PimaIndianDiabetes.R

user

2021-09-26

#Pima Indians Diabetes   
  
# Who is Pima Indians ?  
#"The Pima Indians ( "River People")are a group of Native Americans living in an area   
#consisting of what is now central and southern Arizona.  
  
# Understanding the data  
#The datasets consist of several medical predictor (independent) variables  
#and one target (dependent) variable, Outcome. Independent variables include  
#the number of pregnancies the patient has had, their BMI, insulin level,   
#age, and so on.  
  
  
# Load required libraries  
library(ggplot2)  
library(ggthemes)

## Warning: package 'ggthemes' was built under R version 4.1.1

library(psych)

## Warning: package 'psych' was built under R version 4.1.1

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

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(caret)  
library(reshape2)  
library(corrplot)

## Warning: package 'corrplot' was built under R version 4.1.1

## corrplot 0.90 loaded

# Load the dataset  
dataset=read.csv("C:/Users/user/Downloads/diabetes.csv")  
  
  
# Print first 5 row  
print(head(dataset,5))

## Pregnancies Glucose BloodPressure SkinThickness Insulin BMI  
## 1 6 148 72 35 0 33.6  
## 2 1 85 66 29 0 26.6  
## 3 8 183 64 0 0 23.3  
## 4 1 89 66 23 94 28.1  
## 5 0 137 40 35 168 43.1  
## DiabetesPedigreeFunction Age Outcome  
## 1 0.627 50 1  
## 2 0.351 31 0  
## 3 0.672 32 1  
## 4 0.167 21 0  
## 5 2.288 33 1

# Print last 5 row  
print(tail(dataset,5))

## Pregnancies Glucose BloodPressure SkinThickness Insulin BMI  
## 764 10 101 76 48 180 32.9  
## 765 2 122 70 27 0 36.8  
## 766 5 121 72 23 112 26.2  
## 767 1 126 60 0 0 30.1  
## 768 1 93 70 31 0 30.4  
## DiabetesPedigreeFunction Age Outcome  
## 764 0.171 63 0  
## 765 0.340 27 0  
## 766 0.245 30 0  
## 767 0.349 47 1  
## 768 0.315 23 0

# To View the contents in the dataet  
View(dataset)  
  
  
# Print column names  
print(names(dataset))

## [1] "Pregnancies" "Glucose"   
## [3] "BloodPressure" "SkinThickness"   
## [5] "Insulin" "BMI"   
## [7] "DiabetesPedigreeFunction" "Age"   
## [9] "Outcome"

# Dimention of data  
print(dim(dataset))

## [1] 768 9

# Print Statistical summary  
describe(dataset)

## vars n mean sd median trimmed mad min  
## Pregnancies 1 768 3.85 3.37 3.00 3.46 2.97 0.00  
## Glucose 2 768 120.89 31.97 117.00 119.38 29.65 0.00  
## BloodPressure 3 768 69.11 19.36 72.00 71.36 11.86 0.00  
## SkinThickness 4 768 20.54 15.95 23.00 19.94 17.79 0.00  
## Insulin 5 768 79.80 115.24 30.50 56.75 45.22 0.00  
## BMI 6 768 31.99 7.88 32.00 31.96 6.82 0.00  
## DiabetesPedigreeFunction 7 768 0.47 0.33 0.37 0.42 0.25 0.08  
## Age 8 768 33.24 11.76 29.00 31.54 10.38 21.00  
## Outcome 9 768 0.35 0.48 0.00 0.31 0.00 0.00  
## max range skew kurtosis se  
## Pregnancies 17.00 17.00 0.90 0.14 0.12  
## Glucose 199.00 199.00 0.17 0.62 1.15  
## BloodPressure 122.00 122.00 -1.84 5.12 0.70  
## SkinThickness 99.00 99.00 0.11 -0.53 0.58  
## Insulin 846.00 846.00 2.26 7.13 4.16  
## BMI 67.10 67.10 -0.43 3.24 0.28  
## DiabetesPedigreeFunction 2.42 2.34 1.91 5.53 0.01  
## Age 81.00 60.00 1.13 0.62 0.42  
## Outcome 1.00 1.00 0.63 -1.60 0.02

# Internal structure of data  
print(str(dataset))

## 'data.frame': 768 obs. of 9 variables:  
## $ Pregnancies : int 6 1 8 1 0 5 3 10 2 8 ...  
## $ Glucose : int 148 85 183 89 137 116 78 115 197 125 ...  
## $ BloodPressure : int 72 66 64 66 40 74 50 0 70 96 ...  
## $ SkinThickness : int 35 29 0 23 35 0 32 0 45 0 ...  
## $ Insulin : int 0 0 0 94 168 0 88 0 543 0 ...  
## $ BMI : num 33.6 26.6 23.3 28.1 43.1 25.6 31 35.3 30.5 0 ...  
## $ DiabetesPedigreeFunction: num 0.627 0.351 0.672 0.167 2.288 ...  
## $ Age : int 50 31 32 21 33 30 26 29 53 54 ...  
## $ Outcome : int 1 0 1 0 1 0 1 0 1 1 ...  
## NULL

