Gurugram Flats Price Prediction

Submitted by:
Vinayak Chhabra - 2024010118
Srishti – 2024010106

MCA 2nd Year

Submitted to:

Dr. Anjula Mehto

Assistant Professor

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Computer Science and Engineering Department

Thapar Institute of Engineering and Technology, Patiala

November 2025

TABLE OF CONTENTS

Introduction

Real estate prices in Gurugram have always been dynamic. With the city expanding rapidly through new residential sectors, major corporate offices, metro connectivity, and highway development, the price of a house can change significantly even between neighbouring areas. For homebuyers, investors, and students studying real estate trends, understanding these price variations is important. However, manually estimating the price of a property is often difficult because it depends on many factors such as location, size, property type, and available amenities.

To address this challenge, this project focuses on building a **Machine Learning-based House Price Prediction System for Gurugram**. The idea is to use real historical data of houses and flats in different sectors of the city and train a model that can automatically estimate the price of a property based on its features. Important attributes like **built-up area, number of bedrooms and bathrooms, the sector in which the property is located, and the type of property (flat or independent house)** are used to find patterns that influence the final price.

In today’s real estate market, buyers want data-driven insights instead of rough guesses, and this project aims to provide exactly that. By applying Machine Learning algorithms, we try to understand how different property features contribute to the overall value. The system learns from past listings and generates a price estimate that is more consistent and analytical compared to traditional methods of evaluation.

This project also highlights the practical use of data science techniques—starting from cleaning the dataset, handling missing values, transforming categorical sectors into numerical form, selecting useful features, and applying regression algorithms to build the prediction model. The evaluation of the model helps us understand how accurately it can predict prices when tested on unseen properties.

Overall, **Gurugram House Price Prediction** aims to make property price estimation more accessible, clear, and data-driven. It shows how Machine Learning can simplify decision-making in real estate and help users understand market behaviour in one of India’s fastest-growing cities. The project also provides a learning experience in applying real-world data science concepts to a practical and relevant problem.

Problem Statement

Property pricing in Gurugram has become increasingly complex due to rapid urban development, new residential sectors, varying builder qualities, and constantly changing market conditions. As a result, determining the fair market value of a house or flat is not straightforward. Prices differ significantly even within the same sector, and factors such as built-up area, number of rooms, location advantages, type of property, and nearby amenities contribute to these variations. Most buyers, sellers, and students studying real estate trends rely on general assumptions, personal experience, or broker suggestions, which may not always provide accurate or consistent results.

The key problem is the lack of a systematic, data-driven method to estimate the price of a property in Gurugram. The real estate market is influenced by multiple features at the same time, and manually analysing these factors becomes difficult. Traditional methods do not utilize the potential of historical data, making it challenging to understand how each feature contributes to the final price. With hundreds of listings spread across different sectors, it becomes necessary to develop a more objective and analytical approach.

This project aims to solve this problem by building a Machine Learning model that can predict the approximate price of a house in Gurugram based on actual data. The model uses important features like built-up area, number of bedrooms and bathrooms, sector location, and property type. By learning patterns from the dataset, the system will be able to estimate prices more accurately than manual methods. This approach reduces guesswork and provides a reliable way to understand price behaviour across different residential regions of the city.

The problem also includes challenges such as handling inconsistent or missing data, converting categorical sector names into numerical form, selecting the right features, and choosing a suitable regression algorithm. Addressing these challenges is essential for building a stable and accurate prediction system.

Therefore, the main objective of the problem is to design a predictive model that simplifies house price estimation in Gurugram by using data-driven techniques, making the process more transparent, efficient, and accessible for users who need quick and meaningful insights into the local real estate market.

Overview of the Dataset used

The dataset used in this project was created by web-scraping real estate listings from **99acres.com**, one of India's largest property platforms. Since there is no publicly available structured dataset specifically for Gurugram house prices, web scraping was used to collect real and up-to-date information directly from active property listings. This ensures that the price patterns captured in the dataset closely reflect the current real estate market conditions of Gurugram.

The final dataset contains **over 3,000 rows** of cleaned and structured data, representing individual residential properties across multiple sectors of Gurugram. Each row corresponds to one flat or house listing. The dataset includes **around 19 columns**, covering a wide range of important property features that influence price. These columns help the Machine Learning model understand how different characteristics contribute to the overall value of a property.

Some of the key columns in the dataset include:

- **Location / Sector** – The sector number or locality where the property is located.
- **Built-up Area** – Total covered area (sq. ft), which is one of the strongest indicators of price.
- **Bedrooms (BHK)** – Number of bedrooms.
- **Bathrooms** – Total number of bathrooms.
- **Property Type** – Whether the property is a flat, builder floor, or independent house.
- **Furnishing Status** – Unfurnished, semi-furnished, or fully furnished.
- **Floor Number** – The floor on which the unit is located (ground, mid, or high floor).
- **Age of Property** – Approximate construction age (0–5 years, 5–10 years, etc.).
- **Price –** The target variable that the model learns to predict.
- **Additional Amenities** – Lift, security, power backup, etc. (depending on scraping availability).

