Zhuo Chen, Yidan Pan
Equally contributed

Homework 4

a Report network parameters

Table 1: Parameters of Training BP Network to Fit $f(x) = 1/x$

Network parameters	
Topology	$(1 + 1_{Bias})$ — (10) — 1
Transfer function	tanh with slope of 1
Learning parameters	
Initial weights	drawn from U[-0.1, 0.1]
Learning rate (α)	0.01
Momentum	none
Epoch size ($Epoch$)	200
Stopping criteria	error (Err_{RMSD}) $\leq$ 0.05 OR learn count (t) $>$ $4,000,000$
Error measure(Err_{RMSD})	Square root of the sum of $(D - y)^2$ that averaged over all training or testing samples (see formula (1) below)
Input / output data, representation, scaling	
# training samples (N_{tr})	200 (x values drawn randomly from U[0.1,1])
# test samples (N_{tst})	100 (x values drawn randomly from U[0.1,1])
Scaling of inputs	no scaling
Scaling of outputs	map [global min, global max] to [-1,1]

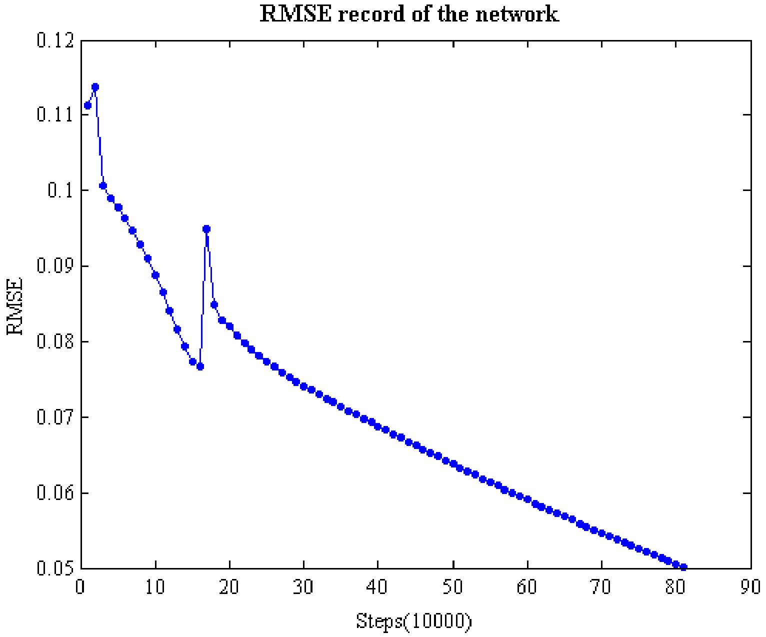
The formula of the error we computed is:

$$(1) \quad Err_{RMSD} = \sqrt{\frac{1}{|X|}\sum_{k=1}^{|X|}(D^k - y^k)^2}$$

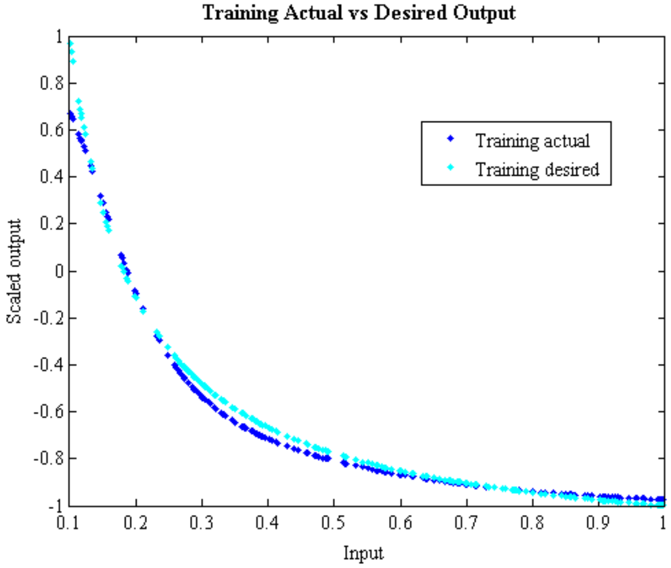
Where k is the sample index, X is either the whole training or testing data set at a particular learn count. Based on the parameter we used, The size of X is constant: 200 for training set, 100 for testing set.

c Plot performance indicator of the network

We use Err_{RMSD}(RMSE) as the performance indicator. The formula is shown as above. The figure shows the RMSE of training set. The RMSE is calculated every 10000 steps(50x200, where 200 is the size of training input set). We saw that two spikes appeared in the process.



