**India Rental Housing Market Analysis**

Author: Desmond Foo Yau Yit [@SHINE-six](https://github.com/SHINE-six)

# Introduction

The rental housing market in India is dynamic and complex, with multiple factors influencing the pricing of rentals. The complex dynamics of this market have been deciphered by a thorough examination of a large dataset on home rents. A vast range of information about rental properties that are currently available is included in this dataset. Some of the details that are included are the Bedroom-Hall-Kitchen (BHK) layout, rental costs, property size, number of floors, area type, locality, city, furnishing status, tenant preferences, number of bathrooms, and contact details. This dataset, which has more than 4700 records and 12 columns, is a useful tool for identifying the variables that influence people's selection of rental houses according to their lifestyles, preferences for certain localities, and family histories.

The main goal of this analysis is to help tenants make decisions, but it is also intended to help property owners make smart real estate investments for optimal rental price. Data exploration, manipulation, transformation, and visualisation are used in the analysis to make it thorough and go beyond the course's standard methodologies by incorporating advanced concepts to improve the data study. To demonstrate methods, graphs and R code will be categorised by certain criteria. The investigation will examine how layout, size, amenities, proximity, and urban factors affect rental costs. This research examines the dataset's unique qualities to help property owners navigate the rental housing market and make financially prudent decisions that meet their goals.

# Data description

The dataset consists of items that represent different rental homes, apartments, and flats. The following parameters are used to provide complete details for each attribute in this dataset:

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Attribute | Type | Description |
| 1 | Posted On | Interval | This attribute is a DATE data type. It is about the date of the house being posted on for rent. |
| 2 | BHK | Ratio | This attribute is a Numeric data type attribute. It represents the number of bedrooms, halls and kitchen. |
| 3 | Rent | Ratio | This attribute is a Numeric data type attribute. It represents the rent price of the property. |
| 4 | Size | Ratio | This attribute is a Numeric data type attribute. It shows the size of the property to-be-rent. |
| 5 | Floor | Ratio | This attribute is a Numeric data type attribute. It shows the number of floors or the floor number of house to be rented. |
| 6 | Area Type | Nominal | This attribute is a Character data type attribute. It is about the method of  calculation of the size attribute. |
| 7 | Area Locality | Nominal | This attribute is a Character data type attribute. It is about the venue of the house which is listed to be rent. |
| 8 | City | Nominal | This attribute is a Character data type attribute. It shows the city that the house to be rent is located. |
| 9 | Furnishing  Status | Nominal | This attribute is a Character data type attribute. It shows whether the house to be rent is furnished, semi-furnished or unfurnished. |
| 10 | Tenant Preferred | Nominal | This attribute is a Character data type attribute. It shows the type of tenants preferred by the owner or agent to rent the house. |
| 11 | Bathroom | Ratio | This attribute is a Numeric data type attribute. This states the number of bathrooms of the house to be rent. |
| 12 | Point of Contact | Nominal | This attribute is a Character data type attribute. This shows to whom the tenant should contact for further inquiries regarding the property. |

Hypothesis

***Furnished houses at different cities, yield the same result of 80% smaller size house having a profit rent price***

# Assumption

With a quick glance at the available cities in the dataset, we found out the record was only collected within the 6 cities of India: Bangalore, Chennai, Delhi, Hyderabad, Kolkata, and Mumbai.

Brief dive into each city:

1. Bangalore (Barb, 2023-07-27) (Srinivas, 1997)
   1. Bangalore is the third most populous city in India, with a population of over 8 million people. This makes it a megacity, and one of the most populous cities in the world.
   2. Bangalore is a major economic hub in India. It is home to a large IT and technology industry, and is known as the "Silicon Valley of India". The city is also a major center for manufacturing, aerospace, and education.
   3. Bangalore has a well-developed infrastructure. The city has a modern transportation system, including a metro system, a bus system, and an airport. It also has a good road network, and a number of hospitals and schools.
2. Chennai (Barb, 2023-07-27) (Kumar, 2022)
   1. Chennai is the fourth most populous city in India, with a population of over 10 million people.
   2. The city has a well-developed infrastructure, including a metro rail system, an airport, and a network of highways.
   3. Chennai is a major economic hub, with a diversified industrial base.
   4. The city is home to a number of educational institutions, including the Indian Institute of Technology Madras and the Madras Christian College.
3. Delhi (Barb, 2023-07-29) (Aggarwal, 2010)
   1. Delhi is the second most populous city in India, with a population of over 20 million people.
   2. The city has a well-developed infrastructure, including a metro rail system, an airport, and a network of highways.
   3. Delhi is a major commercial and industrial center, with a diversified economy that includes manufacturing, IT, and tourism.
   4. The city is home to a number of educational institutions, including the Indian Institute of Technology Delhi and the University of Delhi.
4. Hyderabad (Barb, 2023-07-29)
   1. Hyderabad is the fourth most populous city in India, with a population of over 10 million people.
   2. The city has a well-developed infrastructure, including a metro rail system, an airport, and a network of highways.
   3. Hyderabad is a major commercial and industrial center, with a diversified economy that includes manufacturing, IT, and tourism.
   4. The city is home to a number of educational institutions, including the Indian Institute of Technology Hyderabad and the University of Hyderabad.
5. Kolkata (Barb, 2023-07-29)
   1. Kolkata is the third most populous city in India, with a population of over 14 million people.
   2. The city has a well-developed infrastructure, including a metro rail system, an airport, and a network of highways.
   3. Kolkata is a major commercial and industrial center, with a diversified economy that includes manufacturing, IT, and tourism.
6. Mumbai (Barb, 2023-07-29)
   1. The Mumbai Metro Rail system, which was inaugurated in 2014, is one of the largest and most modern metro systems in India.
   2. The Mumbai Port is one of the largest ports in India, and it is a major hub for trade and commerce.
   3. The IT sector in Mumbai is growing rapidly, and the city is now home to a number of multinational IT companies.
   4. The tourism industry in Mumbai is also growing, and the city is a popular destination for both domestic and international tourists.

