German Credit – CIND 119 Project

# Members

1. Kevin Maciver Riquelme, kmaciver@ryerson.ca
2. Lanzo Siega, lanzo.siega@ryerson.ca
3. Natalia Yuka Hiratani, nhiratani@ryerson.ca
4. Sara Kmair, skmair@ryerson.ca

Summary: This report will present a data analysis in order to bank make a better decision about to provide loan to customers. A database which contains 20 attributes and 1.000 customers will be used for this analysis. Among these attributes, 17 are categorical. In the data cleaning process, it was treated missing values, outliers, normalization, and relationship between variables. Logistic Regression, Naïve Bayes, Decision Tree, and Random Forest are statistical methods that help to classifier if the customer has a good or bad creditability. It was used filters and parameters in order to check the best model to the dataset. ROC (Receiver Operating Characteristics) is the evaluation metric for checking the classification model’s performance. Random Forest shown the best result among the other three methods with 86% of recall. In the post-prediction analysis, k-means and EM were presented as a clustering data, and were performed in the 700 customers. After the process, it was concluded the customers were divided in 3 clusters where each one has different features. In conclusion, the project provides an analysis which management can have more accurate view of customers. It was recommended business strategies such as target marketing, and improve fees to yield more revenue.

# Workload Distribution

In this section, you need to mention who did what in the project.

|  |  |
| --- | --- |
| Member Name | List of Tasks Performed |
| Kevin Maciver Riquelme | Data Cleaning, Post-Prediction Analysis |
| Lanzo Siega | Supervised Algorithm – Naïve Bayes, Conclusions |
| Natalia Yuka Hiratani | Supervised Algorithm – Logistic Regression, Conclusions |
| Sara Kmair | Supervised Algorithm – Decision Trees, Comparison |

# Data Preparation

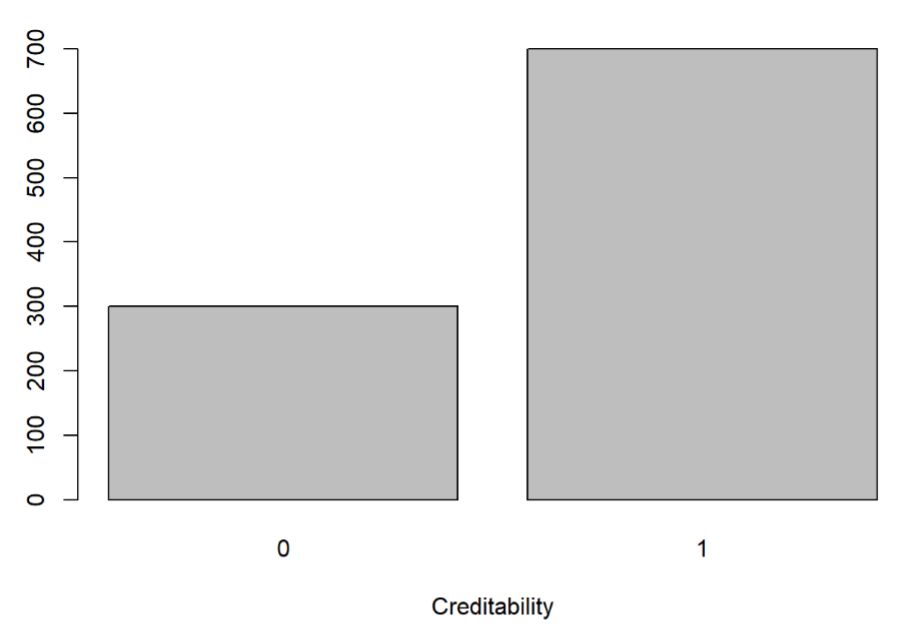
The data set is called German Credit Data and contains 1000 instances, 20 attributes and the class attribute (Creditability). Creditability is a nominal attribute with two values: denied (0), and approved (1), showing a bad or a good credit risk for loan approval. When first uploading the data in R all attributes are integers, although as specified in the data library below, most of these integers have a categorical representation:

1. **Creditability:** The class attribute (qualitative)showing whether the credit rating is good or bad.
2. **Account Balance:** Checking account status (1: < 0 DM, 2: 0<=...<200 DM, 2 > 200 DM, 4: No checking account), where DM= Deutsche Mark (qualitative attribute).
3. **Duration of Credit (month):** Duration of credit in months (numerical)
4. **Payment Status of Previous Credit:** Credit history (qualitative) 0: no credits taken, 1: all credits at this bank paid back duly, 2: existing credits paid back duly till now, 3: delay in paying off in the past, 4: critical account.
5. **Purpose:** Qualitative attribute showing the purpose of the loan (0: New car, 1: Used car, 2: Furniture/Equipment, 3: Radio/Television, 4: Domestic Appliances, 5: Repairs ,6: Education ,7: Vacation, 8: Retraining ,9: Business, 10: Others)
6. **Credit Amount:** Numerical value showing the credit amount
7. Value Savings/Stocks: Qualitative attribute showing average balance in savings and stocks (1: < 100 DM, 2: 100<= ... < 500 DM, 3: 500<= ... < 1000 DM, 4: =>1000 DM, 5: unknown/ no savings account)
8. **Length of current employment:** Qualitative attribute showing length of employment (1: unemployed, 2: < 1 year, 3: 1<=...<4 years, 4: 4<=...<7 years, 5:>=7years).
9. **Instalment percent:** Installment rate in percentage of disposable income (numerical)
10. **Sex & Marital Status:** Qualitative attribute showing gender and marital status (1: male: divorced/separated, 2: female: divorced/separated/married, 3: male: single, 4: male: married/widowed, 5: female: single)
11. **Guarantors: (Qualitative) Guarantors and co-applicants:** (1: none, 2: co-applicant, 3: guarantor)
12. **Duration in Current address:** Qualitative value showing the duration in current address (1: <= 1 year, 1<...<=2 years, 2<...<=3 years, 3:>4years)
13. **Most valuable available asset:** Qualitative attribute showing valuable assets (1: real estate 2: savings agreement/ life insurance, 3: car or other, 4: unknown / no property)
14. **Age (years):** Numerical value showing age in years.
15. **Concurrent Credits:** Installment plans (1: bank, 2: stores, 3: none)
16. **Type of apartment:** Type of housing (1: rent, 2: own, 3: for free)
17. **No of Credits at this Bank:** Numerical value showing number of existing credits at the bank
18. **Occupation:** Job (Qualitative) (1: unemployed/ unskilled - non-resident, 2: unskilled - resident, 3: skilled employee / official, 4: management/ self-employed/highly qualified employee/ officer)
19. **No of dependents:** Numerical value showing number of dependents
20. **Telephone:** Qualitative attribute for telephone number (1: yes, 2: No)
21. **Foreign Worker:** Qualitative attribute showing whether the person is the foreign worker or not (1: yes, 2: no)

Although some of the attributes on the data library are assigned as numeric, in some cases the numbers repeat themselves in a small set. Therefore, those attributes were also set to categorical, (e.g No of dependents, Instalment Percent).

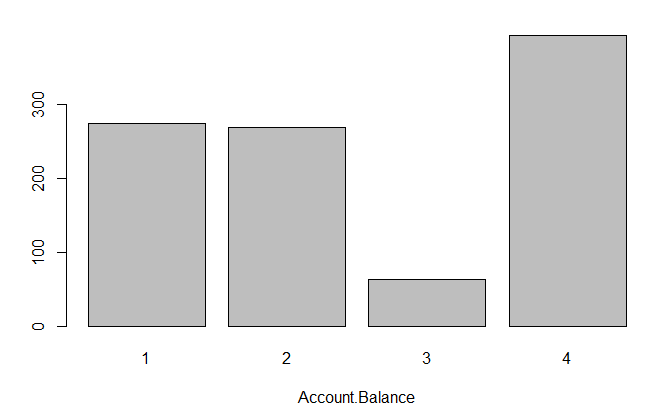
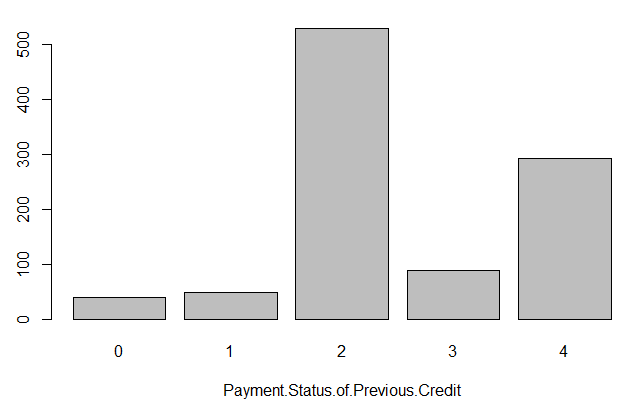
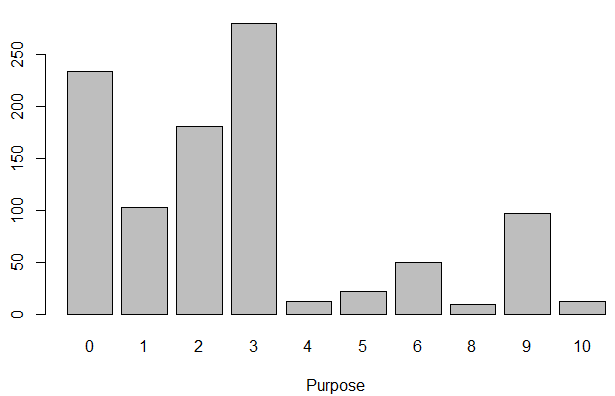
By making the adjustment for the attributes data types the new data contains a total of 17 categorical attributes and only 3 numerical attributes (Age, Durations of Credit and Credit Amount).

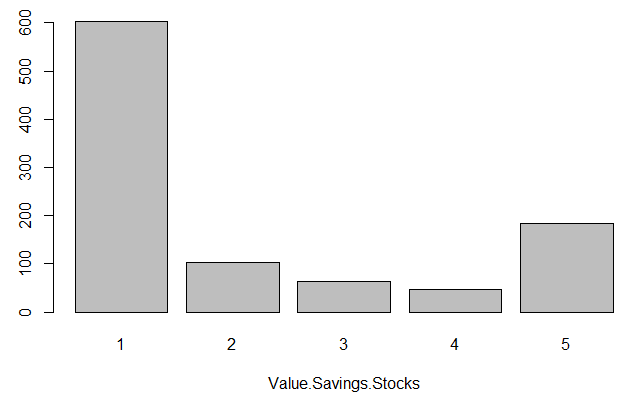
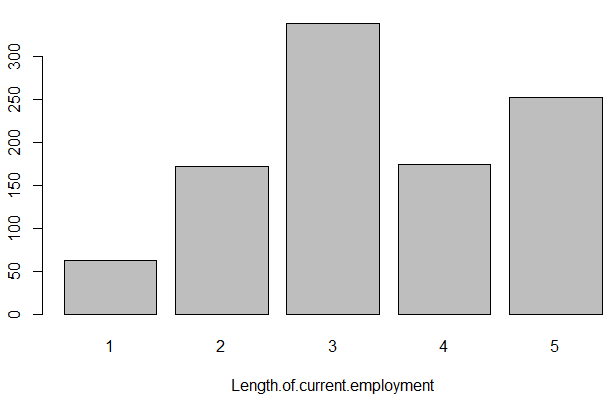
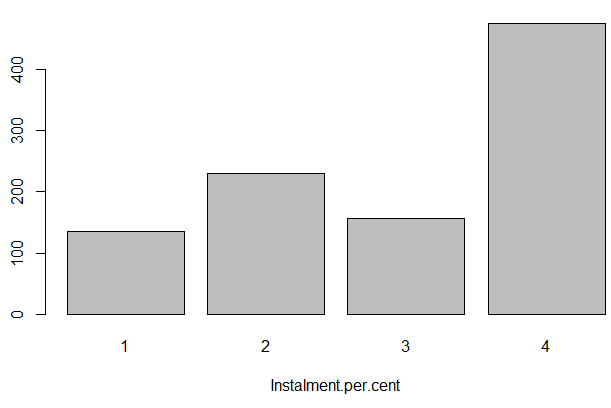
By visualizing the data summary it’s possible to see some imbalance in the class variable. Which is visually confirmed by the bar plot bellow.