# Display columns and display some portions of the data  
#print(glimpse(dataset))  
  
  
# Statistical values  
#print(is.na(dataset))  
print(ncol(dataset))

## [1] 9

print(nrow(dataset))

## [1] 768

print(max(dataset$Outcome))

## [1] 1

print(min(dataset$Outcome))

## [1] 0

print(sort(dataset$Outcome))

## [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [38] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [75] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [112] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [149] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [186] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [223] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [260] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [297] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [334] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [371] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [408] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [445] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [482] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [519] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [556] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [593] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [630] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [667] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [704] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [741] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

print(which.max(dataset$Outcome))# Return the index of the first maximum value

## [1] 1

print(which.min(dataset$Outcome))# Return the index of the first minimum value

## [1] 2

print(mean(dataset$Outcome))

## [1] 0.3489583

print(mean(dataset$Outcome,trim=0.10))

## [1] 0.3116883

print(var(dataset$Outcome))

## [1] 0.2274826

print(median(dataset$Outcome))

## [1] 0

print(mad(dataset$Outcome))# mean absolute division

## [1] 0

print(sd(dataset$Outcome))

## [1] 0.4769514

print(range(dataset$Outcome))

## [1] 0 1

print(quantile(dataset$Outcome))

## 0% 25% 50% 75% 100%   
## 0 0 0 1 1

print(IQR(dataset$Outcome))

## [1] 1

print(t.test(dataset$Outcome))

##   
## One Sample t-test  
##   
## data: dataset$Outcome  
## t = 20.276, df = 767, p-value < 2.2e-16  
## alternative hypothesis: true mean is not equal to 0  
## 95 percent confidence interval:  
## 0.3151731 0.3827436  
## sample estimates:  
## mean of x   
## 0.3489583

# Data visualisation  
  
  
# Create a 2 x 2 plotting matrix  
par(mfrow = c(2, 2))  
  
  
# The $ notation can be used to subset the variable you're interested in.  
  
# Histogram of numerical data  
print(hist(dataset$Pregnancies,col="red"))

## $breaks  
## [1] 0 2 4 6 8 10 12 14 16 18  
##   
## $counts  
## [1] 349 143 107 83 52 20 12 1 1  
##   
## $density  
## [1] 0.2272135417 0.0930989583 0.0696614583 0.0540364583 0.0338541667  
## [6] 0.0130208333 0.0078125000 0.0006510417 0.0006510417  
##   
## $mids  
## [1] 1 3 5 7 9 11 13 15 17  
##   
## $xname  
## [1] "dataset$Pregnancies"  
##   
## $equidist  
## [1] TRUE  
##   
## attr(,"class")  
## [1] "histogram"

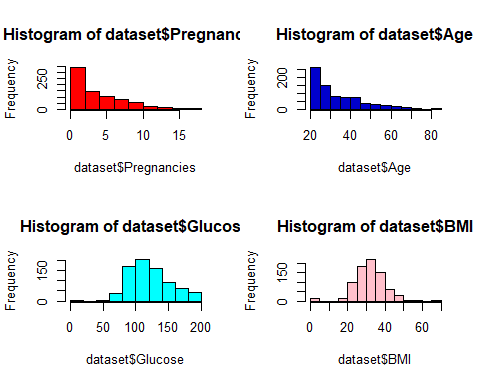
print(hist(dataset$Age,col="blue3"))

## $breaks  
## [1] 20 25 30 35 40 45 50 55 60 65 70 75 80 85  
##   
## $counts  
## [1] 267 150 81 76 76 37 31 23 14 11 1 0 1  
##   
## $density  
## [1] 0.0695312500 0.0390625000 0.0210937500 0.0197916667 0.0197916667  
## [6] 0.0096354167 0.0080729167 0.0059895833 0.0036458333 0.0028645833  
## [11] 0.0002604167 0.0000000000 0.0002604167  
##   
## $mids  
## [1] 22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5 62.5 67.5 72.5 77.5 82.5  
##   
## $xname  
## [1] "dataset$Age"  
##   
## $equidist  
## [1] TRUE  
##   
## attr(,"class")  
## [1] "histogram"

print(hist(dataset$Glucose,col="cyan"))