As can be seen and assumed of, is that all the 6 cities collected based on, in the dataset are, and can be considered as quite of a developed city. With the vision and mindset of a developed city, a critical situational point of view & consideration was taken. Therefore, in the state of the standpoint taken in this project, consideration of that in a developed city, there will be many of the houses of different size, however, because a developed city tends to interest much more of bachelor tenants compared to a family tenants due to a higher employability and busier lifestyle. Also in consideration of that bachelor tenants do and does of more prefer on a smaller house size for the lower rent price, smaller space to maintain, and also the monthly bill & maintenance.

Hence, the hypothesis was made up of that smaller size furnished houses can be rent out at a more of a profit price in conjunction to bigger house size.

Supporting on the ‘more of a profit for smaller size house’ is that there will be more demand for smaller size house as discussed, the additional support from ‘furnished house’ is of most of this bachelor tenant will prefer a readily liveable environment house.

# Additional Features

1. theme\_minimal(): An R function in ggplot2 for applying a clean, minimal theme to plots.
2. anova\_result: Outcome of an ANOVA test, used for comparing means across multiple groups.
3. Tukey\_result: Results of Tukey's HSD test, a post-ANOVA analysis to find which group means differ significantly.
4. Scale\_fill\_discrete: ggplot2 function for customizing colors of discrete variables in plots.
5. %>%: The pipe operator in R, used for chaining sequences of operations.
6. mutate(): An R function in dplyr for creating or modifying columns in a data frame.
7. Loess\_Values: Output from LOESS regression, used for scatterplot smoothing.
8. cor\_matrix: A table showing correlation coefficients between variables.
9. lm\_model: A linear model in statistics, used to understand relationships between variables.

# Data preparation

There is essential setup work to be done before we can begin evaluating the house rent dataset. We'll search for unusual data and values that are absent. Inconsistencies or differences will be fixed, especially when it comes to factors like rent and property size, to make sure the results are accurate. To further our understanding, we could even come up with some new data. Finally, the data will be sorted so that it may be easily analyzed and tested. With this configuration, we can be certain that our research is reliable and that we will have every feature necessary to investigate how various property qualities affect rental decisions, giving property owners valuable information for wise decision-making.

## Data import

A computer code with text

Description automatically generated with medium confidence

The dataset that was presented to us was in the file format of .csv (comma-separated values). Therefore, a function of R was called to read the data values from the csv dataset.

A close up of a number

Description automatically generated

Seeable from the dataset, that this dataset contains a total of 4746 rows and 12 columns.

Figure 1: Rent (Before Clean)

A black text on a white background

Description automatically generated

The dataset examines the dimensions and column names of the dataset. Each representing a specific attribute related to the houses available for rent. These attributes include "Posted.On," "BHK", "Rent”, "Size”, "Floor" , "Area.Type" , "Area.Locality", "City," "Furnishing.Status", "Tenant.Preferred", "Bathroom", and "Point.of.Contact" .



This describe includes statistical measures such as mean, median, standard deviation, minimum, maximum, and quartiles for numerical columns. Additionally, categorical columns will showcase the frequency of unique values and the most common value.

Below is the tabular

A screenshot of a computer

Description automatically generated

Figure 3: Rent (Before Clean)

## Data cleaning

Data cleaning is for the purpose of removing outlier data, removing null values, changing data format into consistency and appropriate, as well as changing the attribute name of the dataset into a understandable and appropriate name.

### Removing outlier

#### Rent

A white background with blue text

Description automatically generated

The code performs data cleaning on the 'house\_data\_v0' dataset and creates a new frame 'house\_data\_v1'. It calculates the quartiles of the "Rent" column to identify outliers and replaces rent values over 100,000 with random samples from within the quartile range. No rows are removed.A computer code with blue text

Description automatically generated

The code above employs a histogram as the primary display, where each bar represents a range of rent values.

A graph of a function

Description automatically generatedBelow is the histogram:A graph of a graph showing a number of different colored bars

Description automatically generated with medium confidence

Figure 2: Rent (After Clean)

In summary, figure 1 represents the rent distribution before cleaning, while figure 2 portrays the distribution after cleaning. The code effectively cleans the dataset and presents insightful visualizations to facilitate data understanding.

A black text on a white background

Description automatically generated

The provided code utilizes a boxplot visualization to display ranges of rent values.

A graph with a green line

Description automatically generatedA graph with numbers and lines

Description automatically generatedBelow is the boxplot:

Figure 4: Rent (After Clean)

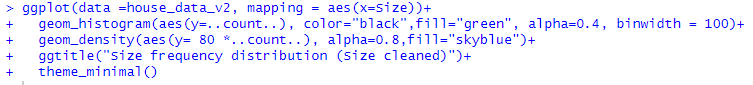
In summary, figure 3 represents the rent distribution before cleaning, while figure 4 portrays the distribution after cleaning.

#### Size

A white background with blue text

Description automatically generated

Coding copies house\_data\_v1 to house\_data\_v2. 'Size' quartiles (25th, 50th, and 75th percentiles) of house\_data\_v2 are computed without NA values. Calculate 'Size' column outliers below or over thresholds (0 and 3500). Replacement of House\_data\_v2 rows with 'Size' values above 3500 with second and third quartile random samples. Viewers show altered data frames. After that, a notice says no lines (rows) were eliminated. Lastly, a message is displayed indicating that no lines (rows) were removed.



The code above employs a histogram as the primary display, where each bar represents a range of size values.

Below is the histogram:

A graph of a number of sizes

Description automatically generatedA graph of a number of green and blue bars

Description automatically generated

Figure 9: BHK

Figure 5: Size (Before Clean)

Figure 6: Size (After Clean)

In summary, figure 5 represents the size before cleaning, while figure 6 portrays the size after cleaning.

A black text on a white background

Description automatically generated

The provided code utilizes a boxplot visualization to display ranges of rent values.

