Lab3 Assignment 3

Shwetha

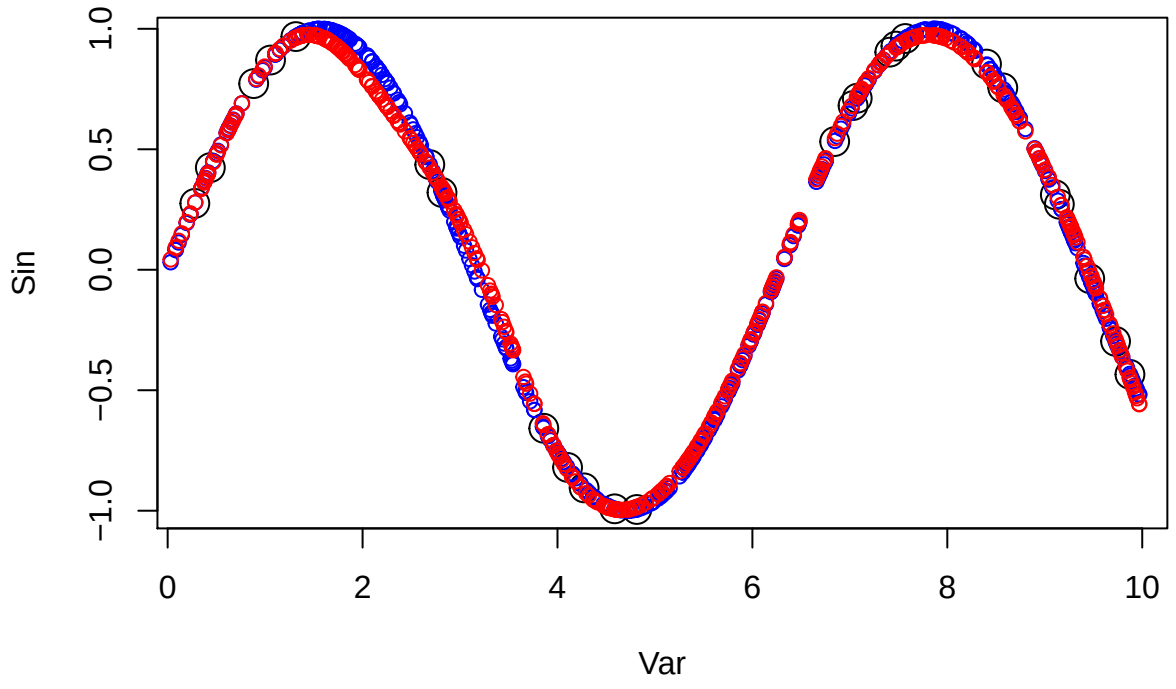
12/3/2020

3. NEURAL NETWORKS

1.

Fitting neural network to learn the trigonometric sine function

```
winit = runif(46,-0.5,0.5)
nn = neuralnet(Sin~Var,tr,hidden = 15,startweights = winit)
train_predict = predict(nn,tr)
test_predict = predict(nn,te)
mse = mean(((te[,2]-test_predict)^2)/475)
```

