UNIVERZITET U SARAJEVU

ELEKTROTEHNIČKI FAKULTET SARAJEVO

DOMAĆA ZADAĆA 1
Zadatak 1

MAŠINSKO UČENJE

Odsjek: Računarstvo i Informatika
Datum: 18.11.2019

Studenti:

- **Mašović Haris, 1689/17993**
- **Muminović Amir, 1661/17744**

Opis seta podataka

U prilogu zadaće 1, dat je set podataka “attrition_train.csv”. Potrebno je na osnovu seta podataka, izgraditi klasifikacijski model koji će utvrditi da li će uposlenik neke kompanije napustiti tu kompaniju (Attrition=yes)

U setu se nalaze podaci o uposlenicima (demografski, o vrsti posla, uspješnosti na poslu, zadovoljstvu na poslu, edukaciji, itd). Dodatne informacije o kategoričkim varijablama (**):

- Education: 1 'Below College', 2 'College', 3 'Bachelor', 4 'Master', 5 'Doctor'
- EnvironmentSatisfaction: 1 'Low', 2 'Medium', 3 'High', 4 'Very High'
- JobInvolvement: 1 'Low', 2 'Medium', 3 'High', 4 'Very High'
- JobSatisfaction: 1 'Low', 2 'Medium', 3 'High', 4 'Very High'
- PerformanceRating: 1 'Low', 2 'Good', 3 'Excellent', 4 'Outstanding'
- RelationshipSatisfaction: 1 'Low', 2 'Medium', 3 'High', 4 'Very High'
- WorkLifeBalance: 1 'Bad', 2 'Good', 3 'Better', 4 'Best'

Zadatak 1 (6 bodova)

U skladu sa jednim od osnovnih principa Mašinskog Učenja “garbage in garbage out”, i prateći korake izgradnje predikcijskih modela, **neophodno je upoznati se sa setom podataka (tipovi varijabli i njihova distribucija, deskriptivna statistika, veze između varijabli, itd) i izvršiti pripremu istog (transformacija, skaliranje, ispunjavanje nedostajućih vrijednosti, itd)**. **Dokumentujte proces istraživanja podataka za svaku varijablu i jasno argumentujte primijenjene metode po osnovu prirode podataka i relacija između podataka.**

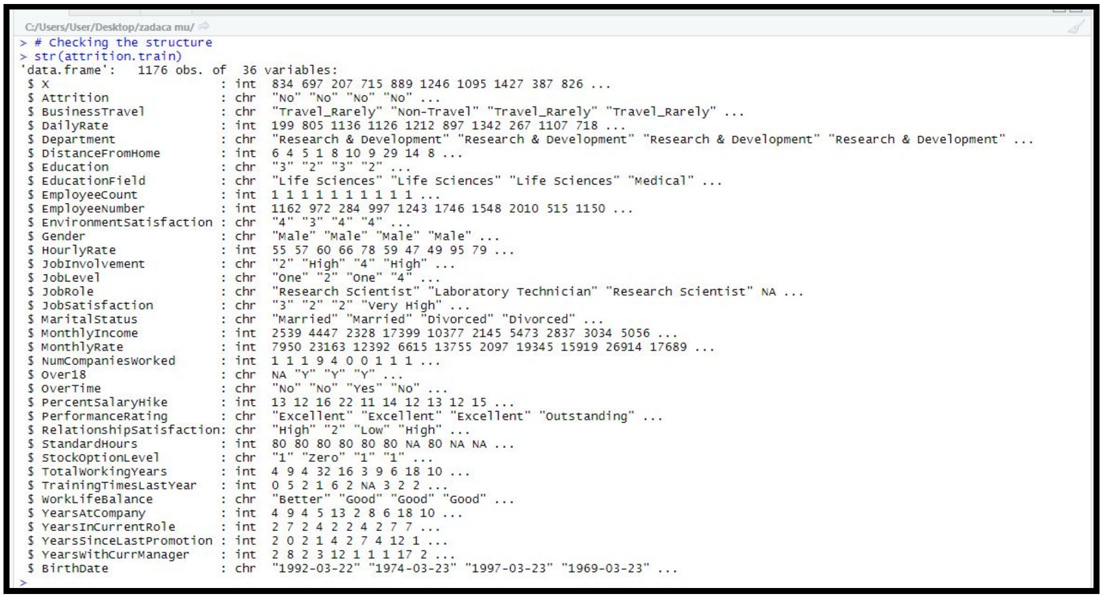
** Prije sveukupnog upoznavanja sa setom podataka, prvo oba seta treba učitati i upoznati se sa vrijednostima. Radi tog razloga će biti učitana oba seta podataka i spojena u jedan (kasnije će se odspojiti nazad na početni slučaj pomoću naše granice) pomoću kojeg ćemo očistiti naše podatke (da ne bi čistili i mijenjali podatke dva puta). To ćemo uraditi sa sljedećim kodom:

```
# Getting to know the data
# Loading training & test sets
attrition.train <- read.csv(file="attrition_train.csv", stringsAsFactors = FALSE, header =
TRUE)
attrition.test <- read.csv(file="attrition_test.csv", stringsAsFactors = FALSE, header = TRUE)

# Creating a new data set with both the test and the train sets
attrition.full <- bind_rows(attrition.train,attrition.test)
# Our divider
LT=dim(attrition.train)[1]
```

Upoznavanje naše trenutne strukture možemo uraditi na sljedeći način:

```
# Checking the structure
str(attrition.train)
```

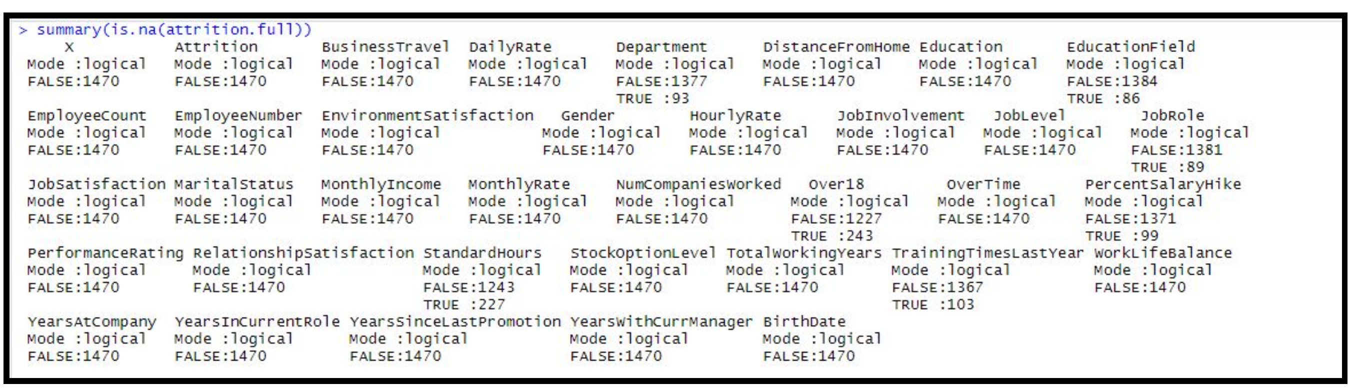
Na osnovu prethodnog poziva dobiti ćemo sljedeći rezultat iz kojeg možemo očitati strukturu našeg data seta na osnovu kojeg vidimo prika tipove varijabla i generalno strukture:



```
C:/Users/User/Desktop/zadaca mu/
> # Checking the structure
> str(attrition.train)
'data.frame':	1176 obs. of  36 variables:
 $ X                       : int  834 697 207 715 889 1246 1095 1427 387 826 ...
 $ Attrition               : chr  "No" "No" "No" "No" ...
 $ BusinessTravel          : chr  "Travel_Rarely" "Non-Travel" "Travel_Rarely" "Travel_Rarely" ...
 $ DailyRate               : int  199 805 1136 1126 1212 897 1342 267 1107 718 ...
 $ Department              : chr  "Research & Development" "Research & Development" "Research & Development" "Research & Development" ...
 $ DistanceFromHome        : int  6 4 5 1 8 10 9 29 14 8 ...
 $ Education               : chr  "3" "2" "3" "2" ...
 $ EducationField          : chr  "Life Sciences" "Life Sciences" "Life Sciences" "Medical" ...
 $ EmployeeCount           : int  1 1 1 1 1 1 1 1 1 1 ...
 $ EmployeeNumber          : int  1162 972 284 997 1243 1746 1548 2010 515 1150 ...
 $ EnvironmentSatisfaction : chr  "4" "3" "4" "4" ...
 $ Gender                  : chr  "Male" "Male" "Male" "Male" ...
 $ HourlyRate              : int  55 57 60 66 78 59 47 49 95 79 ...
 $ JobInvolvement          : chr  "2" "High" "4" "High" ...
 $ JobLevel                : chr  "One" "2" "One" "4" ...
 $ JobRole                 : chr  "Research Scientist" "Laboratory Technician" "Research Scientist" NA ...
 $ JobSatisfaction         : chr  "3" "2" "2" "Very High" ...
 $ MaritalStatus           : chr  "Married" "Married" "Divorced" "Divorced" ...
 $ MonthlyIncome           : int  2539 4447 2328 17399 10377 2145 5473 2837 3034 5056 ...
 $ MonthlyRate             : int  7950 23163 12392 6615 13755 2097 19345 15919 26914 17689 ...
 $ NumCompaniesWorked      : int  1 1 1 9 4 0 0 1 1 1 ...
 $ Over18                  : chr  NA "Y" "Y" "Y" ...
 $ OverTime                : chr  "No" "No" "Yes" "No" ...
 $ PercentSalaryHike       : int  13 12 16 22 11 14 12 13 12 15 ...
 $ PerformanceRating       : chr  "Excellent" "Excellent" "Excellent" "Outstanding" ...
 $ RelationshipSatisfaction: chr  "High" "2" "Low" "High" ...
 $ StandardHours           : int  80 80 80 80 80 80 NA 80 NA NA ...
 $ StockOptionLevel        : chr  "1" "Zero" "1" "1" ...
 $ TotalWorkingYears       : int  4 9 4 32 16 3 9 6 18 10 ...
 $ TrainingTimesLastYear   : int  0 5 2 1 6 2 NA 3 2 2 ...
 $ WorkLifeBalance         : chr  "Better" "Good" "Good" "Good" ...
 $ YearsAtCompany          : int  4 9 4 5 13 2 8 6 18 10 ...
 $ YearsInCurrentRole      : int  2 7 2 4 2 2 4 2 7 7 ...
 $ YearsSinceLastPromotion : int  2 0 2 1 4 2 7 4 12 1 ...
 $ YearsWithCurrManager    : int  2 8 2 3 12 1 1 1 17 2 ...
 $ BirthDate               : chr  "1992-03-22" "1974-03-23" "1997-03-23" "1969-03-23" ...
>
```

Gledanjem u data set zaključeno je da postoji dosta NA vrijednosti u podacima, te sa narednom komandom možemo vidjeti koje varijable imaju NA vrijednosti i koliko ih ima (TRUE predstavlja count):

```
# Get NA summary
summary(is.na(attrition.full))
# Empty values:
# Department-93/1470,
# EducationField-86,
# JobRole-89,
# PercentSalaryHike-99,
# TrainingTimesLastYear-103
```



```
> summary(is.na(attrition.full))
     X            Attrition       BusinessTravel  DailyRate       Department      DistanceFromHome Education       EducationField 
 Mode :logical   Mode :logical   Mode :logical   Mode :logical   Mode :logical   Mode :logical    Mode :logical   Mode :logical  
 FALSE:1470      FALSE:1470      FALSE:1470      FALSE:1470      FALSE:1377      FALSE:1470       FALSE:1470      FALSE:1384     
                                                                 TRUE :93                                         TRUE :86       
 EmployeeCount   EmployeeNumber  EnvironmentSatisfaction   Gender          HourlyRate      JobInvolvement   JobLevel         JobRole       
 Mode :logical   Mode :logical   Mode :logical             Mode :logical   Mode :logical   Mode :logical   Mode :logical   Mode :logical  
 FALSE:1470      FALSE:1470      FALSE:1470                FALSE:1470      FALSE:1470      FALSE:1470      FALSE:1470      FALSE:1381     
                                                                                                                            TRUE :89       
 JobSatisfaction MaritalStatus   MonthlyIncome   MonthlyRate     NumCompaniesWorked   Over18           OverTime        PercentSalaryHike
 Mode :logical   Mode :logical   Mode :logical   Mode :logical   Mode :logical      Mode :logical   Mode :logical   Mode :logical    
 FALSE:1470      FALSE:1470      FALSE:1470      FALSE:1470      FALSE:1470         FALSE:1227      FALSE:1470      FALSE:1371       
                                                                                    TRUE :243                       TRUE :99         
 PerformanceRating RelationshipSatisfaction StandardHours   StockOptionLevel TotalWorkingYears TrainingTimesLastYear WorkLifeBalance
 Mode :logical     Mode :logical            Mode :logical   Mode :logical    Mode :logical     Mode :logical         Mode :logical  
 FALSE:1470        FALSE:1470               FALSE:1243      FALSE:1470       FALSE:1470        FALSE:1367            FALSE:1470     
                                            TRUE :227                                          TRUE :103                            
 YearsAtCompany  YearsInCurrentRole YearsSinceLastPromotion YearsWithCurrManager BirthDate      
 Mode :logical   Mode :logical      Mode :logical           Mode :logical        Mode :logical  
 FALSE:1470      FALSE:1470         FALSE:1470              FALSE:1470           FALSE:1470     
```

Da bi dobili summary od čitavog data seta to možemo uraditi na sljedeći način:

```
# Summary of the data set
summary(attrition.train)
```