Below is the boxplot:

A graph of a number of sizes

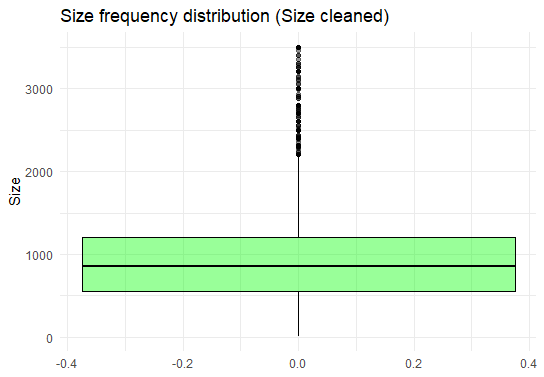
Description automatically generated with medium confidence

Figure 7: Size (Before Clean)

In summary, figure 7 represents the size before cleaning, while figure 8 portrays the size after cleaning.

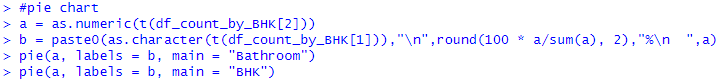
Figure 8: Size (After Clean)

#### BHK

A computer screen shot of a computer code

Description automatically generated

The code analyses home data by bedroom count. The code calculates and assigns n\_uniques\_BHK to house\_data\_v2's unique BHK values. The code sorts the collection by BHK value and counts each unique value. A new data frame, df\_count\_by\_BHK, summarises the findings with two columns: BHK for BHK values and count for their frequency in the dataset. This dataset appears to function without any problems.

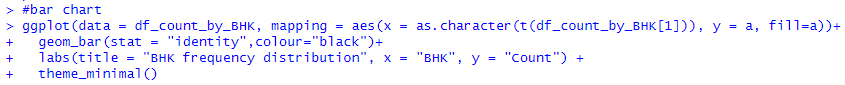


The code builds a pie chart showing the prevalence of BHK (Bedroom, Hall, Kitchen) classes in a dataset. The code computes percentages from BHK type counts.

A pie chart with numbers and a number

Description automatically generatedBelow is the pie chart:

In summary, figure 9 represents the BHK which have six bathroom and the count of each number of bathrooms.



The code uses to build a visually appealing bar plot of BHK (Bedroom, Hall, Kitchen) kinds of frequency distribution.

Below is the bar chart:

A graph with blue bars

Description automatically generated with medium confidence

In summary, figure 10 represents the bar chart of the bathroom which have six BHK and the count of each number of BHK.

Figure 10: BHK

#### Floor

A screenshot of a computer code

Description automatically generated

The code is to perform preprocesses and analyses house\_data\_v2 to determine home floors. First, the code uses n\_distinct () to count the dataset's unique floor values and saves the result in n\_uniques\_Floor. Then, calculate a summary count for each floor.

#### Area Type

A close-up of a white background

Description automatically generated

In this code, we are modifying a house-related dataset. We are creating a new version known as house\_data\_v3. We found that several properties were labelled "Built Area," although they should be called "Carpet Area" instead. Research reveals a closer relationship between "Built Area" and "Carpet Area" than "Super Area". (Figure11*:* (*What Is the Difference Between Super Built up Area and Carpet Area?*, n.d.))

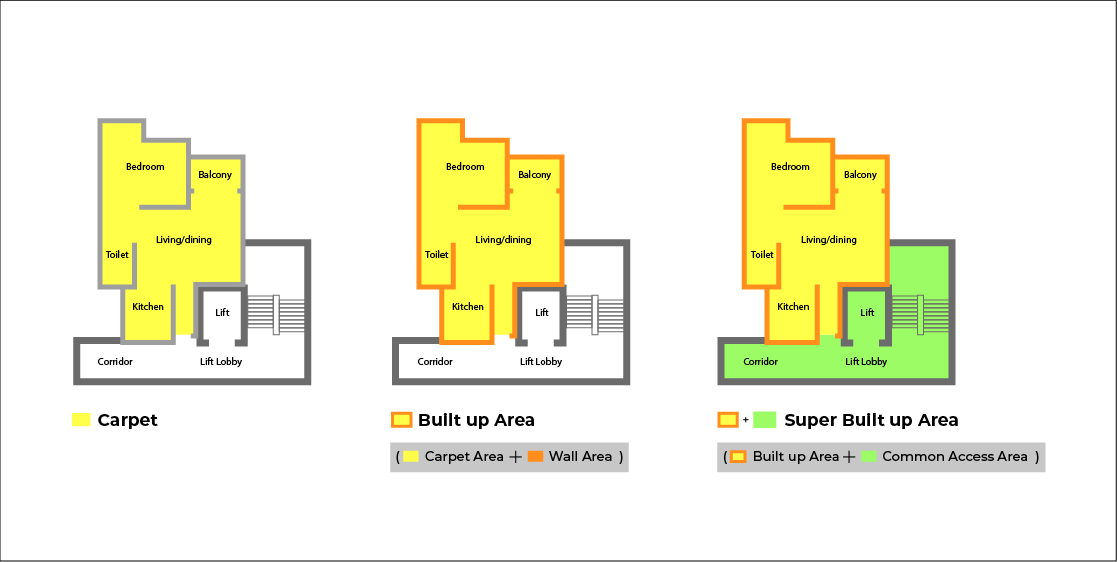


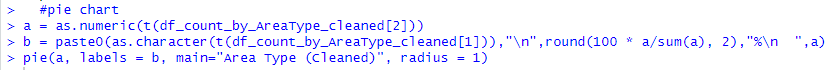
Figure 11: Built Area, Carpet Area, and Super Area

Therefore, we locate the buildings in the sixth column of the dataset and change their designation to "Carpet Area." To clarify and align with how people interpret housing areas, we're replacing "Built Area" with "Carpet Area". This makes the dataset more reliable for examining diverse living situations.

