

# **SPRAWOZDANIE**

Zajęcia: Analiza Procesów Ucznienia

Prowadzący: prof. dr hab. Vasyl Martsenyuk

Laboratorium 5

Data 28.04.2023

Temat: "Modelowanie procesów uczenia maszynowego w pakiecie mlr.  
Trenowanie, ocena i porównywanie modeli w pakiecie mlr"

Wariant: 2

Szymon Białek  
Informatyka II stopień  
stacjonarne  
1 semestr,  
Gr.1

**Wszystkie pliki i komendy można obejrzeć pod linkiem:**

<https://github.com/NynyNoo/Analiza-procesow-uczenia/tree/main/lab5>

## **Zadanie1**

### **Polecenie**

Zadanie 1. Zadanie dotyczy konstruowania drzew decyzyjnych oraz reguł klasyfikacyjnych na podstawie zbioru danych (z wykorzystaniem biblioteki MASS lub datasets). Biopsy

### **Wykorzystane komendy oraz wyniki działania programu**

#### **Wczytanie pakietu rpart**

```
library(rpart)
```

#### **Wczytanie danych**

```
data(infert)
```

#### **Podsumowanie kolumn**

```
summary(infert)
```

#### **Podział danych na zestaw treningowy i testowy**

```
set.seed(123)
```

```
train_indices <- sample(1:nrow(infert), nrow(infert) * 0.7)
```

```
train <- infer[train_indices, ]
```

```
test <- infer[-train_indices, ]
```

#### **Tworzenie modelu drzewa decyzyjnego**

```
model <- rpart(case~., data=train, method="class")
```

#### **Wizualizacja drzewa decyzyjnego**

```
plot(model)
```

```
text(model)
```

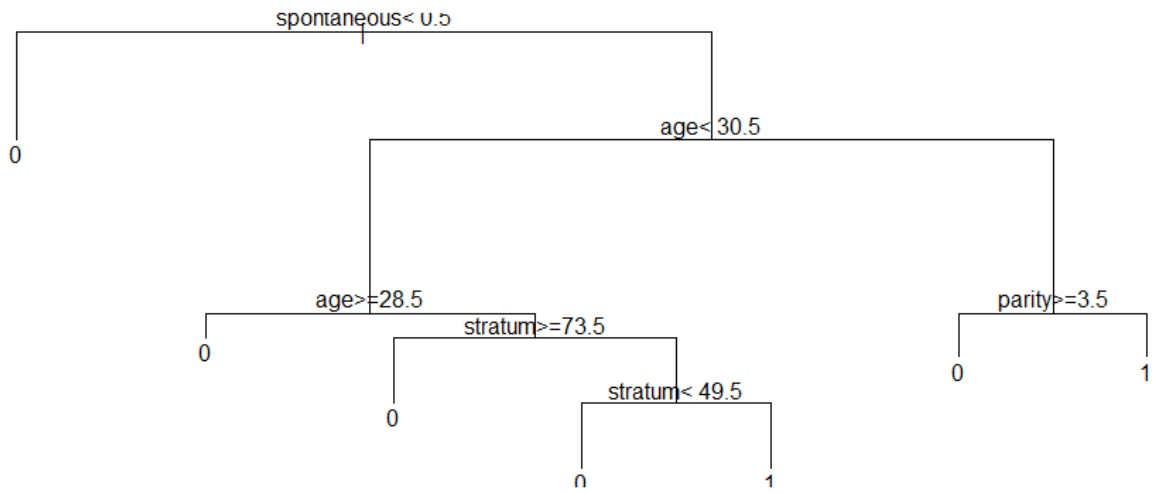
#### **Predykcja na zestawie testowym**

```
predictions <- predict(model, newdata=test, type="class")
```

## Wyniki predykcji

`table(predictions, test$case)`

## Wizualizacja Danych



## Zadanie 2

### Polecenie

Zadanie dotyczy prognozowania oceny klientów (w skali 5-punktowej, Error < 5%) urządzeń RTV AGD, określonych na Zajęciu 1. Rozwiązanie polega na użyciu pakietu mlr. Należy wybrać najlepszą metodę wśród 5 możliwych z punktu widzenia precyzyjności. Wyniki porównywania precyzyjności metod należy przedstawić w postaci graficznej.

### Wykorzystane komendy oraz wyniki działania programu

```
library("rFerts")  
library("randomForestSRC")  
library("mlr")
```

### Wczytaj dane

```
data <- read.csv("macbooki.csv")  
data <- data[3:8]  
data[, 6] <- factor(data[, 6])
```

### Ustaw zadanie

```
task = makeClassifTask(  
  id = deparse(substitute(data)),  
  data,  
  target = "ocena_klientow",  
  weights = NULL,  
  blocking = NULL,  
  coordinates = NULL,  
  positive = NA_character_,  
  fixup.data = "warn",  
  check.data = TRUE  
)
```

### Utwórz listę algorytmów do sprawdzenia

```
lrns <- makeLearners(  
  c(  
    "lda",  
    "rpart",  
    "C50",  
    "rFerts",  
    "h2o.randomForest"  
  ),  
  type = "classif"  
)
```

## Wykonaj benchmark

```
bench <- benchmark(  
  learners = lrns,  
  tasks = task,  
  resamplings = cv5  
)
```

```
Measures: mmce  
=====| 100%  
=====| 100%  
=====| 100%  
=====| 100%  
[Resample] iter 1: 1.0000000  
=====| 100%  
=====| 100%  
=====| 100%  
=====| 100%  
[Resample] iter 2: 1.0000000  
=====| 100%  
=====| 100%  
=====| 100%  
=====| 100%  
[Resample] iter 3: 0.6666667  
=====| 100%  
=====| 100%  
=====| 100%  
=====| 100%  
[Resample] iter 4: 0.6666667  
=====| 100%  
=====| 100%  
=====| 100%  
=====| 100%  
[Resample] iter 5: 0.3333333
```

## Wypisz wynik

bench

```
> bench  
  task.id learner.id mmce.test.mean  
1  data      classif.lda      0.1333333  
2  data      classif.rpart     1.0000000  
3  data      classif.C50       0.6666667  
4  data      classif.rFerns     0.4666667  
5  data      classif.h2o.randomForest 0.7333333  
>
```

## Prezentacja graficzna danych

```
> library(ggplot2)  
> library(dplyr)  
>  
> # Przekształć obiekt BenchmarkResult na data.frame  
> df <- as.data.frame(bench)  
>  
> # Stwórz wykres  
> plot <- ggplot(df, aes(x = learner.id, y = mmce)) +  
+   geom_bar(stat = "identity", fill = "steelblue") +  
+   labs(title = "Średni błąd klasyfikacji",
```

```
+ x = "Algorytm",  
+ y = "Błąd klasyfikacji") +  
+ theme_minimal() +  
+ theme(axis.text.x = element_text(angle = 45, hjust = 1))  
>  
> # Wyświetl wykres  
> print(plot)
```

