Credit Scoring Project

Mehdi Rahmouni

August 18, 2017

Contents

Introduction

The purpose of this code is to help the bank acuuratly predict if a loan applicant will default on his loan or not. For predfctive modeling decision trees and logistic regression models are built.The last part of the code tries to treat data imbalance using different techniques over, under sampling and smote. Reproducibility is important to allow other reproduce the code with ease by using set.seed() and sessioninfo()

This report was developed using R Markdown, which allows for the creation of reproducible documents, from PDF, HTML, to MS Word Documents.

Loading Libraries

```
## graphs and plots
require (ggplot2)
require (plotly)

##ROC calculation
require (pROC)
require (ROCR)

##data manipulation
require (magrittr) ##pipes, data manipulation
require (plyr)
require (dplyr)
require (e1071)
require (gmodels)
##decision trees
require (rpart)
```

```
require (rpart.plot)
require (RColorBrewer)
require (rattle)

##report generation
require (knitr)
require (markdown)
require (devtools)

##treat data imbalance
require (caret)
require (DMwR)
require (purrr)
```

Loading Dataset

```
##for reproductibility purposes
set.seed (100)
##to limit files "hard-coding", for reproductibility
input_file <- "C:/Users/Dell/Google Drive/DAT650/Project/credit.csv"
german_data <- read.csv (input_file)
```

Deleting the ID Variable

```
german_data$OBS. <- NULL
```

Checking Structure

```
str (german_data)
```

```
## 'data.frame':    1000 obs. of  31 variables:
##  $ CHK_ACCT        : int  0 1 3 0 0 3 3 1 3 1 ...
##  $ DURATION        : int  6 48 12 42 24 36 24 36 12 30 ...
##  $ HISTORY         : int  4 2 4 2 3 2 2 2 2 4 ...
##  $ NEW_CAR         : int  0 0 0 0 1 0 0 0 0 1 ...
##  $ USED_CAR        : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ FURNITURE       : int  0 0 0 1 0 0 1 0 0 0 ...
##  $ RADIO.TV        : int  1 1 0 0 0 0 0 0 1 0 ...
##  $ EDUCATION       : int  0 0 1 0 0 1 0 0 0 0 ...
##  $ RETRAINING      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ AMOUNT          : int  1169 5951 2096 7882 4870 9055 2835 6948 3059 5234 ...
##  $ SAV_ACCT        : int  4 0 0 0 0 4 2 0 3 0 ...
##  $ EMPLOYMENT      : int  4 2 3 3 2 2 4 2 3 0 ...
##  $ INSTALL_RATE    : int  4 2 2 2 3 2 3 2 2 4 ...
##  $ MALE_DIV        : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ MALE_SINGLE     : int  1 0 1 1 1 1 1 1 0 0 ...
##  $ MALE_MAR_or_WID : int  0 0 0 0 0 0 0 0 0 1 ...
##  $ CO.APPLICANT    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ GUARANTOR       : int  0 0 0 1 0 0 0 0 0 0 ...
```

```
##  $ PRESENT_RESIDENT: int  4 2 3 4 4 4 4 2 4 2 ...
##  $ REAL_ESTATE     : int  1 1 1 0 0 0 0 0 1 0 ...
##  $ PROP_UNKN_NONE  : int  0 0 0 0 1 1 0 0 0 0 ...
##  $ AGE             : int  67 22 49 45 53 35 53 35 61 28 ...
##  $ OTHER_INSTALL   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ RENT            : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ OWN_RES         : int  1 1 1 0 0 0 1 0 1 1 ...
##  $ NUM_CREDITS     : int  2 1 1 1 2 1 1 1 1 2 ...
##  $ JOB             : int  2 2 1 2 2 1 2 3 1 3 ...
##  $ NUM_DEPENDENTS  : int  1 1 2 2 2 2 1 1 1 1 ...
##  $ TELEPHONE       : int  1 0 0 0 0 1 0 1 0 0 ...
##  $ FOREIGN         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ DEFAULT         : int  0 1 0 0 1 0 0 0 0 1 ...
```

Head()

```
head (german_data, 3)
```

```
##   CHK_ACCT DURATION HISTORY NEW_CAR USED_CAR FURNITURE RADIO.TV EDUCATION
## 1        0        6       4       0        0         0        1         0
## 2        1       48       2       0        0         0        1         0
## 3        3       12       4       0        0         0        0         1
##   RETRAINING AMOUNT SAV_ACCT EMPLOYMENT INSTALL_RATE MALE_DIV MALE_SINGLE
## 1          0   1169        4          4            4        0           1
## 2          0   5951        0          2            2        0           0
## 3          0   2096        0          3            2        0           1
##   MALE_MAR_or_WID CO.APPLICANT GUARANTOR PRESENT_RESIDENT REAL_ESTATE
## 1               0            0         0                4           1
## 2               0            0         0                2           1
## 3               0            0         0                3           1
##   PROP_UNKN_NONE AGE OTHER_INSTALL RENT OWN_RES NUM_CREDITS JOB
## 1              0  67             0    0       1           2   2
## 2              0  22             0    0       1           1   2
## 3              0  49             0    0       1           1   1
##   NUM_DEPENDENTS TELEPHONE FOREIGN DEFAULT
## 1              1         1       0       0
## 2              1         0       0       1
## 3              2         0       0       0
```

Tail()

```
tail (german_data, 3)
```

```
##      CHK_ACCT DURATION HISTORY NEW_CAR USED_CAR FURNITURE RADIO.TV
## 998         3       12       2       0        0         0        1
## 999         0       45       2       0        0         0        1
## 1000        1       45       4       0        1         0        0
##      EDUCATION RETRAINING AMOUNT SAV_ACCT EMPLOYMENT INSTALL_RATE MALE_DIV
## 998          0          0    804        0          4            4        0
## 999          0          0   1845        0          2            4        0
## 1000         0          0   4576        1          0            3        0
```

```
##      MALE_SINGLE MALE_MAR_or_WID CO.APPLICANT GUARANTOR PRESENT_RESIDENT
## 998            1               0            0         0                4
## 999            1               0            0         0                4
## 1000           1               0            0         0                4
##      REAL_ESTATE PROP_UNKN_NONE AGE OTHER_INSTALL RENT OWN_RES NUM_CREDITS
## 998            0              0  38             0    0       1           1
## 999            0              1  23             0    0       0           1
## 1000           0              0  27             0    0       1           1
##      JOB NUM_DEPENDENTS TELEPHONE FOREIGN DEFAULT
## 998    2              1         0       0       0
## 999    2              1         1       0       1
## 1000   2              1         0       0       0
```

Data Conversion

```
##convert the outcome to factor
german_data$DEFAULT <- as.factor (ifelse(german_data$DEFAULT == 1,
                                  "Default" , "NonDefault"))
##convert to factor
german_data$CHK_ACCT <- as.factor(ifelse(german_data$CHK_ACCT == 0, "below 0",
                                   ifelse(german_data$CHK_ACCT == 1, "below 200",
                                   ifelse(german_data$CHK_ACCT == 2, "over 200",
                                   ifelse(german_data$CHK_ACCT == 3, "no_acct",
                                          ""))))
##convert to factor
german_data$PURPOSE <- as.factor((ifelse (german_data$NEW_CAR == 1, "new_car",
                                  ifelse (german_data$USED_CAR == 1, "used_car",
                                  ifelse (german_data$FURNITURE == 1, "furniture",
                                  ifelse (german_data$RADIO.TV == 1, "radio_tv",
                                  ifelse (german_data$EDUCATION == 1, "education",
                                  ifelse (german_data$RETRAINING == 1,"retraining",
                                          "no_purpose")))))))
##convert to factor
german_data$HISTORY <- as.factor(ifelse (german_data$HISTORY == 0, "no_credit",
                                 ifelse (german_data$HISTORY == 1, "credit_paid",
                                 ifelse (german_data$HISTORY == 2, "existing_paid",
                                 ifelse (german_data$HISTORY == 3, "delay",
                                 ifelse (german_data$HISTORY == 4, "critical_acct",
                                         ""))))))
##convert to factor
german_data$SAV_ACCT <- as.factor(ifelse (german_data$SAV_ACCT == 0, "<100",
                                   ifelse (german_data$SAV_ACCT == 1, "100<=...<500",
                                   ifelse (german_data$SAV_ACCT == 2, "500<=...<1000",
                                   ifelse (german_data$SAV_ACCT == 3, "=>1000",
                                   ifelse (german_data$SAV_ACCT == 4, "unk/no_acct",
                                           "")))))
##convert to factor
german_data$JOB <- as.factor(ifelse (german_data$JOB == 0, "unskilled-non-res",
                             ifelse (german_data$JOB == 1, "unskilled-res",
                             ifelse (german_data$JOB == 2, "skilled-emp",
                             ifelse (german_data$JOB == 3, "mgmt/officer", "")))))
##convert to factor
```

```
german_data$EMPLOYMENT <- as.factor(ifelse (german_data$EMPLOYMENT == 0, "unemployed",
                                    ifelse (german_data$EMPLOYMENT == 1,  "<1year",
                                    ifelse (german_data$EMPLOYMENT == 2, "1<=...<4years",
                                    ifelse (german_data$EMPLOYMENT == 3, "4<=...<7years",
                                    ifelse (german_data$EMPLOYMENT == 4, ">=7years",
                                            ""))))))
##merge status in one variable
german_data$STATUS <- as.factor(ifelse (german_data$MALE_DIV==1, "male-div",
                                ifelse (german_data$MALE_SINGLE== 1, "male-single",
                                ifelse (german_data$MALE_MAR_or_WID == 1, "male-marr-wid",
                                        "unknown"))))
##convert to factor
german_data$RESIDENCY <- as.factor(ifelse (german_data$PRESENT_RESIDENT==1, "<=1year",
                                    ifelse (german_data$PRESENT_RESIDENT== 2, "1<..<=2years",
                                    ifelse (german_data$PRESENT_RESIDENT == 3, "2<..<=3years",
                                    ifelse (german_data$PRESENT_RESIDENT == 4, ">4years",
                                            0 )))))
##merge debtors in one variable
german_data$DEBTOR <- as.factor(ifelse (german_data$CO.APPLICANT == 1, "co-applicant",
                                ifelse (german_data$GUARANTOR == 1,"guarantor", "none")))

##merge property in one variable
german_data$PROPERTY <- as.factor(ifelse (german_data$REAL_ESTATE == 1, "realstate",
                                  ifelse (german_data$PROP_UNKN_NONE== 1, "unk-prop",
                                  "none")))
##merge housing in one variable
german_data$HOUSING <- as.factor(ifelse (german_data$RENT == 1, "rent",
                                 ifelse (german_data$OWN_RES == 1, "own-resid",
                                 "unknown")))
##convert to factor
german_data$TELEPHONE <- as.factor(ifelse (german_data$TELEPHONE == 1, "yes", "no"))
##convert to factor
german_data$FOREIGN <- as.factor(ifelse (german_data$FOREIGN == 1, "yes", "no"))
##convert to factor
german_data$INSTALL_RATE <- as.factor(german_data$INSTALL_RATE)
##convert to factor
german_data$PRESENT_RESIDENT <- as.factor(german_data$PRESENT_RESIDENT)
##convert to factor
german_data$OTHER_INSTALL <- as.factor(german_data$OTHER_INSTALL)
##convert to factor
german_data$NUM_DEPENDENTS <- as.factor(german_data$NUM_DEPENDENTS)
##convert to factor
german_data$NUM_CREDITS <- as.factor(german_data$NUM_CREDITS)
```

