# CSE 574 Introduction to Machine Learning
## Programming Assignment 2
# Classification and Regression

**Course Name**: **CSE 574** Introduction to Machine Learning

**Group Number**: 37

**Team Members**:
Arvind Srinivass Ramanathan (arvindsr)
Barath Eswer Nagasubramaniyan (barathes)
Senthil Kumar Laguduva Yadindra Kumar (laguduva)

# Problem 1: Experiment with Gaussian Discriminators

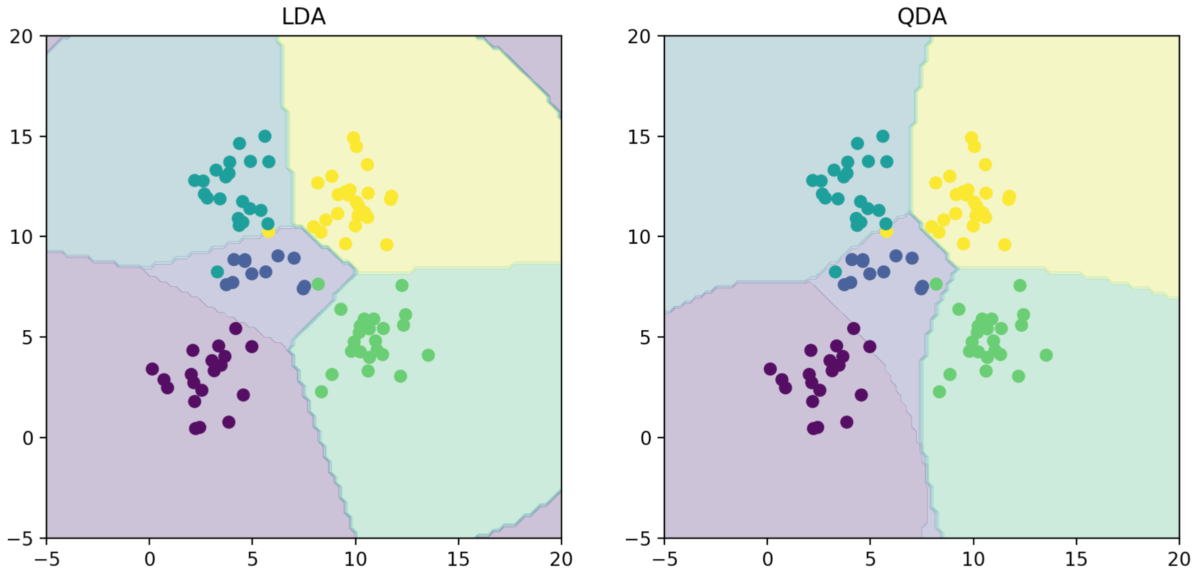



*Figure 1 Discriminating boundary for linear and quadratic*

**LDA Accuracy = 97%**
**QDA Accuracy = 96%**

The difference that can be seen in the above two plots is the fact that we get straight line decision boundaries for Linear Discriminant Analysis (LDA), while we get curved decision boundaries for Quadratic Discriminant Analysis (QDA). The difference between the two classifications is that when it came to LDA, we had only a single covariance matrix in hand for the training data and also, calculated used only one while writing the test function. Here, the covariance matrix being the same results in a straight-line boundary and the same when it comes to the Gaussian density. Classification in the case of LDA is fairly easier.

When it comes to QDA, there was a different covariance matrix for each of the classes, we get a graph where decision boundaries try to give a better fit to the data. And hence, we get the curved decision boundaries. QDA gives more accuracy to the classification in general. But QDA is comparatively more complex and expensive when compared to LDA. In this case, we get more accuracy for the LDA since the amount of data we have is less and is fairly well grouped. When we have a large amount of data with more diverse and complex data, QDA would definitely give a better accuracy.

