

Sadržaj

Uvod.....	2
Učitavanje i analiza podataka.....	2
Učitavanje podataka iz fajla app.csv	2
Duplikati	3
NA i NULL vrednosti	4
Eksplorativna analiza.....	4
Učitavanje podataka iz fajla cb.csv	8
Duplikati	8
NA i NULL vrednosti	8
Eksplorativna analiza.....	8
Spajanje podataka.....	9
Deljenje uzorka po godinama	11
Logistička regresija – razvojni uzorak.....	11
Prag za logističku regresiju	13
Logistička regresija – test uzorak	14
Random forest – razvojni podaci	15
Random forest – test podaci	16
Deljenje uzorka po nausmičnom odabiru podataka	16

Uvod

Ovaj dokument opisuje proces razvoja modela za verovatnoću default-a.

Raspolagali smo sa podacima iz 2011. i 2012. godine.

Najpre smo odstranili duple vrednosti, zamenili NA vrednosti i odstranili outlier-e gde je imalo smisla.

Razvijali smo dva modela: logističku regresiju i random forest. Podelili smo uzorak na razvojni i test prema dva kriterijuma: godina i slučajni odabir podataka u odnosu 65%-35%. Za svaki kriterijum smo testirali oba modela. Rezultat se nije značajno razlikovao na out of sample i out of time kod logističke regresije, dok kod random forest-a jeste.

Random forest je dao veliku tačnost, 99% na razvojnem i 97% na test uzorku, međutim ispostavilo se da je greška 1. vrste na test uzorku jednaka 1. Model bi dozvolio da banka daje kredit klijentima koji bi imali default. Stoga smo odbacili ovaj model.

Logistička regresija sa pragom 0.02 je dala skoro iste rezultate za razvojni i test uzorak. Tačnost logističke regresije je 64%, što možda ne deluje mnogo, ali logistička regresija sa tačnošću od 77% detektuje klijente koji će imati default. U ovom slučaju bolje je da banka odbije dobar kredit, nego da odobri mnogo loših.

Međutim kako True Negative vrednosti ima 63%, možda možemo da se zapitamo da li je ovo ipak dobar model, jer ne želimo da odbijemo 37% klijenata koji bi zaista vratili kredit. Stoga, bilo bi poželjno ispitati i druge modele.

Pratni dokumenti su kodovi sa komentarima, odnosno R fajlovi : `DataScience_BSANDO.R` i `DataScienceRandomSampleSplit_BSANDO.R`.

Kod koji se vidi u ovom dokumentu je kod iz fajla `DataScience_BSANDO.R`, tu je uzorak podeljen prema godinama.

U fajlu `DataScienceRandomSampleSplit_BSANDO.R` je rađen isti postupak samo je uzorak podeljen slučajnim odabirom.

Učitavanje i analiza podataka

Podaci su učitani iz dva izvora, `app.csv` i `cb.csv` (opis podataka je u sekciji 2.3 u dokumentu `zadaci.pdf`).

Učitavanje podataka iz fajla `app.csv`

Prvo smo učitali podatke iz fajla `app.csv`.

```
app<-read_csv("app.csv",col_types = cols (col_integer(), col_integer(), col_date(), col_double(), col_integer(),
col_double(),col_date(), col_character(), col_character(), col_integer(),col_logical(), col_character(),
col_character(), col_double()))
```

Pomoću funkcije *glimpse*, vidimo strukturu podataka i da je broj redova 10436.

1

0

1

1

1

Vidimo zaista da za svaki *id*, *income* ima vrednost 1 i drugu više logičniju vrednost, te ćemo izbaciti vrednosti za ove *id*-eve gde je *income* jednak 1.

```
app<-filter(app, (id %in% dupl_id & income !=1) | !(id %in% dupl_id))
(dupl_id<-app$id[duplicated(app$id)])
```

NA i NULL vrednosti

U ovoj sekciji proverićemo koliko ima *missing values*.

```
(na_count <- sapply(app, function(x) sum(is.na(x))))
  id    bad    date    amount installments    interest    birthdate    sex    family
  0      0      0      0      0      0      0      0      0
children    rent    education    employment    income
  93      0      0      0      0
(null_count <- sapply(app, function(x) sum(is.null(x))))
  id    bad    date    amount installments    interest    birthdate    sex    family
  0      0      0      0      0      0      0      0      0
children    rent    education    employment    income
  0      0      0      0      0
```

Kolona *children* ima mali broj *na* vrednosti. Proverićemo koliko među njima ima klijenata koji su kasnili.

```
table(filter(app, is.na(children))$bad)
0 1
91 2
```

Kako ih nema mnogo, možemo te vrednosti da zamenimo sa nulom.

Eksplorativna analiza

U ovoj sekciji posmatraćemo raspodele promenljivih, kao i njihov odnos sa brojem default-a.

```
table(app$sex)
female  male
 4937  5063
aggregate(bad ~ sex, data = app, mean)
  sex    bad
1 female 0.02167308
2  male 0.02923168
app<-mutate(app, sexMale = as.factor(ifelse(sex == 'male', 1, 0)))
```

Vidimo da je udeo žena i muškaraca sličan, takođe srednja vrednost za default je slična.

Uveli smo novu promenljivu *sexMale* koja ima vrednost 1 ako je klijent musko, u suprotnom 0.

Kada posmatramo promenljivu *family*, možemo da uočimo određene pravilnosti.

```
>aggregate(bad ~ family, data = app, mean)
  family    bad
1 divorced 0.04086957
2  married 0.02075556
3   single 0.03271538
4  widowed 0.02821317
```

Razvedeni i samci imaju veću stopu default-a. Stoga, možemo da transformišemo ovu promenljivu tako da dobija vrednost 1 kada je klijent razveden ili samac, a u suprotnom 0.

```
app<-mutate(app, divorced_single = as.factor(ifelse(family == 'divorced' | family == 'single',1,0)))
```

Klijenti koji iznajmljuju nekretninu u kojoj žive imaju veću stopu default-a.

```
aggregate(bad ~ rent, data = app, mean)
  rent    bad
1 FALSE 0.01891313
2  TRUE 0.05074915
```

Transformisaćemo ovu promenljivu tako da dobija vrednost 1 kada klijent iznajmljuje nekretninu, a 0 u suprotnom slučaju.

```
app$rent_1 <-as.factor(ifelse(app$rent, 1, 0))
```

Kada posmatramo promenljivu koja pokazuje vrstu zaposlenja, vidimo da klijenti koji imaju sopstveni biznis imaju stopu default-a.

```
aggregate(bad ~ employment, data = app, mean)
  employment 0.02678324
  retired    0.01340616
  self-employed 0.04487179
```