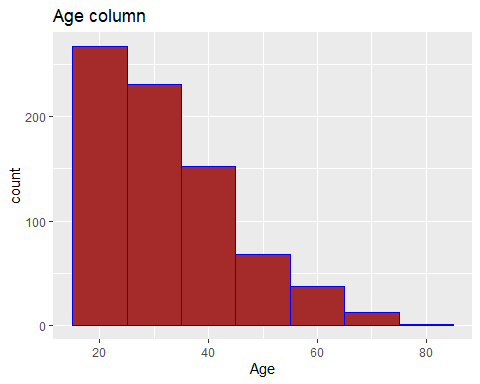
## $breaks  
## [1] 0 20 40 60 80 100 120 140 160 180 200  
##   
## $counts  
## [1] 5 0 4 38 167 205 157 91 60 41  
##   
## $density  
## [1] 0.0003255208 0.0000000000 0.0002604167 0.0024739583 0.0108723958  
## [6] 0.0133463542 0.0102213542 0.0059244792 0.0039062500 0.0026692708  
##   
## $mids  
## [1] 10 30 50 70 90 110 130 150 170 190  
##   
## $xname  
## [1] "dataset$Glucose"  
##   
## $equidist  
## [1] TRUE  
##   
## attr(,"class")  
## [1] "histogram"

print(hist(dataset$BMI,col="pink"))



## $breaks  
## [1] 0 5 10 15 20 25 30 35 40 45 50 55 60 65 70  
##   
## $counts  
## [1] 11 0 0 14 98 180 221 148 61 27 5 2 0 1  
##   
## $density  
## [1] 0.0028645833 0.0000000000 0.0000000000 0.0036458333 0.0255208333  
## [6] 0.0468750000 0.0575520833 0.0385416667 0.0158854167 0.0070312500  
## [11] 0.0013020833 0.0005208333 0.0000000000 0.0002604167  
##   
## $mids  
## [1] 2.5 7.5 12.5 17.5 22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5 62.5 67.5  
##   
## $xname  
## [1] "dataset$BMI"  
##   
## $equidist  
## [1] TRUE  
##   
## attr(,"class")  
## [1] "histogram"

#Age and number of times pregnant are not normal distributions as expected since the underlying population should not be   
#normally distributed either. This 392 observations are just a sample of the original population. On the other hand, the glucose   
#level and BMI seem to follow a normal distribution. When performing any analysis, it is always good to know what is the   
#distribution of the data so all the assumptions for different tests or models can be met."""  
  
  
# Age distribution  
  
age<-ggplot(dataset,aes(x=Age))+geom\_histogram(binwidth=10,col="blue",fill="brown")+  
 labs(title="Age column",x="Age","Count")  
print(age)



# Pregnancy distribution  
print(str(dataset$Pregnancies))

## int [1:768] 6 1 8 1 0 5 3 10 2 8 ...  
## NULL

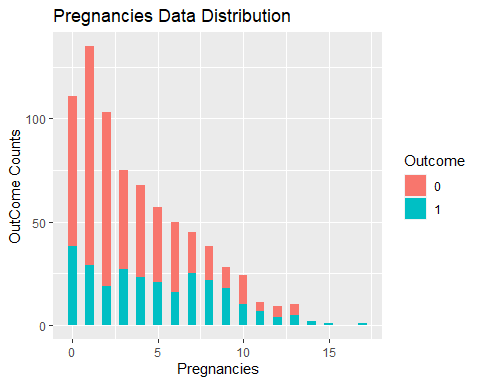
print(table(dataset$Pregnancies))# Create a table for pregnancies

##   
## 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 17   
## 111 135 103 75 68 57 50 45 38 28 24 11 9 10 2 1 1

dataset$Outcome <- as.factor(dataset$Outcome)  
#All 8 independent variables are numeric. There are two outcomes, this data is good for classifciation.   
#Lets change Outcome to categorical Variable  
  
  
# Histogram of Diabetes and non diabetes women  
pd<-ggplot(dataset,aes(x = Pregnancies)) +  
 geom\_histogram(binwidth = 0.5,aes(fill = Outcome,position = "dodge")) +  
 ggtitle("Pregnancies Data Distribution") + ylab("OutCome Counts") +  
 theme\_light() +  
 theme\_update(plot.title = element\_text(hjust = ))

## Warning: Ignoring unknown aesthetics: position

print(pd)



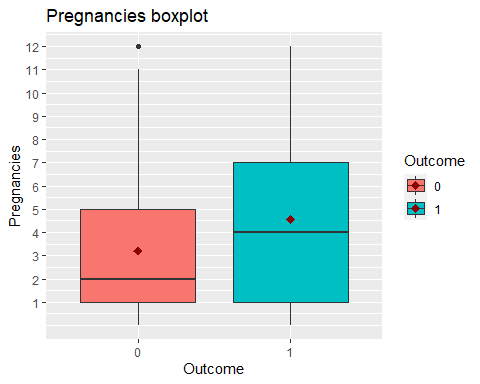
#Pregnancies data is right skewed.  
  
