**Task 1(a)**

**Mean Square Error**

Table I for Mean Squared error and optimizer is SGD (stochastic gradient descent)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Parameters | | | | | | | Results |
| Hidden layer | Units  (Output\_dim) | Learning rate | Momentum rates | Epoch | Decay | Batch  size | Loss & Accu  (train data) |
| 1 | 32 | 0.01 | 0.9 | 50 | 1e-6 | 5 | loss: 0.0031 acc: 0.9869 |
| 1 | 50 | 0.01 | 0.95 | 50 | 1e-4 | 5 | loss: 0.0032 acc: 0.9866 |
| 1 | 50 | 0.001 | 0.80 | 100 | 1e-2 | 10 | loss: 0.0031 acc: 0.9876 |
| 2 | 32, 32 | 0.01 | 0.90 | 100 | 1e-6 | 10 | loss: 0.0076 acc: 0.9578 |
| 2 | 52, 52 | 0.01 | 0.95 | 150 | 1e-2 | 10 | loss: 0.0036 acc: 0.9840 |
| 3 | 45, 45, 45 | 0.10 | 0.90 | 100 | 1e-6 | 5 | loss: 0.0161 acc: 0.8999 |
| 3 | 40, 42, 45 | 0.01 | 0.70 | 100 | 1e-3 | 10 | loss: 0.0897 acc: 0.2162 |
| 3 | 100,142,160 | 0.0314 | 0.95 | 100 | 1e-6 | 10 | loss: 0.0065 acc: 0.9627 |

Table II: Best parameters config for Sum-of-squares error function

|  |  |
| --- | --- |
| Config | Result |
|  |  |
| Num of hidden layers =1 | Execution time: 28.746 Sec |
| Num of hidden layer units =50 | Accuracy (train data): 97.385 |
| Learning rates = 0.001, | Loss (train data): 0.00464 |
| Momentum rates= 0.80, | Accuracy (test data): 94.769 |
| Activation function="ReLU" | Loss (test data): 0.0080 |
| Epoch=100  Decay = 1e-2  Batch\_size = 10 |  |
|  |  |

CONFUSION MATRIX TRAINING DATA

[[60 0 1 0 0 0 0 0 0 0]

[ 0 81 0 0 0 1 0 0 0 1]

[ 0 0 82 0 0 1 1 0 0 0]

[ 0 0 0 92 0 4 0 0 0 0]

[ 1 0 0 0 58 0 1 0 1 0]

[ 0 0 0 0 0 69 0 0 0 0]

[ 0 0 0 0 0 0 70 0 0 0]

[ 0 0 0 0 0 0 0 96 0 0]

[ 0 1 0 1 1 1 0 0 62 0]

[ 0 1 0 0 1 0 0 1 1 75]]

TRAINING DATA class Accuracy

Accuracy of Class 0 is 0.9973226238286479

Accuracy of Class 1 is 0.9946595460614153

Accuracy of Class 2 is 0.9959893048128342

Accuracy of Class 3 is 0.9933333333333333

Accuracy of Class 4 is 0.9933333333333333

Accuracy of Class 5 is 0.9906914893617021

Accuracy of Class 6 is 0.9973226238286479

Accuracy of Class 7 is 0.9986595174262735

Accuracy of Class 8 is 0.9920106524633822

Accuracy of Class 9 is 0.9933333333333333

CONFUSION MATRIX TESTING DATA

[[176 0 0 0 0 2 0 0 0 0]

[ 0 172 0 0 0 1 0 0 2 7]

[ 0 8 169 0 0 0 0 0 0 0]

[ 0 0 3 169 0 3 0 2 3 3]

[ 1 2 0 0 176 0 1 0 1 0]

[ 0 0 1 0 0 180 0 0 0 1]

[ 1 4 0 0 0 0 175 0 1 0]

[ 0 0 0 0 2 7 0 164 1 5]

[ 0 10 0 0 0 5 1 0 153 5]

[ 0 2 0 1 4 3 0 0 1 169]]

