# Spondylo vs Met

## Downloading the libraries

```
library(randomForest)
library(pROC)
library(caret)
library(kernlab)
library(e1071)
library(readxl)
library(Boruta)
```

## Loading the dataset

```
features <- read_excel("E:/BMF/bakalarka/ML/features.xlsx", sheet = "ML")
```

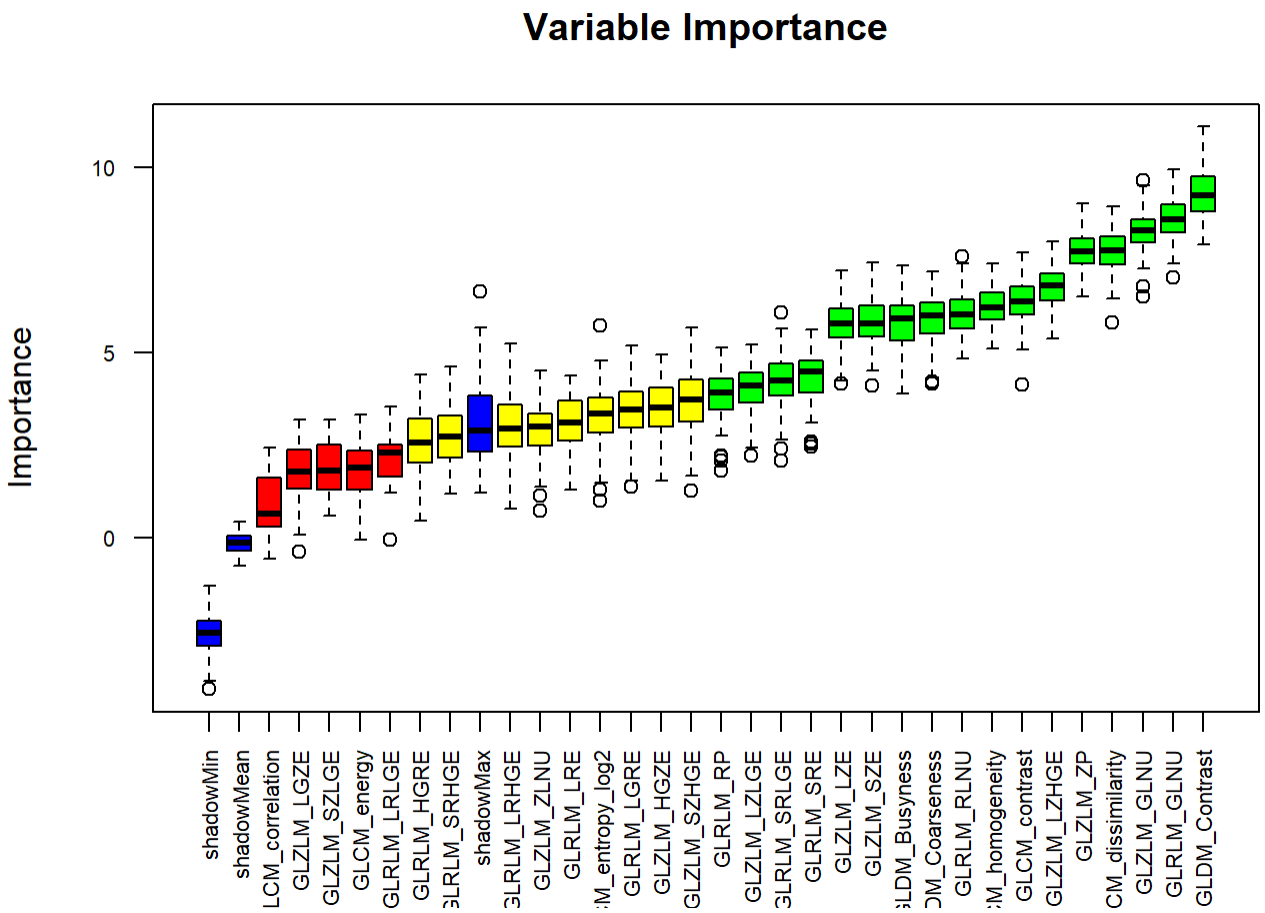
```
#looking at our variables in the dataset

labels(features)
```

```
## [[1]]
##  [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10" "11" "12" "13" "14" "15"
## [16] "16" "17" "18" "19" "20" "21" "22" "23" "24" "25" "26" "27" "28" "29" "30"
## [31] "31" "32" "33" "34" "35" "36" "37" "38" "39" "40" "41" "42" "43" "44" "45"
## [46] "46" "47" "48" "49" "50" "51" "52" "53" "54" "55" "56" "57" "58" "59" "60"
## [61] "61" "62" "63" "64" "65" "66" "67" "68" "69" "70" "71" "72" "73" "74" "75"
## [76] "76" "77" "78" "79" "80"
##
## [[2]]
##  [1] "Disease"            "GLCM_homogeneity"   "GLCM_energy"
##  [4] "GLCM_contrast"      "GLCM_correlation"   "GLCM_entropy_log2"
##  [7] "GLCM_dissimilarity" "GLRLM_SRE"          "GLRLM_LRE"
## [10] "GLRLM_LGRE"         "GLRLM_HGRE"         "GLRLM_SRLGE"
## [13] "GLRLM_SRHGE"        "GLRLM_LRLGE"        "GLRLM_LRHGE"
## [16] "GLRLM_GLNU"         "GLRLM_RLNU"         "GLRLM_RP"
## [19] "NGLDM_Coarseness"   "NGLDM_Contrast"     "NGLDM_Busyness"
## [22] "GLZLM_SZE"          "GLZLM_LZE"          "GLZLM_LGZE"
## [25] "GLZLM_HGZE"         "GLZLM_SZLGE"        "GLZLM_SZHGE"
## [28] "GLZLM_LZLGE"        "GLZLM_LZHGE"        "GLZLM_GLNU"
## [31] "GLZLM_ZLNU"         "GLZLM_ZP"
```

```
# change the quality column to a factor type

features$Disease <- as.factor(features$Disease)
```

