

# SI - SPRAWOZDANIE LAB NR 5

Maciej Budzowski, dzienne, grupa L5

09/05/2020

Link do pliku .tex na platformie OverLeaf: [OVERLEAF](#)

## Użyty zbiór danych

Do wykonania zadania użyłem zbioru **seeds**, linki do źródeł zamieszczam poniżej:

[DANE \(PLIK\)](#) (link prowadzi do oryginalnego pliku)

[DANE \(STRONA\)](#) (link prowadzi do strony datasetu)

**Opis/temat zbioru:** Pomiary właściwości geometrycznych ziaren należących do trzech różnych odmian pszenicy. Zastosowano miękką technikę rentgenowską i pakiet ZIARNA do skonstruowania wszystkich siedmiu atrybutów o wartościach rzeczywistych.

### Autorzy:

Małgorzata Charytanowicz, Jerzy Niewczas

Institute of Mathematics and Computer Science,

The John Paul II Catholic University of Lublin, Konstantynów 1 H

PL 20-708 Lublin, Poland

### Parametry użytego zbioru:

- liczba obserwacji - 210
- liczba cech - 7 (+1 - klasa)
- liczba klas - 3 (1, 2, 3 - odpowiednio "Kama", "Rosa", "Canadian")
- liczba elementów na klasę - "po równo" - 70 elementów dla każdej z klas

### Rodzaje cech w zbiorze:

- area (A) (type - real)
- perimeter (P) (type - real)
- compactness ( $C = 4 \cdot \pi \cdot A / P^2$ ) (type - real)
- length of kernel (type - real)
- width of kernel (type - real)
- asymmetry coefficient (type - real)
- length of kernel groove (type - real)

- class (nazwa klasy)

## Przygotowanie danych

---

W ramach przygotowania danych do pracy musiałem podmienić nazwy klas (w oryginalnym pliku zostały użyte liczby - 1, 2, 3), użyłem w tym celu VSCode oraz narzędzia regex. W skrypcie jako komentarz również pojawia się informacja na ten temat.

Link do oryginalnych danych: [ORYGINALNE](#)

Link do przygotowanych danych: [PRZYGOTOWANE](#)

## Kod

---

Link do projektu ze skryptem: [GITHUB](#)

Klasyfikacji dokonałem przy użyciu skryptu języka R. Opiera się on na kodzie z odbytych ćwiczeń, jednak na potrzeby zadania zaszła potrzeba modyfikacji.

Część właściwego treningu i wypisywania wyników zamknąłem w funkcji, która zostaje wywołana w pętli wraz z odpowiednią wartością **neuralCount** (ilości neuronów). Pozwoliło to na automatyzację procesu. (Podczas prób zauważyłem, że zwiększenie ilości neuronów znacząco wydłuża czas skryptu.)

Pozwoliłem sobie również na zwiększenie parametru **threshold** na 0.1 (domyślna wartość 0.01)

```
nn <- neuralnet(Kama + Rosa + Canadian ~
                area + perimeter + compactness +
                kernel_length + kernel_width +
                asymmetry_coefficient + length_kernel_groove ,
                data = neuralNet_seedsTrain ,
                hidden = c(neuralCount) ,
                stepmax = 1e+06,
```

threshold = 0.1

)

## Wyniki

---

\*Zbiory treningowe i walidacyjne - każdy z nich stanowi połowę całości rekordów.

### Liczba neuronów: 5

prediction	Canadian	Kama	Rosa
Canadian	<b>36</b>	1	0
Kama	4	<b>30</b>	1
Rosa	0	0	<b>33</b>

Tablica 1: *macierz pomyłek dla 5 neuronów w warstwie ukrytej*

prediction	TPR (True Positive Rate)
Canadian	36/40 (90%)
Kama	30/31 (ok. 97%)
Rosa	33/34 (ok. 97%)

Tablica 2: *True Positive Rate dla 5 neuronów w warstwie ukrytej*

**Recognition Rate: 0.9428571**

---

### Liczba neuronów: 10

prediction	Canadian	Kama	Rosa
Canadian	<b>35</b>	1	0
Kama	5	<b>30</b>	0
Rosa	0	0	<b>34</b>

Tablica 3: *macierz pomyłek dla 10 neuronów w warstwie ukrytej*

prediction	TPR (True Positive Rate)
Canadian	35/40 (87,5%)
Kama	30/31 (ok. 97%)
Rosa	34/34 (100%)