After that, we compared the actual output with desired output after training is finished. The data is shown in the figure below:



The desired output vs actual output. The actual output is shown in blue and the desired output is shown in In this run, 812400 steps were taken to make the final RMSE not higher than 0.05 (the average error we accept is 2.5%).

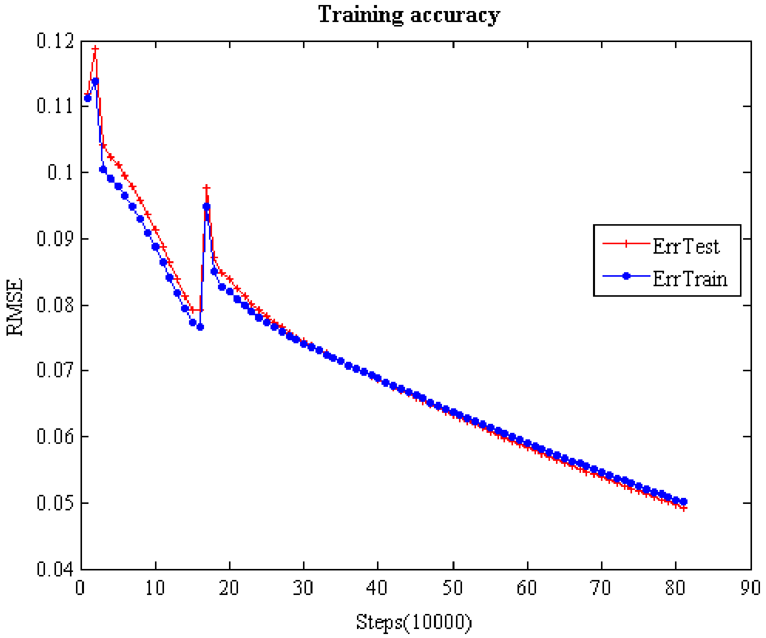
d Number of learning step

We set the criteria as RMSE $\leq$ 0.05 or number of step exceeds 4,000,000. After running the script using the parameters given above for 20 times, the average steps for achieving RMSE $\leq$ 0.05 is 756620, while the standard deviation is 203769. Though the minimum required step is good, further optimization on the parameters is required to generate a more stable learning network.

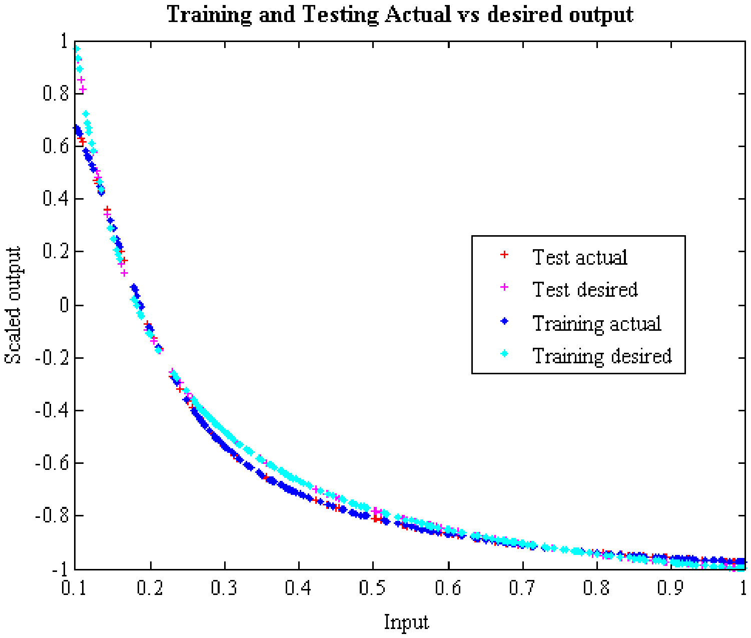
e Training accuracy

The figure showed below compared the RMSE of test set and training set per each 10000 steps (50x200, where 200 is the size of training input set). The RMSE of testing set are shown in red, and the RMSE of training set are shown in blue. The RMSEs of test set in large step number are even lower than the training set.
It is one thing that need to be highlighted that there are actually 50 training runs interval between each point in the figure. For training set, each run goes through 200 samples, and these will be 10000 steps. For test set, each run goes through 100 samples, and these will be 5000 steps.

Note: Here 1 step means 1 sample being taken into the training, so 1 epoch=200 steps (which made the count of steps a huge number).



