Predicting credit risk profile using Machine Learning

~

German Credit Case

free rights image, source: https://www.pexels.com/fr-fr/

Niccolo Cherubini & Ivan Kostine ***HEC Lausanne - Université de Lausanne***

Table of contents

Business understanding

Identifying the best loan applicant profile is essential to maximize profits while minimizing risks for both parties. Based on the data provided to us, we looked for the best prediction solutions for future applicants.

Data exploration

Data understanding & preparation

The customer provided a dataset containing information about a 1000 past credit applicants. Based on 30 criteria, the analysts are able to qualify an applicant with a good or bad final risk rating. In more technical words, the set contained 30 variables and 1 response variable expressing a good risk rating by “1” and a bad credit risk with “0”.
No missing values were detected within the data set.

After looking closely at the nature of the data, we were able to split variables in 2 types: categorical and numerical.
Please refer to the “Annex - Features Information” section to see all the details about the variables.

Categorical Variables

A categorical variable expresses data by levels of categories. For instance, the variable "Job" is divided into 4 categories from 0 to 3:

0: unemployed/unskilled - non resident
1: unskilled - resident
2: skilled employee/official
3: management/self-employed/highly qualified employee/officer

The dataset contains 28 categorical variables:

- 17 variables + 1 (Response variable) are binary: “0” (no) or “1” (yes)
- 10 variables with more than 2 levels with a maximum of 5 levels

The variables install_rate and num_dependents were initially numerical variables. After observation, we realized that they did not contain many distinct observations: num_dependents varied between 1 and 2 while num_dependents between 1 and 4. This is why these variables were considered as categorical.

Incoherent occurrences

Two incoherences were found in two variables:

- The variable "education" which should be binary (0;1) had one occurrence at -1, modified by a 1.
- The variable "guarantor" which should be binary (0;1) had one occurrence at 2, modified by a 1.

Visualization and Correlation Matrix

The graph below shows the distribution of the response variable through the different levels of the checking accounts variable. As we can see, the 4th level contains almost 50% of good ratings for this variable. This could reflect an incoherence in data since this level represents applicants with no checking account, we would have expected this 50% to be attributed to the 3rd level. On the contrary, 45% of applicants with a negative checking account (level 0) have a bad rating.

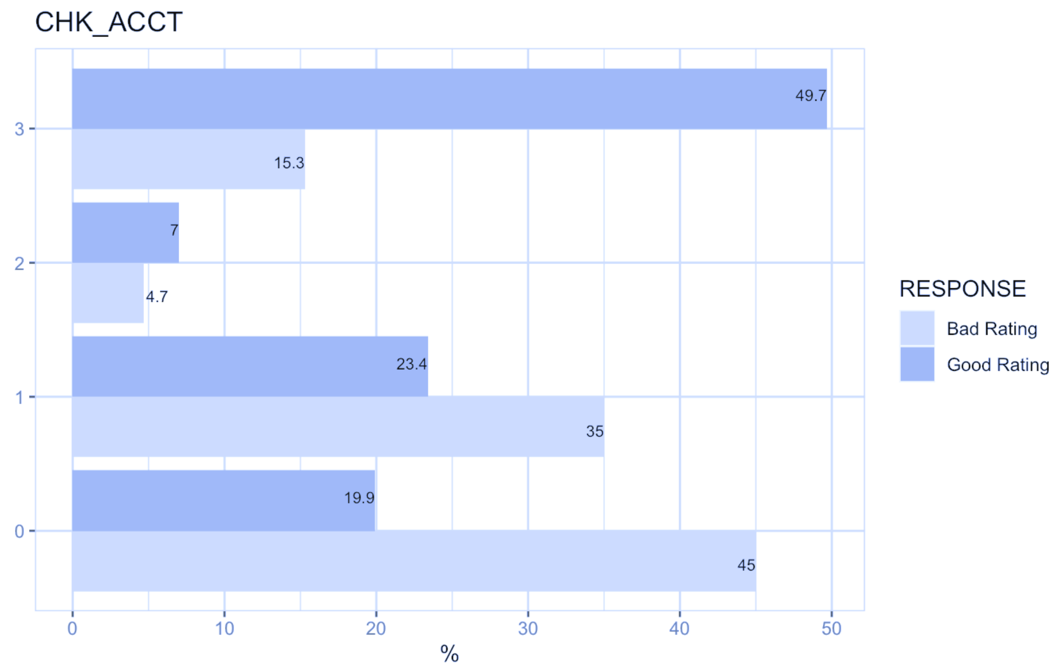

Fig.1: CHK_ACCT vs RESPONSE

This second graph shows the distribution of credit history through the response variable. Once again, some incoherences appear: 82% of applicant with a critical account level (level 4) and 68% of applicant who had delays in paying off past credits (level 3) received a good rating. 62.5 % of level 0 and 57.1% of level 1 applicants were attributed a bad rating when these levels should be positive arguments for a credit request.

