CSE 574 Introduction to Machine Learning

Programming Assignment 2

Classification and Regression

Group 4

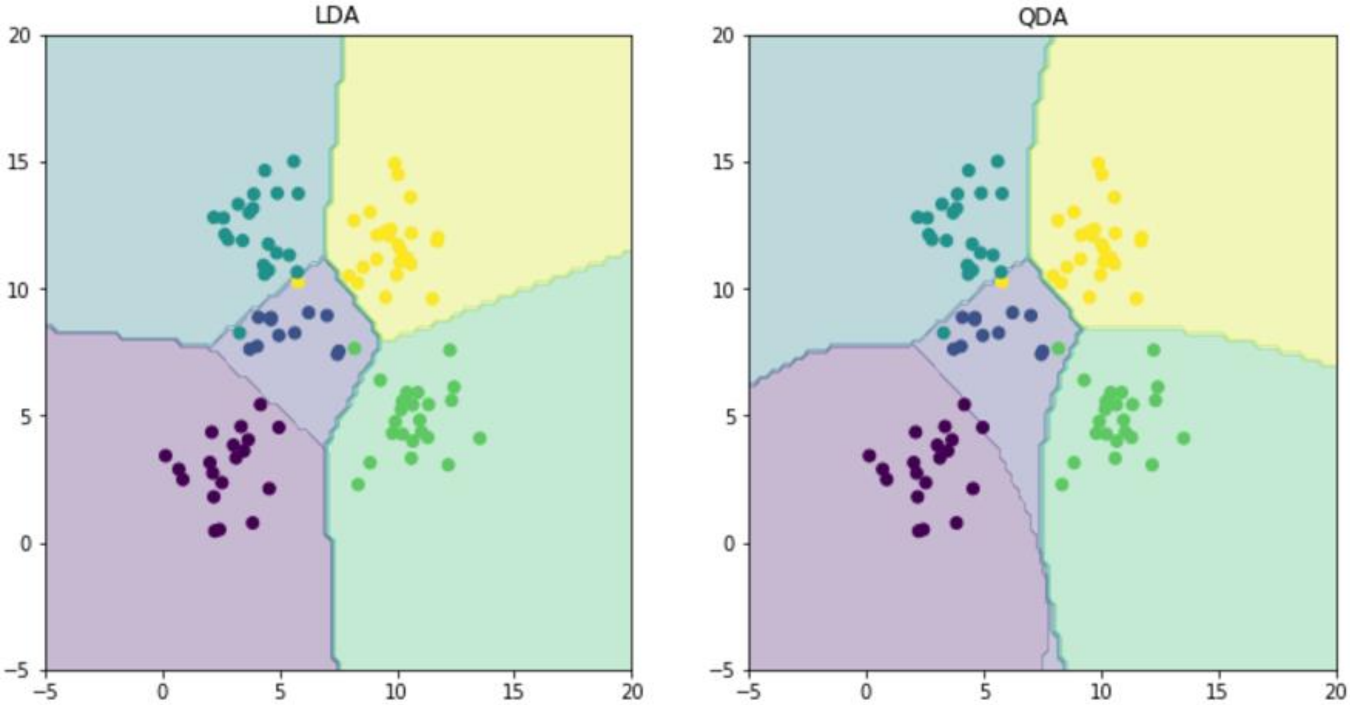
Saurabh Bajoria	50208005
Sumedh Ambokar	50207865
Vidhi Shah	50207090

Problem 1: Experiment with Gaussian Discriminators

Accuracy of LDA: 97%

Accuracy of QDA: 96%

The plots for discriminating boundary for linear and quadratic discriminators are as shown below:



- As seen in the above graphs, the decision boundaries for LDA are linear whereas the boundaries for QDA are curves. This difference is because LDA calculates the covariance matrix for the entire data set whereas QDA calculates different covariance matrices for each class separately.
- QDA estimates a separate matrix for each class, thus the expense of computation increases significantly. On the other hand, LDA assumes a common covariance matrix for the entire data set, so we need to do the computation only once.
- In our case, LDA does a slightly better job at classifying the data because we had very limited set of training data and QDA further divides the data into classes and computes covariance matrix for each class which might be inaccurate.

