PA2 - Linear Models for Supervised Learning

CSE 574 - Neural-networks

Group No: 14

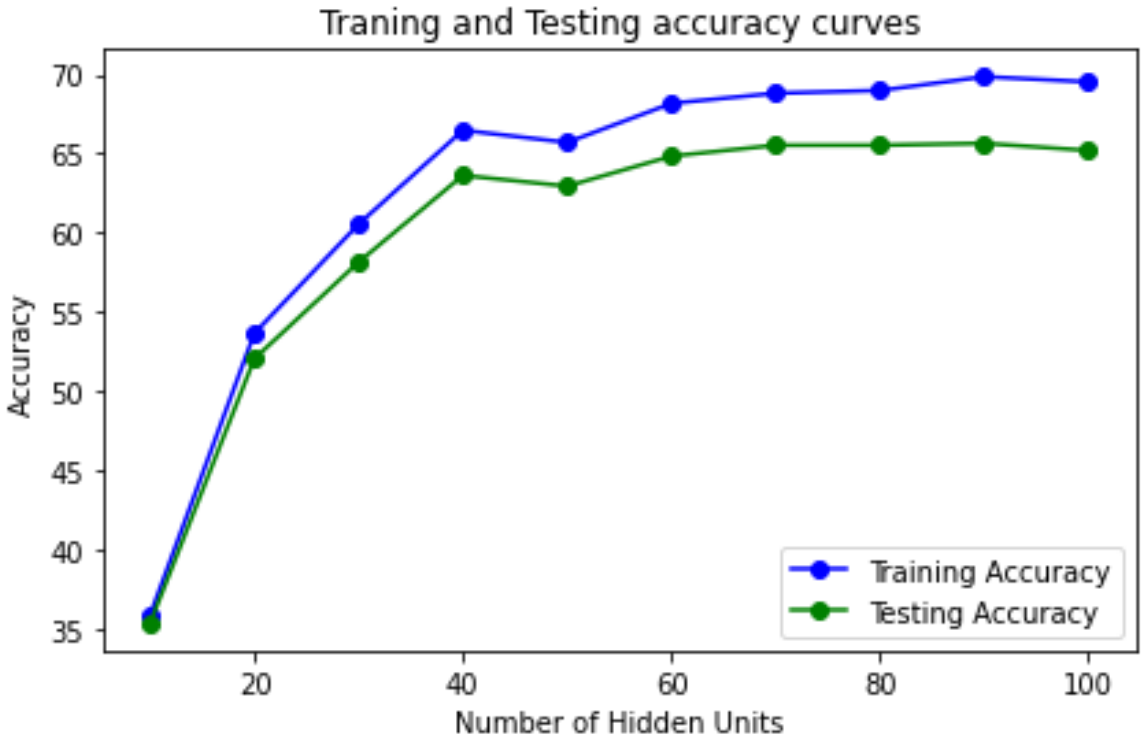
ANOOSHA SEELAM	(lseelam@buffalo.edu)	50388620
VISWAPUJITHA SURESH	(viswapuj@buffalo.edu)	50388544
BHAVISHA KONDHOL	(bhavisha@buffalo.edu)	50388359

1. For the given hidden layer units M=50 and regularization parameter λ = 0,
   - Training accuracy = 66.55%
   - Testing accuracy = 64.05%
   - Training time = 14.68 seconds

2. **Experiment 1:**

   Finding the optimal number of hidden layer units (M)
   - Range of M: 10 to 100 in increments of 10
   - Value of Lambda: 0
   - For each step, we are calculating Training accuracy, Testing accuracy, and time taken for training (Training time)

Number of Hidden layer units (M)	Training Accuracy	Testing Accuracy	Training time(seconds)
10	35.86	35.36	11.96
20	53.65	52.05	11.57
30	60.54	58.15	13.05
40	66.47	63.62	13.89
50	65.70	62.94	14.28
60	68.17	64.84	18.37
70	68.81	65.52	17.05
80	68.98	65.52	16.40
90	69.85	65.64	19.38
100	69.52	65.19	21.43


Traning and Testing accuracy curves
Accuracy
Number of Hidden Units
Training Accuracy
Testing Accuracy
70
65
60
55
50
45
40
35
20
40
60
80
100


Training time as a function of M
Traning time(in sec)
Number of Hidden Units
Training time
20
18
16
14
12
20
40
60
80
100