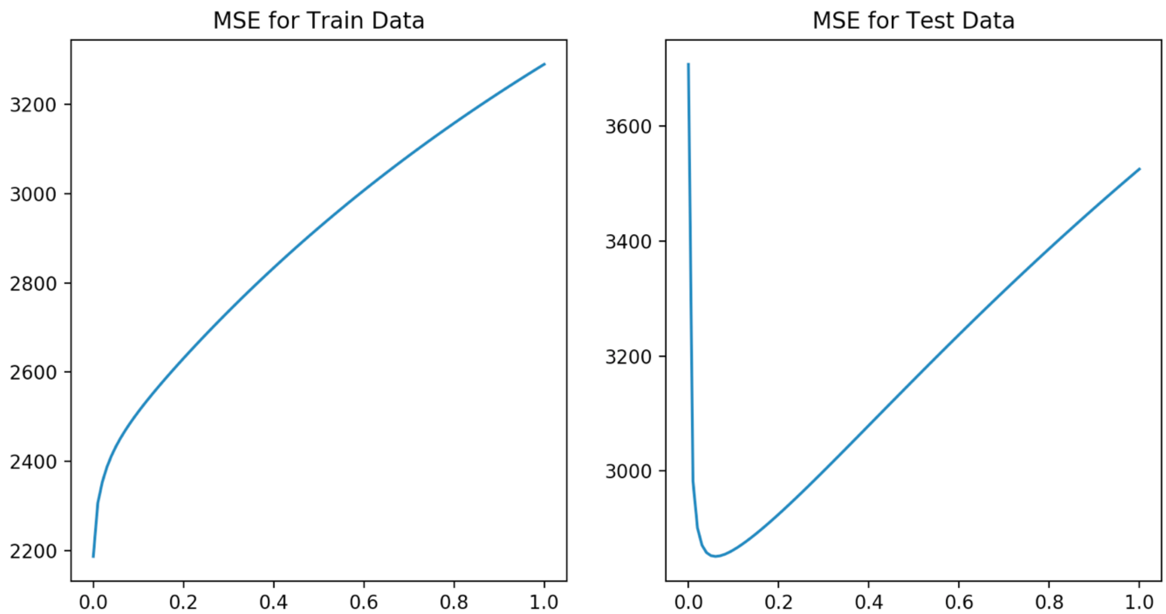
# Problem 2: Experiment with Linear Regression

| | MSE WITHOUT INTERCEPT | MSE WITH INTERCEPT |
|---|---|---|
| TRAIN DATA | 19099.44684 | 2187.160295 |
| TEST DATA | 106775.3616 | 3707.840182 |

*Table 1 MSE with and without intercept*

Linear Regression definitely performs better when we use an intercept. When no intercept is used, the line that is trying to fit the data generally passes through the origin. Hence, it is not a great fit to the actual data. When using an intercept, the line does not pass through the origin and there is more flexibility to more closely fit the data points. When you look at the MSE values, you see that the error value reduces greatly for both the training and the test data, but the reduction is more for the test data. This is because we are trying to reduce the error of an already trained system, and thus we see better results.

