CSE 574: Programming Assignment 1
Classification and Regression

Team:

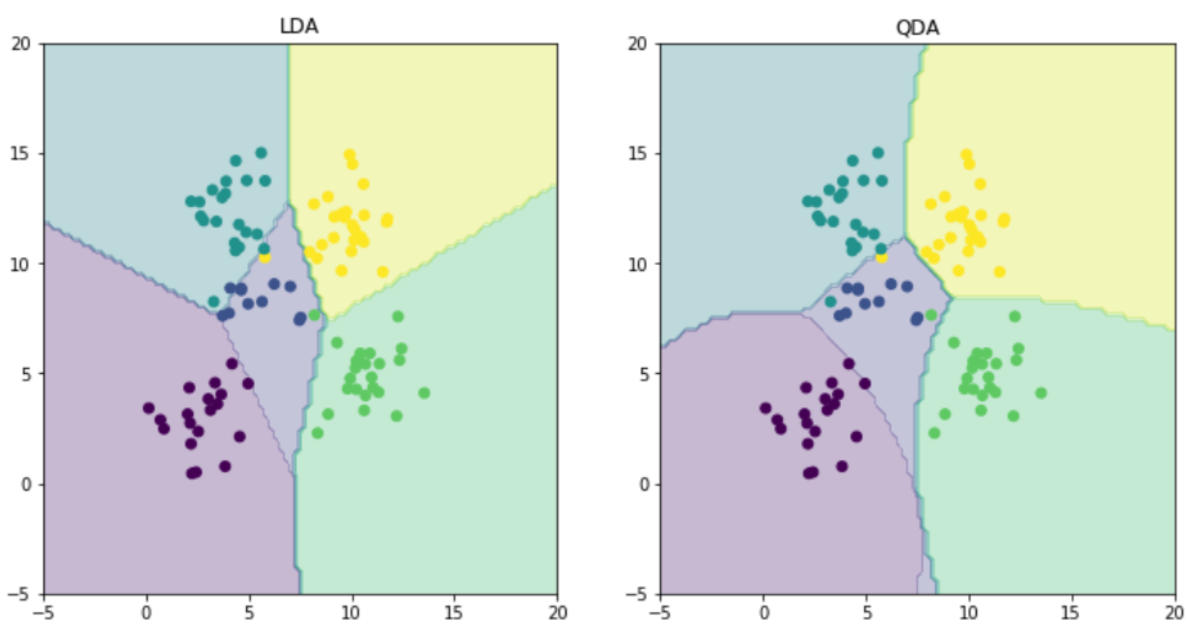
Abhinav Kumar 50247660
Raman Keswani 50246297

Problem 1: Experiment with Gaussian Discriminators

Accuracy obtained on the provided test data set (sample test):

```
LDA Accuracy = 96%
QDA Accuracy = 96%
```

Plot showing the discriminating boundaries:



Reason for difference in two boundaries: LDA assumes the covariance of variables to be common for all output classes (**k**). However, QDA assumes that each of the **k** classes have its own covariance matrix. Because of this reason, QDA can provide non-linear boundaries and LDA can provide linear boundaries. (There is no quadratic term in LDA)

Problem 2: Linear Regression

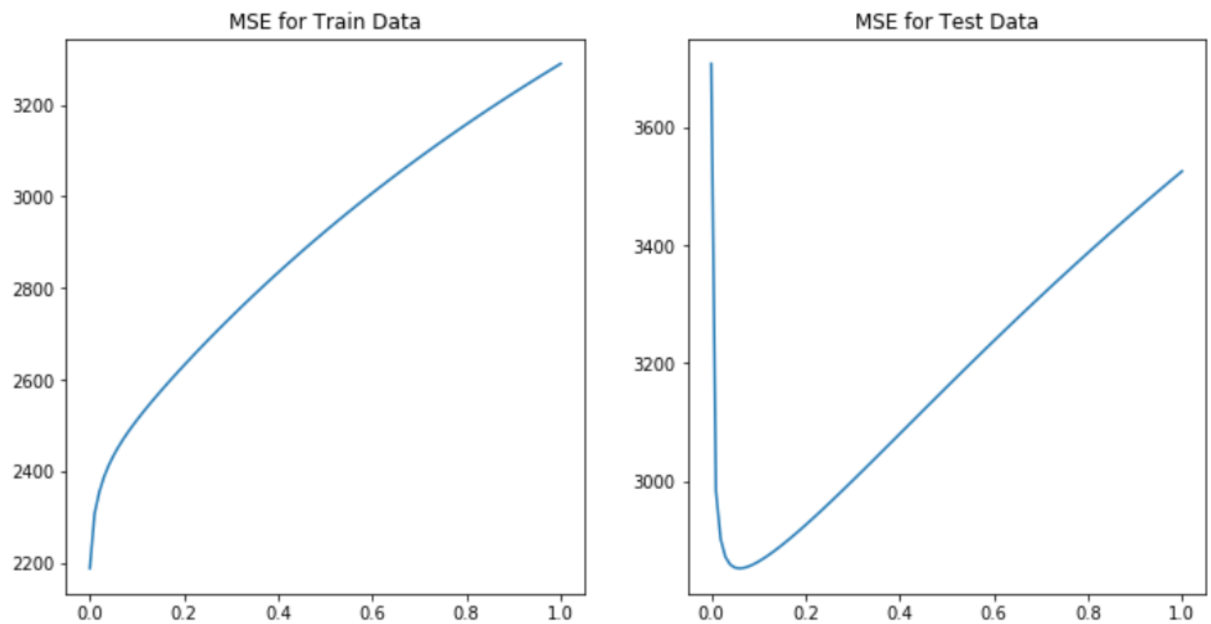
```
MSE without intercept [[106775.36155355]]
MSE with intercept [[3707.84018128]]
```

Clearly, MSE with intercept data is better because of the added bias. With the bias,we can augment our training data by an intercept so that the line fits the data better rather than the line always passing through the origin. In 2-D space, the line will always categorize the classes better if a bias is added.

Problem 3: Ridge Regression

MSE for Training Data: 2468
MSE for Test Data: 2852

The errors on train and test data for different values of lambda:



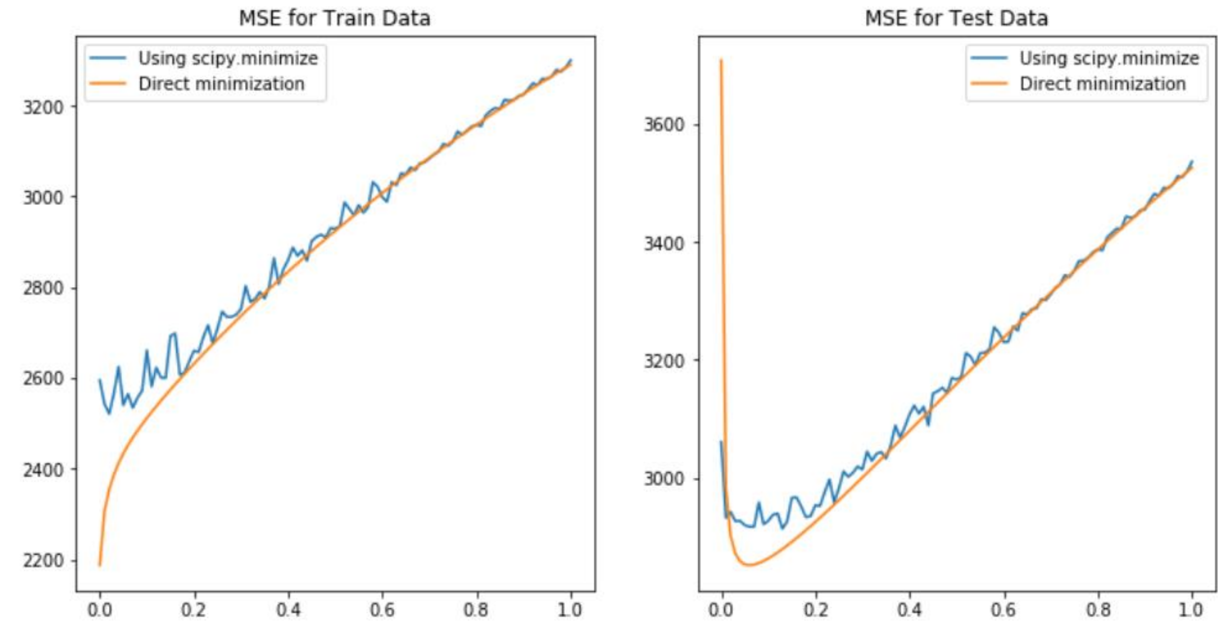
We can observe that weights learned with OLE regression is significantly larger as it tries to minimize the error, whereas weights learned with Ridge on the other hand are smaller as it tries to prevent overfitting of training data. (It tries to ensure the contribution from all the weights are even)

```
Magnitude of weights with OLE = 124531.52652675896
Magnitude of weights with Ridge = 926.5390559158366
```

