WisconsinBreastCancer

Abhinav Ram Bhatta, Prajwal Prashanth, Anviksha Gupta

2022-09-28

a. Major predictors of diagnosis are concave_points, concavity_points_mean, area mean, radius_mean, parameter_mean, area, radius, and perimeter. There are no missing values in the data set. There are outliers. The outliers have been eliminated using Z-score method.

b. 5 leaves in the Decision Tree

c. Major predictors suggested by our tree are: Perimeter, Concave_points_mean, texture

d. Firstly, eliminated the outliers using Z-score and obtained 398 observations. Then to enhance the accuracy cp value has been changed. When cp = 0.04, accuracy of test was 87.5%, when cp was changed to 0.01, Accuracy for Train = 94.49%

e. Accuracy for Train: 94.49% Accuracy for Test: 94.64%

f. Initially, we clean the data for outliers by using the z-score method. Then we combine the trainx and trainy variable to obtain a matrix with the diagnosis. For this new combined variable we applied the decision tree classifiaction. From this, we applied the testx and testy data to the created model and obtain the result. To support the same we have plotted a bivariate analysis and the importance of variable bar graph.

g. From the importance of variable graph, we get the most important variables from which the tree can be affected. Here we used, perimeter_lv, radius_lv, area_lv and perimeter.

```
library("rpart")
library("ggplot2")
library("tidyverse")
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ─
─
## ✔ tibble  3.1.8      ✔ dplyr   1.0.10
## ✔ tidyr   1.2.0      ✔ stringr 1.4.1
## ✔ readr   2.1.2      ✔ forcats 0.5.2
## ✔ purrr   0.3.4
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ─
─
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library("psych")
```

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

```
library("corrplot")
```

```
## corrplot 0.92 loaded
```

```
library("RColorBrewer")

#Load Dataset

trainx=read.csv("/Users/abhinavram/Documents/IDS572 Data Mining/Assignment 1/train
X.csv")
trainy=read.csv("/Users/abhinavram/Documents/IDS572 Data Mining/Assignment 1/train
Y.csv")
testx=read.csv("/Users/abhinavram/Documents/IDS572 Data Mining/Assignment 1/testX.
csv")
testy=read.csv("/Users/abhinavram/Documents/IDS572 Data Mining/Assignment 1/testY.
csv")

#Column Names

colnames(trainx)<-c('radius','texture','perimeter','area','smoothness','compactnes
s','concavity','no_of_concave_contour', 'symmetry', 'fractal_dim',
                 'radius_sd','texture_sd','perimeter_sd','area_sd','smoothness_sd'
,'compactness_sd','concavity_sd','no_of_concave_contour_sd', 'symmetry_sd', 'fract
al_dim_sd',
                 'radius_lv','texture_lv','perimeter_lv','area_lv','smoothness_lv'
,'compactness_lv','concavity_lv','no_of_concave_contour_lv', 'symmetry_lv', 'fract
al_dim_lv')
colnames(testx)<-c('radius','texture','perimeter','area','smoothness','compactness
','concavity','no_of_concave_contour', 'symmetry', 'fractal_dim',
                 'radius_sd','texture_sd','perimeter_sd','area_sd','smoothness_sd'
,'compactness_sd','concavity_sd','no_of_concave_contour_sd', 'symmetry_sd', 'fract
al_dim_sd',
                 'radius_lv','texture_lv','perimeter_lv','area_lv','smoothness_lv'
,'compactness_lv','concavity_lv','no_of_concave_contour_lv', 'symmetry_lv', 'fract
al_dim_lv')
```

