Can You Predict House Prices?

Laboratory Exercise 1

Limeta, Sara Franchesca O. | 3CSC

I. Introduction

In the realm of real estate, the manual pricing of properties presents a significant challenge, given several factors involved, including location, size, amenities, economic indicators, and more. Additionally, these factors are often assessed subjectively, necessitating a more data-driven approach. As India's housing sector continues to grow, the diversity of housing variations further complicates this issue. To address this problem, a model called linear regression is used as a viable solution to predict house prices which is crucial for decision-making.

Linear regression, a statistical technique, is well-suited for forecasting house prices due to its ability to establish a linear relationship between independent variables and dependent variables. By scrutinizing the data, the model identifies patterns and formulates a predictive equation.

In conducting the data exploration, various steps are undertaken to understand the complexity of the data. This includes identifying missing values, searching for inconsistencies, examining the data's structure by analyzing its data types, and reviewing general information about the data to have an overview of what variables will be used in the experiment. Furthermore, Exploratory Data Analysis involves assessing histograms to explore the relationship between the variables and analyzing outliers are also included.

II. Methodology

The dataset comprises 4,746 entries of property listings, which include variables such as the Posting On (date of when the listing was posted), BHK (Bedroom, Hall, and Kitchen), Rent, Size (in square feet), Floor, Area Type, Area Locality, City, Furnishing Status, Tenant Preferred, Bathroom, Point of Contact. The tools utilized are Jupyter Notebook and libraries, including Numpy, Pandas, Matplotlib, and Seaborn.

2.1. Data Exploration

After identifying missing values and inconsistencies, different visualization tools were used to examine different types of data. A histogram was utilized for assessing numerical values, whereas a count plot was used to visualize categorical values. This process is helpful in determining the frequency distribution within each variable and identifying potential outliers. To delve deeper into the analysis, a correlation matrix was used to observe the relationships between numerical variables. To visualize the matrix, the heatmap was generated through the Seaborn library.

2.2. Preprocessing

2.2.1 Outlier Removal

There are several ways how to deal with outliers. In this experiment, outliers were addressed by removing them, as they can significantly affect the model's performance. Using the IQR (Interquartile Range) Method, the

outliers from BHK, Rent, and Size were removed. Since BHK, Rent, and Size are numerical attributes with left-skewed distributions, as observed in their histograms, the IQR Method was the appropriate method for outlier removal. This involves calculating the difference between the upper quartile (Q3) and the lower quartile (Q1).

2.2.2. Feature Engineering and Selecting

One-Hot Encoding

Since the majority of the attributes are categorical, they need to be transformed into values the machine can understand. One-Hot Encoding was applied for City, Point of Contact, Furnishing Status, and Tenant Preferred. This type of encoding is most suitable for nominal variables and attributes with few categories. Given the small number of categories of City, Point of Contact, Furnishing Status, and Tenant Preferred, they enable the creation of binary columns for each unique category. It converts these categorical attributes into binary vectors with 0 indicating absence and 2 indicating presence.

Feature Interaction

Using the correlation matrix, it was identified that there exists a strong correlation between BHK and Bathroom. A new feature was created by multiplying the values of BHK with those of the Bathroom and integrating it with the existing features.

Adding a Feature

It was given that the Size in the data is in square feet. To enhance the dataset and enhance the model's performance, a new feature was introduced by converting the Size values into their corresponding square meter equivalents.

Feature Selection

For feature selection, Manual Backward Selection was employed. The decision to eliminate features was based on the correlation matrix to determine the potentially important features. Through trial and error and assessing the model's performance through MSE (Mean Squared Error) and the Coefficient of Determination, the selected features were adjusted until the model's performance was optimized.

2.2.3. Training and Test Split

Initially, the test and training data were split by assigning all features, excluding the target variable, to one variable while isolating the target variable into its own. Utilizing Scikit-learn, the dataset was divided into training and testing sets with a test size of 0.2 and a random state of 0.42.

2.2.4. Standardization

The StandardScaler was applied to standardize the data of the features from the training and testing data. Both the features of the training and testing data were transformed, but the scaler was only fitted to the training data to prevent any potential data leakage. The standardization of these data ensures data consistency and can ultimately improve the model's performance.

2.3. Modelling

The model used was Linear Regression. The model was trained using the standardized training data of the features along with their corresponding target values, adjusting its parameters to minimize the disparity between the actual and predicted target values in the training data. Subsequently, the model was used to predict the target values for the standardized testing data

features using the predict method. A regularization technique known as Lasso Regularization was also employed. This method aims to prevent overfitting, thereby potentially enhancing the overall performance of the model.

