ML Homework – 1

Participants:

Keshav Narayan UBIT: 50610509

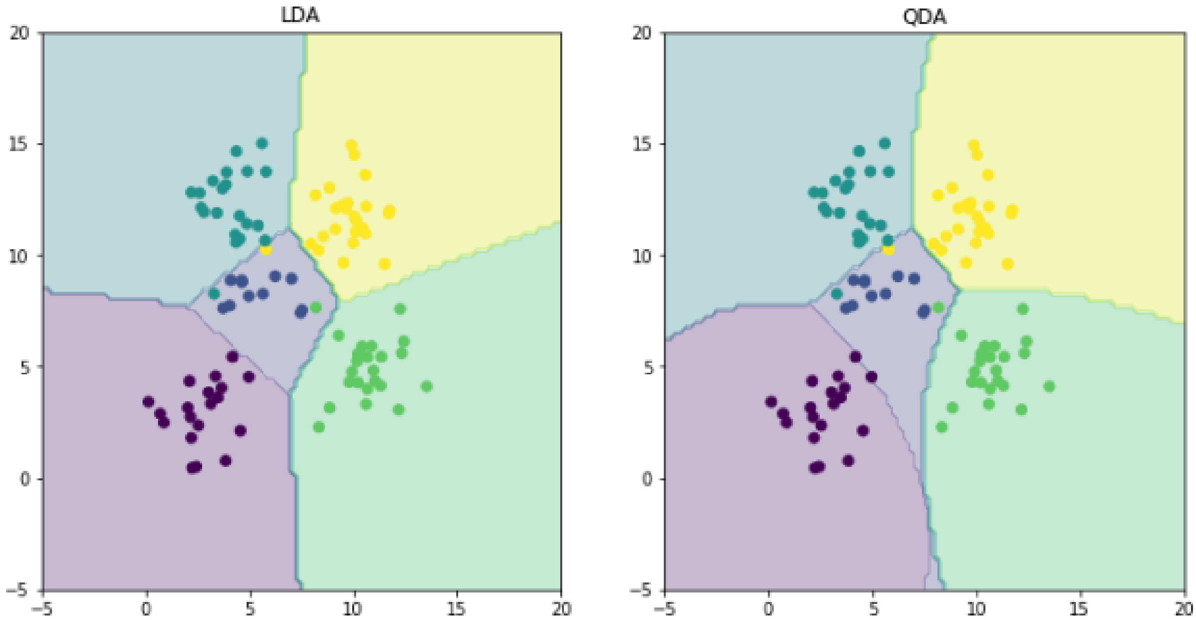
Pramila Yadav UBIT: 50613803

Hari Chandan Gooda UBIT: 50614165

Problem 1:

```
In [77]: runfile('C:/Users/chand/Downloads/Documents/UB/Intro to Machine Learning/Assignment1/Assignment1/basecode/script.py',
wdir='C:/Users/chand/Downloads/Documents/UB/Intro to Machine Learning/Assignment1/Assignment1/basecode')
LDA Accuracy = 0.97
QDA Accuracy = 0.96

In [78]:
```



From the graph and accuracy, we find out that LDA and QDA are very accurate for sample.pickle dataset. Both successfully categorized the test set into five classes. LDA slightly outperformed QDA in this case.

Problem 2:

```
In [78]: runfile('C:/Users/chand/Downloads/Documents/UB/Intro to Machine Learning/Assignment1/Assignment1/basecode/script.py',
wdir='C:/Users/chand/Downloads/Documents/UB/Intro to Machine Learning/Assignment1/Assignment1/basecode')
MSE without intercept [106775.36145629]
MSE with intercept [3707.8401804]
MSE without intercept for training data [19099.44684457]
MSE with intercept for training data [2187.16029493]

In [79]:
```

From the above output, we observe that when bias is included, MSE reduces drastically. From this, we conclude that biases are required to fit the model more accurately, because if the model didn't have a bias, the line which fits the data points would always pass through the origin. Since line could only rotate, it wouldn't fit the input well.

Problem 3:

After calculating MSE (training and testing) using Ridge Regression with the help of the testOLERegression function, we plotted the MSE data against λ values from 0 to 1 with each step being 0.01.

