

INDIAN INSTITUTE OF TECHNOLOGY, DELHI

ASSIGNMENT REPORT

Digit Recognition Neural Network

SHAURYA MOHAN | ENTRY NO. 2019MT10658

SHRAVAN NAWANDAR | ENTRY NO. 2019MT10674

Course - ELL409 | Prof. Seshan Srirangarajan

Compiled on May 20, 2021

Contents

1	Introduction	2
2	Five Fold Validation	3
2.1	Without Regularization	3
2.1.1	Introduction:	3
2.1.2	Finding the optimal number on nodes in hidden layer	3
2.1.3	Finding the best learning rate	5
2.1.4	Conclusion:	8
2.2	With Regularization	9
2.2.1	Finding the best regularization parameter	9
2.2.2	Conclusion:	14
3	Ten Fold Validation	15
3.1	Without Regularization	15
3.1.1	Introduction:	15
3.1.2	Finding the optimal number on nodes in hidden layer	15
3.1.3	Finding the best learning rate:	17
3.1.4	Conclusion:	20
3.2	With Regularization	21
3.2.1	Finding the best regularization parameter:	21
3.2.2	Conclusion:	26
4	Data Visualisation	27

Chapter 1

Introduction

We have used 5 and 10 fold cross validation in this assignment. We are working with 2 different folds to check the impact of the number of folds on the final model chosen. In our way of implementing cross-validation, we have tried to incorporate mini-batch gradient descent in it as well so that we could explore the different types of gradient descents as well since full batch gradient descent was already implemented in assignment 1.

To incorporate mini batch gradient descent, we have first divided the training data set of size of 4500 data points into 5(or 10) parts. Then, while training one model if we want to leave the k^{th} fold out, instead of concatenating all other folds we train our weights on one fold and pass the trained weights to the next fold as so on. In this way our algorithm first uses only one fold to train, then the second fold and so on. This is exactly identical to what happens in mini-batch gradient descent with mini-batch size equal to the size of each fold. Had we concatenated all folds involved in the training process, the algorithm would have been similar to full-batch gradient descent.

In the upcoming sections we have tried to find the best possible set of parameters using cross-validation by using around 30 models to find the best learning rate and then again around 30 models to find the best regularisation parameters. We have also tried to vary the learning rates and regularisation parameters across layers to add better functionalities to the model.

Finally we have tried to visualize the dataset and compared the actual values with the predicted values. Visualizing the data helped us recognize some of the common mistakes our trained model always made in classifying some set of confusing digits (for example 1 and 7). This can help us by indicating a specific area to work upon in the future so as to refine our model further.

Chapter 2

Five Fold Validation

2.1 Without Regularization

2.1.1 Introduction:

In this assignment we have to implement neural network, so for this we have divided the assignment into two parts. In first part we have done five fold validation, that is we have divided the training set into five parts and trained on four parts and validated on the fifth part such that all the parts are used for validation. This helps in selection of hyper-parameters like learning rates and number of nodes in the hidden layer to get the best performing model for the question. In the following section we have first found the optimal number of nodes for a good learning rate (that we determined through preliminary rounds of running the model). Then we get the best learning rate (elaborate methodology in that section only) for the given number of nodes. And hence train for around 4000 iterations to get the best model.

2.1.2 Finding the optimal number on nodes in hidden layer

This section deals with finding the optimum number of nodes in the hidden layer. We have selected the best learning rates that we got from the preliminary runs and used it to compare the 7 hidden layer nodes option. The following heatmap is to find the best number of nodes in the hidden layer.

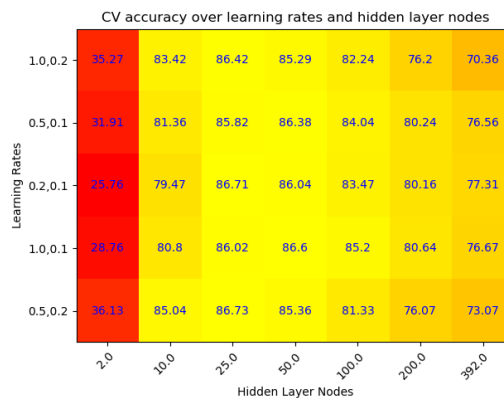


Figure 2.1: Heat Map showing 25 and 50 as best number of nodes

Interpretations: Through the heat map it can be easily be seen that 25 and 50 are the best number of hidden layer nodes but one careful examination one can see 25 edges out 50 generally hence we are going to use 25 as are number of hidden layer nodes. Some other notable observations is that the accuracy increases drastically as we increase number of nodes from 2 to 25 and then gradually decreases as we keep on increasing the nodes to 392. Hence this shows that the best number of hidden layer nodes is around 25 only. Also we can clearly see that 2 is a big no for hidden layer nodes which can be explained by loss of data as number of features are more than the number of nodes in the hidden layer hence causing such bad results.

Below is the table used for the heat map above

L Rate 1	L Rate 2	Nodes	Accuracy
0.5	0.2	2	36.13
0.5	0.2	10	85.04
0.5	0.2	25	86.73
0.5	0.2	50	85.36
0.5	0.2	100	81.33
0.5	0.2	200	76.07
0.5	0.2	392	73.07
1	0.1	2	28.76
1	0.1	10	80.8
1	0.1	25	86.02
1	0.1	50	86.6
1	0.1	100	85.2
1	0.1	200	80.54
1	0.1	392	76.67
0.2	0.1	2	25.76
0.2	0.1	10	79.47
0.2	0.1	25	86.71
0.2	0.1	50	86.04

Figure 2.2: heat map 1 - 18 entries

L Rate 1	L Rate 2	Nodes	Accuracy
0.2	0.1	100	83.47
0.2	0.1	200	80.16
0.2	0.1	392	77.31
0.5	0.1	2	31.91
0.5	0.1	10	81.36
0.5	0.1	25	85.82
0.5	0.1	50	86.38
0.5	0.1	100	84.04
0.5	0.1	200	80.24
0.5	0.1	392	76.56
1	0.2	2	35.27
1	0.2	10	83.42
1	0.2	25	86.42
1	0.2	50	85.29
1	0.2	100	82.24
1	0.2	200	76.20
1	0.2	392	70.36
X	X	X	X

Figure 2.3: heat map 19 - 35 entries

Conclusion: There was a clear demarcation between the column of 25 and 50 hidden nodes as they were the brightest yellow. Signifying that they were the best among all the options. On closer examination we can see that 25 edges out 50 most of the times hence giving 25 as the best number of nodes for the hidden layer.

2.1.3 Finding the best learning rate

In this section we are going to find the best learning rate for the model with 25 hidden layer nodes. We have varied the learning rate 1 and learning rate 2 across values: 0.01,0.1,0.2,0.5,1,10 (6 rates) giving a total 36 models to compare from to find the best learning rate from them. For this we have chosen the best learning rate 2 for a given learning rate 1.

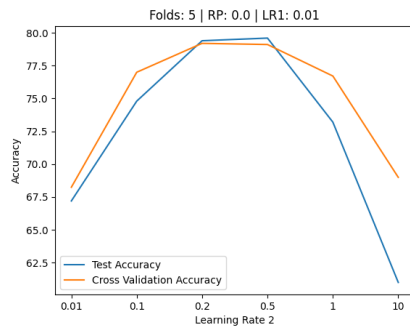


Figure 2.4: graph for LR1 = 0.01

LR1	LR2	CV accuracy	Test accuracy
0.01	0.01	68.24	67.19
0.01	0.1	77	74.8
0.01	0.2	79.2	79.4
0.01	0.5	79.11	79.6
0.01	1	76.71	73.2
0.01	10	69	61

Figure 2.5: Table for the graph

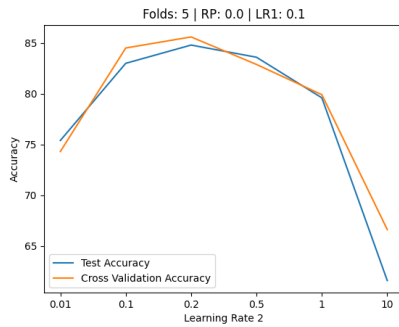


Figure 2.6: graph for LR1 = 0.1

LR1	LR2	CV accuracy	Test accuracy
0.1	0.01	74.31	75.4
0.1	0.1	84.51	83
0.1	0.2	85.6	84.8
0.1	0.5	82.88	83.6
0.1	1	79.91	79.60
0.1	10	66.62	61.6