2.4 Evaluation

2.4.1. Quantitative Evaluation

To evaluate the model's performance, metrics such as Mean Squared Error, Coefficient of Determination, and Coefficients were examined. The lower the MSE is and the closer the Coefficient of Determination is to 1, the better the model's performance is. To interpret the Coefficients, a positive sign signifies a positive relationship between the feature and the target, while a negative sign indicates a negative relationship.

2.4.2 Qualitative Evaluation

This evaluation involves extracting sample data from the features to predict house prices using the model. Through this evaluation, the predicted and actual prices were compared.

III. Experiments

Outliers

Various approaches were considered for handling outliers, involving trial and error to determine whether to retain or remove them. Outliers were removed individually for each numerical attribute (BHK, Rent, Size, and Bathroom), with the impact on model performance assessed. Interestingly, removing outliers from the Bathroom attribute led to a deterioration in model performance, while other numerical attributes showed huge significant improvement.

For the categorical attributes, a distinct approach to remove outliers was done since the IQR method was used for numerical attributes only. This involved establishing a threshold and comparing it with the frequency count of each categorical feature. If the count falls below the threshold, then it will be removed. However, the removal of the categorical attributes also resulted in a decline in performance. A regularization technique known as Ridge Regularization was also employed. This method aims to prevent overfitting, thereby potentially enhancing the overall performance of the model.

Encoding of Categorical Values

The Floor feature could be an important feature as the floor level of a property could mean a higher price.

Initially, a custom algorithm was used to assign a weight for every value of the Floor feature. This was done because the initial idea was to convert Floor levels into a ratio. However, it was recognized that fractions like 1/2 and 15/30, despite having the same ratio, do not hold the same meaning as the 15th floor is higher than the 1st floor. Using the same value for these fractions might lead the machine to interpret them as equivalent. By assigning weights to each value, the intention was to provide the machine with information about the relative importance of different floor levels. The algorithm began by assigning values to categories like Ground, Upper Basement, and Lower Basement to convert them into numerical values, achieved through the use of regular expressions. Subsequently, weights were assigned to all values. However, a drawback of this approach is the potential for inaccuracies in assigning weights to each value.

Another approach, known as frequency encoding, was also employed. This method is

suitable for encoding variables with numerous categories. It involves counting the frequency of each category and assigning this count as the value for that category. However, this technique did not lead to any improvement in the model's performance.

In another attempt, the Floor level and the total floors were segregated. Similar to the previous method, values were assigned for categories such as Ground, Upper Basement, and Lower Basement. Subsequently, two separate columns were generated for the Floor Level and the Total Floors. This separation was implemented to enable the machine to potentially interpret the floor level independently while retaining the total floor information for analysis purposes. However, this approach also resulted in marginal improvement. Following this, feature interaction was explored with the expectation of enhancing the model. Despite multiplying the floor level by the total floor, there was still no noticeable improvement observed.

Another strategy involved solely utilizing the floor level attribute for analysis, omitting the total floor information. Despite the potential value in understanding the height of the property's floor level, this approach also failed to yield any improvement in model performance.

Finally, target encoding was attempted for this attribute. It calculates the mean of the target variable for each specific category and assigns this mean value to that category. However, a potential issue with this method is the risk of overfitting, as the target variable is used for encoding the feature. To mitigate this risk, the target and training data were split before performing the target encoding. Despite this precaution, the feature still did not contribute significantly to improving model performance.

Similar approaches, including target encoding and frequency encoding, were also applied to the Area Locality attribute using the same methodology. However, neither approach resulted in any improvement in model performance.

The Posted On feature was encoded by extracting the month from the values. Given that all values belonged to the year 2022, it was deemed that including the year would not provide significant relevance. Additionally, the day component was omitted, as it was reasoned that minor day-to-day fluctuations would not significantly impact prices. However, despite these adjustments, this feature did not lead to any improvement in model performance.

Feature Interaction

After encoding all the features, a correlation matrix of all the features was created. Using this matrix, the features with strong correlations with each other were identified. With this, a feature interaction was done on Size and Bathroom and Floor and Area Locality. There was a decline on the model's performance with both approaches.

