COL774: Machine Learning. Assignment 3

Arpan Mangal — 2016CS10321

April 2019

1 Decision Trees (and Random Forests)

1.1 Multi-Way Decision Tree

First the data was preprocessed and each numerical attribute was converted into a Boolean attribute.

Next tree was grown using the training data, and train, validation and test set accuracies were plotted against the number of nodes in the tree. (Fig. 1)

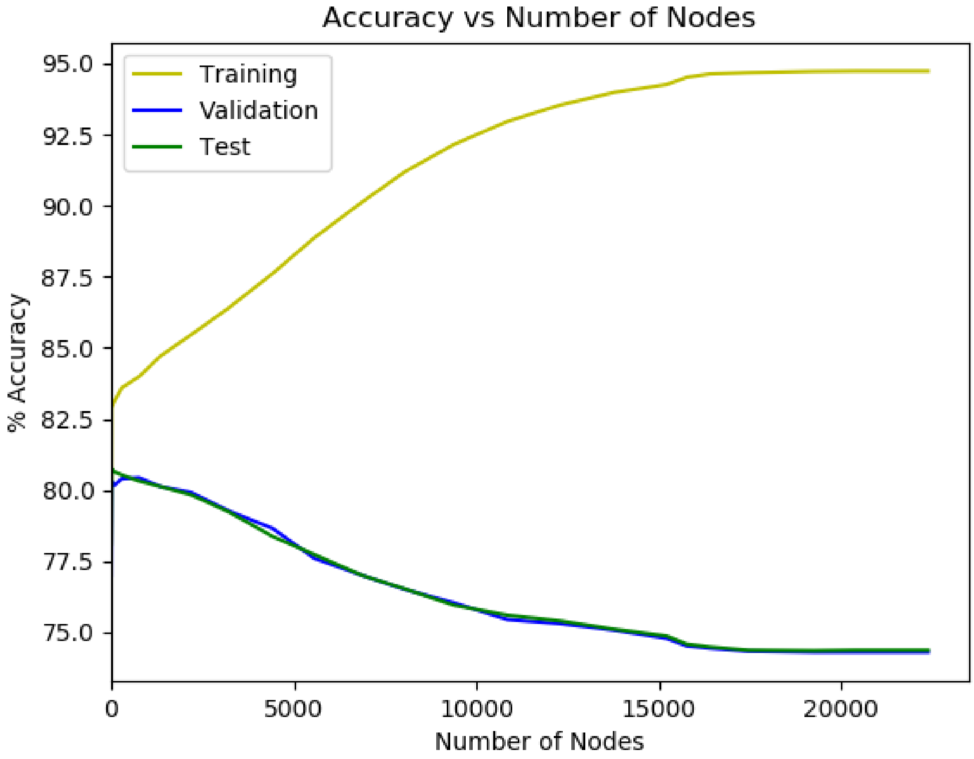


Figure 1: Decision Tree

The tree is grown using DFS. For the plots, trees were grown multiple times with different possible max. depths, in an iterative deepening (ID) pattern.

Final Accuracies for the full tree:

- Training Accuracy: 94.73 %
- Validation Accuracy: 74.30 %
- Test Accuracy: 74.37 %

1.2 Pruning

In this part the tree was grown fully and then pruned bottom-up using the validation set.
The accuracies with the number of nodes is shown in Fig. 2

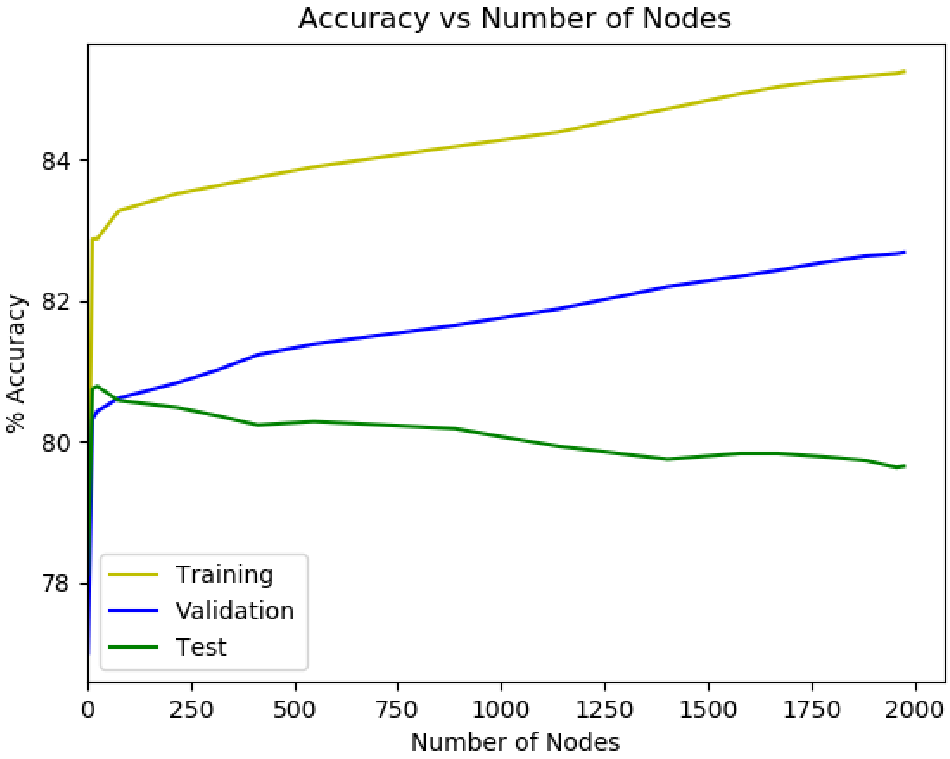


Figure 2: Decision Tree with Pruning

Final Accuracies for the full tree:

- Training Accuracy: 85.25 %
- Validation Accuracy: 82.68 %
- Test Accuracy: 79.65 %

Observations

1. The model performs better than the model in part (a)

2. In part (a) the model overfits only on the train set, but in this the model starts to overfit on the validation set as well.

1.3 Continous Attributes

Numerical attributes weren't preprocessed using the median and the median was calculated at each node online for the respective features. Further the respective feature was not dropped and was reused in some further depth.