Kao rezultat poziva prethodnog imamo:

```
> summary(attrition.train)
       X           Attrition         BusinessTravel       DailyRate       Department        DistanceFromHome  Education
 Min.   :   1.0   Length:1176        Length:1176        Min.   : 103.0   Length:1176        Min.   : 1.000   Length:1176
 1st Qu.: 372.2   Class :character   Class :character   1st Qu.: 464.8   Class :character   1st Qu.: 2.000   Class :character
 Median : 740.5   Mode  :character   Mode  :character   Median : 810.0   Mode  :character   Median : 7.000   Mode  :character
 Mean   : 735.5                                         Mean   : 805.6                      Mean   : 9.259
 3rd Qu.:1093.2                                         3rd Qu.:1162.0                      3rd Qu.:14.000
 Max.   :1470.0                                         Max.   :1496.0                      Max.   :29.000

 EducationField     EmployeeCount EmployeeNumber   EnvironmentSatisfaction    Gender           HourlyRate      JobInvolvement
 Length:1176        Min.   :1     Min.   :   1.0   Length:1176             Length:1176        Min.   : 30.00   Length:1176
 Class :character   1st Qu.:1     1st Qu.: 495.2   Class :character        Class :character   1st Qu.: 48.00   Class :character
 Mode  :character   Median :1     Median :1027.5   Mode  :character        Mode  :character   Median : 66.00   Mode  :character
                    Mean   :1     Mean   :1024.9                                              Mean   : 65.85
                    3rd Qu.:1     3rd Qu.:1546.2                                              3rd Qu.: 84.00
                    Max.   :1     Max.   :2068.0                                              Max.   :100.00

   JobLevel           JobRole          JobSatisfaction    MaritalStatus      MonthlyIncome    MonthlyRate    NumCompaniesWorked
 Length:1176        Length:1176        Length:1176        Length:1176        Min.   : 1051   Min.   : 2097   Min.   :0.000
 Class :character   Class :character   Class :character   Class :character   1st Qu.: 2942   1st Qu.: 8192   1st Qu.:1.000
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Median : 4945   Median :14288   Median :2.000
                                                                             Mean   : 6506   Mean   :14371   Mean   :2.698
                                                                             3rd Qu.: 8384   3rd Qu.:20684   3rd Qu.:4.000
                                                                             Max.   :19999   Max.   :26997   Max.   :9.000

    Over18            OverTime         PercentSalaryHike PerformanceRating  RelationshipSatisfaction StandardHours StockOptionLevel
 Length:1176        Length:1176        Min.   :11.00     Length:1176        Length:1176              Min.   :80    Length:1176
 Class :character   Class :character   1st Qu.:12.00     Class :character   Class :character         1st Qu.:80    Class :character
 Mode  :character   Mode  :character   Median :14.00     Mode  :character   Mode  :character         Median :80    Mode  :character
                                       Mean   :15.24                                                 Mean   :80
                                       3rd Qu.:18.00                                                 3rd Qu.:80
                                       Max.   :25.00                                                 Max.   :80
                                       NA's   :99                                                    NA's   :227
 TotalWorkingYears TrainingTimesLastYear WorkLifeBalance     YearsAtCompany   YearsInCurrentRole YearsSinceLastPromotion
 Min.   : 0.00     Min.   :0.000         Length:1176        Min.   : 0.000   Min.   : 0.00      Min.   : 0.000
 1st Qu.: 6.00     1st Qu.:2.000         Class :character   1st Qu.: 3.000   1st Qu.: 2.00      1st Qu.: 0.000
 Median :10.00     Median :3.000         Mode  :character   Median : 5.000   Median : 3.00      Median : 1.000
 Mean   :11.41     Mean   :2.779                            Mean   : 7.072   Mean   : 4.24      Mean   : 2.205
 3rd Qu.:15.00     3rd Qu.:3.000                            3rd Qu.:10.000   3rd Qu.: 7.00      3rd Qu.: 3.000
 Max.   :40.00     Max.   :6.000                            Max.   :40.000   Max.   :18.00      Max.   :15.000
                   NA's   :103
 YearsWithCurrManager  BirthDate
 Min.   : 0.000       Length:1176
 1st Qu.: 2.000       Class :character
 Median : 3.000       Mode  :character
 Mean   : 4.137
 3rd Qu.: 7.000
 Max.   :17.000
```

Dalje, od ključnog značaja je da saznamo koliko ima unikatnih vrijednosti po svakoj varijabli, i to možemo saznati pomoću sljedećeg koda:

```
# unique values ? - length(unique(x))
apply(attrition.full, 2, function(x) length(unique(x)))
# Just by looking at the set, irrelevant variables: StandardHours, Over18
```

```
> apply(attrition.full, 2, function(x) length(unique(x)))
                       X                Attrition           BusinessTravel                DailyRate               Department
                    1470                        2                        3                      886                        4
        DistanceFromHome                Education           EducationField            EmployeeCount           EmployeeNumber
                      29                        5                        7                        1                     1470
 EnvironmentSatisfaction                   Gender               HourlyRate           JobInvolvement                 JobLevel
                       4                        2                       71                        4                        5
                 JobRole          JobSatisfaction            MaritalStatus            MonthlyIncome              MonthlyRate
                      10                        4                        3                     1349                     1427
      NumCompaniesWorked                   Over18                 OverTime        PercentSalaryHike        PerformanceRating
                      10                        2                        2                       16                        2
RelationshipSatisfaction            StandardHours         StockOptionLevel        TotalWorkingYears    TrainingTimesLastYear
                       4                        2                        4                       40                        8
         WorkLifeBalance           YearsAtCompany       YearsInCurrentRole  YearsSinceLastPromotion     YearsWithCurrManager
                       4                       37                       19                       16                       18
               BirthDate
                      43
> 
```

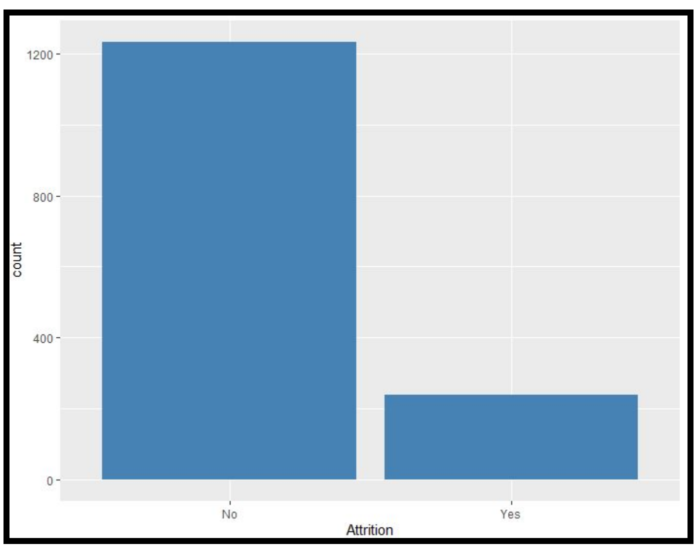
U postavci su naglašene varijable i njihove vrijednosti (**), na osnovu prethodnih dijelova koda i same postavke možemo zaključiti da to nije baš tako u data setovima i shodno tome je to potrebno ispraviti vrijednosti varijabli da bude zadovoljeno:

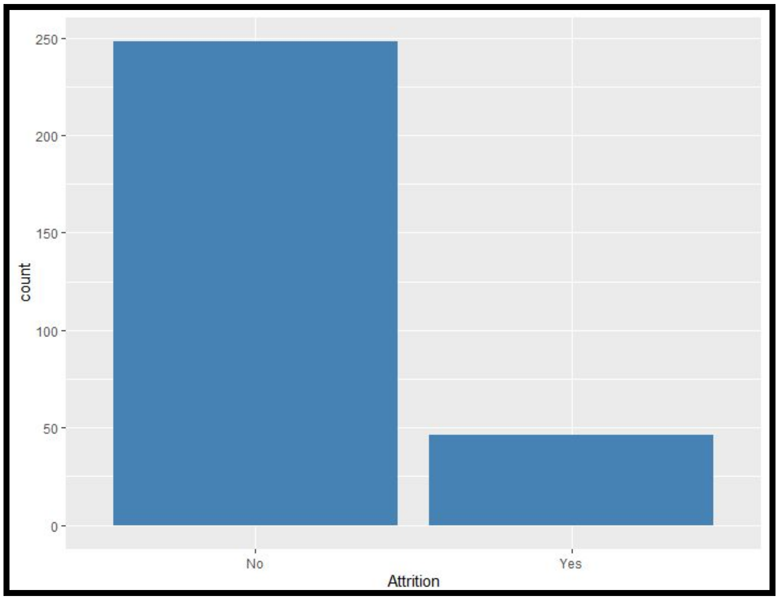
```
#JobLevel: One->1, Five->5
attrition.full$JobLevel <- revalue(attrition.full$JobLevel, c("One"=1,"Five"=5))
#StockOptionLevel: zero change into 0
attrition.full$StockOptionLevel <- revalue(attrition.full$StockOptionLevel, c("Zero"=0))
#EnvironmentSatisfaction: Medium -> 2
attrition.full$EnvironmentSatisfaction <- revalue(attrition.full$EnvironmentSatisfaction,
c("Medium"=2))
#Education: Master->4; Doctor->5
attrition.full$Education <- revalue(attrition.full$Education, c("Master"=4,"Doctor"=5))
#JobInvolvement: High->3
attrition.full$JobInvolvement <- revalue(attrition.full$JobInvolvement, c("High"=3))
#JobSatisfaction: Very High->4
attrition.full$JobSatisfaction <- revalue(attrition.full$JobSatisfaction, c("Very High"=4))
#RelationshipSatisfaction: Low->1, High->3
attrition.full$RelationshipSatisfaction <- revalue(attrition.full$RelationshipSatisfaction,
c("Low"=1,"High"=3))
```

Analizirajući ručno dataset, zaključeno je da ima više vrijednosti No u odnosu na Yes, shodno tome napravljena je provjera za oversampling na čitavom setu podataka. Ukoliko postoji oversampling, treba imati na umu prilikom izrade ML modela. Prvo ćemo vratiti prvobitne setove, te iscrtati grafove vezane za postojanje oversamplinga.

```
# Getting back the data for oversampling test
train_im <- attrition.full[1:LT,]
test_im<-attrition.full[(LT+1):1470,]

#Check if there is oversampling?
ggplot(data=attrition.full,aes(x=Attrition))+geom_bar( fill="steelblue")
ggplot(data=test_im,aes(x=Attrition))+geom_bar( fill="steelblue")
```

Iscrtani plotovi uz ggplot su naredni (ukupni set i test test respektivno):





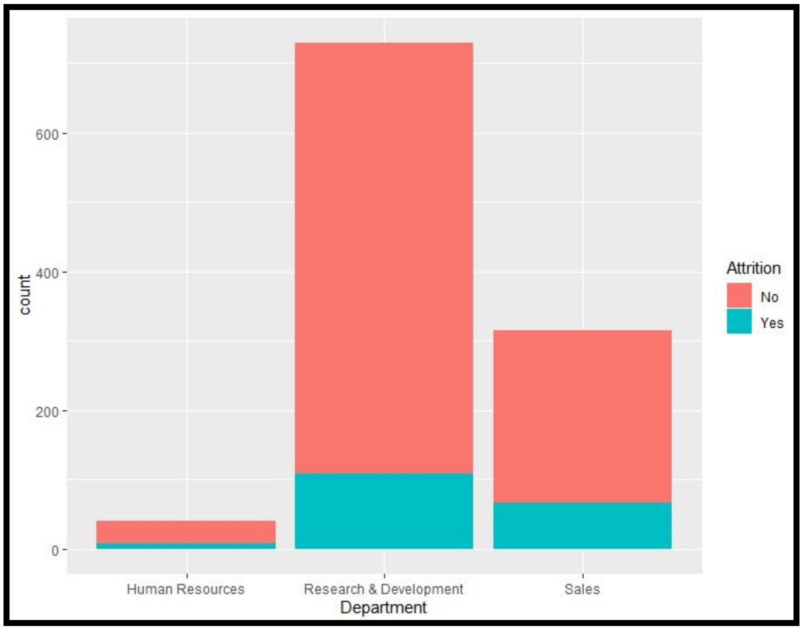
Vidimo da u zajedničkom (ujedno training i test set) je zastupljen oversampling. Vrijednost No u odnosu na Yes je dominantna. **Ovdje nam je kraj uzimanja podataka iz test seta, tj. test će se dalje koristiti samo za *re-valueing* tj. *promjena podataka i za finalno testiranje modela.***

Dalje, pošto imamo 3 kontinualne varijable, njih ćemo normalizirati koristeći min max normalizaciju iz razloga što će nam to omogućiti bliži raspon podataka i mogućnost bolje vrijednosti kasnije modela:

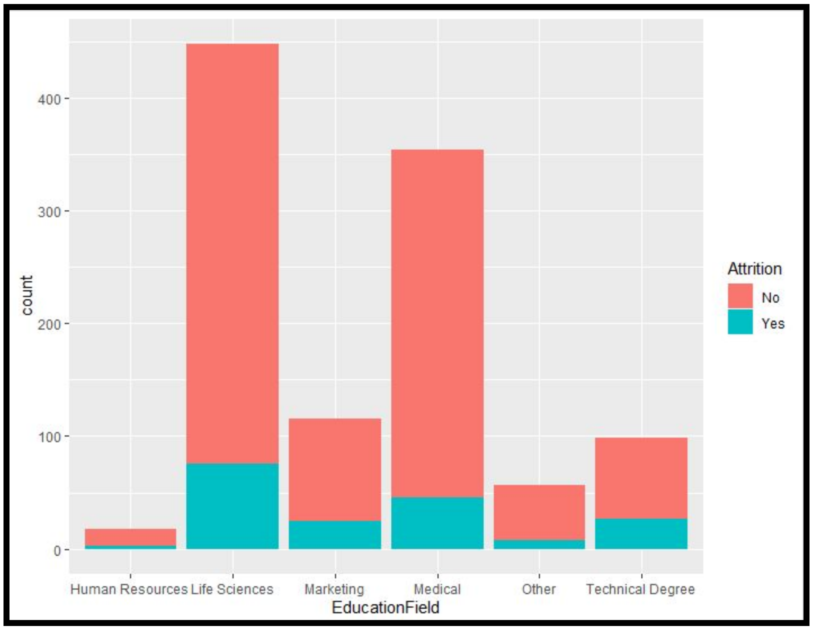
```
# Factor all others
cols<-c("JobLevel","JobRole","StockOptionLevel","EnvironmentSatisfaction","Education","JobInvo
lvement","JobSatisfaction","RelationshipSatisfaction","PerformanceRating","WorkLifeBalance","O
verTime","MaritalStatus","Gender","Department","BusinessTravel","Attrition")
for (i in cols){
  attrition.full[ ,i] <- as.factor(attrition.full[ ,i])
}

# Normalisation
# it doesn’t effect that much,
# so it can be avoided, but we did it also (every little bit helps :) )
normalize <- function(x) {
  return ((x - min(x)) / (max(x) - min(x)))
}

attrition.full$DailyRate<-normalize(attrition.full$DailyRate)
attrition.full$MonthlyRate<-normalize(attrition.full$MonthlyRate)
attrition.full$MonthlyIncome<-normalize(attrition.full$MonthlyIncome)
```

U nastavku ćemo ispitati odnose varijabli koje imaju nedostajuće vrijednosti sa varijablom Attrition, i njihov međusobni odnos, treba napomenuti da još uvijek nismo fillovali varijable sa NA vrijednostima. Razlog tome je ušteda vremena i kucanja (cost effect value), te ćemo postepeno fillovati nedostajuće vrijednosti ukoliko se ukaže potreba (bude uticaj ogroman tih NA vrijednosti ili ih ima dosta) za tim u narednim grafovima. Prvenstveno ćemo ispitati korelacije između tih istih varijabli kroz naredne iteracije, a poslije toga ispitati relevantnost nedostajućih vrijednosti (količina i odnos) kroz grafove.

