

**FOUNDATION OF DATA ANALYTICS**

**PROJECT REPORT**

**CREDIT RISK ANALYSIS**

**Team members:**

**S V S AKHIL RAMAN 19BCE1564**

**M JAYANTH 19BCE1798**

**INTRODUCTION**

The purpose of credit scoring is to classify the applicants into two types: applicants with good credit and applicants with bad credits. When a bank receives a loan application, applicants with good credit have great possibility to repay financial obligation. Applicants with bad credit have high possibility of defaulting. The accuracy of credit scoring is critical to financial institutions profitability. Even 1% improvement on the accuracy of credit scoring of applicants with bad credit will decreases a great loss for financial institutions.

This study aims at addressing this classification problem by using the applicant’s demographic and socio-economic profiles of **German credit data** to examine the risk of lending loan to the customer. We assessed the performance of different Machine learning algorithms (Logistic regression model, Decision tree, random forests, Support vector machines, neural networks) in terms of overall accuracy. For the model optimization, we conducted a comparative assessment of different models combining the effects balanced accuracy and the area under the ROC curve (AUC) values.

.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sr.no** | **CONTENTS** | **Page No** |
| **1** | Objective and Scope | 4 |
| **2** | DATASET | 5 |
| **3** | Tools and Techniques | 6 |
| **4** | Analytic approach | 7 |
| **5** | Data description and preparation | 8 |
| **6** | Exploratory data analysis | 10 |
| **7** | Prediction models   * Logistic Regression * Support Vector Machine * Decision Tree * Random Forest * Neural Network | 26  32  35  37  40 |
| **8** | Conclusion | 41 |
| **9** | References | 42 |

**OBJECTIVE AND SCOPE OF PROJECT**

**1.Object**

* To minimize loss from the bank’s perspective, the bank needs a decision rule regarding who to give approval of the loan and who not to. An applicant’s demographic and socio-economic profiles are considered by loan managers before a decision is taken regarding his/her loan application.

**2.Scope**

* We will be looking into a Credit dataset
* We will derive relations between a person socio-economic life style with his credit ratings
* Based on these trends and patterns we will analysis the data
* Using Models, we will be able to predict whether a client is creditworthy or not

**DATASET**

The German Credit data is a dataset provided by Dr. Hans Hofmann of the University of Hamburg. It’s a publicly available from the UCI Machine Learning repository at the following link: <https://archive.ics.uci.edu/ml/datasets/Statlog+(German+Credit+Data)>.

Getting a glimpse of German data, we observe it’s a data frame containing 21 variables for a total of 1,000 observations. The response variable (**or outcome**) in the dataset corresponds to the **Class** label, which is a binary variable indicating credit risk or credit worthiness with levels Good and Bad. Here, we are going to describe the 20 features and their characteristics:

* Credit rating
* Account balance
* Credit Duration
* Previous Credit
* Credit Purpose
* Credit Amount
* Saving
* Employment duration
* Installments
* Marital status
* Guarantor
* Residence duration
* Current assets
* Age
* Other credits
* Apartment type
* Bank credits
* Occupation
* Dependent
* Telephone
* Foreign worker

**TOOLS AND TECHNIQUES**

We have used the following Analytical techniques / methodology for analysing the Data:

1.Summary of Statistics for each variable

2.Identification of frequency of standard violation for each of the factors

3.Using Graphs and Box Plots to visually represent them

4.Identification of significant Metrological factors through correlation and regression methodology

5.Using Multiple Linear Regression & Neural Network for Model Development

6.Tools used: R, & Excel

7.Techniques: Box Plot, Histogram, Bar Chart, Line Chart, Infographics, Visual Clues, Correlation Matrix, Multiple Linear Regression, Artificial Neural Network

8.We have used R Programming environment and Microsoft Excel for our analysis

**ANALYTIC APPROACH**

The Analytical Approach will involve the following (not necessarily in the order) activities:

•Data extraction from Primary Data source as well as secondary data sources

•Data quality check

•Data cleaning and data preparation

•Study each of the variables by exploring the data

•Study the variables for its relevance for the study

•Identifying Y variable(s).

•Performing Univariate analysis for all variables

•Division of data into train and test

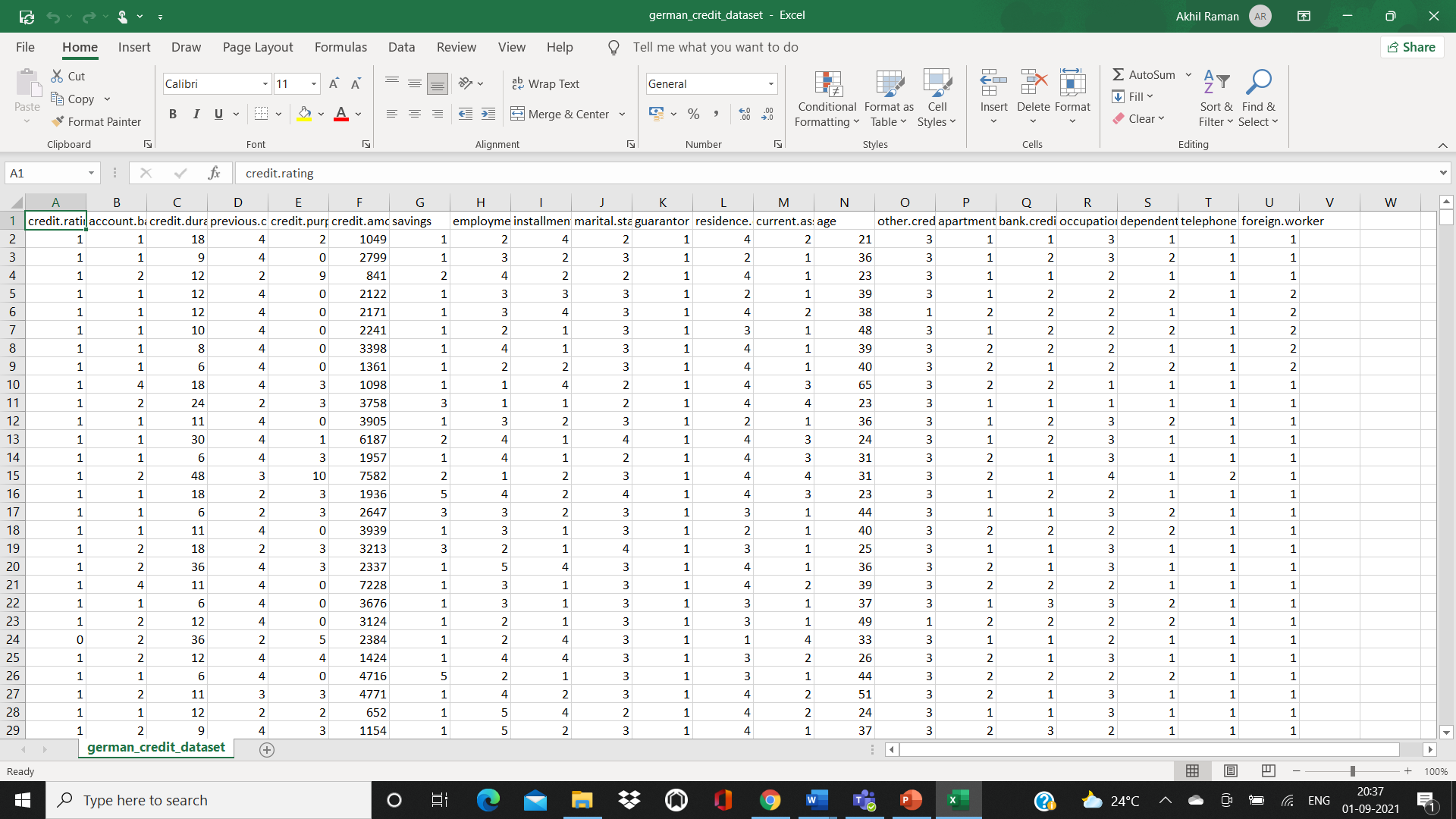
•Model Development

•Final Model

•Model Validation & Model Validation on Test

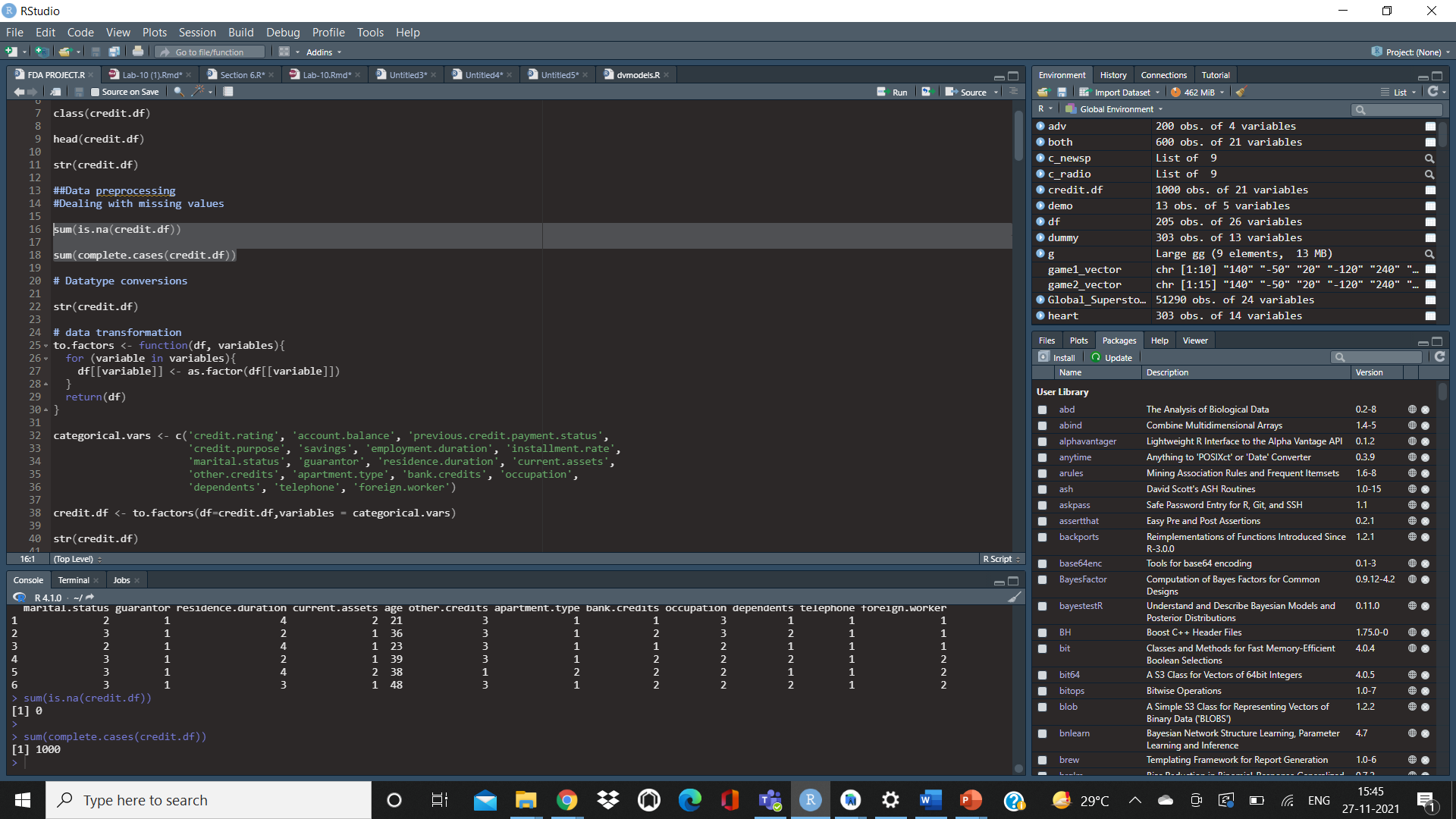
**Data Description and Preparation**

Glimpse of our dataset

****

* DEALING WITH MISSING VALUES

The German credit dataset has no null values, so we can continue to data preparation



* DATA PREPARATION

Variable Transformation

Data transformation is the process of changing the format, structure, or values of data. For data analytics projects, data may be transformed at two stages of the data pipeline.