Check Structure After Transformation

```
head(german_data)
```

```
##    CHK_ACCT DURATION       HISTORY NEW_CAR USED_CAR FURNITURE RADIO.TV
## 1   below 0        6 critical_acct       0        0         0        1
## 2 below 200       48 existing_paid       0        0         0        1
```

```
## 3   no_acct       12 critical_acct       0        0         0        0
## 4   below 0       42 existing_paid       0        0         1        0
## 5   below 0       24         delay       1        0         0        0
## 6   no_acct       36 existing_paid       0        0         0        0
##   EDUCATION RETRAINING AMOUNT    SAV_ACCT    EMPLOYMENT INSTALL_RATE
## 1         0          0   1169 unk/no_acct      >=7years            4
## 2         0          0   5951        <100 1<=...<4years            2
## 3         1          0   2096        <100 4<=...<7years            2
## 4         0          0   7882        <100 4<=...<7years            2
## 5         0          0   4870        <100 1<=...<4years            3
## 6         1          0   9055 unk/no_acct 1<=...<4years            2
##   MALE_DIV MALE_SINGLE MALE_MAR_or_WID CO.APPLICANT GUARANTOR
## 1        0           1               0            0         0
## 2        0           0               0            0         0
## 3        0           1               0            0         0
## 4        0           1               0            0         1
## 5        0           1               0            0         0
## 6        0           1               0            0         0
##   PRESENT_RESIDENT REAL_ESTATE PROP_UNKN_NONE AGE OTHER_INSTALL RENT
## 1                4           1              0  67             0    0
## 2                2           1              0  22             0    0
## 3                3           1              0  49             0    0
## 4                4           0              0  45             0    0
## 5                4           0              1  53             0    0
## 6                4           0              1  35             0    0
##   OWN_RES NUM_CREDITS           JOB NUM_DEPENDENTS TELEPHONE FOREIGN
## 1       1           2   skilled-emp              1       yes      no
## 2       1           1   skilled-emp              1        no      no
## 3       1           1 unskilled-res              2        no      no
## 4       0           1   skilled-emp              2        no      no
## 5       0           2   skilled-emp              2        no      no
## 6       0           1 unskilled-res              2       yes      no
##      DEFAULT   PURPOSE      STATUS    RESIDENCY    DEBTOR  PROPERTY
## 1 NonDefault  radio_tv male-single      >4years      none realstate
## 2    Default  radio_tv     unknown 1<..<=2years      none realstate
## 3 NonDefault education male-single 2<..<=3years      none realstate
## 4 NonDefault furniture male-single      >4years guarantor      none
## 5    Default   new_car male-single      >4years      none  unk-prop
## 6 NonDefault education male-single      >4years      none  unk-prop
##     HOUSING
## 1 own-resid
## 2 own-resid
## 3 own-resid
## 4   unknown
## 5   unknown
## 6   unknown
```

```
tail(german_data)
```

```
##       CHK_ACCT DURATION       HISTORY NEW_CAR USED_CAR FURNITURE RADIO.TV
## 995    no_acct       12 existing_paid       1        0         0        0
## 996    no_acct       12 existing_paid       0        0         1        0
## 997    below 0       30 existing_paid       0        1         0        0
## 998    no_acct       12 existing_paid       0        0         0        1
```

```
## 999    below 0       45 existing_paid       0        0         0        1
## 1000 below 200       45 critical_acct       0        1         0        0
##      EDUCATION RETRAINING AMOUNT     SAV_ACCT    EMPLOYMENT INSTALL_RATE
## 995          0          0   2390  unk/no_acct      >=7years            4
## 996          0          0   1736         <100 4<=...<7years            3
## 997          0          0   3857         <100 1<=...<4years            4
## 998          0          0    804         <100      >=7years            4
## 999          0          0   1845         <100 1<=...<4years            4
## 1000         0          0   4576 100<=...<500    unemployed            3
##      MALE_DIV MALE_SINGLE MALE_MAR_or_WID CO.APPLICANT GUARANTOR
## 995         0           1               0            0         0
## 996         0           0               0            0         0
## 997         1           0               0            0         0
## 998         0           1               0            0         0
## 999         0           1               0            0         0
## 1000        0           1               0            0         0
##      PRESENT_RESIDENT REAL_ESTATE PROP_UNKN_NONE AGE OTHER_INSTALL RENT
## 995                 3           0              0  50             0    0
## 996                 4           1              0  31             0    0
## 997                 4           0              0  40             0    0
## 998                 4           0              0  38             0    0
## 999                 4           0              1  23             0    0
## 1000                4           0              0  27             0    0
##      OWN_RES NUM_CREDITS           JOB NUM_DEPENDENTS TELEPHONE FOREIGN
## 995        1           1   skilled-emp              1       yes      no
## 996        1           1 unskilled-res              1        no      no
## 997        1           1  mgmt/officer              1       yes      no
## 998        1           1   skilled-emp              1        no      no
## 999        0           1   skilled-emp              1       yes      no
## 1000       1           1   skilled-emp              1        no      no
##         DEFAULT   PURPOSE      STATUS    RESIDENCY DEBTOR  PROPERTY
## 995  NonDefault   new_car male-single 2<..<=3years   none      none
## 996  NonDefault furniture     unknown      >4years   none realstate
## 997  NonDefault  used_car    male-div      >4years   none      none
## 998  NonDefault  radio_tv male-single      >4years   none      none
## 999     Default  radio_tv male-single      >4years   none  unk-prop
## 1000 NonDefault  used_car male-single      >4years   none      none
##        HOUSING
## 995  own-resid
## 996  own-resid
## 997  own-resid
## 998  own-resid
## 999    unknown
## 1000 own-resid
```

Sort the dataset

```
sorted_data <- german_data[c(1,2,3,32,10,11,12,13,33,
               34,19,35,22,23,36,26,27,28,29,30,31)]
```

Outcome Distribution

```
table(sorted_data$DEFAULT)
```

```
##
##    Default NonDefault
##        300        700
```

Sorted Data Structure

```
str(sorted_data)
```

```
## 'data.frame':    1000 obs. of  21 variables:
##  $ CHK_ACCT        : Factor w/ 4 levels "below 0","below 200",..: 1 2 3 1 1 3 3 2 3 2 ...
##  $ DURATION        : int  6 48 12 42 24 36 24 36 12 30 ...
##  $ HISTORY         : Factor w/ 5 levels "credit_paid",..: 2 4 2 4 3 4 4 4 4 2 ...
##  $ PURPOSE         : Factor w/ 7 levels "education","furniture",..: 5 5 1 2 3 1 2 7 5 3 ...
##  $ AMOUNT          : int  1169 5951 2096 7882 4870 9055 2835 6948 3059 5234 ...
##  $ SAV_ACCT        : Factor w/ 5 levels "<100","=>1000",..: 5 1 1 1 1 5 4 1 2 1 ...
##  $ EMPLOYMENT      : Factor w/ 5 levels "<1year",">=7years",..: 2 3 4 4 3 3 2 3 4 5 ...
##  $ INSTALL_RATE    : Factor w/ 4 levels "1","2","3","4": 4 2 2 2 3 2 3 2 2 4 ...
##  $ STATUS          : Factor w/ 4 levels "male-div","male-marr-wid",..: 3 4 3 3 3 3 3 3 1 2 ...
##  $ RESIDENCY       : Factor w/ 4 levels "<=1year",">4years",..: 2 3 4 2 2 2 2 3 2 3 ...
##  $ PRESENT_RESIDENT: Factor w/ 4 levels "1","2","3","4": 4 2 3 4 4 4 4 2 4 2 ...
##  $ DEBTOR          : Factor w/ 3 levels "co-applicant",..: 3 3 3 2 3 3 3 3 3 3 ...
##  $ AGE             : int  67 22 49 45 53 35 53 35 61 28 ...
##  $ OTHER_INSTALL   : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ PROPERTY        : Factor w/ 3 levels "none","realstate",..: 2 2 2 1 3 3 1 1 2 1 ...
##  $ NUM_CREDITS     : Factor w/ 4 levels "1","2","3","4": 2 1 1 1 2 1 1 1 1 2 ...
##  $ JOB             : Factor w/ 4 levels "mgmt/officer",..: 2 2 4 2 2 4 2 1 4 1 ...
##  $ NUM_DEPENDENTS  : Factor w/ 2 levels "1","2": 1 1 2 2 2 2 1 1 1 1 ...
##  $ TELEPHONE       : Factor w/ 2 levels "no","yes": 2 1 1 1 1 2 1 2 1 1 ...
##  $ FOREIGN         : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ DEFAULT         : Factor w/ 2 levels "Default","NonDefault": 2 1 2 2 1 2 2 2 2 1 ...
```

Summary Numeric Variables

```
summary(sorted_data[c(2, 5, 13)])
```

```
##     DURATION        AMOUNT           AGE
##  Min.   : 4.0   Min.   :  250   Min.   :19.00
##  1st Qu.:12.0   1st Qu.: 1366   1st Qu.:27.00
##  Median :18.0   Median : 2320   Median :33.00
##  Mean   :20.9   Mean   : 3271   Mean   :35.55
##  3rd Qu.:24.0   3rd Qu.: 3972   3rd Qu.:42.00
##  Max.   :72.0   Max.   :18424   Max.   :75.00
```

Check Missing Values

```
missing<-function(x){
  return (sum(is.na(x)))
}
apply(sorted_data,2,missing)
```

```
##         CHK_ACCT         DURATION          HISTORY          PURPOSE
##                0                0                0                0
##           AMOUNT         SAV_ACCT       EMPLOYMENT     INSTALL_RATE
##                0                0                0                0
##           STATUS        RESIDENCY PRESENT_RESIDENT           DEBTOR
##                0                0                0                0
##              AGE    OTHER_INSTALL         PROPERTY      NUM_CREDITS
##                0                0                0                0
##              JOB   NUM_DEPENDENTS        TELEPHONE          FOREIGN
##                0                0                0                0
##          DEFAULT
##                0
```

Attach Dataset For Data Manipulation

```
attach(sorted_data)
```

Exploratory Data Analysis

Contingency Table, DEFAULT vs CHK_ACCT

```
CrossTable(DEFAULT, CHK_ACCT, digits=1, prop.r=F,
           prop.t=F, prop.chisq=F, chisq=T)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  1000
##
##
##              | CHK_ACCT
##      DEFAULT |   below 0 | below 200 |   no_acct |  over 200 | Row Total |
## -------------|-----------|-----------|-----------|-----------|-----------|
##      Default |       135 |       105 |        46 |        14 |       300 |
##              |       0.5 |       0.4 |       0.1 |       0.2 |           |
## -------------|-----------|-----------|-----------|-----------|-----------|
##   NonDefault |       139 |       164 |       348 |        49 |       700 |
```

```
##              |       0.5 |       0.6 |       0.9 |       0.8 |           |
## -------------|-----------|-----------|-----------|-----------|-----------|
## Column Total |       274 |       269 |       394 |        63 |      1000 |
##              |       0.3 |       0.3 |       0.4 |       0.1 |           |
## -------------|-----------|-----------|-----------|-----------|-----------|
##
##
## Statistics for All Table Factors
##
##
## Pearson's Chi-squared test
## ------------------------------------------------------------
## Chi^2 =  123.7209     d.f. =  3     p =  1.218902e-26
##
##
##
```

