Introduction to Machine Learning

Lab 6

Rasmus Holm

2016-12-15

Contents

Assignment 1

In this assignment I have used a neural network with one hidden layer of 10 units to fit a sinus functions. In order to find the optimal threshold at which the gradient descent will stop as to avoid overfitting I divided the data into train and validation sets each 50% of the data of 50 observations. In figure 1 we can see that the optimal threshold was found at 0.004 which I used to train a new neural network on the complete data set.

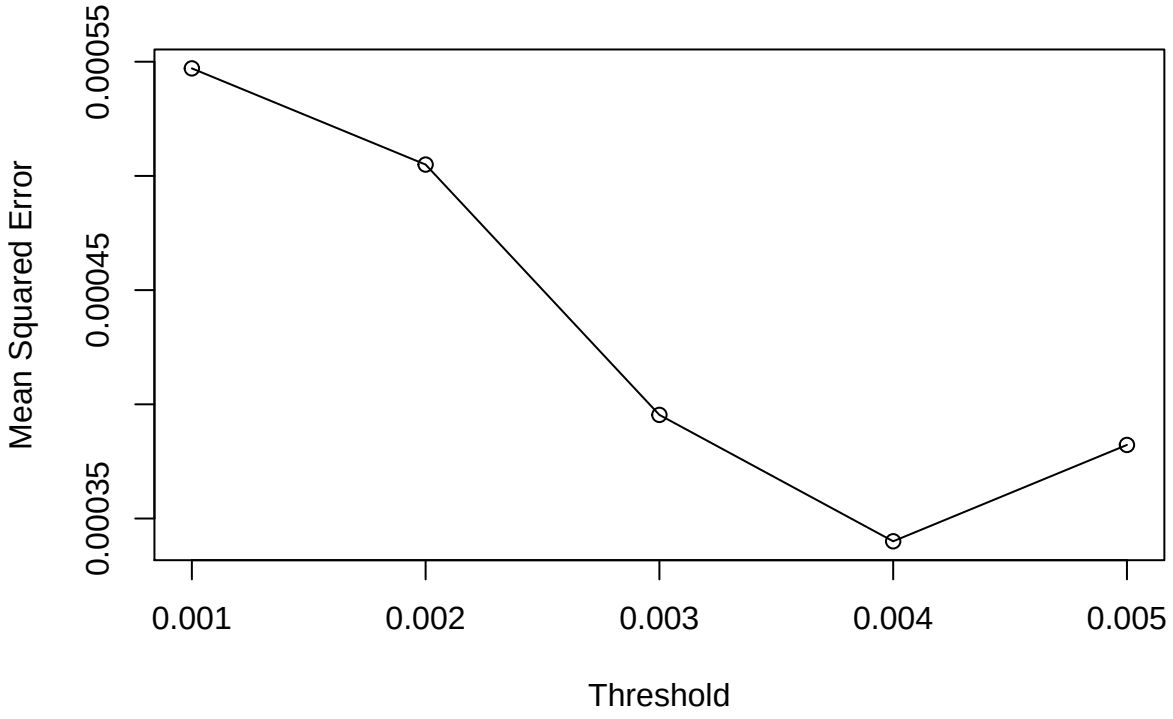

Figure 1: Mean squared error as the threshold increases.

Figure 2 shows the final neural network.