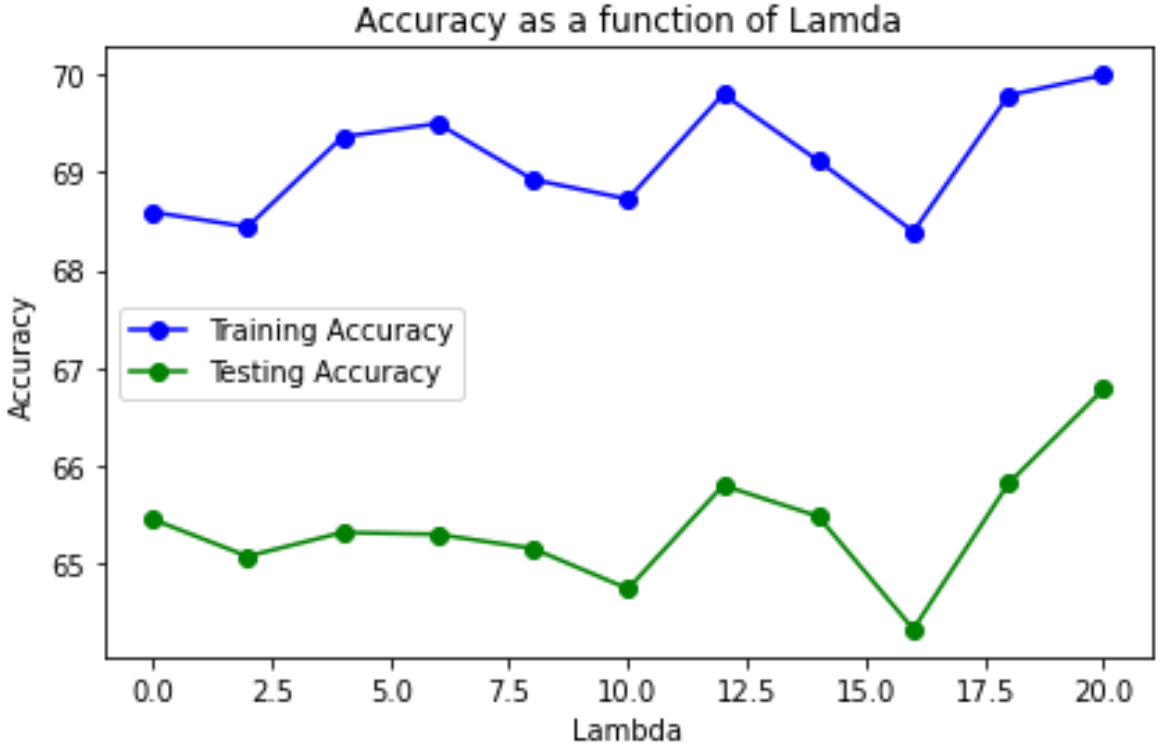
Observation: From the graph, we can see that testing accuracy increases as we increase the number of hidden layer units, but training time also increases as we increase the number of hidden layer units. We can also observe that after a certain M, the testing accuracy won't change much. So, we select **M=80** to maintain the balance between testing accuracy and training time.

