Assignment 3

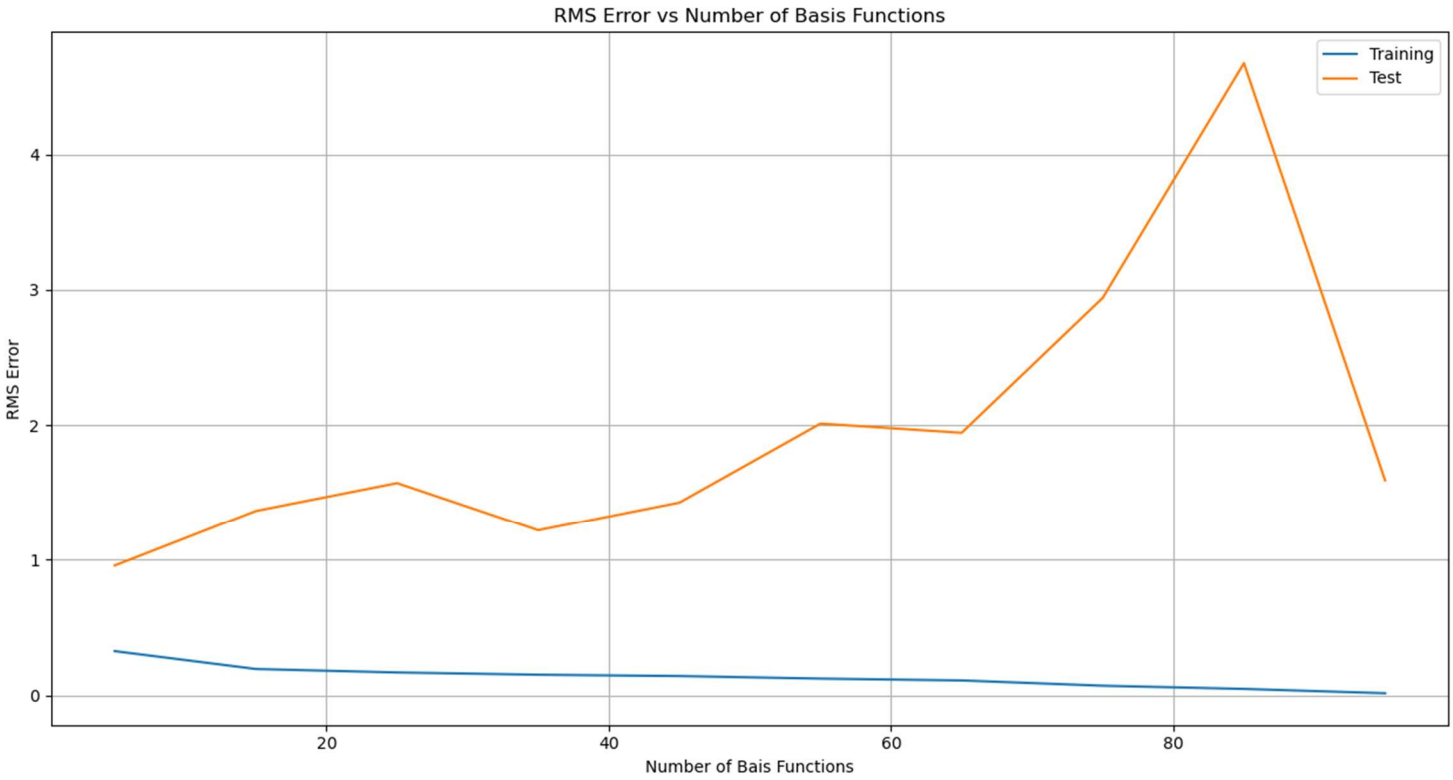
ECE 736: 3D Image Processing and Computer Vision

