L9

Kovel

31 РѕРєС‚СЏР±СЂСЏ 2020 Рі

# Download the data

set.seed(123)  
f\_train <- read.csv2('education\_train.csv', header = TRUE, encoding = 'UNICOD')  
f\_test <- read.csv2('education\_test.csv', header = TRUE, encoding = 'UNICOD')  
f\_train <- f\_train[-1]  
f\_test <- f\_test[-1]

## Висновок: завантажано датасет,який було розподілено на навчальну та тестову вибірки.

# Fitting

# install.packages('rpart')  
library(rpart)  
f\_train$school <- as.factor(f\_train$school)  
f\_test$school <- as.factor(f\_test$school)  
class\_dt = rpart(school ~ ., data = f\_train)

## Висновок: базову модель дерева побудовано на основі всіх змінних.

## Predicting

y <- predict(class\_dt, f\_test[-10], type = 'class')

## Висновок: визначені класи об’єктів (вектор у).

## Confusion Matrix

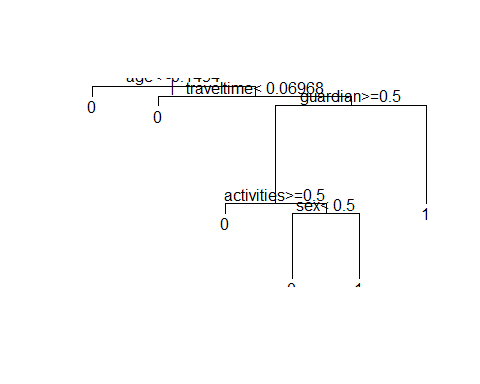
cm = table(f\_test[, 'school'], y)  
print(cm)

## y  
## 0 1  
## 0 112 4  
## 1 10 5

## Висновок: точність моделі – (112+5) / 131 = 89,3 %, частка невірно класифікованих випадків – (10+4) / 131 = 10,7 %. Чутливість – 5 / (10+5) = 34 %, специфічність – 112 / (112+4) = 96 %, тобто модель суттєво більш чутлива до виявленнянегативних випадків. У цьому разі – попадання до школи другого типу.

# Plotting the tree

plot(class\_dt)  
text(class\_dt)

 ##Висновок: візуалізація дозволяє проаналізувати логіку побудови дерева. # Fitting 2 factors

class\_ct = rpart(school ~ age + traveltime, data = f\_train)

## Висновок: проведено навчання моделі дерева рішень із включенням двох значющих факторів.

## Predicting

y <- predict(class\_ct, f\_test[, c('age','traveltime')], type = 'class')

## Висновок: визначено класи об’єктів (вектор у). Для цього використано параметр type = ‘class’.

## Confusion Matrix

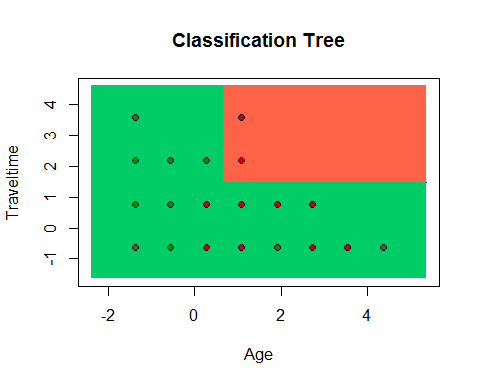
cm = table(f\_test[, 'school'], y)  
print(cm)

## y  
## 0 1  
## 0 115 1  
## 1 13 2

## Висновок: точність моделі – (115+2) / 131 = 89,3 %, частка невірно класифікованих випадків – (13+1) / 131 = 10,7 %. Чутливість – 2 / (13+2) = 14 %, специфічність – 115 / (115+1) = 99,1 %, тобто модель суттєво більш чутлива до виявлення негативних випадків. У цьому разі також – попадання до школи другого типу.

