**School of Chemistry, Chemical Engineering and Biotechnology**

**Division of Chemical and Biomolecular Engineering**

A black background with blue and red text

Description automatically generated

**CB4247**

**Statistics and Computational Inference to Big Data**

**India’s House Prices Prediction**

Name: Nathan Lawira

Matriculation Number: [This field is left blank for the GitHub Copy]

Date: April 14, 2024

**Table of Contents**

[List of Figures 3](#_Toc166594025)

[List of Tables 4](#_Toc166594026)

[List of Equations 5](#_Toc166594027)

[Abstract 6](#_Toc166594028)

[1.0 Introduction 7](#_Toc166594029)

[1.1 Aim 7](#_Toc166594030)

[1.2 Background 7](#_Toc166594031)

[2.0 Body 9](#_Toc166594032)

[2.1 Exploratory Analysis & Data Pre-Processing 9](#_Toc166594033)

[3.2 Regression and Modeling Methods 19](#_Toc166594034)

[3.0 Results and Discussions 22](#_Toc166594035)

[3.1 Linear Regression 22](#_Toc166594036)

[Recommendations 24](#_Toc166594037)

[Conclusion 25](#_Toc166594038)

[References 26](#_Toc166594039)

[Appendix A 27](#_Toc166594040)

[Appendix B 28](#_Toc166594041)

# List of Figures

[Figure 1 A distribution plot, boxplot, violin plot, and probability plot for "Rent" 8](#_Toc166447629)

[Figure 2 Distribution plot, boxplot, violin plot, and probability plot for “Rent” after removal of the extreme outlier 9](#_Toc166447630)

[Figure 3 A distribution plot, boxplot, violin plot, and probability plot for “Size” 9](#_Toc166447631)

[Figure 4 A count plot and a boxplot of “BHK” 10](#_Toc166447632)

[Figure 5 A count plot and a boxplot of “Bathroom” 11](#_Toc166447633)

[Figure 6 A count plot of houses available for rent grouped by “City” 12](#_Toc166447634)

[Figure 7 A count plot of houses available for rent grouped by “Furnishing Status” 12](#_Toc166447635)

[Figure 8 A count plot of houses available for rent grouped by “Area Type” 13](#_Toc166447636)

[Figure 9 A count plot of houses available for rent grouped by “Tenant Preferred” 14](#_Toc166447637)

[Figure 10 A count plot of houses available for rent grouped by “Point of Contact” 15](#_Toc166447638)

[Figure 11 Count plot of houses available for rent grouped by “Posted On” 15](#_Toc166447639)

[Figure 12 Value counts of “Area Locality” (left) and “Floor” (right) 16](#_Toc166447640)

[Figure 13 Code to scale the value of the regressors 18](#_Toc166447641)

[Figure 14 Code to create dummy variables for categorical variables 18](#_Toc166447642)

[Figure 15 Code to calculate sum of squares regression and error 19](#_Toc166447643)

[Figure 16 Code to calculate Fo and respective F value 20](#_Toc166447644)

[Figure 17 Code to find residuals 20](#_Toc166447645)

[Figure 18 Residual plots 21](#_Toc166447646)

[Figure 19 Code to calculate percentage of residual points within [-3, 3] 21](#_Toc166447647)

# List of Tables

[Table 1 Descriptive statistics of the numerical variables 7](#_Toc166332412)

[Table 2 Performance metrics of models before feature importance 21](#_Toc166332413)

[Table 3 Performance metrics of models after feature importance 21](#_Toc166332414)

# List of Equations

**No table of figures entries found.**

# Abstract

A dataset on housing prices in India was explored in this project. The dataset was extracted from Kaggle and analyzed using various Python libraries. Exploratory analysis and pre-processing of the data were conducted via visualization with graphs and removal of extreme outliers. Linear regression, specifically ordinary least squares regression, and various machine learning algorithms were trained on labeled data and used to predict unseen data. ANOVA and residual analysis were conducted for the linear regression model, while feature importance was conducted for the remaining algorithms to find and select the most critical variables. After the critical variables were identified and chosen, all models were retrained on the critical variables, and their performances were reevaluated.

