CREDIT DEFAULT PREDICTION

Shiladitya Bose

2022-07-15

INTRODUCTION

Granting credit to customers is the core business of a bank. In doing so, banks need to have adequate systems to decide to whom to grant credit. Credit scoring is a key risk assessment technique to analyze and quantify a potential obligor's credit risk. Essentially, credit scoring aims at quantifying the likelihood that an obligor will repay the debt. The outcome of the credit scoring exercise is a score reflecting the creditworthiness of the obligor. Throughout the past few decades banks have gathered plenty of information describing the default behavior of their customers. Examples are historical information about a customer's date of birth, gender, income, employment status, and so on. All this data has been nicely stored into huge (e.g., relational) databases or data warehouses. On top of this, banks have accumulated lots of business experience about their credit products. As an example, many credit experts do a pretty good job of discriminating between low-risk and high-risk mortgages using their business expertise only. It is now the aim of credit scoring to analyze both sources of data in more detail and come up with a statistically based decision model that allows scoring future credit applications and ultimately deciding which ones to accept and which to reject. For the historical customers, we know which ones turned out to be good payers and which ones turned out to be bad payers. This good/bad status is now the binary target variable Y, which we will relate to all information available at scoring time about our obligors. The goal of credit scoring is now to quantify this relationship as precisely as possible to assist credit decisions, monitoring, and management. Banks score borrowers at loan application, as well as at regular times during the term of a financial contract (generally loans, loan commitments, and guarantees). Once we have our credit scoring model built, we can then use it to decide whether the credit application should be accepted or rejected, or to derive the probability of a future default. To summarize, credit scoring is a key risk management tool for a bank to optimally manage, understand, and model the credit risk it is exposed to.

OBJECTIVE

1. Missing value imputation
2. Variable selection using correlation plot, Stepwise selection and IV
3. Devolop logistic model based on best selected model
4. Calculate PD
5. Build a Credit Scorecard

DATASET DESCRIPTION

```
## [1] 7500   15
```

Here our dataset contains 7500 data points and 15 columns.

```
## 'data.frame':    7500 obs. of  15 variables:
##  $ Home.Ownership              : chr  "Own Home" "Own Home" "Home Mortgage" "Own Home" ...
##  $ Annual.Income               : num  482087 1025487 751412 805068 776264 ...
##  $ Years.in.current.job        : chr  NA "10+ years" "8 years" "6 years" ...
```

```
##  $ Number.of.Open.Accounts     : num  11 15 11 8 13 12 9 13 17 10 ...
##  $ Years.of.Credit.History     : num  26.3 15.3 35 22.5 13.6 14.6 20.3 12 15.7 24.6 ...
##  $ Maximum.Open.Credit         : num  685960 1181730 1182434 147400 385836 ...
##  $ Number.of.Credit.Problems   : num  1 0 0 1 1 0 0 0 1 0 ...
##  $ Months.since.last.delinquent: num  NA NA NA NA NA NA 73 18 NA 6 ...
##  $ Bankruptcies                : num  1 0 0 1 0 0 0 0 1 0 ...
##  $ Purpose                     : chr  "debt consolidation" "debt consolidation" "debt consolidation"
##  $ Term                        : chr  "Short Term" "Long Term" "Short Term" "Short Term" ...
##  $ Current.Loan.Amount         : num  99999999 264968 99999999 121396 125840 ...
##  $ Current.Credit.Balance      : num  47386 394972 308389 95855 93309 ...
##  $ Monthly.Debt                : num  7914 18373 13651 11338 7180 ...
##  $ Credit.Default              : int  0 1 0 0 0 1 0 1 0 1 ...
```

Here Our Response or Target variable is Credit Default, which contains binary response. "1" stands for Default and "0" stand for Not-Default. And rest of the 14 variables are Expanetory variables.

CHECKING MISSING VALUES AND DUPLICATE VALUES

1.CHECKING DUPLICATE VALUES

```
## [1] 0
```

Our data contains no dulpicate values.

2.CHECKING MISSING VALUES

```
##               Home.Ownership                Annual.Income
##                            0                         1557
##         Years.in.current.job      Number.of.Open.Accounts
##                          371                            0
##      Years.of.Credit.History          Maximum.Open.Credit
##                            0                            0
##    Number.of.Credit.Problems Months.since.last.delinquent
##                            0                         4081
##                 Bankruptcies                      Purpose
##                           14                            0
##                         Term          Current.Loan.Amount
##                            0                            0
##       Current.Credit.Balance                 Monthly.Debt
##                            0                            0
##               Credit.Default
##                            0
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:purrr':
##
##     set_names
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

Hence, our dataset contains a huge number of missing values. So, now we impute missing values using different techniques.

MISSING VALUE IMPUTATION

Now we check the distributions of missing values according to the Credit Default and Not Default of Missing explanatory variables. At first, we will see it for Annual Income. In general, the intution may suggest that the Missing rows in Annual Income implies the customers are not an earning person. So, impute these missing rows with 0 is an idea. But, before doing it, we must see the distributions of missing values according to the Credit Default and Not Default.

```
## Warning: package 'sqldf' was built under R version 4.2.1
```

```
## Loading required package: gsubfn
```

```
## Warning: package 'gsubfn' was built under R version 4.2.1
```

```
## Loading required package: proto
```

```
## Warning: package 'proto' was built under R version 4.2.1
```

```
## Loading required package: RSQLite
```

```
## Warning: package 'RSQLite' was built under R version 4.2.1
```

```
## Credit.Default
##    0    1
## 1028  529
```

Here, ratio of defaulters and non-defaulters is 1:2. So, Impute the missing values using 0 is not a good idea here. So, we use here MissForest algorithm for missing value imputation.

```
## Warning: package 'missForest' was built under R version 4.2.1
```

Then, we will see it for Bankruptcies. In general, here also the intution may suggest that the Missing rows in Bankruptcies implies the customers are not an earning person. So, bankruptcies is not an issue for him/her. So, impute these missing rows with 0 is an idea. But, before doing it, we must see the distributions of missing values according to the Credit Default and Not Default.

```
## Credit.Default
##  0  1
## 10  4
```

Here, also the ratio of defaulters and non-defaulters is nearly 1:2. So, Impute the missing values using 0 is not a good idea here. So, we use here Apriori algorithm for missing value imputation.

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
##
## Attaching package: 'arules'

## The following object is masked from 'package:dplyr':
##
##     recode

## The following objects are masked from 'package:base':
##
##     abbreviate, write

##
##    0    1    2    3    4
## 6660  786   31    7    2

##      Home.Ownership Number.of.Open.Accounts Number.of.Credit.Problems
## 101        Own Home                       9                         0
## 257            Rent                       6                         0
## 258   Home Mortgage                      15                         0
## 899            Rent                      20                         0
## 1405           Rent                       4                         0
## 3064           Rent                       5                         0
## 3253           Rent                       7                         0
## 3352  Home Mortgage                      14                         0
## 3402           Rent                       2                         0
## 3497           Rent                       7                         0
## 4335           Rent                       9                         0
## 5567       Own Home                      12                         0
## 7185           Rent                       3                         0
## 7380       Own Home                      16                         0
##      Bankruptcies              Purpose       Term Current.Loan.Amount
## 101          <NA> educational expenses Short Term            99999999
## 257          <NA>   debt consolidation Short Term            99999999
## 258          <NA>   debt consolidation Short Term              447480
## 899          <NA>   debt consolidation Short Term              456808
## 1405         <NA>                other Short Term               11242
## 3064         <NA>                other Short Term               44814
## 3253         <NA>        business loan Short Term              156970
## 3352         <NA>   debt consolidation Short Term              528968
## 3402         <NA>                other Short Term            99999999
## 3497         <NA> educational expenses Short Term              210166
## 4335         <NA>   debt consolidation Short Term              167882
## 5567         <NA>                other Short Term               92620
## 7185         <NA>   debt consolidation Short Term               46706
## 7380         <NA>       small business Short Term               71170
##      Credit.Default
## 101               A
## 257               A
## 258               A
## 899               B
## 1405              A
## 3064              B
## 3253              A
## 3352              A
## 3402              A
## 3497              A
```

```
## 4335              A
## 5567              A
## 7185              B
## 7380              B
```

```
##            Home.Ownership   Number.of.Open.Accounts Number.of.Credit.Problems
##                         0                         0                         0
##              Bankruptcies                   Purpose                      Term
##                         0                         0                         0
##       Current.Loan.Amount            Credit.Default
##                         0                         0
```

```
##      lhs                                   rhs                  support confidence  coverage      li
## [1]  {Bankruptcies=A,
##       Term=Short Term}                  => {Credit.Default=A} 0.5018702  0.7672044 0.6541544 1.06812
## [2]  {Number.of.Credit.Problems=[0,7],
##       Bankruptcies=A,
##       Term=Short Term}                  => {Credit.Default=A} 0.5018702  0.7672044 0.6541544 1.06812
## [3]  {Term=Short Term}                  => {Credit.Default=A} 0.5675928  0.7666907 0.7403153 1.06740
## [4]  {Number.of.Credit.Problems=[0,7],
##       Term=Short Term}                  => {Credit.Default=A} 0.5675928  0.7666907 0.7403153 1.06740
## [5]  {Purpose=debt consolidation}       => {Credit.Default=A} 0.5725354  0.7217918 0.7932140 1.00489
## [6]  {Number.of.Credit.Problems=[0,7],
##       Purpose=debt consolidation}       => {Credit.Default=A} 0.5725354  0.7217918 0.7932140 1.00489
## [7]  {Bankruptcies=A,
##       Purpose=debt consolidation}       => {Credit.Default=A} 0.5080150  0.7212213 0.7043815 1.00410
## [8]  {Number.of.Credit.Problems=[0,7],
##       Bankruptcies=A,
##       Purpose=debt consolidation}       => {Credit.Default=A} 0.5080150  0.7212213 0.7043815 1.00410
## [9]  {}                                 => {Credit.Default=A} 0.7182741  0.7182741 1.0000000 1.00000
## [10] {Number.of.Credit.Problems=[0,7]}  => {Credit.Default=A} 0.7182741  0.7182741 1.0000000 1.00000
## [11] {Bankruptcies=A}                   => {Credit.Default=A} 0.6387924  0.7180180 0.8896607 0.99964
## [12] {Number.of.Credit.Problems=[0,7],
##       Bankruptcies=A}                   => {Credit.Default=A} 0.6387924  0.7180180 0.8896607 0.99964
```

```
##
##    0    1    2    3    4
## 6674  786   31    7    2
```

