

Problem 1: Decision Trees

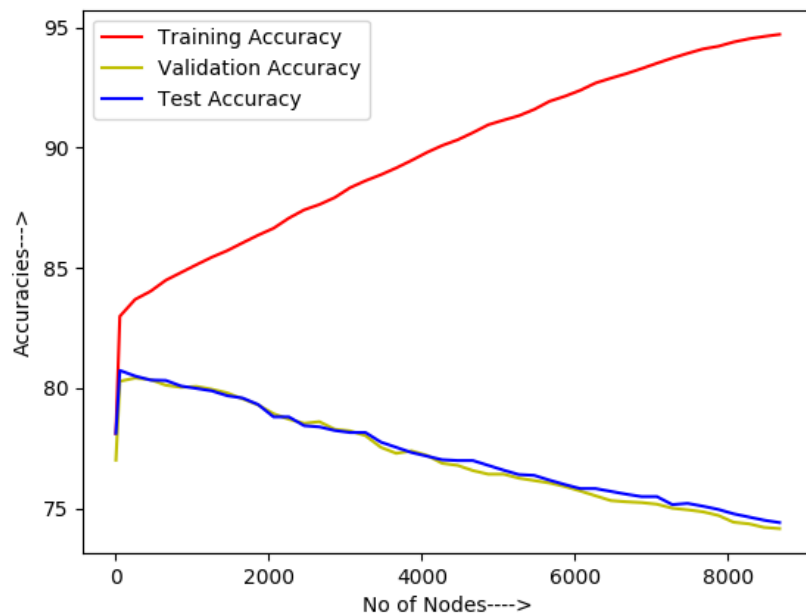
Answer to the problem goes here.

1. As the number of nodes increases (which are considered for prediction), the training accuracy increases, while the validation and testing accuracy decreases. The reason for this behaviour is overfitting. With the increase in the number of nodes, the tree starts overfitting.

Training set Accuracy: 94.71666666666667

Validation set Accuracy: 74.15

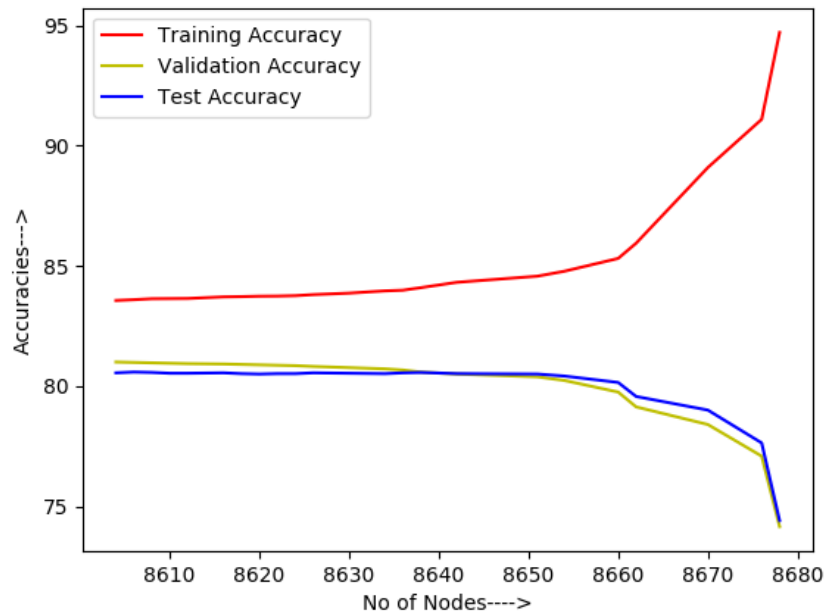
Testing set Accuracy: 74.4



2. When we prune the tree, the number of nodes decreases. Now this avoids overfitting and as a result, the accuracy on the training set reduces , while there is increase in accuracies of testing and validation set.

```
iteration: 0 ,Max_Accuracy 77.08333333333333
iteration: 1 ,Max_Accuracy 78.4
iteration: 2 ,Max_Accuracy 79.13333333333334
iteration: 3 ,Max_Accuracy 79.75
iteration: 4 ,Max_Accuracy 80.23333333333333
iteration: 5 ,Max_Accuracy 80.38333333333334
iteration: 6 ,Max_Accuracy 80.5
iteration: 7 ,Max_Accuracy 80.58333333333333
iteration: 8 ,Max_Accuracy 80.66666666666667
iteration: 9 ,Max_Accuracy 80.71666666666667
iteration: 10 ,Max_Accuracy 80.76666666666667
```

