RASHEED HAMEED
CSC 578 Neural Networks and Deep Learning – Section 910
Homework 3

Claim 1:

Per the first claim wanted to see if the results from this code 578hw3-checktestset provide same results as the html document. The results are the same as portrayed below:

RESULTS:

```
[Epoch 0]  Training: MSE=0.26673129, CE=1.61690061, LL=0.45930834, Correct: 1/1, Acc: 1.00000000
           Test:     MSE=0.32440028, CE=1.85581862, LL=0.58712192, Correct: 0/1, Acc: 0.00000000
[[[1, 1.0, 0.26673128660052947, 1.6169006088912665, 0.4593083419698332]], [[0, 0.0, 0.3244002758397572, 1.855818621521176, 0.5871219174966811]]]

[Epoch 0]  Training: MSE=0.21078666, CE=1.38020720, LL=0.39958448, Correct: 1/1, Acc: 1.00000000
Epoch 0 complete
[[[1, 1.0, 0.2107866577006649, 1.3802072032795016, 0.39958447879106]], []]
```

HTML document results:

```
[Epoch 0] Training: MSE=0.26673129, CE=1.61690061, LL=0.45930834, Correct: 1/1, Acc: 1.00000000
          Test:     MSE=0.32440028, CE=1.85581862, LL=0.58712192, Correct: 0/1, Acc: 0.00000000
[[[1, 1.0, 0.26673128660052947, 1.6169006088912665, 0.4593083419698332]], [[0, 0.0, 0.3244002758397572, 1.855818621521176, 0.58
71219174966811]]]

[Epoch 0] Training: MSE=0.21078666, CE=1.38020720, LL=0.39958448, Correct: 1/1, Acc: 1.00000000
[[[1, 1.0, 0.2107866577006649, 1.3802072032795016, 0.39958447879106]], [[]]]
```

Claim 2a:

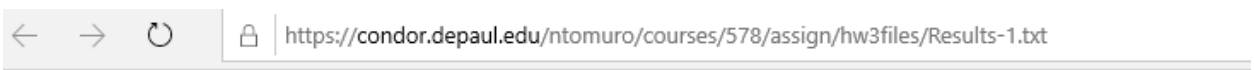
In observation for the second claim where we are looking if the results match the training and the test. The training and test set had different results. The MSE values, the Cross-Entropy values, log-likelihood value and the accuracy values were not matching as portrayed in the snippet below.

```
[Epoch 0]  Training: MSE=0.26673129, CE=1.61690061, LL=0.45930834, Correct: 1/1, Acc: 1.00000000
           Test:     MSE=0.32440028, CE=1.85581862, LL=0.58712192, Correct: 0/1, Acc: 0.00000000
[[[1, 1.0, 0.26673128660052947, 1.6169006088912665, 0.4593083419698332]], [[0, 0.0, 0.3244002758397572, 1.855818621521176,
0.5871219174966811]]]
```

Claim 2b:

Observing if the results matched after implementing the code and choosing choices for having 100 epochs, with min-batch size at 5 and eta at 0.5. We observe the output and the result file provided and found to be exactly matching. The choice of the parameters were set to only run for 100 epochs and the default set at 1.0.

RESULTS 1 BEGINNING OF FILE




```
[Epoch 0] Training: MSE=0.33989623, CE=1.93947802, LL=1.07270335, Correct: 50/150, Acc: 0.33333333
[Epoch 1] Training: MSE=0.35054136, CE=1.98910892, LL=1.13654524, Correct: 50/150, Acc: 0.33333333
[Epoch 2] Training: MSE=0.33898723, CE=1.93723493, LL=1.12030613, Correct: 50/150, Acc: 0.33333333
[Epoch 3] Training: MSE=0.30402667, CE=1.76598458, LL=0.98309968, Correct: 51/150, Acc: 0.34000000
[Epoch 4] Training: MSE=0.27946742, CE=1.65306466, LL=0.88038172, Correct: 100/150, Acc: 0.66666667
[Epoch 5] Training: MSE=0.25916154, CE=1.56196374, LL=0.80792403, Correct: 100/150, Acc: 0.66666667
[Epoch 6] Training: MSE=0.24333059, CE=1.48939578, LL=0.75466382, Correct: 100/150, Acc: 0.66666667
```