For our dataset we have to convert all the categorial variables to factors and the numerical values are kept the same

Categorial variables include : credit rating , previous credit payment status , savings , employee duration , instalment rate , marital status , guarantor , residence duration , current assets , other credits , apartment type , bank credit , apartment type , occupations, dependency , telephones, foreign worker

Numerical variables include: credit amount, credit duration months , age

# 

# **EXPLORATORY DATA ANALYSIS**

To get a better understanding of our dataset and the interdependences of the different variables in our dataset we will perform exploratory data analysis, by the end of this section we will remove the unnecessary attributes from the dataset reduce the factors in certain attributes to improve our prediction models

# 

# 

# 

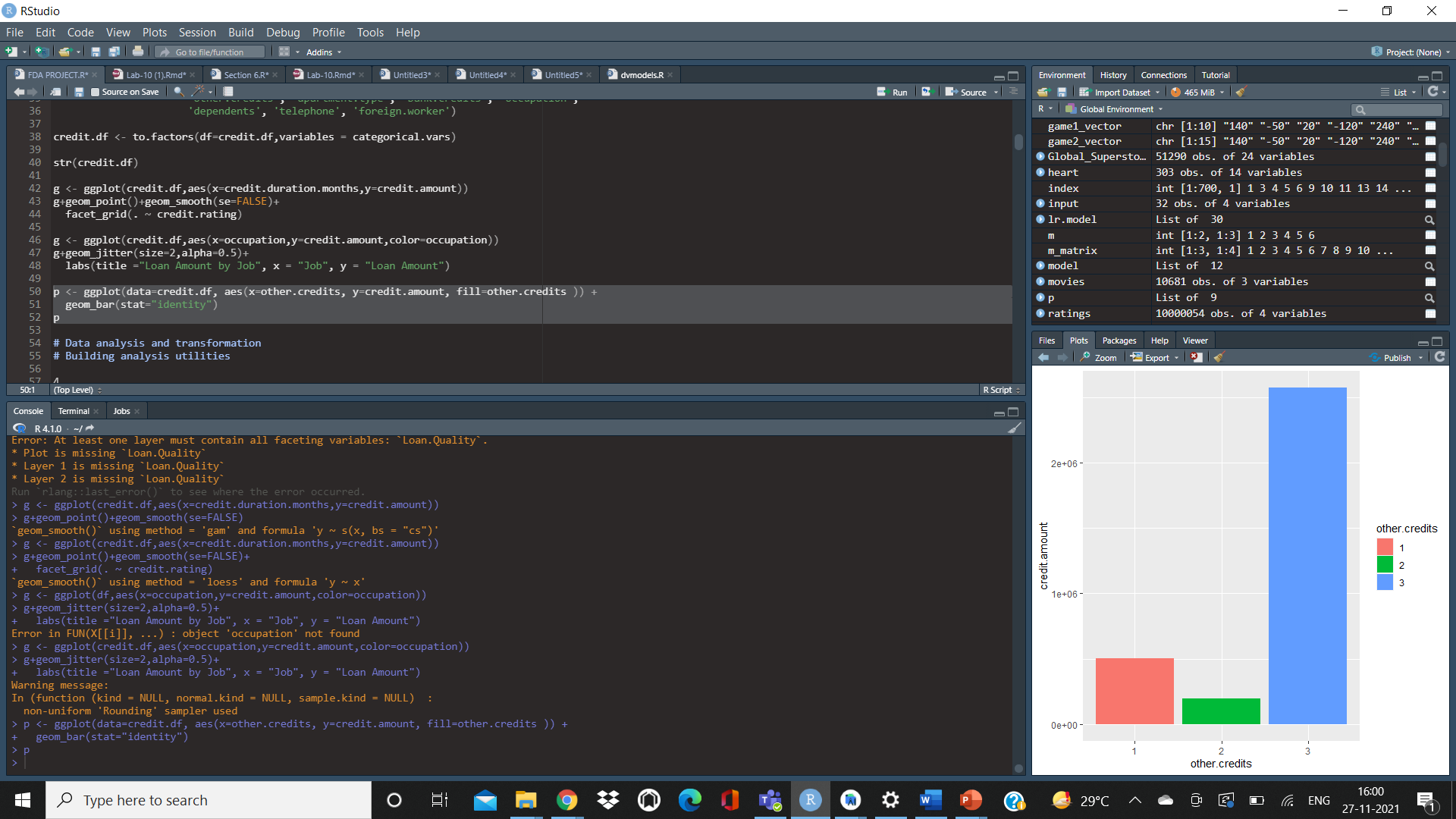


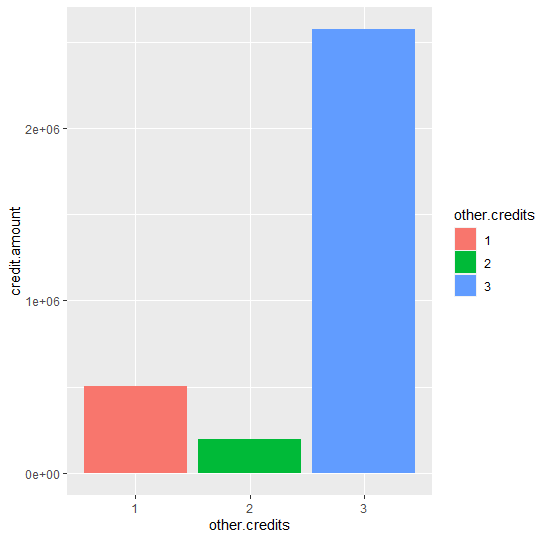
**INFERENCE:**

In occupations

1. Unemployed with no permanent residence
2. Unskilled with permanent residence
3. Skilled workers/civil servants
4. Executive/self-employed/higher civil servants

From the plot we can observe that people under category 3 are taking higher number of loans and higher loan amounts





**INFERENCE**

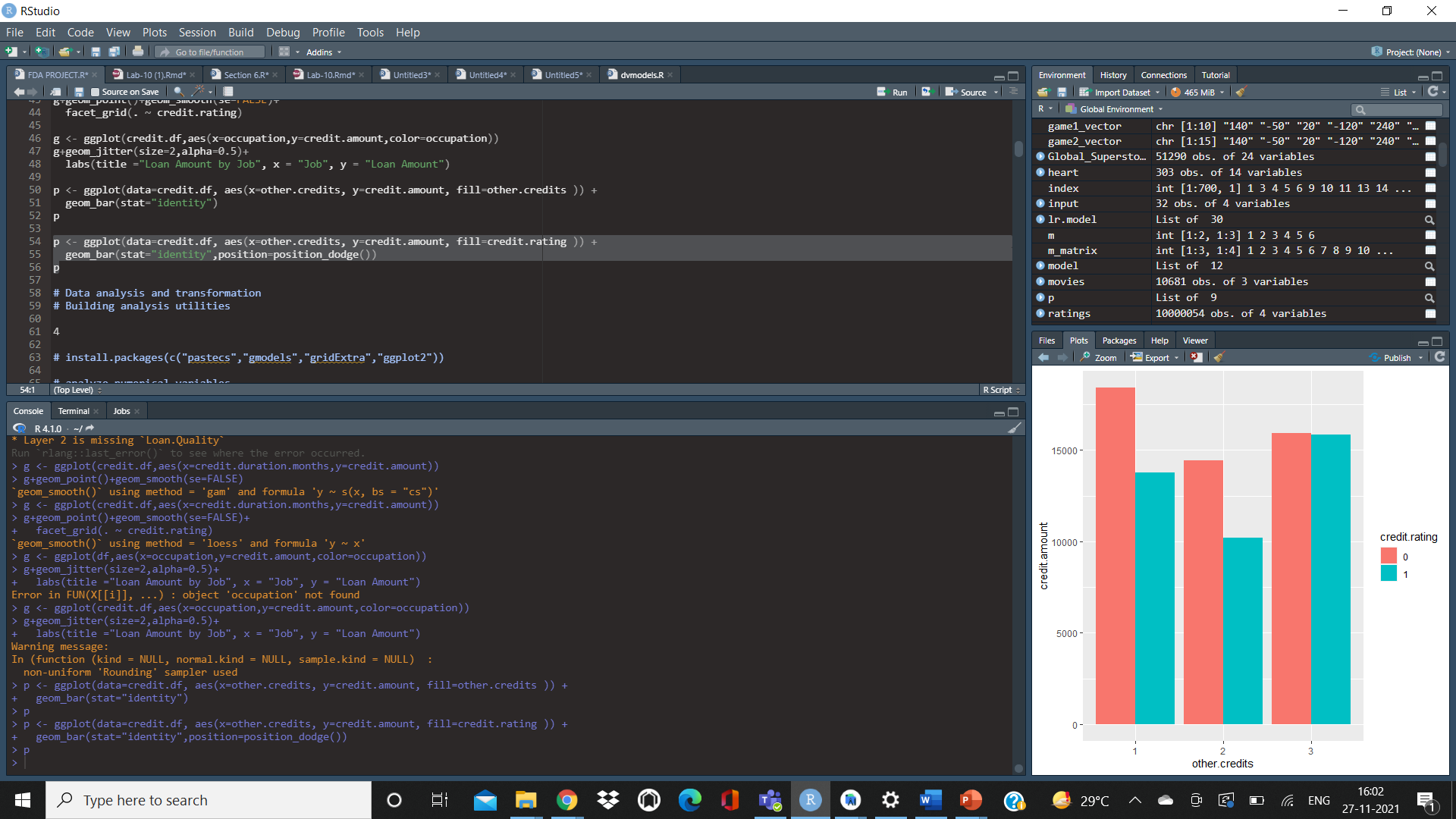
in other credits:

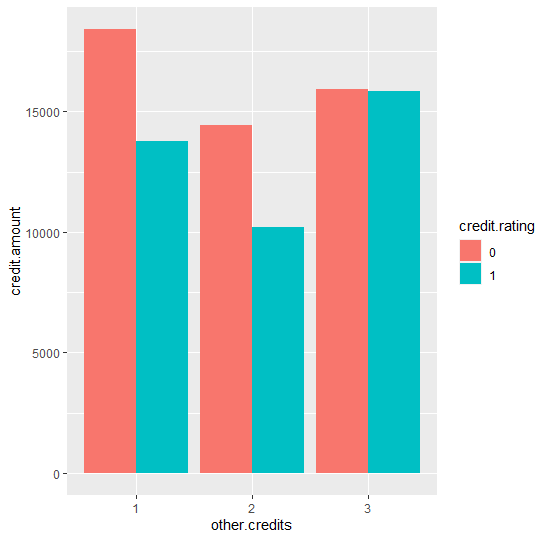
1-Credits with other banks

2-In stores

3-No further credits

We can observe in our dataset most of the clients have no further credits

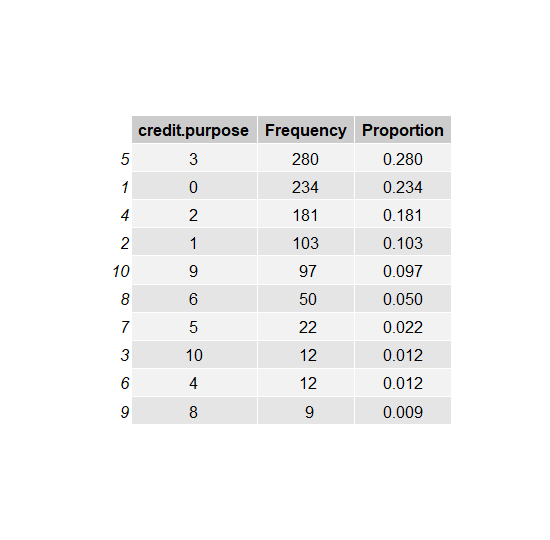




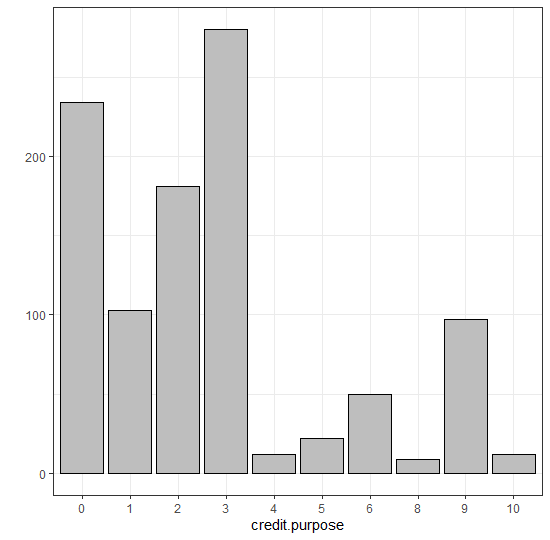
**INFERENCE**

In our dataset clients having other credits at different banks are more in the case of bad credit rating similarly for credits in stores and there is a balance in clients with good and bad credit ratings who have no further credits

**CREDIT PURPOSE**



German credit dataset consists of 10 different credit purposes, we will conduct data transformation and merge these credit purposes to improve our prediction models



0-Others

1-new car

2-used car

3-furniture items

5-Household applications

6-repair

7- (no significance)

8-vacation

9-training

10-business

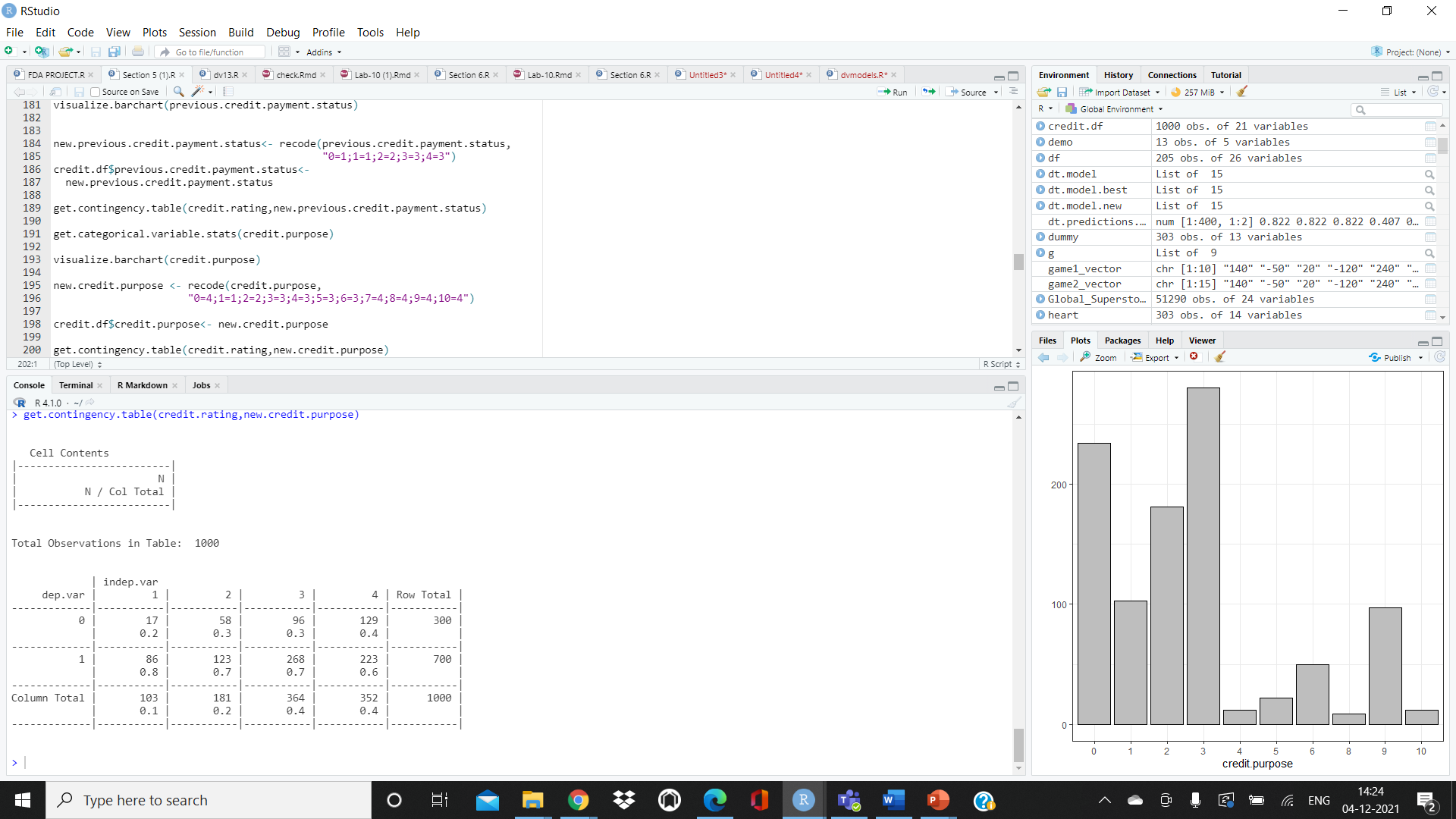
After performing transformation, the related contingency table is