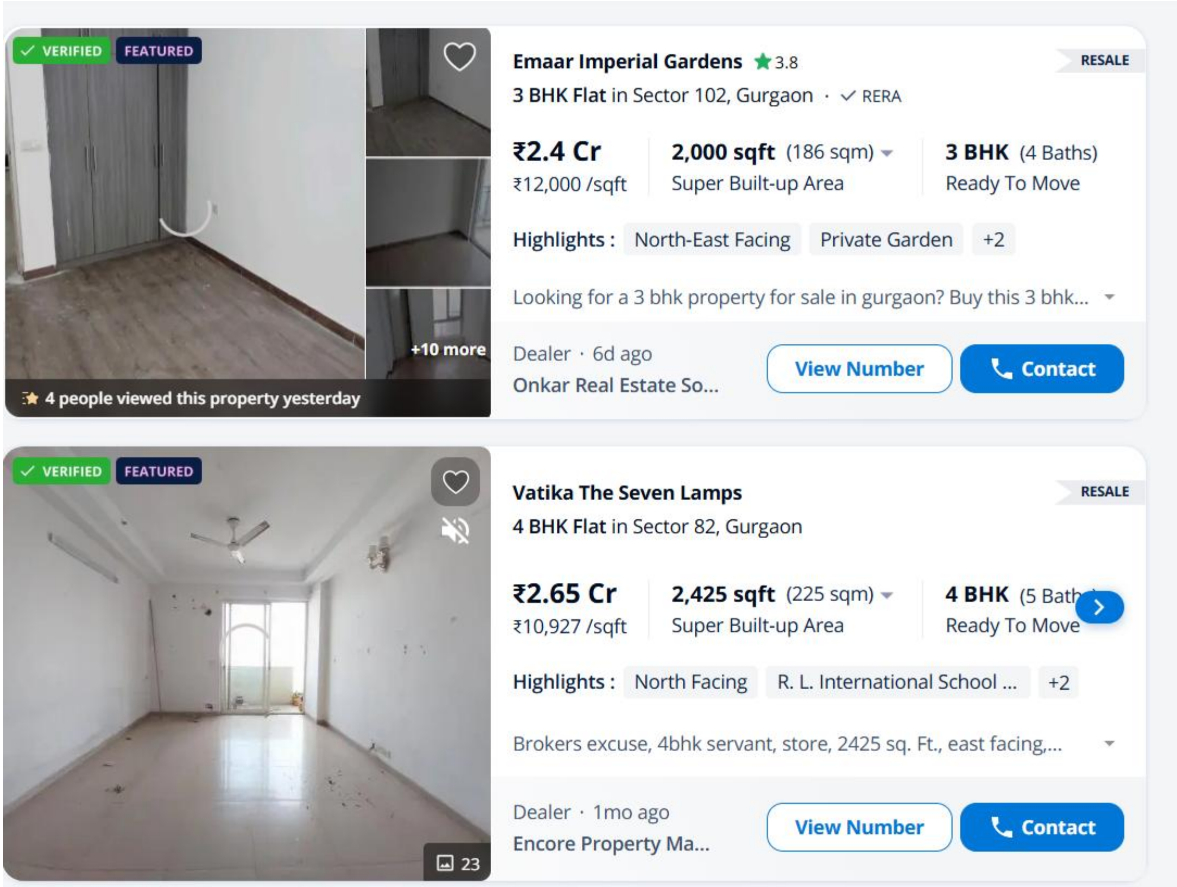
After scraping, the raw data had to be cleaned because many listings contain missing values, inconsistent formatting, or unstructured text. The dataset was cleaned by removing duplicates, handling missing values, standardizing numerical columns, and converting categorical data such as sectors into model-friendly numeric labels.

Project Workflow

The workflow of the *Gurugram House Price Prediction* project consists of a structured series of steps that guide the development of the prediction model from data collection to final evaluation. Each step is designed to ensure that the dataset is reliable, the features are meaningful, and the Machine Learning model is trained effectively to generate accurate predictions. The major stages of the workflow are explained below.

1. Data Collection

The project begins with collecting real estate data by web scraping property listings from 99acres.com. This process provides a large and diverse dataset containing more than 3,000 property records from various sectors of Gurugram. Information such as built-up area, number of rooms, bathrooms, location, property type, and price is extracted for further use.



Emaar Imperial Gardens ★3.8

3 BHK Flat in Sector 102, Gurgaon · ✓ RERA

₹2.4 Cr ₹12,000 /sqft

2,000 sqft (186 sqm) Super Built-up Area

3 BHK (4 Baths) Ready To Move

Highlights : North-East Facing | Private Garden | +2

Looking for a 3 bhk property for sale in gurgaon? Buy this 3 bhk...

Dealer · 6d ago

Onkar Real Estate So...

Vatika The Seven Lamps

4 BHK Flat in Sector 82, Gurgaon

₹2.65 Cr ₹10,927 /sqft

2,425 sqft (225 sqm) Super Built-up Area

4 BHK (5 Bath Ready To Move

Highlights : North Facing | R. L. International School ... | +2

Brokers excuse, 4bhk servant, store, 2425 sq. Ft., east facing,...