  
# Boxplot of Diabetes and non Diabetes women  
opm<-ggplot(data = dataset,aes(x = Outcome, y = Pregnancies)) +  
 geom\_boxplot( aes(fill= Outcome)) +  
 scale\_y\_continuous(breaks = seq(1,12,1),limits = c(0,12)) +  
 ggtitle("Pregnancies boxplot") +  
 stat\_summary(fun.y=mean, colour="darkred", geom="point",   
 shape=18, size=3,show.legend = TRUE) +  
 theme\_gray() +   
 theme\_update(plot.title = element\_text(hjust = 0.5))

## Warning: `fun.y` is deprecated. Use `fun` instead.

print(opm)

## Warning: Removed 14 rows containing non-finite values (stat\_boxplot).

## Warning: Removed 14 rows containing non-finite values (stat\_summary).



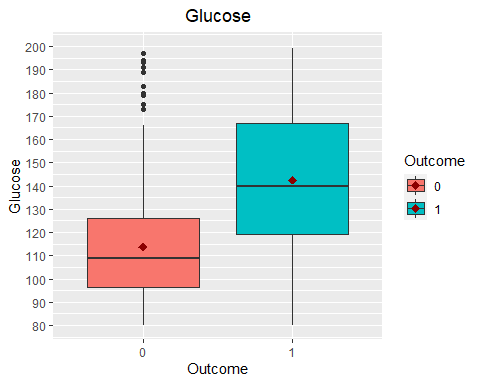
#Box plot shows, woman who had more pregnancies are more prone to diabetes.This may be important variable for model.  
  
  
# Glucose level of Diabetes and non Diabetes women  
ogm<-ggplot(data = dataset,aes(x = Outcome, y = Glucose)) +  
 geom\_boxplot( aes(fill= Outcome)) +  
 scale\_y\_continuous(breaks = seq(80,200,10),limits = c(80,200)) +  
 ggtitle("Glucose") +  
 stat\_summary(fun.y=mean, colour="darkred", geom="point",   
 shape=18, size=3,show.legend = TRUE) +  
 theme\_gray() +   
 theme\_update(plot.title = element\_text(hjust = 0.5))

## Warning: `fun.y` is deprecated. Use `fun` instead.

print(ogm)

## Warning: Removed 41 rows containing non-finite values (stat\_boxplot).

## Warning: Removed 41 rows containing non-finite values (stat\_summary).



#Diabetics woman have high Plasma glucose concentration.   
#On average this value is 140 for diabetics woman while this is quite low for non-diabetics.  
  
#Blood Pressure  
table(dataset$BloodPressure)

##   
## 0 24 30 38 40 44 46 48 50 52 54 55 56 58 60 61 62 64 65 66   
## 35 1 2 1 1 4 2 5 13 11 11 2 12 21 37 1 34 43 7 30   
## 68 70 72 74 75 76 78 80 82 84 85 86 88 90 92 94 95 96 98 100   
## 45 57 44 52 8 39 45 40 30 23 6 21 25 22 8 6 1 4 3 3   
## 102 104 106 108 110 114 122   
## 1 2 3 2 3 1 1

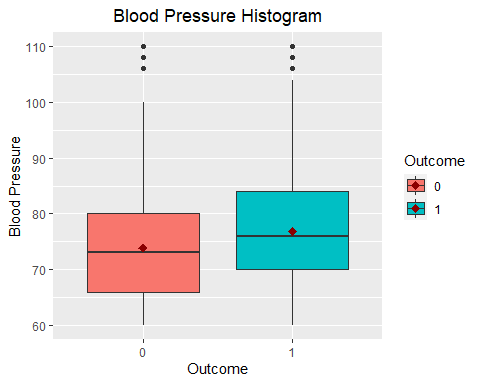
# Blood Pressure level of Diabetes and non Diabetes women  
obm<-ggplot(data = dataset,aes(x = Outcome, y = BloodPressure)) +  
 geom\_boxplot( aes(fill= Outcome)) +  
 scale\_y\_continuous(breaks = seq(60,110,10),limits = c(60,110)) +  
 ylab("Blood Pressure") +  
 ggtitle("Blood Pressure Histogram") +  
 stat\_summary(fun.y=mean, colour="darkred", geom="point",   
 shape=18, size=3,show.legend = TRUE) +  
 theme\_gray() +  
 theme\_update(plot.title = element\_text(hjust = 0.5))

## Warning: `fun.y` is deprecated. Use `fun` instead.

print(obm)

## Warning: Removed 123 rows containing non-finite values (stat\_boxplot).

## Warning: Removed 123 rows containing non-finite values (stat\_summary).



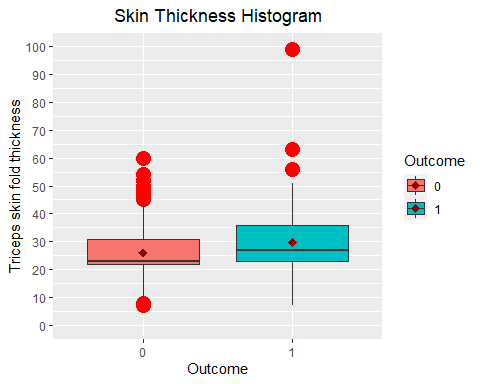
#Diastolic blood pressure for diabetic woman is higher compare to non-diabetics.  
  
