

Neural Network

Varshith konda

20/6/2020

NEURAL NETWORKS

Objective of the following assignment is to use neuralnet package from CRAN to implement neural network.

Our data is 500 uniformly generated random points in the interval $[0,10]$

Training model on training data

Error,threshold and steps data of model during training

## hidden: 5	thresh: 0.01	rep: 1/10	steps: 7285	error: 0.02454	time: 0.47 secs
## hidden: 5	thresh: 0.01	rep: 2/10	steps: 1085	error: 1.35566	time: 0.07 secs
## hidden: 5	thresh: 0.01	rep: 3/10	steps: 1770	error: 0.00744	time: 0.12 secs
## hidden: 5	thresh: 0.01	rep: 4/10	steps: 51675	error: 0.01772	time: 3.15 secs
## hidden: 5	thresh: 0.01	rep: 5/10	steps: 4022	error: 0.00501	time: 0.25 secs
## hidden: 5	thresh: 0.01	rep: 6/10	steps: 3050	error: 0.01323	time: 0.19 secs
## hidden: 5	thresh: 0.01	rep: 7/10	steps: 10307	error: 0.00693	time: 0.64 secs
## hidden: 5	thresh: 0.01	rep: 8/10	steps: 14553	error: 0.00689	time: 0.95 secs
## hidden: 5	thresh: 0.01	rep: 9/10	steps: 11056	error: 0.00309	time: 0.65 secs
## hidden: 5	thresh: 0.01	rep: 10/10	steps: 5572	error: 0.00576	time: 0.35 secs

Here sin values of the points are response variables and points are used as feature for training.

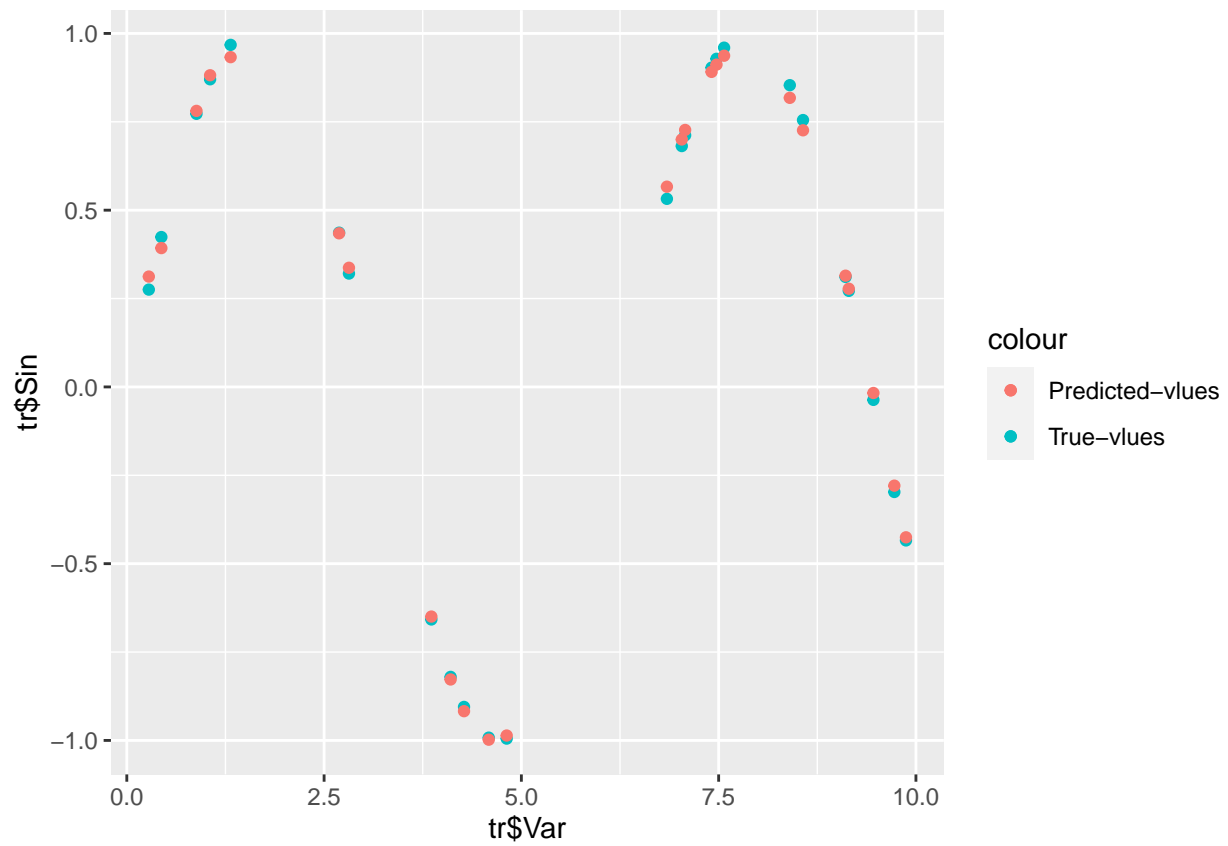
After generating 500 sample points , we are using just 5 percent of the data to train the neural network and 95 percent of the data for testing the accuracy of the network.