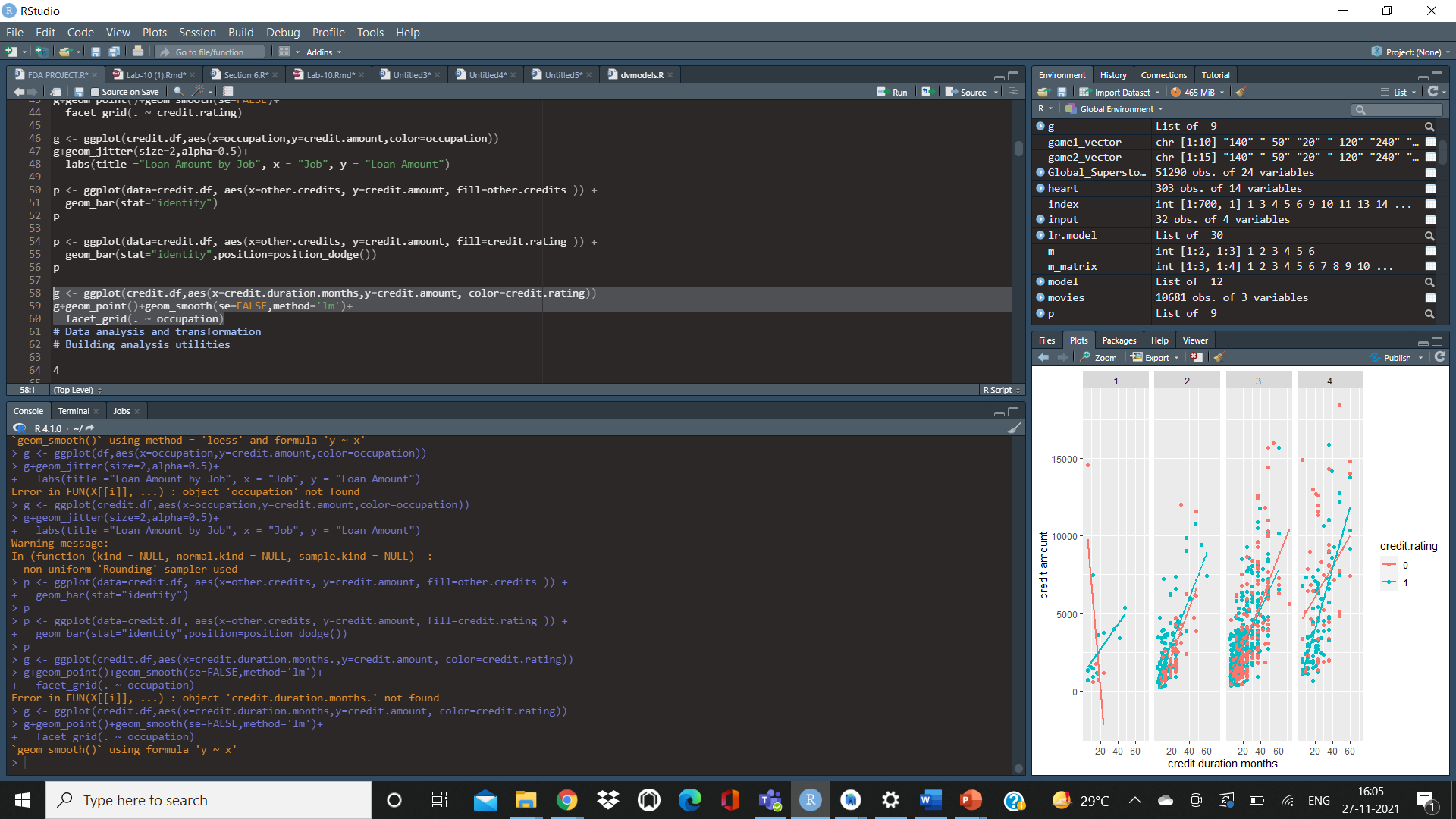
1-New car

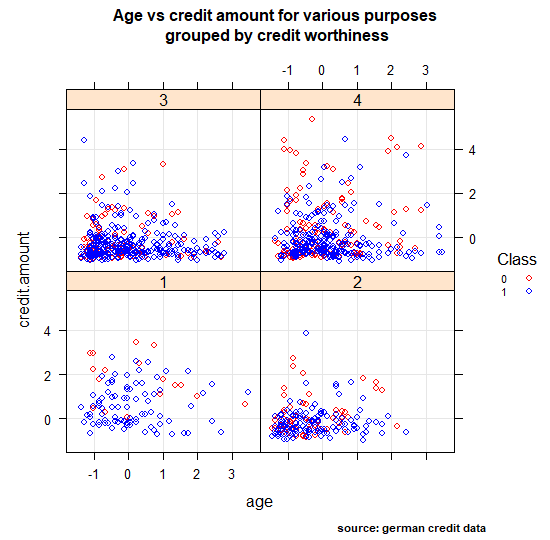
2-old car

3-Home applications

4-Others



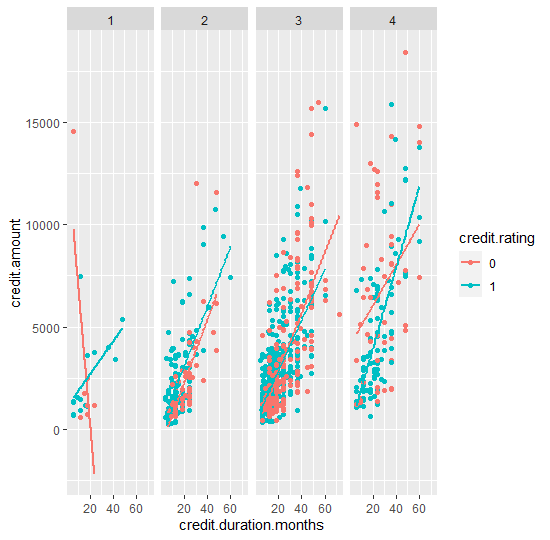




**INFERFENCE:**

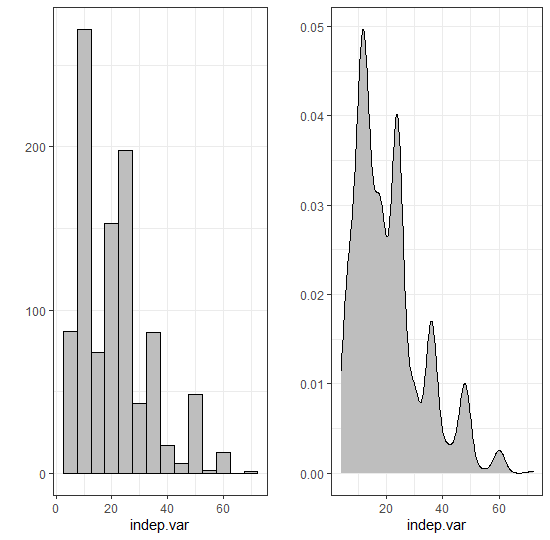
We can observe if the credit purpose is for a new car most of the clients have good credit ratings

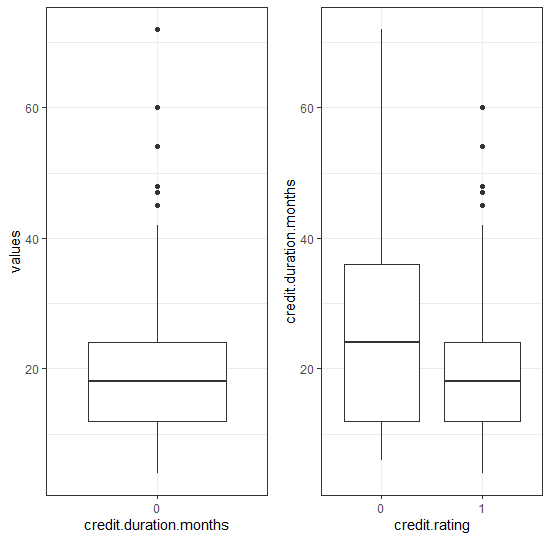
For the category others which consists of vacation , business , repair we can see that more number of Clients with a bad credit rating having loans having a higher credit amount



**CREDIT DURATION**

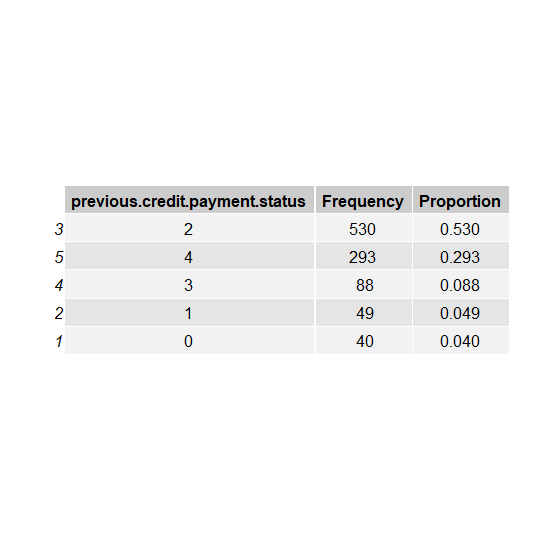


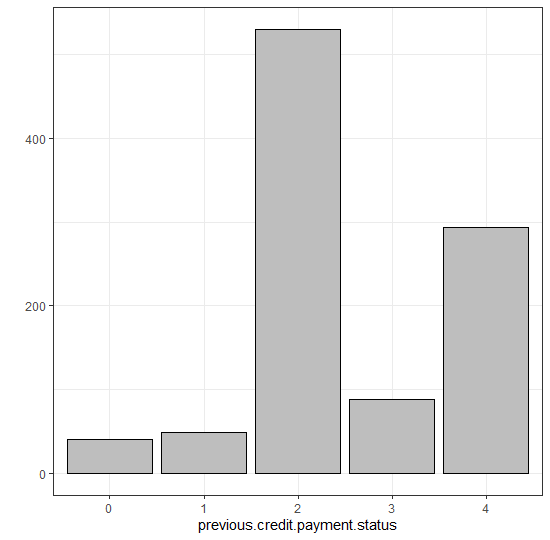




We can observe that median of credit duration is higher for people with a bad credit rating

**PREVIOUS CREDIT PAYMENT STATUS**



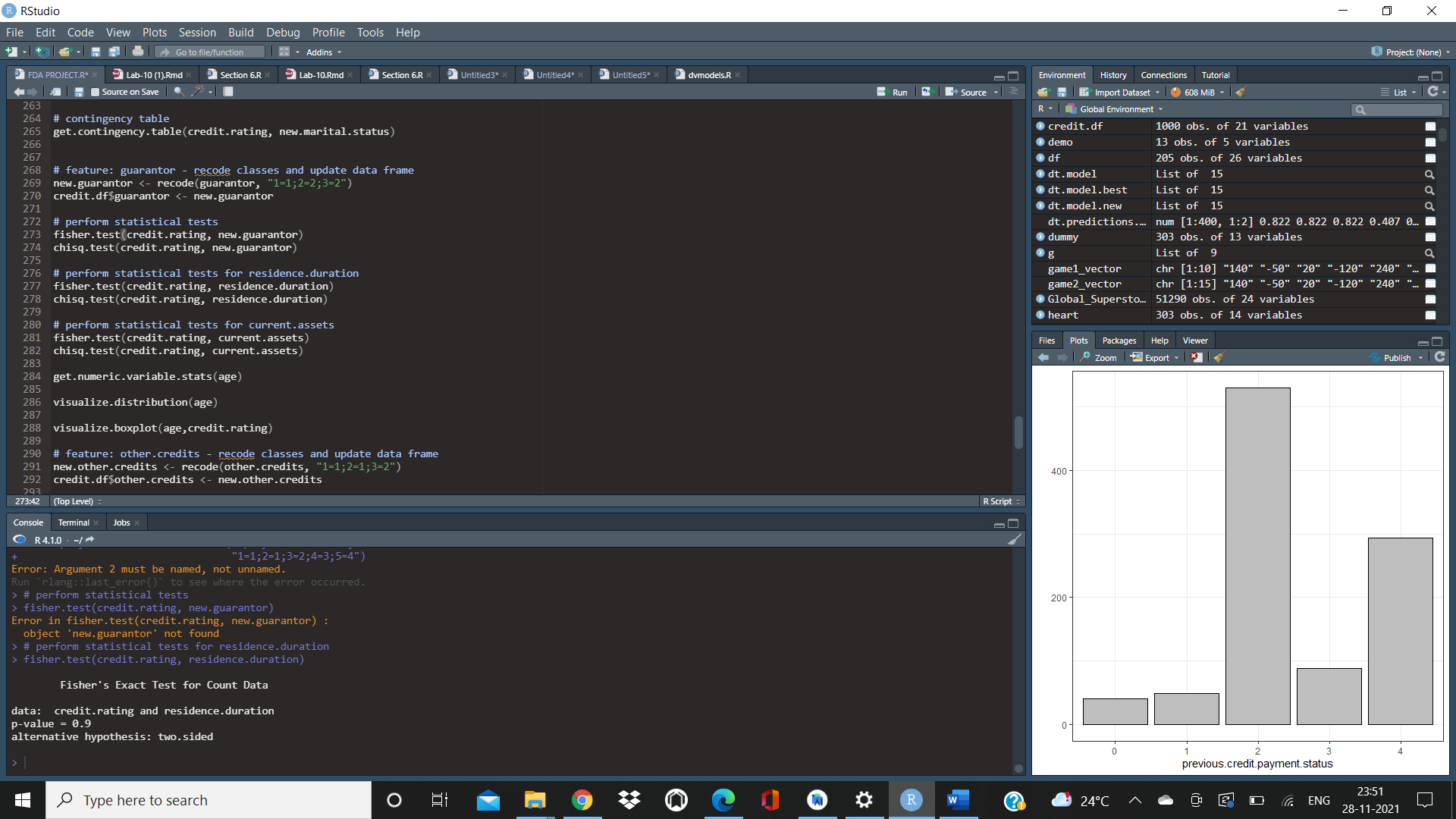


Category 0,1 and 3 are in low frequencies, to make the analysis part easier we conduct data transformation and merge 0 and 1 into 2 and 3 with 4.

