Machine Learning CS60050

Assignment 1
Curve Fitting With Gradient Descent

Shivam Kumar Jha

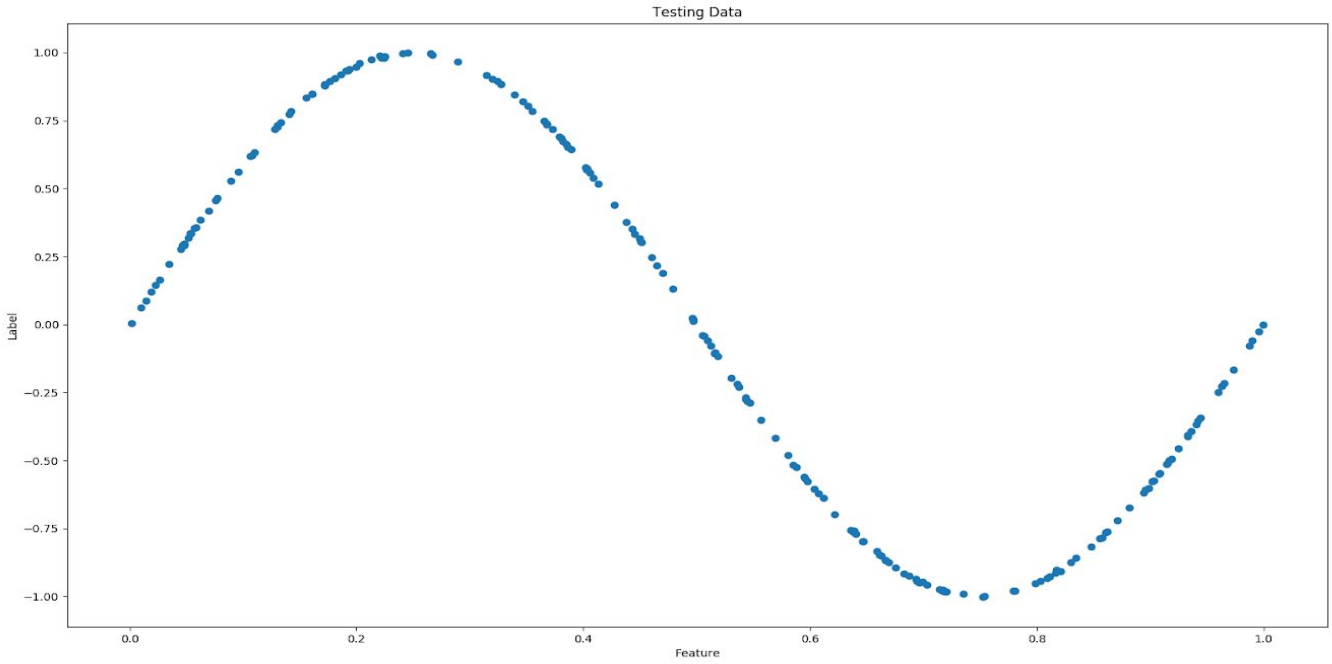
17CS30033

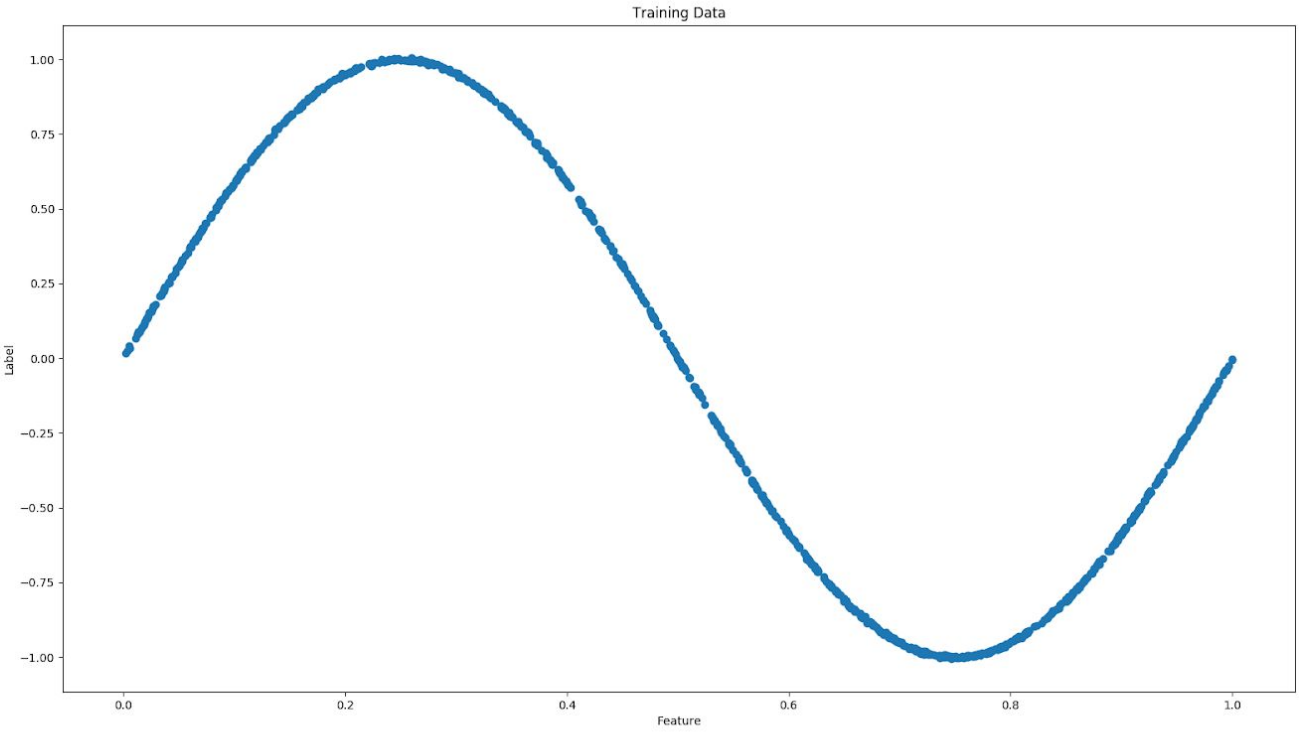
General Information

I've used Python as the language for doing the assignment and information on how to run the code is written in README.pdf. A **learning rate of 0.05** has been used along with a converge criteria that the difference between **two consecutive costs should not be greater than 0.00000001**. All the plots are also present in the submission folder.

Initially, I tried with a more relaxed convergence point but the graphs were not coming with much accuracy. I also started using python itself but later switched to numpy vectors for speed. Vectorization involves expressing mathematical operations, such as the multiplication we're using here, as occurring on entire arrays rather than their individual elements (as in a for-loop). With vectorization, the underlying code is parallelized such that the operation can be run on multiply array elements at once, rather than looping through them one at a time.

Answer To 1a


Testing Data
Label
Feature
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0.0
0.2
0.4
0.6
0.8
1.0


Training Data
Label
Feature
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0.0
0.2
0.4
0.6
0.8
1.0

The above plots are also available in higher resolution as “testing_data_plot.png” and “training_data_plot.png”

Answer to 1b

The predicted labels for the features are present in the folder “predicted_labels” with names “predicted_labels_n.csv” where n represents the degree of the polynomial.

Below are the details for predicted parameters for different degrees of the polynomial.