Stoga, transformisaćemo *employment* tako da dobija vrednost 1 kada klijent ima sopstveni biznis, a 0 u suprotnom.

```
app<-mutate(app, self_empl = as.factor(ifelse(employment == 'self-employed',1,0)))
```

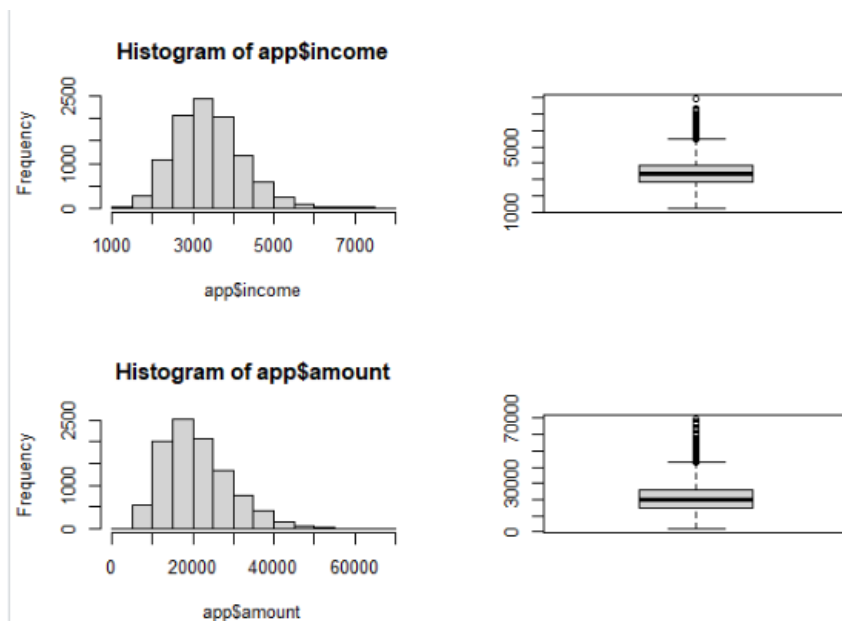
Kada posmatramo obrazovanje u odnosu na stopu default-a, ne možemo da uočimo neku jasnu vezu, te verovatno ova promenljiva neće igrati ulogu u modelu.

```
> aggregate(bad ~ education, data = app, mean)
  education    bad
1         1 0.01570681
2         2 0.02131742
3         3 0.03027806
4         4 0.01901141
app$education <-as.factor(app$education)
```

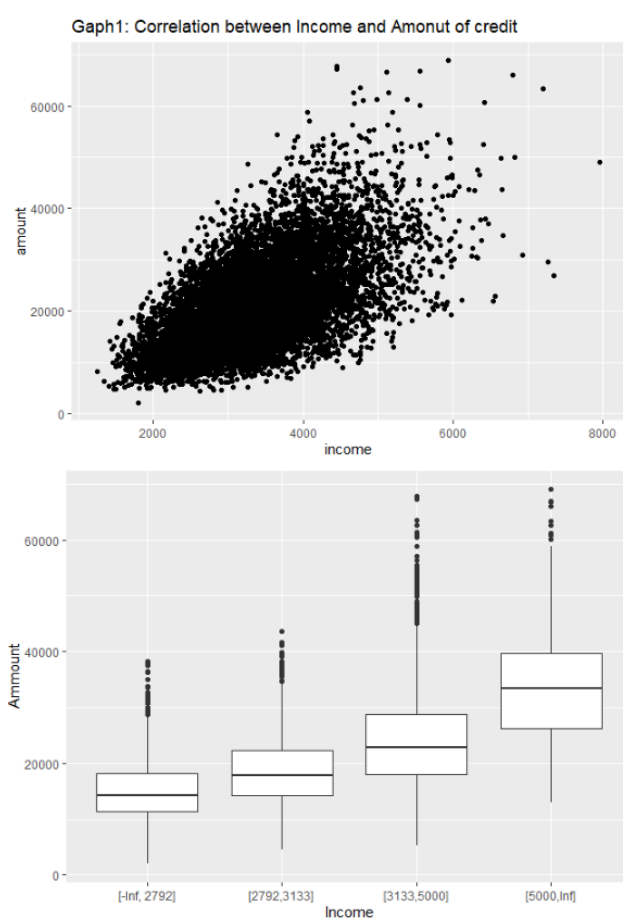
Posmatrajmo sada promenljive *income* i *amount*.

```
par(mfrow = c(2, 2))
hist(app$income)
boxplot(app$income)
hist(app$amount)
boxplot(app$amount)
par(mfrow = c(1, 1))
```

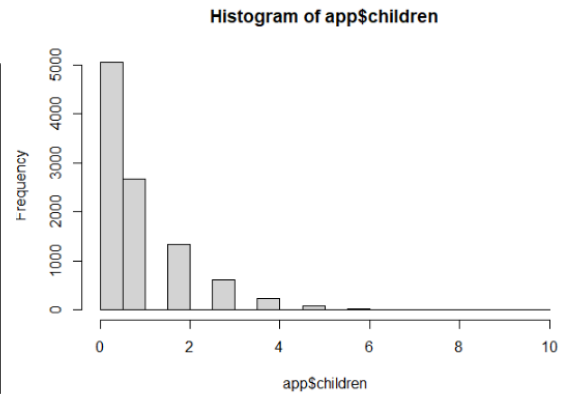
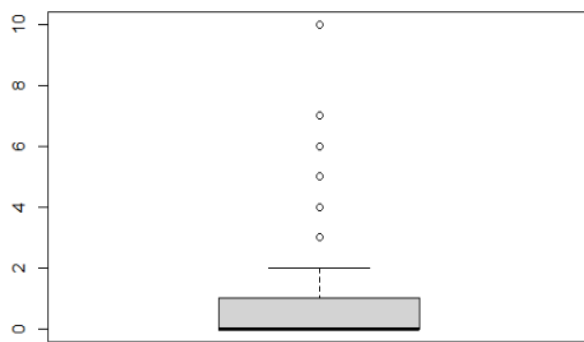
Vidimo da obe promenljive imaju smislene raspodele. Imaju i outlier-e, ali ne previše nelogične vrednosti.



Vrlo je verovatno da su ove dve promenljive povezane. Pogledajmo naredna dva graifka. Možemo primetiti korelaciju izmedju ove dve promenljive, odnosno da veća primanja dovode do većeg iznosa kredita. Outlier-i nisu toliko nerazumni, npr klijenti sa primanjima u opsegu [3133,5000] podgli su kredit od oko 60 000.



