Rowan Yasser 8010

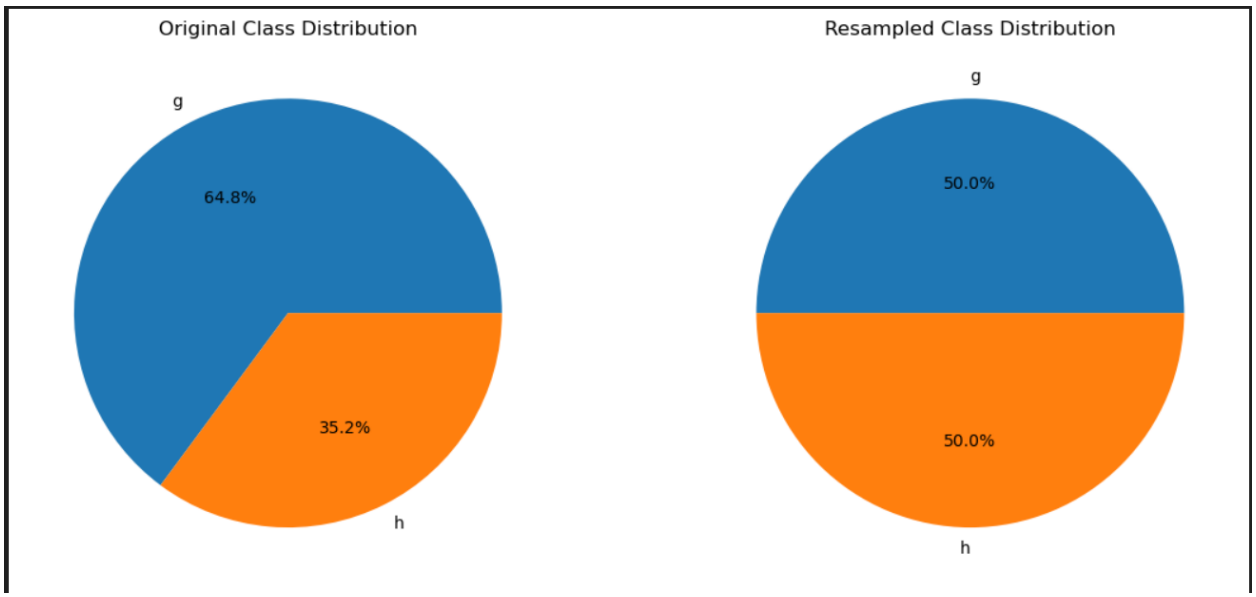
Sara Elmassry 8236

Mazen Elsayed 8247

Assignment 1 Machine Learning

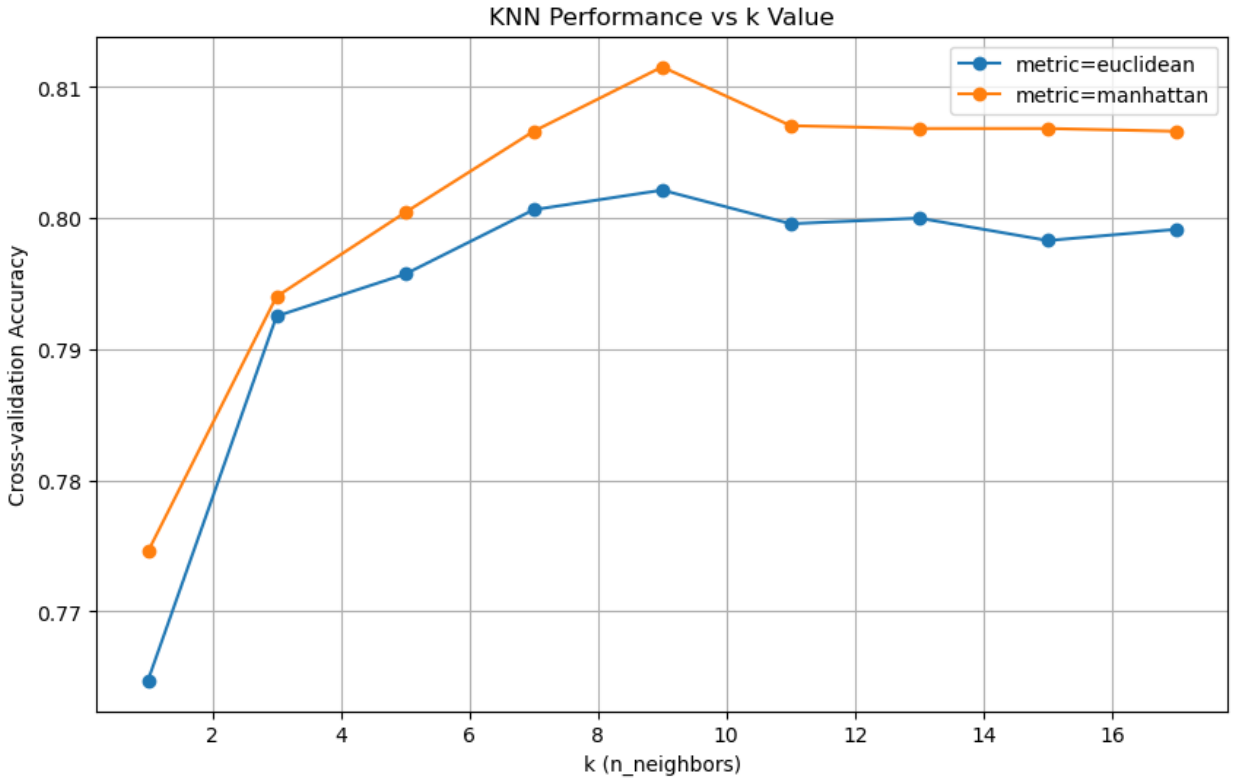
Part 1) Classification:

1. First, we under sampled our data to balance the two classes, the gamma was 64.8% of the data and the hadron was 35.2%, after resampling the two classes each class became 50% (6688 samples for each class).



2. We made a correlation matrix to check the relationship between the features and the target.
3. We separated the gamma data and the hadron data from each other.
4. For each separated data, (gamma and hadron), we split 70% of the data for the training model and 30% for the testing data. Then we took the 30% testing data and split it again to 15% validation and 15% testing.

5. Then we combined the gamma and hadron training data together as well as the testing and validation.
6. We scaled our data using standard scaler and then we fit the data.
7. A grid search was used to output the best hyper parameters we can use to expect the best results: ‘metric': 'Manhattan', 'n_neighbors': 9.
8. The figure below shows that Manhattan performed better and shows the best K.

