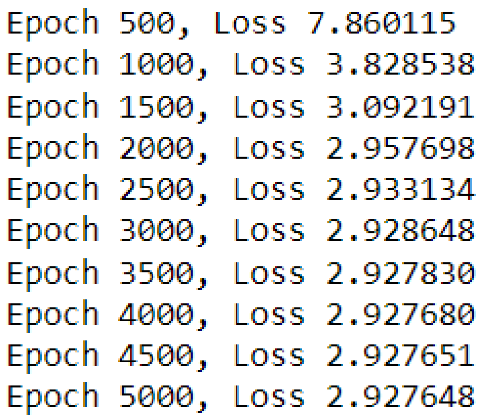
ECGR 4105 – HW # 5

Nahush D. Tambe - 801060297

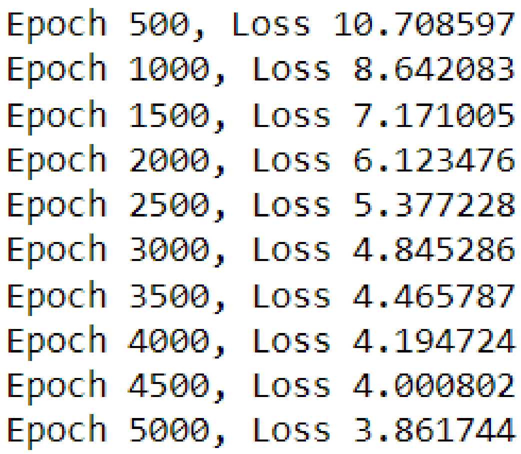
https://github.com/Ntambe25

Problem 1:



```
Epoch 500, Loss 7.860115
Epoch 1000, Loss 3.828538
Epoch 1500, Loss 3.092191
Epoch 2000, Loss 2.957698
Epoch 2500, Loss 2.933134
Epoch 3000, Loss 2.928648
Epoch 3500, Loss 2.927830
Epoch 4000, Loss 2.927680
Epoch 4500, Loss 2.927651
Epoch 5000, Loss 2.927648
```

Figure 1: Loss for every 500 epochs per Training for a Linear Model



```
Epoch 500, Loss 10.708597
Epoch 1000, Loss 8.642083
Epoch 1500, Loss 7.171005
Epoch 2000, Loss 6.123476
Epoch 2500, Loss 5.377228
Epoch 3000, Loss 4.845286
Epoch 3500, Loss 4.465787
Epoch 4000, Loss 4.194724
Epoch 4500, Loss 4.000802
Epoch 5000, Loss 3.861744
```

Figure 2: Loss for every 500 epochs per Training for a Non-Linear Model

As required for problem # 2, the loss for every 500 epochs for both Linear and Non-Linear models is shown in Figure 1 and Figure 2, above. It can be seen that the loss for the Linear model with a learning rate of 1e-2 gave the best results. Different values were explored like the loss with a learning rate of 1e-1 was coming out to be NaN for all of the training. With a value of 1e-3, the loss was coming out to be around 25, which is way higher that the general loss with a learning rate value of 1e-2.