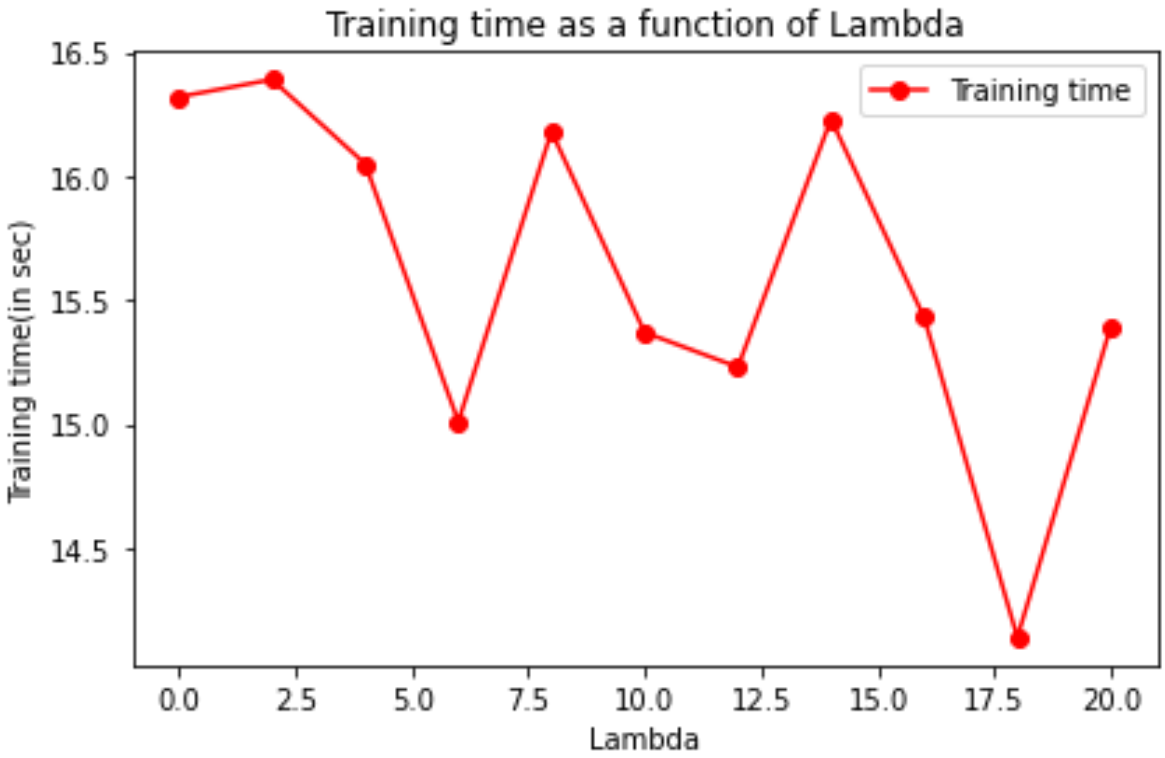
3. **Experiment 2:**

Finding the optimal Lamda (λ)

- Range of Lambda: 0 to 20 in increments of 2
- Number of hidden layer units: 80
- For each step, we are calculating Training accuracy, Testing accuracy, and time taken for training (Training time)

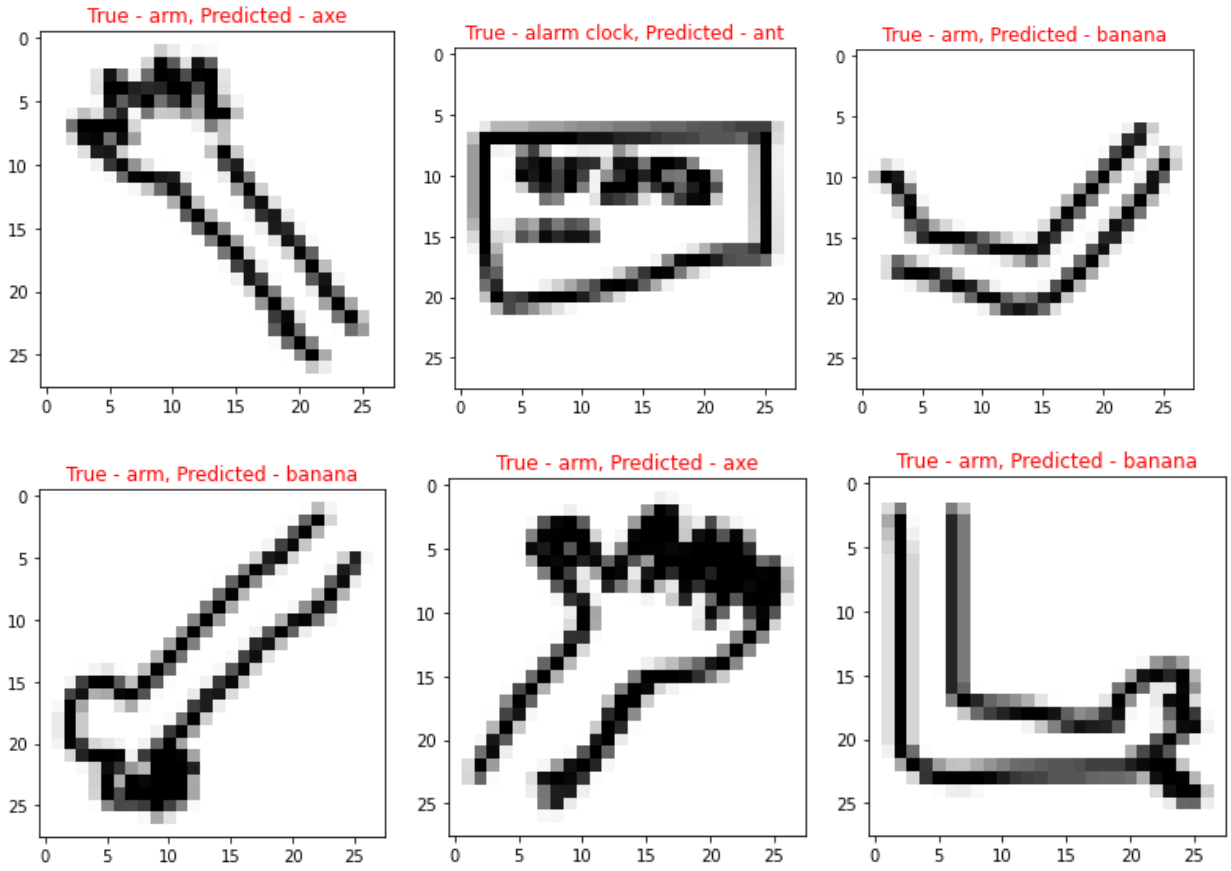
Lambda (λ)	Training Accuracy	Testing Accuracy	Training time(seconds)
0	68.59	65.46	16.32
2	68.44	65.08	16.39
4	69.36	65.32	16.05
6	69.50	65.30	15.01
8	68.92	65.16	16.18
10	68.72	64.74	15.37
12	69.80	65.80	15.23
14	69.12	65.48	16.23
16	68.39	64.33	15.43
18	69.78	65.82	14.14
20	69.99	66.78	15.39