Pranav Rawal

Department of Electrical & Computer Engineering, McMaster University

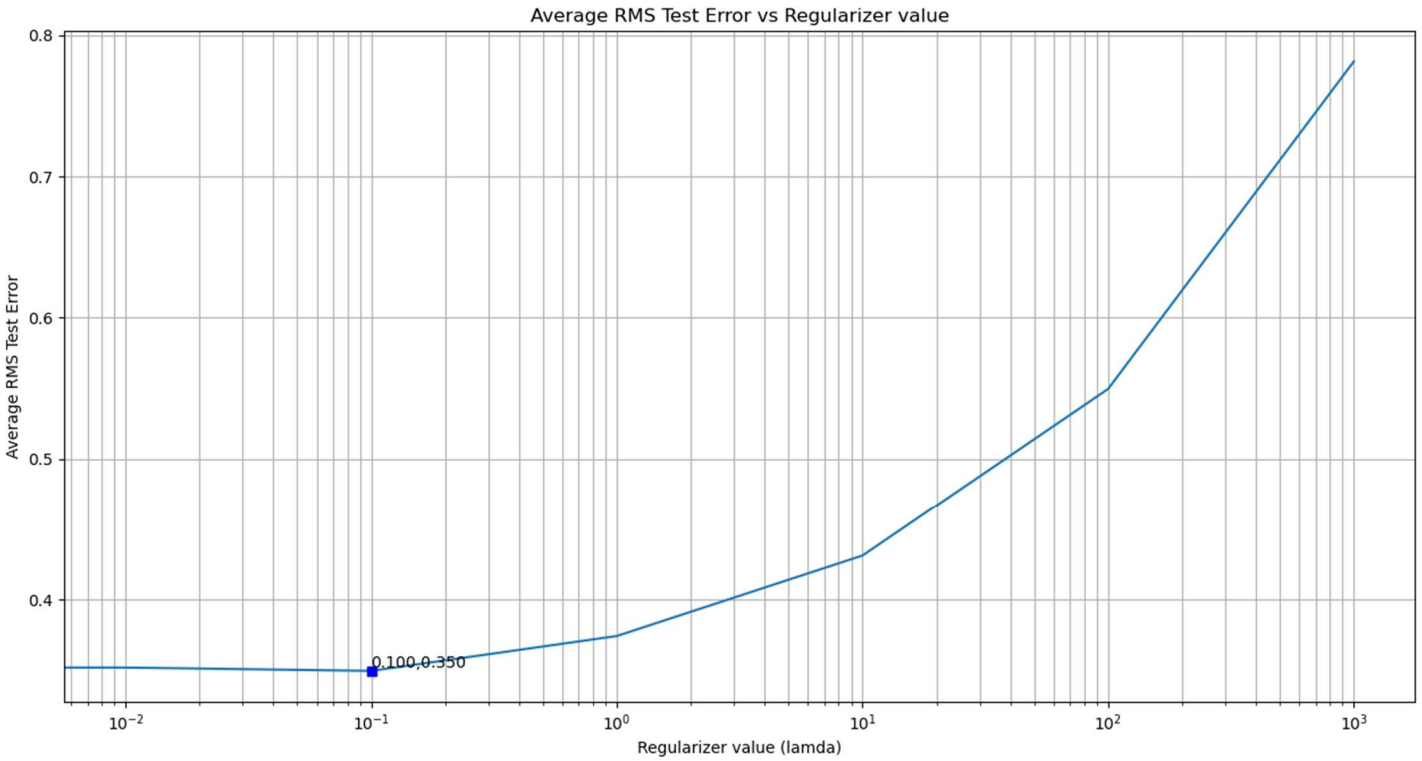
Question 1: Multivariate Linear Regression with Gaussian Basis Functions on AutoMPG dataset.