The trees were grown for different depths. The plots of accuracies vs. number of nodes is shown in Fig. 3

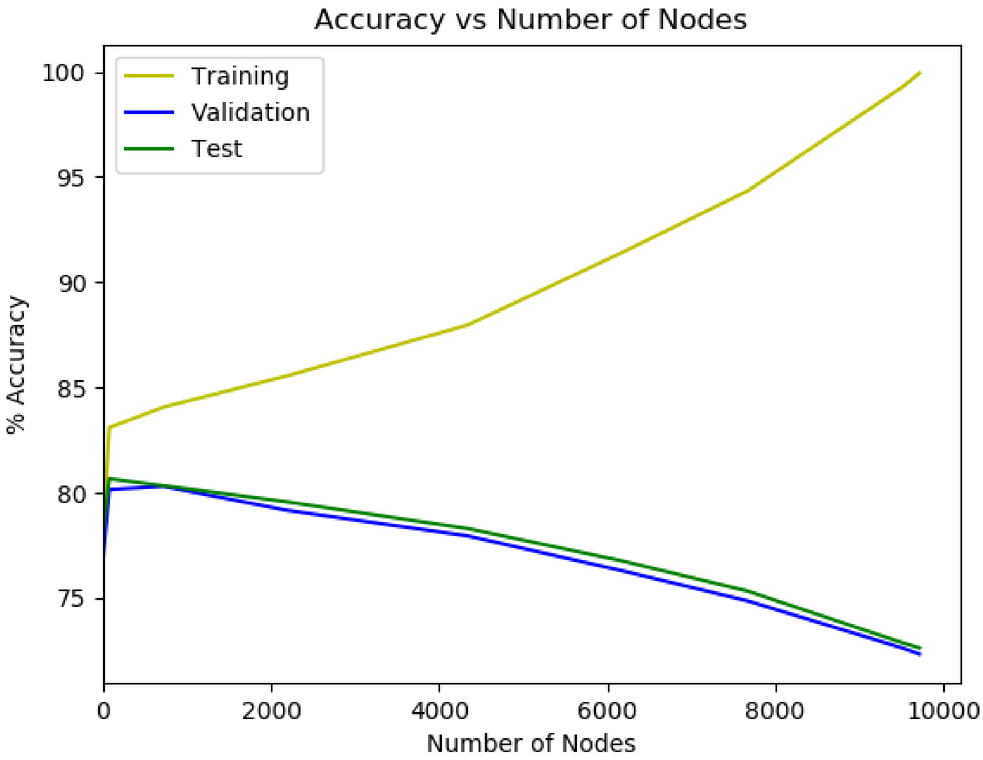


Figure 3: Decision Tree with Continous Attributes

For depth 20:

- Training Accuracy: 99.93 %

- Validation Accuracy: 72.33 %

- Test Accuracy: 72.63 %

Observations

1. The model highly overfitted as depth increased.
2. For the same number of nodes, the accuracies of this model is better than part (a) initially. However with more depth this model severely overfit and thus the accuracy drops.
3. With more depth the model approaches 100 % training accuracy due to leafs becoming purer.

1.4 Decision Trees with scikit-learn

Scikit-learn was used to learn the decision tree over the Training Data.

Various values of the following parameters were experimented with and these are the best values which we get:

- max_depth: 7
- min_samples_split: 2
- min_samples_leaf: 55

Accuracies on the above best parameters:

- Training Accuracy: 83.34 %
- Validation Accuracy: 80.767 %
- Test Accuracy: 81.033 %

Observations

1. The accuracies are much higher than in part (c), because the model in part (c) had essentially overfitted
2. The testing accuracy is higher than in part (b), but the validation accuracy is less than part (b), since while doing pruning the model had partly overfit the validation data.

1.5 One Hot Encoding

Numerical features were one-hot encoded and then passed to Scikit-learn model for training.

Various values of the following parameters were experimented with and these are the best values which we get:

- max_depth: 6
- min_samples_split: 2
- min_samples_leaf: 60

Accuracies on the above best parameters:

- Training Accuracy: 80.461 %
- Validation Accuracy: 79.583 %
- Test Accuracy: 78.333 %

Observations

1. The accuracies are lower than those of non-one-hot encoded data.
2. One possible reason is that making the features as one-hot data introduced sparsity in the data, and with so many features we need considerably more amount of data.

1.6 Random Forests

Scikit-learn random forests were used for learning the model over the training data.

Various values of the following parameters were experimented with and these are the best values which we get:

- n_estimators: 7
- max_features: 0.7
- bootstrap: False
- max_depth: 4

Accuracies on the above best parameters:

- Training Accuracy: 83.183 %
- Validation Accuracy: 80.833 %
- Test Accuracy: 80.867 %

Observations

1. The test accuracy is the best till now.
2. The validation accuracy is the best excluding part(b) where it was more than 82 %.

2 Neural Networks

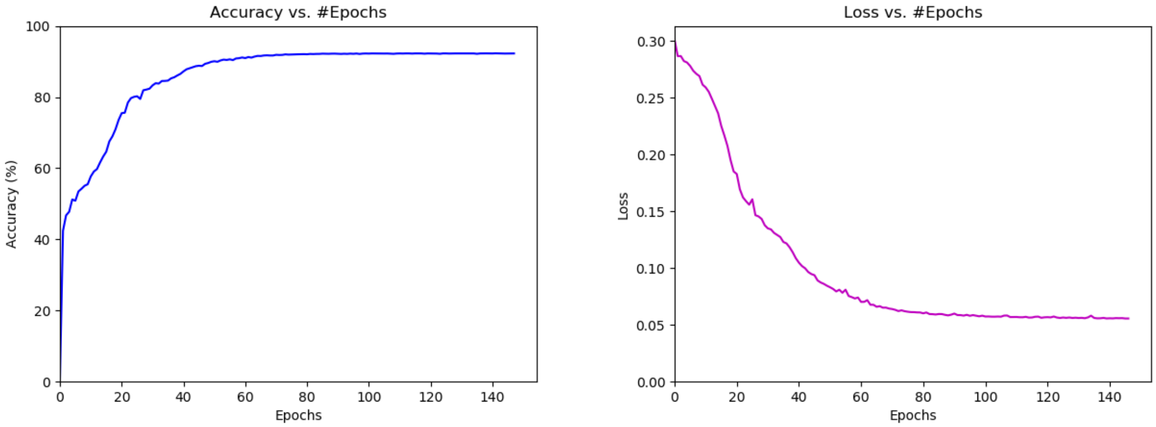
2.1 Preprocessing

Each categorial feature was one-hot-encoded and saved in the processed file.

2.2 Neural Network Architecture

A fully connected neural network was implemented using SGD, MSE loss and sigmoid activations.

