**Homework 1:** Written Responses

**Problem 1**

**1a.** For each dataset, which lambda value gives the least **test** set MSE?

*Response:* Please see the table below. The corresponding graphs can be found in the “graphs” folder.

*Homework 1, Problem 1a:*

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset | Graph Name | Lambda | Minimum Test MSE |
| Train-100-10, test-100-10 | hw1\_1a\_100\_10.png | 9 | 4.16 |
| Train-100-100, test 100-100 | hw1\_1a\_100\_100.png | 22 | 5.07 |
| Train-1000-100, test-1000-100 | hw1\_1a\_1000\_100.png | 27 | 4.32 |
| Train-50(1000)-100, test-1000-100 | hw1\_1a\_50\_1000\_100.png | 8 | 5.51 |
| Train-100(1000)-100, test-1000-100 | hw1\_1a\_100\_1000\_100.png | 19 | 5.20 |
| Train-150(1000)-100, test(1000)-100 | Hw1\_1a\_150\_1000\_100.png | 24 | 4.84 |

**1b.** For each of the datasets 100-100, 50(1000)-100, 100(1000)-100, provide an additional graph with lambda ranging from 1 to 150.

*Response:* The corresponding graphs can be found in the “graphs” folder.

|  |  |
| --- | --- |
| Dataset | Graph Name |
| Train-100-100, test 100-100 | hw1\_1b\_100\_100.png |
| Train-50(1000)-100, test-1000-100 | hw1\_1b\_50\_1000\_100.png |
| Train-100(1000)-100, test-1000-100 | hw1\_1b\_100\_1000\_100.png |

**1c.** Explain why lambda = 0 (i.e., no regularization) gives abnormally large MSEs for those three datasets in (b).

*Response:* Each of the datasets has a high number of parameters in comparison to a relatively low number of observations to train the data on; resulting in very high overfitting. For example, the dataset “Train-100-100” has 100 parameters and only 100 observations, meaning there aren’t enough observations to create an accurate model that uses 100 parameters. The solution to this problem is to either use fewer parameters, such as the 100-10 model that performed much better, or to use significantly more observations to train the models.

**Problem 2**

**2a.** Using CV technique, what is the best choice of lambda value and the corresponding test set MSE for each of the six datasets?

*Response:* Please see the table below for the best choice of lambda value and its corresponding test set MSE for each dataset using the CV technique.

*Homework 1, Problem 2a:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset | CV Lambda | CV Minimum Test MSE | 1(a)  Lambda | 1(a) Min  Test MSE |
| Train-100-10 | 13 | 4.186549 | 9 | 4.16 |
| Train-100-100 | 20 | 4.466572 | 22 | 5.07 |
| Train-1000-100 | 39 | 4.139641 | 27 | 4.32 |
| Train-50(1000)-100 | 24 | 5.285221 | 8 | 5.51 |
| Train-100(1000)-100 | 31 | 4.85221 | 19 | 5.20 |
| Train-150(1000)-100 | 47 | 4.876913 | 24 | 4.84 |

**2b.** How do the values for lambda and MSE obtained from CV compare to the choice of lambda and MSE is question 1(a)?

*Response:* The lambda values from the CV technique are, in general, slightly higher than the lambda values in question 1(a). The greatest difference is from the Train-150(1000)-100 dataset with a difference of 23. The CV MSEs and the MSEs from 1(a) are relatively comparable, with the greatest difference of ~0.6 from dataset Train-100-100.

**2c.** What are the drawbacks of CV?

*Response:* The two biggest drawbacks of cross validation are that it is computationally inefficient and that you are not able to use the entirety of the training set to train the model. Cross validation is computationally inefficient because you must calculate the MSE k number of times (where k is the number of folds) instead of calculating the MSE once using the training dataset and testing the model on the test dataset. In cross validation you cannot use the entirety of the training set because a portion of the training data is reserved as the test data for each run, where you have k runs for the number of folds.

**2d.** What are the factors affecting the performance of CV?

*Response:* The size of your dataset, or number of observations, may be the biggest factor affecting the performance of CV. Because the dataset is being subdivided, where one fold is reserved for testing, it means that there is less data to train the model on. Another factor that can greatly influence the performance of CV is the choice of K folds. The total cost of computation is the number of k folds multiplied by the number of choices of lambda. This means that the greater number of folds (k), the more inefficient the model will perform, with maximum number for k-folds being the number of observations in the training dataset (N).

**Problem 3**

Fix lambda = 1, 25, 150. For each of these values, plot a learning curve for the algorithm using the dataset 1000-100.csv.

*Response:* The corresponding graphs can be found in the “graphs” folder. I understand conceptually that the train and test should look like reflections of each other across the y-axis, almost converging as the subset size increases, but my graphs do not look perfectly like the ideal. My lambda = 1 and Lambda = 25 graphs look like a rough approximation of the ideal, but my Lambda = 150 does not. I think this may be because Lambda = 150 is considerably larger than the optimum lambda value of 27 (according to 1(a)), causing a similar issue that we saw in the models from 1(b) where there was not sufficient data in comparison to the number of parameters. I think this also explains why the Lambda = 25 graph is relatively flatter than the Lambda = 1 graph; since Lambda = 25 is so close to the optimum, the model is more efficient and therefore needs less data to begin converging at an MSE.

*Homework 1, Problem 3:*

|  |  |
| --- | --- |
| Dataset | Graph Name |
| Lambda = 1 | Hw1\_prob3\_lam1 |
| Lambda = 25 | Hw1\_prob3\_lam25 |
| Lambda = 150 | Hw1\_prob3\_lam150 |