DSC520-FinalProject

SumbarajuAditya

2/25/2021

# library

library(knitr)  
library(ggplot2)  
library(pander)  
library(dplyr)

##   
## 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(gridExtra)

##   
## Attaching package: 'gridExtra'

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

library(reshape2)   
library(ISLR)   
library(car)

## Loading required package: carData

##   
## Attaching package: 'car'

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

library(leaps)  
library(glmnet) # glmnet()

## Loading required package: Matrix

## Loaded glmnet 4.1

library(bestglm)   
library(pROC)

## Type 'citation("pROC")' for a citation.

##   
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':  
##   
## cov, smooth, var

library(plyr)

## ------------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------------

##   
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

library(broom)  
library(mapproj)

## Loading required package: maps

##   
## Attaching package: 'maps'

## The following object is masked from 'package:plyr':  
##   
## ozone

library(boot)

##   
## Attaching package: 'boot'

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

library(tree)  
library(randomForest)

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##   
## Attaching package: 'randomForest'

## The following object is masked from 'package:gridExtra':  
##   
## combine

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

## The following object is masked from 'package:ggplot2':  
##   
## margin

library(rpart)

Data importing and cleaning steps

filename <- 'C:/BU/DSC520/assignment\_repo/dsc520/data/final\_project/diabetic\_data.csv'  
diabetic\_data <- read.table(filename, sep = ",", header = T, na.strings = "?")  
#head(data)

For getting the readmission prediction we dont need many columns hence we should be selecting only relevant values to make it simpler.Also we need to remove those columns which mainly has NA values.

diabetic\_data$admission\_type\_id <- as.factor(diabetic\_data$admission\_type\_id)  
diabetic\_data$discharge\_disposition\_id <- as.factor(diabetic\_data$discharge\_disposition\_id)  
diabetic\_data$admission\_source\_id <- as.factor(diabetic\_data$admission\_source\_id)

Cleaning Missing Values Below is the logic to replace bad data “?” and “unknown/invalid values” with NA. I am passing respective columns from diabetic\_data\_clean “diabetic\_data” using ncol and trying to identify the bad data by passing the columns in loop and there by replacing the data with NA.

count <- 0  
for(i in 1:ncol(diabetic\_data))  
 {  
 if(is.factor(diabetic\_data[,i]))  
 {  
 for(j in 1:nrow(diabetic\_data))  
 {  
 if(diabetic\_data[j,i]== "?" | diabetic\_data[j,i]== "Unknown/Invalid" )  
 {  
 count <- count + 1  
 diabetic\_data[j,i] <- NA #replace "?" and "Unknown/Invalid" values with NA  
 }  
 }  
 if(count > 0)  
 {  
 print(c(colnames(diabetic\_data)[i],count))  
 }  
 }  
 count <- 0  
 }

As the data is huge - it is leading to performance issue; hence createda new file which represents clean data.

write.csv(diabetic\_data, file = "diabetic\_data\_clean.csv")

Below are the variables removed: Payer code, weight and Medical Specialty are not included since they have a large number of missing values.

Minor variability variables are excluded. That includes following variables acetohexamide, glimepiride.pioglitazone, metformin.rosiglitazone, metformin.pioglitazone, chlorpropamide, acetohexamide, tolbutamide, acarbose, miglitor, troglitazone, tolazamide, examide, citoglipton, glyburide.metformin, glipizide.metformin, and glimepiride.pioglitazone.

diabetic\_data\_clean <- read.csv("./diabetic\_data\_clean.csv")  
diabetic\_data\_clean$X <- NULL  
diabetic\_data\_clean$weight <- NULL  
diabetic\_data\_clean$payer\_code <- NULL  
#diabetic\_data\_clean$age <- NULL  
#diabetic\_data\_clean$admission\_type\_id <- NULL  
#diabetic\_data\_clean$discharge\_disposition\_id <- NULL  
#diabetic\_data\_clean$admission\_source\_id <- NULL   
diabetic\_data\_clean$medical\_specialty <- NULL  
diabetic\_data\_clean$encounter\_id <- NULL  
#diabetic\_data\_clean$diag\_1 <- NULL  
#diabetic\_data\_clean$diag\_2 <- NULL  
#diabetic\_data\_clean$diag\_3 <- NULL  
diabetic\_data\_clean$examide <- NULL  
diabetic\_data\_clean$citoglipton <- NULL  
diabetic\_data\_clean$acetohexamide <- NULL  
diabetic\_data\_clean$repaglinide <- NULL  
diabetic\_data\_clean$nateglinide <- NULL  
diabetic\_data\_clean$metformin.pioglitazone <- NULL   
diabetic\_data\_clean$metformin.rosiglitazone <- NULL  
diabetic\_data\_clean$chlorpropamide <- NULL  
diabetic\_data\_clean$acetohexamide <- NULL  
diabetic\_data\_clean$miglitol <- NULL   
diabetic\_data\_clean$tolbutamide <- NULL  
diabetic\_data\_clean$acarbose <- NULL  
diabetic\_data\_clean$miglitor <- NULL  
diabetic\_data\_clean$troglitazone <- NULL  
diabetic\_data\_clean$tolazamide <- NULL  
diabetic\_data\_clean$examide <- NULL  
diabetic\_data\_clean$citoglipton <- NULL  
diabetic\_data\_clean$glyburide.metformin <- NULL  
diabetic\_data\_clean$glipizide.metformin <- NULL  
diabetic\_data\_clean$glimepiride.pioglitazone <- NULL  
diabetic\_data\_clean <- na.omit(diabetic\_data\_clean)  
write.csv(diabetic\_data, file = "diabetic\_data\_clean1.csv")  
dim(diabetic\_data\_clean)

