Projektbericht2

Ferdinand Bubeck und Johannes Bubeck

25 3 2021

#import libraries  
library(readr)  
library(dplyr)

## Warning: package 'dplyr' was built under R version 3.6.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.6.3

library(tidyverse)

## Warning: package 'tidyverse' was built under R version 3.6.3

## -- Attaching packages ---------------------------------------------------------------------------------- tidyverse 1.3.0 --

## v tibble 3.0.0 v stringr 1.4.0  
## v tidyr 1.0.2 v forcats 0.5.0  
## v purrr 0.3.3

## Warning: package 'tibble' was built under R version 3.6.3

## Warning: package 'tidyr' was built under R version 3.6.3

## Warning: package 'purrr' was built under R version 3.6.3

## Warning: package 'forcats' was built under R version 3.6.3

## -- Conflicts ------------------------------------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

datTest <- read\_csv("aug\_test.csv")

## Parsed with column specification:  
## cols(  
## enrollee\_id = col\_double(),  
## city = col\_character(),  
## city\_development\_index = col\_double(),  
## gender = col\_character(),  
## relevent\_experience = col\_character(),  
## enrolled\_university = col\_character(),  
## education\_level = col\_character(),  
## major\_discipline = col\_character(),  
## experience = col\_character(),  
## company\_size = col\_character(),  
## company\_type = col\_character(),  
## last\_new\_job = col\_character(),  
## training\_hours = col\_double()  
## )

datTrain <- read\_csv("aug\_train.csv")

## Parsed with column specification:  
## cols(  
## enrollee\_id = col\_double(),  
## city = col\_character(),  
## city\_development\_index = col\_double(),  
## gender = col\_character(),  
## relevent\_experience = col\_character(),  
## enrolled\_university = col\_character(),  
## education\_level = col\_character(),  
## major\_discipline = col\_character(),  
## experience = col\_character(),  
## company\_size = col\_character(),  
## company\_type = col\_character(),  
## last\_new\_job = col\_character(),  
## training\_hours = col\_double(),  
## target = col\_double()  
## )

#Data Preparation  
datTrain\_casted <- datTrain %>%   
 select(-enrollee\_id, -city) %>%  
 mutate(city\_development\_index = as.numeric(city\_development\_index),  
 target = as.factor(target)) %>%   
 as.data.frame() %>%  
 mutate\_if(is.character, as.factor) #converting all variables with type=char to factor  
  
  
datTrain\_notImputed <- datTrain\_casted  
  
datTrain\_NaReduced <- na.omit(datTrain\_notImputed)  
  
#NAs  
detectNA <- function(x){sum(is.na(x))/length(x)\*100}  
apply(datTrain\_NaReduced, 2, detectNA)

## city\_development\_index gender relevent\_experience   
## 0 0 0   
## enrolled\_university education\_level major\_discipline   
## 0 0 0   
## experience company\_size company\_type   
## 0 0 0   
## last\_new\_job training\_hours target   
## 0 0 0

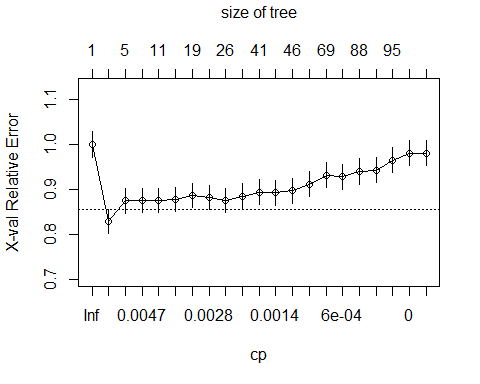
#Test / Training Split  
set.seed(304)  
trainingRows2 <- sort(sample(nrow(datTrain\_NaReduced), nrow(datTrain\_NaReduced)\*.7))  
  
Train\_NaReduced <- datTrain\_NaReduced[trainingRows2,]  
Test\_NaReduced <- datTrain\_NaReduced[-trainingRows2,]  
  
