**SPRAWOZDANIE**

Zajęcia: Analiza Procesów Uczenia

Prowadzący: prof. dr hab. Vasyl Martsenyuk

**Laboratorium 4**

**Temat:** "Podatawy języka R"

**Wariant 1**

Maciej Wojcieszyk

Informatyka II stopień,

stacjonarne (zaoczne),

1 semestr,

Gr.:1

1. **Polecenie:**

Zadanie dotyczy modelowania funkcji matematycznych za

pomoca ̨ sztucznej sieci neuronowej używajać paczkę neuralnet. Rozwaźamy ́miennę niezależną x. Celem jest uzyskanie sieci neuronowej (zmieniając zarówno ilość warstw ukrytych jak i ilość neuronów) wypełniającej warunek Error < 0.01.

1. **Wprowadzane dane:**

f(x) = x ^3 + 2 ∗ x, x ∈ [1; 100]

1. **Wykorzystane komendy:**

library("neuralnet")

myFunction <- function(x) { return(x^3 +(2\*x))}

traininginput <- as.data.frame(runif(100, min = 0, max = 10))

trainingoutput <- myFunction(traininginput)

trainingdata <- cbind(traininginput, trainingoutput)

colnames(trainingdata) <- c("Input", "Output")

net.function <- neuralnet(Output ~ Input, trainingdata,

hidden = 100, threshold = 0.01)

plot(net.function)

testdata <- as.data.frame((1:100))

net.results <- compute(net.function, testdata)

ls(net.results)

print(net.results$net.result)

cleanoutput <- cbind(testdata, myFunction(testdata),

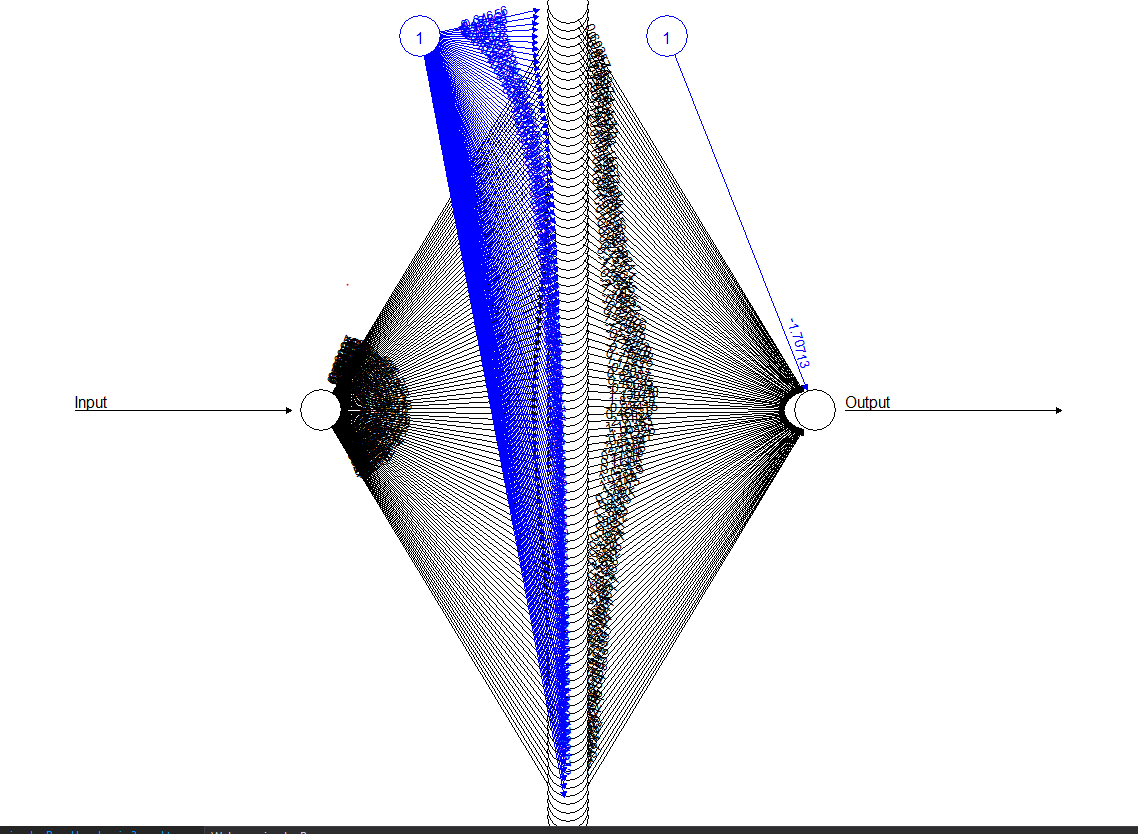
as.data.frame(net.results$net.result))

colnames(cleanoutput) <- c("Input", "Expected Output", "Neural Net

Output")

print(cleanoutput)

1. **Wyniki działań:**



1. **Wnioski:**

Dla danych treningowych – 100 próbek z przedziały 1 do 10, udało się nauczyć sieć neuronową zawierającą 1 warstwę ukrytą z ilością 100 neuronów. Jak widać na załączonych zrzutach ekranu błąd jest minimalny – w granicach błędu założonego przy trenowaniu sieci.