Dealer · 1mo ago

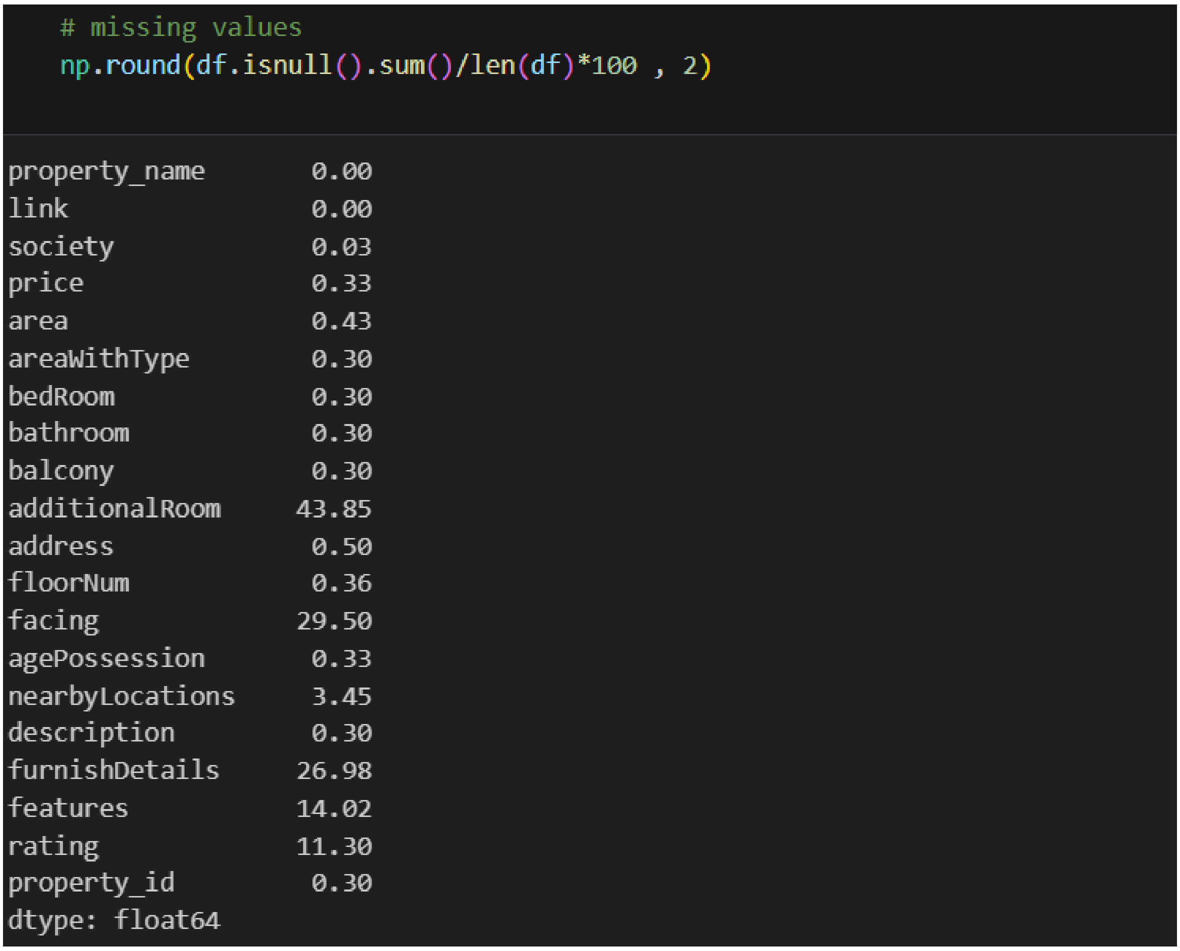
Encore Property Ma...

2. Data Cleaning

The raw scraped data often contains inconsistencies such as missing values, duplicate records, extra symbols, and unstructured text. This step involves:

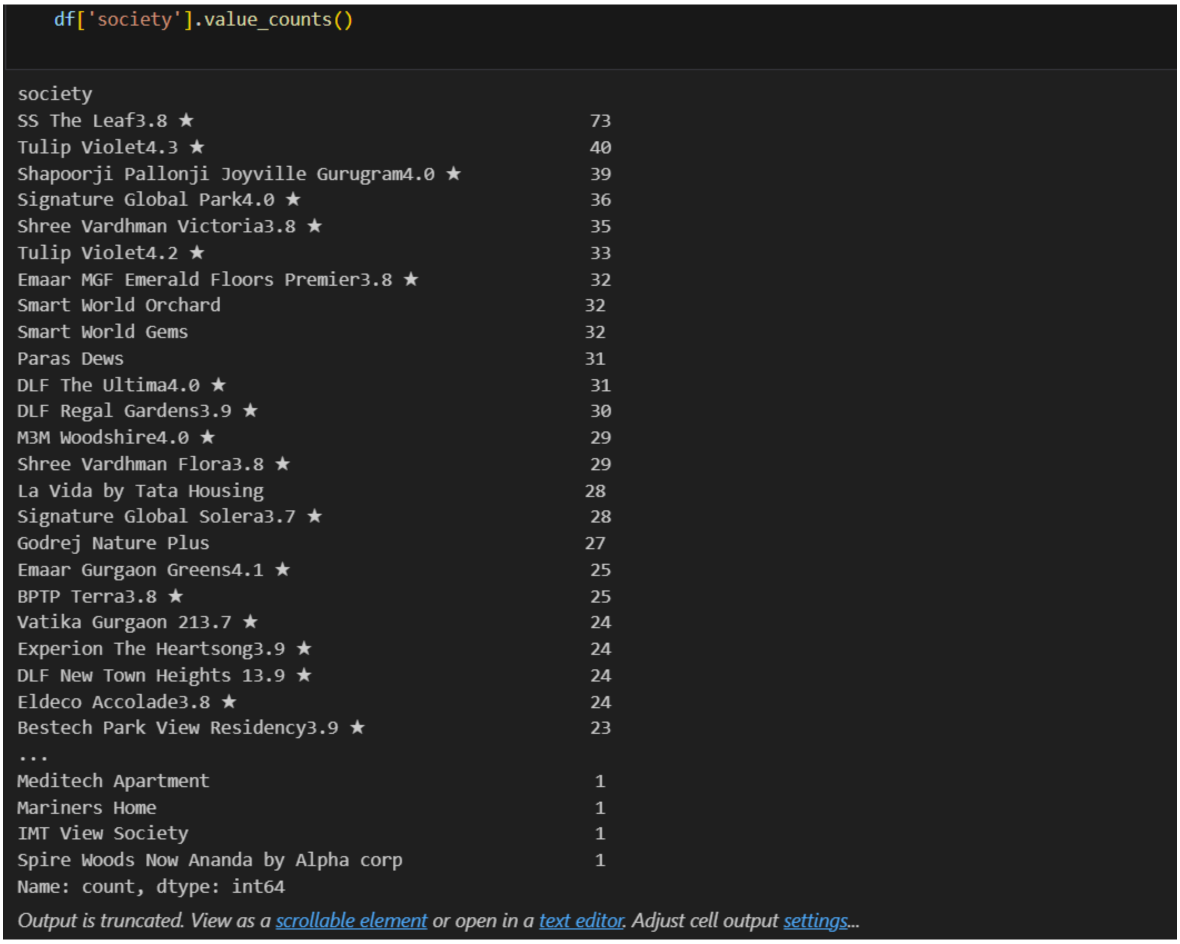
- Removing duplicate entries
- Handling missing values
- Standardizing numerical fields
- Cleaning text-based columns
- Filtering out incomplete or irrelevant entries