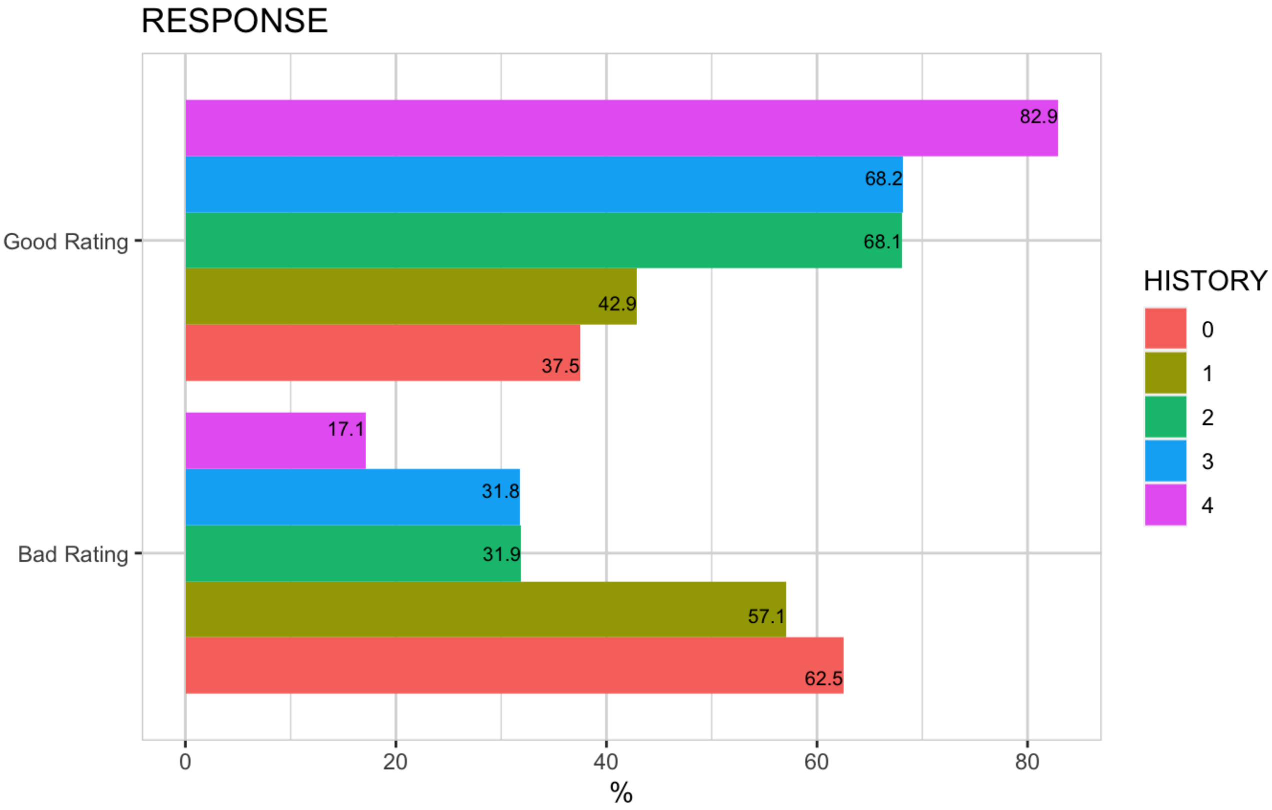

Fig.2: RESPONSE vs HISTORY

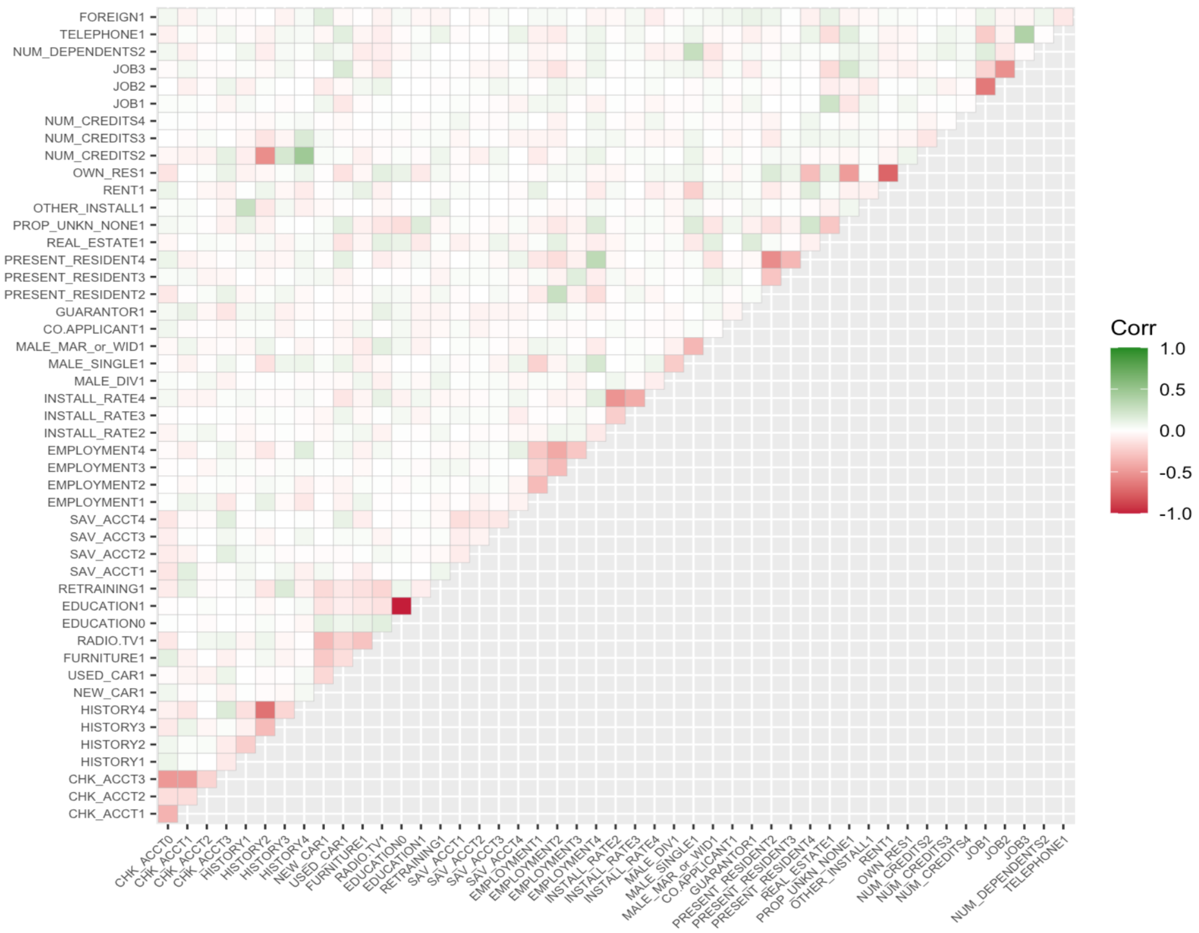

Fig.3: Correlation matrix - categorical variables

The above correlation matrix shows correlation between categorical variables. Red squares indicate a negative correlation and green ones a positive correlation. A positive correlation indicates that an increase/decrease in a variable respectively increases/decreases the other variable. A negative correlation, also known as reversed correlation, indicates that the increase/decrease of a variable respectively decreases/increases the other one. Logically, categorical variables should have negative correlations through their own levels

Strong positive correlations:

- Number of credits level 2 with history level 4
- Telephone level 1 with job level 3
- Present resident level 4 with employment level 4

Strong negative correlations:

- own residence level 1 and and rent level 1
- job 2 and job 1
- history 2 and history 4
- education 1 and education 0

Numerical Variables

