Introduction to Machine Learning

Lab 6

Anton Persson, Emil Klasson Svensson, Mattias Karlsson, Rasmus Holm

2016-12-16

Contents

Assignment 1

In this assignment we used a neural network with one hidden layer consisting of 10 units to learn the trigonometric sine function from 25 observations without any added noise in the interval $[0, 10]$. The threshold is defined as i / 1000, where i = 1, . . ., 10, and we used an additional set of 25 observations for validation to stop the iterations by computing the mean squared error (MSE). The optimal threshold was found to be 4 / 1000 which can be seen in figure 1.

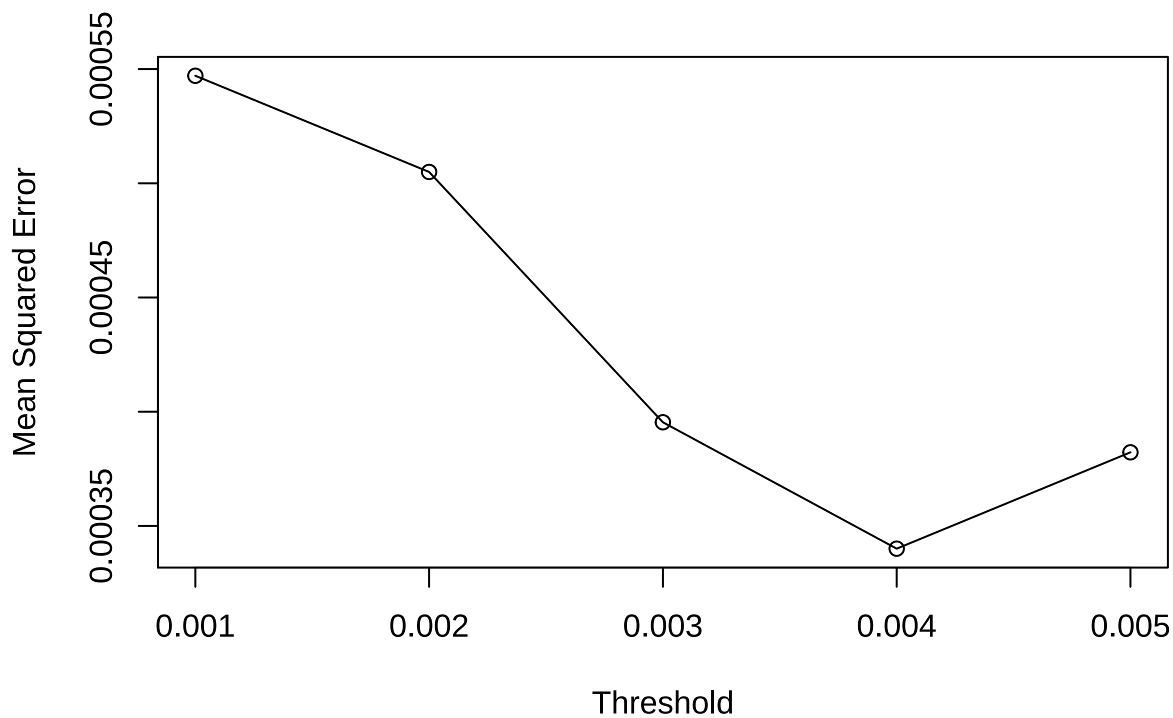

Figure 1: Mean squared error as the threshold increases.

With the optimal threshold we trained a new neural network on the complete data set and the final network is shown in figure 2.

