# Project_4_Navin_Chandra.R

*navin*

*Tue Mar 5 16:46:46 2019*

```
# Reading the CSV dataset

data <- read.csv("CancerData.csv", header = T, stringsAsFactors = F)

#Overall view of the data
str(data)
```

```
## 'data.frame':    569 obs. of  33 variables:
##  $ id                     : int  842302 842517 84300903 84348301 84358402 843786 844359 84458202 844
##  $ diagnosis              : chr  "M" "M" "M" "M" ...
##  $ radius_mean            : num  18 20.6 19.7 11.4 20.3 ...
##  $ texture_mean           : num  10.4 17.8 21.2 20.4 14.3 ...
##  $ perimeter_mean         : num  122.8 132.9 130 77.6 135.1 ...
##  $ area_mean              : num  1001 1326 1203 386 1297 ...
##  $ smoothness_mean        : num  0.1184 0.0847 0.1096 0.1425 0.1003 ...
##  $ compactness_mean       : num  0.2776 0.0786 0.1599 0.2839 0.1328 ...
##  $ concavity_mean         : num  0.3001 0.0869 0.1974 0.2414 0.198 ...
##  $ concave.points_mean    : num  0.1471 0.0702 0.1279 0.1052 0.1043 ...
##  $ symmetry_mean          : num  0.242 0.181 0.207 0.26 0.181 ...
##  $ fractal_dimension_mean : num  0.0787 0.0567 0.06 0.0974 0.0588 ...
##  $ radius_se              : num  1.095 0.543 0.746 0.496 0.757 ...
##  $ texture_se             : num  0.905 0.734 0.787 1.156 0.781 ...
##  $ perimeter_se           : num  8.59 3.4 4.58 3.44 5.44 ...
##  $ area_se                : num  153.4 74.1 94 27.2 94.4 ...
##  $ smoothness_se          : num  0.0064 0.00522 0.00615 0.00911 0.01149 ...
##  $ compactness_se         : num  0.049 0.0131 0.0401 0.0746 0.0246 ...
##  $ concavity_se           : num  0.0537 0.0186 0.0383 0.0566 0.0569 ...
##  $ concave.points_se      : num  0.0159 0.0134 0.0206 0.0187 0.0188 ...
##  $ symmetry_se            : num  0.03 0.0139 0.0225 0.0596 0.0176 ...
##  $ fractal_dimension_se   : num  0.00619 0.00353 0.00457 0.00921 0.00511 ...
##  $ radius_worst           : num  25.4 25 23.6 14.9 22.5 ...
##  $ texture_worst          : num  17.3 23.4 25.5 26.5 16.7 ...
##  $ perimeter_worst        : num  184.6 158.8 152.5 98.9 152.2 ...
##  $ area_worst             : num  2019 1956 1709 568 1575 ...
##  $ smoothness_worst       : num  0.162 0.124 0.144 0.21 0.137 ...
##  $ compactness_worst      : num  0.666 0.187 0.424 0.866 0.205 ...
##  $ concavity_worst        : num  0.712 0.242 0.45 0.687 0.4 ...
##  $ concave.points_worst   : num  0.265 0.186 0.243 0.258 0.163 ...
##  $ symmetry_worst         : num  0.46 0.275 0.361 0.664 0.236 ...
##  $ fractal_dimension_worst: num  0.1189 0.089 0.0876 0.173 0.0768 ...
##  $ X                      : logi  NA NA NA NA NA NA ...
```

```
#summary of data
summary(data)
```

```
##        id              diagnosis          radius_mean      texture_mean
##  Min.   :     8670   Length:569         Min.   : 6.981   Min.   : 9.71
##  1st Qu.:   869218   Class :character   1st Qu.:11.700   1st Qu.:16.17
```

