Predictive Analytics: Credit Risk Scorecard Application Case Study: Group 15

Jacob Heye Hilbrands (12229285) Mustafa Alsudani (1214099)
Moritz Renkin (11807211) Nils Klüwer (12229263)

November, 2022

Abstract

Credit scorecards are crucial tools in the credit assessment process. They are based on the prior performance of clients that share the same traits as a new client. Consequently, the goal of a credit scorecard is to estimate risk because credit scorecards are based on the past behavior of clients that share the same qualities as a prospective client. Therefore, the primary purpose of it is to either approve or reject a new client's loan request. The scorecard's function is to support this option. In this assignment, we'll compare and contrast the binning techniques known as weight-of-evidence (woe) binning and group binning. That's why we have been using the existing scorecard. Moreover, certain model changes and binning splits will be made as a result of the addition of the 7 variables. To evaluate the effectiveness of the scorecard, the Gini coefficient is used. The major results of this research were to: demonstrate that group binning outperforms woe binning in out-of-sample predictions as measured by the Gini coefficient. The dependent variable, credibility, may be predicted more precisely when more independent variables are included.

Contents

Abstract

Credit scorecards are crucial tools in the credit assessment process. They are based on the prior performance of clients that share the same traits as a new client. Consequently, the goal of a credit scorecard is to estimate risk because credit scorecards are based on the past behavior of clients that share the same qualities as a prospective client. Therefore, the primary purpose of it is to either approve or reject a new client's loan request. The scorecard's function is to support this option. In this assignment, we'll compare and contrast the binning techniques known as weight-of-evidence (woe) binning and group binning. That's why we have been using the existing scorecard. Moreover, certain model changes and binning splits will be made as a result of the addition of the 7 variables. To evaluate the effectiveness of the scorecard, the Gini coefficient is used. The major results of this research were to: demonstrate that group binning outperforms woe binning in out-of-sample predictions as measured by the Gini coefficient. The dependent variable, credibility, may be predicted more precisely when more independent variables are included.

1 Introduction

In the form of a score and a likelihood of default, credit scoring models and scorecards estimate the risk that a borrower won't return a loan.

For instance, a credit scorecard may award a borrower points based on the following table for their age and income. As previously noted, a single dependent variable, credibility, was predicted using seven independent variables. As a binning strategy, the Weight-of-Evidence technique was adopted.

Therefore, group binning will be used as an extra binning strategy for this project's execution. To increase the model's Gini coefficient of prediction, a seventh independent variable was added, and the binning breaks were modified.

2 Predictive Analytics Research Methodology

2.1 Predictor Variable Transformation

Transforming predictor variables with the Weight of Evidence (WOE) approach. The WOE describes how well an independent variable may predict the outcome of a dependent variable.

The WOE cannot be determined until the data has been separated into bins. This value is used for future calculations in place of the original value once the WOE has been located. As a result, upon transformation, the WOE will be the same for all lines that include the variable in the same bin.

If the WOE is greater than one, then the Distribution of the Goods is greater than the Bads, if the WOE is smaller than one, then the Distribution of the Goods is less than the Distribution of the Bads.

There are benefits to this transformation: Because the data is categorized with the simple treatment of the missing values, outliers are no longer a concern. Additionally, it supports categorical and continuous numbers.

Using the two functions in R:

woebin: This function determines the best binning.

woebin_ply: This function applies the provided binning information to convert the values in the original data to WOE values.

There will be some drawbacks for WOE as well. One is that binning may result in information loss. Another is the possibility of an unresearched link between the independent variables.

$$WOE = \frac{Distribution of Goods}{Distribution of Bads} \tag{1}$$

2.2 Logistic Regression Analysis

Logistic Regression (also known as Logit-Model) is a statistical Model that can estimate probability of a certain event happening based on one or more independent variables. Its application is widespread in various statistical methods, especially in classification problems and prediction analyses. Contrary to linear regression, the predicted variable in logistic regression is a Bernoulli variable, i.e. a binary random variable k with:

$$k \in \{0, 1\} \tag{2}$$

Formally, the logistic regression model estimates probability p of the Bernoulli k being 1, corresponding the the event in question happening. The logistic function defining $p(x)$ takes on the form:

$$p(k) = \frac{e^{\beta_0 + \beta_1 k}}{1 + e^{\beta_0 + \beta_1 k}} = \frac{1}{1 + e^{-}(\beta_0 + \beta_1 k)} \tag{3}$$

Figure TODO depicts the logistic regression for two examplary attributes.

In the domain of Credit Scoring, logistic regression is one of the most widely used statistical models. (Bolton u. a., 2009, p. 19)

2.3 Building Scorecards

lorem impsum

2.4 Forecasting Scorepoints and default Probabilities

lorem impsum

2.5 Forecasting Accuracy Testing

Include test-train split

3 Empirical results

lorem impsum

3.1 Loading and Preparing the Data

The dataset "germancredit" is used from the library "scorecard". Below the import process is shown. The "germancredit" data has 21 variables, 7 will remain for our scorecard model and 1 dependent variable the "creditability". In the following the explained theoretical approach is used to filter out the 7 most significant predictor variables, relevant for the credibility of a possible customer.

```
# Importing of library and data
library(scorecard)
data("germancredit")
ncol(germancredit)
```

The first step of finding relevant predictor variables is done with the "var_filter()" function. The default limits and rates for iv_limit, missing_rate & identical_rate mentioned below are used. The iv_limit excludes every variables whose information value is lower or equal to 0.02 in respect to our depedent variable "creditability". Through this filtering process 7 predictor variables are already excluded which are not eligible for the scorecard model. In the data_f.df 14 varibales are remaing, 13 posible predictor varibales and one dependent variable "creditablitity".

```r
# Filtering of variables with iv_limit >= 0.02, missing_rate <= 0.95
# and identical_rate <= 0.95
data_f.df = var_filter(germancredit, y="creditability")
```

```r
ncol(data_f.df)
```

```
## [1] 14
```

3.2 Splitting the Data into Train and Test Samples

The remaining 13 possible predictor varibales are splitted with the split_df function into train and test data with a ratio fo 75% train and 25% test data. After that data_f.list is reformated to be useable in the later process.

```r
# Splitting data into train and test data with ratio 0.75
data_f.list = split_df(data_f.df,"creditability",ratio=c(0.75,0.25))
class(data_f.list)
lapply(data_f.list,class)
lapply(data_f.list, dim)
```

The "default.list" is generated for later use In-Sample and Out-of-Sample testing.