Results of training are shown in Fig. 4 and Fig. 5



(a) Accuracy

(b) Loss

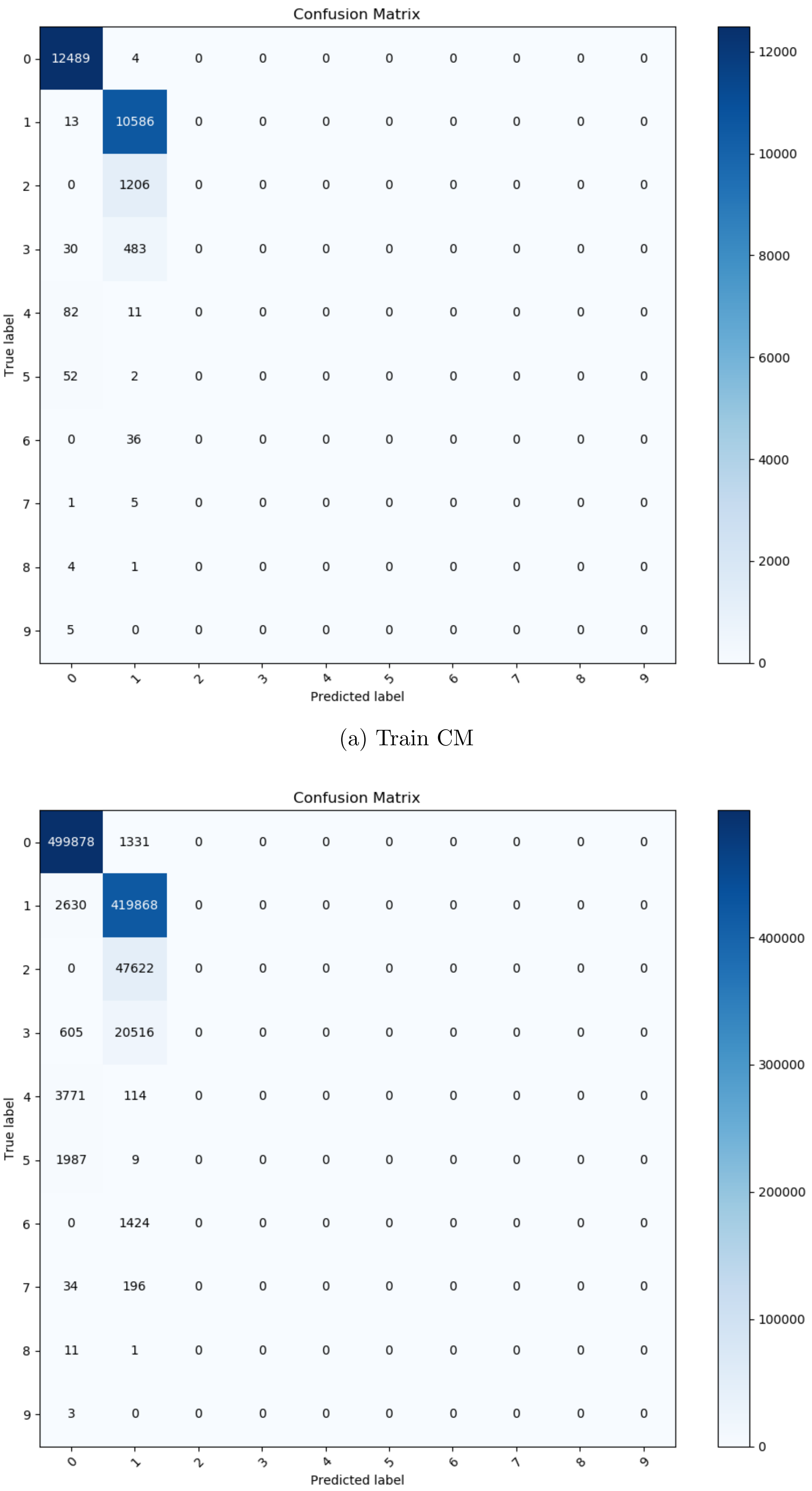
Figure 4: Training Accuracy and Loss

The following parameters were used in the training:

- Input Layer Size: 85
- Output Layer Size: 10
- Number of Hidden Layers: 1
- Hidden Layer Sizes: 20
- Learning Rate: 0.1
- Batch Size: 1

Result of training and testing

- Training Accuracy: 92.26 %
- Testing Accuracy: 91.97 %
- Time Taken: 515 secs



	0	1	2	3	4	5	6	7	8	9
0	12489	4	0	0	0	0	0	0	0	0
1	13	10586	0	0	0	0	0	0	0	0
2	0	1206	0	0	0	0	0	0	0	0
3	30	483	0	0	0	0	0	0	0	0
4	82	11	0	0	0	0	0	0	0	0
5	52	2	0	0	0	0	0	0	0	0
6	0	36	0	0	0	0	0	0	0	0
7	1	5	0	0	0	0	0	0	0	0
8	4	1	0	0	0	0	0	0	0	0
9	5	0	0	0	0	0	0	0	0	0

(a) Train CM

	0	1	2	3	4	5	6	7	8	9
0	499878	1331	0	0	0	0	0	0	0	0
1	2630	419868	0	0	0	0	0	0	0	0
2	0	47622	0	0	0	0	0	0	0	0
3	605	20516	0	0	0	0	0	0	0	0
4	3771	114	0	0	0	0	0	0	0	0
5	1987	9	0	0	0	0	0	0	0	0
6	0	1424	0	0	0	0	0	0	0	0
7	34	196	0	0	0	0	0	0	0	0
8	11	1	0	0	0	0	0	0	0	0
9	3	0	0	0	0	0	0	0	0	0

(b) Test CM

Figure 5: Train and Test Confusion Matrices

2.3 Single Hidden Layers

The network was trained with a single hidden layer with the size from the set {5, 10, 15, 20, 25}.

Here are the metrics:

Size	Train Acc.	Test Acc.	Time Taken
5	69.01	67.31	555.44
10	78.77	77.38	472.89
15	82.58	80.35	180.06
20	90.73	89.93	189.71
25	92.30	92.19	629.61

Plot of the metrics is shown in Fig. 6.

Confusion Matrices of the above are shown in Fig. 7 and are available in the Q2/plots/PartC folder.

Observations

1. Accuracy increase with larger size of the hidden layer
2. With size = 5, most examples are classified in class 0 only, but with larger class size the model learns to predict class 1 too.
3. The data for other classes is too low and the dataset too skewed for it to be able to learn the representation for the other classes.

2.4 Two Hidden Layers

The network was trained with two hidden layers with the size from the set {5, 10, 15, 20, 25}. Both layers had the same size.

Here are the metrics:

Size	Train Acc.	Test Acc.	Time Taken
5	58.37	56.12	115.85
10	85.09	84.07	1013.93
15	81.87	79.61	354.33
20	91.40	90.79	228.04
25	92.29	92.06	381.41

Plot of the metrics is shown in Fig. 8.

Confusion Matrices of the above are shown in Fig. 9 and are available in the Q2/plots/PartD folder.

Observations

1. The accuracy increases marginally over single hidden layer.

2.5 Adaptive Learning Rate

The network was trained with a single hidden layer with the size from the set {5, 10, 15, 20, 25}. The training is expensive and so for comparision purposes network was trained using single layer only.

Here are the metrics:

Size	**Train Acc.**	**Test Acc.**	**Time Taken**
5	68.36	66.88	262.30
10	76.80	75.59	299.55
15	88.06	86.82	406.45
20	92.25	92.08	253.65
25	92.28	92.13	300.02

Plot of the metrics is shown in Fig. 10.

Confusion Matrices of the above are shown in Fig. 11 and are available in the Q2/plots/PartE folder.

Observations

1. The accuracies increases marginally.
2. This happens as the model tries to get closer and closer to the local minima it is on, by decreasing the learning rate.

2.6 RELU Activation

The network was trained with a single hidden layer with the size from the set {5, 10, 15, 20, 25}. The training is expensive and so for comparision purposes network was trained using single layer only. RELU was used as the activation function in all the layers except the last layer where sigmoid was used.