Kad posmatramo broj dece vidimo da je zabeleženo da nekoliko klijenata ima preko sedmoro dece.



```
aggregate(bad ~ children, data = app, mean)
```

children	bad
1	0 0.01838671
2	1 0.02436282
3	2 0.04282494
4	3 0.03618421
5	4 0.05752212
6	5 0.05263158
7	6 0.03846154
8	7 0.00000000
9	10 0.00000000

```
table(app$children)
```

0	1	2	3	4	5	6	7	10
5058	2668	1331	608	226	76	26	6	1

Vidimo da veći broj dece indikuje veću stopu default-a, osim kada klijent ima preko sedmoro dece, što je motiv da pomislimo da ti podaci nisu tačni.

Kako je i boxplot označio ove vrednosti kao outlier-e, sklonićemo ih iz uzorka.

```
app1<-filter(app, children <=6)
```

Broj rata izlaza da je igra veliku ulogu u proceni stope default-a.

installments	bad
1	36 0.02689487
2	48 0.02574686
3	60 0.02366381

Proverili smo, za svaki slučaj da li podaci zaista spadaju u dati vremenski okvir, tj da su podaci iz 2011. i 2012 godine.

```
summary(app1$date)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
"2011-01-01"	"2011-06-29"	"2011-12-31"	"2011-12-30"	"2012-06-30"	"2012-12-31"

Proverili smo da li godina rođenja ima smislene podatke.

```
summary(app1$birthdate)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
"1941-04-19"	"1954-11-29"	"1968-05-19"	"1967-03-08"	"1979-01-13"	"1992-11-23"

Uveli smo novu promenljivu *age* – broj godina klijenta u trenutku apliciranja za kredit.

```
app2<-mutate(app1, age = as.double(difftime(date,birthdate, units="days")/365))
hist(app2$age)
```

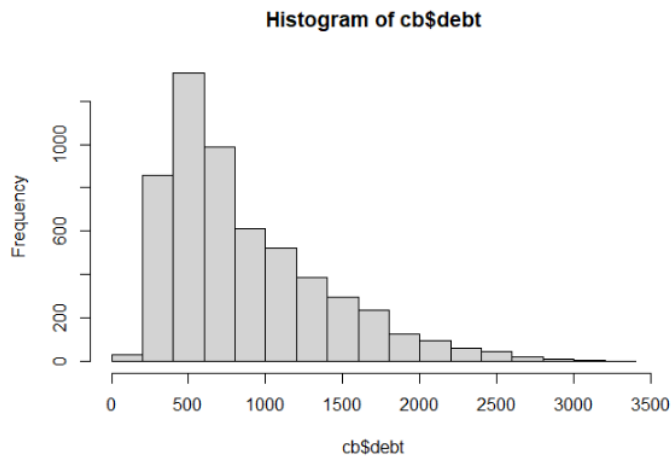


Deluje da su u približnom broju prisutne starosne grupe od 20-70 godina.

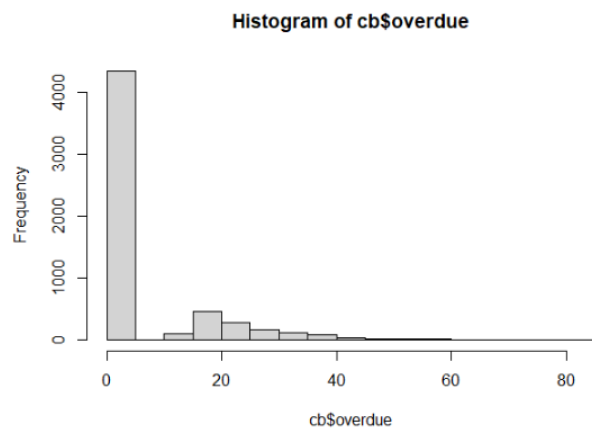
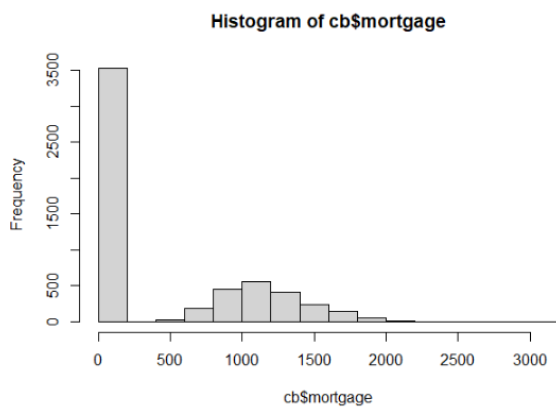
Učitavanje podataka iz fajla cb.csv

