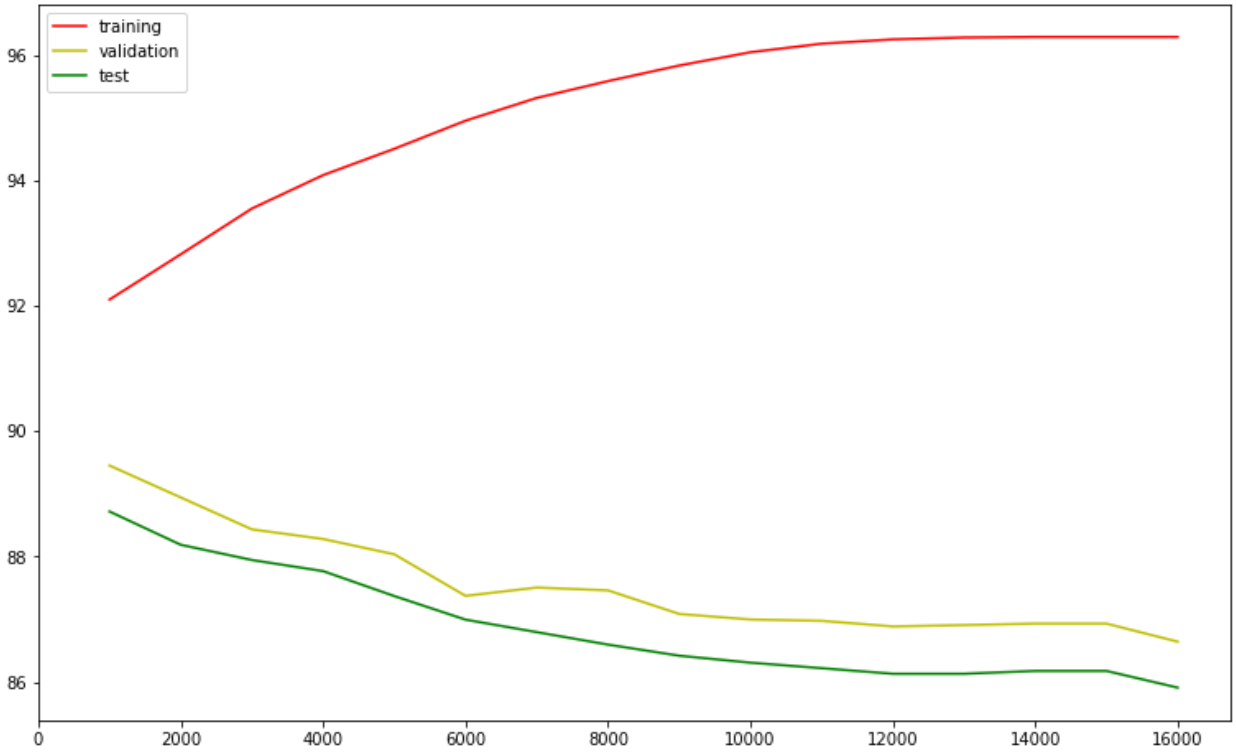
COL772

ASSIGNMENT - 3

QUESTION 1

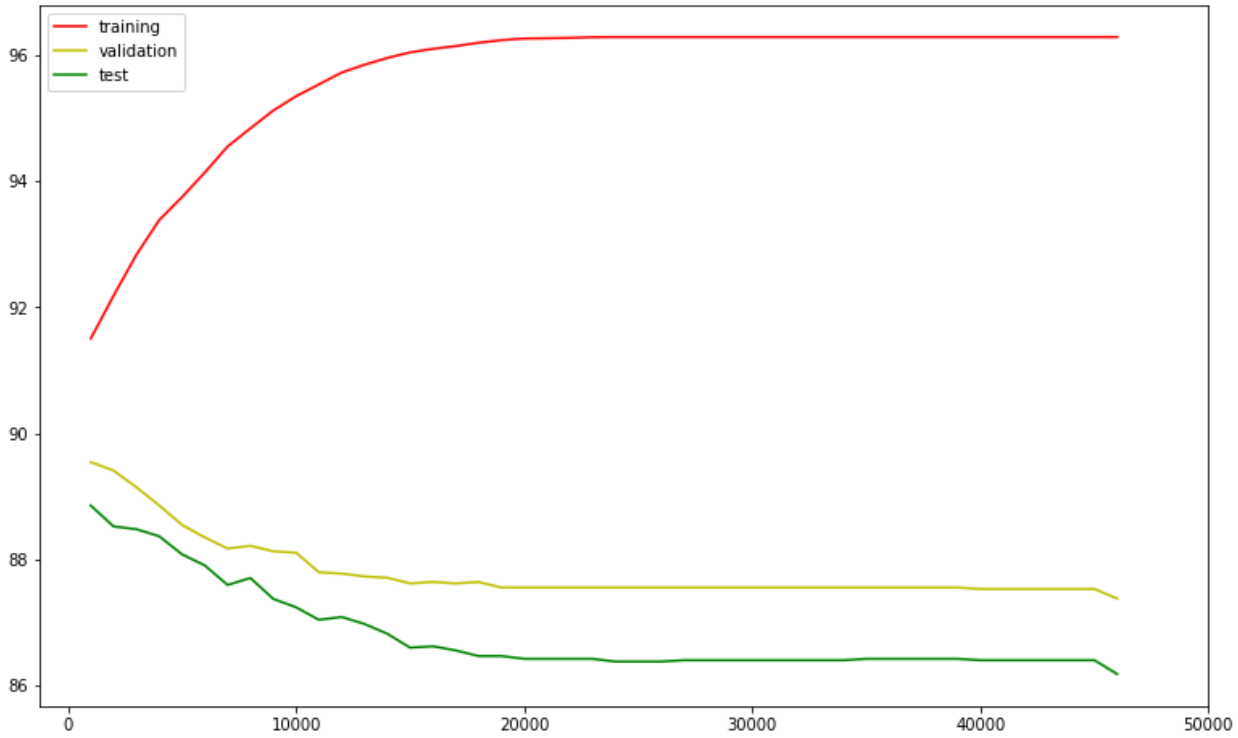
Part A

1. Decision Tree Construction



- Training accuracy : 96.287
- Validation accuracy : 86.643
- Test accuracy : 85.91