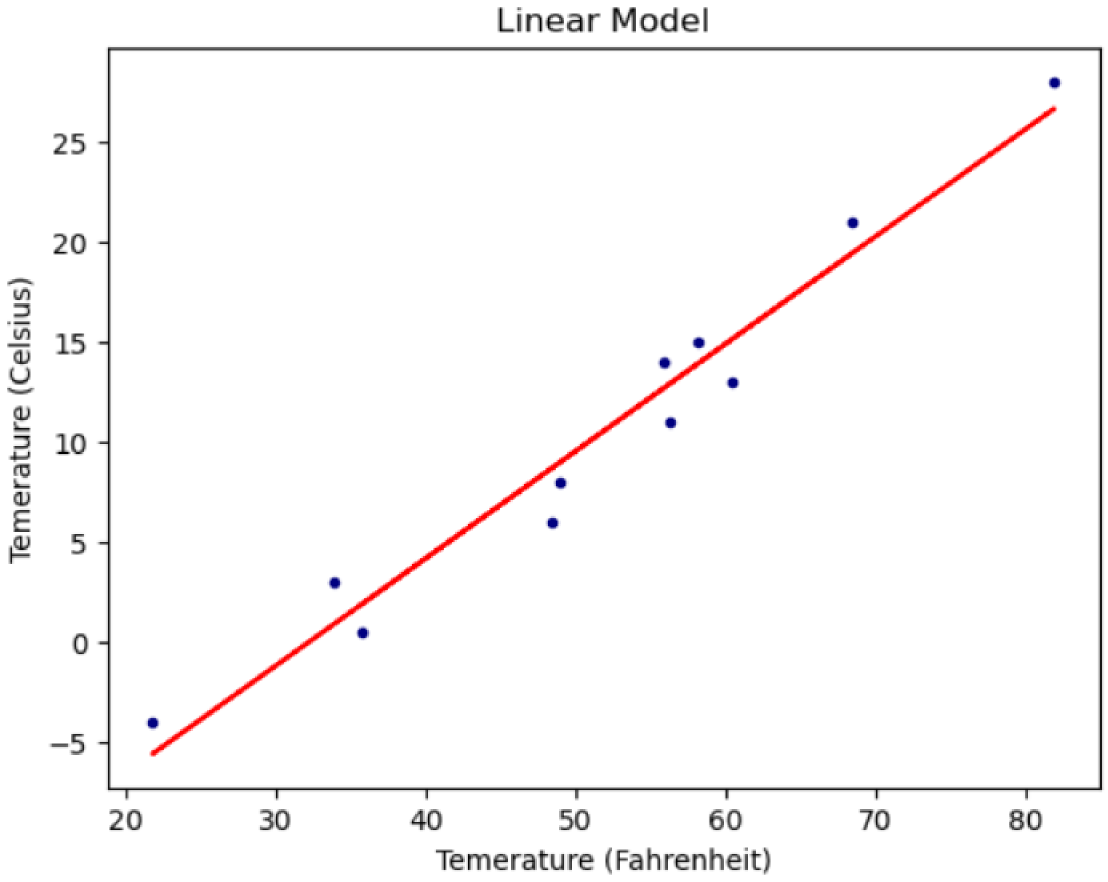


Figure 3: Plot of Linear Model

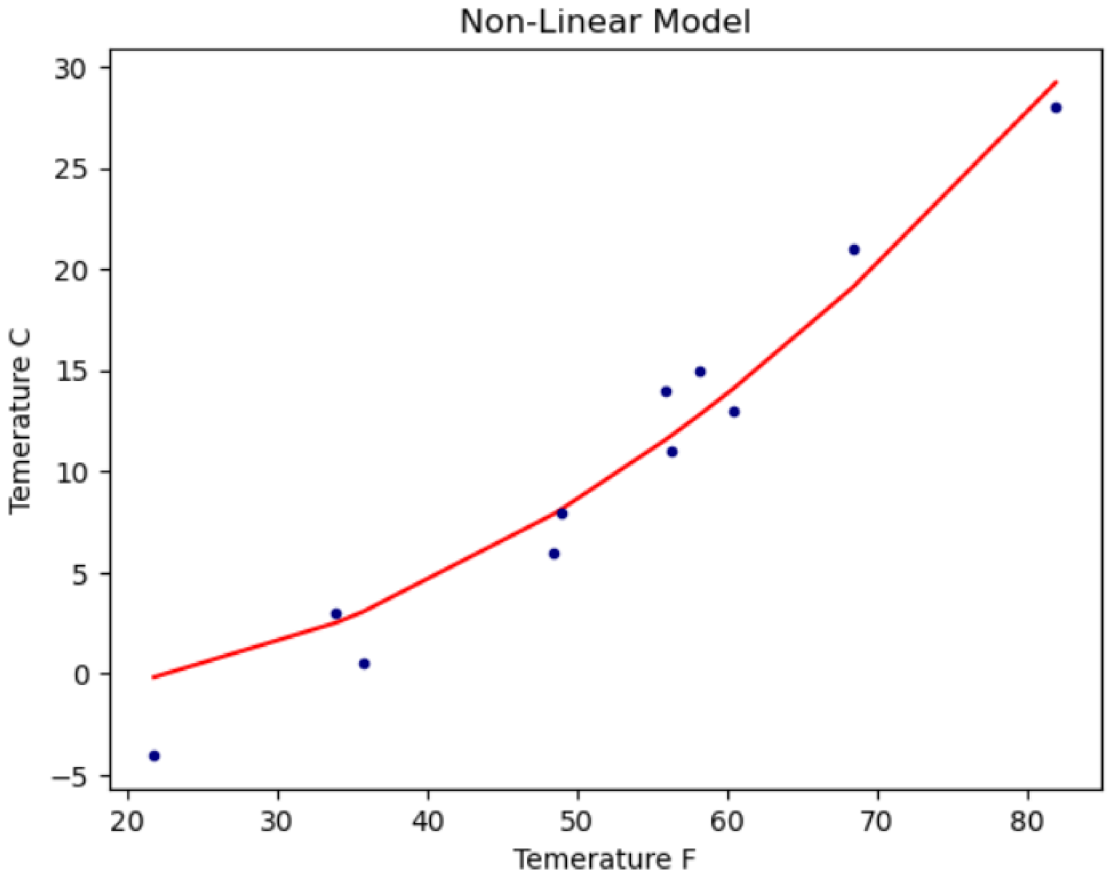


Figure 4: Plot of Non-Linear Model

Looking at the two plots, Figure 3 and Figure 4, above, it can be safely concluded that the Non-Linear