**CONTINGENCY TABLE FOR CREDIT PAYMENT STATUS**

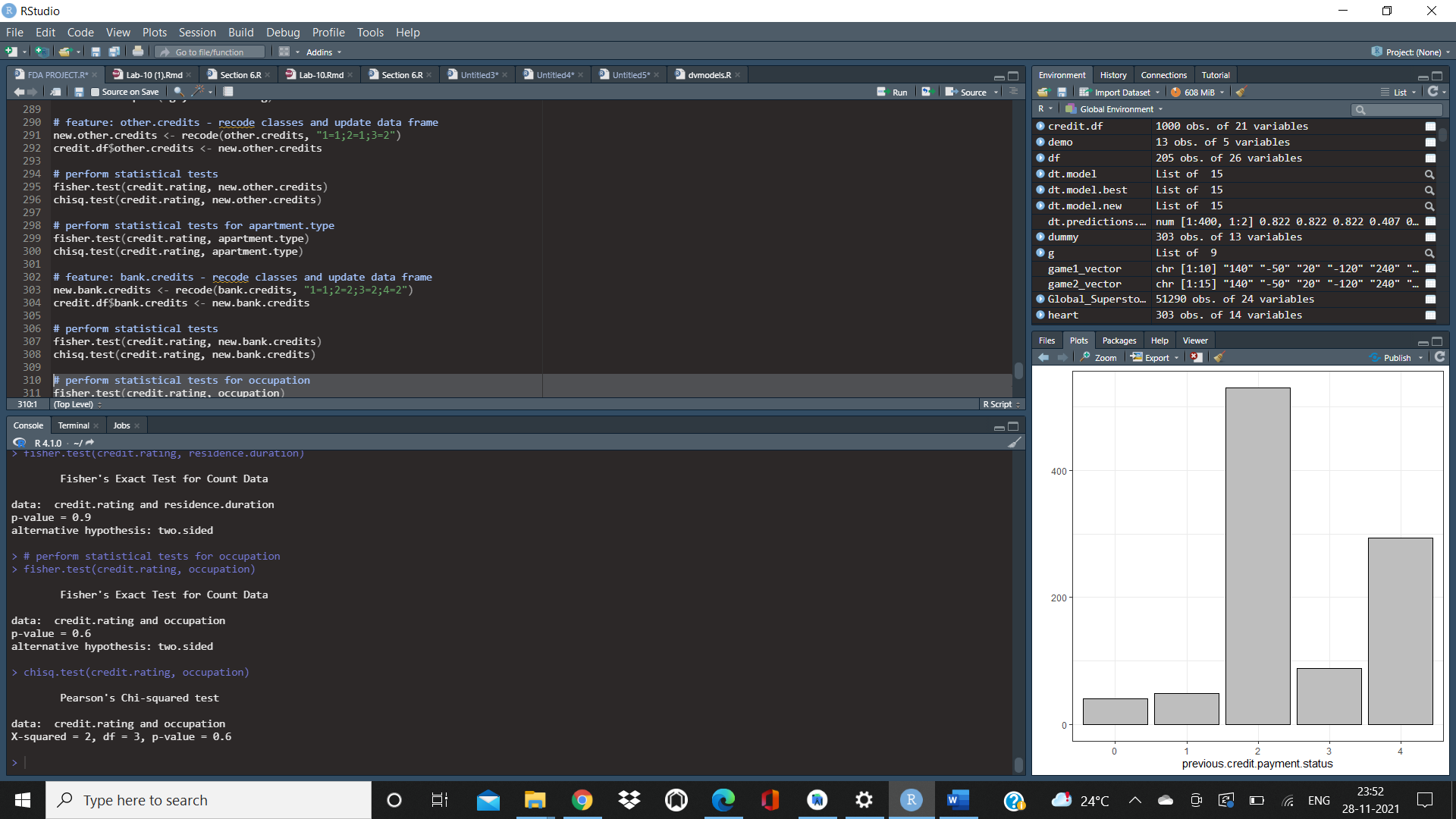


**STATISTICAL TESTS FOR RESIDENCE DURATION**



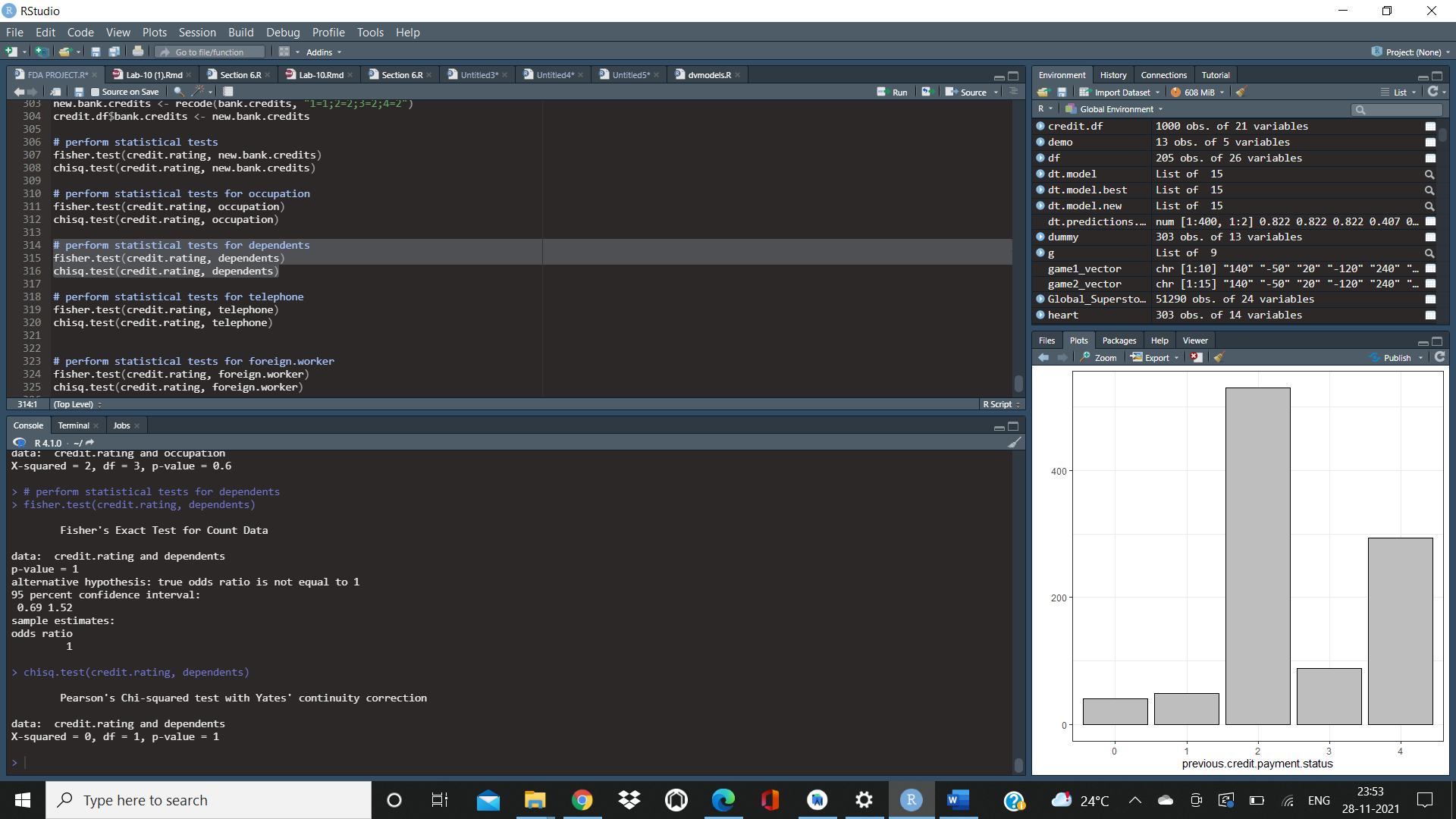
Since P value is greater than 0.05, there is no relation between residence duration and credit rating

**STATISTICAL TESTS FOR OCCUPATION**



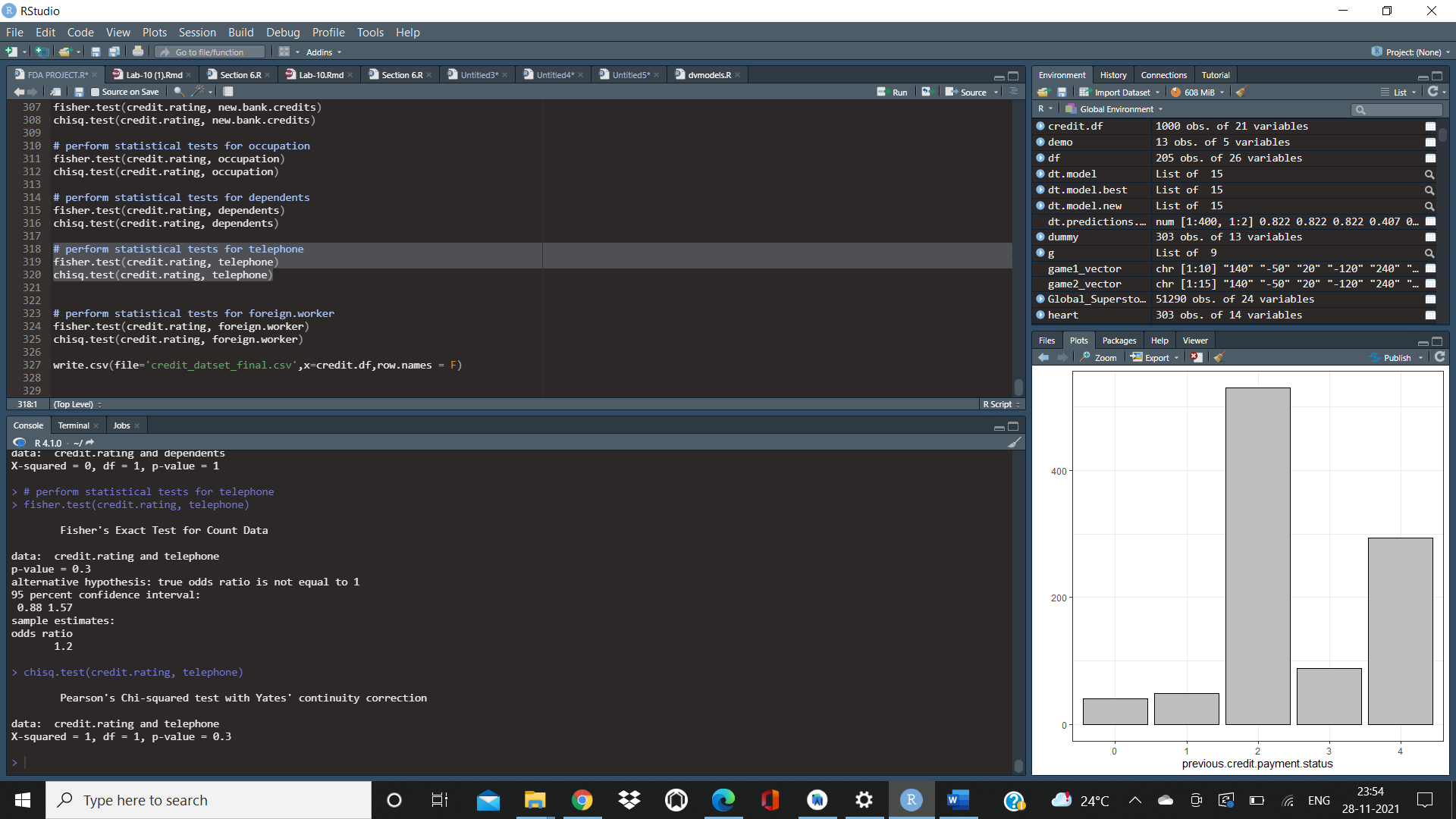
Since P value is greater than 0.05, there is no relation between occupation and credit rating