# Introduction

## 1.1 Aim

The aim of this project is to apply regression and machine learning techniques to analyze a complex real-world dataset and predict values for new data. This dataset contains data on house prices in India and includes multiple regressors and a response variable, also known as regressors and response, respectively. Machine learning algorithms will be trained and used to predict unseen data, and the performance of these algorithms on the dataset will be evaluated. Afterward, the best-performing algorithms will be combined, and their results will be evaluated.

## 1.2 Background

India's housing landscape, a subject of our study, is remarkably varied, from luxurious palaces once home to ancient maharajas, modern skyscraper apartments in vibrant cities, and simple huts in isolated rural areas. This variety displays the growth in India's housing sector that parallels the country's rising income levels. According to the Human Rights Measurement Initiative, India has achieved 60.9% of its potential in providing its citizens with the fundamental right to housing based on income. Regarding housing arrangements, renting, also known as hiring or letting, involves payments for the temporary use of a resource, service, or property owned by someone else. A gross lease is a specific type of rental agreement where the tenant pays a set rent, and the landlord covers all property-related costs, which may include tax payments and utility bills. Renting supports the principles of the sharing economy by allowing the shared use of assets, enhancing both efficiency and access to housing for diverse groups of people (Banerjee, 2022).

A dataset from Kaggle, titled “House Rent Prediction Dataset,” was used for the aims of this project. The dataset was extracted from Magicbricks, a website that provides service for all real estate needs, including property listings, rent payments, home loans, and expert advice. The dataset contains information on over 4700 property listings in India, with a rich number of attributes such as number of bedrooms, rent, property size, city, furnishing status, and others that will be explored and used to train a model in later sections (Banerjee, 2022).

This dataset was selected amongst other datasets due to its nature and usability score. It is suitable for analysis via linear regression and other forms of regression, as shown on the website, and matches the purposes of this project. A usability score on Kaggle displays the ease of use of a dataset based on completeness, credibility, and compatibility (Devrishi, *New usability rating on datasets*). Completeness is evaluated based on the presence of the dataset's title, tags, and description; credibility includes whether the dataset source is specified and traceable; and compatibility depends on whether the dataset’s format is compatible for use and sufficient information is provided for the file and columns. This dataset’s usability score is 10.00, achieving the maximum score in all three criteria mentioned previously, meaning that the dataset is clear, reliable, and easy to use. Therefore, it was selected for analysis in this project.

For this project, the dataset was analyzed using the Python programming language and various libraries, such as NumPy, Pandas, and scikit-learn, to assist with processing data, conducting regression, and training machine learning models. Students were initially required to select either Python or MATLAB for this project’s code, and the former was selected due to greater familiarity. The Python script was written using Jupyter Notebook.

# 2.0 Body

## 2.1 Exploratory Analysis & Data Pre-Processing

After being loaded into a data frame, the dataset was analyzed to find the column names indicating the dataset attributes. As shown below, there were 12 unique columns, which will be explored in the later sections. On the Kaggle site, the dataset was provided with a glossary that explains the columns (Banerjee, 2022):

* **BHK:** Number of Bedrooms, Hall, Kitchen.
* **Rent:** Rent of the Houses/Apartments/Flats.
* **Size:** Size of the Houses/Apartments/Flats in Square Feet.
* **Floor:** Houses/Apartments/Flats situated in which Floor and Total Number of Floors (Example: Ground out of 2, 3 out of 5, etc.)
* **Area Type:** Size of the Houses/Apartments/Flats calculated on either Super Area or Carpet Area or Build Area.
* **Area Locality:** Locality of the Houses/Apartments/Flats.
* **City:** City where the Houses/Apartments/Flats are Located.
* **Furnishing Status:** Furnishing Status of the Houses/Apartments/Flats, either it is Furnished or Semi-Furnished or Unfurnished.
* **Tenant Preferred:** Type of Tenant Preferred by the Owner or Agent.
* **Bathroom:** Number of Bathrooms.
* **Point of Contact:** Whom should you contact for more information regarding the Houses/Apartments/Flats.