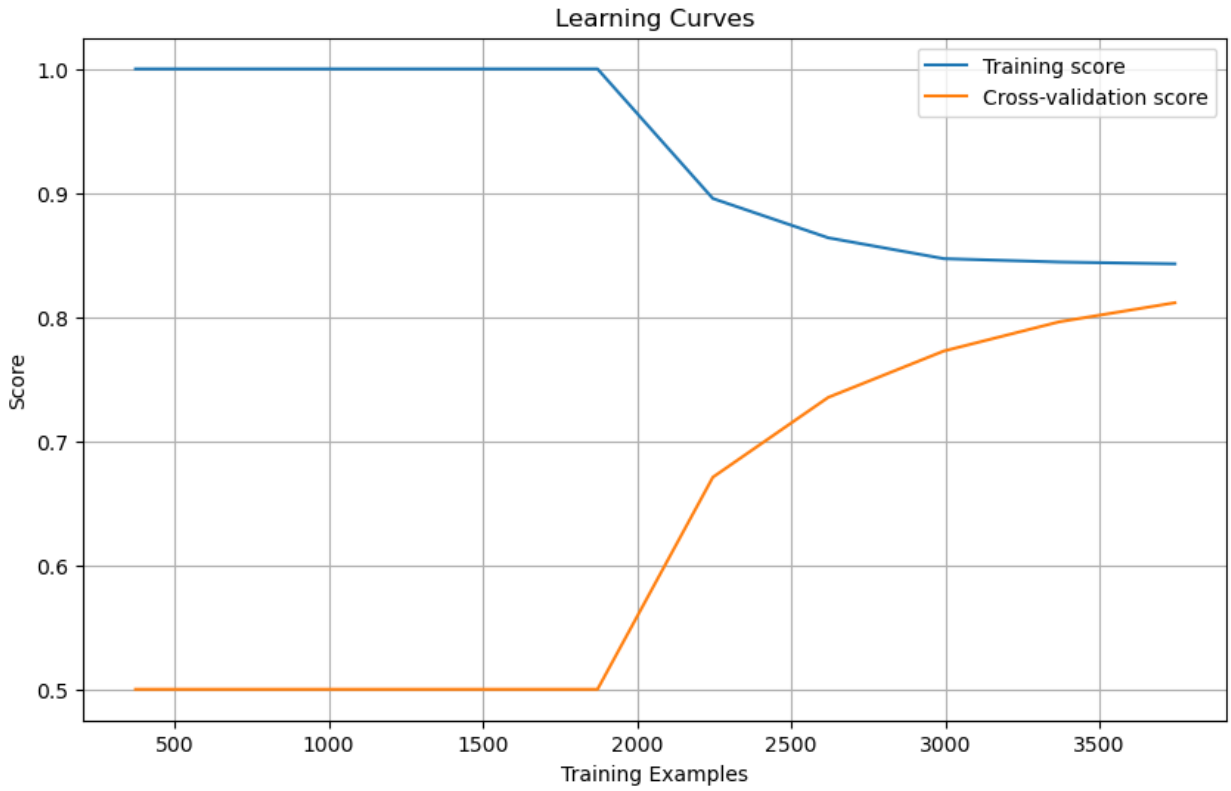


9. Then we used the best k which is 9 and used it for the testing data.

10.The learning curve helps us understand how the model's performance changes as we increase the amount of training data.

What the curves represent:
The blue line (Training score) shows how well the model performs on the training data The orange line (Cross-validation score) shows how