OUTPUT OF CODE BEGINNING OF RESULTS

```
[4]: net1.SGD(iris_data, 100, 5, 0.5)

[Epoch 0]  Training: MSE=0.33989623, CE=1.93947802, LL=1.07270335, Correct: 50/150, Acc: 0.33333333
[Epoch 1]  Training: MSE=0.35054136, CE=1.98910892, LL=1.13654524, Correct: 50/150, Acc: 0.33333333
[Epoch 2]  Training: MSE=0.33898723, CE=1.93723493, LL=1.12030613, Correct: 50/150, Acc: 0.33333333
[Epoch 3]  Training: MSE=0.30402667, CE=1.76598458, LL=0.98309968, Correct: 51/150, Acc: 0.34000000
[Epoch 4]  Training: MSE=0.27946742, CE=1.65306466, LL=0.88038172, Correct: 100/150, Acc: 0.66666667
[Epoch 5]  Training: MSE=0.25916154, CE=1.56196374, LL=0.80792403, Correct: 100/150, Acc: 0.66666667
[Epoch 6]  Training: MSE=0.24333059, CE=1.48939578, LL=0.75466382, Correct: 100/150, Acc: 0.66666667
```

RESULTS 1 END OF FILE

```
[Epoch 94] Training: MSE=0.07979103, CE=0.57712893, LL=0.26837303, Correct: 133/150, Acc: 0.88666667
[Epoch 95] Training: MSE=0.07831577, CE=0.56933537, LL=0.26463285, Correct: 133/150, Acc: 0.88666667
[Epoch 96] Training: MSE=0.07697374, CE=0.56219254, LL=0.26118019, Correct: 134/150, Acc: 0.89333333
[Epoch 97] Training: MSE=0.07573288, CE=0.55554221, LL=0.25795193, Correct: 135/150, Acc: 0.90000000
[Epoch 98] Training: MSE=0.07456498, CE=0.54924517, LL=0.25489053, Correct: 136/150, Acc: 0.90666667
[Epoch 99] Training: MSE=0.07344778, CE=0.54319159, LL=0.25194933, Correct: 136/150, Acc: 0.90666667
```

OUTPUT OF CODE END OF RESULTS

```
[Epoch 95]  Training: MSE=0.07831577, CE=0.56933537, LL=0.26463285, Correct: 133/150, Acc: 0.88666667
[Epoch 96]  Training: MSE=0.07697374, CE=0.56219254, LL=0.26118019, Correct: 134/150, Acc: 0.89333333
[Epoch 97]  Training: MSE=0.07573288, CE=0.55554221, LL=0.25795193, Correct: 135/150, Acc: 0.90000000
[Epoch 98]  Training: MSE=0.07456498, CE=0.54924517, LL=0.25489053, Correct: 136/150, Acc: 0.90666667
[Epoch 99]  Training: MSE=0.07344778, CE=0.54319159, LL=0.25194933, Correct: 136/150, Acc: 0.90666667
[4]: [[[50,
```

Claim 2c:

We attempt to verify if the code is working as needed by implementing the use of **"iris4-20-7-3.dat"** as json document where the configuration of the parameters are such; epochs = 100, mini-batch as 5, eta 0.5, which the stop accuracy set at 0.75.

RESULTS 2 BEGINNING OF FILE


```
[Epoch 0] Training: MSE=0.35101648, CE=1.98991328, LL=1.13575832, Correct: 50/150, Acc: 0.33333333
[Epoch 1] Training: MSE=0.35685102, CE=2.01618235, LL=1.14912048, Correct: 50/150, Acc: 0.33333333
[Epoch 2] Training: MSE=0.35639476, CE=2.01398864, LL=1.14813574, Correct: 50/150, Acc: 0.33333333
[Epoch 3] Training: MSE=0.35437837, CE=2.00490209, LL=1.14575037, Correct: 50/150, Acc: 0.33333333
[Epoch 4] Training: MSE=0.35223003, CE=1.99520139, LL=1.14349029, Correct: 50/150, Acc: 0.33333333
[Epoch 5] Training: MSE=0.35031508, CE=1.98646512, LL=1.14156680, Correct: 50/150, Acc: 0.33333333
[Epoch 6] Training: MSE=0.34871835, CE=1.97904087, LL=1.13981143, Correct: 50/150, Acc: 0.33333333
```