This ensures a clean and accurate dataset for analysis.



```
# missing values
np.round(df.isnull().sum()/len(df)*100 , 2)
```

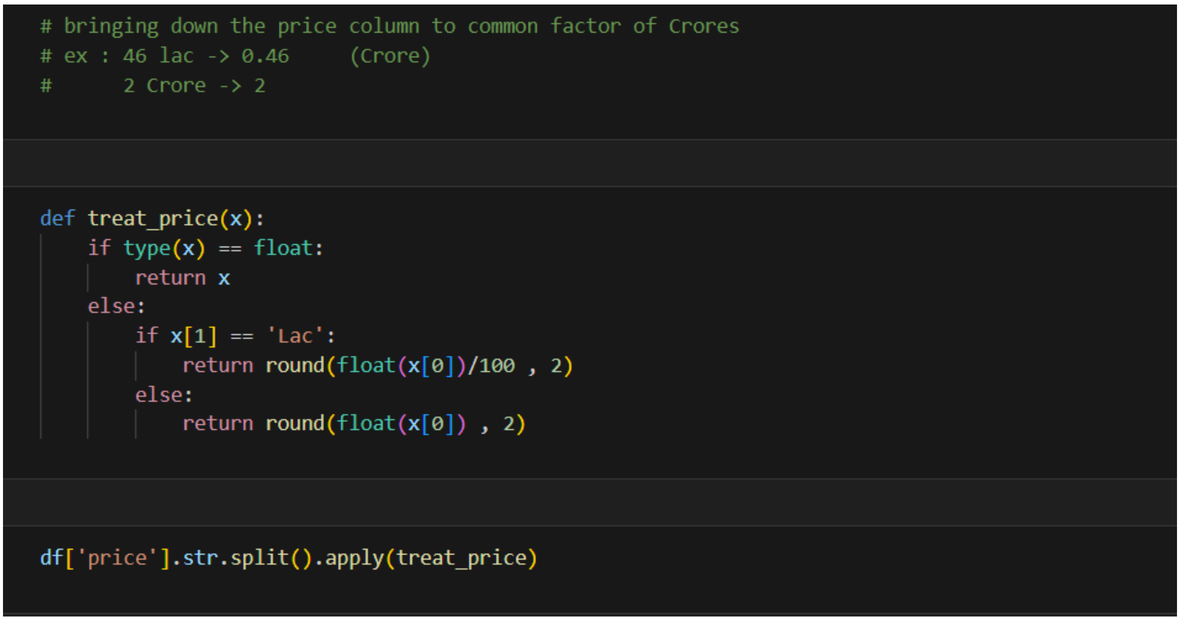
```
property_name        0.00
link                 0.00
society              0.03
price                0.33
area                 0.43
areaWithType         0.30
bedRoom              0.30
bathroom             0.30
balcony              0.30
additionalRoom      43.85
address              0.50
floorNum             0.36
facing              29.50
agePossession        0.33
nearbyLocations      3.45
description          0.30
furnishDetails      26.98
features            14.02
rating              11.30
property_id          0.30
dtype: float64
```



```
df['society'].value_counts()
```

```
society
SS The Leaf3.8 ★                                     73
Tulip Violet4.3 ★                                    40
Shapoorji Pallonji Joyville Gurugram4.0 ★            39
Signature Global Park4.0 ★                           36
Shree Vardhman Victoria3.8 ★                         35
Tulip Violet4.2 ★                                    33
Emaar MGF Emerald Floors Premier3.8 ★                32
Smart World Orchard                                  32
Smart World Gems                                     32
Paras Dews                                           31
DLF The Ultima4.0 ★                                  31
DLF Regal Gardens3.9 ★                               30
M3M Woodshire4.0 ★                                   29
Shree Vardhman Flora3.8 ★                            29
La Vida by Tata Housing                              28
Signature Global Solera3.7 ★                         28
Godrej Nature Plus                                   27
Emaar Gurgaon Greens4.1 ★                            25
BPTP Terra3.8 ★                                      25
Vatika Gurgaon 213.7 ★                               24
Experion The Heartsong3.9 ★                          24
DLF New Town Heights 13.9 ★                          24
Eldeco Accolade3.8 ★                                 24
Bestech Park View Residency3.9 ★                     23
...
Meditech Apartment                                    1
Mariners Home                                         1
IMT View Society                                      1
Spire Woods Now Ananda by Alpha corp                  1
Name: count, dtype: int64
```

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...



```
# bringing down the price column to common factor of Crores
# ex : 46 lac -> 0.46     (Crore)
#      2 Crore -> 2
```

```
def treat_price(x):
    if type(x) == float:
        return x
    else:
        if x[1] == 'Lac':
            return round(float(x[0])/100 , 2)
        else:
            return round(float(x[0]) , 2)
```