# Problem 3: Experiment with Ridge Regression



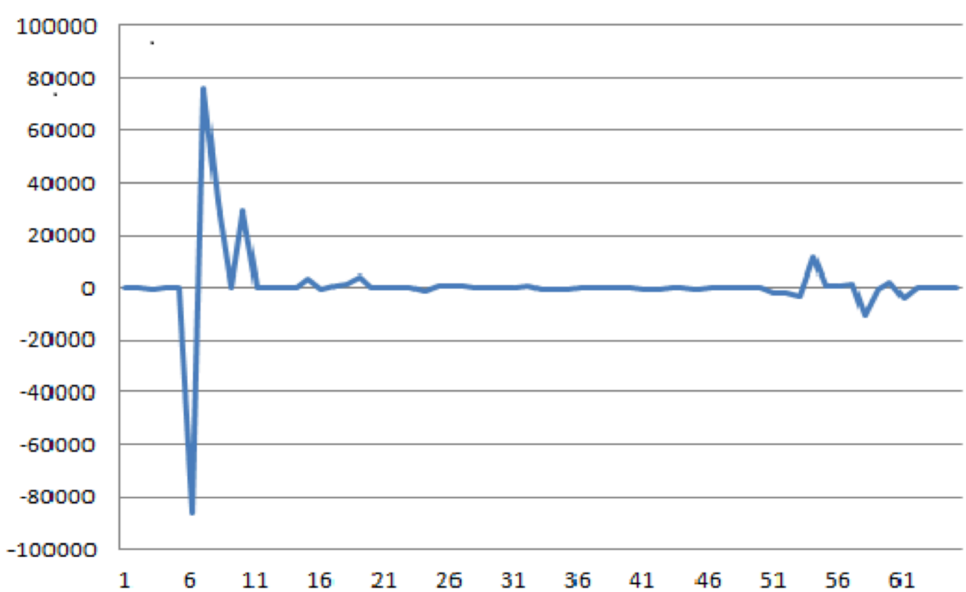




*Figure 2 Weights learnt using linear regression using intercept*

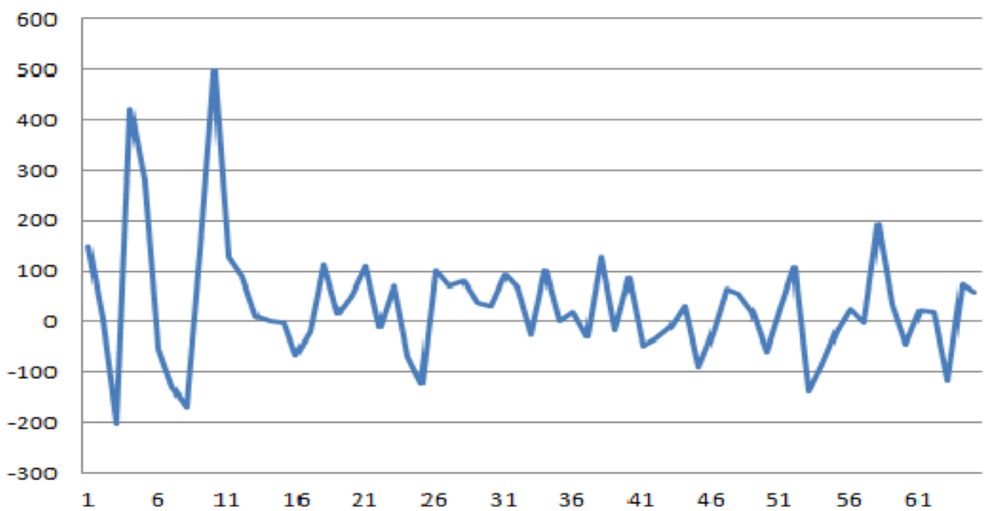



*Figure 3 Weights learnt using ridge regression using intercept*

| Lambda | Error in Train Data | Error in Test Data |
|---|---|---|
| 0.06 | 2451.528491 | 2851.330213 |
| 0.07 | 2468.077553 | 2852.349994 |
| 0.05 | 2433.174437 | 2852.665735 |
| 0.08 | 2483.365647 | 2854.879739 |
| 0.04 | 2412.119043 | 2858.00041 |
| 0.09 | 2497.740259 | 2858.444421 |
| 0.1 | 2511.432282 | 2862.757941 |
| 0.11 | 2524.600039 | 2867.637909 |
| 0.03 | 2386.780163 | 2870.941589 |
| 0.12 | 2537.3549 | 2872.962283 |

Error in data was measured by incrementing the lambda value in small values of 0.01. The system is trained based on what values is generated for train data and the same system was used to obtain MSE values for test data. This is a regularization parameter. Here we observed for optimal Lambda value **0.06**, we obtain the least value of Mean Square Error for test data.

**MSE** for **Training Data** with Intercept using **Linear Regression** : **2187.16029493**
**MSE** for **Test Data** with Intercept using **Linear Regression** : **3707.84018103**