Accuracy after pruning in validation set: 81.0
Training set Accuracy: 83.56111111111112
Validation set Accuracy: 81.0
Testing set Accuracy: 80.55

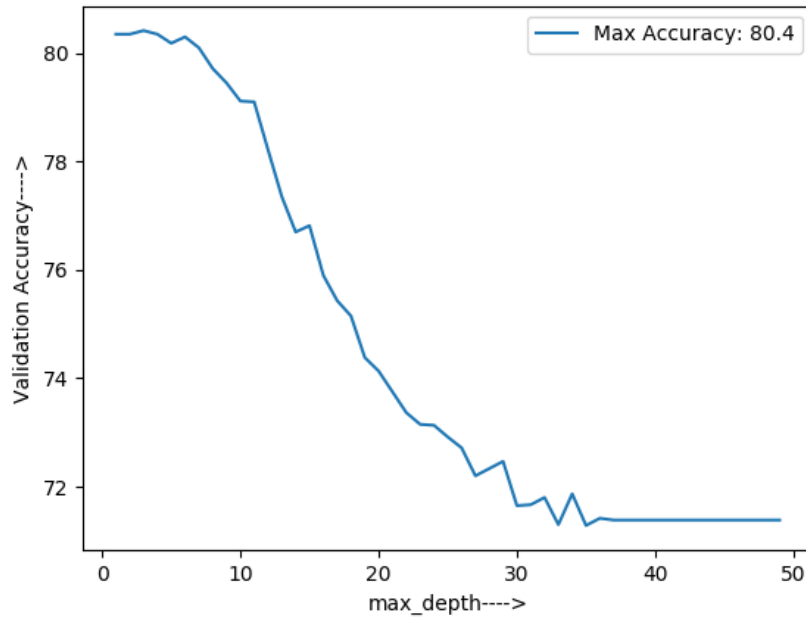



```
Attribute that splitted maximum is: 0 ,split times: 5
Branch 4511 ,Dictionary(attr_no:count): {0: 5, 1: 1, 2: 1, 3: 1, 4: 2, 5: 1, 11: 1
Attribute that splitted maximum is: 0 ,split times: 5
Branch 4512 ,Dictionary(attr_no:count): {0: 5, 1: 1, 2: 1, 3: 1, 4: 2, 5: 1, 11: 1
Attribute that splitted maximum is: 0 ,split times: 5
Branch 4554 ,Dictionary(attr_no:count): {0: 6, 2: 1, 3: 1, 4: 1, 5: 1, 9: 1, 10: 1
Attribute that splitted maximum is: 0 ,split times: 6
Branch 4555 ,Dictionary(attr_no:count): {0: 6, 2: 1, 3: 1, 4: 1, 5: 1, 9: 1, 10: 1
Attribute that splitted maximum is: 0 ,split times: 6
Branch 4556 ,Dictionary(attr_no:count): {0: 5, 2: 1, 3: 1, 4: 1, 5: 1, 9: 1, 10: 1
Attribute that splitted maximum is: 0 ,split times: 5
Branch 4606 ,Dictionary(attr_no:count): {0: 5, 1: 1, 2: 1, 4: 1, 5: 1, 9: 1, 11: 1
Attribute that splitted maximum is: 0 ,split times: 5
Branch 4607 ,Dictionary(attr_no:count): {0: 5, 1: 1, 2: 1, 4: 1, 5: 1, 9: 1, 11: 1
Attribute that splitted maximum is: 0 ,split times: 5
Branch 4610 ,Dictionary(attr_no:count): {0: 5, 1: 1, 2: 1, 3: 1, 4: 1, 5: 1, 9: 1,
Attribute that splitted maximum is: 0 ,split times: 5
Branch 4611 ,Dictionary(attr_no:count): {0: 5, 1: 1, 2: 1, 3: 1, 4: 1, 5: 1, 9: 1,
Attribute that splitted maximum is: 0 ,split times: 5
```

