CSE574 Introduction to Machine Learning

Programming Assignment 1

Classification and Regression

Report

PROJECT GROUP 16

Rohit Balasayee

Vivek Nagaraju

Sriram Venkataramanan

PROBLEM 1: GAUSSIAN DISCRIMINATION

Train both methods using the sample training data (sample train). Report the accuracy of LDA and QDA on the provided test data set (sample test). Also, plot the discriminating boundary for linear and quadratic discriminators. The code to plot the boundaries is already provided in the base code. Explain why there is a difference in the two boundaries.

LDA

Linear Discriminant analysis is a straight forward method applying the generative approach for classification. LDA is based on the assumption,

- **Observation of each class are drawn from a normal distribution**
- **It assumes a common covariance matrix to all the classes present in a data set**

When this assumption holds, LDA approximates the Bayes classifier very closely and the discriminant function produces a linear decision boundary

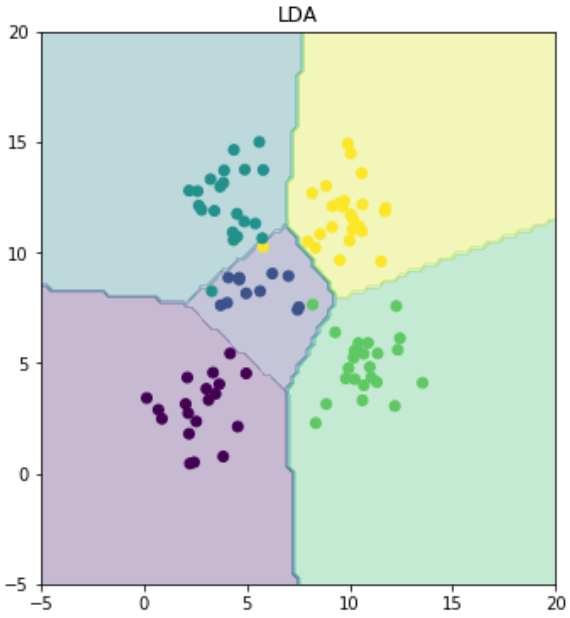
QDA

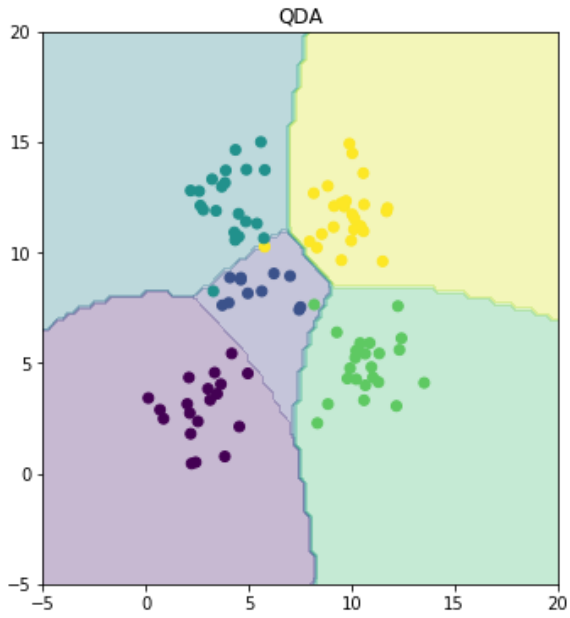
Quadratic discriminant analysis is similar to LDA, without assuming the classes share same covariance matrix, each class has its own covariance matrix. QDA is based on the assumption,

- **Observation of each class are drawn from a normal distribution**
- **It assumes that every class has its own covariance matrix.**

When this assumption holds, QDA approximates the Bayes classifier very closely and the discriminant function produces a linear decision boundary.

- Using the above concepts, the plots for LDA and QDA has been implemented in python and analysed.





ACCURACY OBTAINED IN CASE OF LDA = 97 %

ACCURACY OBTAINED IN CASE OF QDA = 97%

DIFFERENCES INFERRED FROM THE GRAPH

LINEAR DISCRIMANT ANALYSIS(LDA)	QUADRATIC DISCRIMINANT ANALYSIS(QDA)
From the figure of LDA, we can see the **boundaries are more linear** while classifying the data.	From the figure, we can see the **boundaries are more quadratic** while classifying the data.
The two assumptions initially explained such as **class specific mean vector** and **a common covariance matrix to all classes** in data set are implemented in code while training and testing the data set.	The two assumptions initially explained such as **class specific mean vector** and **each class has its own covariance matrix** are implemented in code while training and testing the data set.
The boundaries are linear as there **is no quadratic term** involved in the mathematic expression	The boundaries are quadratic since **quadratic terms** are involved in the mathematic expression.