U ovoj sekciji učit ćemo podatke za druge kreditne obaveze klijenta i analiziraćemo te podatke.

```
cb<-read_csv("cb.csv",col_types = cols (col_integer(), col_double(), col_double(), col_integer()))
glimpse(cb)
#> # A tibble: 5,616 x 4
#>   id      debt      mortgage overdue
#>   <int> <dbl> <dbl> <int>
#> 1     1  541.41    0.00      0
#> 2     3  572.65    0.00     15
#> 3     5 2075.42    0.00      0
#> 4     7  644.91    0.00      0
#> 5     8  582.10    0.00      0
#> 6    12  569.20    0.00      0
#> 7    14  526.37    0.00      0
#> 8    15  853.69    0.00      0
#> 9    16  941.30    0.00      0
#> 10   19  970.41    0.00      0
#> 11   22  1353.23    0.00     83
#> 12   23    0.00    0.00      0
#> 13   24    0.00    0.00      0
#> 14   26    0.00    0.00      0
#> 15   27    0.00    0.00      0
#> 16   28    0.00    0.00      0
#> 17   29    0.00    0.00      0
#> 18   30    0.00    0.00      0
#> 19   31    0.00    0.00      0
#> 20   32    0.00    0.00      0
#> 21   35    0.00    0.00      0
#> 22   36    0.00    0.00      0
#> 23   37    0.00    0.00      0
#> 24   38    0.00    0.00      0
#> 25   39    0.00    0.00      0
#> 26   40    0.00    0.00      0
#> 27   41    0.00    0.00      0
#> 28   42    0.00    0.00      0
#> 29   43    0.00    0.00      0
#> 30   44    0.00    0.00      0
#> 31   45    0.00    0.00      0
#> 32   46    0.00    0.00      0
#> 33   47    0.00    0.00      0
#> 34   48    0.00    0.00      0
#> 35   49    0.00    0.00      0
#> 36   50    0.00    0.00      0
#> 37   51    0.00    0.00      0
#> 38   52    0.00    0.00      0
#> 39   53    0.00    0.00      0
#> 40   54    0.00    0.00      0
#> 41   55    0.00    0.00      0
#> 42   56    0.00    0.00      0
#> 43   57    0.00    0.00      0
#> 44   58    0.00    0.00      0
#> 45   59    0.00    0.00      0
#> 46   60    0.00    0.00      0
#> 47   61    0.00    0.00      0
#> 48   62    0.00    0.00      0
#> 49   63    0.00    0.00      0
#> 50   64    0.00    0.00      0
#> 51   65    0.00    0.00      0
#> 52   66    0.00    0.00      0
#> 53   67    0.00    0.00      0
#> 54   68    0.00    0.00      0
#> 55   69    0.00    0.00      0
#> 56   70    0.00    0.00      0
#> 57   71    0.00    0.00      0
#> 58   72    0.00    0.00      0
#> 59   73    0.00    0.00      0
#> 60   74    0.00    0.00      0
#> 61   75    0.00    0.00      0
#> 62   76    0.00    0.00      0
#> 63   77    0.00    0.00      0
#> 64   78    0.00    0.00      0
#> 65   79    0.00    0.00      0
#> 66   80    0.00    0.00      0
#> 67   81    0.00    0.00      0
#> 68   82    0.00    0.00      0
#> 69   83    0.00    0.00      0
#> 70   84    0.00    0.00      0
#> 71   85    0.00    0.00      0
#> 72   86    0.00    0.00      0
#> 73   87    0.00    0.00      0
#> 74   88    0.00    0.00      0
#> 75   89    0.00    0.00      0
#> 76   90    0.00    0.00      0
#> 77   91    0.00    0.00      0
#> 78   92    0.00    0.00      0
#> 79   93    0.00    0.00      0
#> 80   94    0.00    0.00      0
#> 81   95    0.00    0.00      0
#> 82   96    0.00    0.00      0
#> 83   97    0.00    0.00      0
#> 84   98    0.00    0.00      0
#> 85   99    0.00    0.00      0
#> 86  100    0.00    0.00      0
#> 87  101    0.00    0.00      0
#> 88  102    0.00    0.00      0
#> 89  103    0.00    0.00      0
#> 90  104    0.00    0.00      0
#> 91  105    0.00    0.00      0
#> 92  106    0.00    0.00      0
#> 93  107    0.00    0.00      0
#> 94  108    0.00    0.00      0
#> 95  109    0.00    0.00      0
#> 96  110    0.00    0.00      0
#> 97  111    0.00    0.00      0
#> 98  112    0.00    0.00      0
#> 99  113    0.00    0.00      0
#> 100 114    0.00    0.00      0
#> 101 115    0.00    0.00      0
#> 102 116    0.00    0.00      0
#> 103 117    0.00    0.00      0
#> 104 118    0.00    0.00      0
#> 105 119    0.00    0.00      0
#> 106 120    0.00    0.00      0
#> 107 121    0.00    0.00      0
#> 108 122    0.00    0.00      0
#> 109 123    0.00    0.00      0
#> 110 124    0.00    0.00      0
#> 111 125    0.00    0.00      0
#> 112 126    0.00    0.00      0
#> 113 127    0.00    0.00      0
#> 114 128    0.00    0.00      0
#> 115 129    0.00    0.00      0
#> 116 130    0.00    0.00      0
#> 117 131    0.00    0.00      0
#> 118 132    0.00    0.00      0
#> 119 133    0.00    0.00      0
#> 120 134    0.00    0.00      0
#> 121 135    0.00    0.00      0
#> 122 136    0.00    0.00      0
#> 123 137    0.00    0.00      0
#> 124 138    0.00    0.00      0
#> 125 139    0.00    0.00      0
#> 126 140    0.00    0.00      0
#> 127 141    0.00    0.00      0
#> 128 142    0.00    0.00      0
#> 129 143    0.00    0.00      0
#> 130 144    0.00    0.00      0
#> 131 145    0.00    0.00      0
#> 132 146    0.00    0.00      0
#> 133 147    0.00    0.00      0
#> 134 148    0.00    0.00      0
#> 135 149    0.00    0.00      0
#> 136 150    0.00    0.00      0
#> 137 151    0.00    0.00      0
#> 138 152    0.00    0.00      0
#> 139 153    0.00    0.00      0
#> 140 154    0.00    0.00      0
#> 141 155    0.00    0.00      0
#> 142 156    0.00    0.00      0
#> 143 157    0.00    0.00      0
#> 144 158    0.00    0.00      0
#> 145 159    0.00    0.00      0
#> 146 160    0.00    0.00      0
#> 147 161    0.00    0.00      0
#> 148 162    0.00    0.00      0
#> 149 163    0.00    0.00      0
#> 150 164    0.00    0.00      0
#> 151 165    0.00    0.00      0
#> 152 166    0.00    0.00      0
#> 153 167    0.00    0.00      0
#> 154 168    0.00    0.00      0
#> 155 169    0.00    0.00      0
#> 156 170    0.00    0.00      0
#> 157 171    0.00    0.00      0
#> 158 172    0.00    0.00      0
#> 159 173    0.00    0.00      0
#> 160 174    0.00    0.00      0
#> 161 175    0.00    0.00      0
#> 162 176    0.00    0.00      0
#> 163 177    0.00    0.00      0
#> 164 178    0.00    0.00      0
#> 165 179    0.00    0.00      0
#> 166 180    0.00    0.00      0
#> 167 181    0.00    0.00      0
#> 168 182    0.00    0.00      0
#> 169 183    0.00    0.00      0
#> 170 184    0.00    0.00      0
#> 171 185    0.00    0.00      0
#> 172 186    0.00    0.00      0
#> 173 187    0.00    0.00      0
#> 174 188    0.00    0.00      0
#> 175 189    0.00    0.00      0
#> 176 190    0.00    0.00      0
#> 177 191    0.00    0.00      0
#> 178 192    0.00    0.00      0
#> 179 193    0.00    0.00      0
#> 180 194    0.00    0.00      0
#> 181 195    0.00    0.00      0
#> 182 196    0.00    0.00      0
#> 183 197    0.00    0.00      0
#> 184 198    0.00    0.00      0
#> 185 199    0.00    0.00      0
#> 186 200    0.00    0.00      0
#> 187 201    0.00    0.00      0
#> 188 202    0.00    0.00      0
#> 189 203    0.00    0.00      0
#> 190 204    0.00    0.00      0
#> 191 205    0.00    0.00      0
#> 192 206    0.00    0.00      0
#> 193 207    0.00    0.00      0
#> 194 208    0.00    0.00      0
#> 195 209    0.00    0.00      0
#> 196 210    0.00    0.00      0
#> 197 211    0.00    0.00      0
#> 198 212    0.00    0.00      0
#> 199 213    0.00    0.00      0
#> 200 214    0.00    0.00      0
#> 201 215    0.00    0.00      0
#> 202 216    0.00    0.00      0
#> 203 217    0.00    0.00      0