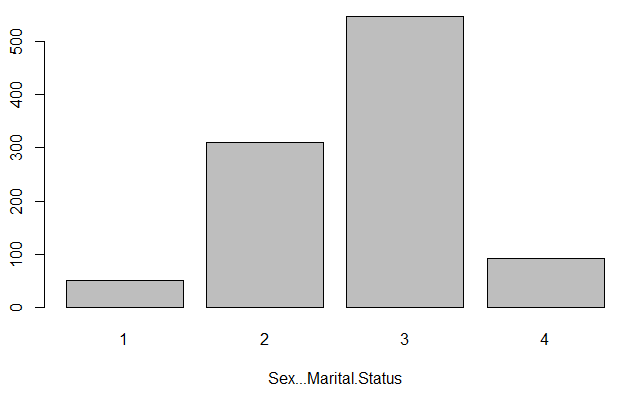
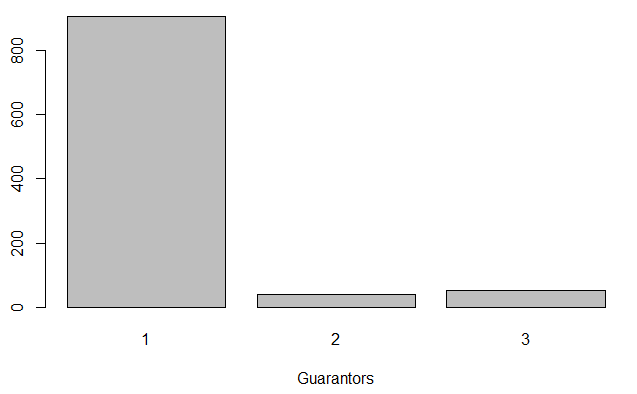
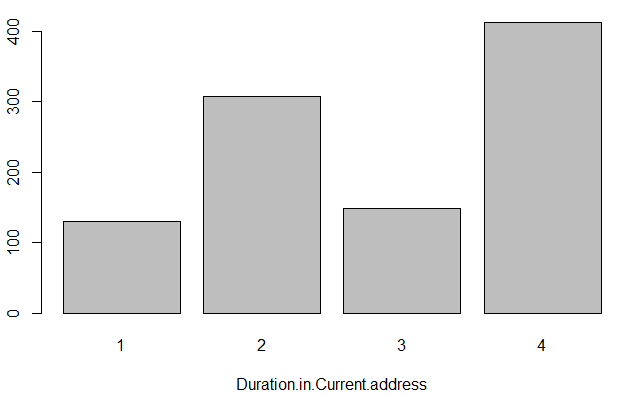


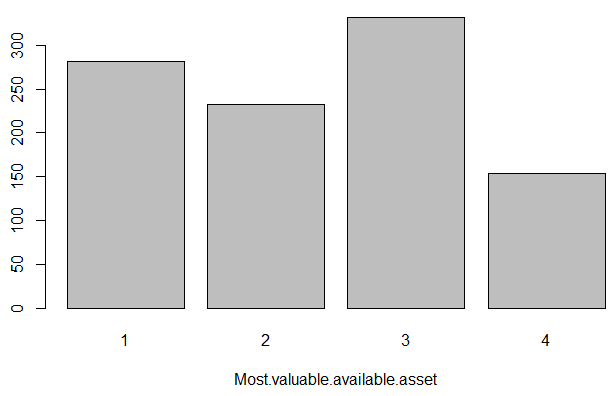
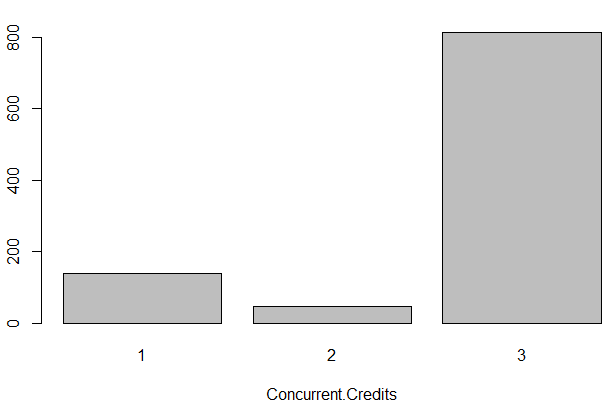
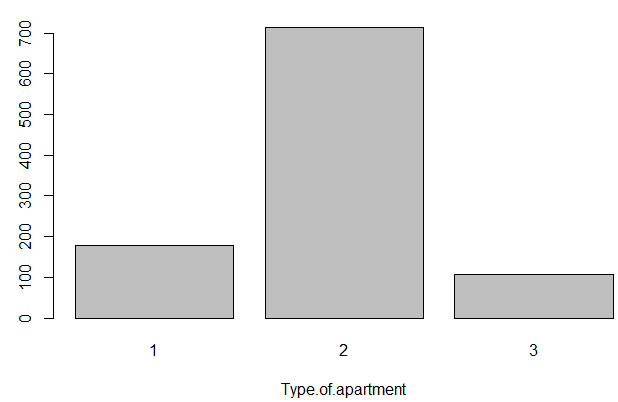
The imbalance will be treated during supervised modeling.

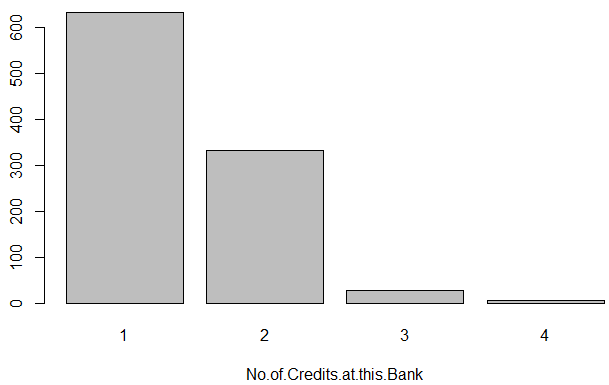
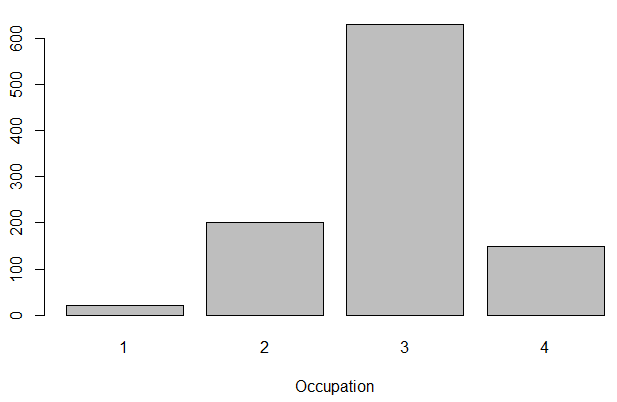
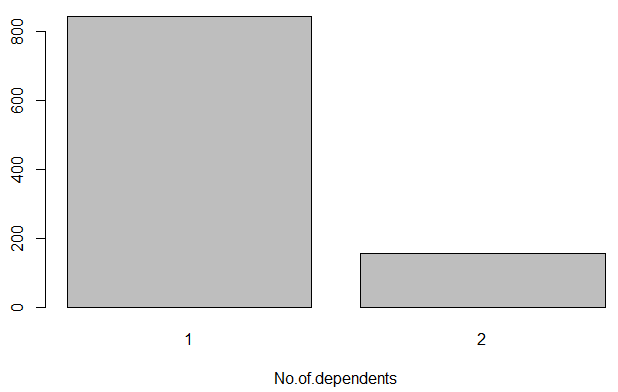
Bar plots were also used for the other categorical attributes.

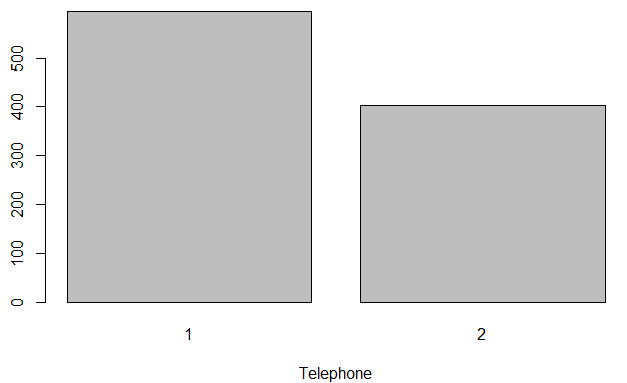
  

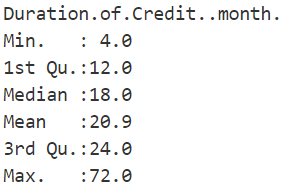
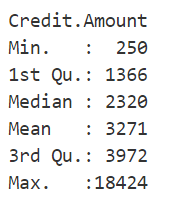
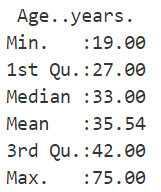
  

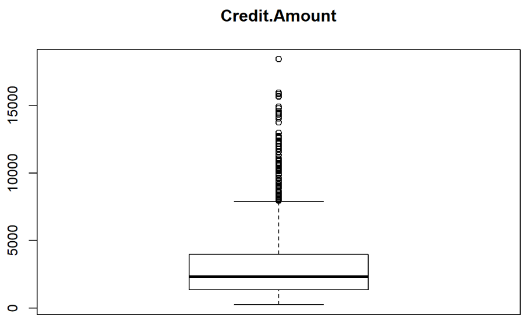
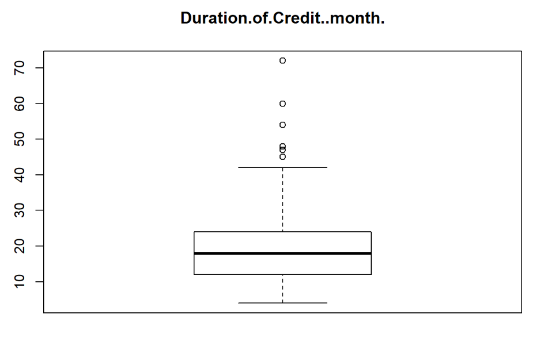
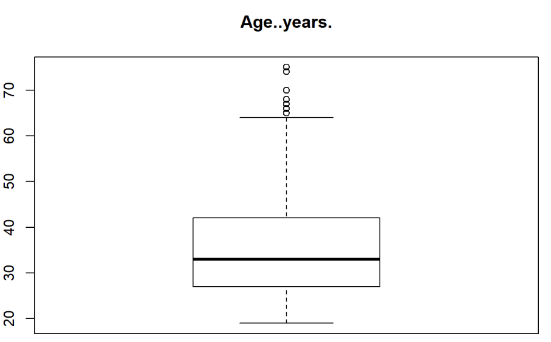
  