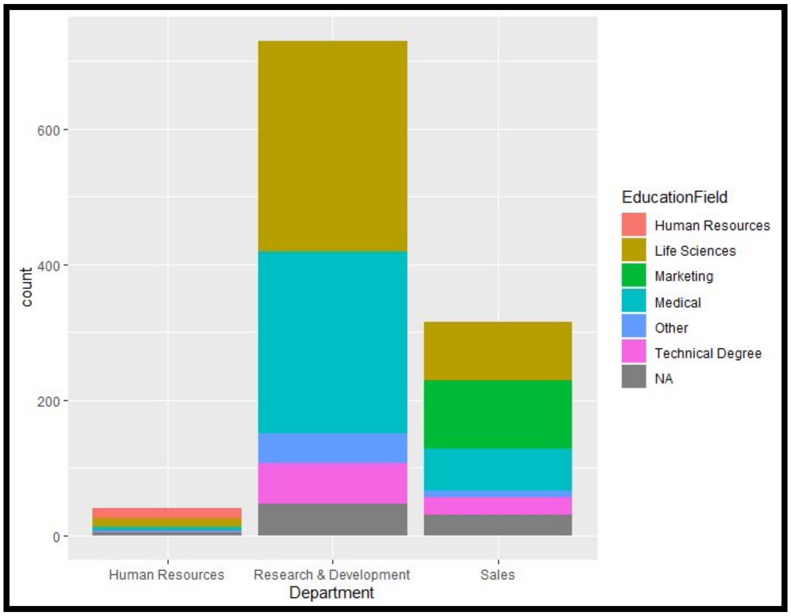
```
# Check if the empty variables are relevant
# Department VS Attrition
ggplot(data =
attrition.train[!is.na(attrition.train$Department),],aes(x=Department,fill=Attrition))+geom_ba
r()
```



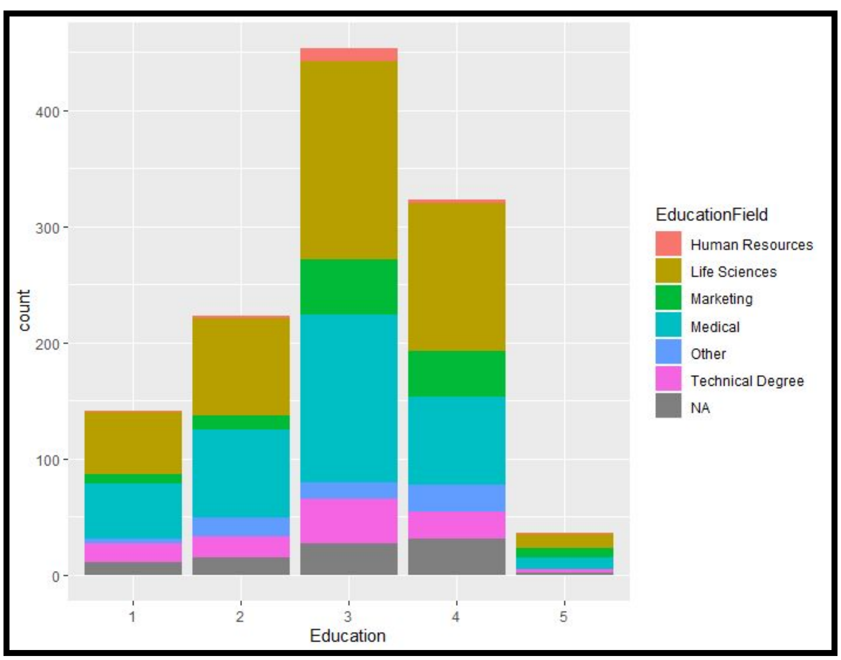

```
# EducationField VS Attrition
ggplot(data =
attrition.train[!is.na(attrition.train$EducationField),],aes(x=EducationField,fill=Attrition))
+geom_bar()
```



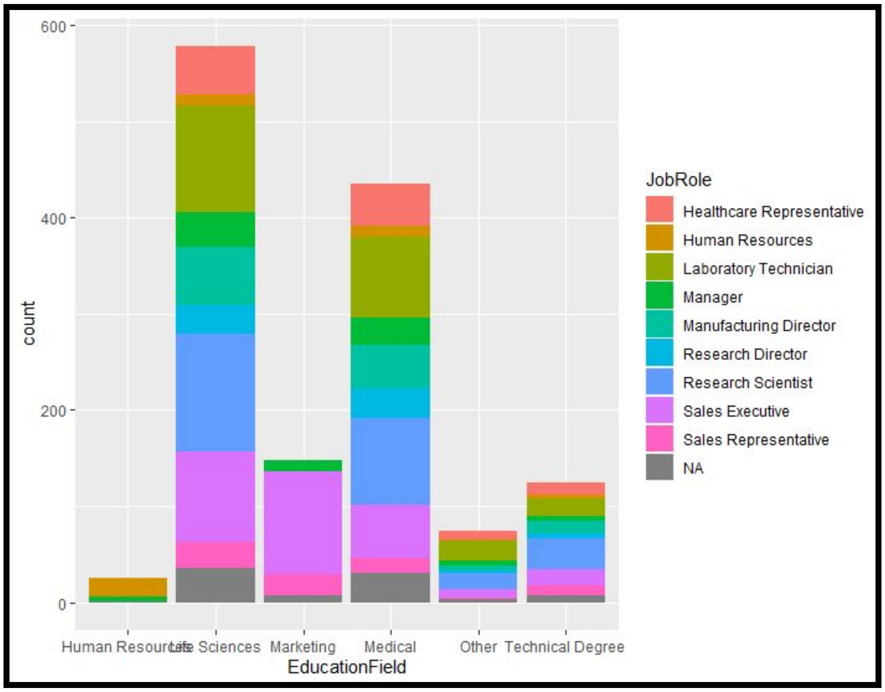
```
# Department VS EducationField
ggplot(data =
attrition.train[!is.na(attrition.train$Department),],aes(x=Department,fill=EducationField))+ge
om_bar()
# Not correlated as expected
```



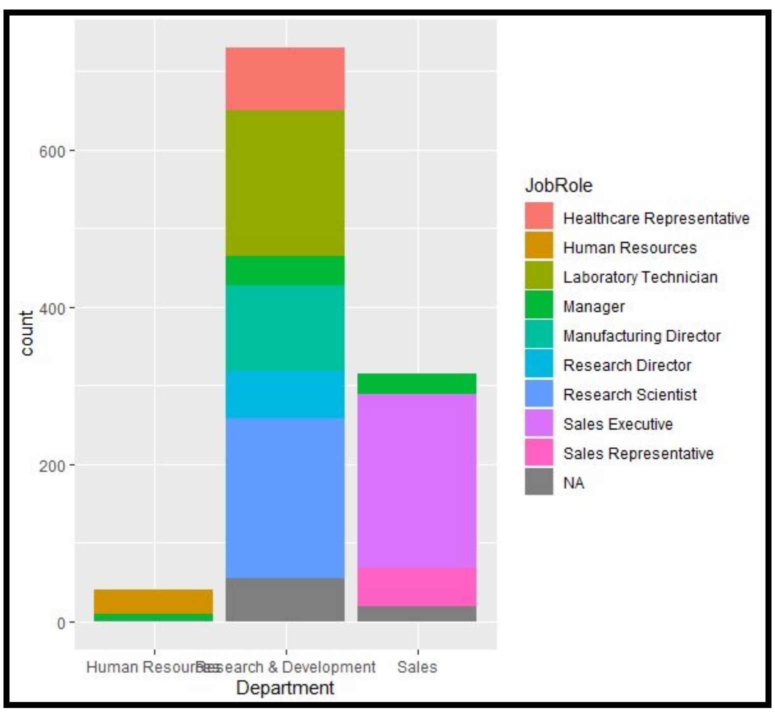

```
# Education VS EducationField
ggplot(data =
attrition.train[!is.na(attrition.train$Education),],aes(x=Education,fill=EducationField))+geom
_bar()
# Not correlated
```



```
# JobRole VS EducationField
ggplot(data =
attrition.train[!is.na(attrition.train$EducationField),],aes(x=EducationField,fill=JobRole))+g
eom_bar()
# Not so correlated
```

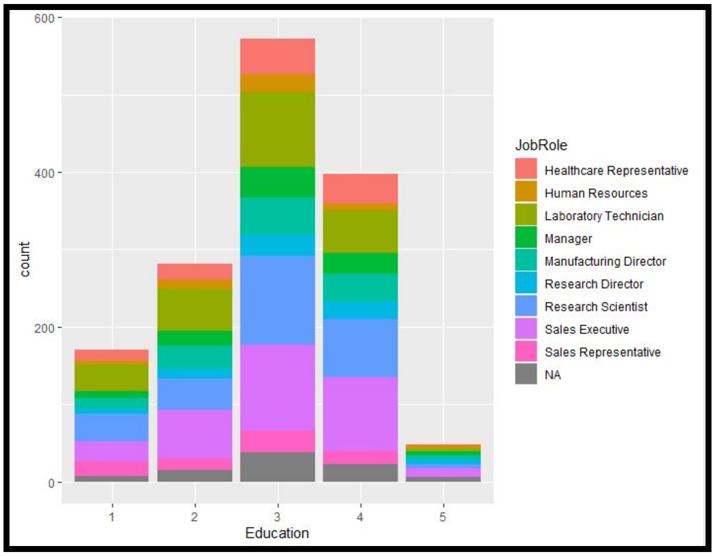



```
# Department VS JobRole
ggplot(data =
attrition.train[!is.na(attrition.train$Department),],aes(x=Department,fill=JobRole))+geom_bar(
)
# Correlated, we shall check what variable to choose
```

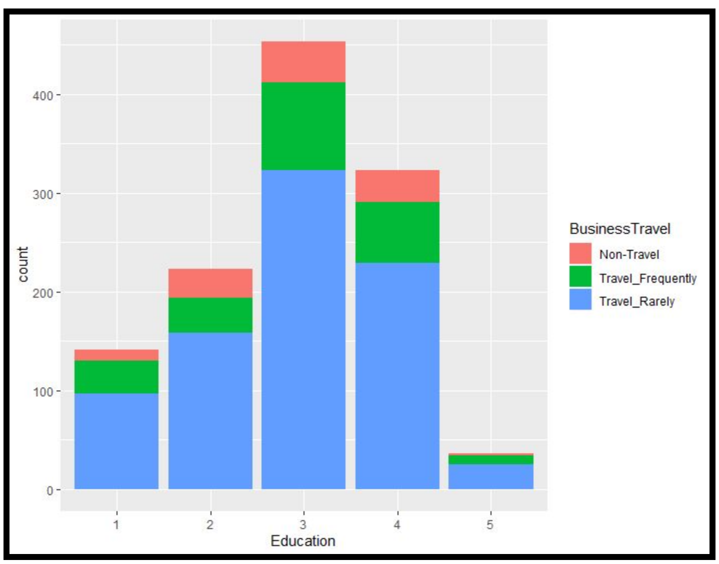


Razlog korelacije predstavlja istojednako ponašanje obje varijable u dosta slučajeva, tj. npr ukoliko je osoba Healthcare representative ona je ujedno i research & department dio varijable departmenta te se može zanemariti određeni dio, dok npr Sales Executive je dio samo Sales i nema veze sa ostalim. Koju ćemo zanemariti varijablu ovisi od ispitivanja i ostalih varijabli koje imaju NA vrijednosti sa ove dvije.