Numerical variables represent quantitative and measurement values. The data set counted 3 features:

- Duration
- Amount
- Age

Incoherent occurrences

The variable "Age" contained an incoherent occurrence indicating a 125 years old applicant, we changed and aligned this occurrence to the previous oldest age in the data: 75 years.

Visualization and Correlation Matrix

The below boxplot shows that there is not a lot of difference within the distribution of the numerical variables compared to the response. In other terms, the 3 variables are almost equally distributed between good and bad ratings. However, we can note that more bad ratings are attributed to a longer duration and that age seems to play a role in the attribution of good ratings.

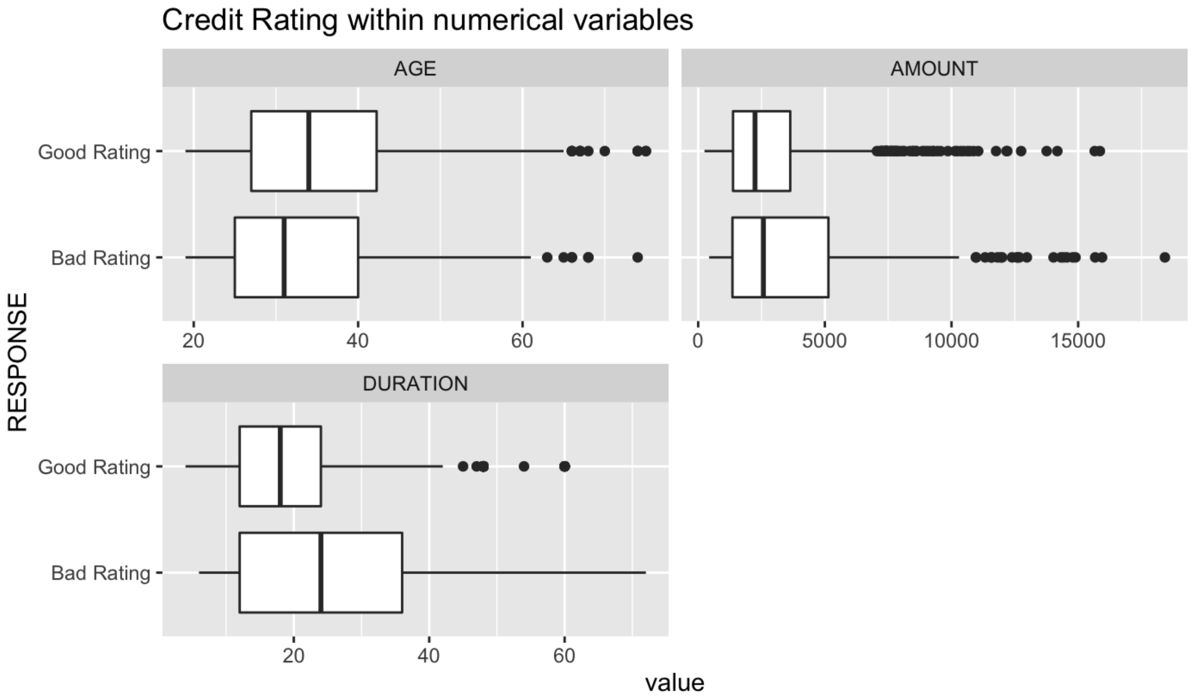

Fig.4: Boxplots - numerical variables

Thanks to the correlation matrix we were able to identify some positive correlation between the amount and age variables. This means that the higher the amount of the loan, the longer the duration tends to be.