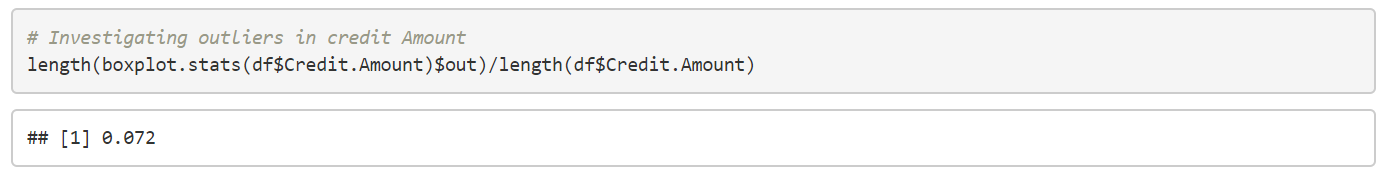


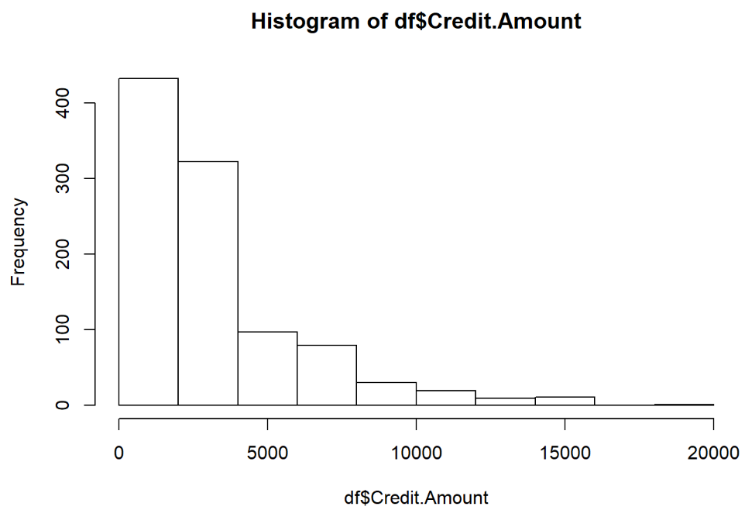
For the numeric attributes the range was evaluated, no incorrect values, such as negative or very high number appear on the data:



Boxplot were also plotted in order to evaluate the distribution as well as outliers.

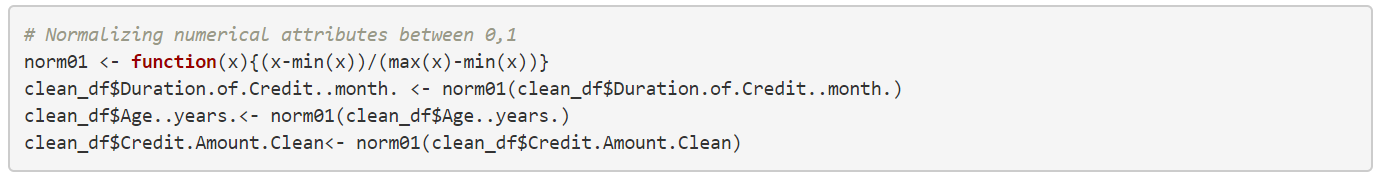
A considerable number of outliers seem to be plotted for Credit Amount. A further analysis shows that 7.2% of the instances are outliers for Credit Amount.





Although some techniques are available to treat these outliers, the numbers do not seem to be incorrect, meaning they can be actual reliable data. Therefore, no treatment for the outliers were made.

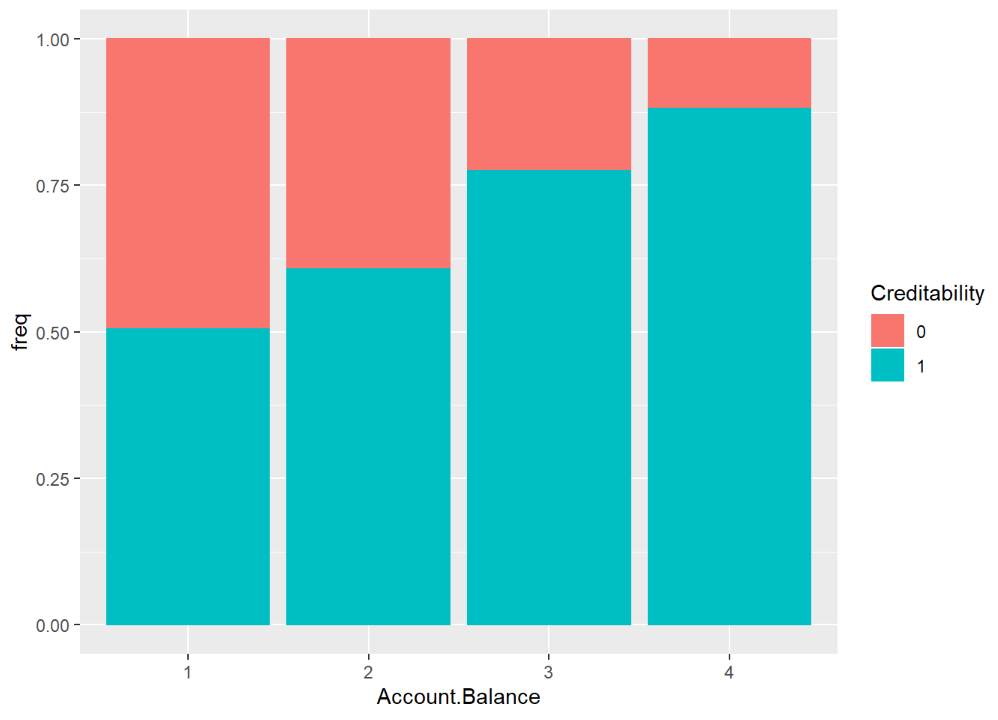
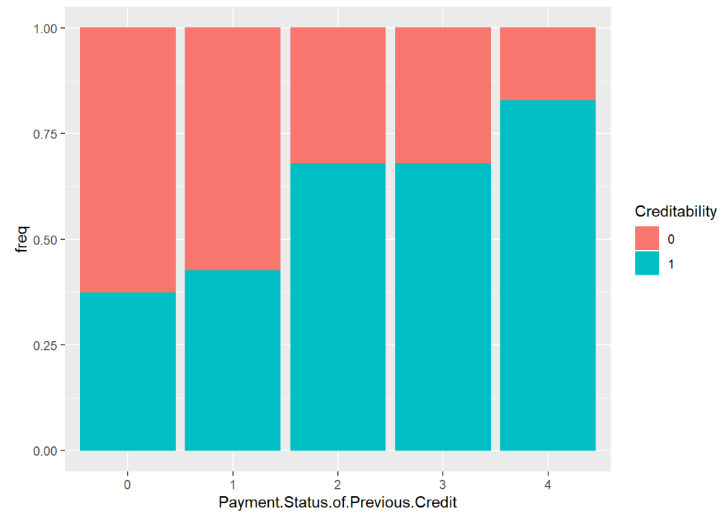
Considering that the numeric attributes have very different ranges, normalization was executed to standardize them between zero and one.

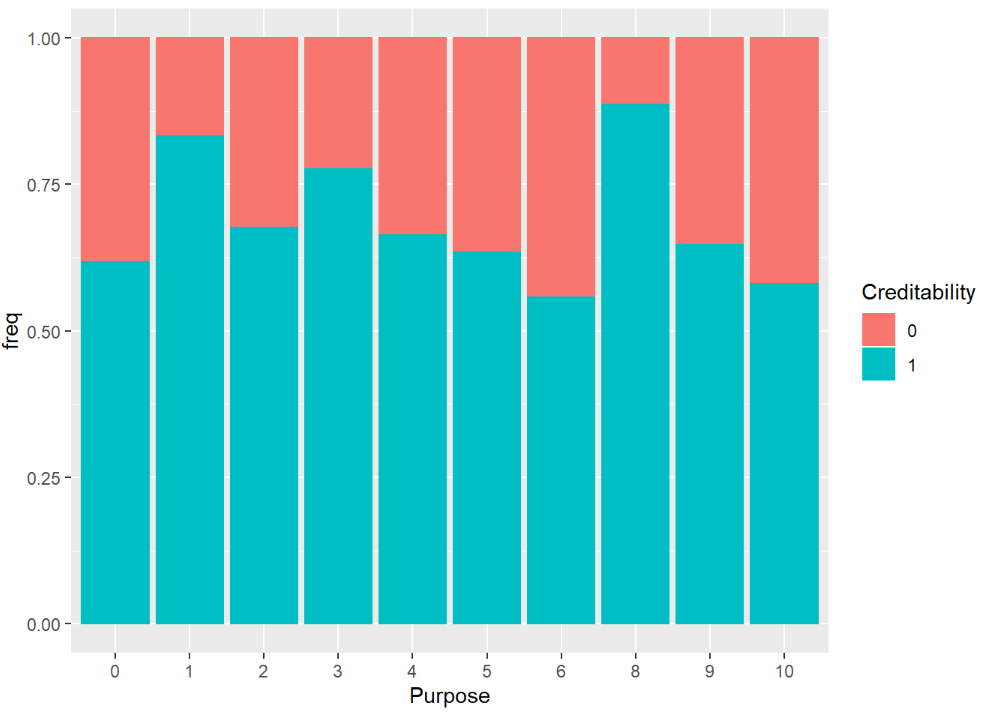
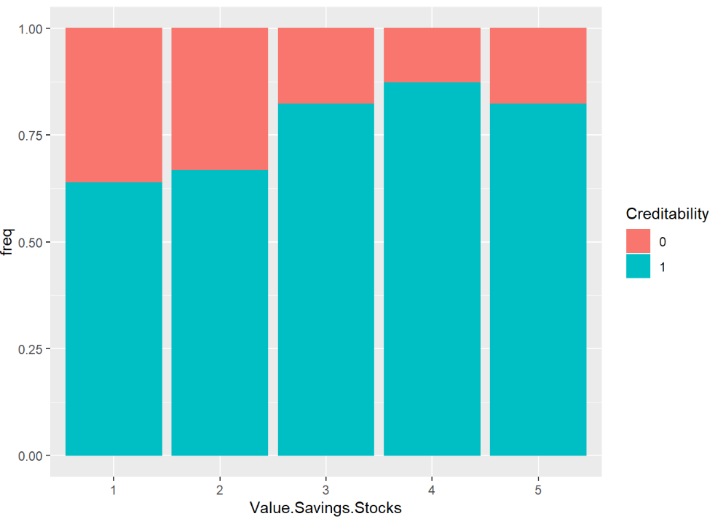


