Performing EDA and Model implementation

Harini G

Problem Description:

The dataset is about the health conductions of people.

Based on the Blood pressure, sugar levels, BMI we have to predict the outcome (0 and 1) i.e., if the person is diseased or not .

Implementing the Machine Algorithms to predict the Outcomes(0 and 1)

Lets us consider class 0: No disease and class 1: Disease

Data Understanding:

- The data is well structured.
- The dataset contains 8 columns and 680 observations.
- The columns in the dataset are integer and numeric in data type.
- There are missing values present in the dataset.
- There are outliers present in the dataset.
- There is no linear relation between any of the variables.
- No variable is following normal distribution.

```
library(pastecs)

## Warning: package 'pastecs' was built under R version 4.0.5

library(ggplot2)
library(car)

## Warning: package 'car' was built under R version 4.0.4

## Loading required package: carData

library(caTools)

## Warning: package 'caTools' was built under R version 4.0.4

library("randomForest")
```

```
## Warning: package 'randomForest' was built under R version 4.0.4

## randomForest 4.6-14

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

##
## Attaching package: 'randomForest'

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

library(rpart)

## Warning: package 'rpart' was built under R version 4.0.4

library(rpart.plot)

## Warning: package 'rpart.plot' was built under R version 4.0.4

library(caret)

## Loading required package: lattice

library(ROSE)

## Warning: package 'ROSE' was built under R version 4.0.5

## Loaded ROSE 0.0-3

library(ROCR)
library(dplyr)

## Warning: package 'dplyr' was built under R version 4.0.5

##
## Attaching package: 'dplyr'

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

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

## The following objects are masked from 'package:pastecs':
##
##     first, last

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

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

data=read.csv("D:/Harini(christ unniversity)/2nd sem subjects/R/DublinTest
dataset.csv")
head(data)

##   BloodPressure RBS FBS Serum.Insulin  BMI   BUN Age Outcome
## 1           117  92   0             0 34.1 0.337  38       0
## 2           109  75  26             0 36.0 0.546  60       0
## 3           158  76  36           245 31.6 0.851  28       1
## 4            88  58  11            54 24.8 0.267  22       0
## 5            92  92   0             0 19.9 0.188  28       0
## 6           122  78  31             0 27.6 0.512  45       0

str(data)

## 'data.frame':    680 obs. of  8 variables:
##  $ BloodPressure: int  117 109 158 88 92 122 103 138 102 90 ...
##  $ RBS          : int  92 75 76 58 92 78 60 76 76 68 ...
##  $ FBS          : int  0 26 36 11 0 31 33 0 37 42 ...
##  $ Serum.Insulin: int  0 0 245 54 0 0 192 0 0 0 ...
##  $ BMI          : num  34.1 36 31.6 24.8 19.9 27.6 24 33.2 32.9 38.2 ...
##  $ BUN          : num  0.337 0.546 0.851 0.267 0.188 0.512 0.966 0.42
0.665 0.503 ...
##  $ Age          : int  38 60 28 22 28 45 33 35 46 27 ...
##  $ Outcome      : int  0 0 1 0 0 0 0 0 1 1 ...

Inference: Many of the variables are in integer and numeric in data type

summary(data)

##  BloodPressure        RBS              FBS        Serum.Insulin
##  Min.   :  0.0   Min.   :  0.0   Min.   : 0.00   Min.   :  0.00
##  1st Qu.: 99.0   1st Qu.: 64.0   1st Qu.: 0.00   1st Qu.:  0.00
##  Median :117.0   Median : 72.0   Median :23.00   Median : 30.50
##  Mean   :120.9   Mean   : 69.1   Mean   :20.64   Mean   : 80.05
##  3rd Qu.:141.0   3rd Qu.: 80.0   3rd Qu.:32.25   3rd Qu.:125.00
##  Max.   :198.0   Max.   :122.0   Max.   :99.00   Max.   :846.00
##       BMI             BUN              Age          Outcome
##  Min.   : 0.00   Min.   :0.0780   Min.   :21.00   Min.   :0.0000
##  1st Qu.:27.30   1st Qu.:0.2487   1st Qu.:24.00   1st Qu.:0.0000
##  Median :32.00   Median :0.3815   Median :30.00   Median :0.0000
##  Mean   :32.05   Mean   :0.4782   Mean   :33.59   Mean   :0.3632
##  3rd Qu.:36.60   3rd Qu.:0.6275   3rd Qu.:41.00   3rd Qu.:1.0000
##  Max.   :67.10   Max.   :2.4200   Max.   :81.00   Max.   :1.0000

Inference: The survey is conducted between the age to 81

stat.desc(data)
```

```
##              BloodPressure          RBS          FBS Serum.Insulin
BMI
## nbr.val       6.800000e+02 6.800000e+02 6.800000e+02    680.000000
6.800000e+02
## nbr.null      5.000000e+00 3.000000e+01 2.010000e+02    332.000000
1.000000e+01
## nbr.na        0.000000e+00 0.000000e+00 0.000000e+00      0.000000
0.000000e+00
## min           0.000000e+00 0.000000e+00 0.000000e+00      0.000000
0.000000e+00
## max           1.980000e+02 1.220000e+02 9.900000e+01    846.000000
6.710000e+01
## range         1.980000e+02 1.220000e+02 9.900000e+01    846.000000
6.710000e+01
## sum           8.219800e+04 4.698800e+04 1.403500e+04  54431.000000
2.179100e+04
## median        1.170000e+02 7.200000e+01 2.300000e+01     30.500000
3.200000e+01
## mean          1.208794e+02 6.910000e+01 2.063971e+01     80.045588
3.204559e+01
## SE.mean       1.250666e+00 7.341221e-01 6.148216e-01      4.575182
3.036444e-01
## CI.mean.0.95  2.455637e+00 1.441422e+00 1.207180e+00      8.983204
5.961948e-01
## var           1.063632e+03 3.664760e+02 2.570438e+02  14233.955209
6.269595e+01
## std.dev       3.261337e+01 1.914356e+01 1.603259e+01    119.306141
7.918077e+00
## coef.var      2.698009e-01 2.770414e-01 7.767836e-01      1.490477
2.470879e-01
##                       BUN          Age      Outcome
## nbr.val      680.00000000 6.800000e+02 680.00000000
## nbr.null       0.00000000 0.000000e+00 433.00000000
## nbr.na         0.00000000 0.000000e+00   0.00000000
## min            0.07800000 2.100000e+01   0.00000000
## max            2.42000000 8.100000e+01   1.00000000
## range          2.34200000 6.000000e+01   1.00000000
## sum          325.20200000 2.283800e+04 247.00000000
## median         0.38150000 3.000000e+01   0.00000000
## mean           0.47823824 3.358529e+01   0.36323529
## SE.mean        0.01293286 4.536496e-01   0.01845647
## CI.mean.0.95   0.02539321 8.907247e-01   0.03623861
## var            0.11373606 1.399426e+02   0.23163606
## std.dev        0.33724777 1.182974e+01   0.48128584
## coef.var       0.70518781 3.522296e-01   1.32499744
```

Inference: Using this function we can know the total, mean, standard deviation, variance, range, null values and missing values.

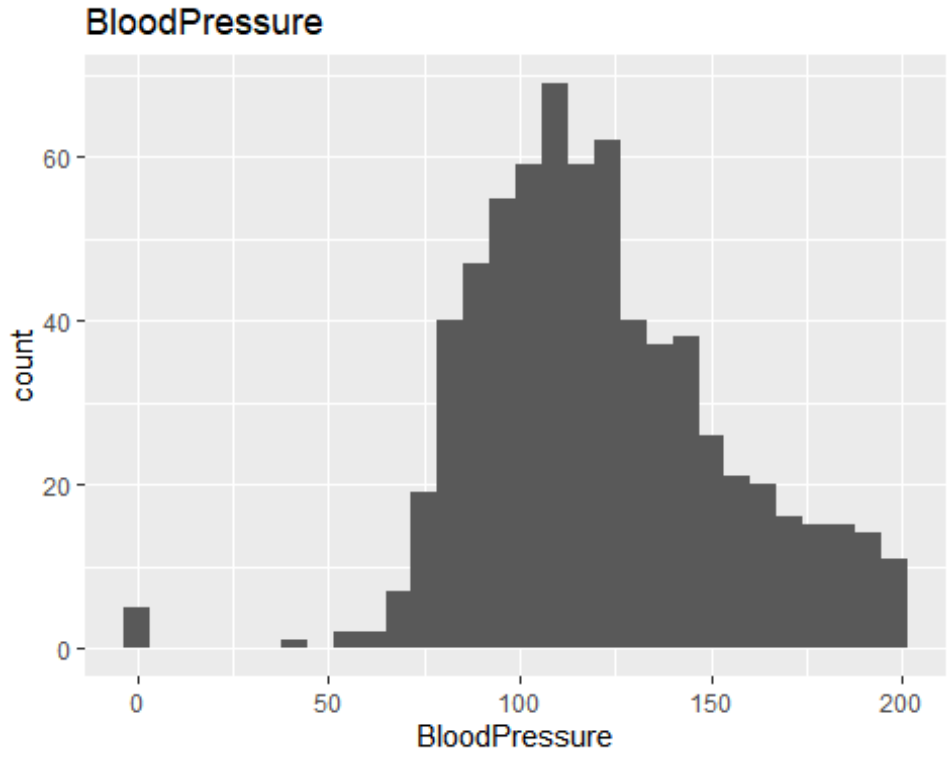
```
colSums(is.na(data))#checking if their are any null values
```

```
## BloodPressure           RBS           FBS Serum.Insulin           BMI
##             0             0             0             0             0
##           BUN           Age       Outcome
##             0             0             0
```

Inference: As we can see that there are no missing values present in the dataset.

```
data=transform(data, Outcome=as.factor(Outcome))
ggplot(data) +aes(x = BloodPressure)
+geom_histogram()+labs(title="BloodPressure")
```

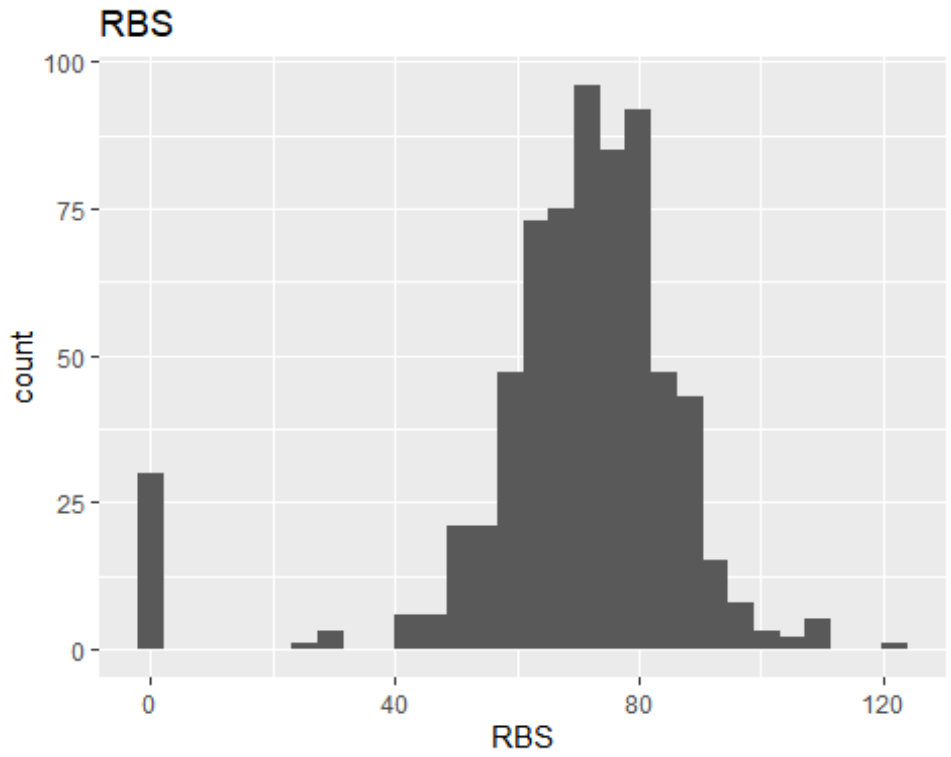
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



inference: the Blood Pressure is having Multi-Modal Distribution

```
ggplot(data) +aes(x = RBS) +geom_histogram()+labs(title="RBS")
```