# Variable selection using Boruta package

```
boruta_output <- Boruta(factor(Disease) ~ ., data=na.omit(features), doTrace=2)
plot(boruta_output, cex.axis=.7, las=2, xlab="", main="Variable Importance")
```




```
# listing the confirmed

boruta_signif <- names(boruta_output$finalDecision[boruta_output$finalDecision %in% c("Confir
med")])
print(boruta_signif)
```

```
##  [1] "GLCM_homogeneity"   "GLCM_contrast"      "GLCM_dissimilarity"
##  [4] "GLRLM_SRE"          "GLRLM_SRLGE"        "GLRLM_GLNU"
##  [7] "GLRLM_RLNU"         "GLRLM_RP"           "NGLDM_Coarseness"
## [10] "NGLDM_Contrast"     "NGLDM_Busyness"     "GLZLM_SZE"
## [13] "GLZLM_LZE"          "GLZLM_LZLGE"        "GLZLM_LZHGE"
## [16] "GLZLM_GLNU"         "GLZLM_ZP"
```

```
# having overview of all the decisions

vypis <- attStats(boruta_output)
print(vypis)
```

```
##                       meanImp medianImp      minImp    maxImp   normHits
## GLCM_homogeneity   6.2426554 6.2045422  5.11265519  7.416507 1.00000000
## GLCM_energy        1.7478149 1.8737314 -0.07052687  3.313086 0.04040404
## GLCM_contrast      6.3714531 6.3826442  4.13686813  7.707649 1.00000000
## GLCM_correlation   0.8692897 0.6449314 -0.58035563  2.437090 0.00000000
## GLCM_entropy_log2  3.3157079 3.3567103  1.00209190  5.736074 0.56565657
## GLCM_dissimilarity 7.7301597 7.7610835  5.79963946  8.956844 1.00000000
## GLRLM_SRE          4.3580194 4.4830333  2.46279706  5.611787 0.83838384
## GLRLM_LRE          3.0922741 3.1154700  1.30043008  4.363921 0.53535354
## GLRLM_LGRE         3.3995105 3.4529656  1.36214462  5.192067 0.60606061
## GLRLM_HGRE         2.5809748 2.5589612  0.46549513  4.389285 0.40404040
## GLRLM_SRLGE        4.2053190 4.2436694  2.07362366  6.067318 0.77777778
## GLRLM_SRHGE        2.7727550 2.7144326  1.16969028  4.625981 0.42424242
## GLRLM_LRLGE        2.1724423 2.3032972 -0.06442910  3.544783 0.04040404
## GLRLM_LRHGE        2.9972872 2.9475213  0.77169886  5.244343 0.52525253
## GLRLM_GLNU         8.5990661 8.5925916  7.03463667  9.940691 1.00000000
## GLRLM_RLNU         6.0726795 6.0367836  4.84540400  7.605072 0.98989899
## GLRLM_RP           3.8413769 3.9228813  1.79574424  5.125778 0.74747475
## NGLDM_Coarseness   5.9134820 5.9920605  4.16697678  7.193477 0.96969697
## NGLDM_Contrast     9.2441623 9.2292930  7.92173594 11.097048 1.00000000
## NGLDM_Busyness     5.7687783 5.9031805  3.88077967  7.361804 0.97979798
## GLZLM_SZE          5.8137908 5.7703979  4.10386209  7.439920 0.97979798
## GLZLM_LZE          5.7448533 5.7700998  4.16754992  7.201573 0.96969697
## GLZLM_LGZE         1.6902980 1.7852138 -0.37425543  3.196783 0.02020202
## GLZLM_HGZE         3.4422126 3.4968056  1.53410850  4.951833 0.59595960
## GLZLM_SZLGE        1.8854706 1.8121693  0.58557341  3.172229 0.00000000
## GLZLM_SZHGE        3.6887627 3.7141956  1.27389191  5.669782 0.62626263
## GLZLM_LZLGE        4.0452884 4.0926689  2.20778759  5.221742 0.80808081
## GLZLM_LZHGE        6.7682455 6.7972466  5.37978275  7.995508 1.00000000
## GLZLM_GLNU         8.2430796 8.2839078  6.49927152  9.653568 1.00000000
## GLZLM_ZLNU         2.9185087 2.9885437  0.72631094  4.499106 0.46464646
## GLZLM_ZP           7.7620171 7.7352533  6.49875078  9.032895 1.00000000
##                     decision
## GLCM_homogeneity   Confirmed
## GLCM_energy         Rejected
## GLCM_contrast      Confirmed
## GLCM_correlation    Rejected
## GLCM_entropy_log2  Tentative
## GLCM_dissimilarity Confirmed
## GLRLM_SRE          Confirmed
## GLRLM_LRE          Tentative
## GLRLM_LGRE         Tentative
## GLRLM_HGRE         Tentative
## GLRLM_SRLGE        Confirmed
## GLRLM_SRHGE        Tentative
## GLRLM_LRLGE         Rejected
## GLRLM_LRHGE        Tentative
## GLRLM_GLNU         Confirmed
## GLRLM_RLNU         Confirmed
## GLRLM_RP           Confirmed
## NGLDM_Coarseness   Confirmed
```

```
## NGLDM_Contrast     Confirmed
## NGLDM_Busyness     Confirmed
## GLZLM_SZE          Confirmed
## GLZLM_LZE          Confirmed
## GLZLM_LGZE          Rejected
## GLZLM_HGZE         Tentative
## GLZLM_SZLGE         Rejected
## GLZLM_SZHGE        Tentative
## GLZLM_LZLGE        Confirmed
## GLZLM_LZHGE        Confirmed
## GLZLM_GLNU         Confirmed
## GLZLM_ZLNU         Tentative
## GLZLM_ZP           Confirmed
```