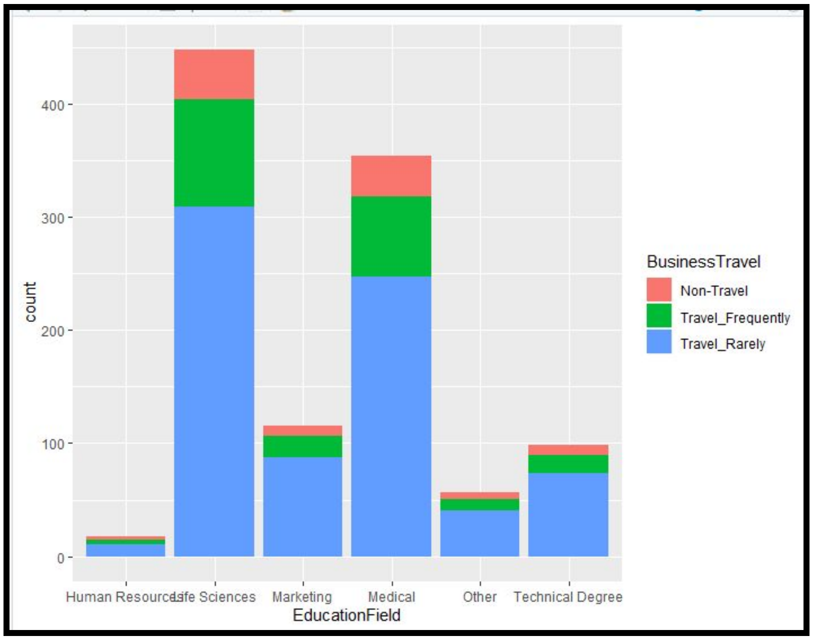
```
# JobRole VS Education
ggplot(data =
attrition.train[!is.na(attrition.train$Education),],aes(x=Education,fill=JobRole))+geom_bar()
# Not correlated
```



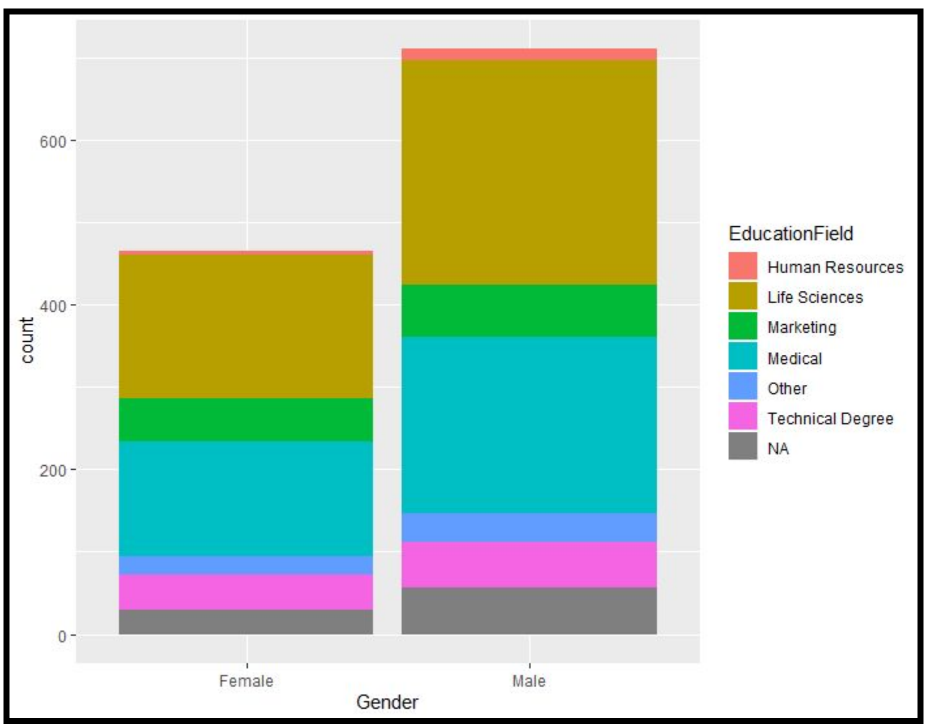

```
# Education VS BusinessTravel
ggplot(data =
attrition.train[!is.na(attrition.train$Education),],aes(x=Education,fill=BusinessTravel))+geom
_bar()
# Not correlated
```



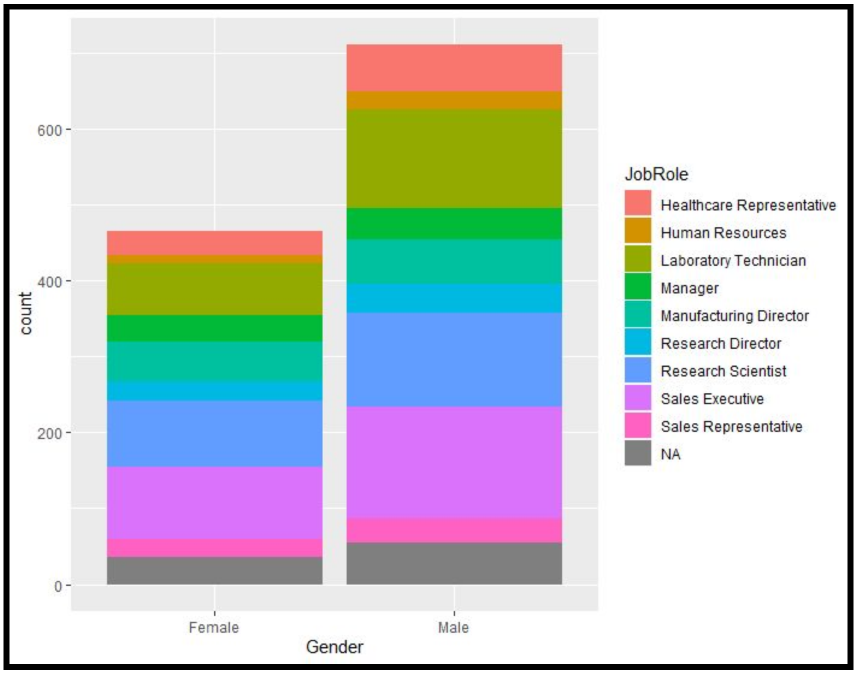
```
# EducationField VS BusinessTravel
ggplot(data =
attrition.train[!is.na(attrition.train$EducationField),],aes(x=EducationField,fill=BusinessTra
vel))+geom_bar()
# Not correlated
```



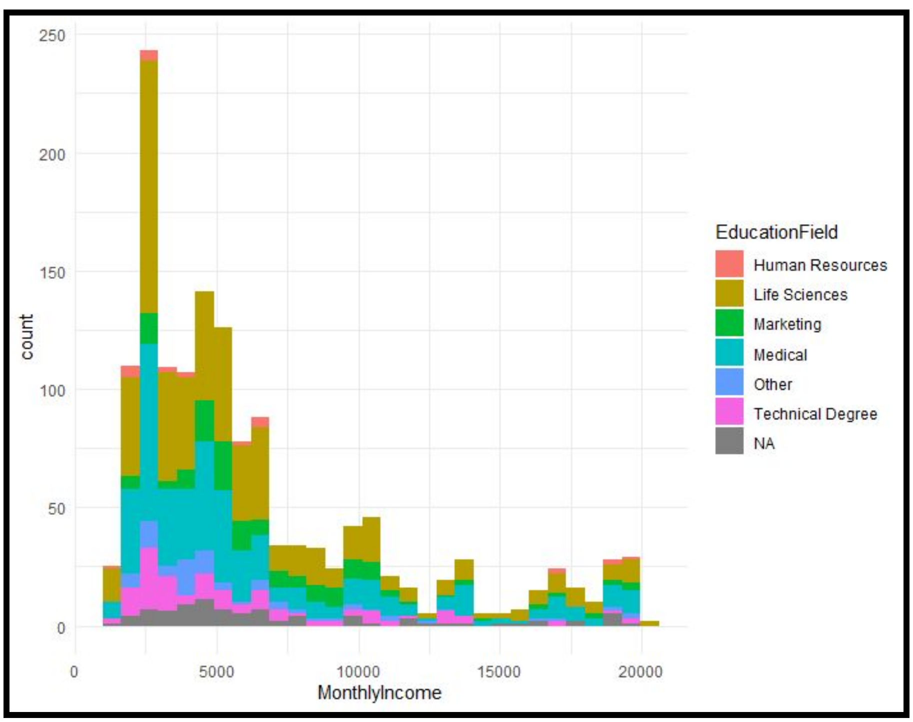

```
# EducationField VS Gender
ggplot(data =
attrition.train[!is.na(attrition.train$Gender),],aes(x=Gender,fill=EducationField))+geom_bar()
# Not correlated
```



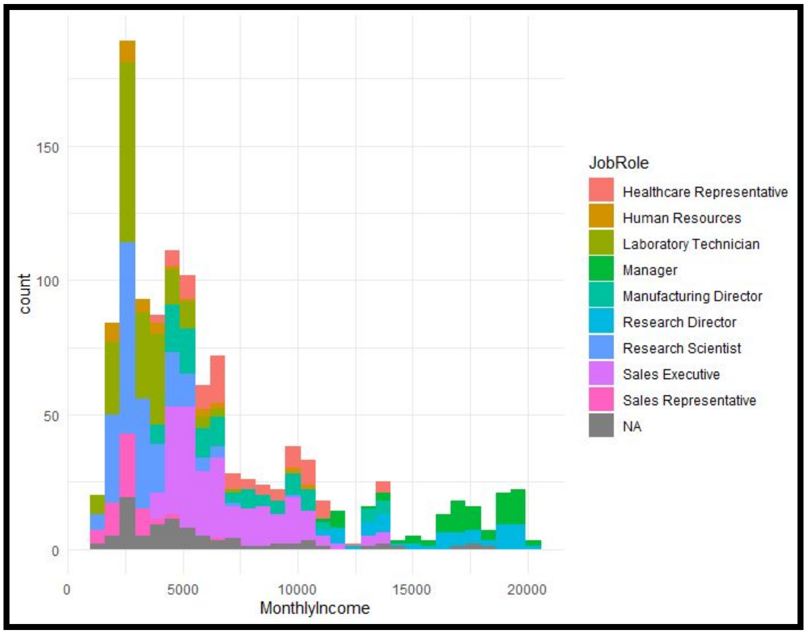
```
# JobRole VS Gender
ggplot(data =
attrition.train[!is.na(attrition.train$Gender),],aes(x=Gender,fill=JobRole))+geom_bar()
# Not correlated
```




```
# EducationField VS MonthlyIncome
ggplot(data =
attrition.train,aes(x=MonthlyIncome,fill=EducationField))+geom_histogram()+theme_minimal()
# Not correlated
```

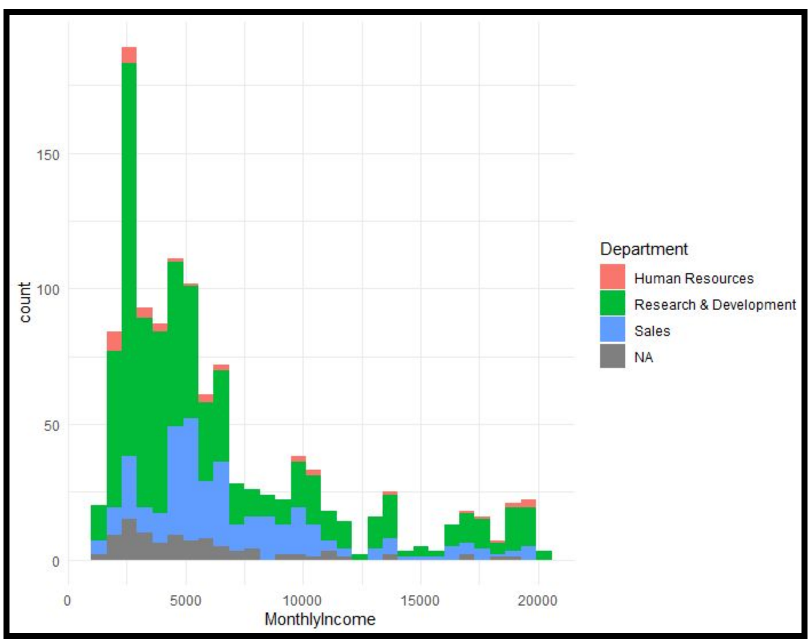


```
# JobRole VS MonthlyIncome
ggplot(data =
attrition.train,aes(x=MonthlyIncome,fill=JobRole))+geom_histogram(bins=30)+theme_minimal()
# Correlated, therefore, we will use MonthlyIncome and not JobRole since it was already
collerated with Department.
```

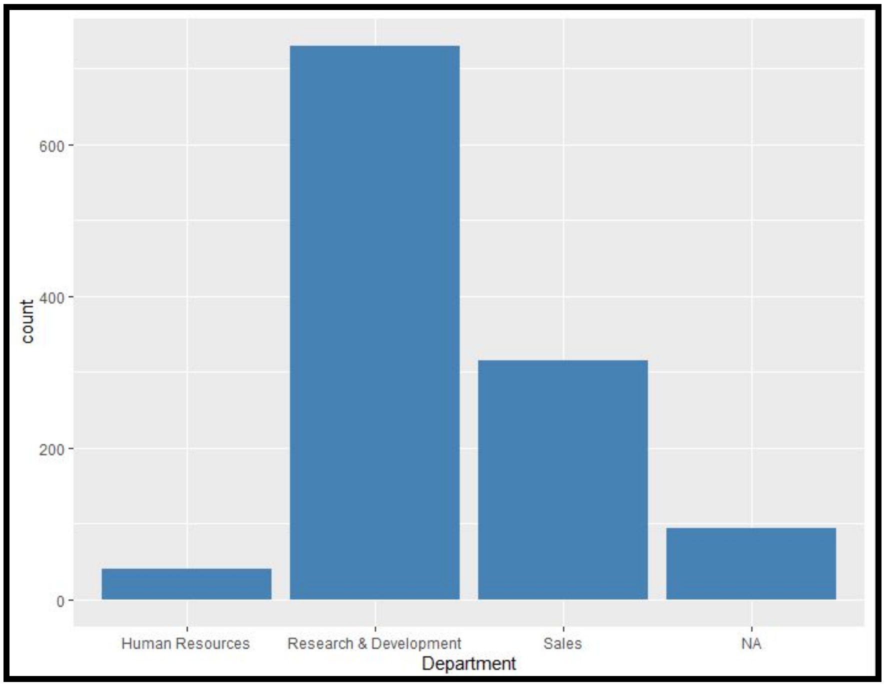


Izabrali smo varijablu Department u odnosu na JobRole, zato sto je ona u vise korelacija u odnosu na Department. Ujedno ćemo popuniti NA vrijednosti Department sa MFV (Most Frequent Value).

