

CS541 HW4 Screenshots

```
homework4_tamishra_anisal.py > findBestHyperparameters
258 best_epochs = besth[0] # best number of epochs
259 best_batch_size = besth[1] # best number of batch_size
260 best_hidden_num = besth[2] # best number of hidden neurons
261 best_h_layers = besth[3] # best number of hidden layers
262 best_learningrate = besth[4] # best learning rate
263 best_alpha = besth[5] # best alpha
264
265 weightsAndBiases = initWeightsAndBiases(best_hidden_num, best_h_layers)
266 weightsAndBiases=copy.deepcopy(bestw)
267 loss, h_h, h_z, yhat = forward_prop(testX, testY, weightsAndBiases, best_hidden_num, best_h_layers)
268
269 print("\nThe Best HyperParameters: \nHidden Layers:", best_h_layers, "\nHidden Layer Neurons: ", best_hidden_num,
270       "\nEpochs: ", best_epochs, "\nBatch size: ", best_batch_size)
271 print("\nLearning rate: ", best_learningrate, "\nAlpha: ", best_alpha)
```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

```
Loss: 0.36965639474551754 Accuracy: 88.88333333333334
Epoch no. 64
Loss: 0.3721489069228046 Accuracy: 88.78333333333333
Epoch no. 65
Loss: 0.3742978232311491 Accuracy: 88.78333333333333
Epoch no. 66
Loss: 0.37722781472526834 Accuracy: 88.75
Epoch no. 67
Loss: 0.3812414969384232 Accuracy: 88.8
Epoch no. 68
Loss: 0.3832383846438546 Accuracy: 88.78333333333333
Epoch no. 69
Loss: 0.3864690270695528 Accuracy: 88.71666666666667

The Best HyperParameters:
Hidden Layers: 3
Hidden Layer Neurons: 81
Epochs: 70
Batch size: 16
Learning rate: 0.005
Alpha: 1e-06
Accuracy (validation data) : 88.94166666666666

Min loss value: 0.3421147290727392

Accuracy on Test data: 88.12
```

Best Accuracy and hyperparameters along loss and accuracy values after each iteration

```
homework4_tamishra_anisal.py > findBestHyperparameters
317 def saveval(nested_list, name):
318     with open(name, 'w') as file:
319         for inner_list in nested_list:
320             line = ','.join(map(str, inner_list)) + '\n'
321             file.write(line)
322
323
324 if __name__ == "__main__":
325     # TODO: Load data and split into train, validation, test sets
326     X_tr = np.reshape(np.load("fashion_mnist_train_images.npy"), (-1, 28 * 28)) / 255
327     trainX = X_tr.T
328     ytr = np.load("fashion_mnist_train_labels.npy")
329     train_Y = ytr
330     X_te = np.reshape(np.load("fashion_mnist_test_images.npy"), (-1, 28 * 28)) / 255
331     testX = X_te.T
332     yte = np.load("fashion_mnist_test_labels.npy")
333     test_Y = yte
334
335     # onehot encoding
336     trainY = np.zeros((train_Y.size, train_Y.max() + 1))
337     testY = np.zeros((test_Y.size, test_Y.max() + 1))
338     trainY[np.arange(train_Y.size), train_Y] = 1
339     testY[np.arange(test_Y.size), test_Y] = 1
340     trainY = trainY.T
341     testY = testY.T
342
```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

KeyboardInterrupt
PS C:\Users\tanish A. Mishra\Documents\College Stuff\WPI\CS541\HW4\New folder> & "C:/Users/Tanish A. Mishra/AppData/Local/Programs/Python/Python310/python.exe" "c:/Users/Tanish A. Mishra/Documents/College Stuff/WPI/CS541/HW4/New folder/homework4_tamishra_anisal.py"
Gradient Check:
2.4632077529957962e-06

Hidden Layers: 3 Neurons in each layer: 64 Batch_size= 16
Learning Rate: 0.01 Alpha: 1e-06 Epochs: 70
Epoch no. 0
Loss: 0.6439679579053116 Accuracy: 76.00833333333334
Epoch no. 1
Loss: 0.5092111516426224 Accuracy: 81.96666666666667
Epoch no. 2