model is better than the baseline Linear model as the best fit line is touching more points in the Non-Linear model’s plot.

Problem 2:

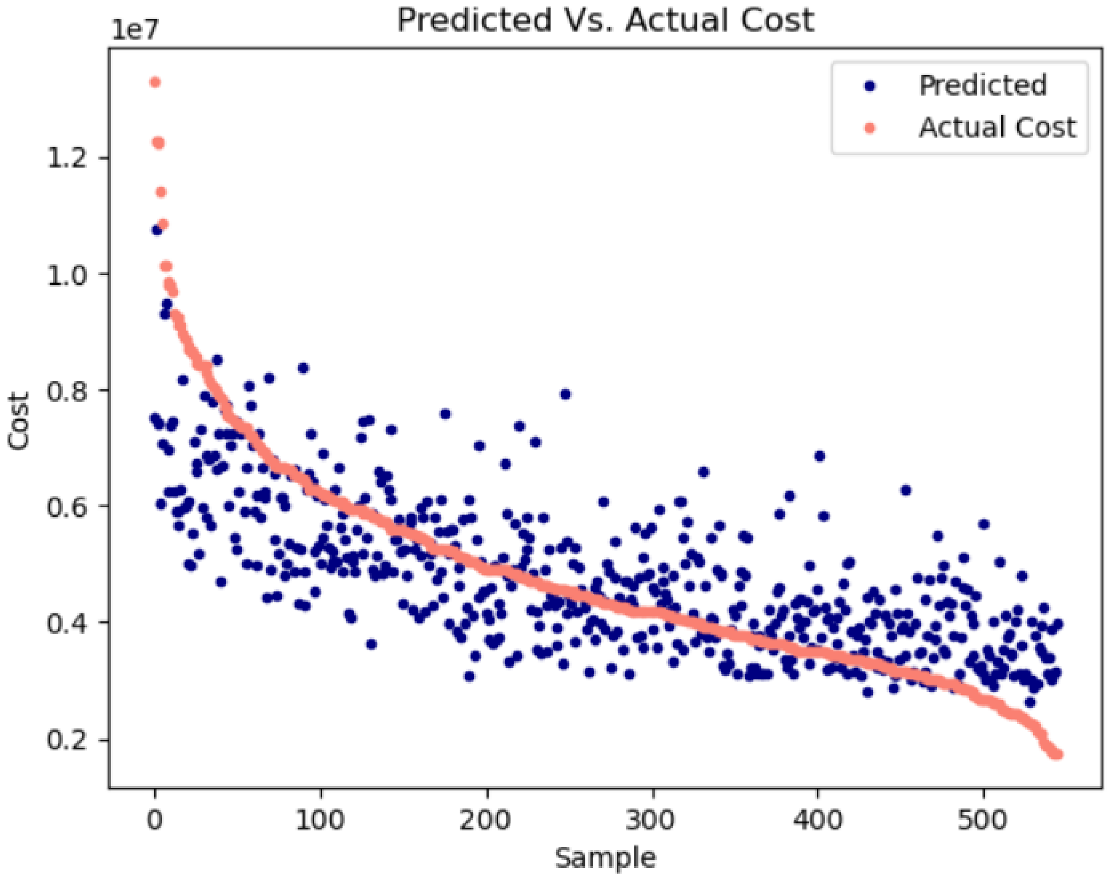
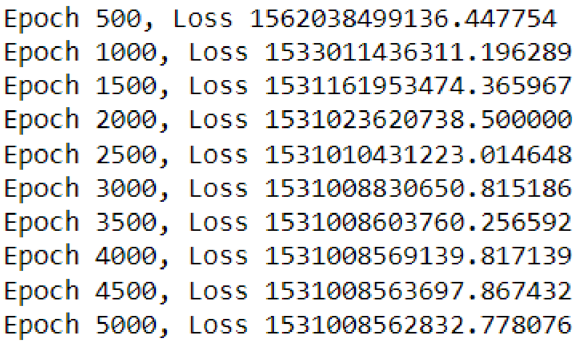
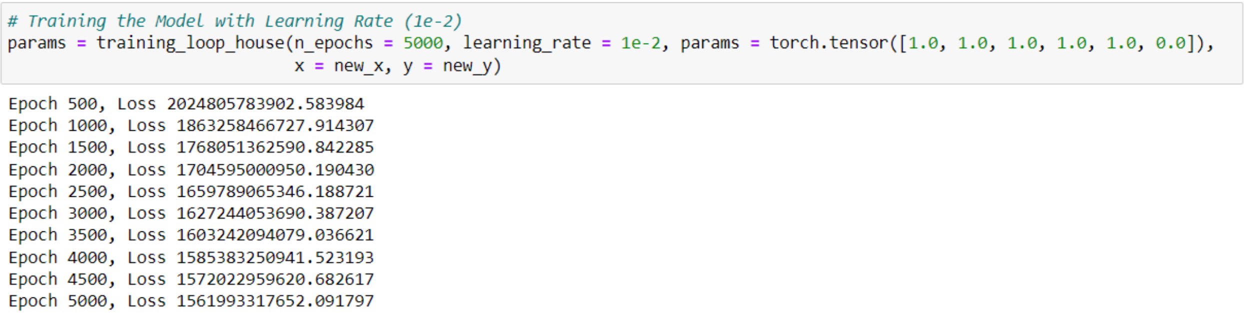