So, Imputation is good enough. Next, we will see it for Years in current job. In general, here also the intution may suggest that the Missing rows in Years in current job implies the customers are not an earning person. So, Years in current job is not an issue for him/her. So, impute these missing rows with <1 year is an idea. But, before doing it, we must see the distributions of missing values according to the Credit Default and Not Default.

```
## Credit.Default
##   0   1
## 234 137
```

Here, also the ratio of defaulters and non-defaulters is nearly 2:3. So, Impute the missing values using <1 year is not a good idea here. So, we use here knn algorithm which use Gower Distance for missing value imputation.

GOWER DISTANCE

One of the most important task while clustering the data is to decide what metric to be used for calculating distance between each data point. In various real-life fields where cluster analysis is commonly used, such as biology, social sciences, or marketing surveys, datasets with both quantitative and categorical variables are often applied. This type of data is referred as mixed data. Many distance metrics exist, and one of them is, the Gower distance (1971) which is used when the data is of Mixed data.

What is Gower's Distance?

Gower's Distance can be used to measure how different two records are. The records may contain combination of logical, categorical, numerical or text data. The distance is always a number between 0 (identical) and 1 (maximally dissimilar). The metrics used for each data type are described below:

quantitative (interval): range-normalized Manhattan distance

ordinal variable is first ranked, then Manhattan distance is used with a special adjustment for ties.

For nominal variables of k categories are first converted into k binary columns and then the Dice coefficient is used.

```
## Loading required package: colorspace

## Loading required package: grid

## VIM is ready to use.

## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues

##
## Attaching package: 'VIM'

## The following object is masked from 'package:missForest':
##
##     nrmse

## The following object is masked from 'package:datasets':
##
##     sleep

##
##  < 1 year    1 year 10+ years   2 years   3 years   4 years   5 years   6 years
##       563       504      2332       705       620       469       516       426
##   7 years   8 years   9 years
##       396       339       259

##
##  < 1 year    1 year 10+ years   2 years   3 years   4 years   5 years   6 years
##       563       504      2469       705       620       469       516       426
##   7 years   8 years   9 years
##       396       573       259
```

So, Imputation is good enough. Then, we will see it for Months Since Last Delinquent. In general, here also the intution may suggest that the Missing rows in Months Since Last Delinquent implies the customers are not an earning person. So, Months Since Last Delinquent is not an issue for him/her. So, impute these missing rows with a very high value, say, 130 is an idea. But, before doing it, we must see the distributions of missing values according to the Credit Default and Not Default.

```
## Credit.Default
##    0    1
## 2951 1130
```

Here, also the ratio of defaulters and non-defaulters is nearly 2:5. So, Impute the missing values using 130 is not a good idea here. So, also we use here knn algorithm which use Gower Distance for missing value imputation.

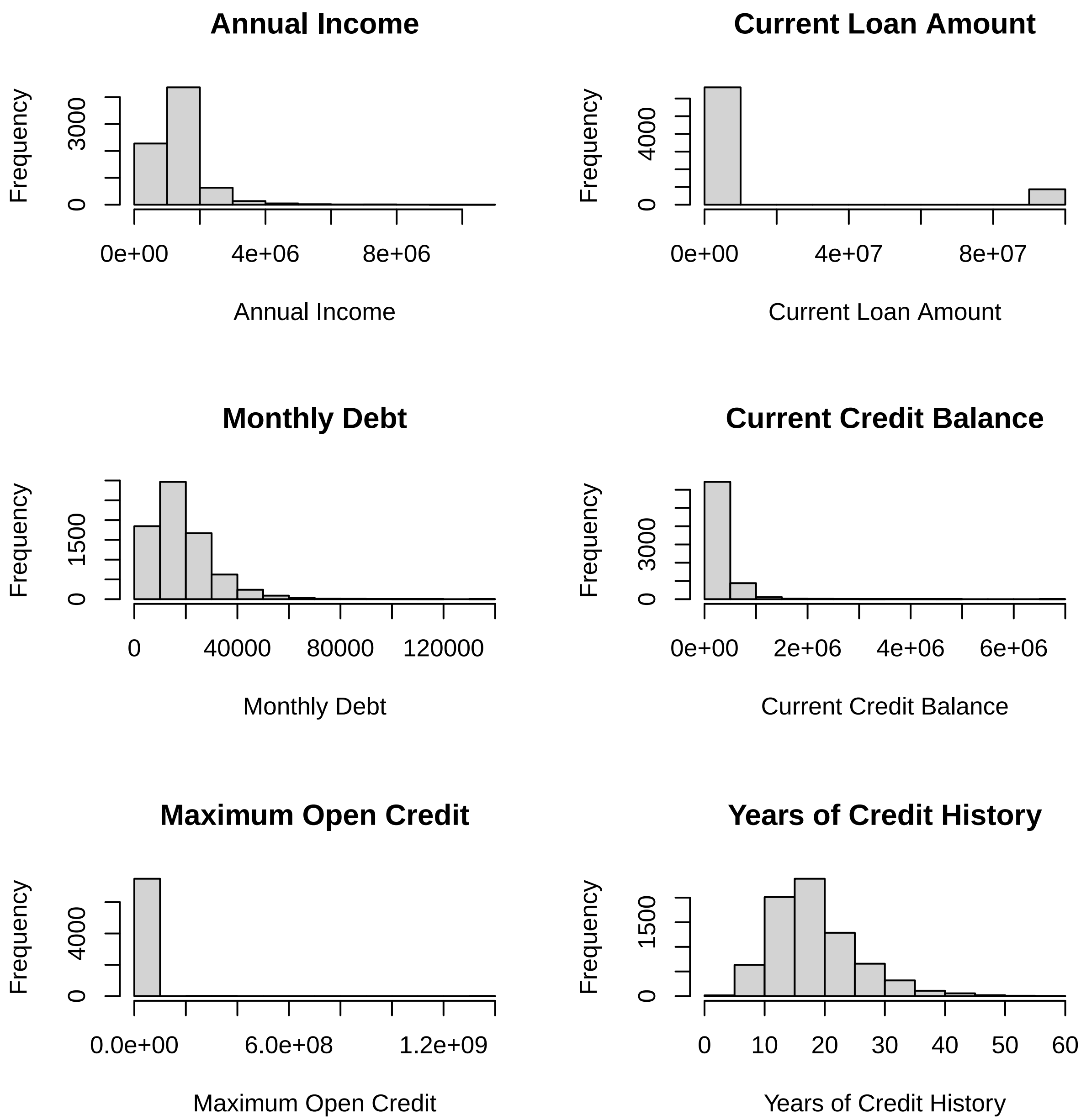
```
##
##   0   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19
##  18  26  25  30  31  51  64  64  68  61  63  51  65  65  76  48  61  58  58  65
##  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39
##  54  47  52  42  59  54  56  46  45  71  53  51  51  68  55  59  46  51  63  49
##  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59
##  48  50  43  45  36  50  46  37  44  25  39  19  26  34  36  36  23  29  24  32
##  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79
##  32  36  23  33  26  28  17  22  36  26  22  30  24  21  25  24  23  21  29  20
##  80  81  82  83  84  86  91  92 118
##  28  19   4   3   1   1   1   1   1
```

```
##
##   0   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19
##  18  26  25  30  32  51  66  80  75  77  82  74  96 134 131  91 134 128 123 155
##  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39
## 141  96 132 119 148 164 159 122 121 225 673 152 125 274 150 227 153 161 185 144
##  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59
## 132 124  95 102  98 134  88  75 109  39  65  35  50  62  65  52  32  55  42  47
##  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79
##  50  44  32  46  37  37  33  32  48  26  22  37  24  30  25  24  23  21  30  20
##  80  81  82  83  84  86  91  92 118
##  28  19   4   3   1   1   1   1   1
```