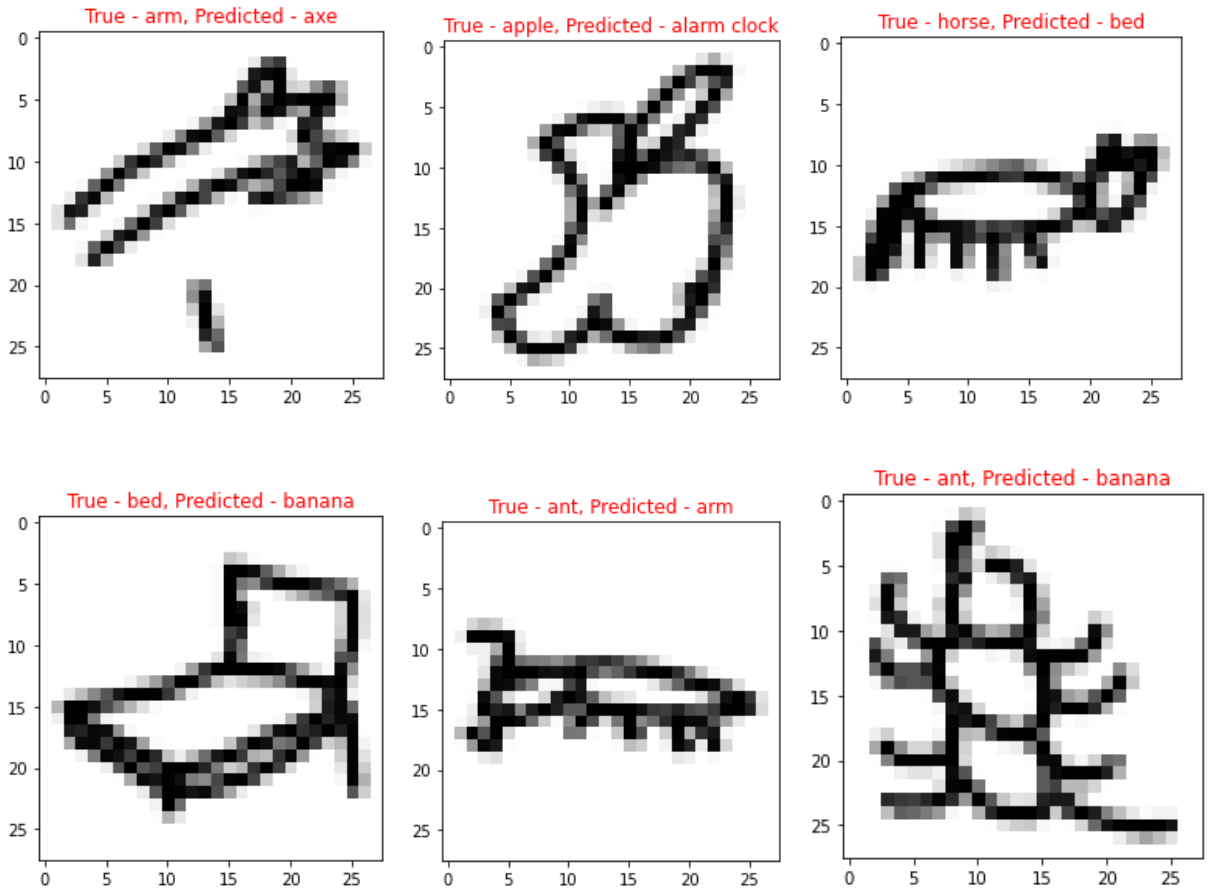

Accuracy as a function of Lamda
Training Accuracy
Testing Accuracy
Accuracy
Lambda
70
69
68
67
66
65
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0