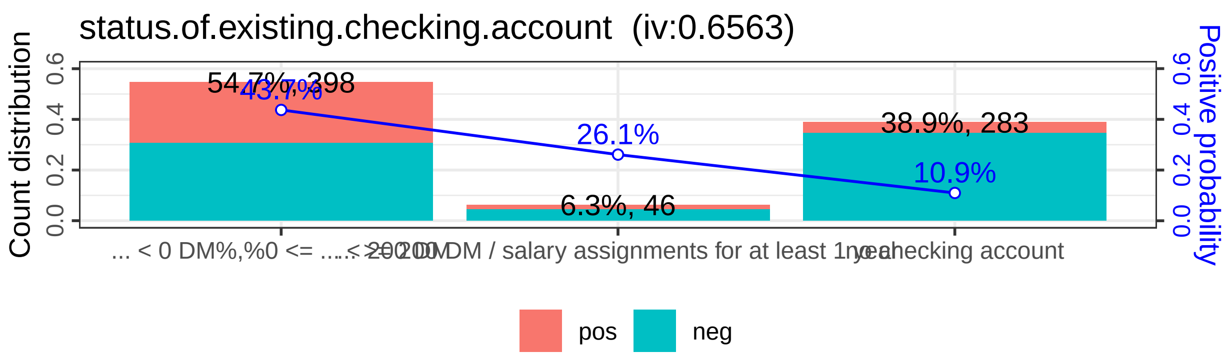
3.3 Weight-Of-Evidence (WOE)-Binning

In the following the WOE-Binning is done with the 'woebin' and 'woebin_ply' function. The breaks are generated automated by the function and are saved in the "breaks.list". For the 'woebin' the default 'method="width"' is used.

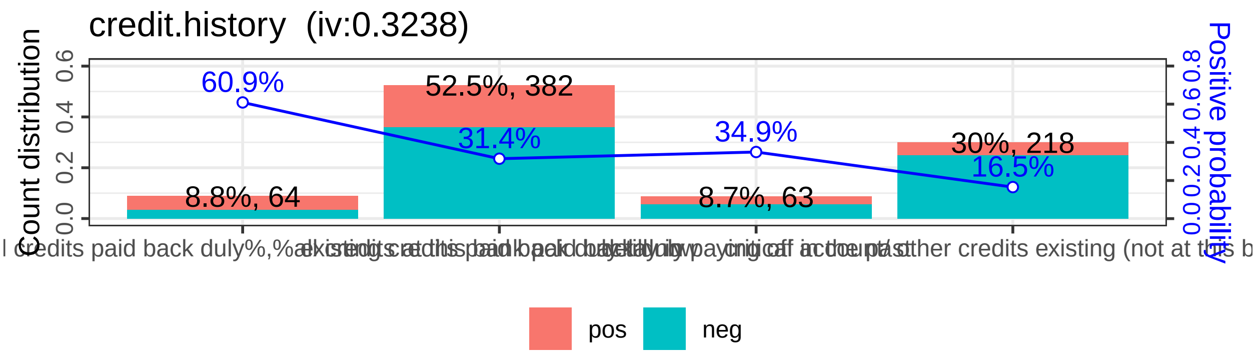
```r
# Binning of train data
# breaks.list is saved and imported seperately
bins.list = woebin(data_f.list$train,
                   "creditability",
                   save_breaks_list = "breaks.list")
```

Below the 7 predictor variables with the highest information value are shown in descending order. This does not mean that these values are the final 7 predictor variables. The Full plots of all predictor variables can be viewed in the Appendix.

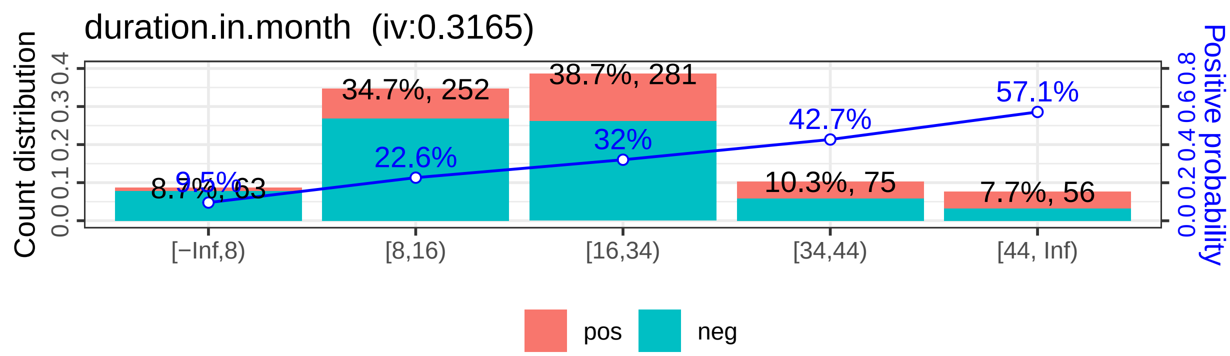
```
## $status.of.existing.checking.account
```