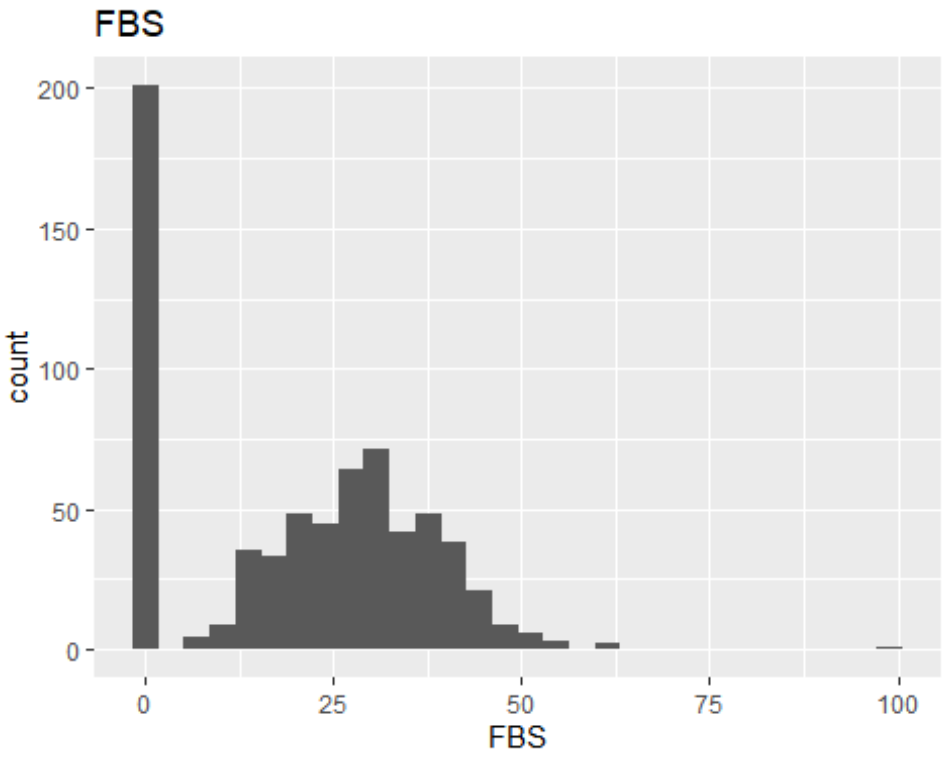
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



inference: the RBs is having Multi-Modal Distribution

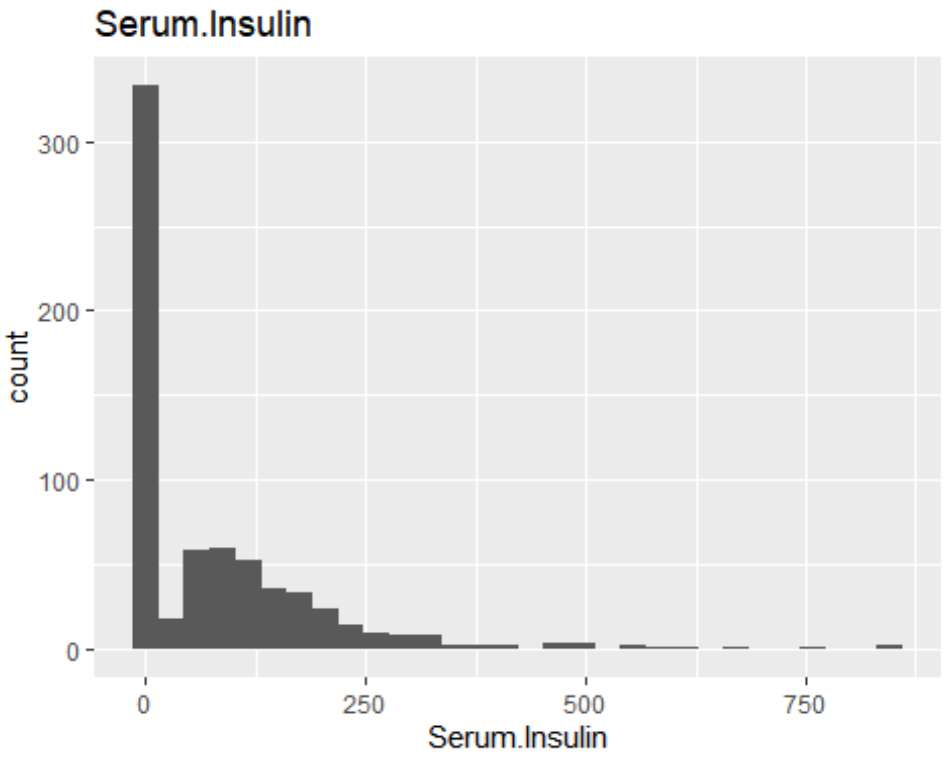
```
ggplot(data) +aes(x = FBS) +geom_histogram()+labs(title="FBS")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



inference: the FBS is having Bi-Modal Distribution

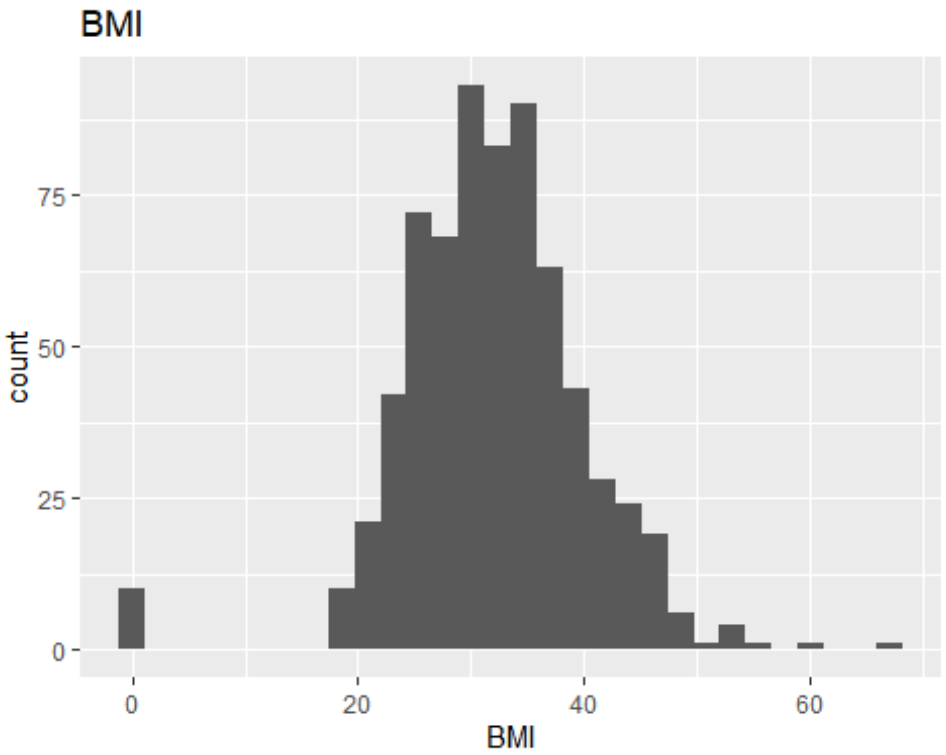
```
ggplot(data) +aes(x = Serum.Insulin)
+geom_histogram()+labs(title="Serum.Insulin")

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



inference: the Serum Insuline is left skewed

```
ggplot(data) +aes(x = BMI) +geom_histogram()+labs(title="BMI")
```

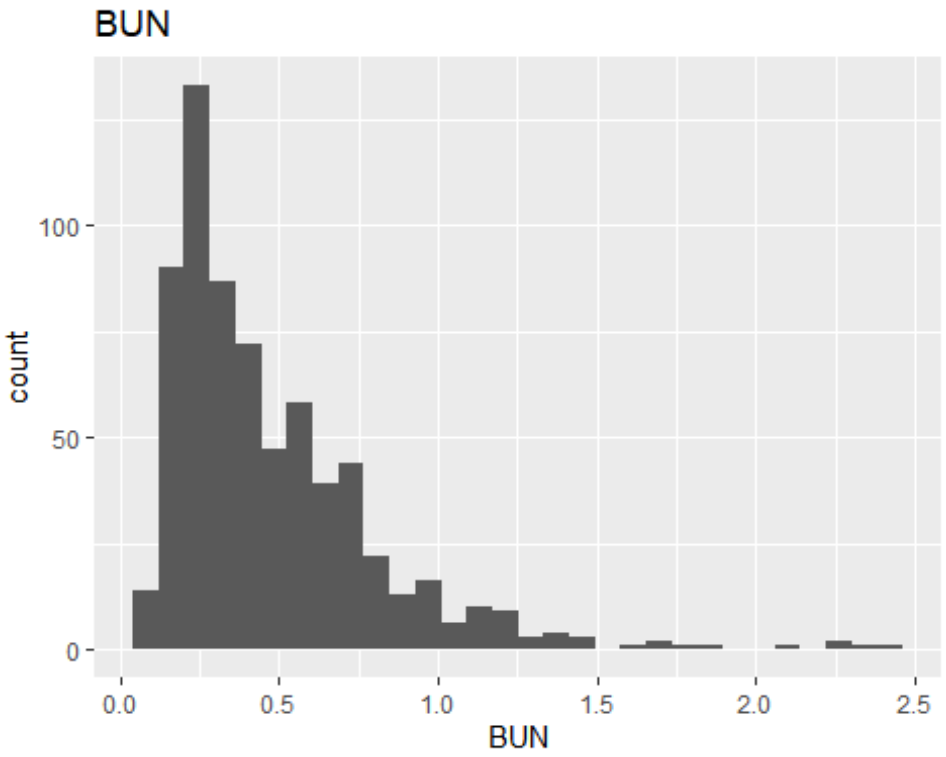
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



inference: the BMI is having Bi-Modal Distribution

```
ggplot(data) +aes(x = BUN) +geom_histogram()+labs(title="BUN")
```

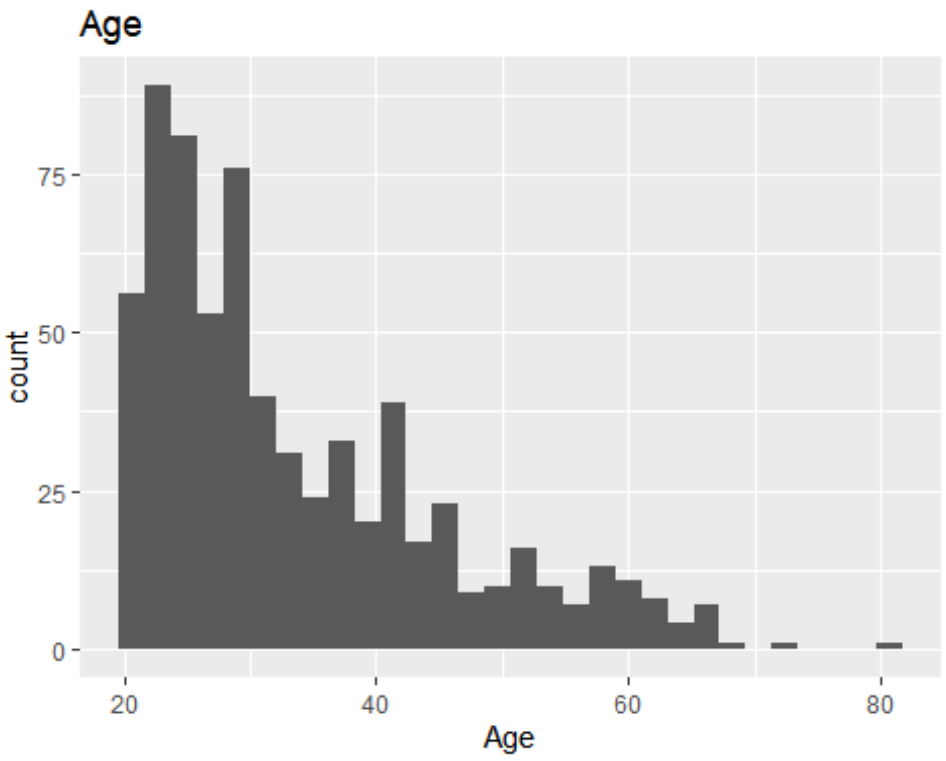
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



