Moshe Binieli - 311800668 - Task 3

So let's start with how I evaluated my neural network, I loaded the train_data and test_inputs, then I split the test_data into 80% of the original size and created validation data which is 20% from the train_data.

To summary it up, if I loaded 1000 rows, I used 800 rows for training purposes and 200 rows for validation purposes.

Attempt number 1:

Neural Network architecture: [784, 256, 10]

Hyper-Parameters: (Learning Rate: 1.0, Epochs: 50, Batch Size: 8)

So I started with neural network architecture which contains the input layer "x" which is 784 neurons, one hidden layer in size of 256 neurons and the output layer of size 10, I used hyperparameters of learning rate: 1.0, epochs: 50 and batch size: 10.

Well, this architecture gives us bad predictions since the learning rate is pretty high, every time we overshoot the local\global minimum which we need to achieve using gradient descent algorithm, I stopped the training at epoch 9 since the correct images that the neural network predicted is low, so this architecture is bad.

Result: Bad.

```
Epoch 0: Ratio, 1613/11000,  Percentage, 14.663636363636364
Epoch 1: Ratio, 1432/11000,  Percentage, 13.018181818181818
Epoch 2: Ratio, 1475/11000,  Percentage, 13.40909090909091
Epoch 3: Ratio, 1645/11000,  Percentage, 14.954545454545453
Epoch 4: Ratio, 1655/11000,  Percentage, 15.045454545454545
Epoch 5: Ratio, 1643/11000,  Percentage, 14.936363636363637
Epoch 6: Ratio, 1583/11000,  Percentage, 14.39090909090909
Epoch 7: Ratio, 1684/11000,  Percentage, 15.309090909090909
Epoch 8: Ratio, 1692/11000,  Percentage, 15.381818181818183
Epoch 9: Ratio, 1805/11000,  Percentage, 16.40909090909091
```

Attempt number 2:

Neural Network architecture: [784, 256, 10]

Hyper-Parameters: (Learning Rate: 0.01, Epochs: 50, Batch Size: 8)

So I understood that the learning rate must be lower, at this architecture I use the same hidden layers but what I changed is the learning rate to 0.01, and as you can see the predictions are pretty good, I reached 86% percent using this hidden layer architecture and hyperparameters.

Result: Good.

```
Epoch 0: Ratio, 8760/11000,  Percentage, 79.63636363636364
Epoch 1: Ratio, 8878/11000,  Percentage, 80.7090909090909
Epoch 2: Ratio, 9035/11000,  Percentage, 82.13636363636364
Epoch 3: Ratio, 9099/11000,  Percentage, 82.71818181818182
Epoch 4: Ratio, 9178/11000,  Percentage, 83.43636363636364
Epoch 5: Ratio, 9148/11000,  Percentage, 83.16363636363636
Epoch 6: Ratio, 9260/11000,  Percentage, 84.18181818181819
Epoch 7: Ratio, 9294/11000,  Percentage, 84.4909090909091
Epoch 8: Ratio, 9235/11000,  Percentage, 83.95454545454545
Epoch 9: Ratio, 9293/11000,  Percentage, 84.48181818181818
Epoch 10: Ratio, 9301/11000,  Percentage, 84.55454545454546
Epoch 11: Ratio, 9263/11000,  Percentage, 84.20909090909092
Epoch 12: Ratio, 9303/11000,  Percentage, 84.57272727272728
Epoch 13: Ratio, 9343/11000,  Percentage, 84.93636363636364
Epoch 14: Ratio, 9371/11000,  Percentage, 85.19090909090909
Epoch 15: Ratio, 9420/11000,  Percentage, 85.63636363636363
Epoch 16: Ratio, 9383/11000,  Percentage, 85.3
Epoch 17: Ratio, 9285/11000,  Percentage, 84.4090909090909
Epoch 18: Ratio, 9378/11000,  Percentage, 85.25454545454545
Epoch 19: Ratio, 9412/11000,  Percentage, 85.56363636363636
Epoch 20: Ratio, 9443/11000,  Percentage, 85.84545454545454
Epoch 21: Ratio, 9414/11000,  Percentage, 85.58181818181818
Epoch 22: Ratio, 9454/11000,  Percentage, 85.94545454545455
Epoch 23: Ratio, 9412/11000,  Percentage, 85.56363636363636
Epoch 24: Ratio, 9480/11000,  Percentage, 86.18181818181819
Epoch 25: Ratio, 9445/11000,  Percentage, 85.86363636363636
Epoch 26: Ratio, 9446/11000,  Percentage, 85.87272727272726
Epoch 27: Ratio, 9487/11000,  Percentage, 86.24545454545455
Epoch 28: Ratio, 9462/11000,  Percentage, 86.01818181818182
Epoch 29: Ratio, 9468/11000,  Percentage, 86.07272727272726
Epoch 30: Ratio, 9494/11000,  Percentage, 86.30909090909091
Epoch 31: Ratio, 9474/11000,  Percentage, 86.12727272727273
------------------------------------------------------------
```