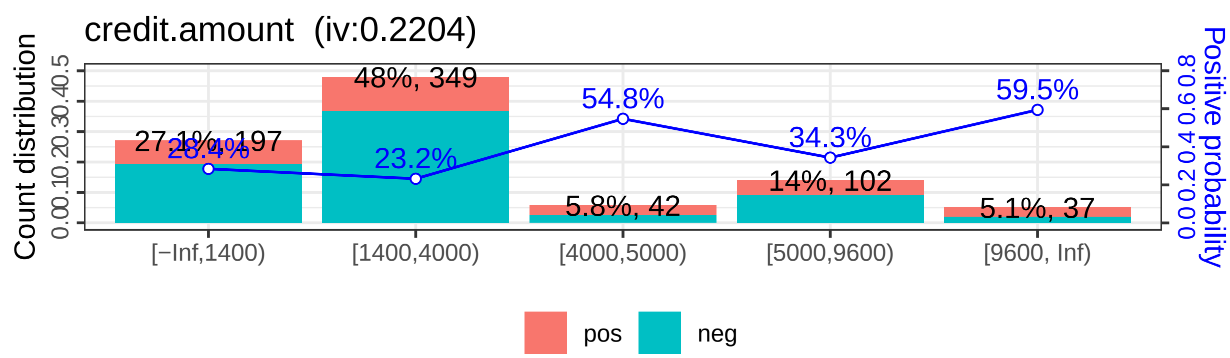

```
## $credit.history
```



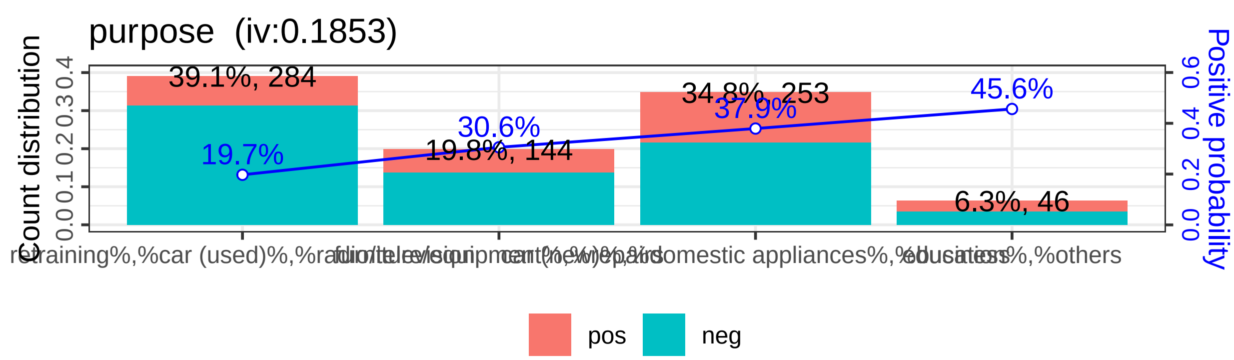

```
## $duration.in.month
```



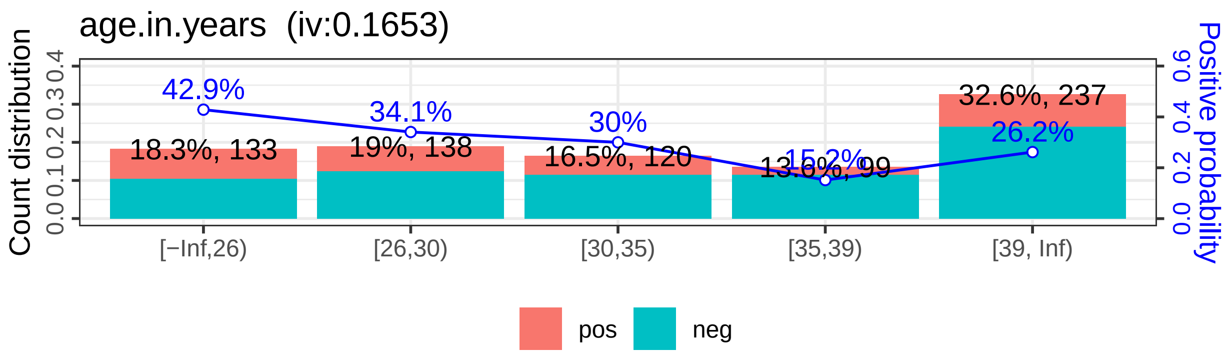

```
## $credit.amount
```



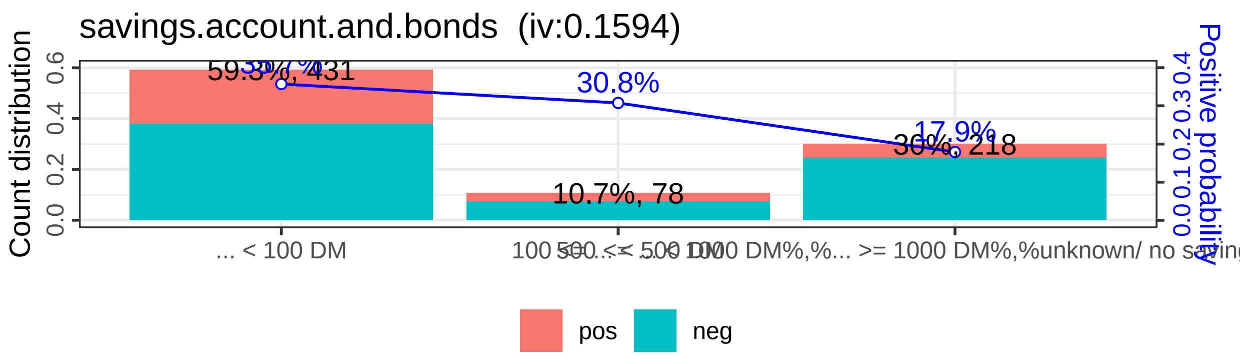

```
## $purpose
```




```
## $age.in.years
```




```
## $savings.account.and.bonds
```



After the WOE-Binning, the train and test data is transformed with the generated 'bins.list' into the prescribed format for later usage in glm and scorecard building.

```
# WOE-Transformation of train and test data
data_woe.list = lapply(data_f.list,
                       function(x) woebin_ply(x, bins.list))
lapply(data_woe.list, class)
lapply(data_woe.list, dim)
```

Transformation of "data_f.list", "bins.list" into data_grp.list - Bin-Group (GRP). We will later use this for validation and the glm. The functionality of building a scorecard with the GRP-Binning isn't implemented in the package at the current state.

```
# Bin-Group (GRP) Transformation of train and test data
data_grp.list = lapply(data_f.list,
                       function(x) woebin_ply(x, bins.list, to = 'bin'))
lapply(data_grp.list, class)
lapply(data_grp.list, dim)
```

3.4 Generalized linear model (glm): Regressing response w.r.t. predictors

In this section, the logistic regression models are built. These models are used to select the seven most significant variables and subsequently to build the scorecard model with the chosen variables.