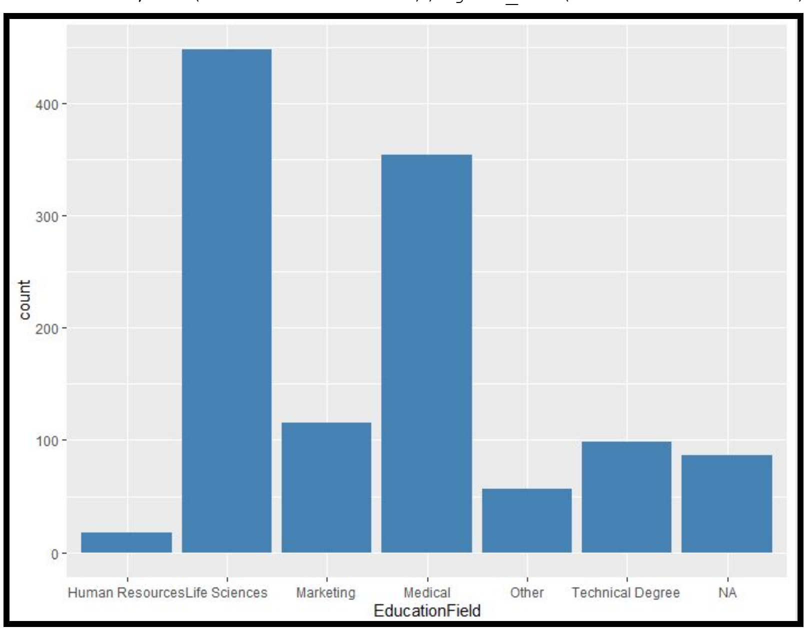
```
# Department VS MonthlyIncome
ggplot(data =
attrition.train,aes(x=MonthlyIncome,fill=Department))+geom_histogram(bins=30)+theme_minimal()
# Not correlated -> Department should be filled with data
```



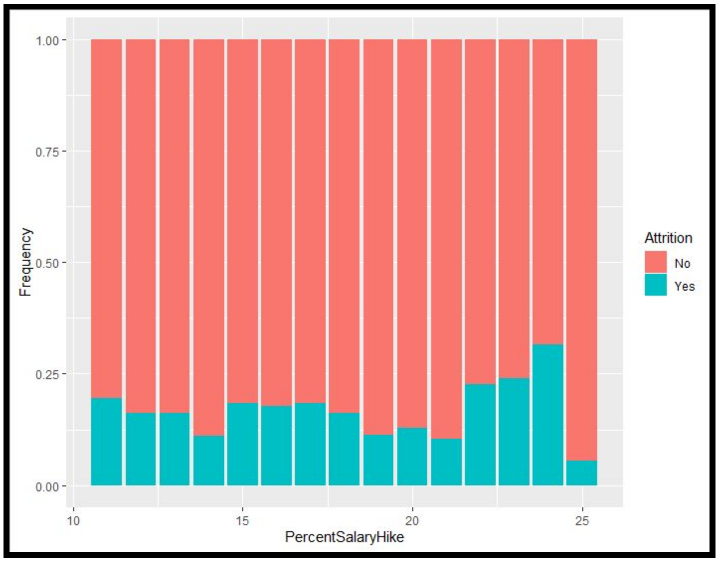
```
# Filling Department values
ggplot(data=attrition.train,aes(x=Department))+geom_bar( fill="steelblue")
```




```
# Number of the NA is not small enough,
# Fill empty fields with the most frequent "Research & Development"
attrition.full$Department[is.na(attrition.full$Department)==TRUE] <- "Research & Development"

# Also checking the count for EducationField
ggplot(data=attrition.train,aes(x=EducationField))+geom_bar( fill="steelblue")
```



```
# Number is not small, filling EducationField - most frequent
attrition.full$EducationField[is.na(attrition.full$EducationField)==TRUE] <-
as.factor("LifeScience")
attrition.full[ ,"EducationField"] <- as.factor(attrition.full[ ,"EducationField"])
# PercentSalaryHike vs Attrition
ggplot(data =
attrition.train[!is.na(attrition.train$PercentSalaryHike),],aes(x=PercentSalaryHike,fill=Attri
tion))+geom_bar(position="fill")+ylab("Frequency")
```



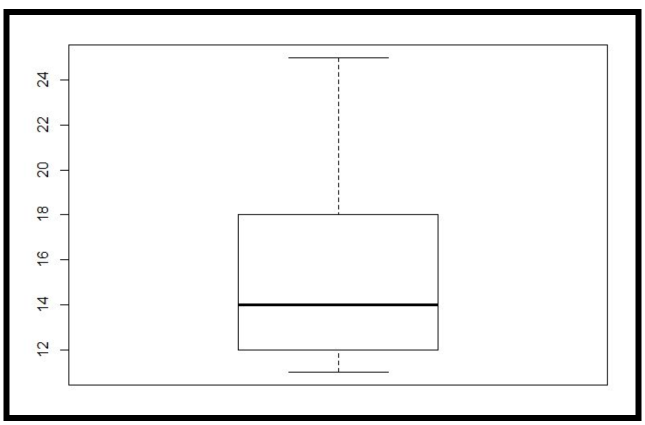
Iz prošlog grafika se uočava nešto veoma specifično, a to je da ljudi sa većinim procentom planiraju napustiti kompaniju. Vidimo da je ovaj procenat najveći na 22 do 24 procenta i najmanji na 25-om procentu. Narednim kodom možemo prikazati njihove karakteristike.

```
# People for whom Attrition=Yes is largest:
filter(attrition.train, PercentSalaryHike == 24)
# All of them have: PerformanceRating=Outstanding & BirthDate in March & mostly
Department=Research & Development

# People for whom Attrition=Yes is smallest:
filter(attrition.train, PercentSalaryHike == 25)
# All of them have: PerformanceRating=Outstanding & BirthDate in March & mostly
Department=Research & Development
```

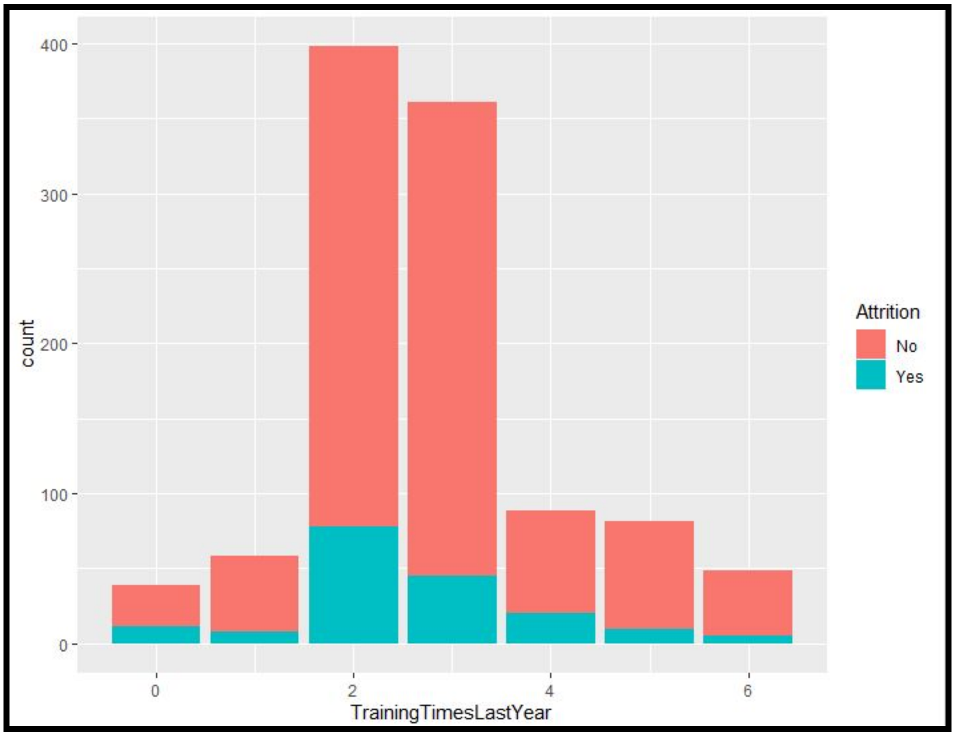
Prethodni podaci filtera (nisu prikazani radi preglednosti, ali generalno ukazuju da odredjeni set podataka se ponavlja za 24 i 25 PercentSalaryHike sto znaci da taj set podataka nije u korelaciji sa ovom varijablom) i grafik ukazuju da je bitno odnos ovog procenta na ljude koji planiraju otići. Pored toga ukazuju da se najviše gledaju varijable PerformanceRating, Department, BirthDate, Gender, WorkLifeBalance i JobInvolvment (kada se filter gornji u terminalu prikaže najviše ima ovih vrijednosti tj. ove varijable najviše utiču na nasu varijablu tj. ponavljaju se).

Na osnovu relativno sličnih podataka za obje varijable možemo pretpostaviti da ova varijabla i varijabla Attrition su u malo većoj korelaciji, dok ova varijabla nije u korelaciji sa ostalim varijablama na osnovu filter podataka.
Shodno tome vrijednosti ovih podataka je bitna, te ćemo popuniti linearnom regresijom (iako nije rađeno na predmetu još, ali predstavlja solidnu varijantu popunjava tražene vrijednosti jer će vrijednost ove varijable biti bitna u fazi odlučivanja).

```
# PercentSalaryHike is relevant
boxplot(attrition.train$PercentSalaryHike) # boxplot ispod
boxplot.stats(attrition.full$PercentSalaryHike)
upper.whisker <- boxplot.stats(attrition.full$PercentSalaryHike)$stats[5]
outlier.filter <- attrition.full$PercentSalaryHike < upper.whisker
PercentSalaryHike.equation = "PercentSalaryHike ~ PerformanceRating + Department +
BirthDate+Gender+WorkLifeBalance+JobInvolvement"
PercentSalaryHike.model <- lm(
  formula = PercentSalaryHike.equation,
  data = attrition.full[outlier.filter,]
)
PercentSalaryHike.row <- attrition.full[
  is.na(attrition.full$PercentSalaryHike),
  c("PerformanceRating", "Department",
"BirthDate","Gender","WorkLifeBalance","JobInvolvement")
  ]
PercentSalaryHike.predictions <- predict(PercentSalaryHike.model,
newdata=PercentSalaryHike.row)
attrition.full[is.na(attrition.full$PercentSalaryHike), "PercentSalaryHike"] <-
PercentSalaryHike.predictions
```




```
# TrainingTimesLastYear vs Attrition
ggplot(data =
attrition.train[!is.na(attrition.train$TrainingTimesLastYear),],aes(x=TrainingTimesLastYear,fi
ll=Attrition))+geom_bar()
```



Na osnovu prethodnih podataka, vidimo i ovdje da broj NA vrijednosti nije zanemariv i treba ih zamijeniti sa MFV:

```
#Filling TrainingTimesLastYear - most frequent
attrition.full$TrainingTimesLastYear[is.na(attrition.full$TrainingTimesLastYear)==TRUE] <- 2
#attrition.full[ ,TrainingTimesLastYear] <- as.factor(attrition.full[ ,TrainingTimesLastYear])
```

Također trebamo BirthDate varijablu pretvoriti u numeric, jer ovako ima previše vrijednosti da bi se gledala kao kategorička.

```
# Converting BirthDate into numeric
library(lubridate)
attrition.full$BirthDate <- ymd(attrition.full$BirthDate)
```

Na kraju ćemo iz seta podataka izbaciti JobRole radi korelacija i varijable koje, gledajući u početni set, imaju iste vrijednosti u svim instancama

```
attrition.full<-select(attrition.full,-c(StandardHours,Over18,JobRole,EmployeeCount,EmployeeNu
mber,X))
```

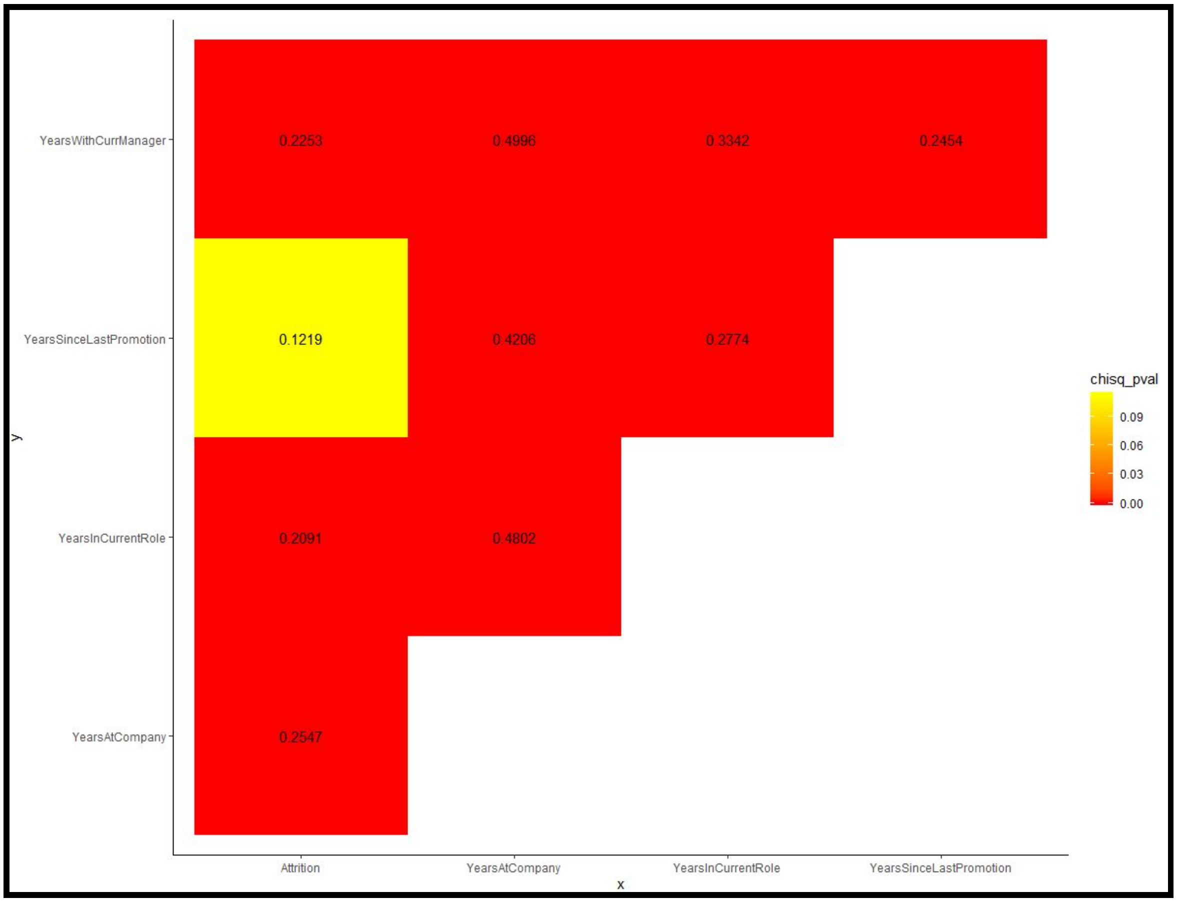
i naravno ponoviti podijele setova opet.

