression, clustering, association rules mining, and visualization. Weka is open source software issued under the GNU General Public License.

The overall tasks are choosing a real-world dataset, define data mining goal and perform necessary data mining tasks to achieve the goal which has a potential for practical use. The target is to build data mining, evaluate the data mining result using appropriate performance measures. By requirement definition Data Mining of type “classification.” is chosen.

# Data Mining Goal

Use financial ratios of companies to arrive at their Issuer credit rating classification. For this purpose, we got 29 various financial ratios for 453 SP 500 companies. We plan to augment this data set with R2000 stock data if required.

The idea is to be able to run various mining algorithms from the Weka set and train the set and use this to see if we can successfully rate a new company. We believe we can also use this framework to rate pre-IPO and private placement companies, if we can adjust the ratio for control, liquidity and other pertinent factors.

In situations where we only have a company’s operating ratios for nonpublic companies such as in initial public offering or in private equity or private debt situation, this could be a very good first cut screen, to see if a deal is worth investigating.

# Detailed description of the dataset

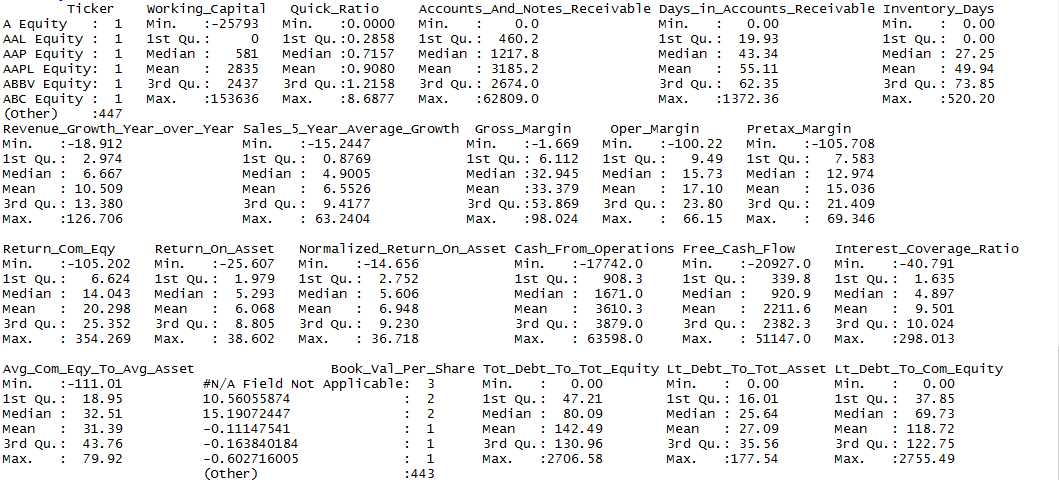
The dataset chosen is financial ratios of 453 SP 500 companies and classification attribute is their Issuer credit rating. There are 29 attributes in data set and they are explained below For this purpose we got 29 various financial ratios for.

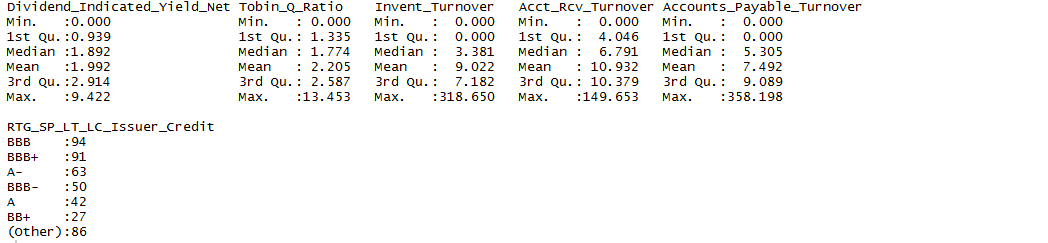
Below is functional description of all 29 Attributes in Original dataset with original dataset attached.