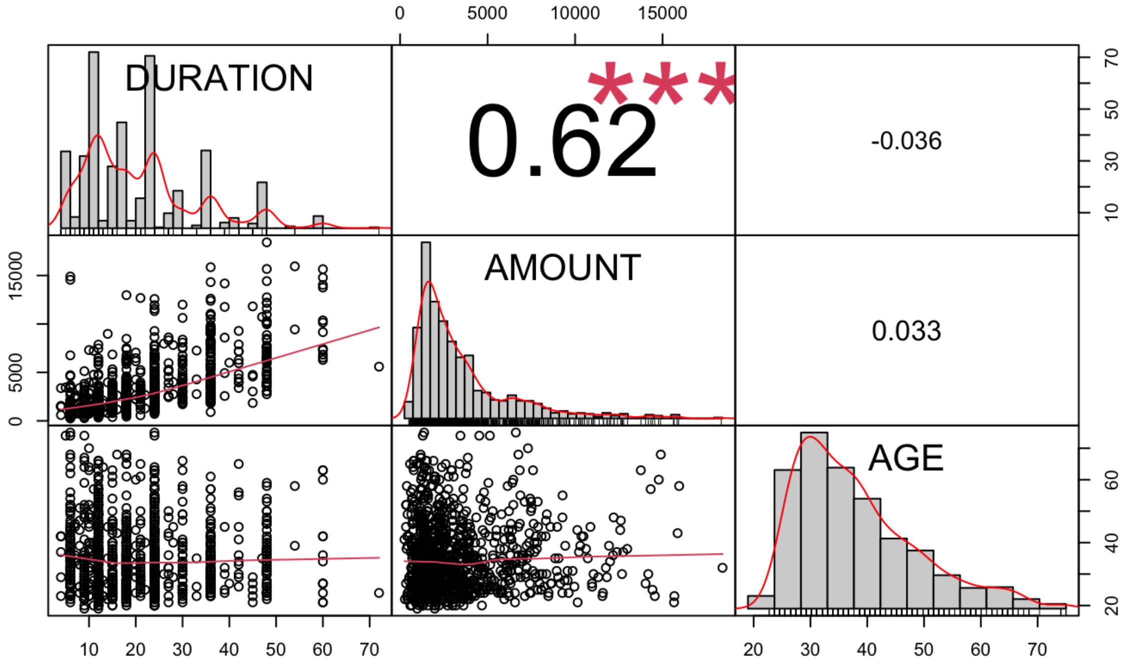

Fig.5: Correlation Matrix - numerical variables

Response Variable

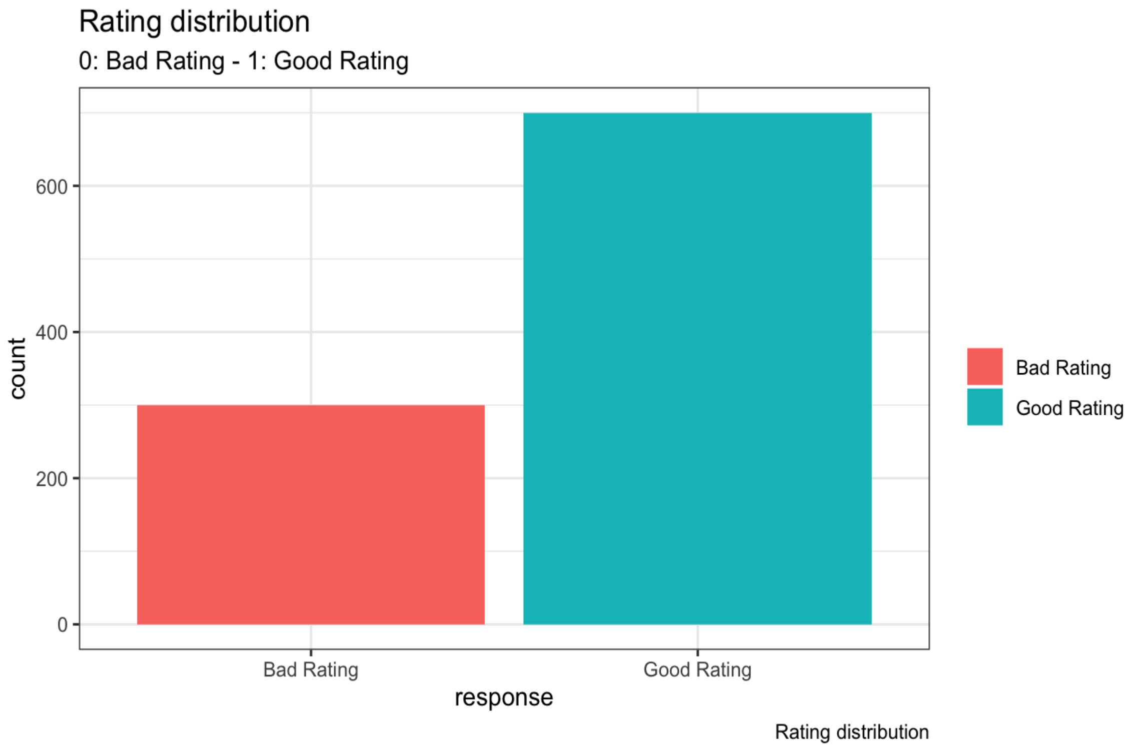

Fig.6: Distribution of the RESPONSE variable

Finally, an overview on the distribution of the response variable was necessary. The above graph shows a higher amount of good ratings. This might represent an issue in prediction because models could tend to predict more easily good ratings and show bad performance at predicting bad ratings. As previous incoherences showed, the error could reside in the reporting of the information about the credit rating.

Modelling

Before processing the data in the different models, we split it into two subsets: a training set containing 80% of the original data and a test set with the 20% left. In order to keep some homogeneity within the two sets, a method called “stratified random split” has been applied. In simple words, this splitting method keeps the original proportion of target distribution in both subsets. For instance, we had a 70% of good rating and 30% of bad rating in the original data set, we will find similar distributions in both train and split set thanks to the stratified random split.

With the training set, we were able to train the models to recognize and rank the results, while the test set allowed us to verify the performance and reliability of a specific model.