Contingency Table DEFAULT vs SAV_ACCT

```
CrossTable(DEFAULT, FOREIGN, digits=1, prop.r=F,
           prop.t=F, prop.chisq=F, chisq=T)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  1000
##
##
##              | FOREIGN
##      DEFAULT |        no |       yes | Row Total |
## -------------|-----------|-----------|-----------|
##      Default |       296 |         4 |       300 |
##              |       0.3 |       0.1 |           |
## -------------|-----------|-----------|-----------|
##   NonDefault |       667 |        33 |       700 |
##              |       0.7 |       0.9 |           |
## -------------|-----------|-----------|-----------|
## Column Total |       963 |        37 |      1000 |
##              |       1.0 |       0.0 |           |
## -------------|-----------|-----------|-----------|
##
##
## Statistics for All Table Factors
##
##
## Pearson's Chi-squared test
```

```
## ------------------------------------------------------------
## Chi^2 =  6.737044     d.f. =  1     p =  0.009443096
##
## Pearson's Chi-squared test with Yates' continuity correction
## ------------------------------------------------------------
## Chi^2 =  5.821576     d.f. =  1     p =  0.01583075
##
##
```

Proportion Tables

```
margin.table(prop.table(table(PRESENT_RESIDENT, OTHER_INSTALL, RESIDENCY,
             NUM_CREDITS, NUM_DEPENDENTS, EMPLOYMENT, TELEPHONE, FOREIGN)), 1)
```

```
## PRESENT_RESIDENT
##     1     2     3     4
## 0.130 0.308 0.149 0.413
```

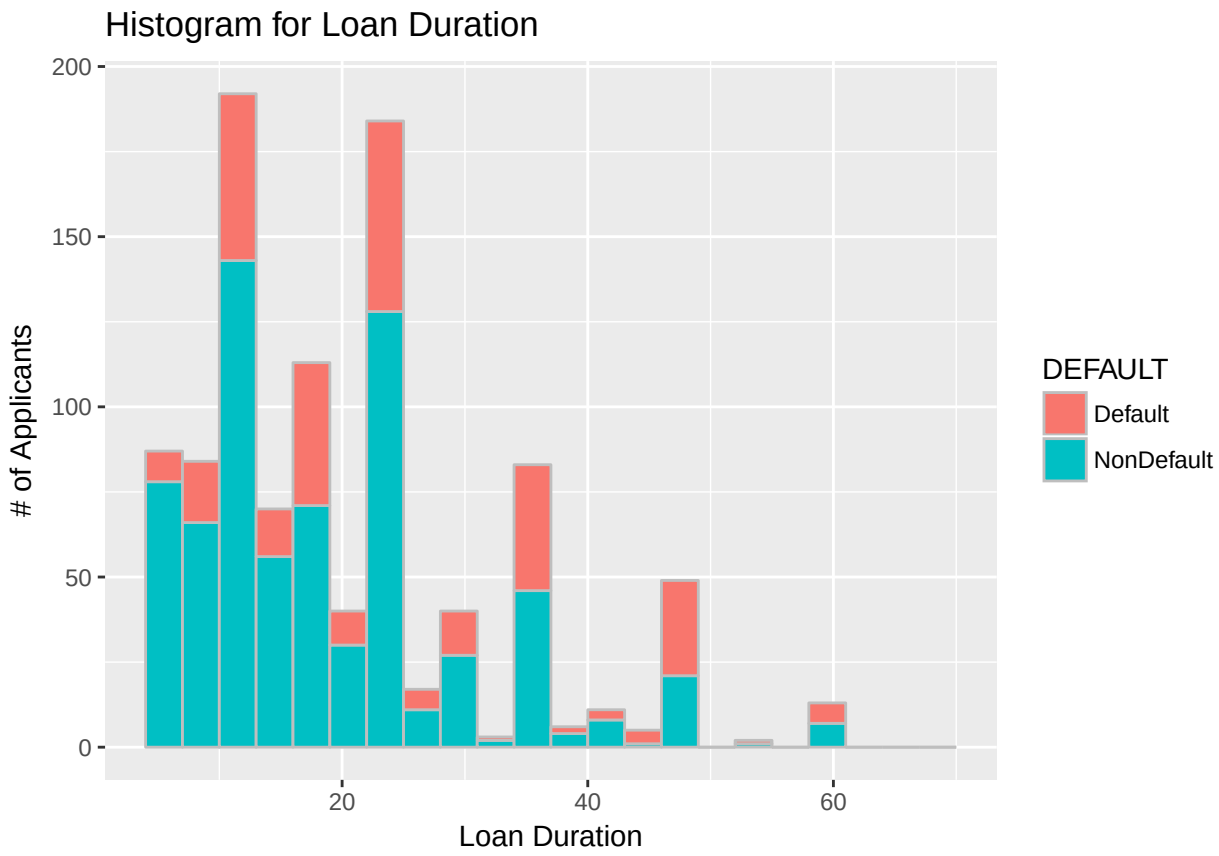
```
margin.table(prop.table(table(PRESENT_RESIDENT, OTHER_INSTALL, RESIDENCY,
             NUM_CREDITS, NUM_DEPENDENTS, EMPLOYMENT, TELEPHONE, FOREIGN)), 3)
```

```
## RESIDENCY
##      <=1year      >4years 1<..<=2years 2<..<=3years
##        0.130        0.413        0.308        0.149
```

Descriptive analysis

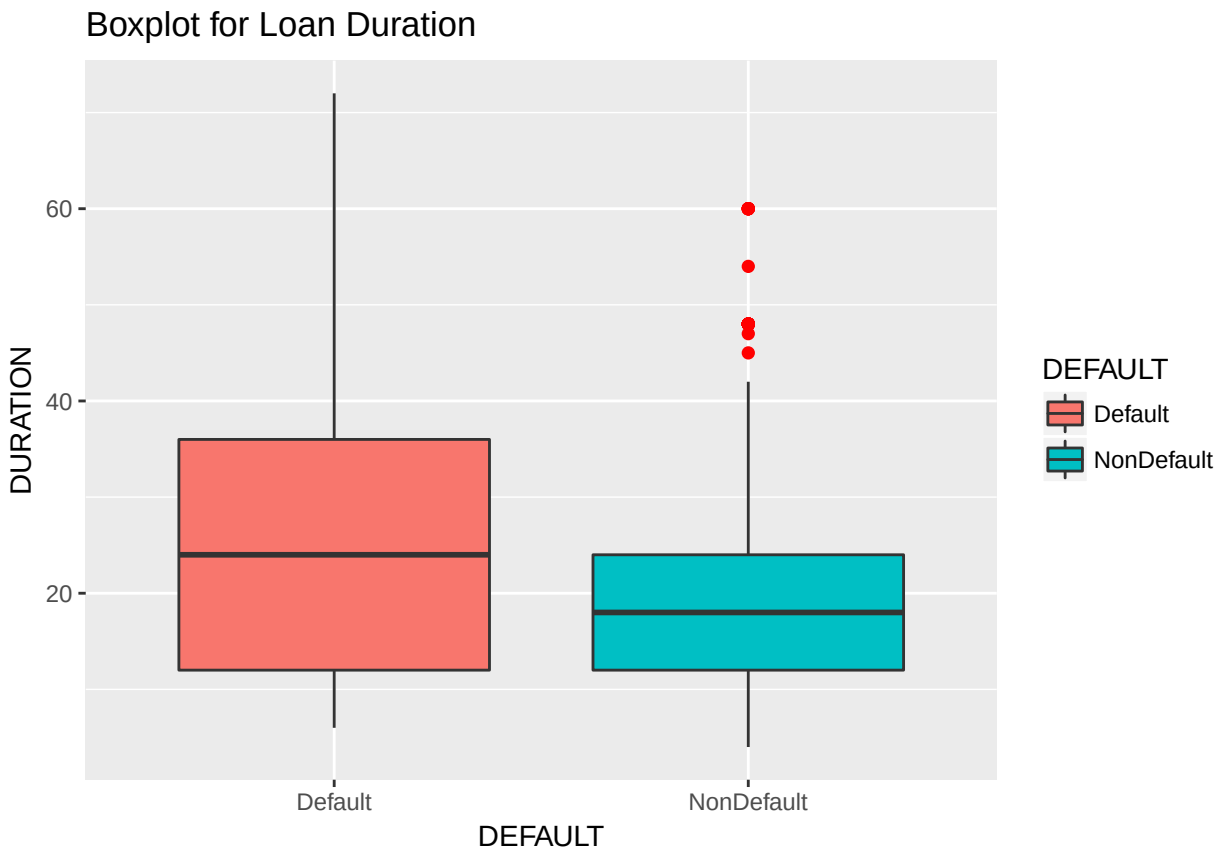
DURATION Histogram

```
##histogram for Loan Duration
ggplot(data = sorted_data, aes(DURATION)) + geom_histogram(breaks = seq(4, 72,
       by =3), col = "grey", aes(fill = DEFAULT)) +
       labs (title = "Histogram for Loan Duration") +
       labs(x = "Loan Duration", y = "# of Applicants")
```



DURATION Boxplot

```
ggplot(sorted_data, aes(x = DEFAULT, y = DURATION, fill = DEFAULT)) +
       geom_boxplot(outlier.colour = "red", outlier.shape = 16,
       outlier.size = 2) + labs (title = "Boxplot for Loan Duration")
```



Model Building

Split data 70,30 (test, train)

```
set.seed(100)
gdata <- sort(sample(nrow(sorted_data), nrow(sorted_data)*0.7))

train <- sorted_data[gdata, ]
test <- sorted_data[-gdata, ]
```

Verify outcome

```
prop.table(table(train$DEFAULT))
```

```
##
##    Default NonDefault
##  0.3257143  0.6742857
```

```
prop.table(table(test$DEFAULT))
```

```
##
##    Default NonDefault
##       0.24       0.76
```

Decision Tree Model

```
set.seed(100)
credit_tree <- rpart(DEFAULT~., data = train, method = "class",
                control = rpart.control(minsplit = 20, minbucket = 7,
                maxdepth = 10, usesurrogate = 2, xval = 10 ))
```

