Лабораторна робота №3. Дерева рішень і випадковий ліс

студентки групи ЕП-61

Лузанової О.С.

Мета роботи – побудова моделі нелінійної регресії - Дерева рішень і випадковий ліс, проаналізувати середньоквадратичні помилки та якість моделі.

Після виконання роботи студент повинен:

ЗНАТИ сутність та принципи побудови моделей нелінійних регресій. А саме - Дерева рішень і випадковий ліс.

УМІТИ організувати та провести моделювання нелінійних регресійних моделей за алгоритмом дерева рішень і випадкового лісу.

Завдання:

1. Побудувати дерево рішень, представити візуалізацію.
2. Побудувати випадковий ліс, представити візуалізацію.
3. На базі випадкового лісу зробити прогноз.
4. Виконати розрахунки на тренувальній та тестовій вибірці.
5. Зробити аналіз та висновки, щодо прогностичних показників моделей.

Зробити висновки щодо якості моделей та прогнозу.

NONLINEAR REGRESSION

# Download the data

#Source files are here  
#setwd('D:/luzanova')  
  
##Features scaling is included in the packages we will work with  
  
#Download the files  
f\_train <- read.csv2('crime\_train.csv', header = TRUE, encoding = 'UNICOD')  
f\_test <- read.csv2('crime\_test.csv', header = TRUE, encoding = 'UNICOD')

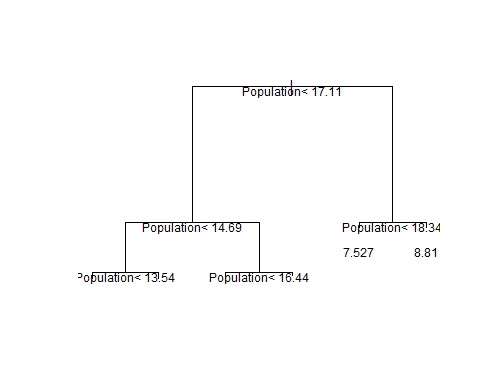
#Висновок: окремо завантажені навчальна і тестова вибірки. # Decision Tree Regression

## Fitting

# install.packages('rpart')  
library(rpart)

## Warning: package 'rpart' was built under R version 3.6.3

model\_dt <- rpart(Homicide ~ Population, f\_train, control = rpart.control(minsplit = 10))  
plot(model\_dt)  
text(model\_dt, pos = 1, cex = 0.75, col = 1, font = 1)

 #Висновок: побудовано дерево рішень, екзогенна змінна – Population.

## Predicting

p\_dt <- predict(model\_dt, f\_test)  
  
train\_mse\_dt <- sum((f\_train$Homicide-predict(model\_dt, f\_train))^2)/length(f\_train$Homicide)  
test\_mse\_dt <- sum((f\_test$Homicide-p\_dt)^2)/length(p\_dt)  
  
train\_mse\_dt

## [1] 1.510216

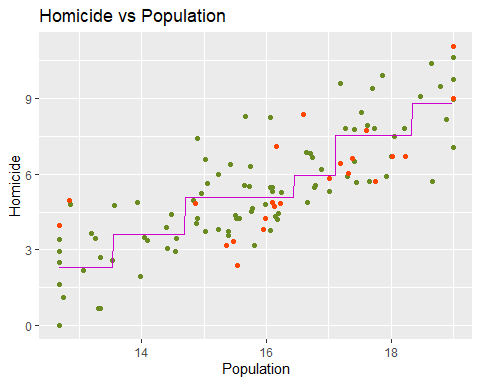
test\_mse\_dt

## [1] 2.187112

#Висновок: значення середньоквадратичної похибки покращилися на навчальній вибірці – 1.51, погіршилися на тестовій вибірці – 2.187. Модель перенавчено.