Before going into details, here is some information about the visualization and interpretation of the results. Since the results are expressed in binary format (0;1) or "bad" and "good" respectively, the clearest way to display them is the confusion matrix that compares the data predicted by a model to the actual value present in the test set. There are four possible outputs:

- The predicted output is a "good score", the actual value is also a "good score".
- The predicted output is a "good grade", the actual value is a "bad grade".
- The predicted output is a "bad grade", actual value is a "bad grade".
- The predicted value is a "bad grade", actual value is a "good grade".

In order to estimate the quality of a model to predict results, the following metrics are used:

- Accuracy: measures the overall capacity of a model to predict the same result as the actual value.
- Sensitivity: measure the ability of the model to predict “good ratings” that are actually “good ratings”.
- Specificity: measure the ability of the model to predict “bad ratings” that are actually “bad ratings”.
- Balanced accuracy: represents the average between Sensitivity and Specificity.

In this particular case, the ideal model has high accuracy and balanced accuracy, the closer sensitivity and specificity are, the better. In other terms, we want a model that can accurately predict both good and bad ratings.

Neural Networks

Briefly explained, a neural network takes input data and processes it through a network of several processors distributed in different layers that will identify and classify this input to finally find and return an output. In this particular case, there are 18 inputs processed through 10 layers.

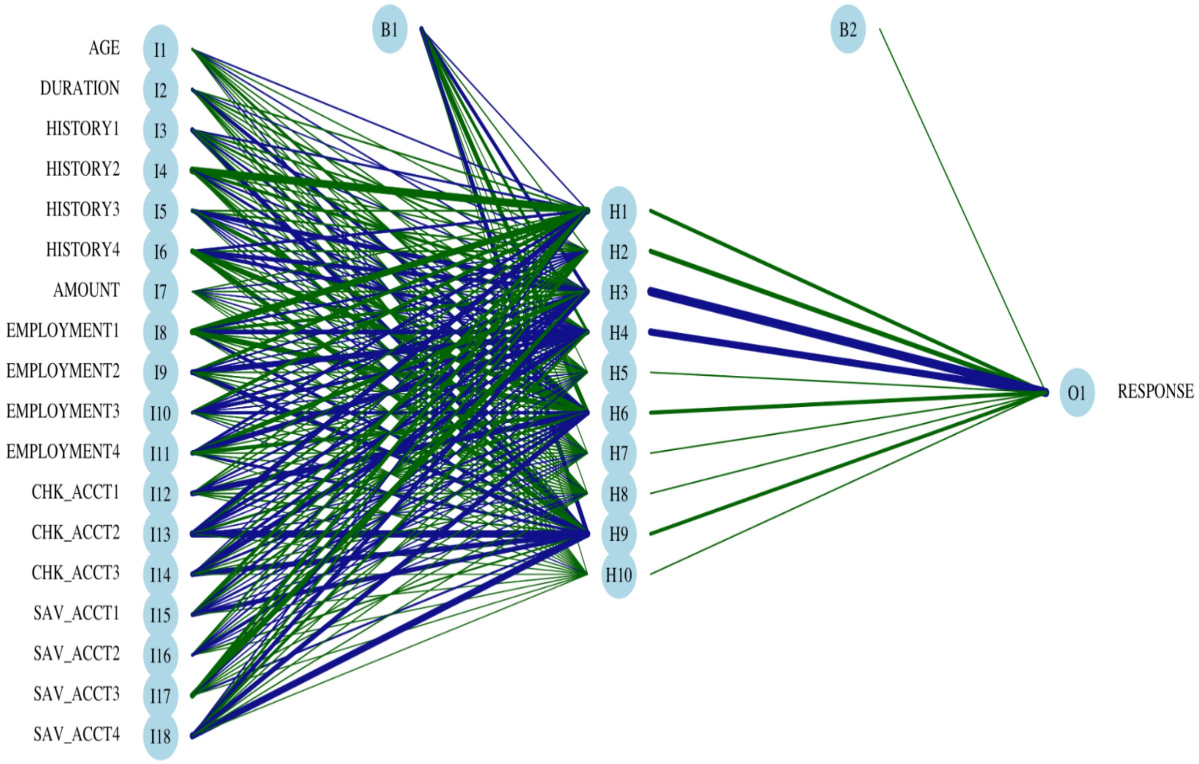

Fig.7: Neural Network

Confusion matrix	Actual value	
Prediction	**Bad Rating**	**Good Rating**
Bad Rating	42	32
Good Rating	48	178

Accuracy: **0.733**

Sensitivity: **0.848**

Specificity: **0.467**

Balanced accuracy: **0.657**

Neural Networks confusion matrix shows a really high sensitivity for a low specificity, this model will tend to predict too many good ratings and neglect bad ratings.

Classification Tree

A classification tree is a structural mapping of binary decisions that lead to a decision. It is composed of branches that represent attributes, while the leaves represent decisions. The decision process starts at the trunk and follows the branches until a leaf is reached. The algorithm iteratively selects the attribute and value that can split a set of samples into two subgroups, minimizing the variability within each subgroup while maximizing the contrast between them. In this particular case the tree has 3 decision nodes and 4 terminal nodes which represent the possible final outputs.