Figure 5: Plot of Predicted Vs. Actual Cost



```
Epoch 500, Loss 1562038499136.447754
Epoch 1000, Loss 1533011436311.196289
Epoch 1500, Loss 1531161953474.365967
Epoch 2000, Loss 1531023620738.500000
Epoch 2500, Loss 1531010431223.014648
Epoch 3000, Loss 1531008830650.815186
Epoch 3500, Loss 1531008603760.256592
Epoch 4000, Loss 1531008569139.817139
Epoch 4500, Loss 1531008563697.867432
Epoch 5000, Loss 1531008562832.778076
```

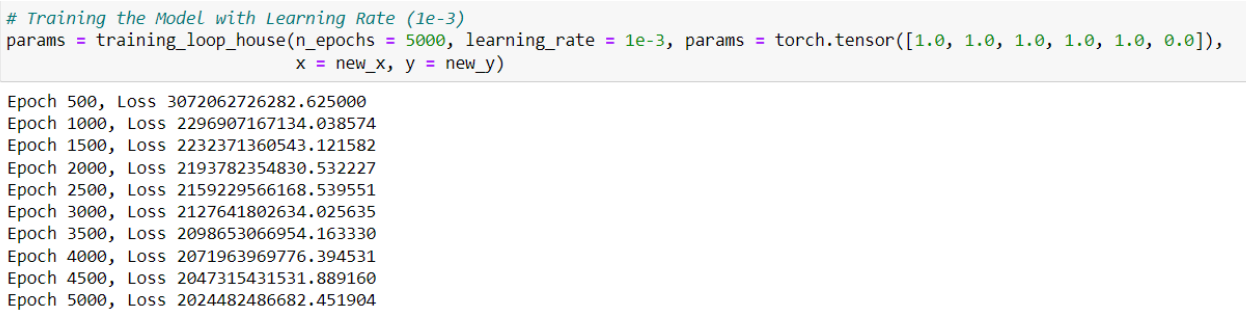
Figure 6: Lowest Loss Values with a Learning rate of 1e-1



```
# Training the Model with Learning Rate (1e-2)
params = training_loop_house(n_epochs = 5000, learning_rate = 1e-2, params = torch.tensor([1.0, 1.0, 1.0, 1.0, 1.0, 0.0]),
                             x = new_x, y = new_y)
```

```
Epoch 500, Loss 2024805783902.583984
Epoch 1000, Loss 1863258466727.914307
Epoch 1500, Loss 1768051362590.842285
Epoch 2000, Loss 1704595000950.190430
Epoch 2500, Loss 1659789065346.188721
Epoch 3000, Loss 1627244053690.387207
Epoch 3500, Loss 1603242094079.036621
Epoch 4000, Loss 1585383250941.523193
Epoch 4500, Loss 1572022959620.682617
Epoch 5000, Loss 1561993317652.091797
```