Summary of dataset

```
summary(trainx)
```

```
##      radius          texture        perimeter          area
##  Min.   : 6.981   Min.   : 9.71   Min.   : 43.79   Min.   : 143.5
##  1st Qu.:11.613   1st Qu.:16.21   1st Qu.: 74.69   1st Qu.: 412.5
##  Median :13.280   Median :18.82   Median : 86.04   Median : 545.6  
##  Mean   :14.107   Mean   :19.38   Mean   : 91.87   Mean   : 655.0  
##  3rd Qu.:15.832   3rd Qu.:21.91   3rd Qu.:103.78   3rd Qu.: 787.0  
##  Max.   :28.110   Max.   :39.28   Max.   :188.50   Max.   :2501.0  
##    smoothness       compactness        concavity       no_of_concave_contour
##  Min.   :0.05263   Min.   :0.01938   Min.   :0.00000   Min.   :0.00000      
##  1st Qu.:0.08609   1st Qu.:0.06609   1st Qu.:0.03038   1st Qu.:0.01979      
##  Median :0.09572   Median :0.09449   Median :0.06406   Median :0.03387      
##  Mean   :0.09620   Mean   :0.10511   Mean   :0.09011   Mean   :0.04895      
##  3rd Qu.:0.10460   3rd Qu.:0.13057   3rd Qu.:0.13205   3rd Qu.:0.07403      
##  Max.   :0.16340   Max.   :0.34540   Max.   :0.42680   Max.   :0.20120      
##     symmetry       fractal_dim        radius_sd        texture_sd    
##  Min.   :0.1060   Min.   :0.04996   Min.   :0.1115   Min.   :0.3621  
##  1st Qu.:0.1623   1st Qu.:0.05797   1st Qu.:0.2319   1st Qu.:0.8280  
##  Median :0.1799   Median :0.06177   Median :0.3175   Median :1.1080  
##  Mean   :0.1808   Mean   :0.06299   Mean   :0.4022   Mean   :1.2278  
##  3rd Qu.:0.1957   3rd Qu.:0.06638   3rd Qu.:0.4702   3rd Qu.:1.4797  
##  Max.   :0.2906   Max.   :0.09744   Max.   :2.8730   Max.   :4.8850  
##   perimeter_sd       area_sd        smoothness_sd      compactness_sd    
##  Min.   : 0.757   Min.   :  6.802   Min.   :0.001713   Min.   :0.002252  
##  1st Qu.: 1.643   1st Qu.: 17.670   1st Qu.:0.005227   1st Qu.:0.013710  
##  Median : 2.280   Median : 23.930   Median :0.006457   Median :0.021145  
##  Mean   : 2.857   Mean   : 40.205   Mean   :0.007163   Mean   :0.026019  
##  3rd Qu.: 3.309   3rd Qu.: 44.947   3rd Qu.:0.008391   3rd Qu.:0.032888  
##  Max.   :21.980   Max.   :542.200   Max.   :0.031130   Max.   :0.135400  
##   concavity_sd     no_of_concave_contour_sd  symmetry_sd      
##  Min.   :0.00000   Min.   :0.000000         Min.   :0.007882  
##  1st Qu.:0.01569   1st Qu.:0.007735         1st Qu.:0.015220  
##  Median :0.02625   Median :0.011000         Median :0.018975  
##  Mean   :0.03285   Mean   :0.011880         Mean   :0.020695  
##  3rd Qu.:0.04273   3rd Qu.:0.014897         3rd Qu.:0.023768  
##  Max.   :0.39600   Max.   :0.052790         Max.   :0.078950  
##  fractal_dim_sd        radius_lv       texture_lv     perimeter_lv   
##  Min.   :0.0008948   Min.   : 7.93   Min.   :12.02   Min.   : 50.41  
##  1st Qu.:0.0022787   1st Qu.:12.97   1st Qu.:21.09   1st Qu.: 83.77  
##  Median :0.0032335   Median :14.88   Median :25.43   Median : 97.39  
##  Mean   :0.0038669   Mean   :16.22   Mean   :25.77   Mean   :106.95  
##  3rd Qu.:0.0045708   3rd Qu.:18.71   3rd Qu.:29.89   3rd Qu.:125.78  
##  Max.   :0.0298400   Max.   :36.04   Max.   :49.54   Max.   :251.20  
##     area_lv       smoothness_lv     compactness_lv     concavity_lv   
##  Min.   : 185.2   Min.   :0.07117   Min.   :0.02729   Min.   :0.0000  
##  1st Qu.: 509.8   1st Qu.:0.11728   1st Qu.:0.14860   1st Qu.:0.1202  
##  Median : 674.0   Median :0.13120   Median :0.21685   Median :0.2298  
##  Mean   : 877.9   Mean   :0.13268   Mean   :0.25810   Mean   :0.2761  
##  3rd Qu.:1063.5   3rd Qu.:0.14625   3rd Qu.:0.34190   3rd Qu.:0.3886  
##  Max.   :4254.0   Max.   :0.22260   Max.   :1.05800   Max.   :1.2520  
##  no_of_concave_contour_lv  symmetry_lv     fractal_dim_lv   
##  Min.   :0.00000          Min.   :0.1565   Min.   :0.05504  
##  1st Qu.:0.06321          1st Qu.:0.2510   1st Qu.:0.07222  
##  Median :0.10160          Median :0.2826   Median :0.08042  
##  Mean   :0.11484          Mean   :0.2904   Mean   :0.08456  
##  3rd Qu.:0.16608          3rd Qu.:0.3178   3rd Qu.:0.09217  
##  Max.   :0.29030          Max.   :0.6638   Max.   :0.20750  
```

