CSE574 Introduction to Machine Learning
Programming Assignment 1
Classification and Regression

Group 7

1. Karan Nisar (karankir)
2. Tejas Dhrangadharia (tejassha)

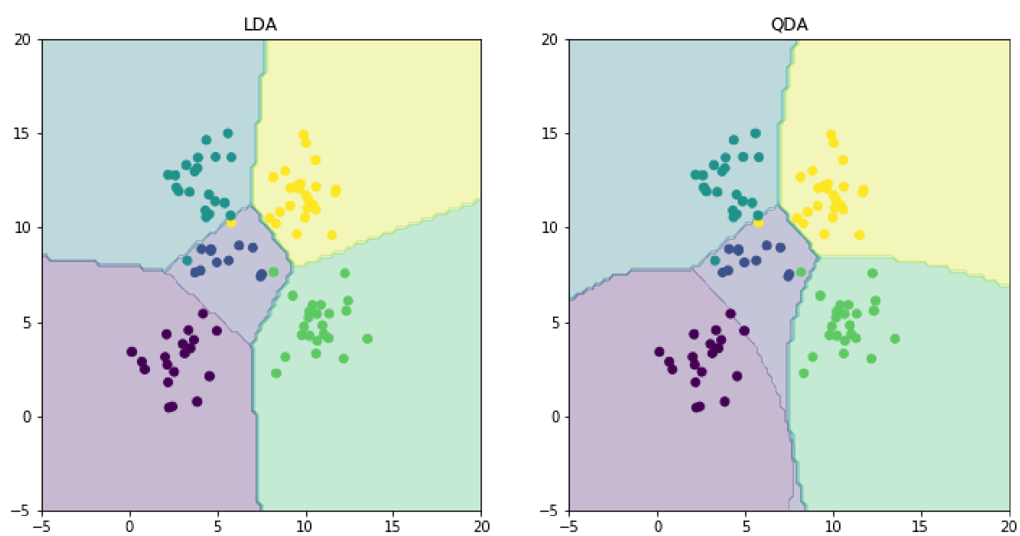
Problem 1: Experiment with Gaussian Discriminators

Implemented Linear Discriminant Analysis(LDA) and Quadratic Discriminant Analysis(QDA), we trained both the methods using the sample training data. Developed two functions `ldaTest` and `qdaTest` which returns the true labels for a given test data set and the accuracy using the labels for the test data set.

Below is the accuracy of LDA and QDA on the provided test data set,

Methods	Accuracy
LDA	97.0
QDA	96.0

And here is the plot of the discriminating boundary for LDA and QDA is,



As we can see in the plot that, there is a difference in discriminating boundary for LDA and QDA. The reason for that is, LDA does linear discriminating boundary between the points belonging to different classes. While on other hand, QDA learns a quadratic one. If the actual boundaries are linear, QDA may have a higher model bias. Here we have a limited data set. And if you divide your already sparse dataset into the constituent classes and compute the covariance matrix for each, that might be inaccurate. In this type of scenario(limited data set), it is better to simplify the entire process and use a common covariance matrix that's computed from the entire data set.

Problem 2: Experiment with Linear Regression

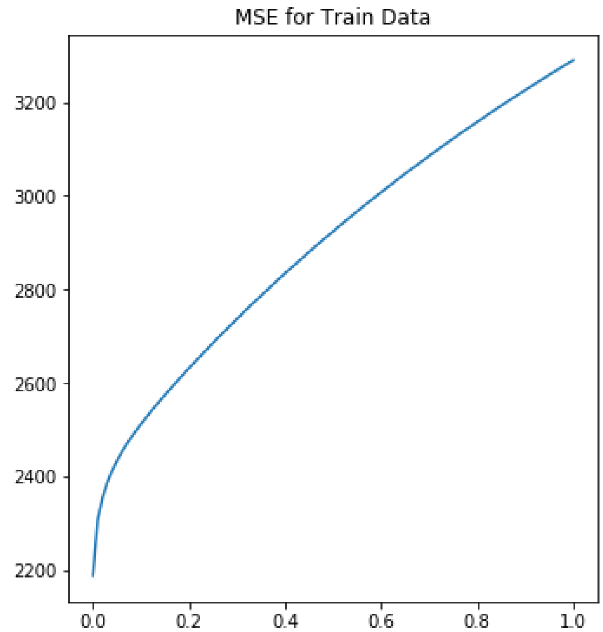
Developed two function `learnOLERegression` and `testOLERegression` for fitting the model and calculating MSE for linear regression respectively. Below is the MSE for test data set with Intercept and without intercept,

