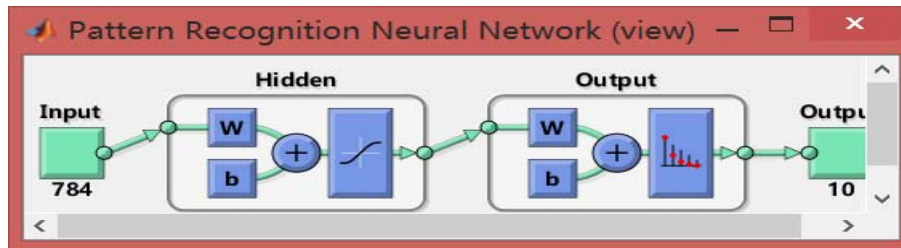


Report

1,

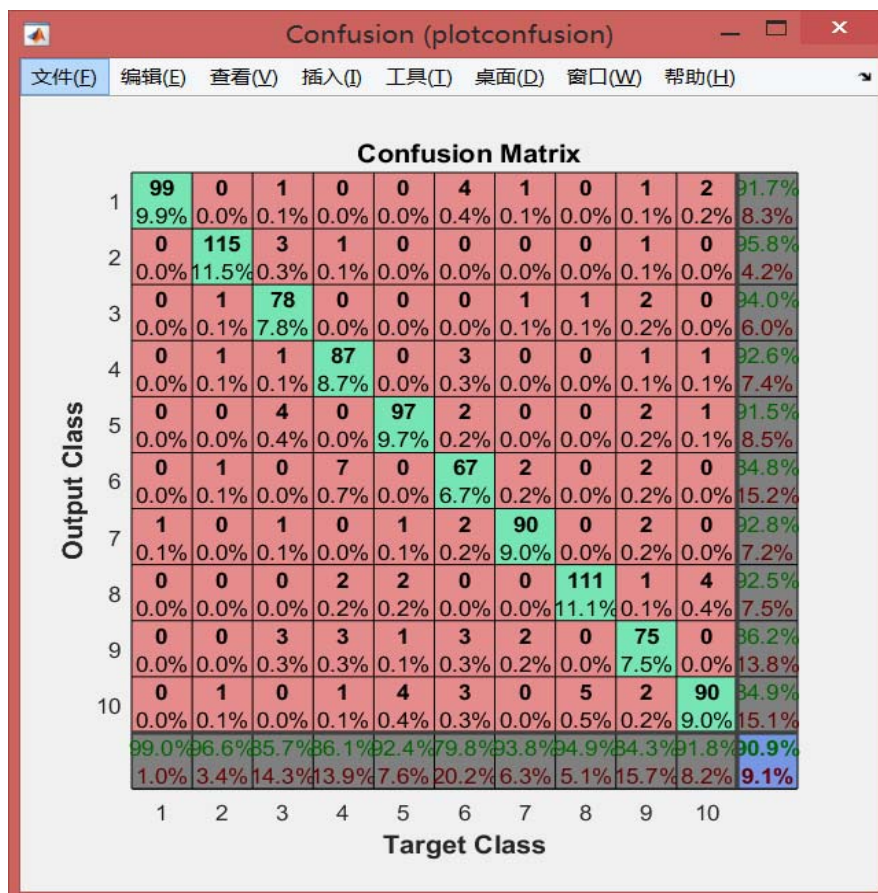


My neural network is a neural network that has just 1 hidden layer, the hidden layer has 26 neurons. The training time is 2 seconds, learning rate is 0.8, validation checks is 6, the epoch is 57 iterations.

Note: I do this coursework in my personal laptop, and I try to run the totally selfsame program in other computers, I found the training time is my computer is much shorter than the time in other computers. Therefore, the training time in this report is normal.

2,

My testing accuracy is 90.8%, the cost time is 2 seconds.



Each column of the confusion matrix represents the prediction category, and each row represents the true attribution category of the data. The green blocks represent the numbers of each correct training results and these accuracies. The (11,11) block represent the total accuracy.

3,

	Number of layers	Neuron numbers	Learning rate	Activation function	Testing accuracy (Time)
1	1	26	0.8	trainscg	90.8%(2s)
2	1	100	0.8	trainscg	91.2%(3s)
3	3	26,26,26	0.8	trainscg	91.9%(30s)
4	1	26	0.8	traingdx	91.1%(5s)
5	1	26	0.1	trainscg	90.9%(2s)

4, The PCA can compress the data in training neural network. The process of compression is equivalent to reduce the data's dimension. This method can extremely save the training time of neural network. Besides, it can make the data set easier to use and removal of noise.

For example: I use feedforwardnet() to train a default single layers neural network, it cost 26 minutes and the accuracy is 87.5%. When I reduce the dimension of "train_feature", this process only cost 3 seconds and the accuracy is 87.2% which is nearly same with the former.