  
#Triceps skin fold thickness  
#Triceps skin-fold thickness normal value for female 23  
table(dataset$SkinThickness)

##   
## 0 7 8 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26   
## 227 2 2 5 6 7 11 6 14 6 14 20 18 13 10 16 22 12 16 16   
## 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46   
## 23 20 17 27 19 31 20 8 15 14 16 7 18 16 15 11 6 5 6 8   
## 47 48 49 50 51 52 54 56 60 63 99   
## 4 4 3 3 1 2 2 1 1 1 1

# Let's replace zero with the median value.  
dataset$SkinThickness <- ifelse(  
 dataset$SkinThickness == 0 ,   
 median(dataset$SkinThickness,na.rm = TRUE),  
 dataset$SkinThickness)  
  
  
# Skin thickness of Diabetes and non Diabetes women  
osm<-ggplot(data = dataset,aes(x = Outcome, y = SkinThickness)) +  
 geom\_boxplot( aes(fill= Outcome),outlier.colour = "red", outlier.size = 5) +  
 scale\_y\_continuous(breaks = seq(0,100,10),limits = c(0,100)) +  
 ylab("Triceps skin fold thickness") +  
 ggtitle("Skin Thickness Histogram") +  
 stat\_summary(fun.y=mean, colour="darkred", geom="point",   
 shape=18, size=3,show.legend = TRUE) +  
 theme\_gray() +  
 theme\_update(plot.title = element\_text(hjust = 0.5))

## Warning: `fun.y` is deprecated. Use `fun` instead.

print(osm)



#Boxplot shows that diabetics woman normally has high skin thickness.   
#Red big dots are outlier but ignoring this outlier to consider the extreme case.  
  
  
table(dataset$BMI)

##   
## 0 18.2 18.4 19.1 19.3 19.4 19.5 19.6 19.9 20 20.1 20.4 20.8 21 21.1 21.2   
## 11 3 1 1 1 1 2 3 1 1 1 2 2 2 4 1   
## 21.7 21.8 21.9 22.1 22.2 22.3 22.4 22.5 22.6 22.7 22.9 23 23.1 23.2 23.3 23.4   
## 1 5 3 2 2 1 2 3 2 1 2 2 4 3 2 1   
## 23.5 23.6 23.7 23.8 23.9 24 24.1 24.2 24.3 24.4 24.5 24.6 24.7 24.8 24.9 25   
## 3 3 2 2 2 4 1 6 4 3 1 4 5 3 1 6   
## 25.1 25.2 25.3 25.4 25.5 25.6 25.8 25.9 26 26.1 26.2 26.3 26.4 26.5 26.6 26.7   
## 3 6 2 4 2 6 2 7 4 3 4 1 3 3 4 1   
## 26.8 26.9 27 27.1 27.2 27.3 27.4 27.5 27.6 27.7 27.8 27.9 28 28.1 28.2 28.3   
## 4 1 2 3 2 4 5 5 7 4 7 2 5 1 2 2   
## 28.4 28.5 28.6 28.7 28.8 28.9 29 29.2 29.3 29.5 29.6 29.7 29.8 29.9 30 30.1   
## 6 3 2 7 2 6 5 1 5 5 4 8 3 5 7 9   
## 30.2 30.3 30.4 30.5 30.7 30.8 30.9 31 31.1 31.2 31.3 31.6 31.9 32 32.1 32.2   
## 1 1 7 7 1 9 5 2 1 12 1 12 2 13 1 1   
## 32.3 32.4 32.5 32.6 32.7 32.8 32.9 33.1 33.2 33.3 33.5 33.6 33.7 33.8 33.9 34   
## 3 10 6 1 3 9 9 3 7 10 1 8 5 5 2 6   
## 34.1 34.2 34.3 34.4 34.5 34.6 34.7 34.8 34.9 35 35.1 35.2 35.3 35.4 35.5 35.6   
## 4 8 6 4 5 5 4 2 6 4 3 2 4 4 7 2   
## 35.7 35.8 35.9 36 36.1 36.2 36.3 36.4 36.5 36.6 36.7 36.8 36.9 37 37.1 37.2   
## 4 5 5 2 3 1 3 2 4 5 1 6 3 1 2 4   
## 37.3 37.4 37.5 37.6 37.7 37.8 37.9 38 38.1 38.2 38.3 38.4 38.5 38.6 38.7 38.8   
## 1 3 2 5 5 3 2 2 3 4 1 2 6 1 3 1   
## 38.9 39 39.1 39.2 39.3 39.4 39.5 39.6 39.7 39.8 39.9 40 40.1 40.2 40.5 40.6   
## 1 4 4 2 1 7 3 1 1 2 3 2 1 1 3 4   
## 40.7 40.8 40.9 41 41.2 41.3 41.5 41.8 42 42.1 42.2 42.3 42.4 42.6 42.7 42.8   
## 1 1 2 1 1 3 2 1 1 2 1 3 3 1 2 1   
## 42.9 43.1 43.2 43.3 43.4 43.5 43.6 44 44.1 44.2 44.5 44.6 45 45.2 45.3 45.4   
## 4 1 1 5 2 2 2 2 1 2 2 1 1 1 3 1   
## 45.5 45.6 45.7 45.8 46.1 46.2 46.3 46.5 46.7 46.8 47.9 48.3 48.8 49.3 49.6 49.7   
## 1 2 1 1 2 2 1 1 1 2 2 1 1 1 1 1   
## 50 52.3 52.9 53.2 55 57.3 59.4 67.1   
## 1 2 1 1 1 1 1 1