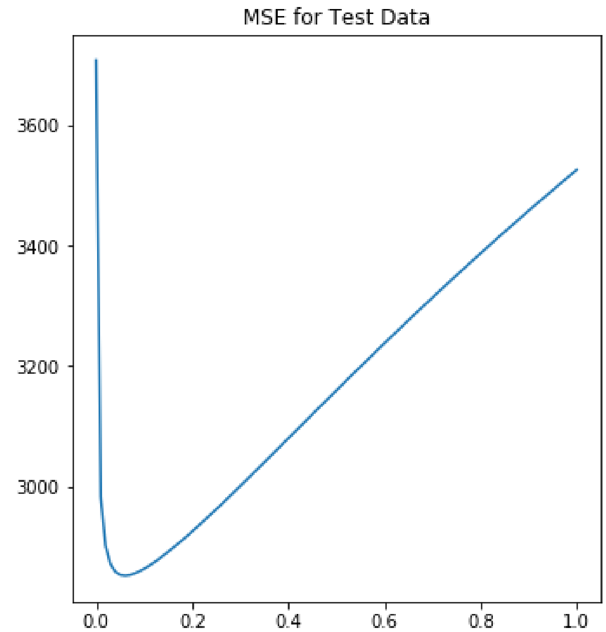
	MSE
MSE without intercept	106775.361555
MSE with intercept	3707.84018132

As we can see from the above result that, MSE with intercept has lower value compare to MSE without intercept. So, MSE with intercept is better in this case.

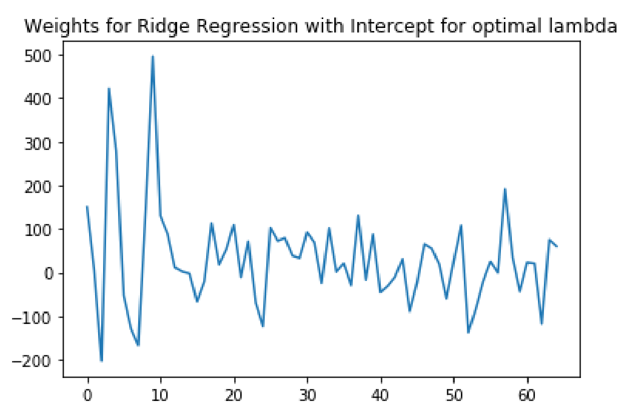
Problem 3: Experiment with Ridge Regression

Calculated the MSE for training and test data set using ridge regression parameters. Following is the plot for errors on train and test data set for different values of λ (λ from 0 to 1 in steps of 0.01).





We can see that for test data the MSE is quite high for λ ~ 0, it falls down drastically and increases linearly. This is because the model is underfitted.
Compared the relative magnitudes of weights learnt using OLE and weights learnt using ridge regression. And below is the comparison for that,



From the above comparison we can say that, weights for OLE have a much higher range (-80,000 to 80,000) compared to ridge regression (-200 to 500). Also, weights for OLE regression have higher deviation compare to Ridge regression, which explains better fit for ridge regression (MSE for ridge is lower than the MSE for OLE).
Comparison of two approaches:

	OLE - MSE	Ridge - MSE
Test data	3707.84018132	2851.33021344
Train data	2187.16029493	2187.16029493