We are not using any activation function during model training as we our main motive is not classification , this is the reason “Linear.output” is TRUE.

Five hidden layers and 10 repetitions were carried out during training process , default weights of the network are set to NULL but we are randomly initializing weights with $[-1,1]$.

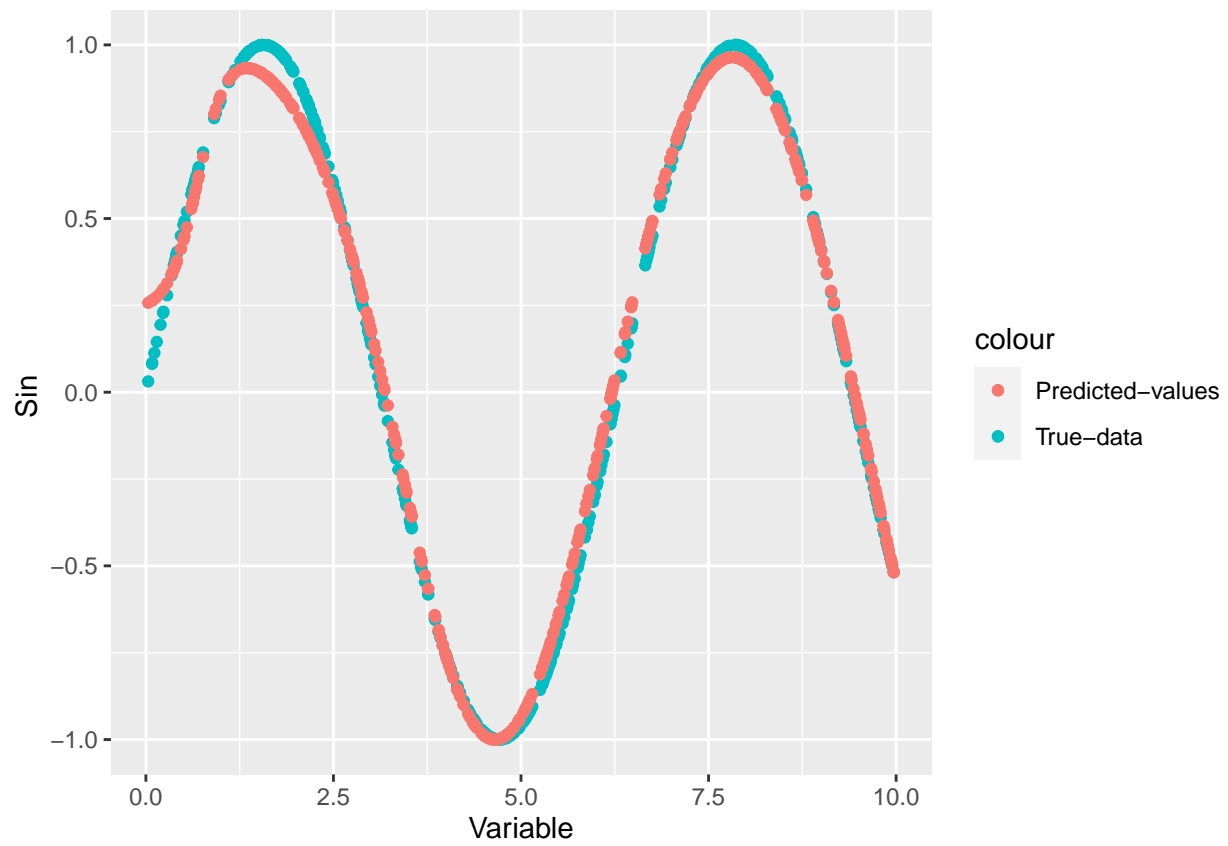
Repetition with lowest error rate and lowest number of steps is used for predicting test values.