well the model performs on validation data The x-axis shows the number of training examples used the y-axis shows the performance score (accuracy in this case)

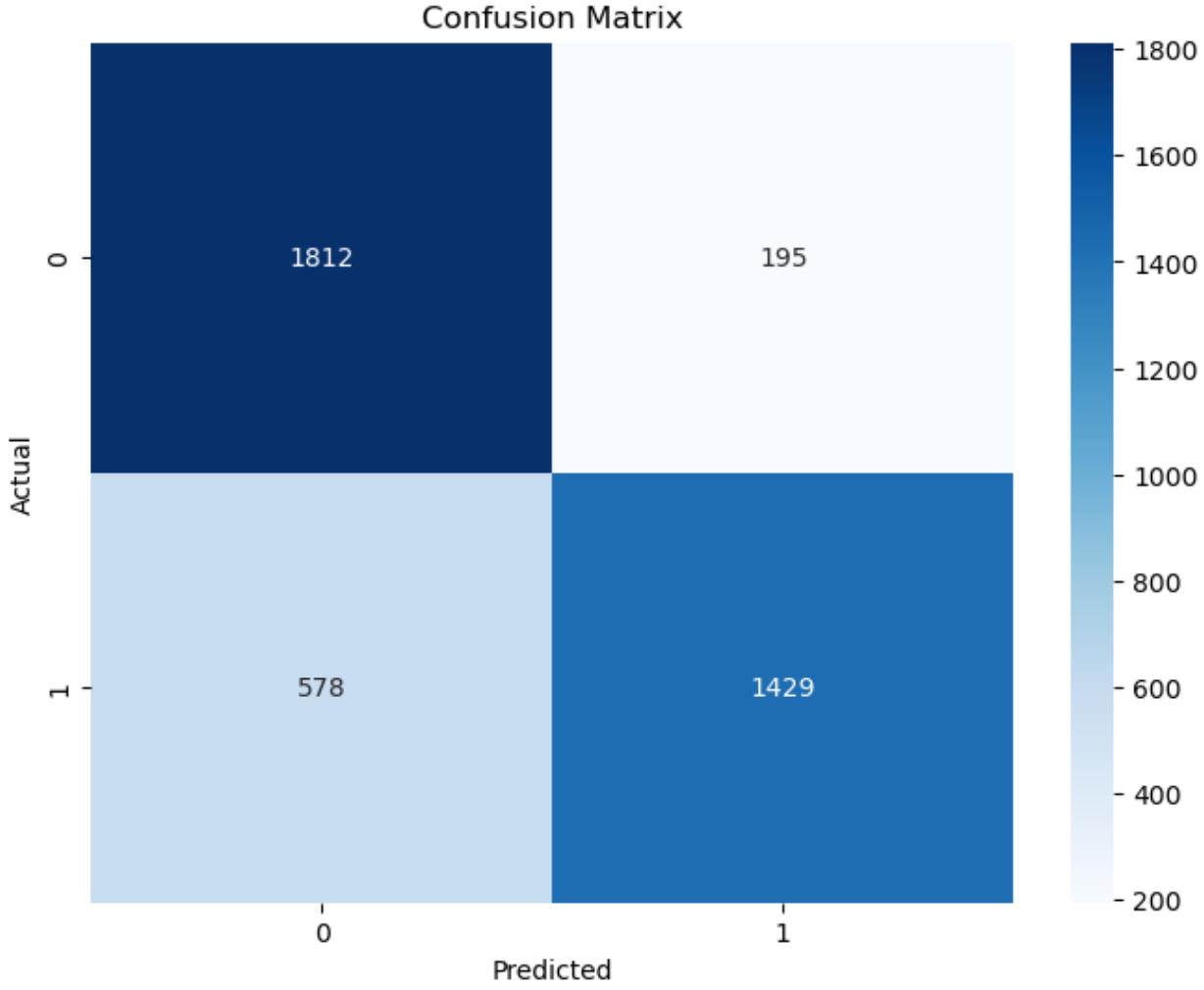
If the training score is much higher than the validation score, the model is overfitting If both scores are low, the model is underfitting If both scores are close and high, the model is performing well