2. Decision Tree with one hot encoded features



- Training accuracy : 96.287
- Validation accuracy : 87.528
- Test accuracy : 86.397

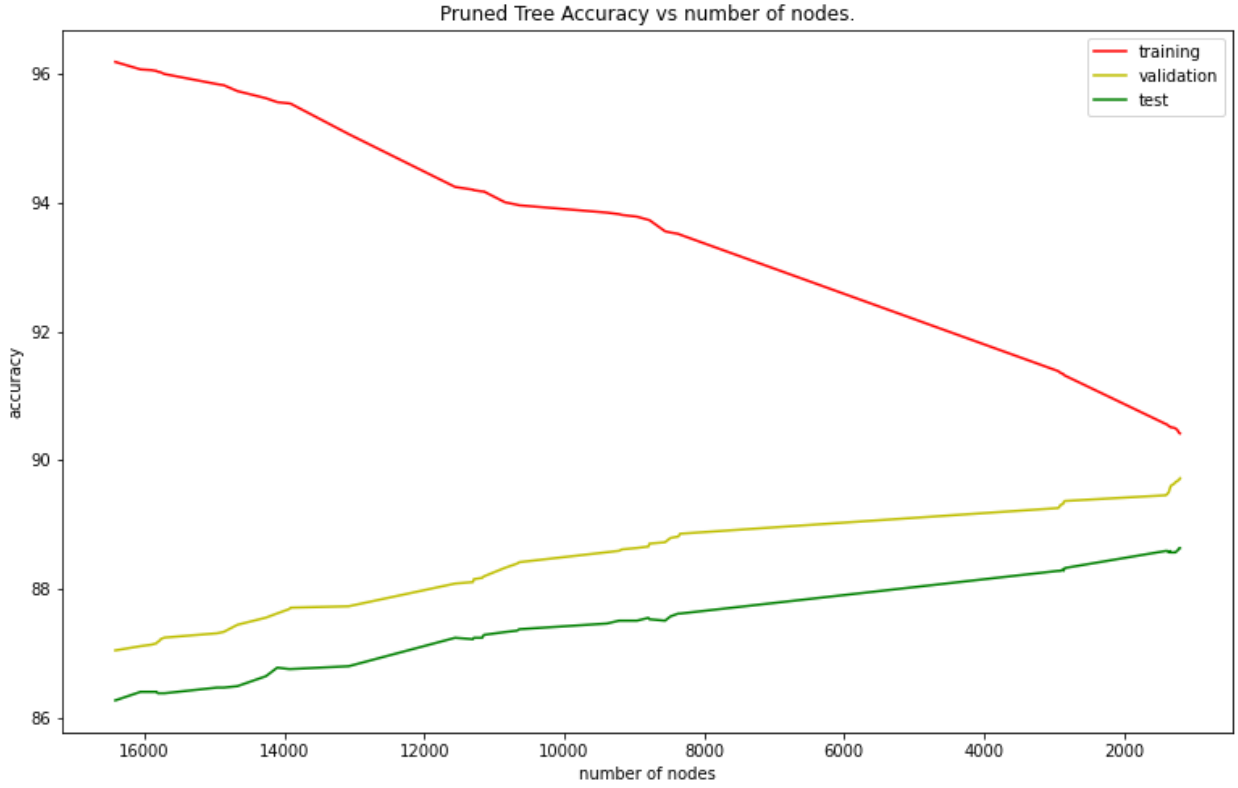
➔ It can be observed that with one hot encoding the number of node in trees almost doubles.

➔ Training accuracy remains same (because it is maximum bound on accuracy), but time for tree construction is higher then part A.

➔ But validation and testing accuracy is better for Part B which implies better generalisation.

Part B

Decision Tree Pruning



- ➔ X-axis denotes number of nodes in tree (as pruning progresses number of nodes decreases)
- ➔ It can be observed that training accuracy decreases while validation and test accuracy increases as number of pruned nodes increase.
- ➔ So clearly pruning helps to reduce overfitting in decision trees.

Part C

Here we have to decide on optimal parameters. So for all combinations of parameter values, oob score, training, validation and test accuracies were calculated.

After that for all parameters we select TOP 10/15 accuracy values and perform intersection on parameter values to get the best set of parameters.

a) Values sorted by **oob accuracy.**

	n_estimators	max_features	sample_split	train_acc	val_acc	test_acc	oob_score
104	450	0.1	10	92.413	89.540	89.294	89.9248
79	350	0.1	10	92.435	89.673	89.317	89.8971
4	50	0.1	10	92.355	89.628	89.427	89.8944
54	250	0.1	10	92.388	89.628	89.294	89.8723
109	450	0.3	10	93.074	90.093	89.250	89.8612
53	250	0.1	8	92.786	89.673	89.184	89.8557
29	150	0.1	10	92.366	89.695	89.361	89.8557
103	450	0.1	8	92.822	89.651	89.162	89.8308
84	350	0.3	10	93.124	89.938	89.073	89.8114
78	350	0.1	8	92.814	89.584	89.184	89.8114

