Predicting House prices using Linear Regression

Nigel Haim N. Sebastian

Febuary 5, 2024

I. Introduction

Linear Regression is the study of relationships between dependent variables and independent variables. From the business and real estate perspective, it helps in decision-making by evaluating trends and forecasting.

It is a mathematical aspect implemented in machine learning to interpret the output coefficient easily. It is also known for its simplified complexity compared to other algorithms. Linear Regression is a fundamental tool for analyzing the relationships of the variables. However, it over-simplifies real-world problems by assuming a linear relationship.

Despite its simplicity, it is prone to miscalculations if the data has any outliers. These can significantly affect the results by changing the integrity and character of variables. Linear Regression always assumes a straight-line relationship.

II. Methodology

Overview: This study provides a better understanding of linear regression applications in machine learning. The model identifies house prices of India's growth in its housing sector. The model's training and testing are done through a provided dataset with a split of 80 for training and 20 for testing. The model is made through the use of different packages in Python.

The findings of this study consist of multiple tests and different methods for implementing ways to improve the model. Range from dropping outliers, mathematical calculations in changing the values, one-hot encoding, and ordinal encoding to represent categorization and other pre-processing methods, as well as adding and removing features.

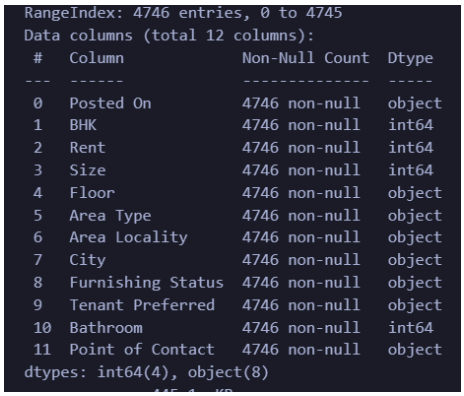

Figure 1 The Initial data set variables

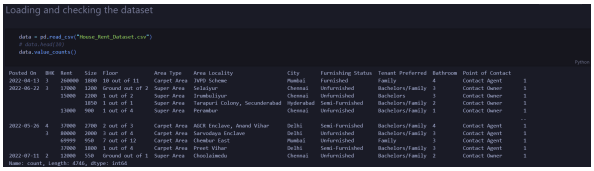

Figure 2 Data Values

Research Design: The dataset values are stored in a .csv file and loaded into a data frame. It consists of 4746 rows (values) and has 12 Columns (variables) with No null values. As observed in Figure 1, all variables do not have any null values. The data frame consists of multiple datatypes that need to be converted into integers for the model.

Check the graphs and observe how each feature affects the house prices. This can be done through the use of Pandas.Dataframe.hist and seaborn.pairplot. Every column of the dataset is a possible feature of the model.

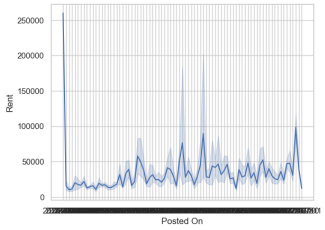

Figure 3 The relationship between the Date Posted and price

Observing the values of each column (feature) in Figure 2. the initial steps in bullet form can be done as follows:

- **Posted On** - Check for any relationship between the Date and the house prices.
- **Floor** - Convert strings into integers and determine a relationship of the Floor to the total number of floors.
- **Area Type** - Apply one-hot encoding
- **City** - Apply one-hot encoding
- **Furnishing Status** - Apply ordinal encoding
- **Tenant Preferred** - Apply one-hot encoding
- **Point of Contact** - Apply ordinal encoding

Afterward, methods of dropping outliers, normalizing the distribution, and applying techniques from the internet will be done. Each change will be observed through the Quantitative Evaluation. Through the R^2 and visualization of a scattered plot. The initial R^2 for this study is 0.32.

III. Experiments

Date and Prices relationship: Since the posted dates are limited to a year, no trend can be determined based on the sorted dates seen in Figure 3. The column date posted was excluded as a feature in the data.