|  |  |
| --- | --- |
| **Data Column Name** | **Data Description** |
| Ticker | Name of the company |
| Working Capital | Measure of a company's operational efficiency and its short-term financial health |
| Quick Ratio | Addition of cash, cash equivalents, short-term investments, and current receivables together then dividing them by current liabilities |
| Accounts and Notes Receivable | Asset of a company, bank or other organization that holds a written promissory note from another party |
| Days in Accounts Receivable | Accounts receivable / average sales per day |
| Inventory Days | Number of days in a year (365 or 360 days) divided by the inventory turnover ratio |
| Revenue Growth Year over Year | Ratio of difference between two consecutive year's annual revenue and to the revenue from the prior year and multiplied by 100. This gives percentage growth rate of total revenue between the two years. |
| Sales 5 Year Average Growth | Average growth of overall Sales over last 5 years |
| Gross Margin | Gross margin is the difference between revenue and cost of goods sold (COGS) divided by revenue. Gross Margin is a type of profit margin |
| Operating Margin | Operating margin is a measure of profitability. It indicates how much of each dollar of revenues is left over after both costs of goods sold and operating expenses are considered |
| Return Com Equity | Measure of the profitability of a business in relation to the equity |
| Return on Asset | Measure of the profitability of a business in relation to the equity |
| Normalized Return on Asset | Calculate Return on Asset removing the effects of seasonality, revenue and expenses that are unusual or one-time influences |
| Cash from Operations | Cash Flow from Operating Activities = Net income + Noncash Expenses + Changes in Working Capital. |
| Free Cash Flow | Free cash flow is the cash a company produces through its operations, less the cost of expenditures on assets |
| Interest Coverage Ratio | The interest coverage ratio is used to determine how easily a company can pay their interest expenses on outstanding debt. The ratio is calculated by dividing a company's earnings before interest and taxes (EBIT) by the company's interest expenses for the same period |
| Average Com Equity to Average Asset | Ratio of Average Equity to Average Asset value |
| Book Val Per Share | Book value per share indicates the book value (or accounting value) of each share of stock. Book value is a company's net asset value, which is calculated by total assets minus intangible assets and liabilities. |
| Tot Debt to Tot Equity | Ratio of Total Debt to Total Equity |
| Lt Debt to Tot Asset | Ratio of Total Debt to Total Asset |
| Lt Debt to Com Equity | Ratio of Total Debt to Total Equity |
| Dividend Indicated Yield Net | Estimated amount of total dividends on a share of stock for the coming year |
| Tobin Q Ratio | Ratio of the market value of a company's assets (as measured by the market value of its outstanding stock and debt) divided by the replacement cost of the company's assets (book value). |
| Invent Turnover | The inventory turnover ratio is calculated by dividing the cost of goods sold for a period by the average inventory for that period |
| Acct Receivable Turnover | Ratio of the net value of credit sales during a given period to average accounts receivable during the same period. |
| Accounts Payable Turnover | Ratio of total purchases made from suppliers, or cost of sales, and dividing it by the average accounts payable amount during the same period |
| RTG SP LT LC Issuer Credit | Credit rating Class Attribute  Distribution of class attribute values  BBB :94  BBB+ :91  A- :63  BBB- :50  A :42  BB+ :27  (Other):86 |

Summary of the Raw Input Data by value in individual attributes





# Data Pre-processing

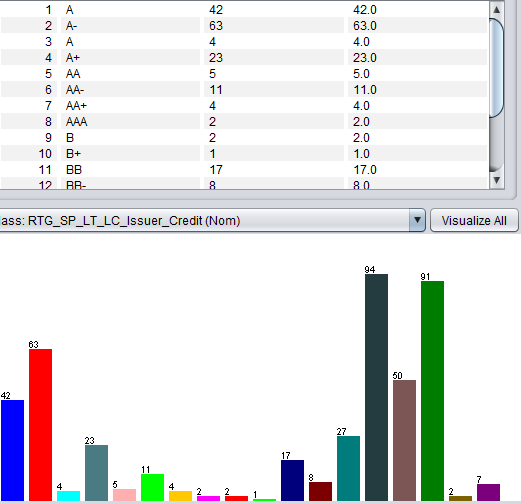
The overall steps in Data Pre processing / Data smoothing to make the Data in acceptable format is as below

## Initial load and classification run in WEKA

The Data File is converted to WEKA File and J48 classification was run. We did preliminary tests on the input file using a holdout method and cross validation using Naïve Bayes and J48. For these tests we considered the input file with all the missing values replaced by mean / mode, stripped version of the file will less number of attributes and tuples, controlling for number of missing values and a version with no missing values.

## Observations

1. There are a total 18 distinct values in Class Attributes



* Attribute “Book Value per Share” was classified by WEKA as neither Numeric nor Categorical. On detail observation we found noise “#NA Field Not Applicable” in data.
* We also noticed there are rows where there are related columns where one of them is zero and the other is non zero.
* We found that Normalized Return on Asset and Return on Asset are very closed.