INFERENCE RELATED TO ACCURACY

There is no difference in accuracy in terms of LDA and QDA in our case for the data given. In our case, the amount of data we have is **less and well grouped,** thus **the accuracy remains the same. Whereas when we have more complex and dense data, QDA tends to perform better than LDA.**

PROBLEM 2: LINEAR REGRESSION

Calculate and report the MSE for training and test data for two cases: first, without using an intercept (or bias) term, and second with using an intercept. Which one is better?

Test Data

MSE obtained without intercept	**106775.3614512301**
MSE obtained with intercept	**3707.840180960696**

Training Data

MSE obtained without intercept	**19099.446844570677**
MSE obtained with intercept	**3707.840180960696**

We can clearly see from the above results that using the intercept gives a lower MSE for both the test data and training data.

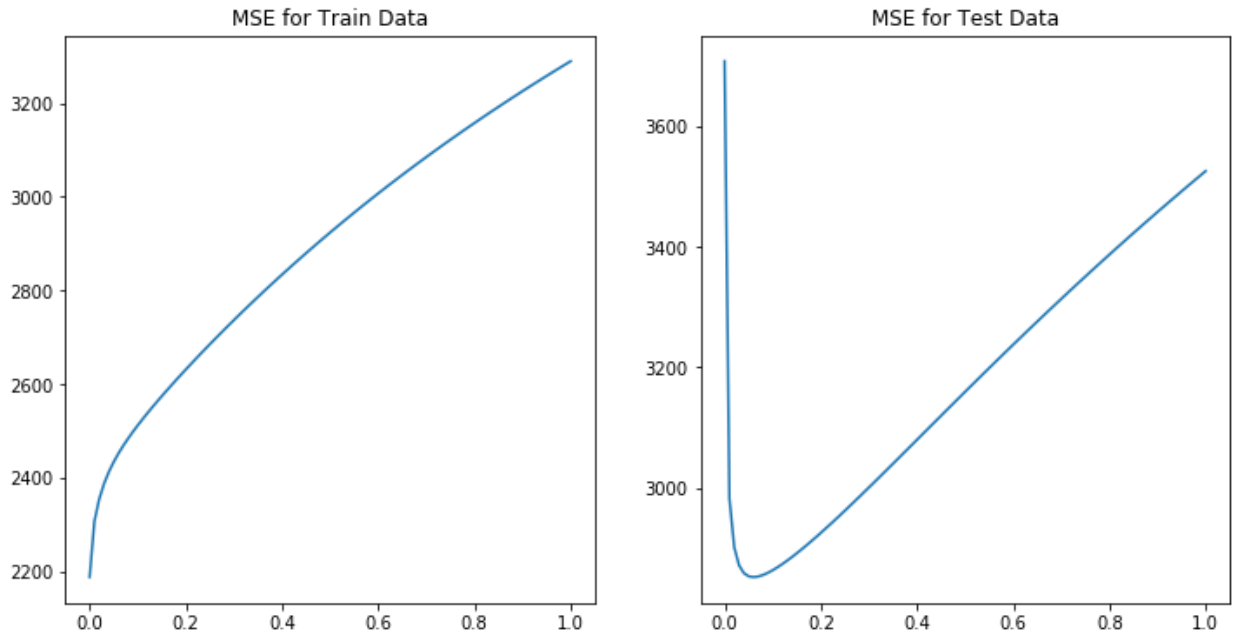
It is not a good when we fit in a model without an intercept, unless we can be sure that the linear approximation of the data passes through the origin, because in doing so we are introducing a bias which in turn leads to a higher MSE.

When intercept is not used the linear model need not pass through the origin and we can fit in the model closer to our data points which in turn lead to a smaller MSE.

Thus, the MSE obtained for training data is drastically lesser than that of the test data. The reason is that we build the model using the training data => the model will fit the training data more closely than the test data and hence the MSE will be lesser for the training data.