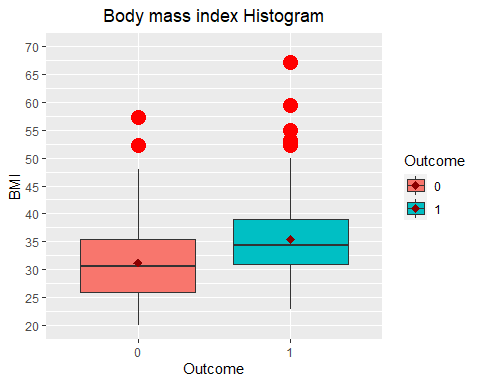
# BMI level of Diabetes and non Diabetes women  
obm<-ggplot(data = dataset,aes(x = Outcome, y = BMI)) +  
 geom\_boxplot( aes(fill= Outcome),outlier.colour = "red", outlier.size = 5) +  
 scale\_y\_continuous(breaks = seq(20,70,5),limits = c(20,70)) +  
 ylab("BMI") +  
 ggtitle("Body mass index Histogram") +  
 stat\_summary(fun.y=mean, colour="darkred", geom="point",   
 shape=18, size=3,show.legend = TRUE) +  
 theme\_gray() +  
 theme\_update(plot.title = element\_text(hjust = 0.5))

## Warning: `fun.y` is deprecated. Use `fun` instead.

print(obm)

## Warning: Removed 24 rows containing non-finite values (stat\_boxplot).

## Warning: Removed 24 rows containing non-finite values (stat\_summary).



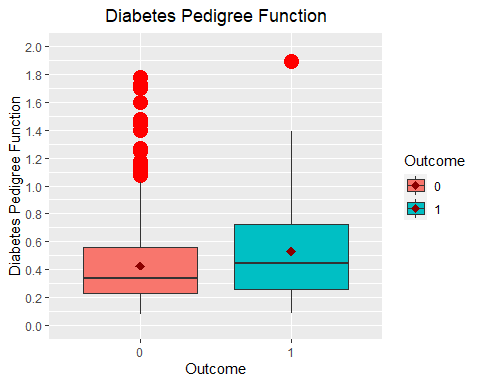
#BMI for diabetics woman is high compare to non-diabetics.  
#There are few outlier, let not treat them to consider the extreme cases of BMI.  
  
  
#Diabetes pedigree function of Diabetes and non Diabetes women  
odpf<-ggplot(data = dataset,aes(x = Outcome, y = DiabetesPedigreeFunction)) +  
 geom\_boxplot( aes(fill= Outcome),outlier.colour = "red", outlier.size = 5) +  
 scale\_y\_continuous(breaks = seq(0,2,0.2),limits = c(0,2)) +  
 ylab("Diabetes Pedigree Function") +  
 ggtitle("Diabetes Pedigree Function") +  
 stat\_summary(fun.y=mean, colour="darkred", geom="point",   
 shape=18, size=3,show.legend = TRUE) +  
 theme\_gray() +  
 theme\_update(plot.title = element\_text(hjust = 0.5))

## Warning: `fun.y` is deprecated. Use `fun` instead.

print(odpf)