3.4.1 Selection of seven variables with Logistic regression w.r.t. WOE-transformed predictors

In order to select the seven most significant variables, a logistic regression is performed on the WOE-transformed predictor variables of the train data. The seven predictor variables which have the lowest significance value, are selected.

```r
data_woe_first_iteration.glm <- glm(creditability ~ .,
                                    family = binomial(),
                                    data = data_woe.list$train)
summary(data_woe_first_iteration.glm)
```

```
##
## Call:
## glm(formula = creditability ~ ., family = binomial(), data = data_woe.list$train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0742  -0.6943  -0.3671   0.7495   2.4750
##
## Coefficients:
##                                                    Estimate Std. Error
## (Intercept)                                        -0.85979    0.09958
## status.of.existing.checking.account_woe             0.82357    0.12779
## duration.in.month_woe                               0.77204    0.19631
## credit.history_woe                                  0.66714    0.18079
## purpose_woe                                         1.01485    0.23555
## credit.amount_woe                                   0.73487    0.21701
## savings.account.and.bonds_woe                       0.80377    0.26016
## present.employment.since_woe                        0.59775    0.33021
## installment.rate.in.percentage.of.disposable.income_woe  2.95015    1.62616
## other.debtors.or.guarantors_woe                     1.17259    0.60659
## property_woe                                        0.12139    0.39847
## age.in.years_woe                                    0.96868    0.26093
## other.installment.plans_woe                         0.62812    0.58047
## housing_woe                                         0.37643    0.40013
##                                                    z value Pr(>|z|)
## (Intercept)                                         -8.634  < 2e-16 ***
## status.of.existing.checking.account_woe              6.445 1.16e-10 ***
## duration.in.month_woe                                3.933 8.40e-05 ***
## credit.history_woe                                   3.690 0.000224 ***
## purpose_woe                                          4.308 1.64e-05 ***
## credit.amount_woe                                    3.386 0.000708 ***
## savings.account.and.bonds_woe                        3.089 0.002005 **
## present.employment.since_woe                         1.810 0.070265 .
## installment.rate.in.percentage.of.disposable.income_woe   1.814 0.069650 .
## other.debtors.or.guarantors_woe                      1.933 0.053226 .
## property_woe                                         0.305 0.760642
## age.in.years_woe                                     3.712 0.000205 ***
## other.installment.plans_woe                          1.082 0.279215
## housing_woe                                          0.941 0.346826
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 886.32  on 726  degrees of freedom
## Residual deviance: 658.56  on 713  degrees of freedom
## AIC: 686.56
##
## Number of Fisher Scoring iterations: 5
```

The following seven predictor variables are selected as they have the lowest significance value (<0.01):

- status.of.existing.checking.account
- duration.in.month
- credit.history
- purpose
- credit.amount
- savings.account.and.bonds
- age.in.years

3.4.2 Logistic regression w.r.t. selected WOE-transformed predictors

The logistic regression model with the seven selected WOE-transformed predictor variables is built on the train data and saved in "data_woe_second_iteration.glm".

```
data_woe_second_iteration.glm <- glm(creditability ~
                                       status.of.existing.checking.account_woe
                                       + duration.in.month_woe
                                       + credit.history_woe
                                       + purpose_woe
                                       + credit.amount_woe
                                       + savings.account.and.bonds_woe
                                       + age.in.years_woe,
                                         family = binomial(),
                                         data = data_woe.list$train)
```

3.4.3 Logistic regression w.r.t. selected GRP-transformed predictors

The logistic regression model with the seven selected GRP-transformed predictor variables is built on the train data and saved in "data_grp_second_iteration.glm".

```
data_grp_second_iteration.glm <- glm(creditability ~
                                       status.of.existing.checking.account_bin
                                       + duration.in.month_bin
                                       + credit.history_bin
                                       + purpose_bin
                                       + credit.amount_bin
                                       + savings.account.and.bonds_bin
                                       + age.in.years_bin,
                                      family = binomial(),
                                      data = data_grp.list$train)
```

3.5 Building the scorecard-model

3.5.1 Scorecard-model w.r.t. WOE-transformed predictors

The scorecard is built with the logistic regression model of the seven WOE-transformed variables and is saved in "scorecard_woe_second_iteration.scm". The scorecard contains the baseline points and the points associated with every bin. In the output of the command `scorecard_woe_second_iteration.scm$purpose`, the points for each bin of the "purpose" predictor variable is displayed. When the purpose of a credit is e.g. furniture, equipment or repairs, the credit applicant receives -3 points for this predictor variable.

```r
scorecard_woe_second_iteration.scm <- scorecard(bins.list,
                                              data_woe_second_iteration.glm)
names(scorecard_woe_second_iteration.scm)
```

```
## [1] "basepoints"                          "status.of.existing.checking.account"
## [3] "duration.in.month"                   "credit.history"
## [5] "purpose"                             "credit.amount"
## [7] "savings.account.and.bonds"           "age.in.years"
```

```r
scorecard_woe_second_iteration.scm$purpose
```

```
##    variable                                        bin count count_distr neg
## 1:  purpose retraining%,%car (used)%,%radio/television   284  0.39064649 228
## 2:  purpose                furniture/equipment%,%repairs   144  0.19807428 100
## 3:  purpose car (new)%,%domestic appliances%,%business   253  0.34800550 157
## 4:  purpose                          education%,%others    46  0.06327373  25
##    pos   posprob         woe       bin_iv  total_iv
## 1:  56 0.1971831 -0.54948057 0.1038486990 0.1853448
## 2:  44 0.3055556  0.03353282 0.0002242187 0.1853448
## 3:  96 0.3794466  0.36261576 0.0487911020 0.1853448
## 4:  21 0.4565217  0.68015999 0.0324807583 0.1853448
##                                        breaks is_special_values points
## 1: retraining%,%car (used)%,%radio/television             FALSE     43
## 2:              furniture/equipment%,%repairs             FALSE     -3
## 3: car (new)%,%domestic appliances%,%business             FALSE    -28
## 4:                         education%,%others             FALSE    -53
```

The scores for the entire "germancredit" data are calculated and saved in "score_woe_second_iteration.df"

```r
score_woe_second_iteration.df = scorecard_ply(germancredit,
                                            scorecard_woe_second_iteration.scm,
                                            only_total_score = FALSE)
```