#> 204 218    0.00    0.00      0
#> 205 219    0.00    0.00      0
#> 206 220    0.00    0.00      0
#> 207 221    0.00    0.00      0
#> 208 222    0.00    0.00      0
#> 209 223    0.00    0.00      0
#> 210 224    0.00    0.00      0
#> 211 225    0.00    0.00      0
#> 212 226    0.00    0.00      0
#> 213 227    0.00    0.00      0
#> 214 228    0.00    0.00      0
#> 215 229    0.00    0.00      0
#> 216 230    0.00    0.00      0
#> 217 231    0.00    0.00      0
#> 218 232    0.00    0.00      0
#> 219 233    0.00    0.00      0
#> 220 234    0.00    0.00      0
#> 221 235    0.00    0.00      0
#> 222 236    0.00    0.00      0
#> 223 237    0.00    0.00      0
#> 224 238    0.00    0.00      0
#> 225 239    0.00    0.00      0
#> 226 240    0.00    0.00      0
#> 227 241    0.00    0.00      0
#> 228 242    0.00    0.00      0
#> 229 243    0.00    0.00      0
#> 230 244    0.00    0.00      0
#> 231 245    0.00    0.00      0
#> 232 246    0.00    0.00      0
#> 233 247    0.00    0.00      0
#> 234 248    0.00    0.00      0
#> 235 249    0.00    0.00      0
#> 236 250    0.00    0.00      0
#> 237 251    0.00    0.00      0
#> 238 252    0.00    0.00      0
#> 239 253    0.00    0.00      0
#> 240 254    0.00    0.00      0
#> 241 255    0.00    0.00      0
#> 242 256    0.00    0.00      0
#> 243 257    0.00    0.00      0
#> 244 258    0.00    0.00      0
#> 245 259    0.00    0.00      0
#> 246 260    0.00    0.00      0
#> 247 261    0.00    0.00      0
#> 248 262    0.00    0.00      0
#> 249 263    0.00    0.00      0
#> 250 264    0.00    0.00      0
#> 251 265    0.00    0.00      0
#> 252 266    0.00    0.00      0
#> 253 267    0.00    0.00      0
#> 254 268    0.00    0.00      0
#> 255 269    0.00    0.00      0
#> 256 270    0.00    0.00      0
#> 257 271    0.00    0.00      0
#> 258 272    0.00    0.00      0
#> 259 273    0.00    0.00      0
#> 260 274    0.00    0.00      0
#> 261 275    0.00    0.00      0
#> 262 276    0.00    0.00      0
#> 263 277    0.00    0.00      0
#> 264 278    0.00    0.00      0
#> 265 279    0.00    0.00      0
#> 266 280    0.00    0.00      0
#> 267 281    0.00    0.00      0
#> 268 282    0.00    0.00      0
#> 269 283    0.00    0.00      0
#> 270 284    0.00    0.00      0
#> 271 285    0.00    0.00      0
#> 272 286    0.00    0.00      0
#> 273 287    0.00    0.00      0
#> 274 288    0.00    0.00      0
#> 275 289    0.00    0.00      0
#> 276 290    0.00    0.00      0
#> 277 291    0.00    0.00      0
#> 278 292    0.00    0.00      0
#> 279 293    0.00    0.00      0
#> 280 294    0.00    0.00      0
#> 281 295    0.00    0.00      0
#> 282 296    0.00    0.00      0
#> 283 297    0.00    0.00      0
#> 284 298    0.00    0.00      0
#> 285 299    0.00    0.00      0
#> 286 300    0.00    0.00      0
#> 287 301    0.00    0.00      0
#> 288 302    0.00    0.00      0
#> 289 303    0.00    0.00      0
#> 290 304    0.00    0.00      0
#> 291 305    0.00    0.00      0
#> 292 306    0.00    0.00      0
#> 293 307    0.00    0.00      0
#> 294 308    0.00    0.00      0
#> 295 309    0.00    0.00      0
#> 296 310    0.00    0.00      0
#> 297 311    0.00    0.00      0
#> 298 312    0.00    0.00      0
#> 299 313    0.00    0.00      0
#> 300 314    0.00    0.00      0
#> 301 315    0.00    0.00      0
#> 302 316    0.00    0.00      0
#> 303 317    0.00    0.00      0
#> 304 318    0.00    0.00      0
#> 305 319    0.00    0.00      0
#> 306 320    0.00    0.00      0
#> 307 321    0.00    0.00      0
#> 308 322    0.00    0.00      0
#> 309 323    0.00    0.00      0
#> 310 324    0.00    0.00      0
#> 311 325    0.00    0.00      0
#> 312 326    0.00    0.00      0
#> 313 327    0.00    0.00      0
#> 314 328    0.00    0.00      0
#> 315 329    0.00    0.00      0
#> 316 330    0.00    0.00      0
#> 317 331    0.00    0.00      0
#> 318 332    0.00    0.00      0
#> 319 333    0.00    0.00      0
#> 320 334    0.00    0.00      0
#> 321 335    0.00    0.00      0
#> 322 336    0.00    0.00      0
#> 323 337    0.00    0.00      0
#> 324 338    0.00    0.00      0
#> 325 339    0.00    0.00      0
#> 326 340    0.00    0.00      0
#> 327 341    0.00    0.00      0
#> 328 342    0.00    0.00      0
#> 329 343    0.00    0.00      0
#> 330 344    0.00    0.00      0
#> 331 345    0.00    0.00      0
#> 332 346    0.00    0.00      0
#> 333 347    0.00    0.00      0
#> 334 348    0.00    0.00      0
#> 335 349    0.00    0.00      0
#> 336 350    0.00    0.00      0
#> 337 351    0.00    0.00      0
#> 338 352    0.00    0.00      0
#> 339 353    0.00    0.00      0
#> 340 354    0.00    0.00      0
#> 341 355    0.00    0.00      0
#> 342 356    0.00    0.00      0
#> 343 357    0.00    0.00      0
#> 344 358    0.00    0.00      0
#> 345 359    0.00    0.00      0
#> 346 360    0.00    0.00      0
#> 347 361    0.00    0.00      0
#> 348 362    0.00    0.00      0
#> 349 363    0.00    0.00      0
#> 350 364    0.00    0.00      0
#> 351 365    0.00    0.00      0
#> 352 366    0.00    0.00      0
#> 353 367    0.00    0.00      0
#> 354 368    0.00    0.00      0
#> 355 369    0.00    0.00      0
#> 356 370    0.00    0.00      0
#> 357 371    0.00    0.00      0
#> 358 372    0.00    0.00      0
#> 359 373    0.00    0.00      0
#> 360 374    0.00    0.00      0
#> 361 375    0.00    0.00      0
#> 362 376    0.00    0.00      0
#> 363 377    0.00    0.00      0
#> 364 378    0.00    0.00      0
#> 365 379    0.00    0.00      0
#> 366 380    0.00    0.00      0
#> 367 381    0.00    0.00      0
#> 368 382    0.00    0.00      0
#> 369 383    0.00    0.00      0
#> 370 384    0.00    0.00      0
#> 371 385    0.00    0.00      0
#> 372 386    0.00    0.00      0
#> 373 387    0.00    0.00      0
#> 374 388    0.00    0.00      0
#> 375 389    0.00    0.00      0
#> 376 390    0.00    0.00      0
#> 377 391    0.00    0.00      0
#> 378 392    0.00    0.00      0
#> 379 393    0.00    0.00      0
#> 380 394    0.00    0.00      0
#> 381 395    0.00    0.00      0
#> 382 396    0.00    0.00      0
#> 383 397    0.00    0.00      0
#> 384 398    0.00    0.00      0
#> 385 399    0.00    0.00      0
#> 386 400    0.00    0.00      0
#> 387 401    0.00    0.00      0
#> 388 402    0.00    0.00      0
#> 389 403    0.00    0.00      0
#> 390 404    0.00    0.00      0
#> 391 405    0.00    0.00      0
#> 392 406    0.00    0.00      0
#> 393 407    0.00    0.00      0
#> 394 408    0.00    0.00      0
#> 395 409    0.00    0.00      0
#> 396 410    0.00    0.00      0
#> 397 411    0.00    0.00      0
#> 398 412    0.00    0.00      0
#> 399 413    0.00    0.00      0
#> 400 414    0.00    0.00      0
#> 401 415    0.00    0.00      0
#> 402 416    0.00    0.00      0
#> 403 417    0.00    0.00      0
#> 404 418    0.00    0.00      0
#> 405 419    0.00    0.00      0
#> 406 420    0.00    0.00      0
#> 407 421    0.00    0.00      0
#> 408 422    0.00    0.00      0
#> 409 423    0.00    0.00      0
#> 410 424    0.00    0.00      0
#> 411 425    0.00    0.00      0
#> 412 426    0.00    0.00      0
#> 413 427    0.00    0.00      0
#> 414 428    0.00    0.00      0
#> 415 429    0.00    0.00      0
#> 416 430    0.0
```