## Visualising

library(ggplot2)  
x\_grid <- seq(min(f\_train$Population), max(f\_train$Population), 0.01)  
ggplot() +  
 geom\_point(aes(f\_train$Population, f\_train$Homicide),colour = 'olivedrab4') +  
 geom\_point(aes(f\_test$Population, f\_test$Homicide),colour = 'orangered') +  
 geom\_line(aes(x\_grid, predict(model\_dt, data.frame(Population = x\_grid))),colour = 'magenta3') +  
 ggtitle('Homicide vs Population') +  
 xlab('Population') +  
 ylab('Homicide')



#Висновок: на графіку червоним позначені точки навчальної вибірки, зеленим – точки тестової вибірки, синім – модельні значення.

# Random forest

## Fitting

# install.packages('randomForest')  
library(randomForest)

## Warning: package 'randomForest' was built under R version 3.6.3

## 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(1234)  
model\_rf = randomForest(x = f\_train['Population'],  
 y = f\_train$Homicide,  
 ntree = 50)

#Висновок: побудовано віпадковий ліс із 50 дерев, екзогенна змінна – Population.

## Predicting

p\_rf <- predict(model\_rf, f\_test)  
  
train\_mse\_rf <- sum((f\_train$Homicide-predict(model\_rf, f\_train))^2)/length(f\_train$Homicide)  
test\_mse\_rf <- sum((f\_test$Homicide-p\_rf)^2)/length(p\_rf)  
  
train\_mse\_rf

## [1] 0.7152313

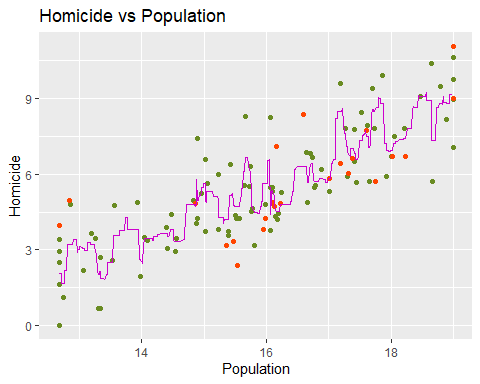
test\_mse\_rf

## [1] 2.152029

#Висновок: значення середньоквадратичної похибки покращилися на навчальній вибірці – 0.715, покращилися на тестовій вибірці – 2,152. Модель перенавчено.

## Visualising

ggplot() +  
 geom\_point(aes(f\_train$Population, f\_train$Homicide),colour = 'olivedrab4') +  
 geom\_point(aes(f\_test$Population, f\_test$Homicide),colour = 'orangered') +  
 geom\_line(aes(x\_grid, predict(model\_rf, data.frame(Population = x\_grid))),colour = 'magenta3') +  
 ggtitle('Homicide vs Population') +  
 xlab('Population') +  
 ylab('Homicide')



#Висновок: на графіку червоним позначені точки навчальної вибірки, зеленим – точки тестової вибірки, синім – модельні значення.

# Saving results

fit <- read.csv2('crime\_fit.csv', header = TRUE, encoding = 'UNICOD')  
fit$p\_dt <- p\_dt  
fit$p\_rf <- p\_rf  
head(fit)

## X p\_sr p\_mr p\_pr p\_dt p\_rf  
## 1 1 6.646040 8.159338 6.591823 5.970051 5.850473  
## 2 2 7.041017 6.275327 7.019894 7.526866 7.027045  
## 3 3 5.806235 7.353415 5.710147 5.078873 4.792396  
## 4 4 6.827411 8.553121 6.787329 7.526866 8.473266  
## 5 5 5.741120 6.312725 5.643411 5.078873 4.517080  
## 6 6 2.015502 1.996263 2.219673 2.305050 2.058364

write.csv2(fit[-1], file = "crime\_fit.csv")

Висновки

В ході виконання Лабораторної роботи №3 було виконано усі поставлені завдання. Модель на базі випадкового лісу дала найкращі результати прогнозування в порівнянні з результатами Лаб. №2, але, варто зазначити, що як і в лінійній регресії, так і в поліноміальній лінійній регресії наявне перенавчання.