```
##  Median :   906024   Mode  :character   Median :13.370   Median :18.84
##  Mean   : 30371831                      Mean   :14.127   Mean   :19.29
##  3rd Qu.:  8813129                      3rd Qu.:15.780   3rd Qu.:21.80
##  Max.   :911320502                      Max.   :28.110   Max.   :39.28
##  perimeter_mean     area_mean      smoothness_mean   compactness_mean
##  Min.   : 43.79   Min.   : 143.5   Min.   :0.05263   Min.   :0.01938
##  1st Qu.: 75.17   1st Qu.: 420.3   1st Qu.:0.08637   1st Qu.:0.06492
##  Median : 86.24   Median : 551.1   Median :0.09587   Median :0.09263
##  Mean   : 91.97   Mean   : 654.9   Mean   :0.09636   Mean   :0.10434
##  3rd Qu.:104.10   3rd Qu.: 782.7   3rd Qu.:0.10530   3rd Qu.:0.13040
##  Max.   :188.50   Max.   :2501.0   Max.   :0.16340   Max.   :0.34540
##  concavity_mean    concave.points_mean symmetry_mean
##  Min.   :0.00000   Min.   :0.00000     Min.   :0.1060
##  1st Qu.:0.02956   1st Qu.:0.02031     1st Qu.:0.1619
##  Median :0.06154   Median :0.03350     Median :0.1792
##  Mean   :0.08880   Mean   :0.04892     Mean   :0.1812
##  3rd Qu.:0.13070   3rd Qu.:0.07400     3rd Qu.:0.1957
##  Max.   :0.42680   Max.   :0.20120     Max.   :0.3040
##  fractal_dimension_mean   radius_se        texture_se      perimeter_se
##  Min.   :0.04996        Min.   :0.1115   Min.   :0.3602   Min.   : 0.757
##  1st Qu.:0.05770        1st Qu.:0.2324   1st Qu.:0.8339   1st Qu.: 1.606
##  Median :0.06154        Median :0.3242   Median :1.1080   Median : 2.287
##  Mean   :0.06280        Mean   :0.4052   Mean   :1.2169   Mean   : 2.866
##  3rd Qu.:0.06612        3rd Qu.:0.4789   3rd Qu.:1.4740   3rd Qu.: 3.357
##  Max.   :0.09744        Max.   :2.8730   Max.   :4.8850   Max.   :21.980
##     area_se        smoothness_se      compactness_se      concavity_se
##  Min.   :  6.802   Min.   :0.001713   Min.   :0.002252   Min.   :0.00000
##  1st Qu.: 17.850   1st Qu.:0.005169   1st Qu.:0.013080   1st Qu.:0.01509
##  Median : 24.530   Median :0.006380   Median :0.020450   Median :0.02589
##  Mean   : 40.337   Mean   :0.007041   Mean   :0.025478   Mean   :0.03189
##  3rd Qu.: 45.190   3rd Qu.:0.008146   3rd Qu.:0.032450   3rd Qu.:0.04205
##  Max.   :542.200   Max.   :0.031130   Max.   :0.135400   Max.   :0.39600
##  concave.points_se   symmetry_se       fractal_dimension_se
##  Min.   :0.000000   Min.   :0.007882   Min.   :0.0008948
##  1st Qu.:0.007638   1st Qu.:0.015160   1st Qu.:0.0022480
##  Median :0.010930   Median :0.018730   Median :0.0031870
##  Mean   :0.011796   Mean   :0.020542   Mean   :0.0037949
##  3rd Qu.:0.014710   3rd Qu.:0.023480   3rd Qu.:0.0045580
##  Max.   :0.052790   Max.   :0.078950   Max.   :0.0298400
##   radius_worst   texture_worst   perimeter_worst    area_worst
##  Min.   : 7.93   Min.   :12.02   Min.   : 50.41   Min.   : 185.2
##  1st Qu.:13.01   1st Qu.:21.08   1st Qu.: 84.11   1st Qu.: 515.3
##  Median :14.97   Median :25.41   Median : 97.66   Median : 686.5
##  Mean   :16.27   Mean   :25.68   Mean   :107.26   Mean   : 880.6
##  3rd Qu.:18.79   3rd Qu.:29.72   3rd Qu.:125.40   3rd Qu.:1084.0
##  Max.   :36.04   Max.   :49.54   Max.   :251.20   Max.   :4254.0
##  smoothness_worst  compactness_worst concavity_worst  concave.points_worst
##  Min.   :0.07117   Min.   :0.02729   Min.   :0.0000   Min.   :0.00000
##  1st Qu.:0.11660   1st Qu.:0.14720   1st Qu.:0.1145   1st Qu.:0.06493
##  Median :0.13130   Median :0.21190   Median :0.2267   Median :0.09993
##  Mean   :0.13237   Mean   :0.25427   Mean   :0.2722   Mean   :0.11461
##  3rd Qu.:0.14600   3rd Qu.:0.33910   3rd Qu.:0.3829   3rd Qu.:0.16140
##  Max.   :0.22260   Max.   :1.05800   Max.   :1.2520   Max.   :0.29100
##  symmetry_worst   fractal_dimension_worst    X
```

```
##  Min.   :0.1565   Min.   :0.05504         Mode:logical
##  1st Qu.:0.2504   1st Qu.:0.07146         NA's:569
##  Median :0.2822   Median :0.08004
##  Mean   :0.2901   Mean   :0.08395
##  3rd Qu.:0.3179   3rd Qu.:0.09208
##  Max.   :0.6638   Max.   :0.20750
```

```
#Checking the first few rows
head(data)
```

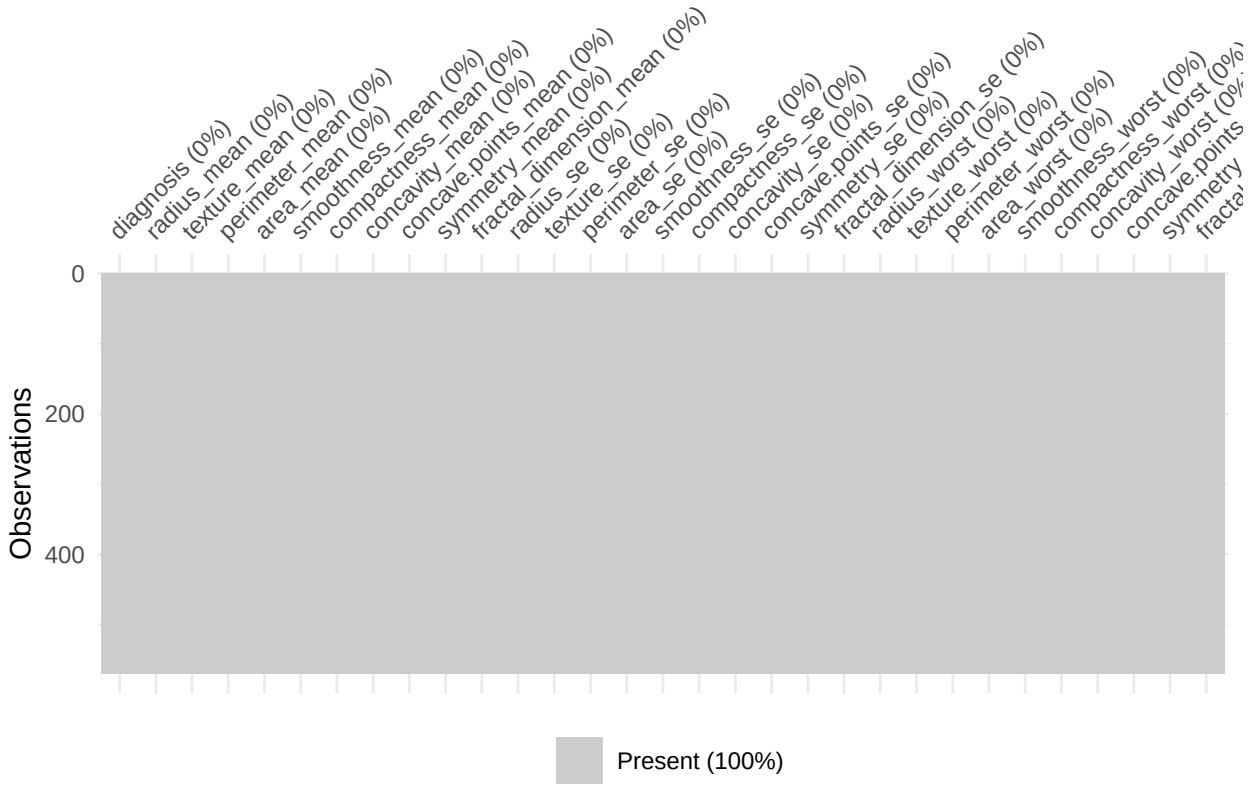
```
##         id diagnosis radius_mean texture_mean perimeter_mean area_mean
## 1   842302         M       17.99        10.38         122.80    1001.0
## 2   842517         M       20.57        17.77         132.90    1326.0
## 3 84300903         M       19.69        21.25         130.00    1203.0
## 4 84348301         M       11.42        20.38          77.58     386.1
## 5 84358402         M       20.29        14.34         135.10    1297.0
## 6   843786         M       12.45        15.70          82.57     477.1
##   smoothness_mean compactness_mean concavity_mean concave.points_mean
## 1         0.11840          0.27760         0.3001             0.14710
## 2         0.08474          0.07864         0.0869             0.07017
## 3         0.10960          0.15990         0.1974             0.12790
## 4         0.14250          0.28390         0.2414             0.10520
## 5         0.10030          0.13280         0.1980             0.10430
## 6         0.12780          0.17000         0.1578             0.08089
##   symmetry_mean fractal_dimension_mean radius_se texture_se perimeter_se
## 1        0.2419                0.07871    1.0950     0.9053        8.589
## 2        0.1812                0.05667    0.5435     0.7339        3.398
## 3        0.2069                0.05999    0.7456     0.7869        