b) Values sorted by column **validation accuracy**

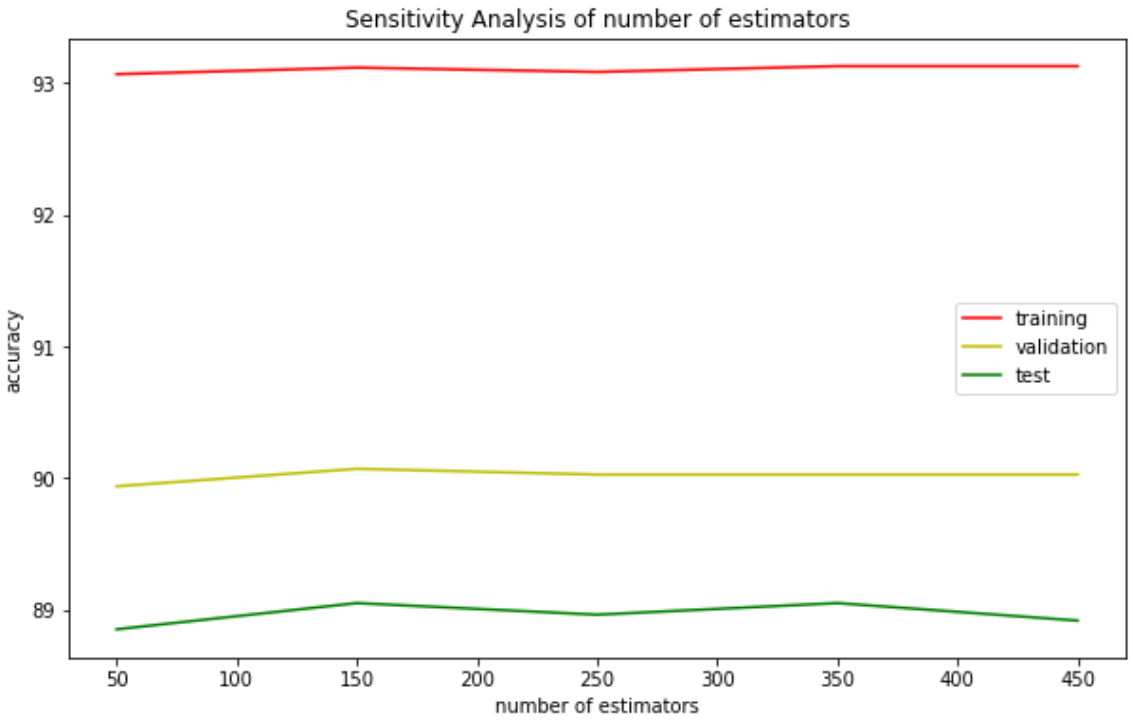
	n_estimators	max_features	sample_split	train_acc	val_acc	test_acc	oob_score
33	150	0.3	8	93.497	90.115	88.896	89.7285
109	450	0.3	10	93.074	90.093	89.250	89.8612
119	450	0.7	10	93.403	90.049	88.764	89.6870
73	250	0.9	8	93.881	90.049	88.653	89.4935
34	150	0.3	10	93.138	90.027	88.963	89.7672
49	150	0.9	10	93.472	90.004	88.808	89.5847
59	250	0.3	10	93.038	90.004	89.051	89.7866
69	250	0.7	10	93.392	89.960	88.719	89.6870
98	350	0.9	8	93.881	89.960	88.609	89.5820
93	350	0.7	8	93.845	89.938	88.631	89.5958
84	350	0.3	10	93.124	89.938	89.073	89.8114
19	50	0.7	10	93.345	89.938	88.675	89.5377
24	50	0.9	10	93.439	89.938	89.007	89.4962
124	450	0.9	10	93.436	89.916	88.830	89.5930
83	350	0.3	8	93.480	89.916	88.985	89.7091

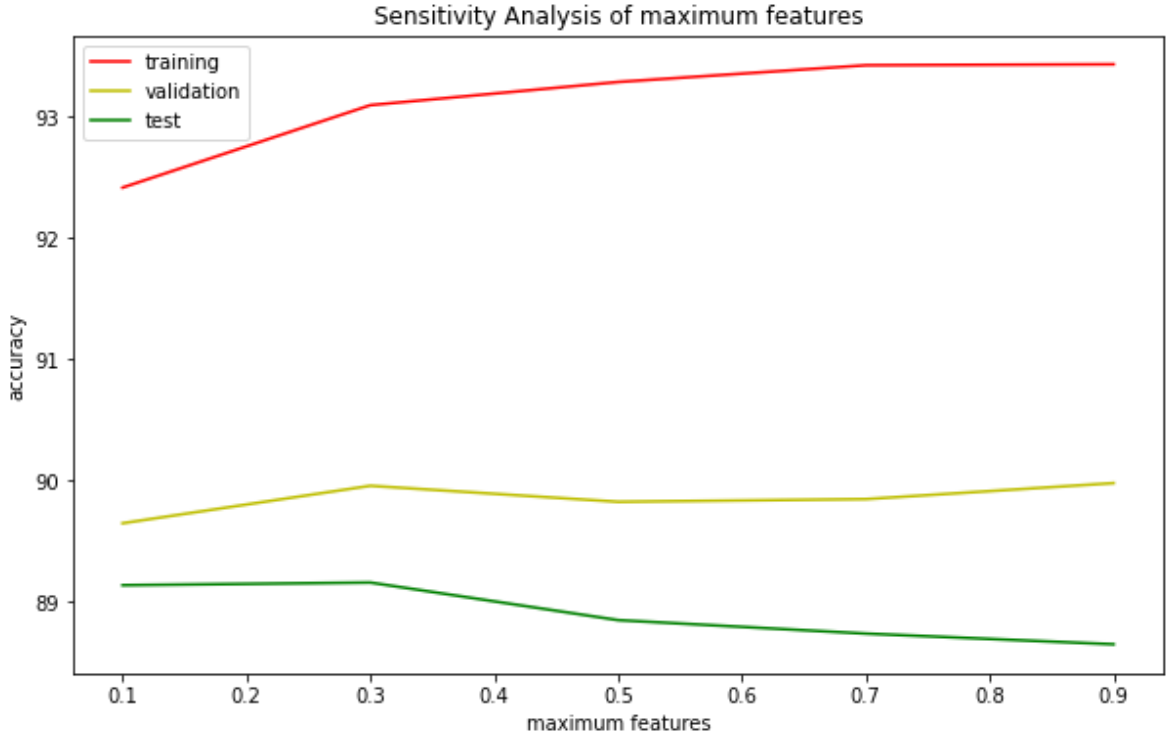
c) Values sorted by column **test accuracy.**