PROBLEM 3 : RIDGE REGRESSION

Calculate and report the MSE for training and test data using ridge regression parameters using the testOLERegression function that you implemented in Problem 2. Use data with intercept. Plot the errors on train and test data for different values of lambda. Vary lambda from 0 (no regularization) to 1 in steps of 0.01.Compare the relative magnitudes of weights learnt using OLE (Problem 2) and weights learnt using ridge regression. Compare the two approaches in terms of errors on train and test data. What is the optimal value for lambda and why?



The above graph shows the **MSE using Ridge Regression** for both the Train Data and Test Data. We analyse as we vary the lambda value from 0 to 1 by steps of 0.01 at each stage and plot the corresponding MSE for both the Training and Test Data.

In Linear Regression (Problem 2), The model may not exactly fit the data due to different outliers and bias in the training data.

Ridge Regression solves this problem by introducing lambda (Regularization Parameter) which will fit the nonlinear curve without overfitting the data.

The below Data shows the MSE for both the training and test data for various values of lambda

MSE (Training)	MSE(Test)	Lambda
[2187.16029493]	[3707.84018145]	0.0
[2306.83221793]	[2982.44611971]	0.01
[2354.07134393]	[2900.97358708]	0.02
[2386.7801631]	[2870.94158888]	0.03
[2412.119043]	[2858.00040957]	0.04
[2433.1744367]	[2852.66573517]	0.05
[2451.52849064]	[2851.33021344]	0.06

```
[2468.07755253] [2852.34999406] 0.07
[2483.36564653] [2854.87973918] 0.08
[2497.74025857] [2858.44442115] 0.09
[2511.43228199] [2862.75794143] 0.1
……
……..
[3264.61386081] [3498.57090566] 0.96
[3270.95717015] [3505.3183244] 0.97
[3277.26258207] [3512.03802854] 0.98
[3283.53048993] [3518.7300819] 0.99
[3289.7612813] [3525.39455263] 1.0
```

Thus, the MSE for Training and test data from the above observation is found as,

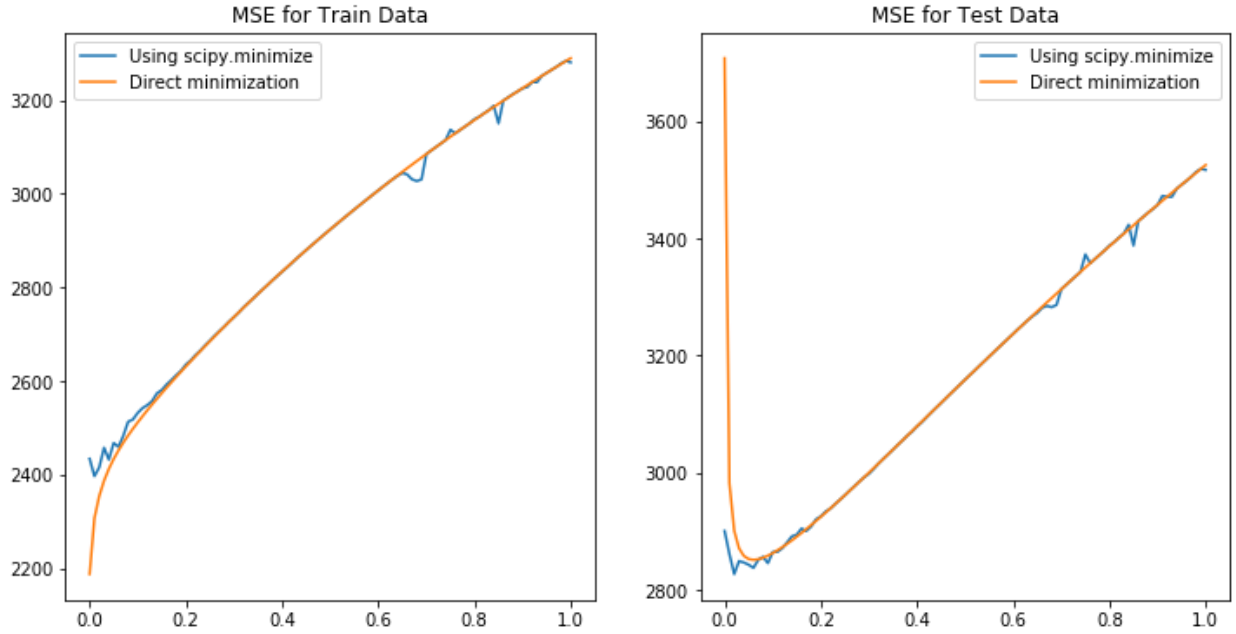
Training data	2451.52849064
Test data	2851.33021344

Thus, we can infer from the above graph, the MSE for the test data is the lowest when the **lambda value is 0.06** => we can conclude, it is the optimal value of Lambda.