The scores for the splitted "germancredit" data (train and test) are calculated and saved in "score_woe_second_iteration.list".

```r
score_woe_second_iteration.list <- lapply(data_f.list,
                                        function(x) scorecard_ply(x,
                                           scorecard_woe_second_iteration.scm))
```

A report can also be generated for this scorecard model which includes information on the dataset, model coefficients, model performance, WOE binning, scorecard, population stability, and gains.

```r
# Report of Scoreboard with WOE-transformed predictor variables
y<-"creditability"
x<-c("status.of.existing.checking.account","duration.in.month","credit.history",
      "purpose","credit.amount","savings.account.and.bonds",
      "age.in.years")

report(data_f.list,
       y,
       x,
       breaks.list,
       seed = NULL,
       save_report = "Report_WOE_second_iteration")
```

3.5.2 Scorecard-model w.r.t. GRP-transformed predictors

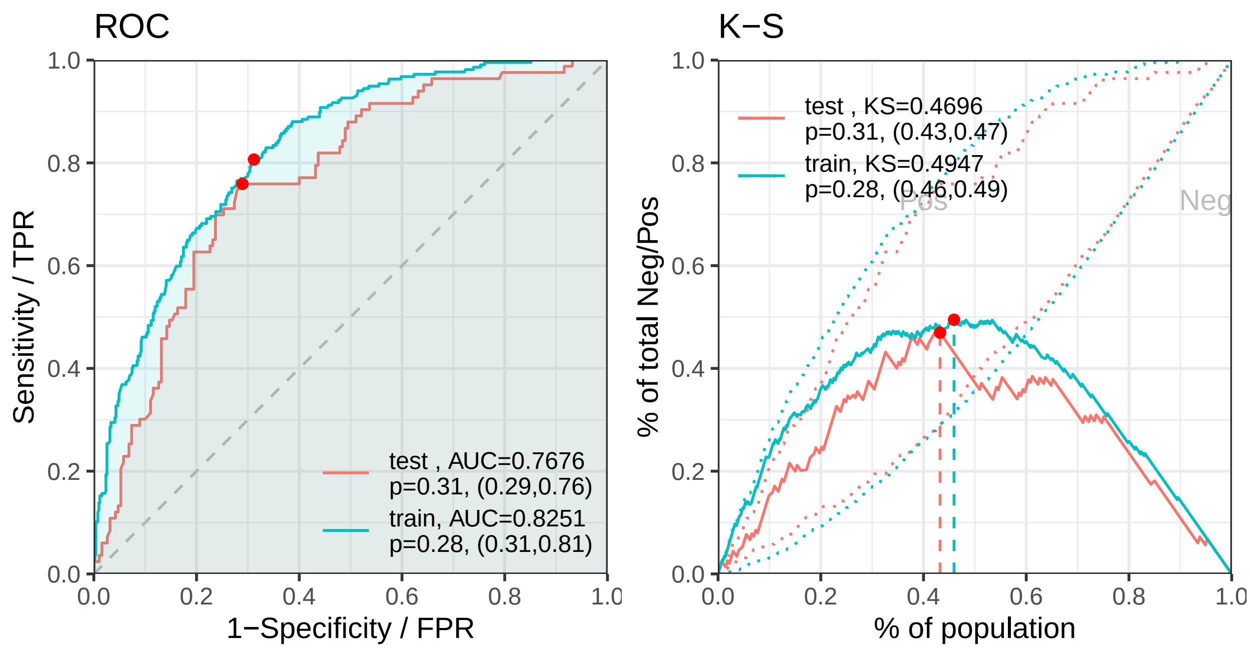
Building a scorecard with GRP-transformed predictor variables is unfortunately not supported by the “scorecard” library. Therefore, no report can be created as well. Nevertheless, the performance of the logistic model of the GRP-transformed predictor variables is compared with the logistic model of the WOE-transformed predictor variables in the next section.

3.6 Predicting (forecasting) probabilities and scorepoints

todo

3.7 Gini Coefficient In-Sample and Out-of-Sample

```
## [INFO] The threshold of confusion matrix is 0.2756.
```




```
## $binomial_metric
## $binomial_metric$train
##         Gini       AUC        R2      RMSE
## 1: 0.6502033 0.8251017 0.2775988 0.3889285
##
## $binomial_metric$test
##         Gini       AUC        R2      RMSE
## 1: 0.5352568 0.7676284 0.1618554 0.4211266
##
##
## $confusion_matrix
## $confusion_matrix$train
##    label pred_0 pred_1     error
## 1:     0    351    159 0.3117647
## 2:     1     42    175 0.1935484
## 3: total    393    334 0.2764787
##
## $confusion_matrix$test
##    label pred_0 pred_1     error
## 1:     0    128     62 0.3263158
```

```
## 2:      1     20     63 0.2409639
## 3: total    148    125 0.3003663
##
##
## $pic
## TableGrob (1 x 2) "arrange": 2 grobs
##   z     cells    name           grob
## 1 1 (1-1,1-1) arrange gtable[layout]
## 2 2 (1-1,2-2) arrange gtable[layout]
```