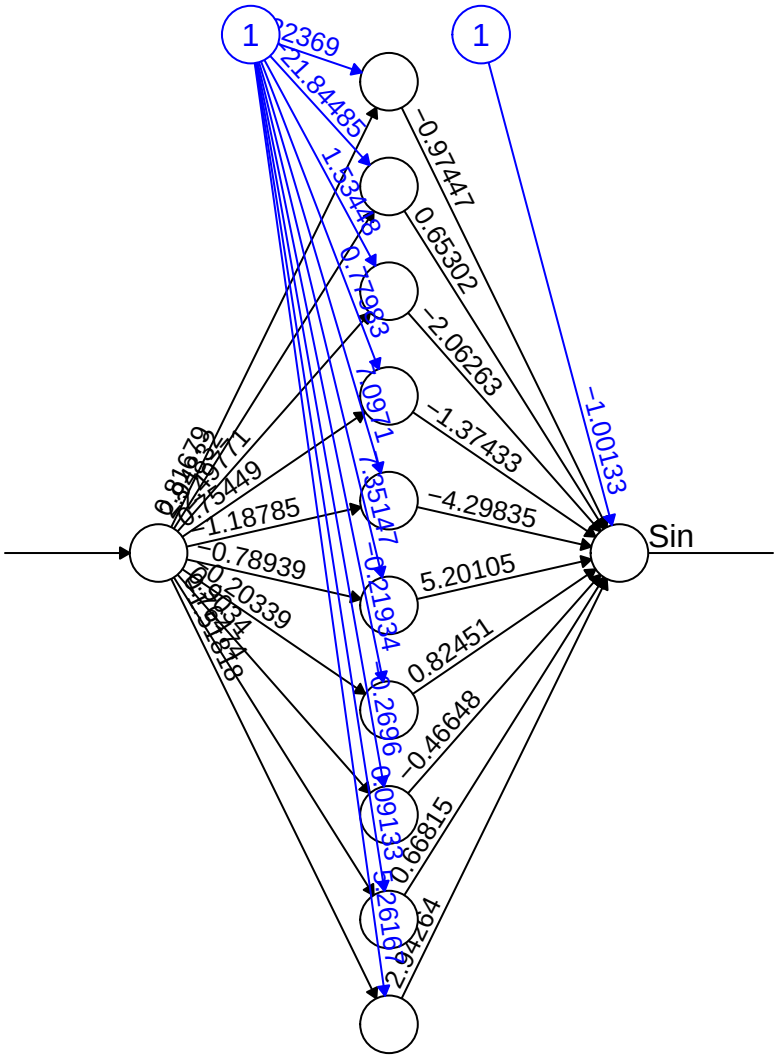

Figure 2: The final neural network.

In figure 3 we can see that the fit is very good where the red points are the predictions and the black points are the true underlying data. This result follows from the theory that a neural network with one hidden layer

can fit any distribution given the right amount of data and hidden units.

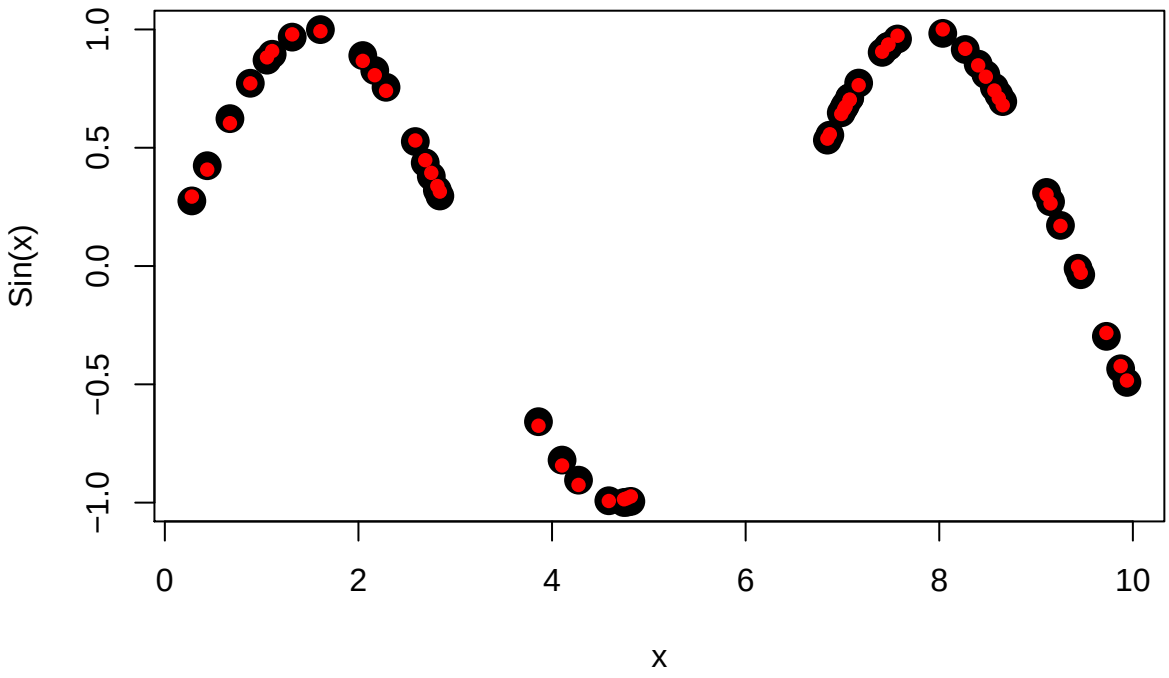

Figure 3: Predictions in red vs the true data in black.

Appendix

Code for Assignment 1

```
library(neuralnet)
set.seed(1234567890)

n <- 50
m <- 10

Var <- runif(n, 0, 10)
data <- data.frame(Var, Sin=sin(Var))

train <- data[1:(n / 2),] # Training
validation <- data[(n / 2 + 1):n,] # Validation

## Random initializaiton of the weights in the interval [-1, 1]
winit <- runif(3 * m + 1, -1, 1)

errors <- rep(0, 10)

for(it in 1:10) {
    nn <- neuralnet(Sin ~ Var, data=train, threshold=it / 1000,
                    hidden=10, startweights=winit)
    predicted <- compute(nn, validation$Var)$net.result
    errors[it] <- sum((predicted - validation$Sin)^2) / nrow(validation)

    if (it > 1 && errors[it] > errors[it - 1]) break
}

optimal_threshold <- it / 1000

nn <- neuralnet(Sin ~ Var, data=data,
                threshold=optimal_threshold,
                hidden=10, startweights=winit)

predictions <- prediction(nn)$rep1
plot(1:it / 1000, errors[1:it], type="o",
     xlab="Threshold", ylab="Mean Squared Error")
plot(nn, information=FALSE)
plot(data, type="p", pch=16, cex=2, xlab="x", ylab="Sin(x)")
points(predictions, col="red", type="p", pch=16, cex=1)
```