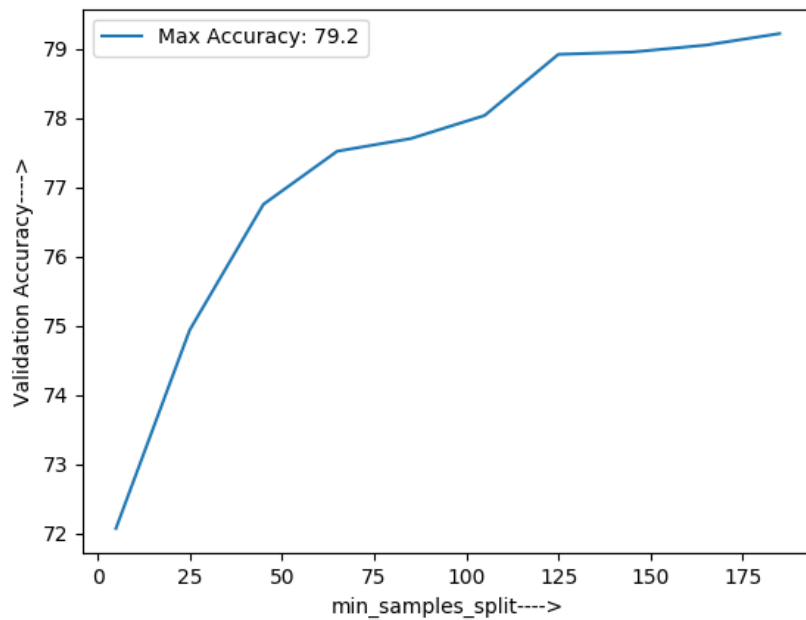
4. Training set Accuracy: 83.32222222222222
Validation set Accuracy: 80.53333333333333

Test set Accuracy: 80.61666666666667

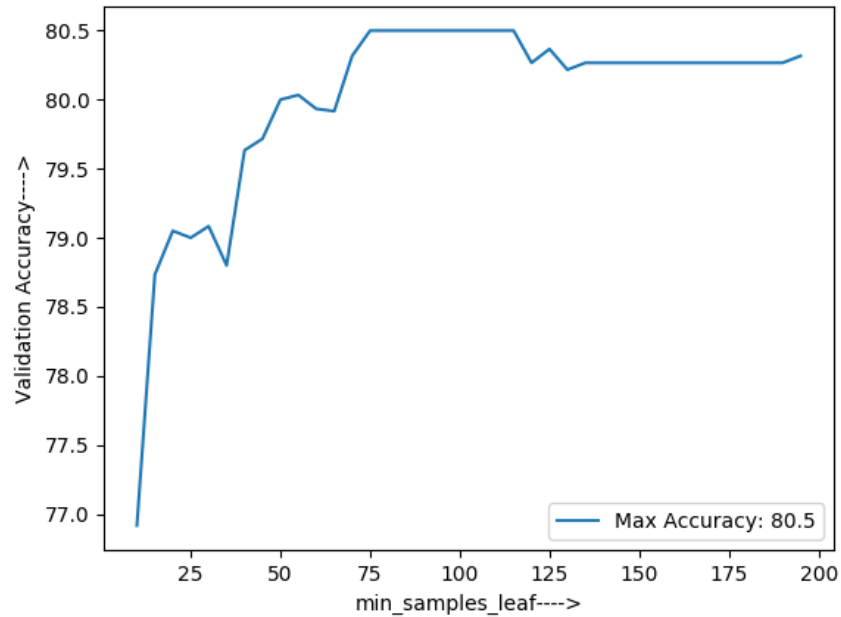
Best Parameters: [criterion='gini', random_state=0, None, 145, 70]
where min_samples_split = 145
min_samples_leaf = 70