Plotting true training values and predicted values from model.



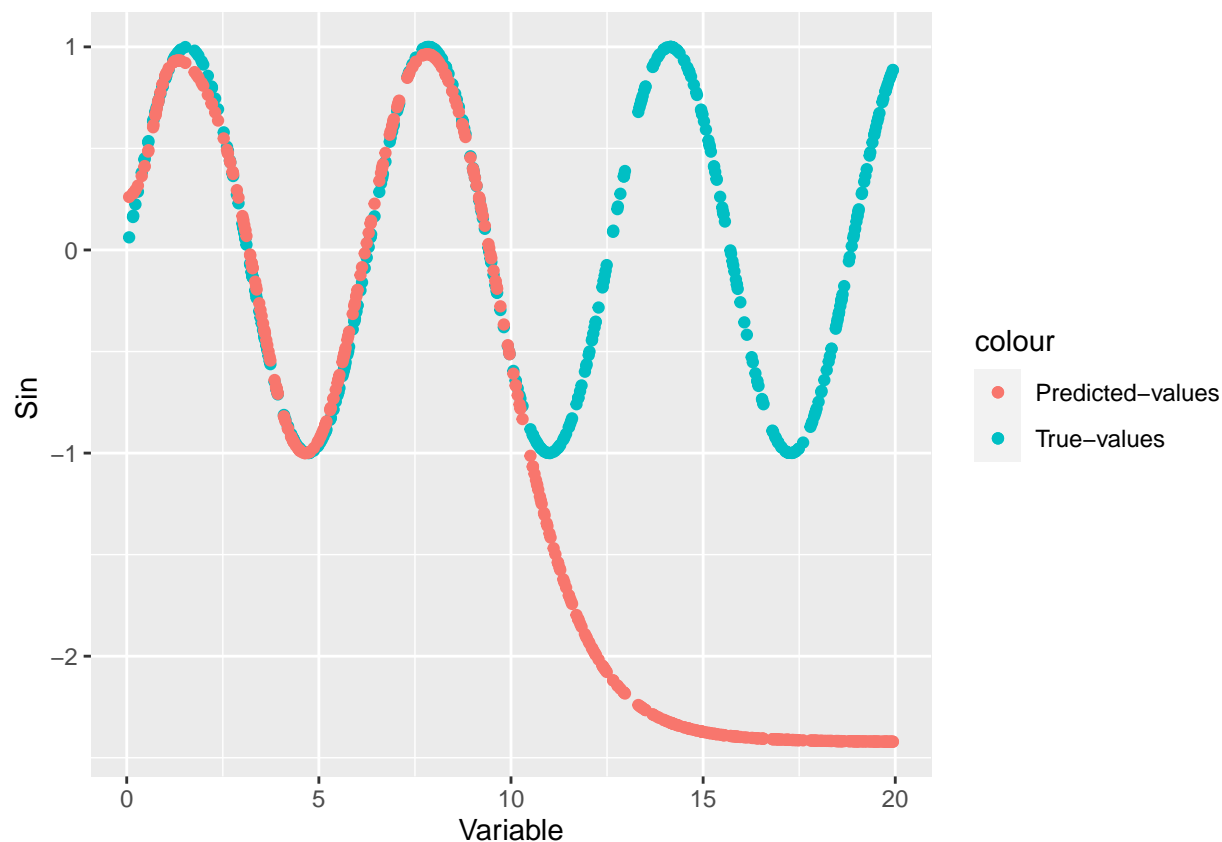
Network performed well for training data with slight margin of errors.

Predicting test data using trained model



Neural net performed moderately with minimal errors , model underperformed while predicting values from 5 to 7.5

Predicting from newly generated data



New data was sampled from uniform distribution in the interval of 0 to 20.