When we compare the weight vector for the Linear and Ridge Regression, we find that the magnitude of the weight vector in the case of Ridge Regression is significantly smaller than that of the Linear Regression. This is because the Regularization parameter controls the weight vector and reduces it significantly.

PROBLEM 4: GRADIENT DESCENT FOR RIDGE REGRESSION

Plot the errors on train and test data obtained by using the gradient descent based learning by varying the regularization parameter lambda. Compare with the results obtained in Problem 3.



The above graph shows the MSE for both training and test data by using the gradient descent algorithm for Ridge Regression. We analyse as we vary the lambda value from 0 to 1 by steps of 0.01 at each stage

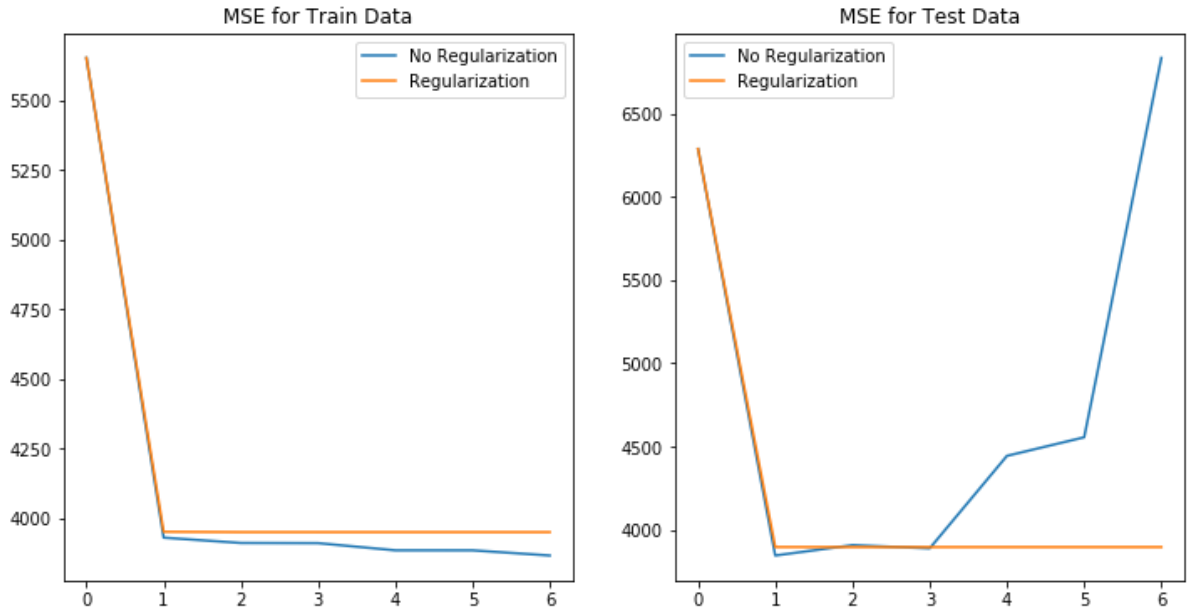
The graphs in both problem 3 and problem are almost similar.

Using Gradient Descent for Ridge Regression is more expensive but it is also suited for complex and diverse data. The advantage of Gradient Descent is that the computation performed is lesser which means it can handle larger data.
By using gradient descent, the computation becomes simple because we will be differentiating the cost function with the summation and will not be converting the equation to matrix to get the weight vector.

PROBLEM 5: Non Linear Regression

Using the **lambda**= 0 and the optimal value of **lambda** found in Problem 3, train ridge regression weights using the non-linear mapping of the data. Vary p from 0 to 6. Note that p = 0 means using a horizontal line as the regression line, p = 1 is the same as linear ridge regression. Compute the errors on train and test data. Comparee the results for both values of **lambda**. What is the optimal value of p in terms of test error in each setting? Plot the curve for the optimal value of p for both values of _ and compare.