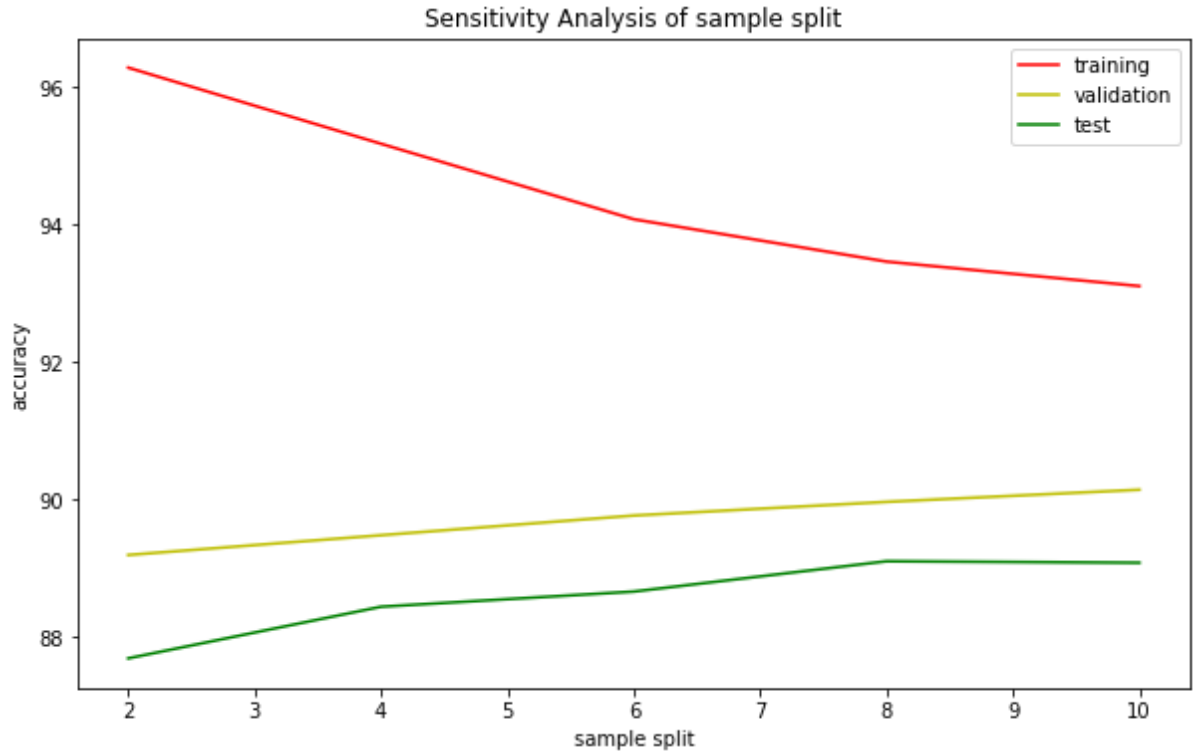
	n_estimators	max_features	sample_split	train_acc	val_acc	test_acc	oob_score
4	50	0.1	10	92.355	89.628	89.427	89.8944
29	150	0.1	10	92.366	89.695	89.361	89.8557
79	350	0.1	10	92.435	89.673	89.317	89.8971
54	250	0.1	10	92.388	89.628	89.294	89.8723
104	450	0.1	10	92.413	89.540	89.294	89.9248
109	450	0.3	10	93.074	90.093	89.250	89.8612
3	50	0.1	8	92.745	89.474	89.228	89.6677
9	50	0.3	10	93.041	89.828	89.206	89.7257
78	350	0.1	8	92.814	89.584	89.184	89.8114
53	250	0.1	8	92.786	89.673	89.184	89.8557
103	450	0.1	8	92.822	89.651	89.162	89.8308
28	150	0.1	8	92.806	89.739	89.162	89.8059
77	350	0.1	6	93.403	89.606	89.140	89.7340
102	450	0.1	6	93.339	89.673	89.095	89.7257
84	350	0.3	10	93.124	89.938	89.073	89.8114

It was observed from all accuracies that the combination at index “84” is best - as it appears in all 3 top values and the difference between accuracy and corresponding max accuracy values is almost same.

Part D


Sensitivity Analysis of number of estimators
accuracy
number of estimators
training
validation
test
93
92
91
90
89
50
100
150
200
250
300
350
400
450


Sensitivity Analysis of maximum features
accuracy
maximum features
training
validation
test
93
92
91
90
89
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9



- ➔ Accuracy for the number of estimators remains almost the same across all values with a small dip at around 250.
- ➔ Max features seem to overfit the decision tree as training accuracy increases while the test accuracy decreases. So the model is highly sensitive to this feature.
- ➔ Sample split seems to help the model for better generalisation, as observed from the graph. Model is also sensitive to Sample split feature.

QUESTION 2

Part A

One hot encoding of features.

Part B

Implemented Neural Network.