a) **question1a.py** is the python script for this question



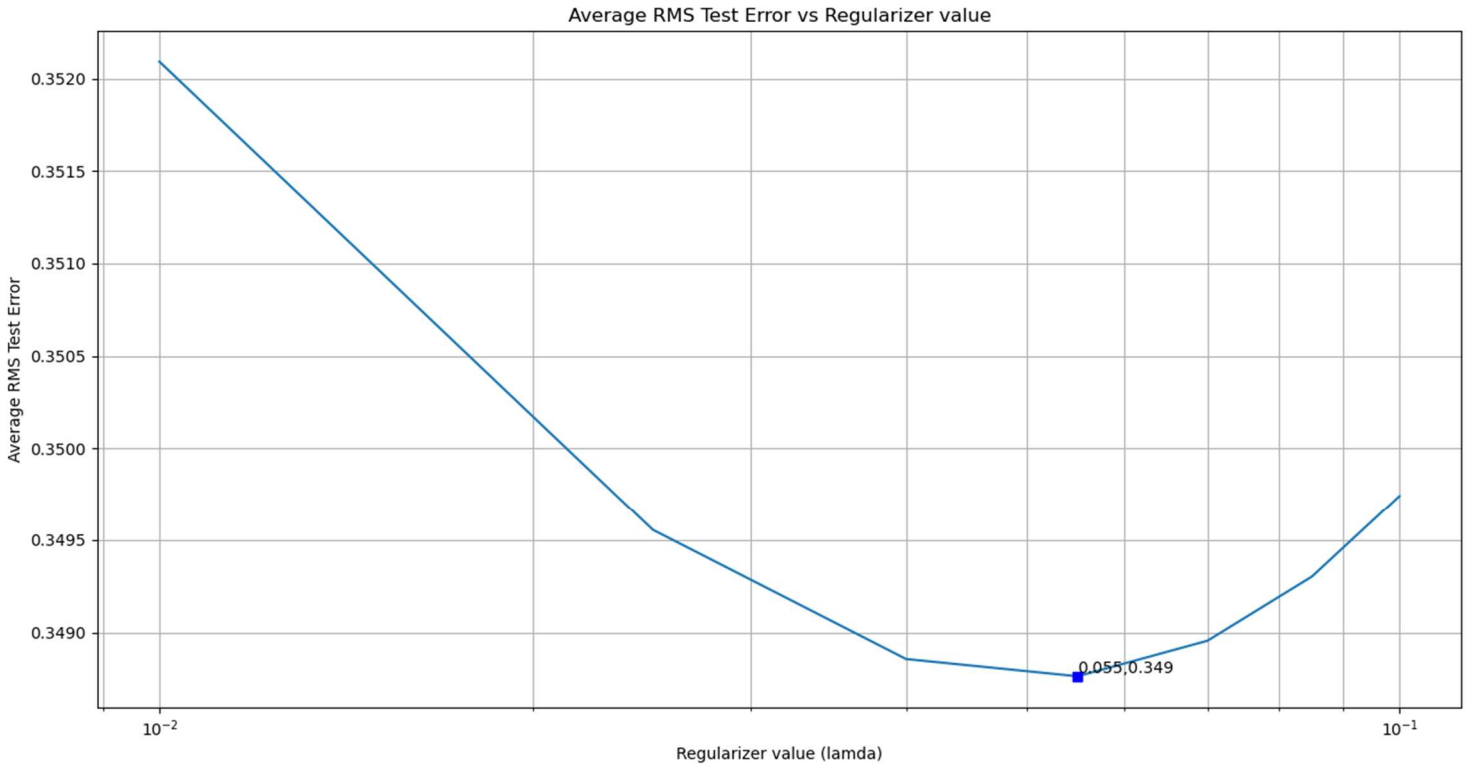
The plot above indicates that with an increase in basis functions we get an increase in test error and decrease in training error. The increase in test error is attributed to overfitting the data as we are creating a model with increasing complexity with more coefficients with very high values. The decrease in training error is a result of the complex model being a very good fit for the training data, however, a terrible fit for any test values.

b) **question1b.py** is the python script for this question



Seeing as how the plot from part (a) was clearly overfitting the dataset, L2 regularization was added to the regression model to get a better fit.

The plot above was generated by doing 10-fold cross validation to determine the best value for lamda, the regularizer value in the L2 regularization. There are values of lamda being tested (0, 0.01, 0.1, 1, 10, 100, 1000). From the plot above we can see that we get a minimum RMS test error for lamda = 0.1. To be more precise on the most ideal regularizer value I have gone one step further and defined a new range of values for lamda. This new range is defined by lamda = np.linspace(0.01,0.1,7), which are seven values from 0.01 to 0.1. The plot for this new range is provided below.



