Programming Assignment 2 - Classification and Regression

Submitted By
Vidyadhar Reddy Annapureddy

LDA & QDA (Problem 1):

We have trained LDA and QDA using the given training data and then tested on given test data. The **accuracy reported by LDA is 97% and for QDA it is 96%**.

LDA separates classes using hyper-plane which works well in the case where the classes are actually linearly separable which is what the case here, as we are getting 97% accuracy for LDA. We can see the scatter plot (Fig 1) given below which clearly tells that the data is almost separable. So the good accuracy is expected from LDA.

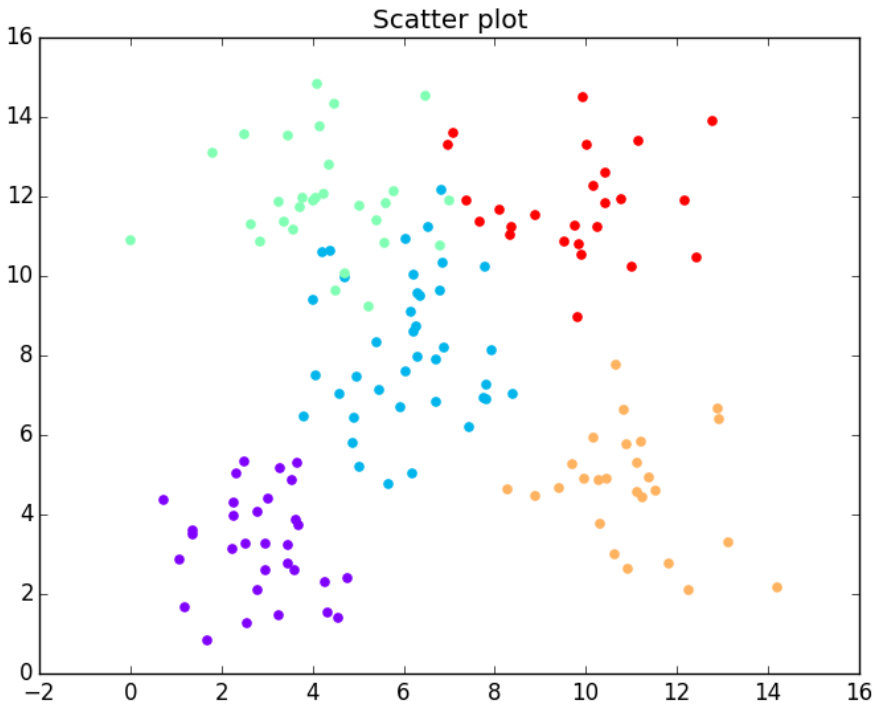

Figure1: scatter plot of test data

The LDA boundary plot is given in Fig 2 which displays the separation boundaries of 5 classes according to the test data.

QDA on the other hand is able to separate the classes by using a curve which can be represented quadratic fashion. This is better suitable when the data cannot be separated linearly but can be using quadratic curves. The decision boundary for QDA is plotted in the Fig 3. QDA will give better accuracy then LDA when the data is not linearly which will be the case in real world mostly.