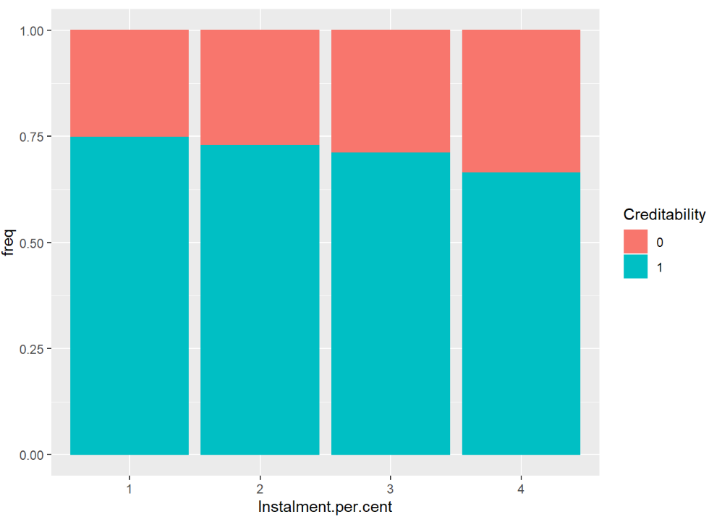
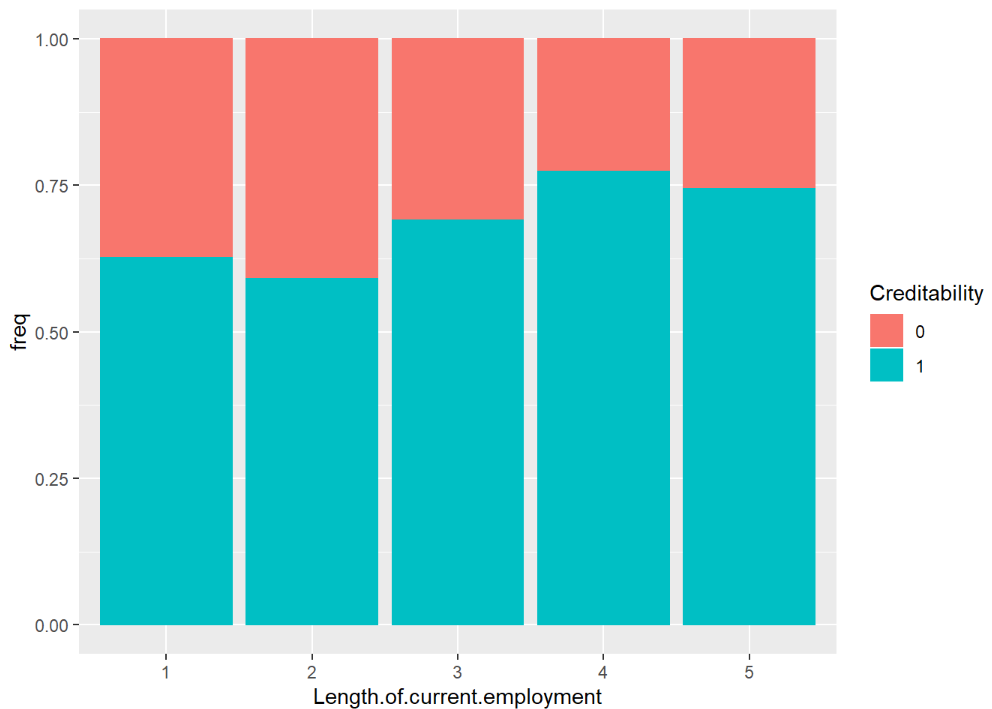
It’s also worth mentioning that the data has no missing values.

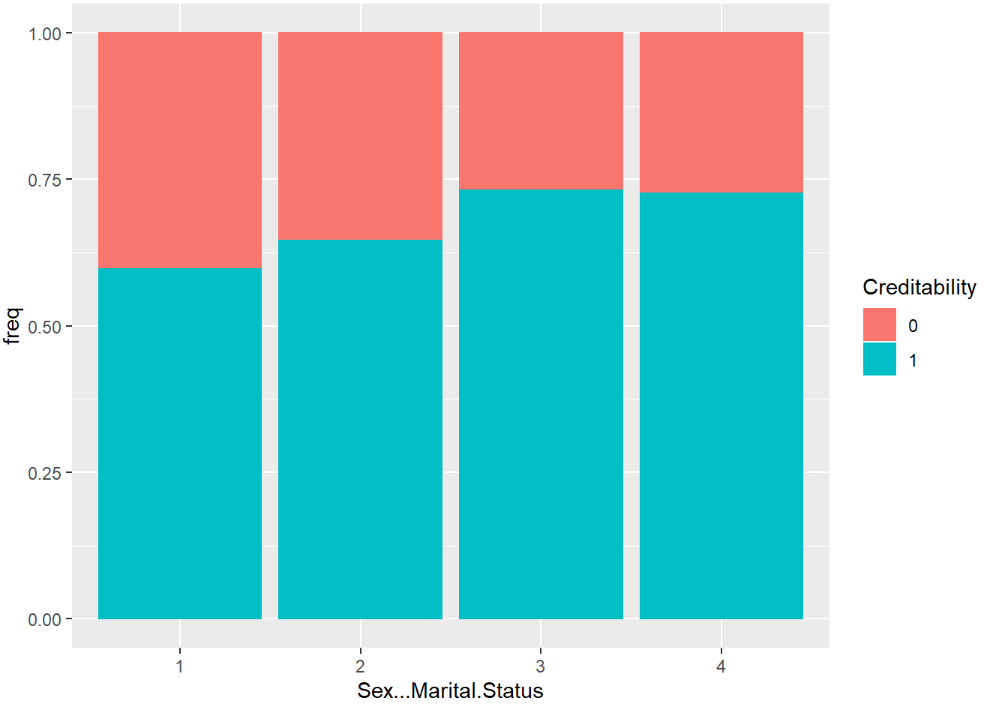
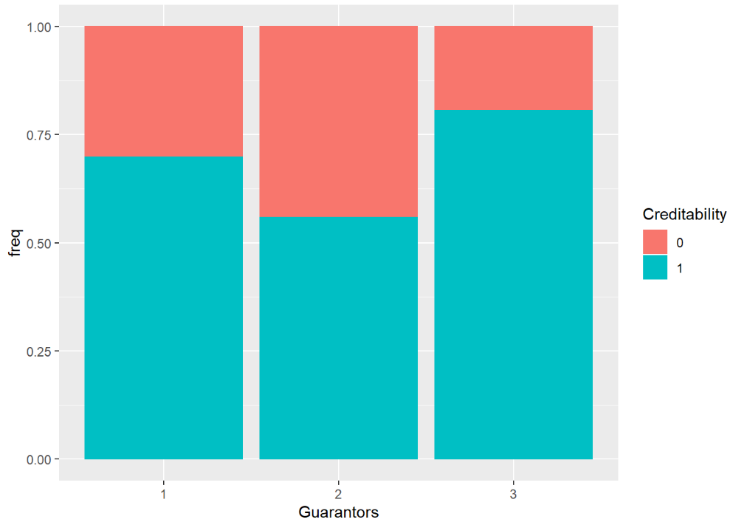


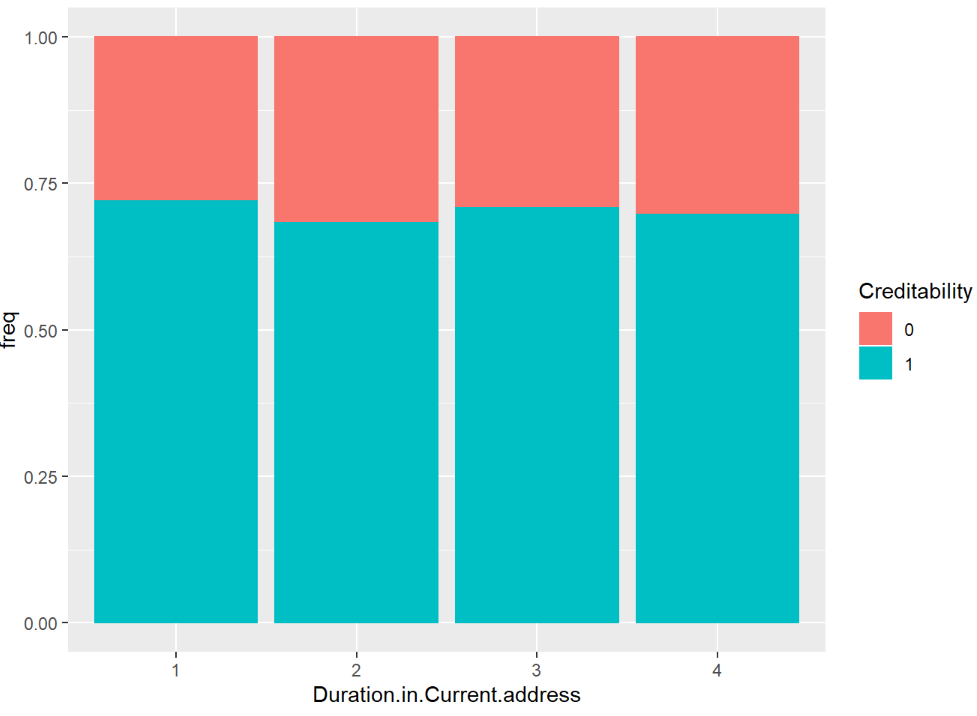
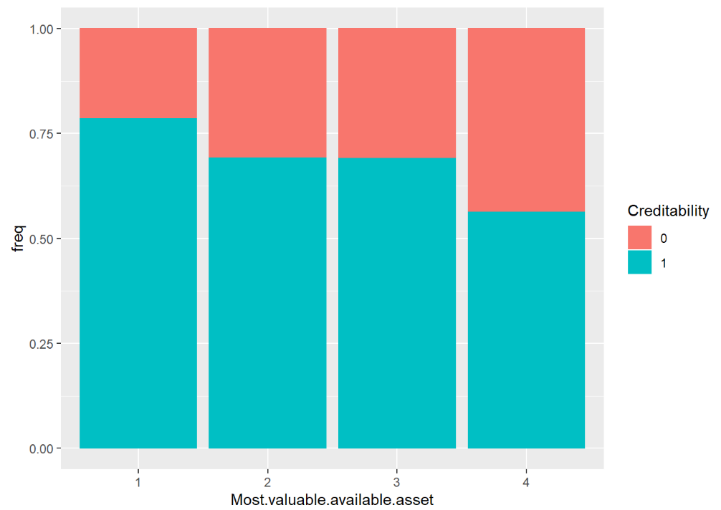
In order to have a primary evaluation of a relation between the attributes and the class, bar plots were elaborated having the percentage of each class over each level for each attribute.

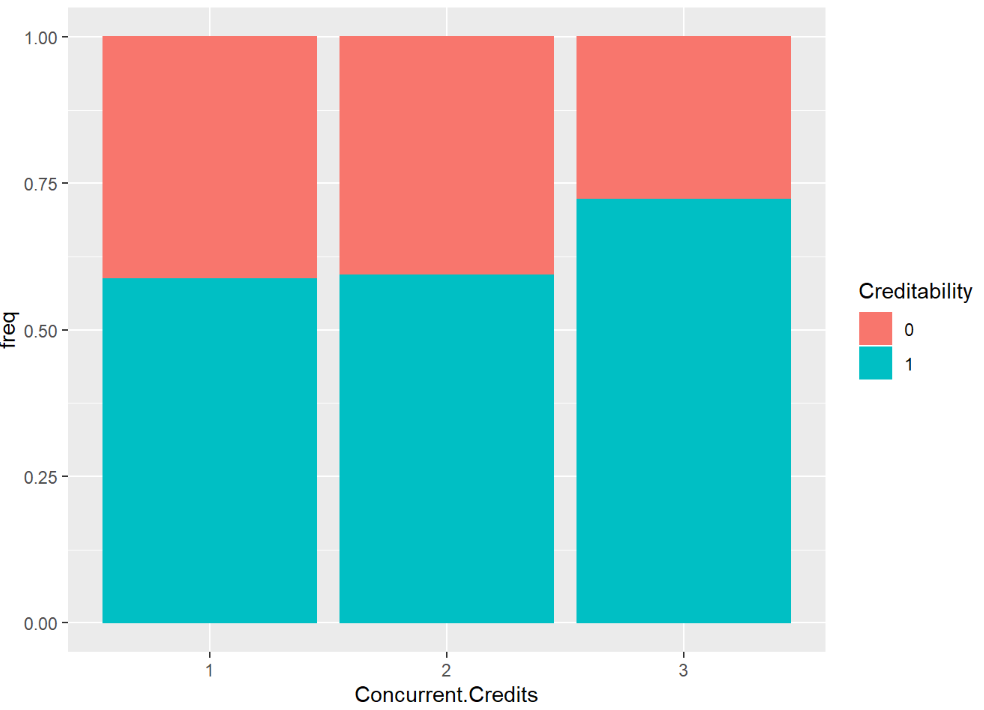
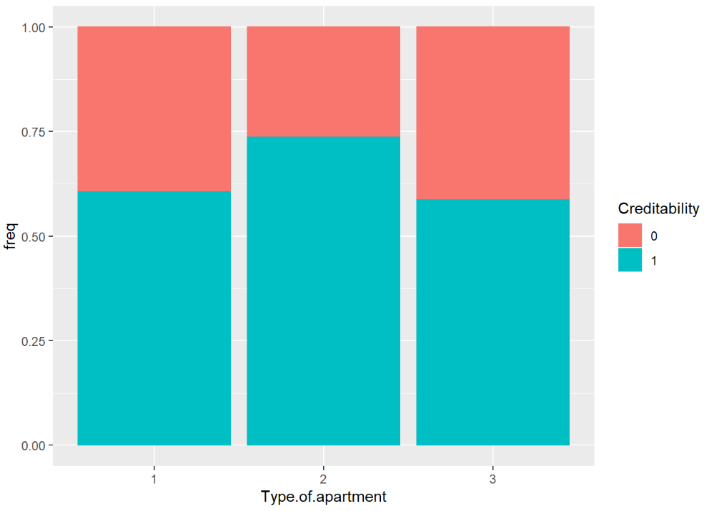