Tree Summary

```
summary(credit_tree)
```

```
## Call:
## rpart(formula = DEFAULT ~ ., data = train, method = "class",
##     control = rpart.control(minsplit = 20, minbucket = 7, maxdepth = 10,
##         usesurrogate = 2, xval = 10))
##   n= 700
##
##           CP nsplit rel error    xerror       xstd
## 1 0.07456140      0 1.0000000 1.0000000 0.05438192
## 2 0.05701754      2 0.8508772 0.9780702 0.05406644
## 3 0.03070175      3 0.7938596 0.9473684 0.05360011
## 4 0.02850877      4 0.7631579 0.9605263 0.05380352
## 5 0.02631579      6 0.7061404 0.9605263 0.05380352
## 6 0.02192982      7 0.6798246 0.9342105 0.05339129
## 7 0.01315789      9 0.6359649 0.8903509 0.05265533
## 8 0.01000000     10 0.6228070 0.9254386 0.05324904
##
## Variable importance
##         CHK_ACCT           AMOUNT         DURATION          HISTORY
##               28               16               14               13
##         SAV_ACCT          PURPOSE PRESENT_RESIDENT        RESIDENCY
##               12                7                2                2
##       EMPLOYMENT              AGE         PROPERTY           DEBTOR
##                2                1                1                1
##      NUM_CREDITS           STATUS
##                1                1
##
## Node number 1: 700 observations,    complexity param=0.0745614
##   predicted class=NonDefault  expected loss=0.3257143  P(node) =1
##     class counts:   228   472
##    probabilities: 0.326 0.674
##   left son=2 (379 obs) right son=3 (321 obs)
##   Primary splits:
##       CHK_ACCT splits as  LLRR,       improve=34.24850, (0 missing)
##       HISTORY  splits as  LRLLL,      improve=14.28245, (0 missing)
##       AMOUNT   < 3909.5 to the right, improve=12.73511, (0 missing)
##       DURATION < 33     to the right, improve=12.08177, (0 missing)
##       SAV_ACCT splits as  LRLRR,      improve= 8.64850, (0 missing)
##   Surrogate splits:
##       SAV_ACCT         splits as  LLLRR,   agree=0.600, adj=0.128, (0 split)
##       HISTORY          splits as  LRLLL,   agree=0.590, adj=0.106, (0 split)
```

```
##       PURPOSE          splits as  LLLLRLR, agree=0.577, adj=0.078, (0 split)
##       RESIDENCY        splits as  LLRL,    agree=0.571, adj=0.065, (0 split)
##       PRESENT_RESIDENT splits as  LRLL,    agree=0.571, adj=0.065, (0 split)
##
## Node number 2: 379 observations,    complexity param=0.0745614
##   predicted class=NonDefault  expected loss=0.469657  P(node) =0.5414286
##     class counts:   178   201
##    probabilities: 0.470 0.530
##   left son=4 (92 obs) right son=5 (287 obs)
##   Primary splits:
##       DURATION < 27.5   to the right, improve=11.245000, (0 missing)
##       PROPERTY splits as  LRL,        improve=10.629530, (0 missing)
##       AMOUNT   < 3998   to the right, improve= 9.493808, (0 missing)
##       HISTORY  splits as  LRRLL,      improve= 7.591871, (0 missing)
##       DEBTOR   splits as  LRL,        improve= 5.123438, (0 missing)
##   Surrogate splits:
##       AMOUNT   < 4195   to the right, agree=0.826, adj=0.283, (0 split)
##       HISTORY  splits as  RRRRL,      agree=0.768, adj=0.043, (0 split)
##       PROPERTY splits as  RRL,        agree=0.765, adj=0.033, (0 split)
##
## Node number 3: 321 observations
##   predicted class=NonDefault  expected loss=0.1557632  P(node) =0.4585714
##     class counts:    50   271
##    probabilities: 0.156 0.844
##
## Node number 4: 92 observations,    complexity param=0.02192982
##   predicted class=Default     expected loss=0.3152174  P(node) =0.1314286
##     class counts:    63    29
##    probabilities: 0.685 0.315
##   left son=8 (59 obs) right son=9 (33 obs)
##   Primary splits:
##       HISTORY    splits as  RRRLL,      improve=2.961356, (0 missing)
##       AGE        < 27.5   to the left,  improve=1.887554, (0 missing)
##       SAV_ACCT   splits as  LRR-R,      improve=1.551890, (0 missing)
##       EMPLOYMENT splits as  LLLRR,      improve=1.499548, (0 missing)
##       RESIDENCY  splits as  RLLR,       improve=1.410215, (0 missing)
##   Surrogate splits:
##       NUM_CREDITS      splits as  LRL-,    agree=0.707, adj=0.182, (0 split)
##       EMPLOYMENT       splits as  LLLLR,   agree=0.663, adj=0.061, (0 split)
##       RESIDENCY        splits as  LLLR,    agree=0.663, adj=0.061, (0 split)
##       PRESENT_RESIDENT splits as  LLRL,    agree=0.663, adj=0.061, (0 split)
##       PURPOSE          splits as  LLLLLLR, agree=0.652, adj=0.030, (0 split)
##
## Node number 5: 287 observations,    complexity param=0.05701754
##   predicted class=NonDefault  expected loss=0.4006969  P(node) =0.41
##     class counts:   115   172
##    probabilities: 0.401 0.599
##   left son=10 (27 obs) right son=11 (260 obs)
##   Primary splits:
##       HISTORY  splits as  LRRRL,      improve=6.892428, (0 missing)
##       PROPERTY splits as  LRL,        improve=6.226454, (0 missing)
##       PURPOSE  splits as  LRLLRRR,    improve=4.607535, (0 missing)
##       DURATION < 11.5   to the right, improve=4.581512, (0 missing)
```

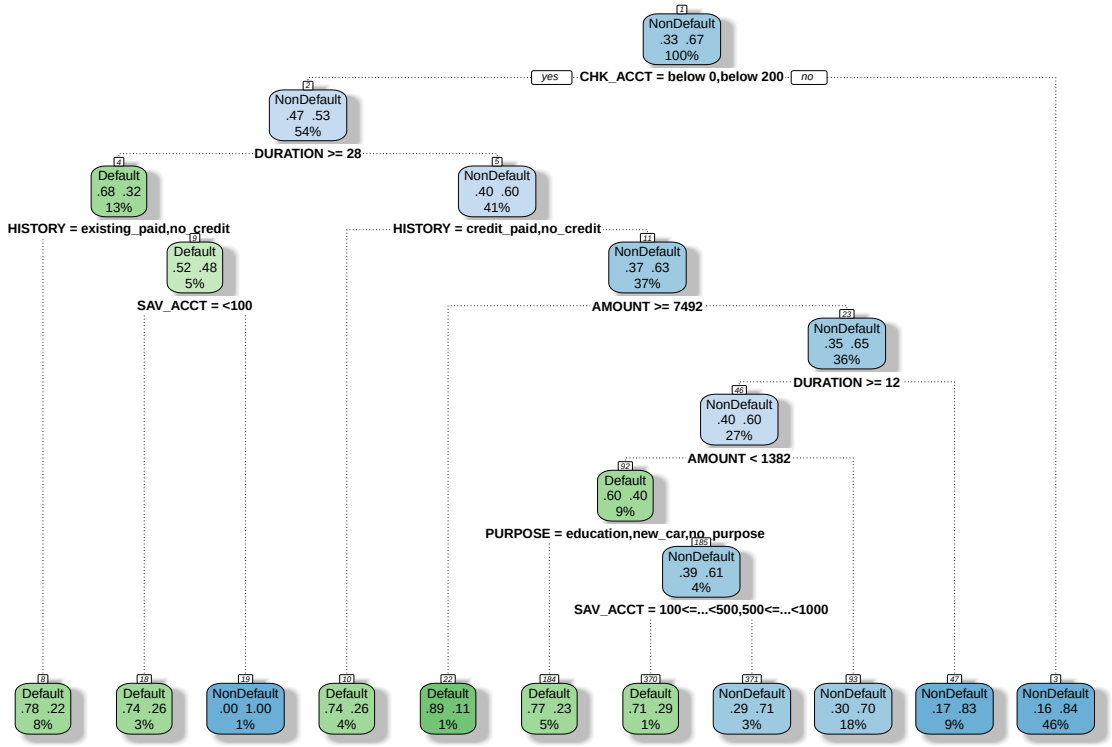
```
##       AMOUNT   < 7491.5 to the right, improve=4.428850, (0 missing)
##
## Node number 8: 59 observations
##   predicted class=Default     expected loss=0.220339  P(node) =0.08428571
##     class counts:    46    13
##    probabilities: 0.780 0.220
##
## Node number 9: 33 observations,    complexity param=0.02192982
##   predicted class=Default     expected loss=0.4848485  P(node) =0.04714286
##     class counts:    17    16
##    probabilities: 0.515 0.485
##   left son=18 (23 obs) right son=19 (10 obs)
##   Primary splits:
##       SAV_ACCT     splits as  LRR-R,      improve=7.615283, (0 missing)
##       CHK_ACCT     splits as  LR--,       improve=3.426025, (0 missing)
##       AGE          < 27.5   to the left,  improve=2.734848, (0 missing)
##       INSTALL_RATE splits as  RLRL,       improve=1.328327, (0 missing)
##       NUM_CREDITS  splits as  RLR-,       improve=1.262626, (0 missing)
##   Surrogate splits:
##       AMOUNT     < 9743   to the left,  agree=0.788, adj=0.3, (0 split)
##       HISTORY    splits as  RLL--,      agree=0.758, adj=0.2, (0 split)
##       PURPOSE    splits as  LLLRLLL,    agree=0.758, adj=0.2, (0 split)
##       EMPLOYMENT splits as  LLLLR,      agree=0.758, adj=0.2, (0 split)
##       CHK_ACCT   splits as  LR--,       agree=0.727, adj=0.1, (0 split)
##
## Node number 10: 27 observations
##   predicted class=Default     expected loss=0.2592593  P(node) =0.03857143
##     class counts:    20     7
##    probabilities: 0.741 0.259
##
## Node number 11: 260 observations,    complexity param=0.03070175
##   predicted class=NonDefault  expected loss=0.3653846  P(node) =0.3714286
##     class counts:    95   165
##    probabilities: 0.365 0.635
##   left son=22 (9 obs) right son=23 (251 obs)
##   Primary splits:
##       AMOUNT   < 7491.5 to the right, improve=5.109902, (0 missing)
##       DURATION < 11.5   to the right, improve=4.807943, (0 missing)
##       DEBTOR   splits as  LRL,        improve=4.310256, (0 missing)
##       HISTORY  splits as  -RLL-,      improve=3.830519, (0 missing)
##       STATUS   splits as  LRRL,       improve=3.681166, (0 missing)
##
## Node number 18: 23 observations
##   predicted class=Default     expected loss=0.2608696  P(node) =0.03285714
##     class counts:    17     6
##    probabilities: 0.739 0.261
##
## Node number 19: 10 observations
##   predicted class=NonDefault  expected loss=0  P(node) =0.01428571
##     class counts:     0    10
##    probabilities: 0.000 1.000
##
## Node number 22: 9 observations
```

```
##   predicted class=Default     expected loss=0.1111111  P(node) =0.01285714
##     class counts:     8     1
##    probabilities: 0.889 0.111
##
## Node number 23: 251 observations,    complexity param=0.02850877
##   predicted class=NonDefault  expected loss=0.3466135  P(node) =0.3585714
##     class counts:    87   164
##    probabilities: 0.347 0.653
##   left son=46 (191 obs) right son=47 (60 obs)
##   Primary splits:
##       DURATION < 11.5   to the right, improve=5.106346, (0 missing)
##       AMOUNT   < 1381.5 to the left,  improve=4.350911, (0 missing)
##       STATUS   splits as  LRRL,       improve=4.348010, (0 missing)
##       DEBTOR   splits as  LRL,        improve=3.823875, (0 missing)
##       PURPOSE  splits as  LLLLRRR,    improve=3.549291, (0 missing)
##   Surrogate splits:
##       AMOUNT  < 670    to the right, agree=0.801, adj=0.167, (0 split)
##       FOREIGN splits as  LR,         agree=0.785, adj=0.100, (0 split)
##       AGE     < 66.5   to the left,  agree=0.777, adj=0.067, (0 split)
##
## Node number 46: 191 observations,    complexity param=0.02850877
##   predicted class=NonDefault  expected loss=0.4031414  P(node) =0.2728571
##     class counts:    77   114
##    probabilities: 0.403 0.597
##   left son=92 (63 obs) right son=93 (128 obs)
##   Primary splits:
##       AMOUNT   < 1381.5 to the left,  improve=7.523125, (0 missing)
##       PURPOSE  splits as  LRLLRRR,    improve=4.588354, (0 missing)
##       DEBTOR   splits as  LRL,        improve=3.324624, (0 missing)
##       HISTORY  splits as  -RRL-,      improve=2.592570, (0 missing)
##       CHK_ACCT splits as  LR--,       improve=2.479747, (0 missing)
##   Surrogate splits:
##       DURATION    < 12.5   to the left,  agree=0.728, adj=0.175, (0 split)
##       PURPOSE     splits as  LRLRRRR,    agree=0.691, adj=0.063, (0 split)
##       AGE         < 20.5   to the left,  agree=0.675, adj=0.016, (0 split)
##       NUM_CREDITS splits as  RRRL,       agree=0.675, adj=0.016, (0 split)
##
## Node number 47: 60 observations
##   predicted class=NonDefault  expected loss=0.1666667  P(node) =0.08571429
##     class counts:    10    50
##    probabilities: 0.167 0.833
##
## Node number 92: 63 observations,    complexity param=0.02631579
##   predicted class=Default     expected loss=0.3968254  P(node) =0.09
##     class counts:    38    25
##    probabilities: 0.603 0.397
##   left son=184 (35 obs) right son=185 (28 obs)
##   Primary splits:
##       PURPOSE       splits as  LRLLRR-, improve=4.458730, (0 missing)
##       PROPERTY      splits as  LRL,     improve=2.697192, (0 missing)
##       OTHER_INSTALL splits as  RL,      improve=2.494394, (0 missing)
##       STATUS        splits as  RRRL,    improve=2.441474, (0 missing)
##       NUM_CREDITS   splits as  LRLR,    improve=2.179138, (0 missing)
```

```
##   Surrogate splits:
##       AGE      < 21.5   to the right, agree=0.651, adj=0.214, (0 split)
##       SAV_ACCT splits as  LLRRL,      agree=0.635, adj=0.179, (0 split)
##       DEBTOR   splits as  RRL,        agree=0.635, adj=0.179, (0 split)
##       AMOUNT   < 679    to the right, agree=0.619, adj=0.143, (0 split)
##       STATUS   splits as  RRLL,       agree=0.619, adj=0.143, (0 split)
##
## Node number 93: 128 observations
##   predicted class=NonDefault  expected loss=0.3046875  P(node) =0.1828571
##     class counts:    39    89
##    probabilities: 0.305 0.695
##
## Node number 184: 35 observations
##   predicted class=Default     expected loss=0.2285714  P(node) =0.05
##     class counts:    27     8
##    probabilities: 0.771 0.229
##
## Node number 185: 28 observations,    complexity param=0.01315789
##   predicted class=NonDefault  expected loss=0.3928571  P(node) =0.04
##     class counts:    11    17
##    probabilities: 0.393 0.607
##   left son=370 (7 obs) right son=371 (21 obs)
##   Primary splits:
##       SAV_ACCT splits as  RRLLR,      improve=1.9285710, (0 missing)
##       JOB      splits as  RL-R,       improve=1.6138270, (0 missing)
##       AGE      < 32.5   to the right, improve=1.3349210, (0 missing)
##       DURATION < 16.5   to the right, improve=1.2071430, (0 missing)
##       PROPERTY splits as  LRL,        improve=0.8571429, (0 missing)
##   Surrogate splits:
##       PROPERTY     splits as  RRL,   agree=0.821, adj=0.286, (0 split)
##       EMPLOYMENT   splits as  RRRLR, agree=0.786, adj=0.143, (0 split)
##       INSTALL_RATE splits as  RRLR,  agree=0.786, adj=0.143, (0 split)
##
## Node number 370: 7 observations
##   predicted class=Default     expected loss=0.2857143  P(node) =0.01
##     class counts:     5     2
##    probabilities: 0.714 0.286
##
## Node number 371: 21 observations
##   predicted class=NonDefault  expected loss=0.2857143  P(node) =0.03
##     class counts:     6    15
##    probabilities: 0.286 0.714
```