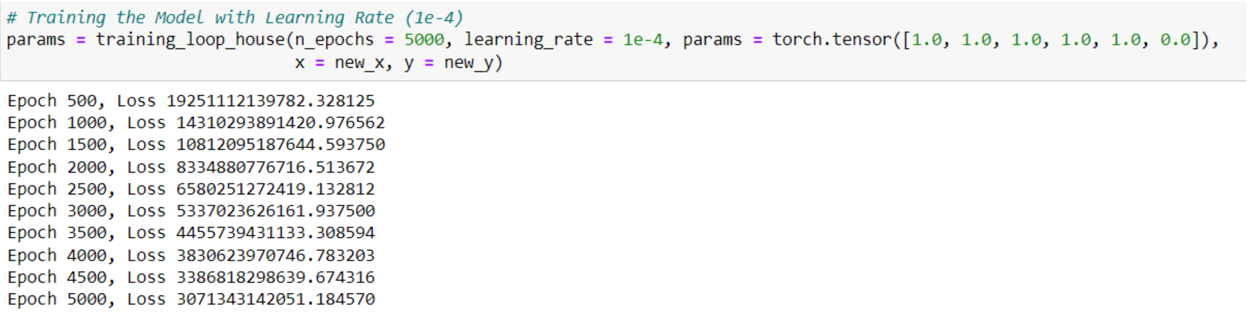
Figure 7: Loss Values with a Learning rate of 1e-2



```
# Training the Model with Learning Rate (1e-3)
params = training_loop_house(n_epochs = 5000, learning_rate = 1e-3, params = torch.tensor([1.0, 1.0, 1.0, 1.0, 1.0, 0.0]),
                             x = new_x, y = new_y)
```

```
Epoch 500, Loss 3072062726282.625000
Epoch 1000, Loss 2296907167134.038574
Epoch 1500, Loss 2232371360543.121582
Epoch 2000, Loss 2193782354830.532227
Epoch 2500, Loss 2159229566168.539551
Epoch 3000, Loss 2127641802634.025635
Epoch 3500, Loss 2098653066954.163330
Epoch 4000, Loss 2071963969776.394531
Epoch 4500, Loss 2047315431531.889160
Epoch 5000, Loss 2024482486682.451904
```

Figure 8: Loss Values with a Learning rate of 1e-3



```
# Training the Model with Learning Rate (1e-4)
params = training_loop_house(n_epochs = 5000, learning_rate = 1e-4, params = torch.tensor([1.0, 1.0, 1.0, 1.0, 1.0, 0.0]),
                             x = new_x, y = new_y)
```