A close-up of a computer code

Description automatically generated

The code creates df\_count\_by\_AreaType\_cleaned by grouping the data by "Area.Type" and using the summarise function. This resulting data frame has two columns: "Area.Type" (such as "Carpet Area" and "Super Area") and "count" (the frequency of each area type in the original data frame). Specifically, the dataset has 2300 "Carpet Area" and 2446 "Super Area" occurrences.



The code creates a pie chart to show the cleaned dataset's area type distribution.

Below is the pie chart:

A pie chart with text on it

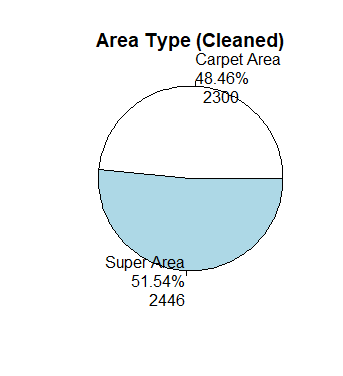
Description automatically generated

Figure 13: Area Type (After Clean)

Figure 12: Area Type (Before Clean)

In summary, figure 12 represents the area before cleaning, while figure 13 portrays the area after cleaning. Notably, it is observed that the cleaning process involved the replacement of "built area" with "carpet area."

#### Area Locality



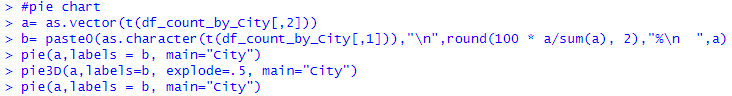
House\_data\_v4 is a copy of house\_data\_v3 created above. The column is cleared by replacing non-numeric Area.Locality values with "NA". IS.na(as.numeric(house\_data\_v4$Area.Locality)) finds rows where Area.Locality cannot be transformed to a numeric value. Those rows have "NA" in column 7. Conditional statements check all non-convertible Area regions.Locality values are now "NA" after cleaning. "Empty" will be written if the cleaning removes all non-convertible values.

#### City

A white background with blue text

Description automatically generated

The code determines that there are six unique cities within a dataset. The data is then grouped by city and the number of records in each city is tallied. The results are organized in a table with two columns: one for the city names and one for the tallies. This table is then displayed, displaying the city names and the number of records for each city. This dataset appears to function without any problems.



The code shows two pie charts are generated using this code to illustrate the distribution of data among several cities.

Below is the 2D and 3D pie chart:

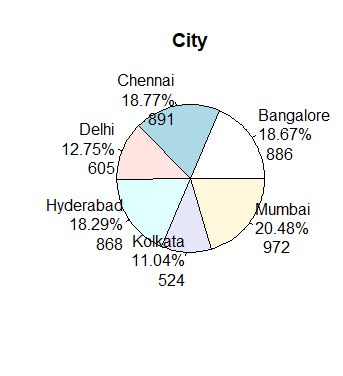
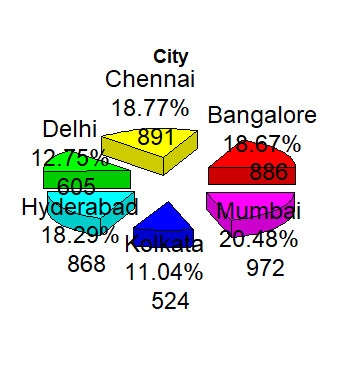
 

Figure 15: City(3D)

Figure 14: City(2D)

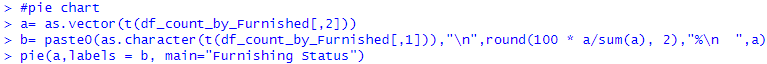
In summary, figure 14 represents the city in 2D and the figure 15 represents the city in 3D. The two pie charts show city counts and combined into a single group.

#### Furnishing Status

A computer screen shot of a computer code

Description automatically generated

The code shows the dataset investigates home attributes. Calculating the number of unique categories in the "Furnishing.Status" column yields three types: furnished, semi-furnished, and unfurnished. The method summarizes the number of households in each furnishing status after classifying the data. The dataset has 680 furnished residences, 2251 semi-furnished dwellings, and 1815 unfurnished buildings, according to the output. This dataset works fine.



A pie chart with text

Description automatically generatedThe code builds a pie chart from home furnishing status data. The function first converts provisioning status numbers from df\_count\_by\_Furnished. The code appears to generate data for a pie chart showing each house's furnishing status percentages and counts.

Figure 16: Furnishing Status

Beside Figure 16 is the 2D pie chart:

#### Tenants Preferred

A computer screen shot of a computer

Description automatically generated

The code analyses tenant housing choices. Bachelors, Bachelors/Family, and Family are the first unique categories in the "Tenant.Preferred" column. The programme then counts residences by renter preference. Bachelors prefer 830 homes, couples 3444, and families 472, according to statistics. This dataset appears to function without any problems.

The code provides a pie chart of tenant preferences by house. After vectorizing summary tenant choice category counts, the function constructs pie chart labels.

Below is the pie chart:

A pie chart with text

Description automatically generated

Figure 17: Tenants Preferred

#### 

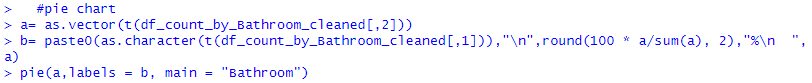
#### Bathroom

A close-up of a data

Description automatically generated

A computer screen with text

Description automatically generated

This code removes records from house\_data\_v4 to create house\_data\_v5. The removal is predicated on a "Bathroom" column value less than 10. The cleaned house\_data\_v5 dataset is analysed next. The code groups and summarises toilet values to count occurrences. The result is tabulated, showing bathroom value categories and counts.

This code makes a pie chart showing the distribution of bathrooms in a dataset of households. It then labels the pie chart with the bathroom count, the dataset proportion, and the actual house count for each category.