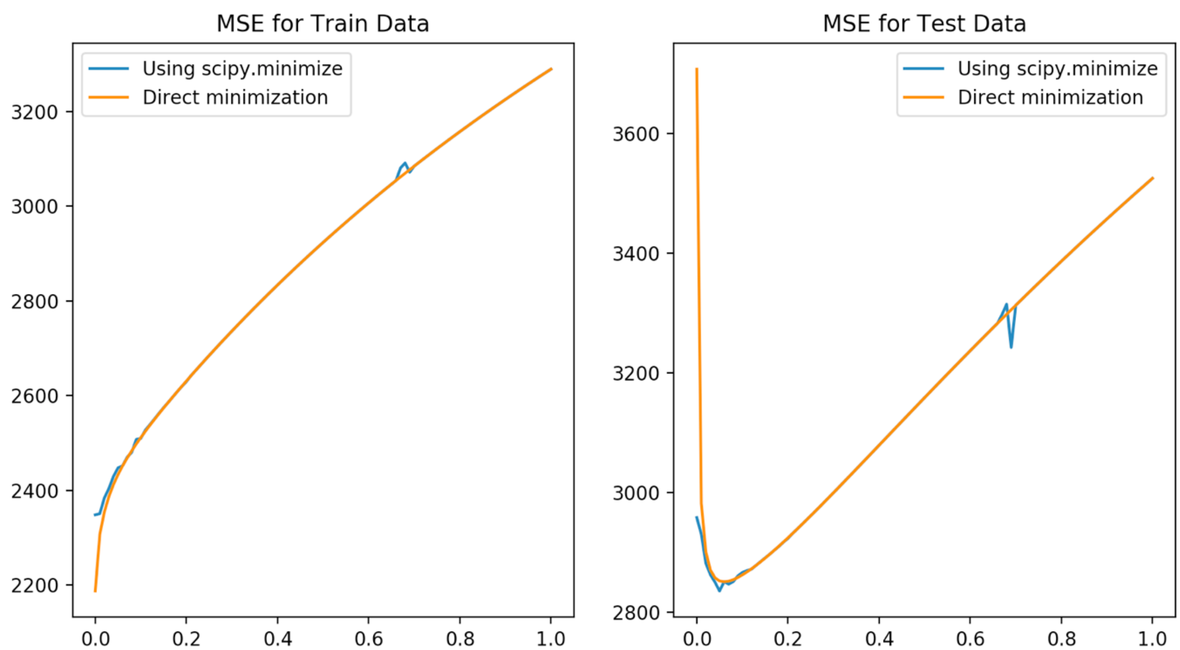
| Lambda | Error in Train Data |
|---|---|
| 0 | 2187.16029493 |

*Table 2 Error in Training Data for **Ridge Regression***

| Lambda | Error in Test Data |
|---|---|
| 0.06 | 2851.330213 |

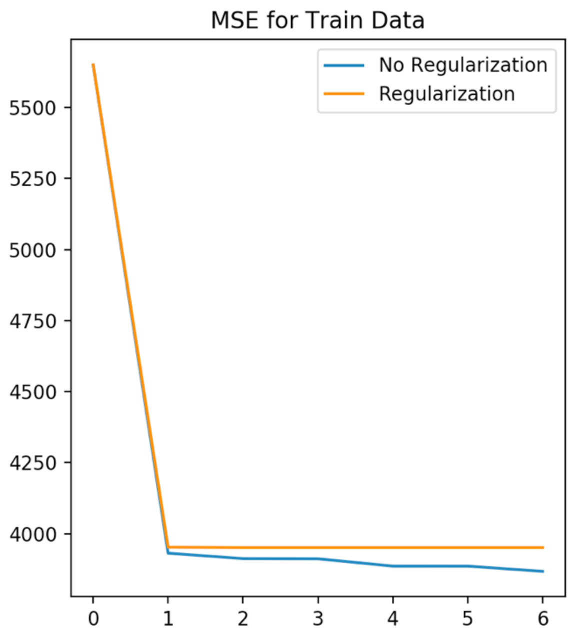
*Table 3 Error in Test Data for **Ridge Regression***

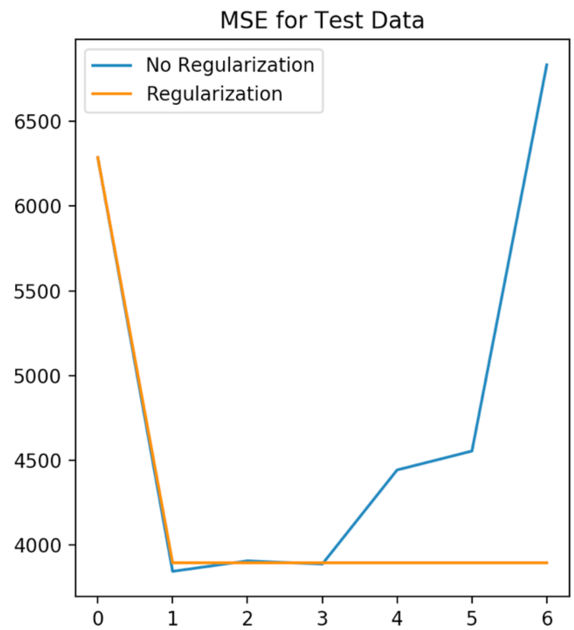
# Problem 4: Using Gradient Descent for Ridge Regression Learning