4.585
## 4        0.2597                0.09744    0.4956     1.1560        3.445
## 5        0.1809                0.05883    0.7572     0.7813        5.438
## 6        0.2087                0.07613    0.3345     0.8902        2.217
##   area_se smoothness_se compactness_se concavity_se concave.points_se
## 1  153.40      0.006399        0.04904      0.05373           0.01587
## 2   74.08      0.005225        0.01308      0.01860           0.01340
## 3   94.03      0.006150        0.04006      0.03832           0.02058
## 4   27.23      0.009110        0.07458      0.05661           0.01867
## 5   94.44      0.011490        0.02461      0.05688           0.01885
## 6   27.19      0.007510        0.03345      0.03672           0.01137
##   symmetry_se fractal_dimension_se radius_worst texture_worst
## 1     0.03003             0.006193        25.38         17.33
## 2     0.01389             0.003532        24.99         23.41
## 3     0.02250             0.004571        23.57         25.53
## 4     0.05963             0.009208        14.91         26.50
## 5     0.01756             0.005115        22.54         16.67
## 6     0.02165             0.005082        15.47         23.75
##   perimeter_worst area_worst smoothness_worst compactness_worst
## 1          184.60     2019.0           0.1622            0.6656
## 2          158.80     1956.0           0.1238            0.1866
## 3          152.50     1709.0           0.1444            0.4245
## 4           98.87      567.7           0.2098            0.8663
## 5          152.20     1575.0           0.1374            0.2050
## 6          103.40      741.6           0.1791            0.5249
##   concavity_worst concave.points_worst symmetry_worst
## 1          0.7119               0.2654         0.4601
## 2          0.2416               0.1860         0.2750
```

```
## 3          0.4504               0.2430         0.3613
## 4          0.6869               0.2575         0.6638
## 5          0.4000               0.1625         0.2364
## 6          0.5355               0.1741         0.3985
##   fractal_dimension_worst  X
## 1                 0.11890 NA
## 2                 0.08902 NA
## 3                 0.08758 NA
## 4                 0.17300 NA
## 5                 0.07678 NA
## 6                 0.12440 NA
```

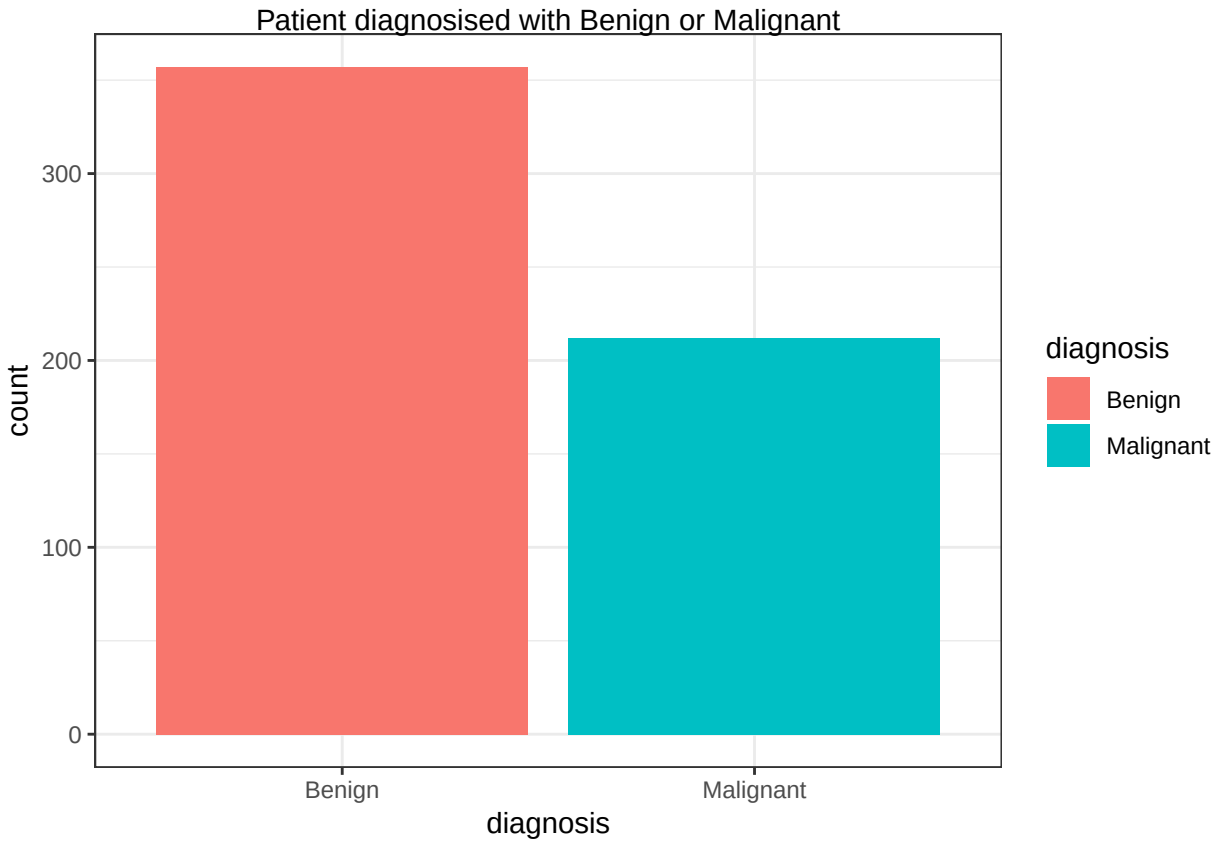
```r
# id and the last column are not required, so we will drop it from our data set
data <- data[,-c(1,33)]

#Converting the target variable "diagnosis into factor from string
data$diagnosis <- factor(data$diagnosis, levels = c("B", "M"),
                         labels = c("Benign", "Malignant"))
#Now check for missing data in the dataset column wise
sapply(data, function(x) sum(is.na(x)))   # This gives zero missing values in every row