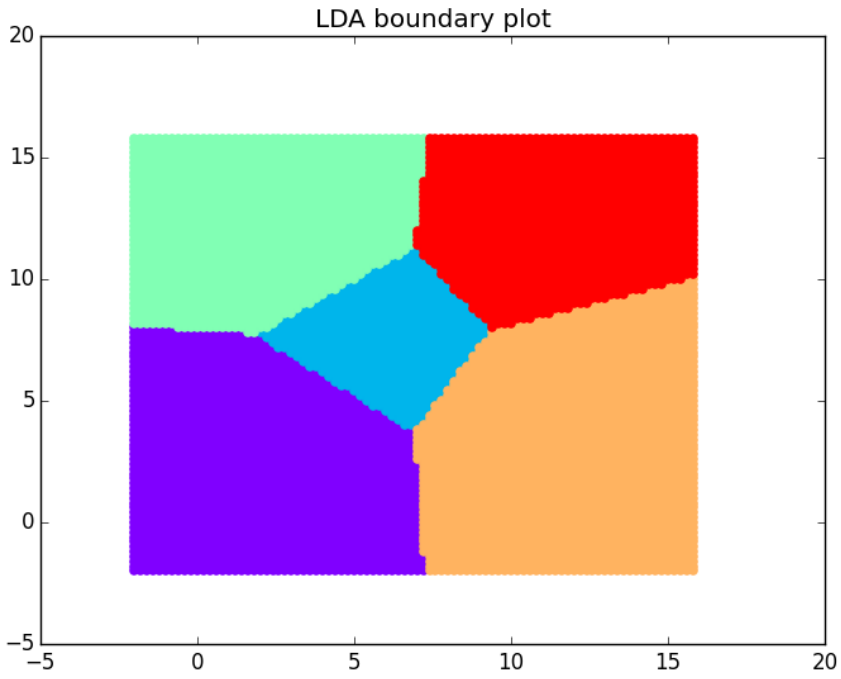

Figure2: LDA boundary plot

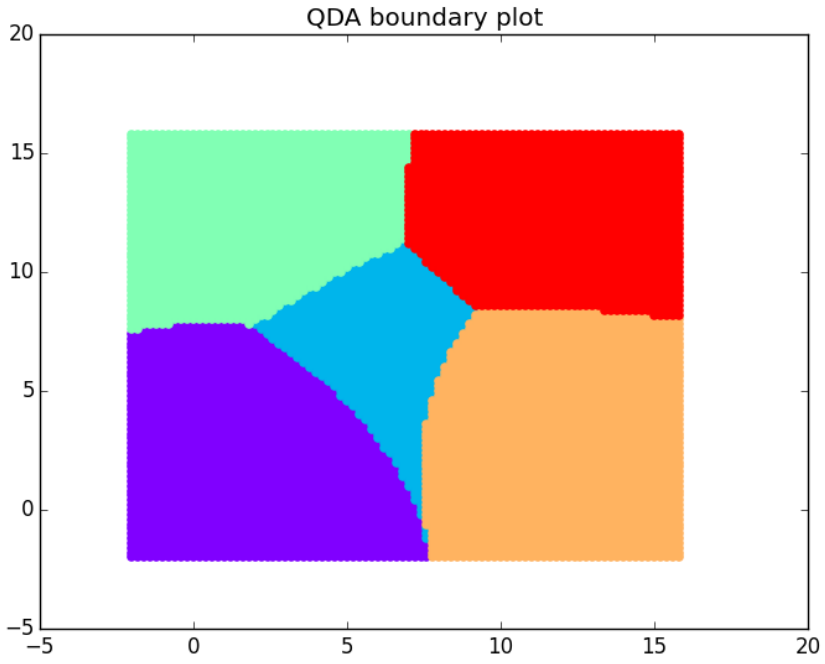

Figure3: QDA boundary plot

Linear Regression (Problem 2):

a. learnOLERegression

The function takes in as input matrices X and y with dimensions (Nxd) and (Nx1) respectively and returns a weight vector having dimensions (dx1).

$$w_{mle} = (X^T X)^{-1} X^T y$$

b. testOLERegression

The function takes in as input matrices Xtest, ytest and weight vector (generated by *learnOLERegression*) with dimensions (Nxd), (Nx1) and (dx1) respectively and returns the root mean square error (RMSE), a scalar for the data.

Findings:

- RMSE for train data without intercept - 8.884
- RMSE for train data with intercept - 3.006
- RMSE for test data without intercept - 23.106
- RMSE for test data with intercept - 4.306

Ridge Regression (Problem 3):

The weights calculated by *learnOLEregression* are smaller in magnitude as compared to the respective counterpart calculated by *learnRidgeRegression.*

$$J(w) = \frac{1}{2}\sqrt{\sum_{i=1}^{n}(y_i - w^T x_i)^2}$$

$$\frac{\partial J(w)}{\partial w} = \frac{1}{N}[w^T(X^T X) - y^T X] + w^T \lambda$$

i. Mean of weights calculated by *learnOLEregression* = 17.54
ii. Mean of weights calculated by *learnRidgeRegression* = 12406.42
iii. Mean of absolute value of weights calculated by *learnOLEregression* = 33.26
iv. Mean of absolute value of weights calculated by *learnRidgeRegression* = 73056.76