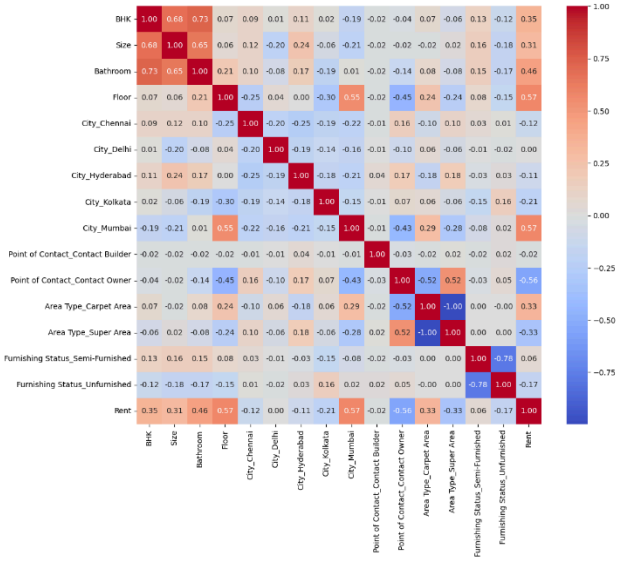

Figure 3.1 Correlation matrix after Encoding of Features

Feature Selection

Initially, all of the features were used for predicting house prices. Several ways were done for feature selection to improve the performance of the model. This includes SelectingKBest, Recursive Feature Elimination, and Principal Component Analysis.

Initially, a value of 20 was assigned to "k," representing the number of best features to be chosen using SelectKBest. With a limited number of features available, this meant that all features were selected. Subsequently, the value of "k" was decreased to explore its impact on model performance. However, adjusting "k" did not lead to improvement, although the model performed optimally when "k" was set to 17.

Two additional techniques, Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA), were applied. However, neither RFE nor PCA led to an improvement in the model's performance. The best results were achieved when using all available features.

Standardization

Multiple techniques were employed to standardize the data, including Min-Max Scaling, StandardScaler, Robust Scaling, and Normalizer. While the first four methods did not notably influence the model's performance, the Normalizer led to significantly worse performance. Among these techniques, StandardScaler yielded the most favorable outcome.

Regularization

Two regularization techniques were utilized. One of the primary advantages of regularization is its ability to prevent overfitting. Implementing these methods could potentially enhance the performance of the model.

IV. Results and Analysis

To analyze the results, a comparison will be made between the demo model and the proposed model. The key distinction between the proposed model and the demo model lies in the fact that the demo model did not remove any outliers and utilized only a limited number of features.

4.1. Removing Outliers

Outliers possess the capability to significantly impact the model parameters, potentially leading to unstable models. Consequently, their removal can result in the development of more stable models. The performance of the model notably improved following the removal of outliers.

By keeping the number of features constant, we can compare the model's performance across different outlier-handling scenarios. This involves comparing scenarios where outliers were retained and where outliers specifically for BHK, Size, and Rent were removed.