```

```
##              diagnosis            radius_mean           texture_mean
##                      0                      0                      0
##         perimeter_mean              area_mean        smoothness_mean
##                      0                      0                      0
##        compactness_mean         concavity_mean     concave.points_mean
##                      0                      0                      0
##          symmetry_mean  fractal_dimension_mean              radius_se
##                      0                      0                      0
##             texture_se           perimeter_se                area_se
##                      0                      0                      0
##          smoothness_se         compactness_se           concavity_se
##                      0                      0                      0
##       concave.points_se            symmetry_se    fractal_dimension_se
##                      0                      0                      0
##            radius_worst          texture_worst        perimeter_worst
##                      0                      0                      0
##              area_worst       smoothness_worst       compactness_worst
##                      0                      0                      0
##         concavity_worst    concave.points_worst          symmetry_worst
##                      0                      0                      0
## fractal_dimension_worst
##                      0
```

```r
# Visulisation of missing data
library(naniar)
vis_miss(data)  # gives the data present or absent in percentage
```




```
#Looking at the number of patients with Malignant and Benign Tumors:
library(ggplot2)
ggplot(data, aes(x= diagnosis, fill=diagnosis)) +geom_bar()+
    ggtitle("Patient diagnosised with Benign or Malignant")+
    theme_bw()+theme(plot.title = element_text(hjust=0.5,size=20,face='bold')+
                         geom_text(aes(label=..count..),stat="count",position = position_stack()))
```

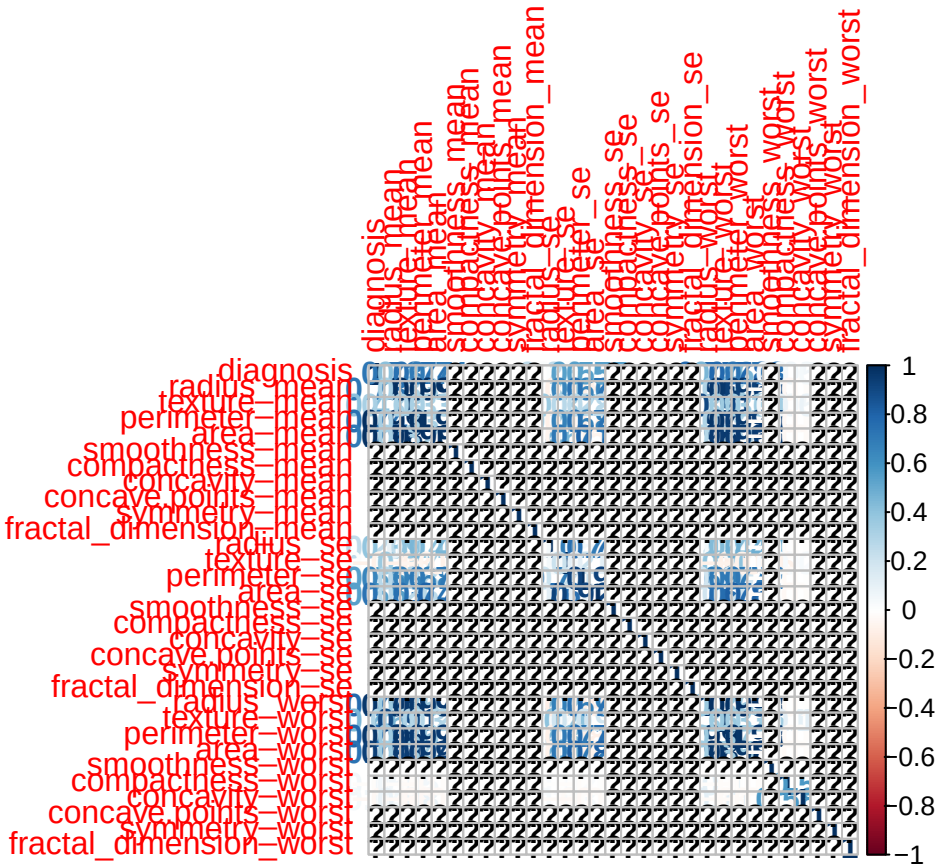



```
#This shows 65% patients had Benign cancer and rest have Malignent

# Now we will see how these mean features are correlated with diagnosis
set.seed(1)
df <- data.frame(data)
df[] <- lapply(df, as.integer)

library(corrplot)
```

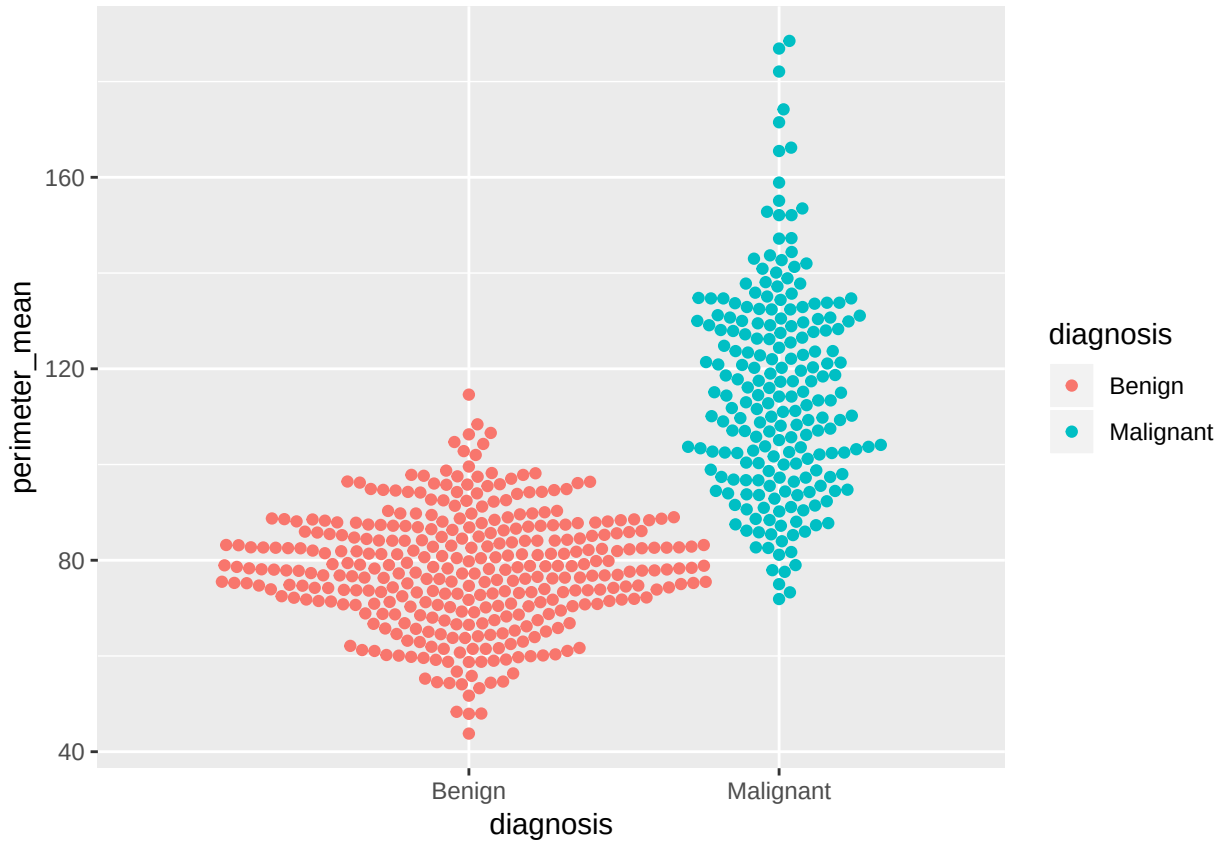
```
## corrplot 0.84 loaded
```

```
corrplot(cor(df, method = 'pearson'), method = 'number')
```

```
## Warning in cor(df, method = "pearson"): the standard deviation is zero
```




```
#1.radius_mean, perimeter_mean, area_mean, compactness_mean, concavity_mean, concave points_mean show h
#2.The other variables do not really show high impact over diagnoses.
library(ggbeeswarm)
ggplot(mapping=aes(diagnosis, perimeter_mean,color=diagnosis), data) +geom_beeswarm(dodge.width=.8,cex=
```




```
library(caret)
```

```
## Loading required package: lattice
```

```
set.seed(100)
index <- createDataPartition(data$diagnosis,p=0.8, list = F, times = 1)
train <- data[index,]
test <- data[-index,]
```

```
#From the correlation plot we got the radius_mean, texture_mean, perimeter_mean, area_mean, radius_se, p
#radius_worst, texture_worst, perimeter_worst, area_worst, texture_worst

reg_model <- glm(diagnosis~ radius_mean+texture_mean+perimeter_mean+area_mean+radius_se+perimeter_se+are
              texture_worst+perimeter_worst+area_worst , train, family = binomial)
```

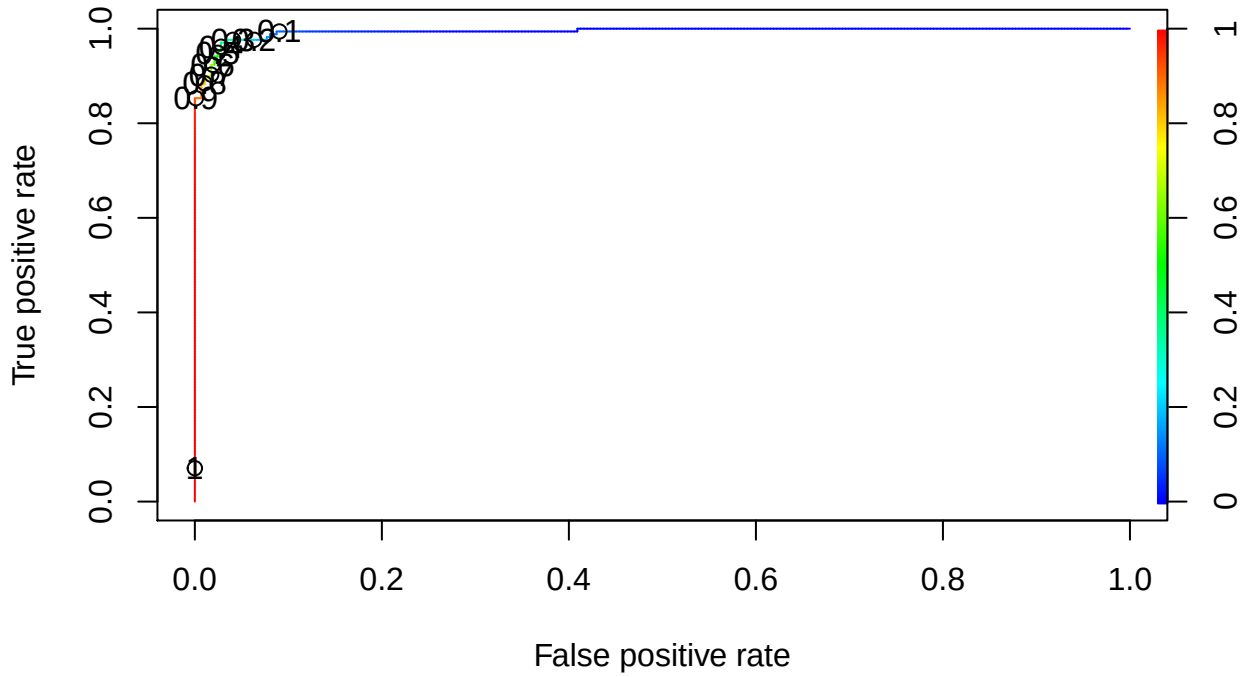
```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(reg_model)
```

```
##
## Call:
## glm(formula = diagnosis ~ radius_mean + texture_mean + perimeter_mean +
##     area_mean + radius_se + perimeter_se + area_se + radius_worst +
##     texture_worst + perimeter_worst + area_worst, family = binomial,
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1069  -0.0910  -0.0156   0.0006   3.3568
```

```
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -11.266364  16.276640  -0.692  0.48882
## radius_mean     -11.621604   4.048610  -2.871  0.00410 **
## texture_mean     -0.126986   0.175430  -0.724  0.46916
## perimeter_mean    1.116850   0.375484   2.974  0.00294 **
## area_mean         0.031321   0.032242   0.971  0.33133
## radius_se         4.191478  18.854534   0.222  0.82408
## perimeter_se     -0.889017   1.251515  -0.710  0.47749
## area_se           0.054913   0.199349   0.275  0.78296
## radius_worst      2.461591   2.457431   1.002  0.31649
## texture_worst     0.381838   0.137507   2.777  0.00549 **
## perimeter_worst   0.072587   0.173027   0.420  0.67484
## area_worst       -0.003911   0.024262  -0.161  0.87195
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 602.315  on 455  degrees of freedom
## Residual deviance:  81.812  on 444  degrees of freedom
## AIC: 105.81
##
## Number of Fisher Scoring iterations: 10
```

```r
#Model performance evaluation

library(ROCR)

pred <-predict(reg_model, train, type='response')

ROCRPred <- prediction(pred, train$diagnosis)
ROCRPref <- performance(ROCRPred, "tpr", "fpr")

plot(ROCRPref, colorize=T, print.cutoffs.at=seq(0.1, by=0.1))
```




```
# Make prediction on the test data set
pred_lm <- predict(reg_model, test, type = "response")


tab <-table(Actualvalue=test$diagnosis, Predictedvalue=pred_lm>0.3)
tab
```

```
##             Predictedvalue
## Actualvalue FALSE TRUE
##   Benign       68    3
##   Malignant     1   41
```

```
#accuracy of the model
sum(diag(tab))/sum(tab)
```

```
## [1] 0.9646018
```

```
1-sum(diag(tab))/sum(tab)
```

```
## [1] 0.03539823
```

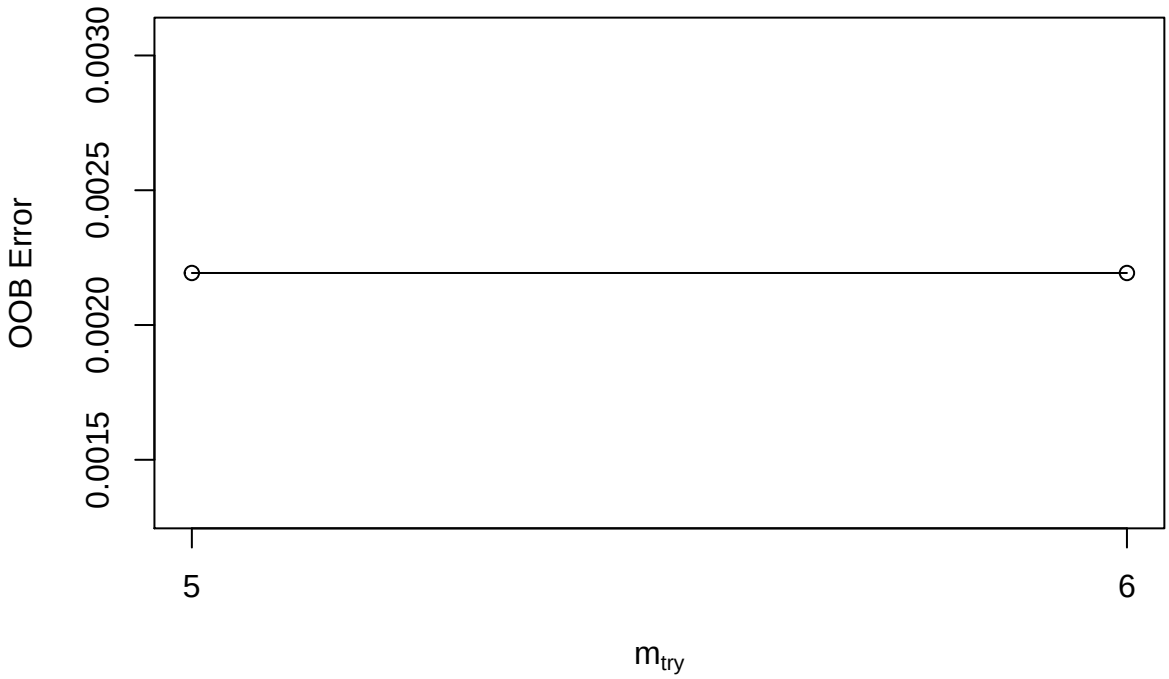
```
# When our threshold is 0.3 we get better result i.e. true negative value goes down and the model accur
```

```
# before creating random forest model , first we find out the optimised "mtry" value
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:ggplot2':
##
##     margin
```

```
bestmtry <-tuneRF(train, train$diagnosis, stepFactor = 1.2, improve = 0.01, trace = T, plot = T)
```

```
## mtry = 5  OOB error = 0.22%
## Searching left ...
## Searching right ...
## mtry = 6     OOB error = 0.22%
## 0 0.01
```




```
rf_model <- randomForest(diagnosis~., data = data)
rf_model
```

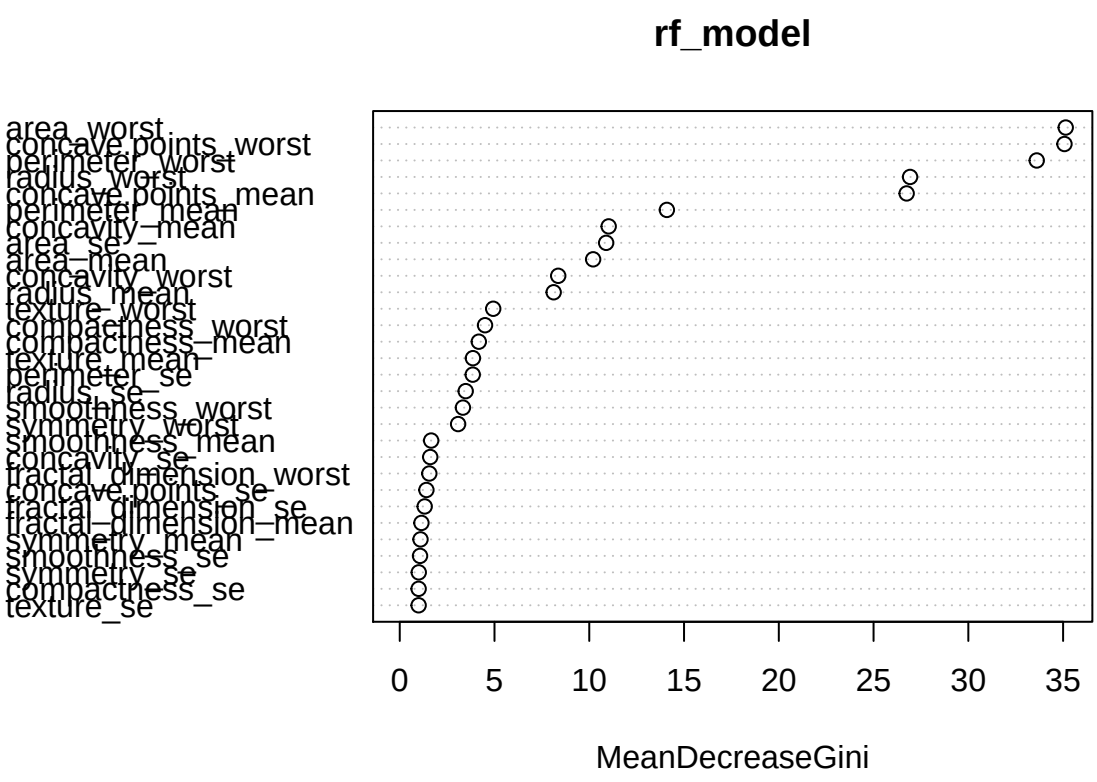
```
##
## Call:
##  randomForest(formula = diagnosis ~ ., data = data)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 5
##
##         OOB estimate of  error rate: 4.04%
## Confusion matrix:
##           Benign Malignant class.error
## Benign       348         9  0.02521008
## Malignant     14       198  0.06603774
```

```
rf_model$importance
```

```
##                         MeanDecreaseGini
## radius_mean                    8.1209979
## texture_mean                   3.8618722
## perimeter_mean                14.0940064
## area_mean                     10.2034283
## smoothness_mean                1.6590460
## compactness_mean               4.1717203
## concavity_mean                11.0239790
```

```
## concave.points_mean           26.7456308
## symmetry_mean                  1.0957337
## fractal_dimension_mean         1.1448434
## radius_se                      3.4800284
## texture_se                     0.9961860
## perimeter_se                   3.8518841
## area_se                       10.8922702
## smoothness_se                  1.0707014
## compactness_se                 0.9972459
## concavity_se                   1.6093025
## concave.points_se              1.4069331
## symmetry_se                    1.0036694
## fractal_dimension_se           1.3164444
## radius_worst                  26.9266566
## texture_worst                  4.9324532
## perimeter_worst               33.6089491
## area_worst                    35.1383195
## smoothness_worst               3.3343276
## compactness_worst              4.5005004
## concavity_worst                8.3617521
## concave.points_worst          35.0727135
## symmetry_worst                 3.0803860
## fractal_dimension_worst        1.5615687
```

```
varImpPlot(rf_model)
```




```
rf_pred <- predict(rf_model, newdata = test, type = "class")
rf_pred
```

```
##         2         9        10        20        23        24        25
## Malignant Malignant Malignant    Benign Malignant Malignant Malignant
```

```
##        32        34        42        43        52        54        57
## Malignant Malignant Malignant Malignant    Benign Malignant Malignant
##        59        60        62        66        67        76        77
##    Benign    Benign    Benign Malignant    Benign Malignant    Benign
##        81       112       113       114       123       126       139
##    Benign    Benign    Benign    Benign Malignant    Benign Malignant
##       142       149       150       160       166       172       177
## Malignant    Benign    Benign    Benign    Benign Malignant    Benign
##       180       184       188       199       202       203       205
##    Benign    Benign    Benign Malignant Malignant Malignant    Benign
##       207       215       218       219       220       228       231
##    Benign Malignant    Benign Malignant Malignant    Benign Malignant
##       235       238       239       240       247       254       262
##    Benign Malignant    Benign Malignant    Benign Malignant Malignant
##       268       283       288       293       296       298       311
##    Benign Malignant    Benign    Benign    Benign Malignant    Benign
##       318       339       340       342       344       351       357
## Malignant    Benign Malignant    Benign Malignant    Benign    Benign
##       359       365       368       371       372       381       383
##    Benign    Benign    Benign Malignant    Benign    Benign    Benign
##       400       402       409       411       412       415       423
##    Benign    Benign Malignant    Benign    Benign Malignant    Benign
##       425       426       445       448       451       471       474
##    Benign    Benign Malignant    Benign    Benign    Benign    Benign
##       483       491       492       498       500       502       509
##    Benign    Benign    Benign    Benign Malignant Malignant    Benign
##       511       515       521       523       524       526       528
##    Benign Malignant    Benign    Benign    Benign    Benign    Benign
##       530       532       539       545       551       552       558
##    Benign    Benign    Benign    Benign    Benign    Benign    Benign
##       565
## Malignant
## Levels: Benign Malignant
```

```
library(caret)

# Now we will create confusion matrix which will give clear picture of predicted variable and actual va
confusionMatrix(table(rf_pred, test$diagnosis))
```

```
## Confusion Matrix and Statistics
##
##
## rf_pred     Benign Malignant
##   Benign        71         0
##   Malignant      0        42
##
##                Accuracy : 1
##                  95% CI : (0.9679, 1)
##     No Information Rate : 0.6283
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 1
##  Mcnemar's Test P-Value : NA
##
##             Sensitivity : 1.0000
```

```
##             Specificity : 1.0000
##          Pos Pred Value : 1.0000
##          Neg Pred Value : 1.0000
##              Prevalence : 0.6283
##          Detection Rate : 0.6283
##    Detection Prevalence : 0.6283
##       Balanced Accuracy : 1.0000
##
##        'Positive' Class : Benign
##
```

```
# This give accuracy of the model is 100%
```