Findings:

- Error for train data without regularization – 3.01
- Error for train data with regularization - 3.18
- Error for test data without regularization - 4.3
- Error for test data with regularization – 3.8

The error on test data with regularization hits a minimum value of 3.78(approx.) at **λ** = 0.002 and increases thereafter, while on training data, the error continuously increases with increasing values of λ. **The optimal value of λ for the experiment is 0.0002.**

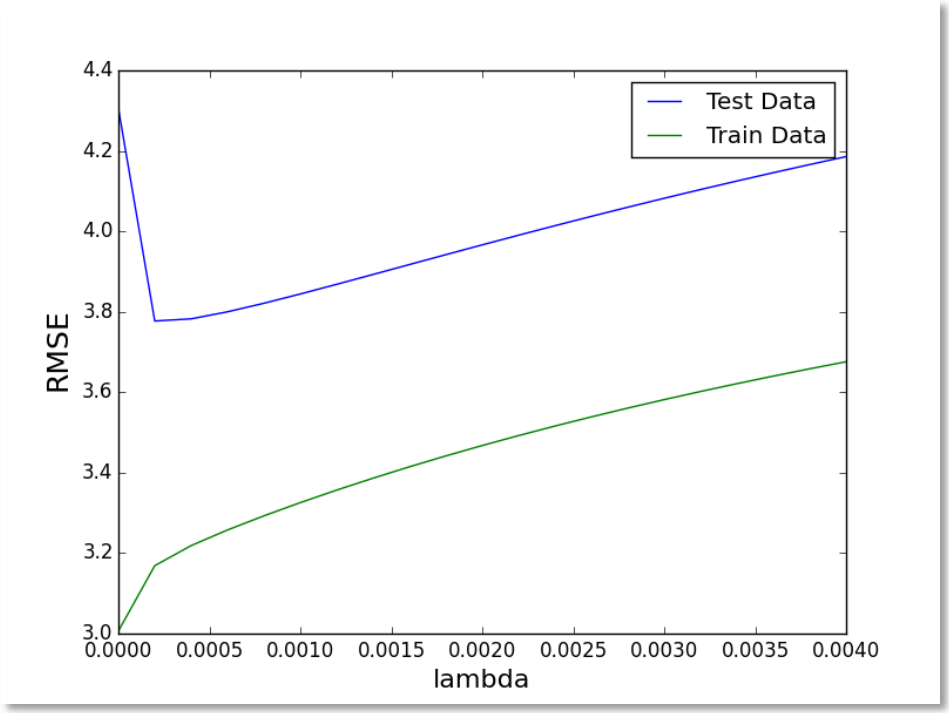

Figure4: Error calculated for train and test data for varying λ

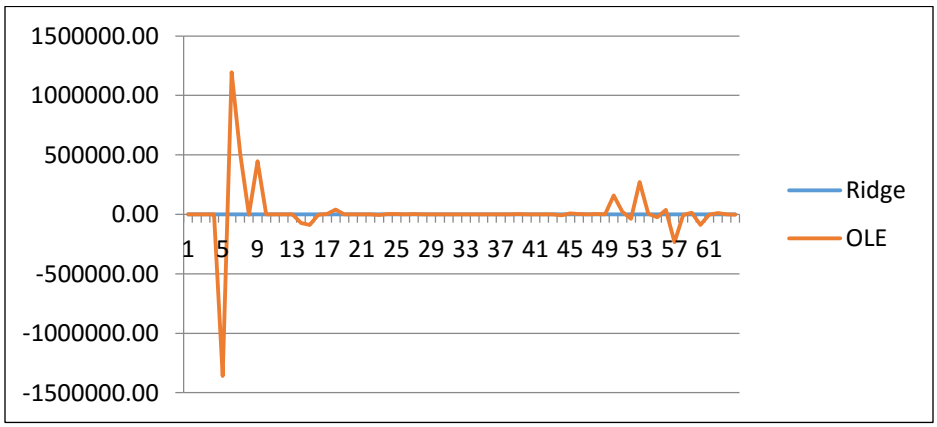

Figure5: True weights calculated by OLE and ridge regression

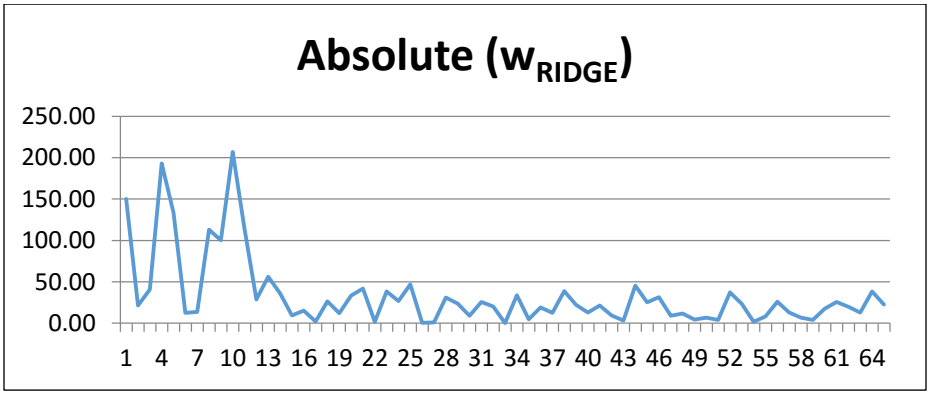

Figure6: absolute values of weights calculated by ridge regression