MSE for Train Data

The MSE value increases when lambda increases, so the least value of MSE is when λ =0. There is no regularization as λ =0.

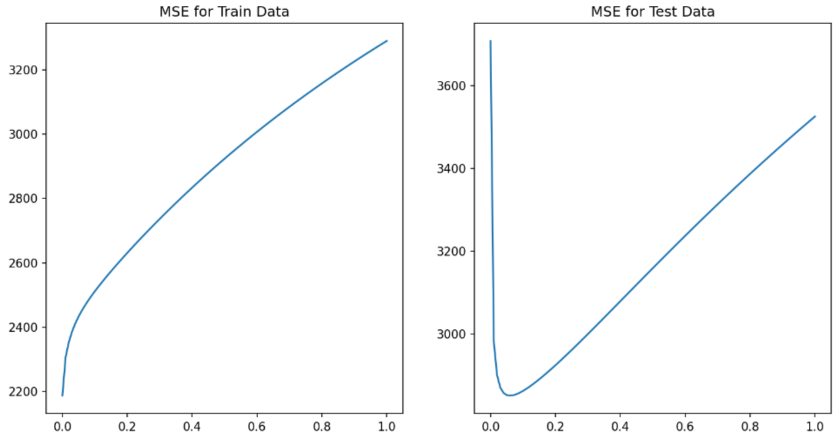
MSE for Test Data

The lowest MSE value is when λ=0.06. This is the best fit as MSE starts increasing again after this point.

lambda	mses3_train	mses3
0	2187.16029	3707.840182
0.01	2306.83222	2982.44612
0.02	2354.07134	2900.973587
0.03	2386.78016	2870.941589
0.04	2412.11904	2858.00041
0.05	2433.17444	2852.665735
0.06	2451.52849	2851.330213
0.07	2468.07755	2852.349994
0.08	2483.36565	2854.879739
0.09	2497.74026	2858.444421
0.1	2511.43228	2862.757941
0.11	2524.60004	2867.637909
0.12	2537.3549	2872.962283
0.13	2549.77689	2878.645869
0.14	2561.92453	2884.626914
0.15	2573.84129	2890.85911
0.16	2585.55987	2897.306659
0.17	2597.10519	2903.941126
0.18	2608.4964	2910.739372
0.19	2619.74839	2917.682164
0.2	2630.87282	2924.753222
0.21	2641.87895	2931.938544
0.22	2652.77413	2939.22593

MSE_test value when λ is 0.06 = 2851.33021344

MSE_train value when λ is 0.06 = 2451.52849064



Comparison between Ridge and OLE errors

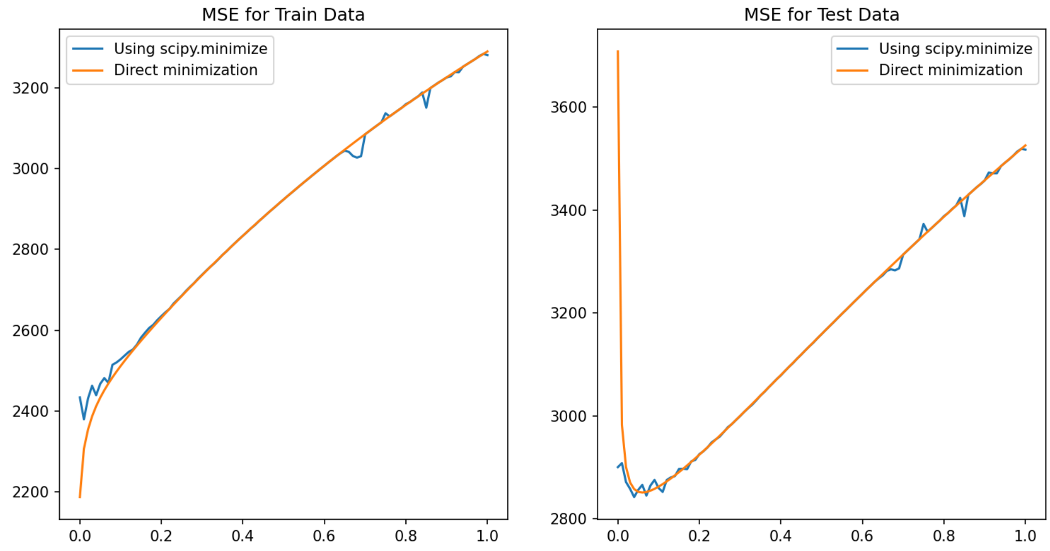
We will compare the "with intercept" data. In Ridge, we get MSE(test) = 2851.33021344 but in OLE, we get MSE(test)=3707.84018171. Hence, we ger lower MSE in ridge regression.