The results obtained by using only Ridge Regression (problem 3) and by using Ridge Regression with Gradient descent (problem 4) are almost the similar. Both give very smooth curves and pretty good results. Ridge Regression with gradient descent is definitely more expensive, but is more suited for more diverse and complex data sets. The advantage of gradient descent is that the computation performed is lesser so it can be used to handle larger data. This turns out to be equal to performing the slightly heavier Ridge Regression on smaller data.

# Problem 5: Non-linear Regression




We ran it for two cases, one without regularization where **lambda = 0**, and another with regularization. Without regularization, for test data, we notice that MSE initially decreases by a huge margin, and reaches its minimum value when p equals 1. For values of p from 2 until 6, the MSE keeps on increasing. This is very typical for a case without regularization, as the data tends to experience over fitting, and tries to adjust itself to fit every point possible.

With Regularization (**lambda = 0.06**), however we see that the graph we get is much smoother. The initial MSE value is very high, and then it decreases again by a big margin at **P = 1**. For 2,3,4,5,6, it almost stays steady; it decreases by very small amounts. This is so because, with regularization, over-fitting of the data is avoided and we get a smooth and simple curve. We reach the minimum value at **P = 4** and then the value almost stays the same.

When it comes to training data, the optimal value for both the cases (with and without Regularization) is reached at **P = 6**.

MSE for Training Data without Regularization: **3866.883449**
MSE for Test Data without Regularization is: **3845.03473**

MSE for Training Data with Regularization: **3950.682335**
MSE for Test Data with Regularization: **3895.582668**

**MSE Values for Train and Test Data without Regularization**

| WITHOUT REGULARIZATION | | |
|---|---|---|
| P | TRAIN DATA | TEST DATA |
| 0 | 5650.710539 | 6286.404792 |
| 1 | 3930.915407 | 3845.03473 |
| 2 | 3911.839671 | 3907.128099 |
| 3 | 3911.188665 | 3887.975538 |
| 4 | 3885.473068 | 4443.327892 |
| 5 | 3885.407157 | 4554.830377 |
| 6 | 3866.883449 | 6833.459149 |

**MSE Values for Train and Test Data with Regularization**

| WITH REGULARIZATION | | |
|---|---|---|
| P | TRAIN DATA | TEST DATA |
| 0 | 5650.711907 | 6286.881967 |
| 1 | 3951.839124 | 3895.856464 |
| 2 | 3950.687312 | 3895.584056 |
| 3 | 3950.682532 | 3895.582716 |
| 4 | 3950.682337 | 3895.582668 |
| 5 | 3950.682335 | 3895.582669 |
| 6 | 3950.682335 | 3895.582669 |

# Problem 6: Interpreting Results

## Conclusion

| METHOD | TRAIN DATA | TEST DATA |
|---|---|---|
| **LINEAR REGRESSION WITHOUT INTERCEPT** | 19099.44684 | 106775.3616 |
| **LINEAR REGRESSION WITH INTERCEPT** | 2187.160295 | 3707.840182 |
| **RIDGE REGRESSION** | 2187.160295 | 2851.330213 |
| **NON LINEAR REGRESSSION WITHOUT REGULARIZATION** | 3866.883449 | 3845.03473 |
| **NON LINEAR REGRESSION WITH REGULARIZATION** | 3950.682335 | 3895.582668 |

Upon observing the MSE values calculated for all the suggested models we can conclude that, Ridge Regressions without Gradient should perform best provided they work with optimal lambda value. We also see a close similarity between the output of a Ridge Regression with Gradient so we can also recommend the same given the best lambda value. However, the MSE for Linear and Non Linear Regression MLE are significantly higher. Thus these methods are not recommended for the data given.

Despite the slight extra computations by Ridge Regression it works well on smaller data sets as opposed to the Ridge Regression with Gradient Descent. But when we work on larger data sets the result of Ridge Regression with Gradient Descent will be better than Ridge Regression and is recommended.