# TRAIN AND TEST SET

```
# randomly selecting 70% of the rows in data

index <- sample(1:nrow(features),size = 0.70*nrow(features))

# showing selected rows for our training set

index
```

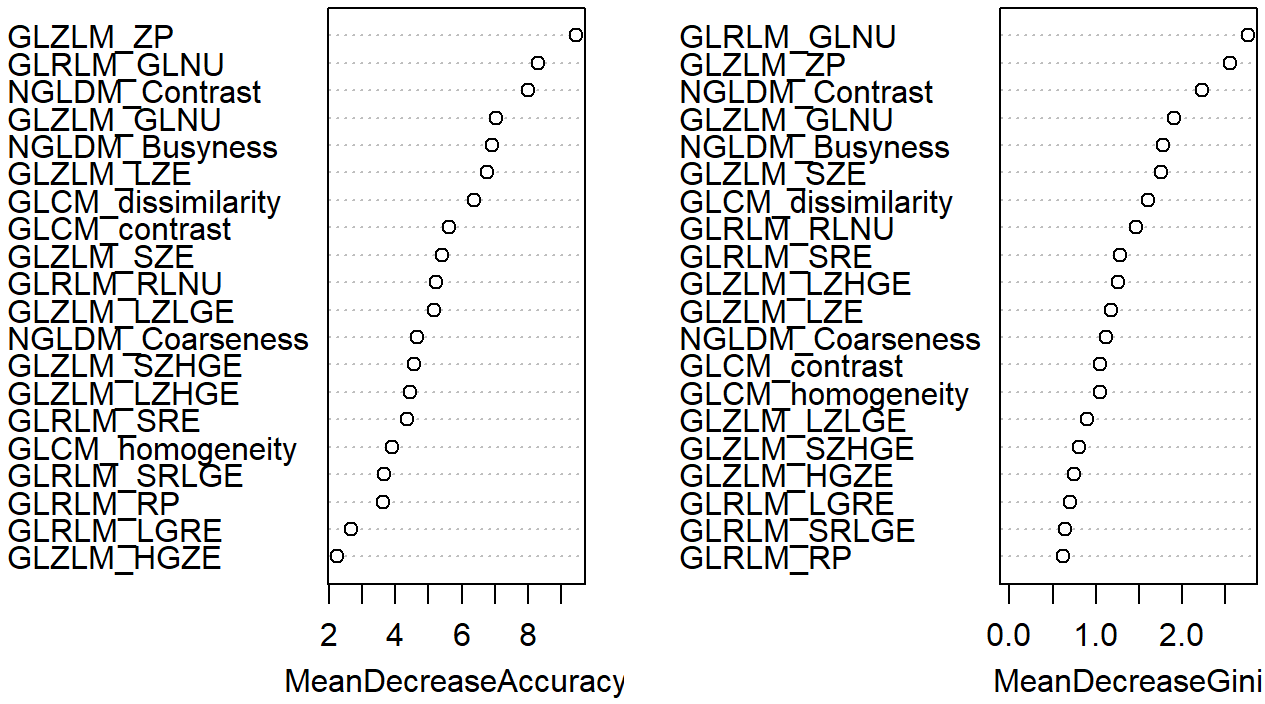
```
##  [1] 23 69 42 31 50 44 79 76 20 80 29  7 71 54 12 19 14 30 47 52 13 36 59 11 17
## [26] 61 32 41 40 67 57 28 43 48 60 63  3  8 64 10 49 39 46 24 26 66 21 38 25 77
## [51]  5  2 74  6 18 70
```

```
# using selected 70% of the data for training set and the rest of 30% for testing set

train.split <- features[index,]
test.split <- features[-index,]
```

## Random Forest Model

```
rf <- randomForest(factor(Disease)~GLCM_homogeneity
                   +GLCM_contrast
                   +GLCM_dissimilarity
                   +GLRLM_SRE
                   +GLZLM_GLNU
                   +GLZLM_ZP
                   +NGLDM_Busyness
                   +GLRLM_LGRE
                   +GLRLM_SRLGE
                   +GLZLM_SZHGE
                   +GLRLM_GLNU
                   +GLRLM_RLNU
                   +GLRLM_RP
                   +GLZLM_LZE
                   +GLZLM_HGZE
                   +GLZLM_LZLGE
                   +GLZLM_LZHGE
                   +NGLDM_Coarseness
                   +NGLDM_Contrast
                   +GLZLM_SZE,data=train.split, ntree=400, mtry=4, na.action = na.omit, impor
tance=TRUE)
```

```
# showing the variable importance performing in the random forest model

varImpPlot(rf)
```