## Warning: Removed 4 rows containing non-finite values (stat\_boxplot).

## Warning: Removed 4 rows containing non-finite values (stat\_summary).



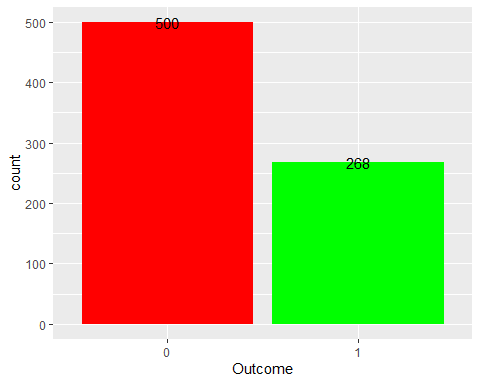
#Check the balancing of data  
table(dataset$Outcome)

##   
## 0 1   
## 500 268

prop.table(table(dataset$Outcome))

##   
## 0 1   
## 0.6510417 0.3489583

ggplot(dataset,aes(Outcome))+  
 geom\_bar(fill=c("red","green"))+  
 geom\_text(stat = "count",aes(label=stat(count),vjust=0.5))



# it seems to be unbalanced  
  
  
# Correlation matrix  
cor\_melt <- melt(cor(dataset[, 1:8]))  
cor\_melt <- cor\_melt[which(cor\_melt$value > 0.5 & cor\_melt$value != 1), ]  
cor\_melt <- cor\_melt[1:3, ]  
print(cor\_melt)

## Var1 Var2 value  
## 8 Age Pregnancies 0.5443412  
## 30 BMI SkinThickness 0.5043073  
## 44 SkinThickness BMI 0.5043073

# correlation values higher than 0.5.  
  
  
#Let's see the correlation between numerical variables. There are variables which are highly correlated.  
#That is the case of Age for example.  
correlat <- cor(dataset[, setdiff(names(dataset), 'Outcome')])  
print(correlat)

## Pregnancies Glucose BloodPressure SkinThickness  
## Pregnancies 1.00000000 0.1294587 0.14128198 0.03256830  
## Glucose 0.12945867 1.0000000 0.15258959 0.15802676  
## BloodPressure 0.14128198 0.1525896 1.00000000 0.16842113  
## SkinThickness 0.03256830 0.1580268 0.16842113 1.00000000  
## Insulin -0.07353461 0.3313571 0.08893338 0.24425015  
## BMI 0.01768309 0.2210711 0.28180529 0.50430728  
## DiabetesPedigreeFunction -0.03352267 0.1373373 0.04126495 0.14297708  
## Age 0.54434123 0.2635143 0.23952795 0.05451384  
## Insulin BMI DiabetesPedigreeFunction  
## Pregnancies -0.07353461 0.01768309 -0.03352267  
## Glucose 0.33135711 0.22107107 0.13733730  
## BloodPressure 0.08893338 0.28180529 0.04126495  
## SkinThickness 0.24425015 0.50430728 0.14297708  
## Insulin 1.00000000 0.19785906 0.18507093  
## BMI 0.19785906 1.00000000 0.14064695  
## DiabetesPedigreeFunction 0.18507093 0.14064695 1.00000000  
## Age -0.04216295 0.03624187 0.03356131  
## Age  
## Pregnancies 0.54434123  
## Glucose 0.26351432  
## BloodPressure 0.23952795  
## SkinThickness 0.05451384  
## Insulin -0.04216295  
## BMI 0.03624187  
## DiabetesPedigreeFunction 0.03356131  
## Age 1.00000000

print(corrplot(correlat,method="ellipse"))

## Warning in corrplot(correlat, method = "ellipse"): Not been able to calculate  
## text margin, please try again with a clean new empty window using {plot.new();  
## dev.off()} or reduce tl.cex