Preliminary analysis showed no missing nor duplicated values. The 12 columns, referred to as variables, were categorized into numerical and categorical variables, as shown by having either “int64” or “object” data types, respectively. The numerical variables are “BHK”, “Rent”, “Size”, and “Bathroom”, while the categorical variables are “Posted On”, “Floor”, “Area Type”, “Area Locality”, “City”, “Furnishing Status”, “Tenant Preferred”, and “Point of Contact”.

Preliminary analysis was done on the numerical variables via descriptive statistics, as shown in Table 1 below. The mean of “Rent” and “Size” are larger than their respective medians, suggesting that these variables are positively skewed.

A screenshot of a computer

Description automatically generated

Table 1 Descriptive statistics of the numerical variables

The numerical variables were analyzed using distribution, box, violin, and probability plots. The plots for “Rent” are shown below in Figure 1. As immediately evident in the box and probability plots, there was an extreme outlier on the rightmost side of both graphs. This extreme outlier was then removed to prevent adverse effects on the models. The graphs after the removal of this outlier are shown in Figure 2. The same graphs were plotted for Size, as shown in Figure 3 below. The distribution, box, and violin plots indicate that both “Rent” and “Size” are positively skewed, as confirmed by Table 1. According to (*1.3.3.21. Normal Probability Plot,* n.d.), the probability plot indicates that “Rent” slightly deviated from a normal distribution as some data points deviate from the straight line.

A screenshot of a graph

Description automatically generated

Figure 1 Distribution, box, violin, and probability plots for "Rent" (from left to right, up to down)

A graph of a graph

Description automatically generated with medium confidence

Figure 2 Distribution, box, violin, and probability plots for “Rent” after removal of the extreme outlier (from left to right, up to down)

A collage of graphs

Description automatically generated

Figure 3 A distribution, box, violin, and probability plot for “Size”(from left to right, up to down)

For BHK and Bathroom, a count and a box plot were used instead since they took on discrete values in a smaller interval than “Rent” and “Size.” The plots for “BHK” and “Bathroom” are shown below in Figures 4 and 5, respectively, indicating that both variables are positively skewed. From the graphs below, most houses available for rent have a “BHK” of 2 and either 1 or 2 bathrooms, potentially indicating higher demand for smaller houses than bigger houses in India.

A comparison of a bar graph

Description automatically generated

Figure 4 A count and a box plot of “BHK”

A comparison of a graph

Description automatically generated

Figure 5 A count and a box plot of “Bathroom”

Afterwards, exploratory analysis was conducted on the categorical variables. Count plots via Seaborn were used to observe their trends. The counterplots for “City,” “Furnishing Status,” “Area Type,” “Tenant Preferred,” “Point of Contact,” and “Posted On” are shown below in Figures 6, 7, 8, 9, 10, and 11, respectively. Each graph will be discussed separately below.

As shown in Figure 6 below, the data of houses were extracted for houses available for rent in Kolkata, Mumbai, Bangalore, Delhi, Chennai, and Hyderabad only, not for other Indian cities. Mumbai had the highest number of houses available for rent, followed by Chennai. This may be because Mumbai and Chennai are two of India's most densely populated cities (Kolb, 2019) and the richest and most developed cities (NoBroker, 2024), which attracts people to work there and thus increases the demand for housing. Meanwhile, Kolkata had the lowest number of houses available for rent.

A graph of houses available for rent

Description automatically generated

Figure 6 A count plot of houses available for rent grouped by “City”

As shown in Figure 7 below, semi-furnished houses were the largest in number, followed by unfurnished and furnished houses. This may indicate tenants’ preferences for budget-friendlier houses and the freedom to furnish them.