Training time as a function of Lambda
Training time
Training time(in sec)
Lambda
16.5
16.0
15.5
15.0
14.5
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

Observation: From the graph, we can see that training time almost remains the same as we increase the lambda. So, we can ignore that. The accuracy gets lower as we increase the lambda. This is because lambda tries to strike a balance between the simplicity of the model and training data fit. If we select low lambda, our model might be complex and overfit the data. If we select high lambda, our model might be too simple and underfit the data. So, to balance the model, we select Lambda = 12.

4. For the given hidden layer units M=80 and regularization parameter λ = 12,
    - Training accuracy = 68.98%
    - Testing accuracy = 65.31%
    - Training time = 15.9 seconds



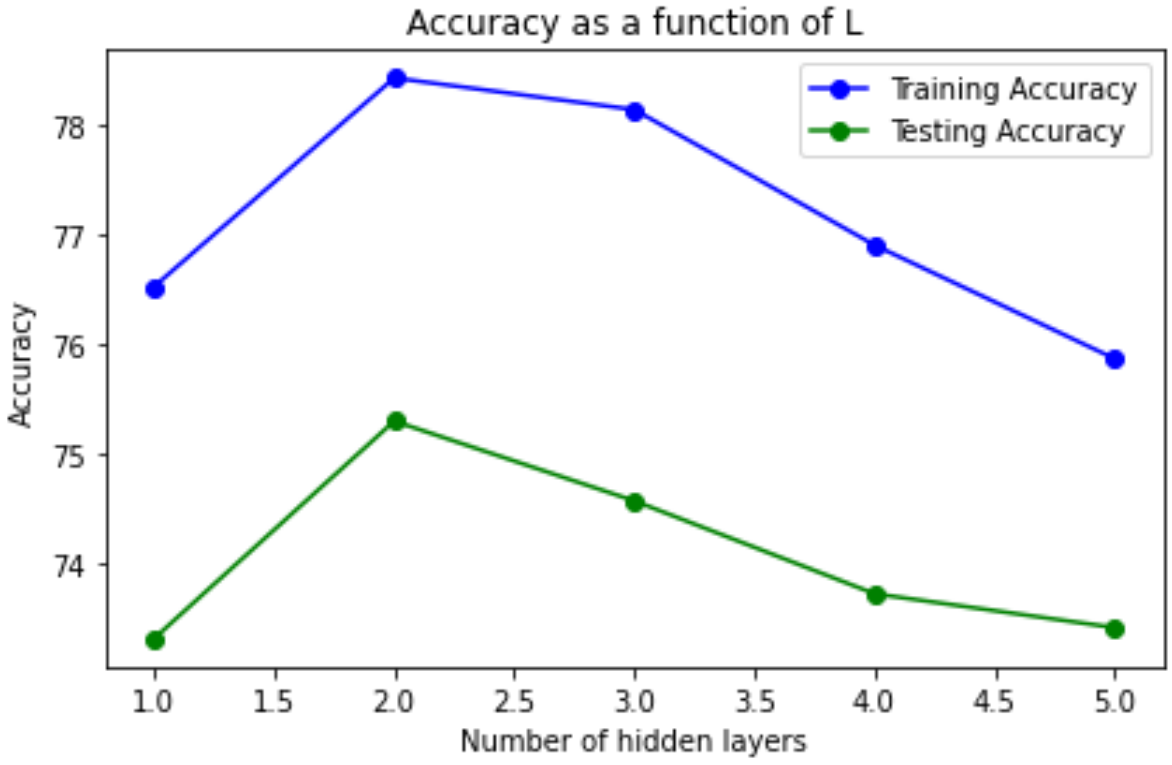


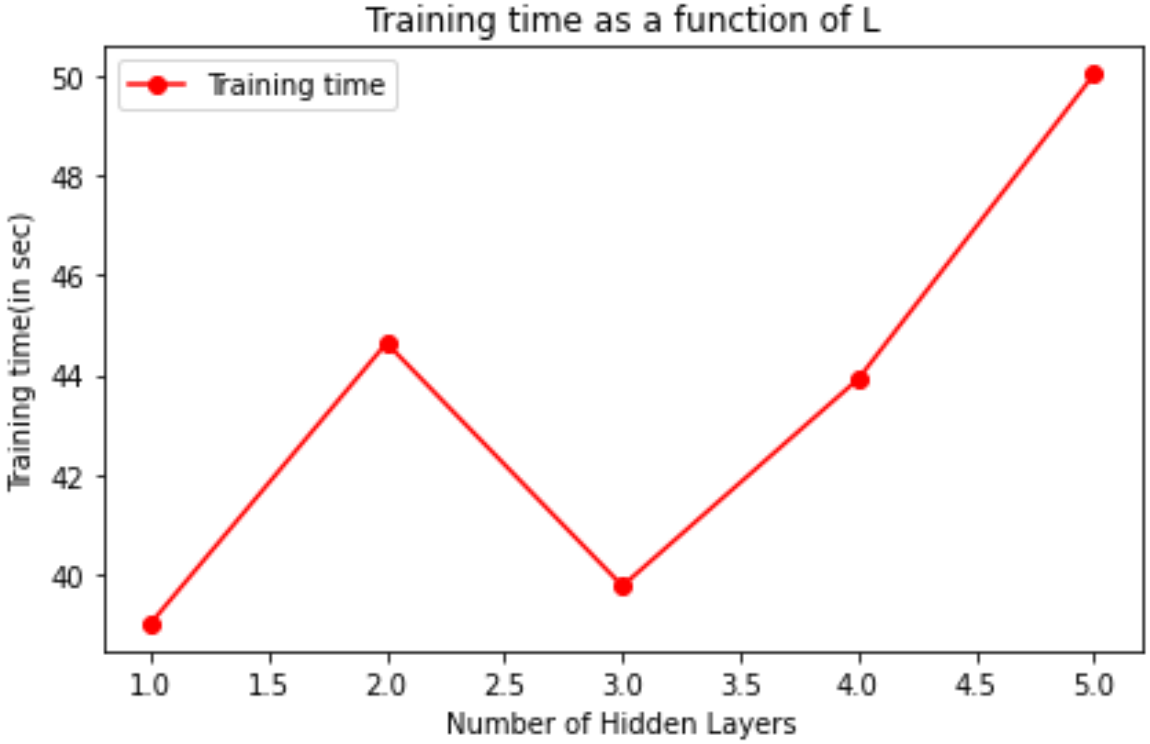
From the above figures, we can see that model always tries to predict any object with a long body like a banana. And a long body with a rectangle-like head as an axe. So, the model is most likely trying to understand the edge features of the object and make a prediction. We can improve the model better by making the model learn the different features between each object. This can be done by increasing the number of hidden layers. These inner deep hidden layers learn the deep features of the object and thus accuracy of the model will be improved.

Report 2:

Problem 1.

L	Training Accuracy	Testing Accuracy	Training time(seconds)
1	76.53	73.31	39.02
2	78.44	75.30	44.64
3	78.15	74.57	39.77