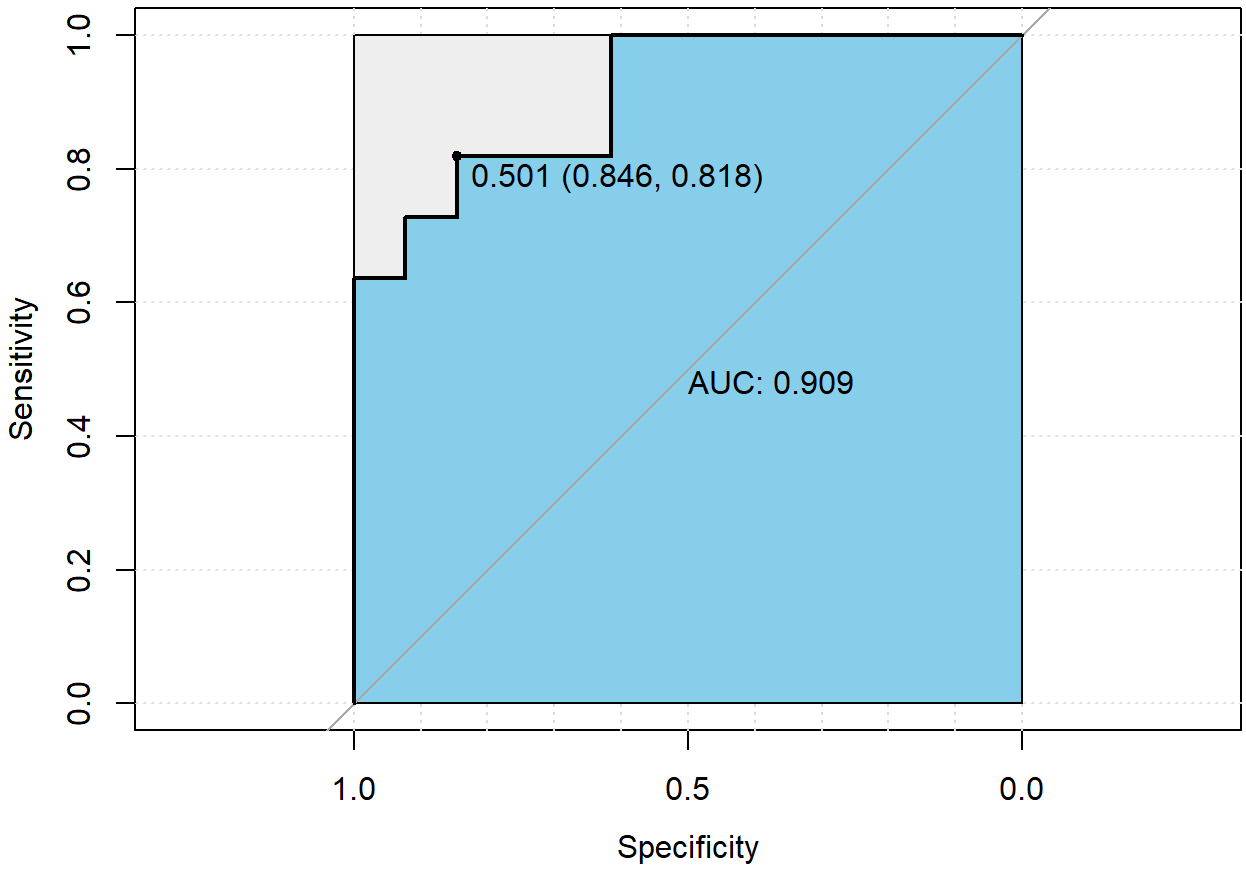

```
# make predictions on our testing data

rf_pred <- predict(rf, test.split)
print(rf_pred)
```

```
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
##  S  S  S  M  M  M  M  M  M  S  M  S  S  S  S  S  S  M  M  S  M  M  M  M
## Levels: M S
```

```
# confusion matrix for the prediction

confusionMatrix(as.factor(rf_pred), as.factor(test.split$Disease))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  M  S
##          M 11  2
##          S  2  9
##
##                Accuracy : 0.8333
##                  95% CI : (0.6262, 0.9526)
##     No Information Rate : 0.5417
##     P-Value [Acc > NIR] : 0.002805
##
##                   Kappa : 0.6643
##
##  Mcnemar's Test P-Value : 1.000000
##
##             Sensitivity : 0.8462
##             Specificity : 0.8182
##          Pos Pred Value : 0.8462
##          Neg Pred Value : 0.8182
##              Prevalence : 0.5417
##          Detection Rate : 0.4583
##    Detection Prevalence : 0.5417
##       Balanced Accuracy : 0.8322
##
##        'Positive' Class : M
##
```

```
# ROC curve - how well it predicted

rf_pred <- predict(rf, test.split, type = "prob")
ROC_rf <- roc(factor(test.split$Disease) ,rf_pred[ ,1])
plot(ROC_rf,
     print.auc=TRUE,
     auc.polygon=TRUE,
     grid=c(0.1, 0.2),
     max.auc.polygon=TRUE,
     print.thres=TRUE,
     auc.polygon.col="skyblue")
```




```
#we can save this model
save(rf , file = 'MyML.rda')
```

