Multi Layer Perceptrons

Introduction

CIFAR-10 Dataset

A multi layer perceptron network is similar to a fully connected neural network, in the sense that they contain a series of linear layers connected together. However, the key difference is that in a MLP, there is an activation function after each layer, which allows the network to learn relationships that are beyond linear. This results in better performance than fully connected neural networks in certain tasks, but for the CIFAR-10 dataset, in this homework, it is not the optimal approach. A ResNet model, which was used in the last class performed a lot better (reaching 85% validation accuracy), while this model only reaches approximately 55%.

Precision: 0.5342946033740245

Recall: 0.5347

F1 Score: 0.5329206299296686

```
[[691  33  48  21  24  11   6  20  91  55]
 [ 40 641  20  22   9   6  15  17  61 169]
 [113  20 393  95 137  70  77  57  17  21]
 [ 49  20  83 374  76 173  94  52  35  44]
 [ 67  11 138  81 451  51  90  69  25  17]
 [ 36  12  66 248  82 379  44  69  38  26]
 [ 15  28  91  98  98  47 548  24  17  34]
 [ 48   9  41  75  89  82  12 594  15  35]
 [147  68  20  21  16  10   5   9 641  63]
 [ 50 140  13  36  10   9  12  42  53 635]]
```

If we double the number of epochs, the result doesn't really change.

Precision: 0.5387330793790339

Recall: 0.5370999999999999

F1 Score: 0.5365017983133075

Confusion Matrix:

[[651 42 47 32 29 15 21 22 98 43]
 [50 632 22 26 14 19   6 29 53 149]
 [87 15 372 86 160 103 71 61 22 23]
 [28 17 82 319 80 281 79 51 31 32]
 [38 13 101 73 478 75 94 91 14 23]
 [20   5 79 149 69 541 35 61 22 19]
 [11 27 75 116 95 70 562 14 16 14]
 [36   7 43 54 77 109 12 607 10 45]
 [124 64 18 22 23 25   8 10 638 68]
 [64 155 17 35 17 37 13 41 50 571]]

All the results are nearly identical, and it seems that this model, in its current state is not able to learn more information from the data.

Widening and deepening the layers (doubling the number of neurons and increasing the amount of layers smoothed out the training graph a little bit, but didn't resolve the problem. The confusion matrix and f1 score are virtually identical.
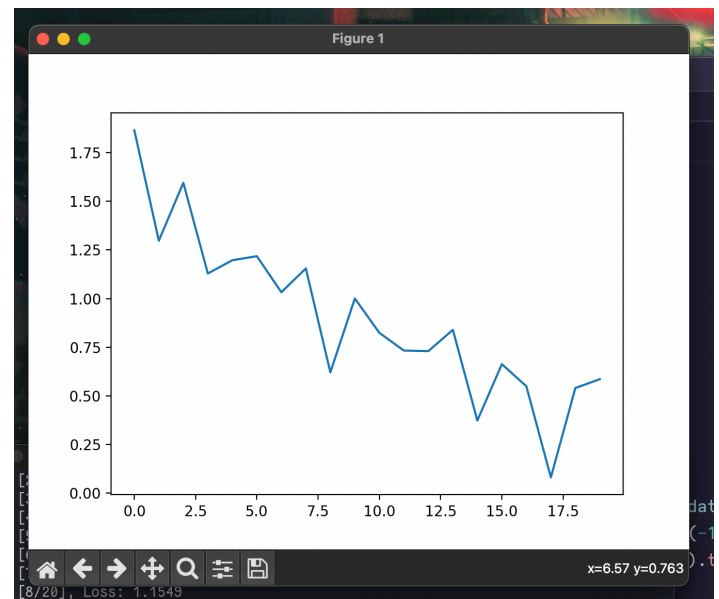
Precision: 0.5556081099812743

Recall: 0.5448000000000001

F1 Score: 0.5452434976289092

Confusion Matrix:

```
[[697  31  53  36  36   7  20  19  55  46]
 [ 54 685   8  32  14  15  14  25  36 117]
 [ 93  21 379 130 118  93  88  45   9  24]
 [ 29  14  67 488  49 181  72  60  13  27]
 [ 62  13 103  93 423  77 116  86  11  16]
 [ 25   9  59 290  46 427  45  71  11  17]
 [ 11  15  54 144  71  46 609  25   6  19]
 [ 47  11  31  83  66  87  14 630   2  29]
 [151  94  16  49  32  18   4  16 530  90]
 [ 87 163  13  35  14  20  16  52  20 580]]
```

Housing Dataset

This is a more suited task to a multilayer perceptron, but I ran out of time and wasn't able to actually do any training.

GitHub URL: https://github.com/Anu78/machine-learning-hw-uncc