Checking for missing values

```
sapply(trainx, function(x) sum(is.na(x)))
```

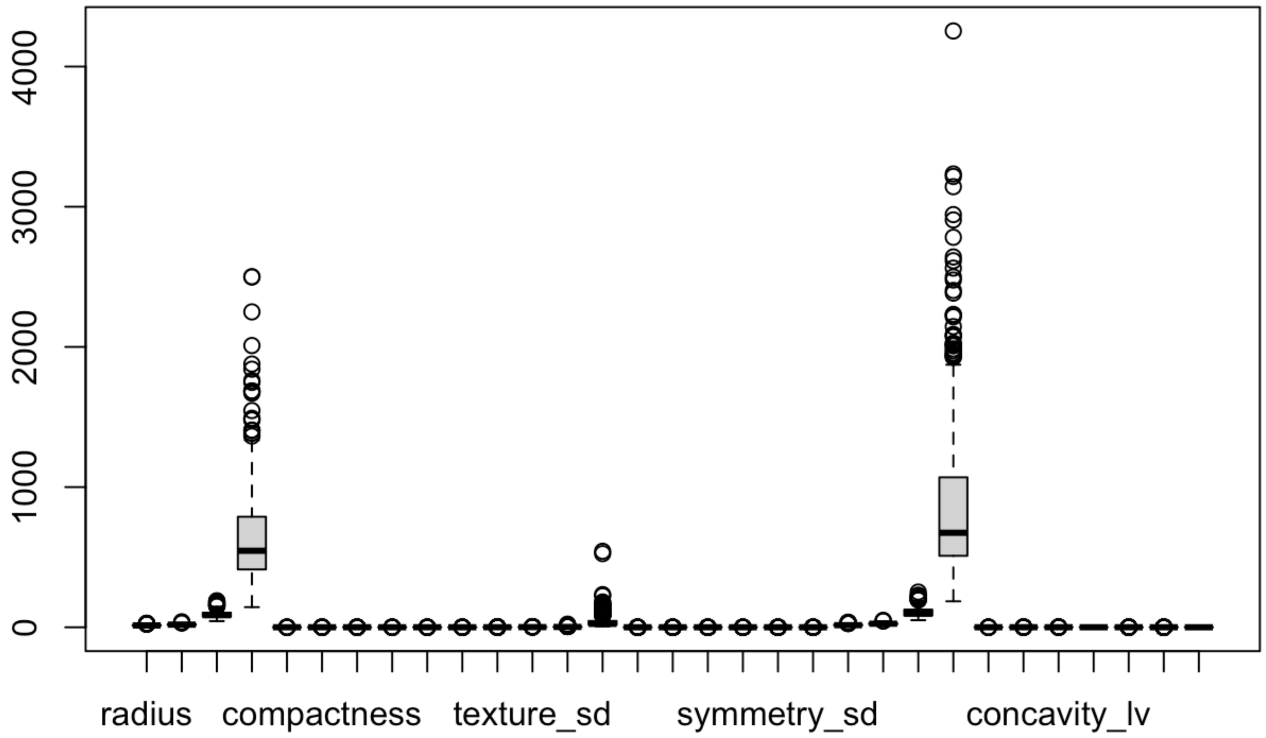
```
##                   radius                  texture                perimeter 
##                        0                        0                        0 
##                     area               smoothness              compactness 
##                        0                        0                        0 
##                concavity    no_of_concave_contour                 symmetry 
##                        0                        0                        0 
##              fractal_dim                radius_sd               texture_sd 
##                        0                        0                        0 
##             perimeter_sd                  area_sd            smoothness_sd 
##                        0                        0                        0 
##           compactness_sd             concavity_sd no_of_concave_contour_sd 
##                        0                        0                        0 
##              symmetry_sd           fractal_dim_sd                radius_lv 
##                        0                        0                        0 
##               texture_lv             perimeter_lv                  area_lv 
##                        0                        0                        0 
##            smoothness_lv           compactness_lv             concavity_lv 
##                        0                        0                        0 
## no_of_concave_contour_lv              symmetry_lv           fractal_dim_lv 
##                        0                        0                        0
```

Combine trainx and trainy

```
train_x_y <- cbind(trainx, trainy)
colnames(train_x_y)<-c('radius','texture','perimeter','area','smoothness','compact
ness','concavity','no_of_concave_contour', 'symmetry', 'fractal_dim',
                    'radius_sd','texture_sd','perimeter_sd','area_sd','smoothness_
sd','compactness_sd','concavity_sd','no_of_concave_contour_sd', 'symmetry_sd', 'fr
actal_dim_sd',
                    'radius_lv','texture_lv','perimeter_lv','area_lv','smoothness_
lv','compactness_lv','concavity_lv','no_of_concave_contour_lv', 'symmetry_lv', 'fr
actal_dim_lv','diagnosis')
```