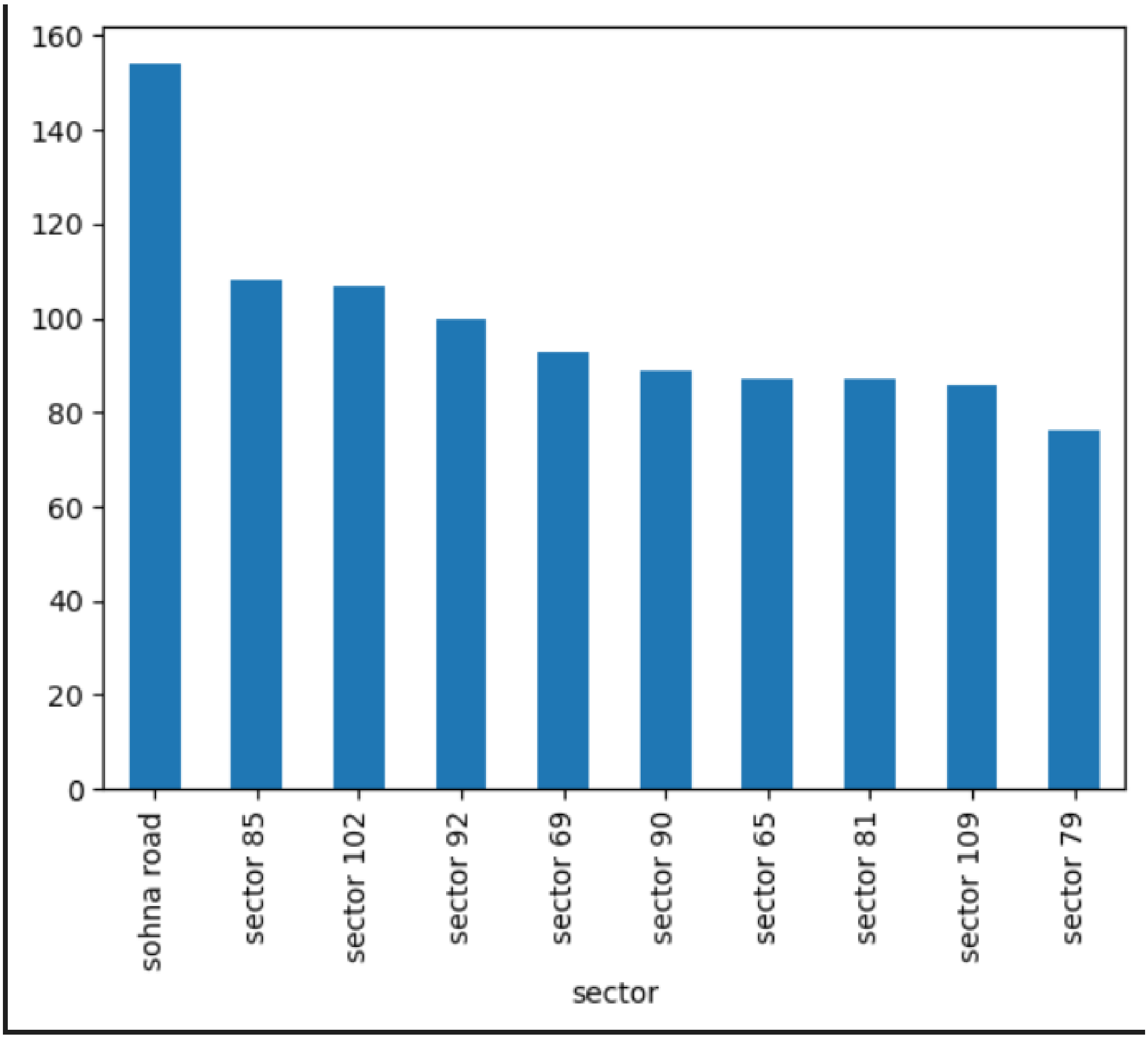
```
df['price'].str.split().apply(treat_price)
```

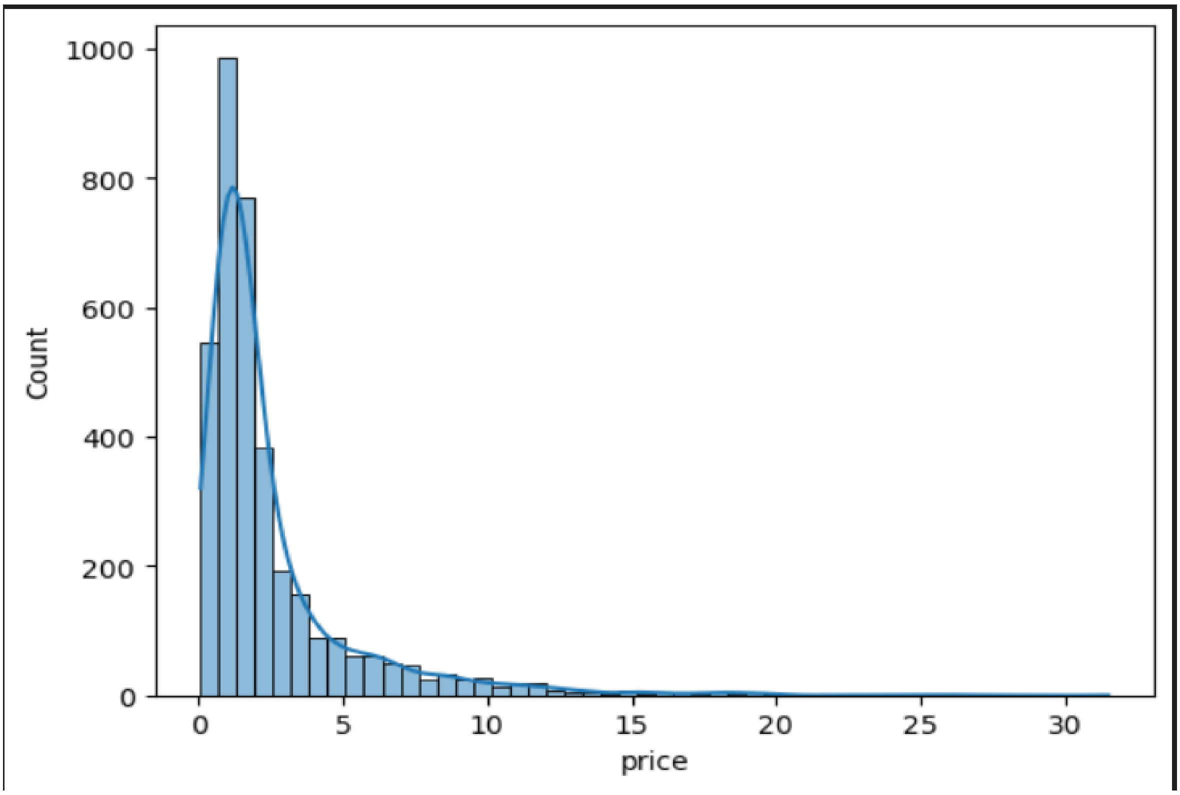
3. Exploratory Data Analysis (EDA)