The optimal value of lambda is 0.07 because it is this value where the MSE for the test data goes from descending to ascending curve. (forming a Parabola)

Problem 4: Gradient Descent for Ridge Regression Learning

Errors on train and test data obtained by using the gradient descent based varying lambda:



Both ridge (Problem 3) and gradient descent approaches provide similar value of lambda for minimum error. But the curve for gradient approach is not smooth as it converges to global minima step by step, and hence fluctuates a bit after each update (especially when it is about to converge)

Problem 5: Non-linear Regression

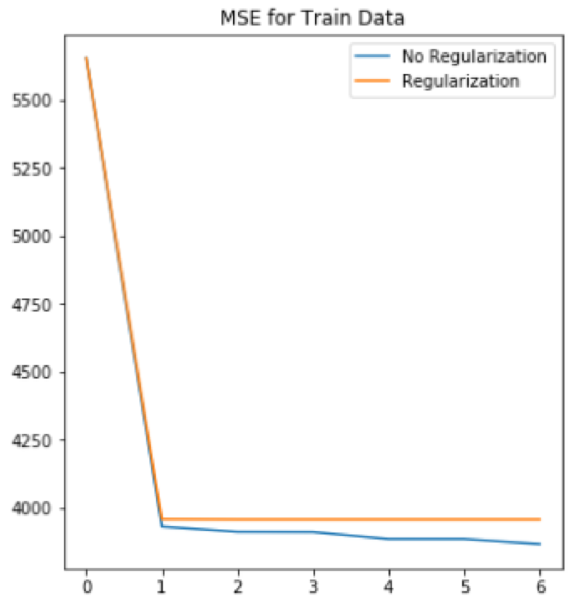
Errors calculated:

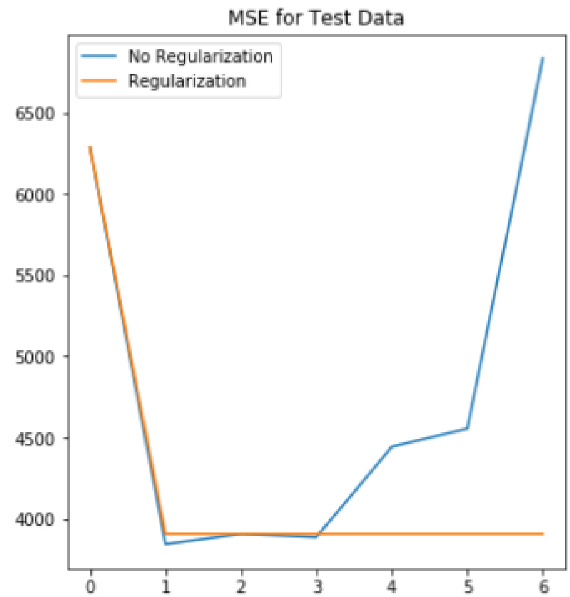
p	Train Data		Test Data	
	Lambda = 0	**Lambda = 0.07**	**Lambda = 0**	**Lambda = 0.07**
0	5650.7105389	5650.71240093	6286.40479168	6286.96173909
1	3930.91540732	3958.37646493	3845.03473017	3907.78832771
2	3911.8396712	3957.34383476	3907.12809911	3907.4243161
3	3911.18866493	3957.33885299	3887.97553824	3907.42092229
4	3885.47306811	3957.33867626	4443.32789181	3907.42084368
5	3885.4071574	3957.33867472	4554.83037743	3907.42084345
6	3866.88344945	3957.33867469	6833.45914872	3907.42084346

Optimal Value of **p** for each setting: (*after rounding off till 1 decimal place)

	Train Data		**Test Data**	
	Lambda = 0 **No Regularization**	**Lambda = 0.07** **Regularization**	**Lambda = 0** **No Regularization**	**Lambda = 0.07** **Regularization**
p	6	1/2/3/4/5/6	1	2/3/4/5/6

Error Comparison Plot for different p values:





Problem 6: Interpreting Results

We can observe that among all of the above approaches, we get minimum error for test data with Ridge Regression.
So we recommend ridge regression for anyone using regression for predicting diabetes level using the input features.
Further, metric should be the lambda for which we get minimum error (with Gradient Descent Approach). Because of the practical difficulties in calculating inverses, such as for singular matrix and computation effort required in general.