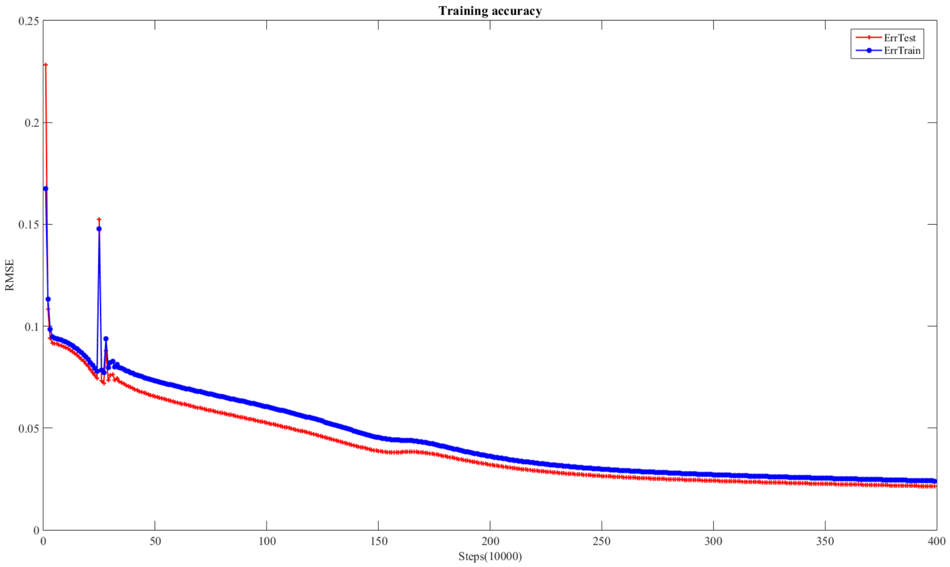
After training is finished, the actual vs desired output for all points of both the training and testing data set is shown in the figure below:
It is clear that the two curves formed by the actual outputs coincide, though the points do not coincide in general. The same pattern is shown in the two curves formed by the desired outputs of training set and testing



set. This data indicates that the recall ability of this net work is pretty good.

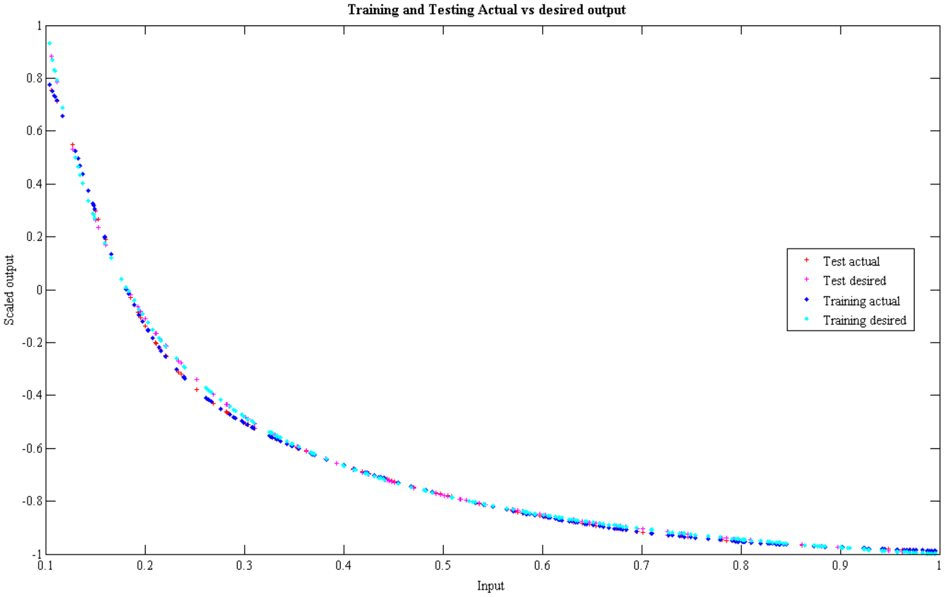
f Increasing the steps

We also tried to increase the number of steps. During training for 20,000 cycles, the changing of RMSE of both training set and test set are shown in the figure below:



The final RMSE of testing set is 0.0214, and the final RMSE of training set is 0.0237. We tried multiple times, and all the trends are similar: we saw some spikes at the very beginning, and then the error rate decreases and becomes flat.

The figure below compared the output pair of testing set and training set. From this we can see that while the general pattern are similar, the fitting is better than the one we performed above.


Training and Testing Actual vs desired output
Scaled output
Input
Test actual
Test desired
Training actual
Training desired