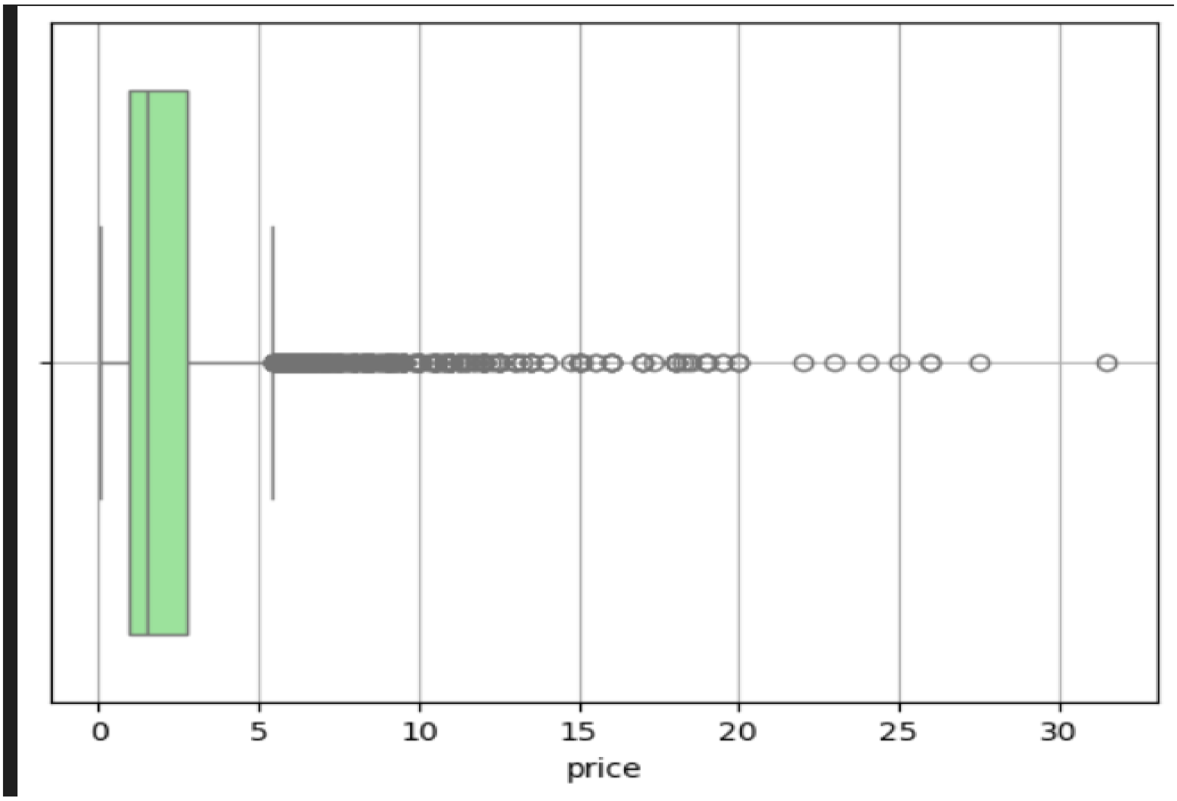
EDA helps in understanding the patterns, trends, and distribution of data. During this stage:

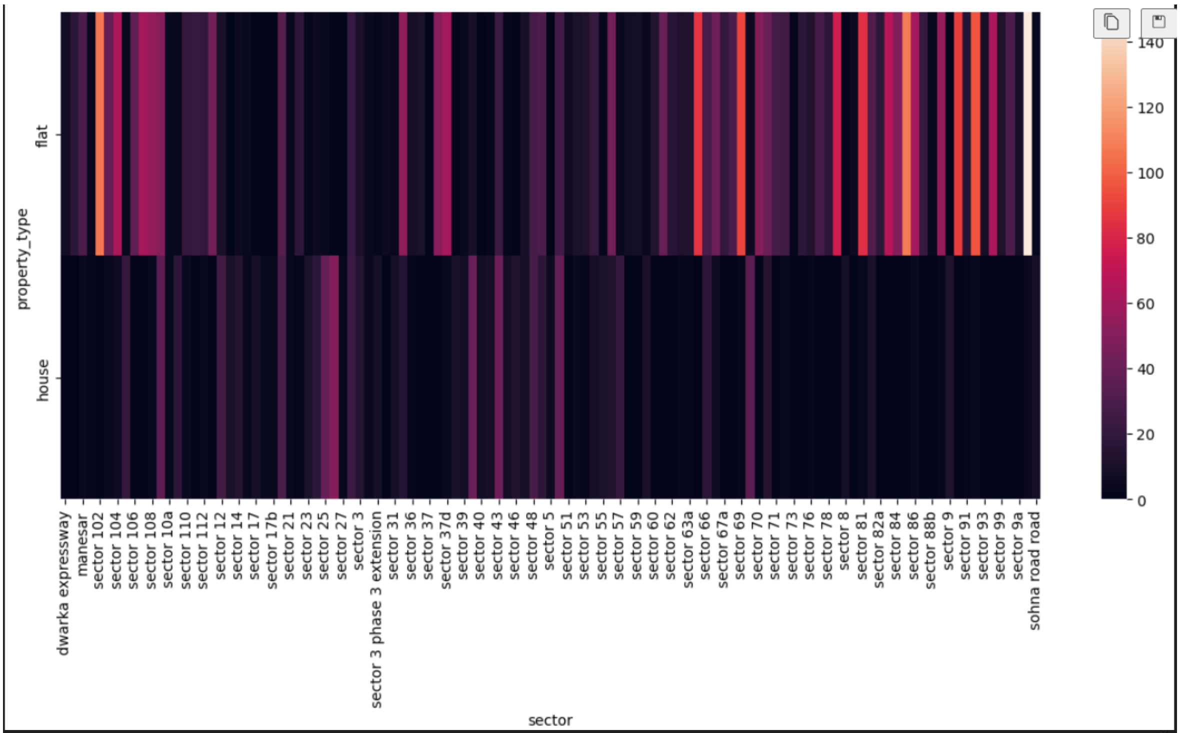
- Statistical summaries are generated
- Outliers and unusual values are identified
- Relationships between features (like area vs. price) are visualized
- Sector-wise and BHK-wise price trends are examined