## Logistic regression model

```
# logistic regression model

lr <- glm(factor(Disease)~GLCM_homogeneity
          +GLCM_contrast
          +GLCM_dissimilarity
          +GLRLM_SRE
          +GLZLM_GLNU
          +GLZLM_ZP
          +NGLDM_Busyness
          +GLRLM_LGRE
          +GLRLM_SRLGE
          +GLZLM_SZHGE
          +GLRLM_GLNU
          +GLRLM_RLNU
          +GLRLM_RP
          +GLZLM_LZE
          +GLZLM_LZLGE
          +NGLDM_Coarseness
          +NGLDM_Contrast
          +GLZLM_SZE, data = train.split, family = "binomial", control= list(maxit=150))
```

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

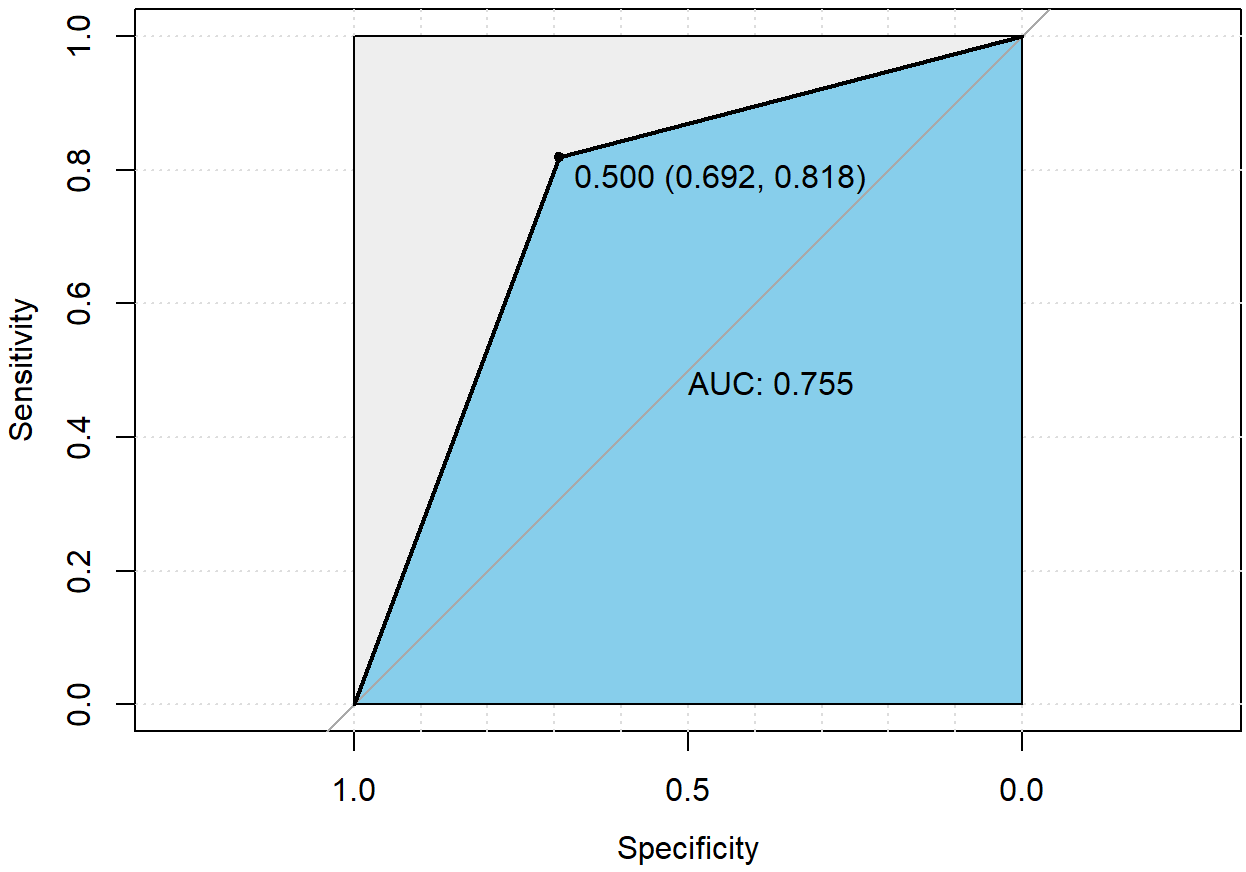
```
print(lr)
```

```
##
## Call:  glm(formula = factor(Disease) ~ GLCM_homogeneity + GLCM_contrast +
##     GLCM_dissimilarity + GLRLM_SRE + GLZLM_GLNU + GLZLM_ZP +
##     NGLDM_Busyness + GLRLM_LGRE + GLRLM_SRLGE + GLZLM_SZHGE +
##     GLRLM_GLNU + GLRLM_RLNU + GLRLM_RP + GLZLM_LZE + GLZLM_LZLGE +
##     NGLDM_Coarseness + NGLDM_Contrast + GLZLM_SZE, family = "binomial",
##     data = train.split, control = list(maxit = 150))
##
## Coefficients:
##        (Intercept)    GLCM_homogeneity       GLCM_contrast  GLCM_dissimilarity
##          1.044e+17          -4.297e+16           1.011e+14          -4.331e+15
##          GLRLM_SRE          GLZLM_GLNU            GLZLM_ZP      NGLDM_Busyness
##         -1.174e+17           2.626e+13           1.984e+16          -2.219e+14
##         GLRLM_LGRE         GLRLM_SRLGE         GLZLM_SZHGE          GLRLM_GLNU
##          4.980e+17          -5.296e+17          -2.145e+11          -2.089e+12
##         GLRLM_RLNU            GLRLM_RP           GLZLM_LZE         GLZLM_LZLGE
##          3.966e+11           3.059e+16           9.411e+10          -8.416e+12
##   NGLDM_Coarseness      NGLDM_Contrast           GLZLM_SZE
##         -5.670e+16          -3.767e+15           1.351e+15
##
## Degrees of Freedom: 55 Total (i.e. Null);  37 Residual
## Null Deviance:       77.56
## Residual Deviance: 360.4     AIC: 398.4
```

```
# make predictions for our testing data
lr_pred <- predict(lr, test.split, type = "response")
lr_pred
```

```
##            1            2            3            4            5            6
## 1.000000e+00 2.220446e-16 1.000000e+00 1.000000e+00 1.000000e+00 2.220446e-16
##            7            8            9           10           11           12
## 2.220446e-16 2.220446e-16 2.220446e-16 1.000000e+00 2.220446e-16 1.000000e+00
##           13           14           15           16           17           18
## 1.000000e+00 1.000000e+00 1.000000e+00 1.000000e+00 2.220446e-16 1.000000e+00
##           19           20           21           22           23           24
## 1.000000e+00 1.000000e+00 2.220446e-16 2.220446e-16 2.220446e-16 2.220446e-16
```

```
# ROC curve - how well it predicted
ROC_lr <- roc(factor(test.split$Disease), lr_pred)
plot(ROC_lr, print.auc=TRUE, auc.polygon=TRUE, grid=c(0.1, 0.2),
     max.auc.polygon=TRUE,print.thres=TRUE,auc.polygon.col="skyblue")
```