Problem 2: Experiment with Linear Regression

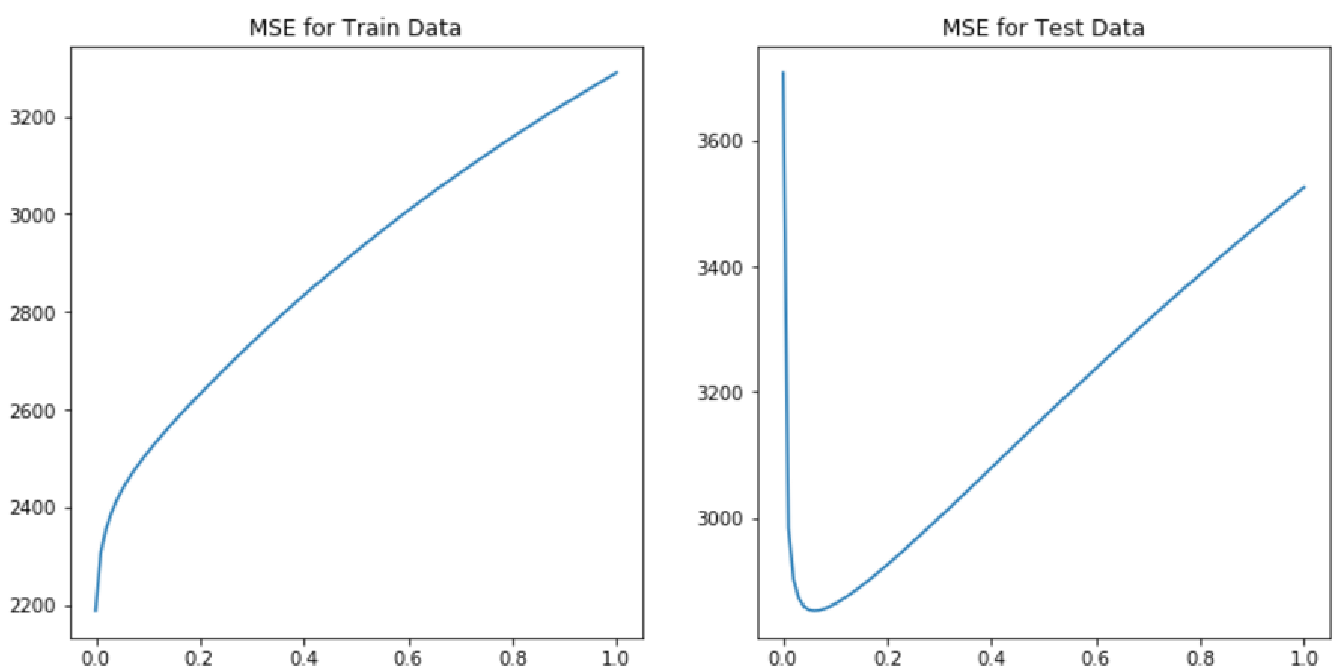
Test Data:
MSE without intercept: 106775.361558
MSE with intercept: 3707.84018132

Train Data:
MSE without intercept: 19099.4468446
MSE with intercept: 2187.16029493

- As mentioned above, the Mean Square error(MSE) for both training and test data is less with intercept as compared to without intercept. This is because, the model without intercept passes through origin and compares the values with zero while the model with intercept compares the values with mean.
- The overall error (with and without intercept) is less in train data as compared to test data which is as expected.
- Also, the error without intercept is around 30 times the error with intercept in test data whereas the error without intercept is around 9 times the error with intercept in train data. This shows that the error gets reduced by a significant amount in case of test data when using the intercept.