These insights help in feature selection and model preparation.

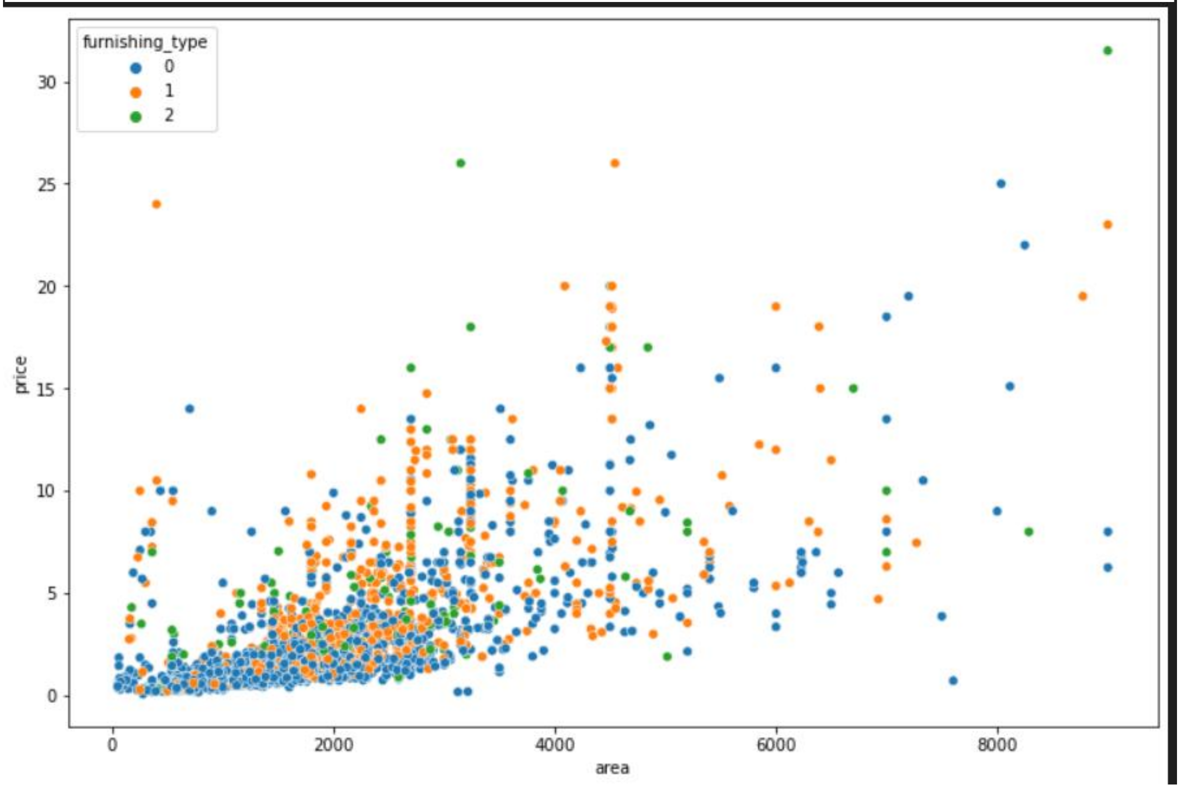



1000
800
600
400
200
0
Count
0
5
10
15
20
25
30
price


0
5
10
15
20
25
30
price



property_type
flat
house
sector
dwarka expressway
manesar
sector 102
sector 104
sector 106
sector 108
sector 10a
sector 110
sector 112
sector 12
sector 14
sector 17
sector 17b
sector 21
sector 23
sector 25
sector 27
sector 3
sector 3 phase 3 extension
sector 31
sector 36
sector 37
sector 37d
sector 39
sector 40
sector 43
sector 46
sector 48
sector 5
sector 51
sector 53
sector 55
sector 57
sector 59
sector 60
sector 62
sector 63a
sector 66
sector 67a
sector 69
sector 70
sector 71
sector 73
sector 76
sector 78
sector 8
sector 81
sector 82a
sector 84
sector 86
sector 88b
sector 9
sector 91
sector 93
sector 99
sector 9a
sohna road road
140
120
100
80
60
40
20
0