Here are the metrics:

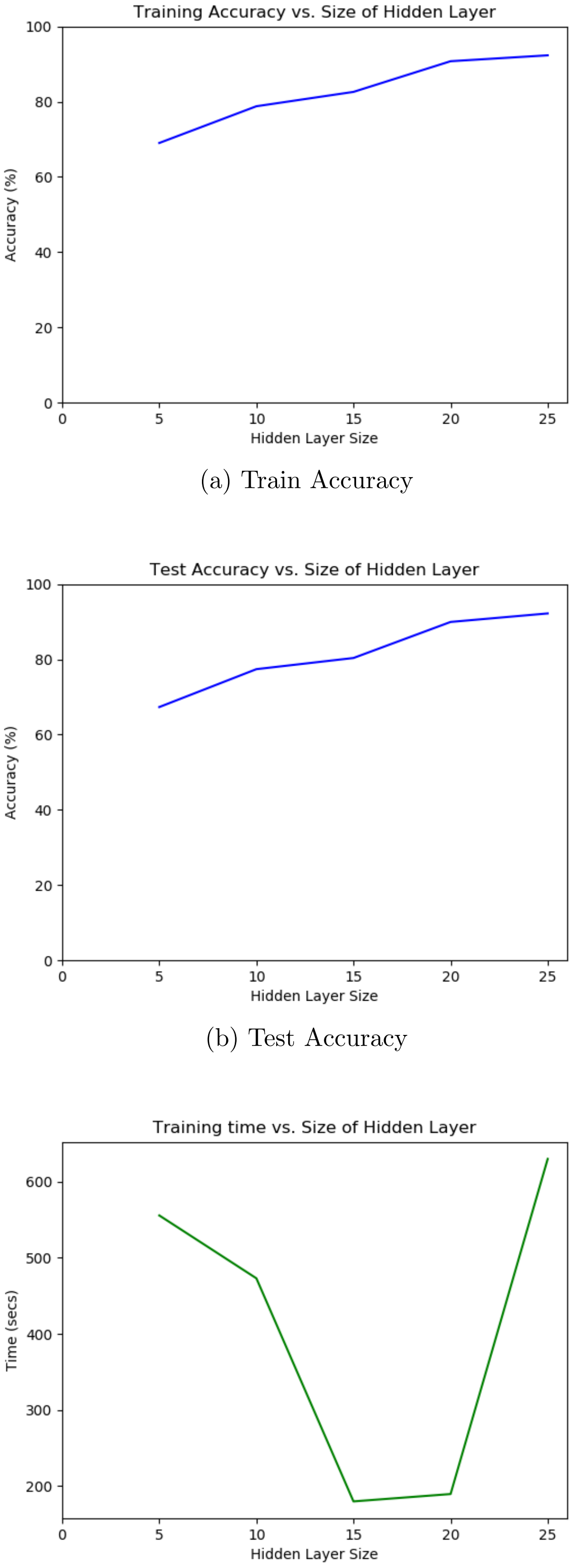
Size	**Train Acc.**	**Test Acc.**	**Time Taken**
5	57.72	56.61	105.58
10	60.37	56.52	200.19
15	82.99	80.54	453.32
20	87.49	85.50	285.86
25	90.73	89.63	466.99

Plot of the metrics is shown in Fig. 12.

Confusion Matrices of the above are shown in Fig. 13 and are available in the Q2/plots/PartF folder.

Observations

1. The model converges in less number of epochs and trains much faster as compared to the sigmoid activation.
2. It gives similar results on train and test accuracies. (Marginal lower accuracy)