Problem 3: Experiment with Ridge Regression

The plots for errors on train and test data against lambda are as shown below:



- As seen above for test data, the value of MSE decreases initially up to a certain value of lambda.
- However, as the lambda is increased beyond this point, the value of MSE increases. This is because significant increase in the value of lambda causes underfitting and thus increases the MSE.

The table below shows the MSE values for test and train data against different values of lambda:

Lambda	Train MSE	Test MSE
0	2187.16	3707.84
0.01	2306.832	2982.446
0.02	2354.071	2900.974
0.03	2386.78	2870.942
0.04	2412.119	2858
0.05	2433.174	2852.666
0.06	2451.528	**2851.33**
0.07	2468.078	2852.35
0.08	2483.366	2854.88
0.09	2497.74	2858.444
0.1	2511.432	2862.758
0.11	2524.6	2867.638
0.12	2537.355	2872.962
0.13	2549.777	2878.646
0.14	2561.925	2884.627
0.15	2573.841	2890.859
0.16	2585.56	2897.307
0.17	2597.105	2903.941
0.18	2608.496	2910.739
0.19	2619.748	2917.682
0.2	2630.873	2924.753
0.21	2641.879	2931.939
0.22	2652.774	2939.226
0.23	2663.564	2946.605
0.24	2674.254	2954.065
0.25	2684.848	2961.599
0.26	2695.349	2969.198
0.27	2705.76	2976.855
0.28	2716.083	2984.564
0.29	2726.32	2992.32
0.3	2736.473	3000.116
0.31	2746.543	3007.948
0.32	2756.533	3015.811
0.33	2766.442	3023.7

Lambda	Train MSE	Test MSE
0.51	2932.26	3166.921
0.52	2940.827	3174.813
0.53	2949.331	3182.689
0.54	2957.773	3190.547
0.55	2966.153	3198.387
0.56	2974.473	3206.208
0.57	2982.732	3214.01
0.58	2990.932	3221.79
0.59	2999.074	3229.55
0.6	3007.157	3237.288
0.61	3015.183	3245.003
0.62	3023.153	3252.695
0.63	3031.066	3260.364
0.64	3038.924	3268.009
0.65	3046.728	3275.629
0.66	3054.477	3283.225
0.67	3062.173	3290.796
0.68	3069.816	3298.341
0.69	3077.406	3305.861
0.7	3084.945	3313.355
0.71	3092.433	3320.822
0.72	3099.871	3328.263
0.73	3107.259	3335.677
0.74	3114.597	3343.064
0.75	3121.886	3350.424
0.76	3129.128	3357.757
0.77	3136.322	3365.062
0.78	3143.468	3372.34
0.79	3150.568	3379.59
0.8	3157.622	3386.813
0.81	3164.63	3394.007
0.82	3171.593	3401.174
0.83	3178.512	3408.313
0.84	3185.387	3415.424