Comparison between Ridge and OLE weights

By looking at the values we can see that Ridge regression weights are lower than OLE weights due to regularization (Both when λ=0.06).

OLE Regression weights when Lambda = 0.06	Ridge Regression weights when Lambda = 0.06
[148.154876]	[150.45959807]
[1.27485208]	[4.80776899]
[-293.38352237]	[-202.90611468]
[414.72544855]	[421.7194576]
[272.08913426]	[279.45107288]
[-86639.45688865]	[-52.29708233]
[75914.46781676]	[-128.59418907]
[32341.62273242]	[-167.50057028]
[221.10121473]	[145.74068096]
[29299.55111699]	[496.30604123]
[125.23036028]	[129.94845775]
[94.41108334]	[88.30438076]
[-93.8628632]	[11.29067689]
[-33.72828002]	[1.88532531]
[3353.19773414]	[-2.58364157]
[-621.09628889]	[-66.89445481]
[791.73653574]	[-20.61939955]
[1767.7603891]	[113.39301454]
[4191.67404957]	[17.99086827]
[119.4381209]	[52.50235963]
[76.61034007]	[109.68765513]
[-15.2001293]	[-10.72779629]
[82.24245937]	[71.67974829]

Problem 4:



MSE for train and test data against varying λ using gradient descent on Ridge Regression is given above.

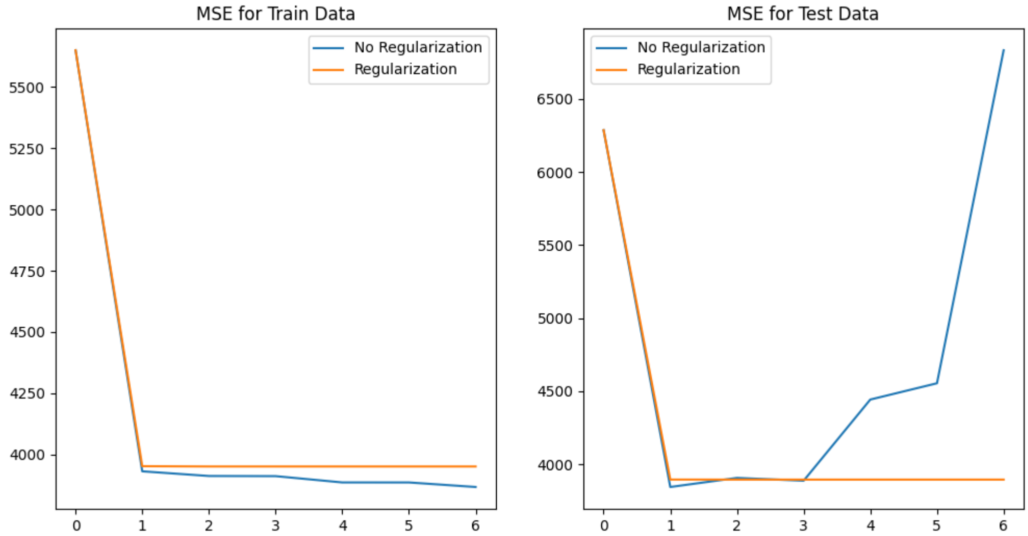
Min(MSE without Gradient Descent) = 2851.33021344 for λ = 0.06

Min(MSE with Gradient Descent) = 2842.456437 for λ = 0.04

Hence, we can conclude that MSE is lower for gradient descent learning.