Figure 2.7: Table for the graph

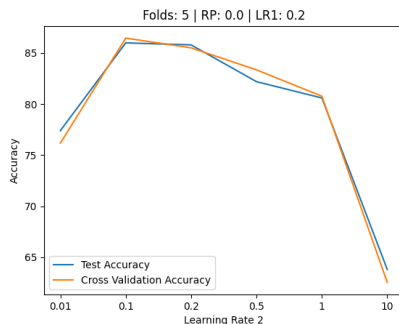


Figure 2.8: graph for LR1 = 0.2

LR1	LR2	CV accuracy	Test accuracy
0.2	0.01	76.17	77.4
0.2	0.1	86.47	86
0.2	0.2	85.51	85.8
0.2	0.5	83.35	82.2
0.2	1	80.78	80.6
0.2	10	62.53	63.80

Figure 2.9: Table for the graph

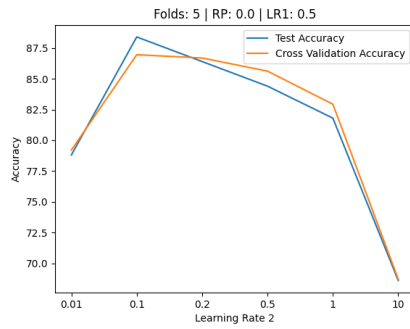


Figure 2.10: graph for $LR1 = 0.5$

LR1	LR2	CV accuracy	Test accuracy
0.5	0.01	79.2	78.8
0.5	0.1	86.95	88.4
0.5	0.2	86.68	86.4
0.5	0.5	85.62	84.39
0.5	1	82.93	81.8
0.5	10	68.71	68.6

Figure 2.11: Table for the graph

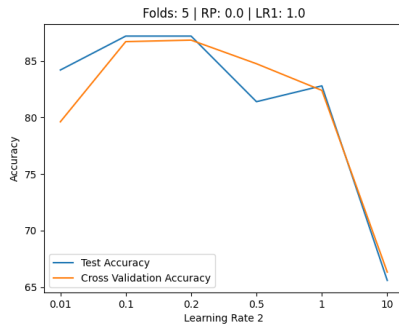


Figure 2.12: graph for $LR1 = 1$

LR1	LR2	CV accuracy	Test accuracy
1	0.01	79.62	84.2
1	0.1	86.71	87.2
1	0.2	86.84	87.2
1	0.5	84.75	81.4
1	1	82.4	82.8
1	10	66.33	65.6

Figure 2.13: Table for the graph

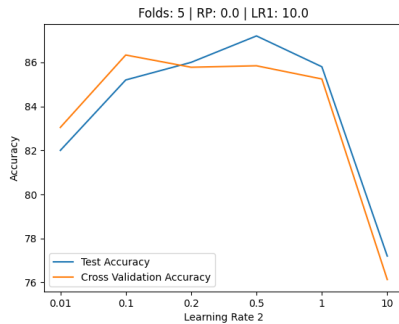


Figure 2.14: graph for $LR1 = 10$

LR1	LR2	CV accuracy	Test accuracy
10	0.01	83.04	82
10	0.1	86.33	85.2
10	0.2	85.77	86
10	0.5	85.84	87.2
10	1	85.24	85.8
10	10	76.13	77.2

Figure 2.15: Table for the graph

Interpretation: From the 6 graphs above we have found the best learning rate for the second layer for each first learning rate simply by comparing the Cross-Validation accuracies. We then run the model for a longer time (4000 iterations) to obtain the train and test accuracy using which we have chosen the final set of learning rates. Below are the graphs for iteration vs accuracy for the best learning rate for each of the first layer learning rates:

Heat map interpretation: From heat map it is clear that 0.01 is not an acceptable rate for either of the first and second learning rates. Also it can be established that 10 is not acceptable for second learning rate. Now if we chop of the first column of the heat map we can see that the brightness of the yellow increases as we go up and left. This shows that learning rate 1 around 1 or 0.5 is good while learning rate 2 around 0.1 and 0.2 is coming out to be good.



Figure 2.16: Heat Map for accuracy over learning rates

LR1	LR2	Train accuracy	Test accuracy
0.01	0.5	75.91	75.2
0.1	0.2	86.24	85.2
0.2	0.1	90.55	87
0.5	0.1	91.22	87.4
1	0.2	91.11	86.2
10	0.1	83.15	79.2

Figure 2.17: Table for the graphs below

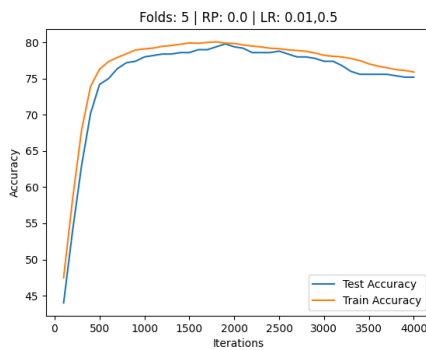


Figure 2.18: graph for LR = [0.01,0.5]

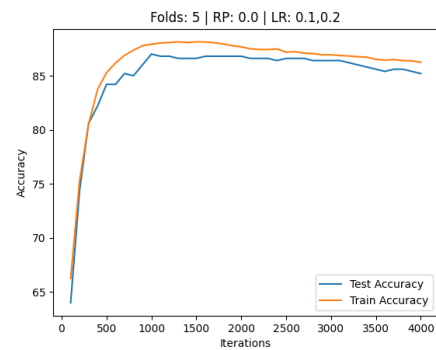


Figure 2.19: graph for LR = [0.1,0.2]



Figure 2.20: graph for LR = [0.2,0.1]

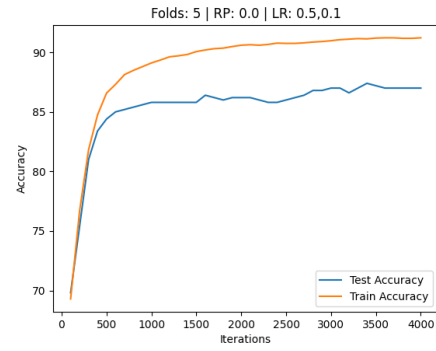


Figure 2.21: graph for LR = [0.5,0.1]

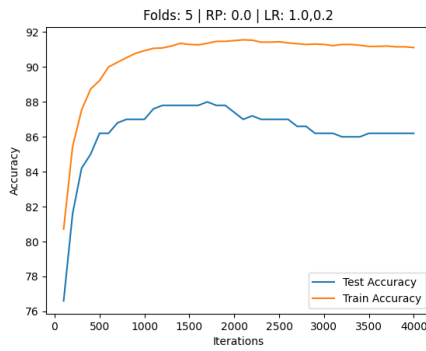


Figure 2.22: graph for LR = [1,0.2]

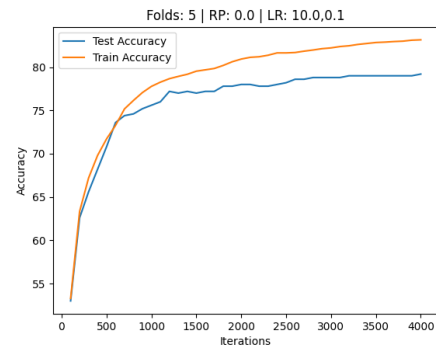


Figure 2.23: graph for LR = [10,0.1]

2.1.4 Conclusion:

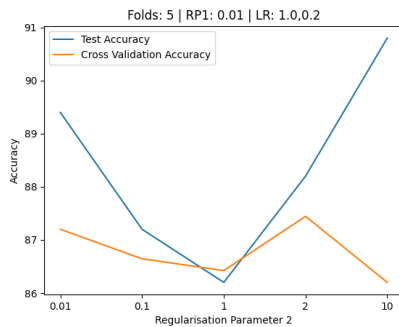
Using the results above we conclude that the optimal learning rate parameters using 5 fold cross validation are [0.5,0.1]. We have obtained a testing accuracy of 87.4 % when tested over 500 unseen data points and trained over the rest.