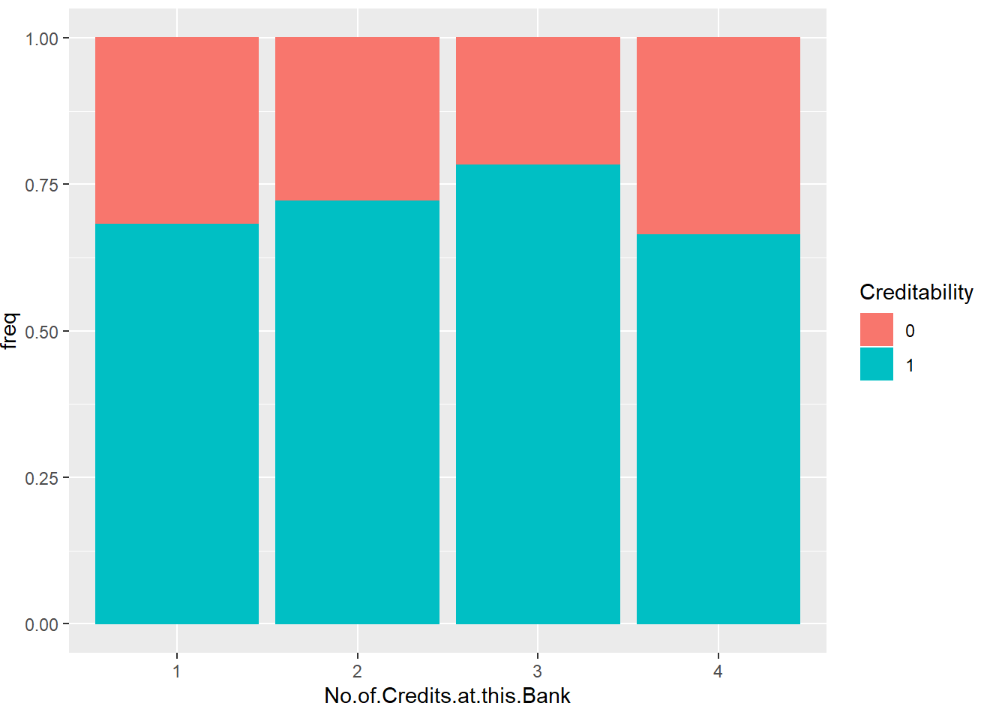
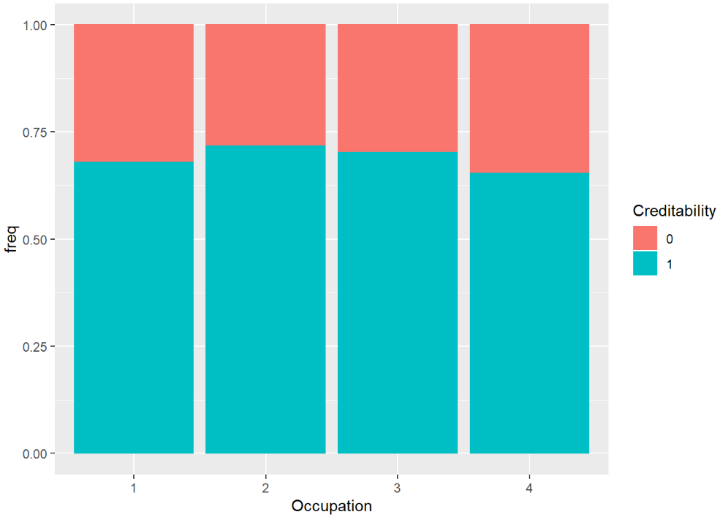


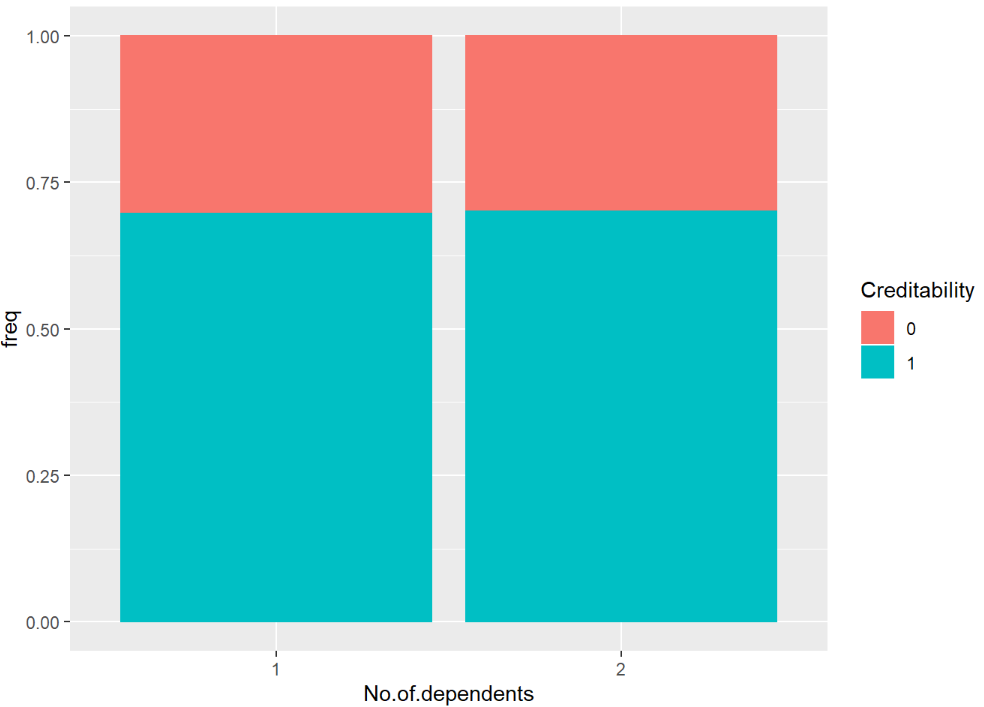
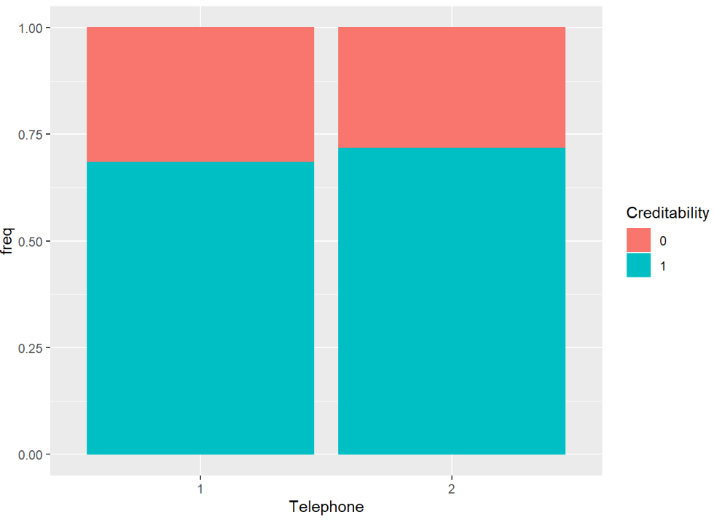
 

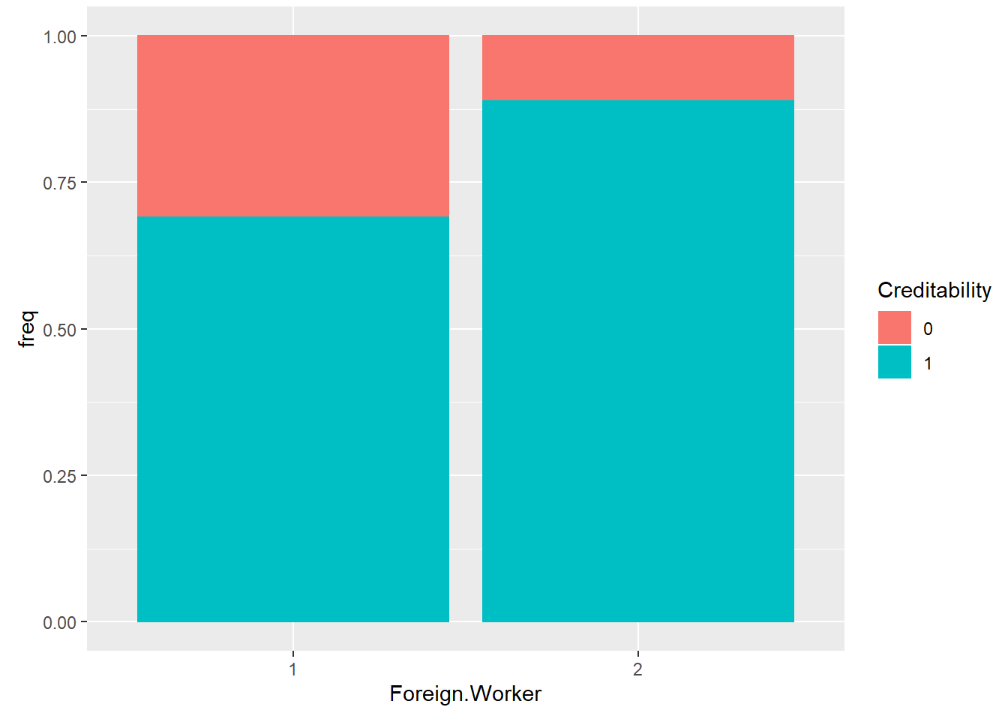












This primary analysis demonstrate that some attributes appear to have some correlation between their levels and creditability (e.g. Account Balance, Payment Status of previous Credit, Foreign Worker). Other attributes appear to have no correlation (e.g. Telephone, Number of Dependents and Duration in current Address).

With the correct data types assigned and normalization done to the numeric attributes the data is now ready for predictive modeling.

# Predictive Modeling/Classification

## Logistic Regression

### How it Works

In statistics, Logistic Regression is a method used to predict a data value based on prior observation. A Logistic Regression model predicts a dependent variable by analyzing the relationship between one or more existing independent variables. In this study, this method was used to predict if the customer is approved to a loan or not.

A picture containing sky

Description automatically generated

The model was run on Unclean and Clean Dataset using 66% split and 10-fold cross validation.

A comparison below shows three variables which are statistically significant for the predictor variable.

