

SPRAWOZDANIE

Zajęcia: Analiza Procesów Ucznia

Prowadzący: prof. dr hab. Vasyl Martsenuk

Laboratorium 5

Data: 20.04.2020r

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

Wariant 1

Zadanie 2

Aleksander Słodczyk

Informatyka II stopień

Stacjonarne

1 semestr

Grupa 2

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 precyzyjności. Wyniki porównywania precyzyjności metod należy przedstawić w postaci graficznej.

2. Wprowadzane dane

Dane to zbiór smartphonów. Został przedstawiony na poniższym obrazku. Na podstawie parametrów smartphona należy określić ocenę klienta. **smartphones** =

	wyswietlacz	pamiec_RAM	pamiec_wbudowana	aparat_foto	ocena_klientow
1	6.7	8	128	64	3
2	6.7	6	128	64	1
3	5.9	4	64	16	2
4	6.4	4	64	25	2
5	6.7	6	128	32	2
6	6.5	4	128	48	3
7	6.1	8	128	16	3
8	6.2	2	32	13	2
9	6.7	8	128	48	4
10	6.7	6	128	12	3
11	5.8	3	32	13	2

3. Wykorzystane komendy:

- **makeResampleDesc()** – Wybiera algorytm próbkujący, w tym przypadku wybrałem algorytm Cross-Validation

```
rdesc = makeResampleDesc(method = "CV", stratify = F)
```

- **makeRegrTask()** – stworzenie zadania dotyczącego regresji, ponieważ chcę przewidzieć wartość na podstawie innych wartości.

```
regr.task = makeRegrTask(id = "Smartphones ocena klientow MSE", data =  
smartphones, target = "ocena_klientow")
```

- **makeLearners()** – Tworzy nowych learnerów, których wykorzystam do stworzenia modelu.

```
regr.learners = makeLearners(c("lm", "randomForest", "nnet", "ctree",
                                "rpart"), type = "regr")
```

- **benchmark()** – porównuje różne algorytmy uczące na podstawie jednego lub więcej tasków, w tym przypadku regr.task

```
regr.comparison = benchmark(learners = regr.learners, tasks = regr.task,
                              resamplings = rdesc)
```

- **getBMRAggrPerformances()**, **getBMRPerformances()**, **plotBMRBoxplots()**, służą do wyświetlania wyników, ostatnia metoda tworzy wykres pudełkowy.

```
getBMRAggrPerformances(regr.comparison)
getBMRPerformances(regr.comparison)
plotBMRBoxplots(regr.comparison)
```

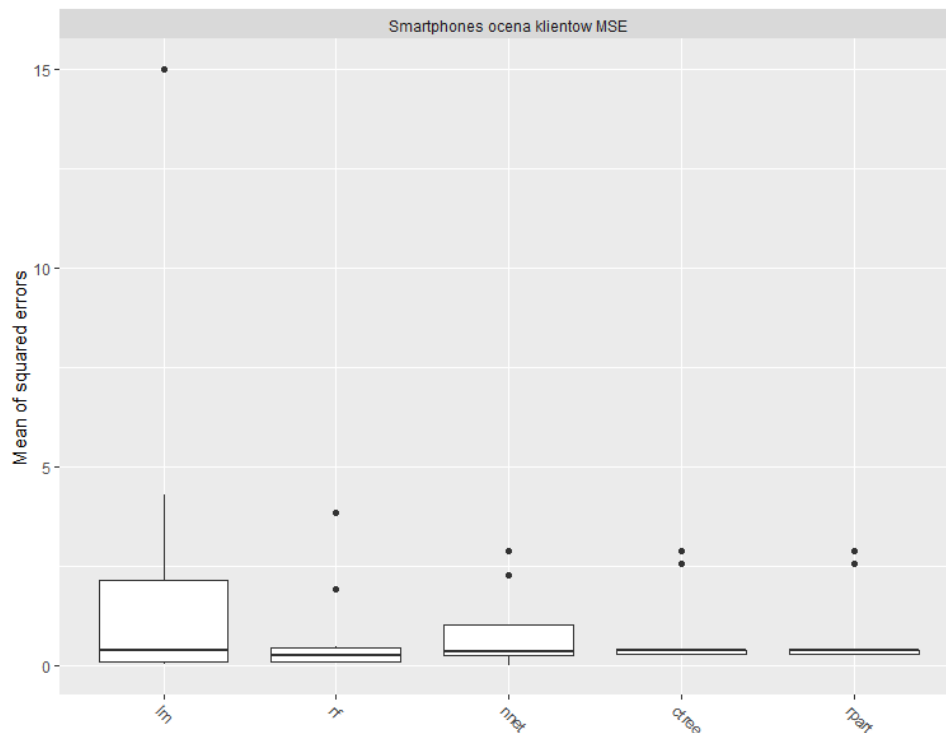
4. Wynik działania

Poniżej przedstawiłem wynik porównania learnerów, których wybrałem do stworzenia modelu.

```
> regr.comparison
```

			task.id	learner.id	mse.test.mean
1	Smartphones	ocena	klientow MSE	regr.lm	2.3690816
2	Smartphones	ocena	klientow MSE	regr.randomForest	0.7623892
3	Smartphones	ocena	klientow MSE	regr.nnet	1.1119990
4	Smartphones	ocena	klientow MSE	regr.ctree	0.7783086
5	Smartphones	ocena	klientow MSE	regr.rpart	0.7783086

Na poniższym obrazku widać wykresy pudełkowe błędu średniokwadratowego poszczególnych learnerów.



Na podstawie błędu średniokwadratowego można stwierdzić, że największy błąd pojawia się przy zastosowaniu zwykłej regresji liniowej (lm). MSE jest największy, jak i wartości odstające. Drugim mniej pewnym algorytmem jest nnet, który ma o połowę mniejszy błąd niż regresja liniowa. Po jednakowych wynikach ctree i rpart można stwierdzić, że te dwa algorytmy są do siebie bardzo podobne.

5. Wnioski:

- Z użytych 5 algorytmów regresji, najpewniejszymi są rpart, ctree lub randomForest.
- Jako najlepszy algorytm wybrałbym randomForest, ponieważ jego błąd średniokwadratowy jest najmniejszy.
- Najprostszy algorytm regresji liniowej dał najgorsze wyniki, czego można się było spodziewać.
- Wszystkie algorytmy mają błąd mniejszy niż 5%

6. Link do githuba:

<https://github.com/allo97/Analiza-procesow-uczenia-Programming-in-R/tree/master/lab5>

Skrypt do algorytmów mlr znajduje się w pliku zadanie/**skrypt3Regr.R**

Wykres pudełkowy znajduje się w pliku

zadanie/**smartphones_ocena_klientow_MSE.png**

Wyniki z konsoli znajdują się w pliku zadanie/**skrypt3Regr_console_log**