2.2 With Regularization

2.2.1 Finding the best regularization parameter

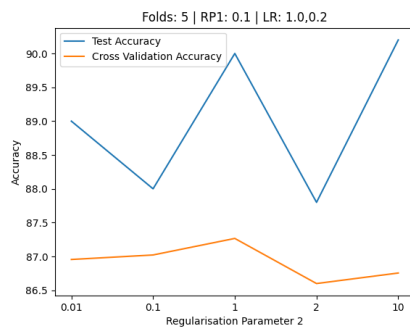
We have selected $[1.0, 0.2]$, $[1.0, 0.1]$, $[0.5, 0.1]$ as the three best learning rates from the section above and we will be finding the best regularisation parameters individually across these learning rates and then compare the cross-validation accuracies across them.

Learning rate: $[1.0, 0.2]$



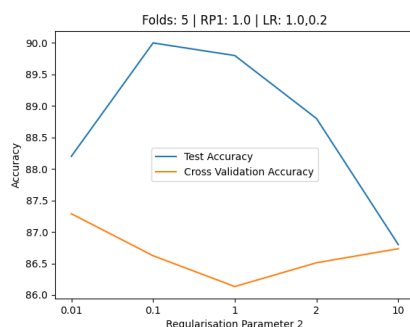
Reg Para 2	CV Accuracy	Test Accuracy
0.01	87.2	89.4
0.1	86.64	87.2
1	86.42	86.2
2	87.44	88.2
10	86.2	90.8

Figure 2.24: Table for Reg Para 1 : 0.01



Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.96	89.0
0.1	87.02	88.0
1	87.27	90.0
2	86.6	87.8
10	86.76	90.2

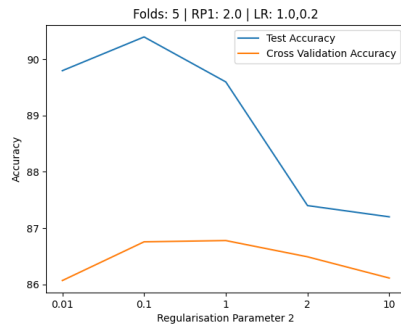
Figure 2.25: Table for Reg Para 1 : 0.1



Reg Para 2	CV Accuracy	Test Accuracy
0.01	87.29	88.2
0.1	86.62	90.0
1	86.13	89.8
2	86.51	88.8
10	86.73	86.8

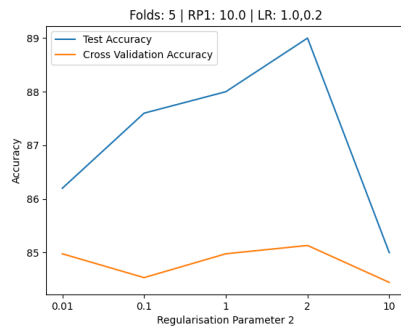
Figure 2.26: Table for Reg Para 1 : 1.0

Question 2



Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.07	89.8
0.1	86.76	90.4
1	86.78	89.6
2	86.49	87.4
10	86.11	87.2

Figure 2.27: Table for Reg Para 1 : 2.0



Reg Para 2	CV Accuracy	Test Accuracy
0.01	84.98	86.2
0.1	84.53	87.6
1	84.98	88.0
2	85.13	89.0
10	84.44	85.0

Figure 2.28: Table for Reg Para 1 : 10.0

Therefore, using the learning rate [1.0,0.2], the maximum cross-validation accuracy is obtained for the regularisation parameter [1,2]. We have trained the model for a longer time (4000 iterations) using these parameters and have achieved a testing set accuracy of 90.2 %.

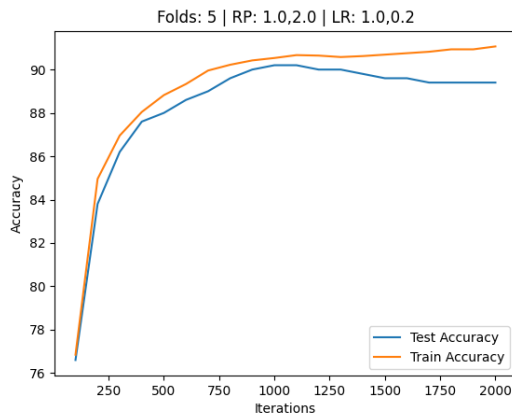


Figure 2.29: Accuracy Vs Iterations

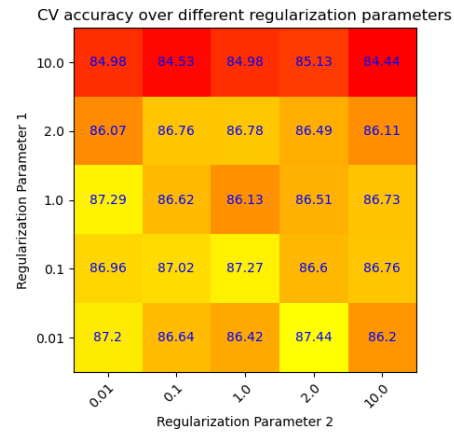
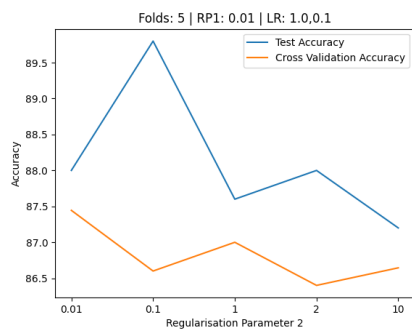


Figure 2.30: Heat Map for LR 1.0,0.2

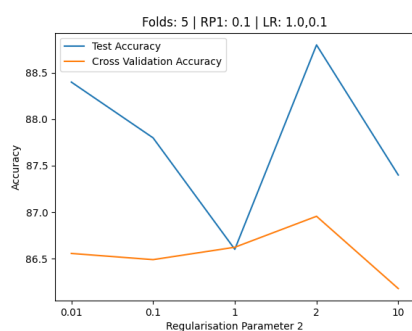
Heat Map: From Heat map too it is clear that [1.0,2.0] is a good regularisation parameter. Also it is clear that 10 is not a good reg parameter for either of the layers while 2 is not a good parameter for the first layer.

Learning rate: [1.0,0.1]



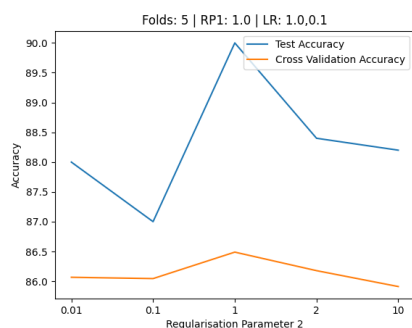
Reg Para 2	CV Accuracy	Test Accuracy
0.01	87.44	88.0
0.1	86.6	89.8
1	87.0	87.6
2	86.4	88.0
10	86.64	87.2

Figure 2.31: Table for Reg Para 1 : 0.01



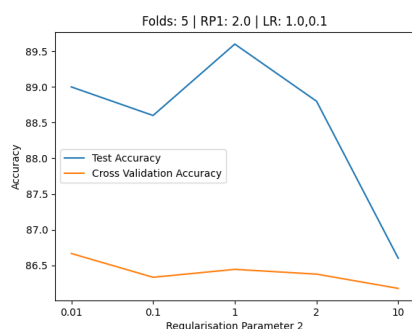
Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.56	88.4
0.1	86.49	87.8
1	86.62	86.6
2	86.96	88.8
10	86.18	87.4

Figure 2.32: Table for Reg Para 1 : 0.1



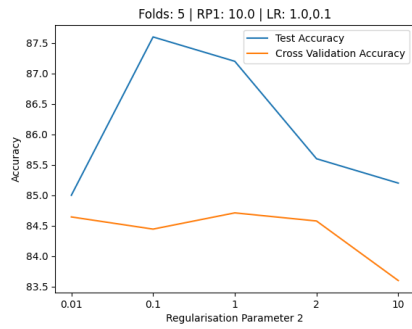
Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.07	88.0
0.1	86.04	87.0
1	86.49	90.0
2	86.18	88.4
10	85.91	88.2

Figure 2.33: Table for Reg Para 1 : 1.0



Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.67	89.0
0.1	86.33	88.6
1	86.44	89.6
2	86.38	88.8
10	86.18	86.6

Figure 2.34: Table for Reg Para 1 : 2.0



Reg Para 2	CV Accuracy	Test Accuracy
0.01	84.64	85.0
0.1	84.44	87.6
1	84.71	87.2
2	84.58	85.6
10	83.6	85.2

Figure 2.35: Table for Reg Para 1 : 10.0

Therefore, using the learning rate [1.0,0.1], the maximum cross-validation accuracy is obtained for the regularisation parameter [0.01,2]. We have trained the model for a longer time (4000 iterations) using these parameters and have achieved a testing set accuracy of 92.4 %.