inference: the BUN is left skewed

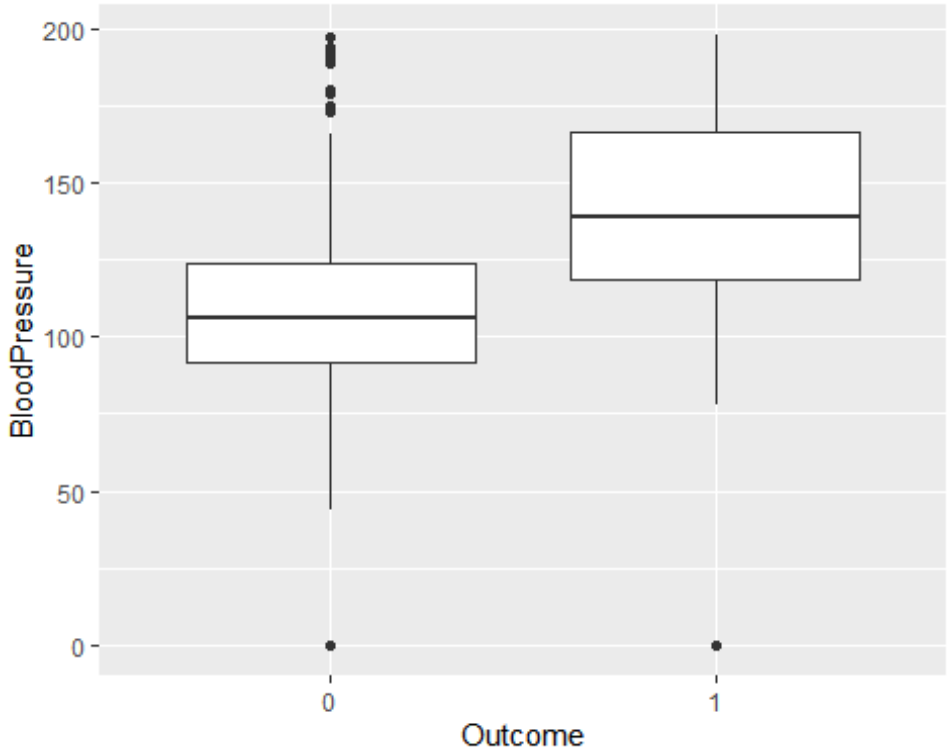
```
ggplot(data) +aes(x = Age) +geom_histogram()+labs(title="Age")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

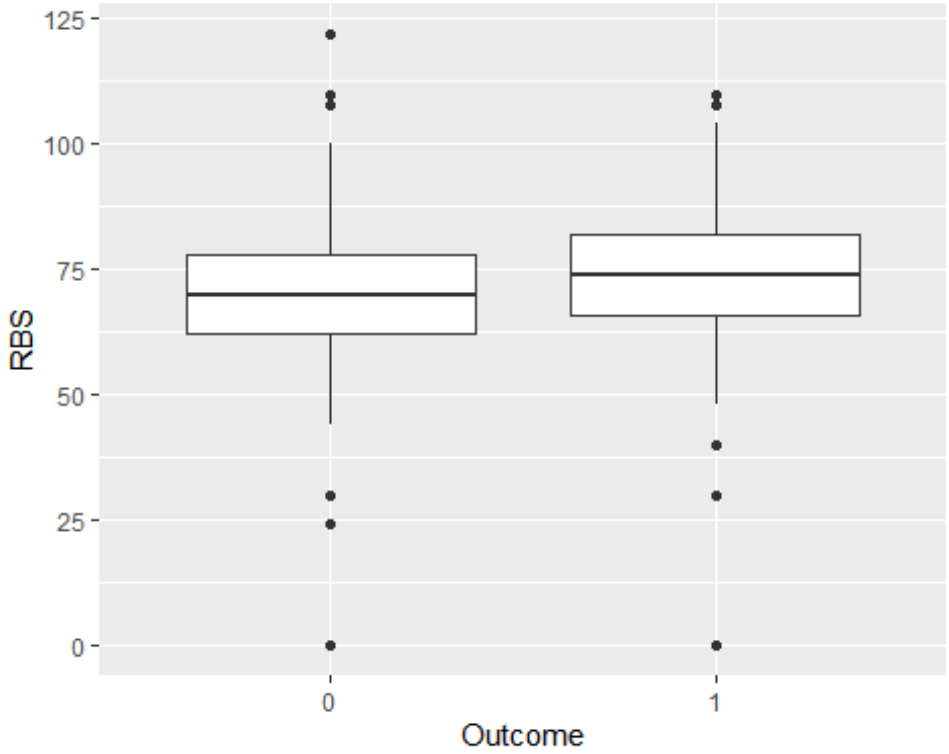


inference: the Serum Insuline is left skewed and following multi-model distribution.

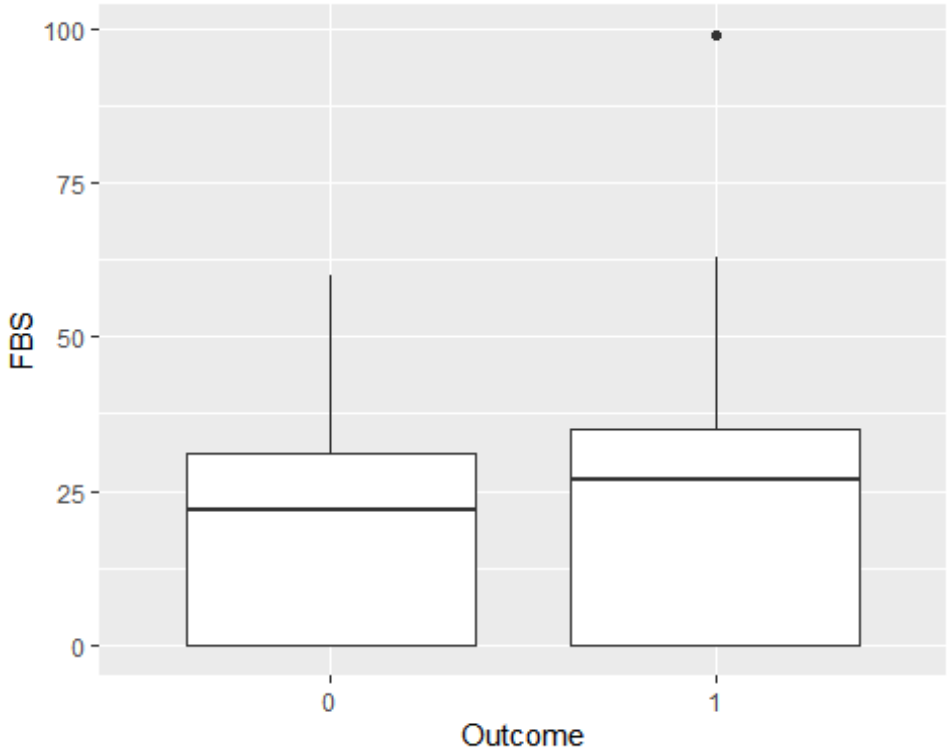
```
ggplot(data) +aes(x = Outcome, y = BloodPressure) +geom_boxplot()
```



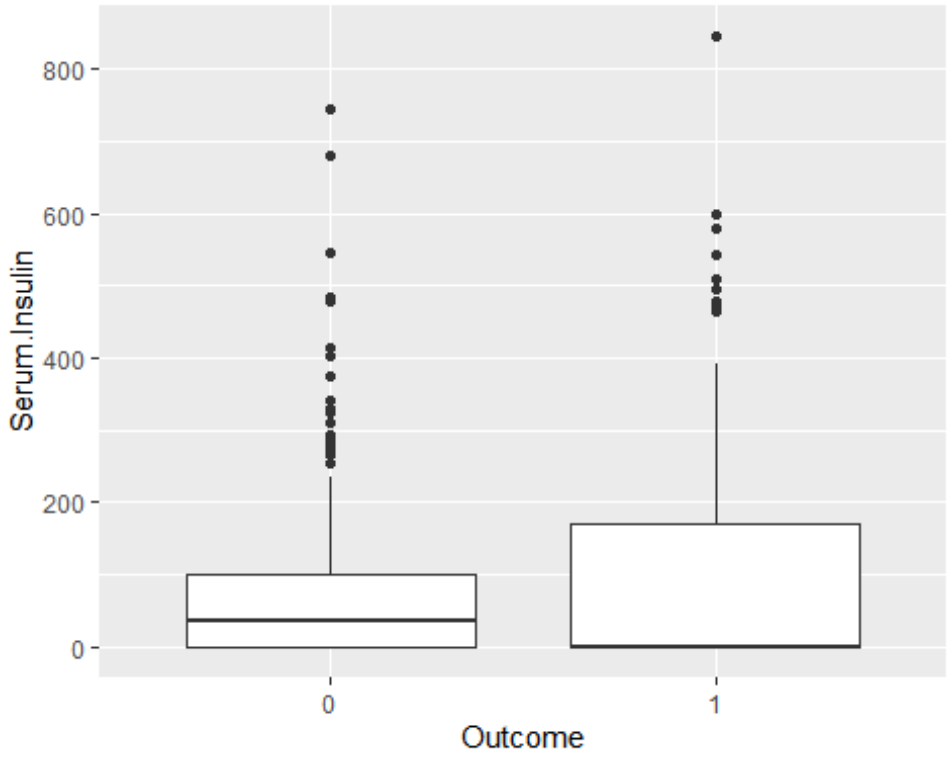

```
ggplot(data) +aes(x = Outcome, y = RBS) +geom_boxplot()
```



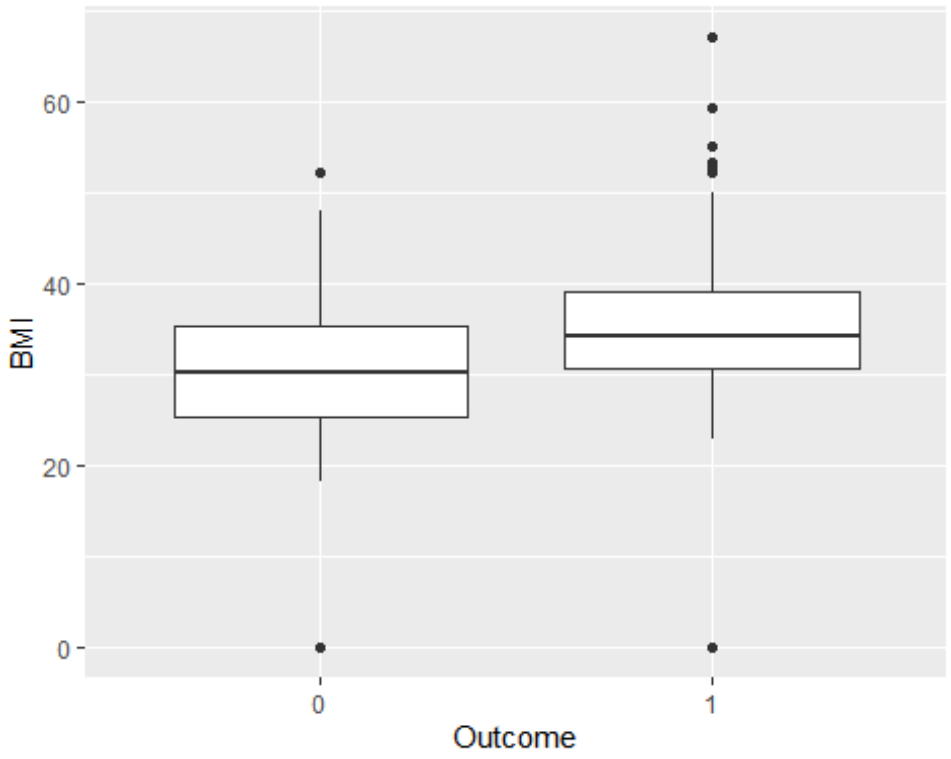

```
ggplot(data) +aes(x = Outcome, y = FBS) +geom_boxplot()
```



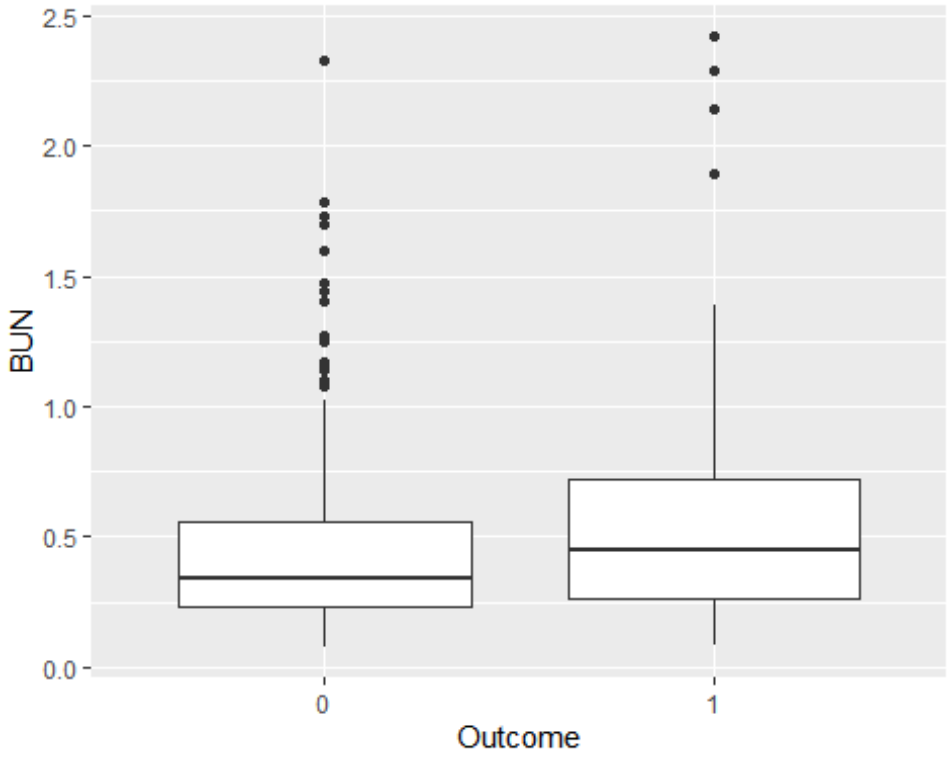