11. We checked the performance

```
Accuracy: 0.8074240159441953
Precision: 0.8190425520951419
Recall: 0.8074240159441953
F1-score: 0.8056546528713431
```

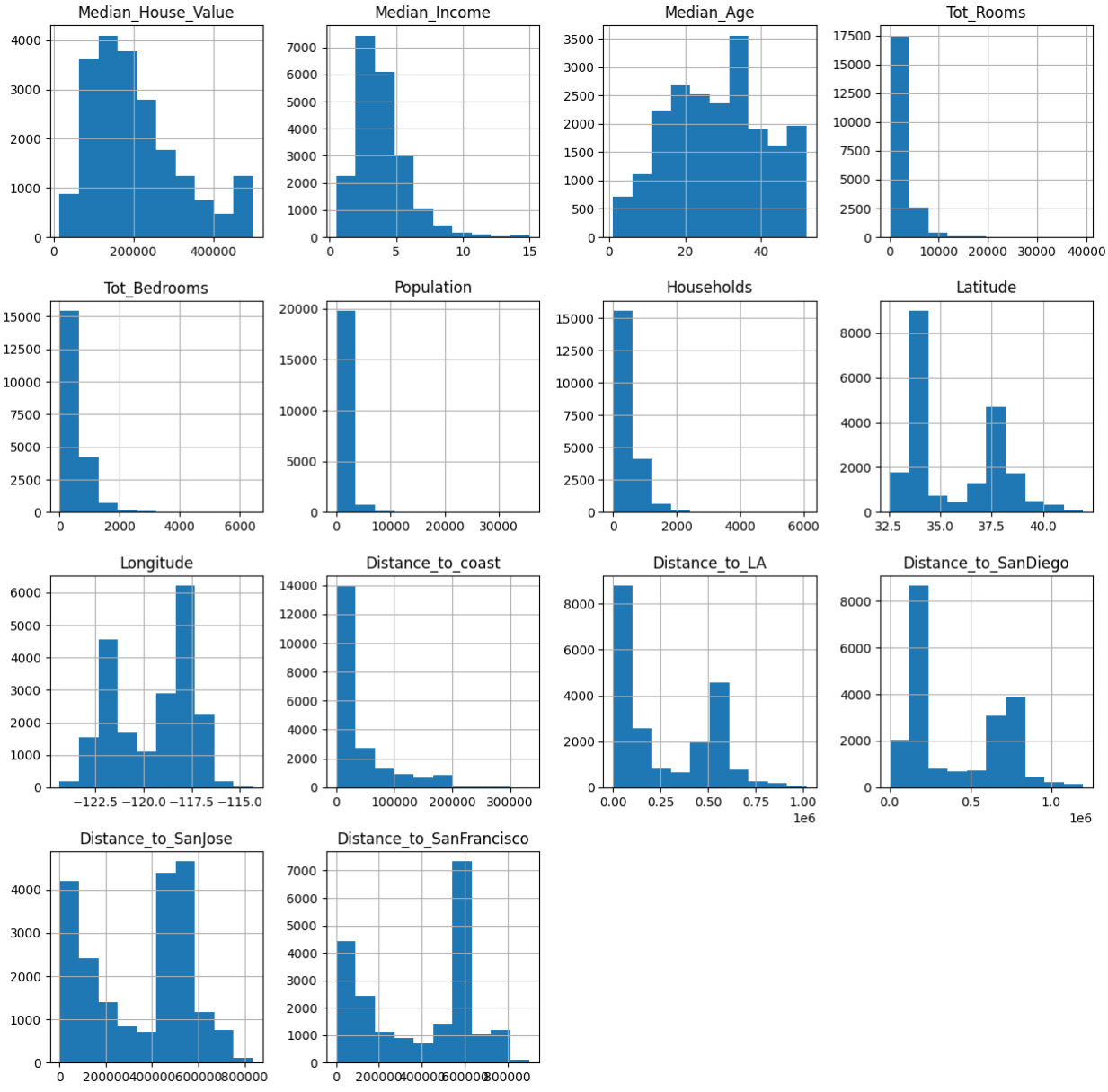
12. Confusion matrix between the y actual and the y predicted:



The confusion matrix reveals that our model demonstrates strong performance, correctly classifying 3241 samples with only 773 missclassied samples, the model achieves a high degree of accuracy, showing its reliability in distinguishing between gamma and hadron classes across most instances.

Part 2) Regression:

1. Visualizing the features using histogram



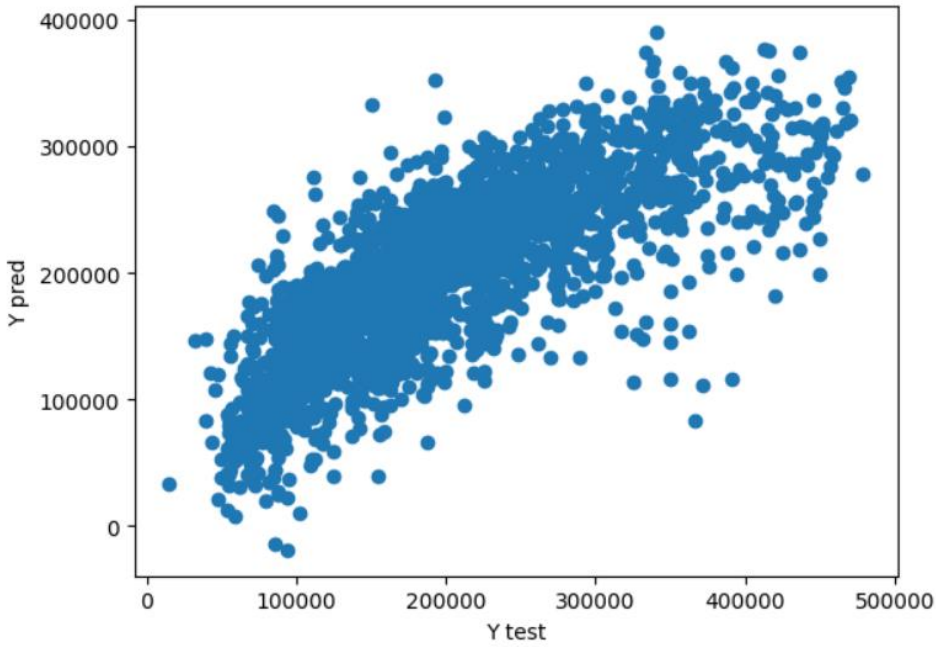
2. We drew a box plot to see the outliers in each feature then calculated how many outliers are there in each feature:

	Feature	Outlier Count
0	Median_House_Value	19569
1	Median_Income	19959
2	Median_Age	20640
3	Tot_Rooms	19353
4	Tot_Bedrooms	19358
5	Population	19444
6	Households	19420
7	Latitude	20640
8	Longitude	20640
9	Distance_to_coast	18264
10	Distance_to_LA	20640
11	Distance_to_SanDiego	20640
12	Distance_to_SanJose	20640
13	Distance_to_SanFrancisco	20640

