Module 5
IT3105

Magnus Gundersen, Simon Borøy-Johnsen

MTDT

November 26, 2015

1 The different ANNs

During developing and testing our solution, we found out that using the activation function T.tanh in all the hidden layers was sufficient in order to achieve a good score (5 points for the training and testing sets, and usually 5 for the demo set as well). Therefore, the only thing that differs between the different ANNs are the numbers of layers, and the number of nodes in each layer.

The table below describes the different ANNs we implemented, with their number of hidden layers and topologies.

ANN	Number of hidden layers	Topology
1	2	[50, 25]
2	2	[100, 50]
3	3	[150, 100, 50]
4	3	[392, 196, 98]
5	3	[784, 392, 196]

2 Comparison

Each ANN was run 20 times. In each run, the network trained over a maximum of 50 epochs. When the error rate reached a certain level (10^{-4}), the training for that run stopped. The training data set was split into batches of size 100. For each batch, the whole batch was fully parsed through the network before the back propagation started.

The table below shows the results after running the following sequence of actions 20 times for each ANN;

1. Train the network using the 'all_flat_mnist_training_cases' file.
2. Classify the MNIST training data set
3. Classify the MNIST testing data set
4. Classify the supplied demo data set

ANN	Avg. scores			Statistics (after training)	
	Train	Test	Demo	Avg. number of epochs	Avg. accuracy
1	5.0	5.0	4.1	50.0	0.99930
2	5.0	5.0	4.2	49.4	0.99986
3	5.0	5.0	4.6	39.4	0.99990
4	5.0	5.0	4.8	31.7	0.99990
5	5.0	5.0	4.9	26.2	0.99993

As can be seen in the table above, both ANN 4 and ANN 5, achieved full scores in most of the runs. However, ANN 5 reached the target error rate in only 26 epochs, while the others needed to push the training data through the network a greater amount of times. We are therefore going to use ANN 5 during the demo session.