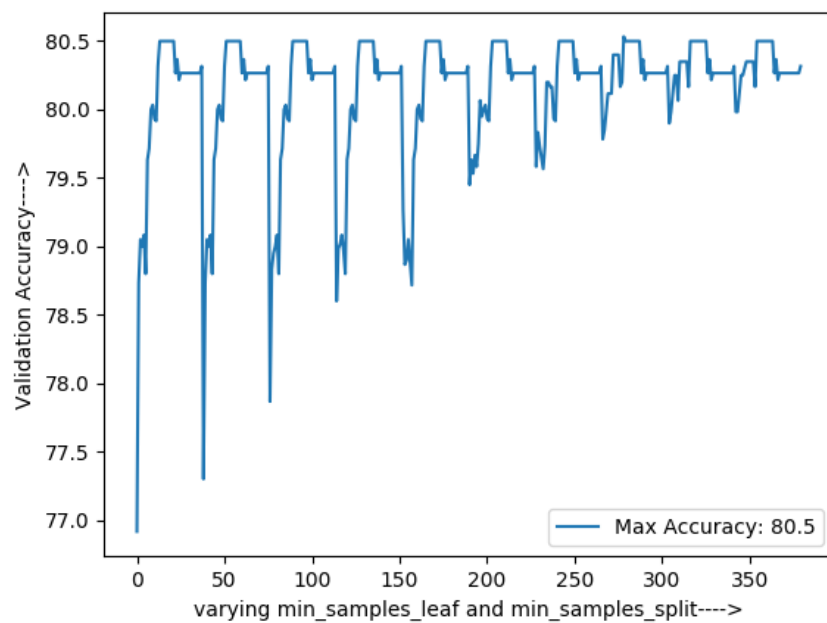
As the max_depth parameter increases, the accuracy on the validation set decreases. This concludes overfitting.



with the Increase in the min_sample_split parameter, the tree is restricted to grow too much, therefore the validation accuracy increases.



The min_sample_leaf has a similar effect on the tree as the min_sample_split parameter. it also avoids overfitting. So It can be concluded that, as we decrease the number of nodes in the tree the validation accuracy increases.



5. First converted the categorical variables into multiple binary variables using One-hot

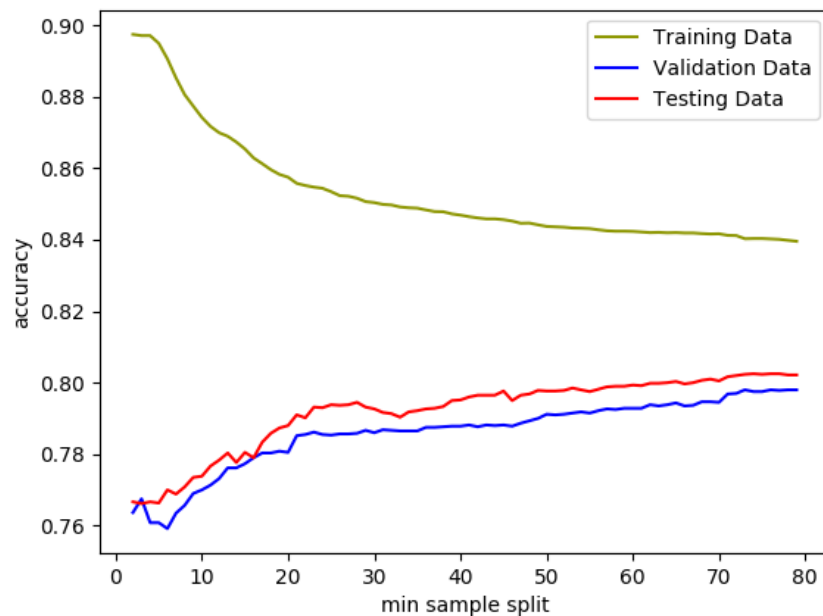
encoding and then computing accuracies on best parameters obtained in previous part.

Training set Accuracy: 83.18333333333334

Validation set Accuracy: 79.88333333333333

Test set Accuracy: 80.41666666666667

We can see that we got better accuracies on previous part on these best parameters.