Plot a fancy tree

```
fancyRpartPlot(credit_tree)
```



Rattle 2017−Aug−18 20:59:59 Dell

Model testing

```
credit_tree_test <- predict(credit_tree, test, type = "class")
mean(credit_tree_test==test$DEFAULT)
```

```
## [1] 0.79
```

Confusion Matrix

```
#confusion matrix on test model
table(pred=credit_tree_test, true=test$DEFAULT)
```

```
##             true
## pred         Default NonDefault
##   Default         39         30
##   NonDefault      33        198
```

CP Table

```
credit_tree$cptable
```

```
##           CP nsplit rel error    xerror       xstd
## 1 0.07456140      0 1.0000000 1.0000000 0.05438192
## 2 0.05701754      2 0.8508772 0.9780702 0.05406644
## 3 0.03070175      3 0.7938596 0.9473684 0.05360011
```

```
## 4 0.02850877      4 0.7631579 0.9605263 0.05380352
## 5 0.02631579      6 0.7061404 0.9605263 0.05380352
## 6 0.02192982      7 0.6798246 0.9342105 0.05339129
## 7 0.01315789      9 0.6359649 0.8903509 0.05265533
## 8 0.01000000     10 0.6228070 0.9254386 0.05324904
```

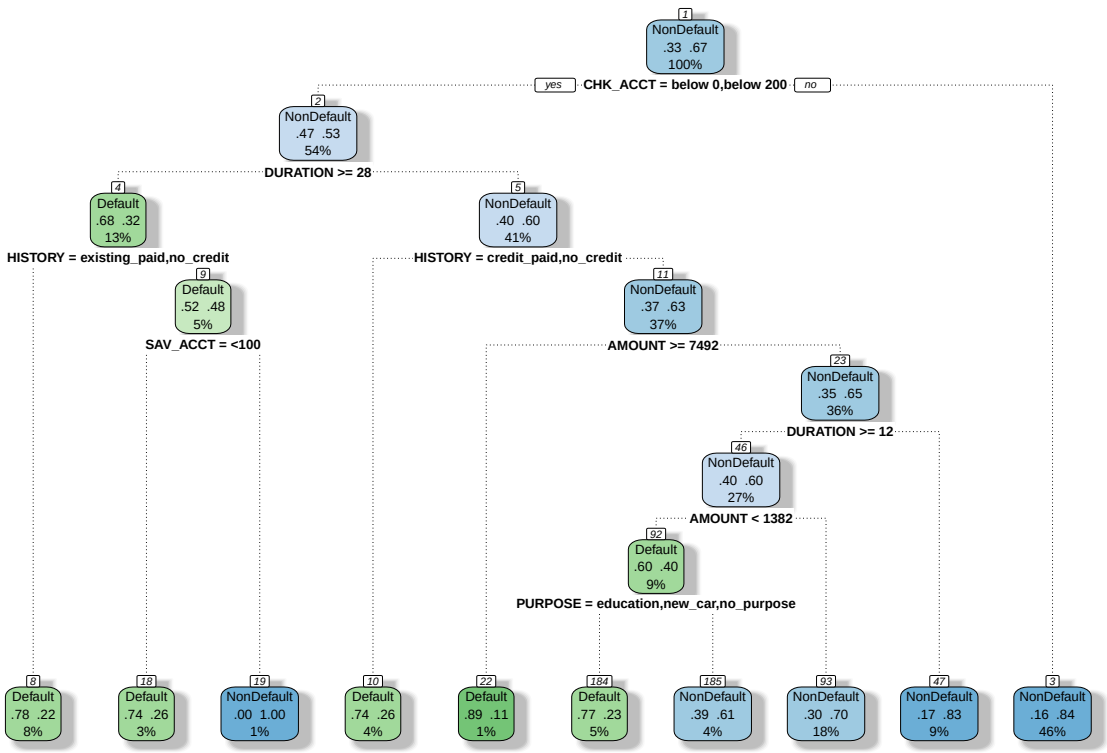
```
bestcp <- credit_tree$cptable[which.min(credit_tree$cptable[, "xerror"]), "CP"]
```

Tree prunning

```
credit_tree_pruned <- prune(credit_tree, cp = bestcp)
```

Plot pruned tree

```
fancyRpartPlot(credit_tree_pruned)
```



CP table (pruned tree)

```
printcp(credit_tree_pruned)
```

```
##
## Classification tree:
## rpart(formula = DEFAULT ~ ., data = train, method = "class",
##     control = rpart.control(minsplit = 20, minbucket = 7, maxdepth = 10,
##         usesurrogate = 2, xval = 10))
```

```
##
## Variables actually used in tree construction:
## [1] AMOUNT   CHK_ACCT DURATION HISTORY  PURPOSE  SAV_ACCT
##
## Root node error: 228/700 = 0.32571
##
## n= 700
##
##         CP nsplit rel error  xerror     xstd
## 1 0.074561      0   1.00000 1.00000 0.054382
## 2 0.057018      2   0.85088 0.97807 0.054066
## 3 0.030702      3   0.79386 0.94737 0.053600
## 4 0.028509      4   0.76316 0.96053 0.053804
## 5 0.026316      6   0.70614 0.96053 0.053804
## 6 0.021930      7   0.67982 0.93421 0.053391
## 7 0.013158      9   0.63596 0.89035 0.052655
```

Confusion matrix (pruned tree) train data

```
conf_matrix_pruned <- table(train$DEFAULT, predict(credit_tree_pruned,type="class"))
rownames(conf_matrix_pruned) <- paste("Actual", rownames(conf_matrix_pruned), sep = ":")
colnames(conf_matrix_pruned) <- paste("Pred", colnames(conf_matrix_pruned), sep = ":")
print(conf_matrix_pruned)
```

```
##
##                      Pred:Default Pred:NonDefault
##   Actual:Default              118             110
##   Actual:NonDefault            35             437
```

Confusion matrix (pruned tree) test data

```
pred_tree_pruned <- predict(credit_tree_pruned, test, type = "class")
confusionMatrix(test$DEFAULT, pred_tree_pruned)
```

```
## Confusion Matrix and Statistics
##
##             Reference
## Prediction   Default NonDefault
##   Default         38         34
##   NonDefault      29        199
##
##                Accuracy : 0.79
##                  95% CI : (0.7395, 0.8347)
##     No Information Rate : 0.7767
##     P-Value [Acc > NIR] : 0.3173
##
##                   Kappa : 0.4103
##  Mcnemar's Test P-Value : 0.6143
##
##             Sensitivity : 0.5672
##             Specificity : 0.8541
```

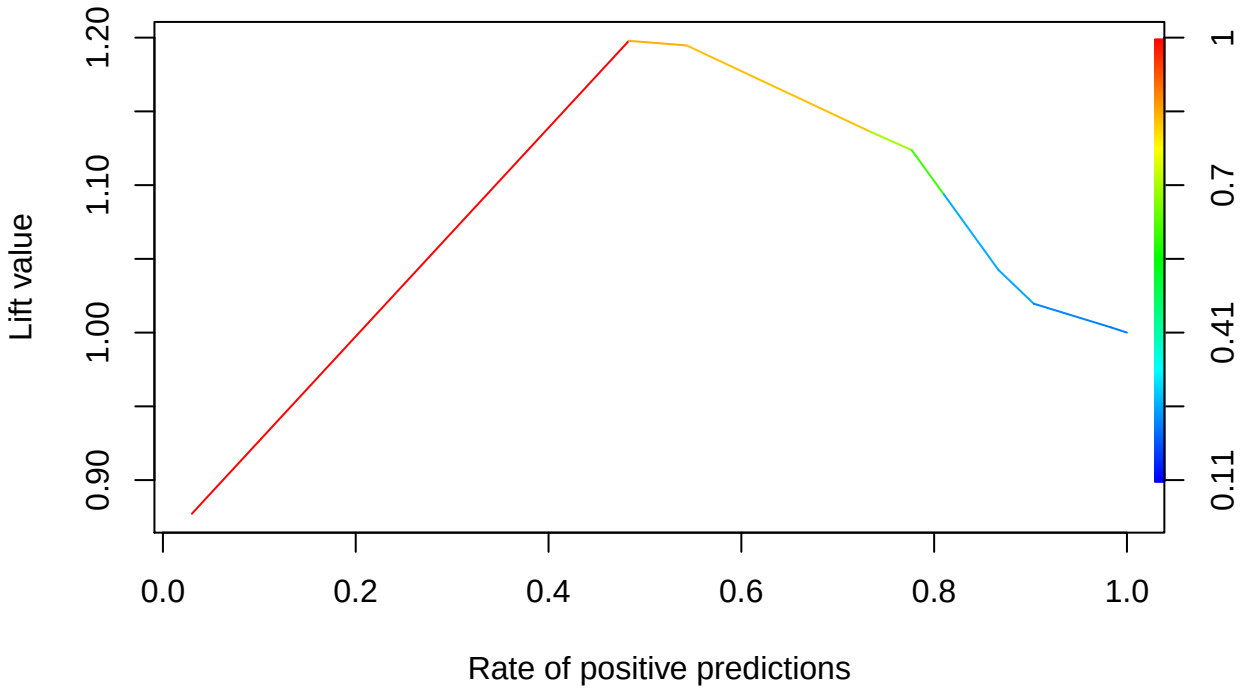
```
##          Pos Pred Value : 0.5278
##          Neg Pred Value : 0.8728
##              Prevalence : 0.2233
##          Detection Rate : 0.1267
##    Detection Prevalence : 0.2400
##       Balanced Accuracy : 0.7106
##
##        'Positive' Class : Default
##
```

