

Data 11.05.2023

Temat:

"Użycie sztucznych sieci neuronowych"

Wariant: 2

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

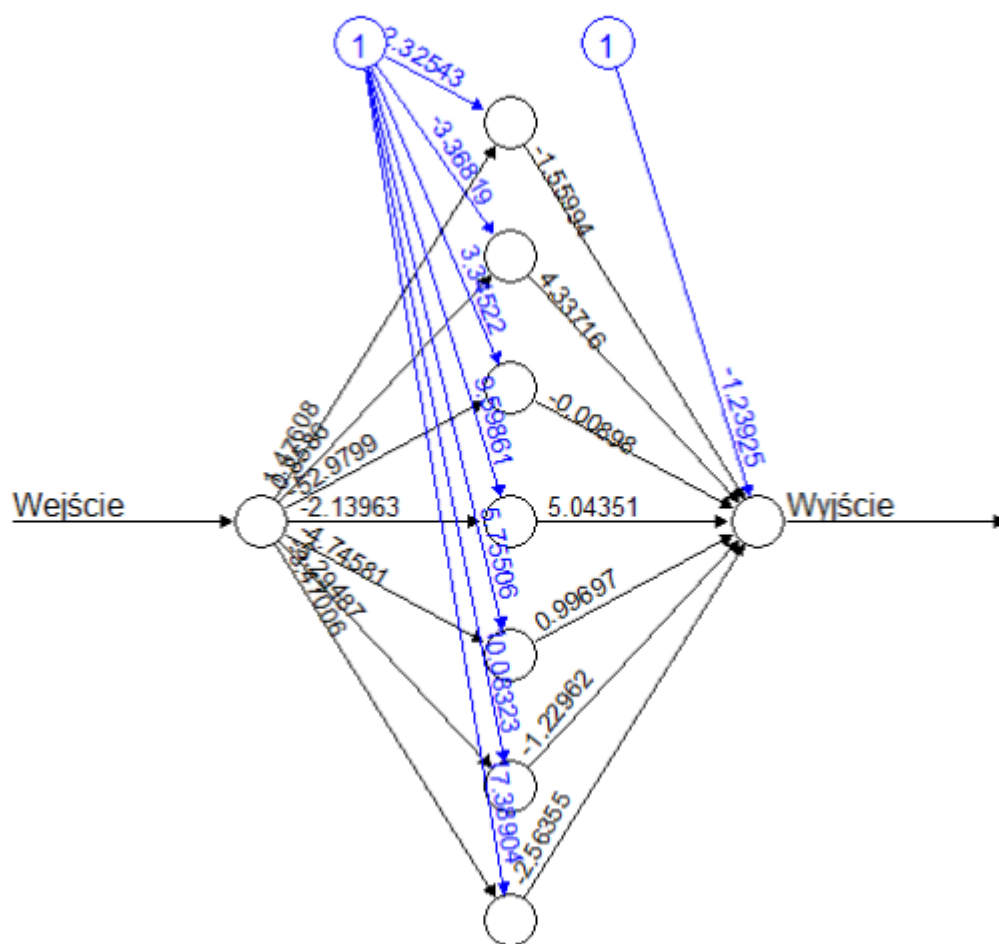
Strona do repozytorium GitHub:

<https://github.com/NynyNoo/Analiza-procesow-uczenia/tree/main/lab3>

Zadanie 1

dotyczy modelowania funkcji matematycznych za pomocą sztucznej sieci neuronowej używając paczkę neuralnet. Rozważamy zmienną niezależną x . Celem jest uzyskanie sieci neuronowej (zmieniając zarówno ilość warstw ukrytych jak i ilość neuronów) spełniającej warunek $\text{Error} < 0.01$.

```
install.packages("neuralnet")
library(neuralnet)
input <- as.data.frame(runif(100,min=0,2*pi))
output <- abs(cos(input))^(abs(sin(input)))
trainingdata <- cbind(input,output)
colnames(trainingdata)<-c("Wejście","Wyjście")
net.sqrt <- neuralnet(Wyjście~Wejście,trainingdata,hidden=7,threshold=0.01,stepmax=1e7)
print(net.sqrt)
plot(net.sqrt,rep="best")
testdata <- as.data.frame(runif(300,min=0,max=20))
net.results <- compute(net.sqrt, testdata)
print(net.results$net.result)
cleanoutput <- cbind(testdata,log(testdata^2),as.data.frame((net.results$net.result)))
colnames(cleanoutput) <- c("Wejście", "Oczekiwane Wyjście", "Wyjście sieci neuronowej")
print(cleanoutput)
```