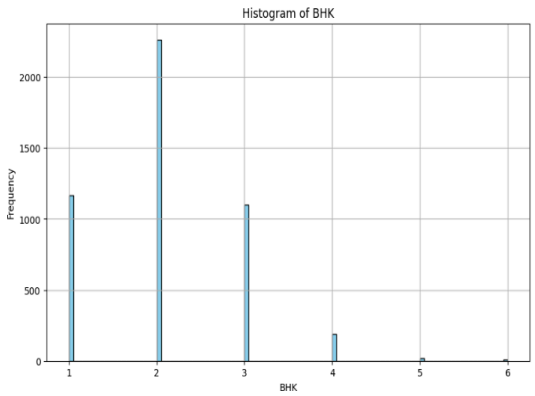

Figure 4.1 Histogram of BHK before removal of outliers

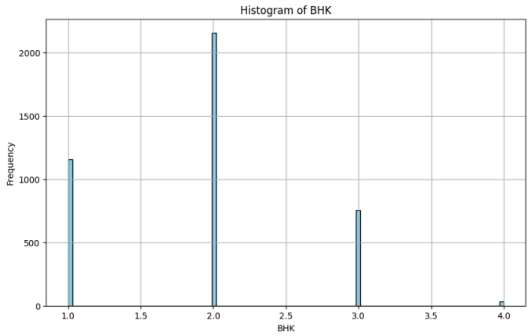

Figure 4.2 Histogram of BHK after removal of outliers

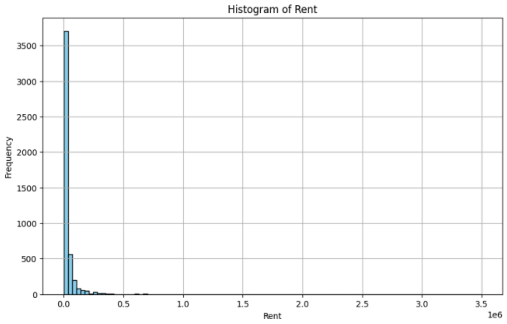

Figure 4.3 Histogram of Rent before removal of outliers

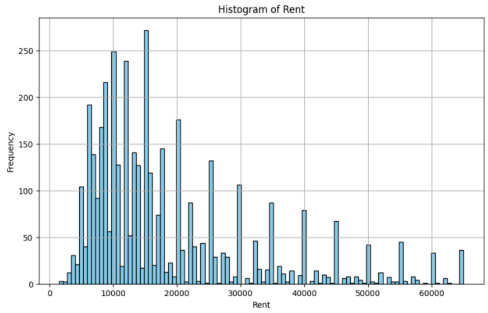

Figure 4.4 Histogram of Rent after removal of outliers

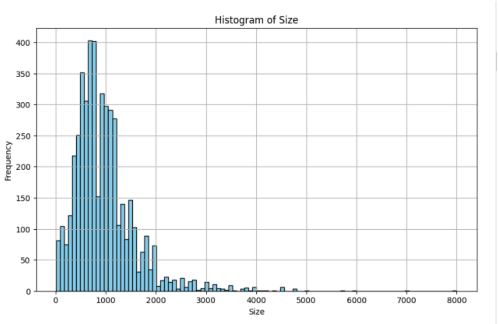

Figure 4.5 Histogram of Size before removal of outliers

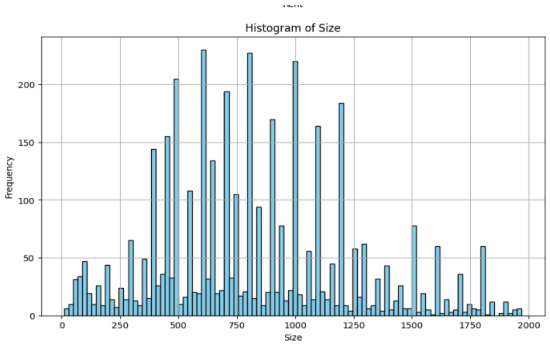

Figure 4.6 Histogram of Size after removal of outliers

	Mean Squared Error	**Coefficient of Determination**
Demo Model (with outliers)	1915010853.37	0.52
Proposed Model (without outliers)	57130281.12	0.69

Table 4.1 Comparison of Removing outliers

4.2. Feature Engineering and Selection

Selecting important features, creating new features, and incorporating feature interaction contributed to an improved model performance. The Coefficient of Determination increased from 32% in the demo model to 56% in the proposed model. Additionally, the Mean Square Error decreased from 2 trillion in the demo model to 1.7 trillion in the proposed model.

	Mean Squared Error	**Coefficient of Determination**
Demo Model (few features)	2693550407.36	0.32
Proposed Model (added important features)	57130281.12	0.69

Table 4.2 Comparison of Adding more features

4.3 Regularization

Integrating regularization into the model resulted in only a marginal enhancement of its performance. This regularization technique prevents overfitting, thus contributing to a more robust model. Lasso Regularization proved to be the most effective in optimizing the model's performance.

Regularization Technique	Mean Squared Error	Coefficient of Determination
Lasso (L1)	56783649.95	0.69
Ridge (L2)	56815934.02	0.69
No Regularization	56803893.69	0.69

Table 4.3 Comparison of Regularization Techniques

4.4 Overall Performance

By incorporating the removal of outliers, conducting feature engineering and selection, and implementing regularization techniques, the model's performance witnessed a significant improvement, soaring from an initial 32% to a commendable 69%. This comprehensive approach addressed various aspects crucial for enhancing model accuracy and robustness.

	Mean Squared Error	Coefficient of Determination
Demo Model	2693550407.36	0.32
Proposed Model	56803893.69	0.69

Table 4.4 Comparison of Overall Model Performance

Upon analyzing the sample data, it was noted that the Proposed Model exhibited a more precise prediction of the actual price, with a difference of 2,640. In contrast, the Demo Model displayed a larger disparity, with a difference of 22,239. This discrepancy underscores the enhanced accuracy and precision achieved by the Proposed Model, highlighting its superiority over the Demo Model in accurately predicting housing prices.

	Demo		Proposed	
	Actual	Predicted	Actual	Predicted
1	10000	32239.375356189103	10000	12640.115452071188

Table 4.5 Comparison of Actual and Predicted Prices from the Sample Data between the Demo and Proposed Model

V. Conclusions and Recommendations

In conclusion, the enhancements implemented in the Proposed Model, including outlier removal, feature engineering and selection, and regularization techniques, have led to a significant improvement of the accuracy compared to the Demo Model.

These improvements have undoubtedly transformed the model into a highly valuable tool, particularly for real estate agents. By providing more accurate predictions of property prices, the model equips agents with crucial insights to better serve their clients.

For further improvements, it is suggested to explore more on the encoding techniques for variables with diverse categories, as target-encoded values of these rare categories can be less reliable. It is recommended to handle these rare categories by combining them into broader categories. Using binning as an encoding technique may be useful for the Floor Value, such as assigning the values as Lower Floors, Middle Floors, and Higher Floors.

It is also recommended to explore other outlier-handling techniques as outliers can be transformed using mathematical functions and replace their values with non-outlying values.

Lastly, there are several ways to engineer features. In this way, more features can be used for more accurate prediction.