4	76.91	73.72	43.92
5	75.87	73.41	50.03


Accuracy as a function of L
Training Accuracy
Testing Accuracy
Accuracy
78
77
76
75
74
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Number of hidden layers


Training time as a function of L
Training time
Training time(in sec)
50
48
46
44
42
40
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Number of Hidden Layers

Observation: From the graph, we can see that accuracy increases as we increase the number of hidden layers and it starts decreasing after a certain number of layers. This is because the model tries to learn the deep features in the objects and thus predicts the correct one. We can observe this when we compare the accuracies from PART2. But as we keep on adding hidden layers, the model becomes complex and it starts to overfit the data.

Here, L = 2 is selected to achieve high accuracy within reasonable training time.

Problem 2.

For the given hidden layer units M=80 and L=2 and 'sigmoid' activation,

- Training accuracy = 78.44%
- Testing accuracy = 75.30%
- Training time = 44.64 seconds

For the given hidden layer units M=80 and L=2 and 'relu' activation,

- Training accuracy = 82.40%
- Testing accuracy = 70.72%
- Training time = 37.14 seconds

For the given hidden layer units M=80 and L=2 and 'tanh' activation,

- Training accuracy = 78.54%
- Testing accuracy = 75.06%
- Training time = 37.14 seconds

Observation: From the accuracies, we can see that sigmoid and tanh perform better than relu. The advantage of tanh is that negative inputs are strongly mapped to negative and positive inputs are strongly mapped to positive.