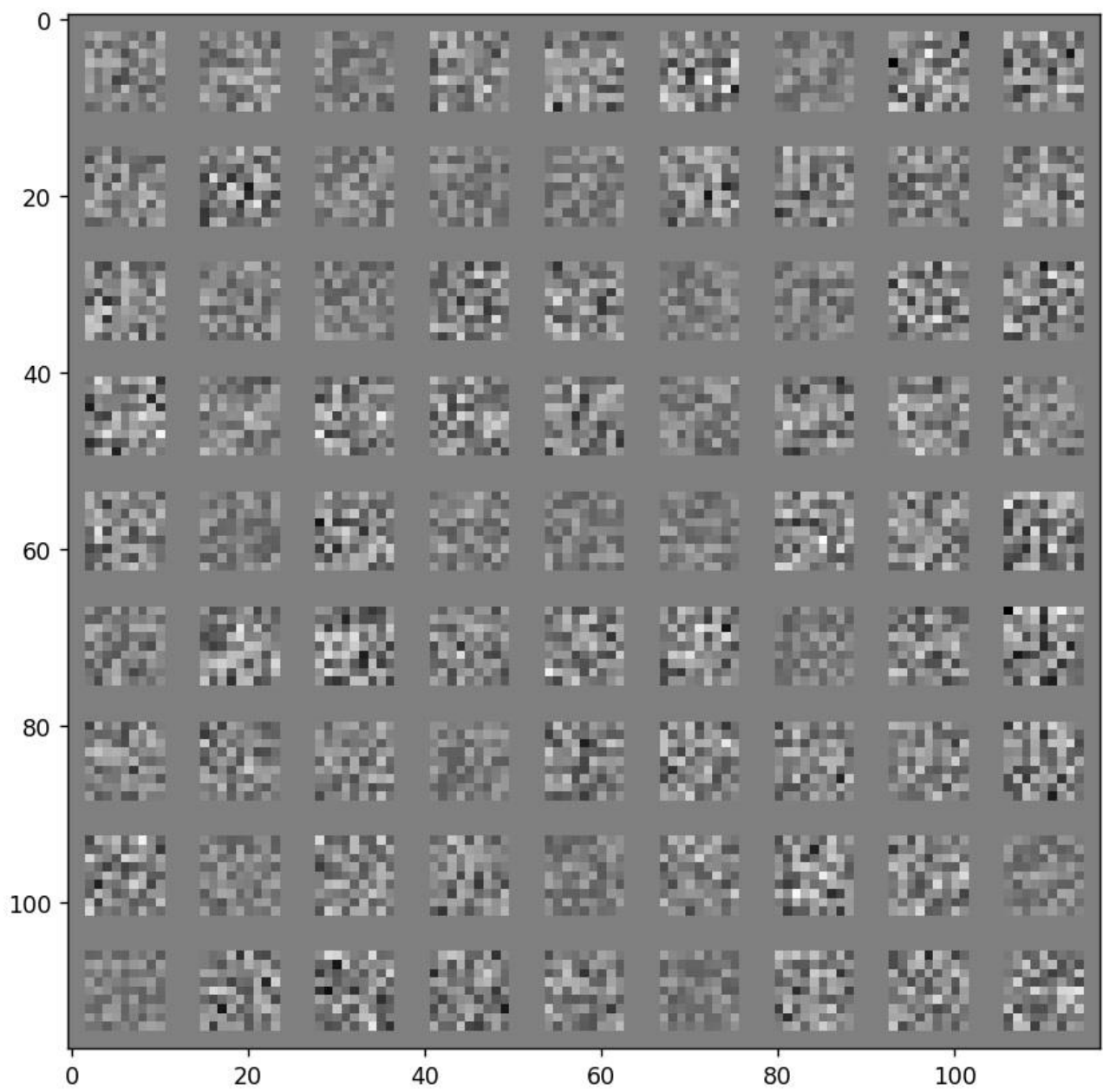
Gradient Check

```
homework4_tamishra_anisal.py > findBestHyperparameters
258 best_epochs = besth[0] # best number of epochs
259 best_batch_size = besth[1] # best number of batch_size
260 best_hidden_num = besth[2] # best number of hidden neurons
261 best_h_layers = besth[3] # best number of hidden layers
262 best_learningrate = besth[4] # best learning rate
263 best_alpha = besth[5] # best alpha
264
265 weightsAndBiases = initWeightsAndBiases(best_hidden_num, best_h_layers)
266 weightsAndBiases=copy.deepcopy(bestw)
267 loss, h_h, h_z, yhat = forward_prop(testX, testY, weightsAndBiases, best_hidden_num, best_h_layers)
268
269 print("\nThe Best HyperParameters: \nHidden Layers:", best_h_layers, "\nHidden Layer Neurons: ", best_hidden_num,
270       "\nEpochs: ", best_epochs, "\nBatch size: ", best_batch_size)
271 print("Learning rate: ", best_learningrate, "\nAlpha: ", best_alpha)
```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Loss: 0.612957024217922 Accuracy: 77.63333333333334
Epoch no. 1
Loss: 0.49761346621199776 Accuracy: 82.1
Epoch no. 2
Loss: 0.45303496480016675 Accuracy: 83.94166666666666
Epoch no. 3
Loss: 0.4258982726267441 Accuracy: 85.03333333333333
Epoch no. 4
Loss: 0.4067922182723614 Accuracy: 85.6
Epoch no. 5
Loss: 0.39024930229295923 Accuracy: 86.24166666666666
Epoch no. 6
Loss: 0.37759410068489463 Accuracy: 86.675
Epoch no. 7
Loss: 0.368900839349503 Accuracy: 86.83333333333333
Epoch no. 8
Loss: 0.363024347769612 Accuracy: 87.025
Epoch no. 9
Loss: 0.35578971017143163 Accuracy: 87.25
Epoch no. 10
Loss: 0.35233104370306584 Accuracy: 87.375
Epoch no. 11
Loss: 0.3511380542168628 Accuracy: 87.30833333333334
Epoch no. 12
Loss: 0.3458163661724052 Accuracy: 87.54166666666667
Epoch no. 13
Loss: 0.3448192707985614 Accuracy: 87.75833333333334
Epoch no. 14

Iterations



Visualisation of $W^{(1)}$ with 81 neurons in each hidden layer