Tablica 4: *True Positive Rate dla 10 neuronów w warstwie ukrytej*

**Recognition Rate: 0.9428571**

---

**Liczba neuronów: 15**

prediction	Canadian	Kama	Rosa
Canadian	<b>36</b>	1	0
Kama	4	<b>30</b>	0
Rosa	0	0	<b>34</b>

Tablica 5: *macierz pomyłek dla 15 neuronów w warstwie ukrytej*

prediction	TPR (True Positive Rate)
Canadian	36/40 (90%)
Kama	30/31 (ok. 97%)
Rosa	34/34 (100%)

Tablica 6: *True Positive Rate dla 15 neuronów w warstwie ukrytej*

**Recognition Rate: 0.952381**

---

**Liczba neuronów: 20**

prediction	Canadian	Kama	Rosa
Canadian	<b>36</b>	1	0
Kama	4	<b>30</b>	1
Rosa	0	0	<b>34</b>

Tablica 7: *macierz pomyłek dla 20 neuronów w warstwie ukrytej*

prediction	TPR (True Positive Rate)
Canadian	36/40 (90%)
Kama	30/31 (ok. 97%)
Rosa	34/34 (100%)

Tablica 8: *True Positive Rate dla 20 neuronów w warstwie ukrytej*

**Recognition Rate: 0.952381**

**Liczba neuronów: 25**

---

prediction	Canadian	Kama	Rosa
Canadian	<b>35</b>	1	0
Kama	5	<b>29</b>	1
Rosa	0	1	<b>34</b>

Tablica 9: *macierz pomyłek dla 25 neuronów w warstwie ukrytej*

prediction	TPR (True Positive Rate)
Canadian	35/40 (87,5%)
Kama	29/31 (ok. 93,5%)
Rosa	34/34 (100%)

Tablica 10: *True Positive Rate dla 25 neuronów w warstwie ukrytej*

**Recognition Rate: 0.9333333**

---

## Wnioski

Czas trenowania wzrastał wraz z ilością neuronów w warstwie ukrytej, a malał wraz ze wzrostem parametru **threshold**.

Wyniki sklasyfikowania dla każdej próby okazały się wysokie i stosunkowo bardzo do siebie zbliżone.

Najwyższy procent prawidłowo sklasyfikowanych obiektów został osiągnięty przy liczbie 15 i 20 neuronów w warstwie ukrytej (ok. 95%).

Procent prawidłowo sklasyfikowanych obiektów wzrastał do liczby 15 neuronów, spadek został zauważony na etapie 25 neuronów.

Procent prawidłowo sklasyfikowanych obiektów osiągnięty przy liczbie 25 neuronów był niższy niż w przy liczbie 5, 10, 15 i 20 neuronów.

Na podstawie osiągniętych wyników wnioskuję, że ilość neuronów w wybranym przeze mnie pakiecie danych nie ma znaczącego wpływu na osiąganе wyniki w klasyfikacji obiektów.

## Oryginalne wyniki z konsoli IDE RStudio

---

```
[1] "Neural_network_number_1—number_of_neurals : 5"
```

```
[1] "Confusion_matrix :"
```

```
prediction Canadian Kama Rosa
```

```
Canadian      36      1      0
```

```
Kama          4     30      1
```

```
Rosa          0      0     33
```

```
[1] "Recognition_rate :"
```

```
[1] 0.9428571
```

```
[1] "Neural_network_number_2—number_of_neurals : 10"
```

```
[1] "Confusion_matrix :"
```

prediction Canadian Kama Rosa

Canadian	35	1	0
Kama	5	30	0
Rosa	0	0	34

[1] "Recognition\_rate:"

[1] 0.9428571

[1] "Neural\_network\_number\_3--number\_of\_neurals:\_15"

[1] "Confusion\_matrix:"

prediction Canadian Kama Rosa

Canadian	36	1	0
Kama	4	30	0
Rosa	0	0	34

[1] "Recognition\_rate:"

[1] 0.952381

[1] "Neural\_network\_number\_4--number\_of\_neurals:\_20"

[1] "Confusion\_matrix:"

prediction Canadian Kama Rosa

Canadian	36	1	0
Kama	4	30	0
Rosa	0	0	34

[1] "Recognition\_rate:"

[1] 0.952381

[1] "Neural\_network\_number\_5--number\_of\_neurals:\_25"

[1] "Confusion\_matrix:"

prediction Canadian Kama Rosa

Canadian	35	1	0
Kama	5	29	0
Rosa	0	1	34

[1] "Recognition\_rate:"

[1] 0.9333333