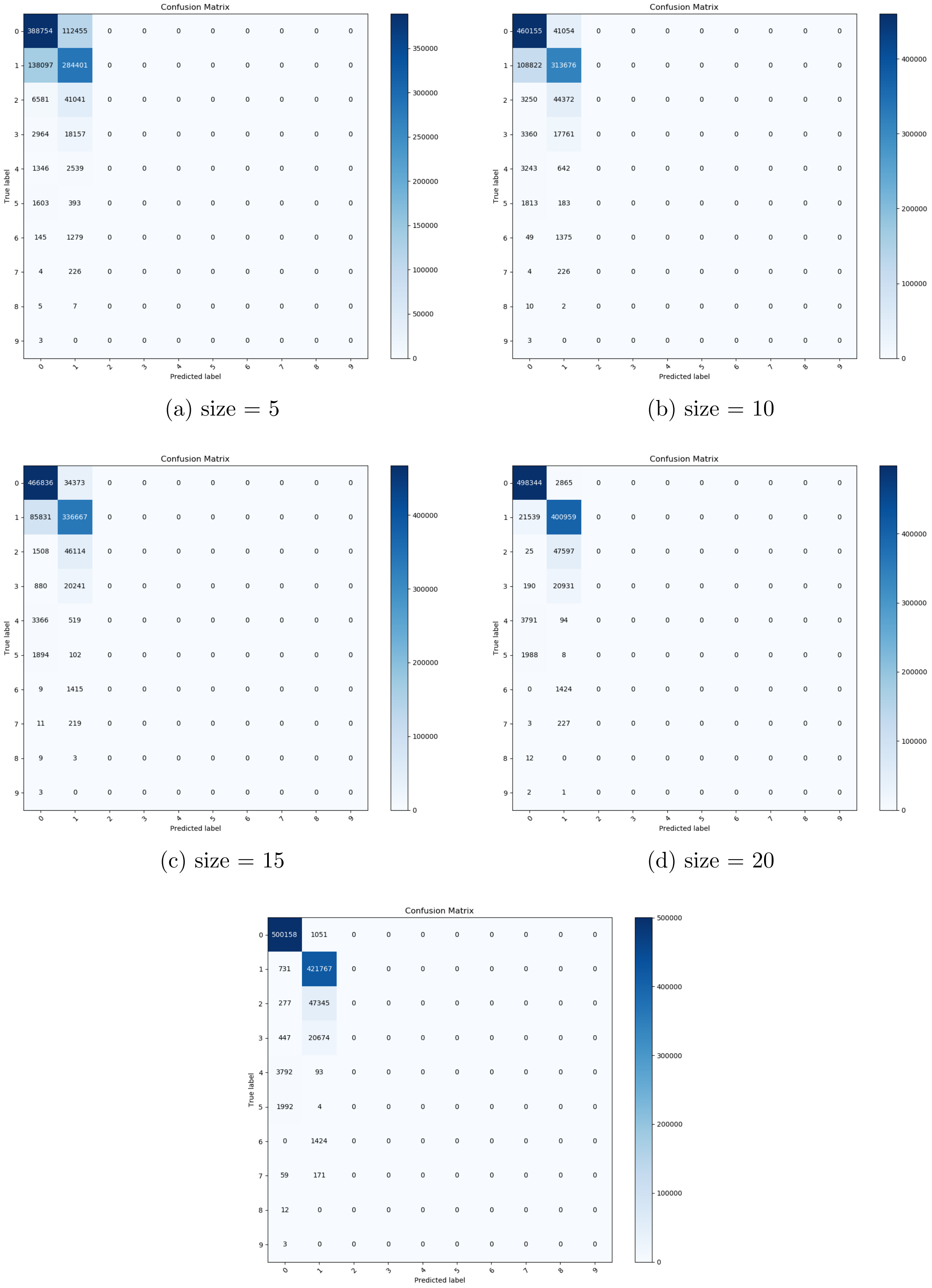


(a) Train Accuracy

(b) Test Accuracy

(c) Train Time

Figure 6: Metrics – Part C



	0	1	2	3	4	5	6	7	8	9
0	388754	112455	0	0	0	0	0	0	0	0
1	138097	284401	0	0	0	0	0	0	0	0
2	6581	41041	0	0	0	0	0	0	0	0
3	2964	18157	0	0	0	0	0	0	0	0
4	1346	2539	0	0	0	0	0	0	0	0
5	1603	393	0	0	0	0	0	0	0	0
6	145	1279	0	0	0	0	0	0	0	0
7	4	226	0	0	0	0	0	0	0	0
8	5	7	0	0	0	0	0	0	0	0
9	3	0	0	0	0	0	0	0	0	0

(a) size = 5

	0	1	2	3	4	5	6	7	8	9
0	460155	41054	0	0	0	0	0	0	0	0
1	108822	313676	0	0	0	0	0	0	0	0
2	3250	44372	0	0	0	0	0	0	0	0
3	3360	17761	0	0	0	0	0	0	0	0
4	3243	642	0	0	0	0	0	0	0	0
5	1813	183	0	0	0	0	0	0	0	0
6	49	1375	0	0	0	0	0	0	0	0
7	4	226	0	0	0	0	0	0	0	0
8	10	2	0	0	0	0	0	0	0	0
9	3	0	0	0	0	0	0	0	0	0

(b) size = 10

	0	1	2	3	4	5	6	7	8	9
0	466836	34373	0	0	0	0	0	0	0	0
1	85831	336667	0	0	0	0	0	0	0	0
2	1508	46114	0	0	0	0	0	0	0	0
3	880	20241	0	0	0	0	0	0	0	0
4	3366	519	0	0	0	0	0	0	0	0
5	1894	102	0	0	0	0	0	0	0	0
6	9	1415	0	0	0	0	0	0	0	0
7	11	219	0	0	0	0	0	0	0	0
8	9	3	0	0	0	0	0	0	0	0
9	3	0	0	0	0	0	0	0	0	0

(c) size = 15

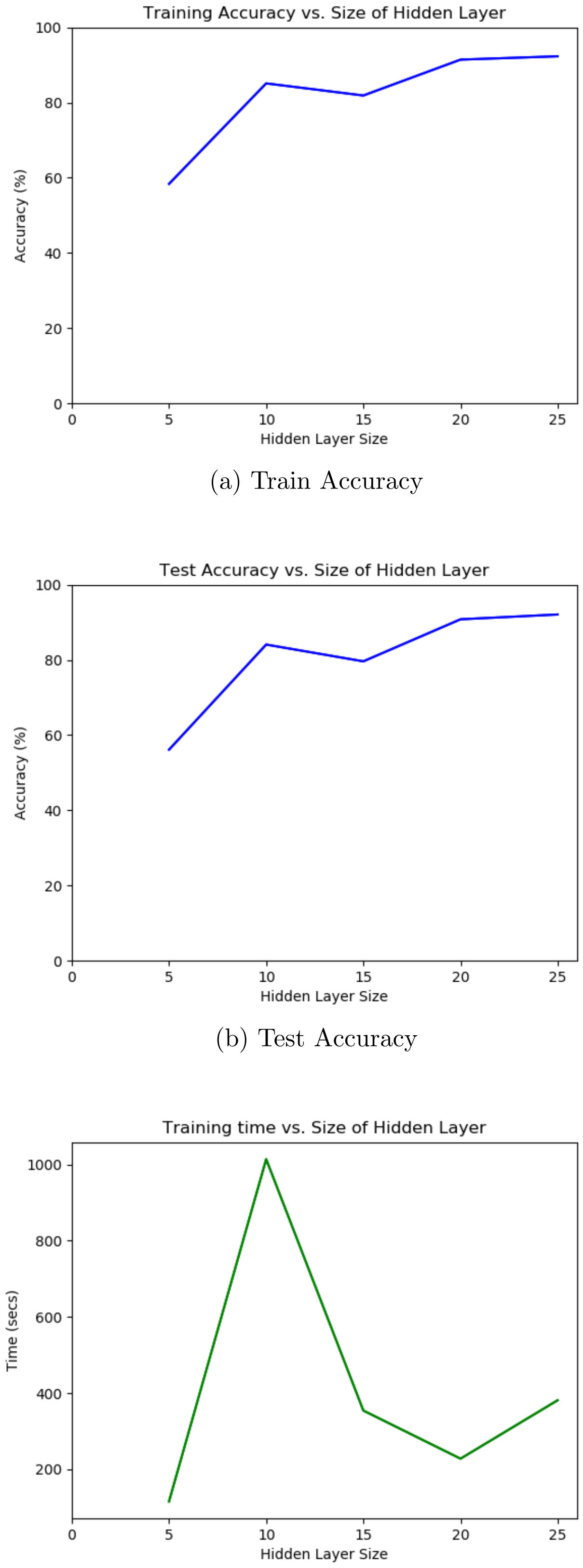
	0	1	2	3	4	5	6	7	8	9
0	498344	2865	0	0	0	0	0	0	0	0
1	21539	400959	0	0	0	0	0	0	0	0
2	25	47597	0	0	0	0	0	0	0	0
3	190	20931	0	0	0	0	0	0	0	0
4	3791	94	0	0	0	0	0	0	0	0
5	1988	8	0	0	0	0	0	0	0	0
6	0	1424	0	0	0	0	0	0	0	0
7	3	227	0	0	0	0	0	0	0	0
8	12	0	0	0	0	0	0	0	0	0
9	2	1	0	0	0	0	0	0	0	0

(d) size = 20

	0	1	2	3	4	5	6	7	8	9
0	500158	1051	0	0	0	0	0	0	0	0
1	731	421767	0	0	0	0	0	0	0	0
2	277	47345	0	0	0	0	0	0	0	0
3	447	20674	0	0	0	0	0	0	0	0
4	3792	93	0	0	0	0	0	0	0	0
5	1992	4	0	0	0	0	0	0	0	0
6	0	1424	0	0	0	0	0	0	0	0
7	59	171	0	0	0	0	0	0	0	0
8	12	0	0	0	0	0	0	0	0	0
9	3	0	0	0	0	0	0	0	0	0

(e) size = 25

Figure 7: Confusion Matrices – Part C

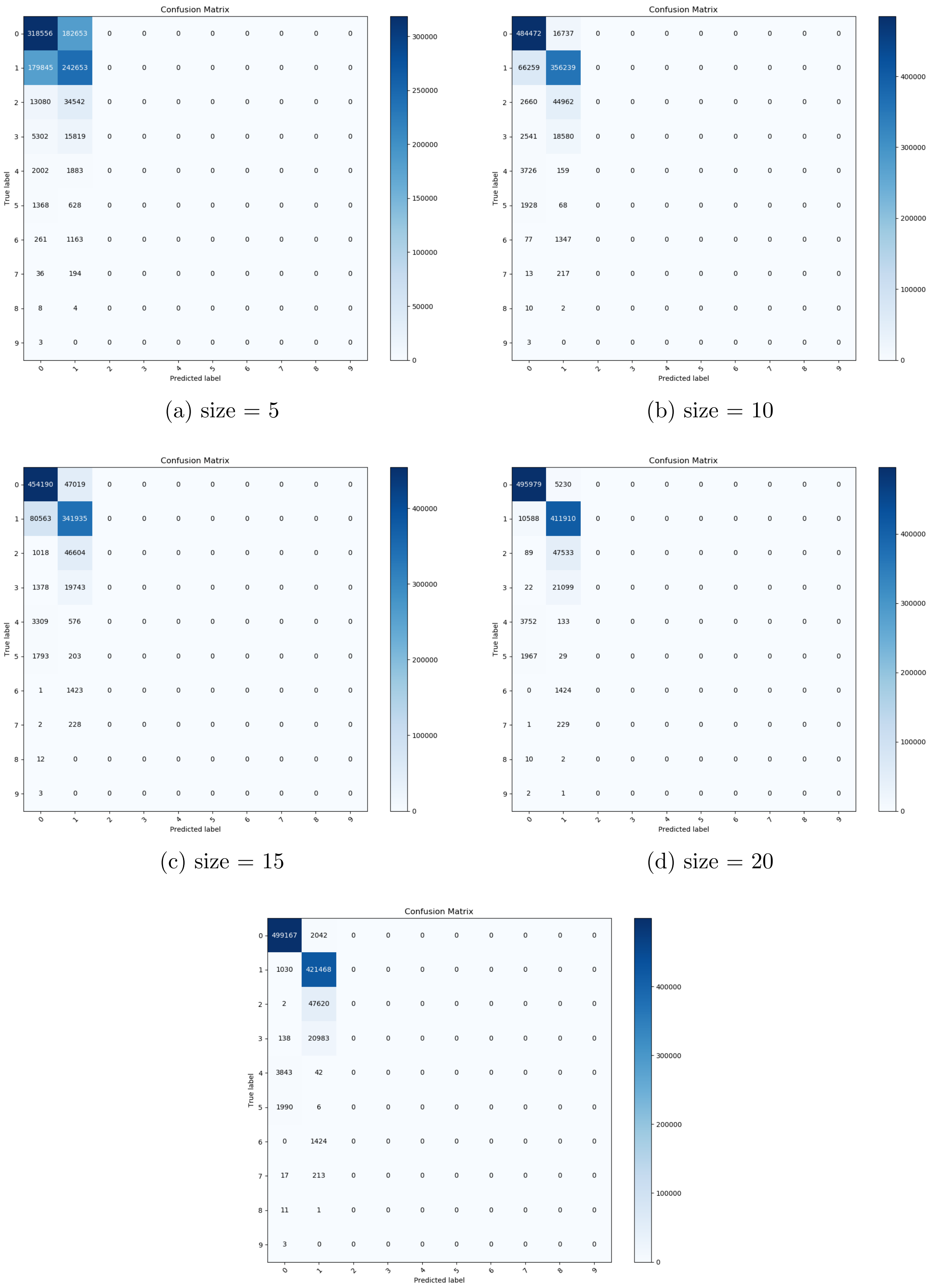


(a) Train Accuracy

(b) Test Accuracy

(c) Train Time

Figure 8: Metrics – Part D



(a) size = 5

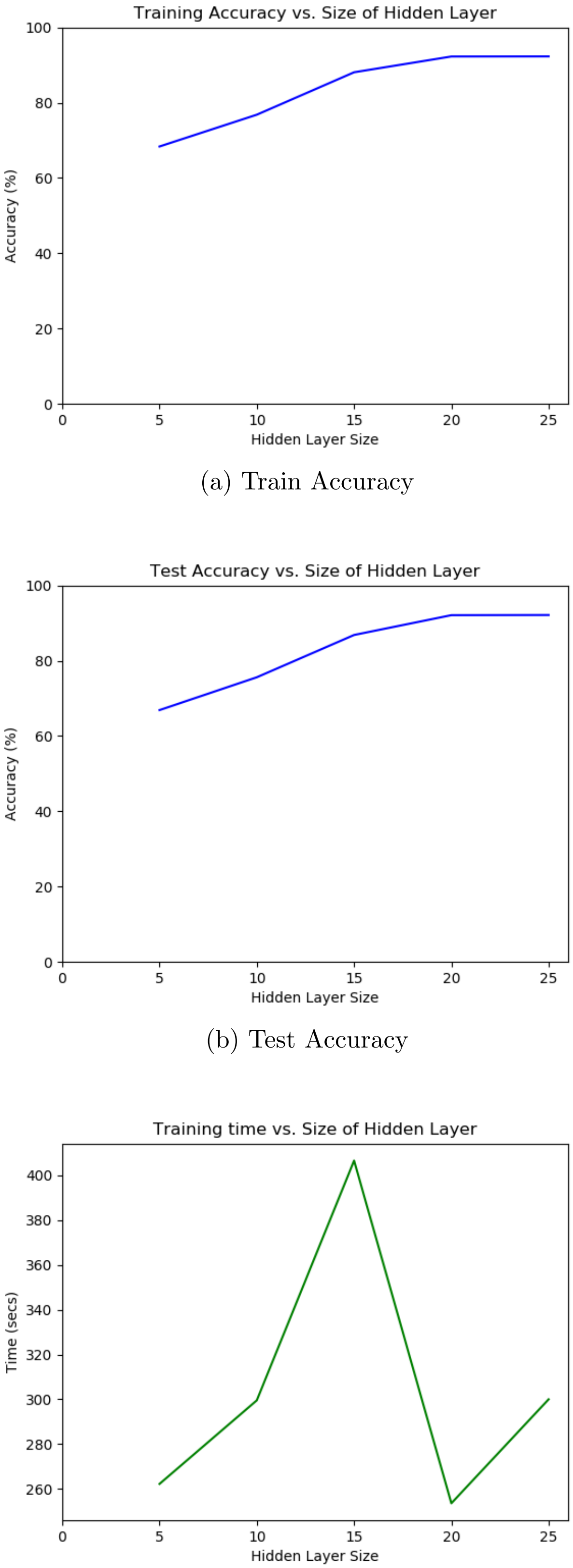
(b) size = 10

(c) size = 15

(d) size = 20

(e) size = 25

Figure 9: Confusion Matrices – Part D

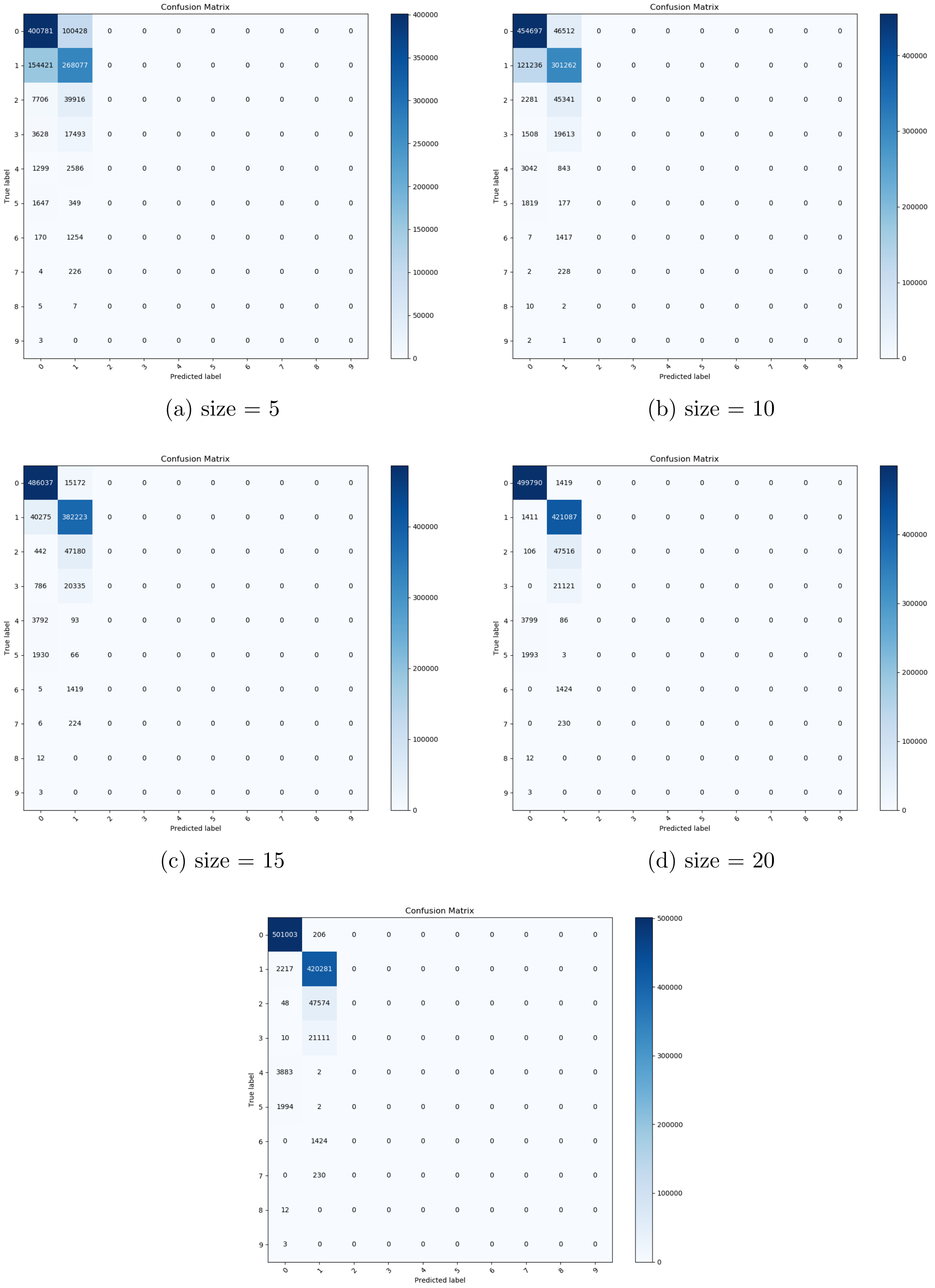


(a) Train Accuracy

(b) Test Accuracy

(c) Train Time

Figure 10: Metrics – Part E



(a) size = 5

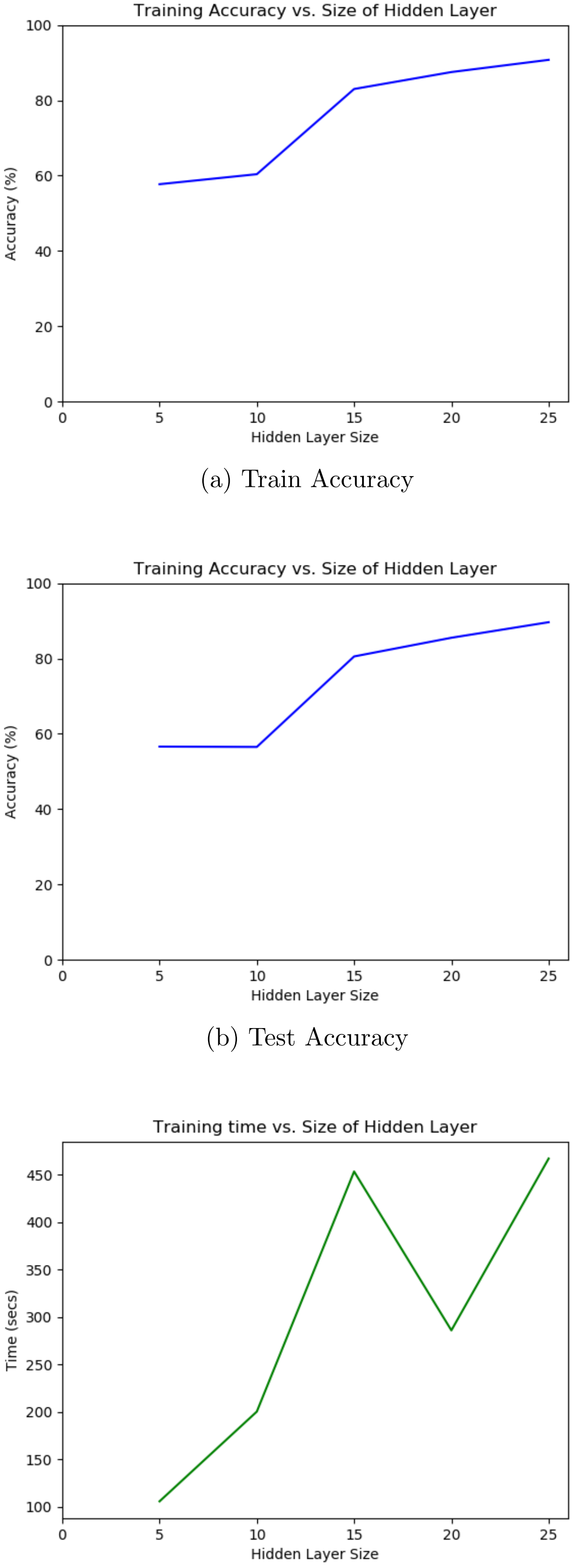
(b) size = 10

(c) size = 15

(d) size = 20

(e) size = 25

Figure 11: Confusion Matrices – Part E

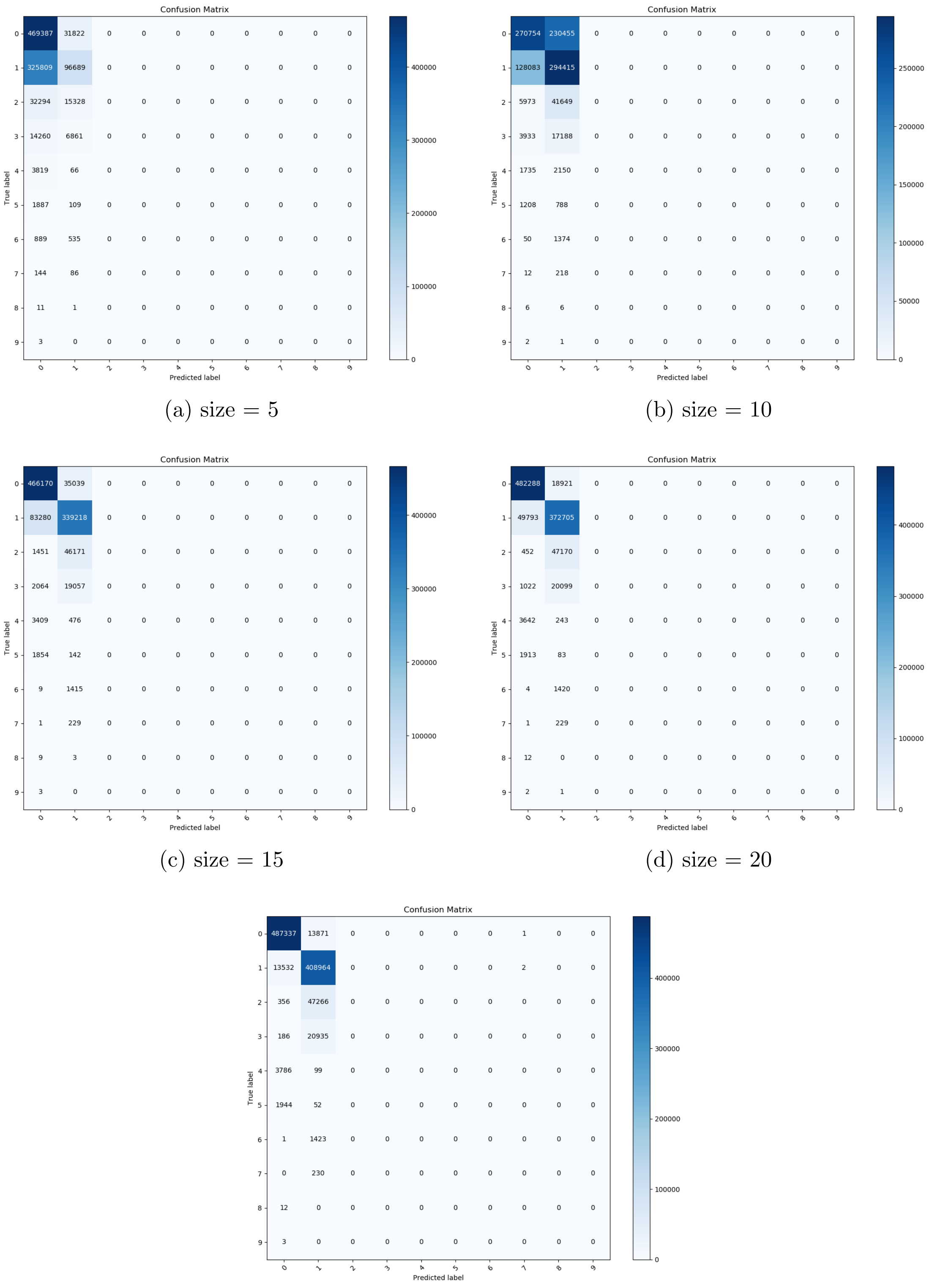


(a) Train Accuracy

(b) Test Accuracy

(c) Train Time

Figure 12: Metrics – Part R



	0	1	2	3	4	5	6	7	8	9
