ASSIGNMENT 1

CS5691 Pattern Recognition and Machine Learning

CS5691 Assignemnt 1

Team Members:

BE17B007	N Sowmya Manojna
PH17B010	Riya Thakkar Anandbhai
PH17B011	Chaithanya Krishna Moorthy

Indian Institute of Technology, Madras

Contents

1 Question 1

1.1 Polynomial Regression

The data for uni-variate polynomial regression is prepared by raising it to the required degree. In case of uni-variate polynomial regression of degree d, the dependent variable, of size $(d, 1)$ is assumed to have the form

$$\vec{y}_{n\times 1} = \phi_{n\times d} W_{d\times 1} \tag{1}$$

The weights corresponding to a given degree is then calculated by using the closed form solution for uni-variate polynomial regression:

$$W = (\phi^T \phi + \lambda I)^{-1} \phi^T \vec{y} \tag{2}$$

Where, λI is the regularization term.

1.2 Grid Search

In order to pick the parameters that best fit the dataset, a grid search was performed on the dataset. Prior to this, the dataset was split into training set, validation set and the testing set, in the ratio 70:10(from the training data):30. The results obtained is as follows:

Degree	**Lambda**	**Train error**	**Validation error**	**Sum Error**
6	0.0	0.044889	0.159636	0.204525
3	0.0	0.672882	1.001484	1.674366
6	0.5	0.708018	1.292034	2.000052
9	0.5	0.750020	1.469413	2.219433
6	1.0	0.978764	1.556590	2.535354
3	0.5	1.017522	1.649754	2.667276
2	1.0	1.169652	1.584834	2.754486
2	0.5	1.059525	1.711646	2.771171
2	2.0	1.471590	1.412142	2.883731
2	0.0	1.014199	1.883134	2.897333
9	1.0	1.040132	1.929033	2.969165
6	2.0	1.390322	1.712728	3.103050
3	1.0	1.375405	1.858616	3.234021
9	2.0	1.354363	2.165779	3.520142
3	2.0	1.851634	1.918302	3.769936
9	10.0	2.281929	1.857270	4.139198
6	10.0	2.797239	1.657453	4.454692
3	10.0	3.085788	1.446564	4.532352
2	10.0	3.474567	1.065662	4.540229
9	50.0	3.342110	1.447933	4.790042
9	100.0	3.782560	1.380623	5.163183
6	50.0	3.746184	1.431362	5.177546

6	100.0	3.939933	1.377351	5.317284
3	50.0	4.446303	1.192392	5.638695
3	100.0	5.249380	1.211820	6.461200
2	50.0	6.080079	1.109623	7.189702
2	100.0	6.788129	1.180800	7.968929
9	0.0	5.063475	92.085167	97.148642

Table 1: Results obtained for Task 1, with sample size of 10

Degree	Lambda	Train error	Validation error	Sum Error
6	0.0	0.094536	0.094379	0.188914
9	0.0	0.093581	0.100752	0.194333
6	0.5	0.128136	0.144512	0.272648
9	0.5	0.134226	0.152565	0.286791
6	1.0	0.187428	0.203255	0.390684
9	1.0	0.186479	0.209008	0.395487
9	2.0	0.289107	0.311716	0.600824
6	2.0	0.302235	0.310699	0.612934
9	10.0	0.766298	0.776521	1.542819
6	10.0	0.787556	0.786211	1.573767
3	1.0	0.939789	0.843160	1.782948
3	0.5	0.935552	0.851141	1.786693
3	2.0	0.955492	0.836864	1.792356
3	0.0	0.934079	0.862605	1.796685
3	10.0	1.245737	1.053461	2.299198
2	0.5	1.591943	1.420648	3.012591
2	1.0	1.592243	1.420474	3.012717
2	0.0	1.591842	1.421021	3.012863
2	2.0	1.593419	1.420708	3.014127
2	10.0	1.626363	1.446713	3.073077
6	50.0	1.586311	1.708840	3.295151
9	50.0	1.620063	1.707757	3.327820
2	50.0	2.072812	1.882810	3.955622
9	100.0	2.138200	2.310223	4.448424
6	100.0	2.168534	2.377912	4.546446
3	50.0	2.379269	2.229847	4.609116
2	100.0	2.696613	2.519404	5.216017
3	100.0	2.947778	2.853756	5.801534

Table 2: Results obtained for Task 1, with sample size of 200

From the table above, we see that the best fit for the data is obtained for degree: 6 and $\lambda : 0$.
The best fit, $d : 6$ and $\lambda : 0$ is visualized as follows:

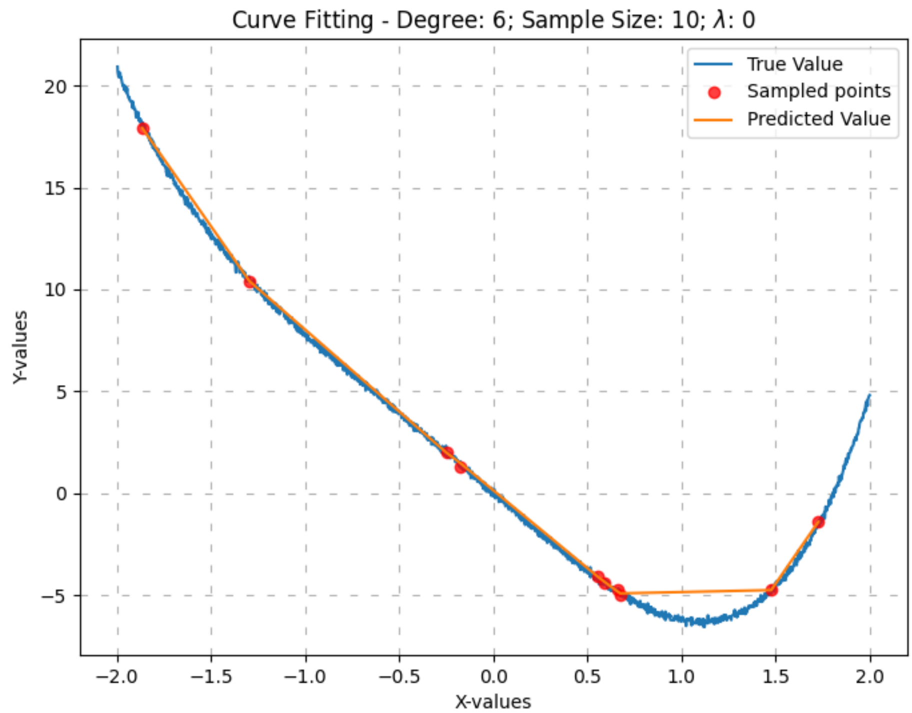


Figure 1: Task 1 - best fit, Sample size: 10

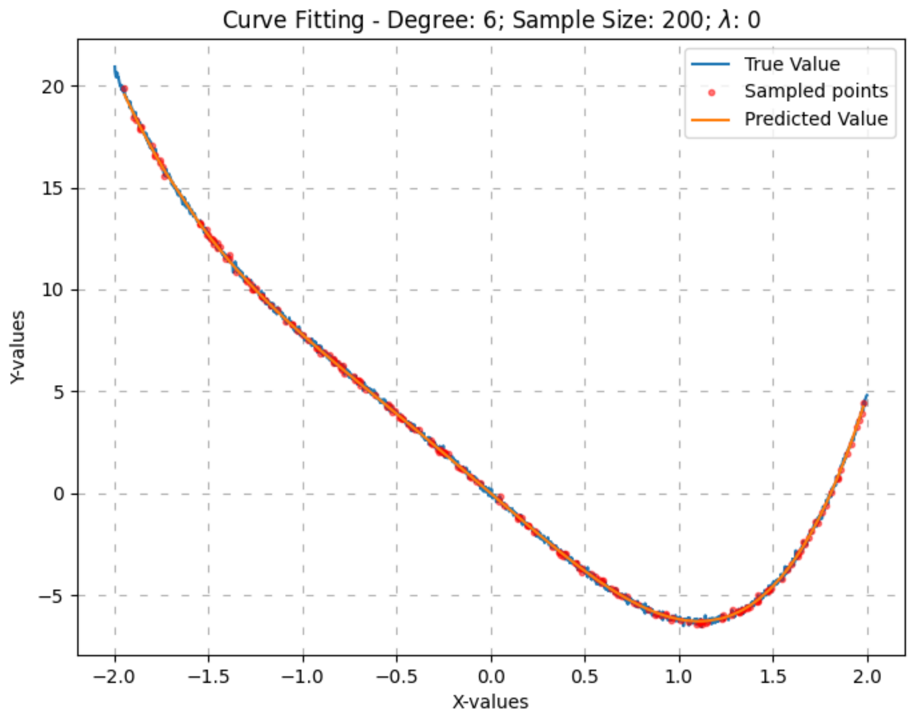


Figure 2: Task 1 - best fit, Sample size: 200

The final training and testing error obtained is as follows:
Training Error: 0.09974659089780814 and Testing Error: 0.09793071099285168

2 Question 2

2.1 Polynomial Regression for bi-variate data

The second dataset is a bi-variate data with 2000 examples. We assume that the target variable is of the form:

$$y = \sum_{i=0} \omega_i \phi_i(x1, x2) + \epsilon \tag{3}$$

Where ω_i are the parameters to be found through regression, $\phi_i(x1, x2)$ is a polynomial in x1 and x2 and ϵ is the normally distributed error.
A breakdown of the steps undertaken is:

1. The function **create_phi** generates the design matrix $\phi(x1, x2)$ for the required degree of complexity.

2. The design matrix is passed to the function **regularized_pseudo_inv** , which generates the moore-penrose inverse of the given design matrix(X) and specified value of regularization parameter lambda(λ).

$$(\lambda I + X^T X)^{-1} X^T \tag{4}$$

3. The function **opt_regularized_param** is then used to obtain optimum values of $\vec{\omega}$

$$\vec{\omega} = [(\lambda I + X^T X)^{-1} X^T].y \tag{5}$$

   Where y is the output as defined in the equation 3.

4. The optimum parameter vector thus obtained can be used to predict the variable y for new inputs.

$$y_{prediction} = X\vec{\omega} \tag{6}$$

2.2 Degree of complexity 2

With degree of complexity set to 2, for a particular data point,

$$y_i = \omega_0 + \omega_1(x1) + \omega_2(x2) + \omega_3(x1)^2 + \omega_4(x2)^2 + \omega_5(x1).(x2) \tag{7}$$

2.3 Data processing

1. The range of variables is same for x1 and x2 = (-16,16), hence no scaling is required, there are no null data values.

2. The data is first shuffled and then split into Train data, Cross-validation data and Test data. Train data sizes = 50,200,500

3. The independent vectors $\vec{x1}$ and $\vec{x2}$ are extracted from the datasets and then design matrices $\phi_{n\times5}$ are created using the function **create_phi**. Here, n is the number of samples in the respective sub dataset.

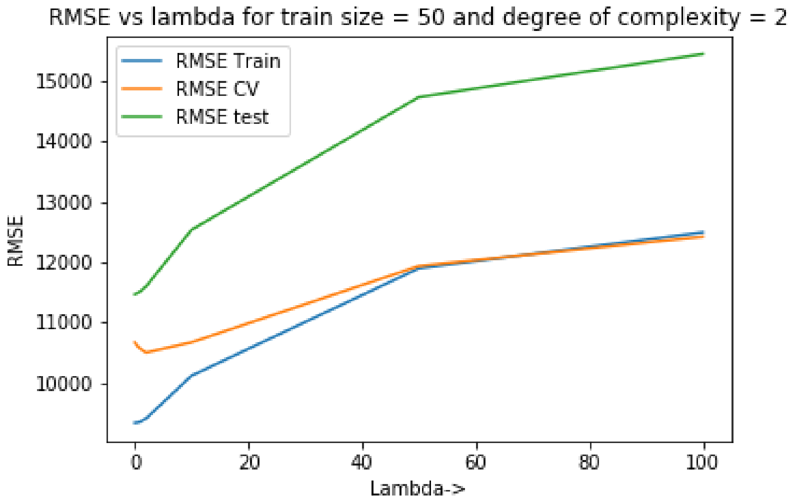


Figure 3: RMSE vs lambda for train data size = 50 and degree of complexity= 2

2.4 Effect of varying regularization parameter, lambda for train data size = 50

After the above pre-processing, The optimum parameter $\vec{\omega}$ is obtained for the following lambda values - 0,0.5,1,2,10,50,100 The variation in RMSE vs lambda is as follows:

Lambda	RMSE Train	RMSE CV	RMSE Test
0	9340.73	10671.59	11468.19
0.5	9346.19	10611.41	11487.13
1	9361.17	10564.66	11516.49
2	9412.58	10503.67	11598.33
10	10120	10672.93	12534.16
50	11897.56	11935.27	14729.56
100	12491.10	12420.38	15443.04

Table 3: Variation in RMSE values with lambda

2.5 Effect of varying regularization parameter, lambda for train data size = 200

The optimum parameter $\vec{\omega}$ is obtained for the following lambda values - 0,0.5,1,2,10,50,100 The variation in RMSE vs lambda is as follows:

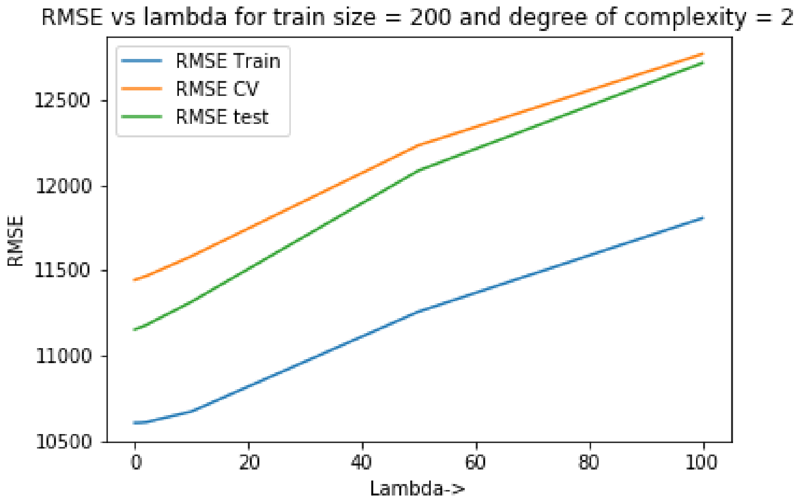


Figure 4: RMSE vs lambda for train data size = 200 and degree of complexity= 2

Lambda	RMSE Train	RMSE CV	RMSE Test
0	10	10671.59	11468.19
0.5	9346.19	10611.41	11487.13
1	9361.17	10564.66	11516.49
2	9412.58	10503.67	11598.33
50	11897.56	11935.27	14729.56
100	12491.10	12420.38	15443.04

Table 4: Variation in RMSE values with lambda

2.6 Effect of varying regularization parameter, lambda for train data size = 500

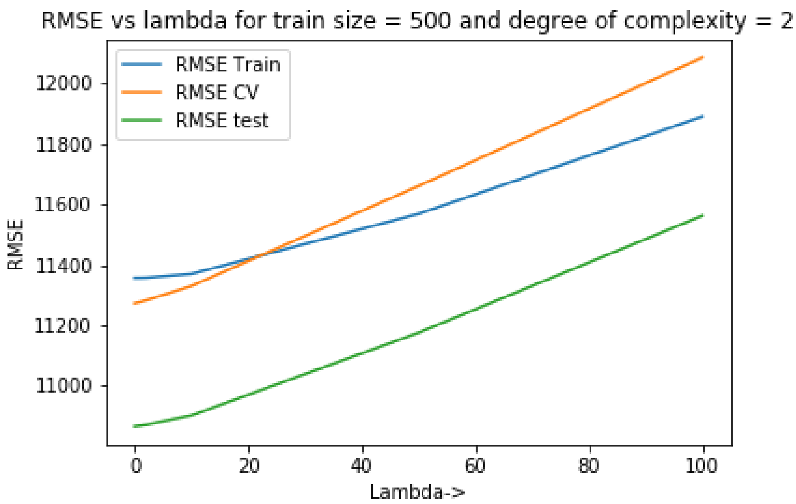


Figure 5: RMSE vs lambda for train data size = 500 and degree of complexity= 2

2.7 Degree of complexity 3

With degree of complexity set to 3, for a particular data point,

$$y_i = \omega_0+\omega_1(x1)+\omega_2(x2)+\omega_3(x1)^2+\omega_4(x2)^2+\omega_5(x1).(x2)+\omega_6(x1)^3+\omega_7(x2)^3+\omega_8(x2)^2(x1)+\omega_9(x1)^2(x2) \tag{8}$$

2.8 Effect of varying regularization parameter, lambda for train data size = 50

After the above pre processing, The optimum parameter $\vec{\omega}$ is obtained for the following lambda values - 0,0.5,1,2,10,50,100 The variation in RMSE vs lambda is as follows:

Lambda	RMSE Train	RMSE CV	RMSE test
0.0	8407.761210828727	11960.392166857815	12390.588660098681
0.5	8414.299614637861	11743.19358761457	12329.126658336485
1.0	8431.988457444206	11558.772331311482	12293.299535838765
2.0	8491.472017549635	11270.79171291384	12280.362852665143
10.0	9240.199538933726	10757.219031210356	13112.100832682248
50.0	10876.901176573218	12040.04490493105	15628.464519645266
100.0	11379.791399813128	12642.949547533039	16474.271301190667

Table 1: Results obtained for Task 2

Variation in rmse values with lambda for complexity= 3 and train size = 50:

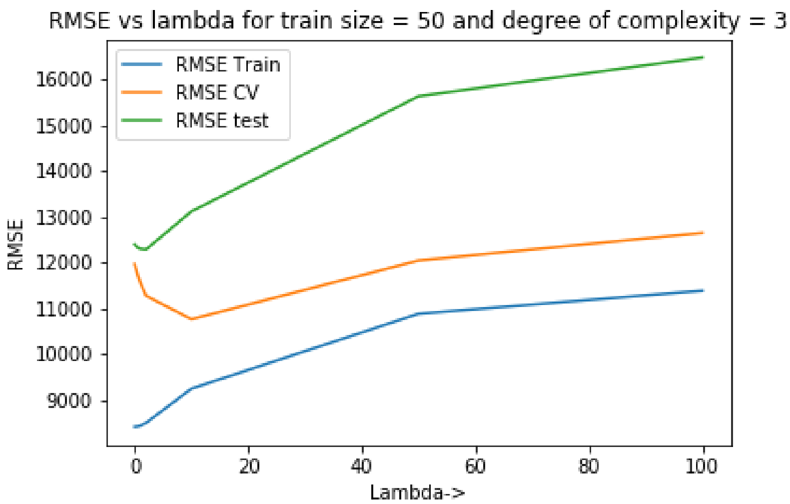


Figure 6: RMSE vs lambda for train data size = 50 and degree of complexity= 3

2.9 Effect of varying regularization parameter, lambda for train data size = 200

Te obtained results are:

Lambda	RMSE Train	RMSE CV	RMSE test
0.0	10322.506441224592	11480.65137585371	11540.748956514151
0.5	10322.74337910385	11484.060426748905	11548.276365031981
1.0	10323.437876912261	11487.78843512935	11556.1380959018
2.0	10326.106384432025	11496.129590042878	11572.780261162094
10.0	10392.059429467075	11592.560024074122	11735.150238100397
50.0	10984.962331062083	12191.950852364467	12578.93878751002
100.0	11522.792243145757	12692.364674117875	13234.250404753897

Table 1: Results obtained for Task 2

Scatter plots of the model prediction using the regularization parameter value 0.01:

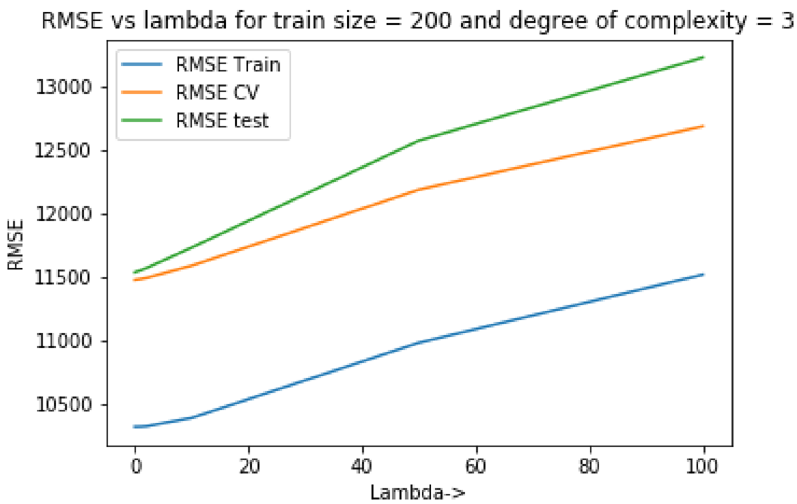


Figure 7: RMSE vs lambda for train data size = 200 and degree of complexity= 3

2.10 Effect of varying regularization parameter, lambda for train data size = 500

The optimum parameter $\vec{\omega}$ is obtained for the following lambda values - 0,0.5,1,2,10,50,100 The variation in RMSE vs lambda is as follows:

2.11 Degree of complexity 6

With the degree of complexity 6,

$$y = \sum_{(} x1)^{\alpha} * (x2)^{\beta} \tag{9}$$

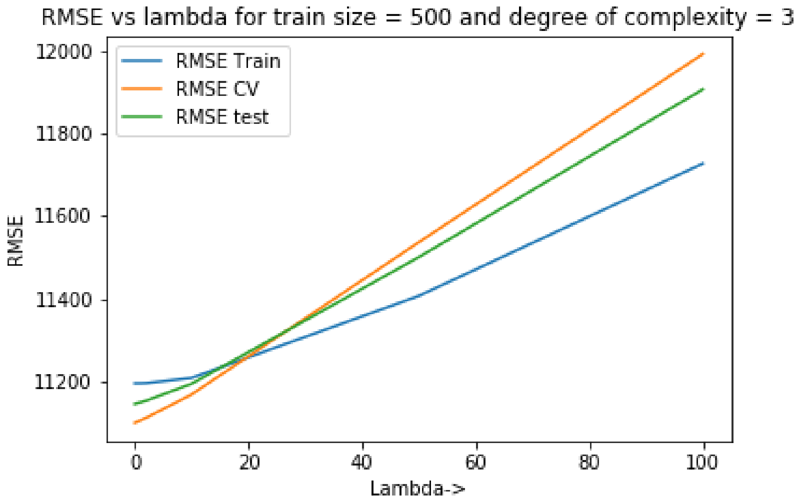


Figure 8: RMSE vs lambda for train data size = 500 and degree of complexity= 3

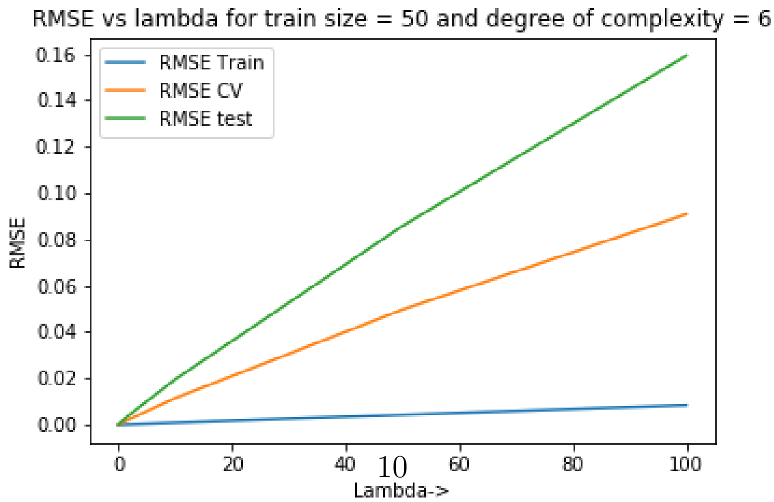
Where $0 \leq \alpha + \beta \leq 6$.