** Moment of improvement of the architecture **

At this point, I realized what you said at the lesson, it's better to have more hidden layers and small ones instead of one huge hidden layer, since it can learn more patterns and etc, so instead of using 1 hidden layer which contains 256 neurons, I created 2 hidden layers which each contains 128 neurons.

Attempt number 3:

Neural Network architecture: [784, 128, 128, 10]

Hyper-Parameters: (Learning Rate: 1.0, Epochs: 50, Batch Size: 8)

So after I changed the hidden layers architecture, I wanted to see how the neural network behaves on kind of the same hyperparameters which are learning rate 1.0 and batch size 8, as you can see the learning rate ruins it completely, so this attempt wasn't good either.

Result: Bad.

```
Epoch 0: Ratio, 1139/11000,  Percentage, 10.354545454545455
Epoch 1: Ratio, 1070/11000,  Percentage, 9.727272727272727
Epoch 2: Ratio, 1070/11000,  Percentage, 9.727272727272727
Epoch 3: Ratio, 1107/11000,  Percentage, 10.063636363636363
Epoch 4: Ratio, 1139/11000,  Percentage, 10.354545454545455
Epoch 5: Ratio, 1085/11000,  Percentage, 9.863636363636363
Epoch 6: Ratio, 1085/11000,  Percentage, 9.863636363636363
Epoch 7: Ratio, 1179/11000,  Percentage, 10.718181818181819
Epoch 8: Ratio, 1070/11000,  Percentage, 9.727272727272727
Epoch 9: Ratio, 1070/11000,  Percentage, 9.727272727272727
Epoch 10: Ratio, 1078/11000,  Percentage, 9.8
Epoch 11: Ratio, 1078/11000,  Percentage, 9.8
Epoch 12: Ratio, 1107/11000,  Percentage, 10.063636363636363
Epoch 13: Ratio, 1088/11000,  Percentage, 9.89090909090909
Epoch 14: Ratio, 1088/11000,  Percentage, 9.89090909090909
Epoch 15: Ratio, 1107/11000,  Percentage, 10.063636363636363
Epoch 16: Ratio, 1098/11000,  Percentage, 9.981818181818182
Epoch 17: Ratio, 1088/11000,  Percentage, 9.89090909090909
```

Attempt number 4:

Neural Network architecture: [784, 128, 128, 10]

Hyper-Parameters: (Learning Rate: 0.001, Epochs: 50, Batch Size: 8)

So at this architecture attempt, I took 2 hidden layers of size 128 and learning rate 0.001, and batch size 8, This architecture is also good but a works not perfectly good, I reached to 81% of correctness but it can be even better if I would give it more time to train since it's SGD architecture.