TESTING DATA class Accuracy

Accuracy of Class 0 is 0.9976567076742824

Accuracy of Class 1 is 0.9792984473835538

Accuracy of Class 2 is 0.993002915451895

Accuracy of Class 3 is 0.9912689173457508

Accuracy of Class 4 is 0.9935822637106184

Accuracy of Class 5 is 0.9866743916570104

Accuracy of Class 6 is 0.9953243717124488

Accuracy of Class 7 is 0.9901162790697674

Accuracy of Class 8 is 0.9826889786497404

Accuracy of Class 9 is 0.9815561959654179

**Cross-Entropy Error (ReLu)**

Table I for Mean Squared error and optimizer is SGD (stochastic gradient descent)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Parameters | | | | | | | Results |
| Hidden layer | Units of hidden layer  (Output\_dim) | Learning rate | Momentum rates | Epoch | Decay | Batch  size | Loss & Accu  (train data) |
| 1 | 32 | 0.01 | 0.9 | 30 | 1e-6 | 5 | loss: 0.0067 acc: 0.9990 |
| 1 | 50 | 0.01 | 0.95 | 25 | 1e-4 | 5 | loss: 0.0063 acc: 0.9990 |
| 1 | 50 | 0.001 | 0.80 | 20 | 1e-2 | 10 | loss: 0.0276 acc: 0.9925 |
| 2 | 50, 40 | 0.01 | 0.90 | 30 | 1e-6 | 10 | loss: 0.0149 acc: 0.9957 |
| 2 | 52, 52 | 0.01 | 0.95 | 50 | 1e-6 | 10 | loss: 0.0043 acc: 0.9993 |
| 3 | 45, 45, 45 | 0.10 | 0.90 | 31 | 1e-6 | 5 | loss: 0.0211 acc: 0.9925 |
| 3 | 30, 50, 100 | 0.01 | 0.95 | 50 | 1e-4 | 10 | loss: 0.0033 acc: 1.0000 |

The last one seems to be the case for the overfitting so considering the one with 2 layer and 52,52 neurons in it.

Table II: Best parameters config for Cross-Entropy error function

|  |  |
| --- | --- |
| Config | Result |
|  |  |
| Num of hidden layers = 2 | Execution time: 16.61 Sec |
| Num of hidden layer units = 52, 52 | Accuracy (train data): 97.90 |
| Learning rates = 0.01, | Loss (train data): 0.105 |
| Momentum rates= 0.95,  Decay = 1e-6 | Accuracy (test data): 95.93 |
| Activation function="ReLU" | Loss (test data): 0.237 |
| Epoch=50  Batch\_size = 10 |  |
|  |  |

CONFUSION MATRIX TRAINING DATA

[[60 0 1 0 0 0 0 0 0 0]

[ 0 81 0 0 0 0 0 0 1 1]

[ 0 0 83 0 0 0 1 0 0 0]

[ 0 0 0 92 0 3 0 0 0 1]

[ 0 0 0 0 61 0 0 0 0 0]

[ 0 0 0 0 0 69 0 0 0 0]

[ 0 0 0 0 0 0 70 0 0 0]

[ 0 0 0 0 0 1 0 93 0 2]

[ 0 1 0 1 1 1 0 0 62 0]

[ 0 0 0 0 1 0 0 0 0 78]]

TRAINING DATA class Accuracy

Accuracy of Class 0 is 0.9986666666666667

Accuracy of Class 1 is 0.9960106382978723

Accuracy of Class 2 is 0.9973368841544608

Accuracy of Class 3 is 0.993368700265252

Accuracy of Class 4 is 0.9973368841544608

Accuracy of Class 5 is 0.993368700265252

Accuracy of Class 6 is 0.9986666666666667

Accuracy of Class 7 is 0.9960106382978723

Accuracy of Class 8 is 0.993368700265252

Accuracy of Class 9 is 0.993368700265252

CONFUSION MATRIX

[[176 0 0 0 0 2 0 0 0 0]

[ 0 181 0 0 0 0 0 0 1 0]

[ 0 4 172 0 0 0 1 0 0 0]

[ 0 0 2 174 0 3 0 0 1 3]

[ 0 0 0 0 177 0 1 0 3 0]

[ 0 0 1 0 0 176 0 0 1 4]

[ 0 0 0 0 2 0 178 0 1 0]

[ 0 0 0 0 1 10 0 158 1 9]

[ 0 8 0 2 0 1 1 0 159 3]

[ 0 0 0 1 1 4 0 0 1 173]]

TESTING DATA class Accuracy

Accuracy of Class 0 is 0.9988412514484357

Accuracy of Class 1 is 0.9925158318940702

Accuracy of Class 2 is 0.9953810623556582

Accuracy of Class 3 is 0.9930875576036866

Accuracy of Class 4 is 0.9953810623556582

Accuracy of Class 5 is 0.9851428571428571

Accuracy of Class 6 is 0.9965317919075144

Accuracy of Class 7 is 0.9879656160458453

Accuracy of Class 8 is 0.9862700228832952

Accuracy of Class 9 is 0.9851428571428571

In comparison to the mean\_squared\_error and the cross entropy error function, it can be seen through the observation that the crossentropy takes less time and provides better timing as compared to the mse.