2.12 Effect of varying regularization parameter, Lambda for train size = 50

Lambda	**RMSE Train**	**RMSE CV**	**RMSE test**
0.0	1.3582344908807028e-07	1.3024937374699014e-07	2.736466435265001e-07
0.5	5.3224862846108924e-05	0.0006389419451237852	0.0011697424661700818
1.0	0.00010241277566647342	0.001259472447857696	0.0022735033897776087
2.0	0.00019571426483158983	0.002466786482370453	0.004369578591012229
10.0	0.0009066540184949185	0.011454915983379615	0.0195422946271592
50.0	0.004319989638299114	0.049682786993700954	0.08571221964007386
100.0	0.008418172149435164	0.09086562659012756	0.15931182940985875

Table 1: Results obtained for Task 2

RMSE variation for complexity = 6 and train size = 50



10.0	0.000128403762827723	0.0001679510933427838	0.000216489022423017
50.0	0.0005954210072570998	0.0007608482959799609	0.0009543646518816395
100.0	0.0011768765498580448	0.0014847238097789764	0.001842159446862891

Table 1: Results obtained for Task 2

RMSE variation for complexity = 6 and train size = 50

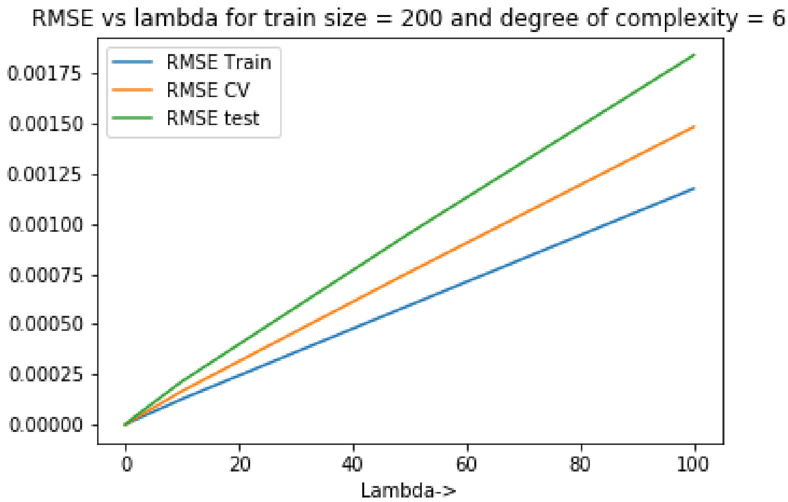


Figure 10: RMSE vs lambda for train data size = 200 and degree of complexity= 6

2.14 Effect of varying regularization parameter, lambda for train size = 500

Lambda	RMSE Train	RMSE CV	RMSE test
0.0	3.5408872821237 27e-08	3.741549292426492e-08	3.4517606283226707e-08
0.5	2.4891560253931053e-06	2.69805964526573e-06	2.5949552832042936e-06
1.0	4.963649710534798e-06	5.3778590424185165e-06	5.176457054060217e-06
2.0	9.866381792471788e-06	1.069014286492481e-05	1.0291657829414182e-05
10.0	4.744117748092903e-05	5.148944778527705e-05	4.9743962467771885e-05
50.0	0.00021913097081968563	0.00023736669064951702	0.00023193961964998602
100.0	0.00042714320882117563	0.00046046329475841634	0.00045296787349301 64

Table 1: Results obtained for Task 2

RMSE variation for complexity = 6 and train size = 500

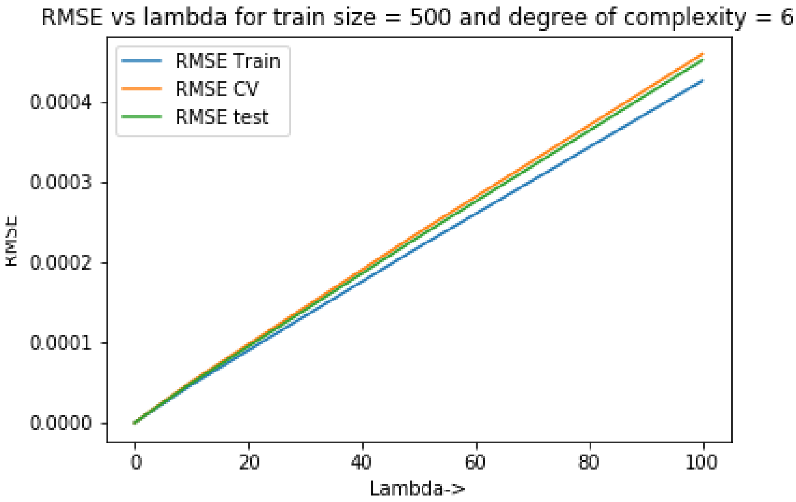


Figure 11: RMSE vs lambda for train data size = 500 and degree of complexity= 6

2.15 Conclusion

The RMSE values are least over train data, cross-validation data as well as test data for degree of complexity = 6, train data size = 500 and regularization parameter, lambda =0.

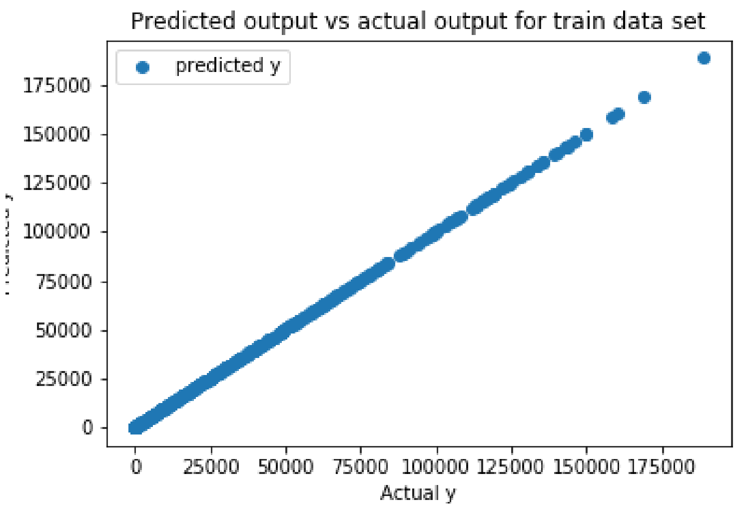


Figure 12: Predicted output vs actual values for train data

The surface plots of approximated functions superimposed with the scatter points are generated when the attached python code is run