Below is the pie chart:

A pie chart with numbers and a number of objects

Description automatically generated A pie chart with numbers and a number

Description automatically generated

Figure 18: Bathroom (Before Clean)

Figure 19: Bathroom (After Clean)

In summary, figure 18 represents the bathroom before cleaning, while figure 19 portrays the bathroom after cleaning. The data cleaning resulted in the removal of instances with a count of 1 bathroom.

A computer code with blue text

Description automatically generated

The code utilizes ggplot2 to build a bar chart indicating home bathroom distribution. The code utilizes plot: Setting the dataset to df\_count\_by\_Bathroom produces an aesthetically mapping with b for bathroom number and a for bathroom percentage.

Below is the bar chart:

A graph of bathroom and bathroom

Description automatically generated with medium confidence A graph of bathroom with numbers and numbers

Description automatically generated with medium confidence

Figure 20: Bathroom (Before Clean)

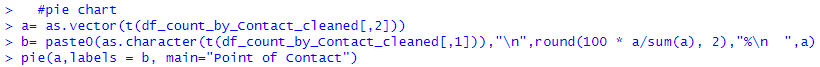
Figure 21: Bathroom (After Clean)

In summary, figure 20 represents the bathroom before cleaning, while figure 21 portrays the bathroom after cleaning. The data cleaning resulted in the removal of instances with a count of 1 bathroom. These visualizations effectively convey the impact of the cleaning process on the bathroom data and highlight the alterations made to enhance data quality and accuracy.

#### Point of Contact

A computer screen shot of a computer code

Description automatically generated

Code for home contact points is cleaned and updated here. The "Point.of.Contact" column's most common "mode" is "Contact Owner" at 3216. Copying the original dataset to house\_data\_v5. This dataset's "Point.of.Contact" labels change from "Contact Builder" to "Contact Owner" after indexing. Df\_count\_by\_Contact\_cleaned is built from the corrected dataset. According to the distribution of two main contact types—"Contact Agent" with 1529 occurrences and "Contact Owner" with 3217 instances—study contact preferences more realistically.

A pie chart is built in this dataset to show the distribution of home contact points in a cleaned dataset.

Below is the pie chart:

A pie chart with numbers and a number of people

Description automatically generated **A pie chart with numbers and a number of people

Description automatically generated**

Figure 21: Point of Contact (Before Clean)

Figure 22: Point of Contact (After Clean)

The visualisations show dataset changes before and after cleaning. Figure 21 shows the bathroom distribution before cleaning, and Figure 22 shows it after. The cleaning process replaced "Contact Builder" with "Contact Owner."

### Remove NA Data

A close-up of a computer screen

Description automatically generated

In house\_data\_v6, the code snippet fixes missing or NA values. Initially, is.na() and colSums() count column missing values. Next counts indicate no missing values across columns, verifying the dataset's completeness. House\_data\_v6\_noChange uses na.omit() to delete NA rows. Finally, the code confirms that row counts did not change between house\_data\_v6 and house\_data\_v6\_noChange.

### Remove Duplicate Data

A close-up of a math equation

Description automatically generated

The house\_data\_v6 function removes duplicates. The code creates house\_data\_v6\_noChange from the original dataset's unique records using unique(). With a count of 0, house\_data\_v6\_noChange shows that row counts have not changed.

### Change Data Format

#### Date

A computer code with blue text

Description automatically generated

A screenshot of a computer

Description automatically generatedThe code formats the house\_data\_v6 "Posted.On" column. First, "Posted.On" is confirmed as a date data type. The next loop successively examines "Posted.On" date strings. Parse\_date\_time() from the lubridate package applies several date formats in this loop. Format() converts parsed dates to "YYYY-MM-DD".

Beside is the Date format after changed (Figure 22):

Figure 22: Date Format

#### Check All Format

A screenshot of a computer program

Description automatically generated

The code above closely explores house\_data\_v7 column data types. It reveals each column's data type. Since "BHK," "Rent," "Size," "Bathroom," and "Point.of.Contact" are "integer" data types, "Posted.On" is "Date". These "character" columns are "Floor," "Area.Type," "Area.Locality," "City," "Furnishing.Status," and "Tenant.Preferred". This thorough examination illuminates each column's data, can be certain the dataset are no mistakes.

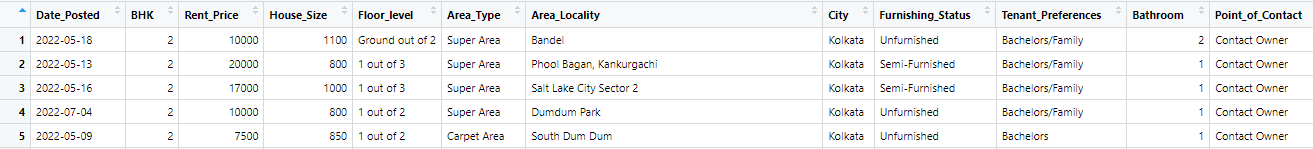
#### Change Data Frame Name

A screenshot of a computer program

Description automatically generated

The code segment renames columns in house\_data\_v6 to create a new dataset entitled house\_data\_v8 with revised column names. The data.frame() function selects columns from the original dataset and renames them to match their content. This transformation involves multiple columns: "Posted.On" becomes "Date\_Posted," "Rent" becomes "Rent\_Price," "Size" becomes "House\_Size," The short form of "Floor" is "Floor\_level," "Tenant.Preferred" becomes "Tenant\_Preferences," "Bathroom" is "No\_of\_Bathroom," and "Point.of.Contact" is "Point\_of\_Contact." The updated dataset, house\_data\_v8, has redesigned column names.

Below is the new columns name of the dataset:



## Data Preprocessing

### Add On Category

#### Size Category

A computer screen shot of a computer code

Description automatically generated

A screenshot of a phone

Description automatically generatedThe provided code performs house\_data\_v8 activities. The "House\_Size" column is summarized to offer statistical measurements such lowest, first quartile, median, mean, third quartile, and maximum. A new character vector named "size\_category" is then created with a length equal to the dataset's rows. Using a loop, each row's "House\_Size" value is checked and assigned a size category label ("Small," "Medium," or "Large") based on certain conditions. Any unusual value generates an error message. The "size\_category" vector is then joined with house\_data\_v8 using cbind() to create house\_data\_v9, which contains the size category information.

