Assignment 1.1 Report

Name: Shubham Sarda Entry Number: 2018TT10958

Part B

Lambda is coming out to be “10” after 10-fold cross validation

Part C

Note 1: The general trend in all these experiments is to create many features (>300) and then using Lassolars (sklearn), find the top 300 features. These features are then used to perform usual Least Squares Regression to give predictions on test set as mentioned in the problem statement. The code for all the experiments can be found in **Assignment1.1.ipynb.** In **linear.py**, we manually create these top 300 features to build the model.

In the code for **Assignment1.1.ipynb**, it has been assumed that the directory containing the notebook also contains the “**data”** folder which can be downloaded from the following link:

https://www.cse.iitd.ac.in/~cs5170401/Assignment_1.zip

Note 2: All experiments were done for different values of lambda (regularization coefficient is LassLars:

lambda=[0.00001,0.001,0.004,0.008,0.01,0.014,0.02,0.03,1 ,3]

Higher values of lambda generally proved to give bad results

Experiments

1. I first did feature selection based on the default features. Since lasso lars normalizes the features, I used weights as a measure of feature importance. I sorted them in descending order of absolute value of weights to find the most predictive features:

Alpha:0.001 RMSE: 20935.55056133305 R2:0.6403946392264617

features selected= 30-> [25 11 30 22 24 23 20 6 8 9 21 10 1 27 2 28 12 18 13 14 5 17 16 26

15 4 7 19 29 3] → these are the indices of features sorted by their predictive power

{'APR Medical Surgical Description': 12726.074687436407, 'Length of Stay': 2616.1696112860704, 'Emergency Department Indicator': -2185.8834017598792, 'APR Severity of Illness Code': 1366.6863813857124, 'APR Risk of Mortality': -1251.3514738190868, 'APR Severity of Illness Description': -1025.2350971765966, 'APR MDC Code': 866.2824171886991, 'Age Group': -845.0000570649697, 'Gender': 600.3249665809576, 'Race': 329.5473077127016, 'APR MDC Description': -237.9187831443522, 'Ethnicity': -222.73018473685806, 'Health Service

Area': 156.78087990419712, 'Payment Typology 2': 133.2477195654222, 'Hospital County': -62.00810796467093, 'Payment Typology 3': 45.23987403799771, 'Type of Admission': 29.954670693089177, 'APR DRG Code': -26.39559375720693, 'Patient Disposition': -21.144120935446942, 'CCS Diagnosis Code': -18.6252606935117, 'Facility Name': 13.602830910425167, 'CCS Procedure Description': 9.005070761749067, 'CCS Procedure Code': -6.6423769295296555, 'Payment Typology 1': -3.2771574590734187, 'CCS Diagnosis Description': -2.1153788084998357, 'Facility Id': -0.7312493298161694, 'Zip Code - 3 digits': -0.6537651504777152, 'APR DRG Description': -0.1323084023420723, 'Birth Weight': 0.0575165308702115, 'Operating Certificate Number': 0.0013996077997524446}

Interpretation: The order of features is also interpretable since total cost would highly depend on whether its surgery or not, length of stay, emergency and illness severity indicators. It would also to some extent be correlated with the race and ethnicity due to the existing financial distribution based on these factors. The dependence would be very low on features like operating certificate number, patient disposition, birth weight etc.

2. Then I tried removing outliers based on z-score>3 using scipy library, but that further increased the error, so dropped this.

Alpha:0.1 RMSE: 21878.483943255895 R2:0.6072782373923614

3. Added x^2 terms for selected features of 1.

Square terms were added so that even regression could be tuned with higher order terms to introduce non linearity.

Alpha:0.001 RMSE: 19793.913

4. Created Polynomial Features using sklearn of order 2 for selected features to introduce more polynomial terms since introduction of square terms decreased the error in previous experiment.

Alpha:0.01 RMSE: 17937.27877359557 R2: 0.7359663830399313 features selected= 263

5. Removed birth weight, operation code number before polynomial creation

Added exponential terms for top 13 features after this

Alpha:0.008 RMSE: 17705.0727469637 R2:0.742747636026755 features selected= 282

6. Since exponential reduced error, therefore added exponential before poly creation and removed last 8 of selected features as less predictive

Alpha:0.008 Error: 17802.113510599396,0.7399187206822082 features selected= 279

7. Alpha:0.001 Error: 17927.34337376135,0.7362292467935065 features selected= 55->

Added OHE of ['APR Medical Surgical Description','Emergency Department Indicator','APR Severity of Illness Code','APR Risk of Mortality','APR Severity of Illness Description','Age Group','Race','Ethnicity','Health Service Area']]

This made it clear that OHE didn't help.

8. Then experimented by grouping by most predictive categorical variables

And adding mean total cost/ length of stay as features before polynomial creation. Experimented with various permutation and combination these categorical variables and achieved :

Alpha:0.01 Error: 17877.541991209255 R2:0.7377122091385717 features selected= 284->

9.

Since exponential gave good results, also experimented with sigmoid function and achieved:

Alpha:0.008 Error: 17806.89849321004,0.7397788701711685 features selected= 299

NOTE: In case features selected by lasso exceeds 300, I select top 299 predictive features based on their absolute value of weights.

10. Experimented with power transform of sklearn library to make data more gaussian like which is one of the assumptions of linear regression. Achieved:

Alpha:0.01 Error: 17801.441335938267 R2:0.7399407827107393 features selected= 299

11. Also experimented with quantile transform

Alpha:0.004 Error: 17890.19738955868 R2:0.7373462021002171 features selected= 188

12. Experimented with different permutation and combinations of power quantile and exponential function. Achieved:

Alpha:0.001 RMSE: 17713.27192137238 R2:0.7425087090274634 features selected= 299

13. Experimented with order 3 polynomials

Alpha:0.001 RMSE: 17813.27192137238 R2:0.7325087090274634 features selected= 299

14. **BEST MODEL—**

RMSE: 17409.56784671678 R2:0.7512472686106204 features selected= 298

Removed birth weight and operation certificate number, and added more exponential features of

['Length of Stay','APR Medical Surgical Description','Emergency Department Indicator','APR Severity of Illness Code','APR Risk of Mortality','APR Severity of Illness Description','APR MDC Code','Age Group','Gender','Race','APR MDC Description','Ethnicity','Health Service Area']]

After polynomial creation, again added those 2 features which were removed to ensure higher order terms of less predictive features were not made

Total 906 features created and finally Lasso used for feature selection. This results in 298 non zero coefficient feature

Since lasso was only supposed to be used for finding the best features, these features were then manually created in **linear.py** to ensure less train time.