0	469387	31822	0	0	0	0	0	0	0	0
1	325809	96689	0	0	0	0	0	0	0	0
2	32294	15328	0	0	0	0	0	0	0	0
3	14260	6861	0	0	0	0	0	0	0	0
4	3819	66	0	0	0	0	0	0	0	0
5	1887	109	0	0	0	0	0	0	0	0
6	889	535	0	0	0	0	0	0	0	0
7	144	86	0	0	0	0	0	0	0	0
8	11	1	0	0	0	0	0	0	0	0
9	3	0	0	0	0	0	0	0	0	0

(a) size = 5

	0	1	2	3	4	5	6	7	8	9
0	270754	230455	0	0	0	0	0	0	0	0
1	128083	294415	0	0	0	0	0	0	0	0
2	5973	41649	0	0	0	0	0	0	0	0
3	3933	17188	0	0	0	0	0	0	0	0
4	1735	2150	0	0	0	0	0	0	0	0
5	1208	788	0	0	0	0	0	0	0	0
6	50	1374	0	0	0	0	0	0	0	0
7	12	218	0	0	0	0	0	0	0	0
8	6	6	0	0	0	0	0	0	0	0
9	2	1	0	0	0	0	0	0	0	0

(b) size = 10

	0	1	2	3	4	5	6	7	8	9
0	466170	35039	0	0	0	0	0	0	0	0
1	83280	339218	0	0	0	0	0	0	0	0
2	1451	46171	0	0	0	0	0	0	0	0
3	2064	19057	0	0	0	0	0	0	0	0
4	3409	476	0	0	0	0	0	0	0	0