**STATISTICAL TESTS FOR DEPENDENTS**



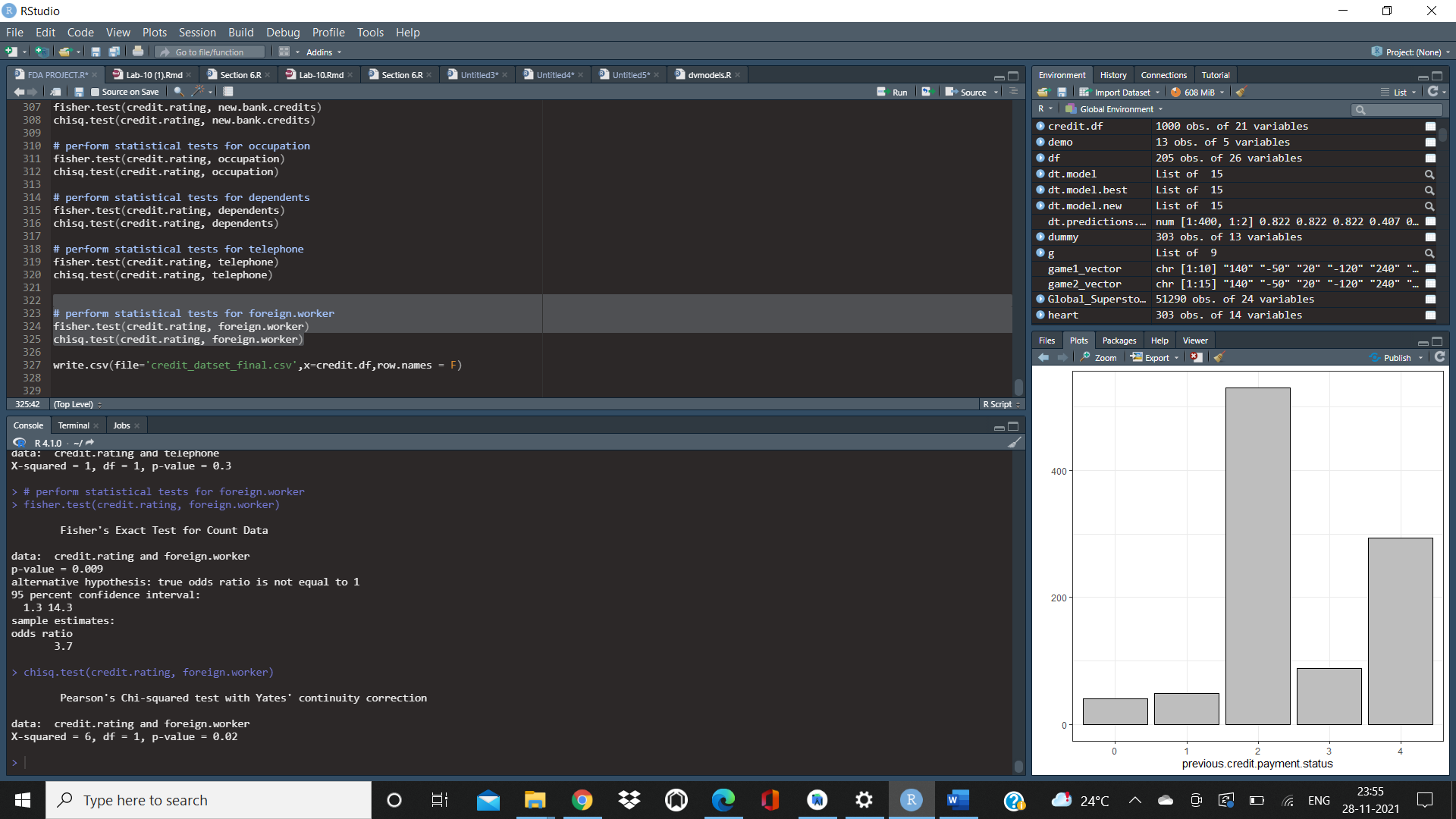
Since P value is greater than 0.05, there is no relation between dependents and credit rating

**STATISTICAL TESTS FOR TELEPHONE**



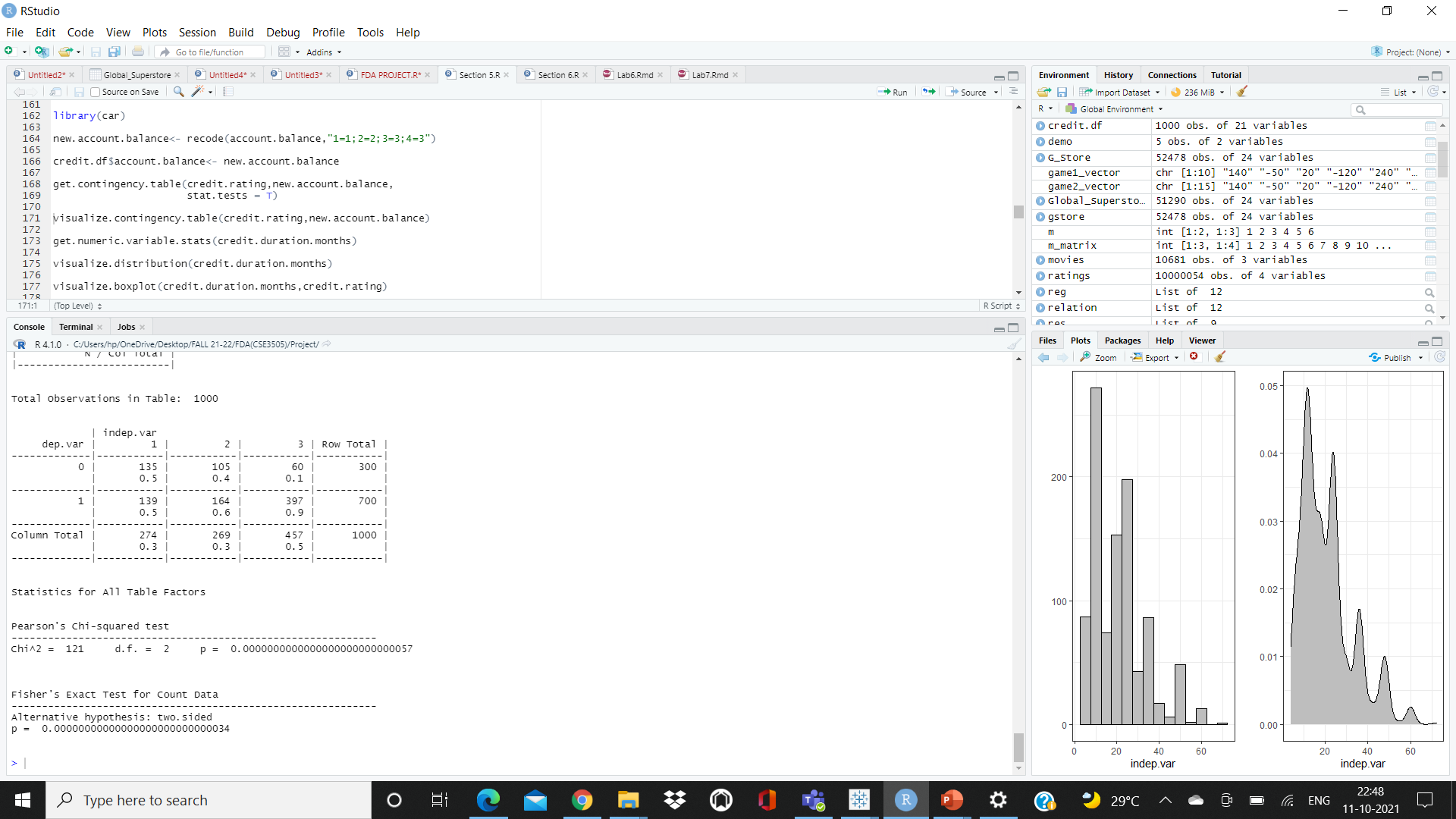
Since P value is greater than 0.05, there is no relation between telephone and credit rating

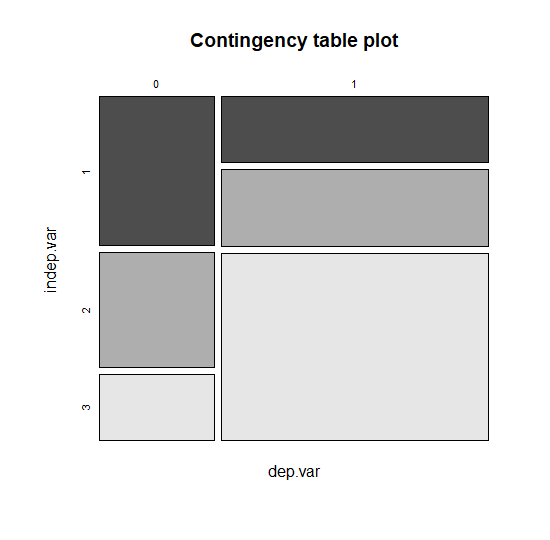
**STATISTICAL TESTS FOR FOREIGN WORKER**



Since P value is greater than 0.05, there is no relation between foreign worker and credit rating

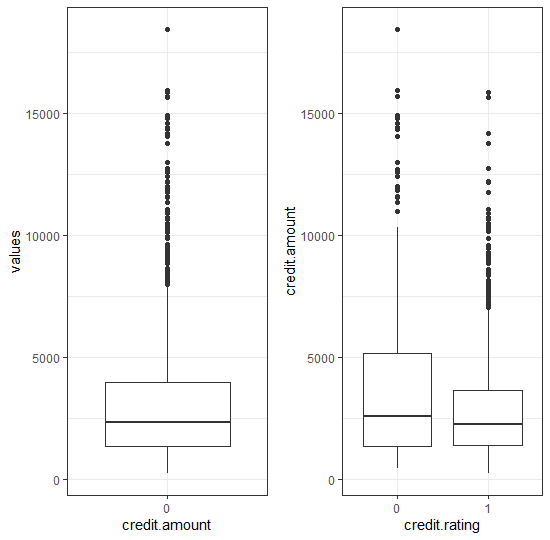
**Account Balance**

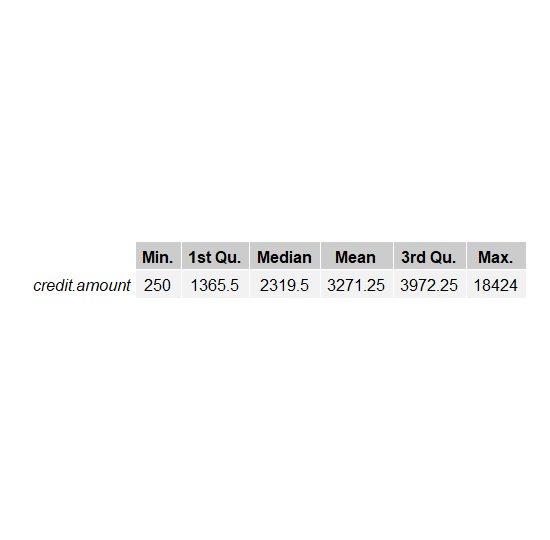
****

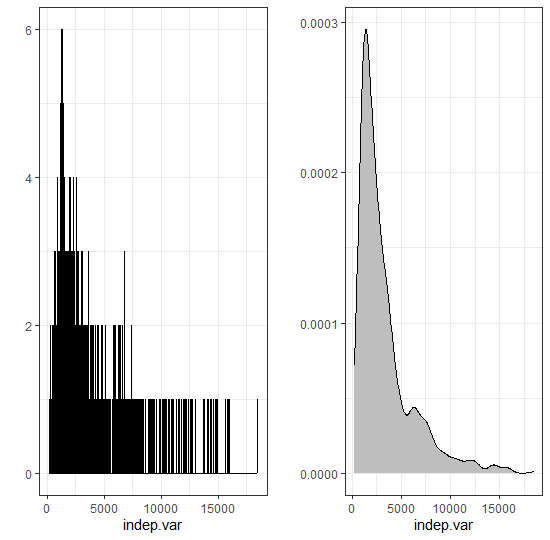
****

We can infer that since the P value is less than 0.05, there is strong co-relation between Account Balance and credit rating