OUTPUT OF CODE BEGINNING OF RESULTS

```
[6]: net3 = network.load_network("iris4-20-7-3.dat")
     net3.SGD(iris_data, 100, 5, 0.5, None, 0.75)
```

```
[Epoch 0]  Training: MSE=0.35101648, CE=1.98991328, LL=1.13575832, Correct: 50/150, Acc: 0.33333333
[Epoch 1]  Training: MSE=0.35685102, CE=2.01618235, LL=1.14912048, Correct: 50/150, Acc: 0.33333333
[Epoch 2]  Training: MSE=0.35639476, CE=2.01398864, LL=1.14813574, Correct: 50/150, Acc: 0.33333333
[Epoch 3]  Training: MSE=0.35437837, CE=2.00490209, LL=1.14575037, Correct: 50/150, Acc: 0.33333333
[Epoch 4]  Training: MSE=0.35223003, CE=1.99520139, LL=1.14349029, Correct: 50/150, Acc: 0.33333333
[Epoch 5]  Training: MSE=0.35031508, CE=1.98646512, LL=1.14156680, Correct: 50/150, Acc: 0.33333333
[Epoch 6]  Training: MSE=0.34871835, CE=1.97904087, LL=1.13981143, Correct: 50/150, Acc: 0.33333333
[Epoch 7]  Training: MSE=0.34745549, CE=1.97297198, LL=1.13794379, Correct: 50/150, Acc: 0.33333333
```

RESULTS 2 END OF FILE

```
[Epoch 57] Training: MSE=0.17197155, CE=1.00920274, LL=0.49413599, Correct: 108/150, Acc: 0.72000000
[Epoch 58] Training: MSE=0.16919844, CE=0.99485113, LL=0.48755799, Correct: 109/150, Acc: 0.72666667
[Epoch 59] Training: MSE=0.16638703, CE=0.98050039, LL=0.48086780, Correct: 109/150, Acc: 0.72666667
[Epoch 60] Training: MSE=0.16355655, CE=0.96620930, LL=0.47414144, Correct: 109/150, Acc: 0.72666667
[Epoch 61] Training: MSE=0.16060140, CE=0.95142646, LL=0.46709139, Correct: 112/150, Acc: 0.74666667
[Epoch 62] Training: MSE=0.15757755, CE=0.93638462, LL=0.45986411, Correct: 113/150, Acc: 0.75333333
```

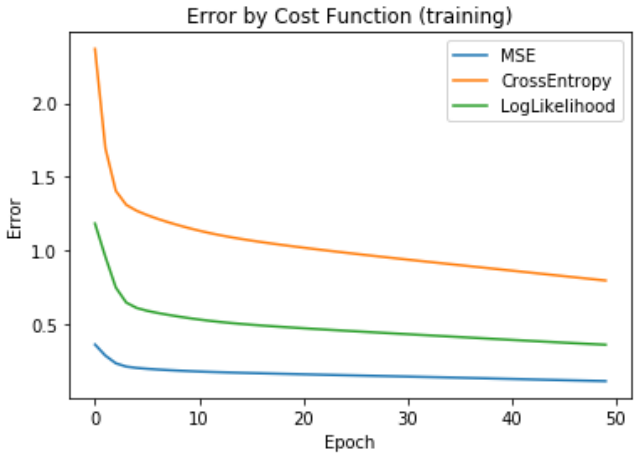
OUTPUT OF CODE END OF RESULTS

```
[Epoch 57]  Training: MSE=0.17197155, CE=1.00920274, LL=0.49413599, Correct: 108/150, Acc: 0.72000000
[Epoch 58]  Training: MSE=0.16919844, CE=0.99485113, LL=0.48755799, Correct: 109/150, Acc: 0.72666667
[Epoch 59]  Training: MSE=0.16638703, CE=0.98050039, LL=0.48086780, Correct: 109/150, Acc: 0.72666667
[Epoch 60]  Training: MSE=0.16355655, CE=0.96620930, LL=0.47414144, Correct: 109/150, Acc: 0.72666667
[Epoch 61]  Training: MSE=0.16060140, CE=0.95142646, LL=0.46709139, Correct: 112/150, Acc: 0.74666667
[Epoch 62]  Training: MSE=0.15757755, CE=0.93638462, LL=0.45986411, Correct: 113/150, Acc: 0.75333333
Epoch 62 complete
```