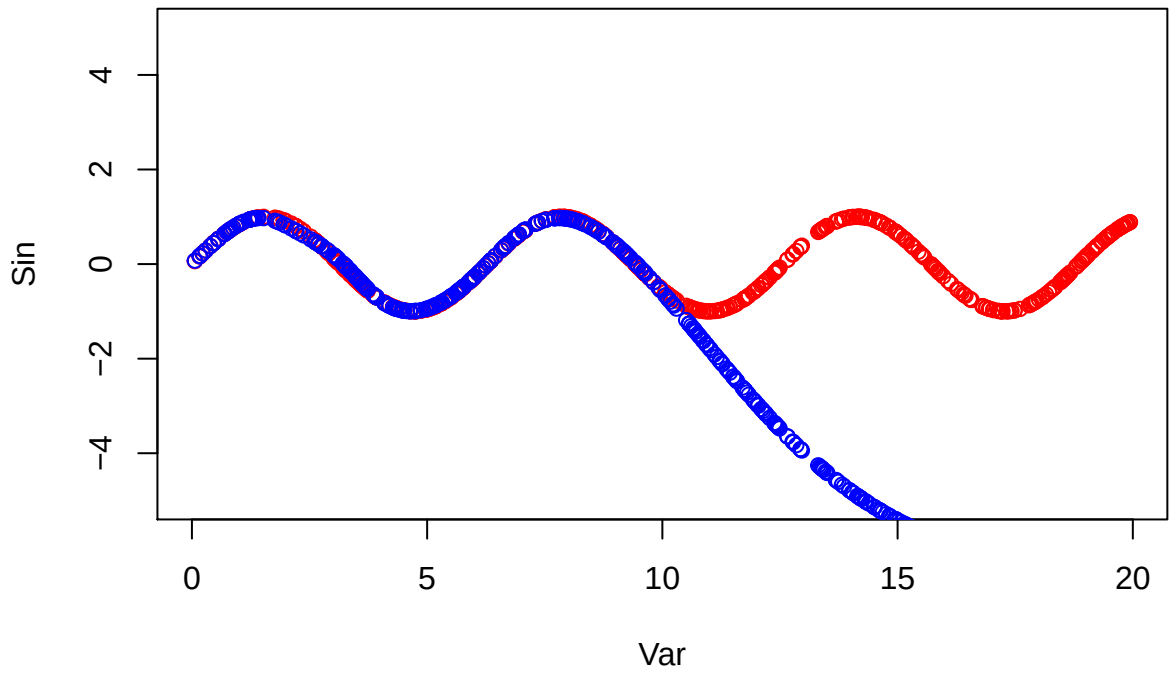


From the above graph we can see that prediction from the neural network on testdata is pretty good with a low mean square error of 2.9393626×10^{-6}

2.

Fitting the neural network for test data of range [0,20]

```
set.seed(1234567890)
Var <- runif(500, 0, 20)
newdata <- data.frame(Var, Sin=sin(Var))
new_predict = predict(nn,newdata)
plot(newdata,col="red", ylim=c(-5,5))
points(newdata$Var,new_predict,col="blue")
```

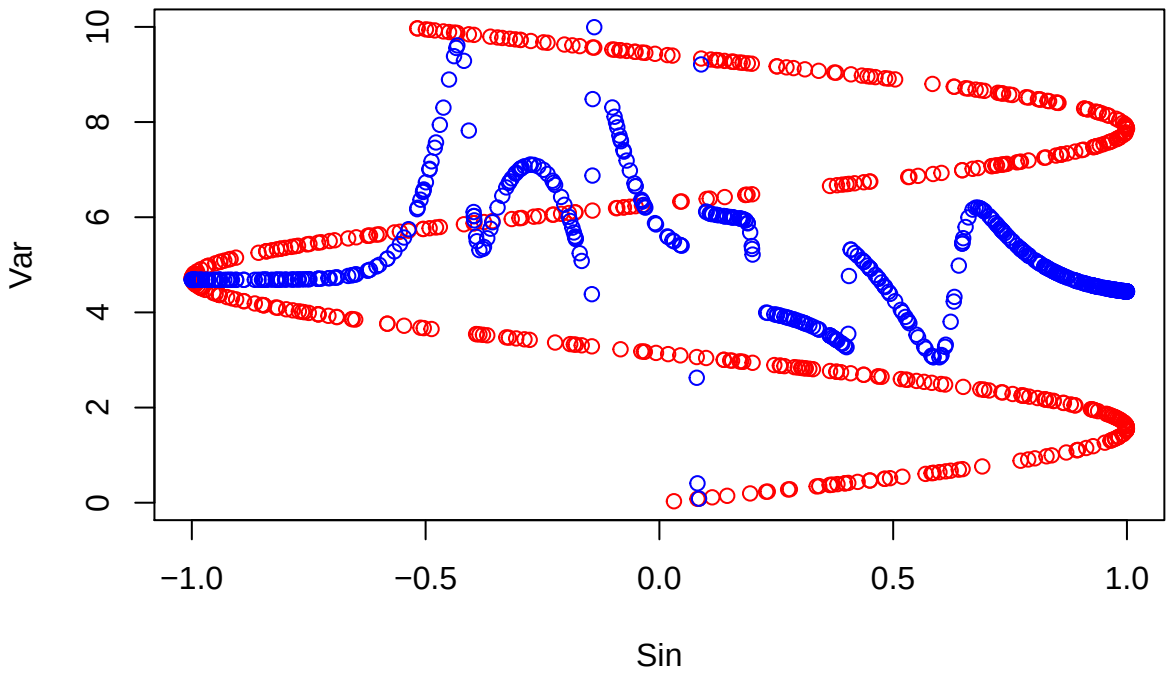


Prediction goes wrong after for the values which are not in training range this is because neural networks are not extrapolation methods. That is to say for regions of the variable space where no training data is available, the output of a neural network is not reliable.

3.

Neural net to predict x from sin(x)

```
set.seed(1234567890)
Var <- runif(500, 0, 10)
newdata <- data.frame(Var, Sin=sin(Var))
w = runif(31,-0.1,0.1)
nn = neuralnet(Var~Sin,newdata,hidden = 10,stepmax=1e8,startweights = w)
```



Sin of a variables oscillates from -1 to 1, from the graph we can see that multiple values of variables give out same output when applied with sine function, this makes it harder for neural network to predict the independent variable from the response of sine function. As result we dont get a good prediction here.