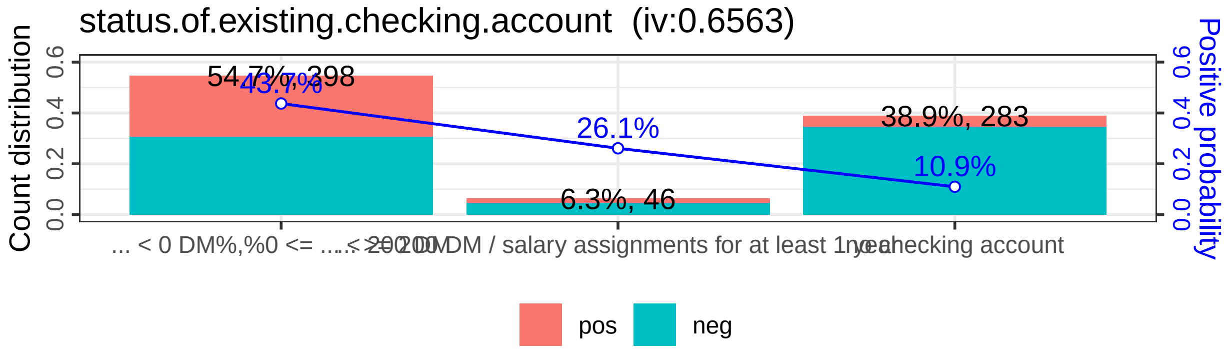
4 Summary

TODO

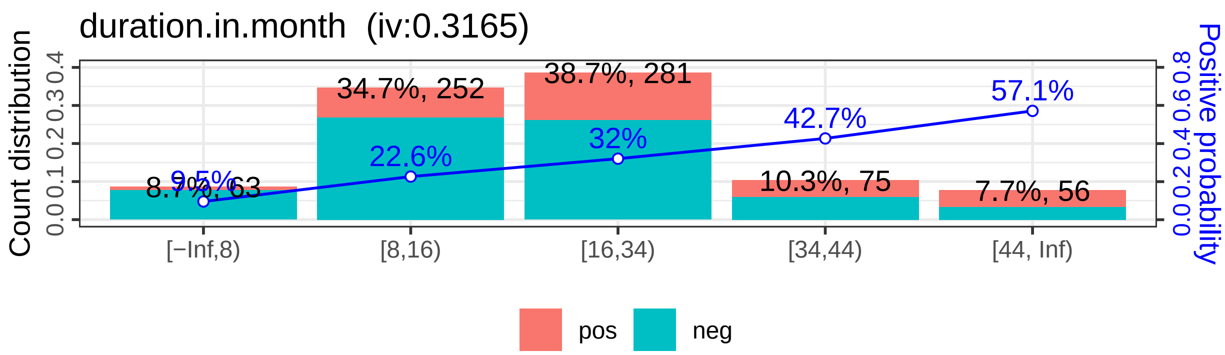
5 Appendix

5.1 Full List of WOE-Binnined predictor variables

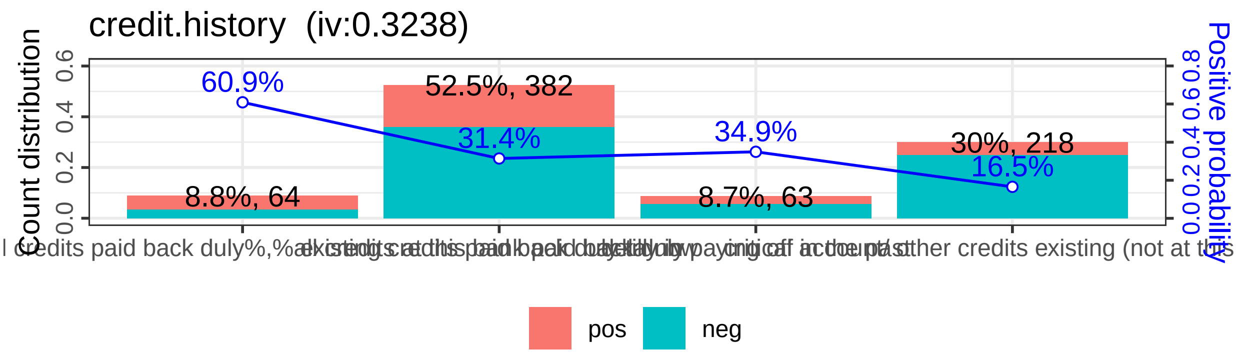
```
## $status.of.existing.checking.account
```