lambda	mses4_train	mses4
0	2433.667834	2900.545156
0.01	2379.454361	2908.438739
0.02	2430.933664	2871.666424
0.03	2462.605706	2858.572312
0.04	2439.163567	2842.456437
0.05	2467.40446	2856.431306
0.06	2481.550497	2866.103357
0.07	2469.692406	2845.372184
0.08	2514.445753	2864.775963
0.09	2520.335126	2875.80759
0.1	2528.140329	2860.071933
0.11	2537.365924	2852.435671
0.12	2546.451525	2875.665845
0.13	2552.222141	2880.672099
0.14	2564.411427	2883.130187
0.15	2581.612513	2897.027108
0.16	2593.905431	2897.354177
0.17	2605.502896	2896.69692
0.18	2613.332189	2911.76713
0.19	2624.971091	2914.557928
0.2	2634.951051	2925.868442
0.21	2644.368691	2931.370803

Problem 5:



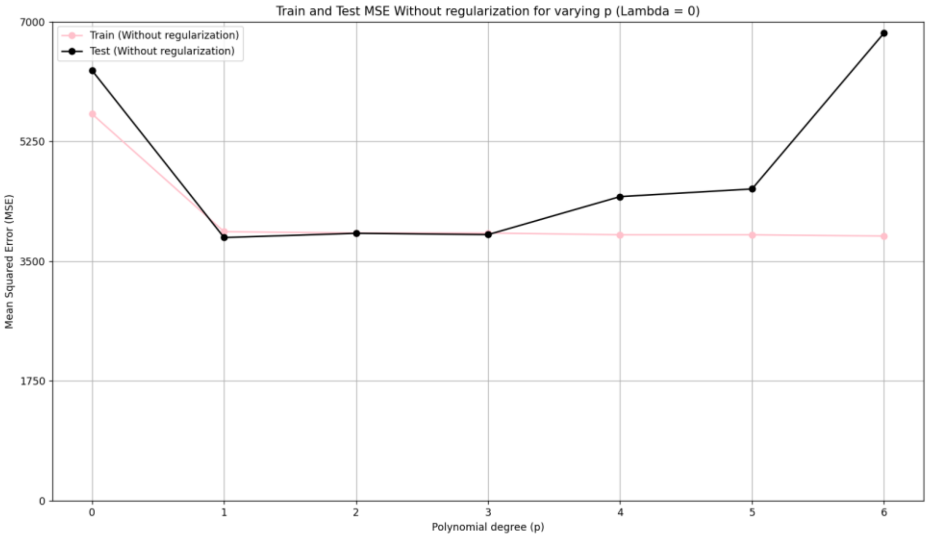
Above two graphs represent the result of mean squared error (MSE) of train and test data model with regularization and without regularization. In ridge regression, λ (lambda) controls the strength of the regularization. If λ is zero it will be a case of no regularization and larger values of λ will have stronger regularization.

In the Train data graph, which is on left side, it is clearly visible that when degree of polynomial(p) is 1, both the blue and orange line have close to similar error. **Hence, P=1 is optimal value**. After p = 1, regularization is straight line across the x axis whereas for no regularization, it keeps on decreasing for p values from 2 to 6. Here, higher order polynomial will fit the curve better hence there is decrease in error.

In the Test data graph, which is on right side, it is evident that for no regularization the error increases after p = 3 whereas regularization (orange line) remains straight line across x axis (which is same as train data in case of regularization). This graph shows overfitting with higher degree polynomial values in case of no regularization hence there is increase in error.

Graph when λ = 0(Without regularization and p value from 0 to 6)

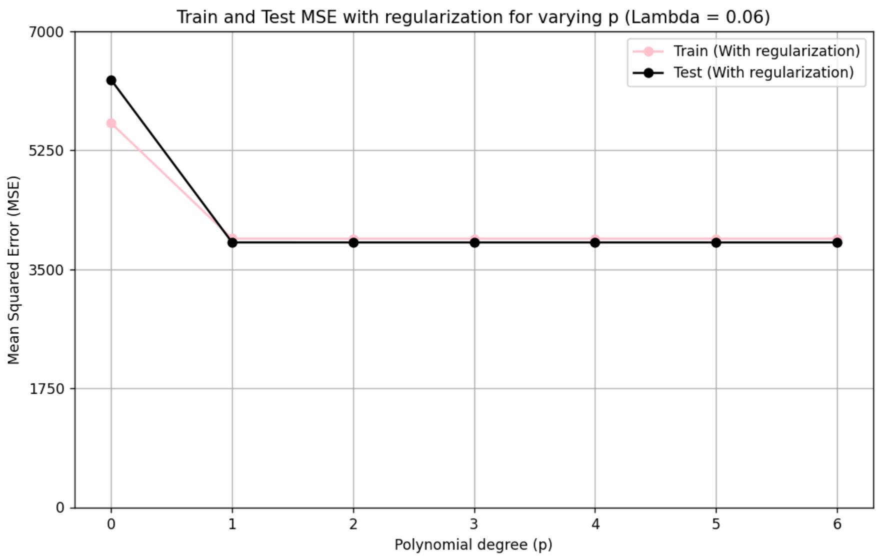
```
p = 0: Train MSE = 5650.710539, Test MSE = 6286.404792
p = 1: Train MSE = 3930.915407, Test MSE = 3845.034730
p = 2: Train MSE = 3911.839671, Test MSE = 3907.128099
p = 3: Train MSE = 3911.188665, Test MSE = 3887.975538
p = 4: Train MSE = 3885.473068, Test MSE = 4443.327892
p = 5: Train MSE = 3885.407157, Test MSE = 4554.830377
p = 6: Train MSE = 3866.883449, Test MSE = 6833.459149
```



From the above graph it is evident that when λ = 0 for test MSE data, the optimal value of p is 1 since test error reaches its minimum value which is equal 3845.034730 whereas for Train MSE data, the optimal value of p is 5 since error is minimum which is equal to 3885.407157. The lines for both the training and testing MSE with regularization are not similar after p = 3. The test error starts increasing after p = 3.

Graph when λ = 0.06(With regularization and p value from 0 to 6)

```
p = 0: Train MSE = 5650.711907, Test MSE = 6286.881967
p = 1: Train MSE = 3951.839124, Test MSE = 3895.856464
p = 2: Train MSE = 3950.687312, Test MSE = 3895.584056
p = 3: Train MSE = 3950.682532, Test MSE = 3895.582716
p = 4: Train MSE = 3950.682337, Test MSE = 3895.582668
p = 5: Train MSE = 3950.682335, Test MSE = 3895.582669
p = 6: Train MSE = 3950.682335, Test MSE = 3895.582669
```



From the above graph it is evident that when λ = 0.06(λ found in Problem 3), for test MSE data, the optimal value of p is 4 since test error reaches its minimum value which is equal to 3895.582668 whereas for Train MSE data, the optimal value of p is 5 or 6 since error is minimum which is equal to 3950.682335. The lines for both the training and testing MSE with regularization are very similar beyond p = 1.

Problem 6:

Approach	Train MSE	Test MSE
Linear Regression without intercept	19099.4468	106775.36145
Linear Regression with intercept	2187.1602	3707.8401
Ridge Regression with optimal Lambda	2451.52849	2851.330213
Ridge Regression using Gradient Descent	2439.163567	2842.456437
Non-linear Regression without regularization	3885.407157	3845.0347
Non-linear Regression with regularization	3950.682335	3895.582668

The above table shows that different regression techniques have different nature of performance in predicting diabetes levels. Linear regression without an intercept performed the worst which is having a very high test MSE of 106,775.36145 when compared to other approaches. Whereas linear regression with an intercept provided much better results, with a significantly lower test MSE of 3707.84, hence adding an intercept improved the model's ability to predict outcomes. **Ridge regression with optimal lambda and gradient descent showed even further improvements**, with the lowest test MSE values of 2851.330213 and 2842.456437 respectively. From the tabular results it is evident that regularization helps in preventing overfitting and better generalization on test data.

Non-linear regression without regularization had a test MSE of 3845.0347 and non-linear regression with regularization showed a slightly higher error at 3895.58. **Therefore, Ridge Regression using Gradient Descent, appear to offer the best balance between minimizing training and testing errors.** Hence, it is most suitable approach for predicting diabetes levels.