Model scoring

```
tree_score = predict(credit_tree_pruned, test, type = "prob")
#storing model performance scores
pred_tree_val <-prediction(tree_score[,2 ],test$DEFAULT)
```

Calculate AUC

```
perf_tree_val <- performance(pred_tree_val,"auc")
plot(performance(pred_tree_val, measure="lift", x.measure="rpp"), colorize=TRUE)
```

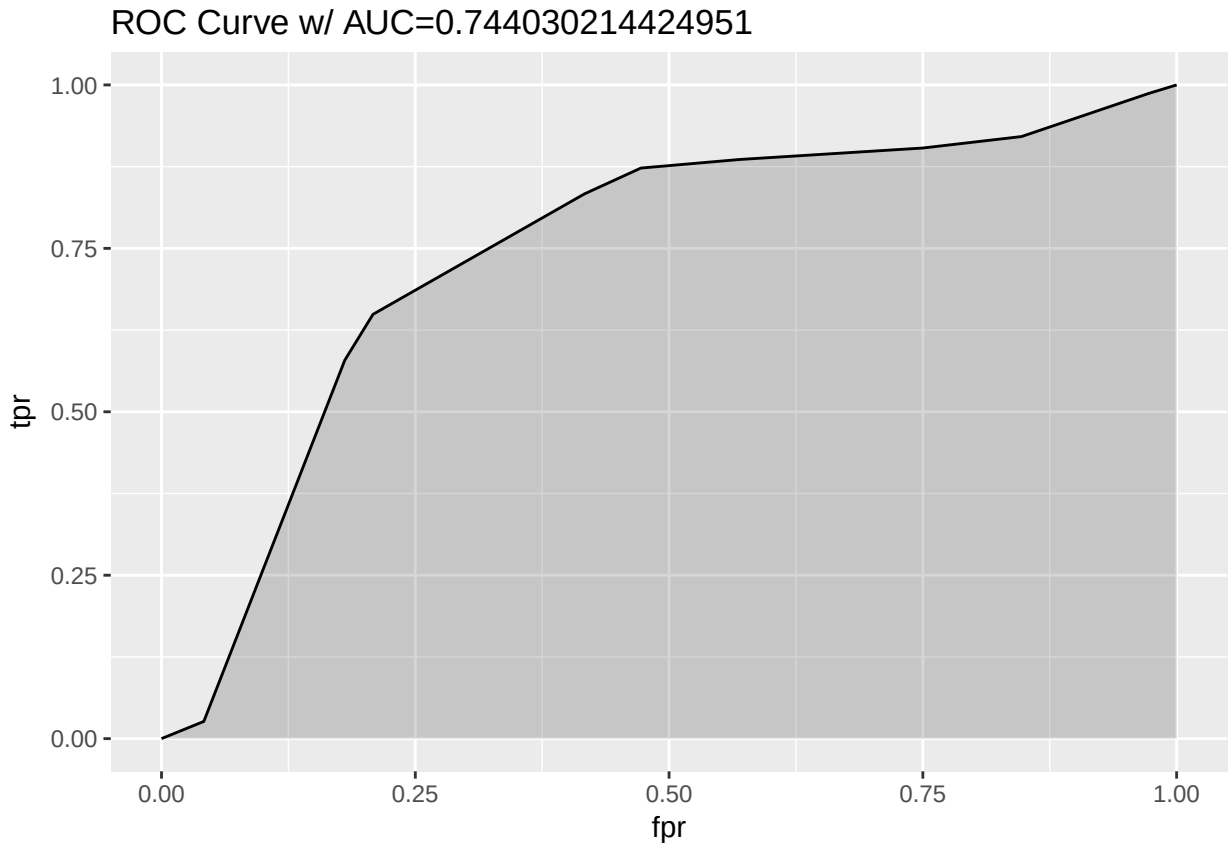


Calculate TPR, FPR

```
perf_tree_val <- performance(pred_tree_val, "tpr", "fpr")
```

ROC using ggplot

```
auc_tree <- performance(pred_tree_val, measure = "auc")
auc_tree <- auc_tree@y.values[[1]]

roc_tree <- data.frame(fpr = unlist(perf_tree_val@x.values),
                       tpr = unlist(perf_tree_val@y.values),
                       model = "rpart")
ggplot(roc_tree, aes(x = fpr, ymin = 0, ymax = tpr)) +
       geom_ribbon(alpha = 0.2) +
       geom_line(aes(y = tpr)) +
       ggtitle(paste0("ROC Curve w/ AUC=", auc_tree))
```



Calculate KS Statistic

```
ks_credit_tree <- max(attr(perf_tree_val, "y.values")[[1]] - (attr(perf_tree_val, "x.values")[[1]]))
ks_credit_tree
```

```
## [1] 0.4407895
```

Logistic Regression Model (Full)

```
full_glm <- glm(DEFAULT ~ ., family=binomial, data = train)
```

Model prediction on test data

```
full_glm_pred <- predict (full_glm, test)
pred_class <- ifelse (full_glm_pred>=0.5, 1,0)
```

Summary

```
summary(full_glm)
```

```
##
## Call:
## glm(formula = DEFAULT ~ ., family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7267  -0.6809   0.3533   0.7028   2.4495
##
## Coefficients: (3 not defined because of singularities)
##                            Estimate Std. Error z value Pr(>|z|)
## (Intercept)              -9.500e-01  1.152e+00  -0.824  0.40971
## CHK_ACCTbelow 200         4.872e-01  2.598e-01   1.875  0.06079 .
## CHK_ACCTno_acct           1.840e+00  2.768e-01   6.648 2.97e-11 ***
## CHK_ACCTover 200          1.099e+00  4.359e-01   2.521  0.01169 *
## DURATION                 -2.597e-02  1.112e-02  -2.336  0.01948 *
## HISTORYcritical_acct      1.572e+00  5.293e-01   2.970  0.00298 **
## HISTORYdelay              7.042e-01  5.685e-01   1.239  0.21543
## HISTORYexisting_paid      3.115e-01  4.553e-01   0.684  0.49385
## HISTORYno_credit          2.530e-01  7.178e-01   0.353  0.72443
## PURPOSEfurniture          1.166e+00  4.837e-01   2.411  0.01591 *
## PURPOSEnew_car            6.908e-01  4.654e-01   1.484  0.13768
## PURPOSEno_purpose         1.212e+00  5.837e-01   2.077  0.03782 *
## PURPOSEradio_tv           1.468e+00  4.798e-01   3.058  0.00223 **
## PURPOSEretraining         9.710e-01  5.504e-01   1.764  0.07767 .
## PURPOSEused_car           1.676e+00  5.829e-01   2.875  0.00404 **
## AMOUNT                   -1.449e-04  5.269e-05  -2.750  0.00596 **
## SAV_ACCT=>1000            1.084e+00  5.892e-01   1.840  0.06581 .
## SAV_ACCT100<=...<500      4.478e-01  3.483e-01   1.286  0.19857
## SAV_ACCT500<=...<1000     2.859e-01  4.694e-01   0.609  0.54248
## SAV_ACCTunk/no_acct       8.050e-01  3.120e-01   2.580  0.00987 **
## EMPLOYMENT>=7years        8.160e-02  3.586e-01   0.228  0.82000
## EMPLOYMENT1<=...<4years   7.862e-02  3.016e-01   0.261  0.79431
## EMPLOYMENT4<=...<7years   6.599e-01  3.712e-01   1.778  0.07545 .
## EMPLOYMENTunemployed     -4.462e-01  5.300e-01  -0.842  0.39986
## INSTALL_RATE2            -7.787e-01  3.883e-01  -2.005  0.04492 *
## INSTALL_RATE3            -8.501e-01  4.224e-01  -2.013  0.04416 *
## INSTALL_RATE4            -1.196e+00  3.790e-01  -3.156  0.00160 **
## STATUSmale-marr-wid       3.938e-01  5.652e-01   0.697  0.48603
## STATUSmale-single         7.812e-01  4.730e-01   1.651  0.09865 .
## STATUSunknown            -7.050e-02  4.813e-01  -0.146  0.88354
## RESIDENCY>4years         -3.947e-01  3.488e-01  -1.132  0.25783
## RESIDENCY1<..<=2years    -8.473e-01  3.555e-01  -2.384  0.01714 *
```

```
## RESIDENCY2<..<=3years   -5.148e-01  4.016e-01  -1.282  0.19989
## PRESENT_RESIDENT2               NA         NA      NA       NA
## PRESENT_RESIDENT3               NA         NA      NA       NA
## PRESENT_RESIDENT4               NA         NA      NA       NA
## DEBTORguarantor          2.205e+00  7.039e-01   3.133  0.00173 **
## DEBTORnone               8.663e-01  4.684e-01   1.849  0.06440 .
## AGE                      8.168e-03  1.111e-02   0.735  0.46229
## OTHER_INSTALL1          -6.027e-01  2.524e-01  -2.388  0.01695 *
## PROPERTYrealstate        3.609e-01  2.575e-01   1.402  0.16104
## PROPERTYunk-prop        -4.207e-01  3.054e-01  -1.377  0.16836
## NUM_CREDITS2            -2.721e-01  2.923e-01  -0.931  0.35193
## NUM_CREDITS3            -1.002e-01  7.539e-01  -0.133  0.89429
## NUM_CREDITS4             1.592e-01  1.399e+00   0.114  0.90935
## JOBskilled-emp          -2.444e-01  3.461e-01  -0.706  0.48005
## JOBunskilled-non-res     2.644e-01  7.780e-01   0.340  0.73403
## JOBunskilled-res        -5.505e-01  4.208e-01  -1.308  0.19082
## NUM_DEPENDENTS2         -3.457e-01  2.983e-01  -1.159  0.24656
## TELEPHONEyes             4.105e-01  2.448e-01   1.677  0.09363 .
## FOREIGNyes               8.168e-01  8.204e-01   0.996  0.31941
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 883.54  on 699  degrees of freedom
## Residual deviance: 636.23  on 652  degrees of freedom
## AIC: 732.23
##
## Number of Fisher Scoring iterations: 5
```