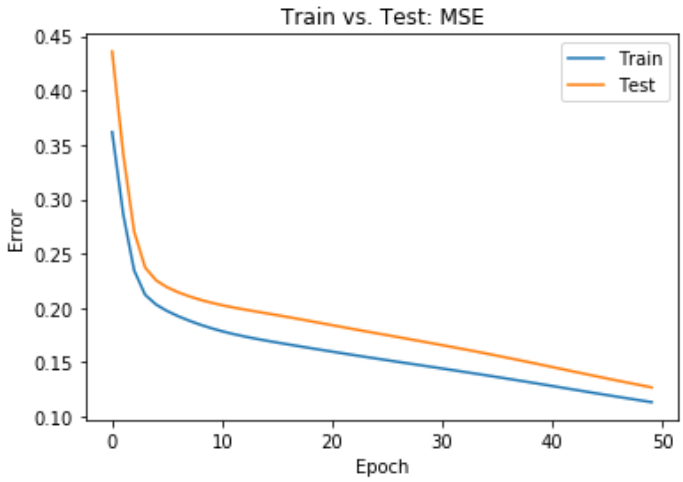
Observing all the snippets above all the results are matching per output provided. I used IBM Cognitive Colab and the answer should be very similar.

Visualization observation 3:

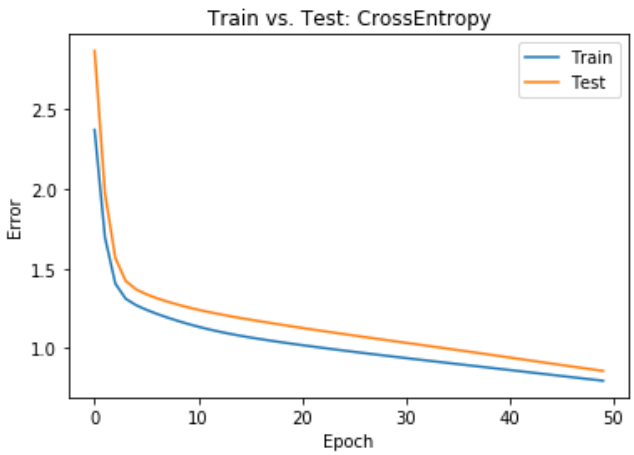
For the Error cost function plotting portrayed below for just the training set. We see MSE having the lowest values for errors, then log likelihood and then the highest for Cross-Entropy functions. It shows for this data set at higher epochs we are reached better accuracy and lower error values. The best performance when the gradient errors are computed with respect to MSE. Overall the weights update to better values.



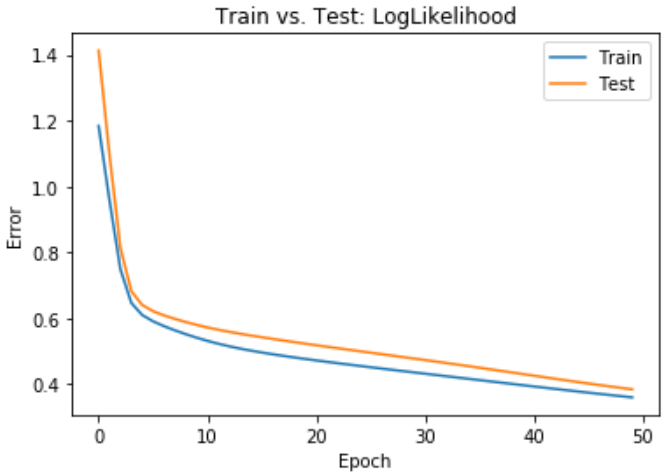
The training set versus the test set for MSE have very similar values and we can say overfitting is not a concern here. We do notice though the error values are little higher for training set between 1-50 Epochs.



The training set versus the test set for Cross-Entropy have very similar values and we can say overfitting is not much of a concern here. We do notice though the error values are little higher for training set between 1-50 Epochs.



The training set versus the test set for log likelihood have very similar values and we can say overfitting is not much of a concern here. We do notice though the error values are little higher for training set. The values between 1-5 epoch are very near than they grow slightly apart between 4 - 50



My reaction and reflection 4:

I did find this exercise challenging but it was a good learning experience as it took a long time to figure out array manipulations. I was stuck on unnecessary errors like missing a parenthesis or incorrect array manipulation of arrays. I did spend a lot of time on this exercise.