NOTE: The parameters for the polynomials are also present in the file “predicted_parameters.txt”.

========== Polynomial Degree 1 ==========

Parameters: [0.91609257 -1.85515881]

Squared Error on Test Data: **0.0955305340264786**

========== Polynomial Degree 2 ==========

Parameters: [0.97374505 -2.20096505 0.34052727]

Squared Error on Test Data: **0.09579851920170313**

========== Polynomial Degree 3 ==========

Parameters: [-0.07671096 10.50647719 -31.22342516 20.91009886]

Squared Error on Test Data: **0.003248850901895813**

========== Polynomial Degree 4 ==========

Parameters: [0.08304287 7.17832471 -15.64570255 -3.95364149 12.65570878]

Squared Error on Test Data: **0.004675914873744095**

========== Polynomial Degree 5 ==========

Parameters: [0.19224162 5.40744098 -10.24041835 -4.97457161 2.51910479 7.5532347]

Squared Error on Test Data: **0.008864550892919437**

========== Polynomial Degree 6 ==========

Parameters: [0.07187978 7.23221242 -15.80129906 -2.21439591 7.14343828 6.20216435 -2.3066975]

Squared Error on Test Data: **0.0045908198829193265**

========== Polynomial Degree 7 ==========

Parameters: [0.03408172 7.64689745 -16.09554391 -3.72309084 6.58913232 8.27255123 3.22195633 -5.75157165]

Squared Error on Test Data: **0.002333572675940087**

========== Polynomial Degree 8 ==========

Parameters: [0.03688883 7.44725993 -14.80938571 -4.95991551 4.64976525 7.86202875 5.78645938 0.48757847 -6.41213449]

Squared Error on Test Data: **0.0014136094363091148**

========== Polynomial Degree 9 ==========

Parameters: [0.05625352 7.05546052 -13.32894872 -5.57128763 2.93826133

6.66167982 6.23979143 3.21321827 -1.1720244 -6.09228559]