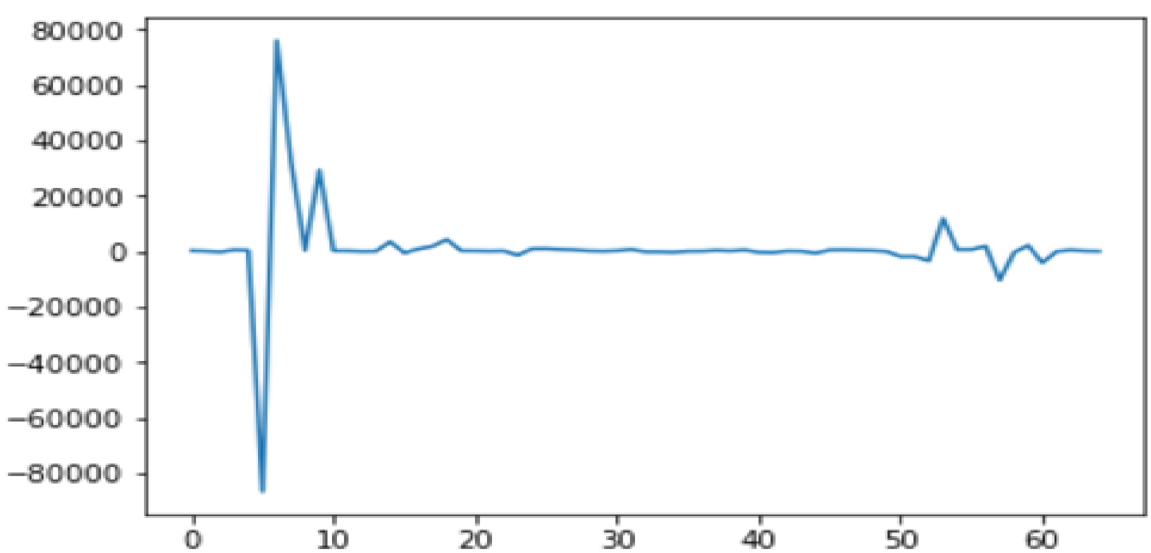
0.34	2776.273	3031.613
0.35	2786.027	3039.545
0.36	2795.704	3047.493
0.37	2805.305	3055.454
0.38	2814.831	3063.425
0.39	2824.284	3071.403
0.4	2833.664	3079.385
0.41	2842.972	3087.37
0.42	2852.208	3095.355
0.43	2861.374	3103.337
0.44	2870.471	3111.316
0.45	2879.498	3119.289
0.46	2888.458	3127.255
0.47	2897.35	3135.212
0.48	2906.176	3143.158
0.49	2914.935	3151.093
0.5	2923.63	3159.014

0.85	3192.218	3422.507
0.86	3199.006	3429.562
0.87	3205.751	3436.589
0.88	3212.454	3443.588
0.89	3219.115	3450.559
0.9	3225.735	3457.501
0.91	3232.315	3464.416
0.92	3238.854	3471.303
0.93	3245.353	3478.162
0.94	3251.812	3484.993
0.95	3258.232	3491.796
0.96	3264.614	3498.571
0.97	3270.957	3505.318
0.98	3277.263	3512.038
0.99	3283.53	3518.73
1	3289.761	3525.395
0.99	3283.53	3518.73
1	3289.761	3525.395

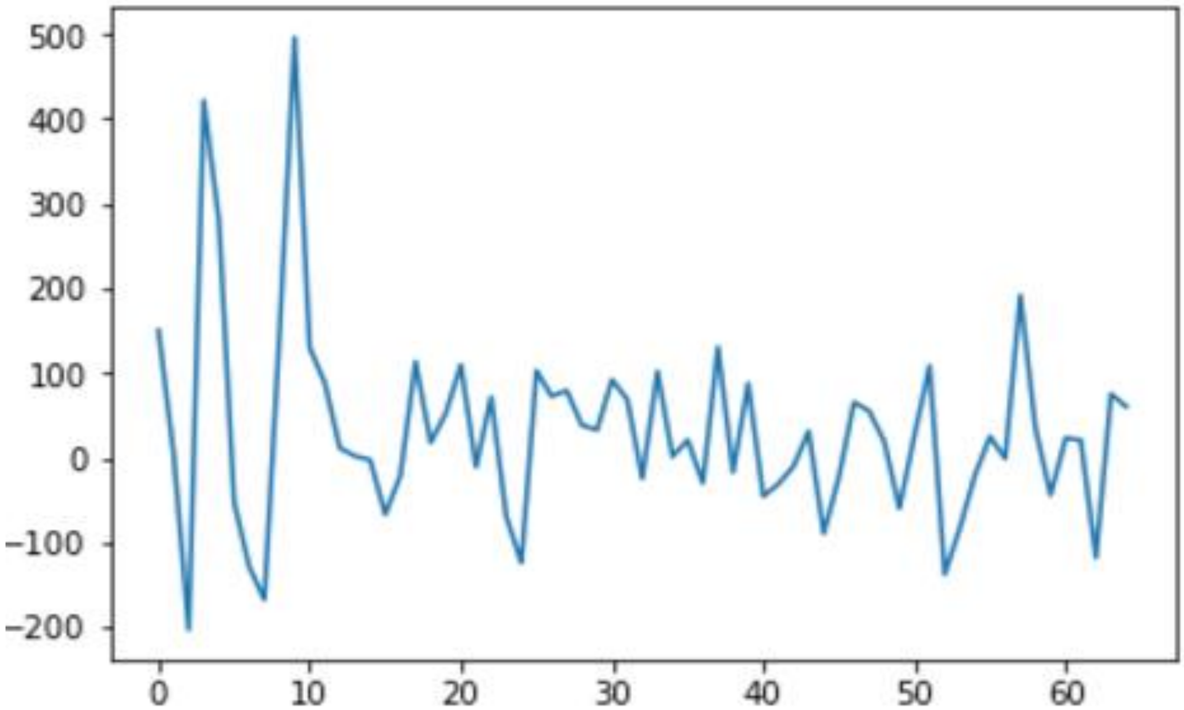
- Highlighted in the table above, the optimal value for Lambda is 0.06. At this value, we observe the least MSE for test data i.e. 2851.33
- Using OLE on train data, the MSE observed with intercept was 2187.16029493. Using Ridge regression, the lowest MSE is 2187.16 for lambda=0
- Similarly, using OLE on test data, the MSE observed with intercept was 3707.84018132. Using Ridge regression, the lowest MSE is 2851.33 for lambda=0.06. Thus, the MSE for ridge regression is significantly lower than the MSE for OLE.
- This indicates that Ridge regression is a better approach than the OLE.