Figure 2.36: Accuracy Vs Iterations

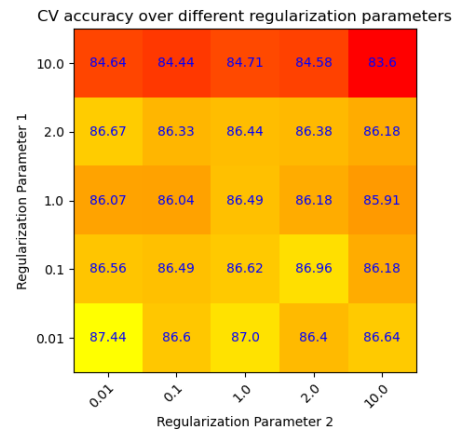
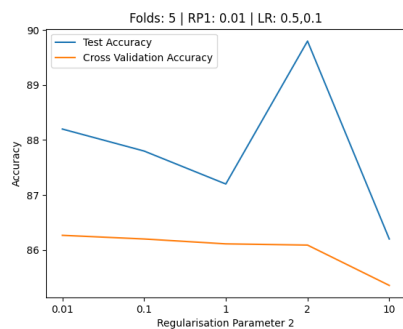


Figure 2.37: Heat Map for 1.0,0.1 LR

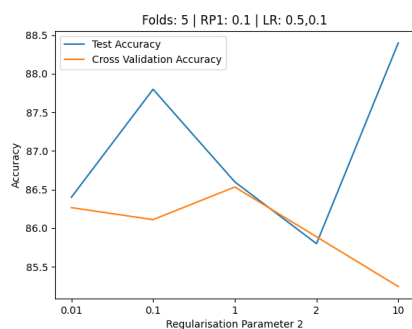
Heat Map: It can also be confirmed from the heat map that [0.01,2] is a good regularization parameter and also we can again see that high reg parameter for the first layer is not recommended hence the better reg parameters are being found in the lower half of the heat map that is when the first regularization parameter is 0.1 or 0.01.

Learning rate: [0.5,0.1]



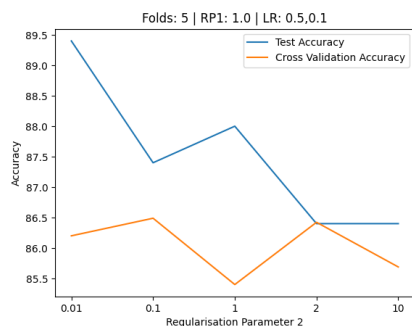
Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.27	88.2
0.1	86.2	87.8
1	86.11	87.2
2	86.09	89.8
10	85.36	86.2

Figure 2.38: Table for Reg Para 1 : 0.01



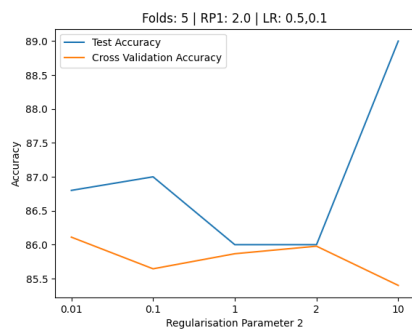
Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.27	86.4
0.1	86.11	87.8
1	86.53	86.6
2	85.89	85.8
10	85.24	88.4

Figure 2.39: Table for Reg Para 1 : 0.1



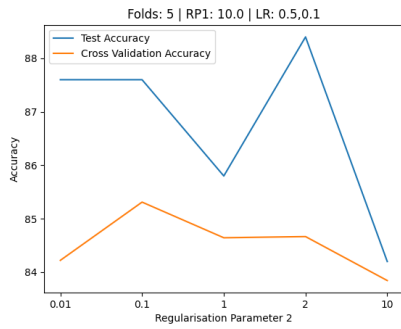
Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.2	89.4
0.1	86.49	87.4
1	85.4	88.0
2	86.42	86.4
10	85.69	86.4

Figure 2.40: Table for Reg Para 1 : 1.0



Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.11	86.8
0.1	85.64	87.0
1	85.87	86.0
2	85.98	86.0
10	85.4	89.0

Figure 2.41: Table for Reg Para 1 : 2.0



Reg Para 2	CV Accuracy	Test Accuracy
0.01	84.22	87.6
0.1	85.31	87.6
1	84.64	85.8
2	84.67	88.4
10	83.84	84.2

Figure 2.42: Table for Reg Para 1 : 10.0

Therefore, using the learning rate $[0.5,0.1]$, the maximum cross-validation accuracy is obtained for the regularisation parameter $[0.1,0.1]$. We have trained the model for a longer time (4000 iterations) using these parameters and have achieved a testing set accuracy of 90.4 %.

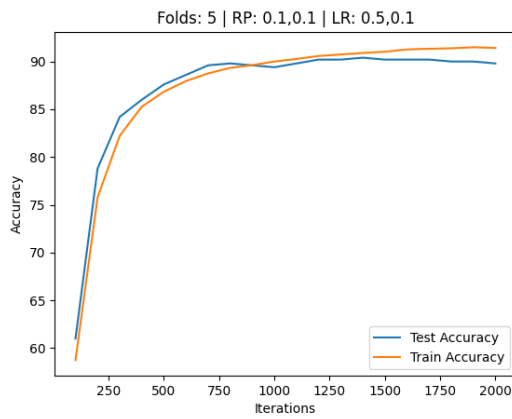


Figure 2.43: Accuracy Vs Iterations

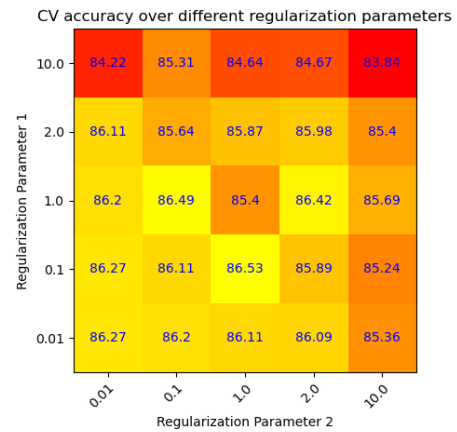


Figure 2.44: Heat Map for LR 0.5,0.1

Heat Map: The heat map again suggests that the first reg parameter must be below 2 while the second reg para must be less than 10. Also $[0.1,0.1]$ is the best reg parameter is confirmed by the heat map.

2.2.2 Conclusion:

Eventually, on comparing the best models across the learning rates, we conclude that the best set of parameters using 5-fold cross validation are $[1.0,0.1]$ as the learning rates and $[0.01,2]$ as the regularisation parameters with 25 hidden layer nodes. We have achieved a testing accuracy of 92.4 % by testing the model on 500 data points and training on the rest.

Chapter 3

Ten Fold Validation

3.1 Without Regularization

3.1.1 Introduction:

Now we are going to find the best model from 10-fold validation. For this again we are first going to find the best number of nodes in the hidden layer by comparing the CV accuracy across various number of nodes in the hidden layer (2,10,25,50,100,200,392) for some good learning rates. Using the heat map we are going to choose the best number of nodes. Once we have the best number of nodes then we are going to use that number of nodes to investigate the best learning rate across 36 different learning rates. Then we will move to regularization.

3.1.2 Finding the optimal number on nodes in hidden layer

This section deals with finding the optimum number of nodes in the hidden layer. We have selected the best learning rates that we got from the preliminary runs and used it to compare the 7 hidden layer nodes option. The following heatmap is to find the best number of nodes in the hidden layer.