Box Plot for Outliers and cleaning the data

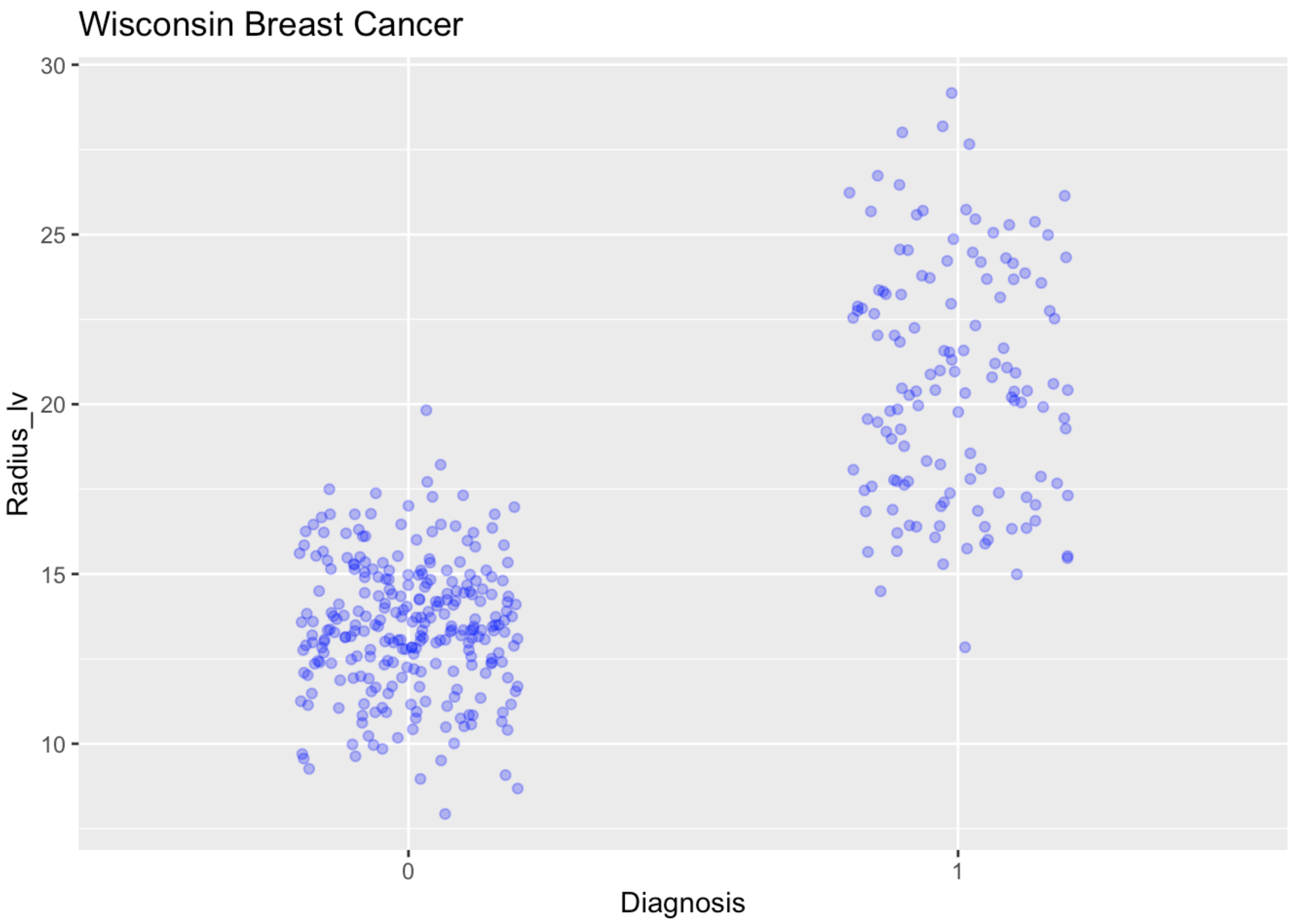
```
boxplot(train_x_y)
```



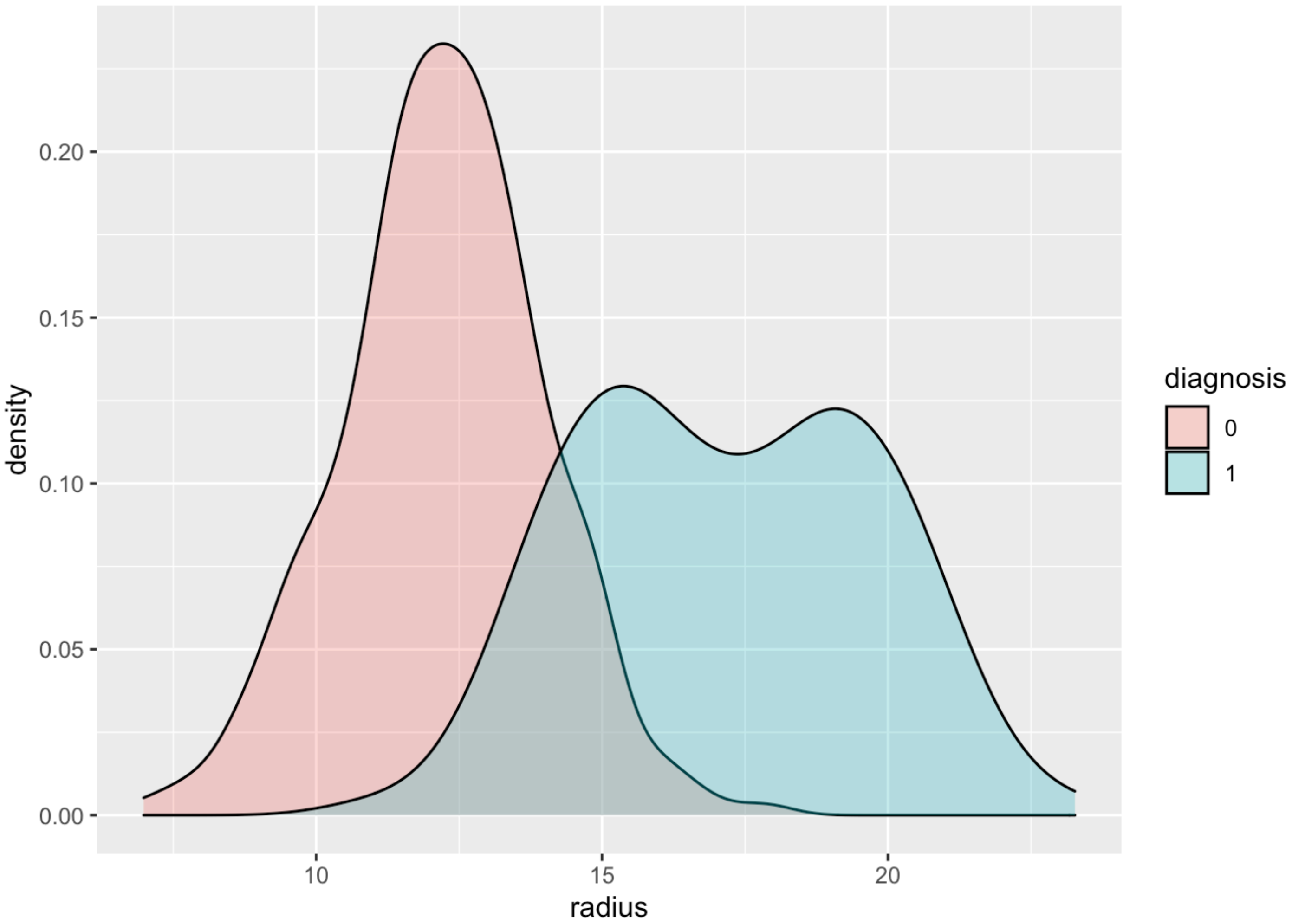
```
outliers<- as.data.frame(sapply(train_x_y, function(train_x_y) (abs(train_x_y- mea
n(train_x_y))/ sd(train_x_y))))

train_x_y_new <- train_x_y[!rowSums(outliers>3), ]

trainx_new = select(train_x_y_new, -diagnosis)
train_x_y_new$diagnosis = as.factor(train_x_y_new$diagnosis)