The following figure plots the magnitudes of weight learnt by OLE and ridge regression:

OLE Regression:



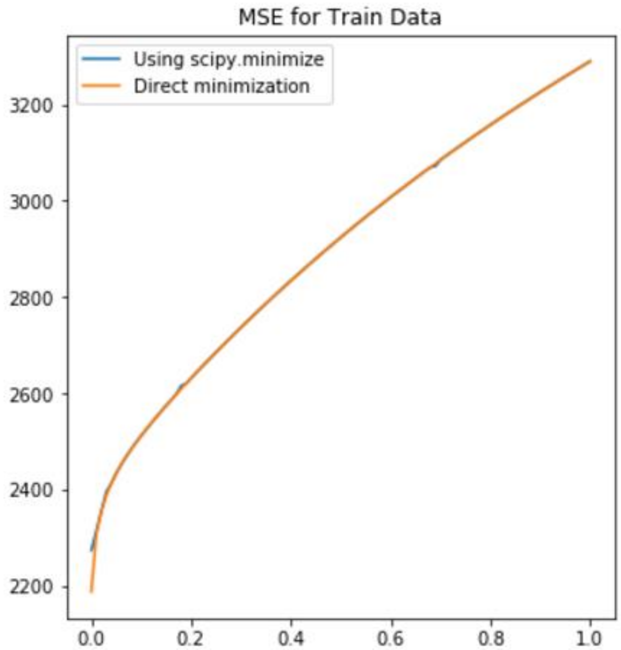
Ridge Regression:

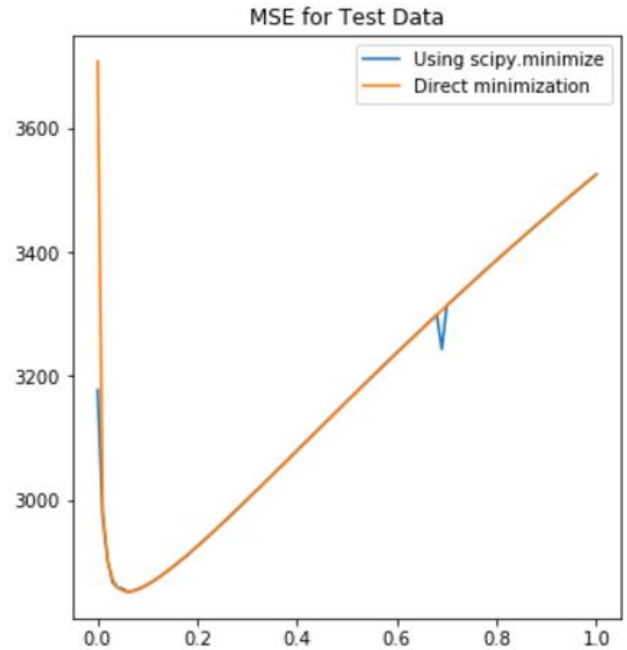


- The magnitudes of weights learnt by OLE regression is much higher than that of Ridge regression, as seen from the above two graphs.
- Also, the variance of weights is much smaller in ridge regression than in OLE. This is because of the regularization incorporated in the Ridge regression which keeps the weights in check.

Problem 4: Using Gradient Descent for Ridge Regression Learning

- Here we are analyzing Ridge Regression using gradient decent.
  The graph is plotted for MSE vs Lambda like the Ridge Regression graph from problem 3

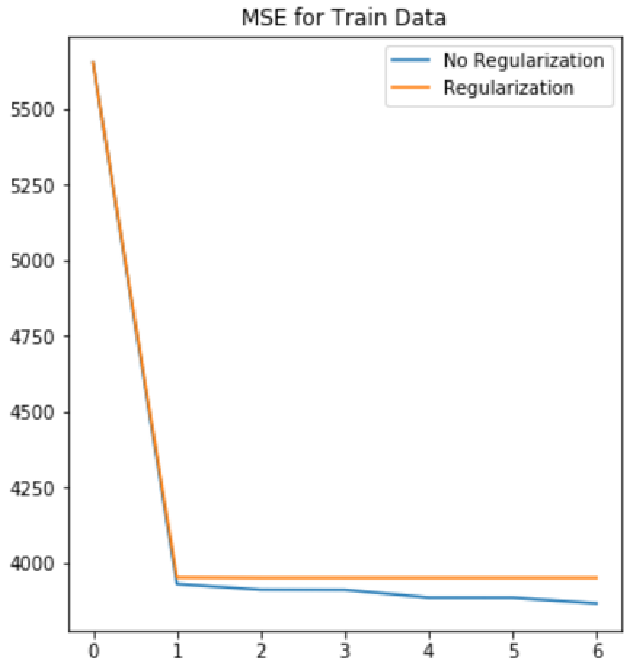


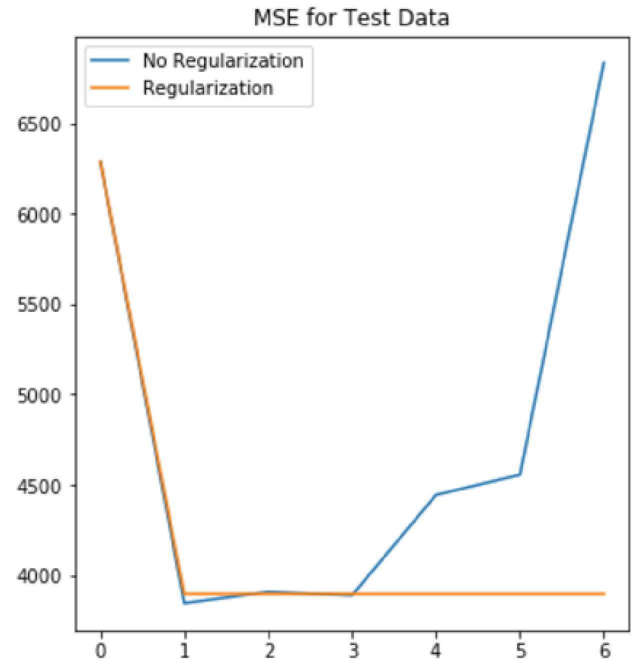


- We can observe that the curve plotted for Ridge Regression using gradient decent is similar to the one from problem 3.
- It is also observed that for test data, MSE value initially decreases but after value of Lambda reaches a threshold, MSE increases with increase in value of Lambda. This is because of underfitting which increases the MSE.
- Thus, gradient decent provides satisfactory results for Ridge Regression and is a convenient way of implementation.