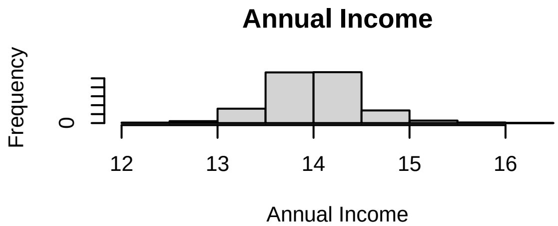
```
##               Home.Ownership      Number.of.Open.Accounts
##                            0                            0
##       Years.of.Credit.History         Maximum.Open.Credit
##                            0                            0
##    Number.of.Credit.Problems                 Bankruptcies
##                            0                            0
##                      Purpose                         Term
##                            0                            0
##          Current.Loan.Amount       Current.Credit.Balance
##                            0                            0
##                 Monthly.Debt               Credit.Default
##                            0                            0
##                Annual_Income         Years.in.current.job
##                            0                            0
## Months.since.last.delinquent
##                            0
```

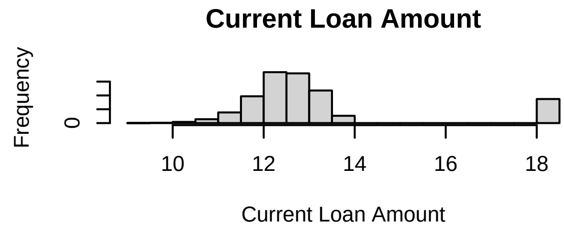
So, we complete our data missing values imputation. Now our data is free from Missing values and go for further analysis.

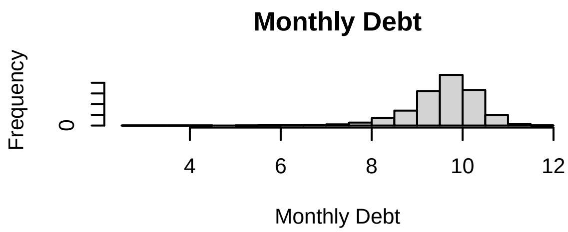
NOW WE CHECK GRAPHICAL PREVIEW OF CONTINUOUS COLUMN & CHECK THEIR CHARACTERISTICS

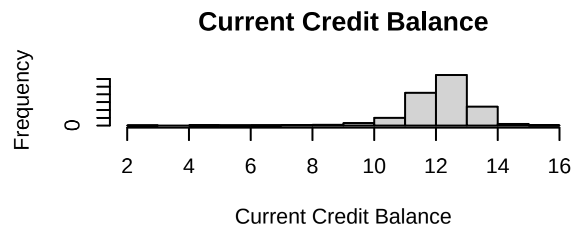


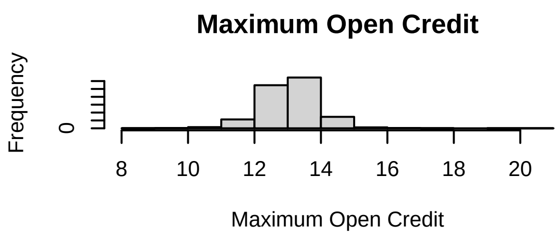
All of these continuous columns are right skewed except the YEARS OF CREDIT HISTORY. So we use log-transformation except this YEARS OF CREDIT HISTORY column to make their distributions nearly bell-shaped.











SPLITTING THE INTO TRAIN AND TEST

We split the dataset into (4:1) ratio for train and test.

```
## [1] 6000   15
```

```
## [1] 1500   15
```

VARIABLE SELECTIONS

An Important Technical Aspect of Developing Logistic Regression: Variable Selection

Variable selection aims at reducing the number of variables in a model. It will make the model more concise and faster to evaluate. Logistic regression has a built-in procedure to perform variable selection. It is based on a statistical hypothesis test to verify whether the coefficient of a variable included in the model is significantly different from zero.

In credit scoring, it is very important to be aware that statistical significance is only one evaluation criterion to consider in doing variable selection. As mentioned before, interpretability is also an important criterion (Martens et al. 2007). In logistic regression, this can be easily evaluated by inspecting the sign of the regression coefficient. It is highly preferable that a coefficient has the same sign as anticipated by the credit expert; otherwise he or she will be reluctant to use the model. Coefficients can have unexpected signs due to multicollinearity issues, noise, or small sample effects. Sign restrictions can be easily enforced in a forward regression setup by preventing variables with the wrong sign from entering the model.

Legal issues also need to be properly taken into account. For example, in the United States, there is the Equal Credit Opportunity Act, which states that no one is allowed to dis- criminate based on gender, age, ethnic origin, nationality, beliefs, and so on. These variables must not be included in a credit scorecard. Other countries have other regulations, and it is important to be aware of this.

NOW CHECKING CORRELATION AMONG NOMINAL CATEGORICAL VARIABLES

```
## Cramer V
##   0.1072
```

```
## Cramer V
##    0.356
```

```
## Cramer V
##   0.1276
```

```
## Cramer V
##  0.06648
```

```
## Cramer V
##  0.09327
```

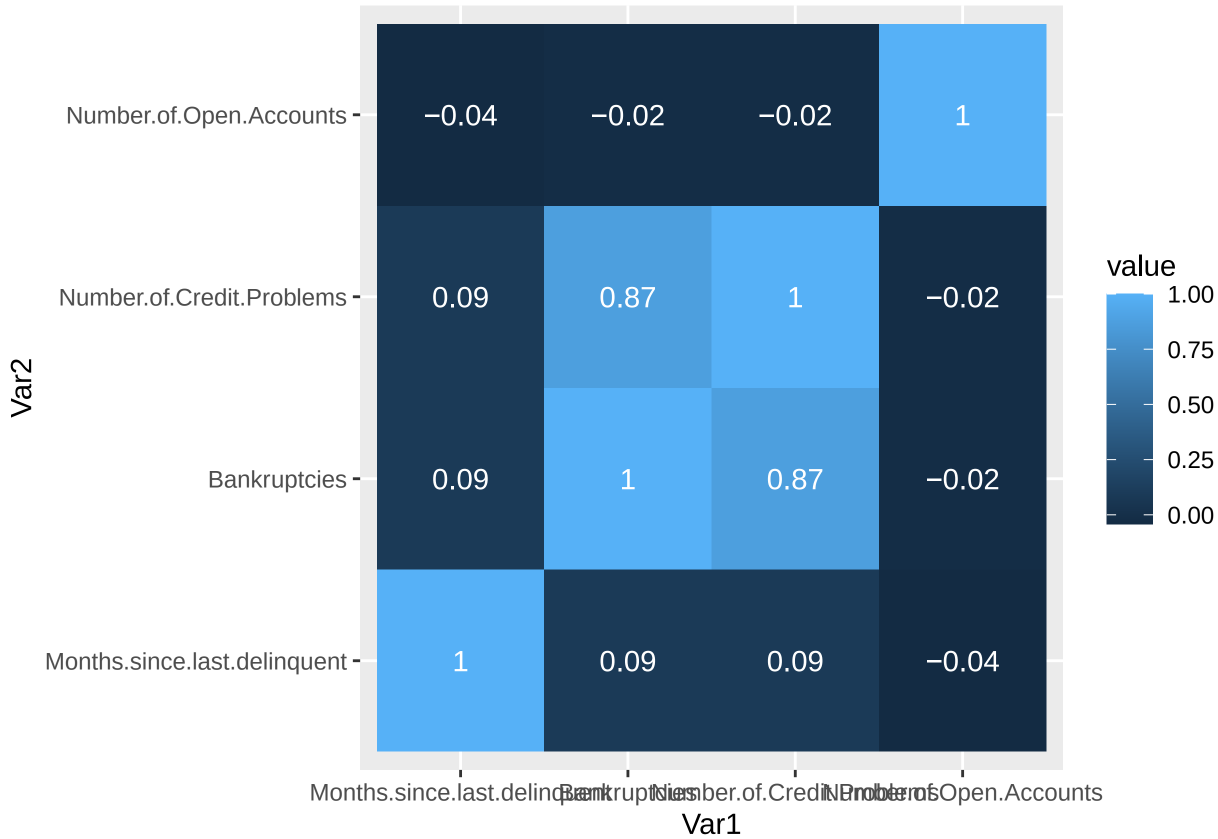
```
## Cramer V
##  0.05905
```

The correlation are not too high. So, we retain all Nominal Categorical columns.