(Napomena : podjela setova i refreshanje podatak se desila nekoliko puta tokom iteracija, tako da su uvijek relevantni podaci)

Dosad smo napravili korelacije za vrijednosti u kojima su falile vrijednosti i napravili korelacije, dalje za sve ostale varijable koje su ostale u setu ćemo napraviti korelacijske matrice i na osnovu grafika tih matrica odlučiti koje varijable treba izbaciti, a koje ne (to ćemo uraditi korištenje chi square metode). U ovoj metodi što je chi square value manja to je varijabla u korelaciji sa Attrition (što ujedno i tražimo). Sada pravimo heat mape za sve ostale varijable iz train seta.

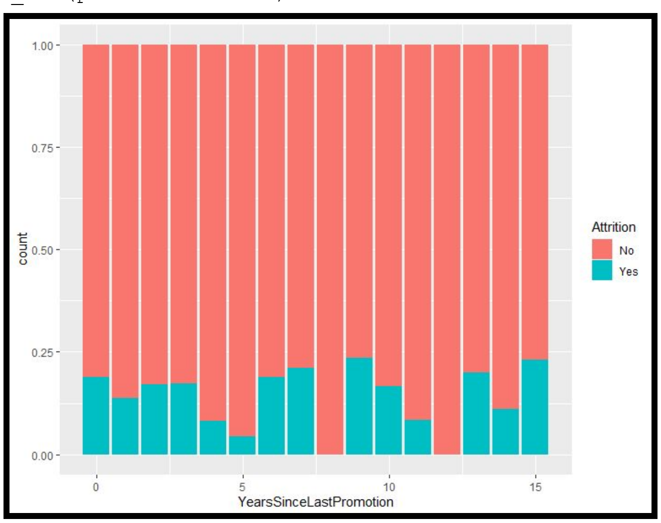
```
# Other Correlations
attrition.full2<-select(attrition.train,c(YearsWithCurrManager,YearsSinceLastPromotion,YearsIn
CurrentRole,YearsAtCompany,Attrition))
df<-data.frame(attrition.full2)
attrition.full3<-select(attrition.train,c(StockOptionLevel,RelationshipSatisfaction,Performanc
eRating,PercentSalaryHike,OverTime,Attrition))
df2<-data.frame(attrition.full3)
attrition.full4<-select(attrition.train,c(JobLevel,JobInvolvement,HourlyRate,Gender,Environmen
tSatisfaction,EducationField,Education,Attrition))
df3<-data.frame(attrition.full4)
attrition.full5<-select(attrition.train,c(WorkLifeBalance,TrainingTimesLastYear,TotalWorkingYe
ars,JobSatisfaction,MonthlyIncome,MaritalStatus,Attrition))
df4<-data.frame(attrition.full5)
attrition.full6<-select(attrition.train,c(NumCompaniesWorked,MonthlyRate,DistanceFromHome,Depa
rtment,DailyRate,BusinessTravel,BirthDate,Attrition))
df5<-data.frame(attrition.full6)
# For every generated df, do correlation plot (change df into df, df2, df3, df4 then df5.) !!!
# Red values indicated high correlations (lower values)
library(tidyverse)
library(lsr)
library(purrr)


f = function(x,y) {
  tbl = df %>% select(x,y) %>% table()
  chisq_pval = round(chisq.test(tbl)$p.value, 4)
  cramV = round(cramersV(tbl), 4)
  data.frame(x, y, chisq_pval, cramV) }

# create unique combinations of column names
# sorting will help getting a better plot (upper triangular)
df_comb = data.frame(t(combn(sort(names(df)), 2)), stringsAsFactors = F)

# apply function to each variable combination
df_res = map2_df(df_comb$X1, df_comb$X2, f)
# plot results
df_res %>%
  ggplot(aes(x,y,fill=chisq_pval))+
  geom_tile()+
  geom_text(aes(x,y,label=cramV))+
  scale_fill_gradient(low="red", high="yellow")+
  theme_classic()
```

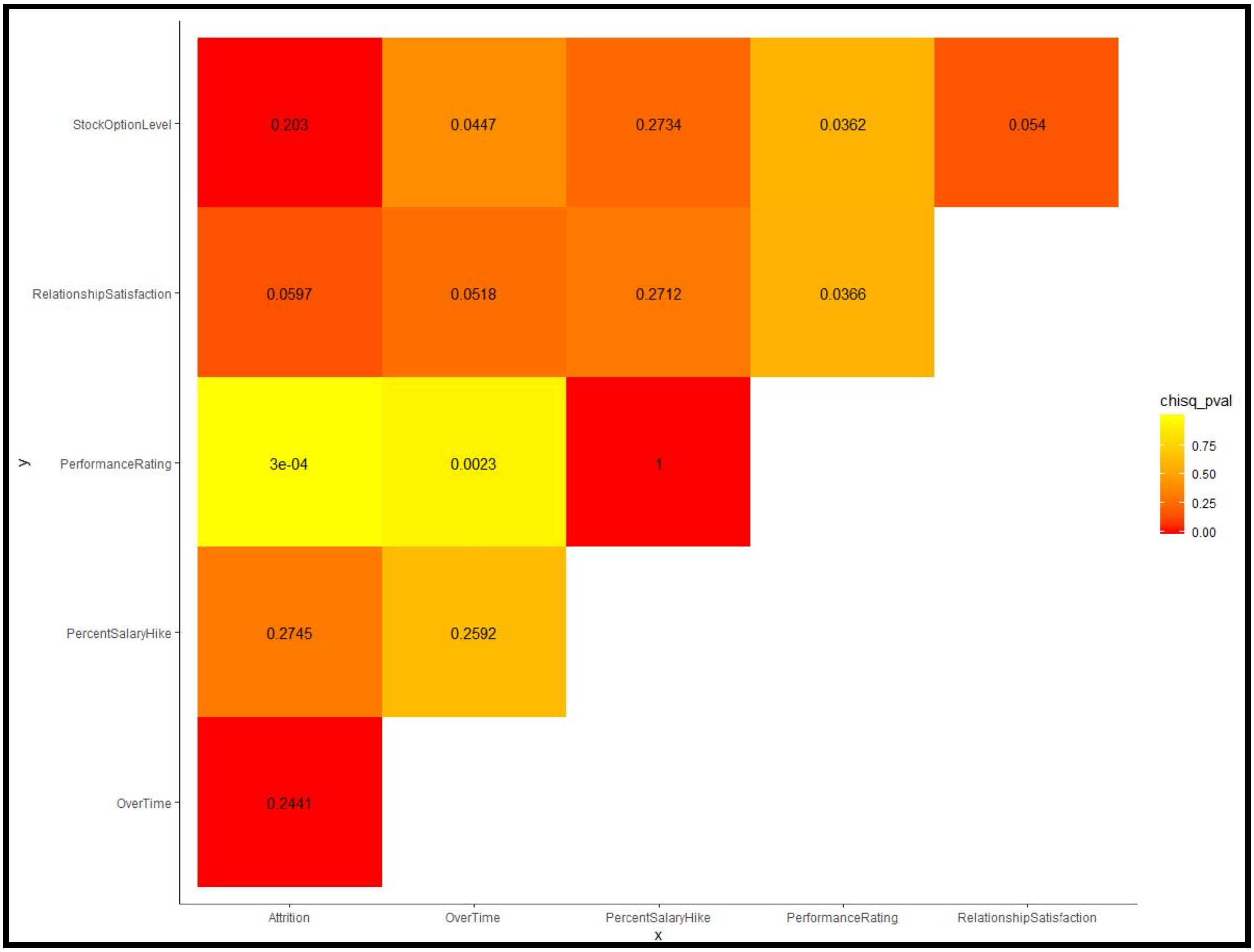
Shodno prethodnim kodom možemo 5 puta izgenerisati korelacijske matrice i vidjeti koje vrijednosti treba izabrati za naš model:



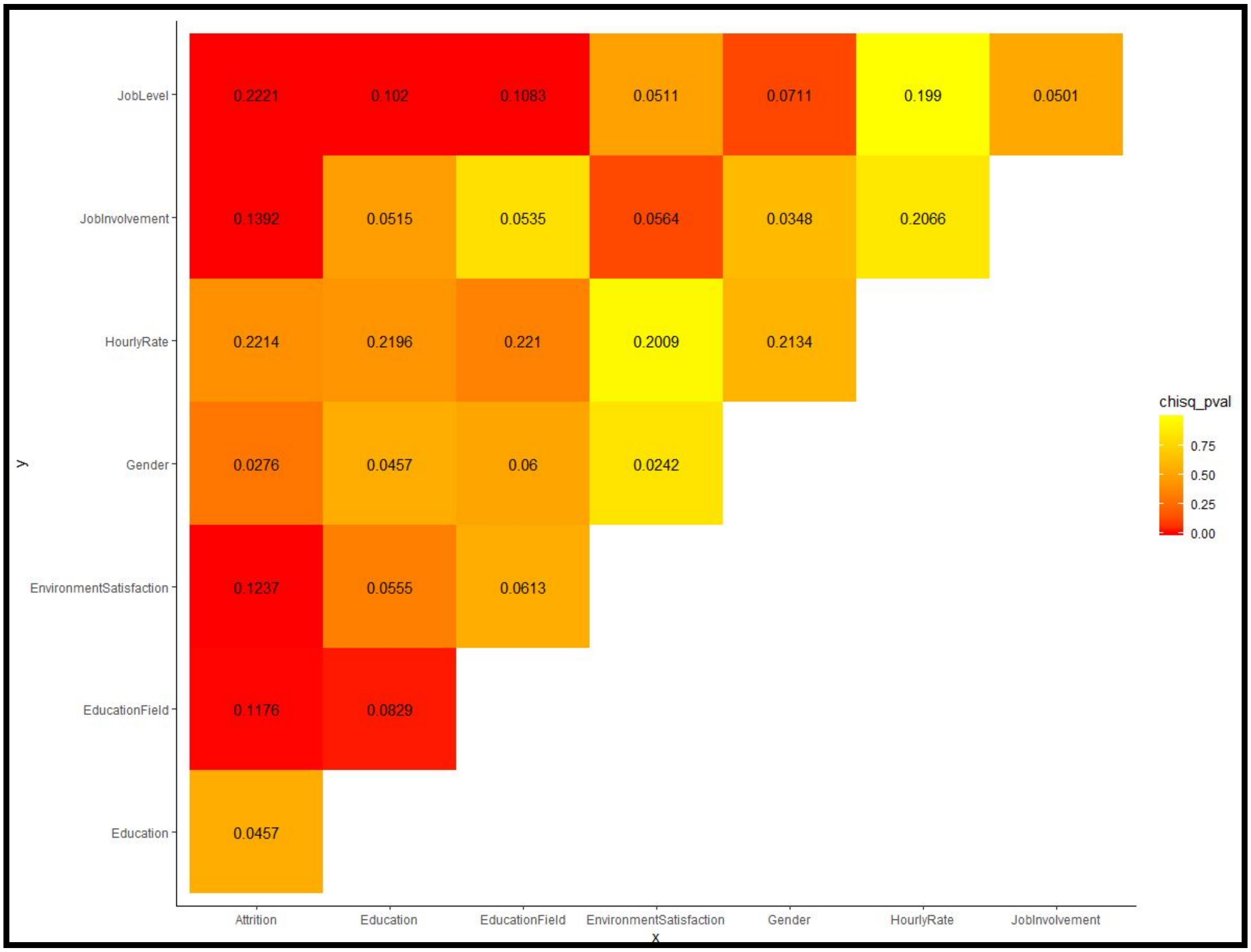
Sve varijable osim YearsSinceLastPromotion su pogodne za korištenje u modelu, dok ova ima visoku vrijednost chi-square.

```
ggplot(data =
attrition.train[!is.na(attrition.train$YearsSinceLastPromotion),],aes(x=YearsSinceLastPromotio
n,fill=Attrition))+geom_bar(position="fill")
```

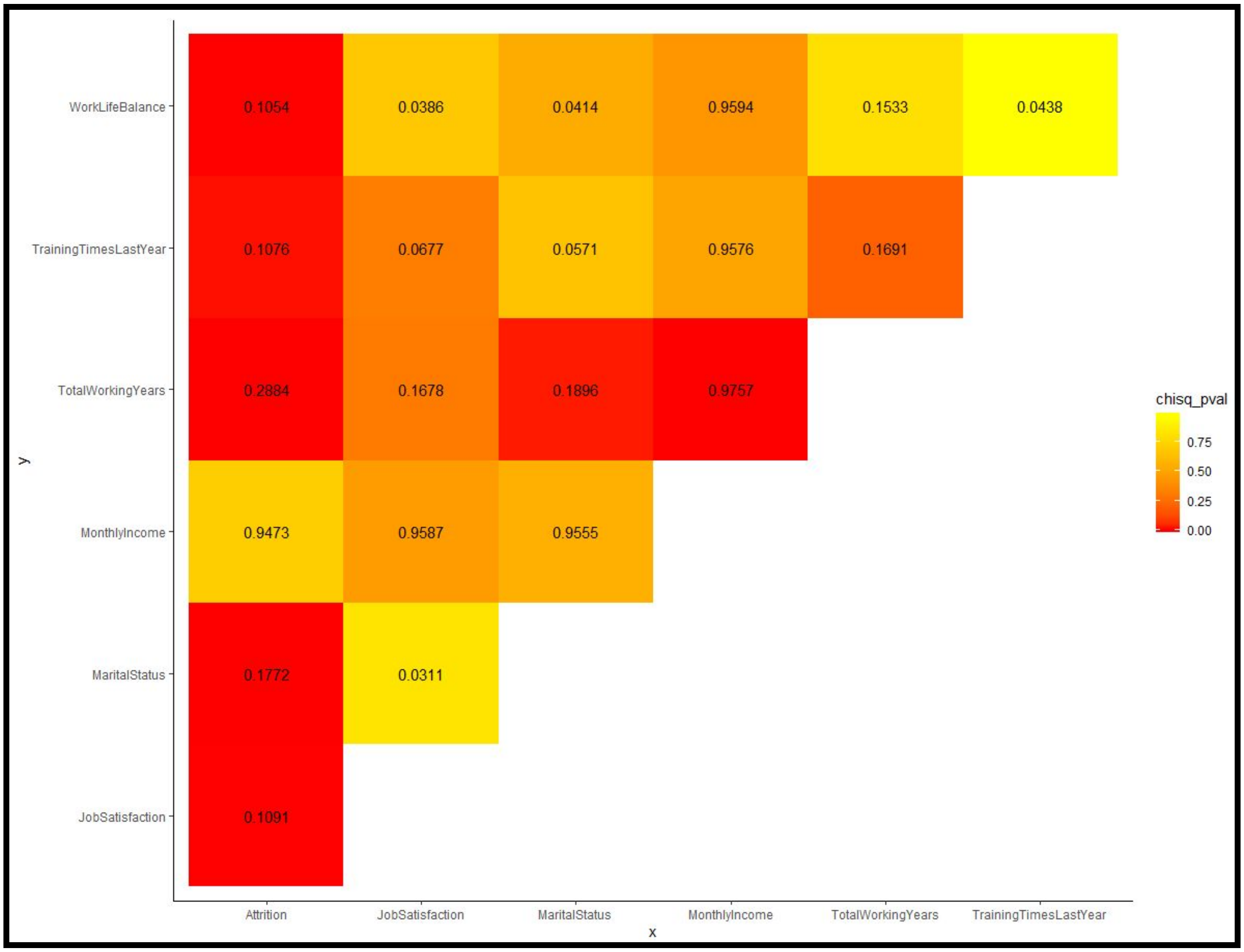




Vidimo da PerformanceRating nije u korelaciji sa Attrtition i nju ne trebamo koristi u modelu. Također vidimo da se u PerformanceRating i PercentSalaryHike u korelaciji te nju možemo izbaciti iz modela. Dobre varijable za naš ML model su : OverTime, PercentSalaryHike i StockOptionLevel.