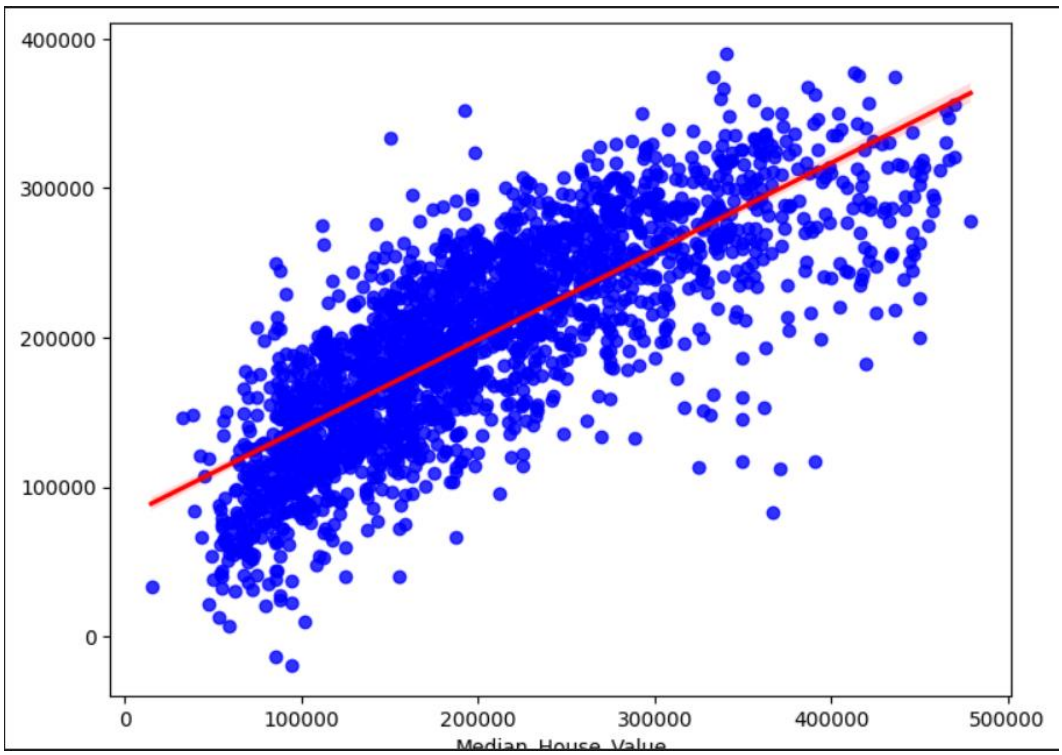
3. Then we reduced the outliers using the quantile method which reduced the outliers by approximately 5000 data points for each feature

4. After removing the outliers, we split the features and our target column then split the data into 70% training data and 30% testing data which was then split again into 15% test data and 15% validation data

5. The data then was scaled using standard scaler and then fit the data for the model



Plotting the data between the Y predict and the Y test.



Plotting the linear regression model line.

6. We used the lasso and the ridge model to test if it is going to fit the data better, but the results were nearly the same.

7. We calculated the mean square error and the mean absolute error:

```
Mean Squared Error LR:  3307228849.0476165
Mean Squared Error Lasso:  3307224286.6204267
Mean Squared Error Ridge:  3307279847.2781982
Mean absolute error LR:  42688.20200800906
Mean absolute error Lasso:  42689.38903502964
Mean absolute error Ridge:  42691.12100005777
```

8. Then we tried the same models in the same way using columns of dataset except the most uncorrelated columns (Total bedrooms, population and Households) to see if the performance will be better.

9. But the error increased

```
Mean Squared Error LR:  3913803978.4021273
Mean Squared Error Lasso:  3913791250.9178443
Mean Squared Error Ridge:  3913817347.6407666
Mean absolute error LR:  46942.383558058646
Mean absolute error Lasso:  46943.34262435514
Mean absolute error Ridge:  46945.16537074284
```

So, it is better to use all the columns of the data as features.