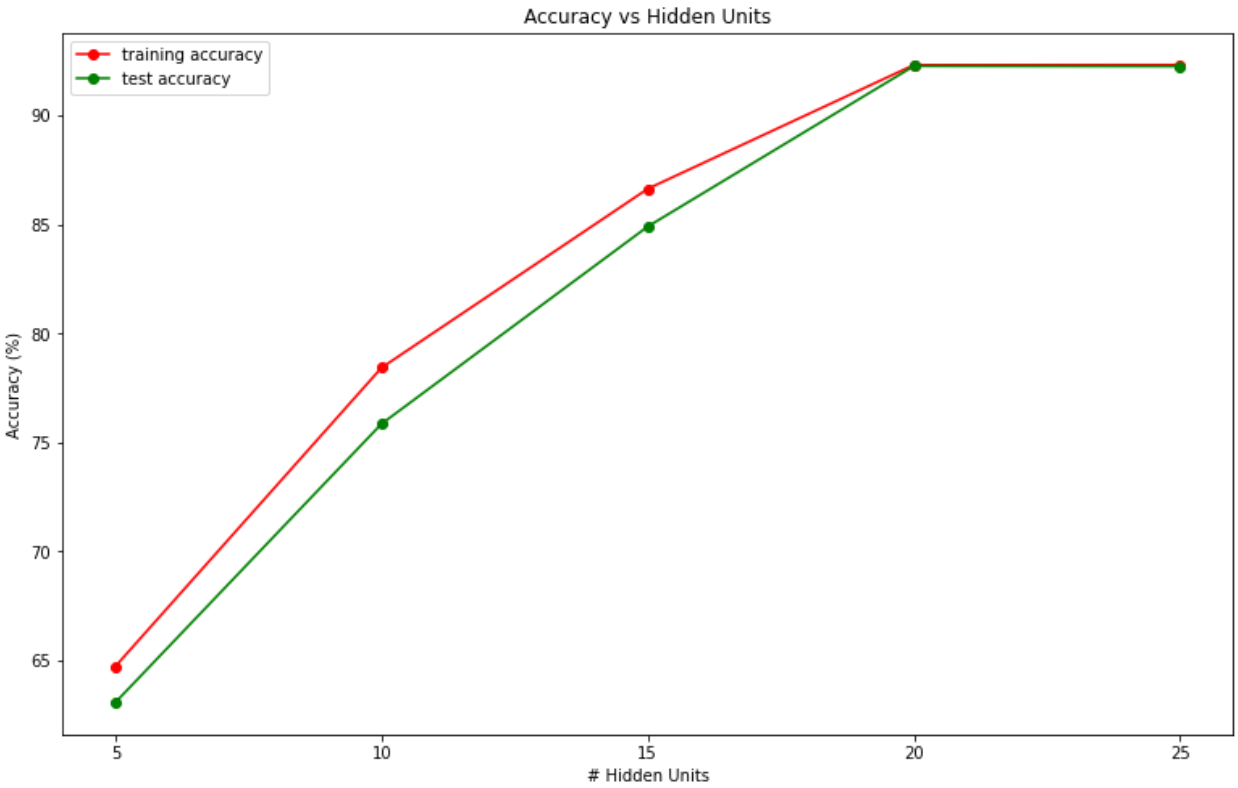
Part C

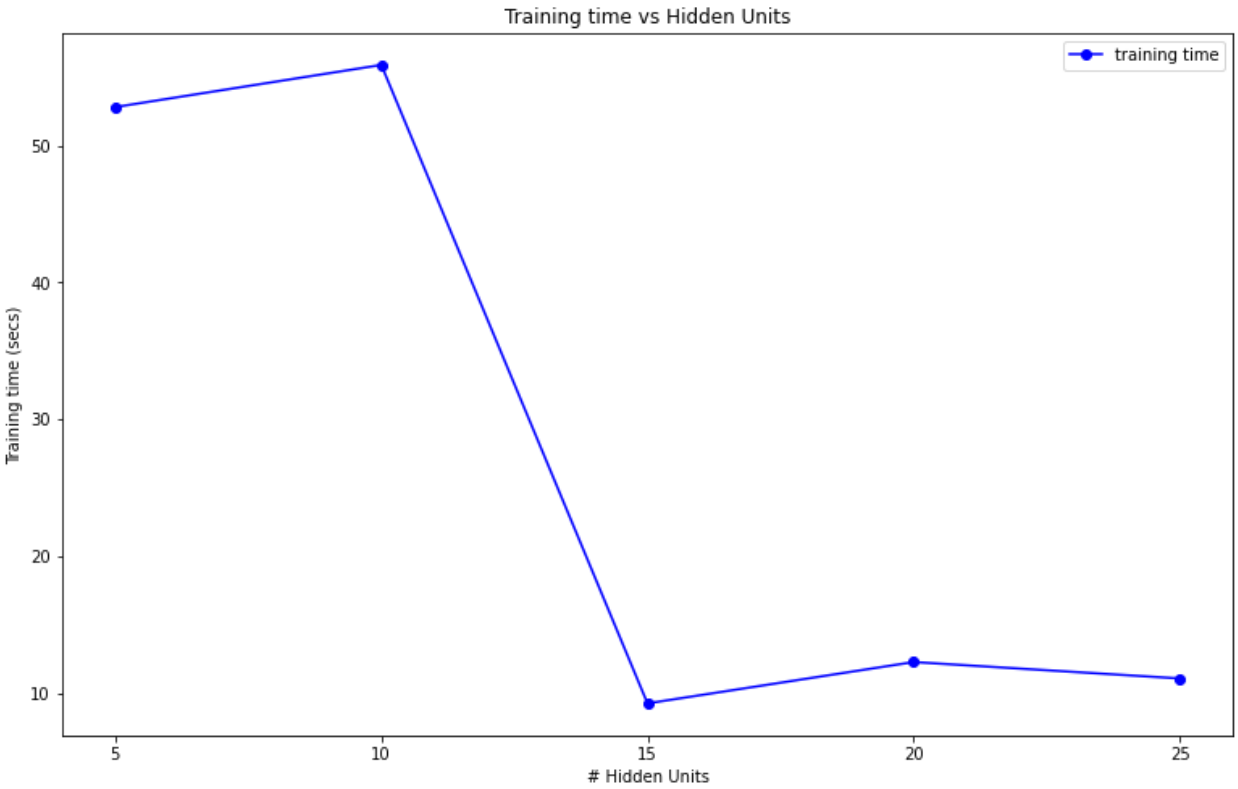
1. **Stopping criteria:**
    - **a.** Difference between loss of last epoch and average of loss of last 5 epochs, given loss in current epoch is less than 0.01.
    - b. Also tried with absolute stopping criteria, but the above one worked best for all cases as the data was also between 0 and 1.

