A Project Report

on

House Prediction Problem using Machine Learning

Submitted in partial fulfillment of the requirement for the award of the degree of

Bachelor of Technology in Computer Science and Engineering

+

Under The Supervision
of

Mr. V. Arul

Assistant Professor

Department of Computer Science and Engineering

Submitted By-Aaditya Prakash Pillai-
18021120023/18SCSE1120025

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING GALGOTIAS UNIVERSITY, GREATER NOIDA
DECEMBER - 2021

SCHOOL OF COMPUTING SCIENCE ANDENGINEERING

GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the project, entitled **" A housing prediction app using machine learning"** in partial fulfillment of the requirements for the award of the **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING**

submitted in the **School of Computing Science and Engineering** of Galgotias University, GreaterNoida, is an original work carried out during the period of **JULY-2021 to DECEMBER-2021**, under the supervision of **Mr.V. ARUL, Assistant Professor**, **Department of Computer Science and Engineering** of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the project has not been submitted by me/us for the award of any otherdegree of this or any other places.

18SCSE1120025-AADITYA PRAKASH PILLAI

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor (Mr.V.Arul,

Assistant Professor)

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of **18SCSE1120025 – AADITYA PRAKASH,** has been held on _____and his/herwork is recommended for the award of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING**.

Signature of Examiner(s) Signature of Supervisor(s)

Signature of Project Coordinator **Signature of Dean**

D

t

e

:

P

l

a

c

e

:

TABLE OF CONTENTS

ABSTRACT

The propose of the project is to implement a house price prediction model of some properties in some cities/towns in India. It's a Machine Learning (ML) model which analyses how housing prices fluctuate daily and are sometimes exaggerated rather than based on worth. I tend to study and employ some machine learning algorithms to train the available dataset and to correctly predict the price for those properties whose prices are not known.

The major focus of this project is on extracting the genuine feature variables using the ML, from an existing dataset and apply ML algorithms for generating prediction models using factors. In this project the focus is on finding the relative best predictive method on the housing dataset with relative less error, thereby increasing accuracy.

Through this project I intend to design an application that can get the most accurate price for the property after evaluating the accuracy of my ML models. The goal of this project is to learn Python and get experience in Data Analytics and Machine Learning.

Acronyms

B.Tech.	Bachelor of Technology
M.Tech.	Master of Technology
BCA	Bachelor of Computer Applications
MCA	Master of Computer Applications
B.Sc. (CS)	Bachelor of Science in Computer Science
M.Sc. (CS)	Master of Science in Computer Science
SCSE	School of Computing Science and Engineering

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

The prediction of accurate housing price has become an essential task for the real-estate parties as there is demand for suitable property either for investment or for personal use, for the ever-growing demands of our country's population.

House prices changes every year, so it is mandatory for a structure to foresee house prices in the future. House price prediction can help in fixing and thereby predicting house prices and customer can evaluate it. Our intension is to employ predictive modelling techniques in machine learning domain to accurately find out house prices using several machine learning techniques.

House price depends on various factors like area, bedrooms, bathrooms, location, drawing room, material used in house, interiors, parking area and mainly on square feet per area. My intention behind proposing this paper is to employ different machine learning techniques for predicting the price based on these metrics.

Some of the challenges include

- Understanding the variations that affect the house price such as the city or town located, new property or resale property, "ready to move in" or "under construction" and so on.
- Selling of old properties is also a challenging task as it is costly to renovate all the property. The price of a property may be affected depending on whether it is a resale property or freshy launched property.
- Data cleaning as many rows have missing data or NAN values.

In this project I have selected this problem to design and provide a solution for giving the best possible prediction for the sale price of a particular property using machine learning algorithms.

Chapter 2

Literature Survey

In this chapter, I am discussing the important aspects of certain papers that I have referred to understand the problem related to prediction of house prices. In these papers the authors, are discussing the different machine learning techniques to solve the House price prediction problem. In the paper [4], the authors have discussed the house price prediction in Bangkok. This paper was trying to find a solution to help property dealers to evaluate the price of their property set and to help their customers make a purchase.

Table 1 - Literature Review

Paper no.	Year	Paper Title	Author	Machine Learning Algorithm implemented / discussed	Comments
1	2020	An Overview of Real Estate Modelling Techniques for House Price Prediction	Mohd, T., Jamil, N. S., Johari, N., Abdullah, L., & Masrom, S.	Linear Regression Time Series	None (Review Paper)
2	2021	Ensemble of Supervised and Unsupervised Learning Models to Predict a Profitable Business Decision	M Heidari, S Zad, S Rafatirad	Linear regression (LR), Decision tree (DT), Random Forest (RF), K-nearest neighbour (KNN), Partial least square (PLS), Naïve bayes (NB), Multiple regression analysis (MRA),Spatial Analysis (SA), Gradient boosting (GB), Ridge Regression, Lasso Regression and Ensemble learning model (ELM).	
3	2018	Identifying Real Estate Opportunities Using Machine Learning [5]	Baldominos, Alejandro, et al		Public listing in Spain
4	2021	House Price and Renovation Prediction Analysis Using Different Machine Learning Algorithms[3]	M. Naga Srinivasa Karthik, M. Rahul Sai Krishna, A. Mary Posonia		

5	2020	Real Estate Value Prediction Using Linear Regression [1]	N. N. Ghosalkar and S. N. Dhage	*Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*	To predict house prices in Mumbai city with Linear Regression. Linear Regression method gave least error
6	2017	Predicting the housing price direction using machine learning techniques [2]	D. Banerjee and S. Dutta,	*IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, 2017	This work considers the issue of changing house price as a classification problem and applies machine learning techniques to predict whether house prices will rise or fall.
7	2017	Modeling House Price Prediction using Regression Analysis and Particle Swarm Optimization[4]	Alfiyatin, Adyan Nur, et al.	*International Journal of Advanced Computer Science and Applications* 8.10 (2017)	Predict the house prices in Bangkok, Thailand, to assist the estate developers to evaluate their property

					and allow customers to decide the right time to buy property.

Chapter 3

Data Handling

Data is often incomplete, inconsistent, and quite often difficult to collect and share. However, the emergence of Data engineering, Data Science and other allied fields have made data acquisition, data processing and data analysis, much more approachable and very popular across several domains. However, data is not available in readily processable format. It has to be preprocessed before it can be selected for any kind of analysis.

The dangers of using raw data is that such unprocessed data can lead to analysts getting incorrect understanding of the data. Information made on incorrect understanding can lead to wrong decisions. Wrong decisions will seriously affect the organizations' decision-making process at all levels of the management and affect the department branches day to day operations and forecasted.[6]

Types of data - Data is mainly divided into two types, quantitative and qualitative.

- Qualitative – They are of two types, Nominal & Ordinal, Nominal is the yes or no type of variables, whereas Ordinal variables have a finite set of distinct values (Excellent, V.Good, Good, Fair ,Poor etc.)
- Quantitative - They are of two types, Discrete and Continuous. Discrete values pertain to values which are fixed and never changing with time. They are obtained by counting sometimes like the number of gold medals won by a country or athlete at the Olympics in a particular year or the number of arrival flights on a particular day, in an airport . The value of continuous variables may change like temperature of a place, price of an item etc.

Centrality Measures – These values help us identify the central tendencies in the data

- Mean is equal to the sum of all the values in the data set divided by the number of values in the data set.
- The median is the middle score for a set of data that has been arranged in order of magnitude.
- The mode is the most frequent score in our data set. (This is the only central tendency measure that can be used with nominal data, which have purely qualitative category assignments.)
- If these 3 values are equal then we can say that we have a uniform distribution.
- The normal distribution is the probability distribution that is symmetric about the mean. It is also known as bell curve.
- Properties: In a standard normal distribution, mean is zero and standard deviation is 1. It has a zero skewness. In such a distribution, the Mean = Median = Mode

Identifying the dispersion in data

- Measures of dispersion: It indicates how large the spread of distribution in around the central tendency.
- Range: Range is the simplest of all measures of dispersion. It is calculated as the difference between maximum and minimum value in dataset.

  Range = X(maximum) - X(minimum)

- Interquartile range (IQR): It is a measure of variability, based on dividing a data set into quartiles i.e. into four parts represented by Q1, Q2,Q3 and Q4. The IQR is least affected by outliers.

  IQR = Q3 - Q1

- Standard Deviation: It is a measure of how spread out the numbers in a distribution are. It is the measure of dispersion of a data from its mean. Denominator term for Sample is n-1

  Assume a population with "N" items. Suppose that we want to take samples of size "n" from that population. If we could list all possible samples of "n" items that could be selected from the population of "N" items, then we could find the SD for each possible sample.

1. Histogram is bar chart which represents a frequency distribution.

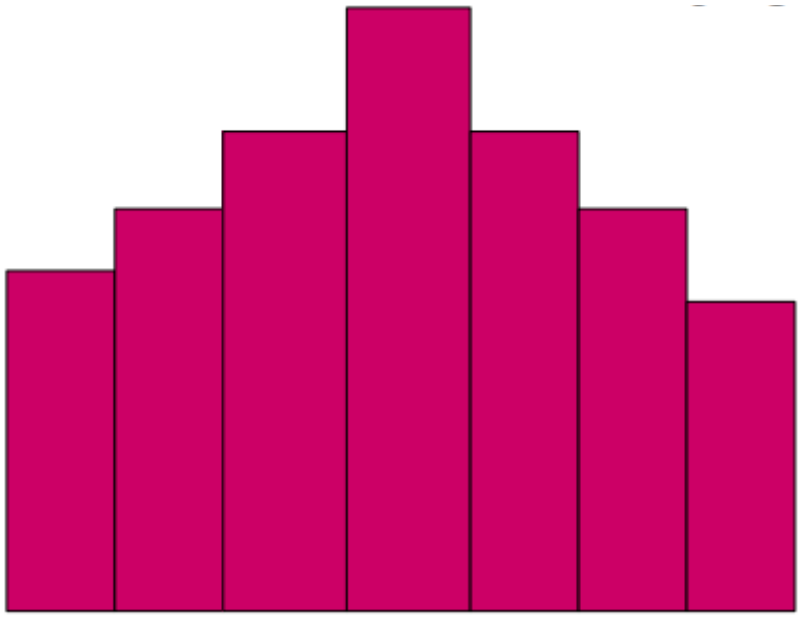


Figure 1 Histogram

Horizontal axis of the histogram represents the data points within an interval called as "bin" and vertical axis represents the corresponding "frequency"

Box and Whisker Plot

It displays the Five-Point summary of the data i.e., minimum, first quartile, median, third quartile, and maximum

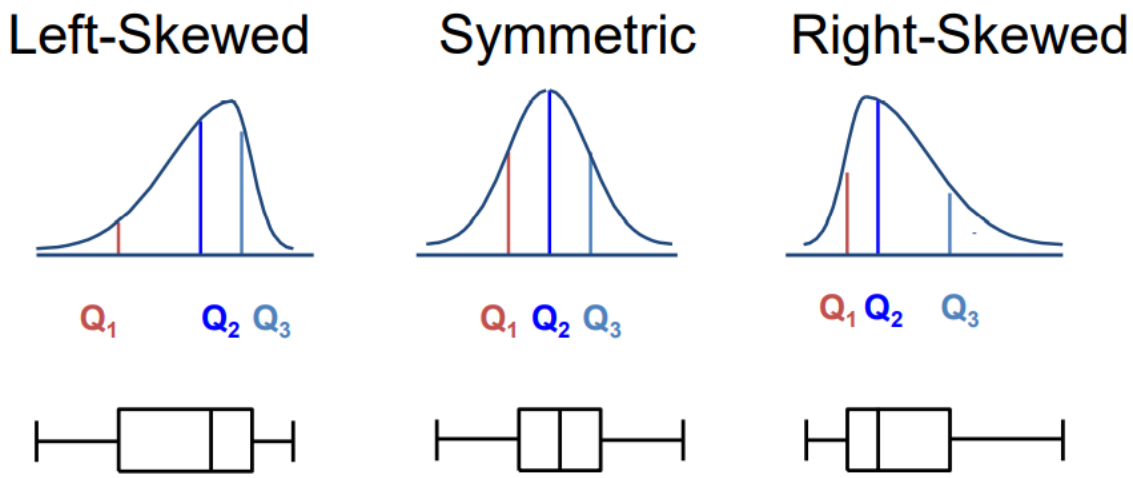


Figure 2 Box Plot and Distribution Shape

In a box plot, box is from the first quartile to the third quartile.

A vertical line goes through the box at the median.

- Minimum value represented through a Whisker is (Q1 -1.5* IQR)
- Maximum value represented through a Whisker is (Q3 + 1.5* IQR)

Any point which is below the Minimum Value and/or above the Maximum value is an outlier.

Data Cleaning - It is very important to deal with outliers and Null or NAN values before analysis on the dataset.