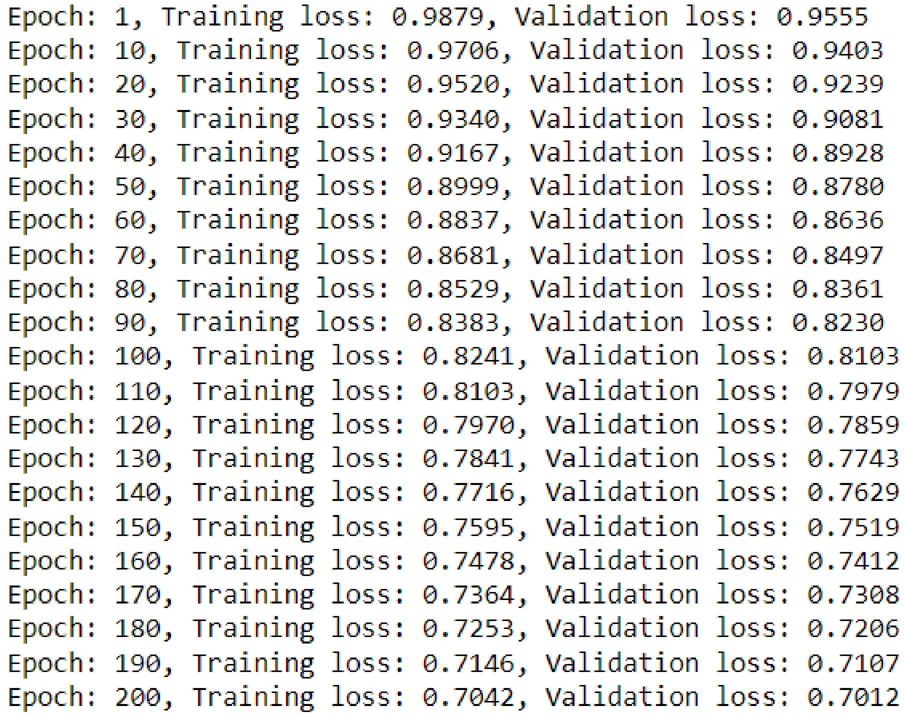
```
Epoch 500, Loss 19251112139782.328125
Epoch 1000, Loss 14310293891420.976562
Epoch 1500, Loss 10812095187644.593750
Epoch 2000, Loss 8334880776716.513672
Epoch 2500, Loss 6580251272419.132812
Epoch 3000, Loss 5337023626161.937500
Epoch 3500, Loss 4455739431133.308594
Epoch 4000, Loss 3830623970746.783203
Epoch 4500, Loss 3386818298639.674316
Epoch 5000, Loss 3071343142051.184570
```

Figure 9: Loss Values with a Learning rate of 1e-4

As required for problem 2, different learning rate values were explored, and the lowest loss was observed with a learning rate of 1e-1. Figure 6, above, shows the same. Figure 7, 8 and 9, report the loss values with a learning rate of 1e-2, 1e-3, and 1e-4.

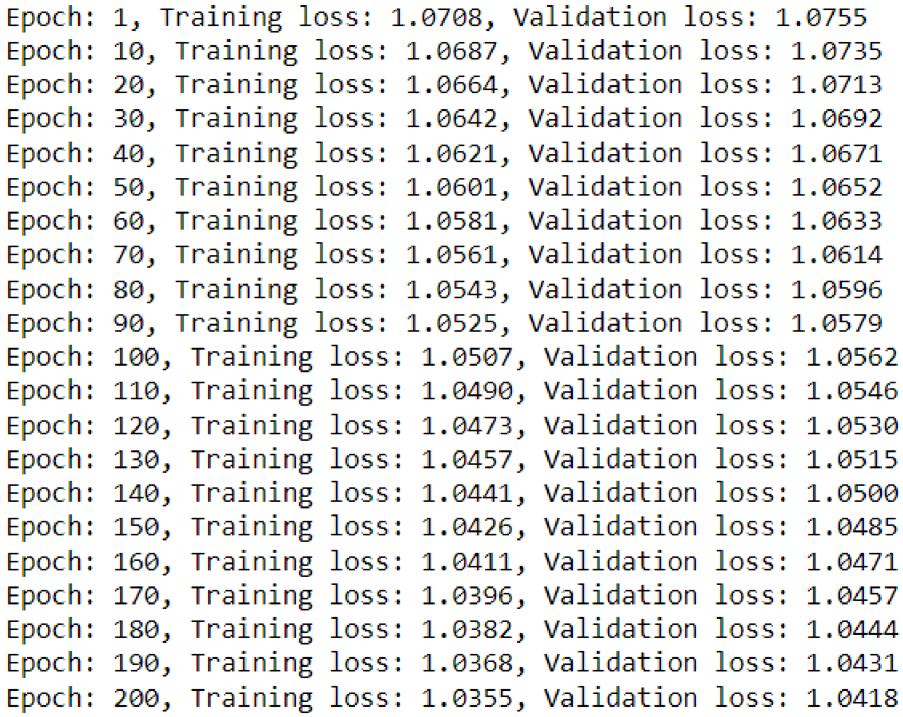
Figure 5, above, shows the plot of the Actual Vs. Predicted Cost with a learning rate of 1e-1 used, which produced the lowest loss values.