From this new plot we can see the minimum average RMS test error occurs for a **lamda of 0.055**. This exercise can be repeated to get a more precise lamda value, however, seeing as how the minimum average RMS test error is not reducing by a significant amount I will stop at this point.

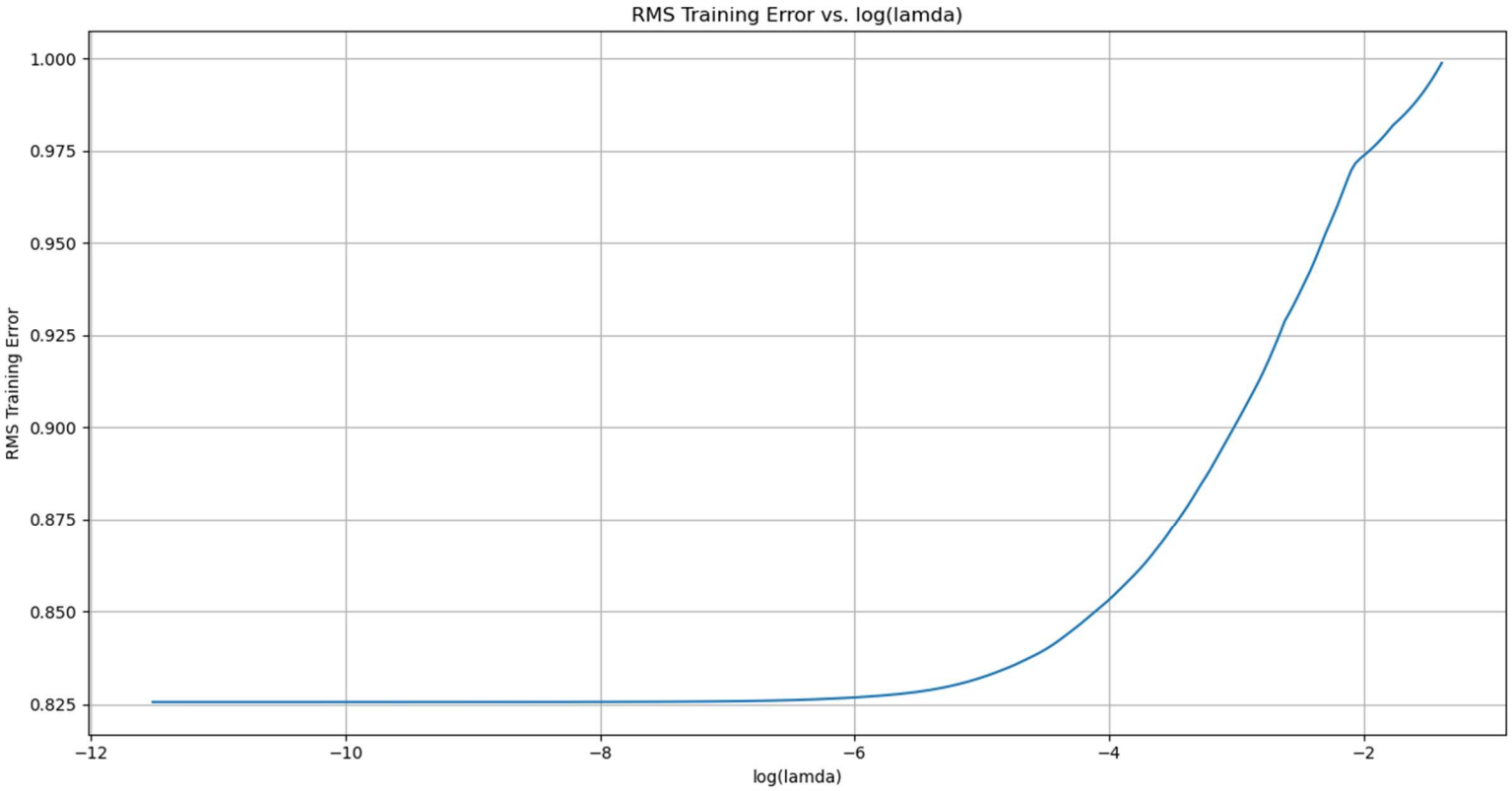
Question 2: Multivariate Linear Regression with Lasso Regularizer

Question2.py is the python script for this question.