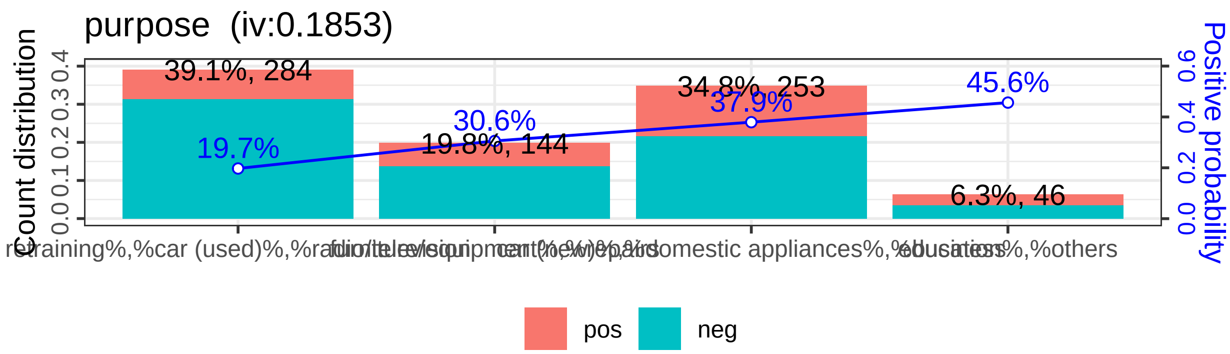

```
##
## $duration.in.month
```



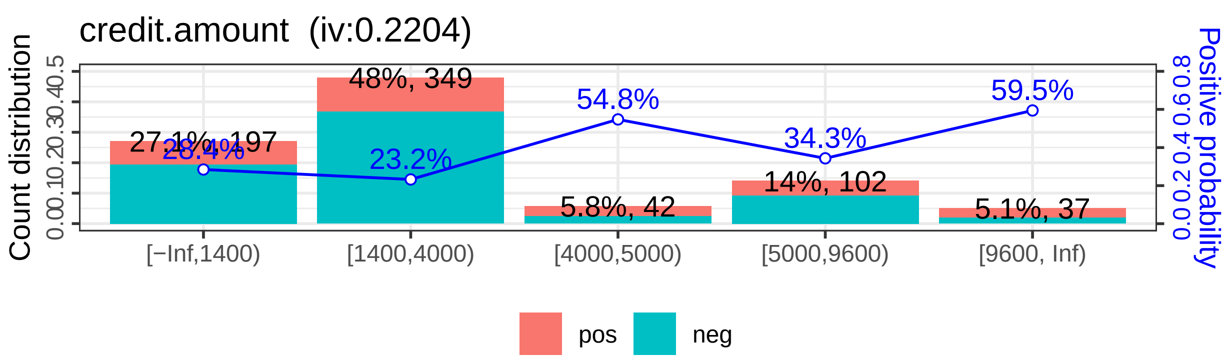

```
##
## $credit.history
```



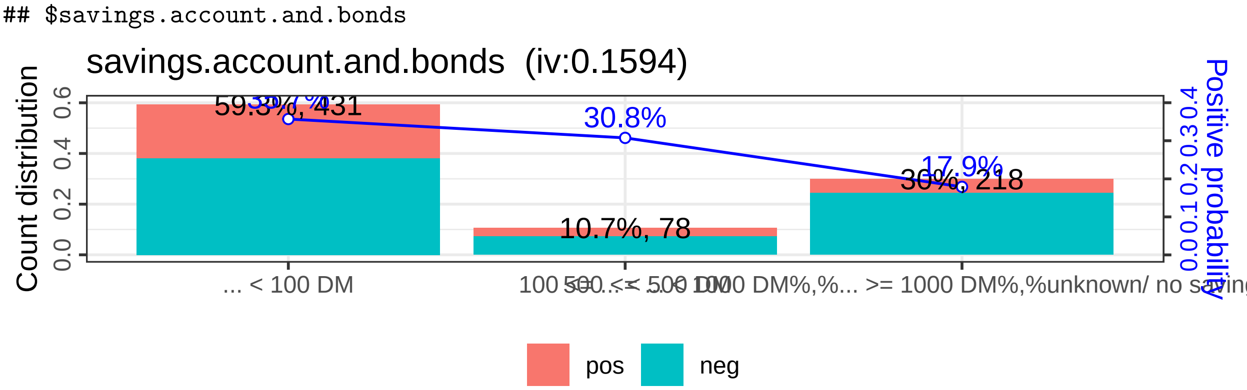

```
##
## $purpose
```




```
##
## $credit.amount
```

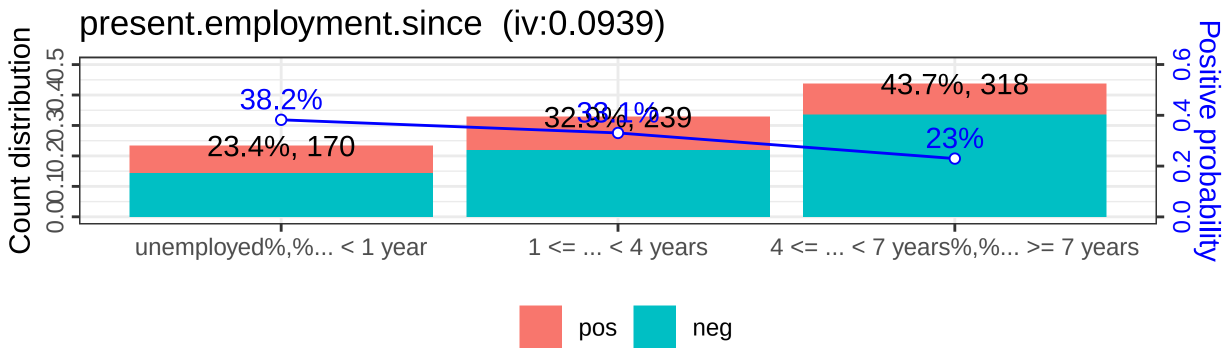



```
##
## $savings.account.and.bonds
```