CHAPTER 4

Software and Hardware Specifications

Software Requirement

1. Anaconda Navigator and Jupyter Notebook
2. Python Version

Step 1 - Anaconda Navigator Installation – This is an open source, free toolkit for single users to perform data science tasks. Download the Anaconda Navigator from the official site. Select the Installer for the Windows Operating System.

Figure 3 Anaconda Installers

Figure 4 Anaconda Installation Window

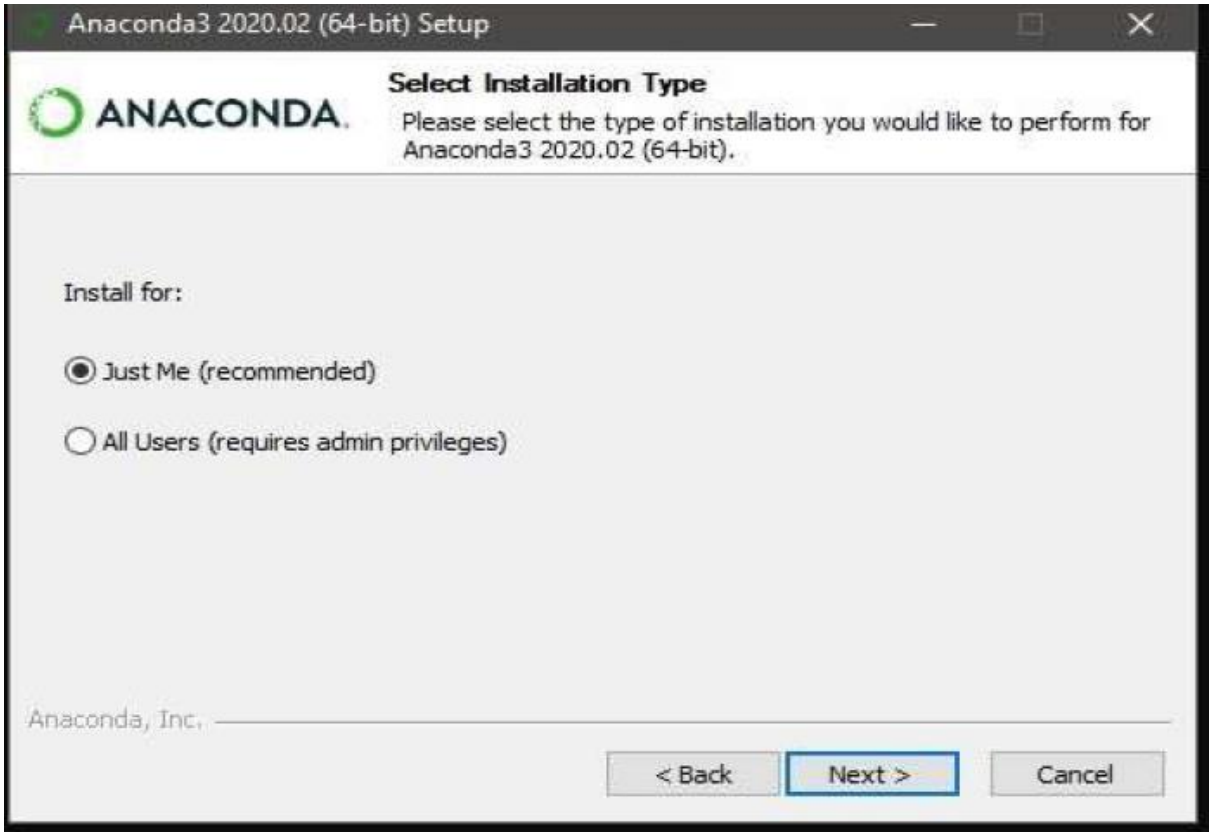


Figure 5 Select Installation type

Step 2 - Select Drive to store Anaconda installation files

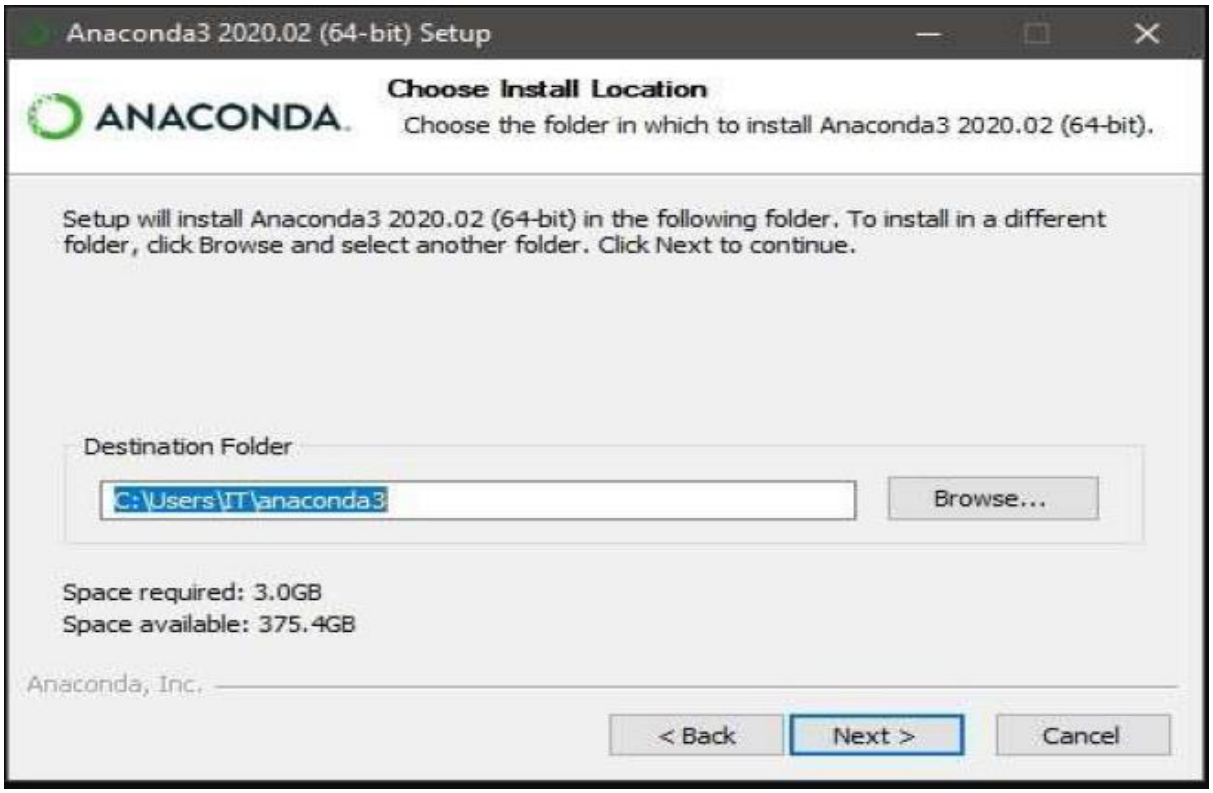


Figure 6 - Store Anaconda installation files

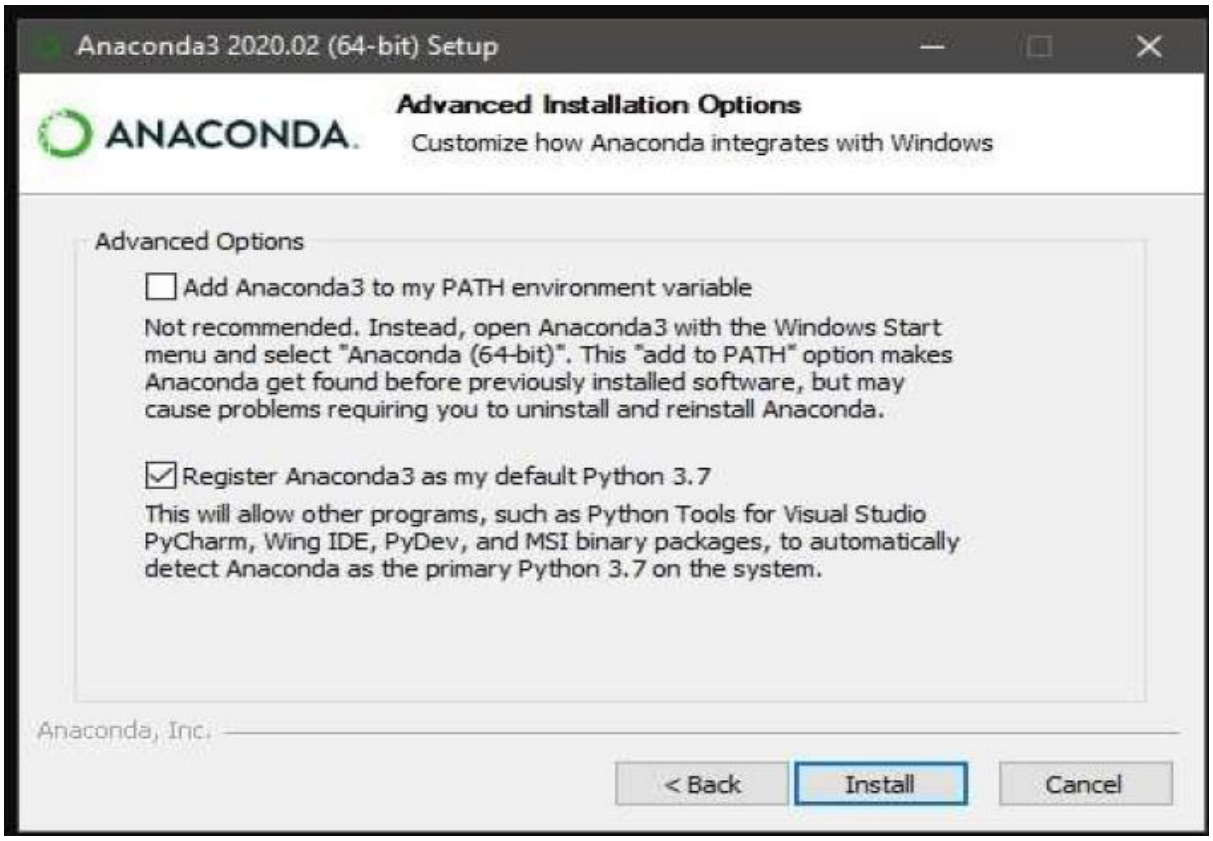


Figure 7 Click Install to complete installation

Figure 8 - Click Finish to complete Setup

1. Step 2 – Launching the Jupyter Notebook

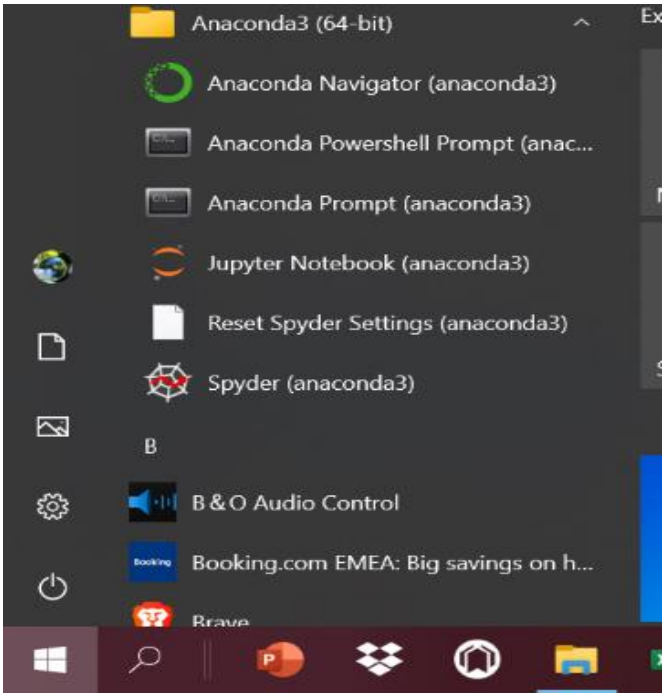


Figure 9 Jupyter Note Book Launcher

2. Python Programming language

It is one of the fastest growing programming languages. Great functionality to deal with mathematics, statistics, and data science applications. Easy to use, easy to debug language that also caters to people with non-programming backgrounds

There are various Python libraries assist programmers and data scientists to deal with data and solve not only business problems but any other domain dealing with data based problems , that require some kind of statistical analysis.

The Pandas library is used to process data. It provides the necessary tools for data cleansing and analysis. It is used in Python to create the data frames for storing and processing the data values.

Table 2 Pandas Library

Operation	Pandas Function
Load or import the data from different sources/formats	read_csv(), read_excel(),
Information about the data - dimension, column dtypes, non-null values and memory usage	info()
Display Basic statistical details of numeric data - quartiles, min, max, mean, std	describe()
Used to detect missing values of an array-like object	isnull()

- -The Seaborn library is a data visualisation library that provides visualisation for statistical models and informative plots. It is integrated with Matplotlib library.
- -The Matplotlib library is a comprehensive library for producing static, animated and interactive plots.
- -The Pyplot package in Matplotlib library is used for plotting 2D graphs.
- -The NumPy library is used for working with Arrays. NumPy arrays are multidimensional. -They are faster and consume less memory than Python lists.

Hardware Requirement

- OS NAME-Microsoft Windows 10 Home Single Language
- SYSTEM MODEL-HP Laptop 15s-fq2xxx
- Proessor-11th Gen Intel® Core™i5-1135G7@2.40GHz,2419 Mh

Chapter 5

Exploratory Data Analysis & Data Visualization

What is Exploratory Data Analysis (EDA)?

The EDA process involves the use of visualization techniques and statistical methods. EDA is crucial step to understand various aspects of the dataset that we have selected for analysis. We can summarize all the important information regarding a particular dataset in this way.

The main purpose to employ EDA is to help the analyst complete the first step of the data analysis. It is very important to acquire a good understanding of the data. Starting with basic information such as the number of entries in the dataset, to the number of variables, mean, mode and median value of each numerical variables in the dataset

Another important reason to employ EDA tools is help the analyst spot any missing values or extreme values (outliers) in the given dataset. After detecting such values, the best strategy to handle *unclean data* can be employed. In this HousePricePrediction project Python tools will be employed to perform EDA.

It is the visual representation of data and its helps to observe, communicate patterns and trends with naked eye. Data visualization helps to communicate information in a manner that is universal, fast, and effective.

Communicating insights to non-technical decision makers is one of the most critical phases in a data science project

Python Libraries for Visualization

- Matplotlib is one of the most popular libraries for data visualizations. It provides high-quality graphics and a variety of plots such as histograms, bar charts, pie charts, etc.
- Seaborn is complementary to Matplotlib and it specifically targets statistical data visualizations.

Univariate Analysis involves the study of how

- continuous variables say X, is distributed within a dataset. An example would be distribution of male or female employees in a dataset. A Histogram is drawn to visualise this analysis.
- the count of the value of a categorical variable X in each category of the dataset. An example would be Distribution of Bachelor, Masters and Research Degree students in a university. A Count plot is used to visualise this analysis.

Bivariate Analysis involves the study of how

- continuous variable X with another Continuous variable Y and how they are correlated.

An example would be Distribution of rainfall with temperature of a place. A scatterplot is used to visualise this analysis. A Scatterplot is drawn to visualise this analysis.

- continuous variable Y changes over time. An example would be Sale of umbrellas during the year. A Line Plot is used to visualise this analysis.
- continuous variable X changes with a categorical value Y. An example would be how does the number of tourists in a place vary with months of the year. A Box Plot or Swarm plot is used to visualise this analysis
- categorical variable X changes with categorical variable Y. An example would be how many diabetes patients are across various age groups. A Stacked Bar plot is used to visualise this analysis.

A saying around matplotlib and seaborn is, "matplotlib tries to make easy things easy and hard things possible, seaborn tries to make a well-defined set of hard things easy too."

Some important functions are displot(), boxplot(), stripplot(), pairplot()

❖ Scatter Plot - A scatter plot uses dots to represent values for two different numeric variables. The position of each dot on the horizontal and vertical axis indicates values for an individual data point. Scatter plots are used to observe relationships between continuous variables.

❖ Bar Plot - A bar chart is a chart that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent. The bars can be plotted vertically or horizontally.

- Stacked Bar plot - Stacked Bar plots are used to show how a larger category is divided into smaller categories and what relationship each category of one variable has with each category of another variable.

❖ Line Plot

A line graph is a graphical display of information that changes continuously over time.

❖ Histogram and skewness in data

A histogram is a graphical display of data using bars of different heights. In a histogram, each bar groups numbers into ranges. Skewness refers to distortion or asymmetry in a symmetrical bell curve in a set of data. If the curve is shifted to the left, it is called left skewed else it is called right skewed.

❖ Count Plot

Count plot shows the count of observations in each category of a categorical variableusing

bars. A count plot can be thought of as a histogram across a categorical, instead of continuous, variable.

❖ Box Plot

A box plot is a type of chart often used in exploratory data analysis to visualize the distribution of numerical data and get an idea about the skewness and outliers in the data by displaying the items included in the five-point summary. The five point summary includes:

-Q1 (the first quartile, or the 25% mark)

-The median (the second quartile, or the 50% marks)

-Q3 (the third quartile, or the 75% mark)

- Swarm Plot

  Swarm is like a categorical scatterplot with non-overlapping points. The data points are adjusted so that they don't overlap. This gives a better representation of the distribution and spread of values.

  Quantitative analysis: Describes and summarizes data numerically

  Visual analysis: Illustrates data with charts, plots, graphs etc.

  Key Libraries for Data Manipulation - NumPy & Pandas

- Numpy - Numerical Python, Fundamental package for scientific computing. A powerful N-dimensional array object – ndarray. Useful in linear algebra, vector calculus, and random number capabilities, etc. If you use the Anaconda distribution, you will automatically be able to use the common libraries, NumPy being one of them.
- Pandas - Extremely useful for data manipulation and exploratory analysis. Built on top of NumPy. Offers two major data structures - Series & DataFrame.A DataFrame is made up of several Series - Each column of a DataFrame is a Series.In a DataFrame, each column can have its own data type unlike NumPy array which creates all entries with the same data type.

  Plots for Data Visualisation

- Distribution Plot

  A distribution plot is a method for visualizing the distribution of observations in data. Relative to a histogram, a distribution plot can produce a graph that is less cluttered and more interpretable, especially when drawing multiple distributions

- Pair Plot

  It is used to visualize relationship across multiple combination of variables in a dataset.

  It gives a square matrix of plots where each numeric variable in data will by shared across the y-axes across a single row and the x-axes across a single column.

  The diagonal plots are univariate distribution plot.

  The plot in the figure shows the pairwise relation between all three variables of the mpg data from Searborn - horsepower, weight, and acceleration.

- Heatmap

  It is used to visualize the spread of values as a rectangular table using color-encoding to highlight very low and very high values.

Chapter 6

Machine Learning

The Machine Learning (ML) programming paradigm is a different from the traditional programming paradigm as it aids programmers and analysts to build models that support decision-making for new data input points by using pre-existing knowledge or data. Machine Learning frees the programmer from the task of making “complex flowcharts or hand-coded rules” to make predictive models. [6]

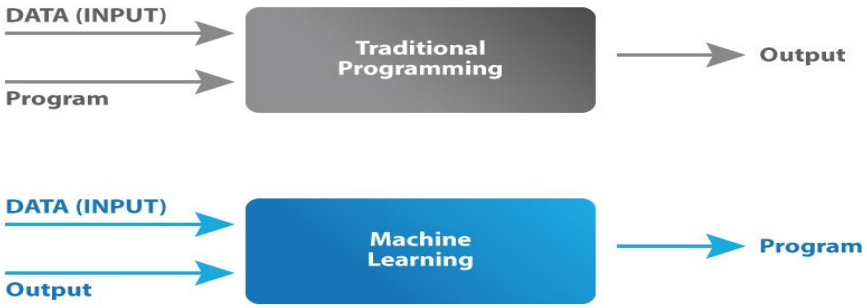


Figure 10 Traditional Programming vs. Machine Learning Paradigms [7]

Reasons for using Machine Learning paradigm

The ML paradigm allows for creating reusable code to make faster and better decisions. This Paradigm requires that enough data (historical) is available to based our predictions. The paradigm has helped to provide solutions across multiple domains where in large volumes of data are available. Machine Learning algorithms are the main drivers of this model, these lagorithms and newer versions are used to perform selected tasks and these algorithms learn from historical data to make accurate predictions.

The ML algorithms are applied on tasks that require classification for example , classifymng emails as spam or not spam. Regression involves tasks where a numerical value is to be predicted from an already available set of variables or regressors. Grouping or clustering which involves grouping similar items together from a large group of dissimilar items.

Machine Learning Project Life Cycle

- Step 1-1. Identify the problem

- Identify type of problem: predictive analytics, prescriptive analytics.
- Identify key people within your organization and outside Get specifications, requirements, priorities, budgets How accurate the solution needs to be?

- Do we need all the data?
- Built internally versus using a vendor solution Vendor comparison, bench.
- Identify stakeholder

- Step 2 -Identify available data sources

  - Extract (or free data set) and check sample data (use sound sampling techniques)

  -Perform EDA (exploratory analysis)

  - Assess quality of data, and value available in data

  - Select tool (R, Excel, Tableau, Python)

- Step 3 Statistical Analyses

  -Use imputation methods as needed

  - Detect / remove outliers Selecting variables (variables reduction)

  -Correlation analysis

  -Model selection (as needed, favor simple models)

  -Sensitivity analysis Cross-validation, model fitting Measure accuracy, provide confidence intervals

- Step 4 . Implementation, development and testing

  -Develop a simple , fast, robust and reusable and scalable application.

Machine Learning Algorithms

In the project I have write code to implement house prediction prices. Decision Trees algorithms are ML models that assist in decision making. The Regression algorithm may not give accurate results hence we have to apply Decision Tree.

CHAPTER 7

Implementation

Objective of the project – To use Machine Learning (ML) algorithms to predict the price of housing units from a given dataset.

The dataset chosen is from a Kaggle dataset (Include REF). The following is the Attributes Description:

Column Description

POSTED_BY Category marking who has listed the property

UNDER_CONSTRUCTION Under Construction or Not

RERA Rera approved or Not

BHK_NO Number of Rooms

BHKORRK Type of property

SQUARE_FT Total area of the house in square feet

READYTOMOVE Category marking Ready to move or Not

RESALE Category marking Resale or not

ADDRESS Address of the property

LONGITUDE Longitude of the property

LATITUDE Latitude of the property

ACKNOWLEDGMENT: The dataset for this hackathon was contributed by Devrup Banerjee

Linear regression is a way to identify a relationship between the independent variable(s) and the dependent variable

2. We can use these relationships to predict values for one variable for given value(s) of other variable(s)

3. It assumes the relationship between variables can be modeled through linear equation or an equation of line.

4. The variable, which is used in prediction is termed as independent/explanatory/regressor where the predicted variable is termed as dependent/target/response variable.

5. In case of linear regression with a single explanatory variable, the linear combination can be expressed as : $\hat{Y} = \beta 0 + \beta 1 X$. The terms β 0& β1 are coefficients.

Best fit line in the linear regression model

• Learning from the data, the model generates a line that fits the data.
• Our aim is to find a regression line that best fits the data

• Mathematically, the line that minimizes the sum of squared error of residuals is called Regression Line or the Best Fit Line.

We can use the scatter plot to understand the correlation between the independent variables
● In the example here, you can see a scatter plot between the tip amount and the total_bill amount
● We can see that there is positive correlation between these two - as the bill amount increases, the tip increases
● The line in blue that you see is the 'best fit'line - those in red are some examples of all other lines that are not the 'best fit'

Deterministic vs Statistical Regression
● Deterministic (or functional) relationships are exact. For example: Fahrenheit = 9/5 * Celsius + 32. Given X, Y is exactly known. See X-Y graph below (left).
● Statistical relationships between Y and X are probabilistic (not exact).
● For example, monthly sales of umbrellas in a company is directly proportional to the average rainfall in the month. See X-Y graph below (right).

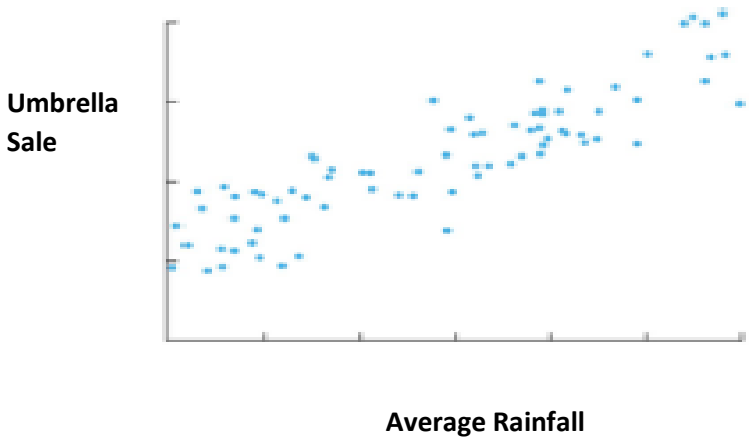


Figure 11 Statistical relationship

Run python Notebook

Dataset Selection

- Selecting Data that is suitable for your analysis project is a very important step of the Data Engg process. The dataset is IndianCitiesHousePrice.
- There are over 400 cities and towns in the dataset.
- So to understand how to predict the prices of houses, we can reduce the dataset to focus on entries from 4 major cities in India.

Cleaning the Data

Using Index values we can clean the dataset of the unrequired data

```
index_names = data[(data['CITY'] == 'Agra')].index

# drop these given row
# indexes from dataFrame
data.drop(index_names, inplace = True)

print((data['CITY']).unique)
```

Steps to clean or trim the Data

1. From the original data set having data from over 400 cities, we have selected 4 cities - Bombay, Chennai, Mumbai, Kolkata.
2. The dataset has 3,82,865 entries and 12 columns.
3. From over three different types of Posts (Owner, Builder, Dealer) we have selected only those adverts, posted by Owner only.
4. The Address column was split into two columns, address and city, to arrange the entries by city.

REFERENCES

1. N. N. Ghosalkar and S. N. Dhage, "Real Estate Value Prediction Using Linear Regression," *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 2018, pp. 1-5, doi: 10.1109/ICCUBEA.2018.8697639.
2. D. Banerjee and S. Dutta, "Predicting the housing price direction using machine learning techniques," *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, 2017, pp. 2998-3000, doi: 10.1109/ICPCSI.2017.8392275.
3. Renovation Prediction Analysis Using Different Machine Learning Algorithms. In: Dash S.S., Panigrahi B.K., Das S. (eds) Sixth International Conference on Intelligent Computing and Applications. Advances in Intelligent Systems and Computing, vol 1369. Springer, Singapore. https://doi.org/10.1007/978-981-16-1335-7_41
4. Alfiyatin, Adyan Nur, et al. "Modeling house price prediction using regression analysis and particle swarm optimization." *International Journal of Advanced Computer Science and Applications* 8.10 (2017): 323-326.
5. Baldominos, Alejandro, et al. "Identifying real estate opportunities using machine learning." *Applied sciences* 8.11 (2018): 2321.
6. Sarkar, Dipanjan, Raghav Bali, and Tushar Sharma. "Practical machine learning with Python." *A Problem-Solvers Guide To Building Real-World Intelligent Systems. Berkely: Apress* (2018).
7. https://devopedia.org/images/article/298/3800.1609407548.jpg

APPENDIX

House Unit Price Prediction using Machine Learning Algorithms - Linear Regression & Decision Tree

1.Data Description:

Train.csv - 29451 rows x 12 columns

Test.csv - 68720 rows x 11 columns

Sample Submission - Acceptable submission format. (.csv/.xlsx file with 68720 rows)

2.Attributes Description:

Column Description

POSTED_BY Category marking who has listed the property

UNDER_CONSTRUCTION Under Construction or Not

RERA Rera approved or Not

BHK_NO Number of Rooms

BHKORRK Type of property

SQUARE_FT Total area of the house in square feet

READYTOMOVE Category marking Ready to move or Not

RESALE Category marking Resale or not

ADDRESS Address of the property

LONGITUDE Longitude of the property

LATITUDE Latitude of the property

ACKNOWLEDGMENT: The dataset for this hackathon was contributed by Devrup Banerjee . We would like to appreciate his efforts for this contribution to the Machinehack community.

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns

# import libraries for building linear regression model
# using statsmodel
from statsmodels.formula.api import ols
import statsmodels.api as sm
from sklearn.linear_model import LinearRegression
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import mean_squared_error, r2_score
#ignore warnings
import warnings
warnings.filterwarnings("ignore")
```

The **Pandas library**is used to process data. It provides the necesary tools for data cleansing and analysis. It is used in Python to create the dataframes.

The **Seaborn library** is a data visualisation library that provides visualisation for statistical models and informative plots. It is integrated with Matplotlib libraty.

The **Matplotlib library** is a comprehensive library for producing static, animated and interactive plots.

The **Pyplot package** in Matplotlib library is used for plotting 2D graphs.

The **NumPy library** is used for working with Arrays. NumPy arrays are multidimensional. They are faster and consume less memory than Python lists.

3. Import the training set

In [2]:
```
data=pd.read_csv('train1.csv')
data1=pd.read_csv('test1.csv')
# data=pd.read_csv('train_4citydata.csv')
#data1=pd.read_csv('test1.csv')
```

4. View the data

In [3]:
```
data.head()
```

Out[3]:

	POSTED_BY	UNDER_CONSTRUCTION	RERA	BHK_NO.	BHK_OR_RK	SQUARE_FT	READY_TO_MOVE
0	Owner	0	0	2	BHK	1300.236407	1
1	Dealer	0	0	2	BHK	1275.000000	1
2	Owner	0	0	2	BHK	933.159722	1
3	Owner	0	1	2	BHK	929.921143	1
4	Dealer	1	0	2	BHK	999.009247	0

5. Data Description

Observations The data consists of the above columns that consist of info of housing units and their prices. These include -are these info posted by Owner,Dealer or Builder -expected price of the unit -RERA approved or not -Resale property -Address - City and location -BHK or RK (Room & Kitchen) Unit -Under Construction or Ready to Move

6. Number of cities in the dataset

In [4]:
```
P=len(data['CITY'].unique())
print("No. of cities in this dataset = ",P)
print()
print("They are ",data['CITY'].unique())
```

```
No. of cities in this dataset =  384

They are  ['Bangalore' 'Mysore' 'Ghaziabad' 'Kolkata' 'Kochi' 'Jaipur' 'Mohali'
 'Chennai' 'Siliguri' 'Noida' 'Raigad' 'Bhubaneswar' 'Wardha' 'Pune'
 'Mumbai' 'Nagpur' 'Deoghar' 'Bhiwadi' 'Faridabad' 'Lalitpur'
 'Maharashtra' 'Vadodara' 'Visakhapatnam' 'Vapi' 'Mangalore' 'Aurangabad'
 'Ottapalam' 'Vijayawada' 'Belgaum' 'Bhopal' 'Lucknow' 'Kanpur'
 'Gandhinagar' 'Pondicherry' 'Agra' 'Ranchi' 'Gurgaon' 'Udupi' 'Indore'
 'Jodhpur' 'Coimbatore' 'Valsad' 'Palghar' 'Surat' 'Varanasi' 'Guwahati'
 'Amravati' 'Anand' 'Tirupati' 'Secunderabad' 'Raipur' 'Vizianagaram'
 'Thrissur' 'Satna' 'Madurai' 'Chandigarh' 'Shimla' 'Gwalior' 'Rajkot'
 'Sonipat' 'Allahabad' 'Berhampur' 'C V Raman Nagar' ' Mahalaxmi 2'
 'Roorkee' 'Dharuhera' 'Latur' 'Durgapur' 'Panchkula' 'Solapur' 'Durg'
 'Jamshedpur' 'Hazaribagh' 'Jabalpur' 'Hosur' 'Morbi' 'Hubli' 'Karnal'
 'Patna' 'Bilaspur' 'Ratnagiri' 'Meerut' 'Kotdwara' 'Jalandhar' 'Amritsar'
 'Patiala' 'Ludhiana' 'Alwar' 'Kota' 'Panaji' ' Thiruvanmiyur' 'Kolhapur'
 'Ernakulam' 'Bhavnagar' 'Bharuch' 'Asansol' 'Jhansi' 'Margao' 'Anantapur'
 'Eluru' 'Bhilai' 'Dehradun' 'Guntur' 'Jalgaon' 'Udaipur' 'Gurdaspur'
 'Neemrana' 'Hassan' 'Sindhudurg' ' Manewada' 'Hoshangabad' 'Kottayam'
 'ranchi' 'Dhanbad' ' Marcela' 'Navsari' 'Bahadurgarh' 'Nellore'
 'next to tata chrome' 'Dhule' 'Tirunelveli' 'Cuttack' 'Haridwar'
 'Nainital' ' Barowaritala' 'Jamnagar' ' Kalapatti' 'Kanchipuram' 'Kadi'
 'Karad' 'Jagdalpur' ' Saidapet' 'Panipat' 'Muzaffarpur'
 ' Sirsi Bhankrota Road' 'Salem' 'Jhunjhunu' 'Gandhidham' 'Junagadh'
 'Moradabad' 'Pallikranai' 'Ahmednagar' ' barwaritola' 'Jalna' 'Bhiwani'
 'Palakkad' 'Kannur' 'Vakola' 'Karjat' 'Akola' 'Jind' ' GANDHINAGAR'
 ' Bellandur' 'Gaya' 'Ambala' 'Ajmer' ' kochi' 'ZAGADE WASTI' 'Hajipur'
 'Dharwad' 'Pudukkottai' 'Kollam' 'Ooty' 'Bhandara' ' Police Lane'
 'Barabanki' ' Pratap Nagar' ' Periyar Nagar' 'Rajpura' 'Palwal' 'Aligarh'
 'Avadi' 'Erode' ' nanganallur' 'Rudrapur' 'Tenali' 'Ongole' 'Nizamabad'
 'Puri' 'Dalhousie' 'Siddipet' 'Solan' 'Darbhanga' ' Opp- JAINAM VIHAR'
 ' Achara' 'Kadapa' 'Kakinada' ' Ponda' ' Manapakkam' 'Agartala'
 ' Vadodara' ' Official Residence of Chief Minister of Himachal Pradesh'
 ' NELLORE' 'Warangal' 'Osmanabad' 'Bhagalpur' ' Old Pardi Naka'
 'Bardhaman' 'Rishikesh' 'Chandrapur' 'Bokaro' ' chembur west'
 ' Gomti Nagar' 'Jharsuguda' 'Bhimavaram' 'Kurnool' 'Amroha' 'Hapur'
 'Sabarkantha' 'mylapore' 'Harda' 'Haldwani' 'kundrathur' 'Ujjain'
 'Thoothukudi' ' Nagaram' 'Thiruvanmiyur' 'Karaikudi' ' Tata Hospital'
 'Mathura' ' Kuthumbakam' 'yamuna Complex' ' karad' ' Tibri Road' 'Rewari'
 'Godhra' 'Kharagpur' 'Srinagar' 'Midnapore' ' Asansol' ' Tardeo'
 'Rayagada' 'Banswara' 'Shirdi' ' Eagle Ridge' ' Samarpan Square' 'karjat'
 'Rohtak' 'Pali' 'Hathras' ' Sikandara' ' Yash plaza' ' Vinayagapuram'
 ' Motera' 'Balasore' ' Poonamallee' ' ELECTRICITY COLONY' 'Chhindwara'
 'Jivarajpark' 'Bareilly' 'opp To Nagarjuna Sch' 'Patan Road'
 'nandi Garden' 'Vidisha' 'umiyanagar2' 'Thanjavur' 'Kangra' 'Bikaner'
 ' Waghapur' 'Rewa' ' Neral' ' Ghaziabad' 'Porbandar' 'Nagaur' 'Nanded'
 'Rourkela' 'Nadiad' ' Uttarahalli' ' S.S.Colony' ' Sarjapura Road'
 ' Bangalore' 'Gulbarga' ' Ramgarh' ' BHAVNAGAR' ' E.M Bypass' 'Palanpur'
 'Bhadrak' 'Kurukshetra' 'boda bag' ' Athwalines' ' Manikantan Nagar'
 'Dibrugarh' 'Sagar' 'Machilipatnam' 'Pathanamthitta' ' besa'
 'LATMA ROAD NO - 9' 'Bankura' 'Jammu' 'Idukki' 'vasant Vihar'
 ' Portryne Mall Road' ' Dombivali East' 'Ambala City' 'Raigarh'
 ' Chunabhati' ' TC Playa Main Road' 'Arrah' 'Nagaon' 'Talaja Road'
 'Karwar' 'Dahod' ' Opposite GRP Ground' 'Nagapattinam' ' Ambattur'
 'Sikar' ' Aligarh' ' Vaishali Nagar West' ' gomti nagar' ' Narayan Pur'
 'Angul' 'Srikakulam' ' Sreebhumi' 'Baddi' ' Jugsalai' ' Madipakkam'
 ' Sanganer' 'Latma road' ' KURSEONG' ' Kalyan Wes' ' chinchwad' 'Raisen'
 ' Machabollaram' ' RR Nagar' 'Hoshiarpur' ' Solan ' 'Beed' ' Amroli'
 'lekhraj metro station' ' haudin road' ' Sullad Road' ' Korattur'
 ' Lohgarh Road' 'Gadarwara' ' Siddhivinayak Nagar' ' Mova'
```

```
 ' Behala- Tollygunge' ' JP Nagar' 'deshaiphet Road ' ' Ranjit Sagar Road'
 ' Pune' ' Kota' ' Dunlop' ' greater noida west' 'Jajpur' 'Govindpuram'
 ' Varadharajapuram Village' ' Palghar 401202' ' Anand Colony'
 ' Syndicate Bank Layout' ' Nagasandra Post' 'Haldia' 'Chittoor'
 'Faizabad' 'Budha more' ' Prabgat Nagar' 'Malappuram' 'Betul'
 'Surendranagar' 'Phagwara' 'Visnagar' 'Rajnandgaon' ' pudupalayam'
 'Raichur' ' Narayanapuram' ' Kukkikatte' ' DERABASSI' 'Chankyapuri'
 ' opp chopati' 'Sambalpur' ' Mallikarjunapuram colony'
 'Opp.Mafatlal Mill' ' Gogol' 'Gondia' ' Bhekrainagar' ' Wagdara'
 ' Tembhode Rd' ' Velapadi Kosapet' 'Bharatpur' 'Bhuj' 'Pin 841407'
 ' Ramapuram' 'Washim' 'medavakkam']
```

7. Exploratory Data Analysis (EDA)

In [5]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 29224 entries, 0 to 29223
Data columns (total 13 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   POSTED_BY           29224 non-null  object 
 1   UNDER_CONSTRUCTION  29224 non-null  int64  
 2   RERA                29224 non-null  int64  
 3   BHK_NO.             29224 non-null  int64  
 4   BHK_OR_RK           29224 non-null  object 
 5   SQUARE_FT           29224 non-null  float64
 6   READY_TO_MOVE       29224 non-null  int64  
 7   RESALE              29224 non-null  int64  
 8   ADDRESS             29224 non-null  object 
 9   CITY                29224 non-null  object 
 10  LONGITUDE           29224 non-null  float64
 11  LATITUDE            29224 non-null  float64
 12  TARGET              29224 non-null  float64
dtypes: float64(4), int64(5), object(4)
memory usage: 2.9+ MB
```

Observations

-There are 29225 rows and 13 columns.
-There are 5 Integer values , 5 Float values and 4 Object values

8. Categorical and Numerical Values

In [6]:
```
cat_test=[cat for cat in data.columns if data[cat].dtype=='object']
num_test=[cat for cat in data.columns if data[cat].dtype=='int64' or data[cat].dtype=='
print(cat_test)
print(num_test)
```

```
['POSTED_BY', 'BHK_OR_RK', 'ADDRESS', 'CITY']
['UNDER_CONSTRUCTION', 'RERA', 'BHK_NO.', 'SQUARE_FT', 'READY_TO_MOVE', 'RESALE', 'LON
GITUDE', 'LATITUDE', 'TARGET']
```

9. Unique values under each Variablein the Dataset

In [7]:
```
# Unique values under each variable/column
data.nunique()
```

Out[7]:
```
POSTED_BY                3
UNDER_CONSTRUCTION       2
RERA                     2
BHK_NO.                 16
BHK_OR_RK                2
SQUARE_FT            19412
READY_TO_MOVE            2
RESALE                   2
ADDRESS               5775
CITY                   384
LONGITUDE             4013
LATITUDE              4005
TARGET                1168
dtype: int64
```

10. Are there any null values in the dataset?

In [8]:
```
data.isnull().any()
```

Out[8]:
```
POSTED_BY            False
UNDER_CONSTRUCTION   False
RERA                 False
BHK_NO.              False
BHK_OR_RK            False
SQUARE_FT            False
READY_TO_MOVE        False
RESALE               False
ADDRESS              False
CITY                 False
LONGITUDE            False
LATITUDE             False
TARGET               False
dtype: bool
```

Observation

Two column ADDRESS and CITY had null values After Data cleaning there are no null values in the dataset

11. Check how many null values are in the dataset

In [9]:
```
display(data.isnull().sum())
```

```
POSTED_BY             0
UNDER_CONSTRUCTION    0
RERA                  0
BHK_NO.               0
BHK_OR_RK             0
SQUARE_FT             0
READY_TO_MOVE         0
RESALE                0
ADDRESS               0
CITY                  0
LONGITUDE             0
LATITUDE              0
TARGET                0
dtype: int64
```

In the training set (dataset)

- ADDRESS column there are 2 null values
- CITY column there are 9 null values

12. Print the null values are in the dataset

In [10]:
```
# print the null value columns
null_columns=data.columns[data.isnull().any()]
print(data[data.isnull().any(axis=1)][null_columns])
print("===========================")
print(data[data.isnull().any(axis=1)][null_columns].size)
```

```
Empty DataFrame
Columns: []
Index: []
===========================
0
```

Observation

1. Before Data cleaning these were the **Null/NaN values in the datset**

- 2225 NaN panvel
- 5219 Bhivpuri NaN
- 8190 NaN Manoramaganj
- 10144 New garia NaN
- 11630 Umbhel Gam NaN
- 13623 Shivneri Colony NaN
- 17655 Goregoan W NaN
- 20824 Sarjapur NaN
- 21942 Adarsh Nagar NaN
- 23485 CIRCUIT HOUSE ROAD NaN
- 28886 Near Arbindo Hospital Opposite London Villa NaN

2. There are no null values in the dataset after data cleaning

13.Descriptive Statistics

In [13]:
```
data.describe().T
```

Out[13]:

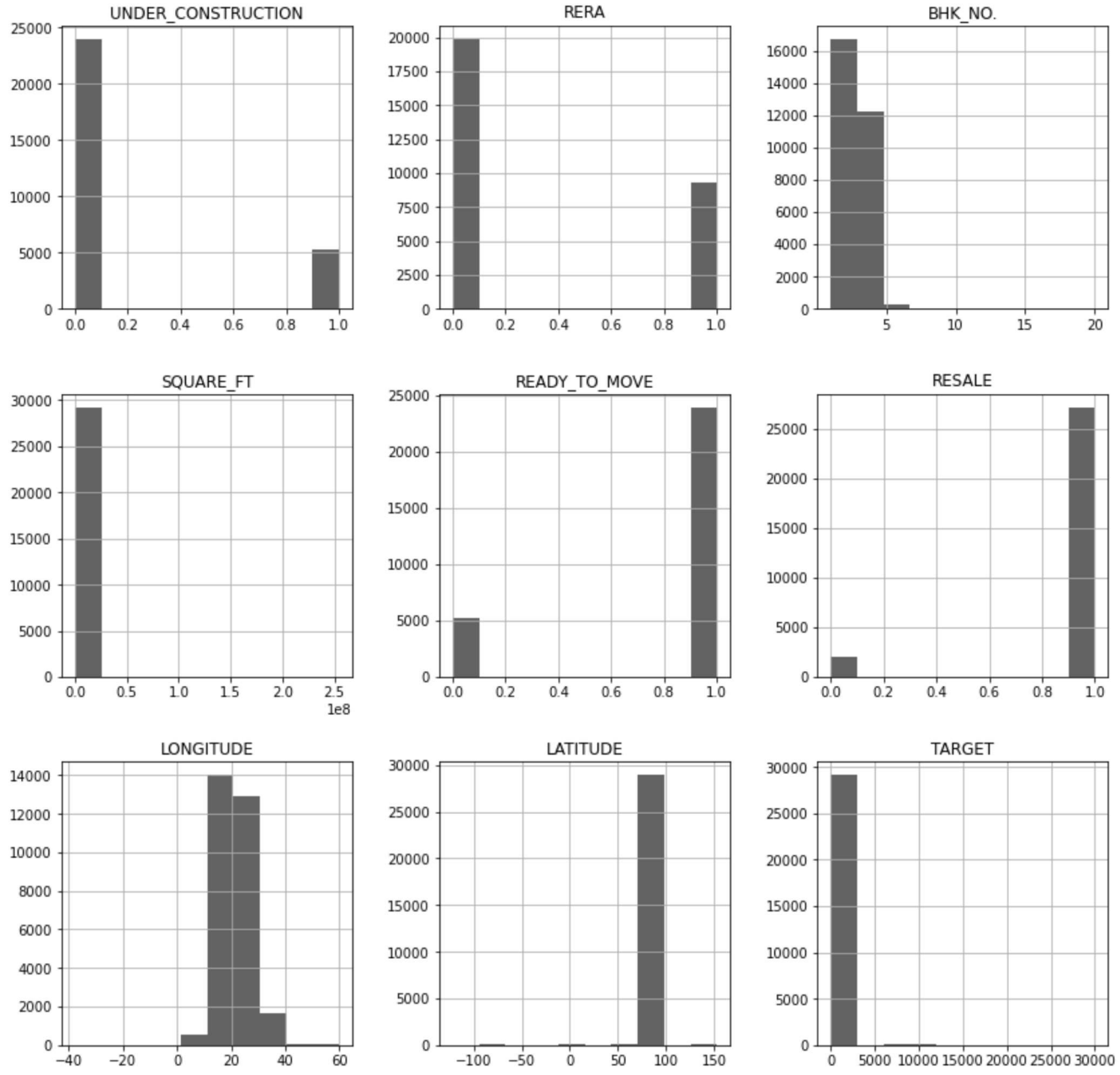
	count	mean	std	min	25%	50%	
UNDER_CONSTRUCTION	29224.0	0.180092	3.842704e-01	0.000000	0.000000	0.000000	0.0
RERA	29224.0	0.319429	4.662635e-01	0.000000	0.000000	0.000000	1.0
BHK_NO.	29224.0	2.394949	8.792454e-01	1.000000	2.000000	2.000000	3.0
SQUARE_FT	29224.0	19938.279801	1.908704e+06	3.000000	900.032144	1175.333871	1553.3
READY_TO_MOVE	29224.0	0.819908	3.842704e-01	0.000000	1.000000	1.000000	1.0
RESALE	29224.0	0.929715	2.556306e-01	0.000000	1.000000	1.000000	1.0
LONGITUDE	29224.0	21.316046	6.174623e+00	-37.713008	18.475864	20.943965	26.9
LATITUDE	29224.0	76.899742	1.038410e+01	-121.761248	73.803730	77.333948	77.9
TARGET	29224.0	142.874851	6.557751e+02	0.250000	38.000000	62.000000	100.0

Observations

-The number of units in the training dataset is 29225. -The area of the housing units range from **3 sq.ft** to **2,55,45,500 sq.ft**. -The minimum number of bedrooms is 1 and maximum is 20. -The minimum price of a housing unit is **0.25 lakhs** and maximum price is **30,000 lakhs**.

In [ ]:
```

```

In [18]:
```
#creating histograms
data.hist(figsize=(14,14))
plt.show()
```



In [20]:
```
#Printing the % sub categories of each category
for i in cat_test:
    print(data[i].value_counts(normalize=True))
    print('='*40)
```

```
Dealer     0.622947
Owner      0.356248
Builder    0.020805
Name: POSTED_BY, dtype: float64
========================================
BHK    0.999179
RK     0.000821
Name: BHK_OR_RK, dtype: float64
========================================
Zirakpur                  0.019265
Whitefield                0.007870
Raj Nagar Extension       0.007357
Thane West                0.005646
Kolshet Road              0.005201
                              ...
528                       0.000034
Hiran Nagri Sector 12     0.000034
Hoodi Circle              0.000034
Lal bangla                0.000034
Garmal                    0.000034
Name: ADDRESS, Length: 5775, dtype: float64
========================================
Bangalore              0.147789
Lalitpur               0.102211
Mumbai                 0.069121
Pune                   0.067889
Noida                  0.060396
                           ...
 Opp- JAINAM VIHAR     0.000034
 E.M Bypass            0.000034
Gadarwara              0.000034
 Bangalore             0.000034
 Mova                  0.000034
Name: CITY, Length: 384, dtype: float64
========================================
```

Observations

- The number of RESALE units are 7 percent.
- The number of RERA approved units are 68 percent.
- The number of adverts posted by Dealer is 63 percent, Owner 35 percent and Builder 2 percent.
    - The number of UNDER_Construction units are 81 percent.
    - The number of BHK units are 99 percent.

Observation

In [30]:
```
# Lets check the relationship between POSTED_BY
data.groupby(['POSTED_BY'])[num_test].mean()
```

Out[30]:

POSTED_BY	UNDER_CONSTRUCTION	RERA	BHK_NO.	SQUARE_FT	READY_TO_MOVE	RESALE	LON
Builder	0.667763	0.672697	2.250000	3893.952525	0.332237	0.008224	2
Dealer	0.230706	0.405054	2.456578	12375.971891	0.769294	0.921615	2
Owner	0.063106	0.149073	2.295649	34098.952789	0.936894	0.997695	2

- The mean price of the units posted by Builder is 234.827 lakhs Rs
- The mean price of the units posted by Dealer is 186.7 lakhs Rs
- The mean price of the units posted by Owner is 60 lakhs Rs

7.What are the number of bedrooms in these housing units?

In [31]:
```
# VIEW THE NUMBER OF BEDROOMS WHICH ARE THERE IN MAXIMUM NUMBER OF HOUSING UNIT
print("Bedrooms"," ", "Units")
print("--------","","------")
print(data['BHK_NO.'].value_counts())
```

```
Bedrooms   Units
--------  ------
2     13211
3     10497
1      3518
4      1714
5       190
6        52
7        11
8        10
20        4
10        4
15        4
9         3
12        3
17        1
11        1
13        1
Name: BHK_NO., dtype: int64
```

OBSERVATION

1. There are

- 13211 units which have 2 bedrooms.
- 10497 units which have 3 bedrooms.
- 3519 units which have 1 bedroom.
- 1714 units with 4 bedrooms.
- 190 units which have 5 bedrooms.
- 52 units with 6 bedrooms.
- 11 units with 10 bedrooms.
- 10 unit with 8 bedrooms.
- 4 units with 20,10,15 bedrooms.
- 3 units with 9 & 12 bedrooms.

- 1 unit with 17,11 amd 13 bedrooms.

2. The maximum numbers of bedrooms in any unit is 20.

8. Arrange the dataset. What is the maximum price & minimum price of any housing unit in the dataset ?

In [29]:
```
# Maximum price & Minimum price of any housing unit in the dataset

data.sort_values(by="TARGET",ascending=False)
```

Out[29]:

	POSTED_BY	UNDER_CONSTRUCTION	RERA	BHK_NO.	BHK_OR_RK	SQUARE_FT	READY_TO
11142	Dealer	0	0	3	BHK	1.875000e+08	
10649	Owner	0	0	3	BHK	2.545455e+08	
15595	Owner	0	0	2	BHK	8.064516e+07	
5908	Dealer	1	1	2	BHK	5.422570e+04	
10541	Dealer	1	1	3	BHK	8.322835e+04	
...	...	...	...	...	...	...	
8270	Owner	0	1	3	BHK	1.333333e+03	
3838	Owner	0	0	10	BHK	4.250000e+04	
20746	Dealer	1	0	2	BHK	1.500000e+03	

Observation

- The maximum price is that of a housing unit in Bangalore, RT Nagar which is 30000 Lakhs.
- The minimum price is that of a housing unit in Bhubaneswar, Bomikhal which is 0.25 Lakhs.

9. Which Indian CITY/IES have Room_Kitchen units.

In [32]:
```
print()
d=data[data['BHK_OR_RK']=="RK"]['CITY']
print('The name/s of the Indian CITY/IES which having Room_Kitchen units:','\n',d.nuniqu
#Indian_billionaires[Indian_billionaires['Age']<50]['Name']
```

```
The name/s of the Indian CITY/IES which having Room_Kitchen units:
 <bound method IndexOpsMixin.nunique of 47       Lalitpur
2340     Lalitpur
2649     Lalitpur
5568     Lalitpur
8376     Lalitpur
9073     Lalitpur
11752    Lalitpur
13555    Lalitpur
14010    Lalitpur
14281    Lalitpur
14653    Lalitpur
14986    Lalitpur
15975    Lalitpur
16423    Lalitpur
17516    Lalitpur
18673    Lalitpur
19817    Lalitpur
20941    Lalitpur
22533    Lalitpur
22540    Lalitpur
23473    Lalitpur
24071    Lalitpur
24479    Lalitpur
26092    Lalitpur
Name: CITY, dtype: object>
```

Observation

- Lalitpur is the only city/town having RK **(Room and Kitchen) only** type of housing units in this dataset.

11. Which 5 cities have the maximum number of housing units on sale?

In [27]:
```
# Which 5 cities have the maximum number of housing units on sale?
print(data.CITY.value_counts().head(),"\n")
```

```
Bangalore    4319
Lalitpur     2987
Mumbai       2020
Pune         1984
Noida        1765
Name: CITY, dtype: int64
```

12. How many Bedroom Housing Units and Room Kitchen units are there in the datset ?

In [26]:
```
# count the number of Bedroom Housing Units and Room Kitchen units only
data['BHK_OR_RK'].value_counts()
```

Out[26]:
```
BHK    29200
RK        24
Name: BHK_OR_RK, dtype: int64
```

13. Are the three measures of central tendency equal?

In [25]:
```
#1
print((data['TARGET'].mean()),'\n')
#2
print((data['TARGET'].mode()),'\n')
#3
print((data['TARGET'].median()),'\n')
```

```
142.87485149192324

0    110.0
dtype: float64

62.0
```

Answer

No the Three Ms (Mean, Median and Mode) (Measures of central tendency) are not equal. This means our dataset is not normally distributed.

14. Which city has the highest number of costliest housing units?

In [24]:
```
#Which are the costliest housing units and what are their details
print(data.sort_values(by='TARGET', ascending=False).head())
```

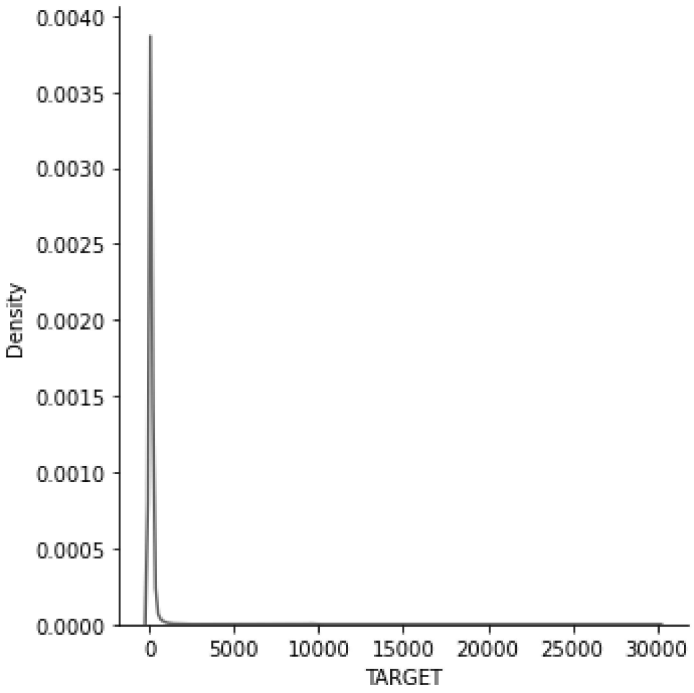
```
      POSTED_BY  UNDER_CONSTRUCTION  RERA  BHK_NO. BHK_OR_RK     SQUARE_FT  \
11142    Dealer                   0     0        3       BHK  1.875000e+08
10649     Owner                   0     0        3       BHK  2.545455e+08
15595     Owner                   0     0        2       BHK  8.064516e+07
5908     Dealer                   1     1        2       BHK  5.422570e+04
10541    Dealer                   1     1        3       BHK  8.322835e+04

       READY_TO_MOVE  RESALE      ADDRESS       CITY  LONGITUDE   LATITUDE  \
11142              1       1   R.T. Nagar  Bangalore  13.018900  77.596300
10649              1       1        Malur  Bangalore  13.021000  77.938000
15595              1       1  Lakkasandra  Bangalore  12.795926  77.331535
5908               0       0   Thane West   Lalitpur  19.180000  72.963330
10541              0       0    Chinchwad       Pune  18.627000  73.782900

        TARGET
11142  30000.0
10649  28000.0
15595  25000.0
5908    9990.0
10541   9910.0
```

Observation

The Housing units with the highest Price is in **Bangalore**, price is **30000 lakhs**, located in RT Nagar.

15. How do you visualise the price of the most number of units?

In [23]:
```
sns.displot(data['TARGET'],kind='kde')
plt.show()
print(data['TARGET'].mode)
```




```
<bound method Series.mode of 0        55.0
1        51.0
2        43.0
3        62.5
4        60.5
         ...
29219    40.0
29220    45.0
29221    27.1
29222    67.0
29223    27.8
Name: TARGET, Length: 29224, dtype: float64>
```

Observation

There are around ___number / percentage of flats with price less than _____.

16. List all the priciest housing units which are in Bangalore City. Arrange the units in alpabetical order by Address.

In [ ]:
```
# List all the cities which are pricier. for the housing units which are in  Bangalore C

bangalore_housingunits = data[data['CITY']=='Bangalore']
print(bangalore_housingunits)

print(bangalore_housingunits['TARGET'].sort_values)
print('The location of the housing units which are above 2000 lakhs:')
bangalore_housingunits[bangalore_housingunits['TARGET']>2000]['ADDRESS']
```

17.Which are the housing units which have area more than 3000 SQUARE FEET? Display them by increasing price.

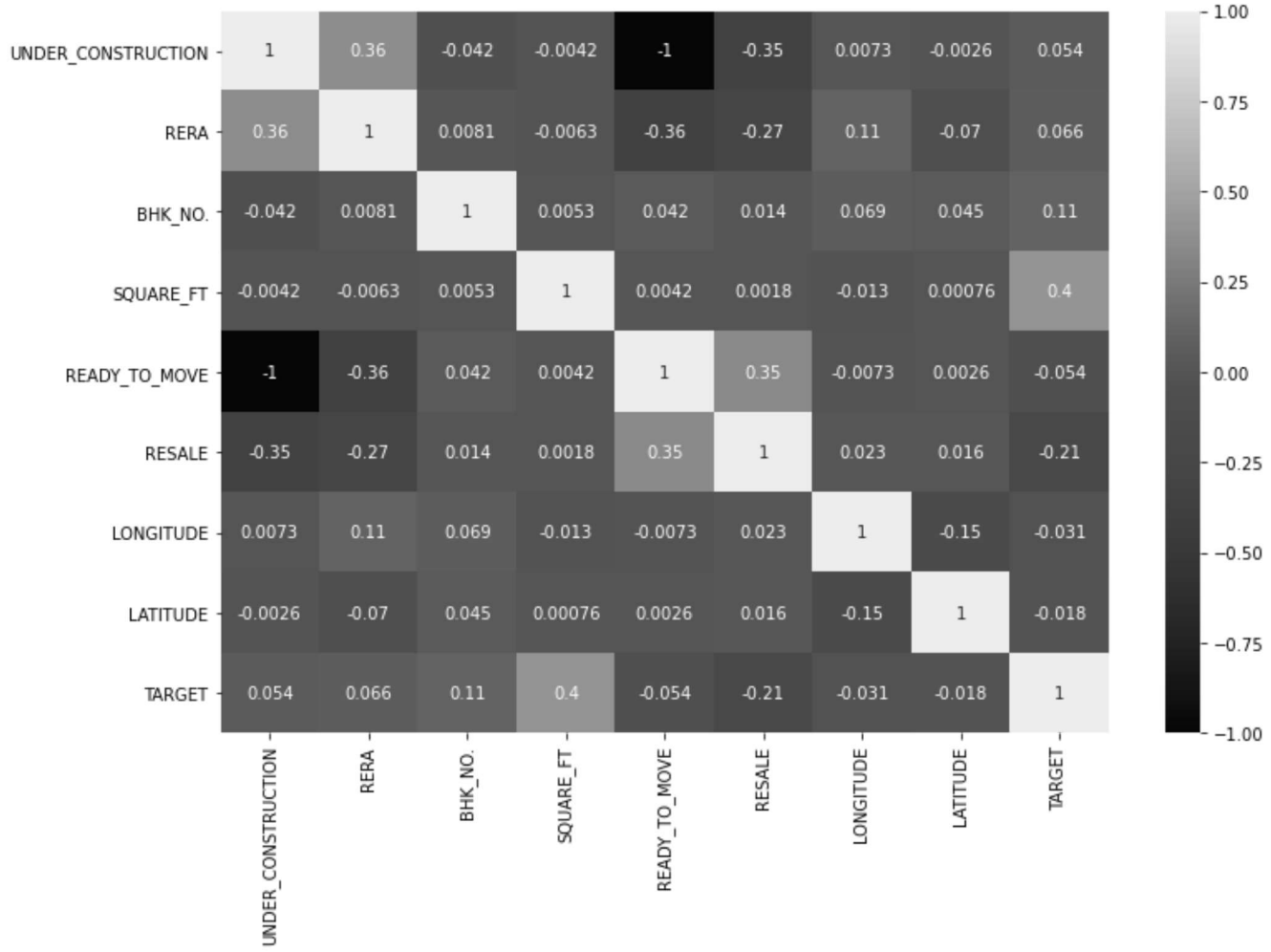
In [ ]:
```
countUnits_B=bangalore_housingunits[bangalore_housingunits['SQUARE_FT']>3000]
countUnits_B.sort_values(by="TARGET", ascending=True)
#(by="BHK_NO.", ascending=True)
```

18. Bivariate Analysis

In [ ]:
```
# Bivariate Analysis
data.corr()
```

Check the correlation using the heatmap

In [35]:
```
#Let's check the correlation using the heatmap
plt.figure(figsize=(12,8))
sns.heatmap(data.corr(), annot=True)
plt.show()
```

