

Ćwiczenia:

Modele regresyjne. Modele klasyfikacyjne

Ocena jakości modeli.

Przykład 1.

Klasyfikacja z kNN – dane iris

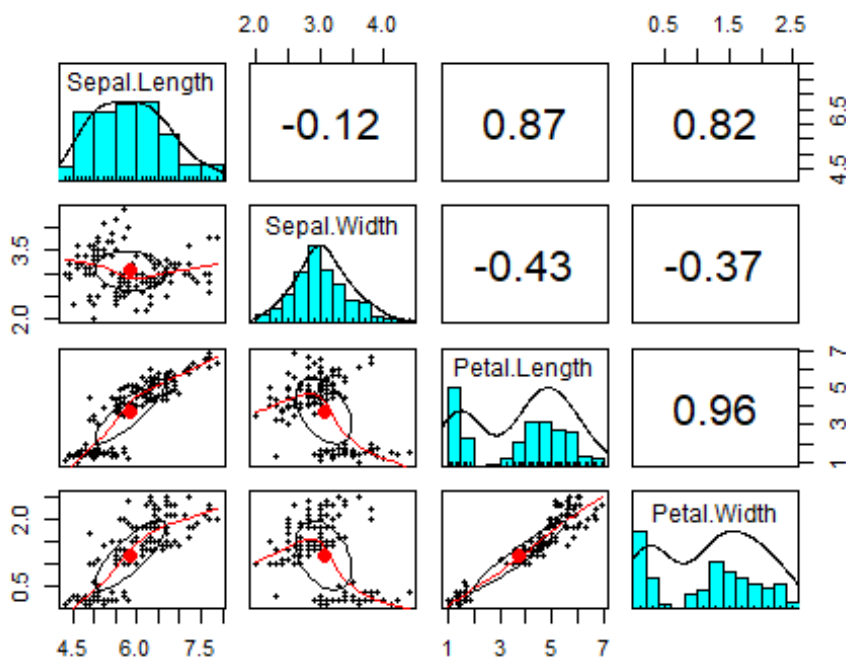
```
table(iris$Species)

##
##      setosa versicolor  virginica
##         50         50         50

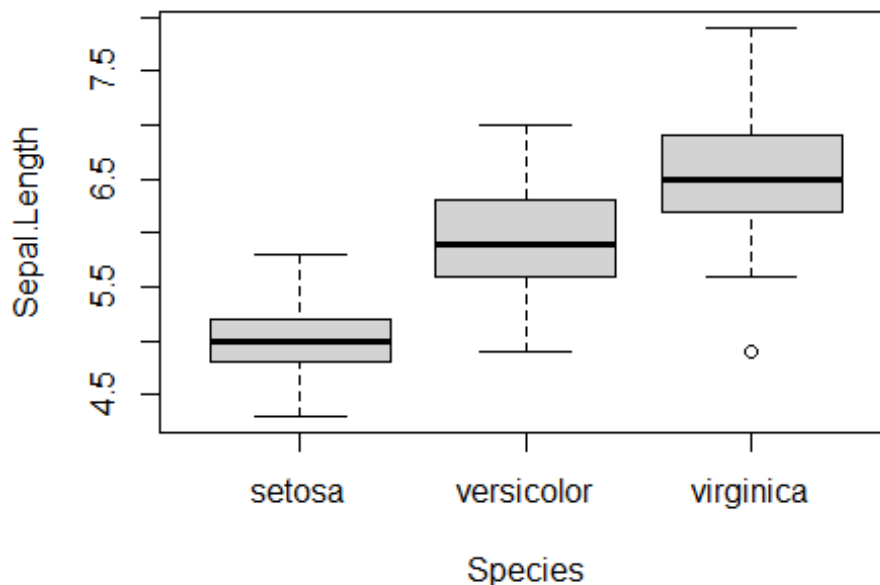
cor(iris[-5])

##              Sepal.Length Sepal.Width Petal.Length Petal.Width
## Sepal.Length      1.0000000 -0.1175698  0.8717538  0.8179411
## Sepal.Width      -0.1175698  1.0000000 -0.4284401 -0.3661259
## Petal.Length      0.8717538 -0.4284401  1.0000000  0.9628654
## Petal.Width       0.8179411 -0.3661259  0.9628654  1.0000000

library(psych)
pairs.panels(iris[-5])
```



```
boxplot(Sepal.Length~Species, iris)
```



```
#własna funkcja do wykonania standaryzacji lub inaczey normalizacji z-score  
stand <-function(x) { (x -mean(x))/(sd(x)) }  
iris_std<- as.data.frame(lapply(iris[,c(1,2,3,4)],stand ))  
iris_std_sp<-data.frame(iris_std, iris$Species)
```

#podzbiory

```
set.seed(123)  
sets <- sample(1:nrow(iris), 0.75 * nrow(iris))  
train_ir<-iris_std_sp[sets,]  
test_ir<-iris_std_sp[-sets,]  
#klasy poszczególnych obserwacji w podzbiorach  
train_ir_class<-iris[sets,5]  
test_ir_class<-iris[-sets,5]
```

```
#pakiet do modeli klasyfikacyjnych kNN  
library(class)
```

#model 1a

```
model_ir3 <- knn(train = train_ir[, 1:4], test = test_ir[,1:4], cl = train_  
ir_class, k=3)  
t_ir3<-table(Species=test_ir_class, Prediction=model_ir3)  
acc_ir3<-mean(test_ir_class == model_ir3)
```

Powtórz dla różnych wartości k. Czasem przyjmuje się optymalną liczbę k jak o pierwiastek z liczby obserwacji w zbiorze uczącym (wartość nieparzysta). Testujemy i szukamy takiego rozwiązania dla którego wybrana metryka (precyz

ja, błąd osiąga optymalną wartość). Można też zastosować V-krotny sprawdzian krzyżowy z pakietu caret.

Dodatkowo oblicz czułość i specyficzność.

kNN model regresyjny dla danych Boston z pakietu MASS

```
set.seed(123)
library(MASS)
library(caret)

#funkcja standaryzująca, zamiast niej może być np. scale()
stand <-function(x) { (x -mean(x))/(sd(x)) }

data("Boston")
sets <- sample(1:nrow(Boston), 0.75 * nrow(Boston))
train<-Boston[sets,]
train_std<- as.data.frame(lapply(train[,c(1:13)],stand ))
train_y<-train[,14]

test<-Boston[-sets,]
test_std<- as.data.frame(lapply(test[,c(1:13)],stand ))
test_y<-test[,14]

knnm1<- knnreg(train_std, train_y, k=5)
str(knnm1)

pred_y = predict(knnm1, data.frame(test_std))

mse = mean((test_y - pred_y)^2)
rmse = caret::RMSE(test_y, pred_y)
R2=(cor(test_y, pred_y))^2

Sprawdź jak wygląda model względem wartości rzeczywistych oraz reszty.
```

Zadanie 2.

Wykonaj modele regresji prostej liniowej, wielorakiej, nieliniowej oraz kNN, wraz z prognozami na 5% obserwacji ze zbioru testowego do predykcji dla danych **real_estate**. Oceń jakość powstałych modeli i porównaj ze sobą.

Zadanie 3.

Dla danych dotyczących **PimaIndian** wykonaj model kNN. Wypróbuj różne wartości k . Wykonaj macierz pomyłek. Oblicz precyzję, czułość i specyficzność. Do zbioru testowego przełącz 30% obserwacji. Dane wymagają preprocessingu, wykonaj analizę danych.