Promenljive *mortgage* i *overdue* nemaju izbalansirane raspodele, odnosno medijana i minimum je 0 za obe kolone, ali za sada nećemo ništa isključivati iz uzorka.



Spajanje podataka

U ovoj sekciji spojićemo podatke iz *app.csv* i *cb.csv* i posmatrati podatke.

```
df<-left_join(app2, cb, by = "id")
glimpse(df)
```

```

Rows: 9,993
Columns: 20
$ id      <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21,...
$ bad     <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
$ date    <date> 2011-01-01, 2011-01-01, 2011-01-01, 2011-01-01, 2011-01-01, 2011-01-01, 2...
$ amount  <dbl> 17200, 25800, 19400, 27300, 37400, 29700, 19100, 19200, 35900, 9200, 26200...
$ installments <int> 48, 48, 48, 48, 36, 36, 36, 48, 48, 36, 48, 36, 48, 36, 48, 36, 48, 48, 60...
$ interest <dbl> 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12...
$ birthdate <date> 1970-01-01, 1970-01-01, 1970-01-01, 1970-01-01, 1970-01-01, 1970-01-01, 1...
$ sex      <chr> "male", "female", "male", "male", "female", "female", "female", "female", ...
$ family   <chr> "married", "married", "married", "widowed", "married", "married", "married...
$ children <dbl> 0, 1, 1, 1, 0, 0, 3, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0, 4, 1, 1, 1, 0, 3, 0,...
$ rent     <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, TRUE, FALSE, TRUE, TRUE, FALSE, TRUE, F...
$ education <chr> "3", "2", "2", "2", "2", "2", "3", "2", "3", "3", "2", "2", "3", "3", "2", ...
$ employment <chr> "retired", "employed", "employed", "retired", "employed", "employed", "emp...

```

Vidimo da ima 4382 reda gde su debt, mortgage i overdue NA, tj da za te klijente nema podataka iz kreditnog biroa.

```

(na_count <- sapply(df, function(x) sum(is.na(x))))
id      bad      date      amount installments interest birthdate
0       0       0       0       0       0       0
sex      family children      rent education employment income
0       0       0       0       0       0       0
rent_1   self_empl age      debt      mortgage      overdue
0       0       0      4382      4382      4382

```

```

na_values <- filter(df, is.na(debt))
table(na_values$bad)

```

```

0 1
4311 71

```

Medju NA vrednostima nema mnogo njih gde je bad=1, tako da ćemo te vrednosti zameniti nulom i kolonu *overdue* ćemo podeliti na 3 klase prema danima kasnjenja.

```

df <- mutate(df, overdue = ifelse(is.na(df$overdue), 0, df$overdue))
df <- mutate(df, overdue_1 = cut(df$overdue, c(-Inf, 0, 40, Inf)))

df <- mutate(df, debt_1 = ifelse(is.na(df$debt), 0, df$debt))
df <- mutate(df, mortgage_1 = ifelse(is.na(df$mortgage), 0, df$mortgage))

```

Kolinearnost

U ovoj sekciji ispitaćemo da li postoji kolienarnost među podacima.

Posmatraćemo prvo podatke iz kreditnog biroa.

```

cor2pcor(cov(cb))
      [,1] [,2] [,3] [,4]
[1,] 1.000000000 -0.009609347 0.007050046 0.003280599
[2,] -0.009609347 1.000000000 0.861929608 -0.007880941
[3,] 0.007050046 0.861929608 1.000000000 0.006088821
[4,] 0.003280599 -0.007880941 0.006088821 1.000000000

```

Postoji povezanost između *mortgage* i *debt*, pa ćemo isključiti *mortgage* iz dalje analize.

```
df1 <- select(df, -mortgage)
```

Posmatramo kolinearnost na celom skupu podataka.

```
test1<-select(df1,amount, income,age,installments,debt_1, children, overdue)
cor2pcor(cov(test1))
      [,1] [,2] [,3] [,4] [,5] [,6] [,7]
[1,] 1.000000000 0.60234302 0.001119749 0.4270728761 0.014580924 -0.004930511 0.0065278098
[2,] 0.602343024 1.000000000 -0.197335847 -0.2489143319 0.148955551 0.064620327 -0.0503289520
[3,] 0.001119749 -0.19733585 1.000000000 -0.0091864260 -0.157463343 0.107009404 0.0237845612
[4,] 0.427072876 -0.24891433 -0.009186426 1.0000000000 -0.016039553 -0.004768982 -0.0002629879
[5,] 0.014580924 0.14895555 -0.157463343 -0.0160395528 1.000000000 0.008354677 0.2366865330
[6,] -0.004930511 0.06462033 0.107009404 -0.0047689823 0.008354677 1.000000000 0.0067923212
[7,] 0.006527810 -0.05032895 0.023784561 -0.0002629879 0.236686533 0.006792321 1.000000000
```

Vidimo da su *income* i *installments* povezani sa *amount* varijablom, te ćemo iz dalje analize isključiti *income* i *installments*.

```
df2<-select(df1, c(-income,-installments))
```

Deljenje uzorka po godinama

Kada smo završili transformaciju/odstranjivanje promenljivih, sada možemo da podelimo uzorak na razvojni i test kako bismo krenuli sa pravljenjem modela.

```
train <-filter(df2, year(date)==2011)
test <-filter(df2, year(date)==2012)
glimpse(train) #5009
glimpse(test) #4984
```

Logistička regresija – razvojni uzorak

U ovoj sekciji razmatraćemo logističku regresiju na razvojnem uzorku.