Credit Amount







From the boxplot we can observe that the outlier values are higher for bad credit clients when compared to clients with a good credit rating

We can infer that median value from credited amount is higher for people with a bad credit rating

Distribution of credit amount Right skewed

* Data partition

# Stratified random sampling is a method of sampling that involves the division of a population into smaller groups known as strata. This is useful for imbalanced datasets, and can be used to give more weight to a minority class. In stratified random sampling, the strata are formed based on members’ shared attributes or characteristics.

# In our case we will use Good/Bad as strata and partition data into 70%-30% as train and test sets. The caret function **createDataPartition()** can be used to create balanced splits of the data

# 

**PREDICTIVE MODEL DEVELOPMENT**

# **LOGESTIC REGRESSION**

# Logistic regression is an approach for predicting the outcome of a categorialdependent variable based on one or more observed features. The probabilities describing the possible outcomes are modelled as a function of the observed variables using a logistic function.

# The goal of logistic regression is to directly estimate the distribution P(Y|X) from the training data.

# In our study, using the **glm()** function of the stats package, we fit our logistic regression model on the training set. Then, in the testing phase, we evaluated our fitted model on the test set with the **predict()** function of the same package.

**Call:**

**glm(formula = formula.init, family = "binomial", data = train)**

**Deviance Residuals:**

**Min 1Q Median 3Q Max**

**-2.5044 -0.6905 0.3700 0.7033 2.1270**

**Coefficients:**

**Estimate Std. Error z value Pr(>|z|)**

**(Intercept) 0.89034 0.93768 0.950 0.342360**

**account.balance2 0.34773 0.25835 1.346 0.178319**

**account.balance3 1.60348 0.26185 6.124 9.15e-10 \*\*\***

**credit.duration.months -0.24570 0.13441 -1.828 0.067548 .**

**previous.credit.payment.status2 0.67793 0.36584 1.853 0.063874 .**

**previous.credit.payment.status3 1.23683 0.38627 3.202 0.001365 \*\***

**credit.purpose2 -1.19562 0.49558 -2.413 0.015841 \***

**credit.purpose3 -1.40389 0.47489 -2.956 0.003114 \*\***

**credit.purpose4 -1.65864 0.45949 -3.610 0.000306 \*\*\***

**credit.amount -0.43728 0.15238 -2.870 0.004110 \*\***

**savings2 0.18083 0.33671 0.537 0.591238**

**savings3 0.91674 0.39081 2.346 0.018988 \***

**savings4 0.92693 0.32505 2.852 0.004350 \*\***

**employment.duration2 0.07653 0.28377 0.270 0.787407**

**employment.duration3 0.83884 0.35548 2.360 0.018289 \***

**employment.duration4 0.35838 0.33981 1.055 0.291585**

**installment.rate2 -0.35725 0.37193 -0.961 0.336789**

**installment.rate3 -0.94958 0.40337 -2.354 0.018567 \***

**installment.rate4 -1.13968 0.35984 -3.167 0.001539 \*\***

**marital.status3 0.62498 0.24206 2.582 0.009825 \*\***

**marital.status4 0.22582 0.36379 0.621 0.534781**

**guarantor2 0.22600 0.35645 0.634 0.526068**

**residence.duration2 -0.43643 0.35231 -1.239 0.215428**

**residence.duration3 -0.09929 0.39482 -0.251 0.801438**

**residence.duration4 0.01703 0.36124 0.047 0.962398**

**current.assets2 -0.68287 0.30476 -2.241 0.025048 \***

**current.assets3 -0.68305 0.28832 -2.369 0.017834 \***

**current.assets4 -1.64596 0.49844 -3.302 0.000959 \*\*\***

**age 0.13729 0.12887 1.065 0.286711**

**other.credits2 0.44379 0.26489 1.675 0.093865 .**

**apartment.type2 0.67952 0.29384 2.313 0.020747 \***

**apartment.type3 0.62108 0.58700 1.058 0.290027**

**bank.credits2 -0.21185 0.27711 -0.765 0.444560**

**occupation2 -0.36868 0.70252 -0.525 0.599725**

**occupation3 -0.38795 0.67718 -0.573 0.566726**

**occupation4 -0.10428 0.72168 -0.144 0.885106**

**dependents2 -0.53344 0.30007 -1.778 0.075447 .**

**telephone2 0.38574 0.23984 1.608 0.107763**

**foreign.worker2 2.17628 0.96562 2.254 0.024211 \***

**---**

**Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**

**(Dispersion parameter for binomial family taken to be 1)**

**Null deviance: 855.21 on 699 degrees of freedom**

**Residual deviance: 619.31 on 661 degrees of freedom**

**AIC: 697.31**

**Number of Fisher Scoring iterations: 5**

**Confusion Matrix and Statistics**

**1 0**

**1 184 46**

**0 26 44**

**Accuracy : 0.76**

**95% CI : (0.7076, 0.8072)**

**No Information Rate : 0.7**

**P-Value [Acc > NIR] : 0.01249**

**Kappa : 0.3898**

**Mcnemar's Test P-Value : 0.02514**

**Sensitivity : 0.8762**

**Specificity : 0.4889**

**Pos Pred Value : 0.8000**

**Neg Pred Value : 0.6286**

**Prevalence : 0.7000**

**Detection Rate : 0.6133**

**Detection Prevalence : 0.7667**

**Balanced Accuracy : 0.6825**

**'Positive' Class : 1**

Feature Selection

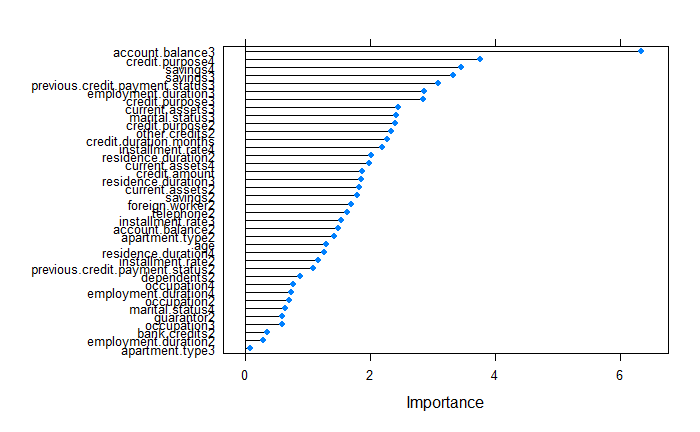
Feature selection is a process where you automatically select those features in your data that contribute most to the prediction variable or output in which you are interested.

Having irrelevant features in your data can decrease the accuracy of many models, especially linear algorithms like linear and logistic regression.

Three benefits of performing feature selection before modeling your data are:

* **Reduces Overfitting**: Less redundant data means less opportunity to make decisions based on noise.
* **Improves Accuracy**: Less misleading data means modeling accuracy improves.
* **Reduces Training Time**: Less data means that algorithms train faster.

In credit dataset the top 5 most important variables are account balance , credit duration , previous credit amount , bank.credits, savings



* Building model using feature selection

**Confusion Matrix and Statistics**

**1 0**

**1 183 51**

**0 27 39**

**Accuracy : 0.74**

**95% CI : (0.6865, 0.7887)**

**No Information Rate : 0.7**

**P-Value [Acc > NIR] : 0.072279**

**Kappa : 0.3299**

**Mcnemar's Test P-Value : 0.009208**

**Sensitivity : 0.8714**

**Specificity : 0.4333**

**Pos Pred Value : 0.7821**

**Neg Pred Value : 0.5909**

**Prevalence : 0.7000**

**Detection Rate : 0.6100**

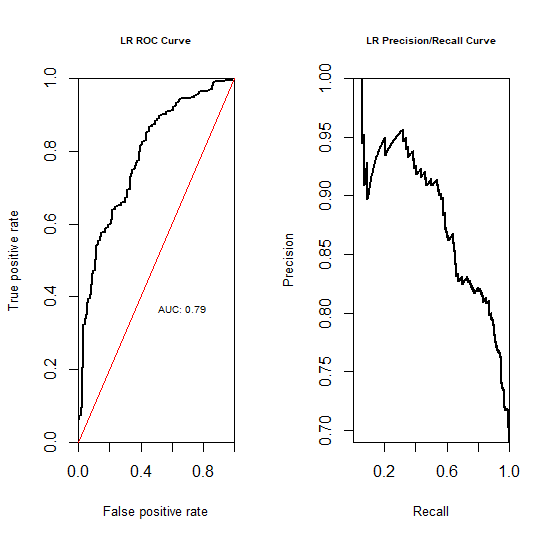
**Detection Prevalence : 0.7800**

**Balanced Accuracy : 0.6524**

**'Positive' Class : 1**

By observing the results of both the logistic regression models, we can infer that logistic regression with feature selection is better when compared to model with feature selection

Using the logistic regression model, we will plot the ROC curve along with the AUC value



AUC ranges in value from 0 to 1. A model whose predictions are 100% wrong has an AUC of 0.0; one whose predictions are 100% correct has an AUC of 1.0.

AUC is desirable for the following two reasons:

* AUC is **scale-invariant**. It measures how well predictions are ranked, rather than their absolute values.
* AUC is **classification-threshold-invariant**. It measures the quality of the model's predictions irrespective of what classification threshold is chosen.

For our case the AUC value is 0.79 which is decent

**Support vector machines**

SVM methods use linear conditions in order to separate out the classes from one another. The idea is to use a linear condition that separates the two classes from each other as well as possible. SVMs were developed by Cortes &Vapnik (1995) for binary classification. Their approach may be roughly sketched with the following tasks : Class separation, Overlapping classes, Nonlinearity and Problem solution which is a quadratic optimization problem.

The principal task, Class separation, lies in looking for the optimal separating hyperplane between the two classes by maximizing the margin between the classes’ closest points (see Figure below)—the points lying on the boundaries are called support vectors, and the middle of the margin is our optimal separating hyperplane. For further details, follow [Meyer and Wien(2015)](https://cran.r-project.org/web/packages/e1071/vignettes/svmdoc.pdf) or Karatzoglou et al (2005).