#Decision Tree  
library(rpart)  
library(rpart.plot)

## Warning: package 'rpart.plot' was built under R version 3.6.3

##1st tree  
set.seed(31)  
dt.default\_NotImputed <- rpart(target~ ., data=Train\_NaReduced, cp=-1)  
  
printcp(dt.default\_NotImputed)

##   
## Classification tree:  
## rpart(formula = target ~ ., data = Train\_NaReduced, cp = -1)  
##   
## Variables actually used in tree construction:  
## [1] city\_development\_index company\_size company\_type   
## [4] education\_level enrolled\_university experience   
## [7] gender last\_new\_job major\_discipline   
## [10] relevent\_experience training\_hours   
##   
## Root node error: 1027/6268 = 0.16385  
##   
## n= 6268   
##   
## CP nsplit rel error xerror xstd  
## 1 0.17137293 0 1.00000 1.00000 0.028534  
## 2 0.00519312 1 0.82863 0.82863 0.026406  
## 3 0.00486855 4 0.81305 0.87439 0.027008  
## 4 0.00462512 6 0.80331 0.87537 0.027020  
## 5 0.00389484 10 0.78481 0.87537 0.027020  
## 6 0.00357027 15 0.76534 0.87731 0.027045  
## 7 0.00292113 18 0.75463 0.88608 0.027157  
## 8 0.00259656 21 0.74586 0.88218 0.027108  
## 9 0.00194742 25 0.73515 0.87537 0.027020  
## 10 0.00162285 36 0.71178 0.88510 0.027145  
## 11 0.00146056 40 0.70497 0.89387 0.027256  
## 12 0.00129828 42 0.70204 0.89192 0.027231  
## 13 0.00097371 45 0.69815 0.89679 0.027293  
## 14 0.00073028 52 0.69133 0.91139 0.027476  
## 15 0.00064914 68 0.67868 0.93087 0.027716  
## 16 0.00055641 71 0.67673 0.92795 0.027680  
## 17 0.00048685 87 0.66699 0.93963 0.027822  
## 18 0.00038948 89 0.66602 0.94255 0.027857  
## 19 0.00032457 94 0.66407 0.96495 0.028125  
## 20 0.00000000 97 0.66310 0.98053 0.028308  
## 21 -1.00000000 342 0.66310 0.98053 0.028308

plotcp(dt.default\_NotImputed)



#Prediction & Performance  
#Train Data  
train.default\_NotImputed.pred <- predict(dt.default\_NotImputed, newdata = Train\_NaReduced, type = "class")  
train.default\_NotImputed.confMatrix <- table(train.default\_NotImputed.pred, Train\_NaReduced[, 12])  
print(train.default\_NotImputed.confMatrix)

##   
## train.default\_NotImputed.pred 0 1  
## 0 5073 513  
## 1 168 514

train.default\_NotImputed.accuracy <- sum(diag(train.default\_NotImputed.confMatrix))/sum(train.default\_NotImputed.confMatrix)  
print(train.default\_NotImputed.accuracy)

## [1] 0.8913529

#Test Data  
test.default\_NotImputed.pred <- predict(dt.default\_NotImputed, newdata = Test\_NaReduced, type = "class")  
test.default\_NotImputed.confMatrix <- table(test.default\_NotImputed.pred, Test\_NaReduced[, 12])  
print(test.default\_NotImputed.confMatrix)

##   
## test.default\_NotImputed.pred 0 1  
## 0 2091 315  
## 1 140 141

test.default\_NotImputed.accuracy <- sum(diag(test.default\_NotImputed.confMatrix))/sum(test.default\_NotImputed.confMatrix)  
print(test.default\_NotImputed.accuracy)