```
ggplot(data) +aes(x = Outcome, y = Serum.Insulin) +geom_boxplot()
```



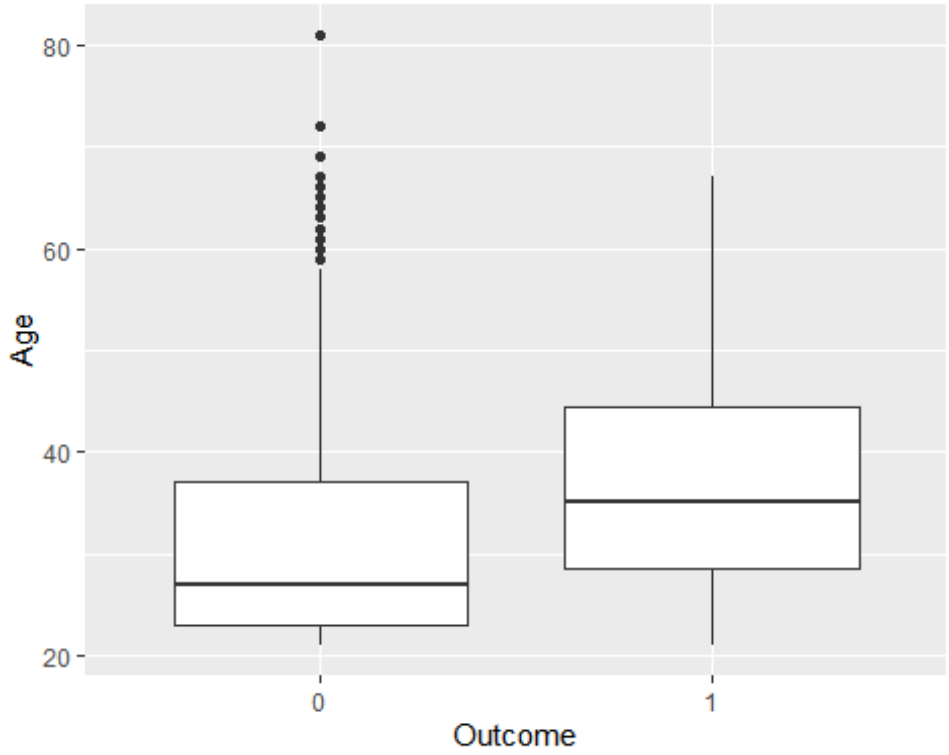

```
ggplot(data) +aes(x = Outcome, y = BMI) +geom_boxplot()
```




```
ggplot(data) +aes(x = Outcome, y = BUN) +geom_boxplot()
```

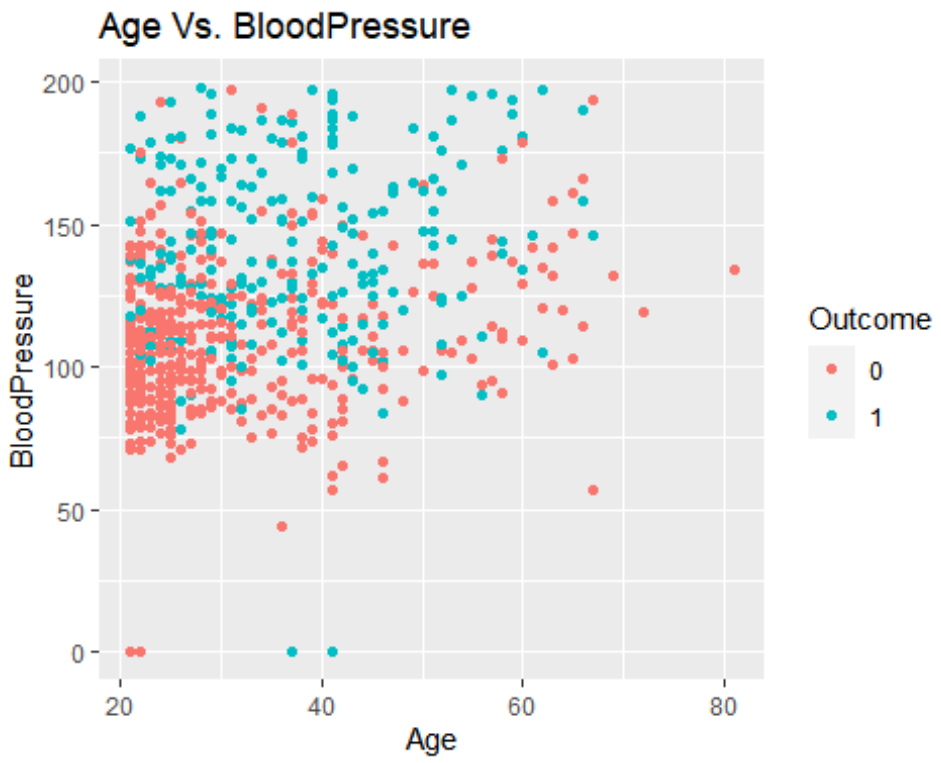



```
ggplot(data) +aes(x = Outcome, y = Age) +geom_boxplot()
```



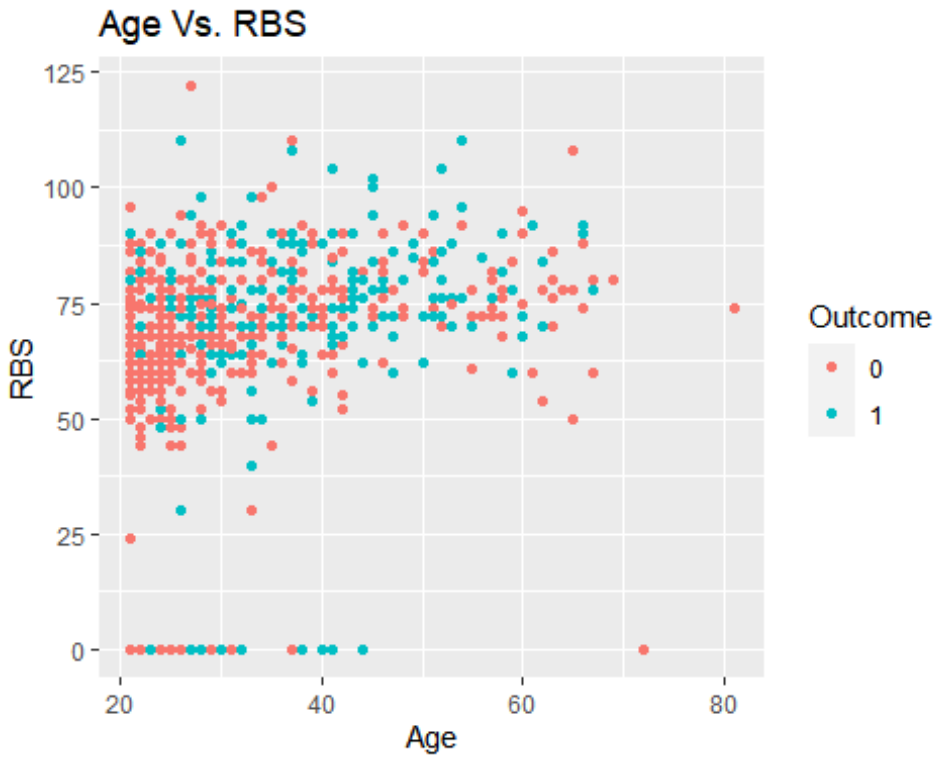
Inference: there are outliers present in almost all the columns.

```
ggplot(data) +aes(x = Age, y = BloodPressure, colour = Outcome) +geom_point()
+scale_color_hue()+labs(title="Age Vs. BloodPressure")
```



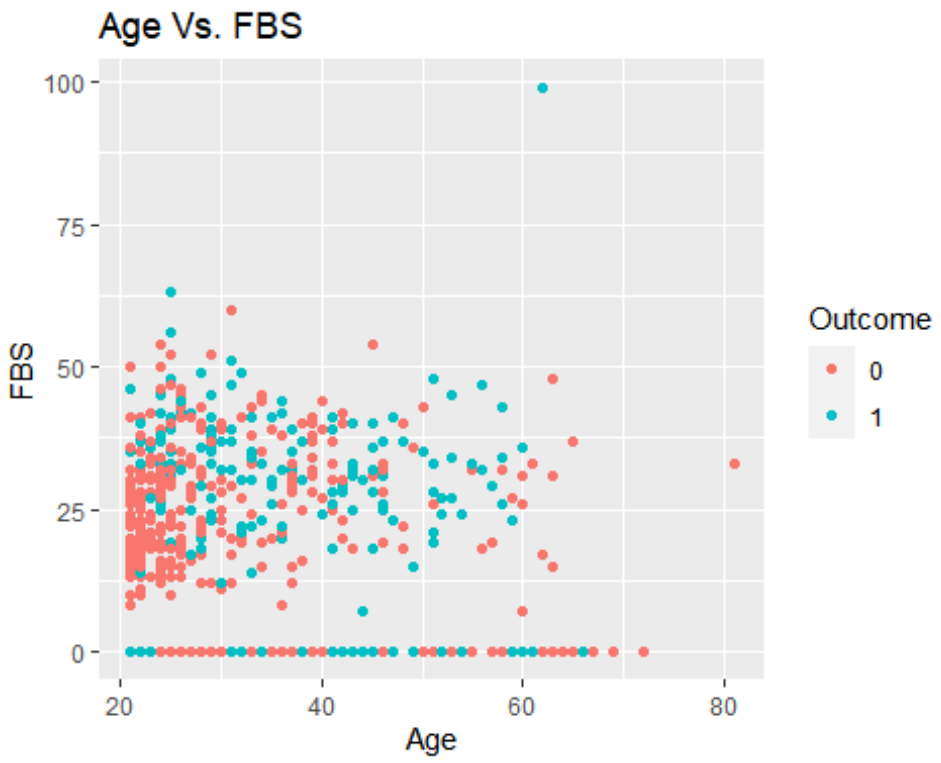
Inference: There is no linear relationship between Age and Blood Pressure

```
ggplot(data) +aes(x = Age, y = RBS, colour = Outcome) +geom_point()
+scale_color_hue()+labs(title="Age Vs. RBS")
```



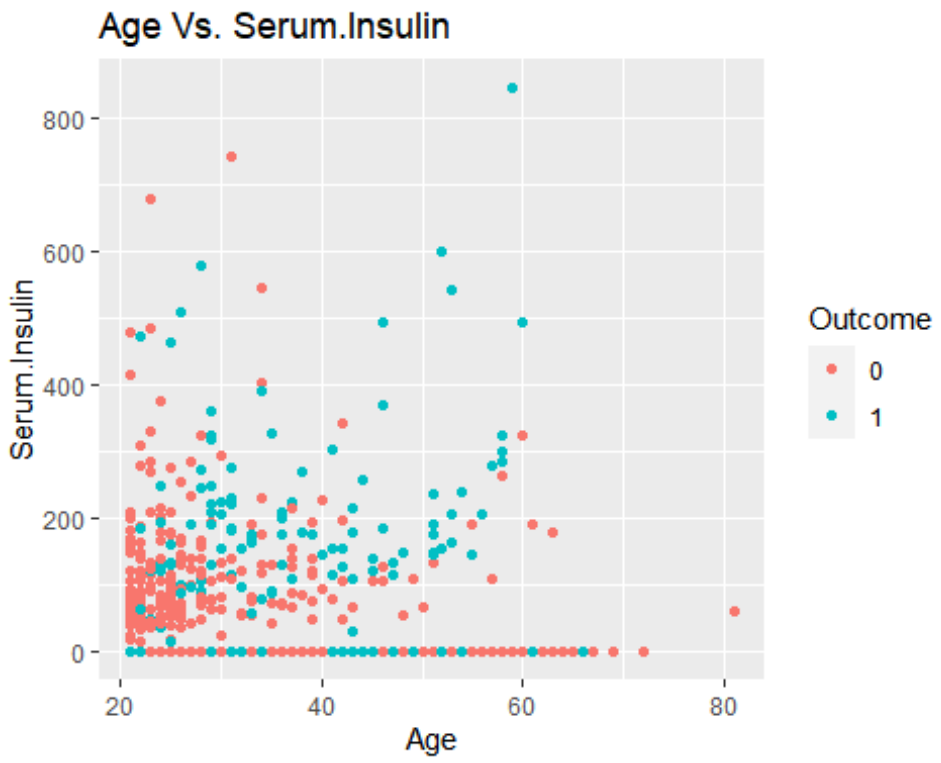
Inference: There is no linear relationship between Age and RBS