The Graph obtained for the problem is:



INFERENCE FROM THE GRAPH

WITHOUT REGULARISATION AND TRAINING DATA (When lambda = 0)

The value of MSE decreases drastically initially till value of p reaches 1. After p reaches 1, the MSE decreases trivially till the value of p reaches from 1 to 6.

WITHOUT REGULARISATION AND TESTING DATA (When lambda = 0)

The value of MSE decreases drastically initially till value of p reaches 1. After p reaches 1, from p = 2 till p=6, the MSE value keeps on increasing. This is due to the concept of overfitting as the data tries to fit every point and it tries to adjust every single point.

WITH REGULARISATION AND TRAINING DATA (When lambda = 0.06)

The value of MSE decreases drastically initially till value of p reaches 1. After p reaches 1, the MSE becomes constant and forms almost a straight line till the value of p reaches 6

WITH REGULARISATION AND TESTING DATA (When lambda = 0.06)

The value of MSE decreases drastically initially till value of p reaches 1. After p reaches 1, from p = 2 till p=6, the MSE value becomes constant unlike the case of testing data without regularisation. It does not try to fit every point and remains steady almost with respect to value of p.

OVERALL OBSERVATION

WITHOUT REGULARISATION	**WITH REGULARISATION**
The graph is **not smoother**	The graph is **smoother**
Overfitting happens.	**Overfitting is prevented**.
The MSE reaches minimum value **when p =6 in case of Training data.**	The MSE reaches minimum value **when p =6 in case of training data**
The MSE reaches minimum value **when p =2 in case of Testing data.**	The MSE reaches minimum value **when p =4 in case of testing data**

MSE VALUES OBTAINED FOR WITHOUT REGULARISATION

P	**TRAINING DATA**	**TESTING DATA**
0	5650.710538897617	6286.404791680896
1	3930.915407315901	**3845.0347301734146**
2	3911.8396712049557	3907.128099107936
3	3911.1886649314497	3887.9755382360145
4	3885.4730681122714	4443.327891813363
5	3885.40715739708	4554.830377434541
6	**3866.8834494460493**	6833.459148718973

MSE VALUES OBTAINED FOR WITH REGULARISATION

P	**TRAINING DATA**	**TESTING DATA**
0	5650.711907032115	6286.881966941448
1	3951.839123560106	3895.8564644739627
2	3950.6873123755195	3895.5840559389176
3	3950.6825315187125	3895.5827159230994
4	3950.682336795369	**3895.582668283526**
5	3950.68233517702	3895.5826687044228
6	**3950.6823351427824**	3895.582668719096

Thus from the above tabularisation, we can observe the MSE values for training and testing data for with and without regularisation.

1) **WITHOUT REGULARISATION**

MSE FOR **TRAINING** DATA	**3866.8834494460493**
MSE FOR **TESTING** DATA	**3845.0347301734146**

2) WITH REGULARISATION

MSE FOR **TRAINING** DATA	**3950.6823351427824**
MSE FOR **TESTING** DATA	**3895.582668283526**

Problem 6: Interpreting results

Using the results obtained for previous 4 problems; make final recommendations for anyone using regression for predicting diabetes level using the input features.

The MSE values obtained in cases of various methods are tabulated as following.

	LINEAR REGRESSION WITHOUT INTERCEPT	**LINEAR REGRESSION WITH INTERCEPT**	**RIDGE REGRESSION**	**NON LINEAR REGRESSION WITHOUT REGULARISATION**	**NON LINEAR REGRESSION WITH REGULARISATION**
Training data	19099.446844570677	3707.840180960696	2451.52849064	3866.8834494460493	3950.6823351427824
Testing data	106775.3614512301	3707.840180960696	2851.33021344	3845.0347301734146	3895.582668283526

FINAL CONCLUSION:

Lower values of MSE determines better model since it means the deviation from the true label is the least which in turn gives rise to higher accuracy of prediction.

After analysing the MSE values as calculated above, we can conclude as following.

- MSE for **Linear and Nonlinear regression** are higher and hence **are not recommended**.
- **Ridge Regression** gives the best possible result if they are used with an optimal lambda value**.** Ridge regression **works well with small amount of data**.
- **Overall Ridge regression using gradient descendent works better for model having large data sets which are complex and diverse. This approach leads to a better efficiency, accuracy and stability with such kinds of data.**