Beside is the size category (Figure 23):

Figure 23: Size Category

A close-up of a code

Description automatically generated

The code creates a pie chart to show house\_data\_v9's size categories. The variables small, medium, and large are produced by counting each size category in the dataset.

Below is the pie chart:

A pie chart with numbers and a pie chart

Description automatically generated

#### Floor Category

A screenshot of a computer program

Description automatically generated

The code transforms house\_data\_v9 floor data. Floor\_category vectors initially store Floor\_level column floor category names. A loop splits each row's Floor\_level, and the first member is stored in the floor\_category vector to reflect the floor category.The floor\_category vector and house\_data\_v9 create house\_data\_v10. Classifying floor levels like "Ground," "Lower Basement," and "Upper Basement" in the code.Use n\_distinct() to get 54 floor types. In addition, the df\_count\_by\_Floor\_cleaned dataframe shows floor category occurrences. Descent by count reveals floor category distribution and counts in the dataframe.

#### Remove unused column for data mining analysis



The code selects columns from house\_data\_v10 using select(). "Date\_Posted," "Floor\_level," and "Area\_Locality" are removed from house\_data\_v11 because these column is unused in the analysis.

Below is the final dataset to use:

A screenshot of a computer

Description automatically generated



After done all the cleaning and processed, the "house\_data\_v11" dataset contains a total of 4745 rows and 11 columns.

# Exploratory Data Analysis (EDA)

## General Analysis

#### General Analysis 1: City Record Distribution

A graph of different colored rectangular shapes

Description automatically generated with medium confidence

Figure 1: City record distribution (Bar)

A pie chart with numbers and a circle

Description automatically generated

Figure 2: : City Record Distribution (Pie)

As seen from the Figure G-1 & Figure G-2 above, the number of record from the 4745 rows of record is quite evenly distributed throughout the 6 cities, with the Mumbai having a slightly more record; and Kolkata having a little less piece of cake at 524, however it is still in a good amount of row to be taken for EDA.

#### General Analysis 2: Attribute correlation heatmapA screenshot of a computer screen Description automatically generated

Figure G-3 show 9 of the 11 attribute (City & Size category not in calculationq1) and their correlation with each other.

Some can be observed of a quite strong correlation between BHK, House Size, & No\_of\_Bathroom, attempting to state:

**“““  
The larger the House Size,   
the larger the value of BHK & No\_of\_Bathroom  
”””**

q1 = City is not an *Ordinal* data type, thus it couldn’t compare with other attribute which is compare for their ordinal data value. Size category is a postprocess of House Size attribute, both are talking about the same thing, thus it is not included.

tech =A screenshot of a computer code

Description automatically generated

Figure G-4: Conversion of discrete attribute into continuous(attempted)(R code)

* Converting the “Ground”, “Upper Basement”, & “Lower Basement” data into “0”, “-1”,& “-2” respectively, this is so it make sense that to continue the counting before the 1,2,3,4,… floor.
* Converting the “Unfurnished”, “Semi-Furnished”, “Furnished”, data into “0”, “1”, “2” respectively, making the furnished to be the higher one for an assumption of Furnished house will have higher Rent\_Price (compared to dependent variable).
* Converting the “Contact Owner”, “Contact Agent” data into “0”, “1” respectively, making Contact agent to be higher for an assumption of Contact Agent will have higher Rent\_Price (compared to dependent variable)
* Converting the “Bachelors”, “Bachelors/Family”, “Family” data into “0”, “1”, “2” respectively, making Family to be higher for an assumption of Family will have higher Rent\_Price (compared to dependent variable)
* Converting the “Super Area”, “Carpet Area” data into “0”, “1” respectively, making Carpet Area to be higher for an assumption of Carpet Area will have higher Rent\_Price (compared to dependent variable)

A screenshot of a computer

Description automatically generated

Figure G-5: Sample output of post-conversion attributes

As seen at Figure G-5, all the 5 mentioned attribute data have been convert into number for the later attempt of correlation graph plot & *general analysis 3: gradient relationship heatmap*.

#### General analysis 3: gradient relationship heatmap

A screenshot of a computer screen

Description automatically generated

Figure G-6: gradient relationship (heatmap)

Figure G-6 show the gradient of a linear regression line if plotted on the graph of Var1 vs Var2, for example, a graph plot with x: BHK vs y:House\_Size, will yield a proportion gradient of 0.8. the proportion is based for the attribute of data to have the range scale from 0:max(attribute) into 0:1 tech.

It can be observed that for the dependent variable – Rent\_Price, it is mostly & strongly affected by No\_of\_Bathroom, BHK, floor\_category, & House\_Size.

***“““***

***The Rent\_Price is strongly influenced by:***

***No\_of\_Bathroom, BHK, floor\_category and House\_Size,***

***ranked***

***”””***

tech = A computer screen shot of a program

Description automatically generated

Figure G-6: Conversion of continuous data into proportion continuous from 0 to 1 (R code)

Using for loop to loop through the whole record of data frame, each attribute data is then convert into a proportion number from 0 to 1, by utilising the attribute data and divide it with the max(attribute), for example, a BHK value of 2 will be divide with 6 (maximum value of BHK in data frame), which then give a result of 0.33.

A screenshot of a table

Description automatically generated

Figure G-7: sample output of gradient of post-proportion conversion attribute

Figure G-7 show some a few of the output gradient data frame that are formed from the conversion of attribute data into a range between 0 and 1. For example, the x:BHK vs y:Rent-Price give a gradient of 0.54, meaning that relatively proportionally speaking, an increase in 1.00 BHK, will yield a proportionally increment of 0.54 Rent\_Price.

## Hypothesis proving

A screenshot of a computer

Description automatically generated

As the hypothesis stated – “Furnished houses at different cities, yield the same result of 80% smaller size house having a profit rent price”.