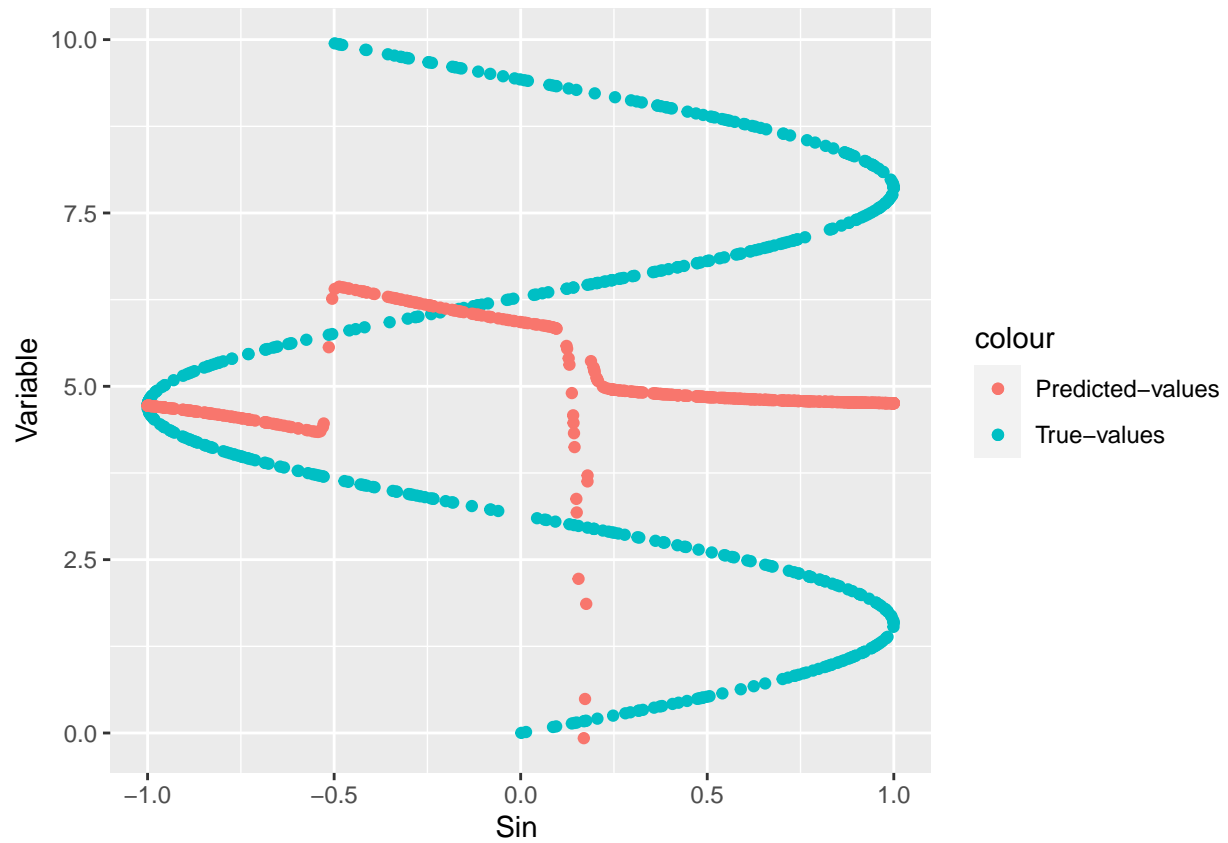
While comparing the predicted values from model with respect to true values, model accurately predicted the values until midway.

Our sample size for training data is very low which can cause “under fitting”, model cannot understand the coherence between features and response variable.

Due to the under fitting of the data error rate of the model can be higher.

Predicting variable from sin values

## hidden: 5	thresh: 0.03	rep: 1/10	steps: 66592	error: 1863.42666	time: 13.41 secs
## hidden: 5	thresh: 0.03	rep: 2/10	steps: 20679	error: 1845.76145	time: 4.09 secs
## hidden: 5	thresh: 0.03	rep: 3/10	steps: 15695	error: 1862.80014	time: 2.99 secs
## hidden: 5	thresh: 0.03	rep: 4/10	steps: 89330	error: 1863.11794	time: 16.5 secs
## hidden: 5	thresh: 0.03	rep: 5/10	steps: 20504	error: 1845.50994	time: 3.86 secs
## hidden: 5	thresh: 0.03	rep: 6/10	steps: 20497	error: 1857.47498	time: 3.84 secs
## hidden: 5	thresh: 0.03	rep: 7/10	steps: 17766	error: 1850.74495	time: 3.35 secs
## hidden: 5	thresh: 0.03	rep: 8/10	steps: 69468	error: 1863.16018	time: 12.86 secs
## hidden: 5	thresh: 0.03	rep: 9/10	steps: 22521	error: 1844.09624	time: 4.24 secs
## hidden: 5	thresh: 0.03	rep: 10/10	steps: 39400	error: 1843.14002	time: 7.39 secs



The algorithm was not converging within the step max range, so the threshold was increased to 0.03 for converging the algorithm within the step max range.

The value on y can have many possibilities from one sin function, which can lead to an improper working model.