3 Question 3

Linear regression using Gaussian basis function is given as

$$y(\vec{x}, \vec{w}) = \sum_{i=0}^{D-1} \omega_i \phi_i(\vec{x}) \tag{10}$$

, where D is a hyperparameter. The basis function

$$\phi_i = e^(- |\vec{x} - \vec{\mu}_i|^2 / \sigma^2) \tag{11}$$

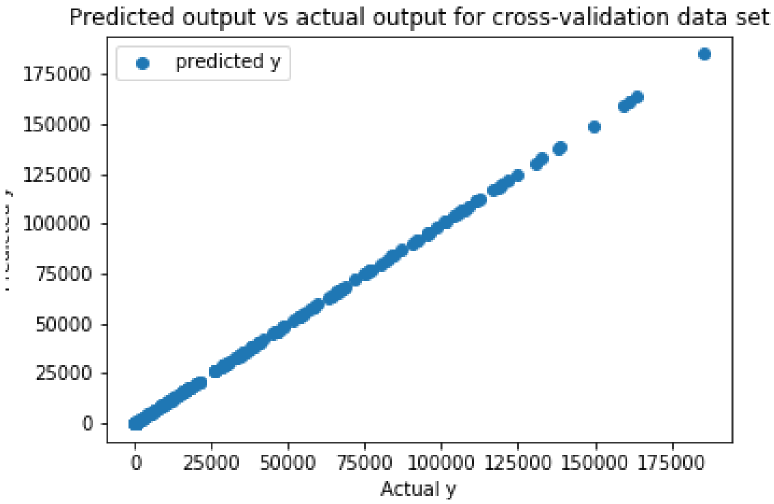


Figure 13: Predicted output vs actual values for cv data

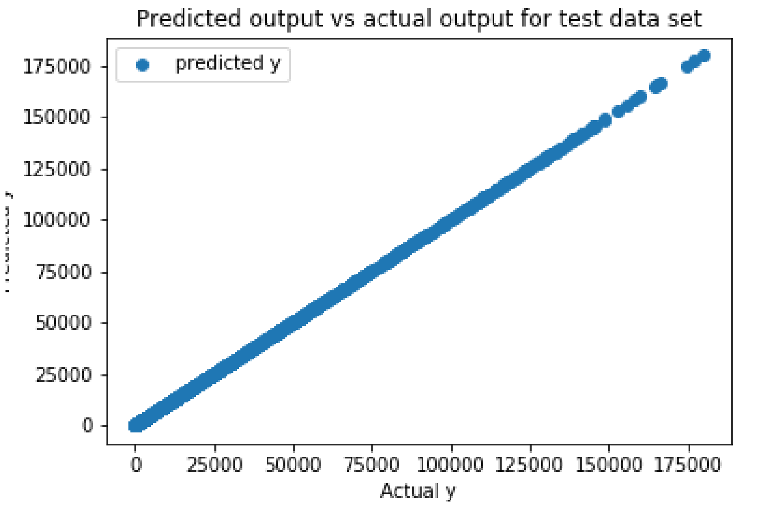


Figure 14: Predicted output vs actual values for test data

where i = 1,2 ... D-1. The μ are the mean vectors for D-1 kernels made from the data set. The value of the mean vectors are found using the KMeans clustering algorithm. In this work, the `sklearn` KMeans function was used. The optimum number of clusters for the dataset 2 - "`function_12D.csv`" was found to be 10 clusters. For the dataset 3 - "`1_bias_clean.csv`", the optimum number of clusters are 9.

3.1 No regularization

The following plots were obtained

3.2 Quadratic Regularization

Optimal parameters using quadratic regularization is given by $\vec{\omega^*} = (\Phi * T\Phi + \lambda I)^{-1}\Phi^T\vec{t}$; λ is the regularization parameter. The values 0.01,0.1, 1.0,5.0,10.0 were used to estimate the optimal parameters and the RMSE on the cross-validation set was calculated for each value. The best peroming model was selected as the one having least RMSE on CV data
For dataset 2, the RMSE values for the Training, CV and Test values corresponding to each

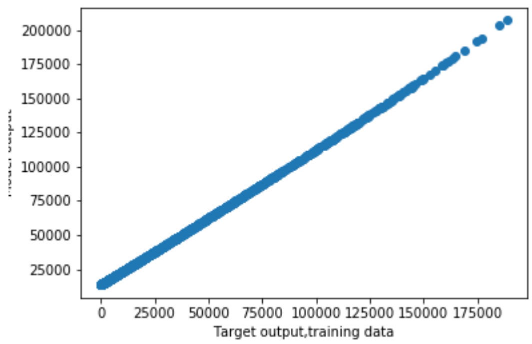


Figure 15: Scatter plot of the target values vs model prediction for Training set of Dataset 2, using linear regression with gaussian basis and no Regularization, $\lambda = 0.01$

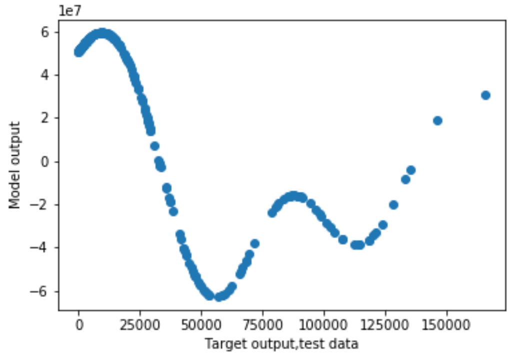


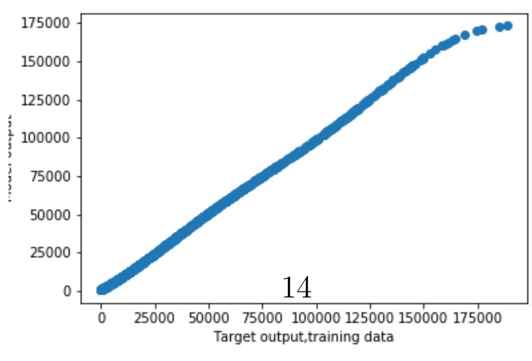
Figure 16: Scatter plot of the target values vs model prediction for Test set of Dataset 2, using linear regression with gaussian basis and no Regularization, $\lambda = 0.01$

lambda value is

Lambda	**RMSE Train**	**RMSE CV**	**RMSE test**
0.01	3059.2231939706166	304419.6502059433	1309688.1097631603
0.1	2967.5181829927037	1260.488802049498	41195.93571016699
1.0	2990.3948869258456	1425.9232970077237	39596.78510421749
5.0	3013.546633117982	1503.9615829712025	39006.81341800579
10.0	3036.360553338793	1541.3504903884834	38684.11470986189

