

# **SPRAWOZDANIE**

Zajęcia: Analiza Procesów Ucznienia

Prowadzący: prof. dr hab. Vasyl Martsenyuk

## **Laboratorium 5\_2**

Data 28.04.2023r.

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

## **Warient 5**

Jarosław Waliczek  
Informatyka  
II stopień,  
stacjonarne  
1 semestr

## 1. 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 precyzji. Wyniki porównywania precyzji metod należy przedstawić w postaci graficznej.

Wariant 5 – Lodówki turystyczne

## 2. Wprowadzane dane:

Tabela z danymi lodówek turystycznych ze strony RTV AGD

## 3. Wykorzystane komendy:

Konfiguracja bibliotek:

```
1 install.packages("randomForest")
2 install.packages("e1071")
3 install.packages("party")
4 install.packages("mlr")
5 install.packages("rFEnns")
6
7 library(randomForest)
8 library(e1071)
9 library(party)
10 library(mlr)
11 library(rFEnns)
12
```

Ładowanie danych wraz z przygotowaniem Factor-ów:

```
3
4 setwd("C:/Users/jaro9/OneDrive/Desktop/apu/zad5_2")
5 lodowki_turystyczne=read.csv("lodowki.csv")
6 lodowki_turystyczne <- lodowki_turystyczne [1:6]
7 lodowki_turystyczne$nazwa = factor(lodowki_turystyczne$nazwa)
8 lodowki_turystyczne$ocena_klientow = factor(lodowki_turystyczne$ocena_klientow)
9
10 summarizeColumns(lodowki_turystyczne)
```

Operacje na danych:

```
summarizeColumns(lodowki_turystyczne)

rdesc = makeResampleDesc("cv", iters = 10)

task = makeClassifTask(id = deparse(substitute(lodowki_turystyczne)), lodowki_turystyczne, "ocena_klientow",
  weights = NULL, blocking = NULL, coordinates = NULL,
  positive = NA_character_, fixup.data = "warn", check.data = TRUE)
lrns <- makeLearners(c("rpart", "c50", "ctree", "naiveBayes", "randomForest"), type = "classif")

bmr <- benchmark(learners = lrns, tasks = task, rdesc, models = TRUE, measures = list(acc, ber))
p_ = getBMRPredictions(bmr)
```

## 4. Wynik działania:

Kod programu dostępny w repozytorium: <https://github.com/Jaro233/APU.git>

### Finalny benchmark:

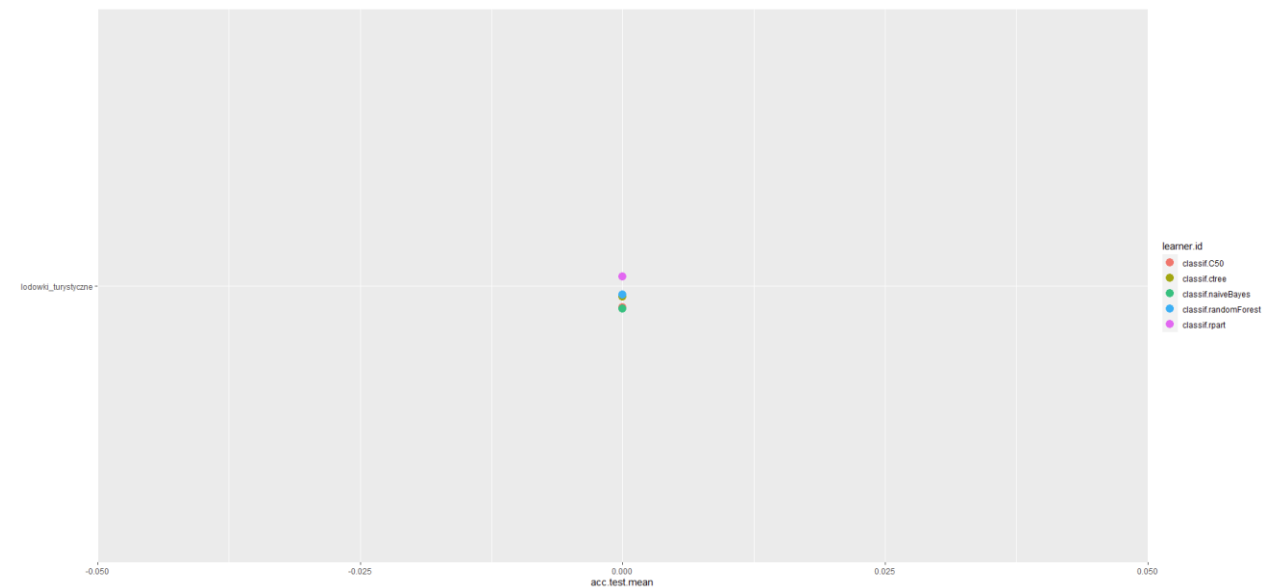
Task: lodowki\_turystyczne, Learner: `classif.randomForest`

Resampling: cross-validation

```
Measures:      acc      ber
[Resample] iter 1: 0.0000000 NaN
[Resample] iter 2: 0.0000000 NaN
[Resample] iter 3: 0.0000000 NaN
[Resample] iter 4: 0.0000000 NaN
[Resample] iter 5: 0.0000000 NaN
[Resample] iter 6: 0.0000000 NaN
[Resample] iter 7: 0.0000000 NaN
[Resample] iter 8: 0.0000000 NaN
[Resample] iter 9: 0.0000000 NaN
[Resample] iter 10: 0.0000000 NaN
```

Aggregated Result: `acc.test.mean=0.0000000,ber.test.mean= NaN`

### Wizualizacja:



## 5. Wnioski:

Dzięki bibliotece mlr możliwe było prognozowanie oceny klientów.