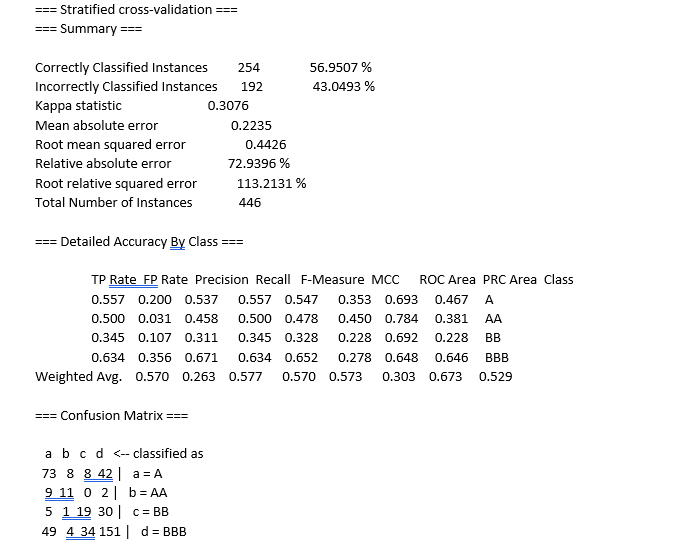
## Manual preprocessing actions

Based on the above observations following update/massaging was performed on data.

* Rows with Attribute “Book Value per Share” as “#NA Field Not Applicable” removed from in dataset.
* Rows with related ratios as zero removed from the dataset.
* Rows with classifiers BB+/BBB+ OR BB-/BBB- are updated to BBB for benefit of classifier analysis. The same is performed on As.

## Initial Classification runs

* We run J48 Decision tree classifier and Naïve Bayes after above pre-process of data and the results were found as below.



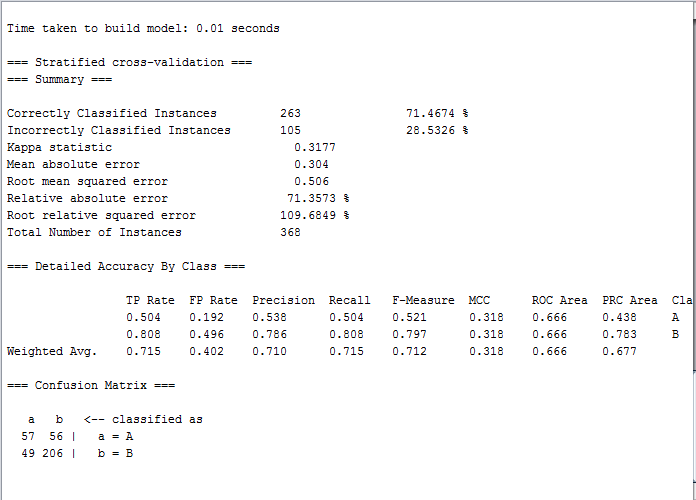
* The performance noted under 60% and we are inclined to further pre-processing of the dataset.

## Smoothing class attributes and revised run

* We smoothed the data further by Binning the classifier values to a relative smaller set of A and B. We classified all A/AA/AAA from revised File to A and all B/BB/BBBs to B for benefit of Classification models. Our objective was to use the pre-processed file and using the different classification schemas come up with a model that will give us an ability to classify a company into A or B rating just using it financial ratios.

## Revised classification runs

* Result of revised classification runs is recorded below. We start with this File as our pre-processed data file and proceed with Attribute selection and classification runs.



# Detailed description of data mining procedure

## Overview of Attribute selections

We used the following 5 Weka attribute selection algorithms to generate 5 input files for further classification.

1. InfoGainAttributeEval : Evaluates the worth of an attribute by measuring the information gain with respect to the class. We applied InfoGainAttributeEval with rank filer of .03. Any attribute with a rank less than .03 was dropped. We chose this as all below this had a rank of 0.This brought the number of attributes to 10 from the original 26.
2. Principal Components: Performs a principal components analysis and transformation of the data. We configured it to give up to 5 principal components that explained 95% of the variance. This brought the number of attributes to 5 principal components from the original 26.
3. CorrelationAttributeEval : Evaluates the worth of an attribute by measuring the correlation (Pearson's) between it and the class. We chose a rank limit of .1 . All observation below this rank were deleted. Below this we were getting negative ranks. This brought the number of attributes to 15 from the original 26.
4. ReliefFAttributeEval :Evaluates the worth of an attribute by repeatedly sampling an instance and considering the value of the given attribute for the nearest instance of the same and different class. (uses 10 neighbors). We used .003 as the rank limit for this. All attributes below this were dropped. Most below this had 0 rank. This brought the number of attributes to 17 from the original 26.
5. SymmetricalUncertAttributeEval :Evaluates the worth of an attribute by measuring the symmetrical uncertainty with respect to the class. We used the rank limit of .003. All attributes below this were dropped. This brought the number of attributes to 10 from the original 26.

.

