

Shengkai Xu
ECG 4106 - RealTime AI
03/04/2022
Homework2

Github: <https://github.com/ric3b0wl/RealTimeAI/tree/master/hw2>

Problem 1

Build a fully connected neural network for the housing dataset you did in previous homework. For training and validation use 80% (training) and 20% (validation) split. For this part, only use one hidden layer with 8 nodes. Train your network for 200 epochs. Report your training time, training loss, and evaluation accuracy after 200 epochs. Analyze your results in your report. Make sure to submit your code by providing the GitHub URL of your course repository for this course.

With only a single hidden layer with 8 nodes, the training for the model has very low accuracy. The model did not perform that well under a low number of the hidden layer, the model was not able to learn the feature of the datasets.

```
Sequential(
  (0): Linear(in_features=5, out_features=6, bias=True)
  (1): Tanh()
  (2): Linear(in_features=6, out_features=4, bias=True)
  (3): Tanh()
  (4): Linear(in_features=4, out_features=2, bias=True)
  (5): Tanh()
  (6): Linear(in_features=2, out_features=1, bias=True)
)
```

```
Epoch 0, Training Loss is 0.7305, Validation Loss is 0.7499
Epoch 10, Training Loss is 0.6749, Validation Loss is 0.6937
Epoch 20, Training Loss is 0.6237, Validation Loss is 0.6417
Epoch 30, Training Loss is 0.5764, Validation Loss is 0.5938
Epoch 40, Training Loss is 0.5328, Validation Loss is 0.5495
Epoch 50, Training Loss is 0.4925, Validation Loss is 0.5086
Epoch 60, Training Loss is 0.4554, Validation Loss is 0.4709
Epoch 70, Training Loss is 0.4211, Validation Loss is 0.4360
Epoch 80, Training Loss is 0.3894, Validation Loss is 0.4038
Epoch 90, Training Loss is 0.3602, Validation Loss is 0.3740
Epoch 100, Training Loss is 0.3332, Validation Loss is 0.3465
Epoch 110, Training Loss is 0.3084, Validation Loss is 0.3211
Epoch 120, Training Loss is 0.2854, Validation Loss is 0.2976
Epoch 130, Training Loss is 0.2642, Validation Loss is 0.2759
Epoch 140, Training Loss is 0.2446, Validation Loss is 0.2559
Epoch 150, Training Loss is 0.2266, Validation Loss is 0.2373
Epoch 160, Training Loss is 0.2099, Validation Loss is 0.2203
Epoch 170, Training Loss is 0.1946, Validation Loss is 0.2045
Epoch 180, Training Loss is 0.1804, Validation Loss is 0.1899
Epoch 190, Training Loss is 0.1673, Validation Loss is 0.1764
Epoch 200, Training Loss is 0.1553, Validation Loss is 0.1640
```

With multiple-layer available, model performance is much better than the model before.

```
0.weight torch.Size([6, 5])
0.bias torch.Size([6])
2.weight torch.Size([4, 6])
2.bias torch.Size([4])
4.weight torch.Size([2, 4])
4.bias torch.Size([2])
6.weight torch.Size([1, 2])
6.bias torch.Size([1])
Parameter containing:
tensor([0.0305], requires_grad=True)
```

```
Epoch 0, Training Loss is 0.0465, Validation Loss is 0.0484
Epoch 10, Training Loss is 0.0436, Validation Loss is 0.0454
Epoch 20, Training Loss is 0.0410, Validation Loss is 0.0427
Epoch 30, Training Loss is 0.0387, Validation Loss is 0.0403
Epoch 40, Training Loss is 0.0366, Validation Loss is 0.0381
Epoch 50, Training Loss is 0.0347, Validation Loss is 0.0361
Epoch 60, Training Loss is 0.0331, Validation Loss is 0.0343
Epoch 70, Training Loss is 0.0315, Validation Loss is 0.0327
Epoch 80, Training Loss is 0.0302, Validation Loss is 0.0313
Epoch 90, Training Loss is 0.0290, Validation Loss is 0.0300
Epoch 100, Training Loss is 0.0279, Validation Loss is 0.0288
Epoch 110, Training Loss is 0.0269, Validation Loss is 0.0277
Epoch 120, Training Loss is 0.0260, Validation Loss is 0.0268
Epoch 130, Training Loss is 0.0252, Validation Loss is 0.0259
Epoch 140, Training Loss is 0.0245, Validation Loss is 0.0251
Epoch 150, Training Loss is 0.0238, Validation Loss is 0.0244
Epoch 160, Training Loss is 0.0232, Validation Loss is 0.0238
Epoch 170, Training Loss is 0.0227, Validation Loss is 0.0232
Epoch 180, Training Loss is 0.0223, Validation Loss is 0.0227
Epoch 190, Training Loss is 0.0218, Validation Loss is 0.0222
Epoch 200, Training Loss is 0.0215, Validation Loss is 0.0218
```