NOW CHECKING CORRELATION AMONG ORNINAL CATEGORICAL VARIABLES

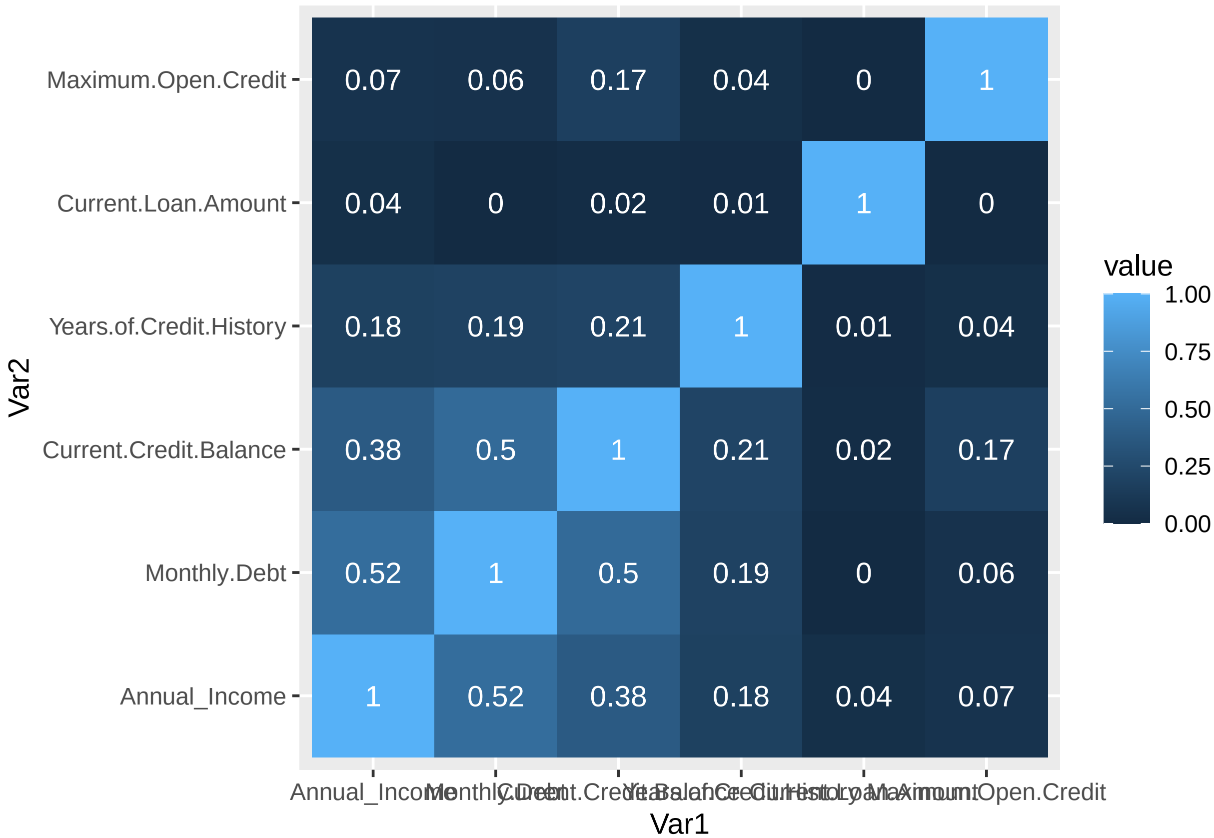
```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```



Here, We can see the correlation between Number of Credit Problem and Bankruptcies is significantly high. So, We drop one column among them. We retain Bankruptcies but drop Number of Credit Problems, as we think Bankruptcies is much significant in Credit default problem than Number of Credit Problems.

NOW CHECKING CORRELATION AMONG CONTINUOUS VARIABLES



From the correlation matrix, the correlations are not much high to infer that any 2 column have correlation among them. For better understand about the correlation, we use VARIANCE INFLATION FACTOR.

VARIANCE INFLATION FACTOR(VIF)

Now we will investigate whether the continuous regressors in our dataset are involved in multicollinearity or not. In a regression problem with multiple regressors , multicollinearity refers to a near-linear relationship among the regressors. Multicollinearity may happen due to overspecification of model, bad data collection or sampling techniques, inclusion of too many higher order terms in a polynomial regression model etc. Multicollinearity has some serious consequences eg. exceptionally high value of parameter estimates, large variances of some parameter estimators. Several multicollinearity diagnostic measures are available. Here we have used "Variance Inflation Factor" to detect multicollinearity among the continuous variables of our dataset. The variance inflation factor for the jth explanatory variable (when all the regressors are scaled to unit norm) is defined as:

$VIF_j = \frac{1}{1 - R_j^2}$

where

R_j^2

denotes the coefficient of determination obtained when Xj is regressed on the remaining regressor variables.

In practice, usually a VIF > 5 indicates that the corresponding explanatory variable is involved in multicollinearity. Here we will use an iterative algorithm that drops variable with highest VIF and then checks V

IF again and then drop until VIF of all variables is less than 5.

```
##
## Attaching package: 'scorecard'

## The following object is masked from 'package:tidyr':
##
##     replace_na

## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:scorecard':
##
##     vif

## The following object is masked from 'package:arules':
##
##     recode

## The following object is masked from 'package:dplyr':
##
##     recode

## The following object is masked from 'package:purrr':
##
##     some

##           Annual_Income            Monthly.Debt  Current.Credit.Balance
##                1.539606                1.806577                3.274368
## Years.of.Credit.History     Current.Loan.Amount     Maximum.Open.Credit
##                1.081478                1.002424                2.936635
```

So, VIFs are less than 5. So, we infer that the continuous columns are not mutually correlated.

Logistic Regression for Developing a Scorecard Model

Logistic regression is a very popular credit scoring classification technique due to its simplicity and good performance. Just as with linear regression, once the parameters have been estimated, the regression can be evaluated in a straightforward way, contributing to its operational efficiency. From an interpretability viewpoint, it can be easily transformed into an interpretable, user-friendly, points-based credit scorecard.

Next we use STEPWISE SELECTION.

```
## [INFO] creating woe binning ...
## [INFO] Binning on 6000 rows and 14 columns in 00:00:14

## [INFO] converting into woe values ...

## [INFO] converting into woe values ...

## Start:  AIC=6332.1
## train$Credit.Default ~ Home.Ownership_woe + Number.of.Open.Accounts_woe +
##     Years.of.Credit.History_woe + Maximum.Open.Credit_woe + Bankruptcies_woe +
##     Purpose_woe + Term_woe + Current.Loan.Amount_woe + Current.Credit.Balance_woe +
##     Monthly.Debt_woe + Annual_Income_woe + Years.in.current.job_woe +
##     Months.since.last.delinquent_woe
##
##                                    Df Deviance    AIC
```

```
## - Bankruptcies_woe                 1   6304.8 6330.8
## <none>                               6304.1 6332.1
## - Current.Credit.Balance_woe       1   6307.3 6333.3
## - Years.of.Credit.History_woe      1   6312.1 6338.1
## - Home.Ownership_woe               1   6314.6 6340.6
## - Purpose_woe                      1   6315.6 6341.6
## - Maximum.Open.Credit_woe          1   6324.7 6350.7
## - Number.of.Open.Accounts_woe      1   6331.3 6357.3
## - Monthly.Debt_woe                 1   6336.7 6362.7
## - Months.since.last.delinquent_woe  1   6342.0 6368.0
## - Years.in.current.job_woe         1   6364.8 6390.8
## - Current.Loan.Amount_woe          1   6438.2 6464.2
## - Annual_Income_woe                1   6480.2 6506.2
## - Term_woe                         1   6522.5 6548.5
##
## Step:  AIC=6330.82
## train$Credit.Default ~ Home.Ownership_woe + Number.of.Open.Accounts_woe +
##     Years.of.Credit.History_woe + Maximum.Open.Credit_woe + Purpose_woe +
##     Term_woe + Current.Loan.Amount_woe + Current.Credit.Balance_woe +
##     Monthly.Debt_woe + Annual_Income_woe + Years.in.current.job_woe +
##     Months.since.last.delinquent_woe
##
##                                   Df Deviance    AIC
## <none>                               6304.8 6330.8
## + Bankruptcies_woe                 1   6304.1 6332.1
## - Current.Credit.Balance_woe       1   6308.3 6332.3
## - Years.of.Credit.History_woe      1   6313.3 6337.3
## - Home.Ownership_woe               1   6315.5 6339.5
## - Purpose_woe                      1   6316.5 6340.5
## - Maximum.Open.Credit_woe          1   6324.7 6348.7
## - Number.of.Open.Accounts_woe      1   6331.9 6355.9
## - Monthly.Debt_woe                 1   6337.3 6361.3
## - Months.since.last.delinquent_woe  1   6342.4 6366.4
## - Years.in.current.job_woe         1   6365.6 6389.6
## - Current.Loan.Amount_woe          1   6439.2 6463.2
## - Annual_Income_woe                1   6480.4 6504.4
## - Term_woe                         1   6523.9 6547.9
```