A graph of a house

Description automatically generated with medium confidence

Figure 7 A count plot of houses available for rent grouped by “Furnishing Status”

As shown in Figure 8 below, most houses’ sizes were displayed in the super or carpet areas and scarcely built areas. Carpet area means all the floor area covered by a carpet, while the built house is the total carpet area added by the wall area. The super area includes the carpet area and the area of common spaces (Mishra, 2024). Since the houses listed using built area were much lower than the other two categories, these data points were outliers and thus removed.

A graph showing houses available for rent

Description automatically generated

Figure 8 A count plot of houses available for rent grouped by “Area Type”

As shown in Figure 9 below, a majority of house owners, 3442 out of 4743, do not have a preference for tenants.

A graph showing a number of houses available

Description automatically generated

Figure 9 A count plot of houses available for rent grouped by “Tenant Preferred”

As shown in Figure 10 below, the point of contact for most houses available for rent was the owner, followed by agents. This may indicate landowners’ preferences to negotiate directly with tenants rather than through an agent. Since the number of houses preferring to contact builders was much less than the other two categories, this data point was removed as an extreme outlier.

A graph of houses available for rent

Description automatically generated

Figure 10 A count plot of houses available for rent grouped by “Point of Contact”

As shown in Figure 11 below, “Posted On,” a categorical variable, had many unique values, with some having very little count. Thus, this variable in this state will not be useful in regression and training the machine learning models. To tackle this issue, “Posted On” was divided into four numerical variables, called “Day Posted,” “Day of the Week Posted,” “Month Posted,” and “Quarter Posted,” to preserve information and consider the potential effect of the date of posting on rent prices. The year of posting is not considered because all listings were posted in 2022, so it is not a variable but a constant value. The code that achieved this is attached to Appendix A.

A graph showing a number of houses

Description automatically generated

Figure 11 A count plot of houses available for rent grouped by “Posted On”

“Area Locality” and “Floor” were not plotted on count plots as they had too many unique values, so running a Python script to plot a count plot for both was very time-consuming. Instead, their value counts were obtained using the methods shown below in Figure 12. Like “Posted On,” these two variables in this state had many unique values, with some having little count, so they will not be useful in regression and training the machine learning models.

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

Figure 12 Value counts of “Area Locality” (left) and “Floor” (right)

To tackle this problem, “Floor” was split into two numerical variables, “Floor Level” and “Total Floor,” by splitting each string value in “Floor” with “out of” as a delimiter. Thus, “Floor Level” contained the number preceding “out of” while “Total Floor” contained the number trailing “out of.” Four-string values in “Floor” did not have “out of” and were removed. The code that achieved this is attached to Appendix B. “Area Locality,” on the other hand, was removed since converting this variable suitable for regression and machine learning is difficult.

A correlation between the numerical variables was obtained and plotted using a heatmap shown below in Figure 13. According to it, “Bathroom” had the highest correlation with “Rent,” followed by “Size” and “BHK.” The highest correlation was found between “Bathroom” and “BHK,” which is reasonable since the number of bathrooms is included in “BHK.” The next highest correlations are found between “Bathroom” and “Size,” then “BHK” and “Size,” which is also reasonable because as the number of rooms in the house increases, the area of the house should follow a similar trend. However, for the purposes of this project, only the relationship between the response variable, “Rent,” and the regressors, the other variables, will be used for analysis and training machine learning algorithms, and not the relationship between one regressor and another. Additionally, “Day Posted” and “Day of the Week Posted” had the weakest correlation with rent, with values close to 0.

A colorful chart with white text

Description automatically generated with medium confidence

Figure 13 Heatmap of numeric variables’ correlation

## 3.2 Regression and Modeling Methods

This project used linear regression, specifically ordinary least squares (OLS) regression, to conduct regression analysis on the dataset. It was selected over nonlinear regression for simplicity. A linear regression model was trained on the training data and predicted the response for the test data. Afterward, its performance was evaluated via the error and R2 values. ANOVA analysis was conducted to determine whether there was a significant statistical relationship between the response and at least one of the regressors. Also, residual analysis was performed to validate the assumptions of using OLS, particularly whether the error follows a normal distribution with constant variance. The results of these analyses will be discussed in the “Results and Discussions” section.