```
ggplot(data) +aes(x = Age, y = FBS, colour = Outcome) +geom_point()
+scale_color_hue()+labs(title="Age Vs. FBS")
```



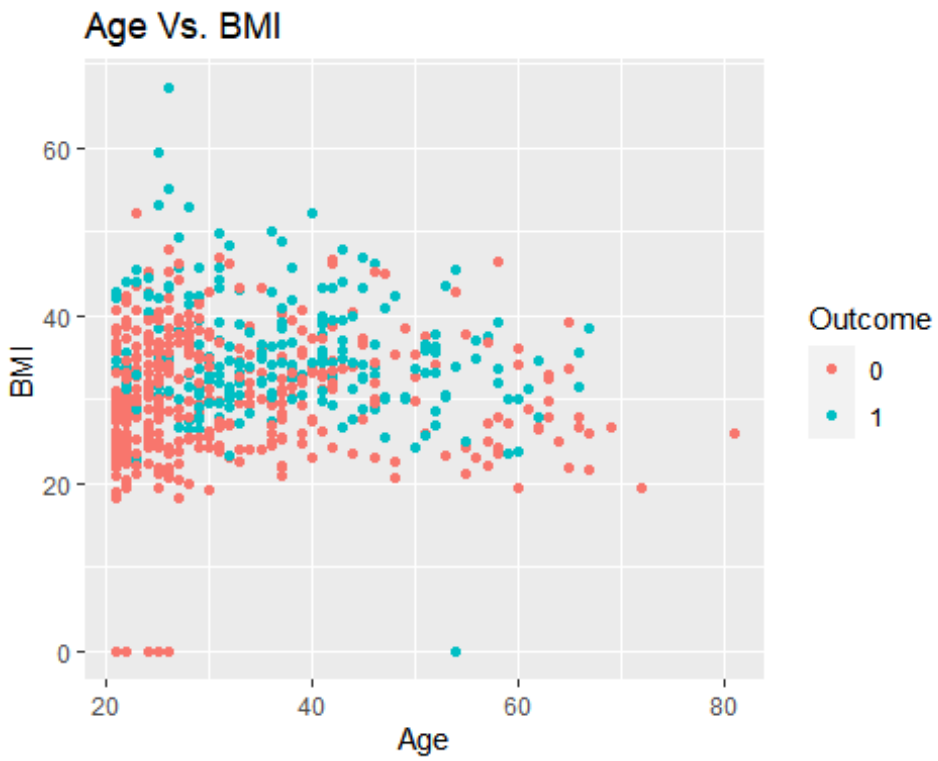
Inference: There is no linear relationship between Age and FBS

```
ggplot(data) +aes(x = Age, y = Serum.Insulin, colour = Outcome) +geom_point()
+scale_color_hue()+labs(title="Age Vs. Serum.Insulin")
```



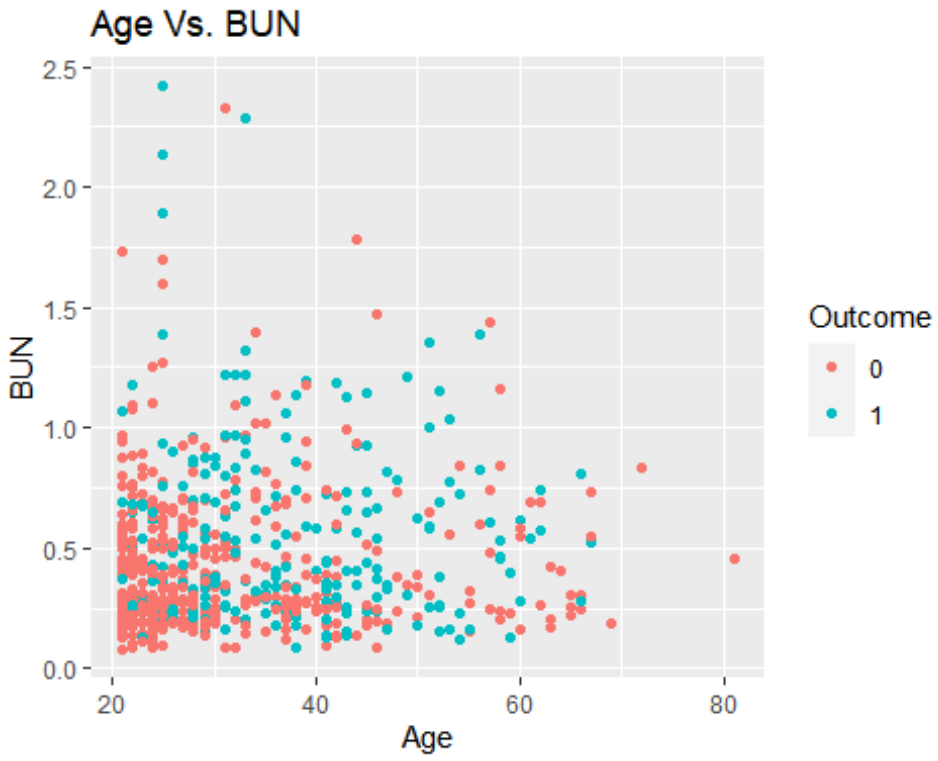
Inference: There is no linear relationship between Age and Serum.Insulin

```
ggplot(data) +aes(x = Age, y = BMI, colour = Outcome) +geom_point()
+scale_color_hue()+labs(title="Age Vs. BMI")
```



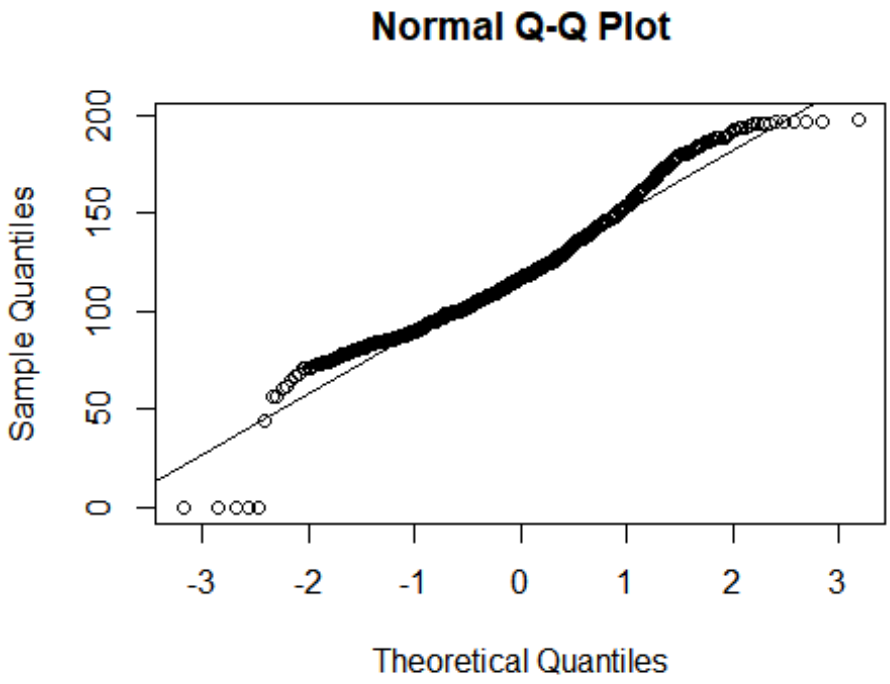
Inference: There is no linear relationship between Age and BMI

```
ggplot(data) +aes(x = Age, y = BUN, colour = Outcome) +geom_point()
+scale_color_hue()+labs(title="Age Vs. BUN")
```

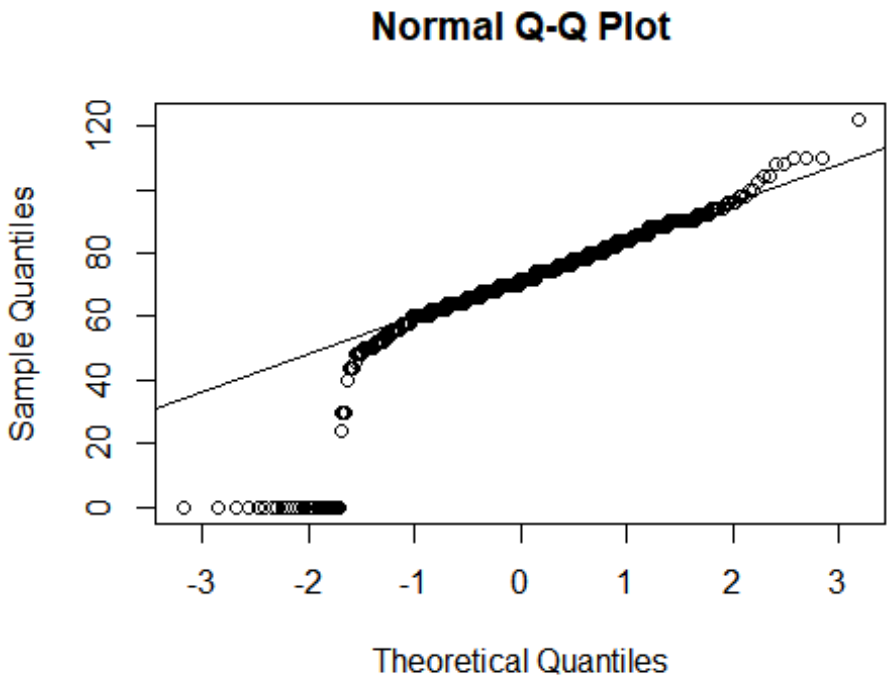


Inference: There is no linear relationship between Age and BUN

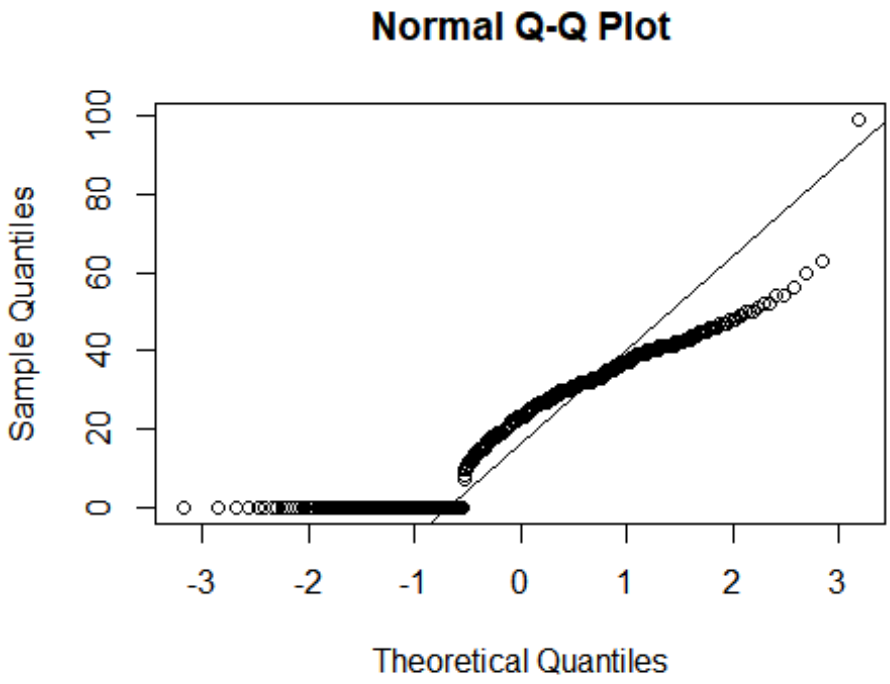
```
# Draw points on the qq-plot:
qqnorm(data$BloodPressure)
# Draw the reference line:
qqline(data$BloodPressure)
```



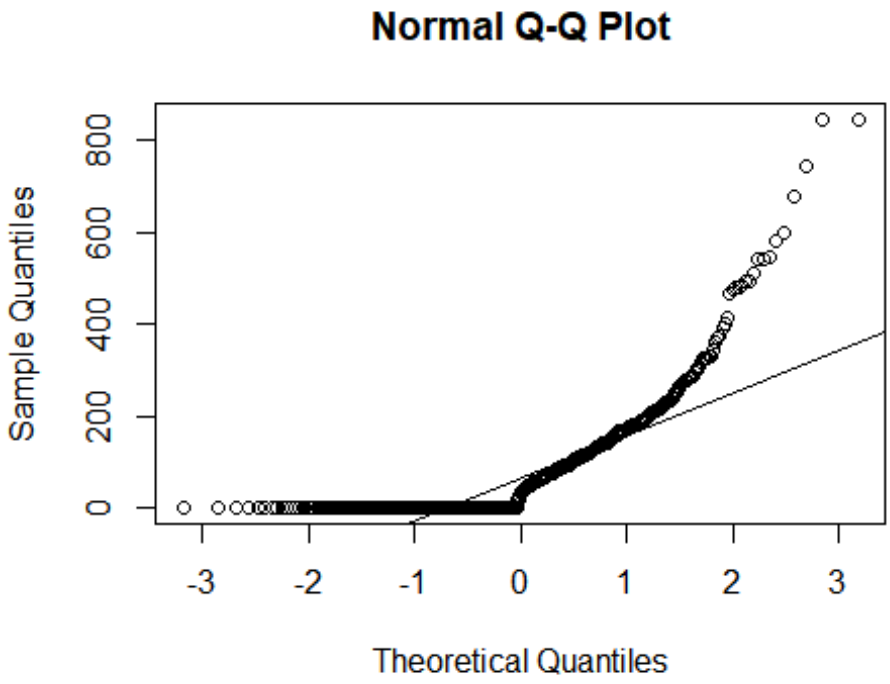