5	1854	142	0	0	0	0	0	0	0	0
6	9	1415	0	0	0	0	0	0	0	0
7	1	229	0	0	0	0	0	0	0	0
8	9	3	0	0	0	0	0	0	0	0
9	3	0	0	0	0	0	0	0	0	0

(c) size = 15

	0	1	2	3	4	5	6	7	8	9
0	482288	18921	0	0	0	0	0	0	0	0
1	49793	372705	0	0	0	0	0	0	0	0
2	452	47170	0	0	0	0	0	0	0	0
3	1022	20099	0	0	0	0	0	0	0	0
4	3642	243	0	0	0	0	0	0	0	0
5	1913	83	0	0	0	0	0	0	0	0
6	4	1420	0	0	0	0	0	0	0	0
7	1	229	0	0	0	0	0	0	0	0
8	12	0	0	0	0	0	0	0	0	0
9	2	1	0	0	0	0	0	0	0	0

(d) size = 20

	0	1	2	3	4	5	6	7	8	9
0	487337	13871	0	0	0	0	0	1	0	0
1	13532	408964	0	0	0	0	0	2	0	0
2	356	47266	0	0	0	0	0	0	0	0
3	186	20935	0	0	0	0	0	0	0	0
4	3786	99	0	0	0	0	0	0	0	0
5	1944	52	0	0	0	0	0	0	0	0
6	1	1423	0	0	0	0	0	0	0	0
7	0	230	0	0	0	0	0	0	0	0
8	12	0	0	0	0	0	0	0	0	0
9	3	0	0	0	0	0	0	0	0	0

(e) size = 25

Figure 13: Confusion Matrices – Part F