After OLS regression analysis, various machine learning algorithms were trained on the same training data and predicted the response for the same test data. These algorithms are Ridge, Lasso, Bayesian Ridge, K-Nearest Neighbors, Decision Tree, Random Forest, Gradient Boosting, Support Vector, CatBoost, LightGBM, and XGBoost. Their performances were evaluated via the mean absolute error, mean squared error, root mean squared error, and R2 values.

Rent was chosen as the response variable. In contrast, “Area Type,” “Size,” “BHK,” “Bathroom,” “City,” “Furnishing Status,” “Tenant Preferred,” “Day Posted,” “Day of the Week Posted,” “Month Posted,” “Quarter Posted,” “Floor Level,” and “Total Floors” were chosen as the regressors. The original dataset was split into the training and test dataset in a 0.8 to 0.2 ratio, specifying a random state 42 to achieve the same results every time the script runs. Before the models were trained, the value of the numerical regressors was because some models were sensitive to the magnitude of the regressor values, namely K-Nearest Neighbors, Support Vector, Ridge, and Lasso (Jaadi, 2023). Additionally, dummy variables were used to represent categorical variables by creating Boolean variables representing each unique categorical value (Garavaglia et al., n.d.). The final list of variables used for modeling is shown below in Table 2.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Response Variable** | **Regressors** | | | | |
| Rent (y) | BHK (x1) | Size (x2) | Bathroom (x3) | Day Posted (x4) | Day of the Week Posted (x5) |
| Month Posted (x6) | Quarter Posted (x7) | Floor Level (x8) | Total Floors (x9) | Area Type\_Super Area (x10) |
| City\_Chennai (x11) | City\_Delhi (x12) | City\_Hyderabad (x13) | City\_Kolkata (x14) | City\_Mumbai (x15) |
| Furnishing Status\_Semi Furnished (x16) | Furnishing Status\_Unfurnished (x17) | Tenant Preferred\_Bachelors/Family (x18) | Tenant Preferred\_Family (x19) | Point of Contact\_Owner (x20) |

Table 2 List of the response variable and regressors and their representations used for modeling

After the machine learning algorithms were trained, their performance metrics were calculated and observed. The algorithms with the lowest errors and better fit the dataset were identified and selected for feature importance analysis. Features of high importance across these models were chosen, and all models were retrained on these features alone. The performance of these new models was evaluated via mean absolute error, mean squared error, root mean squared error and R2 values. The R2-adjusted values before and after feature importance were also considered to identify whether removing non-significant regressors improved the model (Team, 2023).

# 3.0 Results and Discussions

## 3.1 Linear Regression

A linear regression model was fitted using the LinearRegression() method from the sklearn library. The intercept and coefficients of the fitted model were retrieved., and the equation is shown below.

Equation 1 Fitted linear regression model equation

After the linear regression model was fitted, an ANOVA analysis was conducted to determine whether a significant linear relationship existed between the response and at least one of the regressors (*Lesson 3: SLR Evaluation | STAT 462*, n.d.). This is done by first assuming the response variable follows an F distribution, of which the value can be calculated by dividing the mean squared error (MSE) from the regression mean squared (MSR) shown in equations 2 and 3 below (*3.5 - the Analysis of Variance (ANOVA) Table and the F-test | STAT 462*, n.d.). The final F value, F0, shown in equation 4, will determine whether to reject the initial assumption. yi is the actual value of each sample, ȳ is the sample mean of y, ŷi is the predicted value each sample of y by the linear regression model, n is the number of samples, p is the number of estimators, and k is the number of regressors.

Equation 2 Mean Squared Error for ANOVA

Equation 3 Regression Mean Squared for ANOVA

Equation 4 F0 value

From equation 1, the number of regressors, k, was 20 as more variables were created due to dummy variables. The number of estimators, p, was 21, as there was an intercept in the linear regression model. The number of samples, n, was the number of rows of data points, which was 4738. Thus, using Python, the final F value was found to be 259.88.