Problem 2

Create a fully connected Neural Network for all 10 classes in CIFAR-10 with only one hidden layer with the size of 512. Train your network for 200 epochs. Report your training time, training loss and evaluation accuracy after 200 epochs. Analyze your results in your report. Make sure to submit your code by providing the GitHub URL of your course repository for this course. (25pt)

In this problem, we download the database of CIFAR10.

With only one hidden layer, the result of the performance is expected to be very low. But it was not as bad as the result we are getting from problem 1, I think it is due to the quantity of dataset in CIFAR10.

```
Validation Accuracy: 0.804500
```

```
Sequential(  
  (0): Linear(in_features=3072, out_features=512, bias=True)  
  (1): Tanh()  
  (2): Linear(in_features=512, out_features=2, bias=True)  
  (3): LogSoftmax(dim=1)  
)
```

```
Epoch: 0, Training Loss: 0.285458  
Epoch: 10, Training Loss: 0.122038  
Epoch: 20, Training Loss: 0.052608  
Epoch: 30, Training Loss: 0.026766  
Epoch: 40, Training Loss: 0.015452  
Epoch: 50, Training Loss: 0.009215  
Epoch: 60, Training Loss: 0.006043  
Epoch: 70, Training Loss: 0.004266  
Epoch: 80, Training Loss: 0.003212  
Epoch: 90, Training Loss: 0.002559  
Epoch: 100, Training Loss: 0.002127  
Epoch: 110, Training Loss: 0.001824  
Epoch: 120, Training Loss: 0.001598  
Epoch: 130, Training Loss: 0.001424  
Epoch: 140, Training Loss: 0.001285  
Epoch: 150, Training Loss: 0.001170  
Epoch: 160, Training Loss: 0.001073  
Epoch: 170, Training Loss: 0.000990  
Epoch: 180, Training Loss: 0.000919  
Epoch: 190, Training Loss: 0.000856  
Epoch: 200, Training Loss: 0.000800
```

With multiple hidden layers, the linear model is able to perform much better than the previous model.

Validation Accuracy: 0.813500

```
Sequential(  
  (0): Linear(in_features=3072, out_features=1024, bias=True)  
  (1): Tanh()  
  (2): Linear(in_features=1024, out_features=512, bias=True)  
  (3): Tanh()  
  (4): Linear(in_features=512, out_features=128, bias=True)  
  (5): Tanh()  
  (6): Linear(in_features=128, out_features=2, bias=True)  
)
```

```
Epoch: 0, Training Loss: 0.787299  
Epoch: 10, Training Loss: 0.527168  
Epoch: 20, Training Loss: 0.232282  
Epoch: 30, Training Loss: 0.052824  
Epoch: 40, Training Loss: 0.013029  
Epoch: 50, Training Loss: 0.002018  
Epoch: 60, Training Loss: 0.001679  
Epoch: 70, Training Loss: 0.000408  
Epoch: 80, Training Loss: 0.000220  
Epoch: 90, Training Loss: 0.000148  
Epoch: 100, Training Loss: 0.000105  
Epoch: 110, Training Loss: 0.000078  
Epoch: 120, Training Loss: 0.000060  
Epoch: 130, Training Loss: 0.000048  
Epoch: 140, Training Loss: 0.000040  
Epoch: 150, Training Loss: 0.000034  
Epoch: 160, Training Loss: 0.000030  
Epoch: 170, Training Loss: 0.000027  
Epoch: 180, Training Loss: 0.000024  
Epoch: 190, Training Loss: 0.000022  
Epoch: 200, Training Loss: 0.000020
```