ECE 607 Project 2 Part 1

Task 1

Files: task\_1.m proj\_2\_regress.m

The file proj\_2\_regress.m is the function I wrote to perform linear regression on a given set of data. It takes as inputs the file containing the training data, the file containing the data to be fit to the model, and the number of columns of the training data which are to be treated as inputs. The output of the function is the model’s predicted outputs from the inputs in the data.

The file task\_1.m is the main file for this task. In this file, I apply the function proj\_2\_regress to the given training data and the testing data. The file outputs the mean, standard deviation, and max of the training and testing error for each output. For the error, I used the following formula where y is the actual output and t is the desired output: . These results are reproduced in the below table.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Error | Output 1 | Output 2 | Output 3 | Output 4 | Output 5 | Output 6 | Output 7 |
| Training Mean | 2.92% | 22.25% | 28.26% | 19.12% | 9.71% | 16.44% | 94.16% |
| Training Standard Deviation | 2.04% | 25.03% | 33.38% | 16.15% | 14.92% | 14.58% | 714.78% |
| Training Max | 17.22% | 206.75% | 278.77% | 148.52% | 128.38% | 121.70% | 19055.67% |
| Testing Mean | 2.85% | 23.07% | 29.52% | 19.23% | 10.40% | 16.53% | 166.44% |
| Testing Standard Deviation | 1.94% | 26.33% | 33.79% | 16.63% | 18.02% | 14.59% | 1410.42% |
| Testing Max | 9.36% | 175.43% | 215.43% | 126.04% | 175.77% | 110.44% | 37460.17% |

As can be seen from the table, the model performed best for output 1 and worst for output 7.

Task 2

Files: task\_2.m proj\_2\_class.m

The file proj\_2\_class.m is a function that performs linear classification. It takes as inputs the file containing the training data, the file containing the data to be classified, and lambda, which is the regularization coefficient. For task 2, I set lambda equal to zero. The function takes the last column of the training file to be the classes and all previous columns to be the inputs. The file to be classified may or may not contain outputs; the same number of columns as are inputs in the training file are taken to be inputs. The function uses the 1-of-k encoding scheme to convert the classes into a row vector of outputs then uses the same algorithm as in task one to perform linear regression with these outputs. The predicted class is then taken to be the class corresponding with the highest predicted output.

The main file, task\_2.m, applies this function to the training and testing data from project 1. Then it calculates the accuracy by dividing the number of correctly classified rows by the total number of rows and multiplying by 100 to get a percent. The training accuracy is 85.5% and the testing accuracy is 88%.

Task 3

Files: task\_3.m proj\_2\_class.m

The main file for this task is task\_3.m. It uses the same function for classification as Task 2, but I used a for loop to run though each given value of lambda. Accuracy was calculated in the same way as in task 2, and the results are reported in the following table:

|  |  |  |
| --- | --- | --- |
| Lambda | Training Accuracy | Testing Accuracy |
| 0.01 | 87.97% | 86.00% |
| 0.1 | 87.97% | 86.03% |
| 0.5 | 87.97% | 86.07% |
| 1 | 88.03% | 86.03% |
| 5 | 88.10% | 86.23% |
| 10 | 88.20% | 86.43% |

As can be seen from the table, the model achieves the best accuracy for both training and testing with the regularization coefficient set to 10.