2. **Testing model with different hidden units (single layer)**

# hidden units	training accuracy(%)	test accuracy(%)	training time (secs)
5	64.718	63.063	52.81
10	78.425	75.85	55.891
15	86.637	84.904	9.249
20	92.323	92.27	12.27
25	92.327	92.259	11.07

Plots added below.


Accuracy vs Hidden Units
training accuracy
test accuracy
Accuracy (%)
90
85
80
75
70
65
5
10
15
20
25
Hidden Units


Training time vs Hidden Units
training time
Training time (secs)
50
40
30
20
10
5
10
15
20
25
Hidden Units

3. Confusion Matrix

a. Hidden unit 5

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**422807**	78402	0	0	0	0	0	0	0	0
	1	214952	**207546**	0	0	0	0	0	0	0	0
	2	13979	33643	0	0	0	0	0	0	0	0
	3	5184	15662	0	275	0	0	0	0	0	0
	4	3059	826	0	0	0	0	0	0	0	0
	5	1686	310	0	0	0	0	0	0	0	0
	6	308	1116	0	0	0	0	0	0	0	0
	7	8	211	0	11	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. Hidden Unit 10

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**458618**	42591	0	0	0	0	0	0	0	0
	1	122620	**299878**	0	0	0	0	0	0	0	0
	2	2136	45486	0	0	0	0	0	0	0	0
	3	3565	17556	0	0	0	0	0	0	0	0
	4	2959	926	0	0	0	0	0	0	0	0
	5	1853	143	0	0	0	0	0	0	0	0
	6	30	1394	0	0	0	0	0	0	0	0
	7	11	219	0	0	0	0	0	0	0	0
	8	7	5	0	0	0	0	0	0	0	0
	9	3	0	0	0	0	0	0	0	0	0

c. Hidden Unit 15

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**482837**	18372	0	0	0	0	0	0	0	0
	1	56297	**366201**	0	0	0	0	0	0	0	0
	2	709	46913	0	0	0	0	0	0	0	0
	3	726	20395	0	0	0	0	0	0	0	0
	4	3612	273	0	0	0	0	0	0	0	0
	5	1906	90	0	0	0	0	0	0	0	0
	6	3	1421	0	0	0	0	0	0	0	0
	7	10	220	0	0	0	0	0	0	0	0
	8	9	3	0	0	0	0	0	0	0	0
	9	2	1	0	0	0	0	0	0	0	0

d. Hidden Unit 20

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**500804**	405	0	0	0	0	0	0	0	0
	1	599	**421899**	0	0	0	0	0	0	0	0
	2	55	47567	0	0	0	0	0	0	0	0
	3	1641	19480	0	0	0	0	0	0	0	0
	4	3707	178	0	0	0	0	0	0	0	0
	5	1991	5	0	0	0	0	0	0	0	0
	6	0	1424	0	0	0	0	0	0	0	0
	7	58	172	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

e. Hidden Unit 25

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**500988**	221	0	0	0	0	0	0	0	0
	1	895	**421603**	0	0	0	0	0	0	0	0
	2	170	47452	0	0	0	0	0	0	0	0
	3	1240	19881	0	0	0	0	0	0	0	0
	4	3726	159	0	0	0	0	0	0	0	0
	5	1994	2	0	0	0	0	0	0	0	0
	6	0	1424	0	0	0	0	0	0	0	0
	7	96	134	0	0	0	0	0	0	0	0
	8	11	1	0	0	0	0	0	0	0	0
	9	2	1	0	0	0	0	0	0	0	0