```
# Draw points on the qq-plot:
qqnorm(data$RBS)
# Draw the reference line:
qqline(data$RBS)
```



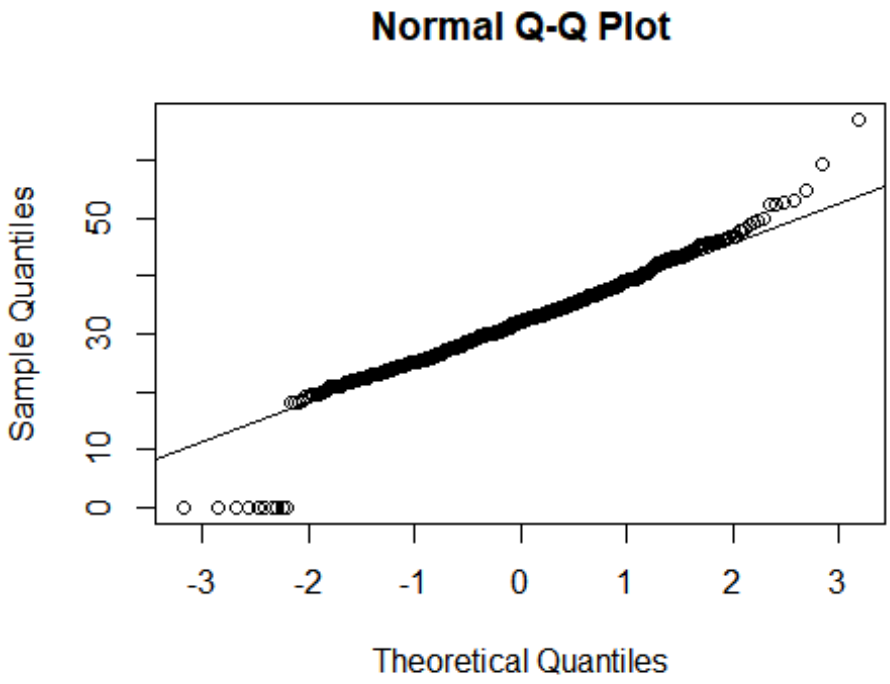

```
# Draw points on the qq-plot:
qqnorm(data$FBS)
# Draw the reference line:
qqline(data$FBS)
```



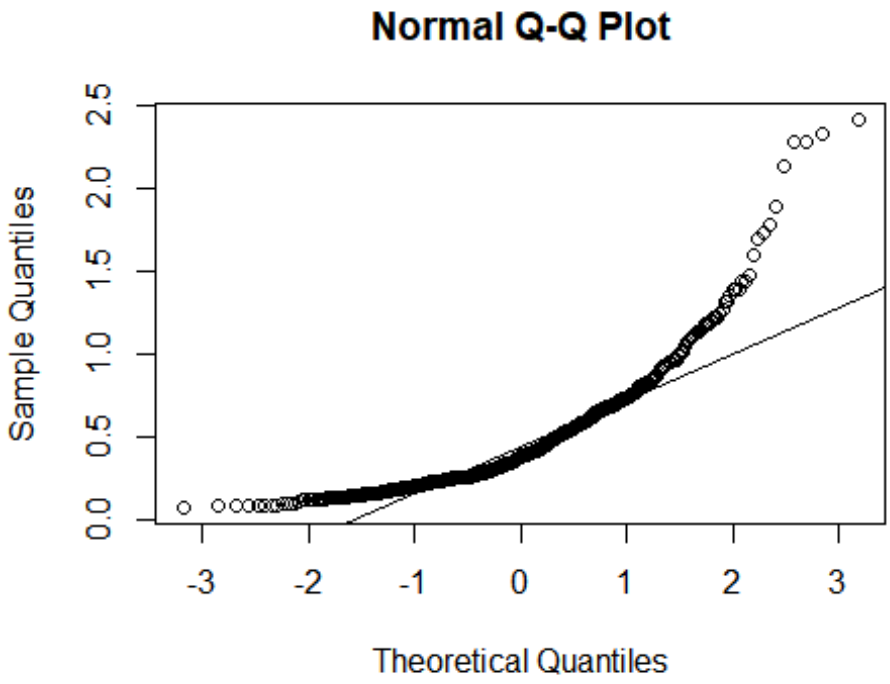

```
# Draw points on the qq-plot:
qqnorm(data$Serum.Insulin)
# Draw the reference line:
qqline(data$Serum.Insulin)
```



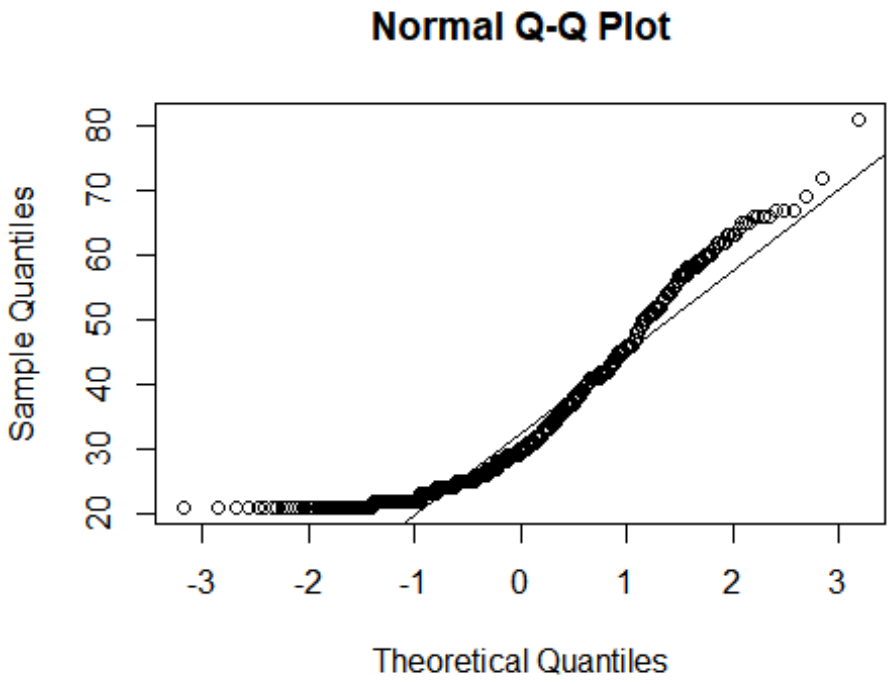

```
# Draw points on the qq-plot:
qqnorm(data$BMI)
# Draw the reference line:
qqline(data$BMI)
```




```r
# Draw points on the qq-plot:
qqnorm(data$BUN)
# Draw the reference line:
qqline(data$BUN)
```

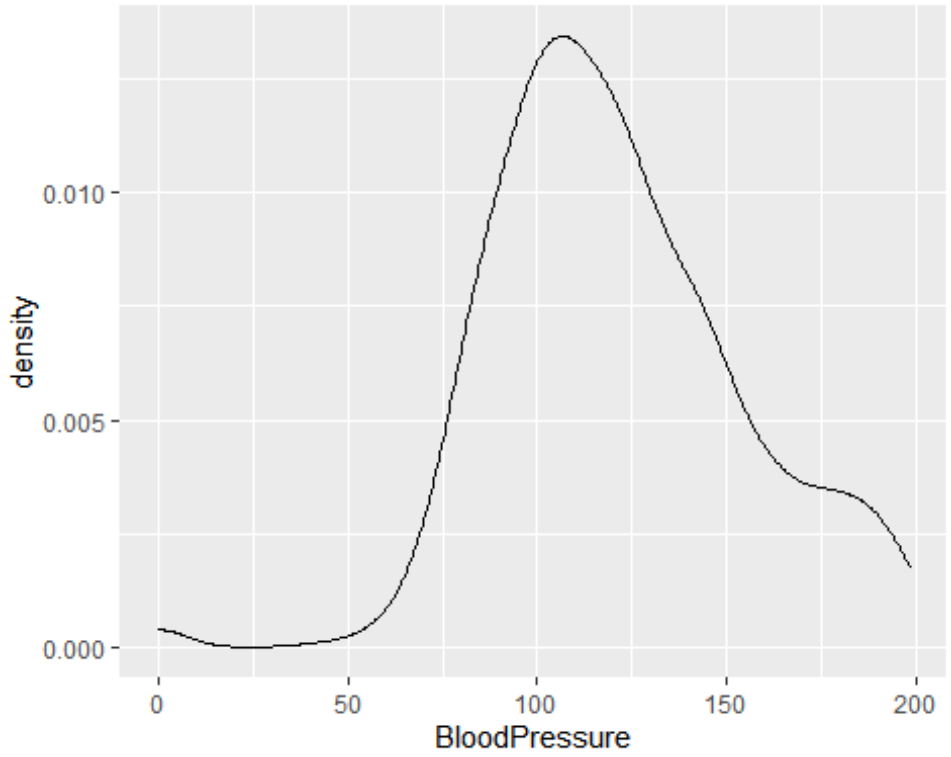



```
# Draw points on the qq-plot:
qqnorm(data$Age)
# Draw the reference line:
qqline(data$Age)
```



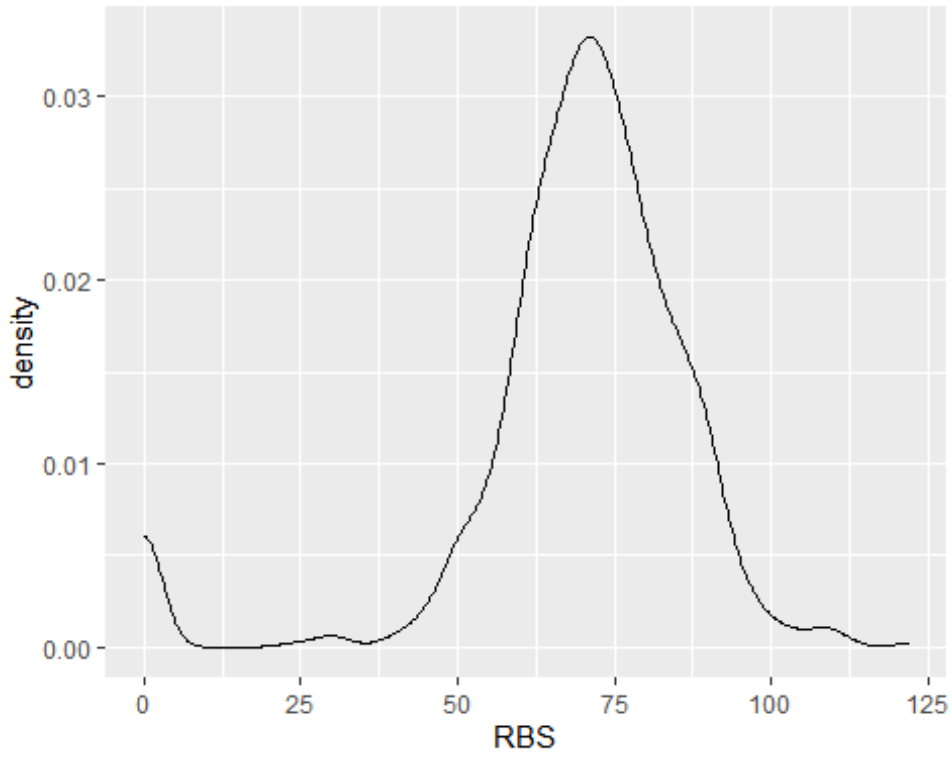
Inference: the points are not lined in a straight line, therefore the residuals are not following normally distributed.

```
ggplot(data) +aes(x = BloodPressure) +geom_density()
```



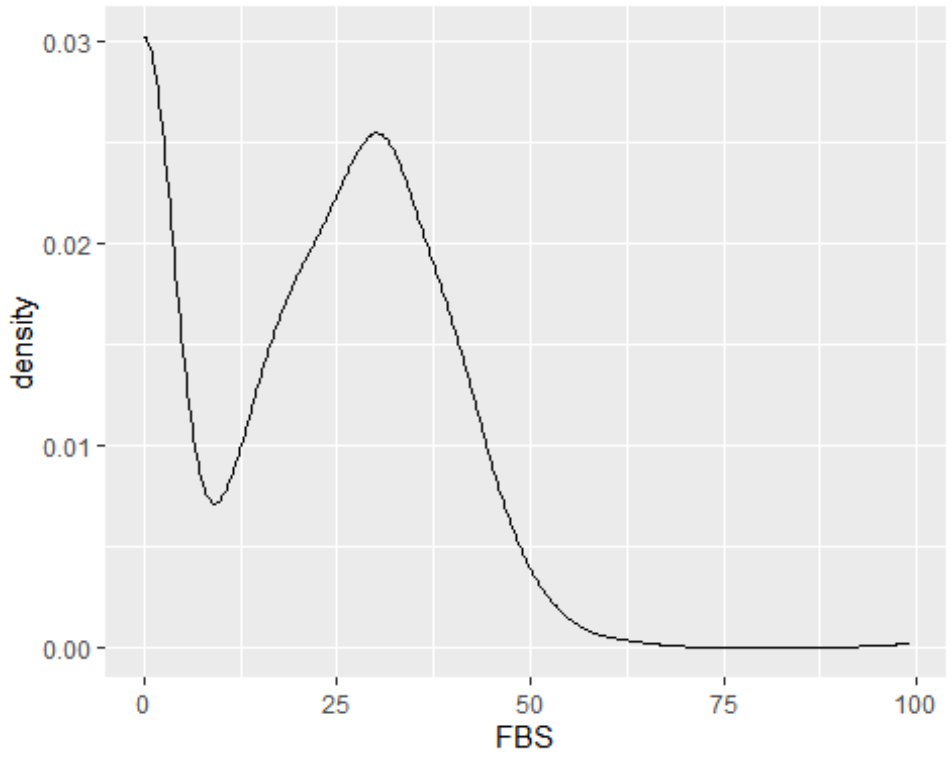
Inference: Blood Pressure column is following Multi-model distribution.