### Conclusion:

Hypothesis correctness: partially-correct

Finding: Not all city yield 80% of smaller furnished size house having profit rent price. It is only true in 3 of the 6 Cities – Hyderabad, Chennai, & Kolkata.

A screenshot of a computer screen

Description automatically generated

Figure H-1: house size vs rent price for each size category for furnishing status

## Foo Yau Yit

### Objective - smaller size house make more profit compared to bigger size

#### Objective Analysis 1: City record distribution

A graph of different colored bars

Description automatically generated

Figure F-1: City vs Record Distribution in each Size category (complex bar)

It was able to be observed that proportionally speaking:

* Delhi & Mumbai have more small size houses
* Chennai has more medium size houses
* Hyderabad has more large size houses

Assumption:

* Mumbai & Delhi is a high population density City (Ronald, 2021). That is why there is a high demand for housing land, therefore most of the houses are smaller in size to accommodate the population
* For similar reason, Chennai with a relatively average population density in India (Ronald, 2021). Yield it to having more medium size house
* Same old same reason, Hyderabad is the most relatively low population density in India (Ronald, 2021). Hence, of it having more large size house

#### Objective Analysis 2: Size vs rent

A graph showing a line of dots

Description automatically generated with medium confidence

Figure F-2: House Size vs Rent\_Price (scatter)

Figure F-2 show that a positive gradient increase for house size vs rent price. Meaning a larger house size, the higher the rent price.

A graph of different colored dots

Description automatically generated

Figure F-3: House Size vs Rent Price for each City (Scatter)

Figure F-3 show that in most case of the cities, the larger the house size, the higher the rent price. However, Mumbai unique out as the larger the house size, the lower the rent price.assumption

assumption = it is heavily influenced by other attributes, such of strong rent price influenced attribute are less in favor with Mumbai large size house.

A screenshot of a computer screen

Description automatically generated

Figure F-4: House Size vs Rent Prive for each City and each Size category (scatter)

Figure F-4 further detailed the figure F-3 into each size category to observe the linear regression relationship.

The in depth observe of Mumbai show that the Rent Price increase strongly at the small size category, and begin to reduce as it approach Medium size, and then becoming negative relationship when in large size category.

A screenshot of a computer

Description automatically generatedFigure F-5 show the value of gradient for each City and each Size category. Further detailing on the said observation.

Figure F-5: table of gradient for each city & each size

A chart with different colored squares

Description automatically generated

Figure F-6: City vs Rent Price (boxplot)

Figure F-6 show of that Mumbai has the highest average rent.

A screenshot of a graph

Description automatically generated

Figure F-7: House Size vs Rent Price for each size category & each City (boxplot)

Figure F-7 show that within the highest average rent price of Mumbai shown in Figure F-6, the Medium size house has the highest average Rent Price in Mumbai. In contrast to other 5 cities, where Large size has the highest average Rent Price.

#### Objective Analysis 3: Size vs BHK

A diagram of different colored shapes

Description automatically generated

Figure F-8: Size category vs BHK (+mean line) for each City (Violin)

Figure F-8 show a healthy positive relation between size category and BHK, meaning that the larger the house, the larger the BHK.

#### Objective Analysis 4: Size vs Area Type

A graph of a graph showing a couple of blue and red shapes

Description automatically generated with medium confidence

Figure F-9: Area Type vs House Size (Boxplot & Violin)

Figure F-10 show that there is a slight biastest by “Carpet Type” on the value of house size, however on the big scale, the Figure F-9 show that this doesn’t greatly impact the overall house size as exaggerated as a seeable pattern.

A white background with black text

Description automatically generatedFigure F-10 state the slight differences in mean. This slight bias isn’t random as test shown.

Figure F-10: mean of house Size for each Area Type

test = t-test references (appendix)

#### Objective Analysis 5: Size vs Furnishing Status

A screenshot of a graph

Description automatically generated

Figure F-11: Furnishing Status vs House Size for each City (Boxplot & Violin)

Figure F-11 show that there are a higher impact of both Furnished & Un-Furnished houses on the house-size, however unfurnished house are more typical to be a smaller house size.

This also imply that smaller size house are more common to be “Unfurnished”

A graph of different colored bars

Description automatically generated with medium confidence

Figure F-12: Furnishing Status vs Record Distribution vs Size category for each City (complex bar)

Figure F-12 show that from the Figure F-1, further classifying each size-city into their furnishing status, and can be seen that Mumbai has many of the small size house at Unfurnishing status.

#### Objective Analysis 6: Size vs Tenant preferences

A diagram of different colored shapes

Description automatically generated

Figure F-13: Tenant Preferences vs Size (boxplot & violin)

Figure F-13 show that in more common, that a larger house size will be more preferably to be rent to family, in contrast to Bachelor & Bachelor/Family.

Assumption =

* it make sense for that a bigger house size will be more suitable for Family Tenants than Bachelor.
* A pie chart with numbers and a blue circle

  Description automatically generatedBetween Bachelor and Bachelor/Family, it should have imply that Bachelor/Family will have a slightly higher house size preference in compare to bachelor, however within psychology effect opinion study, an owner will much more wish for their houses to be expose to more people for more opportunity, that’s why many owner will go with both the selection – “Bachelors/Family”, compare to just “Bachelors” only. Figure F-14 at the right has fully support on this assumption.

Figure F-14: Tenant Preference record distribution (pie)

#### Objective Analysis 7: Size vs Bathroom

A graph with different colored squares

Description automatically generated

Figure F-15: No of Bathroom vs House size (boxplot)

Figure F-15 show a steady positive relationship between no of bathroom a house have and the house size.

#### Objective Analysis 8: Size vs Point of Contact

A screenshot of a graph

Description automatically generated

Figure F-15: Point of Contact vs House Size (boxplot & violin)

Figure F-15 show Contact Agent is more favour with higher House Size compare to Contact Owner.