Problem 3:



```
Epoch: 1, Training loss: 0.9879, Validation loss: 0.9555
Epoch: 10, Training loss: 0.9706, Validation loss: 0.9403
Epoch: 20, Training loss: 0.9520, Validation loss: 0.9239
Epoch: 30, Training loss: 0.9340, Validation loss: 0.9081
Epoch: 40, Training loss: 0.9167, Validation loss: 0.8928
Epoch: 50, Training loss: 0.8999, Validation loss: 0.8780
Epoch: 60, Training loss: 0.8837, Validation loss: 0.8636
Epoch: 70, Training loss: 0.8681, Validation loss: 0.8497
Epoch: 80, Training loss: 0.8529, Validation loss: 0.8361
Epoch: 90, Training loss: 0.8383, Validation loss: 0.8230
Epoch: 100, Training loss: 0.8241, Validation loss: 0.8103
Epoch: 110, Training loss: 0.8103, Validation loss: 0.7979
Epoch: 120, Training loss: 0.7970, Validation loss: 0.7859
Epoch: 130, Training loss: 0.7841, Validation loss: 0.7743
Epoch: 140, Training loss: 0.7716, Validation loss: 0.7629
Epoch: 150, Training loss: 0.7595, Validation loss: 0.7519
Epoch: 160, Training loss: 0.7478, Validation loss: 0.7412
Epoch: 170, Training loss: 0.7364, Validation loss: 0.7308
Epoch: 180, Training loss: 0.7253, Validation loss: 0.7206
Epoch: 190, Training loss: 0.7146, Validation loss: 0.7107
Epoch: 200, Training loss: 0.7042, Validation loss: 0.7012
```

Figure 10: Results for Part a of Problem 3



```
Epoch: 1, Training loss: 1.0708, Validation loss: 1.0755
Epoch: 10, Training loss: 1.0687, Validation loss: 1.0735
Epoch: 20, Training loss: 1.0664, Validation loss: 1.0713
Epoch: 30, Training loss: 1.0642, Validation loss: 1.0692
Epoch: 40, Training loss: 1.0621, Validation loss: 1.0671
Epoch: 50, Training loss: 1.0601, Validation loss: 1.0652
Epoch: 60, Training loss: 1.0581, Validation loss: 1.0633
Epoch: 70, Training loss: 1.0561, Validation loss: 1.0614
Epoch: 80, Training loss: 1.0543, Validation loss: 1.0596
Epoch: 90, Training loss: 1.0525, Validation loss: 1.0579
Epoch: 100, Training loss: 1.0507, Validation loss: 1.0562
Epoch: 110, Training loss: 1.0490, Validation loss: 1.0546
Epoch: 120, Training loss: 1.0473, Validation loss: 1.0530
Epoch: 130, Training loss: 1.0457, Validation loss: 1.0515
Epoch: 140, Training loss: 1.0441, Validation loss: 1.0500
Epoch: 150, Training loss: 1.0426, Validation loss: 1.0485
Epoch: 160, Training loss: 1.0411, Validation loss: 1.0471
Epoch: 170, Training loss: 1.0396, Validation loss: 1.0457
Epoch: 180, Training loss: 1.0382, Validation loss: 1.0444
Epoch: 190, Training loss: 1.0368, Validation loss: 1.0431
Epoch: 200, Training loss: 1.0355, Validation loss: 1.0418
```

Figure 11: Results for Part b of Problem 3

Part a of Problem # 3 was performed with only one hidden layer with 8 nodes. An extension of part a, for part b, two more hidden layers were added, and the same operation was performed. Figure 10, and Figure 11, above, show the same. Comparing the model from part b, it can be concluded that the model in part a is better as the loss is less than that of the loss observed in part b.