```
ggplot(data) +aes(x = RBS) +geom_density()
```



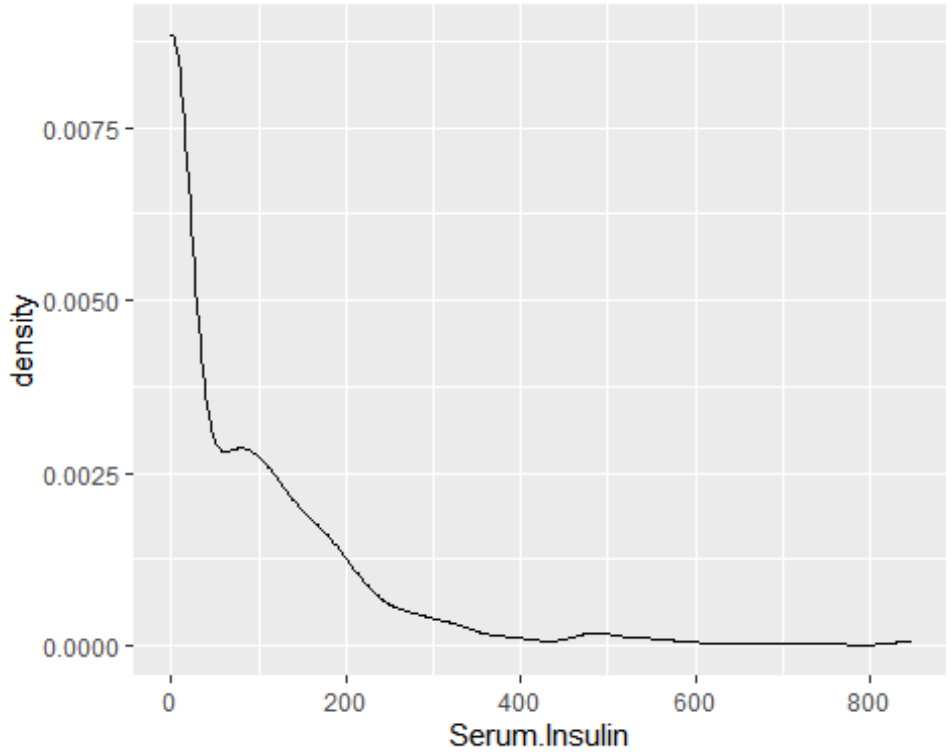
Inference: RBS column is following multi model distribution.

```
ggplot(data) +aes(x = FBS) +geom_density()
```



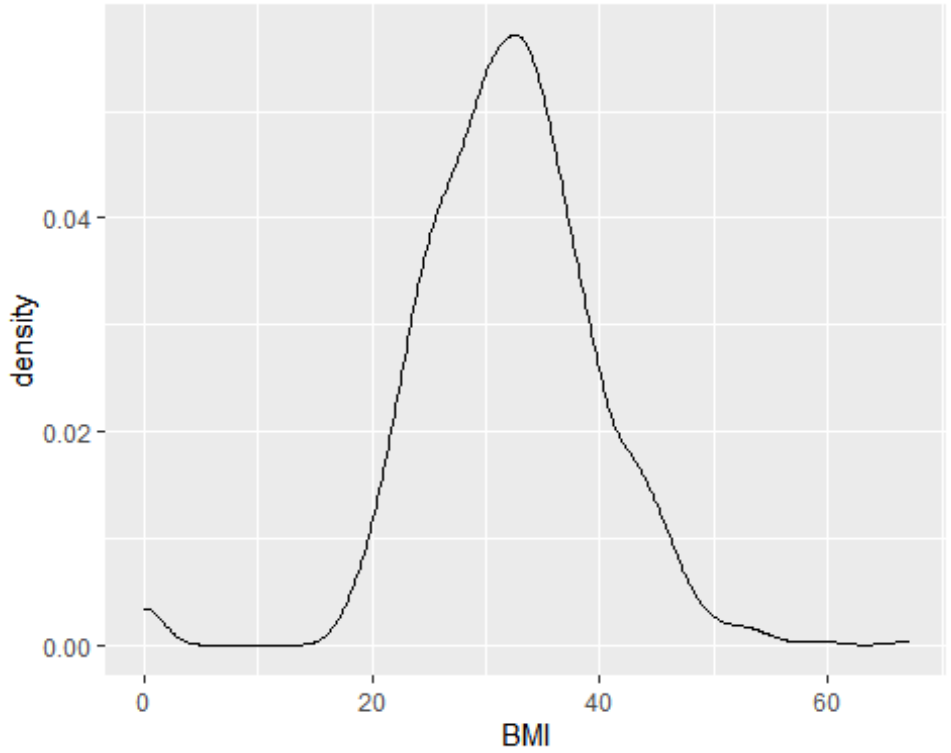
Inference: FBS column is following multi-model distribution.

```
ggplot(data) +aes(x = Serum.Insulin) +geom_density()
```



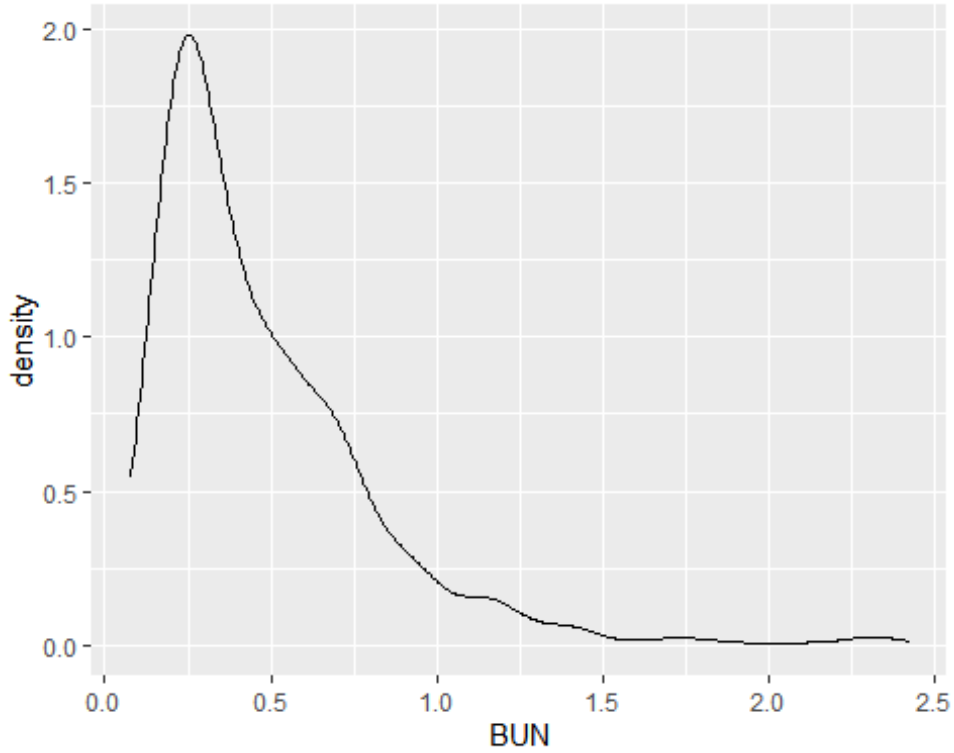
Inference: Serum.Insulin column is left skewed

```
ggplot(data) +aes(x = BMI) +geom_density()
```



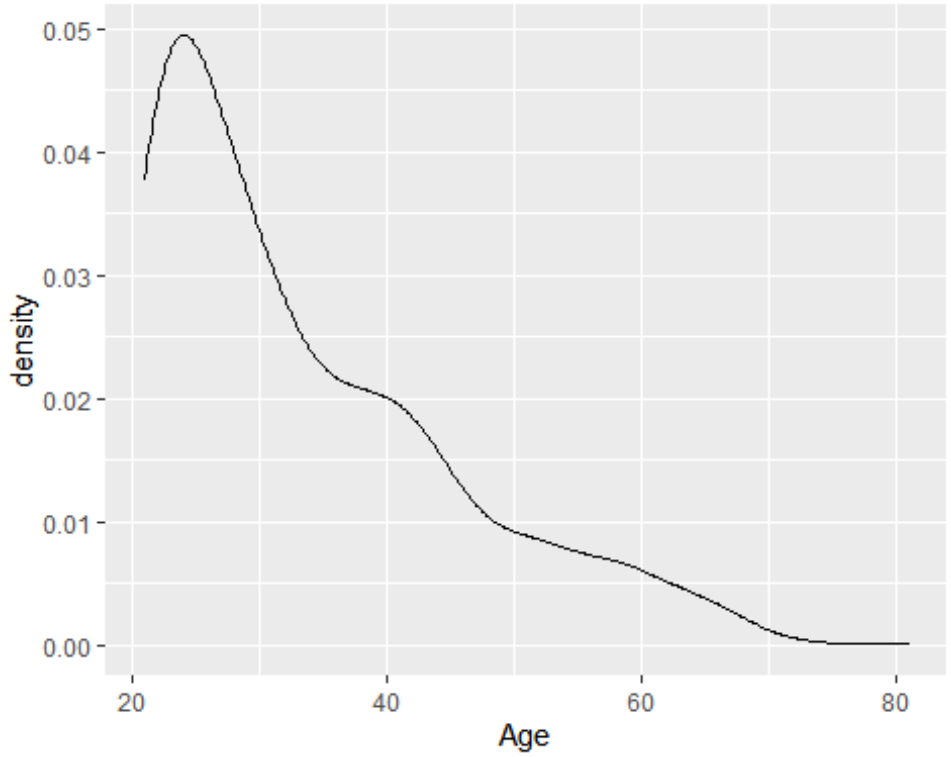
Inference: BMI column is following normal distribution.

```
ggplot(data) +aes(x = BUN) +geom_density()
```



Inference: BUN column is left skewed

```
ggplot(data) +aes(x = Age) +geom_density()
```



Inference: Age column is left skewed

Inference: from density plot also we could conclude that the data is not following normal disrtibution

```
table(data$Outcome)

##
##   0   1
## 433 247

The dataset is imbalance.

sample=sample.split(data$Outcome,SplitRatio=0.75)
train=subset(data,sample==TRUE)
test=subset(data,sample==FALSE)

splitted the data into 75% training and 25% testing


dim(train)

## [1] 510   8

The observations in the training data is 510

table(train$Outcome)
```

```
##
##   0   1
## 325 185

prop.table(table(train$Outcome))

##
##         0         1
## 0.6372549 0.3627451
```

We could clearly see that data is imbalanced

```
#over sampling
train<- ovun.sample(Outcome ~ ., data = train, method = "over",N = 650)$data
table(train$Outcome)

##
##   0   1
## 325 325
```

The imbalanced data is made balance by using Upsampling technique.

Implementing Machime Learning Algorithms.

1. Logitic regression.

```
lmModel=glm(Outcome~.,family=binomial,data=train)
summary(lmModel)

##
## Call:
## glm(formula = Outcome ~ ., family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6812  -0.8507   0.0006   0.8435   2.2948
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -7.4413562  0.7282976 -10.217  < 2e-16 ***
## BloodPressure  0.0333679  0.0038053   8.769  < 2e-16 ***
## RBS           -0.0096789  0.0058594  -1.652  0.09856 .
## FBS           -0.0093569  0.0069992  -1.337  0.18127
## Serum.Insulin -0.0007649  0.0009739  -0.785  0.43224
## BMI            0.0884107  0.0154008   5.741 9.43e-09 ***
## BUN            0.8467279  0.3276078   2.585  0.00975 **
## Age            0.0267530  0.0089193   2.999  0.00270 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 901.09  on 649  degrees of freedom
## Residual deviance: 681.80  on 642  degrees of freedom
## AIC: 697.8
##
## Number of Fisher Scoring iterations: 4
```

From here we could see that blood pressure,BMI are significiant at 0% level of significance.

Bun and age are significant at 1% level of significance.

RBS is significant at 10% LOS

Lower the null and residual deviation better the model.