## [1] 0.8306662

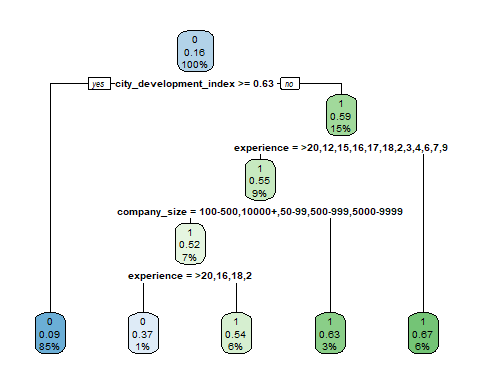
#search cp with lowest Cross-Validation Error  
xerror.min2 <- dt.default\_NotImputed$cptable[which.min(dt.default\_NotImputed$cptable[,4]),]  
cp.best2 <- xerror.min2[1]  
cp.best2

## CP   
## 0.005193119

#cp splitted by n=4  
cp.manual <- 0.00486855  
  
#pruning  
dt.pruned\_manually <- prune(dt.default\_NotImputed, cp=cp.manual)  
  
summary(dt.pruned\_manually)

## Call:  
## rpart(formula = target ~ ., data = Train\_NaReduced, cp = -1)  
## n= 6268   
##   
## CP nsplit rel error xerror xstd  
## 1 0.171372931 0 1.0000000 1.0000000 0.02853365  
## 2 0.005193119 1 0.8286271 0.8286271 0.02640640  
## 3 0.004868550 4 0.8130477 0.8743914 0.02700787  
##   
## Variable importance  
## city\_development\_index experience company\_size   
## 97 3 1   
##   
## Node number 1: 6268 observations, complexity param=0.1713729  
## predicted class=0 expected loss=0.1638481 P(node) =1  
## class counts: 5241 1027  
## probabilities: 0.836 0.164   
## left son=2 (5322 obs) right son=3 (946 obs)  
## Primary splits:  
## city\_development\_index < 0.632 to the right, improve=410.435000, (0 missing)  
## experience splits as RLRLLLLLLLLLLRLRRRRRRL, improve= 49.528340, (0 missing)  
## enrolled\_university splits as RLL, improve= 10.407600, (0 missing)  
## last\_new\_job splits as LRRRRR, improve= 8.838932, (0 missing)  
## major\_discipline splits as LLLRLR, improve= 6.270787, (0 missing)  
## Surrogate splits:  
## experience splits as RLLLLLLLLLLLLLLLLLLLLL, agree=0.85, adj=0.003, (0 split)  
##   
## Node number 2: 5322 observations  
## predicted class=0 expected loss=0.08756107 P(node) =0.8490747  
## class counts: 4856 466  
## probabilities: 0.912 0.088   
##   
## Node number 3: 946 observations, complexity param=0.005193119  
## predicted class=1 expected loss=0.4069767 P(node) =0.1509253  
## class counts: 385 561  
## probabilities: 0.407 0.593   
## left son=6 (583 obs) right son=7 (363 obs)  
## Primary splits:  
## experience splits as RLRRRLRRLLLLRLRLLRLLRL, improve=6.388913, (0 missing)  
## company\_size splits as RRLLLLLL, improve=3.733937, (0 missing)  
## last\_new\_job splits as LRRLRL, improve=3.026805, (0 missing)  
## city\_development\_index < 0.5045 to the left, improve=2.858243, (0 missing)  
## major\_discipline splits as LRLRLL, improve=1.982597, (0 missing)  
## Surrogate splits:  
## last\_new\_job splits as LLLLLR, agree=0.622, adj=0.014, (0 split)  
## major\_discipline splits as LLLLRL, agree=0.618, adj=0.006, (0 split)  
## training\_hours < 215 to the left, agree=0.618, adj=0.006, (0 split)  
## company\_type splits as LLLLRL, agree=0.617, adj=0.003, (0 split)  
##   
## Node number 6: 583 observations, complexity param=0.005193119  
## predicted class=1 expected loss=0.4528302 P(node) =0.09301213  
## class counts: 264 319  
## probabilities: 0.453 0.547   
## left son=12 (423 obs) right son=13 (160 obs)  
## Primary splits:  
## company\_size splits as RRLRLLLL, improve=2.671618, (0 missing)  
## last\_new\_job splits as LRRLRL, improve=2.378418, (0 missing)  
## company\_type splits as RRRRLR, improve=2.316375, (0 missing)  
## experience splits as -L---L--LLRL-L-RR-RR-R, improve=1.945171, (0 missing)  
## training\_hours < 277 to the right, improve=1.432617, (0 missing)  
## Surrogate splits:  
## company\_type splits as RLLLLL, agree=0.750, adj=0.087, (0 split)  
## major\_discipline splits as LRLLLL, agree=0.731, adj=0.019, (0 split)  
## education\_level splits as L-LR-, agree=0.727, adj=0.006, (0 split)  
## experience splits as -L---L--LLLL-R-LL-LL-L, agree=0.727, adj=0.006, (0 split)  
## training\_hours < 2.5 to the right, agree=0.727, adj=0.006, (0 split)  
##   
## Node number 7: 363 observations  
## predicted class=1 expected loss=0.3333333 P(node) =0.05791321  
## class counts: 121 242  
## probabilities: 0.333 0.667   
##   
## Node number 12: 423 observations, complexity param=0.005193119  
## predicted class=1 expected loss=0.4822695 P(node) =0.06748564  
## class counts: 204 219  
## probabilities: 0.482 0.518   
## left son=24 (62 obs) right son=25 (361 obs)  
## Primary splits:  
## experience splits as -L---R--RLRL-L-RR-RR-R, improve=3.129584, (0 missing)  
## company\_type splits as RRRLLR, improve=1.606023, (0 missing)  
## last\_new\_job splits as LRRLRR, improve=1.425794, (0 missing)  
## training\_hours < 4.5 to the right, improve=1.243704, (0 missing)  
## major\_discipline splits as LLRRLR, improve=1.168982, (0 missing)  
## Surrogate splits:  
## major\_discipline splits as LLRRRR, agree=0.858, adj=0.032, (0 split)  
##   
## Node number 13: 160 observations  
## predicted class=1 expected loss=0.375 P(node) =0.02552648  
## class counts: 60 100  
## probabilities: 0.375 0.625   
##   
## Node number 24: 62 observations  
## predicted class=0 expected loss=0.3709677 P(node) =0.009891512  
## class counts: 39 23  
## probabilities: 0.629 0.371   
##   
## Node number 25: 361 observations  
## predicted class=1 expected loss=0.4570637 P(node) =0.05759413  
## class counts: 165 196  
## probabilities: 0.457 0.543