Afterwards, the , in Figures 19 and 20, the respective F0 value is 301.64, while the calculated f-distribution value with α = 0.05, k = 15, p = 16, n= 4742 was 1.669. F0 was significantly higher than the calculated f-distribution, which means that there existed a significant linear relationship between one of the regressors with the response variable.

Residual analysis of the linear regression was conducted by calculating the residuals by subtracting the predicted values from the actual values, then subtracting each residual by its sample mean and dividing it by the standard deviation. The code that accomplished this is shown below in Figure 21.

Afterwards, the standardized residuals were plotted in three different plots, as shown in Figure 22 below. The first plot shows that the residuals roughly follow a normal distribution. The second plot shows an increasing trend in predicted and actual values despite some outliers. The third plot shows that the points crowd around the 0 line and have approximately the same number of points above and below the 0 line. However, we can observe a downward trend in the third plot, indicating that the model has room for improvement.

A group of graphs showing the results of a statistical analysis

Description automatically generated

Figure 19 Residual plots

From the observations of the plots above, we can assume that the residuals roughly follow a normal distribution, which means that the error/noise is approximately normally distributed with constant variance. This assumption was later verified by calculating the percentage of points within [-3, 3] shown in code below in Figure 23. 98.25% of the residuals lie within [-3, 3], which indicates that the residuals roughly follows a normal distribution of 0 mean and 1 standard deviation. Points that lie outside of [-3, 3] can be considered outliers and removed in later models to improve the accuracy of the Linear Regression model.

A screenshot of a computer code

Description automatically generated

Figure 20 Code to calculate percentage of residual points within [-3, 3]

The performance of each model on the test data is shown below in Table 2. As shown in Table 2, Random Forests outperforms both linear regression and XGBoost as it had a higher R2 value and lower errors overall. However, the values remained low, so feature importance was conducted on XGBoost and Random Forest, and the models were retrained on variables with larger importance. The results of the retrained model are shown below in Table 3.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Linear Regression** | **XGBoost** | **Random Forest** |
| **Mean Absolute Error** | 22459.91626822261 | 13969.80044152086 | **13555.038700181447** |
| **Mean Squared Error** | 2016663801.8021157 | 1836858580.516088 | **1376275766.0331059** |
| **Root Mean Squared Error** | 44907.28005348482 | 42858.58817688805 | **37098.1908727785** |
| **R2** | 0.4898127774395711 | 0.5353009378200008 | **0.651821830727462** |
| **R2 Adjusted** | 0.4821654314911278 | 0.5283354272519922 | **0.6466028860060322** |

Table 3 Performance metrics of models before feature importance

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Linear Regression** | **XGBoost** | **Random Forest** |
| **Mean Absolute Error** | 22495.018859406766 | 13081.956977225203 | **12969.870560850226** |
| **Mean Squared Error** | 2022771937.5304327 | 1485256805.4294655 | **1238607104.6472921** |
| **Root Mean Squared Error** | 44975.23693690154 | 38539.02963788094 | **35193.850381100565** |
| **R2** | 0.48826750608622627 | 0.6242511797584249 | **0.686650041519601** |
| **R2 Adjusted** | 0.48391233592525795 | 0.6210533174584967 | **0.683983233362321** |

Table 4 Performance metrics of models after feature importance

Comparing the two tables above, the root mean squared error of linear regression increased slightly while the errors of both XGBoost and Random Forest decreased after feature importance. Additionally, the R2 value of linear regression decreased and the R2 value of XGBoost and Random Forest increased after feature importance. This means that both XGBoost’s and Random Forest’s performances improved after feature importance, while the linear regression model decreased slightly. However, the R2 adjusted value of all models increased after feature importance, indicating that the models trained after feature importance were better models compared to the first models.

# Recommendations

To improve the performance of all the models, some recommendations for future projects can be made.