```
pred<-predict(lmModel,test,type="response")
s_pred_num <- ifelse(pred > 0.5, 1, 0)
s_pred <- factor(s_pred_num, levels=c(0, 1))
s_pred
```

```
##   4   6  15  17  22  37  38  40  41  42  48  51  53  57  63  64  67  68
75  80
##   0   0   1   1   0   0   0   0   0   0   0   0   0   0   0   0   1   0
0   0
##  81  82  88  90  95  98 100 101 102 107 108 113 117 119 122 126 129 131
133 143
##   0   1   1   0   1   1   0   1   1   0   0   0   0   0   1   1   0   1
0   1
## 147 152 153 161 171 172 175 176 182 186 194 195 197 201 213 217 222 239
240 247
##   1   0   0   0   1   0   0   0   0   0   0   1   0   1   0   1   0   1
0   0
## 248 249 256 258 259 262 263 264 267 268 274 281 288 291 293 296 298 299
302 304
##   0   0   0   1   1   0   0   1   1   0   0   1   0   1   0   0   1   1
1   1
## 305 308 310 311 316 319 325 326 328 331 335 336 338 342 344 348 349 353
357 360
##   0   1   1   1   1   0   0   0   1   1   1   1   0   1   0   0   0   0
0   1
## 362 363 367 370 379 383 385 390 397 400 405 408 411 412 425 427 434 438
448 454
##   0   1   1   0   0   0   0   0   1   1   0   1   0   1   1   1   0   0
0   0
## 456 459 462 464 475 482 486 492 496 499 506 516 517 524 532 534 543 546
547 548
##   1   1   0   0   0   0   0   0   0   0   0   0   1   0   1   0   0   0
0   0
## 551 557 563 565 566 570 573 575 578 579 590 596 598 605 607 611 613 615
618 622
##   1   0   0   0   0   1   0   0   1   1   0   1   0   1   0   1   1   1
```

```
0   1
## 623 624 625 642 644 648 659 661 666 667
##   0   1   0   1   1   1   0   1   1   1
## Levels: 0 1

pscl::pR2(lmModel)["McFadden"]

## fitting null model for pseudo-r2

##  McFadden
## 0.2433635

The value is less than 0.4,the model is not best fit for the prediction.

caret::varImp(lmModel)

##                 Overall
## BloodPressure 8.7687861
## RBS           1.6518582
## FBS           1.3368494
## Serum.Insulin 0.7853678
## BMI           5.7406704
## BUN           2.5845781
## Age           2.9994476

From we can we could known that the variables Blood Pressure, BMi play
important role in prediction.

car::vif(lmModel) #Checking for multicollinearity

## BloodPressure           RBS           FBS Serum.Insulin           BMI
##      1.131711      1.206103      1.480775      1.415874      1.229166
##           BUN           Age
##      1.033329      1.166642

All the values are less than 5, it indicates no multicollinearity.

anova(lmModel, test="Chisq")

## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Outcome
##
## Terms added sequentially (first to last)
##
##
##               Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                            649     901.09
## BloodPressure  1  165.214       648     735.88 < 2.2e-16 ***
## RBS            1    0.115       647     735.76  0.734024
```

```
## FBS            1    0.039       646     735.72  0.843973
## Serum.Insulin  1    0.464       645     735.26  0.495926
## BMI            1   36.408       644     698.85 1.601e-09 ***
## BUN            1    7.868       643     690.98  0.005031 **
## Age            1    9.185       642     681.80  0.002440 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
confusionMatrix(table(train[,8], train$Outcome))
```

```
## Confusion Matrix and Statistics
##
##
##       0   1
##   0 325   0
##   1   0 325
##
##                Accuracy : 1
##                  95% CI : (0.9943, 1)
##     No Information Rate : 0.5
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
##             Sensitivity : 1.0
##             Specificity : 1.0
##          Pos Pred Value : 1.0
##          Neg Pred Value : 1.0
##              Prevalence : 0.5
##          Detection Rate : 0.5
##    Detection Prevalence : 0.5
##       Balanced Accuracy : 1.0
##
##        'Positive' Class : 0
##
```

```
The accuracy of the training data is 1
```

```
confusionMatrix(table(s_pred, test$Outcome))
```

```
## Confusion Matrix and Statistics
##
##
## s_pred  0  1
##      0 86 16
##      1 22 46
##
##                Accuracy : 0.7765
##                  95% CI : (0.7063, 0.8367)
```

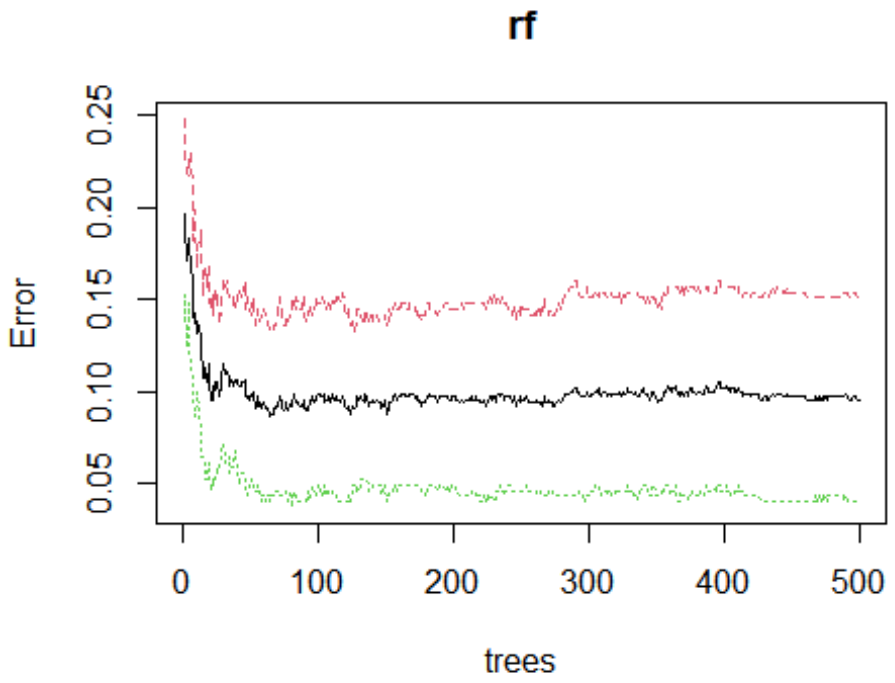
```
##     No Information Rate : 0.6353
##     P-Value [Acc > NIR] : 5.331e-05
##
##                   Kappa : 0.5274
##
##  Mcnemar's Test P-Value : 0.4173
##
##             Sensitivity : 0.7963
##             Specificity : 0.7419
##          Pos Pred Value : 0.8431
##          Neg Pred Value : 0.6765
##              Prevalence : 0.6353
##          Detection Rate : 0.5059
##    Detection Prevalence : 0.6000
##       Balanced Accuracy : 0.7691
##
##        'Positive' Class : 0
##
```

The Accuracy of the test data is 77.65%

38 observations are missed classified.

2. Random Forest

```
rf=randomForest(Outcome~.,data=train)
plot(rf)
```



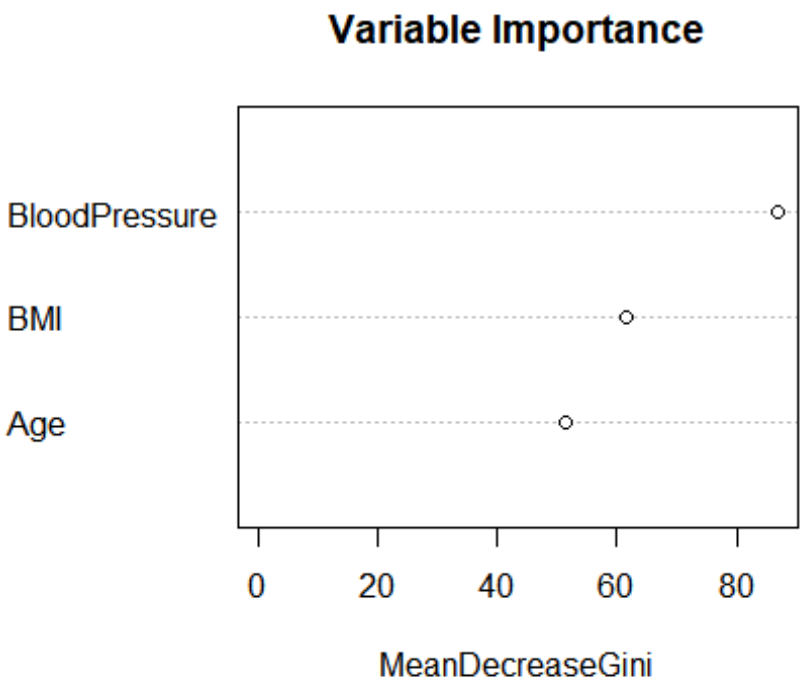
Red line represents MCR of class not having diseases,

green line represents MCR of class having diseases and

black line represents overall MCR or OOB error.

Overall error rate is what we are interested in which seems considerably good.

```
varImpPlot(rf,sort = T,main = "Variable Importance",n.var = 3)
```



From this plot we can known that blood Pressure, BMI and Age are playing important role in predicting the output.

```
var.imp <- data.frame(importance(rf,type = 2))
var.imp

##                MeanDecreaseGini
## BloodPressure         86.89067
## RBS                   28.79319
## FBS                   25.85699
## Serum.Insulin         24.83939
## BMI                   61.56190
## BUN                   43.03274
## Age                   51.46057
```

By using `MeanDecreaseGini` also we could know that BloodPressure, BMI, Age play important role in predicting the outcomes.

```
library(e1071)
rf_pred<-predict(rf,test,type="response")


confusionMatrix(table(train[,8], train$Outcome))
```

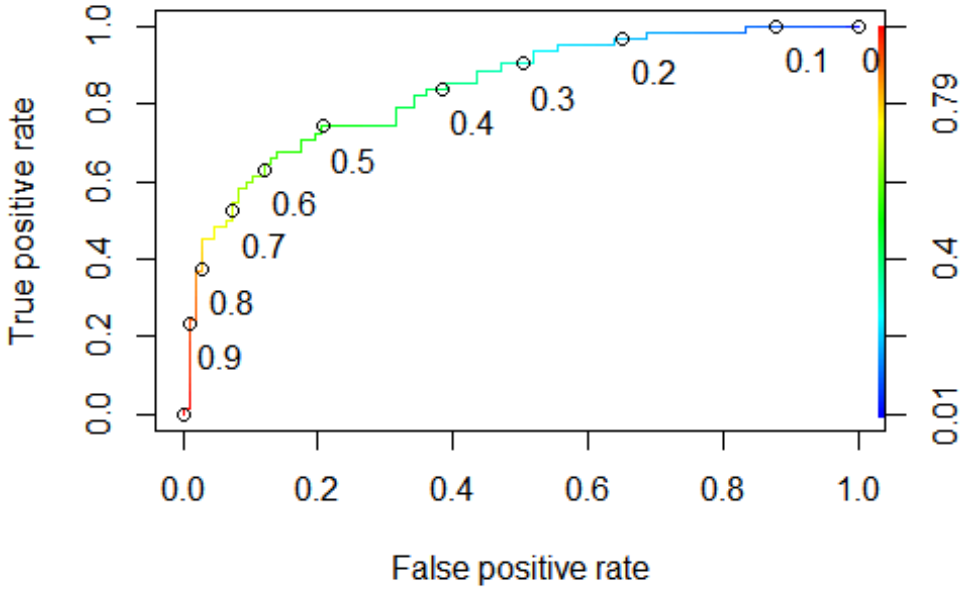
```
## Confusion Matrix and Statistics
##
##
##       0   1
##   0 325   0
##   1   0 325
##
##                Accuracy : 1
##                  95% CI : (0.9943, 1)
##     No Information Rate : 0.5
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
##             Sensitivity : 1.0
##             Specificity : 1.0
##          Pos Pred Value : 1.0
##          Neg Pred Value : 1.0
##              Prevalence : 0.5
##          Detection Rate : 0.5
##    Detection Prevalence : 0.5
##       Balanced Accuracy : 1.0
##
##        'Positive' Class : 0
##
```

```
Accuracy of the training data is 1


confusionMatrix(table(rf_pred, test$Outcome))
```

```
## Confusion Matrix and Statistics
##
##
## rf_pred  0  1
##       0 88 23
##       1 20 39
##
##                Accuracy : 0.7471
##                  95% CI : (0.6748, 0.8105)
```

```
##     No Information Rate : 0.6353
##     P-Value [Acc > NIR] : 0.001266
##
##                   Kappa : 0.4485
##
##  Mcnemar's Test P-Value : 0.760368
##
##             Sensitivity : 0.8148
##             Specificity : 0.6290
##          Pos Pred Value : 0.7928
##          Neg Pred Value : 0.6610
##              Prevalence : 0.6353
##          Detection Rate : 0.5176
##    Detection Prevalence : 0.6529
##       Balanced Accuracy : 0.7219
##
##        'Positive' Class : 0
##

Accuracy of the test data is 74.71%

43 observations are missed classified.
```

```
library(ROCR)
ROCRpred <- prediction(pred, test$Outcome)
ROCRperf <- performance(ROCRpred, measure = "tpr", x.measure = "fpr")
plot(ROCRperf, colorize = TRUE, text.adj = c(-0.2,1.7), print.cutoffs.at =
seq(0,1,0.1))
```




```
auc<-performance(ROCRpred,measure="auc")
auc<-auc@y.values[[1]]
auc
```

```
## [1] 0.8405018
```

```
The ROC, higher the curve better the model.

The area under that curve is 84%.



Conclusion: The Logistic regression provides better accuracy and there are
less mis observations in Logistic regression compared to random forest.
```