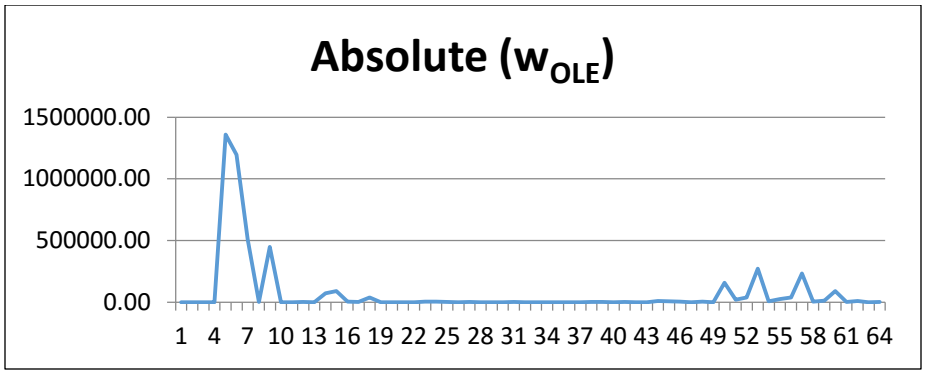

Figure7: absolute values of weights calculated by OLE regression

Gradient Descent for Ridge Regression (Problem 4):

The plot for error on train and test data obtained using gradient descent based learning with λ is shown in the graph (Fig 7). The error is high initially for the λ as 0. But after λ increases, the regularization parameters comes into picture and the error for test data decreased till the optimum value and then started to increase.

The values are almost similar to the values obtained in Problem 3 with some small changes in the values corresponding to λ before the optimum value (0.0002) which should be the case because both methods does the same thing using different techniques.