Error: 0.035586 Steps: 44989

Zadanie 2. To zadanie dotyczy prognozowania cen urządzeń RTV AGD (błąd ≤ 100 PLN), określonych w zadaniu 1. Wykorzystując metodę sztucznych sieci neuronowych, należy opracować plik w języku R z wykorzystaniem pakietu neuralnet. Sprawozdania w postaci pliku R, obrazu sieci neuronowej oraz wyników z konsoli (dowolny plik tekstowy) należy zachować w zdalnym repozytorium (np. GitHub).

dla danych z pierwszych zajęć:

```
library(neuralnet)
```

```
library(ggplot2)
```

```
setwd("D:/MGR/APU/lab3/lab3")
```

```
dataset <- read.csv("macbooki.csv")
```

```
ocena_klientow <- dataset[["ocena_klientow"]]
```

```
cena <- dataset[["cena"]]
```

```
#przygotowanie danych
```

```
compare.trainingdata <- cbind(ocena_klientow, cena)
```

```
#scalowanie ocen
```

```
scaled.ocena_klientow <- as.data.frame(scale(ocena_klientow))
```

```
#trenowanie
```

```
trainingdata <- cbind(scaled.ocena_klientow, cena)
```

```
colnames(trainingdata) <- c("ocena_klientow", "cena")
```

```
 #(error  $\leq 100$  z l)
```

```
net.price <- neuralnet(cena~ocena_klientow, trainingdata, hidden=c(5,3,3), threshold<-100,
```

```
lifesign <- "full")
```

```
plot(net.price)
```

```
testdata <- data.frame(c(20,130))
```

```
scaled.testdata <- as.data.frame(scale(testdata))
```

```
#prognozowanie
```

```
net.results <- compute(net.price, scaled.testdata)
```

```
fixed_cena <- cbind(testdata, as.data.frame(net.results$net.result))
```

```
colnames(fixed_cena) <- c("ocena_klientow", "cena")
```

```
print(fixed_cena)
```

```
"", "nazwa", "ekran", "pamiec_RAM", "dysk", "cena", "liczba_opinii", "ocena_klientow", "status_opinii"
"1", "Macbook Apple 1", "13.3", "8", "256", 7000, 100, "0", "50-100 opinii"
"2", "Macbook Apple 2", "13.3", "8", "512", 9000, 150, "0.5", "więcej 100 opinii"
"3", "Macbook Apple 3", "13.3", "8", "1000", 11000, 200, "1", "więcej 100 opinii"
"4", "Macbook Apple 4", "16", "16", "512", 12000, 300, "1.5", "więcej 100 opinii"
"5", "Macbook Apple 5", "16", "16", "1000", 14000, 250, "2", "więcej 100 opinii"
"6", "Macbook Apple 6", "16", "16", "2000", 16000, 400, "2.5", "więcej 100 opinii"
"7", "Macbook Apple 7", "14", "8", "256", 9000, 50, "3", "50-100 opinii"
"8", "Macbook Apple 8", "14", "8", "512", 10000, 80, "3.5", "50-100 opinii"
"9", "Macbook Apple 9", "15.4", "16", "512", 13000, 120, "4", "więcej 100 opinii"
"10", "Macbook Apple 10", "15.4", "16", "1000", 15000, 180, "4.5", "więcej 100 opinii"
"11", "Macbook Pro M1X", "16 cali", "16 GB", "1 TB", 9999, 0, "5", "nie ma"
"12", "Macbook Air", "13,3\"", "8 GB", "512 GB", 9999, 20, "4.5", "mniej 50 opinii"
"13", "Macbook Pro", "13,3\"", "16 GB", "1 TB", 12999, 25, "4", "mniej 50 opinii"
"14", "Macbook Pro 16", "16\"", "16 GB", "512 GB", 18999, 10, "5", "mniej 50 opinii"
"15", "Macbook Pro M1", "13,3\"", "8 GB", "512 GB", 10999, 15, "3.5", "mniej 50 opinii"
```

powstał taki graf:

