

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 1

Aleksander Słodczyk

Informatyka II stopień

Stacjonarne

1 semestr

Grupa 2

1. Polecenie:

Zadanie dotyczy konstruowania drzew decyzyjnych oraz reguł klasyfikacyjnych na podstawie zbioru danych (library(MASS lub datasets)).

Wariant 1 – iris.

2. Wprowadzane dane

Dane to zbiór 150 kwiatów „iris”. Został przedstawiony na poniższym obrazku. Na podstawie długości płatków należy określić gatunek Irysa.

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa
11	5.4	3.7	1.5	0.2	setosa
12	4.8	3.4	1.6	0.2	setosa
13	4.8	3.0	1.4	0.1	setosa

3. Wykorzystane komendy:

Drzewo decyzyjne - rpart

- Komenda **rpart** tworzy model drzewa decyzyjnego. Na wyjściu jest model rpart

```
model <- rpart(Species ~., data = iris)
```

- Komenda **rpart.plot(model)** rysuje model drzewa rpart z wypisanymi poszczególnymi właściwościami drzewa.
- Komenda **predict** przewiduje na podstawie modelu gatunek irysa

```
model %>% predict(newdata, "class")
```

Reguły klasyfikacyjne – C5.0

- Komenda **C5.0** tworzy model drzewa decyzyjnego. Na wyjściu jest model C5.0

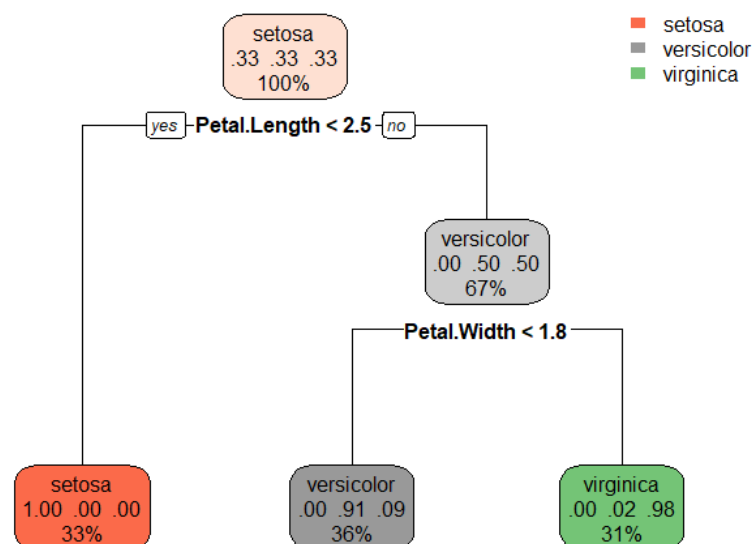
```
mod1 <- C5.0(Species ~ ., data = iris, rules = TRUE)
```

- Komenda **plot(model)** rysuje model drzewa C5.0 z wypisanymi poszczególnymi wartościami gałęzi drzewa.
- Komenda **summary(model)** tworzy podsumowanie wyników drzewa

4. Wynik działania

Drzewo decyzyjne - rpart

Dla drzewa decyzyjnego rpart otrzymałem następujący model drzewa:



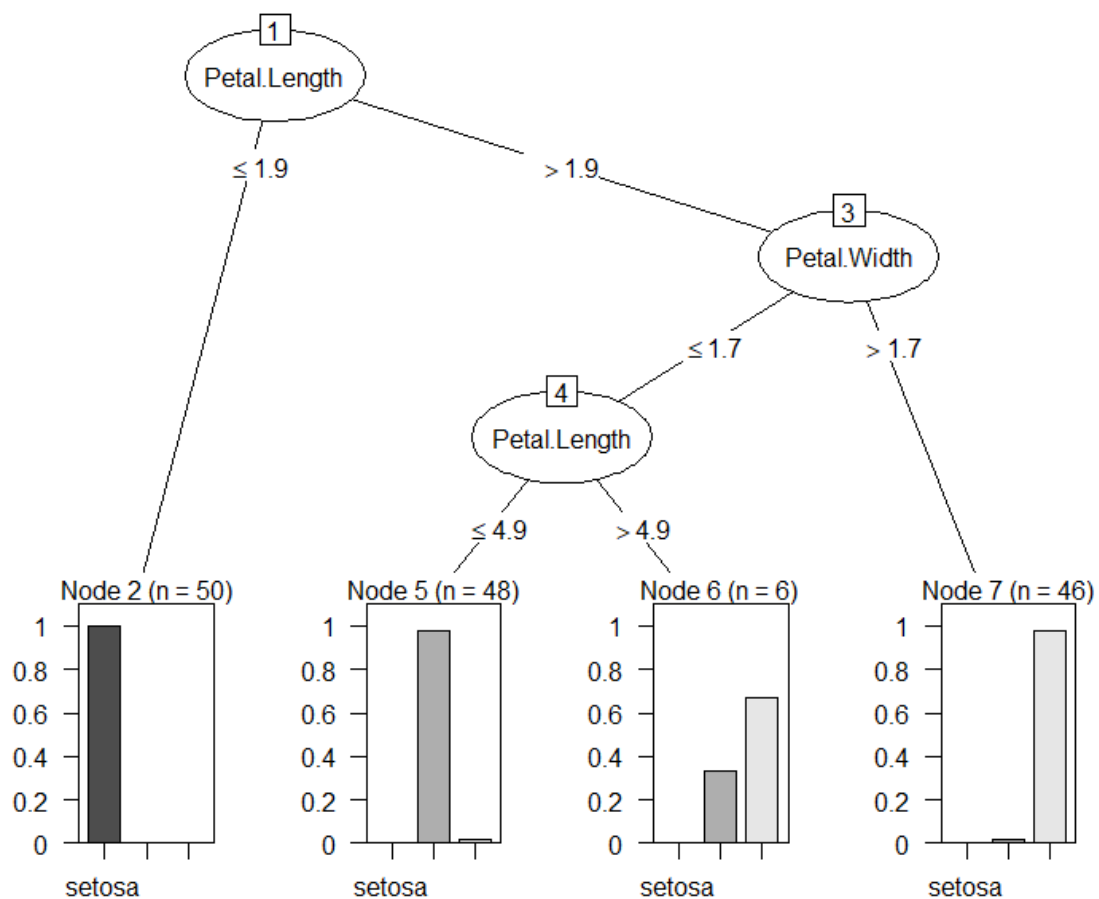
Po utworzeniu przykładowego kwiatu, na podstawie modelu wybrany został gatunek irysa *virginica*, co odpowiada modelowi drzewa decyzyjnego.

```
> newdata <- data.frame(  
+   Sepal.Length = 6.5, Sepal.Width = 3.0,  
+   Petal.Length = 5.2, Petal.Width = 2.0  
+ )  
> model %>% predict(newdata, "class")  
1  
virginica  
Levels: setosa versicolor virginica
```

Z drzewa wynika, że najbardziej znaczące są zmienne Petal.Length i Petal.Width, na podstawie których da się określić już gatunek irysa.

Reguły klasyfikacyjne – C5.0

Dla reguł klasyfikacyjnych otrzymałem następujący model drzewa:



Po wykonaniu metody C5.0 oraz summary dostałem następujące wyniki:

```
Call:
C5.0.formula(formula = Species ~ ., data = iris, rules = TRUE)

C5.0 [Release 2.07 GPL Edition]          Fri May 08 18:01:03 2020
-----

Class specified by attribute `outcome'

Read 150 cases (5 attributes) from undefined.data

Rules:

Rule 1: (50, lift 2.9)
      Petal.Length <= 1.9
      -> class setosa [0.981]

Rule 2: (48/1, lift 2.9)
      Petal.Length > 1.9
      Petal.Length <= 4.9
      Petal.Width <= 1.7
      -> class versicolor [0.960]

Rule 3: (46/1, lift 2.9)
      Petal.Width > 1.7
      -> class virginica [0.958]

Rule 4: (46/2, lift 2.8)
      Petal.Length > 4.9
      -> class virginica [0.938]

Default class: setosa

Evaluation on training data (150 cases):

      Rules
-----
      No      Errors
      4      4 ( 2.7%)  <<

      (a)  (b)  (c)  <-classified as
      ---  ---  ---  ---
      50
           47    3
           1    49
              (a): class setosa
              (b): class versicolor
              (c): class virginica

Attribute usage:

96.00% Petal.Length
62.67% Petal.Width
```

Dla reguł klasyfikacyjnych również okazuje się, że największy wpływ na gatunek irysa mają zmienne odpowiadające za szerokość i długość płatk irysa czyli Petal.Width i Petal.Length. Różnią się tu jednak wartościami oraz jest większy podział drzewa. Błąd w tym przypadku wyniósł 2.7%.

5. Wnioski:

- Oba drzewa przedstawiają podobny model oceny gatunku irysa.
- Reguły klasyfikacyjne zawierają więcej rozgałęzień, więc są przez to dokładniejsze

6. Link do githuba:

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

Skrypty do drzewa decyzyjnego i reguł klasyfikacyjnych znajdują się odpowiednio w pliku zadanie/**skrypt1.R** i zadanie/**skrypt2.R**

Obraz sieci znajduje się w pliku zadanie/ **DrzewoDecyzyjneC50.png** i zadanie/**DrzewoDecyzyjneRpart.png**

Wyniki z consoli znajdują się w pliku zadanie/**skrypt1_console_log** i zadanie/**skrypt2_console_log**