In case of mean\_squared\_error, the Back propagation can take long time if the gradient is small but on the other hand in case of the crossentropy, due to the presence of the log it can deal with the small changes as well and thus preferred more.

**Task 1(b)**

The ReLu for cross-entropy function remains same as shown in Task1(a).

**Cross-Entropy Error (tanh)**

Table I for Mean Squared error and optimizer is SGD (stochastic gradient descent)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Parameters | | | | | | | Results |
| Hidden layer | Units of hidden layer  (Output\_dim) | Learning rate | Momentum rates | Epoch | Decay | Batch  size | Loss & Accu  (train data) |
| 1 | 32 | 0.01 | 0.9 | 50 | 1e-6 | 5 | loss: 0.0085 acc: 0.9993 |
| 1 | 50 | 0.01 | 0.95 | 30 | 1e-4 | 5 | loss: 0.0108 acc: 0.9990 |
| 1 | 50 | 0.001 | 0.80 | 50 | 1e-2 | 10 | loss: 0.0124 acc: 0.9990 |
| 2 | 50, 40 | 0.01 | 0.90 | 30 | 1e-6 | 10 | loss: 0.0191 acc: 0.9987 |
| 2 | 52, 52 | 0.01 | 0.95 | 50 | 1e-2 | 10 | loss: 0.0070 acc: 1.0000 |
| 3 | 45, 45, 45 | 0.10 | 0.90 | 31 | 1e-6 | 5 | loss: 0.0906 acc: 0.9729 |
| 3 | 30, 50, 100 | 0.01 | 0.95 | 50 | 1e-4 | 10 | loss: 0.0111 acc: 0.9980 |

Again not considering the case for the 100% accuracy as this could be overfitting due to more number of epochs run.

Considering the 1 hidden layer for 32 neurons

Table II: Best parameters config for Cross-Entropy error function

|  |  |
| --- | --- |
| Config | Result |
|  |  |
| Num of hidden layers =1 | Execution time: 28.56 Sec |
| Num of hidden layer units =32 | Accuracy (train data): 98.1699 |
| Learning rates = 0.01, | Loss (train data): 0.084 |
| Momentum rates= 0.90, | Accuracy (test data): 95.88 |
| Activation function="ReLU" | Loss (test data): 0.128 |
| Epoch=50  Decay = 1e-6  Batch\_size = 5 |  |

CONFUSION MATRIX TRAINING DATA

[[61 0 0 0 0 0 0 0 0 0]

[ 0 81 0 0 1 0 0 0 0 1]

[ 0 1 82 0 0 0 1 0 0 0]

[ 0 0 0 92 0 4 0 0 0 0]

[ 0 0 0 0 61 0 0 0 0 0]

[ 0 0 0 0 0 69 0 0 0 0]

[ 0 0 0 0 0 0 70 0 0 0]

[ 0 1 0 0 0 0 0 95 0 0]

[ 0 1 0 0 1 1 0 0 63 0]

[ 0 1 0 0 1 0 0 0 0 77]]

TRAINING DATA class Accuracy

Accuracy of Class 0 is 1.0

Accuracy of Class 1 is 0.9920739762219286

Accuracy of Class 2 is 0.99734395750332

Accuracy of Class 3 is 0.9947019867549669

Accuracy of Class 4 is 0.9960212201591512

Accuracy of Class 5 is 0.9933862433862434

Accuracy of Class 6 is 0.9986702127659575

Accuracy of Class 7 is 0.9986702127659575

Accuracy of Class 8 is 0.9960212201591512

Accuracy of Class 9 is 0.9960212201591512

CONFUSION MATRIX

[[176 0 0 0 0 2 0 0 0 0]

[ 0 178 0 0 0 0 0 0 3 1]

[ 0 7 170 0 0 0 0 0 0 0]

[ 0 0 3 172 0 3 0 2 0 3]

[ 0 0 0 0 178 0 0 0 3 0]

[ 0 0 1 0 0 178 0 0 0 3]

[ 0 2 0 0 2 0 176 0 1 0]

[ 0 0 0 0 0 3 0 167 1 8]

[ 0 7 0 0 0 5 1 0 156 5]

[ 0 0 0 2 2 3 0 0 1 172]]

TESTING DATA class Accuracy

Accuracy of Class 0 is 0.998840579710145

Accuracy of Class 1 is 0.9885255306942053

Accuracy of Class 2 is 0.9936562860438293

Accuracy of Class 3 is 0.9925115207373272

Accuracy of Class 4 is 0.9959537572254336

Accuracy of Class 5 is 0.9885255306942053

Accuracy of Class 6 is 0.9965297860034702

Accuracy of Class 7 is 0.9919401266551525

Accuracy of Class 8 is 0.9845714285714285

Accuracy of Class 9 is 0.9840091376356368

While considering the relu and tanh, it can be seen that the tanh has better learned capability for the same parameters as in case of relu. But the relu converged faster as compared to the tanh. Very close similarity can be seen in this as values are quite similar.