**Rationale for choosing the above attribute selection algorithms:**

In finance we use PCA, Correlation and Euclidean distance between asset values quite a bit. During the pre-preliminary testing we found that tree algorithms generate better results than Naïve Bayes. Based on the above facts, we chose InfoGainAttributeEval, Principal Components, CorrelationAttributeEval, ReliefFAttributeEval and SymmetricalUncertAttributeEval.

## Overview of Classification Algorithms run

After generating the 5 input file using the above attribute selection algorithms, we applied the following classifiers to each of the input file as well as the raw input file:

1. Naïve Bayesian

statistical classifier that predicts class membership probability that a given tuple belongs to a class, assuming class conditional independence.

1. J48 ( C4.5)

algorithm used to generate a decision tree developed by Ross Quinlan mentioned earlier. C4.5 is an extension of Quinlan's earlier ID3 algorithm. The decision trees generated by C4.5 can be used for classification, and for this reason, C4.5 is often referred to as a statistical classifier.

1. KNN with 5 nearest neighbors

K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). We are used the nearest 5 neighbors.

1. Random Forest

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.Random decision forests correct for decision trees' habit of overfitting to their training set.

1. Bagging with each of the classifiers.

Bootstrap aggregating, also called bagging, is a machine learning ensemble meta-algorithm designed to improve the stability and accuracy of machine learning algorithms used in statistical classification and regression. It also reduces variance and helps to avoid overfitting. Although it is usually applied to decision tree methods, it can be used with any type of method. Bagging is a special case of the model averaging approach.

# Interpretation of Data Mining Results and Evaluation

The above discussed classifiers were applied with 10-fold cross validation. Cross-validation is a technique for evaluating ML models by training several ML models on subsets of the available input data and evaluating them on the complementary subset of the data. Use cross-validation to detect overfitting, i.e. failing to generalize a pattern.

The synopsis of the result is given below: (The confusion matrices are attached in the appendix). We are considering the following performance measures to compare the different classifier results:

1. Correctly Classified Rate
2. Incorrectly Classified Rate
3. TP Rate
4. FP Rate
5. F Measure
6. ROC Area

## InforGainAttributeSelection

Results for input selected using InfoGainAttributeEval attribute selection algorithms are summarized below.





## CorrelationAttributeSelection





## PrincipalComponentAnalysis





## ReleIf





## SymetricUncertain





## No Attribute Selection





# Classifier Performance Evaluation and model selection

We took the results of each performance measure for each of the classifier runs and obtained the optimal score for each measure across all of 20 runs. That generated a set of optimal scores for each measure across the 20 runs plus the run with original base file without any attribute selection. Below are the optimal scores for each of the measure across the 20 runs.



We marked the grid in green for each run when the score matched the best scores for that performance measure. Then comparing the number of green cells, we arrived at the best performing attribute selection / classifier combination.

The Correlation based attribute selection with Random Forest under Bagged ensemble generates the best result. It has the top accuracy and lowest error rate. It also has the best F measures for both the A and B attributes. The ROC curve is in the top three amongst the result.



# Discussion and Conclusion

The model using Random forest classification on ReleiF attribute selection has the best performance. The reason for choosing this model is given above.

We started off with data that had missing data as well as redundancy, both explicit and implicit. We removed the explicit ones and use the attribute selection to remove the implicit ones. This is evident from the fact that the rate went from ~60% for the raw file to ~75% in the preprocessed and smoothened data. Then we run suitable selection of 5 attribute selections and classifications and were able to reach 80.16% for the final model.

To get the better accuracy out of the data model, the data set needs to be enhanced and the exercise of multiple attribute selection and classification repeated. In our case the model is a first step that classifies a company to the compressed set of class values. We plan to get a larger set consisting of Russel 2000 and MSCI indices and re-run the process to ensure we are classifying into the full slate of the attribute values.

During data mining, there is always missing data, which is treated in different ways. The final model is produced, usually after removing the data or correcting the data, to get optimal performance. In practice there is a margin of error around this success rate because at time the input data is missing and has to be interpolated. Hence we would recommend to run the selected model through the original data and provide the users the margin of error to expect in terms of performance. Having a range of performance metrics sets up a realistic expectation.

# Individual Contribution

|  |  |
| --- | --- |
| Overview and Organizing Document | Krishanu Ghosh |
| Data Description | Krishanu Ghosh |
| Data Mining Goal | Ravi Shastri |
| Pre-Processing Data / Data Smoothing | Krishanu Ghosh |
| Classifier runs | Ravi Shastri |
| Result Interpretation / documentation | Ravi Shastri |
| Summary / Final conclusion | Ravi and Krishanu |

# Appendix

Output of WEKA Attribute selection and Classifier runs uploaded as “classfilerOutputFile.doc”.