Assumption: A larger house size house might be expose to lesser people/ smaller market, as mostly larger house size will be more favour for family, which is in opinion lesser in proportion in the market in contrast to bachelors. Thus, owner let agent to handle the renting as agent are more expertise and have a better/bigger approach on the market of family tenants.

#### Objective Analysis 9: Size vs Floor Category

A green line and blue dots

Description automatically generated

Figure F-16: Floor Category vs House Size

Figure F-16 show that with increasing floor level, the house Size also increase.

### Question: Which City & which size is the most ideal to get most profit? why?

A screenshot of a computer

Description automatically generated

Figure F-16: result of most profit combination attribute on Rent\_price

City: Mumbai

Size: Small

Combination occurred at top 10: **3**

Why?:

**A screenshot of a computer screen

Description automatically generated**

Figure F-17: heatmap of gradient relationship in Mumbai

According to this heatmap of the attribute gradient relationship with each other in Mumbai, we can see that Rent\_Price is most influenced by:

***“““***

***Ranked***

***Furnishing\_Status -> Area\_Type***

***”*** ***””***

* And Mumbai Small houses has the highest proportion of furnished house as shown at figure  
  A graph with different colored rectangles

  Description automatically generated
* And Mumbai Small houses has the highest proportion of Carpet Area house as shown at figure  
  A graph of different colored squares

  Description automatically generated
* And Mumbai Small houses has the highest mean Rent

Correction explanation:

It is seen that in Figure F-16, the BHK doesn’t have great impact on the Rent\_Price, however this doesn’t indicate that it doesn’t exactly have a relationship or have opposing relationship with the rent\_price, as the occurrence count is doesn’t add up to great number, a t-test was taken to check the null hypothesis of BHK having no effect on the Rent Price.   
A screenshot of a computer code

Description automatically generated  
Figure F-20: t-test of BHK\_1 & BHK\_2

Can be seen in Figure F-20, the p-value is 0.2465, which is much larger than the rejecting null hypo value of p=0.05, meaning that the BHK in F-16 doesn’ indicate much pattern and impact on the rent\_price, and cannot be assumed of any patter.

Conclusion: Therefore in conclusion, Furnishing\_Status and Area\_Type attribute has the largest impact on the Rent\_Price of a houses, which directly impacting the profit as well, and Mumbai with small size\_category has the most frequent count in this 2 attribute.

Conclusion

In line with our objective, the analysis reveals a dynamic interplay between rent pricing, the number of bedrooms (BHKs), bathroom counts, and tenant preferences. The study underscores the necessity to consider nuanced factors when constructing and marketing housing units to cater to different tenant demographics. It also validates our objective that houses with more BHKs and bathrooms, tailored for tenants who are either Bachelors or part of a Family, are likely to command higher rent prices.

Causes of Higher Rent Prices:

1. Number of BHKs:

* Bachelors: Bachelors are more likely to choose studio or one-bedroom apartments, but they are willing to pay more for flats with more bedrooms if they have more bathrooms.
* Family/bachelors: This group seems to be more flexible and open to a variety of BHK layouts. They care most about bathrooms, and a higher BHK number usually means a higher rent price, especially if there are also more toilets.
* Families: Choose more than one. They are ready to pay more for a 1 or 2 BHK home with more bathrooms, but not for a 3 BHK home with more bathrooms.

2. Number of bathrooms:

* Bachelors ：Bachelors are willing to pay more for more bathrooms, no matter how many BHK there are.
* Family/bachelors: Family/bachelors are willing to pay more for more bathrooms.
* Families: Families are ready to pay more for 1 and 2 BHK homes with more than one bathroom. But this doesn't hold true for 3 BHK houses.

3. Tenant Preferences:

* Bachelors should pay more attention to bathrooms, since they probably value privacy and ease of use.
* Family/bachelors: Again, bathrooms are a top concern, which shows that this mixed group wants both comfort and functionality.
* Family: Families value more complicated factors when it comes to 3 BHK homes, which means that location, amenities, and square footage may come into play.

# References

Bard, Google AI. "Bangalore, the City of Development” Bard, Google AI, 2023-07-27. Accessed 2023-07-27. <https://g.co/bard/share/4ffd425d6025>

Srinivas, S. (1997). A STUDY OF BANGALORE, INDIA. <https://core.ac.uk/download/pdf/1902776.pdf>

Bard, Google AI. " Chennai, a developed city” Bard, Google AI, 2023-07-27. Accessed 2023-07-27. <https://g.co/bard/share/9a0c0e52793c>

Kumar, N. (2022). Urban Development Analysis of chennai. Academia.edu. <https://www.academia.edu/35478404/Urban_Development_Analysis_of_chennai>

Bard, Google AI. " Delhi, a developed city” Bard, Google AI, 2023-07-29. Accessed 2023-07-29. <https://g.co/bard/share/38e11072e634>

Aggarwal, S. (2010). Delhi Towards a Healthy City. ResearchGate; unknown. <https://www.researchgate.net/publication/301556562_Delhi_Towards_a_Healthy_City>

Bard, Google AI. " Hyderabad, a developed city” Bard, Google AI, 2023-07-29. Accessed 2023-07-29. <https://g.co/bard/share/8ab33b86b694>

Bard, Google AI. " Kolkata, a developed city” Bard, Google AI, 2023-07-29. Accessed 2023-07-29. <https://g.co/bard/share/dddcb989f20f>

Bard, Google AI. " Mumbai, a developed city” Bard, Google AI, 2023-07-29. Accessed 2023-07-29. <https://g.co/bard/share/454a0dec7943>

*What Is The Difference Between Super Built Up Area And Carpet Area?* (n.d.). What Is the Difference Between Super Built up Area and Carpet Area? <https://www.mypropertyboutique.com/blog/what-is-the-difference-between-super-built-up-area-and-carpet-area->

Ronald. (2021, May 30). Top 10 Populated Cities In India. India Internet Player Society (IIPS). https://iipsindia.org/top-10-populated-cities-in-india/