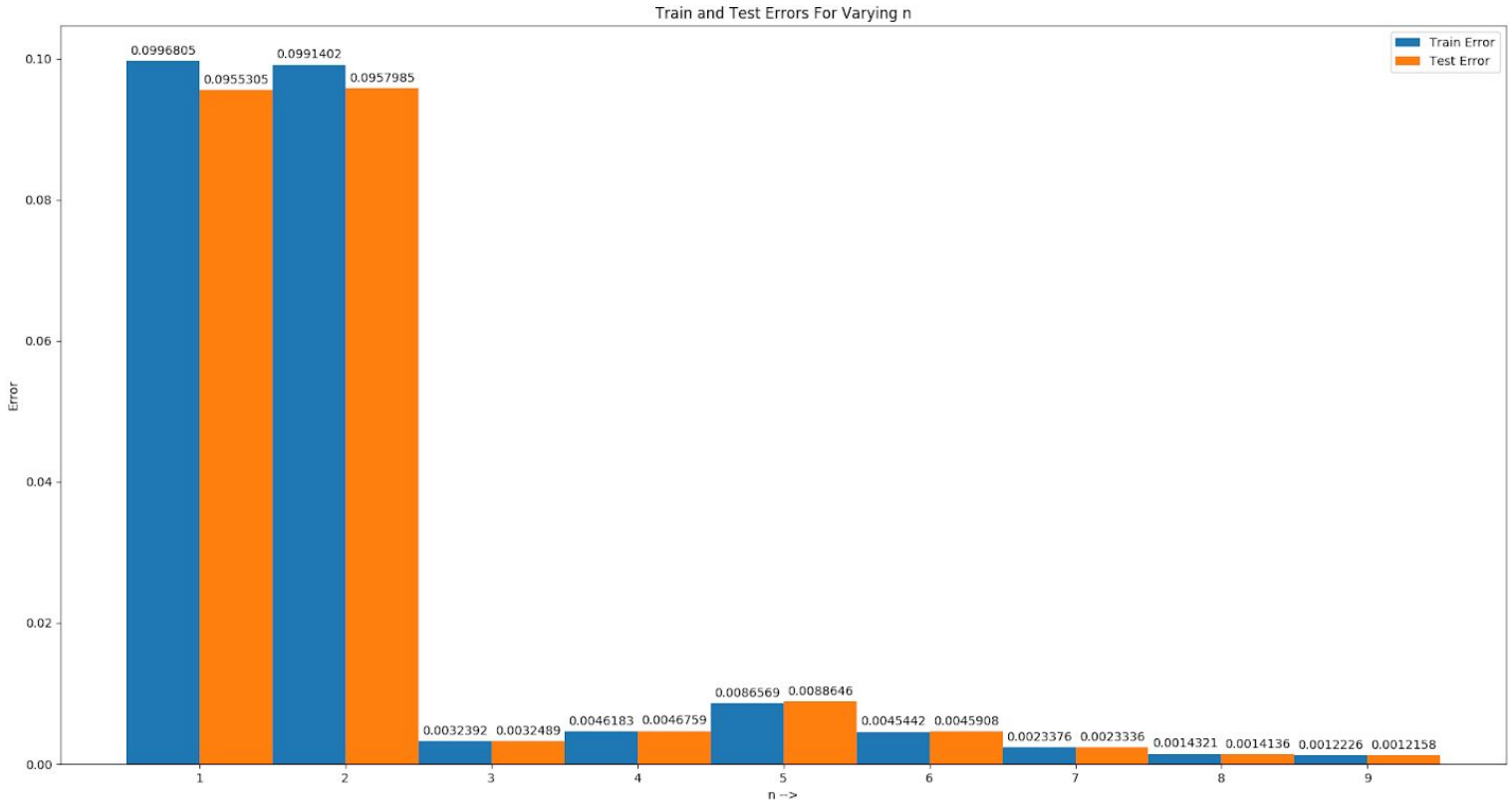
Squared Error on Test Data: **0.0012158219405710774**

Answer to 2a

All the plots of the predicted polynomials are available in the folder “polynomial_plots” with names “pol_n.png” where n denotes the degree of the polynomial.

Answer to 2b

A higher resolution version of the same is available in the folder “result_plots” with name “train_test_varying_n.png”.



Clearly we can see that the predicted polynomial of degree 9 has the least amount of squared error in both test data and train data and hence it is the best fit.

Answer to 3

The best curve as we can see above **is one with degree 9** and the **worst is one with degree 1** based on the training error. Below are the results obtained on regularisation over varying lambdas with same convergence as initial. Later on the graphs are present with discussion.

======= Lasso Regularisation Best Curve: Polynomial Degree 9 Lambda 0.25

Polynomial is: [0.05625352 7.05546052 -13.32894872 -5.57128763 2.93826133

6.66167982 6.23979143 3.21321827 -1.1720244 -6.09228559]

New Parameters: [0.05625285 7.05546795 -13.32896229 -5.57129117 2.93826698

6.66168828 6.23979789 3.21322016 -1.17202806 -6.09229484]

Lasso Error on Test Data: 0.03392160575909278

Squared Error on Test Data: 0.0012158117236797175

====== Lasso Regularisation Best Curve: Polynomial Degree 9 Lambda 0.5

Polynomial is: [0.05625352 7.05546052 -13.32894872 -5.57128763 2.93826133

6.66167982 6.23979143 3.21321827 -1.1720244 -6.09228559]

New Parameters: [0.05625285 7.05546795 -13.32896229 -5.57129117 2.93826698

6.66168828 6.23979789 3.21322016 -1.17202806 -6.09229484]

Lasso Error on Test Data: 0.06662739979450584

Squared Error on Test Data: 0.0012158117236797106

====== Lasso Regularisation Best Curve: Polynomial Degree 9 Lambda 0.75

Polynomial is: [0.05625352 7.05546052 -13.32894872 -5.57128763 2.93826133

6.66167982 6.23979143 3.21321827 -1.1720244 -6.09228559]

New Parameters: [0.01776313 7.48270728 -14.11403396 -5.76781709 3.27053167

7.1462221 6.60114965 3.31093535 -1.38481659 -6.60919563]

Lasso Error on Test Data: 0.10518821757656271

Squared Error on Test Data: 0.0007410192650143116

======= Lasso Regularisation Best Curve: Polynomial Degree 9 Lambda 1

Polynomial is: [0.05625352   7.05546052 -13.32894872 -5.57128763   2.93826133

  6.66167982   6.23979143   3.21321827 -1.1720244   -6.09228559]

New Parameters: [-9.49355229e-03 7.78753550e+00   -1.46846410e+01   -5.89274640e+00

 3.52381478e+00 7.48865030e+00 6.83783752e+00 3.35737857e+00

-1.54169143e+00 -6.94548252e+00]

Lasso Error on Test Data: 0.145710642213836

Squared Error on Test Data: 0.0005374632748063601

====== Lasso Regularisation Worst Curve: Polynomial Degree 1 Lambda 0.25

Polynomial is: [0.91609257   -1.85515881]

New Parameters: [0.91610343   -1.85517822]

Lasso Error on Test Data: 0.09726269470720293

Squared Error on Test Data: 0.09553064367284046

====== Lasso Regularisation Worst Curve: Polynomial Degree 1 Lambda 0.5

Polynomial is: [0.91609257 -1.85515881]

New Parameters: [0.91610343   -1.85517822]

Lasso Error on Test Data: 0.09899474574156539

Squared Error on Test Data: 0.09553064367284046

====== Lasso Regularisation Worst Curve: Polynomial Degree 1 Lambda 0.75

Polynomial is: [0.91609257 -1.85515881]

New Parameters: [0.91610343 -1.85517822]

Lasso Error on Test Data: 0.10072679677592786

Squared Error on Test Data: 0.09553064367284046

======= Lasso Regularisation Worst Curve: Polynomial Degree 1 Lambda 1

Polynomial is: [0.91609257   -1.85515881]

New Parameters: [0.91610343   -1.85517822]

Lasso Error on Test Data: 0.10245884781029033

Squared Error on Test Data: 0.09553064367284046

======== Ridge Regularisation Best Curve: Polynomial Degree 9 Lambda 0.25

Polynomial is: [0.05625352   7.05546052 -13.32894872 -5.57128763   2.93826133

  6.66167982   6.23979143   3.21321827 -1.1720244   -6.09228559]

New Parameters: [-0.04467353   8.19246439 -15.49491187 -5.98302215   3.94239721

  7.92802112   7.04607581   3.30122257 -1.7762257   -7.22796958]

Ridge Error on Test Data: 0.336267119678139

Squared Error on Test Data: 0.0004251772730618497

======= Ridge Regularisation Best Curve: Polynomial Degree 9 Lambda 0.5

Polynomial is: [0.05625352 7.05546052 -13.32894872 -5.57128763 2.93826133
6.66167982 6.23979143 3.21321827 -1.1720244 -6.09228559]

New Parameters: [-0.05023656 8.26231917 -15.66034002 -5.9618738 4.05558518
7.99673553 7.0295167 3.23273861 -1.83006013 -7.19500243]

Ridge Error on Test Data: 0.6809627622880461

Squared Error on Test Data: 0.0004176813363171439

======= Ridge Regularisation Best Curve: Polynomial Degree 9 Lambda 0.75

Polynomial is: [0.05625352 7.05546052 -13.32894872 -5.57128763 2.93826133
6.66167982 6.23979143 3.21321827 -1.1720244 -6.09228559]

New Parameters: [-0.05244609 8.29201594 -15.73851101 -5.94179712 4.11648763
8.02397128 7.00628035 3.18528447 -1.85723613 -7.15558754]

Ridge Error on Test Data: 1.0260039009285136

Squared Error on Test Data: 0.0004139500492847 9473

======= Ridge Regularisation Best Curve: Polynomial Degree 9 Lambda 1

Polynomial is: [0.05625352   7.05546052 -13.32894872 -5.57128763   2.93826133

  6.66167982   6.23979143   3.21321827 -1.1720244   -6.09228559]

New Parameters: [-0.05376647   8.31068731 -15.79111996 -5.92433597   4.16050875

  8.04029179   6.98447367   3.14722931 -1.87628668 -7.11945286]

Ridge Error on Test Data: 1.3713580181678686

Squared Error on Test Data: 0.00041113993666686

======= Ridge Regularisation Worst Curve: Polynomial Degree 1 Lambda 0.25

Polynomial is: [0.91609257   -1.85515881]

New Parameters: [0.91610343   -1.85517822]

Ridge Error on Test Data: 0.09820622600512167

Squared Error on Test Data: 0.09553064367284046

======= Ridge Regularisation Worst Curve: Polynomial Degree 1 Lambda 0.5

Polynomial is: [0.91609257   -1.85515881]

New Parameters: [0.91743721   -1.85756138]

Ridge Error on Test Data: 0.10090964423669806

Squared Error on Test Data: 0.09554436258894075

======= Ridge Regularisation Worst Curve: Polynomial Degree 1 Lambda 0.75

Polynomial is: [0.91609257 -1.85515881]

New Parameters: [0.91825182 -1.85901692]

Ridge Error on Test Data: 0.10361385842300783

Squared Error on Test Data: 0.09555298907247321

======= Ridge Regularisation Worst Curve: Polynomial Degree 1 Lambda 1

Polynomial is: [0.91609257 -1.85515881]

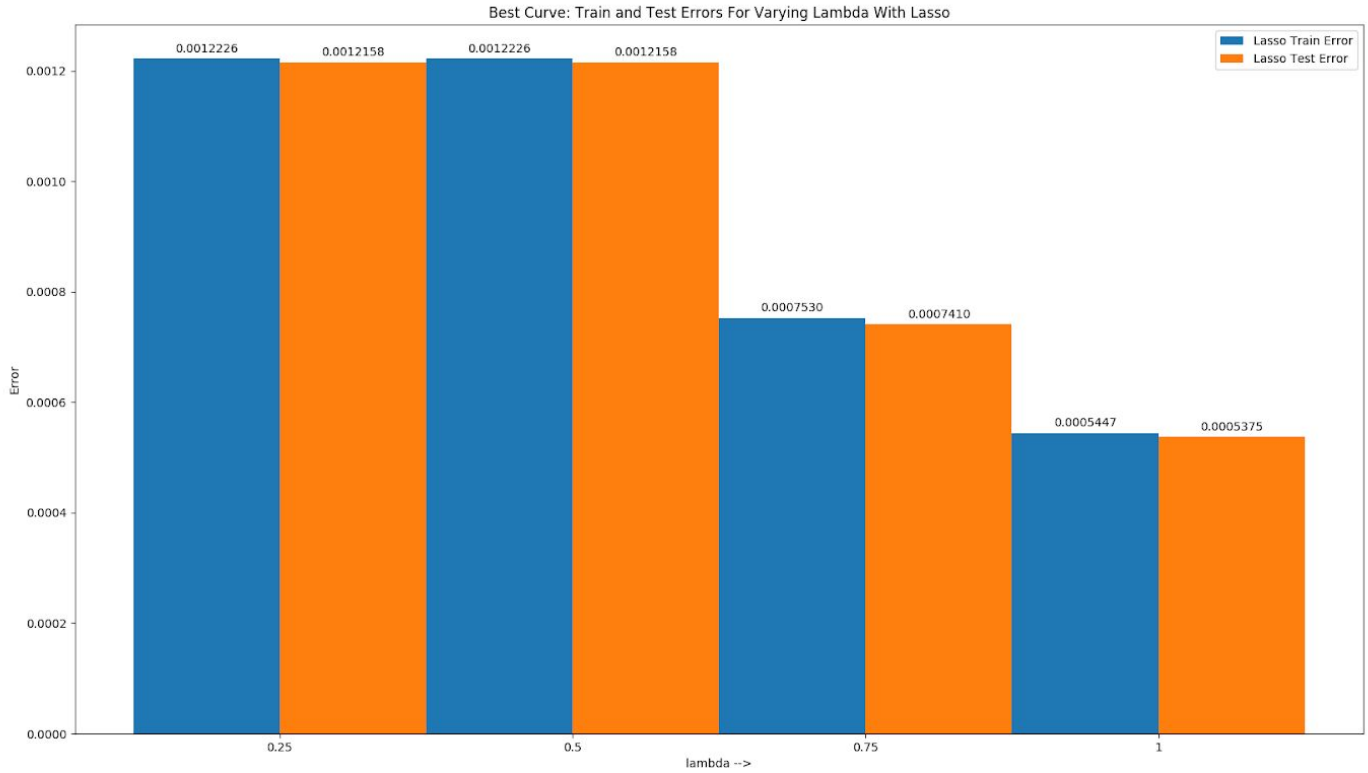
New Parameters: [0.91853589 -1.85952448]

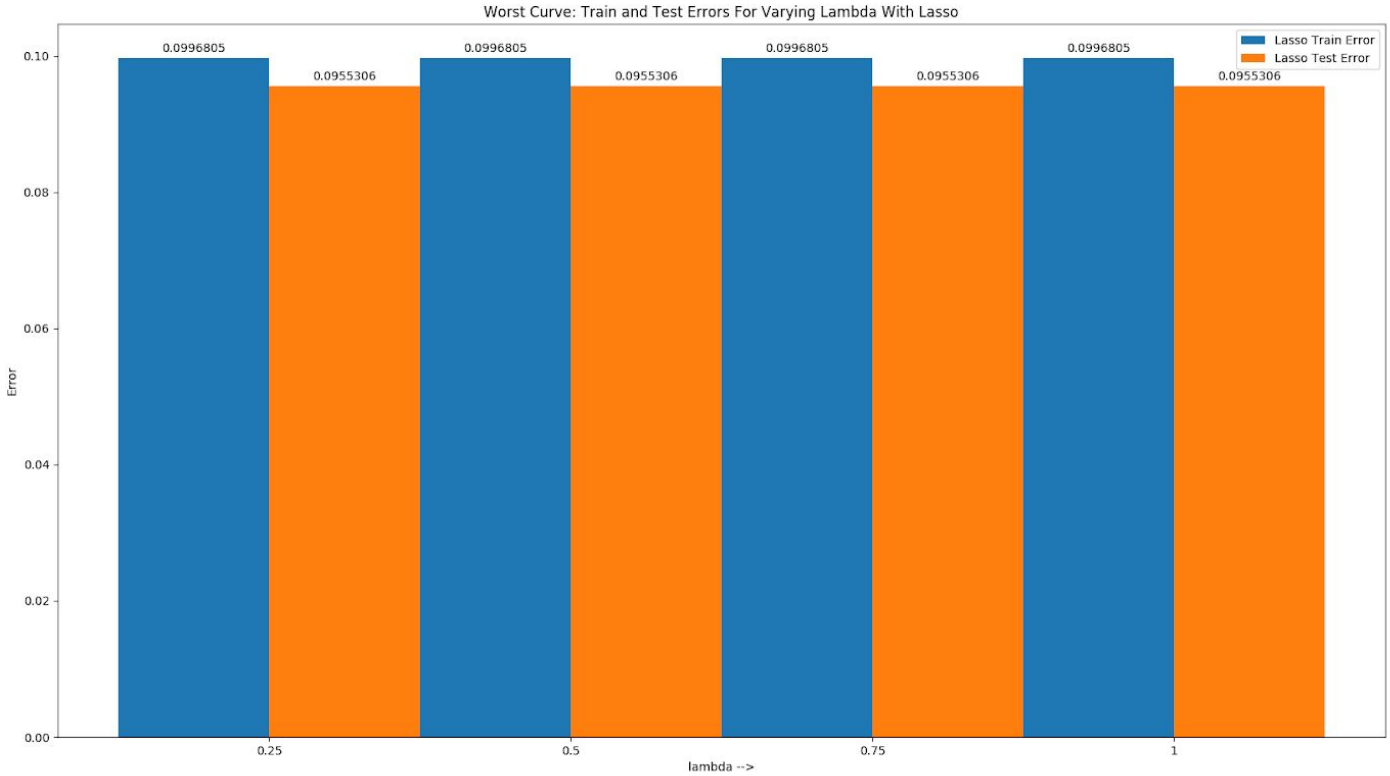
Ridge Error on Test Data: 0.10630989004657294

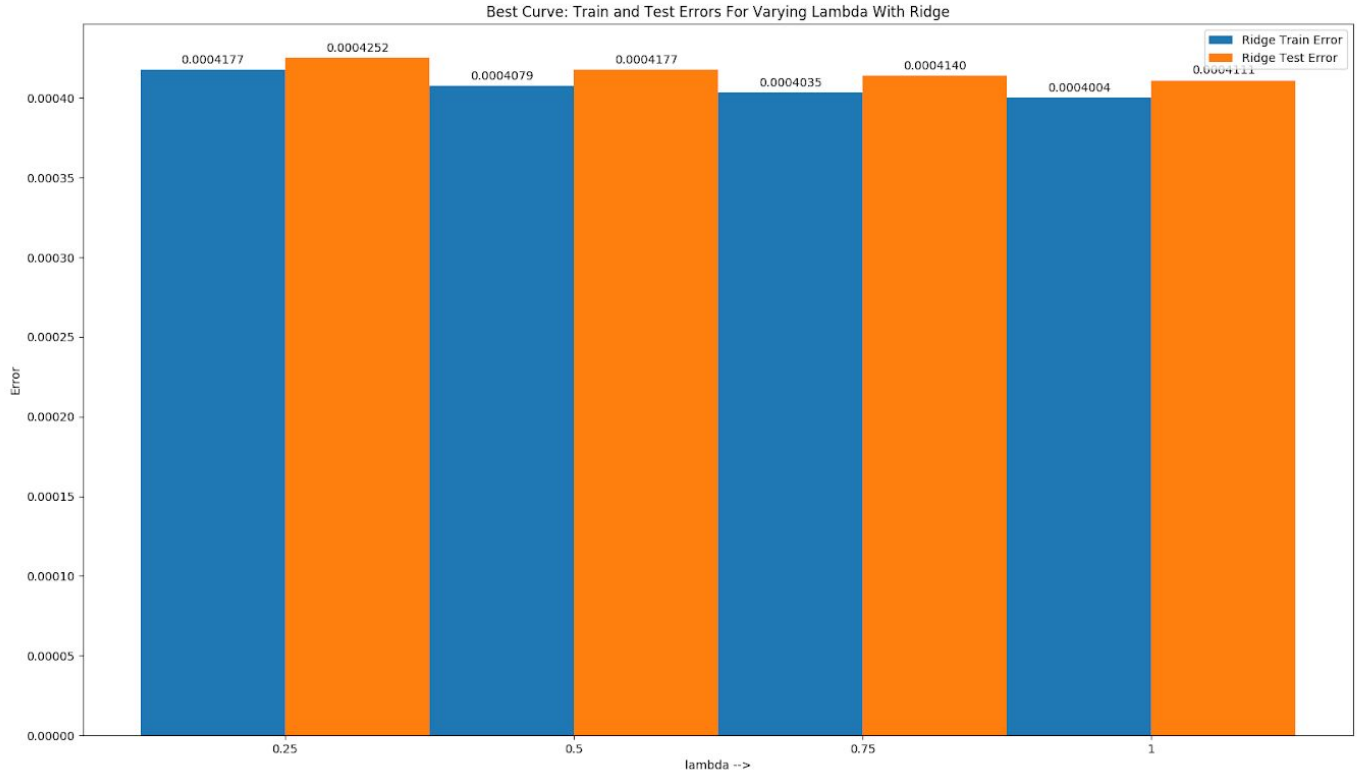
Squared Error on Test Data: 0.09555604135863971

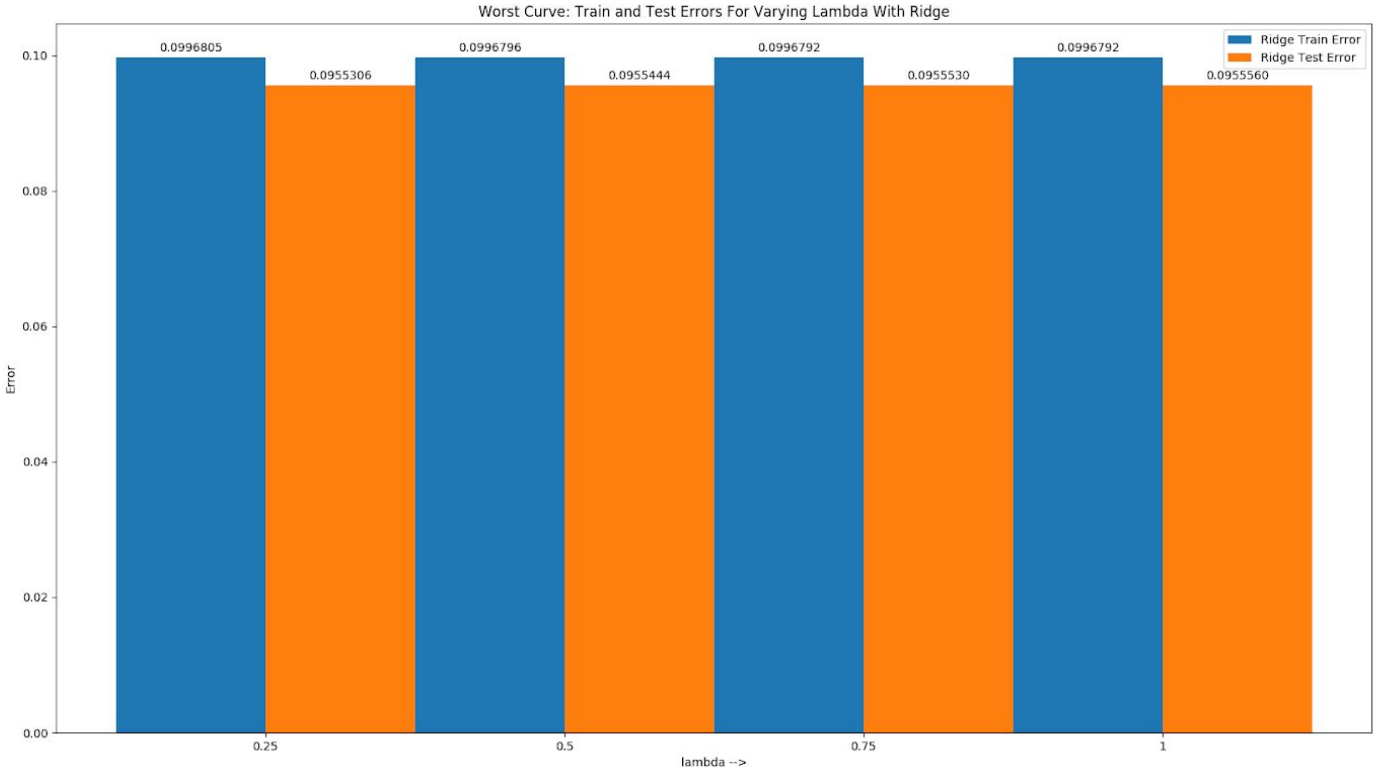
Below are the result graphs for varying lambda with best curve (degree 9) and worst curve (degree 2). The same are also available in the folder “result_plots”. For a better resolution image, please have a look into the folder.

I would prefer Lasso Regularisation with Lambda value 1 since it is then that we get the least error on the test data. The Lasso algorithm reduces the dependency on terms (such as constant) in order to make the curve fit better. We can clearly see this working when the coefficient of X^0 is reduced to -9.49355229e-03 in the last iteration of lasso over best curve with lambda value 1. We do not observe any particular high values that can be reduced by ridge and hence the change with ridge is minimal while lasso punishes the factors which have very less contribution to the actual data.


Best Curve: Train and Test Errors For Varying Lambda With Lasso
Lasso Train Error
Lasso Test Error
0.0012226
0.0012158
0.0012226
0.0012158
0.0007530
0.0007410
0.0005447
0.0005375
Error
0.0000
0.0002
0.0004
0.0006
0.0008
0.0010
0.0012
0.25
0.5
0.75
1
lambda -->


Worst Curve: Train and Test Errors For Varying Lambda With Lasso
Lasso Train Error
Lasso Test Error
0.0996805
0.0955306
0.0996805
0.0955306
0.0996805
0.0955306
0.0996805
0.0955306
Error
0.00
0.02
0.04
0.06
0.08
0.10
0.25
0.5
0.75
1
lambda -->


Best Curve: Train and Test Errors For Varying Lambda With Ridge
Ridge Train Error
Ridge Test Error
0.0004177
0.0004252
0.0004079
0.0004177
0.0004035
0.0004140
0.0004004
0.00040
0.00035
0.00030
0.00025
0.00020
0.00015
0.00010
0.00005
0.00000
Error
0.25
0.5
0.75
1
lambda -->


Worst Curve: Train and Test Errors For Varying Lambda With Ridge
Ridge Train Error
Ridge Test Error
0.0996805
0.0955306
0.0996796
0.0955444
0.0996792
0.0955530
0.0996792
0.0955560
0.10
0.08
0.06
0.04
0.02
0.00
Error
0.25
0.5
0.75
1
lambda -->

References:

The following sources were used for help during the completion of the assignment.

- https://hackernoon.com/practical-machine-learning-ridge-regression-vs-lasso-a00326371ece
- https://pandas.pydata.org/pandas-docs/stable/10min.html
- https://www.programiz.com/python-programming/matrix
- https://realpython.com/python-matplotlib-guide/
- https://towardsdatascience.com/one-simple-trick-for-speeding-up-your-python-code-with-numpy-1afc846db418
- CS60050 Slide 1 and 2