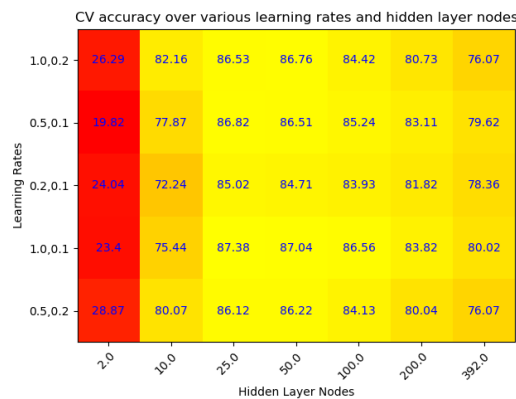


Figure 3.1: Heat Map showing 25 and 50 as best number of nodes

Through the heat map it can be easily be seen that 25 and 50 are the best number of hidden layer nodes but one careful examination one can see 25 edges out 50 generally hence we are going to use 25 as are number of hidden layer nodes.

Below is the table used for the heat map above

L Rate 1	L Rate 2	Nodes	Accuracy
0.5	0.2	2	28.86
0.5	0.2	10	80.66
0.5	0.2	25	86.02
0.5	0.2	50	86.22
0.5	0.2	100	84.13
0.5	0.2	200	80.04
0.5	0.2	392	76.06
1	0.1	2	23.39
1	0.1	10	75.44
1	0.1	25	86.37
1	0.1	50	87.04
1	0.1	100	86.55
1	0.1	200	83.82
1	0.1	392	80.02
0.2	0.1	2	24.04
0.2	0.1	10	72.24
0.2	0.1	25	83.02
0.2	0.1	50	84.71

Figure 3.2: heat map 1 - 18 entries

L Rate 1	L Rate 2	Nodes	Accuracy
0.2	0.1	100	83.93
0.2	0.1	200	81.82
0.2	0.1	392	78.35
0.5	0.1	2	19.82
0.5	0.1	10	77.86
0.5	0.1	25	83.82
0.5	0.1	50	86.51
0.5	0.1	100	85.24
0.5	0.1	200	83.11
0.5	0.1	392	79.62
1	0.2	2	26.28
1	0.2	10	82.15
1	0.2	25	86.53
1	0.2	50	86.75
1	0.2	100	84.42
1	0.2	200	80.73
1	0.2	392	76.06
X	X	X	X

Figure 3.3: heat map 19 - 35 entries

Interpretations: From the heat map it can be seen clearly that the accuracy increases as we increase the number of nodes in the hidden layer from 2 to 25 after which it decreases and decreases till 392 nodes. This can be seen from the shade of columns of each number of nodes. And 25 is coming out to be the best among all the other models.

Conclusion: There was a clear demarcation between the column of 25 and 50 hidden nodes as they were the brightest yellow. Signifying that they were the best among all the options. On closer examination we can see that 25 edges out 50 most of the times hence giving 25 as the best number of nodes for the hidden layer.

3.1.3 Finding the best learning rate:

In this section we are going to find the best learning rate for the model with 25 hidden layer nodes. We have varied the learning rate 1 and learning rate 2 across values: 0.01,0.1,0.2,0.5,1,10 (6 rates) giving a total 36 models to compare from to find the best learning rate from them. For this we have chosen the best learning rate 2 for a given learning rate 1.

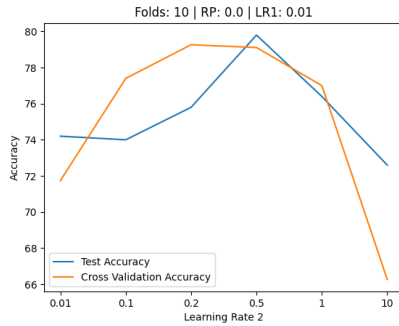


Figure 3.4: graph for LR1 = 0.01

LR1	LR2	CV accuracy	Train accuracy
0.01	0.01	71.73	74.2
0.01	0.1	77.40	74
0.01	0.2	79.26	75.8
0.01	0.5	79.11	79.80
0.01	1	77	76.4
0.01	10	66.26	72.6

Figure 3.5: Table for the graph

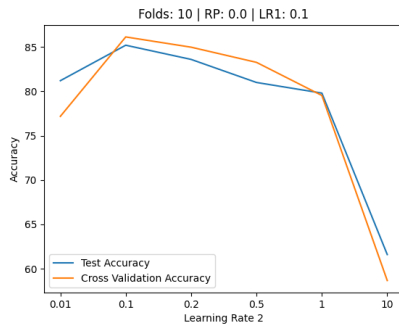


Figure 3.6: graph for LR1 = 0.1

LR1	LR2	CV accuracy	Train accuracy
0.1	0.01	77.17	81.2
0.1	0.1	86.13	85.2
0.1	0.2	84.97	83.6
0.1	0.5	83.26	81
0.1	1	79.53	79.80
0.1	10	58.66	61.6

Figure 3.7: Table for the graph

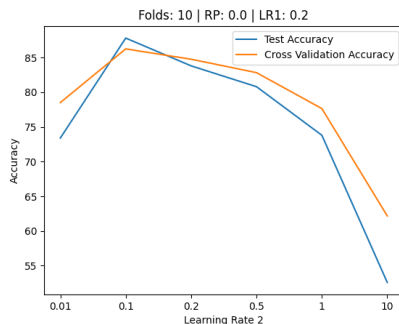


Figure 3.8: graph for LR1 = 0.2

LR1	LR2	CV accuracy	Train accuracy
0.2	0.01	78.51	73.4
0.2	0.1	86.24	87.8
0.2	0.2	84.75	83.8
0.2	0.5	82.82	80.8
0.2	1	77.64	73.8
0.2	10	62.17	52.6

Figure 3.9: Table for the graph

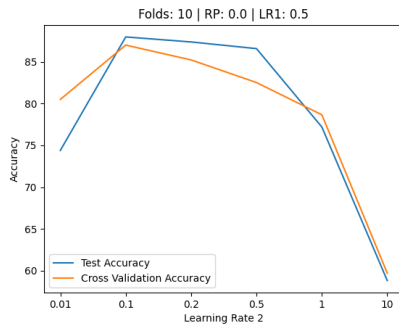


Figure 3.10: graph for $LR1 = 0.5$

LR1	LR2	CV accuracy	Train accuracy
0.5	0.01	80.51	74.4
0.5	0.1	87.02	88
0.5	0.2	85.24	87.4
0.5	0.5	82.53	86.6
0.5	1	78.66	77.2
0.5	10	59.68	58.8

Figure 3.11: Table for the graph

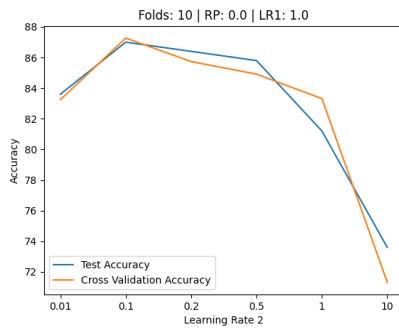


Figure 3.12: graph for $LR1 = 1$

LR1	LR2	CV accuracy	Train accuracy
1	0.01	83.24	83.6
1	0.1	87.26	87
1	0.2	85.73	86.4
1	0.5	84.91	85.8
1	1	83.31	81.2
1	10	71.31	73.6

Figure 3.13: Table for the graph

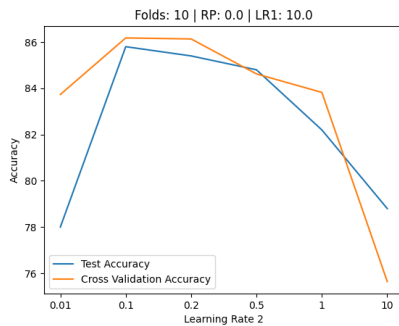


Figure 3.14: graph for $LR1 = 10$

LR1	LR2	CV accuracy	Train accuracy
10	0.01	83.73	78
10	0.1	86.17	85.8
10	0.2	86.13	85.39
10	0.5	84.62	84.8
10	1	83.82	82.2
10	10	75.64	78.8

Figure 3.15: Table for the graph

Interpretation: From the 6 graphs above we have found the best learning rate for the second layer for each first learning rate simply by comparing the Cross-Validation accuracies. We then run the model for a longer time (4000 iterations) to obtain the train and test accuracy using which we have chosen the final set of learning rates. Below are the graphs for iteration vs accuracy for the best learning rate for each of the first layer learning rates:

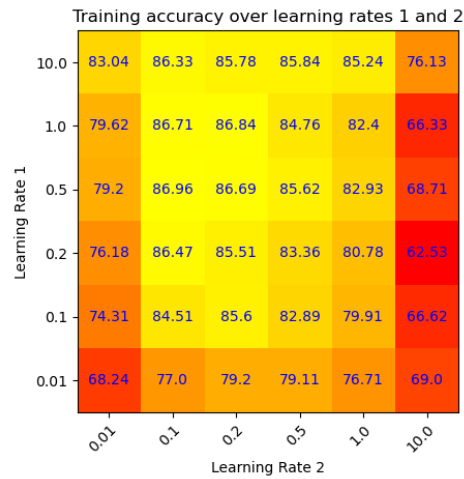


Figure 3.16: Heat Map for accuracy over learning rates

Heat map interpretation: From heat map it is clear that 0.01 is not an acceptable rate for either of the first and second learning rates. Also it can be established that 10 is not acceptable for second learning rate. Now if we chop of the first column of the heat map we can see that the brightness of the yellow increases as we go up and left. This shows that learning rate 1 around 1 or 0.5 is good while learning rate 2 around 0.1 and 0.2 is coming out to be good.