The output shows when people with longer duration of credits or the installment rate is high are less likely to have a loan approval. Also, people are more likely to have a loan approval when the purpose is to buy a used car or radio/television.

**66% on unclean dataset**

|  |
| --- |
| Coefficients:  Estimate Std. Error z value Pr(>|z|)  Duration.of.Credit..month. -3.286e-02 1.230e-02 -2.672 0.007534 \*\*  Purpose1 2.121e+00 5.118e-01 4.143 3.43e-05 \*\*\*  Purpose3 1.040e+00 3.195e-01 3.256 0.001129 \*\*  Instalment.per.cent -4.244e-01 1.165e-01 -3.644 0.000268 \*\*\* |

**66% on clean dataset**

|  |
| --- |
| Coefficients:  Estimate Std. Error z value Pr(>|z|)  Duration.of.Credit..month. -4.23241 0.76810 -5.510 3.58e-08 \*\*\*  Purpose1 1.53454 0.48145 3.187 0.00144 \*  Purpose3 0.91340 0.32359 2.823 0.00476 \*\*  Instalment.per.cent -0.25262 0.10604 -2.382 0.01720 \* |

**10-fold cross validation on unclean dataset**

|  |
| --- |
| Coefficients:  Estimate Std. Error z value Pr(>|z|)  Duration.of.Credit..month. -3.498e-02 1.001e-02 -3.495 0.000474 \*\*\*  Purpose1 1.789e+00 4.097e-01 4.366 1.26e-05 \*\*\*  Purpose3 9.033e-01 2.603e-01 3.470 0.000520 \*\*\*  Instalment.per.cent -2.906e-01 9.353e-02 -3.107 0.001888 \*\* |

**10-fold cross validation on clean dataset:**

|  |
| --- |
| Coefficients:  Estimate Std. Error z value Pr(>|z|)  Duration.of.Credit..month. -3.52636 0.62789 -5.616 1.95e-08 \*\*\*  Purpose1 1.64292 0.40252 4.082 4.47e-05 \*\*\*  Purpose3 0.93484 0.26001 3.595 0.000324 \*\*\*  Instalment.per.cent -0.17592 0.08712 -2.019 0.043454 \* |

### Parameters and Filter used

No parameters or filters were used for this model.

### Best Results

Four iterations were running on the dataset. The best result was obtained in 10-fold cross validation on the clean dataset, where:

* Recall: 87.67%
* Precision: 84.21%
* False Positive Rate: 44.44%
* Accuracy: 79%

## Naïve Bayes

### How it Works

The Naïve Bayes classifier is a probabilistic classifier that calculates the probability of a class occurring given an instance. This calculation is based on Bayes’ rule, which is written as:

P(B│A)= (P(A│B)P(B))/P(A)

This rule states that the probability of B (a class) given A (an instance) is equal to the probability of A given B multiplied by the probability of B, all divided by the probability of A. This classifier assumes that attributes are independent of each other.

The steps of using this classifier on this data set are:

1. Balancing the data set in order to ensured that no bias was present in the calculation of the probability.
2. Measuring the probability of each value of the class occurring.
3. Calculating the probabilities of each instance given their assigned class. Each calculation of probability in this step could then be used to predict the probability of the unclassified instances in the test set.
4. Calculating the probabilities of a class being assigned for each unclassified instance in the test set.
5. Classifying each instance in the test based on the class with the higher probability of being assigned to that instance.

### Parameters and Filter used

Before the classifier was run, the unfiltered data set indicated that 300 instances were classed as 0, and 700 instances were classed as 1. The class imbalance suggested that an algorithm could be biased towards classifying each new instance as the most common class. During the preprocessing stage, a filter was run on the data set to address this problem. The Resample filter was used on the cleaned data set. The Resample filter produces a random subsample of a dataset using either sampling with replacement or without replacement. In this case the bias factor was set to 1 so it created a subsample with replacement of 500 instances with creditability equals zero and 500 instances with replacement of creditability equals one.

Some iterations were ran with a Kernel Estimator on the numeric attributes. During the data cleaning process, the numeric attributes were found to not have a normal distribution. The kernel estimator calculates probability density for these numeric attributes through non-parametric measures; thus decreasing the error that could be generated from using parametric measures on non-normal distribution.

During the testing phase of the algorithm, iterations were tested with either 66% train-test split or 10-cross fold validations. The latter yielded better results as it increased accuracy, recall, and false positive rate.

Finally, some iterations were executed with removed attributes that showed little correlation to creditability removed, as illustrated in the data cleaning section (e.g Marital status, N0 of dependents, telephone).

### Best Results

Thirteen iterations were run on the data set. The best result yielded was from an iteration of Resampled cleaned data, with a kernel estimator used on the numeric values. A kernel estimator was used for the numeric values (Duration of Credit Month, and Age) because they did not appear to have a normal distribution. This iteration results evaluated in creditability = 1 were:

* Recall – 79%
* Precision – 78%
* False Positive Rate – 22%
* Total Accuracy – 79%

## Decision Trees

### How It works

Decision tree is one of the classification method by classification we mean automatically assign class to new observation with features(attributes), using previous observation.

The tree consists of nodes and edges. each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label. The goal is to end up with pure leafs (leafs that contain observations of one particular class).

### Parameters and Filter used

The decision tree model used was the J48 at Weka.

In order to balance the creditability, the Class Balance filter in Weka was used. The Class Balance filter reweights the instances in the data so that each class has the same total weight.

For the test options, iterations were executed with a 66% train-test split and 10-cross fold validations. The latter yielding better results.

Finally, some iterations were executed removing attributes that showed little correlation to creditability as illustrated in the data cleaning section (e.g Duration in Current address, N0 of dependents, telephone).

### Best Results

Twelve iterations were run on the data set. The best result yielded was from an iteration of Class balanced cleaned data. The best results evaluated in creditability = 1 were:

* Recall – 62%
* Precision – 67%
* False Positive Rate – 31%
* Total Accuracy – 65%

## Random Forests

### How it Works

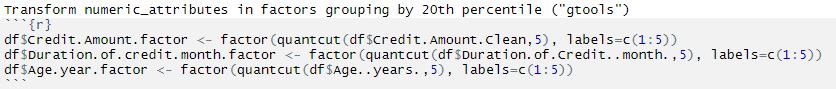
Random Forrest work with the same principle as Decision trees, by actually performing multiple decision trees at a time and outputting the class that is the mode for all decision trees in the case of classification problems. This algorithm helps correcting the decision trees’ habit for overfitting during training.

[[1]](#footnote-1)

### Parameters and Filter used

In order to balance the creditability, the Resample and Class Balancer filters in Weka were used. Being the Resample the filter that yielded the best results

Since the 10 cross fold seemed to have better results in decision trees, the same test option was applied. Some of the iterations were executed with all the attributes as categorical, this was obtained by factoring the numeric attributes in 5 levels corresponding to each 20th percentile.



Finally, some iterations were executed modifying the max depth to 10 and removing attributes that showed little correlation to creditability as illustrated in the data cleaning section (e.g Duration in Current address, N0 of dependents, telephone).

### Best Results

Of the five iteration ran the best results, when evaluated for creditability =1, were:

* Recall – 86%
* Precision – 91%
* False Positive Rate – 9%
* Total Accuracy – 88%

## Comparing Algorithms

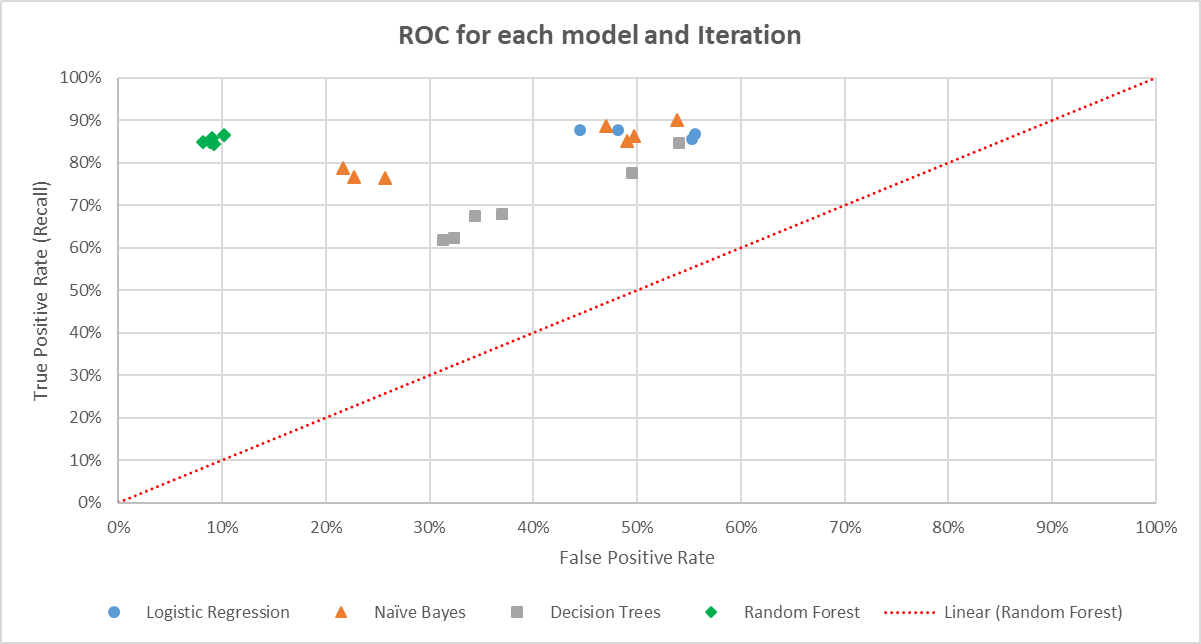
### Method for comparing

Of the metrics delivered by each iteration and algorithm, the False Positive have a more important factor in this data. That is because, since creditability is used to decide the approval of a loan, a bigger false positive rate implies in a larger risk for the Bank.

The method chosen to compare the algorithms was the ROC graph. The ROC graph indicates the false positive rate (FPR) and recall rate of an iteration of the algorithm. A point with a low FPR and high recall is observed to be better because a high ratio of recall to FPR indicates that more True Positive values are being generated than False Positive. The ROC line indicates points in which Recall and FPR are equally as likely to occur. A point that lies on or near this line is observed to be worst model because the probability of this point being a True Positive is 50%; thus the point is being randomly classified.

### Results

Given all the algorithms and the iterations that presented the largest difference, the global ROC curved is showed below:



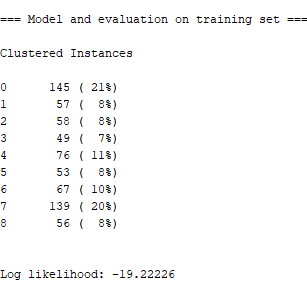
The graph above shows that for this data the random forest algorithms presented the best results.

# Post-prediction Analysis

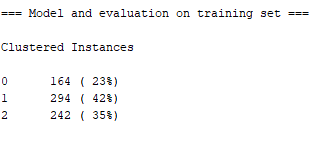
## Preparation

For a post-prediction analysis, we are focused on the instances which creditability = 1. Since most of the attributes in the data set are categorical, a further data preparation was done in order to factorized de numeric attributes in 5 levels, corresponding to each 20th percentile of the attribute. The data was filter in order to maintain only the 700 instances which creditability was equal to one.

Two clustering algorithms were used (Simple k-means and EM). When running an EM clustering in WEKA for the data, while leaving the set number of clusters to be automatically selected by cross validation, the number of optimal clusters is 9.



The problem relies that a greater number of cluster will fragmentize the clients in a way that interpreting the profiles will give highly specific results. Therefore, a number of 3 clusters was chosen since it appears to be more distributed and allows a better interpretation of each profile regarding client segmentation.