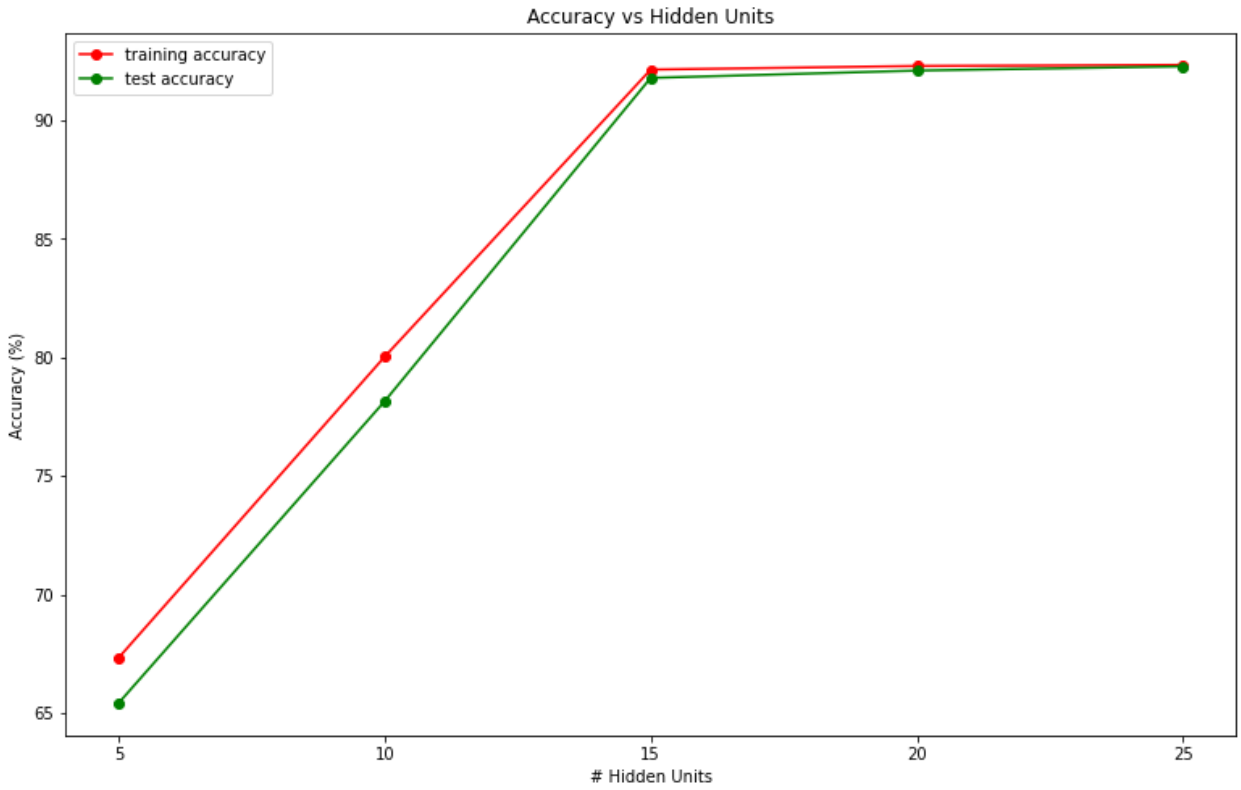
- ➔ It can be seen that accuracy increases with number of units and hence the total number of correct predictions for class 0 and 1 also increases.
- ➔ For some classes eg. class 7 - mis classification as 0 increases.
- ➔ For some it reduces drastically and then increases by small amounts eg class 2.
- ➔ Also it can be observed that if mis classification of a class as 0 increases then mis classification as 1 will decrease.

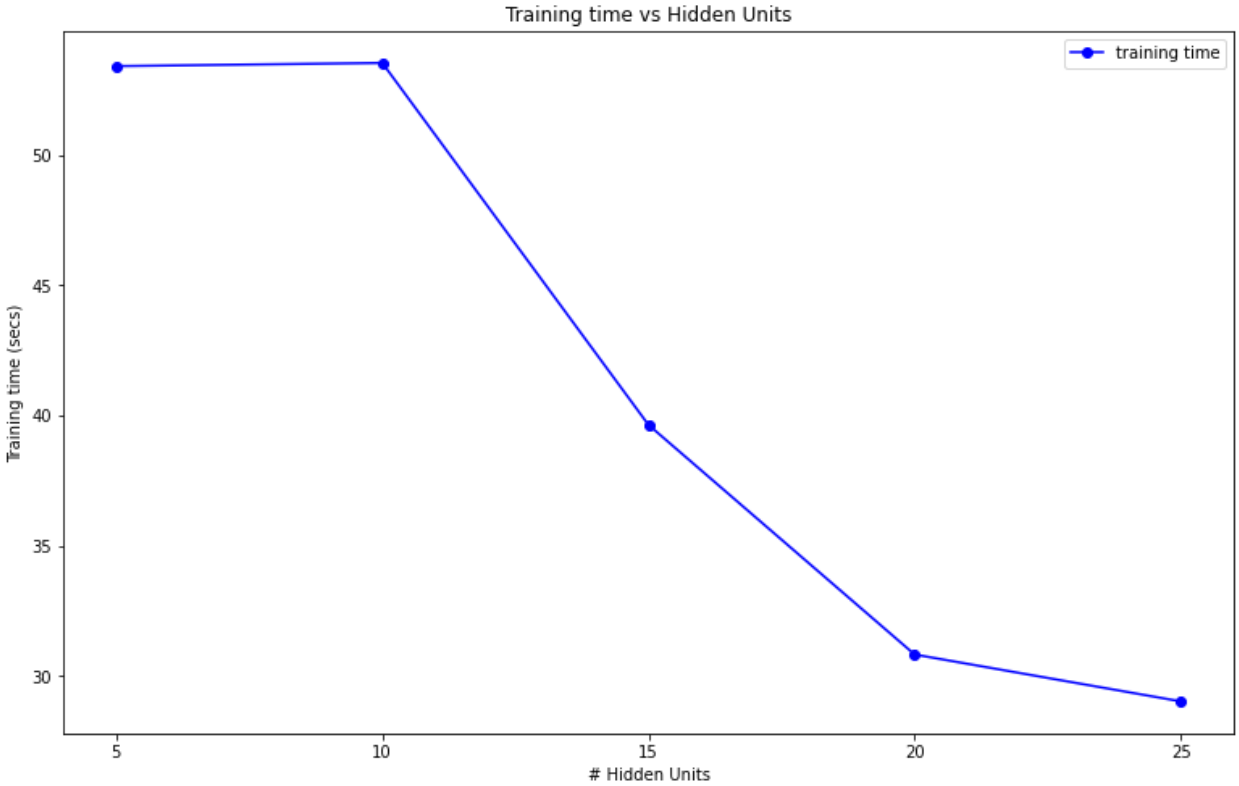
Part D

Decaying Learning Rate

1. Testing model with different hidden units (single layer)

# hidden units	training accuracy(%)	test accuracy(%)	training time (secs)
5	67.321	65.412	53.42
10	80.016	78.14	53.543
15	92.127	91.776	39.647
20	92.279	92.085	30.83
25	92.323	92.263	29.03





Here for larger number of hidden units, convergence is faster. May be a complex model and slower learning rate results in faster convergence.