Iterations vs Accuracy: Below lies the iterations vs the accuracy graph for the best models we got for each of the first learning rates.

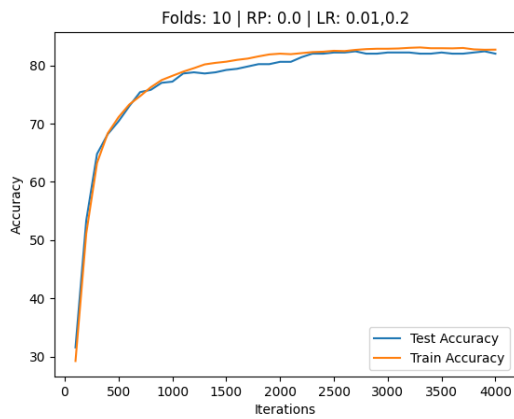


Figure 3.17: graph for LR = [0.01,0.2]

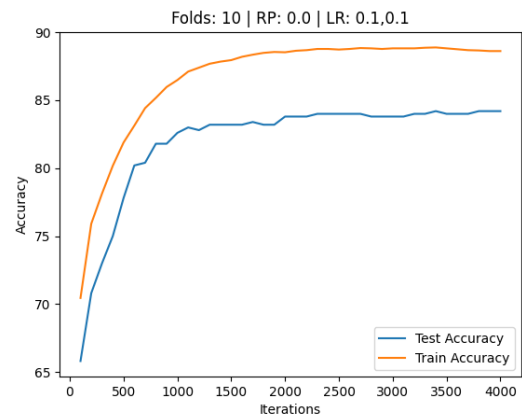


Figure 3.18: graph for LR = [0.1,0.1]

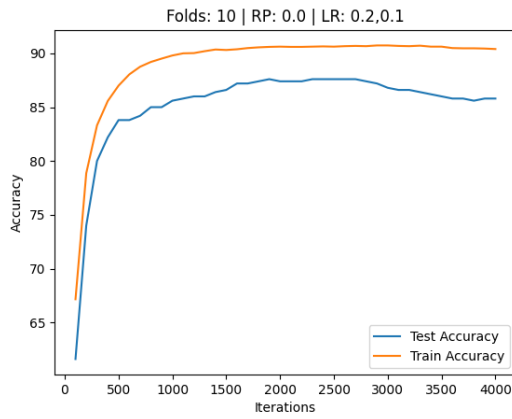


Figure 3.19: graph for LR = [0.2,0.1]

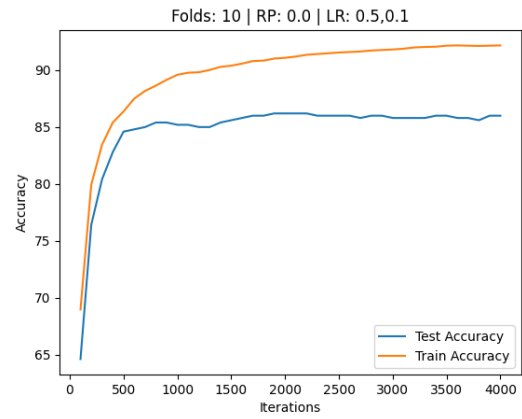


Figure 3.20: graph for LR = [0.5,0.1]

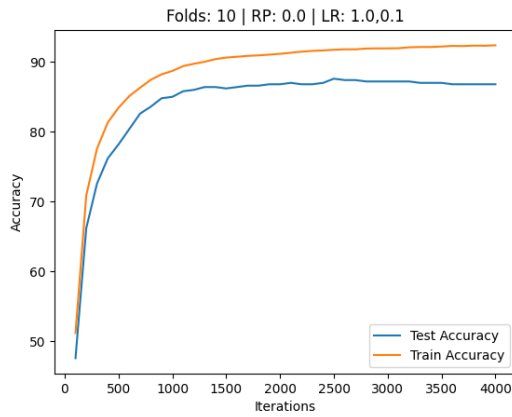


Figure 3.21: graph for LR = [1,0.1]



Figure 3.22: graph for LR = [10,0.1]

3.1.4 Conclusion:

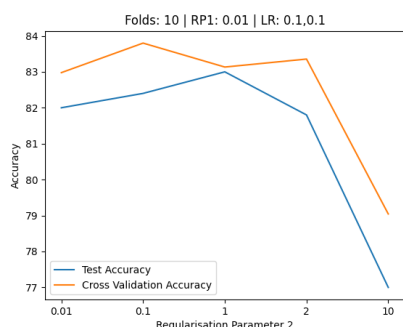
Using the results above we conclude that the optimal learning rate parameters using 10 fold cross validation are [1,0.1]. We have obtained a testing accuracy of 86.8 % when tested over 500 unseen data points and trained over the rest.

3.2 With Regularization

3.2.1 Finding the best regularization parameter:

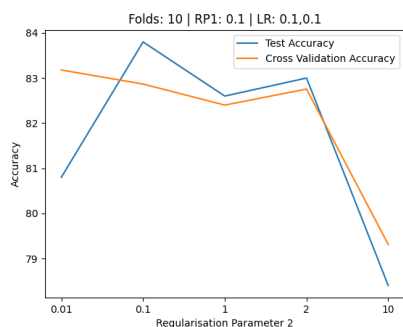
We have selected $[0.1, 0.1]$, $[0.5, 1]$, $[1, 0.1]$ as the three best learning rates from the section above and we will be finding the best regularisation parameters individually across these learning rates and then comparing the cross-validation accuracies across them.

Learning rate: $[0.1, 0.1]$



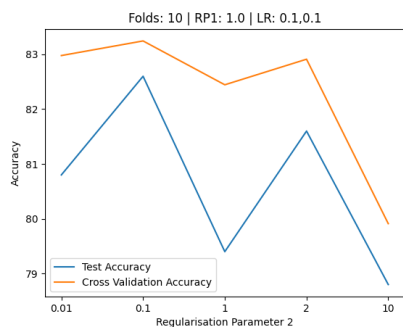
Reg Para 2	CV Accuracy	Test Accuracy
0.01	82.98	82.0
0.1	83.8	82.4
1.0	83.13	83.0
2.0	83.36	81.8
10.0	79.04	77.0

Figure 3.23: Table for Reg Para 1 : 0.01



Reg Para 2	CV Accuracy	Test Accuracy
0.01	83.18	80.8
0.1	82.87	83.8
1.0	82.4	82.6
2.0	82.76	83.0
10.0	79.31	78.4

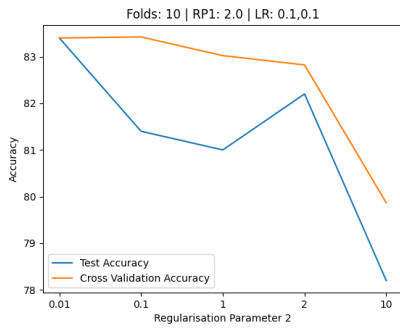
Figure 3.24: Table for Reg Para 1 : 0.1



Reg Para 2	CV Accuracy	Test Accuracy
0.01	82.98	80.8
0.1	83.24	82.6
1.0	82.44	79.4
2.0	82.91	81.6
10.0	79.91	78.8