```
##
## Call:  glm(formula = train$Credit.Default ~ Home.Ownership_woe + Number.of.Open.Accounts_woe +
##     Years.of.Credit.History_woe + Maximum.Open.Credit_woe + Purpose_woe +
##     Term_woe + Current.Loan.Amount_woe + Current.Credit.Balance_woe +
##     Monthly.Debt_woe + Annual_Income_woe + Years.in.current.job_woe +
##     Months.since.last.delinquent_woe, family = binomial, data = train_woe)
##
## Coefficients:
##                      (Intercept)                Home.Ownership_woe
##                          -0.9280                            0.7692
##      Number.of.Open.Accounts_woe       Years.of.Credit.History_woe
##                           1.3654                            0.8409
##          Maximum.Open.Credit_woe                       Purpose_woe
##                           0.8948                            1.1541
##                         Term_woe           Current.Loan.Amount_woe
##                           1.1340                            0.8191
```

```
##       Current.Credit.Balance_woe                    Monthly.Debt_woe
##                           0.7389                              1.2721
##                Annual_Income_woe          Years.in.current.job_woe
##                           1.1561                              1.0377
## Months.since.last.delinquent_woe
##                           0.6114
##
## Degrees of Freedom: 5999 Total (i.e. Null);  5987 Residual
## Null Deviance:       7170
## Residual Deviance: 6305  AIC: 6331
```

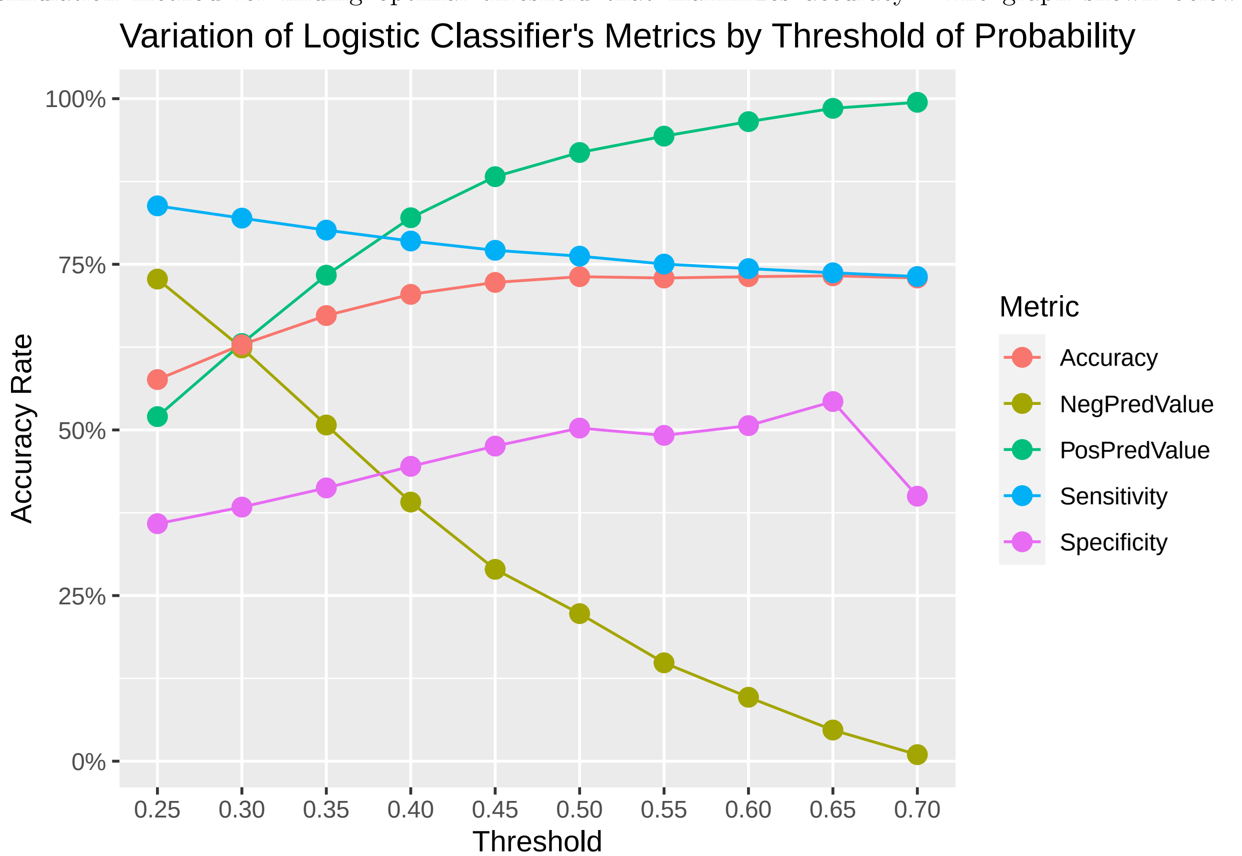
Now, we check the the selected model is nearly the saturated model or not, using deviance statistics.

```
## [1] TRUE
```

So, The Selected model is good.

CALCULATE PD (SEARCH OPTIMAL THRESOLD OF PROBABILITY THAT MAXIMIZE ACCURACY

Accuracy criterion is widely used for evaluating model performance in context of credit scoring. However this measure is totally affected by threshold of probability that we select for classifier. I use simulation method for finding optimal threshold that maximizes accuracy.: The graph shown below:



These results reveals that optimal threshold of probability maximizing Accuracy is 0.5. However note that Accuracy should not be the most vital goal for for-profit businesses.

```
## Loading required package: lattice
```

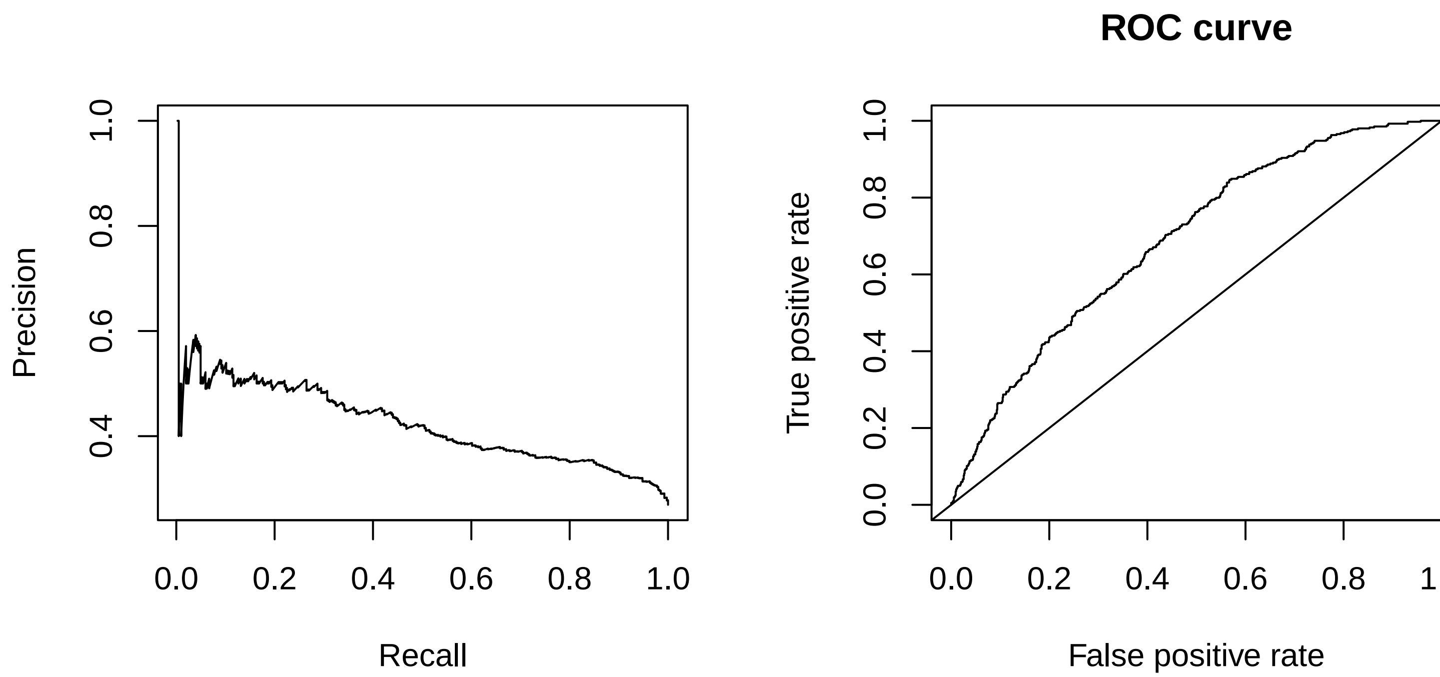
```
##
## Attaching package: 'caret'
## The following object is masked from 'package:purrr':
##
##     lift
```

Now, the we construct the confusion matrix,