Call:

```
glm(formula = bad ~ sexMale + divorced_single + children + self_empl +
    rent_1 + debt_1 + overdue_1 + amount + education + age, family = binomial,
    data = train)
```

Deviance Residuals:

```
    Min     1Q  Median     3Q     Max
-1.3101 -0.2349 -0.1625 -0.1160  3.2670
```

Coefficients:

```
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.979e+01  3.705e+02 -0.053  0.95741
sexMale1      4.094e-01  1.952e-01  2.097  0.03596 *
divorced_single1 8.250e-01  2.111e-01  3.908 9.30e-05 ***
children      3.469e-01  7.376e-02  4.704 2.56e-06 ***
self_empl1    6.775e-01  2.626e-01  2.580 0.00987 **
rent_1        1.431e+00  2.381e-01  6.011 1.85e-09 ***
debt_1        9.542e-04  1.632e-04  5.848 4.98e-09 ***
overdue_1(0,40] 5.349e-01  2.411e-01  2.219 0.02652 *
overdue_1(40, Inf] 1.276e+00  6.173e-01  2.067 0.03876 *
amount        2.799e-06  1.127e-05  0.248 0.80384
education2    1.361e+01  3.705e+02  0.037 0.97069
education3    1.378e+01  3.705e+02  0.037 0.97034
education4    1.390e+01  3.705e+02  0.038 0.97009
age           4.647e-03  8.690e-03  0.535 0.59283
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1102.87 on 5008 degrees of freedom
Residual deviance: 986.77 on 4995 degrees of freedom
AIC: 1014.8

Vidimo da *amount*, *age* i *education* nisu značajne promenljive za ovaj model, te ćemo ih isključiti.

Call:
glm(formula = bad ~ sexMale + divorced_single + children + self_empl +
rent_1 + debt_1 + overdue_1, family = binomial, data = train)

Deviance Residuals:
Min 1Q Median 3Q Max
-1.2142 -0.2320 -0.1623 -0.1159 3.3076

Coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept) -5.8188604 0.2758760 -21.092 < 2e-16 ***
sexMale1 0.4051087 0.1948932 2.079 0.0377 *
divorced_single1 0.8194438 0.2081597 3.937 8.26e-05 ***
children 0.3530242 0.0722264 4.888 1.02e-06 ***
self_empl1 0.6654711 0.2610282 2.549 0.0108 *
rent_1 1.3954418 0.2235381 6.243 4.31e-10 ***
debt_1 0.0009253 0.0001518 6.094 1.10e-09 ***
overdue_1(0,40] 0.5503324 0.2398831 2.294 0.0218 *
overdue_1(40,Inf] 1.2434276 0.6167676 2.016 0.0438 *

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1102.87 on 5008 degrees of freedom
Residual deviance: 992.28 on 5000 degrees of freedom
AIC: 1010.3

Vidimo da su sve promenljive značajne za model.
Sada ćemo da izračunamo verovatnoće default-a.

```
train_lr<- mutate(train, p = predict(train_model1, type = "response"))
mean(train_lr$p)
0.02315832
summary(train$p)
Min. 1st Qu. Median Mean 3rd Qu. Max.
0.003707 0.007405 0.014746 0.023158 0.030293 0.475248
table(train_lr$bad)
0 1
4893 116 #neizbalansiran broj onih koji su kasnili sa kreditom i onih koji nisu
```

Kada smo razdvojili verovatnoće na klase vidimo da za svaku klasu model preceni verovatnoću default-a.

```

train_lr$p_class <- cut(train_lr$p, 5)
table(p_class)
merge(aggregate(bad ~ p_class, data = train_lr, function(x) c(length = length(x), sum = sum(x)))
      , aggregate(p ~ p_class, data = train_lr, mean))
  p_class bad.length bad.sum      p
1 (0.00324,0.098]    4901   101 0.02036444
2 (0.098,0.192]     93    13 0.13016403
3 (0.192,0.287]     10     1 0.20446418
4 (0.287,0.381]      2     0 0.30162307
5 (0.381,0.476]      3     1 0.48025296

```

Prag za logističku regresiju

Kako bismo našli prag, odnosno koja vrednost verovatnoće nam govori da će klijent doći do defaulta-a, posmatrajmo matricu konfuzije za različite pragove.

Ukoliko kao prag stavimo 0.04, odnosno da odlučimo da odbijemo klijente koji imaju veću ili jednaku verovatnoću, a prihvatimo klijente sa manjom verovatnoćom, pogledajmo da li bismo bili u pravu.

```

train_lr <- mutate(train_lr, bad_predicted = ifelse(p<=0.04,0,1))

# Calculate the model's accuracy
mean(train_lr$bad == train_lr$bad_predicted)
[1] 0.8562587

```

Vidimo da je preciznost modela 85.56%, što može da nas navede da pomislimo da je model dobar, ali ova informacija nam ne govori koliko smo pogodili da je klijent imao default.

Pogledajmo matricu konfuzije

```

table(ifelse(train_lr$bad==1,'Default True','Default False'), train_lr$bad_predicted)
  0  1
Default False 4236 657
Default True   63  53

```

Vidimo da dobro pogađamo oni koji zaista neće imati default, ali da od klijenata koji će imati default, sa ovim modelom odbili bi samo 46% njih. Što je u ovom slučaju veliki procenat, ne bi bilo dobro za banku da odobri toliko veliki procenat loših kredita.