Table 1: RMSE values for Training, test and CV data, for dataset 2 using quadratic regularization for "$N_T min$"

Scatter plots of the model prediction using the regularization parameter value 0.01:



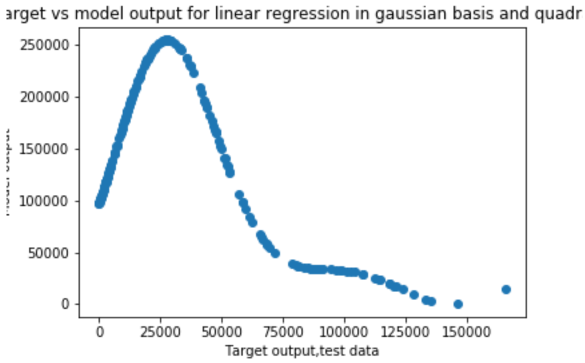


Figure 18: Scatter plot of the target values vs model prediction for Test set of Dataset 2, using linear regression with gaussian basis and quadratic Regularization, $\lambda = 0.01$

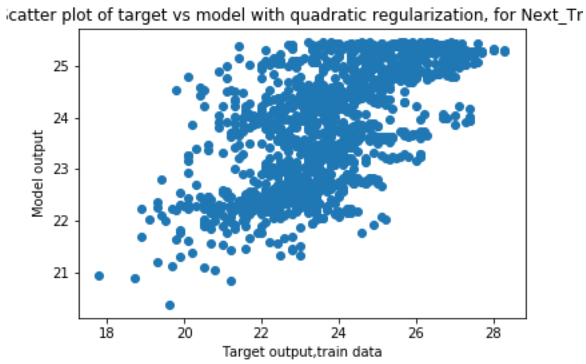


Figure 19: Scatter plot of the target values vs model prediction for Training set of Dataset 3, using linear regression with gaussian basis and quaadratic Regularization, $\lambda = 0.01$ for "NTmin" output variable

3.3 Tikhonov Regularization

The Tikhonov regularization term is given by $\vec{\omega^*} = (\Phi * T\Phi + \lambda\tilde{\Phi})^{-1}\Phi^T\vec{t}$. The $\tilde{\Phi}$ term is defined as

$$\tilde{\Phi} = [\tilde{\phi}]_{i,j=1}^{K} \tag{12}$$

where K is the number of clusters and λ is the regularization parameter. The values 0.01,0.1, 1.0,5.0,10.0 were used to estimate the optimal parameters and the RMSE on the cross-validation set was calculated for each value. The best peroming model was selected as the one having least RMSE on CV data

Applying Tikhonov regularization to the bivariate dataset, the optimal value of was estimated to be 0.01. The table for the RMSE values for the Training, CV and Test values corresponding to each *lambda* value is

Lambda	RMSE Train	RMSE CV	RMSE test
0.01	78112040.25241715	80416219813.42682	43898825037.538
0.1	78629878.58895023	276304216179.1798	176596027875.55765
1.0	79471104.52781227	357583741029.9827	104372422450.83612
5.0	77593937.82583737	272444591755.6569	174802698267.85687
10.0	77693846.61754198	242869557997.72043	158196054705.92648

Table 1: RMSE values for Training, test and CV data, for dataset 2 using Tikhonov regularization

Scatter plots of the model prediction using the regularization parameter value 0.01:

0.01	4929.01653053444	414848.3042013735	3312196.0992440097
0.1	4624.091536077471	1669.205290237633	36902.34473960538
1.0	4637.926167783597	2073.9375316355663	39568.04405922714
5.0	4652.374704231647	2257.2672544567195	40450.17809627787
10.0	4667.40003673528	2346.920490034092	40853.959722564214

Table 1: RMSE values for Training, test and CV data, for dataset 3 using Tikhonov regularization, for "$\mathrm{N}_T max$"

plots were obtained: figures 27 and 28.

4 Conclusion

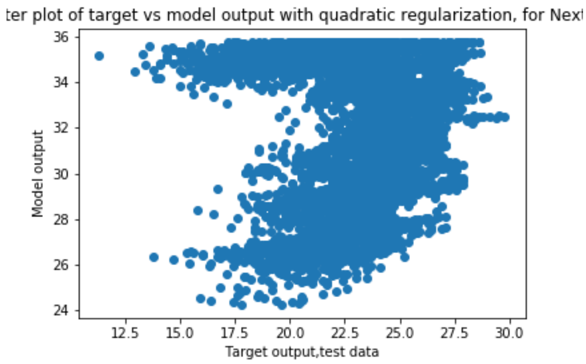


Figure 20: Scatter plot of the target values vs model prediction for Test set of Dataset 2, using linear regression with gaussian basis and quadratic Regularization, $\lambda = 0.01$, for "NTmin" output variable

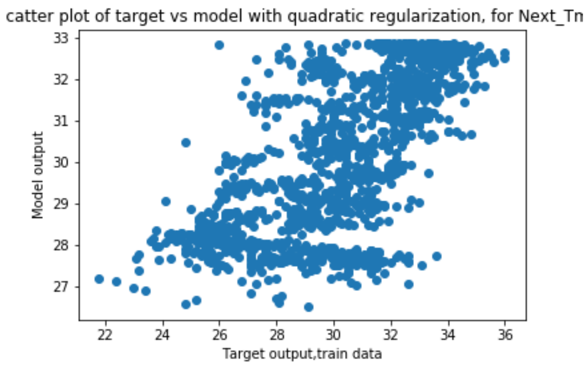


Figure 21: Scatter plot of the target values vs model prediction for Training set of Dataset 3, using linear regression with gaussian basis and quaadratic Regularization, $\lambda = 0.01$ for "NTmax" output variable