```
## Confusion Matrix and Statistics
##
##
## fraud_or_not_STEP    0    1
##                 0 1009  315
##                 1   87   89
##
##                Accuracy : 0.732
##                  95% CI : (0.7088, 0.7543)
##     No Information Rate : 0.7307
##     P-Value [Acc > NIR] : 0.467
##
##                   Kappa : 0.1715
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.9206
##             Specificity : 0.2203
##          Pos Pred Value : 0.7621
##          Neg Pred Value : 0.5057
##              Prevalence : 0.7307
##          Detection Rate : 0.6727
##    Detection Prevalence : 0.8827
##       Balanced Accuracy : 0.5705
##
##        'Positive' Class : 0
##
```

The Accuracy of the model is 85.47%. Now we check the ROC and PRECISION-RECALL curve.

```
## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
## The following object is masked from 'package:colorspace':
##
##     coords
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```



The AUC is

```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

## Area under the curve: 0.6895
```

Weight of Evidence (WOE) and Information Value (IV)

Prior to building a binary classification model (e.g., Logistic Regression, etc.), a common step is to perform variable screening and exploratory data analysis. This is the step where we get to know the data and weed out variables that are either ill-conditioned or simply contain no information that will help us predict the action of interest. Note that the purpose of this step should not to be confused with that of multiple-variable selection techniques, such as stepwise regression, where the variables that go into the final model are selected. Rather, this is a precursory step designed to ensure that the approaches deployed during the final modeling phases are set up for success.

The weight of evidence (WOE) and information value (IV) provide a great framework for for exploratory analysis and variable screening for binary classifiers. WOE and IV have been used extensively in the credit risk world for several decades, and the underlying theory dates back to the 1950s.

WOE and IV are simple, yet powerful techniques to perform variable transformation and selection. These concepts have huge connection with the logistic regression modeling technique. It is widely used in credit scoring to measure the separation of good vs bad customers. In addition, the advantages of WOE transformation are:

Handles missing values.

Handles outliers.

The transformation is based on logarithmic value of distributions. This is aligned with the logistic regression output function No need for dummy variables.

By using proper binning technique, it can establish monotonic relationship (either increase or decrease) between the independent and dependent variable

According to Baesens et al. (2016) and Siddiqi (2012), WOE and IV analysis enable one to:

Consider each variable's independent contribution to the outcome.

Detect linear and non-linear relationships.

Rank variables in terms of "univariate" predictive strength.

Visualize the correlations between the predictive variables and the binary outcome.

Seamlessly compare the strength of continuous and categorical variables without creating dummy variables.

Seamlessly handle missing values without imputation.

Assess the predictive power of missing values.

By convention the values of the IV statistic for variable selec tion can be used as follows:

Less than 0.02: the predictor is not useful for modeling (separating the Goods from the Bads).

From 0.02 to 0.1: the predictor has only a weak relationship to the Goods/Bads odds ratio.

From 0.1 to 0.3: the predictor has a medium strength relationship to the Goods/Bads odds ratio.

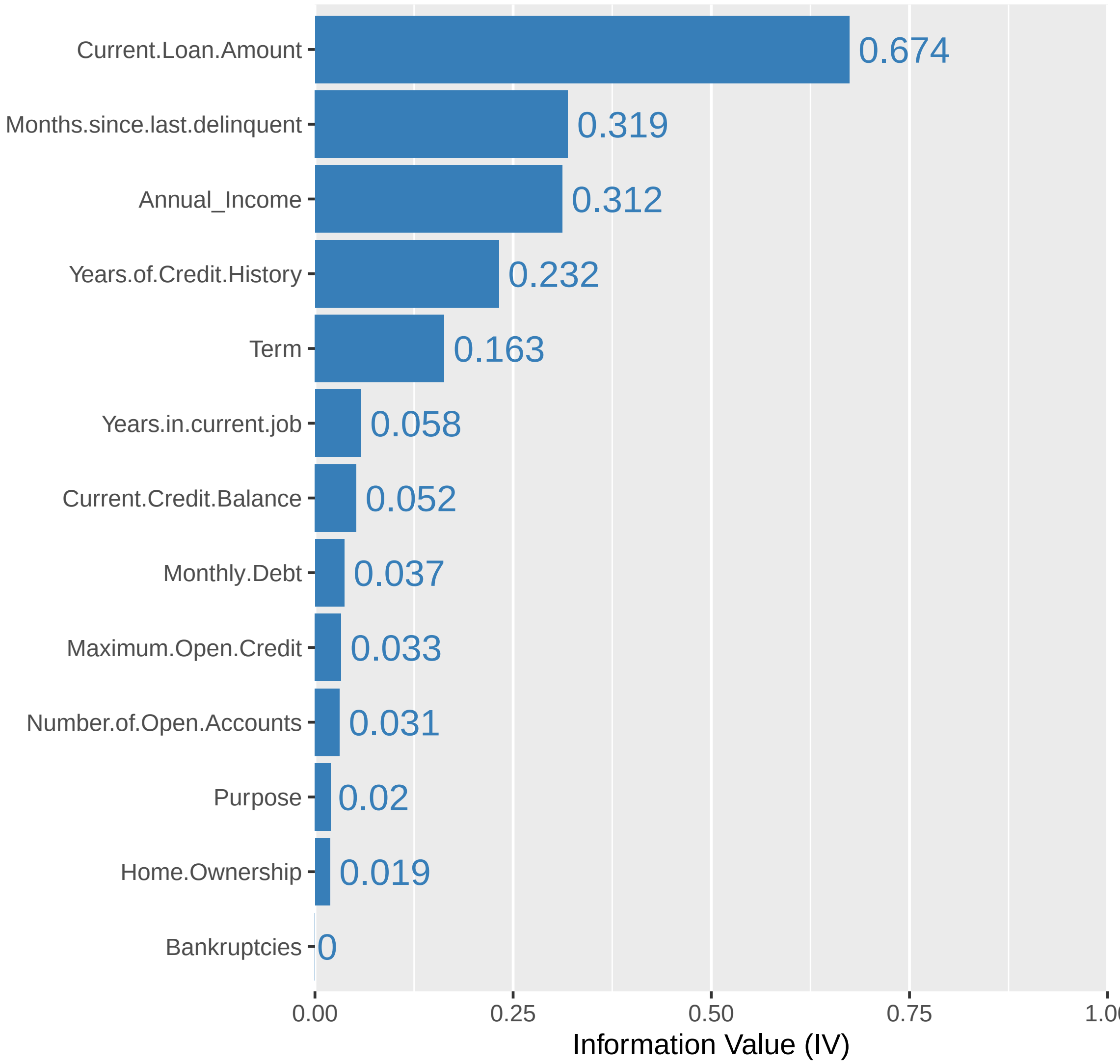
From 0.3 to 0.5: the predictor has a strong relationship to the Goods/Bads odds ratio.

Higher than 0.5: we should check carefully when selecting variables have IV higher than 0.5 because of suspicious relationship.

```
##                        variable  info_value
##  1:         Current.Loan.Amount 0.673518352
##  2: Months.since.last.delinquent 0.319068807
##  3:               Annual_Income 0.311970744
##  4:      Years.of.Credit.History 0.231821130
##  5:                        Term 0.162609983
##  6:         Years.in.current.job 0.058468533
##  7:       Current.Credit.Balance 0.051953409
##  8:                Monthly.Debt 0.037331093
##  9:         Maximum.Open.Credit 0.033350544
## 10:      Number.of.Open.Accounts 0.031120795
## 11:                     Purpose 0.019901237
## 12:              Home.Ownership 0.018828744
## 13:                Bankruptcies 0.000480281
```