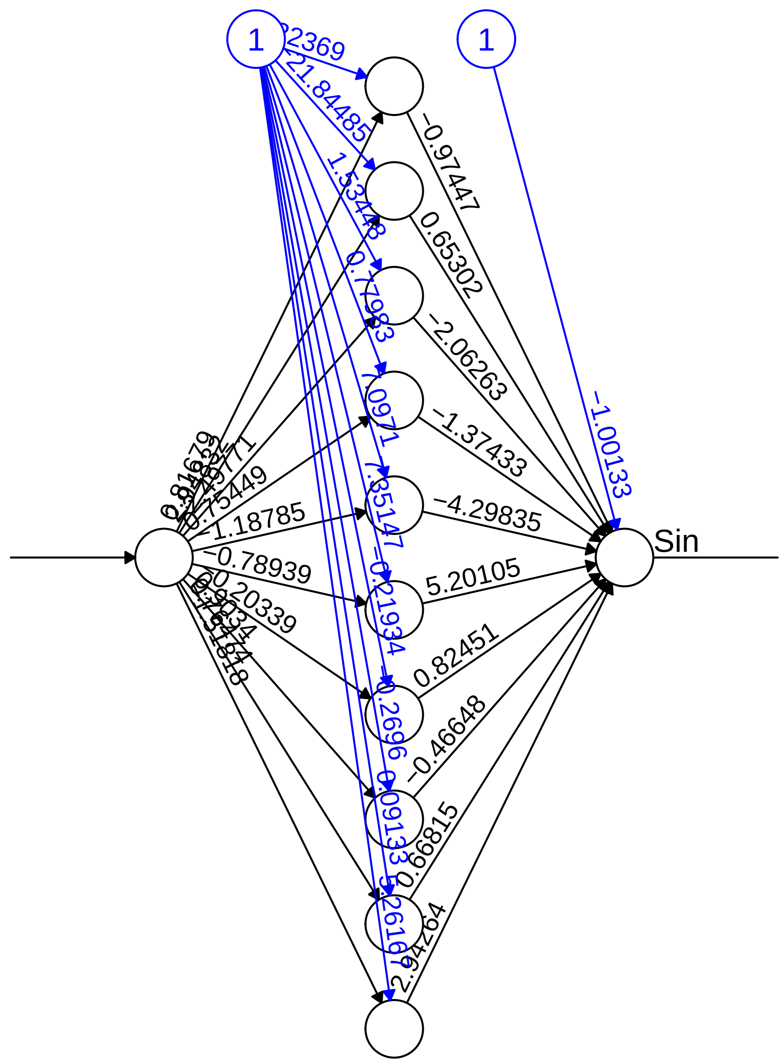

Figure 2: The final neural network.

In order to see how well the neural network could mimic the sine function we plotted the predicted values against the observed values which can be seen in figure 3. We can see that it follow the observed data very well as expected and it follows from the theory that a neural network with a single hidden layer can fit any function arbitrarily well given the right amount of data, hidden units, and iterations.

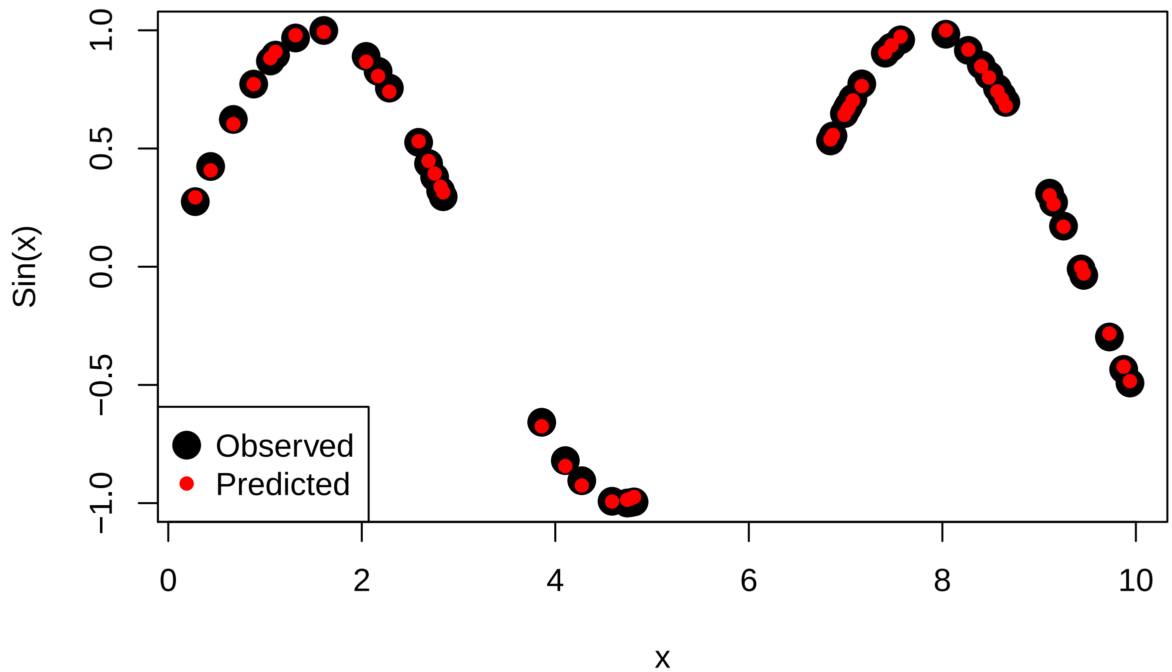

Figure 3: Predictions in red vs the true data in black.

Appendix

Code for Assignment 1

```r
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)
legend("bottomleft", legend=c("Observed", "Predicted"),
       col=c("Black", "Red"), pch=c(16, 16),
       pt.cex=c(2, 1))
```

Contributions

We divided the work into two parts and discussed/compiled the results in pairs. Then we all discussed our findings together as a whole group and checked that everyone had similar/understood the results.