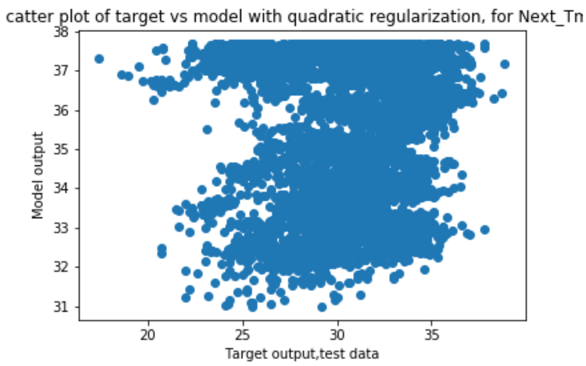


Figure 22: Scatter plot of the target values vs model prediction for Test set of Dataset 2, using linear regression with gaussian basis and quadratic Regularization, $\lambda = 0.01$, for "NTmax" output variable

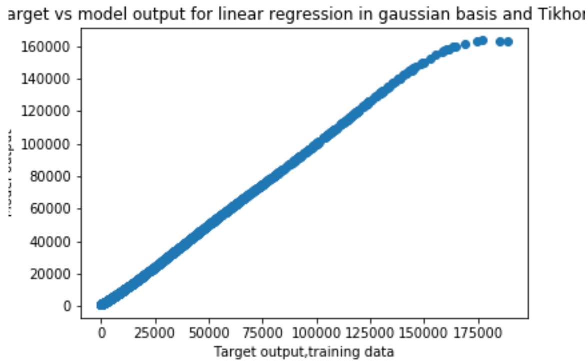


Figure 23: Scatter plot of the target values vs model prediction for Training set of Dataset 2, using linear regression with gaussian basis and Tikhonov Regularization, $\lambda = 0.01$

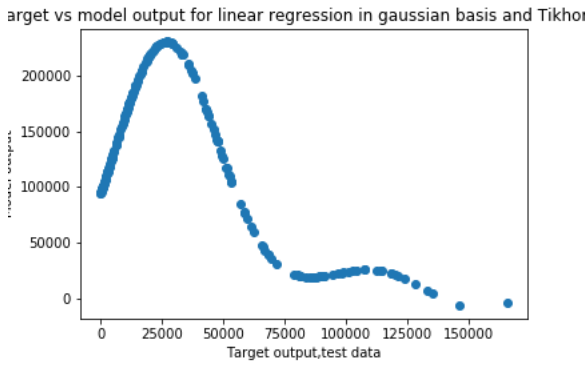


Figure 24: Scatter plot of the target values vs model prediction for Test set of Dataset 2, using linear regression with gaussian basis and Tikhonov Regularization, $\lambda = 0.01$

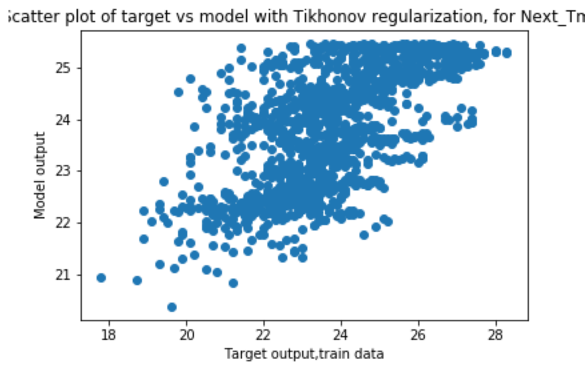


Figure 25: Scatter plot of the target values vs model prediction for Training set of Dataset 3, using linear regression with gaussian basis and Tikhonov Regularization, $\lambda = 0.1$

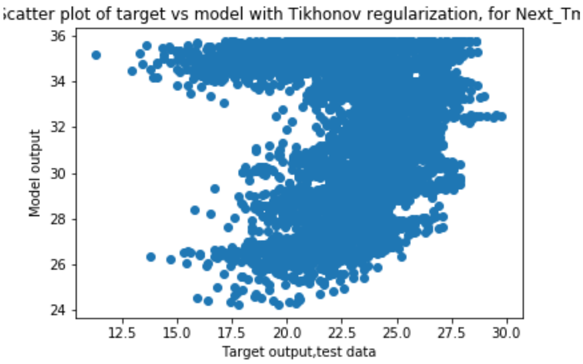


Figure 26: Scatter plot of the target values vs model prediction for Test set of Dataset 3, using linear regression with gaussian basis and Tikhonov Regularization, $\lambda = 0.01$

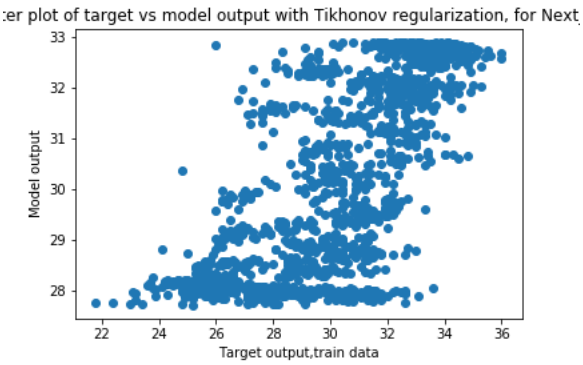


Figure 27: Scatter plot of the target values vs model prediction for Training set of Dataset 3, using linear regression with gaussian basis and Tikhonov Regularization, $\lambda = 0.1$

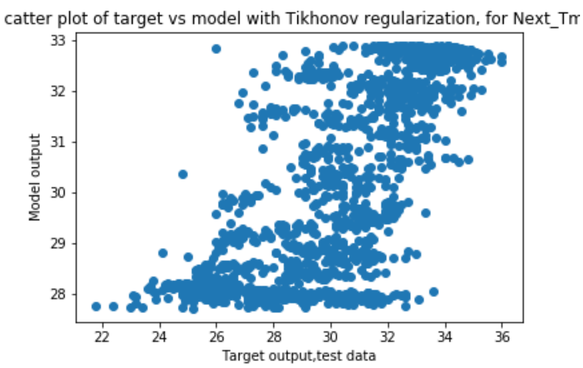


Figure 28: Scatter plot of the target values vs model prediction for Test set of Dataset 3, using linear regression with gaussian basis and Tikhonov Regularization, $\lambda = 0.1$