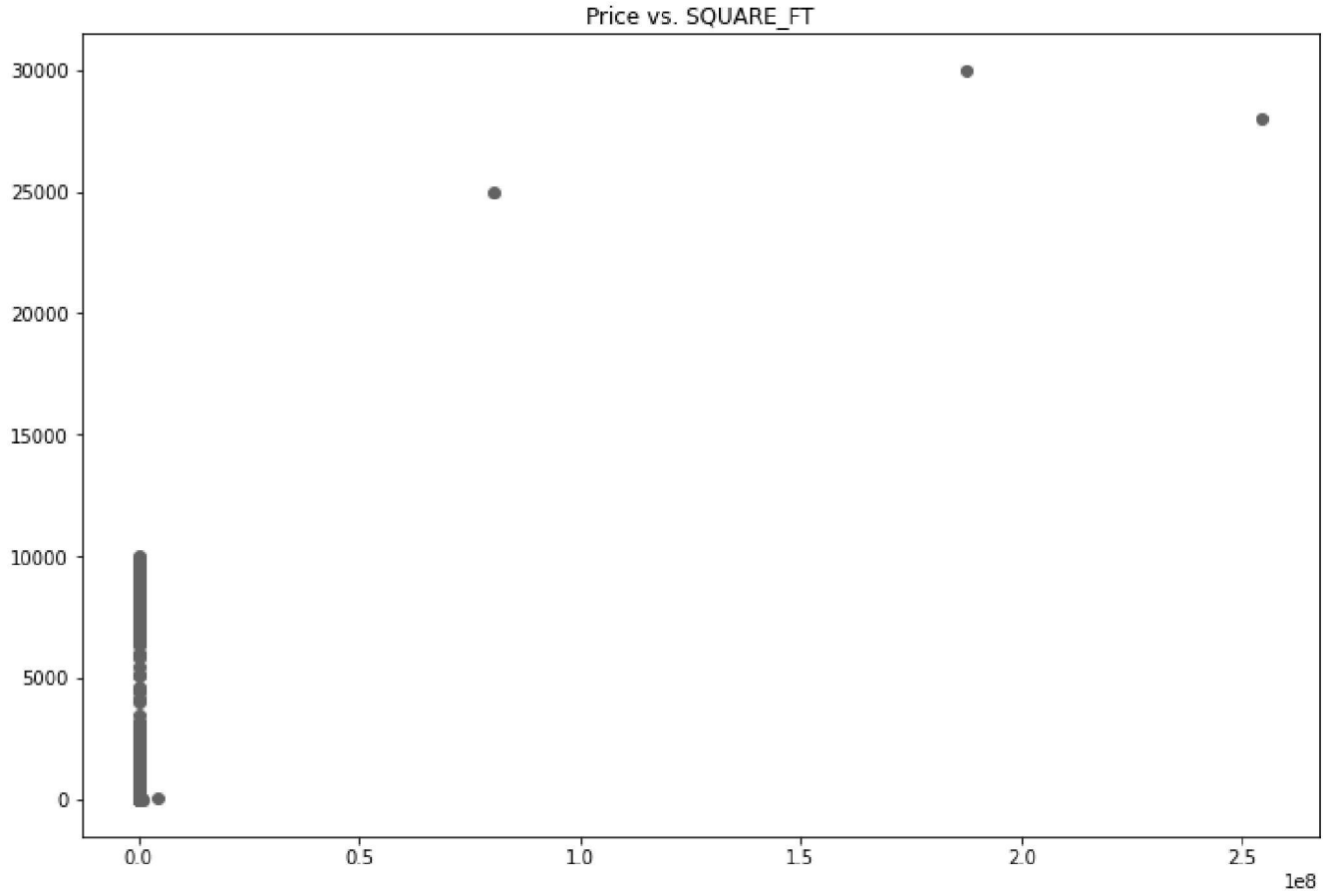


Observation

None of the variables are showing high correlation. Only the READY_TO_MOVE variable is showing high negative correlation. which indicates that the READY_TO_MOVE units price will decrease.

In [34]:
```
plt.figure(figsize=(12,8))
plt.scatter(data.SQUARE_FT, data.TARGET)
plt.title("Price vs. SQUARE_FT")
plt.show
```

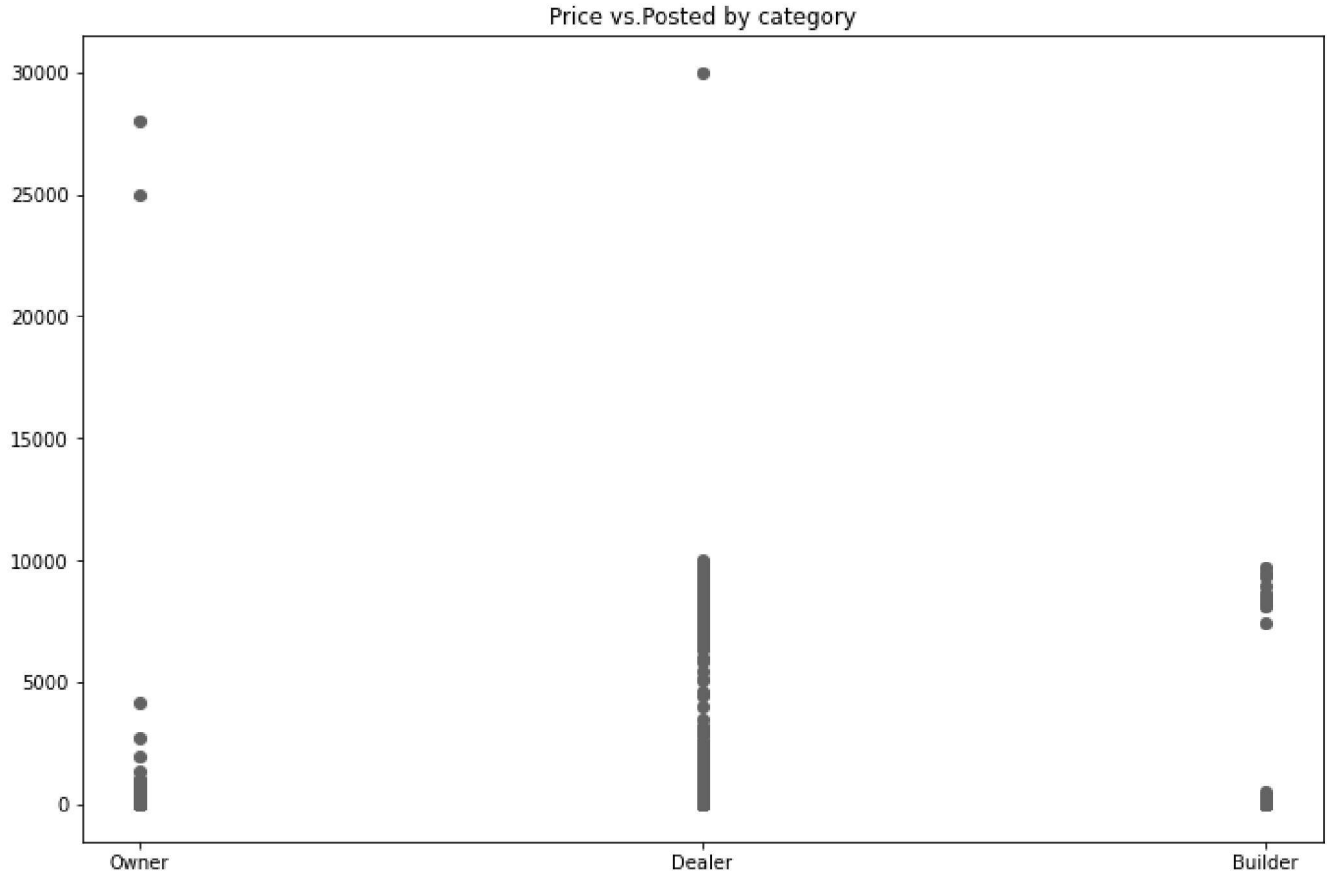
Out[34]: <function matplotlib.pyplot.show(close=None, block=None)>



In [ ]:

In [33]:
```
plt.figure(figsize=(12,8))
plt.scatter(data.POSTED_BY, data.TARGET)
plt.title("Price vs.Posted by category")
plt.show
```

Out[33]: <function matplotlib.pyplot.show(close=None, block=None)>



In [ ]:
```
plt.figure(figsize=(15,6))
ax = sns.barplot(x=data['BHK_NO.'], y=data['TARGET'])
ax.set_xticklabels(ax.get_xticklabels(), rotation=0)
ax.set_title('Bedrooms VS Price', fontsize=14)
```

Model Building - Approach

1. Build model on the train data
2. Cross-validating the model
3. Test the data on test set

In [ ]:
```
from scipy import stats
data['TARGET'] = data['TARGET'].replace([data['TARGET'][np.abs(stats.zscore(data['TARGE
```

In [ ]:
```
plt.figure(figsize=(15,10))
ax = sns.distplot(data['TARGET'], kde=True)
ax.set_title('Distplot of Price', fontsize=14)
```

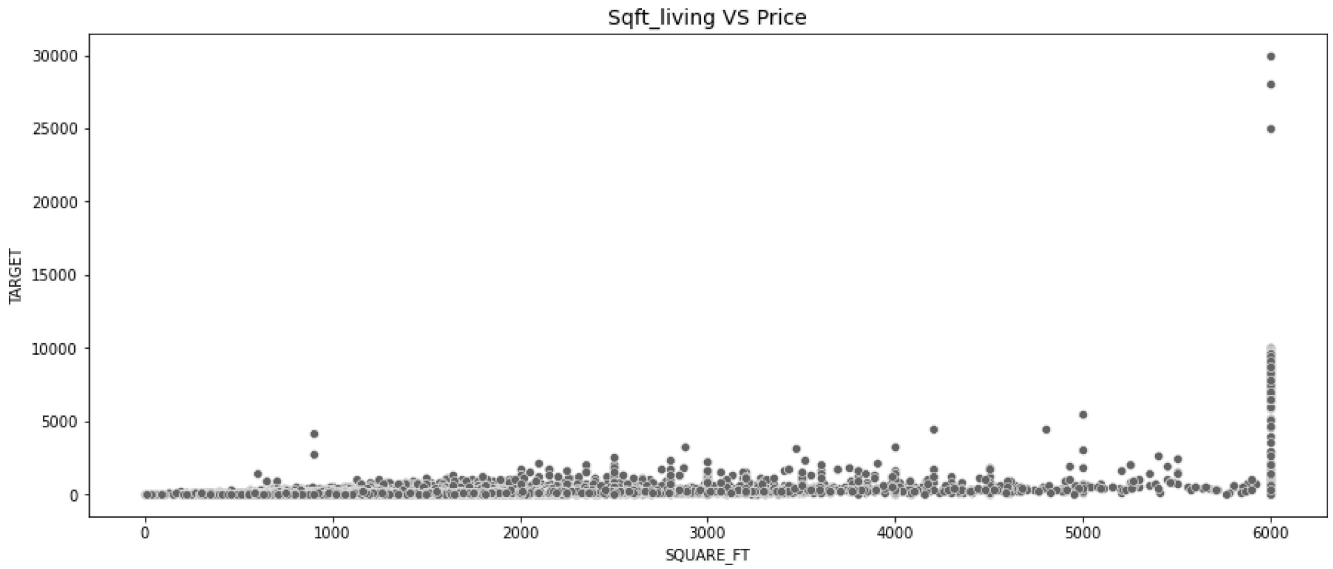
OBSERVATION

There are some outliers and majority of units are oriented towards the left. This is a left-skewed distribution.

In [45]:
```
data['SQUARE_FT'] = np.where((data.SQUARE_FT >6000 ), 6000, data.SQUARE_FT)
```

In [44]:
```
plt.figure(figsize=(15,6))
ax = sns.scatterplot(data=data, x="SQUARE_FT", y="TARGET")
ax.set_title('Sqft_living VS Price', fontsize=14)
```

Out[44]:
```
Text(0.5, 1.0, 'Sqft_living VS Price')
```



Linear Regression

Regression (Machine Learning - Supervised Learning)

X consists of the READY_TO_MOVE indicator and other controls UNDER_CONSTRUCTION, RERA,RESALE, SQUARE_FT, includes 10 regressors

In [48]:
```
from sklearn.linear_model import LinearRegression
from sklearn import metrics

Y = data['TARGET'] #target variable
X = data[['READY_TO_MOVE','UNDER_CONSTRUCTION','RERA','RESALE','SQUARE_FT']]

# define the model
model = LinearRegression()
# fit the linear regression to the regressors and target variable.
results = model.fit(X,Y) #train the model

# print the intercept value  Beta_0)
print("Intercept(Beta_0 value)",results.intercept_)
```

```
Intercept(Beta_0 value) 62.10481215419202
```

In [49]:
```
# coefficient values of the other Regressors
pd.DataFrame(results.coef_.reshape(1,-1),columns=X.columns)
```

Out[49]:

	READY_TO_MOVE	UNDER_CONSTRUCTION	RERA	RESALE	SQUARE_FT
0	-13.182518	13.182518	-3.773328	-476.609922	0.399206

Observations

- The value of READY_TO_MOVE is quite high, this indicates that READY_TO_MOVE housing units cost higher.
- The value of RERA is very high,this indicates that housing units with RERA have higher value.
- The value of UNDER_CONSTRUCTION is quite low, this indicates that housing units which are just being launched or under construction, will be cheaper.
    - There is very little effect of the area of the housing unit (in square feet) on the price.
    - If the house is a resale property, this will affect the price of the housing unit. Though in the data set the **age of the unit** is not mentioned.