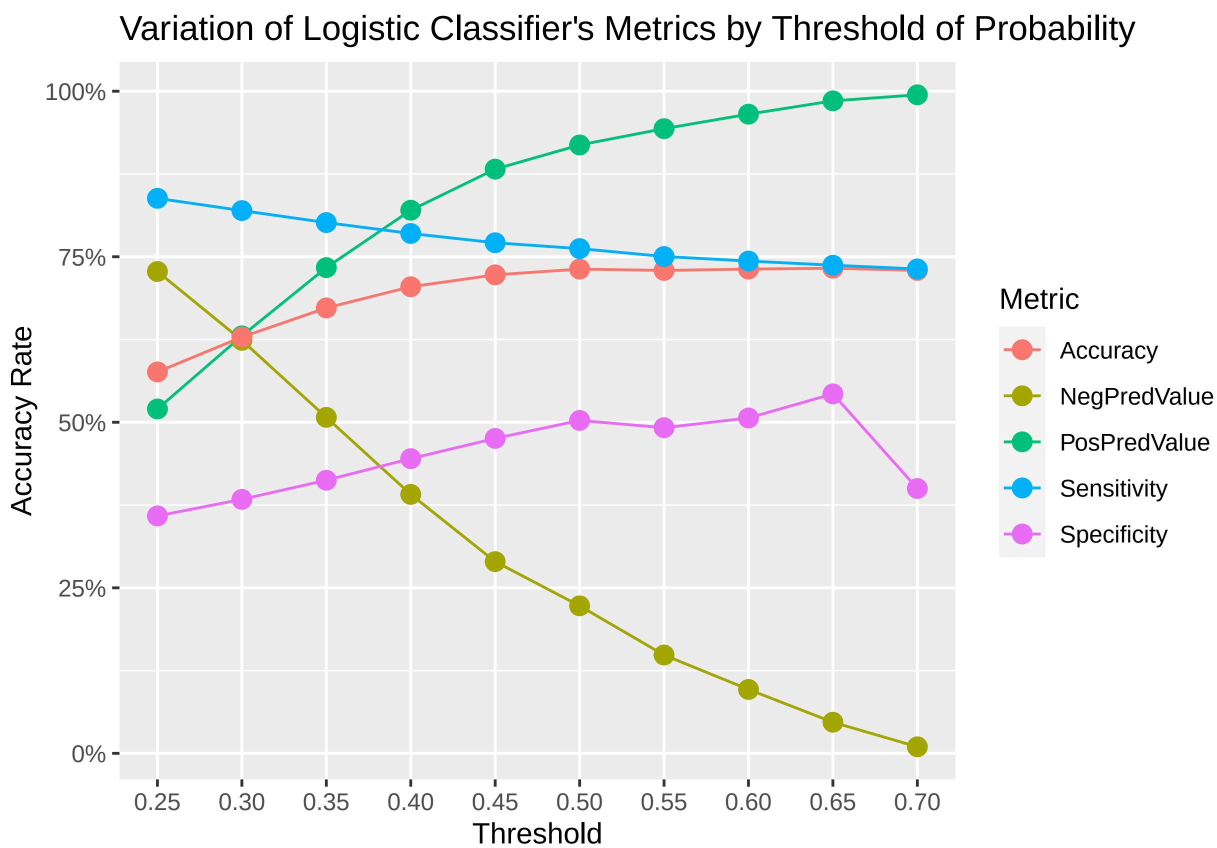
The Graphical view of IV values

Figure 7: Information Value (IV) for All Variables



CALCULATE PD (SEARCH OPTIMAL THRESOLD OF PROBABILITY THAT MAXIMIZE ACCURACY

Accuracy criterion is widely used for evaluating model performance in context of credit scoring. However this measure is totally affected by threshold of probability that we select for classifier. I use simulation method for finding optimal threshold that maximizes accuracy. The graph shown below:



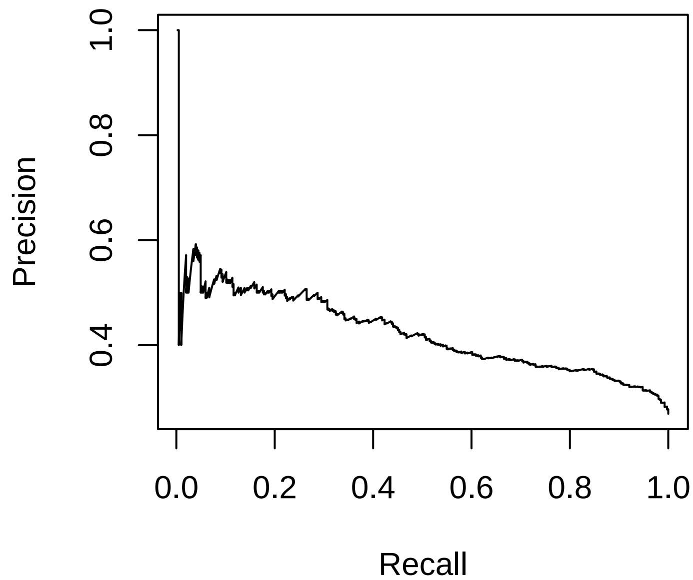
These results reveals that optimal threshold of probability maximizing Accuracy is 0.5. However note that Accuracy should not be the most vital goal for for-profit businesses.

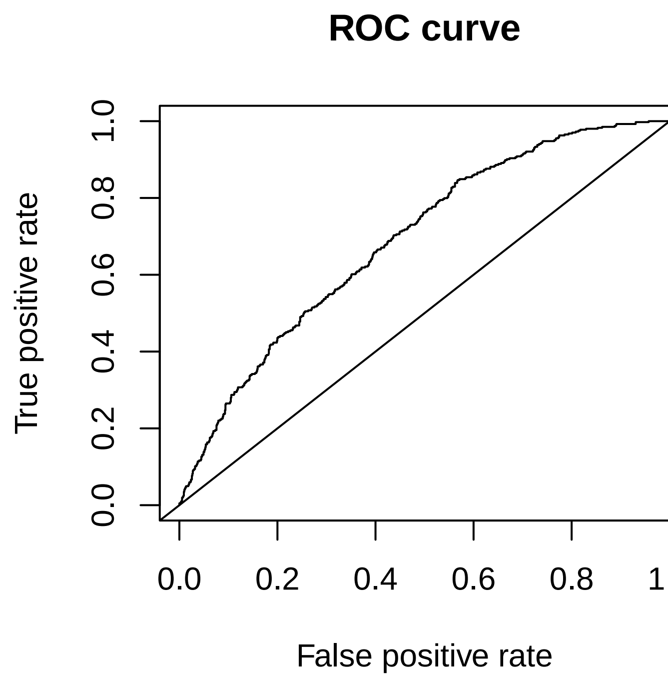
Now, the we construct the confusion matrix,

```
## Confusion Matrix and Statistics
##
##
## fraud_or_not_IV    0    1
##               0 1009  315
##               1   87   89
##
##                Accuracy : 0.732
##                  95% CI : (0.7088, 0.7543)
##     No Information Rate : 0.7307
##     P-Value [Acc > NIR] : 0.467
##
##                   Kappa : 0.1715
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.9206
##             Specificity : 0.2203
##          Pos Pred Value : 0.7621
##          Neg Pred Value : 0.5057
##              Prevalence : 0.7307
##          Detection Rate : 0.6727
##    Detection Prevalence : 0.8827
```

```
##       Balanced Accuracy : 0.5705
##
##        'Positive' Class : 0
##
```

The Accuracy of the model is 85.47%. Now we check the ROC and PRECISION-RECALL curve.





The AUC is

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
## Area under the curve: 0.6895
```

All 2 models has similar accuracy. But, we select model Selected using IV as IV plays a important role in Credit Default Prediction.

So, finally we select the model chossen using IV.

So, Total Probability of Misclassification of the model is

```
##     0
## 0.268
```

Specificity

```
## [1] 22.0297
```

Sensitivity

```
## [1] 92.06204
```

False Negative

```
## [1] 7.937956
```

False Positive

```
## [1] 77.9703
```

Precision

```
## [1] 0.5414384
```

Recall

```
## [1] 0.9206204
```

F1_Score

```
## [1] 0.3409297
```

Phi_coefficient

```
##
## Attaching package: 'psych'

## The following object is masked from 'package:car':
##
##     logit

## The following object is masked from 'package:scorecard':
##
##     describe

## The following object is masked from 'package:rcompanion':
##
##     phi

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha

## [1] 0.19
```

Now, we check the the selected model is nearly the saturated model or not, using deviance statistics.

```
## [1] FALSE
```

So, the model is good. Goodness of fit

```
## X-squared
##      TRUE
```

Contingency coefficient

```
## X-squared
## 0.1884754
```

Probabilities of Default by Group for Test Data

Table 1: Probabilities of Default by Group for Test Data

Credit.Default	min	max	median	mean	n
Default	0.0356	0.7316	0.3580	0.364	404
NonDefault	0.0078	0.7260	0.2372	0.255	1096

Some Criteria for Model Evaluation in Context of Credit Scoring

It is impossible to use a scoring model effectively without knowing how accurate it is. First, one needs to select the best model according to some criteria for evaluating model performance. The methodology of credit scoring models and some measures of their quality have been discussed in surveys conducted by Hand and Henley (1997), Thomas (2000), and Crook at al. (2007). However, until just ten years ago, the general literature devoted to the issue of credit scoring was not substantial. Fortunately, the situation has improved in the last decade with the publication of works by Anderson (2007), Crook et al. (2007), Siddiqi (2006), Thomas et al. (2002), and Thomas (2009), all of which address the topic of credit scoring. The most used criteria in context of credit scoring are:

Gain or lift is a measure of the effectiveness of a classification model calculated as the ratio between the results obtained with and without the model. Gain and lift charts are visual aids for evaluating performance of classification models. However, in contrast to the confusion matrix that evaluates models on the whole population gain or lift chart evaluates model performance in a portion of the population.

Scorecards based on our model provide scores. A score is a measure that allows lenders to rank customers from high risk (low score) to low risk (high score) and as such provides a relative measure of credit risk. Scores are unlimited and can be measured within any range; they can even be negative. A score is not the same as a probability. A probability also allows us to rank, but on top of that, since it is limited between 0 and 1, it also gives an absolute interpretation of credit risk. Hence, probabilities provide more information than scores do. For application scoring, one does not need well-calibrated probabilities of default. However, for other application areas such as regulatory capital calculation in a Basel setting, as we will discuss later, calibrated default probabilities are needed (Van Gestel and Baesens 2009).