Problem 5: Non-linear Regression

The below graphs show the MSE plotted against different values of higher order polynomials p both for non-regularization and regularization.





p	For lambda=0	For lambda=0.06
0	5650.710539	5650.711907
1	3930.915407	3951.839124
2	3911.839671	3950.687312
3	3911.188665	3950.682532
4	3885.473068	3950.682337
5	3885.407157	3950.682335
6	3866.883449	3950.682335

Train Data

p	For lambda=0	For lambda=0.06
0	6286.404792	6286.881967
1	3845.03473	3895.856464
2	3907.128099	3895.584056
3	3887.975538	3895.582716
4	4443.327892	3895.582668
5	4554.830377	3895.582669
6	6833.459149	3895.582669

Test Data

- As seen above, for non-regularized non-linear regression on Test Data, the error is high for p=0 as the regression line is horizontal which means that no learning is done from the training data. Error is minimum for p=1 and increases after that, resulting in overfitting of data as there is no regularization. However, with regularization, the value of MSE does not increase, as lambda provides regularization and avoids overfitting of data.

- For training data, the value of error is minimum for p=6 with regularization. When regularization is not done, the value of error is maximum for p=0 and decreases gradually as compared to non-regularization on test data as we learn on the same data set.
- **Optimal Values of P:**

  For test data:
  1. Regularization: 6
  2. Non-Regularization: 1

  For train data:
  1. Regularization: 6
  2. Non-Regularization: 5

Problem 6: Interpreting Results

The below table shows the Mean Squared Error(MSE) calculated on both training and test data for all the regression models:

Model	**Train data**	**Test data**
OLE	2187.16029	3707.84018
Ridge	2451.528	2851.33
Gradient Descent	2451.53381	2851.32567
Non-Linear (Without Regularization)	3866.88345	3845.03473
Non-Linear (With Regularization)	3950.68234	3895.58267

- We have analyzed all the regression models based on two prominent features which are MSE and Execution Time.
- We will be considering the testing error for evaluating the MSE of the regression models. Testing error is an appropriate indicator of how accurate the algorithm behaves for classification.
- Execution time can also be a crucial metric for systems where there is a resource crunch.

MSE:

- Regression without using an intercept tends to provide high values of both training and testing errors. Thus, we can eliminate it from our recommendation for the best regression model.
- Linear Regression using an intercept does provide lesser values of both training and testing errors but other models have better results. Linear Regression has a lower value of training error but has significantly high value of testing error.
- Non-Linear models have high value of training and testing errors but take lesser execution time.

- Ridge Regression models have best results when it comes to testing errors and can be most accurate for the given data.

Execution Time:
The below table shows the Execution time taken for all the regression models:

Model	Time(Secs)
OLE	0.005985
Ridge	0.000576
Gradient Descent	0.025502
Non-Linear	0.000501

- Execution time does not influence the performance of an algorithm when data is small but plays a key role to analyze an algorithm when large data is involved.
- As per our observations the nonlinear models execute in least amount of time. Ridge regression has the next best timing values.
- But as observed above Ridge Regression provide significantly better results when compared to nonlinear models.
- Ridge Regression using matrix inversion has better execution time but won't be feasible while working with large data as the data might not have the required structure.
- Thus, Ridge Regression with gradient decent is the most convenient and accurate regression model in the given scenario. Gradient decent can provide satisfactory results with appropriate selection of value of Lambda with minimal loss in accuracy.

References:

1. https://rpubs.com/ryankelly/LDA-QDA
2. https://en.wikipedia.org/wiki/Linear_regression