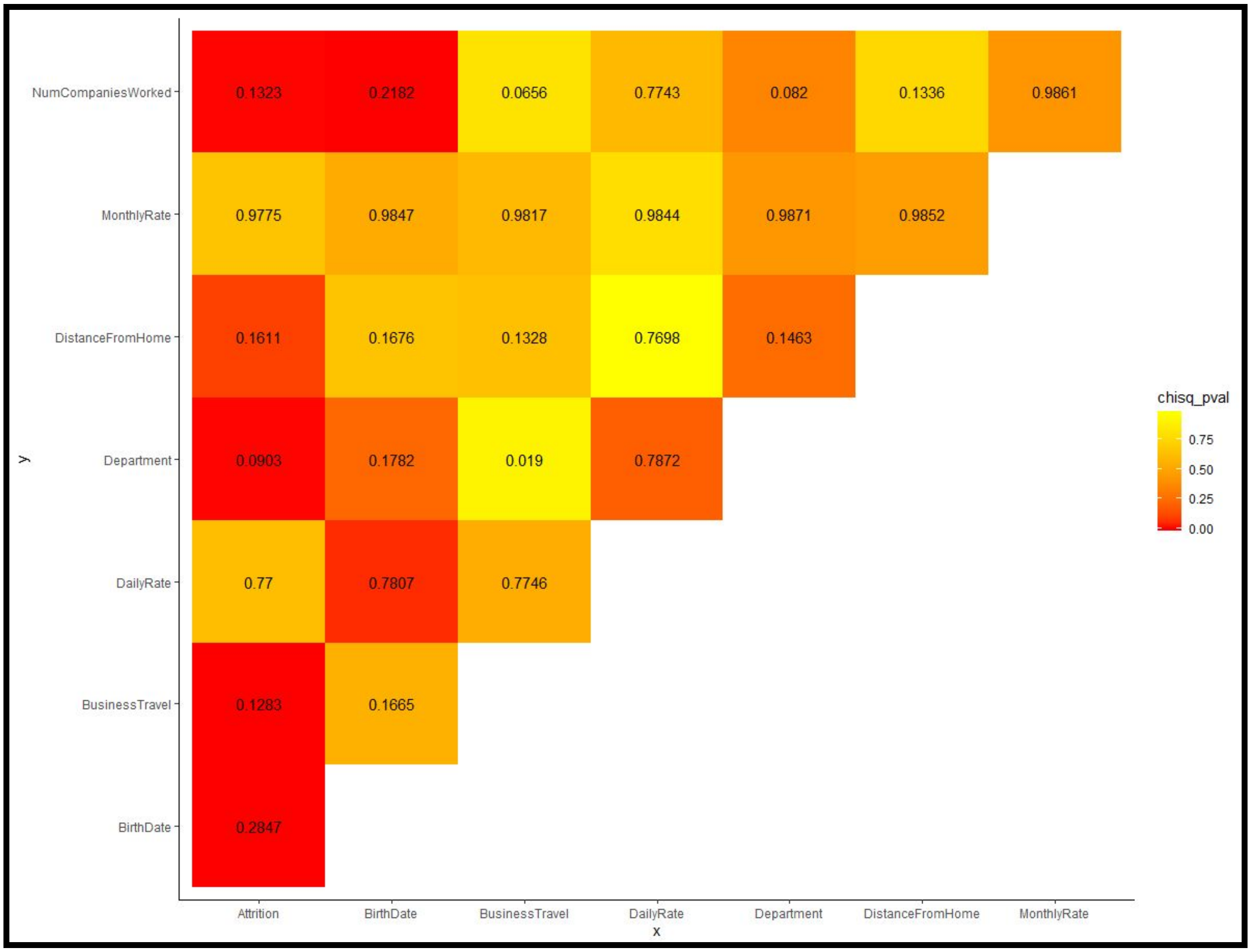


Vidimo da su dobre varijable za naš ML model : JobLevel, JobInvolvment, EnvironmentSatisfction, EducationField.



Vidimo da su dobre varijable za naš ML model: WorkLifeBalance, TrainingTimesLastYear, TotalWorkingYears, JobSatisfaction, MaritalStatus.

Također vidimo da MonthlyIncome nije u korelaciji i da TotalWorkingYears & MonthlyIncome su u korelaciji tako da MonthlyIncome nećemo uključivati u model.



Vidimo da su dobre varijable za naš ML model: NumCompaniesWorked, DistanceFromHome, Department, BusinessTravel. Dok također vidimo da su DailyRate & NumCompaniesWorked u korelaciji sa BirthDate. Shodno tome DailyRate i BirthDate nećemo uzimati u obzir za naš ML model.

Također dosadašnji paketi koji su korišteni (ukoliko nisu prikazani u nekim dijelovima koda):

```
library(ggplot2)
library(plyr)
library(dplyr)
library(GGally)
library(randomForest)
library(caret)
library(corrplot)
library(RColorBrewer)
```

Na kraju kad posložimo sve i sumiramo imamo 19 varijabli na osnovu kojih možemo praviti dalje modele: NumCompaniesWorked, DistanceFromHome, Department, BusinessTravel, WorkLifeBalance, TrainingTimesLastYear, TotalWorkingYears, JobSatisfaction, MaritalStatus, JobLevel, HourlyRate, JobInvolment, EnvironmentSatisfaction, EducationField, OverTime, PercentSalaryHike, StockOptionLevel, YearsWithCurrManager, YearsInCurrentRole, YearsAtCompany.

Kod za formiranje finalnih setova podataka koji će se koristiti za kreiranje modela su sljedeći (koristimo i trening set i set podataka) koji su sada očiščeni, a zadržane su iste informacije.

```
#Conclusion: Variables for ML: correlated with Attrition but not so with other variables

train_im <- attrition.full[1:LT,
c("Attrition","NumCompaniesWorked","DistanceFromHome","Department","BusinessTravel","WorkLifeBalance","TrainingTimesLastYear","TotalWorkingYears","JobSatisfaction","MaritalStatus","JobLevel","HourlyRate","JobInvolvement","EnvironmentSatisfaction","EducationField","OverTime","PercentSalaryHike","StockOptionLevel","YearsWithCurrManager","YearsInCurrentRole","YearsAtCompany")]

test_im<-attrition.full[(LT+1):1470,c("Attrition","NumCompaniesWorked","DistanceFromHome","Department","BusinessTravel","WorkLifeBalance","TrainingTimesLastYear","TotalWorkingYears","JobSatisfaction","MaritalStatus","JobLevel","HourlyRate","JobInvolvement","EnvironmentSatisfaction","EducationField","OverTime","PercentSalaryHike","StockOptionLevel","YearsWithCurrManager","YearsInCurrentRole","YearsAtCompany")
```

Izgradite najmanje tri predikcijska modela, a koja trebaju biti iz porodice modela “drvo odlučivanja”, tj. metoda klasičnog drveta odlučivana (paketi tree ili C50), metode bagging i random forests (paketi randomForest i C50), ili metodu boosting za drvo odlučivanja (paket gbm i istoimena funkcija).

Kako biste istrenirali, testirali i uporedili vaše modele, potrebno je da koristiti najmanje tri metode unakrsne validacije (holdout, cross-validation, bootstrapping, itd).

Dokumentujte proces izgradnje modela, njihovog treniranja i testiranja. Evaluirajte vaše modele pomoću konfuzijske matrice (tačnost, specifičnost, osjetljivost, kappa statistika, itd) (paket caret, funkcija confusionMatrix()).

** Kad smo očistili naše podatke i napravili (niz koraka objašnjenih u zadatku 2) pogodan set za kreiranje modela, sad ćemo i krenuti sa kreiranjem istih. Također treba napomenuti da train_im sadrži naš testni set iz zadatka 2, dok train_im sadrži train set.

Koristiti ćemo 3 podjele podataka kao što je traženo u postavci. Za sve 3 naredne metode ćemo definisati train (nad njim raditi cross validaciju i bootstrapping) i test set:

```
smp_size <- floor(0.70 * nrow(train_im)) # getting the 70% of the dataset
train_ind <- sample(seq_len(nrow(train_im)), size = smp_size)

train <- train_im[train_ind, ] # train contains 70%
test <- train_im[-train_ind, ] # our test set contains 30%
```

1. Holdout

   1.1 Tree

```
model_dt <-
C5.0.formula(Attrition~NumCompaniesWorked+DistanceFromHome+Department+BusinessTravel+Wo
rkLifeBalance+TrainingTimesLastYear+TotalWorkingYears+JobSatisfaction+MaritalStatus+Job
Level+HourlyRate+JobInvolvement+EnvironmentSatisfaction+EducationField+OverTime+Percent
SalaryHike+StockOptionLevel+YearsWithCurrManager+YearsInCurrentRole+YearsAtCompany,
data=train)

pred.train.dt <- predict(model_dt,test,type = "class")
confusionMatrix(table(test$Attrition,pred.train.dt))
```

Na osnovu prethodnog koda dobiti ćemo sljedeće rezultate:

```
Confusion Matrix and Statistics

     pred.train.dt
       No Yes
  No  277  22
  Yes  33  21

               Accuracy : 0.8442
                 95% CI : (0.8021, 0.8804)
    No Information Rate : 0.8782
    P-Value [Acc > NIR] : 0.9760

                  Kappa : 0.344

 Mcnemar's Test P-Value : 0.1775

            Sensitivity : 0.8935
            Specificity : 0.4884
         Pos Pred Value : 0.9264
         Neg Pred Value : 0.3889
             Prevalence : 0.8782
         Detection Rate : 0.7847
   Detection Prevalence : 0.8470
      Balanced Accuracy : 0.6910

       'Positive' Class : No

>
```

1.2 RandomForest

```
model_rf_y
<-randomForest(Attrition~NumCompaniesWorked+DistanceFromHome+Department+BusinessTravel+WorkLifeBalance+TrainingTimesLastYear+TotalWorkingYears+JobSatisfaction+MaritalStatus+JobLevel+HourlyRate+JobInvolvement+EnvironmentSatisfaction+EducationField+OverTime+PercentSalaryHike+StockOptionLevel+YearsWithCurrManager+YearsInCurrentRole+YearsAtCompany,data=train)

pred.train.rf <- predict(model_rf_y,test)

confusionMatrix(table(test$Attrition,pred.train.rf))
```

Na osnovu prethodnog koda dobiti ćemo sljedeće rezultate:

```
> confusionMatrix(table(test$Attrition,pred.train.rf))
Confusion Matrix and Statistics

     pred.train.rf
       No Yes
  No  292   2
  Yes  48  11

               Accuracy : 0.8584
                 95% CI : (0.8176, 0.893)
    No Information Rate : 0.9632
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2609

 Mcnemar's Test P-Value : 1.966e-10

            Sensitivity : 0.8588
            Specificity : 0.8462
         Pos Pred Value : 0.9932
         Neg Pred Value : 0.1864
             Prevalence : 0.9632
         Detection Rate : 0.8272
   Detection Prevalence : 0.8329
      Balanced Accuracy : 0.8525

       'Positive' Class : No
```

1.3 Gbm

```
model_dt_a <-
train(Attrition~NumCompaniesWorked+DistanceFromHome+Department+BusinessTravel+WorkLifeB
alance+TrainingTimesLastYear+TotalWorkingYears+JobSatisfaction+MaritalStatus+JobLevel+H
ourlyRate+JobInvolvement+EnvironmentSatisfaction+EducationField+OverTime+PercentSalaryH
ike+StockOptionLevel+YearsWithCurrManager+YearsInCurrentRole+YearsAtCompany,
data=train, method = "gbm", distribution="bernoulli")

pred.train.dt <- predict(model_dt_a,test,type = "raw")

confusionMatrix(table(test$Attrition,pred.train.dt))
```

Na osnovu prethodnog koda dobiti ćemo sljedeće rezultate:

```
> confusionMatrix(table(test$Attrition,pred.train.dt))
Confusion Matrix and Statistics

     pred.train.dt
       No Yes
  No  292   7
  Yes  38  16

               Accuracy : 0.8725
                 95% CI : (0.8332, 0.9055)
    No Information Rate : 0.9348
    P-Value [Acc > NIR] : 1

                  Kappa : 0.3568

 Mcnemar's Test P-Value : 7.744e-06

            Sensitivity : 0.8848
            Specificity : 0.6957
         Pos Pred Value : 0.9766
         Neg Pred Value : 0.2963
             Prevalence : 0.9348
         Detection Rate : 0.8272
   Detection Prevalence : 0.8470
      Balanced Accuracy : 0.7903

       'Positive' Class : No
```