The package e1071 offers an interface to the award-winning C++- implementation by Chang & Lin(2001), [libsvm](https://www.csie.ntu.edu.tw/~cjlin/libsvm/)

In our study, we used the **svm()** function of that package, which was designed to be as intuitive as possible (we also compared with results of **ksvm()** function of the kernlab package). Models are fitted on the training set and predicted on unseen data(test set) as usual.

**Call:**

**svm(formula = formula.init, data = train, kernel = "radial", cost = 100, gamma = 1)**

**Parameters:**

**SVM-Type: C-classification**

**SVM-Kernel: radial**

**cost: 100**

**Number of Support Vectors: 700**

**( 490 210 )**

**Number of Classes: 2**

**Levels:**

**0 1**

**Confusion Matrix and Statistics**

**1 0**

**1 210 90**

**0 0 0**

**Accuracy : 0.7**

**95% CI : (0.6447, 0.7513)**

**No Information Rate : 0.7**

**P-Value [Acc > NIR] : 0.5284**

**Kappa : 0**

**Mcnemar's Test P-Value : <2e-16**

**Sensitivity : 1.0**

**Specificity : 0.0**

**Pos Pred Value : 0.7**

**Neg Pred Value : NaN**

**Prevalence : 0.7**

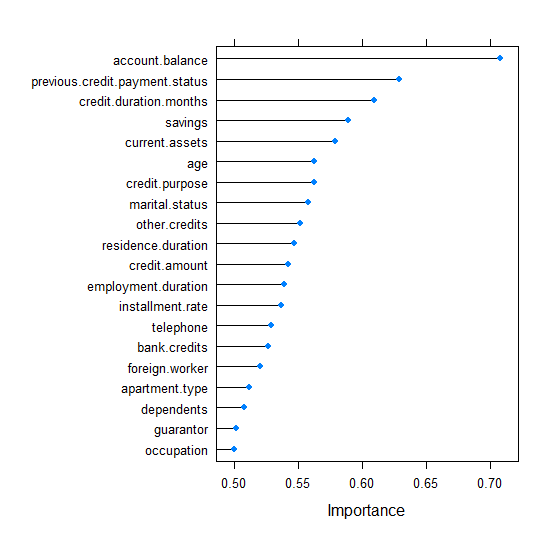
**Detection Rate : 0.7**

**Detection Prevalence : 1.0**

**Balanced Accuracy : 0.5**

**'Positive' Class : 1**

* Feature selection



**Confusion Matrix and Statistics**

**1 0**

**1 188 52**

**0 22 38**

**Accuracy : 0.7533**

**95% CI : (0.7005, 0.8011)**

**No Information Rate : 0.7**

**P-Value [Acc > NIR] : 0.0238850**

**Kappa : 0.3509**

**Mcnemar's Test P-Value : 0.0007485**

**Sensitivity : 0.8952**

**Specificity : 0.4222**

**Pos Pred Value : 0.7833**

**Neg Pred Value : 0.6333**

**Prevalence : 0.7000**

**Detection Rate : 0.6267**

**Detection Prevalence : 0.8000**

**Balanced Accuracy : 0.6587**

**'Positive' Class : 1**

# 000009.png

# **DECISION TREES**

# Decision trees create a hierarchical partitioning of the data, which relates the different partitions at the leaf level to the different classes. The hierarchical partitioning at each level is created with the use of a split criterion. The split criterion may either use a condition (or predicate) on a single attribute, or it may contain a condition on multiple attributes. The former is referred to as a univariate split, whereas the latter is referred to as a multivariate split. The overall approach is to try to recursively split the training data so as to maximize the discrimination among the different classes over different nodes. The discrimination among the different classes is maximized, when the level of skew among the different classes in a given node is maximized.

# In our study, we built our classification tree model on the training set using the rpart function of the rpart package. By default, **rpart()** function uses the Gini impurity measure to split the node. The higher the Gini coefficient, the more different instances within the node. We predicted the fitted tree on the test set with the predict() function.

**Confusion Matrix and Statistics**

**1 0**

**1 172 50**

**0 38 40**

**Accuracy : 0.7067**

**95% CI : (0.6516, 0.7576)**

**No Information Rate : 0.7**

**P-Value [Acc > NIR] : 0.4283**

**Kappa : 0.2739**

**Mcnemar's Test P-Value : 0.2410**

**Sensitivity : 0.8190**

**Specificity : 0.4444**

**Pos Pred Value : 0.7748**

**Neg Pred Value : 0.5128**

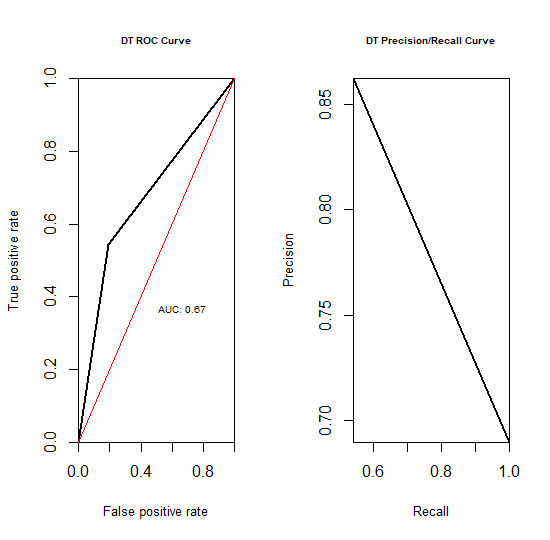
**Prevalence : 0.7000**

**Detection Rate : 0.5733**

**Detection Prevalence : 0.7400**

**Balanced Accuracy : 0.6317**

**'Positive' Class : 1**



**AUC value is 0.67**

# **RANDOM FOREST**

# Random Forest can be regarded as a variant of Bagging approach. It follows the major steps of Bagging and uses decision tree algorithm to build base classifiers. Besides Bootstrap sampling and majority voting used in Bagging, Random Forest further incorporates random feature space selection into training set construction to promote base classifiers’ diversity.

# In our study, we used the **randomForest()** function of the randomForest package which implements [Breiman’s random forest algorithm](https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf) (based on Breiman and Cutler’s original Fortran code), to build our random forest models on training set . Then , we predicted the latter model on test set with the **predict()** function.

**Call:**

**randomForest(formula = formula.init, data = train, importance = T, proximity = T)**

**Type of random forest: classification**

**Number of trees: 500**

**No. of variables tried at each split: 4**

**OOB estimate of error rate: 23%**

**Confusion matrix:**

**0 1 class.error**

**0 91 119 0.56666667**

**1 42 448 0.08571429**

**Confusion Matrix and Statistics**

**1 0**

**1 270 21**

**0 7 102**

**Accuracy : 0.93**

**95% CI : (0.9004, 0.953)**

**No Information Rate : 0.6925**

**P-Value [Acc > NIR] : < 2e-16**

**Kappa : 0.8303**

**Mcnemar's Test P-Value : 0.01402**

**Sensitivity : 0.9747**

**Specificity : 0.8293**

**Pos Pred Value : 0.9278**

**Neg Pred Value : 0.9358**

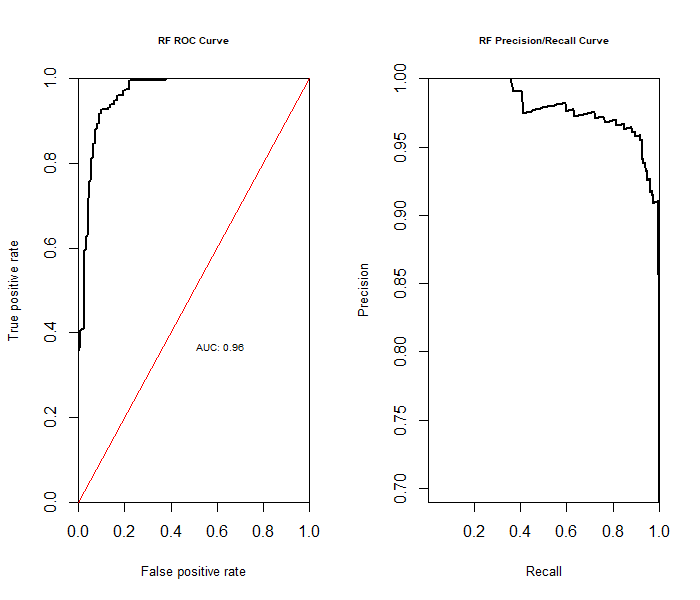
**Prevalence : 0.6925**

**Detection Rate : 0.6750**

**Detection Prevalence : 0.7275**

**Balanced Accuracy : 0.9020**

**'Positive' Class : 1**



# Neural Network

# An artificial neural network (ANN) or neural net is a graph of connected units representing a mathematical model of biological neurons. Those units are sometimes referred to as processing units, nodes, or simply neurons. The units are connected through unidirectional or bidirectional arcs with weights representing the strength of the connections between units. This is inspired from the biological model in which the connection weights represent the strength of the synapses between the neurons, inhibiting or facilitating the passage of signals.

# In our study, we principally used the nnet package by Ripley et al(2016) . We also took into account the NeuralNetTools(Visualization and Analysis Tools for Neural Networks) and neuralnet packages.

# **Confusion Matrix and Statistics**

# **X1 X0**

# **X1 248 52**

# **X0 29 71**

# 

# **Accuracy : 0.7975**

# **95% CI : (0.7547, 0.8358)**

# **No Information Rate : 0.6925**

# **P-Value [Acc > NIR] : 1.542e-06**

# 

# **Kappa : 0.4985**

# 

# **Mcnemar's Test P-Value : 0.01451**

# 

# **Sensitivity : 0.8953**

# **Specificity : 0.5772**

# **Pos Pred Value : 0.8267**

# **Neg Pred Value : 0.7100**

# **Prevalence : 0.6925**

# **Detection Rate : 0.6200**

# **Detection Prevalence : 0.7500**

# **Balanced Accuracy : 0.7363**

# 

# **'Positive' Class : X1**

# 000007.png

# **CONCLUSION**

For selecting the best model we will be comparing accuracy , specificity , sensitivity and AUC value of the models

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MODEL | ACCURACY | SPECIFICITY | SENSITIVITY | AUC VALUE |
| LOGISTIC REGRESSION | 0.76 | 0.8 | 0.45 | 0.8 |
| SUPPORT VECTOR MACHINE | 0.75 | 0.9 | 0.5 | 0.79 |
| DECISION TREE | 0.7 | 0.8 | 0.44 | 0.67 |
| RANDOM FOREST | 0.94 | 0.97 | 0.84 | 0.96 |
| NEURAL NETWORK | 0.8 | 0.9 | 0.57 | 0.81 |

By comparison we can conclude that Random forest is best suited for our analysis, it has the highest accuracy, a very good auc value and decent specificity and sensitivity values

Decision tree is the worst model with accuracy being equal to no information rate.

# **REFERENCES**