```

Bivariate Analysis

```
ggplot(data=train_x_y_new, aes(x=diagnosis, y=radius_lv, group = 1)) +

  geom_jitter(alpha=0.3,
              color =" blue",
              width = 0.2) +
  labs(title="Wisconsin Breast Cancer", x="Diagnosis", y="Radius_lv")
```

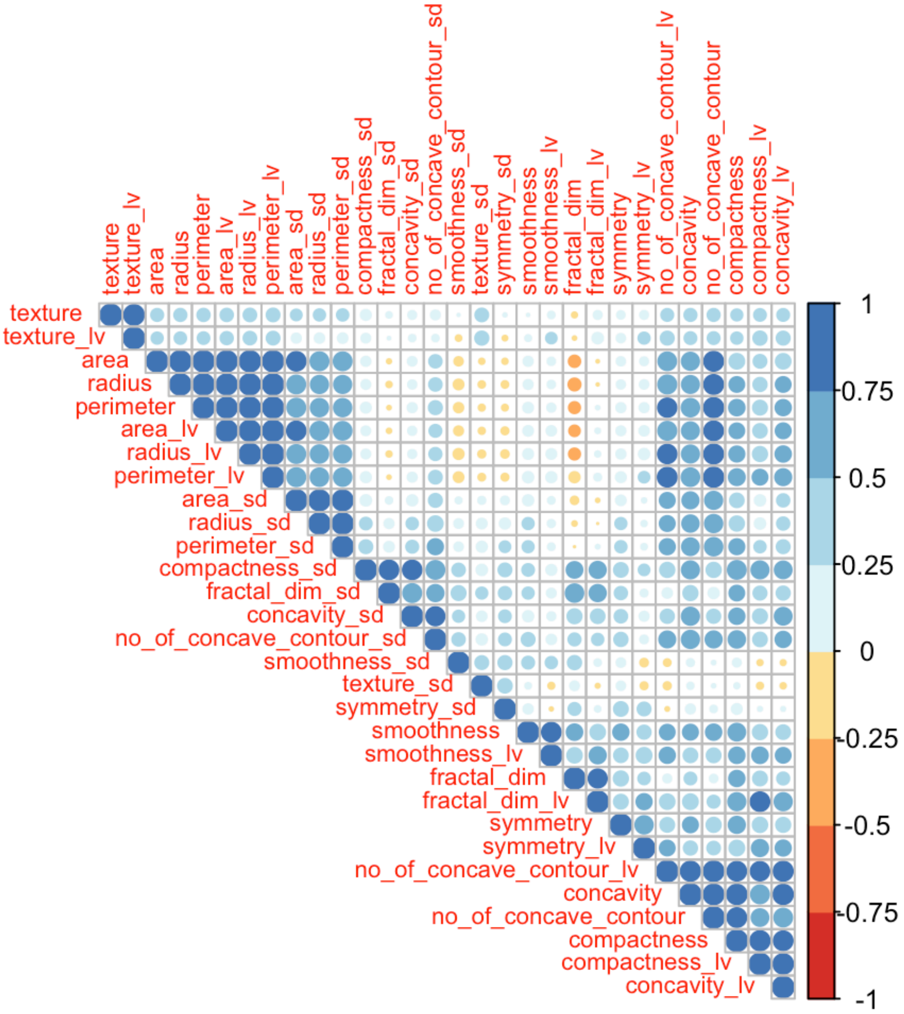


```
ggplot(data=train_x_y_new, aes(x=radius, fill=diagnosis)) +
  geom_density(alpha=.3)
```