6. Training set Accuracy: 99.93888888888888

Validation set Accuracy: 80.36666666666666

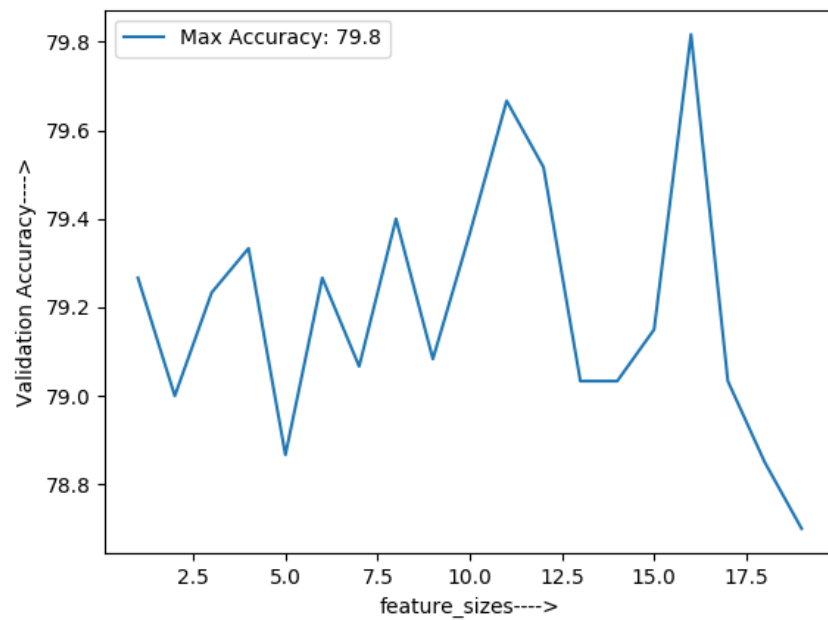
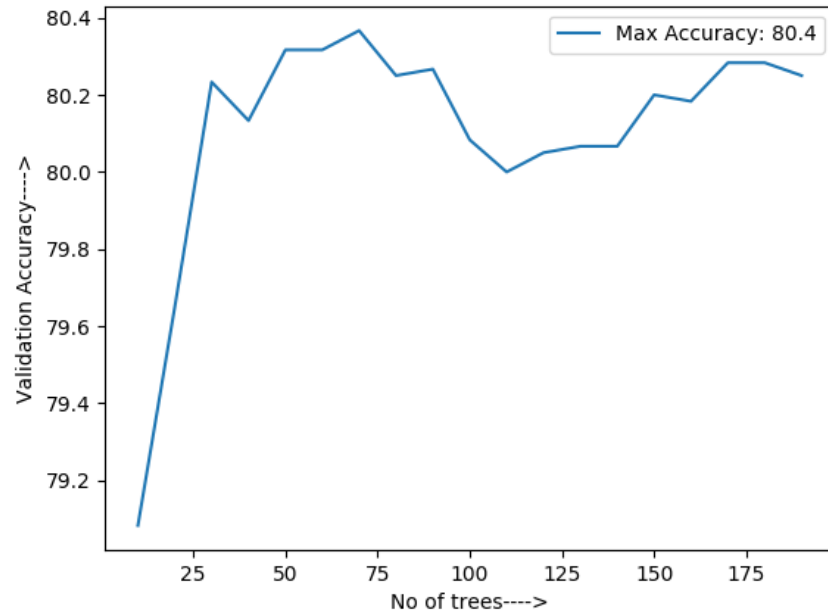
Test set Accuracy: 80.13333333333334

Best Parameters: ['gini', 0, 70]

where n_estimators = 70

bootstrap = true

Here we can observe that the training set accuracies are higher which means training data overfits.



