project2\_rf\_classification

EMVP

2023-05-02

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.4.0 ✔ purrr 1.0.1   
## ✔ tibble 3.1.8 ✔ dplyr 1.0.10  
## ✔ tidyr 1.3.0 ✔ stringr 1.5.0   
## ✔ readr 2.1.4 ✔ forcats 1.0.0   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(caret)

## Loading required package: lattice  
##   
## Attaching package: 'caret'  
##   
## The following object is masked from 'package:purrr':  
##   
## lift

library(randomForest)

## randomForest 4.7-1.1  
## Type rfNews() to see new features/changes/bug fixes.  
##   
## Attaching package: 'randomForest'  
##   
## The following object is masked from 'package:dplyr':  
##   
## combine  
##   
## The following object is masked from 'package:ggplot2':  
##   
## margin

data <- read.csv('/Users/mustafayigitisik/Desktop/stuff/semesters/spring 2023/ams 580/project2/Titanic.csv')  
data <- subset(data, select = -c(PassengerId,Name,Ticket,Cabin))  
# Remember we have added "PassengerId" into the removal list  
data <- subset(data, is.na(Age) == FALSE)  
data$Survived <- as.factor(data$Survived)  
str(data)

## 'data.frame': 714 obs. of 8 variables:  
## $ Survived: Factor w/ 2 levels "0","1": 1 2 2 2 1 1 1 2 2 2 ...  
## $ Pclass : int 3 1 3 1 3 1 3 3 2 3 ...  
## $ Sex : chr "male" "female" "female" "female" ...  
## $ Age : num 22 38 26 35 35 54 2 27 14 4 ...  
## $ SibSp : int 1 1 0 1 0 0 3 0 1 1 ...  
## $ Parch : int 0 0 0 0 0 0 1 2 0 1 ...  
## $ Fare : num 7.25 71.28 7.92 53.1 8.05 ...  
## $ Embarked: chr "S" "C" "S" "S" ...

dim(data)[1] # 714 passengers

## [1] 714

set.seed(123)  
split <- data$Survived %>%   
createDataPartition(p = 0.75, list = FALSE)  
train <- data[split,]  
test <- data[-split,]  
str(train) #536

## 'data.frame': 536 obs. of 8 variables:  
## $ Survived: Factor w/ 2 levels "0","1": 2 2 1 1 2 2 2 1 1 2 ...  
## $ Pclass : int 1 3 3 3 2 3 1 3 3 2 ...  
## $ Sex : chr "female" "female" "male" "male" ...  
## $ Age : num 38 26 35 2 14 4 58 20 14 55 ...  
## $ SibSp : int 1 0 0 3 1 1 0 0 0 0 ...  
## $ Parch : int 0 0 0 1 0 1 0 0 0 0 ...  
## $ Fare : num 71.28 7.92 8.05 21.07 30.07 ...  
## $ Embarked: chr "C" "S" "S" "S" ...

str(test) #178

## 'data.frame': 178 obs. of 8 variables:  
## $ Survived: Factor w/ 2 levels "0","1": 1 2 1 2 1 1 2 1 1 1 ...  
## $ Pclass : int 3 1 1 3 3 3 2 1 1 3 ...  
## $ Sex : chr "male" "female" "male" "female" ...  
## $ Age : num 22 35 54 27 39 31 34 19 28 21 ...  
## $ SibSp : int 1 1 0 0 1 1 0 3 1 0 ...  
## $ Parch : int 0 0 0 2 5 0 0 2 0 0 ...  
## $ Fare : num 7.25 53.1 51.86 11.13 31.27 ...  
## $ Embarked: chr "S" "S" "S" "S" ...

#Q2  
set.seed(123)  
model <- train(  
 Survived ~., data = train, method = "rf",  
 trControl = trainControl("cv", number = 10),  
 importance = TRUE  
 )  
# Best tuning parameter  
model$bestTune

## mtry  
## 1 2

model$finalModel

##   
## Call:  
## randomForest(x = x, y = y, mtry = param$mtry, importance = TRUE)   
## Type of random forest: classification  
## Number of trees: 500  
## No. of variables tried at each split: 2  
##   
## OOB estimate of error rate: 21.27%  
## Confusion matrix:  
## 0 1 class.error  
## 0 284 34 0.1069182  
## 1 80 138 0.3669725

#Q3  
#Sensitivity  
138/(80+138)

## [1] 0.6330275

#Specificity  
284/(284+34)

## [1] 0.8930818

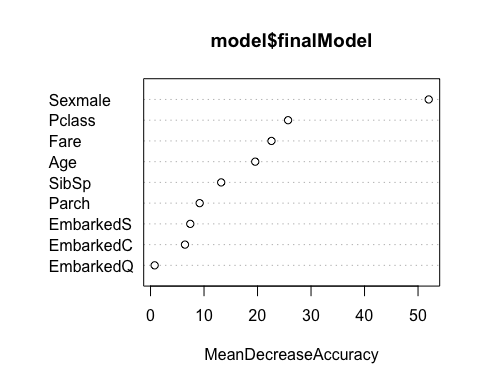
#Accuracy  
(138+284)/(138+284+80+34)

## [1] 0.7873134

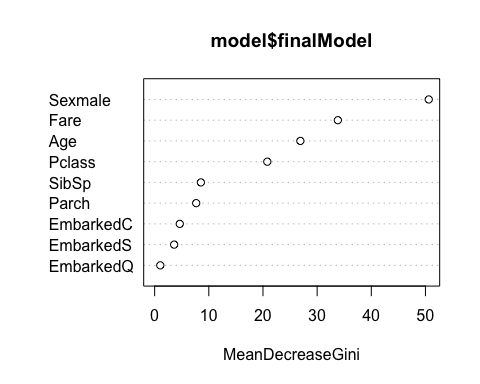
#Confusion matrix  
pred <- model %>% predict(test)  
confusionMatrix(factor(pred), factor(test$Survived))

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 90 19  
## 1 16 53  
##   
## Accuracy : 0.8034   
## 95% CI : (0.7373, 0.8591)  
## No Information Rate : 0.5955   
## P-Value [Acc > NIR] : 2.711e-09   
##   
## Kappa : 0.5891   
##   
## Mcnemar's Test P-Value : 0.7353   
##   
## Sensitivity : 0.8491   
## Specificity : 0.7361   
## Pos Pred Value : 0.8257   
## Neg Pred Value : 0.7681   
## Prevalence : 0.5955   
## Detection Rate : 0.5056   
## Detection Prevalence : 0.6124   
## Balanced Accuracy : 0.7926   
##   
## 'Positive' Class : 0   
##

#Q4  
# Plot MeanDecreaseAccuracy  
varImpPlot(model$finalModel, type = 1)



# Plot MeanDecreaseGini  
varImpPlot(model$finalModel, type = 2)



#Q5  
varImp(model)

## rf variable importance  
##   
## Importance  
## Sexmale 100.000  
## Pclass 39.324  
## Fare 32.425  
## Age 29.287  
## SibSp 15.131  
## Parch 11.011  
## EmbarkedS 9.140  
## EmbarkedC 8.673  
## EmbarkedQ 0.000

# Q6  
sqrt(36)

## [1] 6