Anova to discard unsignificant predictors

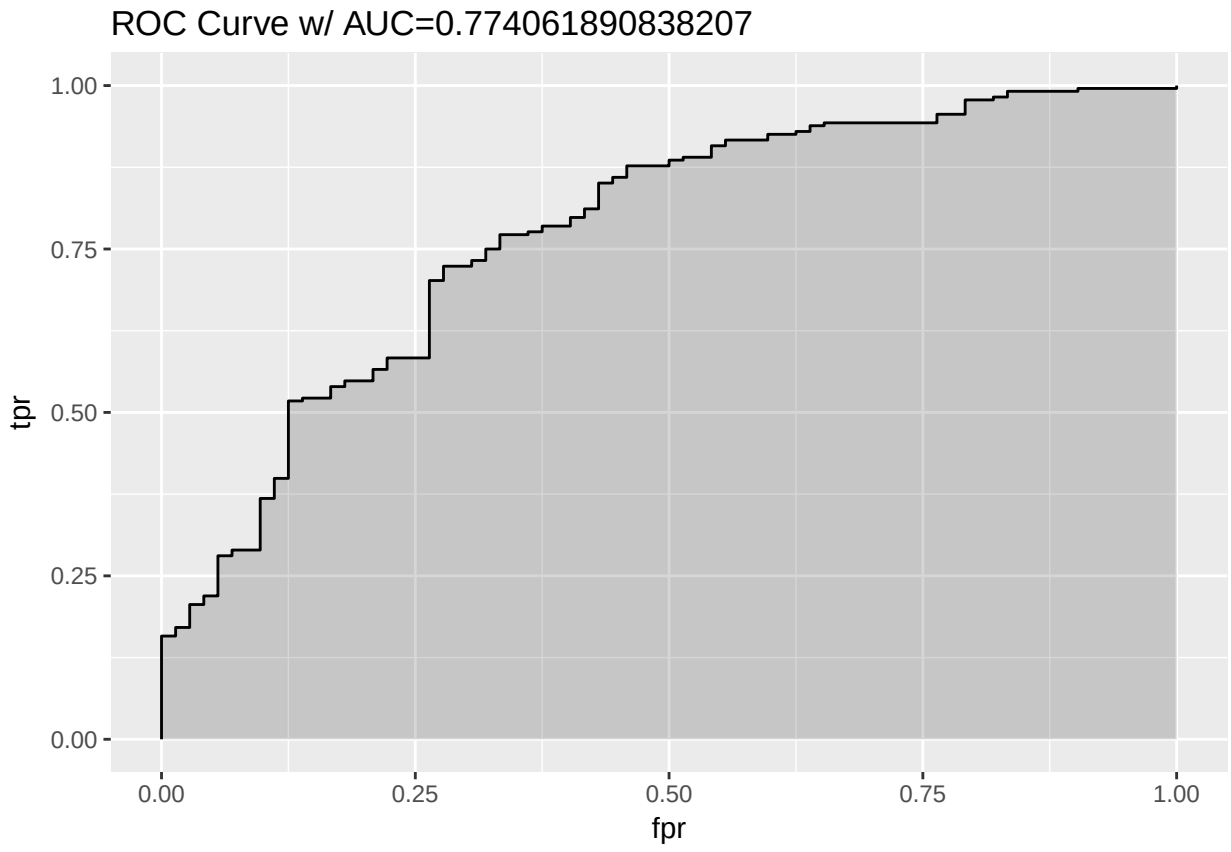
```
anova(full_glm, test = 'Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: DEFAULT
##
## Terms added sequentially (first to last)
##
##
##                 Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                699     883.54
## CHK_ACCT         3   90.779       696     792.76 < 2.2e-16 ***
## DURATION         1   33.131       695     759.63 8.615e-09 ***
## HISTORY          4   25.153       691     734.48 4.686e-05 ***
## PURPOSE          6   19.354       685     715.12  0.003606 **
## AMOUNT           1    2.843       684     712.28  0.091757 .
## SAV_ACCT         4    8.645       680     703.64  0.070624 .
## EMPLOYMENT       4    9.986       676     693.65  0.040672 *
```

```
## INSTALL_RATE      3    7.941       673     685.71  0.047242 *
## STATUS            3   10.159       670     675.55  0.017264 *
## RESIDENCY         3    6.365       667     669.19  0.095126 .
## PRESENT_RESIDENT  0    0.000       667     669.19
## DEBTOR            2   11.994       665     657.19  0.002486 **
## AGE               1    0.528       664     656.66  0.467341
## OTHER_INSTALL     1    6.624       663     650.04  0.010061 *
## PROPERTY          2    2.653       661     647.39  0.265429
## NUM_CREDITS       3    1.111       658     646.28  0.774348
## JOB               3    4.748       655     641.53  0.191226
## NUM_DEPENDENTS    1    1.547       654     639.98  0.213584
## TELEPHONE         1    2.609       653     637.37  0.106287
## FOREIGN           1    1.142       652     636.23  0.285309
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Plotting ROC using ggplot

```
pred_auc_glm<-prediction(full_glm_pred, test$DEFAULT)
pref_auc_glm<-performance(pred_auc_glm, measure = "tpr", x.measure = "fpr")
auc_glm <- performance(pred_auc_glm, measure = "auc")
auc_glm <- auc_glm@y.values[[1]]
roc_glm <- data.frame(fpr = unlist(pref_auc_glm@x.values),
                      tpr = unlist(pref_auc_glm@y.values), model="GLM")
ggplot(roc_glm, aes(x = fpr, ymin = 0, ymax = tpr)) + geom_ribbon(alpha = 0.2) +
       geom_line(aes(y = tpr)) + ggtitle(paste0("ROC Curve w/ AUC=", auc_glm))
```



Logistic Model (Optimized)

```
opt_glm <- glm(DEFAULT ~ CHK_ACCT + DURATION + HISTORY + PURPOSE + DEBTOR +
                 EMPLOYMENT + STATUS + OTHER_INSTALL + SAV_ACCT ,
                 family = binomial, data = train)
```

Predicting on test data

```
opt_glm_pred <- predict(opt_glm, test)
opt_pred_class <- ifelse(opt_glm_pred>=0.5, 1, 0)
```

Summary

```
summary(opt_glm)
```

```
##
## Call:
## glm(formula = DEFAULT ~ CHK_ACCT + DURATION + HISTORY + PURPOSE +
##     DEBTOR + EMPLOYMENT + STATUS + OTHER_INSTALL + SAV_ACCT,
##     family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7325  -0.7906   0.4071   0.7362   2.2502
##
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)
## (Intercept)              -2.213681   0.908645  -2.436 0.014841 *
## CHK_ACCTbelow 200         0.523214   0.248271   2.107 0.035080 *
## CHK_ACCTno_acct           1.722913   0.261172   6.597 4.20e-11 ***
## CHK_ACCTover 200          1.244458   0.414691   3.001 0.002692 **
## DURATION                 -0.047723   0.008615  -5.539 3.04e-08 ***
## HISTORYcritical_acct      1.412535   0.477639   2.957 0.003103 **
## HISTORYdelay              0.596991   0.534526   1.117 0.264054
## HISTORYexisting_paid      0.379318   0.440452   0.861 0.389127
## HISTORYno_credit          0.079224   0.668287   0.119 0.905633
## PURPOSEfurniture          1.209036   0.453246   2.668 0.007642 **
## PURPOSEnew_car            0.734246   0.443094   1.657 0.097502 .
## PURPOSEno_purpose         1.197657   0.563509   2.125 0.033557 *
## PURPOSEradio_tv           1.500407   0.451398   3.324 0.000888 ***
## PURPOSEretraining         1.136211   0.514139   2.210 0.027110 *
## PURPOSEused_car           1.530283   0.532630   2.873 0.004065 **
## DEBTORguarantor           2.315967   0.671613   3.448 0.000564 ***
## DEBTORnone                0.933910   0.450165   2.075 0.038024 *
## EMPLOYMENT>=7years        0.096605   0.312025   0.310 0.756861
## EMPLOYMENT1<=...<4years -0.045975   0.277123  -0.166 0.868233
## EMPLOYMENT4<=...<7years   0.681867   0.350559   1.945 0.051765 .
## EMPLOYMENTunemployed     -0.313815   0.432189  -0.726 0.467774
## STATUSmale-marr-wid       0.493044   0.528228   0.933 0.350617
## STATUSmale-single         0.556975   0.441998   1.260 0.207622
```

```
## STATUSunknown           -0.039505   0.450221  -0.088 0.930079
## OTHER_INSTALL1          -0.636581   0.241763  -2.633 0.008461 **
## SAV_ACCT=>1000           0.941229   0.539866   1.743 0.081255 .
## SAV_ACCT100<=...<500     0.324497   0.324882   0.999 0.317884
## SAV_ACCT500<=...<1000    0.429363   0.459830   0.934 0.350436
## SAV_ACCTunk/no_acct      0.736659   0.295345   2.494 0.012623 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 883.54  on 699  degrees of freedom
## Residual deviance: 668.87  on 671  degrees of freedom
## AIC: 726.87
##
## Number of Fisher Scoring iterations: 5
```

Anova on test data

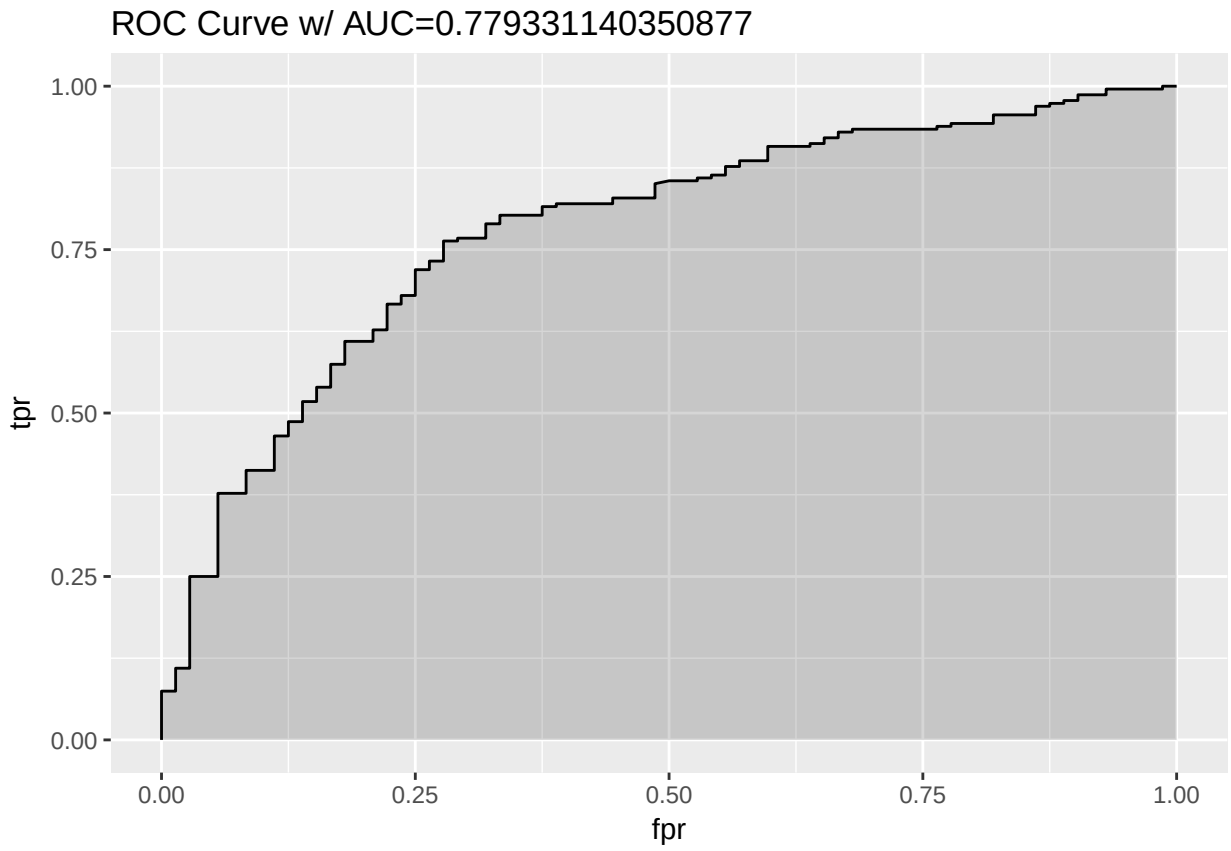
```
anova(opt_glm, test = 'Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: DEFAULT
##
## Terms added sequentially (first to last)
##
##
##               Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                            699     883.54
## CHK_ACCT       3   90.779       696     792.76 < 2.2e-16 ***
## DURATION       1   33.131       695     759.63 8.615e-09 ***
## HISTORY        4   25.153       691     734.48 4.686e-05 ***
## PURPOSE        6   19.354       685     715.12  0.003606 **
## DEBTOR         2   13.170       683     701.95  0.001381 **
## EMPLOYMENT     4    9.528       679     692.43  0.049172 *
## STATUS         3    7.209       676     685.22  0.065526 .
## OTHER_INSTALL  1    7.146       675     678.07  0.007514 **
## SAV_ACCT       4    9.201       671     668.87  0.056270 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ROC with ggplot

```
opt_auc_pred<-prediction(opt_glm_pred, test$DEFAULT)
opt_auc_pref<-performance(opt_auc_pred, measure = "tpr", x.measure = "fpr")
auc_opt_glm <- performance(opt_auc_pred, measure = "auc")
auc_opt_glm <- auc_opt_glm@y.values[[1]]
roc_opt_glm <- data.frame(fpr = unlist(opt_auc_pref@x.values),
```

```
                      tpr = unlist(opt_auc_pref@y.values), model="GLM")
ggplot(roc_opt_glm, aes(x = fpr, ymin = 0, ymax = tpr)) + geom_ribbon(alpha=0.2) +
       geom_line(aes(y = tpr)) + ggtitle(paste0("ROC Curve w/ AUC=", auc_opt_glm))
```



