COL341
A1.1 Report

Ishaan Watts
2019PH10629

Part B

Best Regularisation parameter reported after 10-Fold Cross Validation --> **10**
Metric used --> 1 – (L2 norm of residual errors / L2 norm of Y_test)

Part C

For feature engineering I used different methods on the smaller dataset train.csv

Ideas tried:

Split the data in the ratio 90:10
Metric --> R2 score from LassoLars.score()

1. Calculated the metric after normalising the dataset using mean and standard deviation ~ **60.5**
2. Split the data according to number of unique values in each column into numerical and categorical columns.
Tried different values , i.e. 20,10,6, of the maximum threshold for unique values in a column for selecting that column as categorical.
Metric improved to ~ **62.5** in each case.
3. After this further added polynomial features with maximal degree of 2. Now since features were 1540 for the threshold of 10, used lasso to count features with 0 weight and almost all features were removed and gave the metric as ~ **61.5.**
4. Tried using polynomial features only for numerical columns and not categorical also gave the metric as **~ 62.5.**
5. Now normalised the data and just constructed polynomial features without dividing the dataset and got 495 features. This gave metric score of ~ **70.8**.
Concluded that one hot encoding was not increasing the metric along with polynomial features and hence did not use it.
6. Further tried handpicking features by first taking all 30C2 combinations and adding only 1 feature at a time seeing the increase in score. Length of stay affected the score the most. Took the top 80 combinations. Total features 110.

Then applied polynomial till degree 4 and again handpicked top 80 features.
This resulted in a total of 190 and lasso removed 17 more.
This gave metric of ~ **72.7** – best so far. Unfortunately when averaged it over all 10-folds it resulted to be ~ **66** and hence dropped this idea.

Final Method used:

Normalised the dataset and then constructed polynomial features.
Total of **495** features and using lasso brought it down.

Final testing on train_large.csv using Google Colab:

Used 10-fold to choose the best alpha for Lasso --> **0.0003**
Values tried = [0.0001, 0.0003, 0.001, 0.003, 0.01]
Final features --> **296**
Final metric using 10-fold and best lambda --> **0.6804439503434712**
Columns to be dropped mentioned in feature_selection.py

Further calculations:

Imported R2 score from sklearn.metrics.
Calculated W using Moore-Penrose after transforming the data using selected features.

1. Trained on train_large.csv and calculated R2 score on train.csv --> ~ **68**
1. Trained on train.csv and calculated R2 score on train_large.csv --> ~ **67.7**