G53MLE Assignment 1 Group Report

Topic: Artifical Neural Networks

Group 3
Group Members:
Zisi Ye
Yiming Zhang
Zhihan Zhang
Afraz Hussain
Zher Shin Tan

Explanation of the parameters chosen (e.g. topology, learning rule, learning rate, nr. of epochs, sizes,) and reports on the difficulties encountered.

Topology

The team intialized a 2-layer network with 20 neurons in the hidden layer. Before team members made this decision, members had tested several networks with different layers and neurons.

Network with more neurons tend to be more accurate, but as the number of neuron increases, training increases significantly.

Besides, members also changed the number of hidden layers. The results, however, showed that it did not have much impact on the overall performance. Therefore, the default 2-layer network was applied.

Learning rate

In order to change the learning rate, members altered the value of ‘mu’. The results showed that when the value of ‘mu’ was around 0.02, network tends to perform the best.

Learning rule

Team members decided to use ‘learngdm’ because it was slightly better than ‘leanrngd’ in terms of accuracy. With the use of ‘learngdm’, lowest accuracy percentage can be improved by up to 2%.

Number of epochs

In the nntool, the default value of epochs is 1000, however, after the team tested sets of data and found out the number should not be this large. Instead, the team selected 100 as the number of the epochs.

Basically, there are two reasons account for this. One is that some training fuction like Bayesian Regularization, the training time per epoch is slow and thus the number of epochs should be fairly small. Another reason is that in the experiments, most of training processes stopped at 20th to 90th epochs, so 100 should be big enough for the training.

Performance function and goal

The regression results of using ‘MSE’ or ‘SSE’ are similar, and slightly better than using ‘MSEREG’, so team members decided to use ‘MSE’. In the training network, members chose to use MSE as performance f function and set the goal to 0 to minimize the error.

Max fail

The team set the max_fail in the training network to 20. The default max_fail is 6, and the team found that the network stopped training after 6 failures. This somehow prevented networks from yielding better results.

Difficulties

Some team members just had their first approach to Matlab, or the neural network toolbox in it, it took some time to understand the resulting graphs, for instance the perfomance validation graph.

When the team tried to change some parameters, such as min_grad, the team failed to find a better value, so the default value was applied in the network.

Besides, the result of the neural network kept changing everytime, so it was a problem for team members to actually see the best result of the network. So, team member took some time to get a good result before changing the network parameters.

Classification results per cross-validation fold, just recall, precision and F_1 measure, not confusion matrix. This implies that the students will have to write a small script that splits the given dataset into training and test sets.

The team used 10 fold cross validation method which means there will be 10 iterations during the training and testing process. At each iteration, 10% of the input data/target is used to test, while the rest 90% of input data/target can be implemented to train the network. Brifely, after the team trained a network, it then be used to plot a confusion matrix which contains predicated labels and target outputs at each iteration.

(notes that each target value corresponds to one recall, one precision and one F1 measure) chart of recall,precision,F1 measure

	iteration 1	iteration2	iteration3	iteration4	iteration5	iteration6	iteration7	iteration8	iteration9	iteration10
recall (target 1)	66.7%	28.6%	16.7%	83.3%	80%	100%	100%	40%	50%	100%
precision (target 1)	20%	33.3%	20%	55.6%	50.0%	30%	62.5%	22.2%	50%	33.3%
F1 measure (target 1)	30.8%	30.8%	18.2%	66.7%	61.5%	46.2%	76.9%	28.6%	50%	50.0%
recall (target 2)	75%	77.8%	42.9%	50%	81.8%	50%	75%	42.9%	80%	77.8%
precision (target 2)	66.7%	50.0%	75%	71.4%	75%	75%	100%	33.3%	80%	77.8%
F1 measure (target 2)	70.6%	60.9%	54.6%	58.8%	78.3%	60%	85.7%	37.5%	80%	77.8%
recall (target 3)	33.3%	42.9%	75%	60%	88.9%	85.7%	100%	55.6%	100%	77.8%
precision (target 3)	71.4%	37.5%	60%	100%	72.7%	50%	54.5%	71.4%	77.8%	77.8%
F1 measure (target 3)	45.4%	40.0%	66.7%	75%	80.0%	63.2%	70.6%	62.5%	87.5%	77.8%
recall (target 4)	83.3%	83.3%	76.9%	100%	78.9%	70.6%	75%	94.1%	93.3%	78.6%
precision (target 4)	90.9%	88.2%	83.3%	70.6%	100%	100%	69.2%	88.9%	87.5%	91.7%
F1 measure (target 4)	86.9%	85.7%	80.0%	82.8%	88.2%	82.8%	72.0%	91.4%	90.3%	84.6%
recall (target 5)	50%	66.7%	80.0%	55.6%	57.1%	50.0%	55.6%	63.6%	75%	81.3%
precision (target 5)	44.4%	88.9%	66.7%	83.3%	80%	100%	100%	100%	90%	86.7%
F1 measure (target 5)	47.0%	76.2%	72.7%	66.7%	66.6%	66.7%	71.5%	77.8%	81.8%	83.9%
recall (target 6)	80%	87.5%	88.2%	78.6%	70.0%	87.5%	76.2%	83.3%	83.3%	92.3%
precision (target 6)	80%	100%	83.3%	68.8%	70.0%	82.4%	88.9%	90.9%	90.9%	85.7%
F1 measure (target 6)	80%	93.3%	85.7%	73.4%	70.0%	84.9%	82.1%	86.9%	86.9%	88.9%

Explain what action you took to ensure generalisation of the network and overcome the problem of overfitting.

Early stopping

Early stopping technique was used to improve generalisation of the network. In this technique, the data was divided into three subset, the training set, test set, and validation set. Besides, the result from using this technique helped to overcome the problem of overfitting.

K-fold cross validation

The team approached K-fold cross validation to overcome the overfitting problem. The purpose of using K-fold cross validation is to prevent overfitting of the network. The technique is almost same as early stopping by dividing data into subsets. If the team just used the whole input data to train the network, then the network may be overfit. However, by applying 10-fold cross validation, the team got 10 different confusion matrix at the end of the experiment. After processing these data, team members got the results of average confusion matrix and recall,precision, F_1 measure. These values can describe the performance of the network in a more reasonable way.

Average cross validation classification results, that include:

o Confusion matrix.

o Average recall and precision rates.

(*Hint:* you can derive them directly from the previously computed confusion matrix)

o The F_1-measure derived from the recall and precision rates of the previous step.

The average recall and precision rates for each class can be calulated by suming up the results from 10 iterations.

confusion matrix

	target 1	target 2	target 3	target 4	target 5	target 6
output 1	28	9	4	6	4	2
output 2	18	57	3	4	1	1
output 3	6	4	59	9	4	5
output 4	1	5	12	124	1	6
output 5	18	4	5	1	69	9
output 6	3	5	7	4	5	114

average recall,precision and F1 measure

	target 1	target 2	target 3	target 4	target 5	target 6
recall	59.6%	67.9%	67.8%	83.2%	65.1%	82.6%
precision	37.8%	67.9%	65.6%	87.3%	82.1%	83.2%
F1 measure	42.3%	67.9%	66.7%	85.2%	72.6%	82.9%