One hot encoding: The features City, Tenant Preferred, and Point of Contact only consist of three to five possible values. Applying One hot encoding categorizes the true or false values based on the category given from the Data set. This increases the R^2 to 0.52.

Different Scalers: The model was also tested to use three different scalers. As seen in Table 1, all scalers resulted equally; therefore any scaler for this case can be used.

Scaler	R^2
Standard Scaler	0.57
MinMaxScaler	0.57
RobustScaler	0.57

Table 1: Different scalers and their corresponding R^2

Ordinal Encoding: Similar to One hot encoding, Ordinal Encoding was applied on Furnishing status where the values were ranked on Furnished, Semi-Furnished, and Unfurnished. This method did not improve the R^2 but was still included to ensure the model's integrity.

Ratio of the Floor and the total of floors: The floor column contains the values of a string. Applying the equation of dividing the floor number and the total number of floors. The data is also checked to see if only one value and the Dividend are more significant than the divisor. These will be dropped from the data set. They are resulting in a decrease in the number of rows to 4732. Ground, Upper Basement, and Lower Basement values are substituted with corresponding values (Table 2). Upon getting the Floor ratio, the distribution coefficient increased to 0.57.

Floor	Substituted value
Ground	1
Upper Basement	2
Lower Basement	3

Table 2: Corresponding values substituted for non-integer values

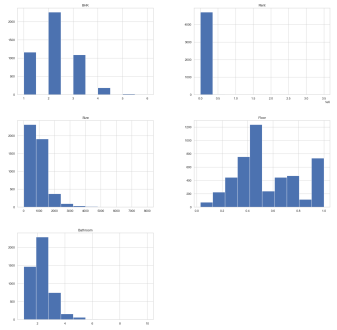

Figure 4: Histogram of BHK, Rent, Size, Floor, and Bathroom

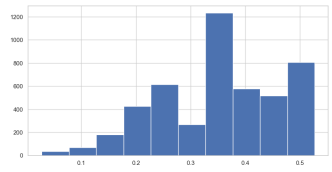

Figure 5: Applied Logarithmic function on Floor data

Data Distribution: The bar figures of BHK, Rent, Size, and Floor do not show a normal distribution. Therefore, transform the data through logarithmic transformation to determine the normal distribution. Applying the formula to the Floor figure maintains the R^2 at 0.57 (Figure 5). Since the Bathroom and Size do not contain any float. It decreases the R^2 to 0.51.

Logarithmic transformation of Prices: As seen in Figure 6, the result of the model with the R^2 of 0.57, the scatter diagram shows that the model predicted some negative prices. Upon checking the data set, no house prices have negative values. Applying logarithmic transformation on the Prices exponentially increases the R^2 to 0.84. However, checking the graph in Figure 7 shows that the prices did not start from 0. This invalidates the method of normalizing the feature.

Identifying and Dropping price outliers: Visualized by Figure 8, the findings identified that one of the causes of the negative predicted prices is the outliers. Some prices are a long distance from their neighboring prices. It is then tested that three different thresholds filter out the data. Any price exceeding the threshold will be dropped. However, as

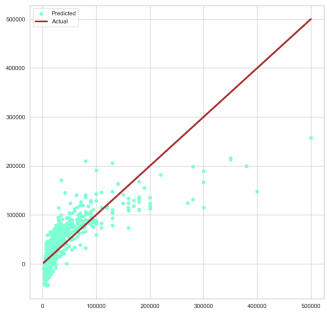

Figure 6: Predicted Negative Prices

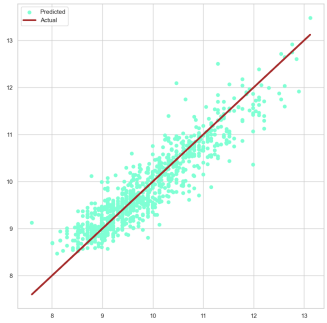

Figure 7: Logarithmic transformed prices

seen in Table 3, the number of dropped values increases as the Price threshold decreases. Despite a huge drop in data, the R^2 increases.