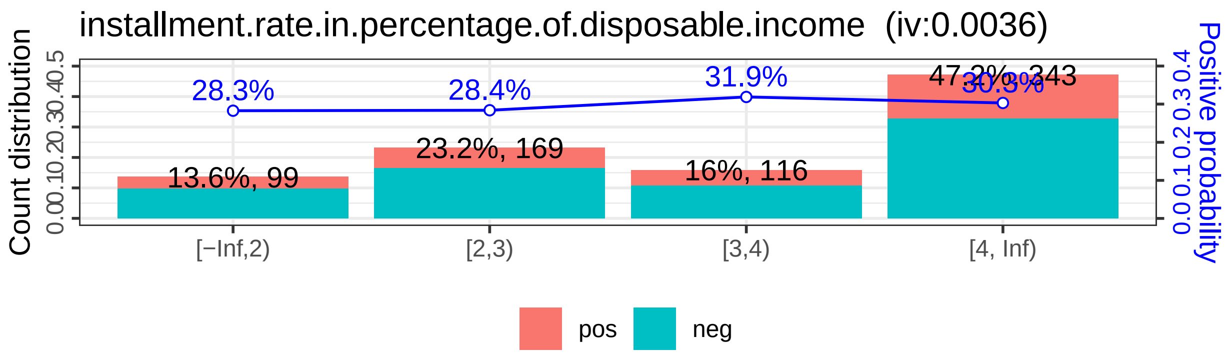



```
##
```

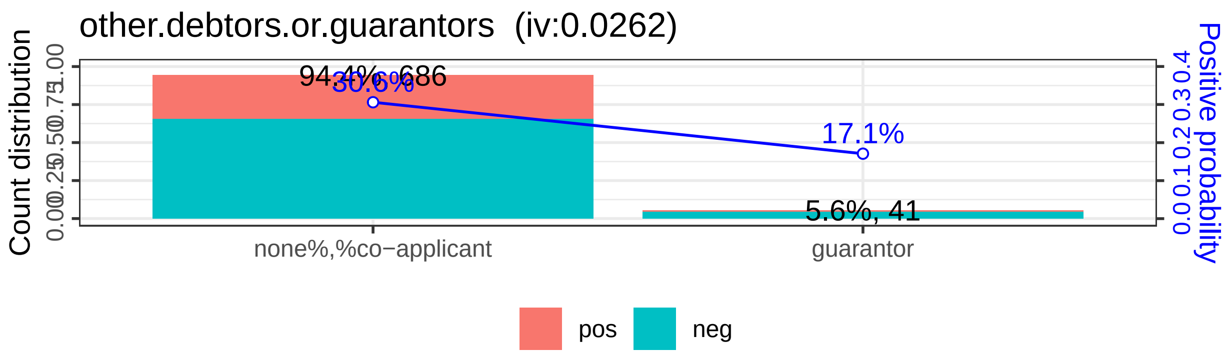
```
## $present.employment.since
```



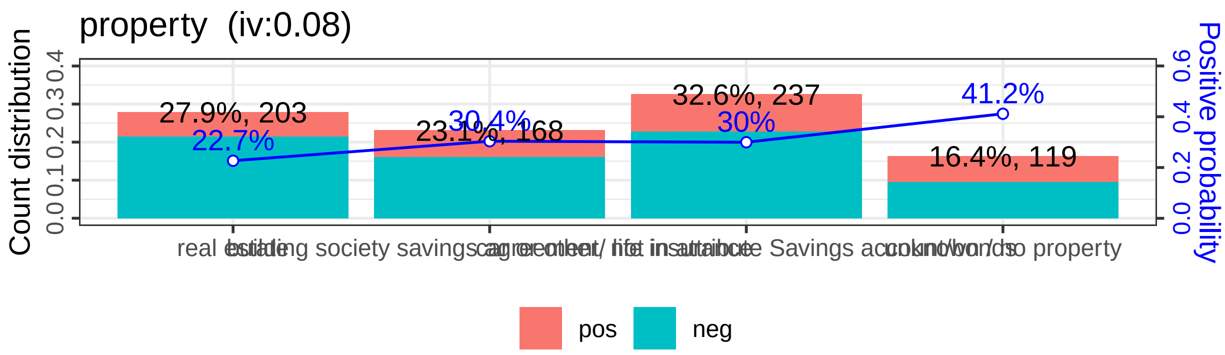

```
##
## $installment.rate.in.percentage.of.disposable.income
```



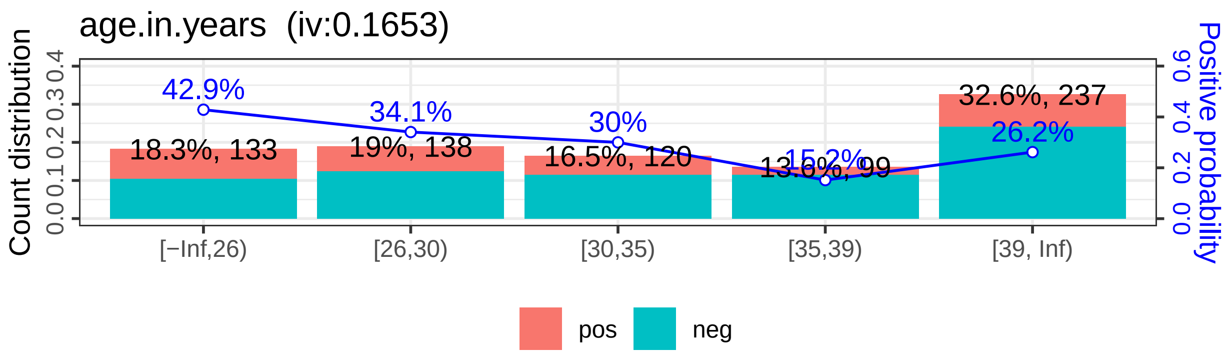

```
##
## $other.debtors.or.guarantors
```



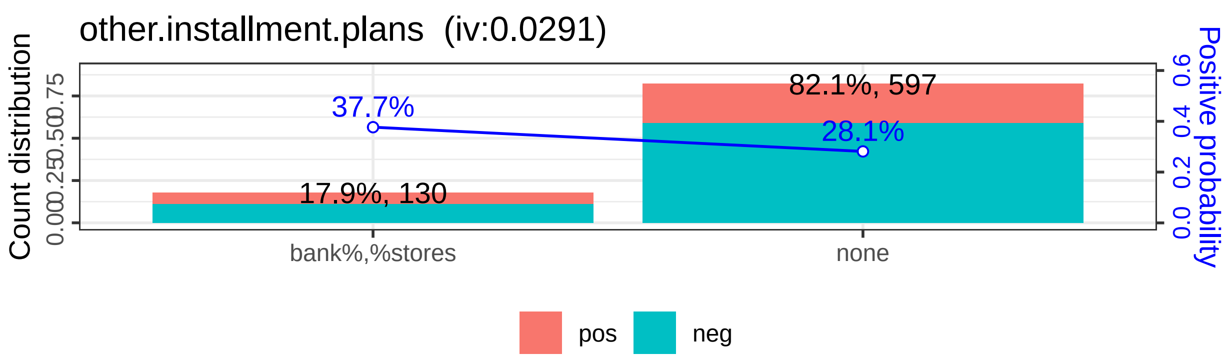

```
##
## $property
```



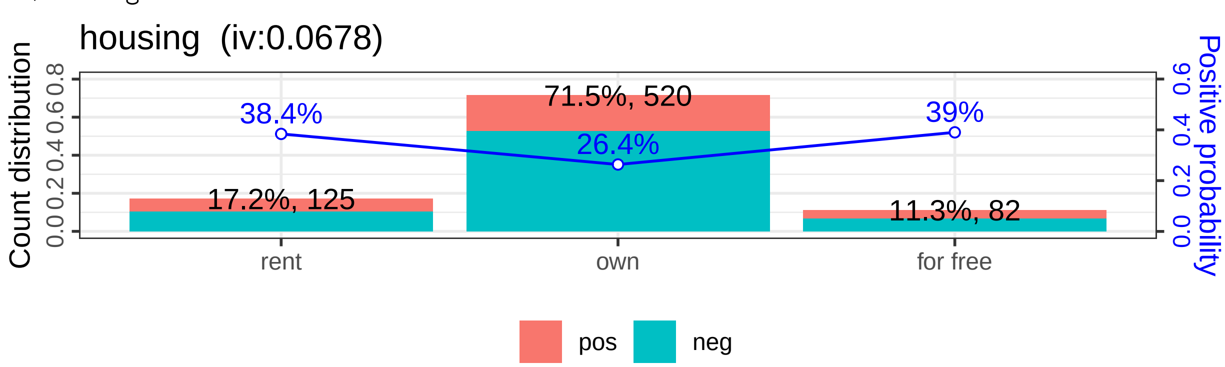

```
##
## $age.in.years
```




```
##
## $other.installment.plans
```




```
##
## $housing
```



References

[Bolton u. a. 2009] Bolton, C. ; Mathematics, University of Pretoria. Department o. ; Mathematics, Applied: *Logistic Regression and Its Application in Credit Scoring.* University of Pretoria, 2009 https://books.google.at/books?id=K7B3MwEACAAJ