$$J(w) = \frac{1}{2N}[(y - Xw)^T * (y - Xw)] + \frac{1}{2}\lambda w^T w$$

$$\frac{d(J(w))}{dw} = \frac{1}{N}(-y^T x + w^T(x^T x)) + \lambda w^T$$

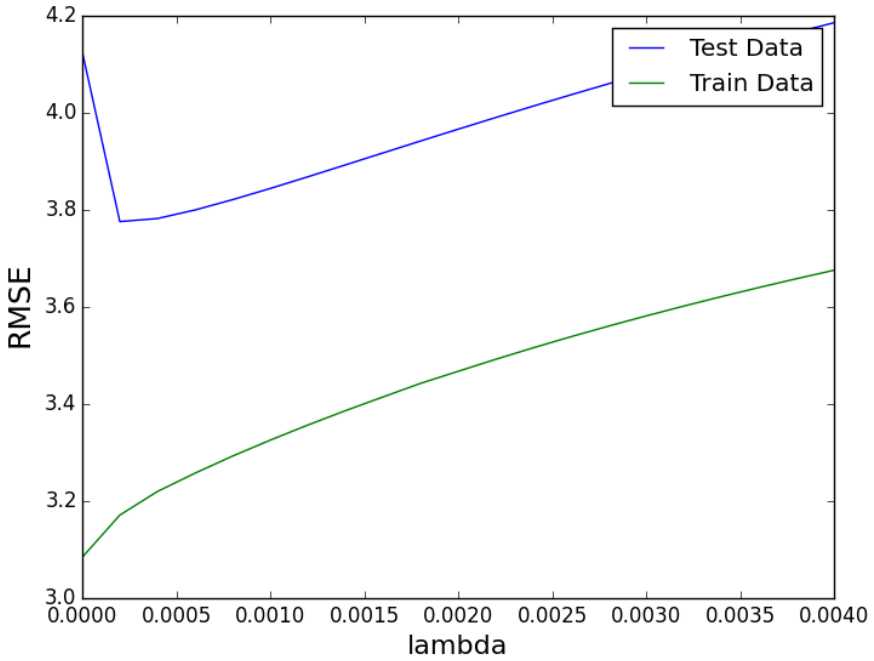

Figure7: RMSE vs λ for train and test data

Non-linear Regression (Problem 5):

We can see from Fig 8 that error for test data is higher than the train data which is expected for both optimal λ and λ=0.

For test data the error is high as compared to train data. For optimal λ the value of error becomes almost constant and it is unaffected by the non-linearity of train data. But for the λ zero, the error is increasing after power 3. So we can say that at optimal λ, the regularization term is able to handle the error and keeping it constant. So it is almost unaffected by the non-linearity of the input data. For train data the error values are almost constant for both λ optimal and zero. Without regularization, for test data, the optimal value of P is 1 while with regularization, it is 2. The optimal value of P for train data without regularization is 6, while it is 5 and 6 with regularization. But we can say it is almost same values for p 1 to 6.

RMSE Values	Test data		Train data	
	λ=0	λ=0.0002	λ=0	λ=0.0002
p=0	5.60642702	5.60659858	4.83218828	4.83218866
p=1	**4.38465206**	4.40623928	4.03031662	4.03759718
p=2	4.41991408	**4.40616766**	4.02052568	4.03690644
p=3	4.40906767	4.40616843	4.02019112	4.03690408
p=4	4.71345303	4.40616843	4.00695319	4.03690397
p=5	4.77222714	4.40616843	4.00691920	**4.03690396**
p=6	5.84527978	4.40616843	**3.99735628**	**4.03690396**