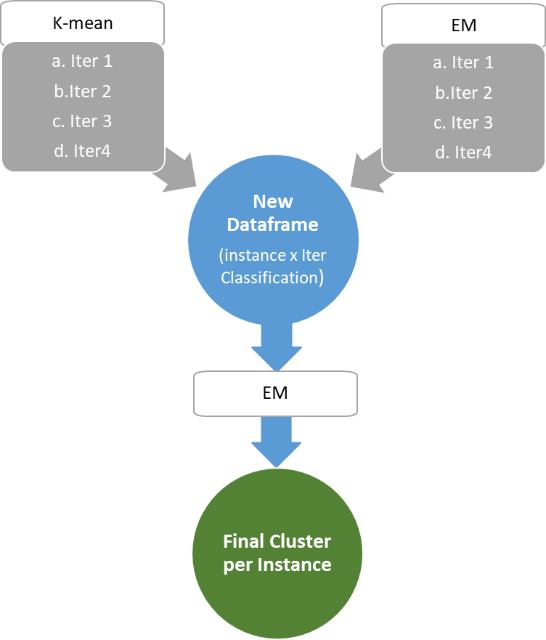
Usually k-means and EM apply only for numerical attributes. In that case, it may be necessary to convert the data set into the new data format converting categorical attributes to binary. While WEKA provides filters to accomplish all of these preprocessing, that’s not necessary for clustering in WEKA. This is because WEKA does that automatically, therefore it handles categorical values interpreting each category level in a Euclidian space[[2]](#footnote-2).

The difference between K-means and EM is that EM applies Gaussian probability for each point which allows a different cluster shape than simple K-means.

## Clustering the data

In order to minimize any bias from initial positions for K-means and EM, four iterations where run for each algorithms, modifying the seed values. After the iterations a new table was created having all 700 instances and the cluster classification for each iteration. Finally, EM was run again in this new table, therefore assigning the final clusters the majority of every iteration.

The figure bellow describes the process for extracting the final cluster assignment:



## Feature Selection

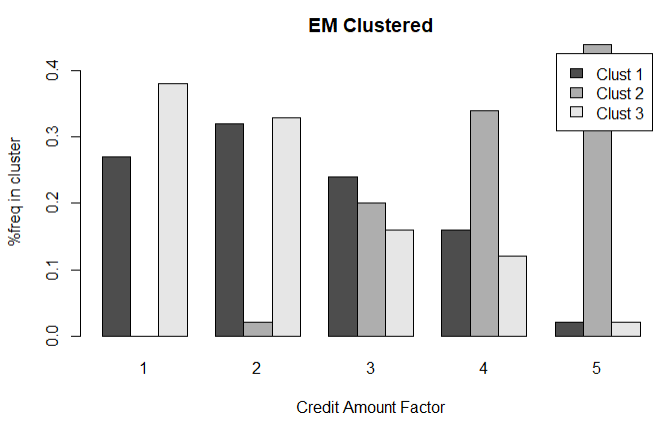
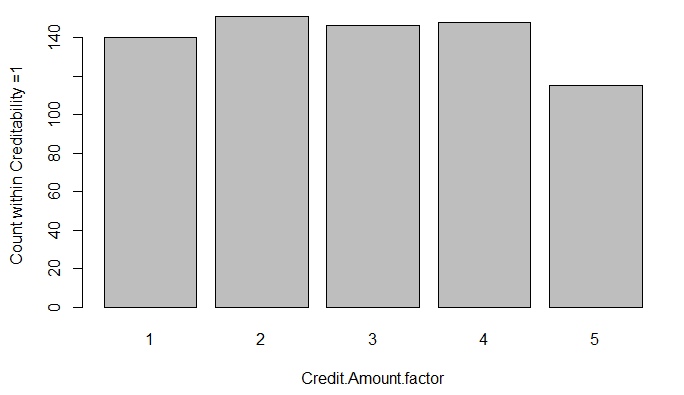
Having the cluster assignment for each instances this column was then added to the data with the remaining attributes. With cluster assignment as a new class, an attribute selection was executed in order to evaluate which attributes had more influence in the clustering classification. The attribute evaluator chosen was “InfoGainAttributeEval”.



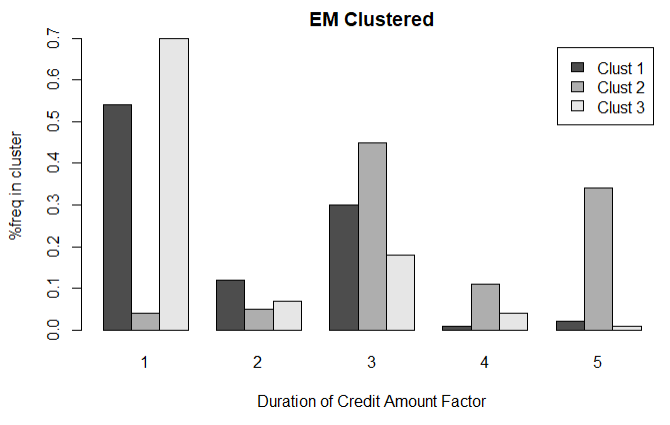
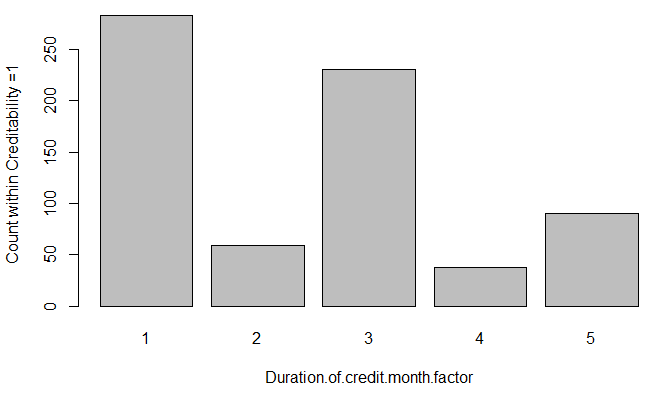
The table above show the first ten attributes ranked in order of information gain.

## Cluster Analysis

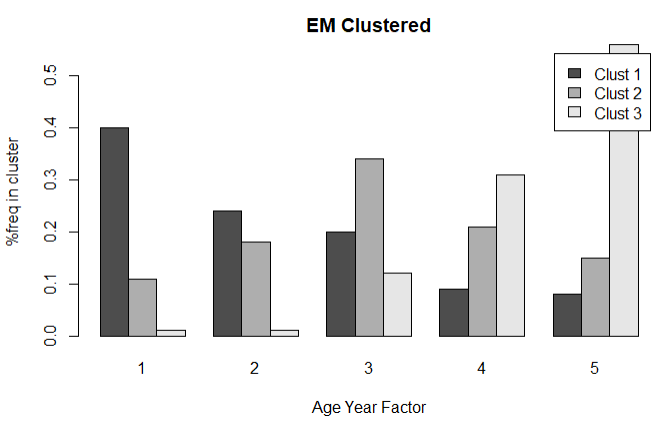
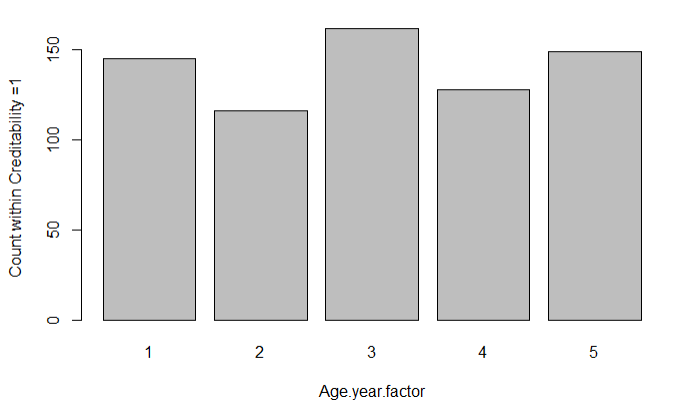
The following bar plots show the frequency of the selected attributes for each cluster under the EM algorithm, as well as an interpretation of the results:



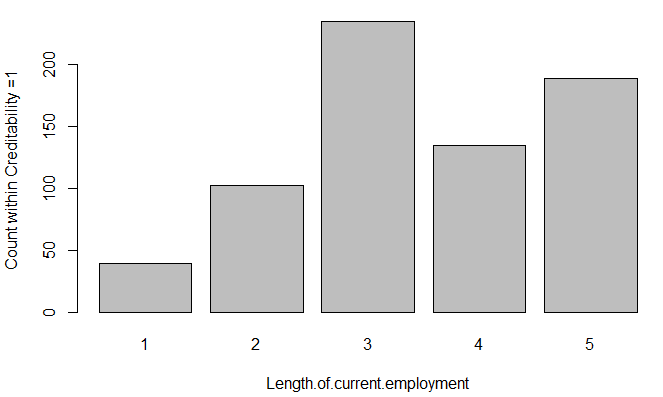
The graph on the left show that the majority of clients on Cluster 2 applied for bigger credit amount while for Cluster 3 the majority applied for lower credit amounts and Cluster 1 has a more normal distribution a little skewed towards smaller credit amount. The graph on the right shows the number of clients for each credit amount level.



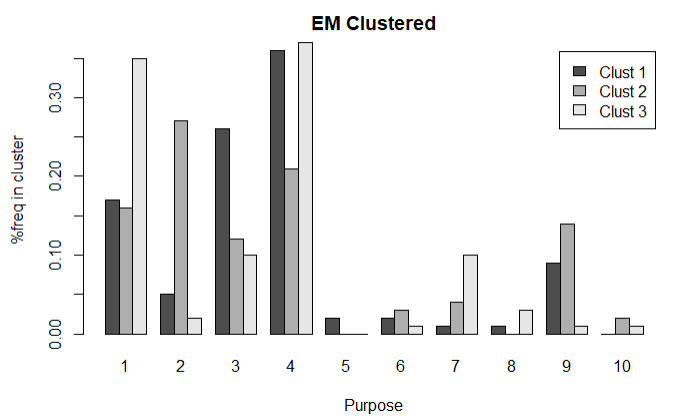
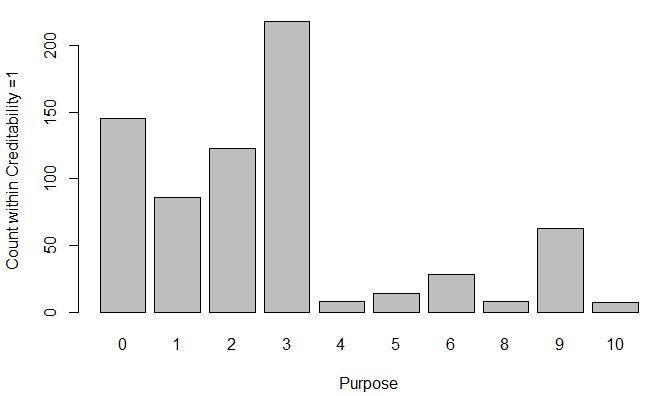
The graph on the left show that the majority of clients on Cluster 3 applied for shorter credit durations while for Cluster 2 clients applied for intermediate to long credit durations, and Cluster 1 applied for short and intermediate credit durations. The graph on the right shows that there are more clients that applied for short and intermediate credit durations.



The graph on the left show that the majority of the clients in Cluster 3 have higher age while Cluster 2 is more normally distributed towards the median of the age of the clients. Finally cluster 1 appear to have a normally distribution skewed towards younger ages. The graph on the right shows that age has a uniformly distribution within the clients.



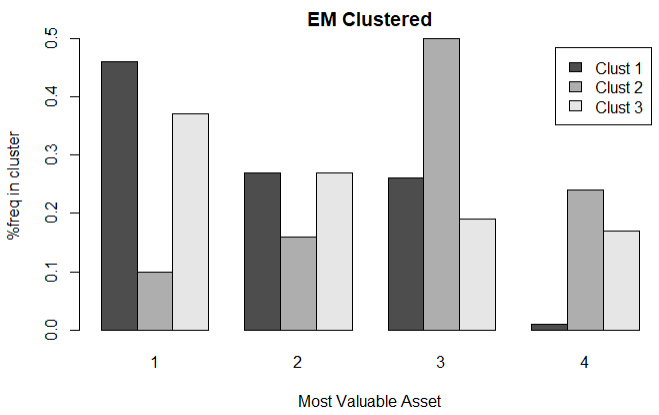
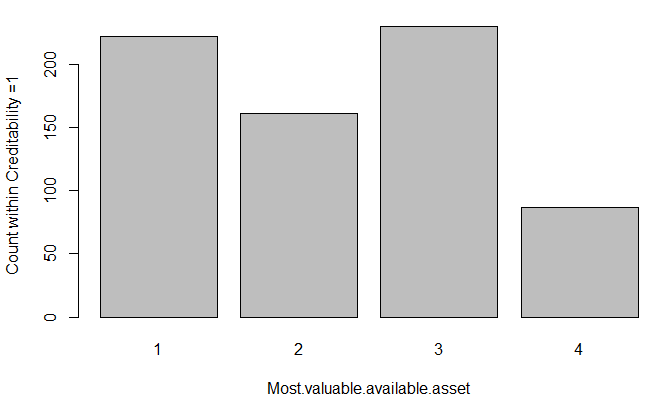
The graph on the left show that the majority of Cluster 3 is represented by clients with longer time (> =7 years), while Cluster 1 is better represented by clients with a length 4 years or less and Cluster 2 is better represented by clients with more than 1 year. The graph on the right shows that length of current employment has a better representation for clients between 1 years or more.



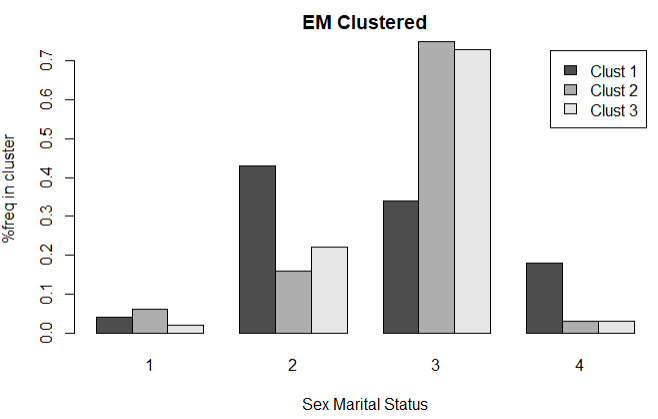
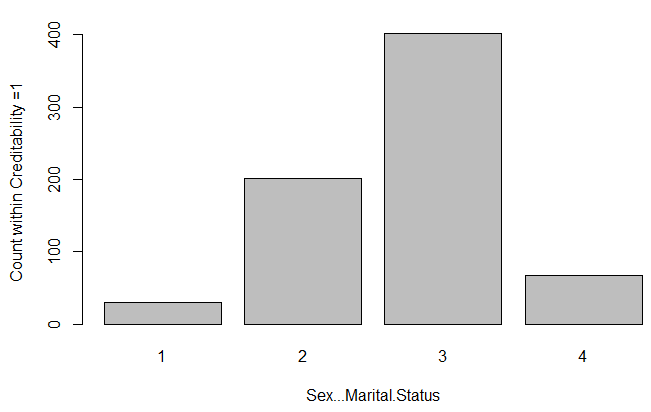
The purpose for each cluster seem to be fairly distributed, although most of the representation stands for the purpose such as:

1. Used car;
2. Furniture/Equipment
3. Radio/Television
4. Domestic Appliances.

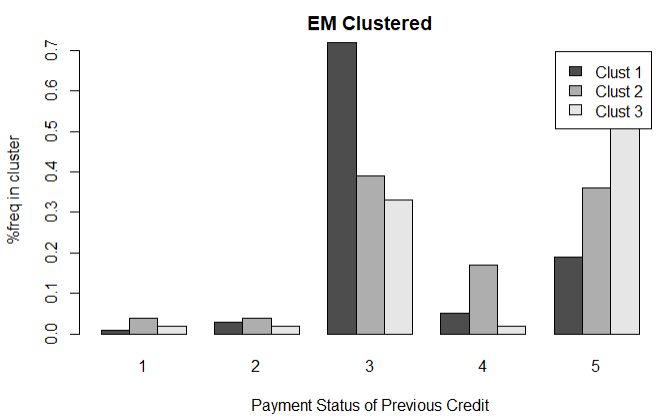
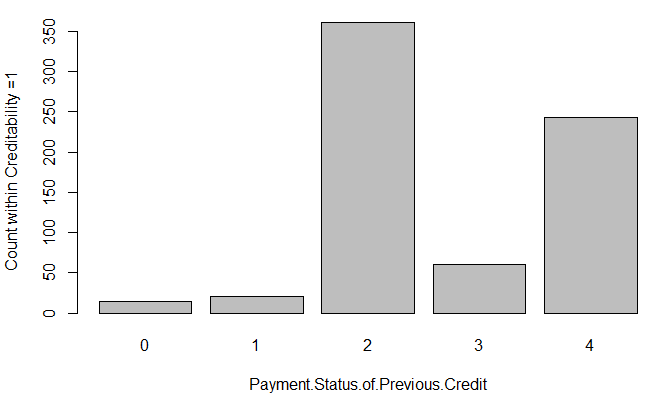
The graph on the left shows that most client have indeed one of the purposes listed above.



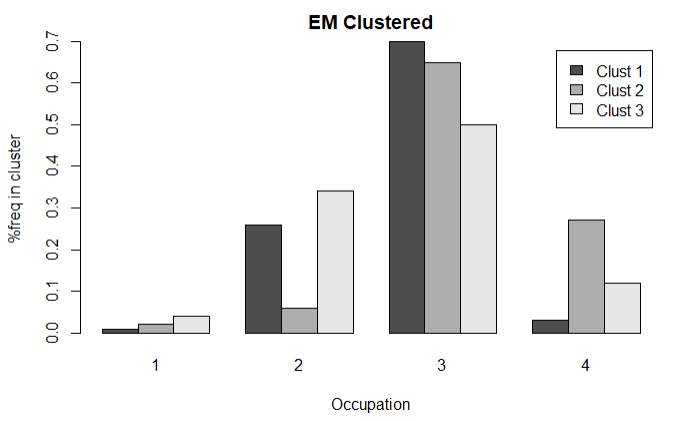
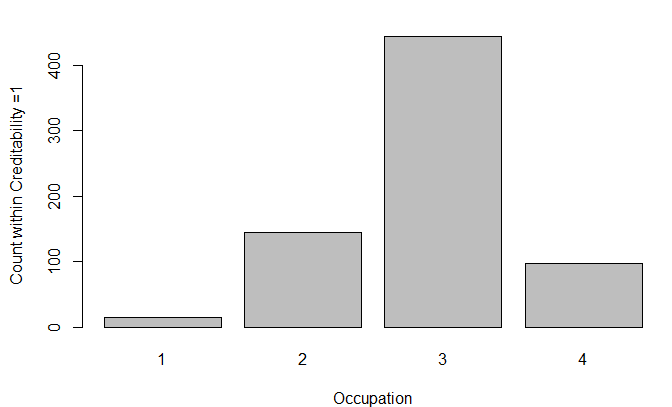
The graph on the left show that the majority of Cluster 1 is represented by clients that have better valuable assets (real estate, savings agreement/ life insurance, car or other). Cluster 2 is more represented by clients that have lower or no valuable assets. Cluster 3 is more fairly distributed within the levels. The graph on the right shows that most of the clients are between levels 1, 2 and 3.



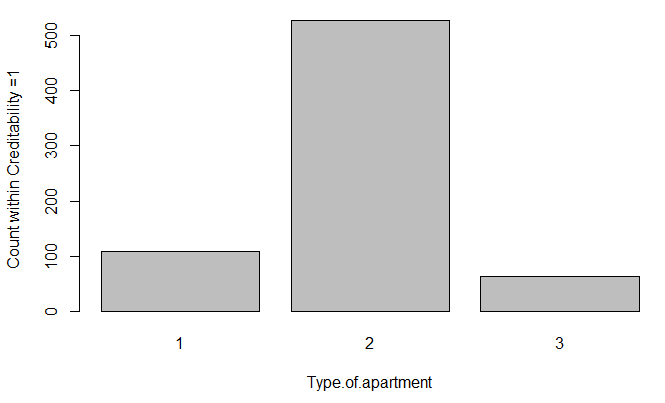
The graph on the left show that the majority of Cluster 2 and 3 are represented by clients that are single and male. Cluster 1 is more fairly distributed within clients that are single or non-single males or non-single females. The graph on the right shows that most of the clients are between levels 2 and 3.



The graph on the left show that the majority of Cluster 1 is represented by clients that existing credits have been paid back duly till now. Cluster 2 are represented by clients that either have good paid back records or have critical accounts. Cluster 3 is mostly represented by clients that have critical accounts. The graph on the right shows that most of the clients have either paid back duly till now, or have critical accounts.



The graph on the left show that the majority of Cluster 1 is represented by clients that are either unskilled resident or skilled employee/official. Cluster 2 are represented by clients that either skilled employee/official or management/self-employed/highly qualified employee/officer. Cluster 3 is most fairly distributed within levels two, three and four. The graph on the right shows that most of the clients are skilled employee/official.



The graph on the left show that the majority of clients within all clusters own their apartment. This is corroborated by the graph on the left. Although, is worth mentioning that Cluster 1 has no representation from clients with free-housing and have the greater representation from clients that pay rent.

## Client Segmentation

The results bellow indicates the following client profile for each cluster:

The graph bellow indicates the cluster size:

# Conclusions & Recommendations

### Supervised Algorithms

Out of the four supervised algorithms that were run (Logistic Regression, Decision Tree, Naïve Bayes, Random Forest), the Random Forest algorithm yielded the iterations with the optimal result. The optimal iteration of Random Forest was filtered with Resample, and was tested using 10-Fold Cross Validation. This iteration indicated 88% accuracy, 86% recall and 9% false positive rate.

While Random Forest provided optimal results, further analysis may be needed in order to decrease the false positive rate, increase the recall rate, and increase accuracy. Different algorithms may also be used; as other algorithms may yield better results than what was calculated by Random Forest.

For the client utilizations the algorithm can be further improve with more data, or, in the case the results are already within client’s expectations, this algorithm can be used to reduce the time of new customer loan analysis.

### Post-prediction

During post-prediction analysis the focus was on client segmentation, the cluster process used the combining results of k-means and EM algorithms.

The final results presented three different profiles based on the analysis of the frequency of each cluster for each attribute level.

This profiles can now be used for different business strategies such as, target marketing, increase different profiles market share or improved fees to yield more revenue.

In this scenario 3 clusters were selected for client profile. Although increasing the number of cluster can eventually fragmentize to much the clients, future analysis can be done to see if a clearer profile can be generated

1. Source: https://towardsdatascience.com/understanding-random-forest-58381e0602d2 [↑](#footnote-ref-1)
2. Source: http://facweb.cs.depaul.edu/mobasher/classes/ect584/weka/k-means.html [↑](#footnote-ref-2)