Results: Good, but we could do better with more epochs.

```
Epoch 0: Ratio, 7925/11000,  Percentage, 72.04545454545455
Epoch 1: Ratio, 7955/11000,  Percentage, 72.31818181818181
Epoch 2: Ratio, 8207/11000,  Percentage, 74.60909090909091
Epoch 3: Ratio, 8328/11000,  Percentage, 75.7090909090909
Epoch 4: Ratio, 8490/11000,  Percentage, 77.18181818181819
Epoch 5: Ratio, 8482/11000,  Percentage, 77.10909090909091
Epoch 6: Ratio, 8484/11000,  Percentage, 77.12727272727273
Epoch 7: Ratio, 8582/11000,  Percentage, 78.01818181818182
Epoch 8: Ratio, 8612/11000,  Percentage, 78.29090909090908
Epoch 9: Ratio, 8654/11000,  Percentage, 78.67272727272727
Epoch 10: Ratio, 8610/11000,  Percentage, 78.27272727272727
Epoch 11: Ratio, 8630/11000,  Percentage, 78.45454545454545
Epoch 12: Ratio, 8684/11000,  Percentage, 78.94545454545454
Epoch 13: Ratio, 8564/11000,  Percentage, 77.85454545454546
Epoch 14: Ratio, 8669/11000,  Percentage, 78.80909090909091
Epoch 15: Ratio, 8674/11000,  Percentage, 78.85454545454546
Epoch 16: Ratio, 8668/11000,  Percentage, 78.8
Epoch 17: Ratio, 8629/11000,  Percentage, 78.44545454545454
Epoch 18: Ratio, 8683/11000,  Percentage, 78.93636363636364
Epoch 19: Ratio, 8788/11000,  Percentage, 79.89090909090909
Epoch 20: Ratio, 8720/11000,  Percentage, 79.27272727272727
Epoch 21: Ratio, 8784/11000,  Percentage, 79.85454545454546
Epoch 22: Ratio, 8772/11000,  Percentage, 79.74545454545454
Epoch 23: Ratio, 8628/11000,  Percentage, 78.43636363636364
Epoch 24: Ratio, 8583/11000,  Percentage, 78.02727272727272
Epoch 25: Ratio, 8783/11000,  Percentage, 79.84545454545454
Epoch 26: Ratio, 8782/11000,  Percentage, 79.83636363636364
Epoch 44: Ratio, 8926/11000,  Percentage, 81.14545454545454
```

Attempt number 5:

Neural Network architecture: [784, 128, 128, 10]

Hyper-Parameters: (Learning Rate: 0.001, Epochs: 50, Batch Size: 1)

So this time I was thinking to mess around with the mathematics a little bit, as we learned, gradient descent and stochastic gradient descent archives the same goal, each one of them does it a little differently, I decided to try this architecture with a batch size of 1, to see how it works, and the results are pretty good.

