CSE 301 Assessment 1 Report

1. Introduction

Recently, image recognition has important usage such as auto-driving and data processing field. Meanwhile, several artificial neuron network (ANN) models including MLP, CNN, competitive learning, SOM, and RBF have been widely applied to image recognition from the 1990s and can improve the recognition accuracies significantly. For example, CNN model was applied for traffic sign classification by the H. H. Aghdam and E. J. Heravi [1]. However, a single neuron network may often exhibit overfitting behavior in the training stage. In addition, the performance of an ANN model depends of choice of different parameters or algorithms used and a number of researchers focus on this open question [2]. This report would present the experiment which implements multi-layer perceptron (MLP) with back-propagation training algorithm and hybrid model of Radial Basis Function (RBF) network, where the RBF centers are the prototypes from unsupervised competitive learning (k-means clustering and SOM) for traffic sign collection and vehicle logo set. Furthermore, the convergence performance would be presented and the relationship between models and parameters would be discussed.

2. Models and Methods

Firstly, multi-layer perceptron (MLP) networks, which consists of an input layer, one or more hidden layer and an output layer, is a class of feedforward artificial neural network. Moreover, in MLP networks, neurons would use a nonlinear activation function which can define the output of nodes. In addition, the number of hidden layer and neurons would determine the speed of convergence and performance of model. For the weight updated, the MLP networks usually follow delta rule which is a popular back-propagation training algorithm in the training stage. Furthermore, the momentum is to stabilize the weight change using a combination of the gradient decreasing term with a fraction of the previous weight change

Next, radial basis function (RBF) network is an artificial neural network which would use radial basis functions as activation functions. Therefore, RBF networks often have an input layer, a hidden layer with a non-linear RBF activation function and a linear output layer, Radial basis function networks have a number of uses, including function approximation, time series prediction, classification, and system control. In addition, the most important feature is the method which can denote the model with centers [3].

3. Training and Testing

In this experiment, two datasets would be used for classification. One is a collection of traffic sign images for six different types. The other one is the representation of five different types of vehicle logo images. In addition, training and test set are 80% and 20% respectively.

Procedure for MLP

1. Load trafficsign.mat
2. Divide the training dataset and testing dataset into 80% and 20%
3. Create a feed-forward backpropagation network and have all variables parameterized with Neural Network Toolbox
4. Train the network model and use an evaluation function to get the accuracy of the model in both the training set and the testing set, speed of convergence.
5. Repeat the training stage the above procedure for 100 times and record average value.
6. Vary the number of hidden units and repeating above steps

Similarly, loading logo.mat and run the experiment again.

Procedure for RBF with K-means

1. Load trafficsign.mat
2. Divide the training dataset and testing dataset into 80% and 20%
3. Set a specific *k* value and use *kmeans* with *gaussian* method to find the centers of the hidden nodes, then get *C* matrix.
4. Calculate the weight W
5. Input training dataset or testing dataset to the network and get the corresponding target vector respectively, and then use an evaluation function to get the accuracy of the mode.
6. Repeat step 1-5 for 100 times and record the average accuracy.
7. Similarly, loading logo.mat and run the experiment again.

Procedure for RBF with SOM

1. Load trafficsign.mat
2. Divide the training dataset and testing dataset into 80% and 20%
3. Set a specific Map size ***map_1***, and generate the SOM map *sM* with use the ***som_make*** method of somtoolbox. Set the center of RBF is

$$C = sM.codebook$$

4. Calculate the weight W
5. Input training dataset or testing dataset to the network and get the corresponding target vector respectively, and then use an evaluation function to get the accuracy of the mode
6. Repeat step 1-5 for 100 times and record the average accuracy
7. Similarly, loading logo.mat and run the experiment again.

Experimental Results

Hidden units	Speed of convergence (CPU time)	Accuracy of the results in training set (%)	Accuracy of the results in testing set (%)

10	5.2359	81.3732	77.2143
25	5.9547	93.0634	90.1429
50	7.1172	96.1268	91.5714
75	7.893	96.2852	91.3571
100	8.9125	97.007	91.4286
150	10.6828	97.3239	89.8571
200	11.5641	96.4965	87.3571
250	Early Stop		

For the MLP experiment, there are methods and results of experiment and simple analysis.

Traffic Sign Collection:

Vary the number of hidden units from 10 to 250 and other parameters are set by default values.

Table 1 Accuracy and Running time of MLP when hidden units change

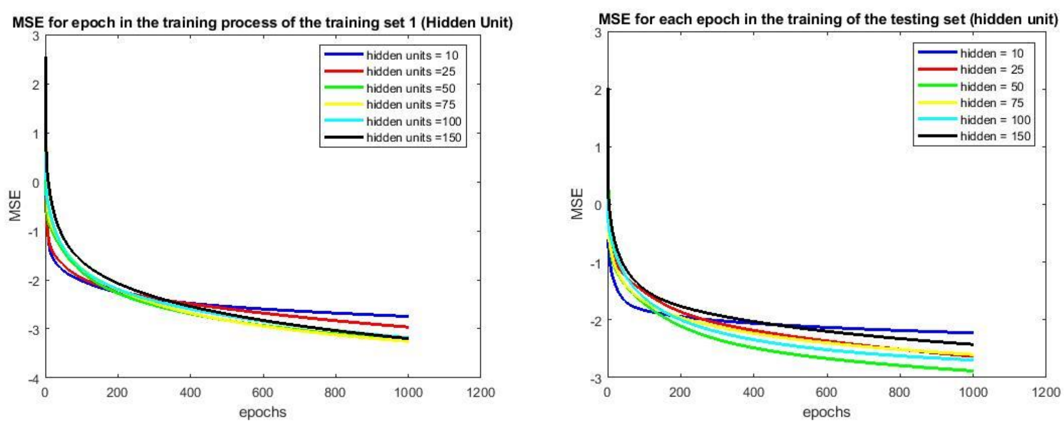

Figure 1 MSE for each epoch in the training process of the training set or testing set when hidden units change

As a result, both the accuracy of the results in training set and testing set would be increased with increasing the number of hidden units. However, the model which has 50 hidden units can provide the highest accuracy of results in testing set 91.5714%. After this, the accuracy of results in testing set would not increase but the accuracy of the results in training set would increase continually. Obviously running time of MLP has positive correlation with the hidden units increasing. In addition, the training would be stopped early because of overfitting when hidden units >= 250.

Set hidden units = 50, changing the learning rate:

Learning rate	Accuracy of the results in training set	Accuracy of the results in testing set
0.01	96.2852	91.5
0.05	99.1725	95.5
0.10	99.4542	97.6429
0.15	97.5	94.6429
0.20	Early Stop	

Table 2 Accuracy of MLP when learning rate changes

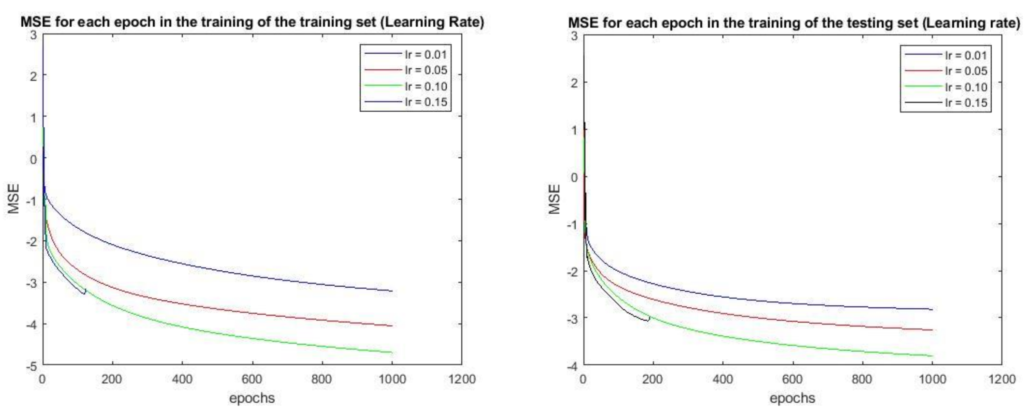

Figure 2 MSE for each epoch in the training process of the training set or testing set when learning rate changes

As a result, both the accuracy of the results in training set and testing set would be increased with increasing the learning rate. However, the model which has 0.10 learning rate can provide the highest accuracy of results in testing set 97.6429% and in training set 99.4542%. After this, the accuracy of results would not increase. In addition, the training would be stopped early because of overfitting when learning rate >= 0.20.

Set the learning rate = 0.10, changing the momentum:

Momentum	Accuracy of the results in training set	Accuracy of the results in testing set
0.60	98.9965	95.3571
0.65	99.2606	95.5714
0.70	96.4261	93.6429
0.75	99.4894	95.6429
0.80	99.2958	95
0.85	95.3169	93.1429
0.9	Overfitting	Overfitting

Table 3 Accuracy of MLP when momentum changes

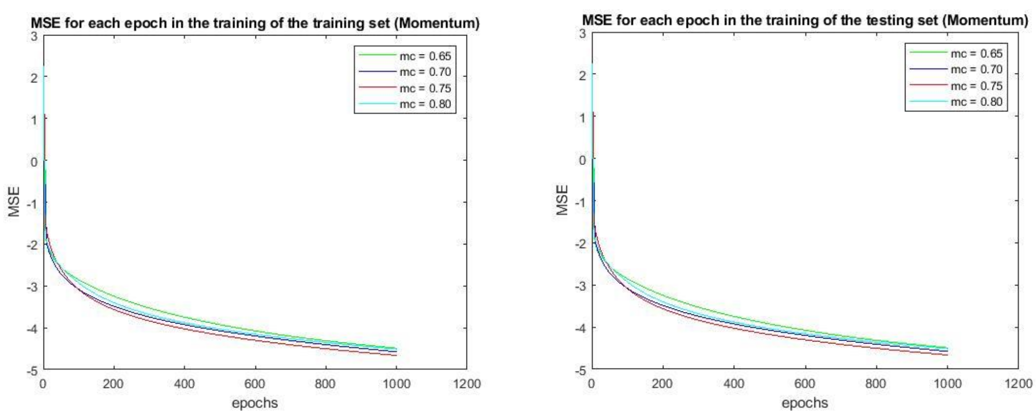

Figure 3 MSE for each epoch in the training process of the training set or testing set when momentum changes

As a result, both the accuracy of the results in training set and testing set would be increased with increasing momentum. However, the model which has 0.75 momentum can provide the highest accuracy of results in testing set 95.6429% and in training set 99.4894%. After this, the accuracy of results would not increase. In addition, the training would be stopped early because of overfitting when momentum >= 0.9.

Therefore, the MLP model which has best performance has hidden units = 50, learning rate = 0.10, momentum = 0.75.

Vehicles Logo Collection:

Vary the number of hidden units from 10 to 200 and other parameters are set by default values.

Hidden units	Speed of convergence	Accuracy of the results in training set	Accuracy of the results in testing set
10	2.0094	83.257	78.4286
25	4.0758	93.5915	86.9565
50	5.8172	96.8085	90
75	6.4688	94.2553	82.6087
100	6.9125	96.3204	81.7143
150	7.5563	94.1489	68.2609
200	Early stop		

Table 4 Accuracy and Running time of MLP when hidden units change

As a result, both the accuracy of the results in training set and testing set would be increased with increasing the number of hidden units. However, the model which has 50 hidden units can provide the highest accuracy of results in testing set 90%. After this, the accuracy of results in testing set would not increase but the accuracy of the results in training set would increase continually. Obviously running time of MLP has positive correlation with the hidden units increasing. In addition, the training would be stopped early because of overfitting when hidden units >= 200.

Set hidden units = 50, changing the learning rate:

Learning rate	Accuracy of the results in training set	Accuracy of the results in testing set
0.01	97.0423	92.8571
0.05	98.9437	96.4286
0.10	98.2979	90.4348
0.15	Early stop	

Table 5 Accuracy of MLP when learning rate changes

As a result, both the accuracy of the results in training set and testing set would be increased with increasing the learning rate. However, the model which has 0.05 learning rate can provide the highest accuracy of results in testing set 95.4286% and in training set98.9437%. After this, the accuracy of results would not increase. In addition, the training would be stopped early because of overfitting when learning rate >= 0.15.

Set the learning rate = 0.10, changing the momentum:

Momentum	Accuracy of the results in training set	Accuracy of the results in testing set
0.60	99.4718	96.5714
0.65	Early Stop	

0.70	98.5915	94.2857
0.75	99.5958	97
0.80	99.7535	97.2857
0.85	99.6479	96.5714
0.9	Early Stop	

Table 6 Accuracy of MLP when momentum changes

As a result, both the accuracy of the results in training set and testing set would be increased with increasing momentum. However, the model which has 0.80 momentum can provide the highest accuracy of results in testing set 97.2857% and in training set 99.7535%. After this, the accuracy of results would not increase. In addition, the training would be stopped early because of overfitting when momentum >= 0.9 or momentum = 0.65.

As a result, the MLP model which has best performance is hidden units = 75, learning rate = 0.10 ,momentum = 0.80.

For the RBF network with k-means experiment, there are methods and results of experiment and simple analysis.

Vary the k value which means the number of prototypes.