## [1] 98053 30

Below is the Summary & Structure of the data;

str(diabetic\_data\_clean)

## 'data.frame': 98053 obs. of 30 variables:  
## $ patient\_nbr : int 55629189 86047875 82442376 42519267 82637451 84259809 114882984 48330783 63555939 89869032 ...  
## $ race : chr "Caucasian" "AfricanAmerican" "Caucasian" "Caucasian" ...  
## $ gender : chr "Female" "Female" "Male" "Male" ...  
## $ age : chr "[10-20)" "[20-30)" "[30-40)" "[40-50)" ...  
## $ admission\_type\_id : int 1 1 1 1 2 3 1 2 3 1 ...  
## $ discharge\_disposition\_id: int 1 1 1 1 1 1 1 1 3 1 ...  
## $ admission\_source\_id : int 7 7 7 7 2 2 7 4 4 7 ...  
## $ time\_in\_hospital : int 3 2 2 1 3 4 5 13 12 9 ...  
## $ num\_lab\_procedures : int 59 11 44 51 31 70 73 68 33 47 ...  
## $ num\_procedures : int 0 5 1 0 6 1 0 2 3 2 ...  
## $ num\_medications : int 18 13 16 8 16 21 12 28 18 17 ...  
## $ number\_outpatient : int 0 2 0 0 0 0 0 0 0 0 ...  
## $ number\_emergency : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ number\_inpatient : int 0 1 0 0 0 0 0 0 0 0 ...  
## $ diag\_1 : chr "276" "648" "8" "197" ...  
## $ diag\_2 : chr "250.01" "250" "250.43" "157" ...  
## $ diag\_3 : chr "255" "V27" "403" "250" ...  
## $ number\_diagnoses : int 9 6 7 5 9 7 8 8 8 9 ...  
## $ max\_glu\_serum : chr "None" "None" "None" "None" ...  
## $ A1Cresult : chr "None" "None" "None" "None" ...  
## $ metformin : chr "No" "No" "No" "No" ...  
## $ glimepiride : chr "No" "No" "No" "No" ...  
## $ glipizide : chr "No" "Steady" "No" "Steady" ...  
## $ glyburide : chr "No" "No" "No" "No" ...  
## $ pioglitazone : chr "No" "No" "No" "No" ...  
## $ rosiglitazone : chr "No" "No" "No" "No" ...  
## $ insulin : chr "Up" "No" "Up" "Steady" ...  
## $ change : chr "Ch" "No" "Ch" "Ch" ...  
## $ diabetesMed : chr "Yes" "Yes" "Yes" "Yes" ...  
## $ readmitted : chr ">30" "NO" "NO" "NO" ...  
## - attr(\*, "na.action")= 'omit' Named int [1:3713] 1 20 21 22 55 66 67 88 100 112 ...  
## ..- attr(\*, "names")= chr [1:3713] "1" "20" "21" "22" ...

#head (diabetic\_data\_clean)  
# Raw diabetic\_data\_clean:  
dim(diabetic\_data)

## [1] 101766 50

# Clean diabetic\_data\_clean  
dim(diabetic\_data\_clean)

## [1] 98053 30

write.csv(diabetic\_data\_clean, file = "diabetic\_data\_clean1.csv")

#diabetic\_data\_clean <- subset(diabetic\_data\_clean, select = -c(encounter\_id)) # delete ID variables  
names(diabetic\_data\_clean)[length(diabetic\_data\_clean)] <- "y"   
diabetic\_data\_clean$y <- as.character(diabetic\_data\_clean$y)  
diabetic\_data\_clean$y[which(diabetic\_data\_clean$y == ">30" | diabetic\_data\_clean$y == "NO")] <- "0"  
diabetic\_data\_clean$y[which(diabetic\_data\_clean$y == "<30")] <- "1"  
diabetic\_data\_clean$y <- as.factor(diabetic\_data\_clean$y)  
  
#rename variables  
names(diabetic\_data\_clean) <- c("pers", "race", "gend", "agem","admt", "disc", "adms","hosp", "labp", "proc", "nmed", "nout", "emer", "ninp", "mod1", "mod2", "mod3", "diag", "mglu", "A1Cr", "metf", "glim", "glip", "glyb", "piog", "rosi", "insu", "chge", "diab", "y")  
  
# Data partition  
set.seed(43)  
pos <- sample(1:nrow(diabetic\_data\_clean), round(nrow(diabetic\_data\_clean)\*0.3))  
testing <- diabetic\_data\_clean[pos, -1]   
training <- diabetic\_data\_clean[-pos, -1]   
  