Problem 2:Neural Networks

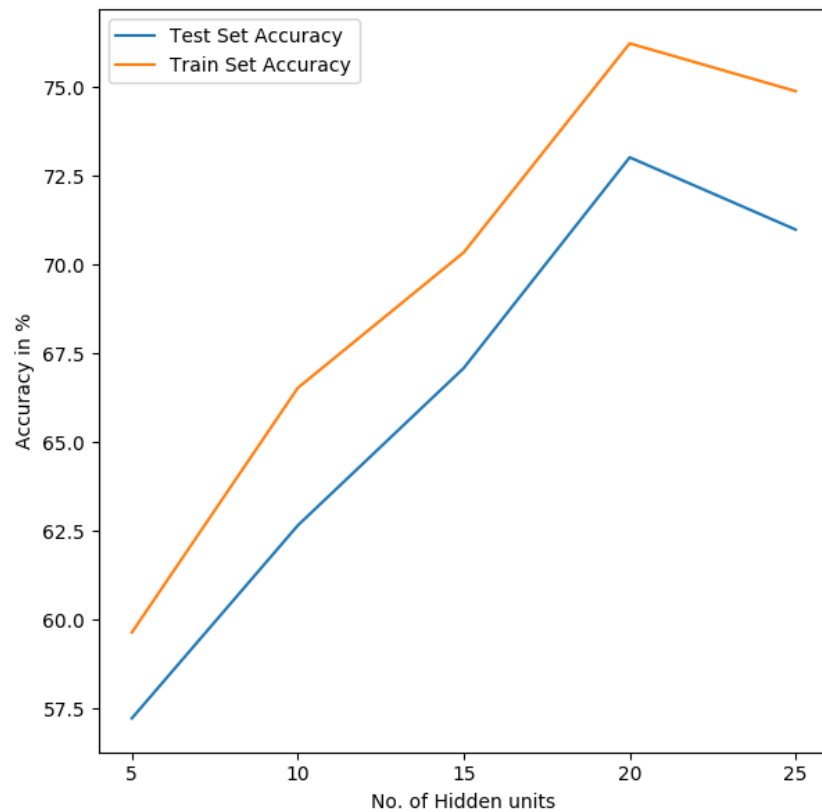
Neural Networks

1. [csv files link](#).
2. Neural Network implemented.
3. Single hidden layer
Stopping Criteria: Number of epochs = 500
Time Taken: 40 mins
Accuracy for each hidden unit: 5,10,15,20,25
I have taken learning rate = 1.0
For training data-

- 59.64
- 66.52
- 70.34
- 76.22
- 74.88

For testing data-

- 57.23
- 62.65
- 67.08
- 73.01
- 70.98



confusion matrix for each hidden layer:

For hidden layers=5

```
[[9723 2770    0    0    0    0    0    0    0    0]
 [5152 5447    0    0    0    0    0    0    0    0]
 [ 460  746    0    0    0    0    0    0    0    0]
 [ 147  366    0    0    0    0    0    0    0    0]
 [  46   47    0    0    0    0    0    0    0    0]
 [  43   11    0    0    0    0    0    0    0    0]
 [   8   28    0    0    0    0    0    0    0    0]
 [   0    6    0    0    0    0    0    0    0    0]
 [   5    0    0    0    0    0    0    0    0    0]
 [   3    2    0    0    0    0    0    0    0    0]]
```

For hidden layers=10

```
[[10039 2454    0    0    0    0    0    0    0    0]
 [      ]
```

Assignment 3

[3818	6781	0	0	0	0	0	0	0	0]
[225	981	0	0	0	0	0	0	0	0]
[45	468	0	0	0	0	0	0	0	0]
[39	54	0	0	0	0	0	0	0	0]
[43	11	0	0	0	0	0	0	0	0]
[2	34	0	0	0	0	0	0	0	0]
[0	6	0	0	0	0	0	0	0	0]
[4	1	0	0	0	0	0	0	0	0]
[4	1	0	0	0	0	0	0	0	0]]

For hidden layers=15

[[10304	2189	0	0	0	0	0	0	0	0]
[3548	7051	0	0	0	0	0	0	0	0]
[215	991	0	0	0	0	0	0	0	0]
[30	483	0	0	0	0	0	0	0	0]
[73	20	0	0	0	0	0	0	0	0]
[49	5	0	0	0	0	0	0	0	0]
[1	35	0	0	0	0	0	0	0	0]
[0	6	0	0	0	0	0	0	0	0]
[4	1	0	0	0	0	0	0	0	0]
[5	0	0	0	0	0	0	0	0	0]]

For hidden layers=20

```
[[10722 1771 0 0 0 0 0 0 0 0]
 [ 2874 7725 0 0 0 0 0 0 0 0]
 [ 121 1085 0 0 0 0 0 0 0 0]
 [ 31 482 0 0 0 0 0 0 0 0]
 [ 72 21 0 0 0 0 0 0 0 0]
 [ 48 6 0 0 0 0 0 0 0 0]
 [ 0 36 0 0 0 0 0 0 0 0]
 [ 0 6 0 0 0 0 0 0 0 0]
 [ 5 0 0 0 0 0 0 0 0 0]
 [ 5 0 0 0 0 0 0 0 0 0]]
```

For hidden layers=25

```
[[11642 851 0 0 0 0 0 0 0 0]
 [ 2162 8437 0 0 0 0 0 0 0 0]
 [ 42 1164 0 0 0 0 0 0 0 0]
 [ 28 485 0 0 0 0 0 0 0 0]
 [ 81 12 0 0 0 0 0 0 0 0]
 [ 50 4 0 0 0 0 0 0 0 0]
 [ 0 36 0 0 0 0 0 0 0 0]
 [ 0 6 0 0 0 0 0 0 0 0]
 [ 4 1 0 0 0 0 0 0 0 0]
 [ 5 0 0 0 0 0 0 0 0 0]]
```

As the number of hidden units are increased, the accuracy over the training and testing set increases. The training time also increases as the number of hidden units are increased.