**Convolution neural network**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Parameters | | | | | | Results |
| Hidden layer | Units of hidden layer  (Output\_dim) | Learning rate | Epoch | filter | Batch  size | Loss & Accu  (train data) |
| 1 | 128 | 0.01 | 50 | 32 | 5 | loss: 0.0105 acc: 0.9961 |
| 1 | 128 | 0.01 | 50 | 64 | 5 | loss: 0.0043 acc: 0.9984 |
| 1 | 256 | 0.01 | 30 | 32 | 10 | loss: 0.0225 - acc: 0.9944 |
| 1 Along with 2 dropout to prevent overfitting | 128 | 0.01 | 30 | 32 | 10 | loss: 0.0668 - acc: 0.9797 |

Here I will be consider the hidden layer units as a power of 2. Therefore it has high values and then convolution network has performed well in the testing data with 98% as compared to the above neural network configuration.

Convolution neural network go through so many processes such as Convolution, Max pooling, Flattening and then fully connected neural network. To prevent overfitting I used Dropout with almost 20% neurons.

Table II: Best parameters config for Cross-Entropy error function

|  |  |
| --- | --- |
| Config | Result |
|  |  |
| Num of hidden layers = 1  Num of convolution layer = 1 | Execution time: 70.70 Sec |
| Num of hidden layer units = 128 | Accuracy (train data): 98.82 |
| Learning rates = 0.01, | Loss (train data): 0.074 |
| Momentum rates= 0.90,  Decay = 1e-6 | Accuracy (test data): 98.10 |
| Activation function="ReLU" | Loss (test data): 0.1283 |
| Epoch=50  Batch\_size = 5 |  |
|  |  |

CONFUSION MATRIX TRAINING DATA

[[61 0 0 0 0 0 0 0 0 0]

[ 0 80 0 1 0 0 2 0 0 0]

[ 0 0 83 0 0 0 1 0 0 0]

[ 0 0 0 95 0 1 0 0 0 0]

[ 0 0 0 0 60 0 1 0 0 0]

[ 0 0 0 0 0 69 0 0 0 0]

[ 0 0 0 0 0 0 70 0 0 0]

[ 0 0 0 0 0 0 0 96 0 0]

[ 0 0 0 0 0 0 0 1 65 0]

[ 0 0 0 0 1 0 0 1 0 77]]

TRAINING DATA class Accuracy

Accuracy of Class 0 is 1.0

Accuracy of Class 1 is 0.9960474308300395

Accuracy of Class 2 is 0.9986789960369881

Accuracy of Class 3 is 0.9973614775725593

Accuracy of Class 4 is 0.9973614775725593

Accuracy of Class 5 is 0.9986789960369881

Accuracy of Class 6 is 0.9947368421052631

Accuracy of Class 7 is 0.9973614775725593

Accuracy of Class 8 is 0.9986789960369881

Accuracy of Class 9 is 0.9973614775725593

CONFUSION MATRIX

[[178 0 0 0 0 0 0 0 0 0]

[ 0 180 0 0 0 0 1 0 0 1]

[ 0 0 174 0 1 0 0 1 1 0]

[ 0 0 0 180 0 1 0 2 0 0]

[ 0 0 0 0 177 0 1 0 0 3]

[ 0 0 0 0 0 179 0 0 0 3]

[ 0 0 0 0 0 0 181 0 0 0]

[ 0 0 0 0 0 0 0 167 0 12]

[ 0 0 0 0 0 0 0 1 170 3]

[ 0 1 0 0 0 1 0 0 1 177]]

TESTING DATA class Accuracy

Accuracy of Class 0 is 1.0

Accuracy of Class 1 is 0.9983012457531144

Accuracy of Class 2 is 0.9983012457531144

Accuracy of Class 3 is 0.9983012457531144

Accuracy of Class 4 is 0.9971719457013575

Accuracy of Class 5 is 0.9971719457013575

Accuracy of Class 6 is 0.9988668555240793

Accuracy of Class 7 is 0.9910061832490163

Accuracy of Class 8 is 0.996608253250424

Accuracy of Class 9 is 0.9860178970917226