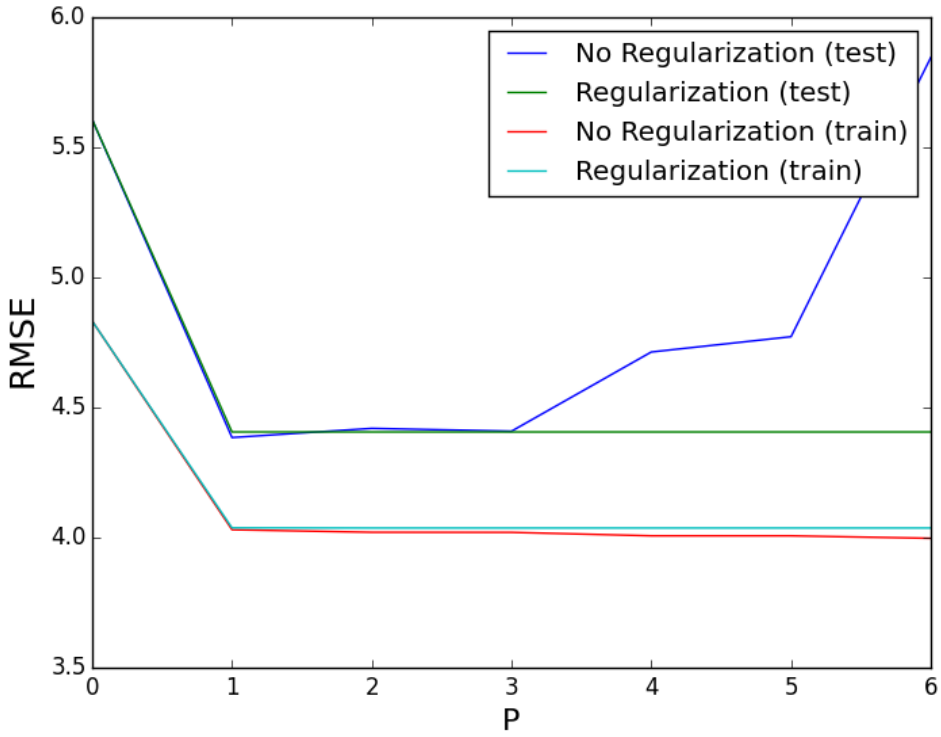

Figure8: RMSE vs P for train and test data

Interpreting Results (Problem 6):

Based on the experiments conducted above, the magnitude of error value calculated for test data with regularization, wherever applicable, appears to be minimum of the ridge regression (3.8) followed by gradient descent method for ridge regression (4), followed by ordinary least squares method (4.3) and yielding non-linear regression as the most erroneous method of all with an error value of 5.6.

The results coincide with our expectations of error values for the conventional ridge regression and ridge regression with gradient descent method coming to be very close. Although linear regression refers to any approach to model the relationship between one or more variables, ordinary least squares method is used to find the simple linear regression of a set of data.

We conclude the order of efficiency as (most efficient first):

1. Ridge Regression
2. Ridge Regression by Gradient Descent
3. Ordinary Least Squares
4. Non-Linear Regression

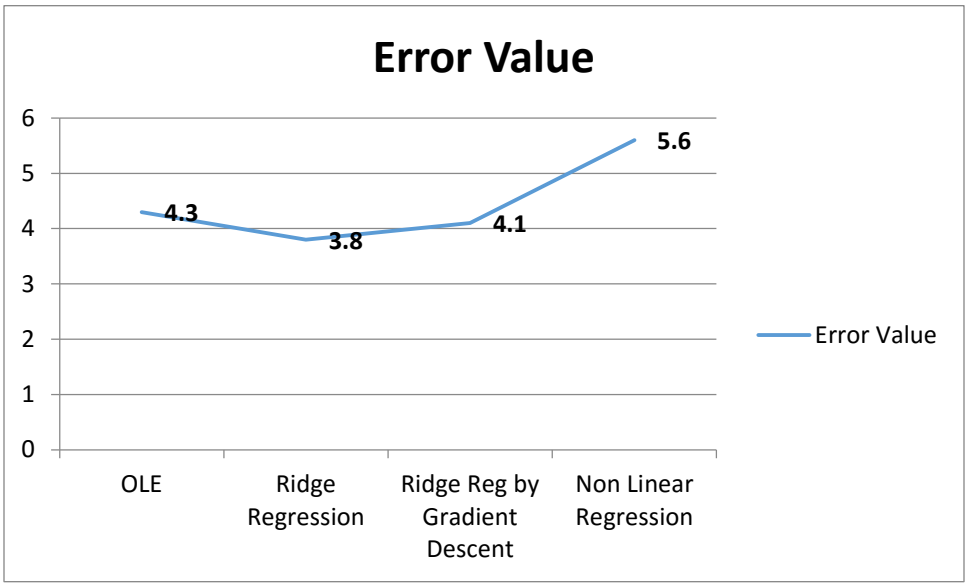

Figure9: Error values for test data for various regression methods