# Visualising the Test set results

library(ggplot2)  
set = f\_test[,c('age','traveltime','school')]  
X1 = seq(min(set['age']) - 1, max(set['age']) + 1, by = 0.01)  
X2 = seq(min(set['traveltime']) - 1, max(set['traveltime']) + 1, by = 0.01)  
grid\_set = expand.grid(X1, X2)  
colnames(grid\_set) = c('age', 'traveltime')  
y\_grid = predict(class\_ct, grid\_set, type = 'class')  
plot(set[, -3],  
 main = 'Classification Tree',  
 xlab = 'Age', ylab = 'Traveltime',  
 xlim = range(X1), ylim = range(X2))  
contour(X1, X2, matrix(as.numeric(y\_grid), length(X1), length(X2)), add = TRUE)  
points(grid\_set, pch = '.', col = ifelse(y\_grid == 1, 'tomato', 'springgreen3'))  
points(set, pch = 21, bg = ifelse(set[, 3] == 1, 'red3', 'green4'))

 ##Висновок: на графіку червоним позначені випадки потрапляння до школи першого типу, зеленим – ло другого. Зеленим виділена зона високої ймовірності потрапляння до другого типу. Модель описує нелінійний варіант розподіляючої кривої. # Fitting Random Forest Classification to the Training set

# install.packages('randomForest')  
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

set.seed(123)  
class\_rf = randomForest(school ~ age + traveltime, data = f\_train, ntree = 10)

## Висновок: проведено навчання моделі випадкового лісу.

## Predicting

y <- predict(class\_rf, f\_test[, c('age','traveltime')])

## Висновок: визначені класи об’єктів (вектор у). Для цього використано параметр type = ‘class’.

## Confusion Matrix

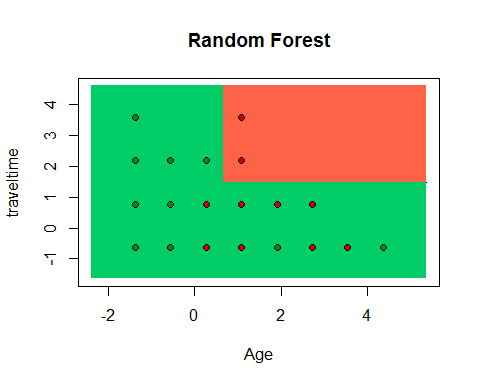
cm = table(f\_test[, 'school'], y)  
print(cm)

## y  
## 0 1  
## 0 115 1  
## 1 13 2

## Висновок: точність моделі – (115+2) / 131 = 89,3 %, частка невірно класифікованих випадків – (13+1) / 131 = 10,7 %. Чутливість – 2 / (13+2) = 14 %, специфічність – 115 / (115+1) = 99,1 %, тобто модель суттєво більш чутлива до виявлення негативних випадків. У цьому разі всі ці характеристики моделі залишилися на тому самому рівні.

# Visualising the Test set results

set = f\_test[,c('age','traveltime','school')]  
X1 = seq(min(set['age']) - 1, max(set['age']) + 1, by = 0.01)  
X2 = seq(min(set['traveltime']) - 1, max(set['traveltime']) + 1, by = 0.01)  
grid\_set = expand.grid(X1, X2)  
colnames(grid\_set) = c('age', 'traveltime')  
y\_grid = predict(class\_rf, grid\_set)  
plot(set[, -3],  
 main = 'Random Forest',  
 xlab = 'Age', ylab = 'traveltime',  
 xlim = range(X1), ylim = range(X2))  
contour(X1, X2, matrix(as.numeric(y\_grid), length(X1), length(X2)), add = TRUE)  
points(grid\_set, pch = '.', col = ifelse(y\_grid == 1, 'tomato', 'springgreen3'))  
points(set, pch = 21, bg = ifelse(set[, 3] == 1, 'red3', 'green4'))

 ##Висновок:красне поле-ймовірність того,що учень навчвється у школі першого типу, зелене - другого.