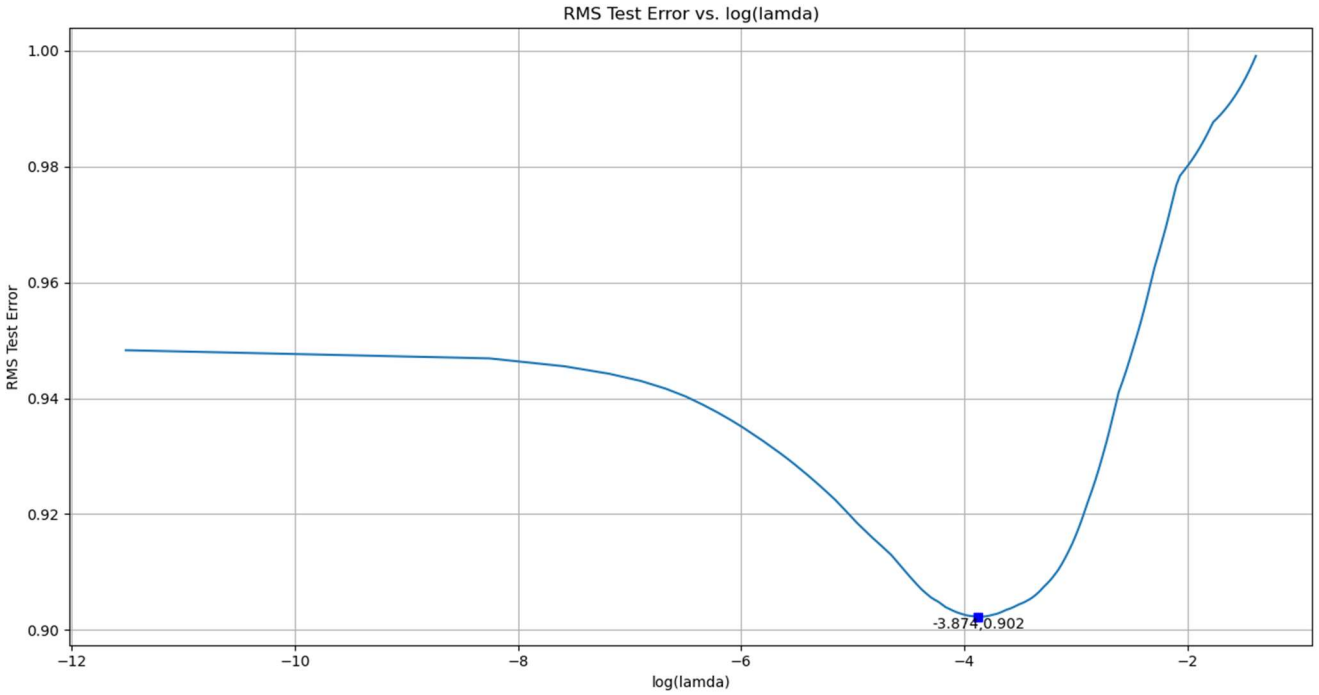
The musicdata dataset has 90 measures of timbre for each of the songs. Its not necessary that all of the 90 features be relevant for predicting which year a new song may have been released, therefore lasso regression can be used to determine which of the 90 features are most useful for the prediction.