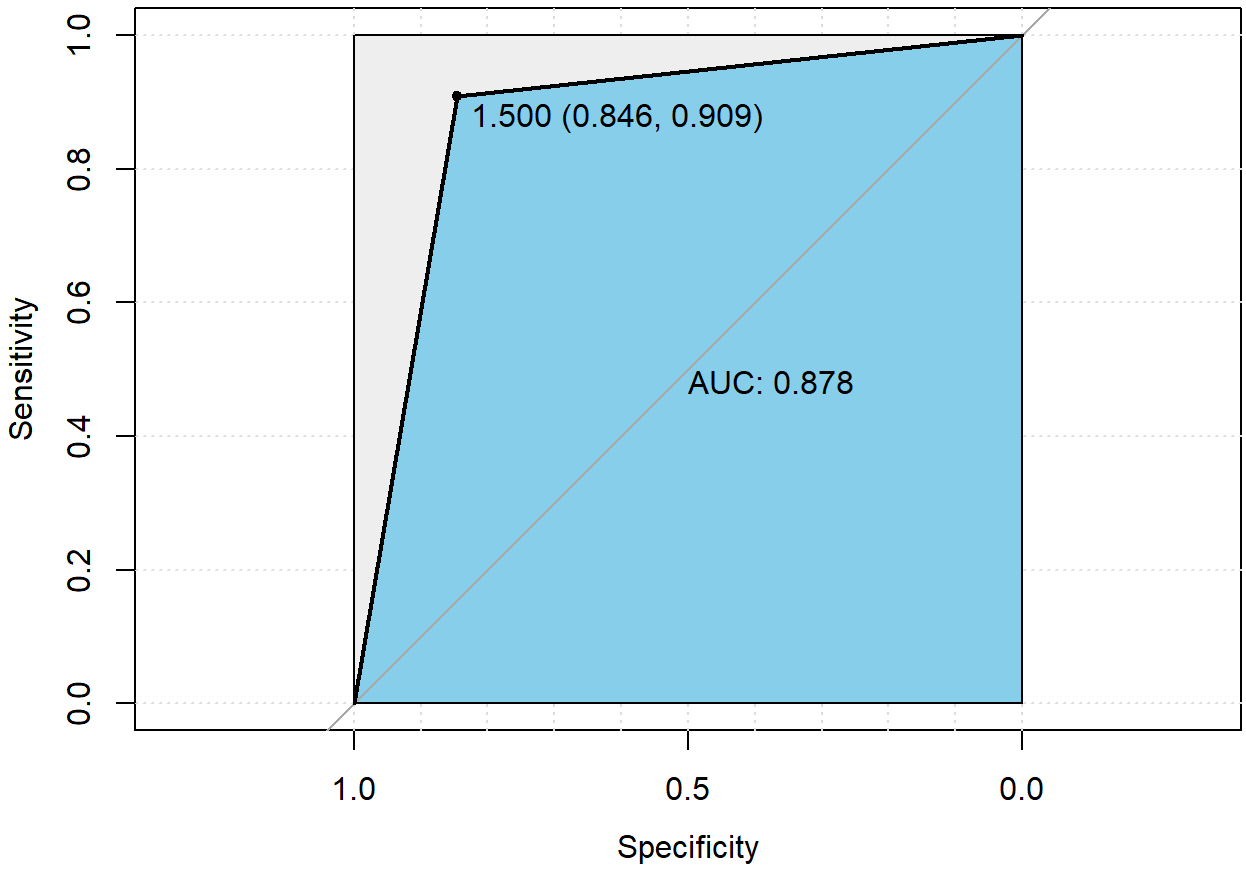



## Support vector machines

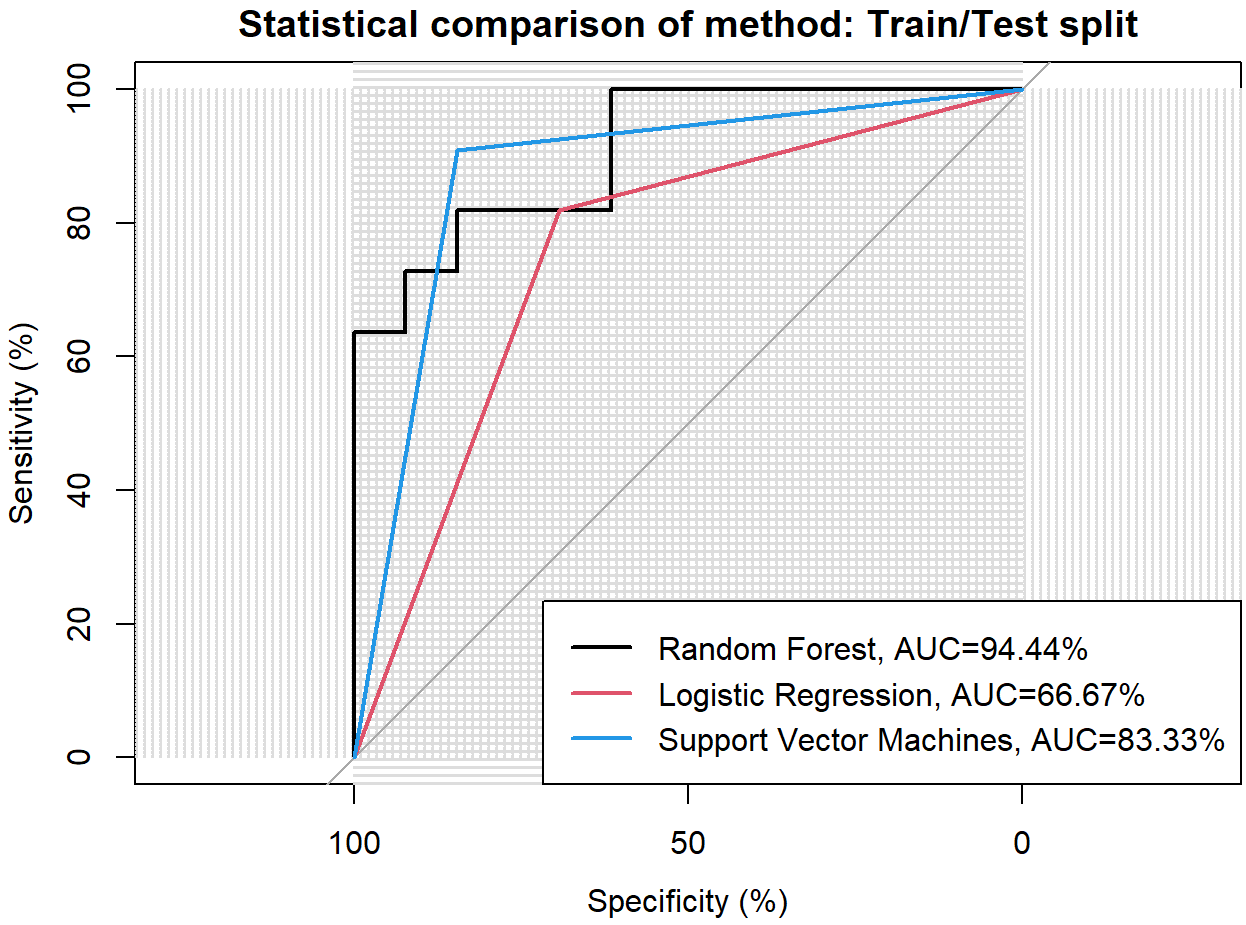
```
# support vector machine model

svm_traintest <- svm(factor(Disease)~GLCM_homogeneity
                     +GLCM_contrast
                     +GLCM_dissimilarity
                     +GLRLM_SRE
                     +GLZLM_GLNU
                     +GLZLM_ZP
                     +NGLDM_Busyness
                     +GLRLM_LGRE
                     +GLRLM_SRLGE
                     +GLZLM_SZHGE
                     +GLRLM_GLNU
                     +GLRLM_RLNU
                     +GLRLM_RP
                     +GLZLM_LZE
                     +GLZLM_LZLGE
                     +NGLDM_Coarseness
                     +NGLDM_Contrast
                     +GLZLM_SZE,data=train.split)
```

```
# making predictions on our testing data
svm_traintest_pred <- predict(svm_traintest, test.split, type = "prob")
```

```
# ROC curve - how well it predicted

ROC_svm1 <- roc(factor(test.split$Disease), as.ordered(svm_traintest_pred))
plot(ROC_svm1,print.auc=TRUE, auc.polygon=TRUE, grid=c(0.1, 0.2),
     max.auc.polygon=TRUE,print.thres=TRUE,auc.polygon.col="skyblue")
```




```
# making comparison of all three methods

roc.rf1 <- plot.roc(factor(test.split$Disease), rf_pred[ ,2], main="Statistical comparison of
method: Train/Test split", col="1", percent=TRUE,grid=c(0.1, 0.2))
roc.lr1 <- lines.roc(factor(test.split$Disease), lr_pred, col="2", percent=TRUE)
roc.svm1 <- lines.roc(factor(test.split$Disease), as.ordered(svm_traintest_pred), col="4", pe
rcent=TRUE)

legend("bottomright", legend=c("Random Forest, AUC=94.44%", "Logistic Regression, AUC=66.6
7%", "Support Vector Machines, AUC=83.33%"), col=c("1", "2", "4"), lwd=2)
```