4. Two hidden layers.

Stopping Criteria: Number of epochs = 500

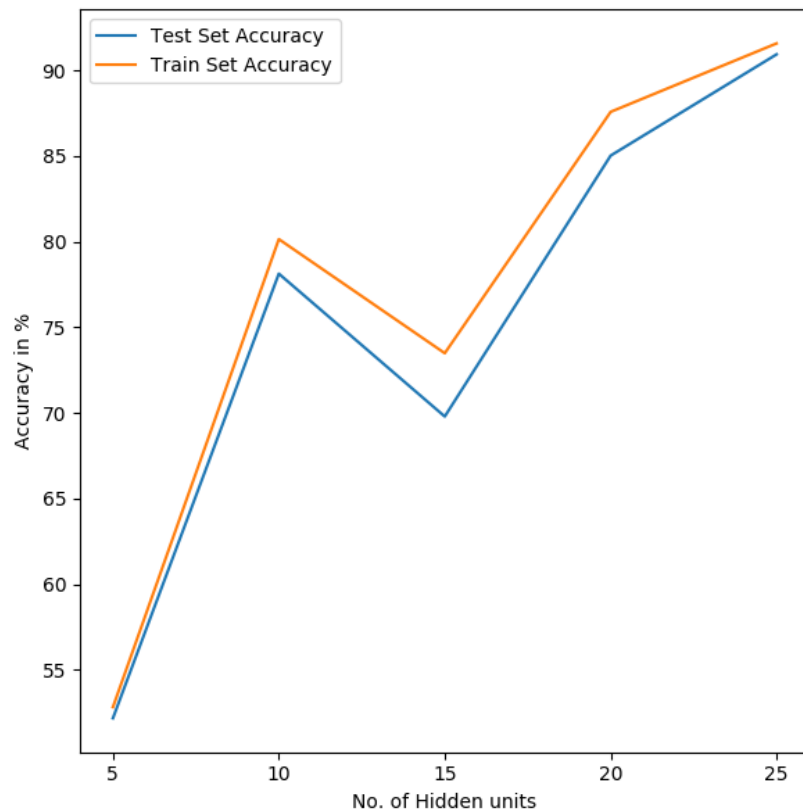
Accuracy for each hidden unit: 5,10,15,20,25

For training data-

- 52.81
- 80.13
- 73.47
- 87.56
- 91.56

For testing data-

- 52.16
- 78.11
- 69.78
- 85.01
- 90.92



5. Adaptive Learning Rate

One hidden layers.