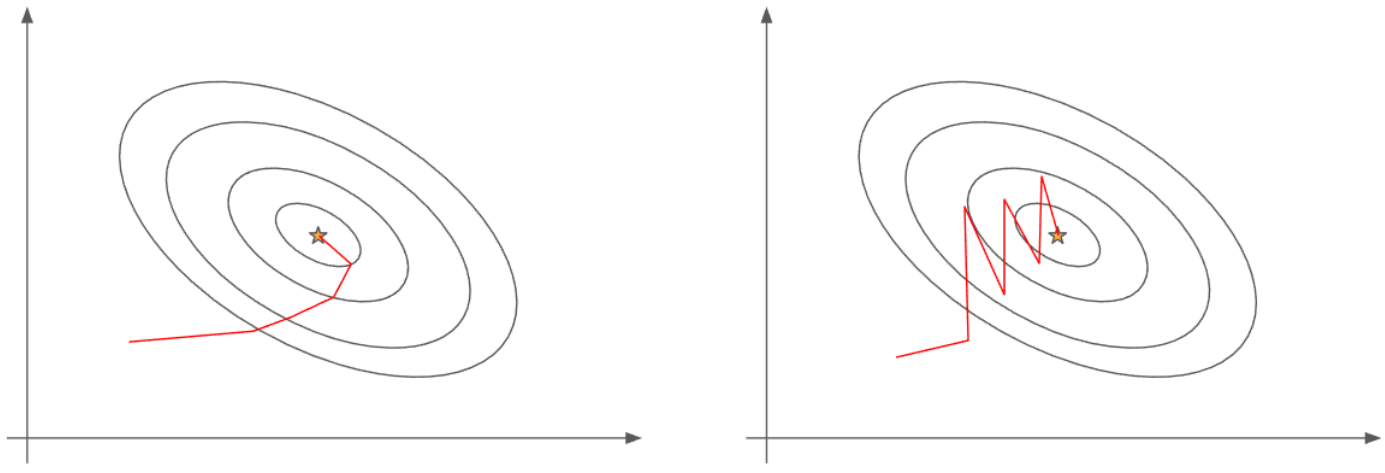
Result: Good.

```
Epoch 0: Ratio, 8356/11000,  Percentage, 75.96363636363637
Epoch 1: Ratio, 8815/11000,  Percentage, 80.13636363636364
Epoch 2: Ratio, 8755/11000,  Percentage, 79.5909090909091
Epoch 3: Ratio, 8876/11000,  Percentage, 80.69090909090909
Epoch 4: Ratio, 8897/11000,  Percentage, 80.88181818181818
Epoch 5: Ratio, 8968/11000,  Percentage, 81.52727272727273
Epoch 6: Ratio, 9068/11000,  Percentage, 82.43636363636364
Epoch 7: Ratio, 9135/11000,  Percentage, 83.04545454545455
Epoch 8: Ratio, 9074/11000,  Percentage, 82.4909090909091
Epoch 9: Ratio, 9196/11000,  Percentage, 83.6
Epoch 10: Ratio, 9268/11000,  Percentage, 84.25454545454546
Epoch 11: Ratio, 9221/11000,  Percentage, 83.82727272727273
Epoch 12: Ratio, 9256/11000,  Percentage, 84.14545454545454
Epoch 13: Ratio, 9247/11000,  Percentage, 84.06363636363636
Epoch 14: Ratio, 9304/11000,  Percentage, 84.58181818181818
Epoch 15: Ratio, 9278/11000,  Percentage, 84.34545454545454
Epoch 16: Ratio, 9314/11000,  Percentage, 84.67272727272727
Epoch 17: Ratio, 9343/11000,  Percentage, 84.93636363636364
Epoch 18: Ratio, 9390/11000,  Percentage, 85.36363636363636
Epoch 19: Ratio, 9334/11000,  Percentage, 84.85454545454544
Epoch 20: Ratio, 9355/11000,  Percentage, 85.04545454545455
Epoch 21: Ratio, 9390/11000,  Percentage, 85.36363636363636
Epoch 22: Ratio, 9380/11000,  Percentage, 85.27272727272728
Epoch 23: Ratio, 9386/11000,  Percentage, 85.32727272727273
Epoch 24: Ratio, 9390/11000,  Percentage, 85.36363636363636
Epoch 25: Ratio, 9413/11000,  Percentage, 85.57272727272728
```

And at epoch 35 I reached to 86% correctness.

```
Epoch 35: Ratio, 9465/11000,  Percentage, 86.04545454545455
```

So what is the difference between attempt number 4 and attempt number 5?



Well, all we need it to visualize that differences, as you can see gradient descent updates the weights and biases every time, and stochastic gradient descent updates weights and biases after working on X amount of data, SGD is faster than GD, and I could feel the speed when I run the networks once on GD and once at SGD.

Attempt number 6:

Neural Network architecture: [784, 258, 128, 64, 32, 10]

Hyper-Parameters: (Learning Rate: 0.01, Epochs: 50, Batch Size: 4)

This time I said I will make big hidden layers and see what's going on, well I must say this time everything just worked really really bad, at epoch 3 I decided to stop running the ANN because it was just bad.

Result: Bad.

```
Epoch 0: Ratio, 1085/11000,  Percentage, 9.863636363636363
Epoch 1: Ratio, 1085/11000,  Percentage, 9.863636363636363
Epoch 2: Ratio, 1085/11000,  Percentage, 9.863636363636363
Epoch 3: Ratio, 1085/11000,  Percentage, 9.863636363636363
```

Conclusion:

Well, the learning rate plays a really strong role here to learn the neural network, there is a difference in performance from gradient descent to stochastic gradient descent, too many layers doesn't necessarily make the neural network model good, it actually ruins it as you can see in attempt number 6.

So ANN architecture of [input, 128, 128, output] works pretty well with a learning rate of 0.001, the batches play a role also but I talked about it at the differences between GD and SGD.