Price Threshold	Number of dropped values	R^2
150000	198	0.65
120000	264	0.69
100000	318	0.70

Table 3: Corresponding values substituted for non-integer values

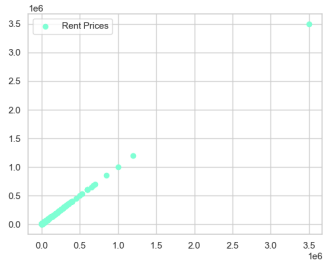

Figure 8: Scattered plots of all prices

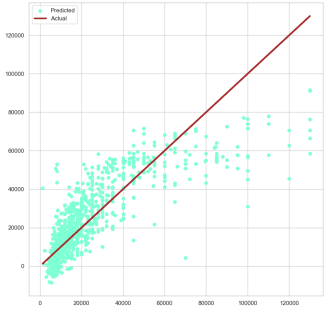

Figure 9: Price predictions with R^2 of 0.65

IV. Results and Analysis The final version of the model has a R^2 of 0.65 with an MSE of 178731484.25. However, dropping too many values could affect predictions since the case is a rapid growth of India's housing industry. The preprocessed features also added a positive change to the results. However, dropping rent prices more significantly than the 150000 threshold is necessary to improve the results exponentially. The tradeoff is losing more data. Logarithmic transformation also helps with some features, but not the house price. It can be used to address skewed data. However, it cannot be used to mitigate outliers.

V. Conclusions and Recommendations

Conclusions

With the findings provided, this study concludes that linear Regression relies on the accuracy and how each value is close to each other on the dataset. It also depends on outlier detection to have a good coefficient of determination and obtain a smaller MSE since the study only had a four-thousand-row data set. It is necessary to drop data to get a better result in training and testing the model.

Overall, the model's prediction performs well on the 100000 price threshold. Any data greater than this threshold slowly disrupts the integrity and price flexibility of the model.

Recommendations

The Linear Regression model can predict rent prices. However, its setbacks cannot predict high prices. It is recommended that based on the features that will be inputted, the model is expected only to indicate a smaller price. It is also recommended to look at different outputs and graphs when training the linear regression model. The R^2 cannot be relied on alone. For example, applying the logarithmic transformation on the price states that the model's performance exponentially improves; however, based on the results on the scatter diagram, it renders the prices in a different position, rendering the model unreliable.

Future experiments

Observing the data set, the study has excluded Area Locality as a feature since its values consist of different strings that one hot encoding or ordinal encoding cannot handle. Clustering the strings before including the feature in the database is recommended. However, it is not stated in the study of the attempted experiment since, with the lack of duration for this feature, clustering the strings takes five minutes for every runtime for the model.

VI. References

Andy. (2019, May 12). Logarithmic Transformation in Linear Regression Models: Why and when. DEV Community. https://dev.to/rokaandy/logarithmic-transformation-in-linear-regression-m

Feng, C., Wang, H., Lu, N., Chen, T., He, H., Lü, Y., Tu, X. (2014). Log-transformation and its implications for data analysis. PubMed. https://doi.org/10.3969/j.issn.1002-0829.2014.02.009

GeeksforGeeks. (2023, March 30). ML Advantages and Disadvantages of Linear Regression. https://www.geeksforgeeks.org/ml-advantages-and-disadvantages-of-linear-regress

GetRegressionive predicted values after linear Regression. (n.d.). Cross Validated. https://stats.stackexchange.com/questions/145383/getting-negative-predicted-values-after-linear-re

Ghosh, B. (2022, June 24). The magic of linear regression model. https://www.linkedin.com/pulse/magic-linear-regression-model-bhagyashree-ghosh#:~:text=Linear%20regressions%20can%20be%20used,forecast%20sales%20in%20future%20months.

Schneider, A., Hommel, G., Blettner, M. (2010). Linear Regression analysis. Deutsches Arzteblatt International. https://doi.org/10.3238/arztebl.2010.0776