1. The variables Area Locality, Floor, and Posted On may be explored and used for regression and training the model to see their influence on it. In particular, the variable Floor can be split into two other variables, Floor Location and Total Floor, to be used for regression and modelling.
2. Different pre-processing techniques can be applied, such as removing outliers beyond the interquartile range. Box-Cox transformation can be used to transform the variables' distribution into a more normal distribution, thereby improving the accuracy of the linear regression.
3. Different machine learning algorithms beyond the ones used here can be explored and used for prediction. Machine learning algorithms such as Neural Networks tend to be more robust against data that deviate from linearity and as such, perform better compared to other models.

# Conclusion

The aims of this project were met by using linear regression and machine learning models such as XGBoost and Random Forests. The models were trained on supervised data and used to predict unseen data, and their performances were evaluated using R2 values and mean errors. ANOVA analysis was also conducted on the linear model, and a significant relationship between one of the regressors and the response variable was determined. A residual analysis was also conducted to determine whether the assumption holds true for the error term following a normal distribution with constant variance, and it was shown that aside from some outliers, the assumption was plausible. Feature importance analysis was conducted on the XGBoost and Random Forest model, and then four features with the highest importance shared between the two models were selected. The three models were retrained only on the selected regressors, and the metrics show that the model improved. However, the errors for all three remain high, and the R2 value remained low. Hence, there are many improvements to be made in improving the performance of these models.

# References

*1.3.3.21. Normal probability plot*. (n.d.). https://www.itl.nist.gov/div898/handbook/eda/section3/normprpl.htm

*3.5 - The Analysis of Variance (ANOVA) table and the F-test | STAT 462*. (n.d.). https://online.stat.psu.edu/stat462/node/107/

Banerjee, S. (2022, August 20). *House Rent Prediction Dataset*. Kaggle. https://www.kaggle.com/datasets/iamsouravbanerjee/house-rent-prediction-dataset?resource=download

Devrishi. (n.d.). *New usability rating on datasets*. Kaggle. https://www.kaggle.com/discussions/product-feedback/93922

Garavaglia, S., Sharma, A., & Dun & Bradstreet. (n.d.). *A SMART GUIDE TO DUMMY VARIABLES: FOUR APPLICATIONS AND a MACRO*. https://stats.oarc.ucla.edu/wp-content/uploads/2016/02/p046.pdf

Jaadi, Z. (2023, August 4). *When and why to standardize your data*. Built In. https://builtin.com/data-science/when-and-why-standardize-your-data

Kolb, E. (2019, July 11). 75,000 people per square mile? These are the most densely populated cities in the world. *USA Today*. Retrieved April 12, 2024, from https://www.usatoday.com/picture-gallery/news/world/2019/07/11/the-50-most-densely-populated-cities-in-the-world/39664445/

*Lesson 3: SLR Evaluation | STAT 462*. (n.d.). https://online.stat.psu.edu/stat462/node/80/

Mishra, S. (2024, April 1). *Difference in carpet area, built-up area and super built-up area*. Housing News. https://housing.com/news/real-estate-basics-part-1-carpet-area-built-up-area-super-built-up-area/

NoBroker. (2024, February 23). *Top 10 richest cities in India – Wealthiest urban centers in the country*. The NoBroker Times. https://www.nobroker.in/blog/richest-cities-in-india/

Team, C. (2023, November 21). *Adjusted R-squared*. Corporate Finance Institute. https://corporatefinanceinstitute.com/resources/data-science/adjusted-r-squared/#:~:text=What%20is%20the%20Adjusted%20R,a%20regression%20model%20or%20not.

# Appendix A

Below is the code script to convert the “Posted On” variable into four numerical variables: “Day Posted,” “Day of the Week Posted,” “Month Posted,” and “Quarter Posted.”

A screenshot of a computer

Description automatically generated

# Appendix B

Below is the code script to convert the “Floor” variable into two numerical variables, “Floor Level” and “Total Floors.”

A screenshot of a computer program

Description automatically generated