Stopping Criteria: Number of epochs = 500

Accuracy for each hidden unit: 5,10,15,20,25

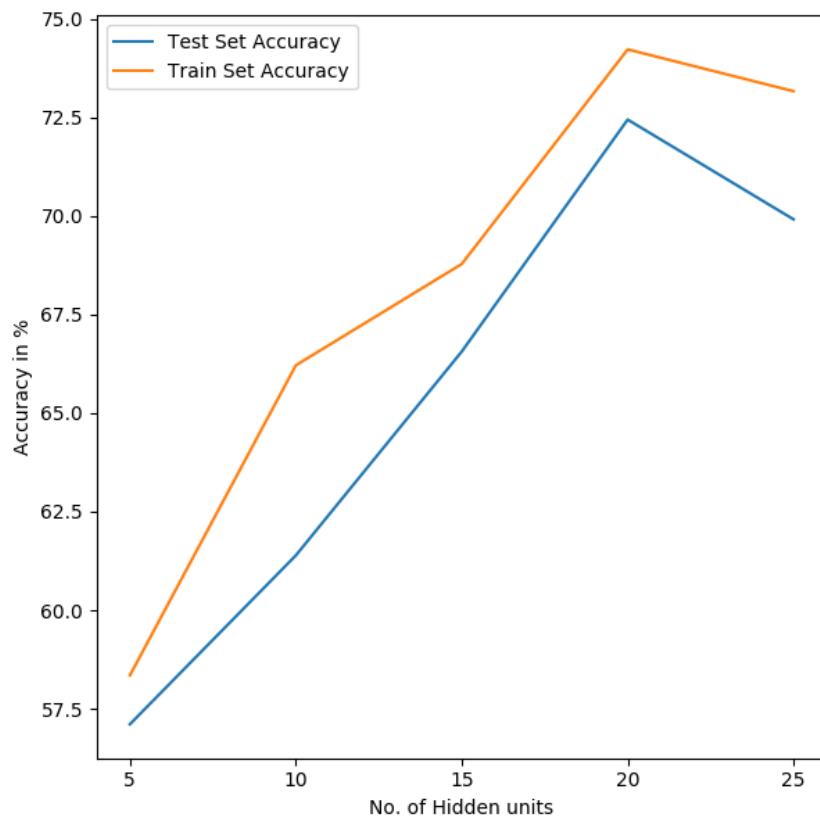
For training data-

- 58.35
- 66.21

- 68.78
- 74.22
- 73.16

For testing data-

- 57.11
- 61.39
- 66.56
- 72.44
- 69.91



Two hidden layers.

Stopping Criteria: Number of epochs = 500

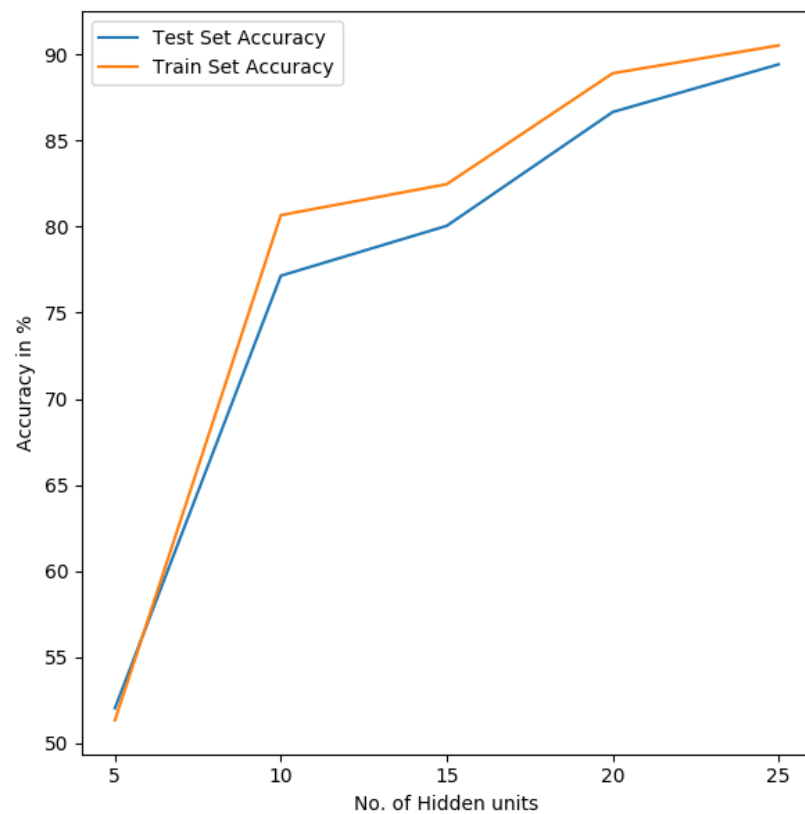
Accuracy for each hidden unit: 5,10,15,20,25

For training data-

- 51.33
- 80.67
- 82.47
- 88.91
- 90.53

For testing data-

- 52.03
- 77.15
- 80.05
- 86.66
- 89.43



6. ReLU

One hidden layer

Stopping Criteria: Number of epochs = 500

Accuracy for each hidden unit: 5,10,15,20,25

For training data-

- 57.61
- 61.21
- 64.88
- 84.76
- 77.52

For testing data-

- 56.23
- 59.74
- 60.92
- 83.03
- 74.56