Compare Regression Models

```
anova(full_glm, opt_glm, test = 'Chisq')
```

```
## Analysis of Deviance Table
##
## Model 1: DEFAULT ~ CHK_ACCT + DURATION + HISTORY + PURPOSE + AMOUNT +
##     SAV_ACCT + EMPLOYMENT + INSTALL_RATE + STATUS + RESIDENCY +
##     PRESENT_RESIDENT + DEBTOR + AGE + OTHER_INSTALL + PROPERTY +
##     NUM_CREDITS + JOB + NUM_DEPENDENTS + TELEPHONE + FOREIGN
## Model 2: DEFAULT ~ CHK_ACCT + DURATION + HISTORY + PURPOSE + DEBTOR +
##     EMPLOYMENT + STATUS + OTHER_INSTALL + SAV_ACCT
##   Resid. Df Resid. Dev  Df Deviance Pr(>Chi)
## 1       652     636.23
## 2       671     668.87 -19   -32.64  0.02644 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Treating Imbalance

Create different training sets

```
ctrl <- trainControl(method = "repeatedcv", repeats = 3, classProbs = TRUE,
                     summaryFunction = twoClassSummary, savePredictions = TRUE)
```

First set

```
set.seed(100)
orig_fit <- train(DEFAULT ~ .,data = train, method = "glm", family = binomial,
                  tuneLength = 5, metric = "ROC", trControl = ctrl)
```

Extract ROC value

```
test_roc <- function(model, data) {

  roc(data$DEFAULT,
      predict(model, data, type = "prob")[, "NonDefault"])
}

orig_fit %>%
  test_roc(data = test) %>%
  auc()
```

```
## Area under the curve: 0.7741
```

Create model weights

```
model_weights <- ifelse(train$DEFAULT == "NonDefault",
                        (1/table(train$DEFAULT)[1]) * 0.5,
                        (1/table(train$DEFAULT)[2]) * 0.5)
```

Ensure all models use same set.seed()

```
ctrl$seeds <- orig_fit$control$seeds
```

Build a model with weights

```
weighted_fit <- train(DEFAULT ~ .,data = train, method = "glm",
                      family = binomial, tuneLength = 5, metric = "ROC",
                      trControl = ctrl, weights = model_weights)
```

Build a down-sampled model

```
ctrl$sampling <- "down"
down_fit <- train(DEFAULT ~ .,data = train, method = "glm",
                  family = binomial, tuneLength = 5, metric = "ROC",
                  trControl = ctrl)
```

Build up-sampled model

```
ctrl$sampling <- "up"
up_fit <- train(DEFAULT ~ .,data = train, method = "glm",
                family = binomial, tuneLength = 5, metric = "ROC",
                trControl = ctrl)
```

Build a SMOTE model

```
ctrl$sampling <- "smote"
smote_fit <- train(DEFAULT ~ .,data = train, method = "glm",
                   family = binomial, tuneLength = 5, metric = "ROC",
                   trControl = ctrl)
```

Examine results

```
model_list <- list(original = orig_fit,
                   weighted = weighted_fit,
                   down = down_fit,
                   up = up_fit,
                   SMOTE = smote_fit)
```

Extract AUC values

```
#extract AUC values
model_list_roc <- model_list %>%
  map(test_roc, data = test)

model_list_roc %>%
  map("auc")
```

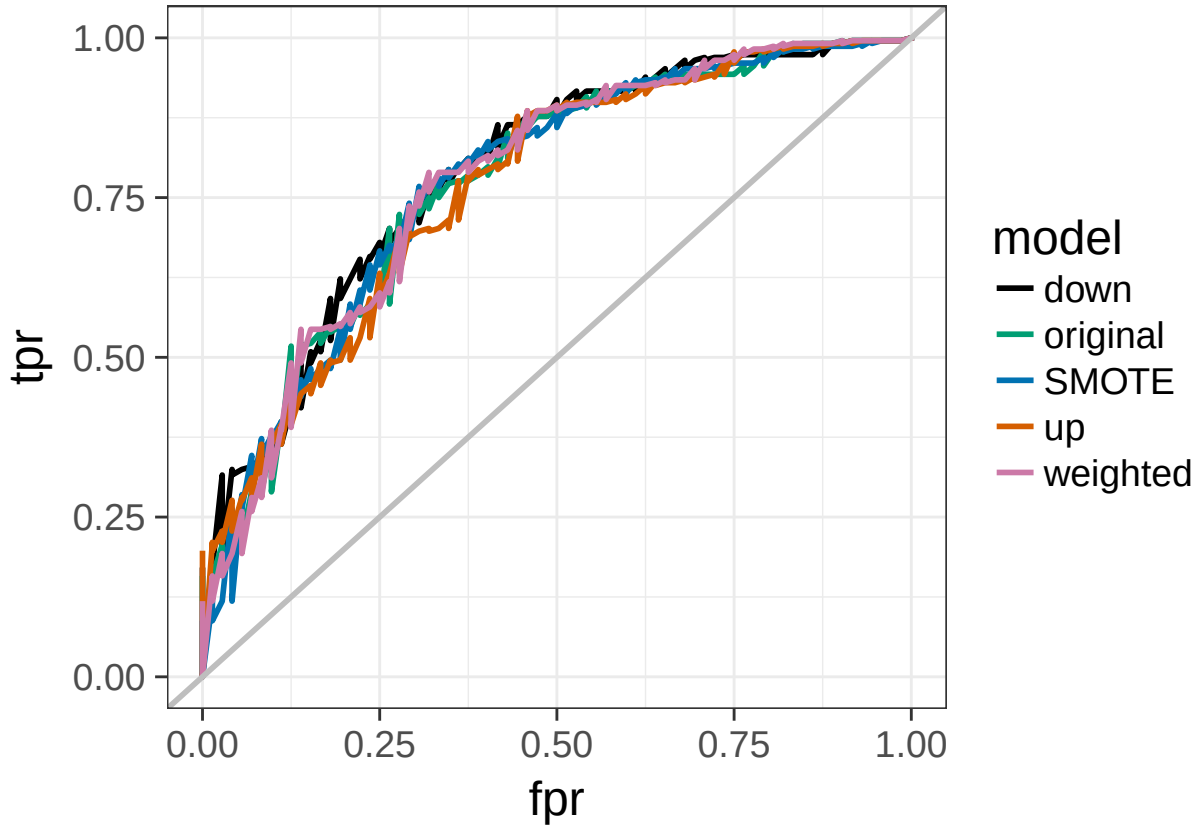
```
## $original
## Area under the curve: 0.7741
##
## $weighted
## Area under the curve: 0.7767
##
## $down
## Area under the curve: 0.7885
##
## $up
```

```
## Area under the curve: 0.7668
##
## $SMOTE
## Area under the curve: 0.7739
```

```
results_list_roc <- list(NA)
num_mod <- 1

for(the_roc in model_list_roc){

  results_list_roc[[num_mod]] <-
    data_frame(tpr = the_roc$sensitivities,
               fpr = 1 - the_roc$specificities,
               model = names(model_list)[num_mod])

  num_mod <- num_mod + 1
}
```

Plot ROC curve for five models

```
results_df_roc <- bind_rows(results_list_roc)

custom_col <- c("#000000", "#009E73", "#0072B2", "#D55E00", "#CC79A7")

ggplot(aes(x = fpr,  y = tpr, group = model), data = results_df_roc) +
  geom_line(aes(color = model), size = 1) +
  scale_color_manual(values = custom_col) +
  geom_abline(intercept = 0, slope = 1, color = "gray", size = 1) +
  theme_bw(base_size = 18)
```



List environment and package info

```
getwd()
```

```
## [1] "C:/Users/Dell/OneDrive/Documents"
```

```
sessionInfo()
```

```
## R version 3.4.0 (2017-04-21)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 15063)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] purrr_0.2.3        DMwR_0.4.1         caret_6.0-76
##  [4] lattice_0.20-35    devtools_1.13.3    markdown_0.8
```

```
##  [7] knitr_1.17         rattle_4.1.0       RColorBrewer_1.1-2
## [10] rpart.plot_2.1.2   rpart_4.1-11       gmodels_2.16.2
## [13] e1071_1.6-8        dplyr_0.7.2        plyr_1.8.4
## [16] magrittr_1.5       ROCR_1.0-7         gplots_3.0.1
## [19] pROC_1.10.0        plotly_4.7.1       ggplot2_2.2.1
##
## loaded via a namespace (and not attached):
##  [1] httr_1.3.0         tidyr_0.7.0        jsonlite_1.5
##  [4] viridisLite_0.2.0  splines_3.4.0      foreach_1.4.3
##  [7] gtools_3.5.0       assertthat_0.2.0   TTR_0.23-2
## [10] stats4_3.4.0       yaml_2.1.14        backports_1.1.0
## [13] quantreg_5.33      glue_1.1.1         digest_0.6.12
## [16] minqa_1.2.4        colorspace_1.3-2   htmltools_0.3.6
## [19] Matrix_1.2-9       pkgconfig_2.0.1    SparseM_1.77
## [22] scales_0.4.1       gdata_2.18.0       MatrixModels_0.4-1
## [25] lme4_1.1-13        tibble_1.3.3       mgcv_1.8-17
## [28] car_2.1-5          withr_2.0.0        nnet_7.3-12
## [31] lazyeval_0.2.0     quantmod_0.4-10    pbkrtest_0.4-7
## [34] memoise_1.1.0      evaluate_0.10.1    nlme_3.1-131
## [37] MASS_7.3-47        xts_0.10-0         class_7.3-14
## [40] tools_3.4.0        data.table_1.10.4  stringr_1.2.0
## [43] munsell_0.4.3      bindrcpp_0.2       compiler_3.4.0
## [46] caTools_1.17.1     rlang_0.1.2        nloptr_1.0.4
## [49] iterators_1.0.8    RGtk2_2.20.33      htmlwidgets_0.9
## [52] labeling_0.3       bitops_1.0-6       rmarkdown_1.6
## [55] gtable_0.2.0       ModelMetrics_1.1.0 codetools_0.2-15
## [58] curl_2.8.1         abind_1.4-5        reshape2_1.4.2
## [61] R6_2.2.2           zoo_1.8-0          bindr_0.1
## [64] rprojroot_1.2      KernSmooth_2.23-15 stringi_1.1.5
## [67] parallel_3.4.0     Rcpp_0.12.12
```