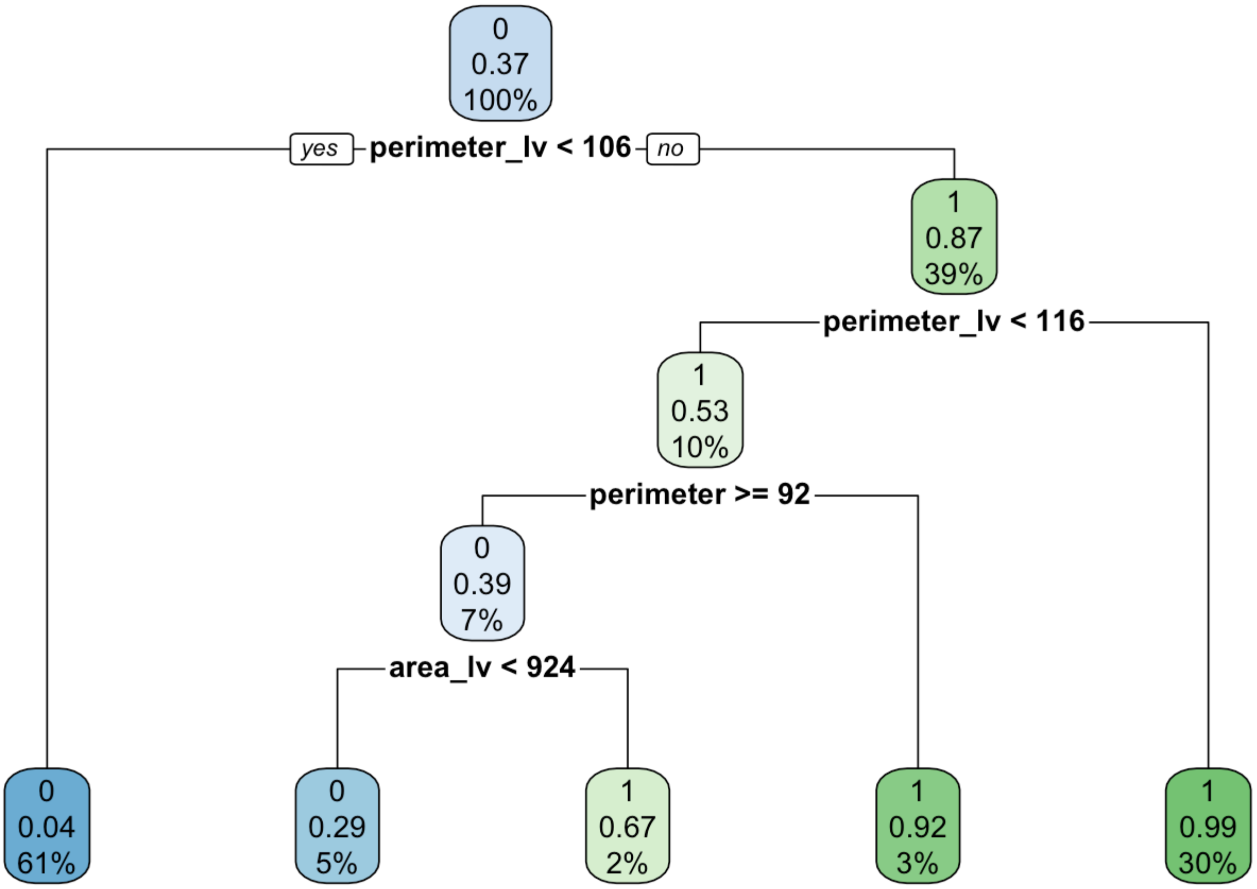
Correlation Plot

```
cor_graph <- cor(trainx)
corrplot(cor_graph, type="upper",order= "hclust", tl.cex = 0.7,col=brewer.pal(n=8,
name="RdYlBu"))
```

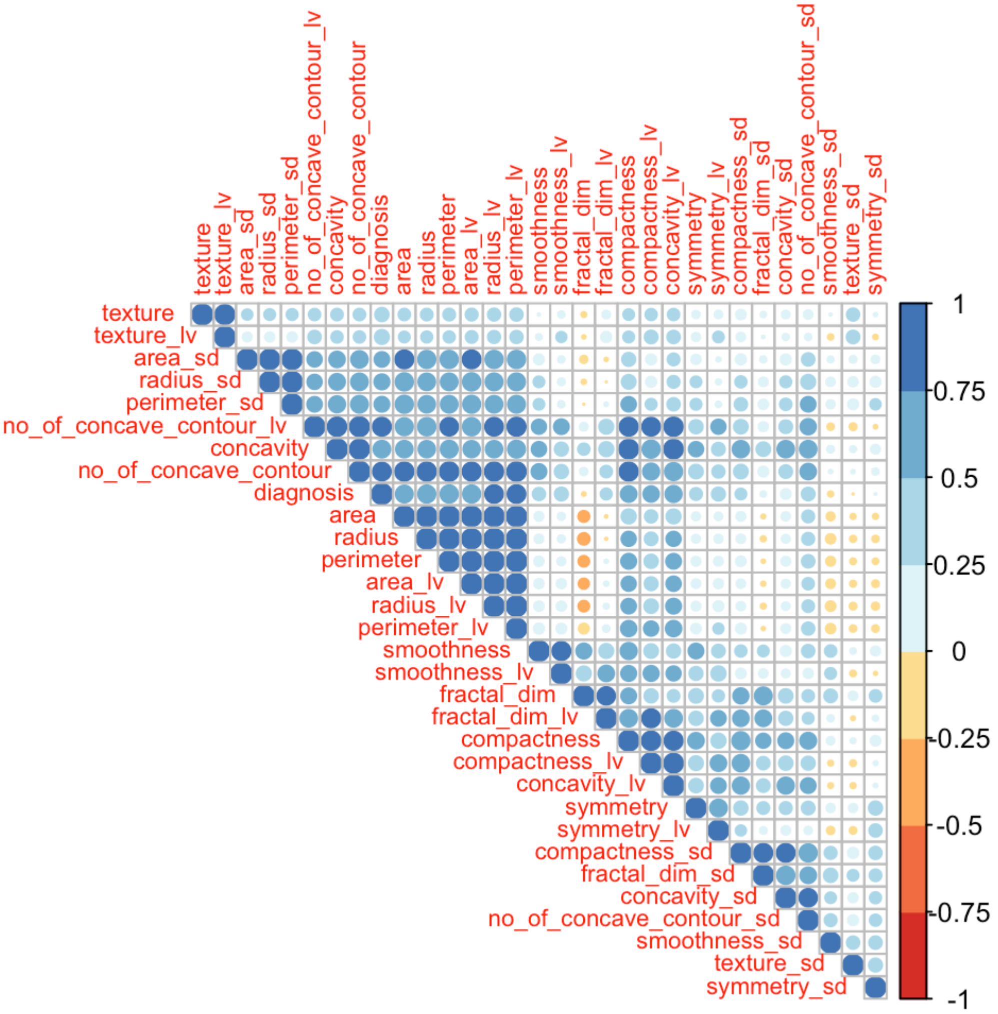


Decision Tree

```
library(rpart.plot)
fit=rpart(diagnosis~perimeter_lv + radius_lv + area_lv + perimeter , data=train_x_
y, parms = list(split="information"), method = 'class')
rpart.plot(fit, extra = 106)
```



```
train_x_y$diagnosis <- as.numeric(train_x_y$diagnosis)
cor_graph <- cor(train_x_y)
corrplot(cor_graph, type="upper",order= "hclust",tl.cex = 0.7,col=brewer.pal(n=8,
name="RdYlBu"))
```



Confusion Matrix for combined

```
t_pred=predict(fit, trainx, type='class')
table(train_x_y$diagnosis, t_pred)
```

```
##    t_pred
##       0   1
##   0 280   6
##   1  19 149
```

```
confusion_mat = table(train_x_y$diagnosis, t_pred)
acc = sum(diag(confusion_mat))/sum(confusion_mat)
print(confusion_mat)
```