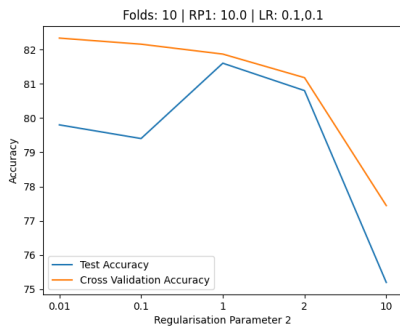
Figure 3.25: Table for Reg Para 1 : 1.0

Question 3



Reg Para 2	CV Accuracy	Test Accuracy
0.01	83.4	83.4
0.1	83.42	81.4
1.0	83.02	81.0
2.0	82.82	82.2
10.0	79.87	78.2

Figure 3.26: Table for Reg Para 1 : 2.0



Reg Para 2	CV Accuracy	Test Accuracy
0.01	82.33	79.8
0.1	82.16	79.4
1.0	81.87	81.6
2.0	81.18	80.8
10.0	77.44	75.2

Figure 3.27: Table for Reg Para 1 : 10.0

Therefore, using the learning rate $[0.1,0.1]$, the maximum cross-validation accuracy is obtained for the regularisation parameter $[0.01,0.01]$. We have trained the model for a longer time (4000 iterations) using these parameters and have achieved a testing set accuracy of 90.8 %.

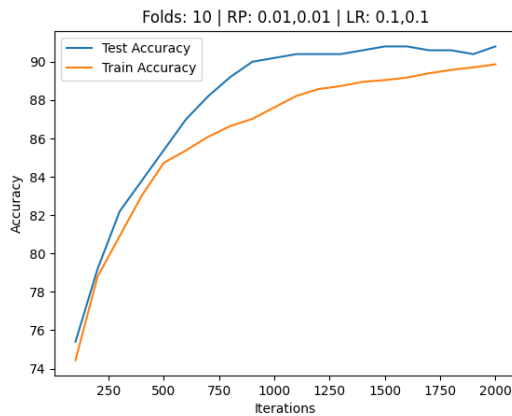


Figure 3.28: Accuracy Vs Iterations

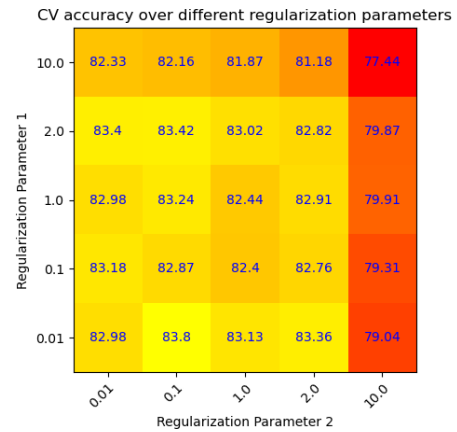
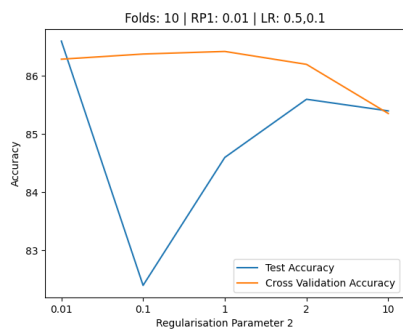


Figure 3.29: Heat Map for LR 0.1,0.1

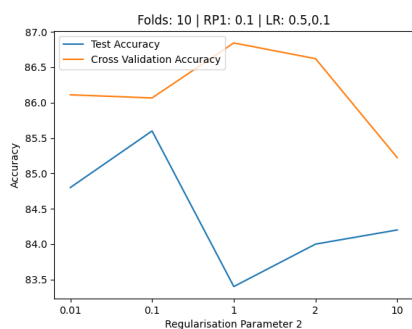
Heat Map: The heat map again suggests that the first reg parameter must be below 2 while the second reg para must be less than 10. Also $[0.01,0.01]$ is the best reg parameter is confirmed by the heat map.

Learning rate: [0.5,0.1]



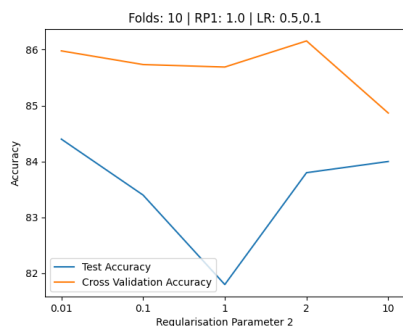
Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.29	86.6
0.1	86.38	82.4
1.0	86.42	84.6
2.0	86.2	85.6
10.0	85.36	85.4

Figure 3.30: Table for Reg Para 1 : 0.01



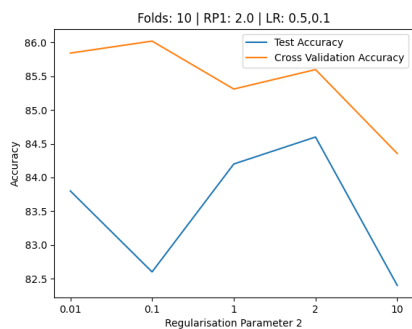
Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.11	84.8
0.1	86.07	85.6
1.0	86.84	83.4
2.0	86.62	84.0
10.0	85.22	84.2

Figure 3.31: Table for Reg Para 1 : 0.1



Reg Para 2	CV Accuracy	Test Accuracy
0.01	85.98	84.4
0.1	85.73	83.4
1.0	85.69	81.8
2.0	86.16	83.8
10.0	84.87	84.0

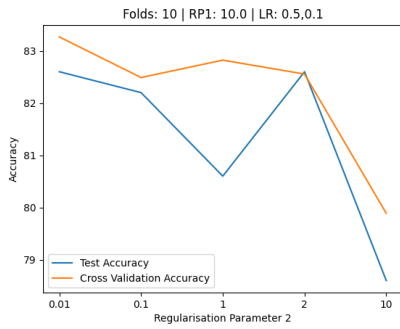
Figure 3.32: Table for Reg Para 1 : 1.0



Reg Para 2	CV Accuracy	Test Accuracy
0.01	85.84	83.8
0.1	86.02	82.6
1.0	85.31	84.2
2.0	85.6	84.6
10.0	84.36	82.4

Figure 3.33: Table for Reg Para 1 : 2.0

Question 3



Reg Para 2	CV Accuracy	Test Accuracy
0.01	83.27	82.6
0.1	82.49	82.2
1.0	82.82	80.6
2.0	82.56	82.6
10.0	79.89	78.6

Figure 3.34: Table for Reg Para 1 : 10.0

Therefore, using the learning rate $[0.5,0.1]$, the maximum cross-validation accuracy is obtained for the regularisation parameter $[2.0,0.01]$. We have trained the model for a longer time (2000 iterations) using these parameters and have achieved a testing set accuracy of 89.8 %.

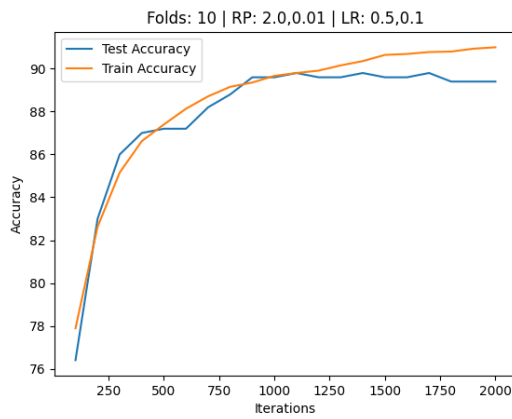


Figure 3.35: Accuracy Vs Iterations

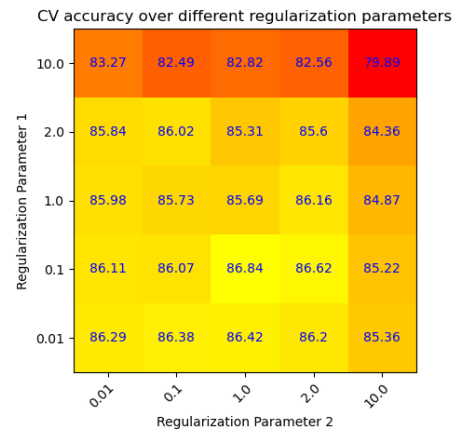
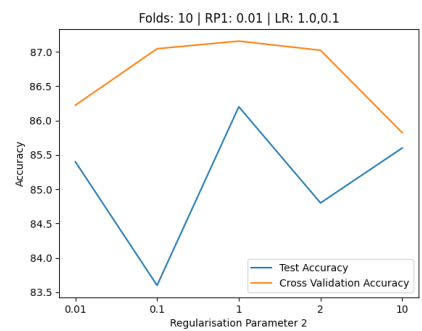


Figure 3.36: Heat Map for LR 0.5,0.1

Heat Map: The heat map again suggests that the first reg parameter must be below 2 while the second reg para must be less than 10. Also $[2,0.1]$ is the best reg parameter is confirmed by the heat map.