## $corr  
## Pregnancies Glucose BloodPressure SkinThickness  
## Pregnancies 1.00000000 0.1294587 0.14128198 0.03256830  
## Glucose 0.12945867 1.0000000 0.15258959 0.15802676  
## BloodPressure 0.14128198 0.1525896 1.00000000 0.16842113  
## SkinThickness 0.03256830 0.1580268 0.16842113 1.00000000  
## Insulin -0.07353461 0.3313571 0.08893338 0.24425015  
## BMI 0.01768309 0.2210711 0.28180529 0.50430728  
## DiabetesPedigreeFunction -0.03352267 0.1373373 0.04126495 0.14297708  
## Age 0.54434123 0.2635143 0.23952795 0.05451384  
## Insulin BMI DiabetesPedigreeFunction  
## Pregnancies -0.07353461 0.01768309 -0.03352267  
## Glucose 0.33135711 0.22107107 0.13733730  
## BloodPressure 0.08893338 0.28180529 0.04126495  
## SkinThickness 0.24425015 0.50430728 0.14297708  
## Insulin 1.00000000 0.19785906 0.18507093  
## BMI 0.19785906 1.00000000 0.14064695  
## DiabetesPedigreeFunction 0.18507093 0.14064695 1.00000000  
## Age -0.04216295 0.03624187 0.03356131  
## Age  
## Pregnancies 0.54434123  
## Glucose 0.26351432  
## BloodPressure 0.23952795  
## SkinThickness 0.05451384  
## Insulin -0.04216295  
## BMI 0.03624187  
## DiabetesPedigreeFunction 0.03356131  
## Age 1.00000000  
##   
## $corrPos  
## xName yName x y corr  
## 1 Pregnancies Pregnancies 1 8 1.00000000  
## 2 Pregnancies Glucose 1 7 0.12945867  
## 3 Pregnancies BloodPressure 1 6 0.14128198  
## 4 Pregnancies SkinThickness 1 5 0.03256830  
## 5 Pregnancies Insulin 1 4 -0.07353461  
## 6 Pregnancies BMI 1 3 0.01768309  
## 7 Pregnancies DiabetesPedigreeFunction 1 2 -0.03352267  
## 8 Pregnancies Age 1 1 0.54434123  
## 9 Glucose Pregnancies 2 8 0.12945867  
## 10 Glucose Glucose 2 7 1.00000000  
## 11 Glucose BloodPressure 2 6 0.15258959  
## 12 Glucose SkinThickness 2 5 0.15802676  
## 13 Glucose Insulin 2 4 0.33135711  
## 14 Glucose BMI 2 3 0.22107107  
## 15 Glucose DiabetesPedigreeFunction 2 2 0.13733730  
## 16 Glucose Age 2 1 0.26351432  
## 17 BloodPressure Pregnancies 3 8 0.14128198  
## 18 BloodPressure Glucose 3 7 0.15258959  
## 19 BloodPressure BloodPressure 3 6 1.00000000  
## 20 BloodPressure SkinThickness 3 5 0.16842113  
## 21 BloodPressure Insulin 3 4 0.08893338  
## 22 BloodPressure BMI 3 3 0.28180529  
## 23 BloodPressure DiabetesPedigreeFunction 3 2 0.04126495  
## 24 BloodPressure Age 3 1 0.23952795  
## 25 SkinThickness Pregnancies 4 8 0.03256830  
## 26 SkinThickness Glucose 4 7 0.15802676  
## 27 SkinThickness BloodPressure 4 6 0.16842113  
## 28 SkinThickness SkinThickness 4 5 1.00000000  
## 29 SkinThickness Insulin 4 4 0.24425015  
## 30 SkinThickness BMI 4 3 0.50430728  
## 31 SkinThickness DiabetesPedigreeFunction 4 2 0.14297708  
## 32 SkinThickness Age 4 1 0.05451384  
## 33 Insulin Pregnancies 5 8 -0.07353461  
## 34 Insulin Glucose 5 7 0.33135711  
## 35 Insulin BloodPressure 5 6 0.08893338  
## 36 Insulin SkinThickness 5 5 0.24425015  
## 37 Insulin Insulin 5 4 1.00000000  
## 38 Insulin BMI 5 3 0.19785906  
## 39 Insulin DiabetesPedigreeFunction 5 2 0.18507093  
## 40 Insulin Age 5 1 -0.04216295  
## 41 BMI Pregnancies 6 8 0.01768309  
## 42 BMI Glucose 6 7 0.22107107  
## 43 BMI BloodPressure 6 6 0.28180529  
## 44 BMI SkinThickness 6 5 0.50430728  
## 45 BMI Insulin 6 4 0.19785906  
## 46 BMI BMI 6 3 1.00000000  
## 47 BMI DiabetesPedigreeFunction 6 2 0.14064695  
## 48 BMI Age 6 1 0.03624187  
## 49 DiabetesPedigreeFunction Pregnancies 7 8 -0.03352267  
## 50 DiabetesPedigreeFunction Glucose 7 7 0.13733730  
## 51 DiabetesPedigreeFunction BloodPressure 7 6 0.04126495  
## 52 DiabetesPedigreeFunction SkinThickness 7 5 0.14297708  
## 53 DiabetesPedigreeFunction Insulin 7 4 0.18507093  
## 54 DiabetesPedigreeFunction BMI 7 3 0.14064695  
## 55 DiabetesPedigreeFunction DiabetesPedigreeFunction 7 2 1.00000000  
## 56 DiabetesPedigreeFunction Age 7 1 0.03356131  
## 57 Age Pregnancies 8 8 0.54434123  
## 58 Age Glucose 8 7 0.26351432  
## 59 Age BloodPressure 8 6 0.23952795  
## 60 Age SkinThickness 8 5 0.05451384  
## 61 Age Insulin 8 4 -0.04216295  
## 62 Age BMI 8 3 0.03624187  
## 63 Age DiabetesPedigreeFunction 8 2 0.03356131  
## 64 Age Age 8 1 1.00000000  
##   
## $arg  
## $arg$type  
## [1] "full"

#In this analysis, we used the diabetic patient health management follow-up data  
#We have combined feature selection and imbalanced processing techniques.