2. Confusion Matrix

- Hidden Unit 5

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**419560**	81649	0	0	0	0	0	0	0	0
	1	187937	**234561**	0	0	0	0	0	0	0	0
	2	9602	38020	0	0	0	0	0	0	0	0
	3	4555	16566	0	0	0	0	0	0	0	0
	4	1102	2783	0	0	0	0	0	0	0	0
	5	1706	290	0	0	0	0	0	0	0	0
	6	140	1284	0	0	0	0	0	0	0	0
	7	15	215	0	0	0	0	0	0	0	0
	8	3	9	0	0	0	0	0	0	0	0
	9	0	3	0	0	0	0	0	0	0	0

- Hidden Unit 10

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**459316**	41893	0	0	0	0	0	0	0	0
	1	100411	**322087**	0	0	0	0	0	0	0	0
	2	1358	46264	0	0	0	0	0	0	0	0
	3	2129	18992	0	0	0	0	0	0	0	0
	4	3593	292	0	0	0	0	0	0	0	0
	5	1827	169	0	0	0	0	0	0	0	0
	6	10	1414	0	0	0	0	0	0	0	0
	7	2	228	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

- Hidden Unit 15

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**499280**	1929	0	0	0	0	0	0	0	0
	1	4014	**418484**	0	0	0	0	0	0	0	0
	2	70	47552	0	0	0	0	0	0	0	0
	3	16	21105	0	0	0	0	0	0	0	0
	4	3811	74	0	0	0	0	0	0	0	0
	5	1990	6	0	0	0	0	0	0	0	0
	6	0	1424	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	2	1	0	0	0	0	0	0	0	0

- Hidden Unit 20

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**499874**	1335	0	0	0	0	0	0	0	0
	1	1520	**420978**	0	0	0	0	0	0	0	0
	2	4	47618	0	0	0	0	0	0	0	0
	3	1	21120	0	0	0	0	0	0	0	0
	4	3868	17	0	0	0	0	0	0	0	0
	5	1990	6	0	0	0	0	0	0	0	0
	6	0	1424	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

- Hidden Unit 25

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**501100**	109	0	0	0	0	0	0	0	0
	1	970	**421528**	0	0	0	0	0	0	0	0
	2	0	47622	0	0	0	0	0	0	0	0
	3	398	20723	0	0	0	0	0	0	0	0
	4	3759	126	0	0	0	0	0	0	0	0
	5	1996	0	0	0	0	0	0	0	0	0
	6	0	1424	0	0	0	0	0	0	0	0
	7	46	184	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

➔ Here for decaying learning rate, the training is a bit slower i.e. takes more training time but test accuracy improves.

Part E

Comparing ReLU and Sigmoid. Architecture of form (85 -- 100 -- 10)

1. Sigmoid in hidden layer.
    - Training time: 40 secs
    - Training accuracy: 92.331
    - Test accuracy: 92.367
    - Confusion matrix:

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUAL VALUES	0	**501180**	29	0	0	0	0	0	0	0	0
	1	8	**422490**	0	0	0	0	0	0	0	0
	2	110	47512	0	0	0	0	0	0	0	0
	3	207	20914	0	0	0	0	0	0	0	0
	4	3832	52	0	0	1	0	0	0	0	0
	5	1993	3	0	0	0	0	0	0	0	0
	6	0	1424	0	0	0	0	0	0	0	0
	7	65	165	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

2. ReLU in hidden layer.
    - Training time: 18 secs
    - Training accuracy: 92.331
    - Test accuracy: 92.365
    - Confusion matrix:

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
ACTUA	0	**501150**	59	0	0	0	0	0	0	0	0
	1	2	**422496**	0	0	0	0	0	0	0	0
	2	0	47622	0	0	0	0	0	0	0	0

L	3	0	21121	0	0	0	0	0	0	0	0
	4	3797	88	0	0	0	0	0	0	0	0
V	5	1990	6	0	0	0	0	0	0	0	0
A	6	0	1424	0	0	0	0	0	0	0	0
L	7	0	230	0	0	0	0	0	0	0	0
U	8	10	2	0	0	0	0	0	0	0	0
E	9	2	1	0	0	0	0	0	0	0	0
S											

3. <u>ReLU with decaying learning Rate.</u>
    - Training time: 52 secs
    - Training accuracy: 92.339
    - Test accuracy: 92.288
    - Confusion matrix:

		PREDICTED VALUES									
		0	1	2	3	4	5	6	7	8	9
A	0	**500715**	494	0	0	0	0	0	0	0	0
C	1	329	**422168**	1	0	0	0	0	0	0	0
T	2	0	47622	0	0	0	0	0	0	0	0
U	3	0	21121	0	0	0	0	0	0	0	0
A	4	3685	200	0	0	0	0	0	0	0	0
L	5	1992	4	0	0	0	0	0	0	0	0
V	6	0	1424	0	0	0	0	0	0	0	0
A	7	0	230	0	0	0	0	0	0	0	0
L	8	11	1	0	0	0	0	0	0	0	0
U	9	2	1	0	0	0	0	0	0	0	0
E											
S											

Part F

- Using MLP Classifier.
- Max iterations = 500
- Training accuracy = 99%
- Test accuracy = 95%
- Training time = 4.51 sec