2. Cross Validation

2.1 C5.0

```
train_control <- trainControl(method="cv", number=10)
model_dt_k<-train(Attrition~NumCompaniesWorked+DistanceFromHome+Department+BusinessTrav
el+WorkLifeBalance+TrainingTimesLastYear+TotalWorkingYears+JobSatisfaction+MaritalStatu
s+JobLevel+HourlyRate+JobInvolvement+EnvironmentSatisfaction+EducationField+OverTime+Pe
rcentSalaryHike+StockOptionLevel+YearsWithCurrManager+YearsInCurrentRole+YearsAtCompany
, data=train, method = "C5.0", trControl=train_control)

pred.train.dt <- predict(model_dt_k,test,type = "raw")

confusionMatrix(table(test$Attrition,pred.train.dt))
```

Na osnovu prethodnog koda dobiti ćemo sljedeće rezultate:

```
Confusion Matrix and Statistics

     pred.train.dt
       No Yes
  No  289   5
  Yes  48  11

               Accuracy : 0.8499
                 95% CI : (0.8083, 0.8855)
    No Information Rate : 0.9547
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2391

 Mcnemar's Test P-Value : 7.968e-09

            Sensitivity : 0.8576
            Specificity : 0.6875
         Pos Pred Value : 0.9830
         Neg Pred Value : 0.1864
             Prevalence : 0.9547
         Detection Rate : 0.8187
   Detection Prevalence : 0.8329
      Balanced Accuracy : 0.7725

       'Positive' Class : No
```

2.2 RandomForest

```
train_control <- trainControl(method="cv", number=10)
# fix the parameters of the algorithm
grid <- expand.grid(.fL=c(0), .usekernel=c(FALSE))
# train the model
model_rf_k<-randomForest(Attrition~NumCompaniesWorked+DistanceFromHome+Department+BusinessTravel+WorkLifeBalance+TrainingTimesLastYear+TotalWorkingYears+JobSatisfaction+MaritalStatus+JobLevel+HourlyRate+JobInvolvement+EnvironmentSatisfaction+EducationField+OverTime+PercentSalaryHike+StockOptionLevel+YearsWithCurrManager+YearsInCurrentRole+YearsAtCompany,data=train,trControl=train_control,tuneGrid=grid)

pred.train.dt <- predict(model_rf_k,test,type = "class")

confusionMatrix(table(test$Attrition,pred.train.dt))
```

Na osnovu prethodnog koda dobiti ćemo sljedeće rezultate:

```
> confusionMatrix(table(test$Attrition,pred.train.dt))
Confusion Matrix and Statistics

     pred.train.dt
       No Yes
  No  292   2
  Yes  48  11

               Accuracy : 0.8584
                 95% CI : (0.8176, 0.893)
    No Information Rate : 0.9632
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2609

 Mcnemar's Test P-Value : 1.966e-10

            Sensitivity : 0.8588
            Specificity : 0.8462
         Pos Pred Value : 0.9932
         Neg Pred Value : 0.1864
             Prevalence : 0.9632
         Detection Rate : 0.8272
   Detection Prevalence : 0.8329
      Balanced Accuracy : 0.8525

       'Positive' Class : No
```

2.3 Gbm

```
train_control <- trainControl(method="cv", number=10)
model_dt_b<-train(Attrition~NumCompaniesWorked+DistanceFromHome+Department+BusinessTrav
el+WorkLifeBalance+TrainingTimesLastYear+TotalWorkingYears+JobSatisfaction+MaritalStatu
s+JobLevel+HourlyRate+JobInvolvement+EnvironmentSatisfaction+EducationField+OverTime+Pe
rcentSalaryHike+StockOptionLevel+YearsWithCurrManager+YearsInCurrentRole+YearsAtCompany
, data=train, method = "gbm", distribution="bernoulli", trControl=train_control)

pred.train.dt <- predict(model_dt_b,test,type = "raw")

confusionMatrix(table(test$Attrition,pred.train.dt))
```

Na osnovu prethodnog koda dobiti ćemo sljedeće rezultate:

```
> confusionMatrix(table(test$Attrition,pred.train.dt))
Confusion Matrix and Statistics

     pred.train.dt
       No Yes
  No  293   6
  Yes  40  14

               Accuracy : 0.8697
                 95% CI : (0.83, 0.903)
    No Information Rate : 0.9433
    P-Value [Acc > NIR] : 1

                  Kappa : 0.3223

 Mcnemar's Test P-Value : 1.141e-06

            Sensitivity : 0.8799
            Specificity : 0.7000
         Pos Pred Value : 0.9799
         Neg Pred Value : 0.2593
             Prevalence : 0.9433
         Detection Rate : 0.8300
   Detection Prevalence : 0.8470
      Balanced Accuracy : 0.7899

       'Positive' Class : No
```

3. Bootstrapping

3.1 C5.0

```
train_control <- trainControl(method="boot", number=10)
model_dt_e <-
train(Attrition~NumCompaniesWorked+DistanceFromHome+Department+BusinessTravel+WorkLifeB
alance+TrainingTimesLastYear+TotalWorkingYears+JobSatisfaction+MaritalStatus+JobLevel+H
ourlyRate+JobInvolvement+EnvironmentSatisfaction+EducationField+OverTime+PercentSalaryH
ike+StockOptionLevel+YearsWithCurrManager+YearsInCurrentRole+YearsAtCompany,
data=train, method = "C5.0", trControl=train_control)

pred.train.dt <- predict(model_dt_e,test,type = "raw")

confusionMatrix(table(test$Attrition,pred.train.dt))
```

Na osnovu prethodnog koda dobiti ćemo sljedeće rezultate:

```
> confusionMatrix(table(test$Attrition,pred.train.dt))
Confusion Matrix and Statistics

     pred.train.dt
       No Yes
  No  285   9
  Yes  45  14

               Accuracy : 0.847
                 95% CI : (0.8052, 0.8829)
    No Information Rate : 0.9348
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2733

 Mcnemar's Test P-Value : 1.908e-06

            Sensitivity : 0.8636
            Specificity : 0.6087
         Pos Pred Value : 0.9694
         Neg Pred Value : 0.2373
             Prevalence : 0.9348
         Detection Rate : 0.8074
   Detection Prevalence : 0.8329
      Balanced Accuracy : 0.7362

       'Positive' Class : No
```

3.2 RandomForest

```
train_control <- trainControl(method="boot", number=10)
# train the model
model_rf_b<-randomForest(Attrition~NumCompaniesWorked+DistanceFromHome+Department+Busin
essTravel+WorkLifeBalance+TrainingTimesLastYear+TotalWorkingYears+JobSatisfaction+Marit
alStatus+JobLevel+HourlyRate+JobInvolvement+EnvironmentSatisfaction+EducationField+Over
Time+PercentSalaryHike+StockOptionLevel+YearsWithCurrManager+YearsInCurrentRole+YearsAt
Company,data=train,trControl=train_control)

pred.train.dt <- predict(model_rf_b,test,type = "class")

confusionMatrix(table(test$Attrition,pred.train.dt))
```

Na osnovu prethodnog koda dobiti ćemo sljedeće rezultate:

```
Confusion Matrix and Statistics

     pred.train.dt
       No Yes
  No  293   1
  Yes  48  11

               Accuracy : 0.8612
                 95% CI : (0.8207, 0.8955)
    No Information Rate : 0.966
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2685

 Mcnemar's Test P-Value : 4.983e-11

            Sensitivity : 0.8592
            Specificity : 0.9167
         Pos Pred Value : 0.9966
         Neg Pred Value : 0.1864
             Prevalence : 0.9660
         Detection Rate : 0.8300
   Detection Prevalence : 0.8329
      Balanced Accuracy : 0.8880

       'Positive' Class : No
```

3.3 Gbm

```
train_control <- trainControl(method="boot", number=10)

model_dt_c<-train(Attrition~NumCompaniesWorked+DistanceFromHome+Department+BusinessTrav
el+WorkLifeBalance+TrainingTimesLastYear+TotalWorkingYears+JobSatisfaction+MaritalStatu
s+JobLevel+HourlyRate+JobInvolvement+EnvironmentSatisfaction+EducationField+OverTime+Pe
rcentSalaryHike+StockOptionLevel+YearsWithCurrManager+YearsInCurrentRole+YearsAtCompany
, data=train, method = "gbm", distribution="bernoulli", trControl=train_control)

pred.train.dt <- predict(model_dt_c,test,type = "raw")

confusionMatrix(table(test$Attrition,pred.train.dt))
```

Na osnovu prethodnog koda dobiti ćemo sljedeće rezultate:

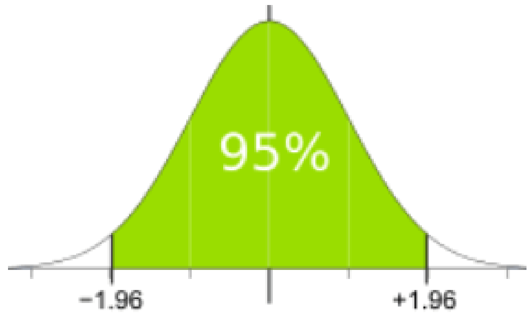
```
> confusionMatrix(table(test$Attrition,pred.train.dt))
Confusion Matrix and Statistics

     pred.train.dt
       No Yes
  No  294   5
  Yes  35  19

               Accuracy : 0.8867
                 95% CI : (0.8489, 0.9178)
    No Information Rate : 0.932
    P-Value [Acc > NIR] : 0.9994

                  Kappa : 0.4339

 Mcnemar's Test P-Value : 4.533e-06

            Sensitivity : 0.8936
            Specificity : 0.7917
         Pos Pred Value : 0.9833
         Neg Pred Value : 0.3519
             Prevalence : 0.9320
         Detection Rate : 0.8329
   Detection Prevalence : 0.8470
      Balanced Accuracy : 0.8426

       'Positive' Class : No
```

Za ostvarene rezultate predikcije, objasnite značenje polja 95% CI, P-Value [Acc > NIR], Pos Pred Value, Prevalence, Detection Rate, Detection Prevalence i Balanced Accuracy.

Da ne bismo objašnjavali svaki paremātar svakog modela zasebno, objasniti ćemo ovdje generalno značenje svakog od parametara u modelu:

95% CI

- I (intervali pouzdanosti) služe kao indikatori koliko za preciznost predviđene vrijednosti. Oni Procjena intervala je zasnovana na statističkim osobinama podataka (mean, mode, median, lin. regresija). Svaki interval ima svojstvo pokrivenosti tj. pouzdanosti. 95% CI znači da će 95% intervala sadržavati prosječnu vrijednost populacije.



Primjer 95% CI

P-Value [Acc > NIR]

- P-Value je vrijednost koja je korisna prilikom određivanja značaja rezultata dobivenom testom hipoteze u statistici. U test se uključuju početna hipoteza i njena alternativa. P vrijednost može imati vrijednost u rasponu od 0 do 1. U slučaju da je p vrijednost manja od 0.05, to indicira da treba odustati od početne hipoteze. Vrijednost veća od 0.05 indicira da treba očuvati početnu pretpostavku i odbaciti alternativeu. Vrijednosti blizu 0.05 se smatraju marginalnim te na osnovu njih se ništa ne može zaključiti.

Pos Pred Value

- Predstavlja omjer svih pozivitnih rezultata i stvarno pozitivnih rezultata. PPV prikazuje tačnost korištene metode. Vrijednost PPV-a se može pronaći koristeći Bayesovu teoremu. PPV se može izračunati koristeći sljedeću formulu:

$$PPV = \frac{broj\_true\_positive}{broj\_true\_positive + broj\_false\_positive}$$

Prevalence

- Broj koji označava koliki je procenat populacije pozivivan. Za dobijanje ove vrijednosti, najčešće se koriste metrike tačnosti, preciznosti, opoziva i osjetljivosti.

Detection Rate

- Proporcija testnih elemenata sa određenom osobinom i elemenata koji su pozitivno klasificirane sa tom osobinom.

Detection Prevalence

- Detection prevalence je broj predviđenih pozitivnih događaja, uključujući lažne pozitivne i stvarne pozitivne, podijeljen sa ukupnim brojem predviđanja.

Balanced Accuracy

- Balanced Accuracy definišemo kao odnos prosječnog broja opozvanih vrijednosti iz jedne klase. Koristi za rad sa nebalansiranom skupovima podataka u problemima binarne i višeklasne klasifikacije.

Na osnovu ostvarenih rezultata odaberite najbolji predikcijski model i obrazložite vas odabir.

Dijeljenje podataka	Način treniranja modela	Pos pred value (>= 0.65)	Balanced Accuracy (>= 0.6)	Kappa statistics (>= 0.25)	Lower Bound CI (>= 0.8)
Holdout	C5.0	0.9264	0.6910	0.3440	0.8021
Holdout	RandomForest	0.9932	0.8525	0.2609	0.8176
Holdout	Gbm	0.9766	0.7903	0.3568	0.8332
Cross Validation	C5.0	0.9830	0.7725	0.2391	0.8083
Cross Validation	RandomForest	0.9932	0.8525	0.2609	0.8176
Cross Validation	Gbm	0.9799	0.7899	0.3223	0.8300
Bootstrapping	C5.0	0.9694	0.7362	0.2733	0.8052
Bootstrapping	RandomForest	0.9966	0.8880	0.2685	0.8207
Bootstrapping	Gbm	0.9833	0.8426	0.4339	0.8489

U prethodnoj tabeli su prikazanje najveće vrijednosti parametara dobijene za svaki model. Prvi inicijalni izbor bi bio model Bootstrapping - RandomForest jer daje najbolje rezultate sa prva dva parametra nad našim testnim setom train_im podataka (test), ali veoma blizu po vrijednostima je model Boostrapping - Gbm, tj. Pos Pred Value i Balanced Accuracy su veoma blize, dok raziku u Kappa statistics nismo mogli ne uvidjeti, a ujedno ovaj model daje najveću vrijednost za Lower Bound CI.

Žrtvujući malo Pos pred value i Balanced Accuracy, izabrali smo model Boostrapping - Gbm kao model koji ima najbolje overall performanse.