```

# Sensitivity 0.4568966 - TruePositive / (TruePositive + FalseNegative)
53/(63+53)

```

```

#Specificity 0.8657265 - TrueNegative / (FalsePositive + TrueNegative)
4236 /(4236+657)

```

Pokušajmo da promenimo prag.

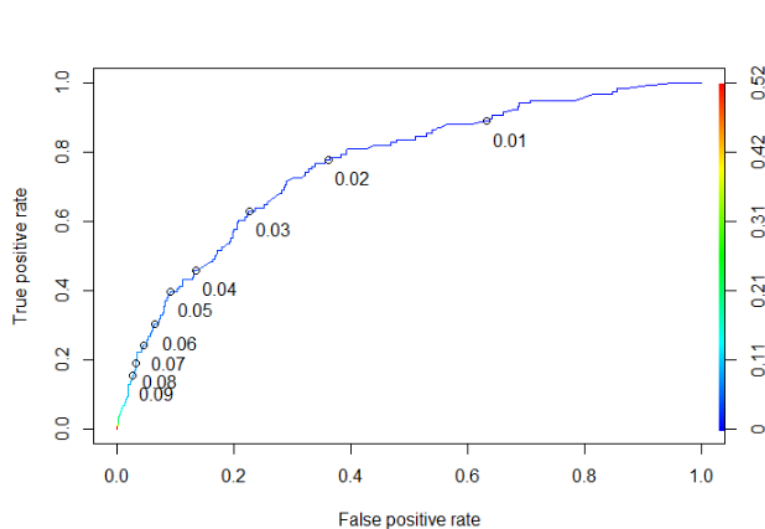
```

ROCRpred = prediction(train_lr$p, train_lr$bad)
ROCRperf = performance(ROCRpred, "tpr", "fpr")

plot(ROCRperf)
# Add colors
plot(ROCRperf, colorize=TRUE)

```

```
# Add threshold labels
plot(ROCperf, colorize=TRUE, print.cutoffs.at=seq(0.01,0.09,by=0.01), text.adj=c(-0.2,1.7))
```



Kako u ovom slučaju želimo da imamo više onih koji su True Positive, jer je bolje da odbijemo neke klijente koji bi otplatili kredit nego da damo kredit klijentima koji ga neće vratiti, probajmo da kao prag uzmemo 0.02, jer se sa slike čini da bi taj prag bio odgovarajući.

```
train_lr1 <- mutate(train_lr, bad_predicted = ifelse(p<=0.02,0,1))
mean(train_lr1$bad == train_lr1$bad_predicted) # 0.6412458
table(ifelse(train_lr1$bad==1,'Default True','Default False'), train1$bad_predicted)
```

```
      0      1
Default False 3122 1771
Default True   26   90
```

```
# Sensitivity 0.7758621 - TruePositive / (TruePositive + FalseNegative)
90/(26+90)
```

```
#Specificity 0.6380544 - TrueNegative / (FalsePositive + TrueNegative)
3122 /(3122 +1771)
```

Vidimo da, iako ovaj model ima manju preciznost, bolji je što se tiče pogađanja klijenata koji ne bi otplatili kredit.

Logistička regresija – test uzorak

Sada ćemo videti kako se naš model ponaša na podacima koje nije video do sada.

```
test_lr <- mutate(test_lr, p = predict(train_model1, type = "response", newdata = test))
test_lr $p <- predict(train_model_lr1, type = "response", newdata = test_lr)

p_class_test_lr <- cut(test_lr$p, 5)
merge(aggregate(bad ~ p_class_test_lr, data = test, function(x) c(length = length(x), sum = sum(x)))
, aggregate(p ~ p_class_test, data = test_lr, mean))
p_class_test bad.length bad.sum      p
1 [0.00261,0.0736]    4776    112 0.01837484
```

```

2 (0.0736,0.144]    169    20 0.10034934
3 (0.144,0.215]    32     6 0.17455995
4 (0.215,0.285]     4     0 0.23694455
5 (0.285,0.356]     3     1 0.32143267

```

```

summary(test_lr$p)
summary(train_lr$p)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.002962 0.006697 0.013840 0.022515 0.026933 0.355943
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.002962 0.006842 0.013893 0.023158 0.028326 0.521510

```

Vidimo da je verovatnoća malo veća sa razvojnim podacima.

Kada koristimo isti prag 0.02, rezultati su sledeći. Tačnost i True Positive vrednosti se nisu mnogo promenile u odnosu na razvojni model, što znači da nije došlo do overfitting-a.

```

test_lr <- mutate(test_lr, bad_predicted = ifelse(p<=0.02,0,1))
# Calculate the model's accuracy
mean(test_lr$bad == test_lr$bad_predicted) # 0.6494783
table(ifelse(test_lr$bad==1,'Default True','Default False'), test_lr$bad_predicted)
      0      1
Default False 3133 1712
Default True   35  104

```

```

# Sensitivity 0.7482014 - TruePositive / (TruePositive + FalseNegative)
104/(35+104)

```

```

#Specificity 0.646646- TrueNegative / (FalsePositive + TrueNegative)
3133 /(3133 +1712)

```

Random forest – razvojni podaci

U ovoj sekciji posmatraćemo model napravljen pomoću algoritma random forest.

```

train_rf<-mutate(train, bad=as.factor(bad))
test_rf<-mutate(test, bad=as.factor(bad))
model_rf <- randomForest(bad ~ sexMale + divorced_single + children + self_empl + rent_1 + debt_1 + overdue_1 +
amount + education +age,data=train_rf, importance = TRUE)

```

Uključili smo one promenljive koje smo isključili iz logističke regresije jer bi random forest trebao sam da proceni koje promenljive nisu značajne.

```

table(ifelse(train_rf$bad==1,'Default True','Default False'), train_rf$p1)
mean(train_rf$bad == train_rf$p1) #0.9960072
      0      1
Default False 4893   0
Default True   20  96

```

```

# Sensitivity 0.7769784 - TruePositive / (TruePositive + FalseNegative)
96/(20 +96) #0.8275862

```

```

#Specificity 0.6315789 - TrueNegative / (FalsePositive + TrueNegative)
4893/(4893)

```

Vidimo da je tačnost modela velika oko 99%, kao i da je veliki postotak True Positive vrednosti.

Random forest – test podaci

Pogledajmo kako se random forest ponaša na podacima koje nije video pre.

```
test_rf<-mutate(test_rf, p1 = predict(model_rf, test_rf,type = "class"))
mean(test_rf$bad==test_rf$p1) #0.9721108
table(ifelse(test_rf$bad==1,'Default True','Default False'), test_rf$p1)
```

	0	1
Default False	4845	0
Default True	139	0

Vidimo da se random forest jako lose ponaša sa klijentima koju su imali default. Iako je tačnost modela velik, oko 97%, nikako ne bismo smeli da prihvatimo ovo kao model jer bi to značilo da bi banka prihvatila klijente koji bi imali default.

Deljenje uzorka po nausmičnom odabiru podataka

U fajlu DataScienceRandomSampleSplit_BSANDO.R smo podeili uzorak slučajnim odabirom u odnosu 65% - 35%. Ponovili smo isti proces kao i kod podele uzorka prema godinama. Rezultati se nisu značajno razlikovali.