# XX and y   
data <- training  
y <- data$y  
response <- "readmitted"   
XX <- model.matrix(y ~., data)[, -1]  
Y <- y

yprop <- prop.table(table(data$y))

# subset data which excludes duplicate data points  
# data2 <- data[!duplicated(data$pers), ]   
# data3 <- data[!duplicated(c(paste0(data$pers, data$agem))), ]   
# demographics  
ytab1 <- prop.table(table(data$race))   
crosstab <- with(data, table(race, y))  
ytab2 <- prop.table(crosstab, margin = 2)   
df.race <- data.frame(round(rbind(ytab1, t(ytab2)), 3))  
names(df.race)[1] <- "Missing"  
rownames(df.race) <- c("Total", "y = 0", "y = 1")  
df.race

## Missing Asian Caucasian Hispanic Other  
## Total 0.193 0.006 0.765 0.020 0.015  
## y = 0 0.193 0.006 0.765 0.020 0.016  
## y = 1 0.196 0.006 0.769 0.017 0.013

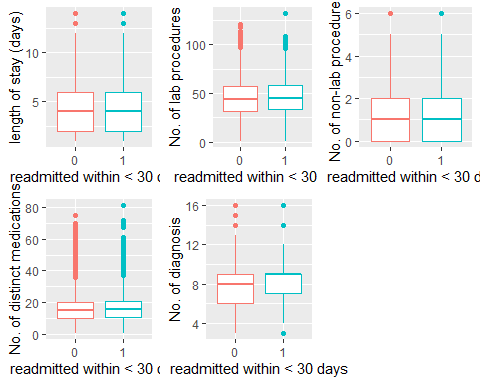
ytab1 <- prop.table(table(data$gend))   
crosstab <- with(data, table(gend, y))  
ytab2 <- prop.table(crosstab, margin = 2)   
df.gender <- data.frame(round(rbind(ytab1, t(ytab2)), 3))  
rownames(df.gender) <- c("Total","y = 0", "y = 1")  
df.gender

## Female Male Unknown.Invalid  
## Total 0.540 0.460 0  
## y = 0 0.540 0.460 0  
## y = 1 0.542 0.458 0

ytab1 <- prop.table(table(data$agem))   
crosstab <- with(data, table(agem, y))  
ytab2 <- prop.table(crosstab, margin = 2)   
df.age <- data.frame(round(rbind(ytab1, t(ytab2)), 3))  
rownames(df.age) <- c("Total", "y = 0", "y = 1")  
colnames(df.age) <- c("0~19", "20~59", "60~79", "80+")  
df.age

## 0~19 20~59 60~79 80+ NA NA NA NA NA NA  
## Total 0.001 0.005 0.015 0.036 0.094 0.170 0.224 0.258 0.169 0.028  
## y = 0 0.001 0.005 0.015 0.036 0.095 0.173 0.224 0.256 0.167 0.028  
## y = 1 0.000 0.003 0.019 0.036 0.089 0.148 0.225 0.271 0.184 0.027

# names(data)  
# df <- subset(data, select = c(race, gender, y))  
# knitr::kable(df)  
# continuous data  
plot1 <- ggplot(data, aes(x = y, y = hosp, color = y)) + geom\_boxplot() + theme(legend.position = "none") +  
 labs(y = "length of stay (days)", x = "readmitted within < 30 days")   
plot2 <- ggplot(data, aes(x = y, y = labp, color = y)) + geom\_boxplot() + theme(legend.position = "none") +  
 labs(y = "No. of lab procedures", x = "readmitted within < 30 days")   
plot3 <- ggplot(data, aes(x = y, y = proc, color = y)) + geom\_boxplot() + theme(legend.position = "none") +  
 labs(y = "No. of non-lab procedures", x = "readmitted within < 30 days")   
plot4 <- ggplot(data, aes(x = y, y = nmed, color = y)) + geom\_boxplot() + theme(legend.position = "none") +  
 labs(y = "No. of distinct medications", x = "readmitted within < 30 days")   
plot8 <- ggplot(data, aes(x = y, y = diag, color = y)) + geom\_boxplot() + theme(legend.position = "none") +  
 labs(y = "No. of diagnosis", x = "readmitted within < 30 days")   
grid.arrange(plot1, plot2, plot3, plot4, plot8, nrow = 2)



plot5 <- ggplot(data, aes(x = y, y = nout, color = y)) + geom\_boxplot() + theme(legend.position = "none") +  
 labs(y = "No. of outpatient", x = "readmitted within < 30 days")   
plot6 <- ggplot(data, aes(x = y, y = emer, color = y)) + geom\_boxplot() + theme(legend.position = "none") +  
 labs(y = "No. of emergency", x = "readmitted within < 30 days")   
plot7 <- ggplot(data, aes(x = y, y = ninp, color = y)) + geom\_boxplot() + theme(legend.position = "none") +  
 labs(y = "No. of inpatient", x = "readmitted within < 30 days")   
grid.arrange(plot5, plot6, plot7, nrow = 1)