For traffic sign dataset:

k	Speed of convergence	Accuracy of the results in training set	Accuracy of the results in testing set
10	0.059531	0.95458	0.94714
30	0.11641	0.99669	0.98271
50	0.15453	0.99975	0.98857
70	0.1925	0.99996	0.98743

90	0.25203	0.99999	0.98414
110	0.29531	1	0.98271

Table 7 Accuracy and Running time of RBF1 when k value changes

As a result, both the accuracy of the results in training set and testing set would be increased with increasing k value. However, the model which k = 50 can provide the highest accuracy of results in testing set 0.98857%. After this, the accuracy of results in testing set would not increase but the accuracy of the results in training set would increase continually. Obviously running time of RBF has positive correlation with the k value increasing.

For vehicle logo dataset:

K	Speed of convergence	Accuracy of the results in training set	Accuracy of the results in testing set
5	0.025156	0.86649	0.8413
10	0.031719	0.99021	0.97478
20	0.047344	0.99904	0.98217
30	0.053281	0.99968	0.97348
40	0.060937	0,99998	0.95957
50	0.068437	1	0.96652

Table 8 Accuracy and Running time of RBF1 when k value changes

As a result, both the accuracy of the results in training set and testing set would be increased with increasing k value. However, the model which k = 20 can provide the highest accuracy of results in testing set 0.98217%. After this, the accuracy of results in testing set would not increase but the accuracy of the results in training set would increase continually. Obviously running time of RBF has positive correlation with the k value increasing.

For the RBF network with SOM experiment, there are methods and results of experiment and simple analysis.

Vary the SOM map size which means the number of prototypes.

For traffic sign dataset:

Map Size	Speed of convergence	Accuracy of the results in training set	Accuracy of the results in testing set
3*3	0.14438	0.91218	0.89657
4*4	0.19328	0.97644	0.96386
5*5	0.22562	0.99398	0.97914
6*6	0.23016	0.99838	0.98771
7*7	0.30078	0.99961	0.986
8*8	0.46141	1	0.98493

Table 9 Accuracy and Running time of RBF2 when SOM size changes

As a result, both the accuracy of the results in training set and testing set would be increased with increasing of map size. However, the model which map size = 6*6 can provide the highest accuracy of results in testing set 98.771%. After this, the accuracy of results in testing set would not increase but the accuracy of the results in training set would increase continually. Obviously running time of RBF has positive correlation with the map size increasing.

For vehicle sign dataset:

Map Size	Speed of convergence	Accuracy of the results in training set	Accuracy of the results in testing set

3*3	0.099687	0.99032	0.9813
4*4	0.11375	0.99936	0.98609
5*5	0.14578	1	0.98913
6*6	0.198891	1	0.98565
7*7	0.23516	1	0.98391

Table 10 Accuracy and Running time of RBF2 when SOM size changes

As a result, both the accuracy of the results in training set and testing set would be increased with increasing of map size. However, the model which map size = 5*5 can provide the highest accuracy of results in testing set 98.913%. After this, the accuracy of results in testing set would not increase but the accuracy of the results in training set would increase continually. Obviously running time of RBF has positive correlation with the map size increasing.

Comparison

We need to compare the best MLP model from above steps with the RBF models (namely RBF1 and RBF2) using the performance indicators average accuracy (or error rate) in testing set and confusion matrix.

For the first dataset:

Model	Accuracy of the results in testing set (%)	Speed of convergence
MLP (Hidden units = 50 Learning rate = 0.10 Momentum = 0.75)	96.74	6.2786
RBF1 with k-means (k = 50)	98.687	0.16031
RBF2 with SOM (Map size	98.163	0.93594

= 6*6)		

Table 11 Accuracy and running time of Tested models

For the second dataset:

Model	Accuracy of the results in testing set	Speed of convergence
MLP (Hidden units = 75 Learning rate = 0.10 Momentum = 0.80)	94.2857	6.9281
RBF with k-means (k = 20)	99.391	0.0631
RBF with SOM (SOM size = 5*5)	99.565	0.99565

Table 12 Accuracy and running time of Tested models

Obviously, we can find that accuracy of RBF1 with k-means and RBF2 with SOM are almost the same, however, the RBF2 with SOM network costs 6 times running time than RBF with SOM. MLP also performs a good classification rate and has the longest running time.

There are confusion matrixes which can describe the performance of each classification model on a set of test data for which the true values are known.