In [50]:
```
data_train_values=data.values
X_train=data_train_values[:,0:3]
y_train=data_train_values[:,3]
```

In [39]:
```
data_train=data.drop((["UNDER_CONSTRUCTION", "RERA", "BHK_NO.","POSTED_BY","READY_TO_MO
data_train
```

Out[39]:

	SQUARE_FT	LONGITUDE	LATITUDE	TARGET
0	1300.236407	12.969910	77.597960	55.0
1	1275.000000	12.274538	76.644605	51.0
2	933.159722	12.778033	77.632191	43.0
3	929.921143	28.642300	77.344500	62.5
4	999.009247	22.592200	88.484911	60.5
...	...	...	...	...
29219	1062.134891	15.866670	74.500000	40.0
29220	2500.000000	27.140626	78.043277	45.0
29221	1022.641509	26.928785	75.828002	27.1
29222	927.079009	12.900150	80.227910	67.0
29223	896.774194	26.832353	75.841749	27.8

29224 rows × 4 columns

In [51]:
```
data_test=data1.drop((["UNDER_CONSTRUCTION", "RERA", "BHK_NO.","POSTED_BY","READY_TO_MO
data_test
```

Out[51]:

	SQUARE_FT	LONGITUDE	LATITUDE
0	545.171340	21.262000	73.047700
1	430.477830	22.700000	72.870000
2	500.000000	21.716412	73.004076
3	1653.225806	11.655167	92.728718
4	2549.162418	26.173068	73.261350
...	...	...	...
24914	1161.995899	28.698133	77.335746
24915	2250.409165	28.427662	77.339119
24916	740.031343	13.090000	80.270000
24917	1259.689922	22.749766	86.216352
24918	900.205761	26.860560	80.915830

24919 rows × 3 columns

In [52]:
```
data_train_values=data_train.values
X_train=data_train_values[:,0:3]
y_train=data_train_values[:,3]
X_train
```

Out[52]:
```
array([[1300.236407 ,   12.96991  ,   77.59796  ],
       [1275.       ,   12.274538 ,   76.644605 ],
       [ 933.1597222,   12.778033 ,   77.632191 ],
       ...,
       [1022.641509 ,   26.928785 ,   75.828002 ],
       [ 927.0790093,   12.90015  ,   80.22791  ],
       [ 896.7741935,   26.832353 ,   75.841749 ]])
```

In [53]:
```
X_train.shape,y_train.shape
```

Out[53]:
```
((29224, 3), (29224,))
```

In [54]:
```
y_train=y_train.reshape(-1,1)
X_train.shape,y_train.shape
```

Out[54]:
```
((29224, 3), (29224, 1))
```

In [55]:
```
data_test_values=data_test.values
X_test=data_test_values[:,0:3]
X_test.shape
X_test
```

Out[55]:
```
array([[ 545.1713396,   21.262    ,   73.0477   ],
       [ 430.4778304,   22.7      ,   72.87     ],
       [ 500.       ,   21.716412 ,   73.004076 ],
       ...,
       [ 740.0313425,   13.09     ,   80.27     ],
       [1259.689922 ,   22.749766 ,   86.216352 ],
       [ 900.2057613,   26.86056  ,   80.91583  ]])
```

In [56]:
```
lin_reg_model=LinearRegression()
lin_reg_model.fit(X_train,y_train)
```

Out[56]:
```
LinearRegression()
```

In [57]:
```
y_prediction=lin_reg_model.predict(X_train)
y_prediction
```

Out[57]:
```
array([[165.12146855],
       [168.647937  ],
       [165.61520081],
       ...,
       [124.41280385],
       [161.48798298],
       [124.67407739]])
```

In [58]:
```
rmse=np.sqrt(mean_squared_error(y_train,y_prediction))
print("Linear Regression ==>RMSE:",rmse)
print("Linear Regression ===>R_square:",r2_score(y_train,y_prediction))
```

```
Linear Regression ==>RMSE: 599.1916763368284
Linear Regression ===>R_square: 0.16509606285186174
```

In [59]:
```
data_train['Predicted_target']=y_prediction
data_train
```

Out[59]:

	SQUARE_FT	LONGITUDE	LATITUDE	TARGET	Predicted_target
0	1300.236407	12.969910	77.597960	55.0	165.121469
1	1275.000000	12.274538	76.644605	51.0	168.647937
2	933.159722	12.778033	77.632191	43.0	165.615201
3	929.921143	28.642300	77.344500	62.5	116.903920
4	999.009247	22.592200	88.484911	60.5	119.561879
...	...	...	...	...	...
29219	1062.134891	15.866670	74.500000	40.0	160.591547
29220	2500.000000	27.140626	78.043277	45.0	120.763270
29221	1022.641509	26.928785	75.828002	27.1	124.412804
29222	927.079009	12.900150	80.227910	67.0	161.487983
29223	896.774194	26.832353	75.841749	27.8	124.674077

29224 rows × 5 columns

OBSERVATION

The Predicted target price is not correctly predicted used Linear Regression as none of the variables are have correlation with the TARGET variable. The RMSE value is very large this explains the deviation of the price by such a large value.

In [60]:
```
# using Decision Tree Model we will attempt to predict the price

decision_tree_model= DecisionTreeRegressor()
decision_tree_model.fit(X_train,y_train)
y_prediction_tree=decision_tree_model.predict(X_train)

y_prediction_tree_test=decision_tree_model.predict(X_test)
y_prediction_tree_test

print("Decision Tree===>RMSE:",np.sqrt(mean_squared_error(y_train,y_prediction_tree)))
print("Decision Tree===>R_square:",r2_score(y_train,y_prediction_tree))
```

```
Decision Tree===>RMSE: 8.746811337035151
Decision Tree===>R_square: 0.9998220883104755
```

In [61]:
```
data_train['predicted_price_DT']=y_prediction_tree
data_train
```

Out[61]:

	SQUARE_FT	LONGITUDE	LATITUDE	TARGET	Predicted_target	predicted_price_DT
0	1300.236407	12.969910	77.597960	55.0	165.121469	55.0
1	1275.000000	12.274538	76.644605	51.0	168.647937	51.0
2	933.159722	12.778033	77.632191	43.0	165.615201	43.0
3	929.921143	28.642300	77.344500	62.5	116.903920	62.5
4	999.009247	22.592200	88.484911	60.5	119.561879	60.5
...	...	...	...	...	...	...
29219	1062.134891	15.866670	74.500000	40.0	160.591547	40.0
29220	2500.000000	27.140626	78.043277	45.0	120.763270	45.0
29221	1022.641509	26.928785	75.828002	27.1	124.412804	27.1
29222	927.079009	12.900150	80.227910	67.0	161.487983	67.0
29223	896.774194	26.832353	75.841749	27.8	124.674077	27.8

29224 rows × 6 columns

Observations

The RMSE value is quite low hence the predicted price is very close to the actual TARGET price. Hence we have been able to correctly predict the price using Decision Tree Algorithm which gave us very good accuracy.

Map of cities in the dataset

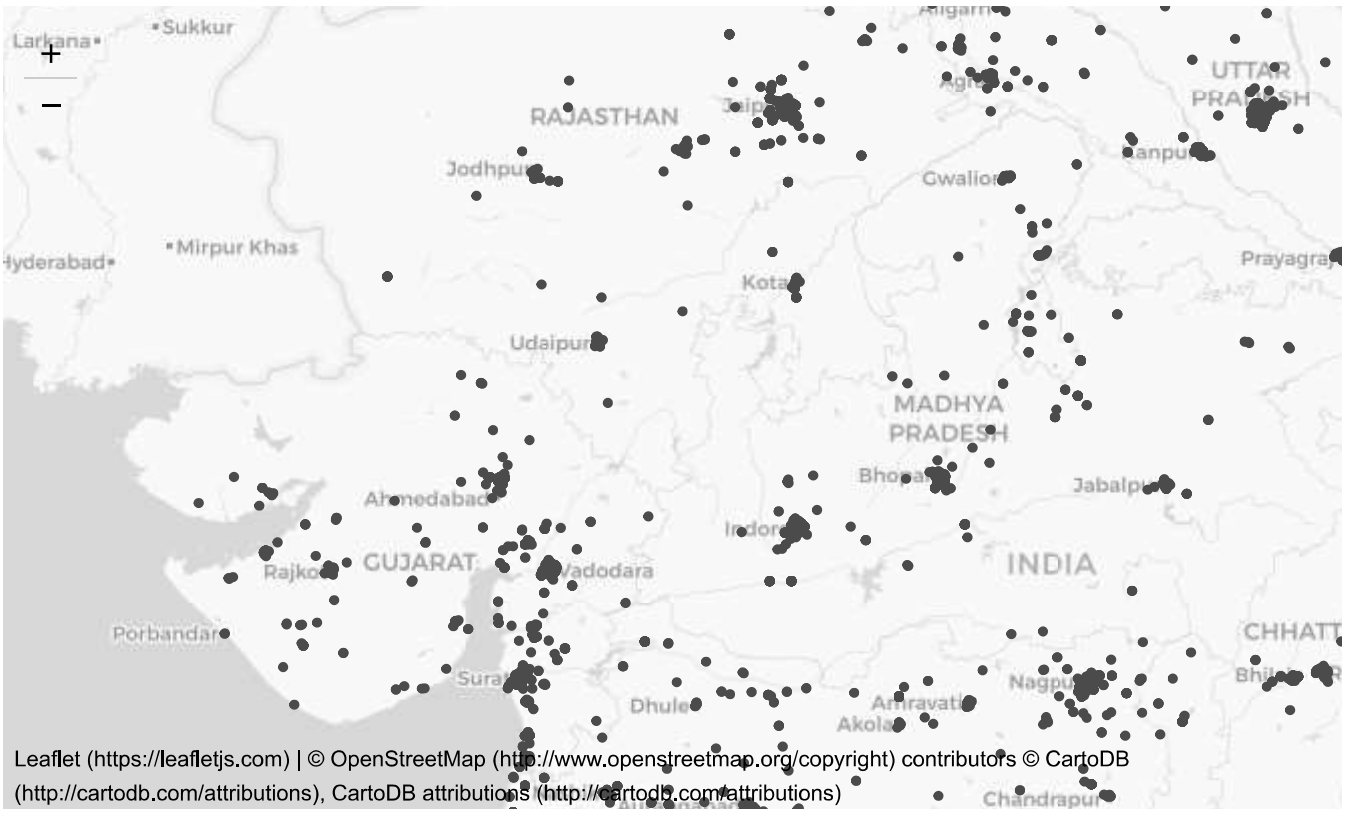
In [ ]:
```
```

In [63]:
```
# This code is attributed to kaggle community  developer
!pip install folium

import folium
from folium import Choropleth, Circle, Marker
from folium.plugins import HeatMap, MarkerCluster
```

```
Requirement already satisfied: folium in c:\users\hp\anaconda3\lib\site-packages (0.1
2.1.post1)
Requirement already satisfied: numpy in c:\users\hp\anaconda3\lib\site-packages (from
folium) (1.20.1)
Requirement already satisfied: requests in c:\users\hp\anaconda3\lib\site-packages (fr
om folium) (2.25.1)
Requirement already satisfied: branca>=0.3.0 in c:\users\hp\anaconda3\lib\site-package
s (from folium) (0.4.2)
Requirement already satisfied: jinja2>=2.9 in c:\users\hp\anaconda3\lib\site-packages
(from folium) (2.11.3)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\hp\anaconda3\lib\site-pack
ages (from jinja2>=2.9->folium) (1.1.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\hp\anaconda3\lib\site-pa
ckages (from requests->folium) (2020.12.5)
Requirement already satisfied: idna<3,>=2.5 in c:\users\hp\anaconda3\lib\site-packages
(from requests->folium) (2.10)
Requirement already satisfied: chardet<5,>=3.0.2 in c:\users\hp\anaconda3\lib\site-pac
kages (from requests->folium) (4.0.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\hp\anaconda3\lib\site
-packages (from requests->folium) (1.26.4)
```

In [66]:
```
map = folium.Map(location=[22.00,78.00], tiles='cartodbpositron', zoom_start=6)
for i in range(0,len(data)):
    Circle(
      location=[data.iloc[i]['LONGITUDE'], data.iloc[i]['LATITUDE']],
     radius=100,
    color='green').add_to(map)

#Display the map
map
```

Out[66]:



In [ ]:

In [ ]: