**German Credit**

**Background**

The second-oldest profession in the world, money lending has existed since the invention of money. However, the systematic assessment of credit risk is a very recent development, as lending was formerly largely dependent on reputation and incredibly sketchy data. The third President of the United States, Thomas Jefferson, was continually in debt and untrustworthy with his loan payments, but people kept giving him money anyhow. The Retail Credit Company was established to disseminate credit information only around the turn of the 20th century. Equifax, one of the top three credit scoring companies, is now that business (the other two are Transunion and Experion).

These days, the use of local and individual human judgment is mostly irrelevant in the credit reporting process. Numerous customer and transactional details are used by credit agencies and other major financial firms that issue credit to consumers to make predictions about the likelihood of defaults and other unfavorable events.

**Data**

This study focuses on a historical transition to predictive modeling at an early stage when records were classified as having good or bad credit by humans. The German Credit dataset contains 1000 records with 30 variables, each representing a previous credit applicant. In 700 cases, each applicant received a "good credit" or "poor credit" rating (300 cases). The values of these variables for the first four records are displayed in ***Table 1***. ***Table 2*** provides explanations for each variable. Additionally, based on the values of the 30 predictor variables, new credit applicants can be categorized as either good or bad credit risks.

According to assessments, the following are the effects of misclassification: The expenses of a false positive—saying an application is a good credit risk when you shouldn't—outweigh the advantages of a real positive—saying an applicant is a good credit risk—by a factor of five. ***Table 3*** provides an overview. The average net profit per loan as stated in ***Table 4*** was used to create the opportunity cost table. We use these tables to evaluate how well the various models perform since decision-makers are accustomed to thinking

Table : First four records from German Credit dataset

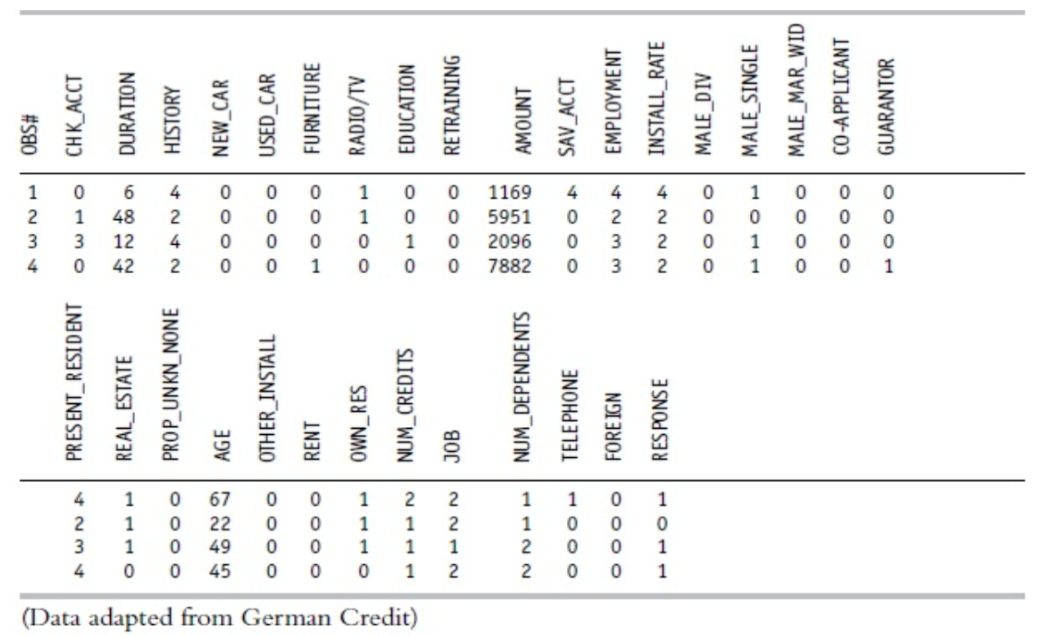


Table : Variables for the German Credit Dataset



The original dataset had a variety of category variables, some of which were converted into a collection of binary variables and others of which were retained in their original format to be handled as numerical variables. Information taken from German Credit.

Table : Opportunity Cost Table (Deutsche Marks)

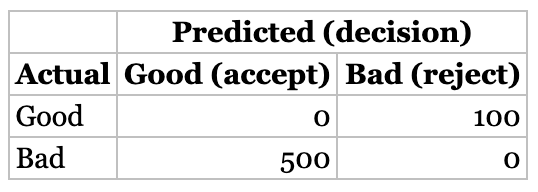
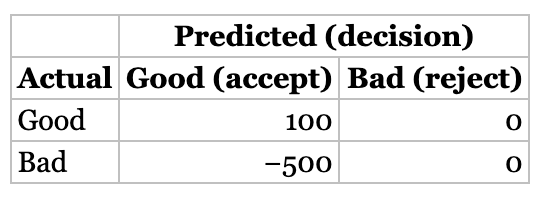


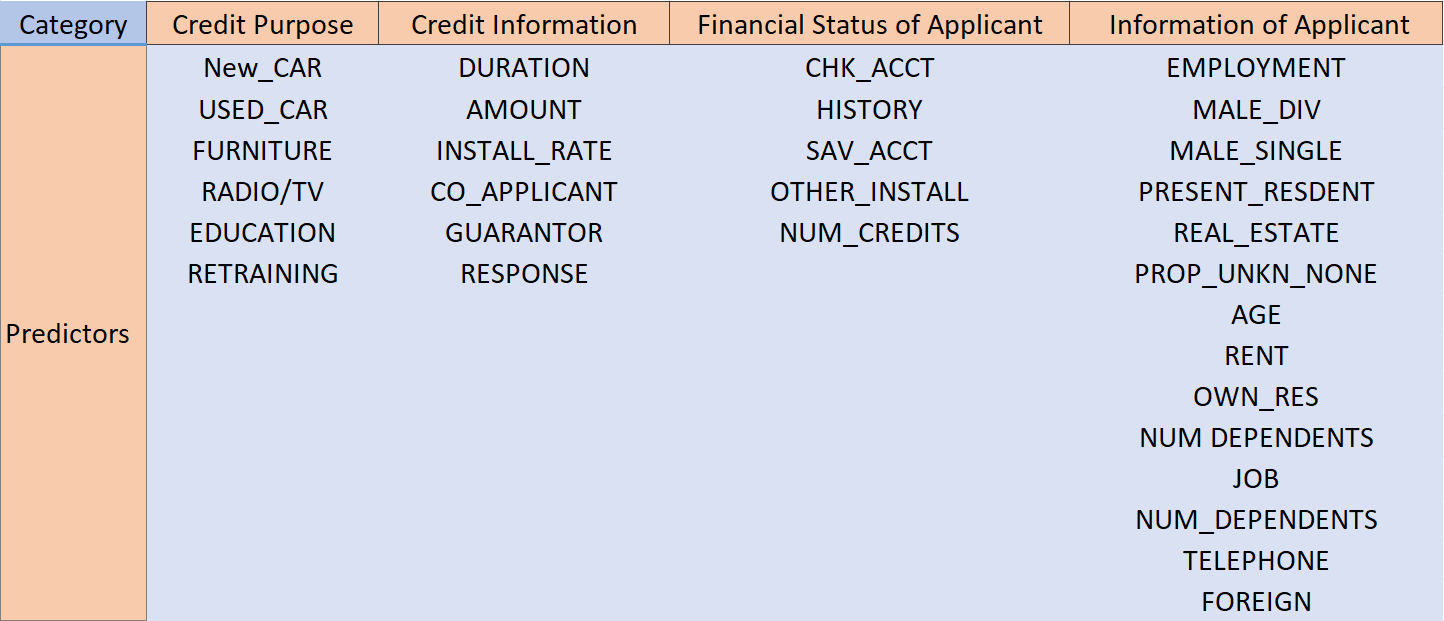
Table : Average Net Profit (Deutsche Marks)



### Assignment

**1. Review the predictor variables (descriptive statistics) and guess what their role in a credit decision might be. Are there any surprises in the data?**

**Predictor variables**



**Surprising data**

|  |  |
| --- | --- |
| **Variable Name** | **Reason / Comments** |
| AMOUNT | This variable indicates how much debt an individual already has. |
| RENT / OWN\_RES | The complement of these two is one another. It would have the same financial outcome whether they own and rent or own and don't rent. Only one will be used. |
| NEW\_CAR / USED\_CAR | As a rule, the same variable is used, therefore these variables look odd.  The owner of a used automobile also owns a brand-new vehicle. There should only be one employed for analysis, as a result. I only intend to utilize NEW\_CAR. |
| DURATION | This reveals the length of time the creditor has been paying on time or not at all. |
| CHK\_ACCT / SAV\_ACCT | The person may be able to repay more money if they have a larger income. |

**Others**

All other factors are normal and unremarkable.

**2. Partition the dataset into 60% training and 40% validation (set the seed to 12345). Develop at least two classification models of your choice. Describe the two models that you chose, with sufficient details (method, parameters variables, etc.) so that it can be replicated.**

**Model selection**

Utilizing the data mining methods of classification trees, logistic regression and KNN, in R. We also divide the data into training and validation divisions and create classification models.

**Data exploration**

After talking about the data and information other banks use to determine which customers get credits, we'll compare it to the data provided in the GC dataset and carefully examine and investigate the various variables. In addition to examining the correlations between the variables, we also want to make sure that the dataset is free of major outliers, inaccurate or misleading data, and confusing or significant outliers.

All of these factors should enable us to identify a good creditor (grant loan) or a bad creditor in the GC dataset (reject application). Checking account status and average savings account balance, two categorical variables in the dataset, may cause the results to be skewed. The highest categories, 3 and 4, were given to not having a checking account and not having a saving account or having one that is unknown. In order to resolve this issue, we changed the hierarchical order of both sets of data by designating category "0" for neither no checking account nor no savings account, respectively, and modifying the other categories as necessary.

We separate the datasets into training set and validation set by 4:6

set.seed(12345)

train.index <- sample(row.names(GC.df), 0.6\*dim(GC.df)[1])

valid.index <- setdiff(row.names(GC.df), train.index)

train.df <- GC.df[train.index, ]

valid.df <- GC.df[valid.index, ]

**Data Selection**

Therefore we decide create 2 datasets. All variables are included in the first dataset. We chose the entire dataset because we wanted an extreme point to gauge how well our additional data reduction performed. Furthermore, as was already said, the majority of the data in GC's dataset is crucial for other banks operating in the same sector.

For the second model, we decide to take out the following variables:

GC.df = subset(GC.df, select = -c(1,6,8,10,20,26,29,30,31) )

#6 We delete USED\_CAR since we think it is duplicated with NEW\_CAR, even though a new car may be more expensive than a used car on average.

#8 We delete RADIO/TV variable since we think it is repetitive compare to FURNITURE, since radio/TV can be part of the furniture.

#10 We delete RETRAINING and contain the EDUCATION. Since it is all part of the education even retraining means the age of applicants can be bigger in general. But the purpose are similar.

#26 We delete OWN\_RES since it is overlaping with REAL-ESTATE, because they all in a way indicate that whether applicants own property.

# 31.We delete foreign worker since its low observation which may cause the bias in our model.

#20/#29/#30 We delete the Present resident/Telephone/People for whom liable to provide maintenance due to their low correlation to responsive variable.

**Data Normalization**

We normalize both datasets due to the high variance created by numerical variables.

norm.values <- preProcess(train.df[, c(2,7,10,18)], method=c("center", "scale"))

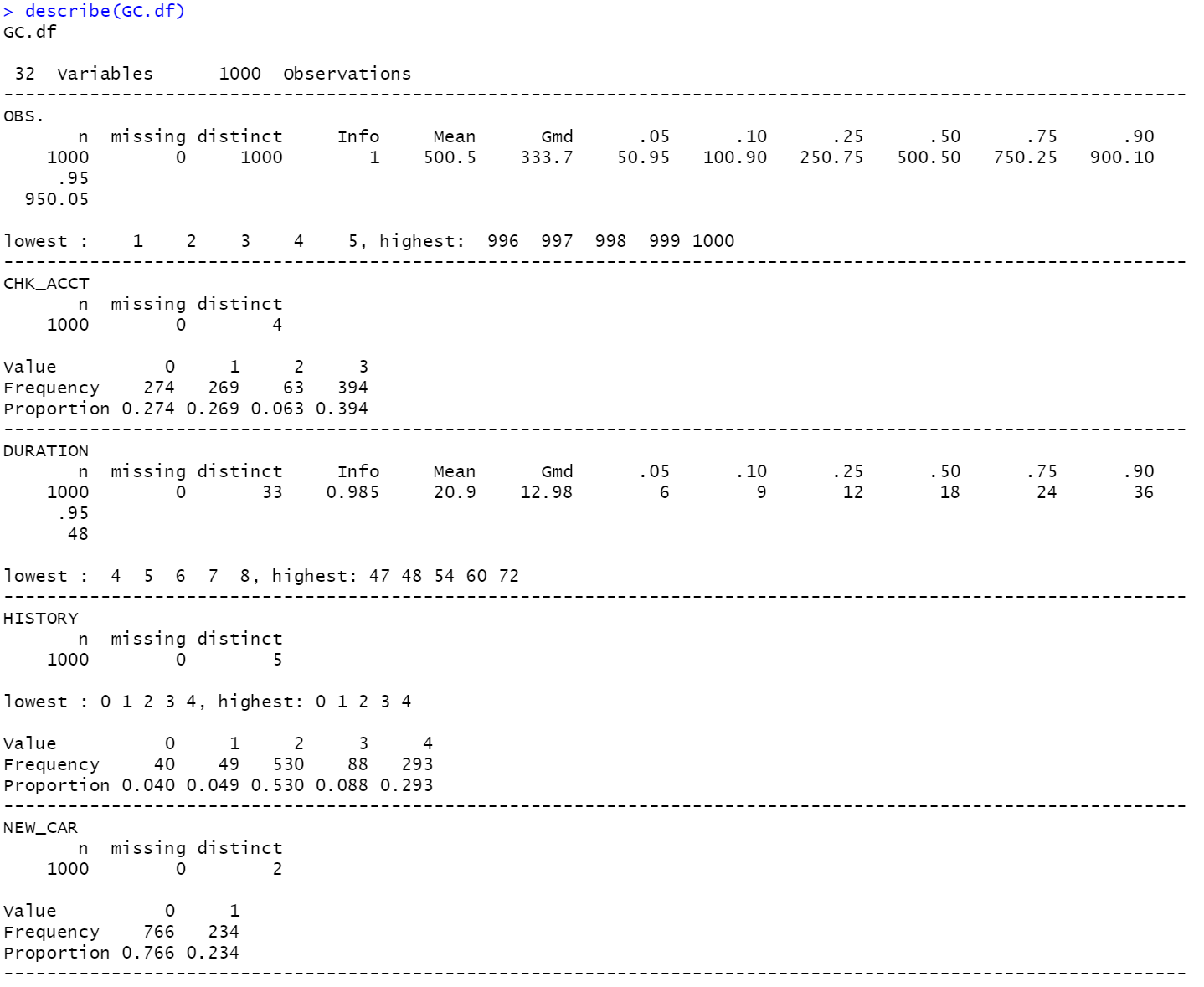
train.norm.df[, c(2,7,10,18)] <- predict(norm.values, train.df[, c(2,7,10,18)])

valid.norm.df[, c(2,7,10,18)] <- predict(norm.values, valid.df[, c(2,7,10,18)])

GC.norm.df[, c(2,7,10,18)] <- predict(norm.values, GC.df[, c(2,7,10,18)])

**Describe Classification tree and Logistic model with details**

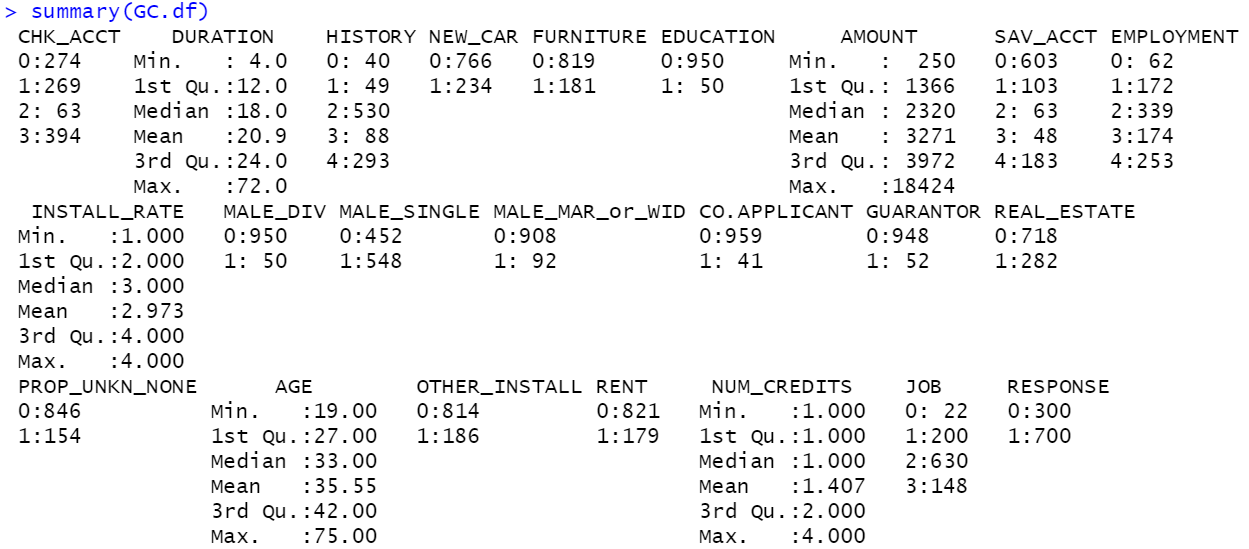
This model has 1000 observations and 32 variables like CHK\_ACCT, DURATION, HISTORY, NEW\_CAR ···, and we can use the function > describe(GC.df) to know some details about these parameter variables, as shown below:



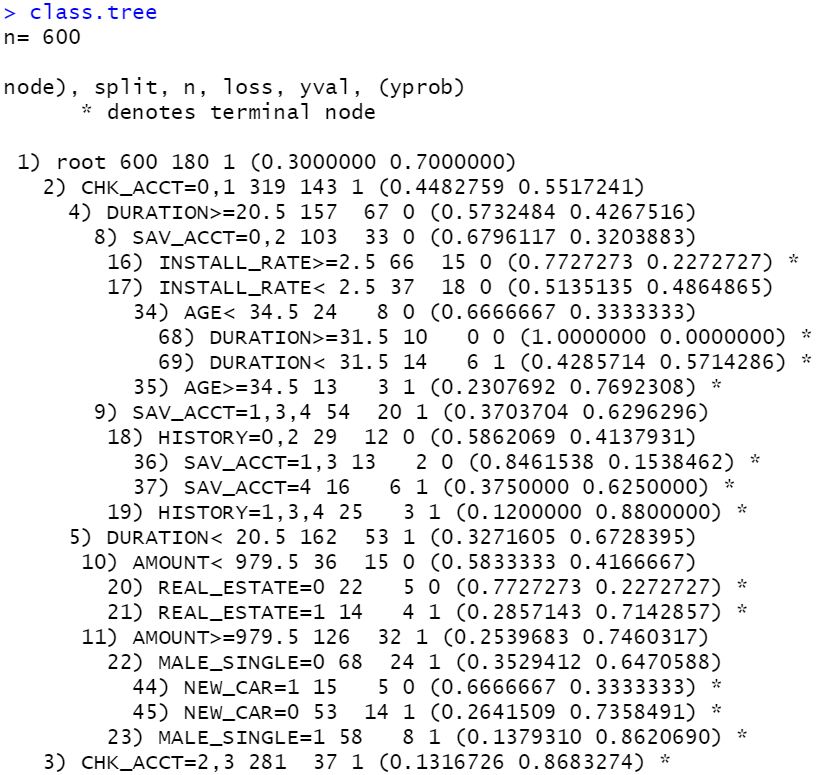
**We don't show all the information for each variable in this report, you can get all the details in the code using the method above. Through analysis, removing some irrelevant variables like OBS#, we will select 23 variables with important influence among the 32 variables as predictors.**

**Classification trees in R**

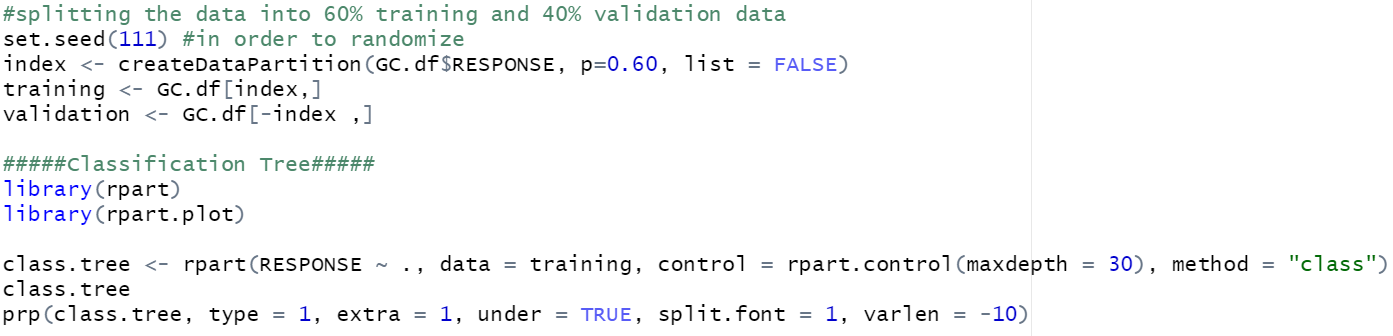
Using GC.df <- GC.df[, -c(1,6,8,10,20,26,29,30,31)], we remove some low-impact variables, and then, we can in the function summary(GC.df) , to view the specific information of the 23 variables after screening.



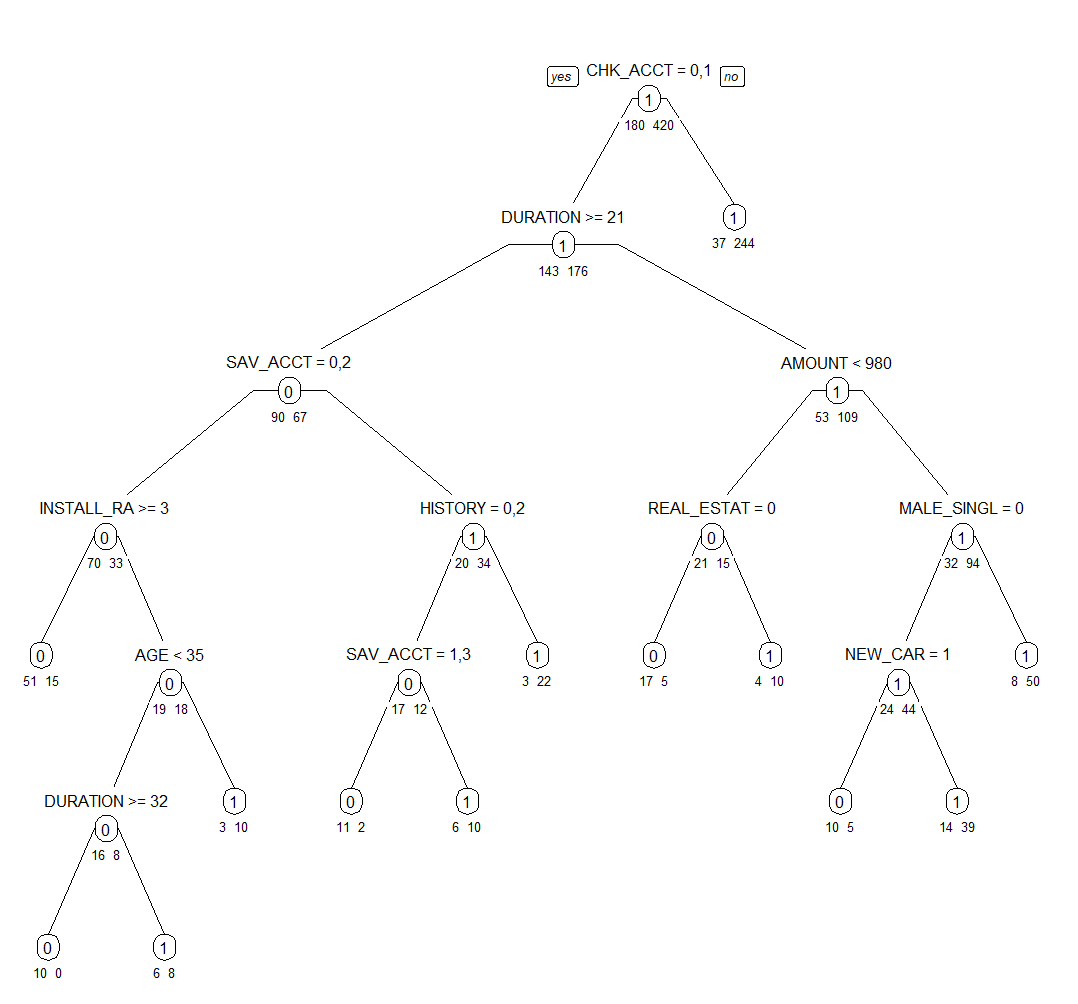
> class.tree



> prp(class.tree, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10)



**Classification Trees**



**Logistical Regression in R**

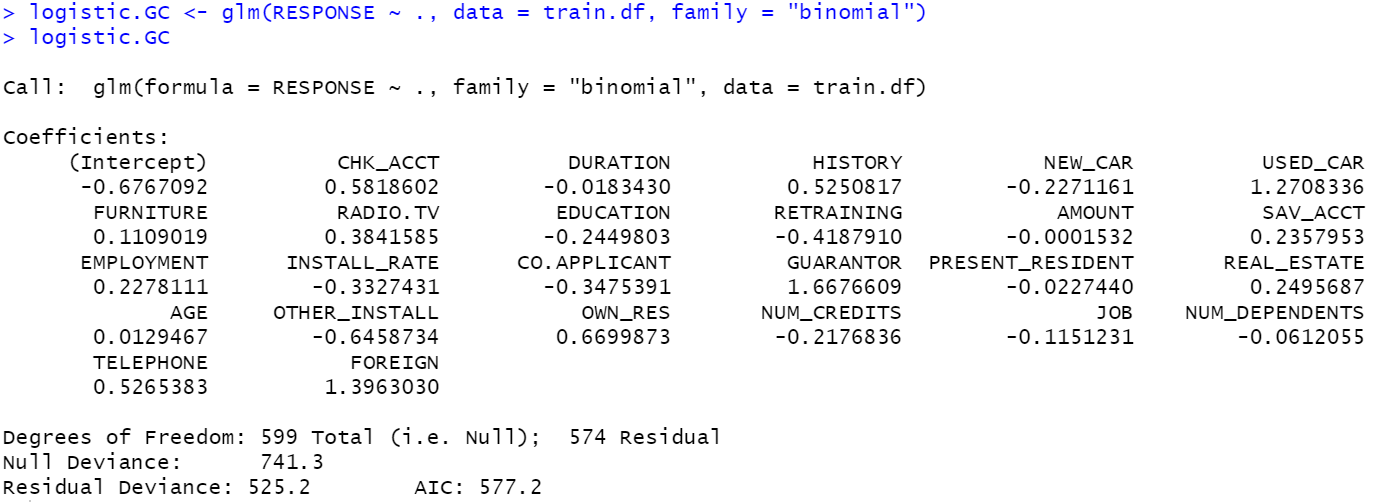
The goal of the logistic regression approach is to forecast the classification's result based on predictor factors. We do a regression analysis on our data sample to determine the correlation between the applicant's categorization and the explanatory factors, in this example, the responses to our questionnaire. The dependent variable is transformed into a probability score that indicates the likelihood that a loan will be approved for an applicant with a certain combination of qualities based on prior judgments.

**Implementation**

> logistic.GC

> summary(logistic.GC)

This way can help us know the relationship between German Credit and other factors.

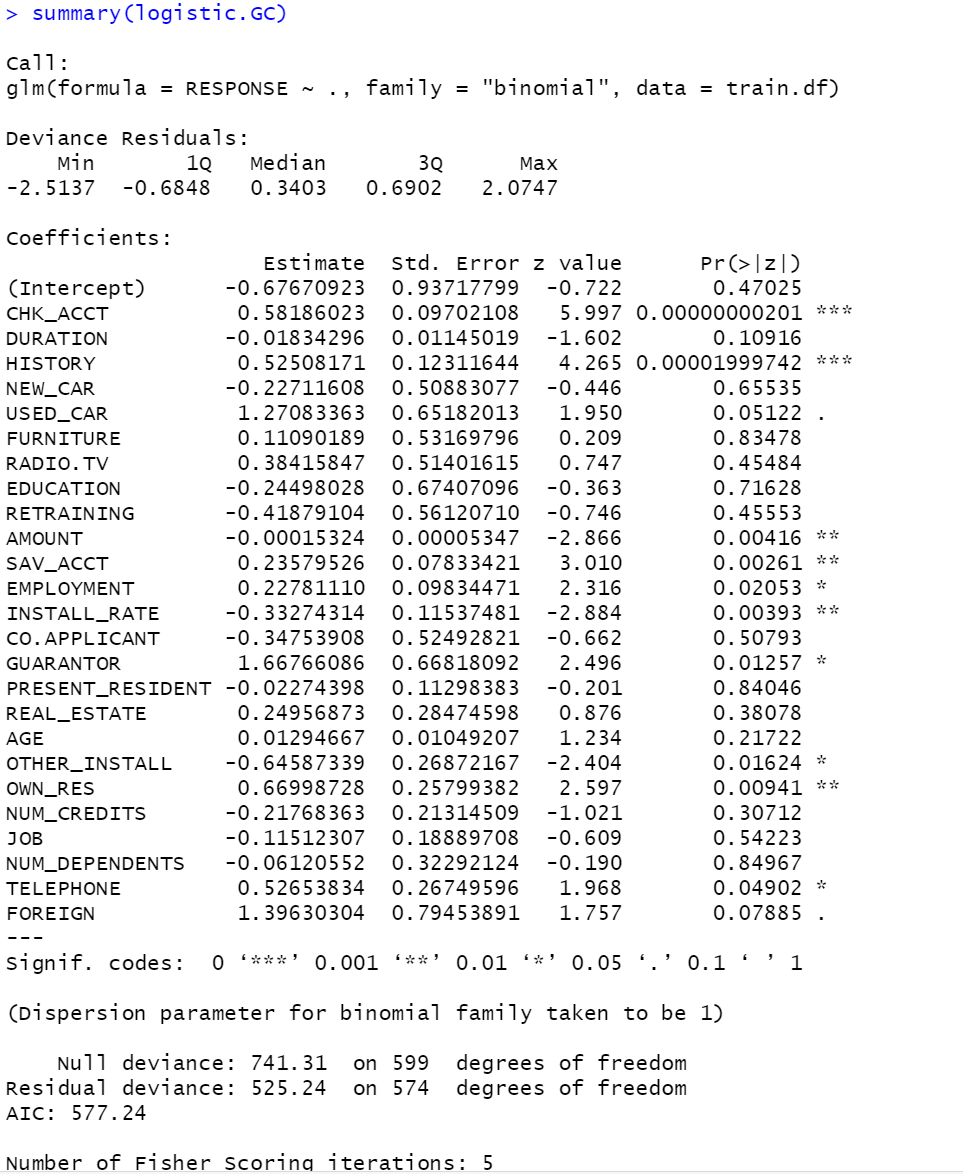


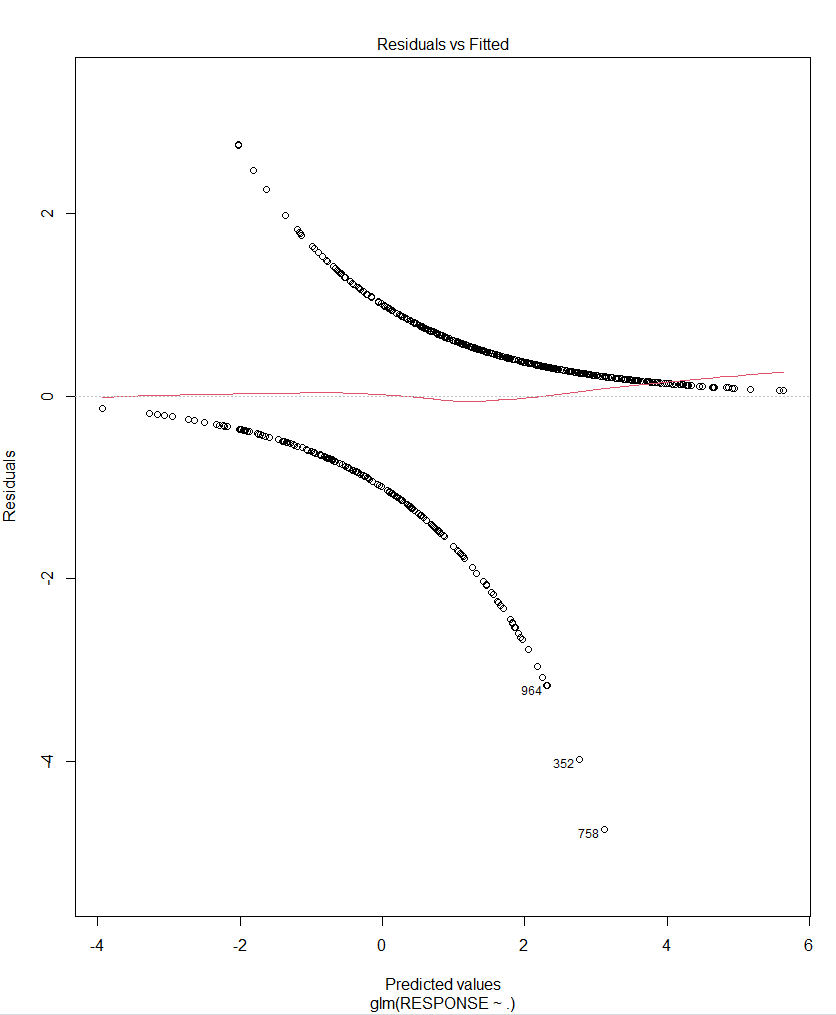
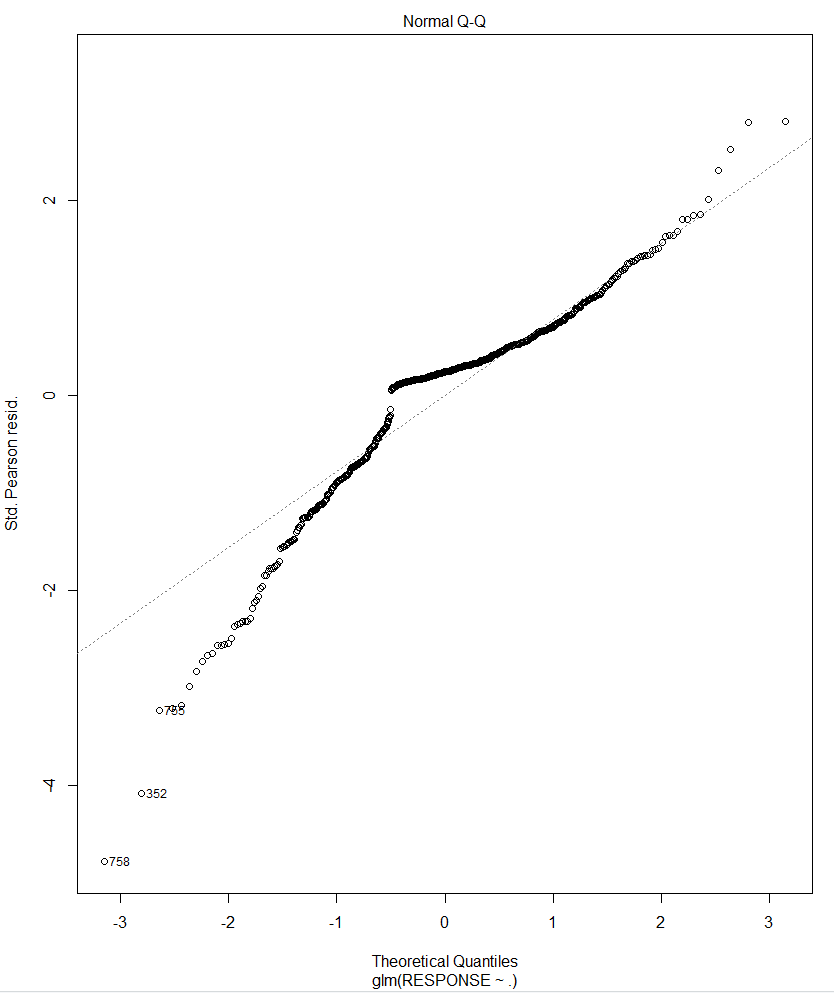
Some details to explain

Each of our models received the logistic regression treatment. The error rate and net profitability of our sample model are determined by one setscrew for our statistical approach. The cutoff value is this.

The cutoff for the likelihood of success is determined by the value, therefore admitted candidates. The likelihood of approving a candidate increases with the chosen cutoff value.

As a result, a lower cutoff value might raise the possibility of a type two mistake, which is the acceptance of a candidate without the necessary loan attributes. The best cutoff value in our report depends on the particular data sample.



**KNN Model**

When using KNN method to classify the customers, we decided to use two datasets we created before in order to get different and various insights from forming two models. And comparing the statistics between two models will help us to decide the final KNN model for prediction.

We decide to test K value from 1 to 44 for both sets and find the optimal accuracy value. And we found out the accuracy bestly achieve around 0.77. And we choose the smallest K to improve the performance.

accuracy.df <- data.frame(k = seq(1, 44, 1), accuracy = rep(0, 44))

for(i in 1:44) {

knn.pred <- knn(train.norm.df[,1:22], valid.norm.df[,1:22],

cl = train.norm.df[,23], k = i)

accuracy.df[i, 2] <- confusionMatrix(knn.pred, as.factor(valid.norm.df[, 23]))$overall[1]

}

accuracy.df

plot(accuracy.df)

**BEST K**

By generating the dataframe to track accuracy rate. We made final decesionTable

Description automatically generated with medium confidence Text, letter

Description automatically generated

**Model with First dataset** **Model with Second dataset**

We found out for the first dataset which include all variables, best K is 21

knn.pred.whole <- knn(train.whole.norm.df[,1:30], valid.whole.norm.df[,1:30], cl=train.whole.norm.df[,31], k=21)

We use K= 24 for the second model using second datasets with variables selection.

knn.pred1 <- knn(train.norm.df[,1:22], valid.norm.df[,1:22], cl=train.norm.df[,23], k=24)

**Model Selection**

A picture containing text

Description automatically generated

A picture containing text

Description automatically generated

By comparing two models we build. We found the first model has better performance.

**3. Choose one model from each technique and report the confusion matrix and the cost/gain matrix for the validation data. Which technique has the highest net profit?**

**KNN Model**

**Confusion Matrix**

Table

Description automatically generated with low confidence

**Profit/gain Matrix**

|  |  |  |
| --- | --- | --- |
|  | Predicted(decision) | |
| Actual | Good(accept) | Bad(reject) |
| Good | 100\*257 = 25700 | 0 |
| Bad | -500\*66 = -33000 | 0 |

**PROFIT:-7300**

**Logistical Regression**

**Confusion Matrix**

A picture containing chart

Description automatically generated

**Profit/gain Matrix**

|  |  |  |
| --- | --- | --- |
|  | Predicted(decision) | |
| Actual | Good(accept) | Bad(reject) |
| Good | 100\*242 = 24200 | 0 |
| Bad | -500\*63 = -31500 | 0 |

**PROFIT:-7300**

**Classification trees**

**Confusion Matrix**

A picture containing text

Description automatically generated

**Profit/gain Matrix**

|  |  |  |
| --- | --- | --- |
|  | Predicted(decision) | |
| Actual | Good(accept) | Bad(reject) |
| Good | 100\*245 = 24500 | 0 |
| Bad | -500\*63 = -31500 | 0 |

**PROFIT:-7000**

**In conclusion, the Classification tree model has the highest profit which is -7000. Though they are all negative**