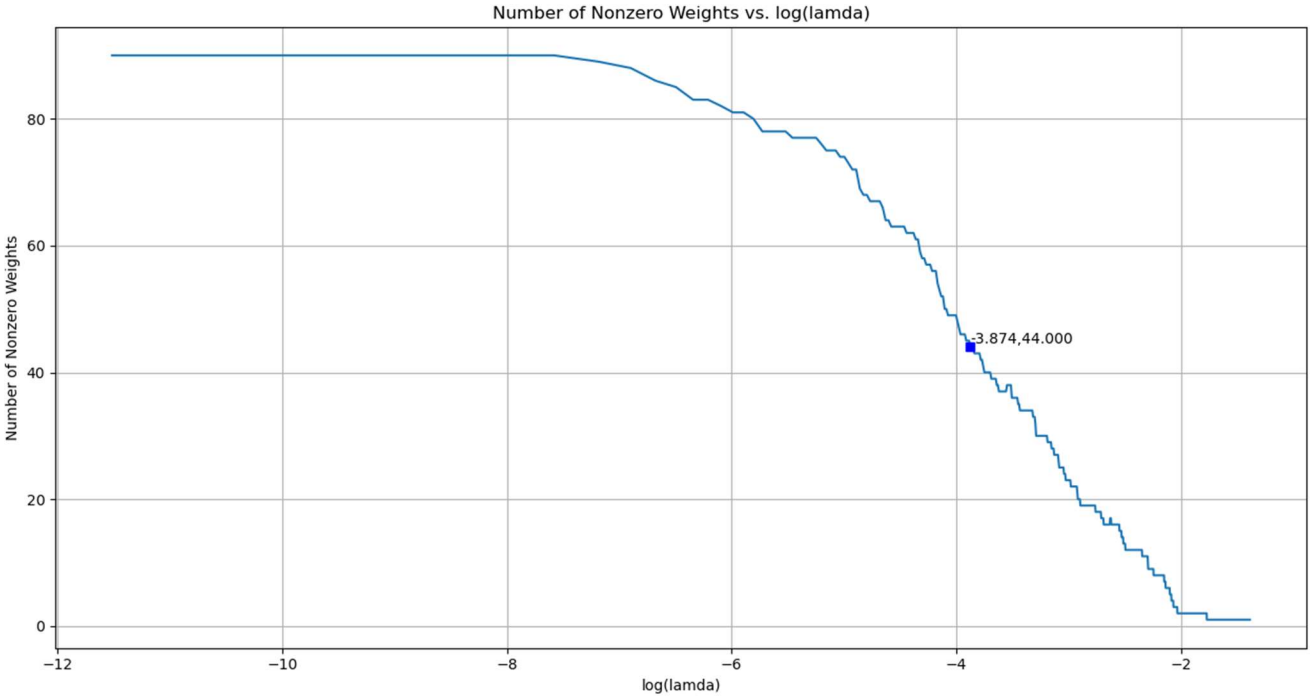
Lasso (L1) and Ridge (L2) Regression are similar in the sense that they force the coefficients of the estimated model to be smaller in value and prevent overfitting the data by providing some value for lamda, the regularizer value. But where they differ is that Lasso regularization will set the coefficients of irrelevant features to zero, whereas Ridge Regression will set the coefficients to very low value but not zero.

In this exercise we will see that by increasing the value of lamda from zero to a higher value in the lasso regularizer, the number of nonzero terms in the coefficient vector will reduce. The range of values for lamda I have chosen are 1e-5 to 0.25. On the next page there are three plots for log(lamda) vs. RMS test error, log(lamda) vs. RMS training error, and log(lamda) vs. Number of Nonzero coefficients.



As we can see the RMS training error increases as we penalize larger values for the coefficients by increasing the lamda value in lasso regularizer. This is because by penalizing large values for the coefficients we are creating a model that does not fit the training data as well, but the goal of any regularization technique is to get more accurate predictions for the test set and other new inputs not already encountered in the training of the model.





The previous two plots show us that a log(lamda) value of -3.874 is related to a local minimum in the RMS test error, which is related to 44 nonzero coefficients.

Next steps could be to do L2 regularization on the 44 features related to the 44 nonzero coefficients to further reduce the RMS test error.

Question 3: Logistic Regression and Support Vector Machine for Classification

```
Logistic Regression Confusion Matrix:
 [[24  1]
 [ 0 25]]
Logistic Regression Accuracy: 0.98
Logistic Regression Precision: 0.9615384615384616
Logistic Regression Recall: 1.0
SVM Confusion Matrix:
 [[22  3]
 [ 0 25]]
SVM Accuracy: 0.94
SVM Precision: 0.8928571428571429
SVM Recall: 1.0
```

Question3.py is the python script for this question.

I am using scikit learn's libraries to run logistic regression and SVM models. Both are very accurate but for this particular dataset, logistic regression seems to be slightly more accurate.