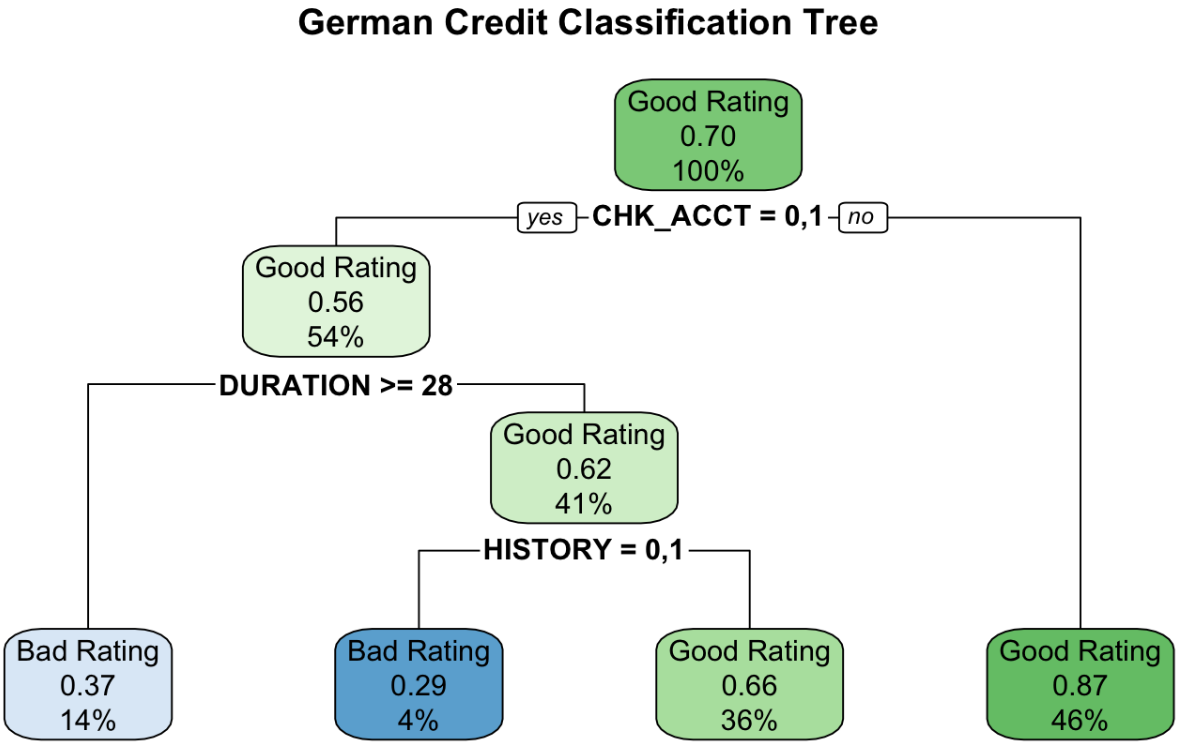

Fig.8: Classification Tree

Confusion matrix	Actual value	
Prediction	**Bad Rating**	**Good Rating**
Bad Rating	33	20
Good Rating	57	190
Accuracy: **0.743** *Sensitivity:* **0.905** *Specificity:* **0.367** *Balanced accuracy:* **0.636**		

Classification tree confusion matrix shows an extremely high sensitivity for a really low specificity. This model will tend to exclusively predict good ratings and totally neglect bad ratings.

Random Forest

In simple words a Random Forest processes randomly a lot of decision trees working as an ensemble and selects the best outputs through a voting method. The variable importance shows the variables that were used the most by the model. The model selects the variables that help the most in the decision splitting of the trees.

The most important variables are for instance: credit amount, checking account, age, credit duration, credit history, savings accounts and employment (seniority at work).

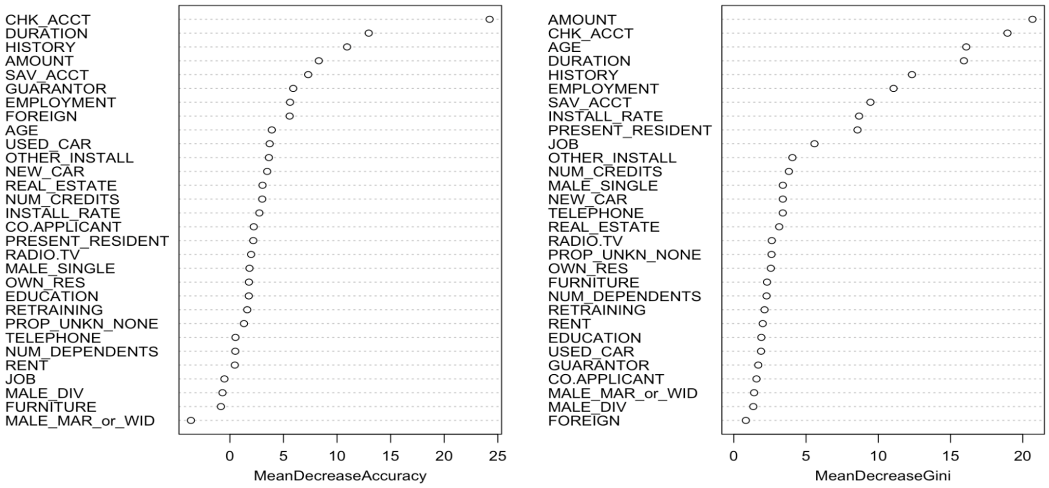

Fig.8: Random Forest - Variables importance

Confusion matrix	Actual value	
Prediction	**Bad Rating**	**Good Rating**
Bad Rating	65	63
Good Rating	25	147
Accuracy: **0.707** *Sensitivity:* **0.7** *Specificity:* **0.722** *Balanced accuracy:* **0.711**		

During the design of the model, a cross-validation method that will generate subsets of the training set and submit them for testing during the training phase of the model has been applied. This allows to improve the accuracy of the model.
Finally, the Random Forest confusion matrix shows equal sensitivity and specificity with a good overall accuracy and balanced accuracy. This model would be effective in predicting both good and bad results.