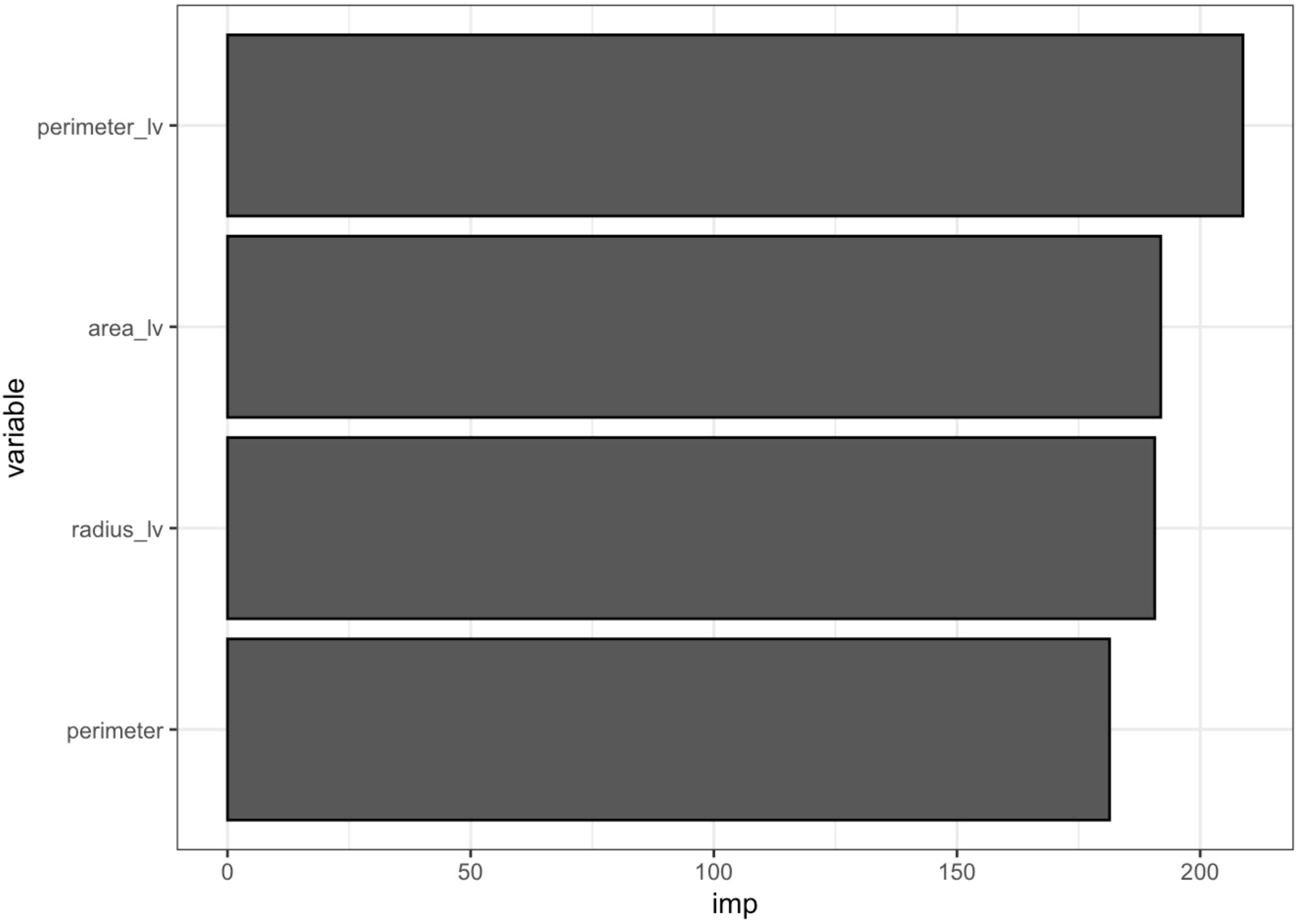
```
##    t_pred
##       0   1
##   0 280   6
##   1  19 149
```

```
print(acc)
```

```
## [1] 0.9449339
```

Bar graph imp vs variable

```
df=data.frame(imp=fit$variable.importance)
df2=df %>%
  tibble::rownames_to_column()%>%
  dplyr::rename("variable"= rowname)%>%
  dplyr::arrange(imp)%>%
  dplyr::mutate(variable = forcats::fct_inorder(variable))
ggplot2::ggplot(df2) +
  geom_col(aes(x=variable, y=imp), col="black", show.legend = F) +
  coord_flip() +
  scale_fill_grey() +
  theme_bw()
```



Confusion Matrix for test

```
t_pred=predict(fit, testx, type='class')
names(testy)=c("diagnosis")
```

Accuracy for Test

```
fit_test <- rpart(diagnosis~.,data=train_x_y_new, parms = list(split="information"
), method = 'class',control=rpart.control(minsplit=5,minbucket=3,cp=0.01))

t_pred=predict(fit_test, testx, type='class')
table(testy$diagnosis, t_pred)
```

```
##    t_pred
##      0  1
##   0 30  1
##   1  2 23
```

```
confusion_mat_test = table(testy$diagnosis, t_pred)
accTest = sum(diag(confusion_mat_test))/sum(confusion_mat_test)
print(confusion_mat_test)
```

```
##    t_pred
##      0  1
##   0 30  1
##   1  2 23
```

```
print(accTest)
```

```
## [1] 0.9464286
```