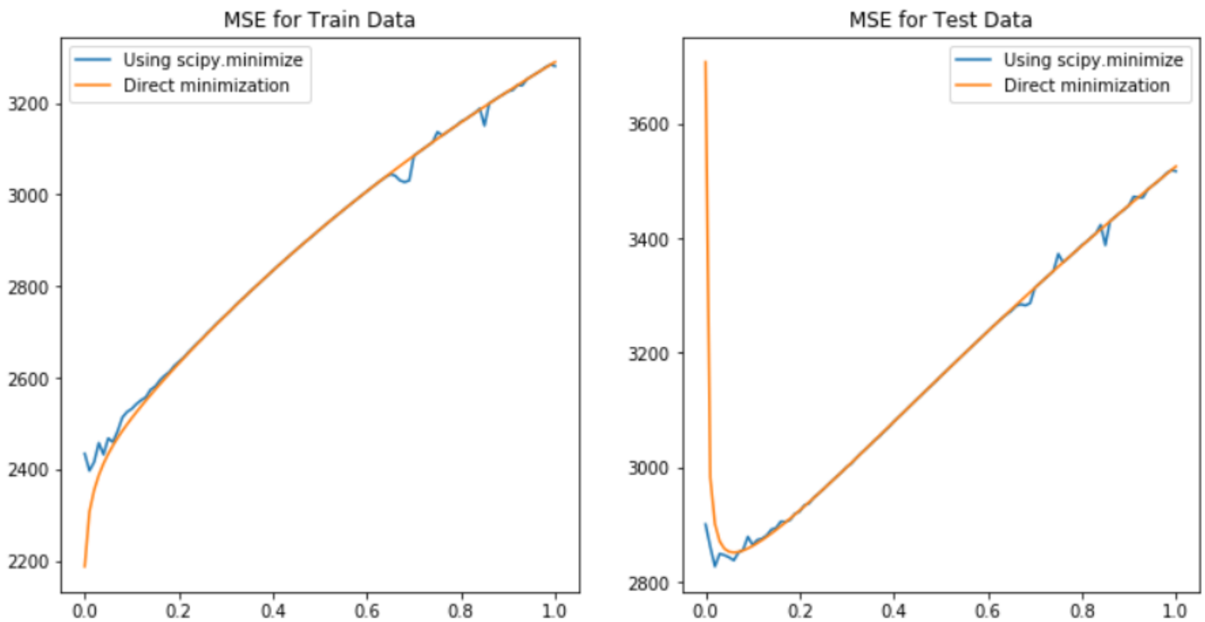
We can say from the above comparison that, MSE for test data using Ridge regression is lower than the OLE. So, Ridge regression is better than the OLE.
And below is the MSE for optimal lambda for training and test data using ridge regression parameters,

	Optimal Lambda	MSE
Test data	0.06	2851.33021344
Train data	0.0	2187.16029493

These are the optimal values of λ. The reason for that is, on these λ values we got minimum MSE for test and train data using Ridge regression.

Problem 4: Using Gradient Descent for Ridge Regression Learning

Below is the plot of errors on train and test data obtained using the gradient descent-based learning by varying the regularization parameter λ and comparing that with the errors obtained using Ridge regression.



From the above plots, we can say that both the plots look almost similar but the minimum value for MSE is changed from 2851.33021344 (Ridge Regression) to 2826.95234102 (Gradient Descent) for test data.
And for train data also it changed from 2187.16029493 to 2396.44210894.

Problem 5: Non-linear Regression

Compute the error on test and train data using λ = 0 and optimal value of λ. Below is the comparison for result of both the values of λ.

	No Regularization	Regularization	p value

Train data	5650.7105389	5650.71190703	0
	3930.91540732	3951.83912356	1
	3911.8396712	3950.68731238	2
	3911.18866493	3950.68253152	3
	3885.47306811	3950.6823368	4
	3885.4071574	3950.68233518	5
	3866.88344945	3950.68233514	6
Test data	6286.40479168	6286.88196694	0
	3845.03473017	3895.85646447	1
	3907.12809911	3895.58405594	2
	3887.97553824	3895.58271592	3
	4443.32789181	3895.58266828	4
	4554.83037743	3895.5826687	5
	6833.45914872	3895.58266872	6

From the above comparison, for train data we can say that, for $\lambda=0$ the optimal value of p is 6 and MSE is 3866.88344945 and for optimal λ the optimal value of p is 6 and MSE is 3950.68233514. For test data, for $\lambda = 0$ the optimal value of p is 1 and MSE is 3845.03473017 and for optimal λ the optimal value of p is 6 and MSE is 3895.58266872.

Problem 6: Interpreting Results

After computing training and test error using various approaches, we can say that, using Gradient Descent with optimal regularization parameter λ we can achieve minimum MSE on the test data without overfitting.