Linear Discriminant Analysis

LDA is a model that looks for linear combinations of variables which best explain the data. It was the first model to be applied in bankruptcy predictions by using accounting ratios and other financial variables, and it is still among the leading models to this date. In the case of bank loan risk prediction, this model can be used to classify the potential candidate as bad or good.

We built our model according to the following equation:

Response ~ *

where * represents all the other features of the provided dataset. To build our model and perform predictions we splitted the data set in a training and a testing subset (containing 80% and 20% of the data respectively). We then trained the prediction model using the training data set and evaluated its accuracy thanks to the test data set.

The following table summarizes the results of the prediction model:

Confusion matrix	Actual value	
Prediction	**Bad Rating**	**Good Rating**
Bad Rating	33	26
Good Rating	27	114
Accuracy: **0.735** *Sensitivity*: **0.55** *Specificity*: **0.8143** *Balanced accuracy*: **0.6821**		

We also built the following ROC graph showing the prediction capacity of the model. An AUC score close to 1 would mean that the model is very good at separating classes and does a perfect prediction:

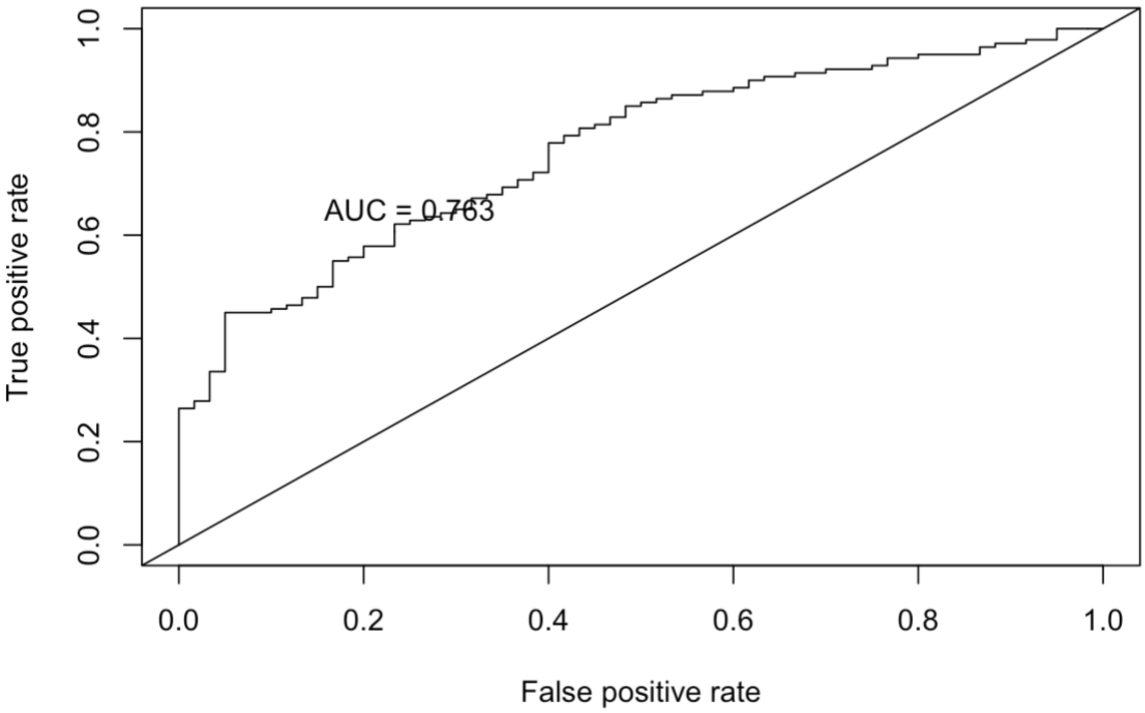

Fig.9: AUC - LDA

In our case, the AUC score is 0.763. This means that the model shows a medium performance at separating classes (predicting bad or good scores) and that it potentially could be improved by selecting different explanatory variables.

Logistic Regression

Some applications of logistic regression are machine learning, most medical fields and social sciences. In fact, logistic regression can be used to predict the risk of developing a disease given certain characteristics of a patient. It can also be used to predict the likelihood of a homeowner defaulting on a mortgage. Due to similarities with the German credit risk classification, we decided to perform predictions by using logistic regression.

The initial model we used is the following:

Response ~ *

where * represents all the variables of the data set.

After creating the model, we observed that a lot of features were not significant enough to impact it. For this reason we performed an AIC Backward variable selection and we were able to reduce the amount of variables used for building the model. In fact we kept only 17 variables with the following model:

RESPONSE ~ CHK_ACCT + DURATION + HISTORY + NEW_CAR + USED_CAR + EDUCATION + AMOUNT + SAV_ACCT + EMPLOYMENT + INSTALL_RATE + MALE_SINGLE + GUARANTOR + PROP_UNKN_NONE + OTHER_INSTALL + RENT + TELEPHONE + FOREIGN

After creating predictions, we computed the confusion matrix and compared the predictions with the test set. The following table summarizes the results:

Confusion matrix	Actual value	
Prediction	**Bad Rating**	**Good Rating**
Bad Rating	34	24
Good Rating	26	116
Accuracy: **0.75** *Sensitivity:* **0.8286** *Specificity:* **0.5667** *Balanced accuracy:* **0.6976**		

We observe that 116 predictions of good ratings were actually good ratings, 34 predictions of bad ratings were actually bad ratings, 24 predictions of bad ratings were actually good ratings and 26 predictions of good ratings were actually bad ratings.

We can note that there is a gap between sensitivity and specificity, which means that the model tends to predict more good ratings (true positives). It can be partially explained by the difference in the amount of good scores and bad scores of the variable Response. Finally, the accuracy of the model is slightly lower than several other models and therefore we don't recommend this model.