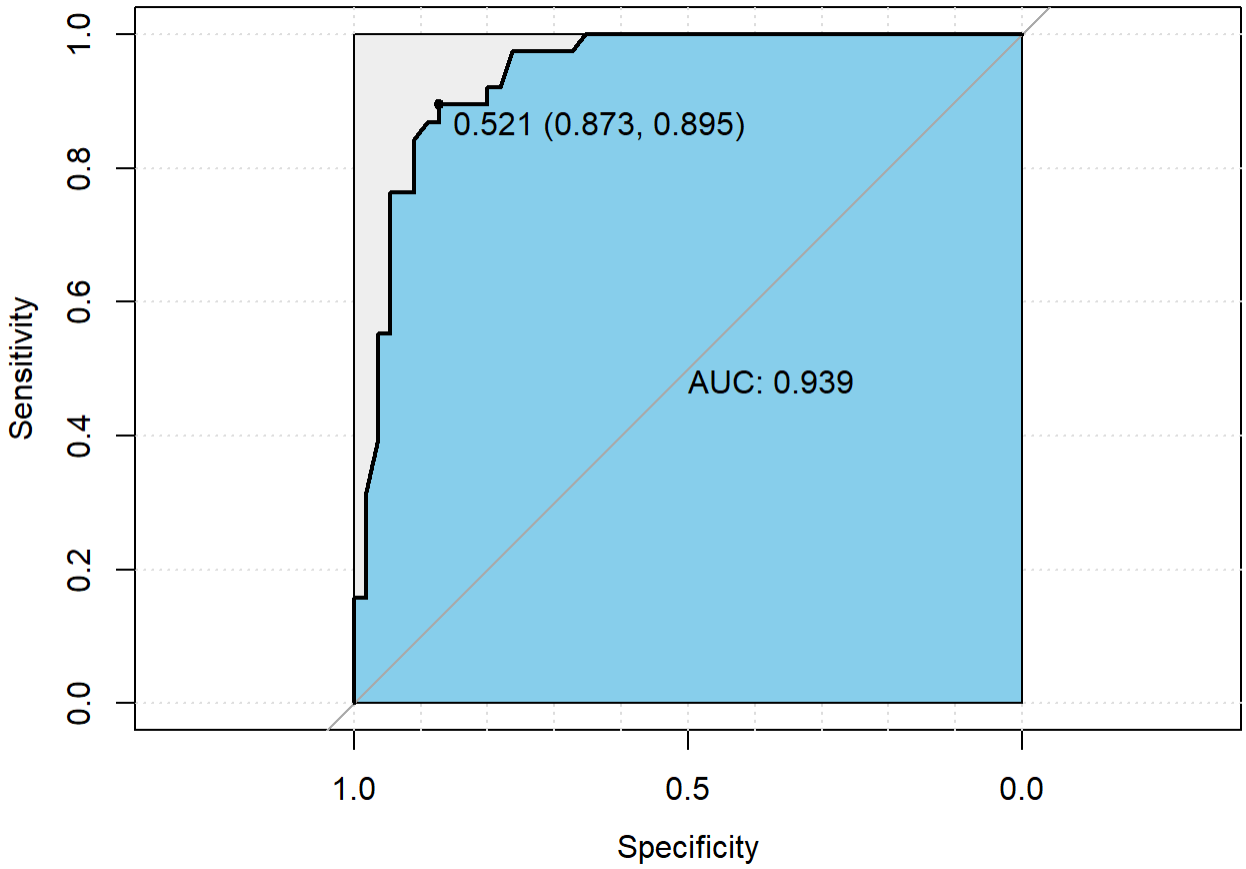

## Validation of the best model - random forest model

```
# uploading our validation set

validacia <- read_excel("validacia.xlsx")
```

```
## New names:
## * `` -> ...2
## * `` -> ...37
## * `` -> ...38
## * `` -> ...39
## * `` -> ...40
## * ...
```

```
# making predictions

valid <- predict(rf, validacia, type = "prob")
```

```
# plotting ROC curve

ROC_rf_valid <- roc(factor(validacia$Disease) ,valid[ ,2])

plot(ROC_rf_valid, print.auc=TRUE, auc.polygon=TRUE, grid=c(0.1, 0.2),
     max.auc.polygon=TRUE,print.thres=TRUE,
     auc.polygon.col="skyblue")
```




```
# we can show confidence intervals with: ci.auc(ROC_rf_valid)
```