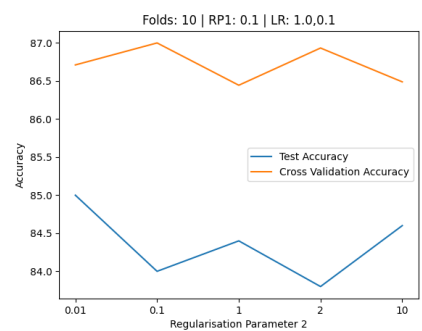
Question 3

Learning rate: [1.0,0.1]



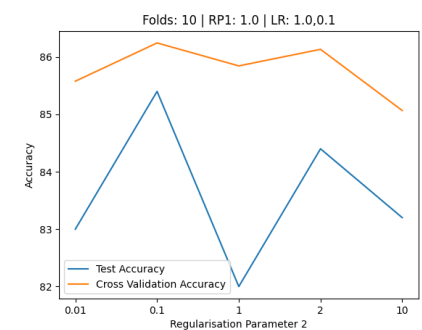
Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.22	85.4
0.1	87.04	83.6
1.0	87.16	86.2
2.0	87.02	84.8
10.0	85.82	85.6

Figure 3.37: Table for Reg Para 1 : 0.01



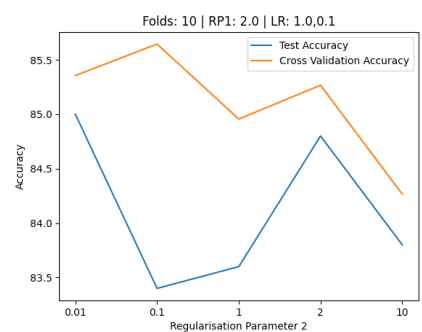
Reg Para 2	CV Accuracy	Test Accuracy
0.01	86.71	85.0
0.1	87.0	84.0
1.0	86.44	84.4
2.0	86.93	83.8
10.0	86.49	84.6

Figure 3.38: Table for Reg Para 1 : 0.1



Reg Para 2	CV Accuracy	Test Accuracy
0.01	85.58	83.0
0.1	86.24	85.4
1.0	85.84	82.0
2.0	86.13	84.4
10.0	85.07	83.2

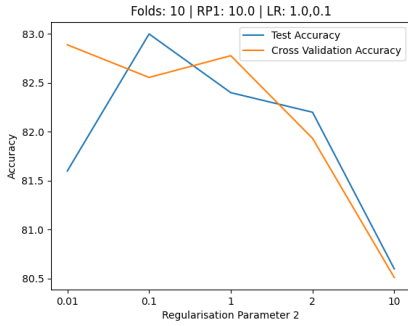
Figure 3.39: Table for Reg Para 1 : 1.0



Reg Para 2	CV Accuracy	Test Accuracy
0.01	85.36	85.0
0.1	85.64	83.4
1.0	84.96	83.6
2.0	85.27	84.8
10.0	84.27	83.8

Figure 3.40: Table for Reg Para 1 : 2.0

Question 3



Reg Para 2	CV Accuracy	Test Accuracy
0.01	82.89	81.6
0.1	82.56	83.0
1.0	82.78	82.4
2.0	81.93	82.2
10.0	80.51	80.6

Figure 3.41: Table for Reg Para 1 : 10.0

Therefore, using the learning rate [1.0,0.1], the maximum cross-validation accuracy is obtained for the regularisation parameter [0.01,2.0]. We have trained the model for a longer time (2000 iterations) using these parameters and have achieved a testing set accuracy of 90.2 %.

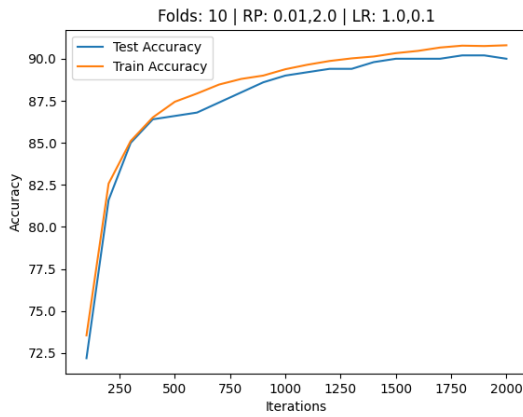


Figure 3.42: Accuracy Vs Iterations

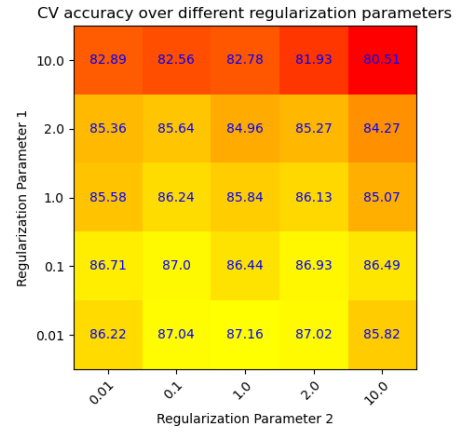


Figure 3.43: Heat Map for LR 1.0,0.1

Heat Map: The heat map again suggests that the first reg parameter must be below 2 while the second reg para must be less than 10. Also [0.01,2.0] is the best reg parameter is confirmed by the heat map.



















































3.2.2 Conclusion:

Eventually, on comparing the best models across the learning rates, we conclude that the best set of parameters using 10-fold cross validation are [0.1,0.1] as the learning rates and [0.01,0.01] as the regularisation parameters with 25 hidden layer nodes. We have achieved a testing accuracy of 90.8 % by testing the model on 500 data points and training on the rest.

Chapter 4

Data Visualisation

In this chapter we have used the best model found using cross validation techniques above and used those parameters to visualise the data. We have randomly selected 50 data points from the testing data set and used the input pixel values to visualise the digit it represents and compared it with our predictions.

Pred :[0.] 	Pred :[7.] 	Pred :[1.] 	Pred :[2.] 	Pred :[4.] 
Pred :[4.] 	Pred :[8.] 	Pred :[8.] 	Pred :[7.] 	Pred :[1.] 
Pred :[4.] 	Pred :[2.] 	Pred :[5.] 	Pred :[8.] 	Pred :[2.] 
Pred :[3.] 	Pred :[3.] 	Pred :[5.] 	Pred :[5.] 	Pred :[9.] 
Pred :[0.] 	Pred :[9.] 	Pred :[4.] 	Pred :[8.] 	Pred :[5.] 
Pred :[7.] 	Pred :[3.] 	Pred :[0.] 	Pred :[6.] 	Pred :[9.] 
Pred :[8.] 	Pred :[2.] 	Pred :[6.] 	Pred :[4.] 	Pred :[9.] 
Pred :[4.] 	Pred :[9.] 	Pred :[4.] 	Pred :[0.] 	Pred :[2.] 
Pred :[5.] 	Pred :[0.] 	Pred :[2.] 	Pred :[8.] 	Pred :[7.] 
Pred :[0.] 	Pred :[8.] 	Pred :[7.] 	Pred :[2.] 	Pred :[8.] 

Observations: We observed some interesting decisions which the model makes while predicting digits.

Pred :[7.]



Pred :[5.]



Pred :[8.]



- The first image has a clear confusion between the digits 1 and 7. But the model predicts 7 since the image has a little horizontal line on the top which increases the probability of the image being a 7 rather than a 1.
- The second image is a confusion between 5 and 6 since the lower end of the image almost touches the upper part making the 5 look like a 6.
- The third image is incorrectly predicted by the model. This is because 5 looks very similar to 8 here since the image looks slightly compressed in height making the upper and lower ends look attached and hence imparting the image features of the digit 8.