Support Vector Machines

The last models we used are the Support Vector Machines (SVMs). SVMs are useful in many applications such as image classification, handwritten character recognition or text categorization. To build our model we decided to use the linear kernel.

In order to select the right parameters for the model we performed some hyperparameter tuning. To do so, we created lists of parameters and we tested the models several times by replacing the parameters at each time until we found the one that performs the best.

The results of the SVM with linear kernel are the following:

Confusion matrix	Actual value	
Prediction	**Bad Rating**	**Good Rating**
Bad Rating	36	12
Good Rating	24	128
Accuracy: **0.82** *Sensitivity:* **0.9143** *Specificity:* **0.6000** *Balanced accuracy:* **0.7571**		

We observe that the model performed the best among all the models that we used so far with an accuracy of 0.82 and a balanced accuracy of 0.7571.

Results

To synthetize, we have produced the following results:

Model	Accuracy	Sensitivity	Specificity	Balanced accuracy
Neural Network	0.733	0.848	0.467	0.657
Classification Tree	0.743	0.905	0.367	0.636
Random Forest	0.707	0.7	0.722	0.711
LDA	0.735	0.55	0.8143	0.6821
Logistic Regression	0.75	0.8286	0.5667	0.6976
SVM Linear	0.82	0.9143	0.6000	0.7571

From the table above, we can see that the best overall models are Random Forest and the SVM using the linear kernel.

Business advice

From the results of our work, we can see that the best machine learning models to predict the response variable are the SVM using a linear kernel and the Random Forest. We recommend these two models as they provide a high level of accuracy and a good sensitivity-specificity trade-off. However, it is important to note that for the SVM Linear, we can see a gap between Sensitivity and Specificity. This can happen because the data we were provided with was highly disproportionate between good and bad ratings in the RESPONSE variable (cf. figure 6: Rating distribution of the RESPONSE variable). Therefore, we recommend training the model with a more balanced distribution between the two classes before definitely using it. Moreover, during our study we observed some incoherences and errors in the data we were provided with, we recommend updating the data collection process in order to avoid that in the future and therefore enhance the quality of future predictions.

Annex - Features Information

Var.#	Variable Name	Description	Variable Type	Description
1.	OBS#	Observation No.	Categorical	
2.	CHK_ACCT	Checking account status	Categorical	$0 : < 0$ DM $1 : 0 < \cdots < 200$ DM $2 : \geq 200$ DM 3 : no checking account
3.	DURATION	Duration of credit in months	Numerical	
4.	HISTORY	Credit history	Categorical	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.	NEW_CAR	Purpose of credit	Binary	car (new) 0 : No, 1 : Yes
6.	USED_CAR	Purpose of credit	Binary	car (used) 0 : No, 1 : Yes
7.	FURNITURE	Purpose of credit	Binary	furniture/equipment 0 : No, 1 : Yes
8.	RADIO/TV	Purpose of credit	Binary	radio/television 0 : No, 1 : Yes
9.	EDUCATION	Purpose of credit	Binary	education 0 : No, 1 : Yes
10.	RETRAINING	Purpose of credit	Binary	retraining 0 : No, 1 : Yes
11.	AMOUNT	Credit amount	Numerical	
12.	SAV_ACCT	Average balance in savings account	Categorical	$0 : < 100$ DM $1 : 100 \leq \cdots < 500$ DM $2 : 500 \leq \cdots < 1000$ DM $3 : \geq 1000$ DM 4 : unknown/no savings account
13.	EMPLOYMENT	Present employment since	Categorical	0 : unemployed $1 : < 1$ year $2 : 1 \leq \cdots < 4$ years $3 : 4 \leq \cdots < 7$ years $4 : \geq 7$ years
14.	INSTALL_RATE	Installment rate as % of disposable income	Numerical	
15.	MALE_DIV	Applicant is male and divorced	Binary	0 : No, 1 : Yes
16.	MALE_SINGLE	Applicant is male and single	Binary	0 : No, 1 : Yes
17.	MALE_MAR_WID	Applicant is male and married or a widower	Binary	0 : No, 1 : Yes
18.	CO−APPLICANT	Application has a co-applicant	Binary	0 : No, 1 : Yes
19.	GUARANTOR	Applicant has a guarantor	Binary	0 : No, 1 : Yes
20.	PRESENT_RESIDENT	Present resident since - years	Categorical	$0 : \leq 1$ year $1 : 1 < \cdots \leq 2$ years $2 : 2 < \cdots \leq 3$ years $3 :> 4$ years
21.	REAL_ESTATE	Applicant owns real estate	Binary	0 : No, 1 : Yes
22.	PROP_UNKN_NONE	Applicant owns no property (or unknown)	Binary	0 : No, 1 : Yes
23.	AGE	Age in years	Numerical	
24.	OTHER_INSTALL	Applicant has other installment plan credit	Binary	0 : No, 1 : Yes
25.	RENT	Applicant rents	Binary	0 : No, 1 : Yes
26.	OWN_RES	Applicant owns residence	Binary	0 : No, 1 : Yes
27.	NUM_CREDITS	Number of existing credits at this bank	Numerical	
28.	JOB	Nature of job	Categorical	0 : unemployed/unskilled - non-resident 1 : unskilled - resident 2 : skilled employee/official 3 : management/self-employed/highly qualified employee/officer
29.	NUM_DEPENDENTS	Number of people for whom liable to provide maintenance	Numerical	
30.	TELEPHONE	Applicant has phone in his or her name	Binary	0 : No, 1 : Yes
31.	FOREIGN	Foreign worker	Binary	0 : No, 1 : Yes