```
## [INFO] creating woe binning ...
## [INFO] Binning on 6000 rows and 13 columns in 00:00:13

## [INFO] creating woe binning ...

## [INFO] converting into woe values ...
## [INFO] Woe transformating on 6000 rows and 12 columns in 00:00:11

## [INFO] converting into woe values ...
```

Predictor	Group	Scorecard
basepoints	NA	455
Home.Ownership	Have Mortgage% to %Home Mortgage	8
Home.Ownership	Own Home	-7
Home.Ownership	Rent	-7
Months.since.last.delinquent	From -Inf to 24	1
Months.since.last.delinquent	From 24 to 30	-13
Months.since.last.delinquent	From 30 to 32	51
Months.since.last.delinquent	From 32 to 36	3
Months.since.last.delinquent	From 36 to Inf	-8
Years.in.current.job	8 years	62
Years.in.current.job	3 years% to %2 years% to %4 years% to %9 years	5
Years.in.current.job	5 years% to %6 years% to %7 years% to %< 1 year% to %1 year	-2
Years.in.current.job	10+ years	-14
Annual_Income	From -Inf to 13.9	-29
Annual_Income	From 13.9 to 14	-8
Annual_Income	From 14 to 14.2	49
Annual_Income	From 14.2 to 14.4	-5
Annual_Income	From 14.4 to Inf	40
Monthly.Debt	From -Inf to 8.4	41
Monthly.Debt	From 8.4 to 9	-5

Predictor	Group	Scorecard
Monthly.Debt	From 9 to 9.3	15
Monthly.Debt	From 9.3 to 9.5	-17
Monthly.Debt	From 9.5 to 10.3	-1
Monthly.Debt	From 10.3 to Inf	-9
Current.Credit.Balance	From -Inf to 10.8	4
Current.Credit.Balance	From 10.8 to 11.6	-5
Current.Credit.Balance	From 11.6 to 12	8
Current.Credit.Balance	From 12 to 13	-2
Current.Credit.Balance	From 13 to Inf	1
Current.Loan.Amount	From -Inf to 13	-6
Current.Loan.Amount	From 13 to 13.5	-27
Current.Loan.Amount	From 13.5 to Inf	84
Term	Short Term	21
Term	Long Term	-52
Purpose	educational expenses% to %vacation% to %moving% to %major purchase% to %buy a car% to %home improvements	11
Purpose	debt consolidation	2
Purpose	other% to %wedding% to %take a trip% to %medical bills% to %buy house% to %business loan% to %small business% to %renewable energy	-18
Maximum.Open.Credit	From -Inf to 12.2	-14
Maximum.Open.Credit	From 12.2 to 12.8	1
Maximum.Open.Credit	From 12.8 to 13.4	-7
Maximum.Open.Credit	From 13.4 to 13.8	6
Maximum.Open.Credit	From 13.8 to Inf	20
Years.of.Credit.History	From -Inf to 11	-10
Years.of.Credit.History	From 11 to 20	-1
Years.of.Credit.History	From 20 to 24	9
Years.of.Credit.History	From 24 to 28	14
Years.of.Credit.History	From 28 to Inf	-5
Number.of.Open.Accounts	From -Inf to 6	2
Number.of.Open.Accounts	From 6 to 9	14
Number.of.Open.Accounts	From 9 to 11	-5
Number.of.Open.Accounts	From 11 to 12	23
Number.of.Open.Accounts	From 12 to Inf	-11

SCORECARD POINTS BY GROUP FOR TEST DATA (SELECTION BASED ON IV)

Table 3: Scorecad Points by Group for Test Data

Credit.Default	min	max	median	mean	n
Default	315	625	430	433	404
NonDefault	317	738	472	482	1096

KEY CHARACTERISTICS OF A USEFUL SCORECARD MODEL

Before bringing a scorecard into production, it needs to be thoroughly evaluated. Depending on the exact setting and usage of the model, different aspects may need to be assessed during evaluation in order to ensure the model is acceptable for implementation. Key characteristics of successful scorecard model are:

INTERPRETABILITY:

A scorecard needs to be interpretable. In other words, a deeper understanding of the detected default behavior is required, for instance to validate the scorecard before it can be used. This aspect involves a certain degree of subjectivism, since interpretability may depend on the credit expert's knowledge. The interpretability of a model depends on its format, which in turn is determined by the adopted analytical technique. Models that allow the user to understand the underlying reasons why the model signals a customer to be a defaulter are called white box models, whereas complex, incomprehensible, mathematical models are often referred to as black box models.

STATISTICAL ACCURACY

Refers to the detection power and the correctness of the scorecard in labeling customers as defaulters. Several statistical evaluation criteria exist and may be applied to evaluate this aspect, such as the hit rate, lift curves, area under the curve (AUC), and so on. Statistical accuracy may also refer to statistical significance, meaning that the patterns that have been found in the data have to be valid and not the consequence of noise. In other words, we need to make sure that the model generalizes well and is not overfitted to the historical data set.

ECONOMICAL COST

Developing and implementing a scorecard involves a significantcost to an organization. The total cost includes the costs togather, preprocess, and analyze the data, and the costs to putthe resulting scorecards into production. In addition, the softwarecosts as well as human and computing resources should betaken into account. Possibly also external (e.g., credit bureau)data has to be bought to enrich the available in-house data.Clearly it is important to perform a thorough cost-benefit analysisat the start of the credit scoring project, and to gain insight intothe constituent factors of the return on investment of building ascorecard system.

REGULATORY COMPLIANCES

A scorecard should be in line and compliant with all applicable regulations and legislation. In a credit scoring setting, the Basel Accords specify what information can or cannot be used and how the target (i.e., default) should be defined. Other regulations (e.g., with respect to privacy and/or discrimination) should also be respected.

SOME PRACTICAL ASPECTS OF SCORECARD MODEL USING BY BANK

The most important usage of application scores is to decide on loan approval. The scores can also be used for pricing purposes. Risk-based pricing (sometimes also referred to as risk-adjusted pricing) sets the price or other characteristics (e.g., loan term, collateral) of the loan based on the perceived risk as measured by the application score. A lower score will imply a higher interest rate and vice versa.

There are still many unresolved aspects of the credit rating (for example, selecting best model that maximizes profit or turning model parameter) but because my time resources is limited, these interesting issues will be presented in an upcoming post.

LIMITATIONS

Although credit scoring systems are being implemented and used by most banks nowadays, they do face a number of limitations. A first limitation concerns the data that is used to estimate credit scoring models.

Since data is the major, and in most cases the only, ingredient to build these models, its quality and predictive ability is key to the models' success.

The quality of the data refers, for example, to the num- ber of missing values and outliers, and to the recency and representativity of the data. Data quality issues can be difficult to detect without specific domain knowledge, but have an important impact on the scorecard development and resulting risk measures. The availability of high-quality data is a very important prerequisite for building good credit scoring models. However, not only does the data need to be of high quality, but it should be predictive as well, in the sense that the captured characteristics are related to the customer's likelihood of defaulting.

In addition, before constructing a scorecard model, we need to thoroughly reflect on why a customer defaults and which characteristics could potentially be related to this. Customers may default because of unknown reasons or information not available to the financial institution, thereby posing another limitation to the performance of credit scoring models. The statistical techniques used in developing credit scoring models typically assume a data set of sufficient size containing enough defaults. This may not always be the case for specific types of portfolios where only limited data is available, or only a low number of defaults is observed. For these types of portfolios, one may have to rely on alternative risk assessment methods using, for example, expert judgment based on the five Cs, as discussed earlier.

REFERENCES

Siddiqi, N. (2012). Credit risk scorecards: developing and implementing intelligent credit scoring. John Wiley & Sons.

Listen Data, Web blog