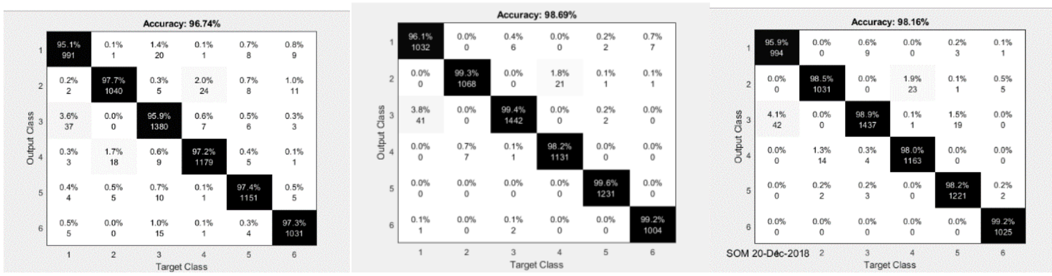

Figure 13 Confusion matrixes of MLP, RBF1, RBF2 models for dataset 1

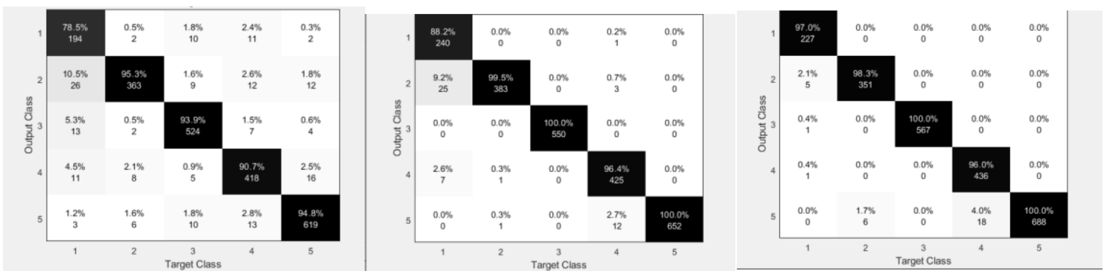

Figure 14 Confusion matrixes of MLP, RBF1, RBF2 models for dataset 2

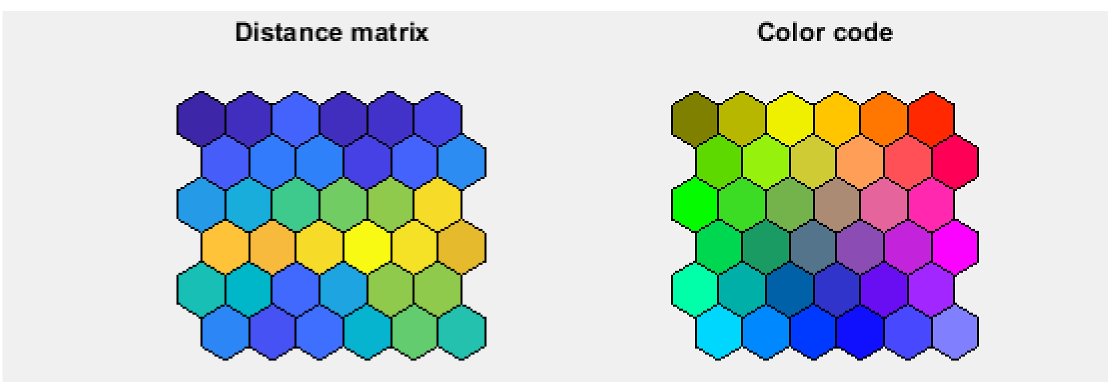

Figure 15 SOM image

Conclusion and Future Planning

This report proposes a hybrid classification of traffic sign and vehicle logo based on MLP, RBF, k-means and SOM. It is known that RBF network is one of the most successful machine learning tools for classification. Therefore, the result can confirm this conclusion. Obtained classification time and recognition rates of RBF are better than the running time and the accuracy of MLP applied on the same dataset set.

The reason for this situation is RBF has a good efficiency on classification when data is clipped and sorted in order, especially for the large datasets. Advantage of RBF networks based hybrid algorithms cannot perform in this situation. Meanwhile, in this experiment, RBF with K-means has a better classification rate and cost less CPU space than RBF with SOM when centers are the same. It is also proved that RBF with SOM need more powerful CPU and time to classify the data.

Reference

[1] Bailing. Zhang, *Handwritted Digit Recignition by Neual 'Gas' Model and Population Decoding*, Callaghan, Australia,1998.

[2] N. Assdrin, A. Smith and D. Turnbull, *Classifying Facial Expression with Radial Basis Function Networks, using Gradient Descent and K-means*, San Diego, USA, 2003.

[3] H. H. Aghdam and E. J. Heravi, *Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification*. Springer, 2017.

[3] Graupe, D.; Kordylewski, H. *Network based on SOM (Self-Organizing-Map) modules combined with statistical decision tools Proceedings of the 39th Midwest Symposium on Circuits and Systems*. 1. pp. 471–474 vol.1, 1996

(Some points are from CSE301 Slides).