Housing Price Prediction

In this project, we are going to do data analysis on the housing price in the Boston area by using a linear regression model. And the files included are one dataset CSV file and one ipynb Jupyter Notebook file.

DATASET FILE:

By applying a linear regression model to the data set we collected from the Internet[1], this multi-factor regression model would be created and trained based on a part of the dataset. After training, this model was anticipated to provide a relatively precise housing price prediction.

	Regressor Code	Regressor
1	CRIM	per capita crime rate by town
2	ZN	proportion of residential land zoned for lots over 25,000 sq.ft.
3	INDUS	proportion of non-retail business acres per town.
4	CHAS	Charles River dummy variable (1 if tract bounds river; 0 otherwise)
5	NOX	nitric oxides concentration (parts per 10 million)
6	RM	average number of rooms per dwelling
7	AGE	proportion of owner-occupied units built prior to 1940
8	DIS	weighted distances to five Boston employment centers
9	RAD	index of accessibility to radial highways
10	TAX	full-value property-tax rate per $10,000
11	PTRATIO	pupil-teacher ratio by town
12	B	1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town
13	LSTAT	% lower status of the population
14	MEDV	Median value of owner-occupied homes in $1000's

[1] https://www.kaggle.com/datasets/vikrishnan/boston-house-prices/code?datasetId=1815&sortBy=voteCount

NOTEBOOK FILE:

We have made two models, one of these dropped the median housing price and another one did not. The reason why we dropped the MEDV in the first model is because this regressor stands for the actual price, which is the standard for comparing the difference between the prediction and the actual price.

Result of the regression model:

R^2: 0.7180617917082988

Adjusted R^2: 0.7123516507808718

MAE: 3.5229299314858222

MSE: 24.49293182486547

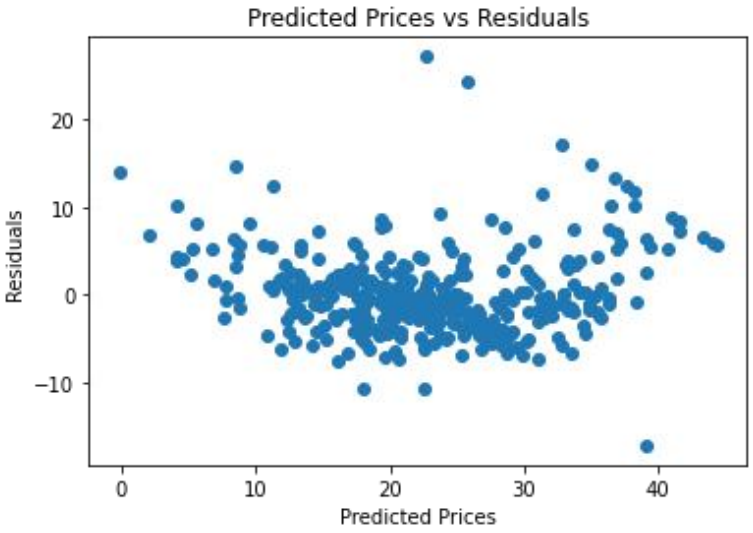
RMSE: 4.949033423292418

From the result of the multi-factor regression model, we can see that the R^2 and the adjusted R^2 is close to 72%, the number shows that around 72% of the total dataset could be explained by our trained model, which is a relatively good result.

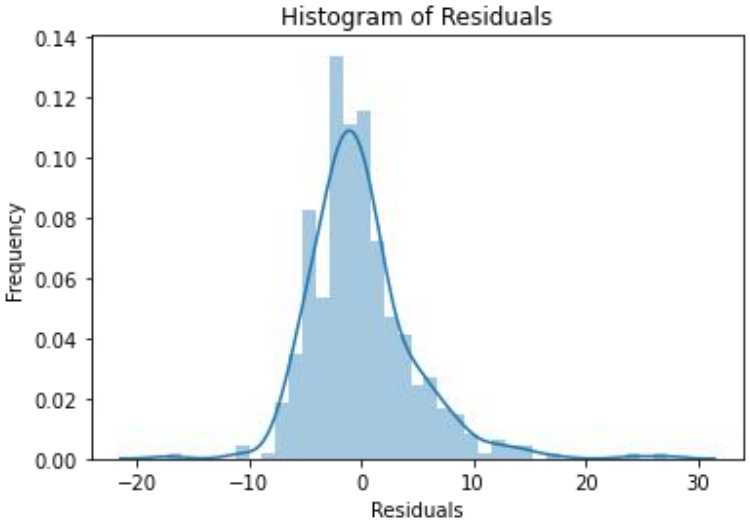
Graph of the result of the prediction and actual price:

After training, here is the result of the prediction of our model. The X-axis shows the actual housing price and the Y-axis shows the predicted price. If the prediction perfectly matched the actual housing price, it would show a perfect straight line with the X-axis equal to Y-axis. In this graph, we can see that though it did not show a perfect straight line, there is still a clear and recognizable trend, a trend that shows that the prediction matched relatively well with the actual result.

Graph of residuals analysis



In the two residual analysis graphs above, we can see that most of the residuals fell in the range from -10 to 10 and showed a clear shape that there are only few outside outliers. This trend proves the trustworthiness of our multiple regression model.



CPMPARE DIFFERENT REFRESSION MODELS

In addition to using the multi-regression model, we also used the random forest regressor for sample fitting. A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The following are the results of the random forest regressor regression model.

R^2: 0.9774135927379937

Adjusted R^2: 0.9769561465149657

MAE: 0.8868613861386134

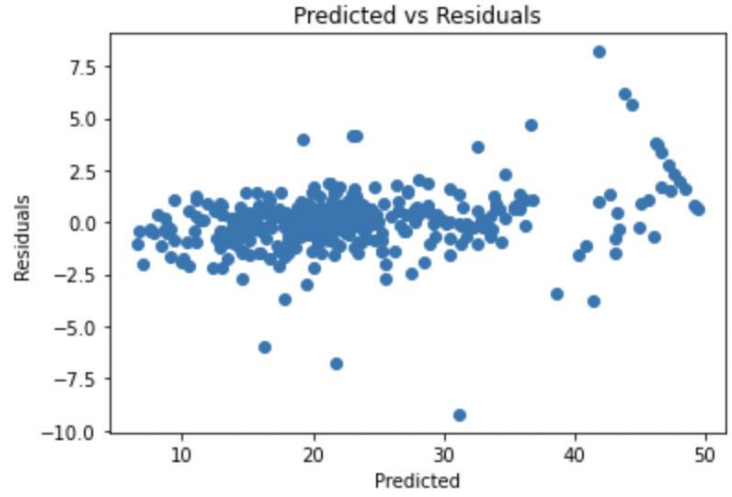
MSE: 1.9621580792079198

RMSE: 1.4007705305323637

Compared with the multi-regression model we did earlier, we can see that the R square(0.9774) of random forest regressor is much larger than the R square(0.7181) of multi-regression model. Which means that the result of the random forest regressor is more accurate than multi-regression model.

We can also see that the MSE of a random forest regressor is also smaller than that of multi-regression model. As we know, mean squared error (MSE) is the average of the summation of the squared difference between the actual output value and the predicted output value. Our goal is to reduce the MSE as much as possible.

In this graph, we can see a perfect line with the X-axis almost equal to the Y-axis, which means that the prediction matches the actual house price perfectly.



In the residual analysis graphs above, we can see that most of the residuals fell in the range from -2.5 to 2.5. This trend proves the highly trustworthiness for our random forest regression model.

COMPARE THE RESULTS TO OTHER DATA SETS

We used the multi-regression model to analyze the housing price data in California, and set the same test size and random state for both data sets.

Result of the regression model:

R^2: 0.640094792430529

Adjusted R^2: 0.6399203246573631

MAE: 50626.79448983832

MSE: 4811134397.884201

RMSE: 69362.34135238084

CONCLUSION

In this project, we used a linear regression model and a random forest regression model to analyze the data on house prices in the Boston area. The results prove that our predicted house prices match perfectly with the actual house prices. In addition, we used California house price data for comparison testing. Through this project, we learned how to use Python databases for modern data science, including Matplotlib, NumPy, Pandas, Scikit-learn, and Seaborn, to visualize data.