rpart.plot(dt.pruned\_manually)



#Train Data  
train.pruned\_NotImputed.pred <- predict(dt.pruned\_manually, newdata = Train\_NaReduced, type = "class")  
train.pruned\_NotImputed.confMatrix <- table(train.pruned\_NotImputed.pred, Train\_NaReduced[, 12])  
print(train.pruned\_NotImputed.confMatrix)

##   
## train.pruned\_NotImputed.pred 0 1  
## 0 4895 489  
## 1 346 538

train.pruned\_NotImputed.accuracy <- sum(diag(train.pruned\_NotImputed.confMatrix))/sum(train.pruned\_NotImputed.confMatrix)  
print(train.pruned\_NotImputed.accuracy)

## [1] 0.8667837

#Test Data  
test.pruned\_NotImputed.pred <- predict(dt.pruned\_manually, newdata = Test\_NaReduced, type = "class")  
test.pruned\_NotImputed.confMatrix <- table(test.pruned\_NotImputed.pred, Test\_NaReduced[, 12])  
print(test.pruned\_NotImputed.confMatrix)

##   
## test.pruned\_NotImputed.pred 0 1  
## 0 2070 235  
## 1 161 221

test.pruned\_NotImputed.accuracy <- sum(diag(test.pruned\_NotImputed.confMatrix))/sum(test.pruned\_NotImputed.confMatrix)  
print(test.pruned\_NotImputed.accuracy)

## [1] 0.8526237