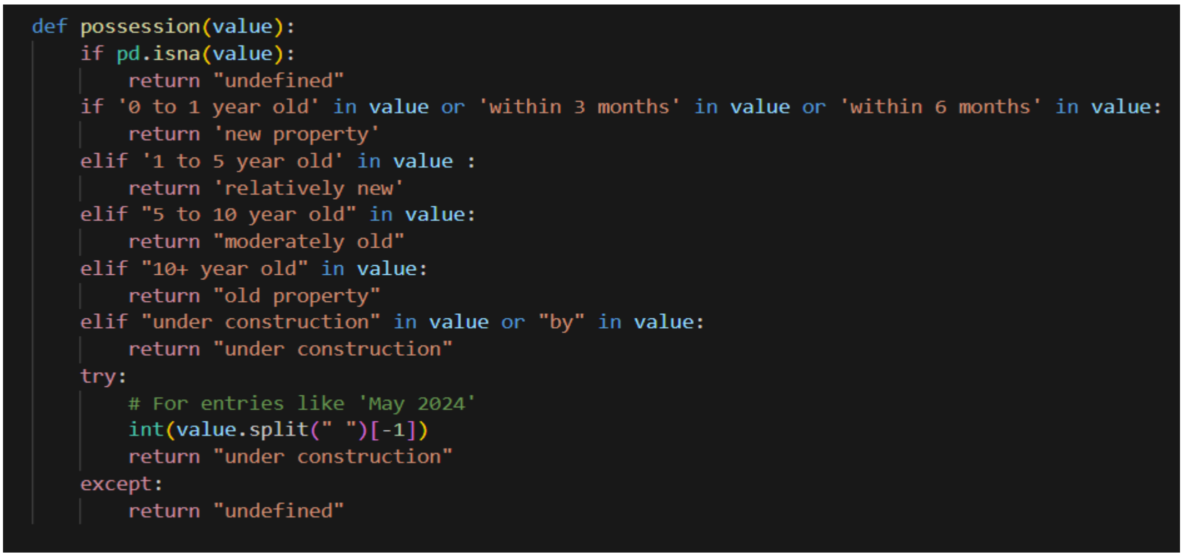
furnishing_type
0
1
2
price
30
25
20
15
10
5
0
area
0
2000
4000
6000
8000

4. Feature Engineering

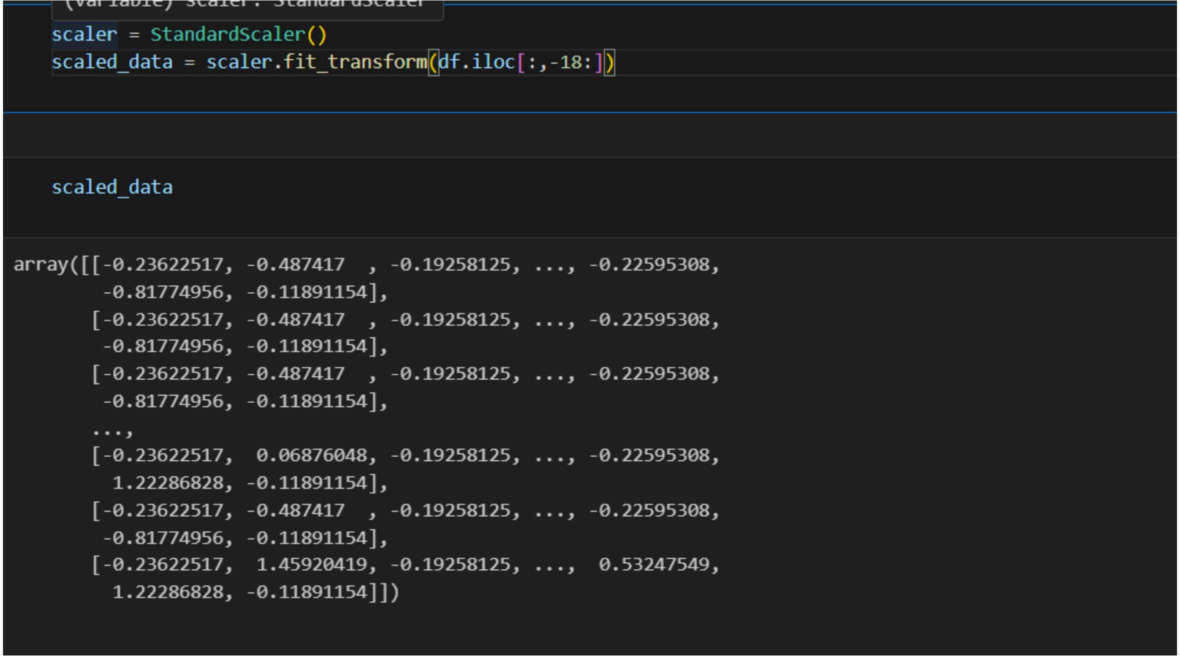
In this step, the dataset is transformed into a format suitable for Machine Learning algorithms. Key tasks include:

- Converting categorical data (e.g., sectors, furnishing status) into numerical labels
- Scaling or normalizing continuous variables
- Selecting the most important features influencing price
- Creating any new derived features if needed

This step ensures that the model can correctly interpret the property attributes.



```python
def possession(value):
    if pd.isna(value):
        return "undefined"
    if '0 to 1 year old' in value or 'within 3 months' in value or 'within 6 months' in value:
        return 'new property'
    elif '1 to 5 year old' in value :
        return 'relatively new'
    elif "5 to 10 year old" in value:
        return "moderately old"
    elif "10+ year old" in value:
        return "old property"
    elif "under construction" in value or "by" in value:
        return "under construction"
    try:
        # For entries like 'May 2024'
        int(value.split(" ")[-1])
        return "under construction"
    except:
        return "undefined"
```



```
  (variable) scaler: StandardScaler
scaler = StandardScaler()
scaled_data = scaler.fit_transform(df.iloc[:,-18:])
```

```
scaled_data
```

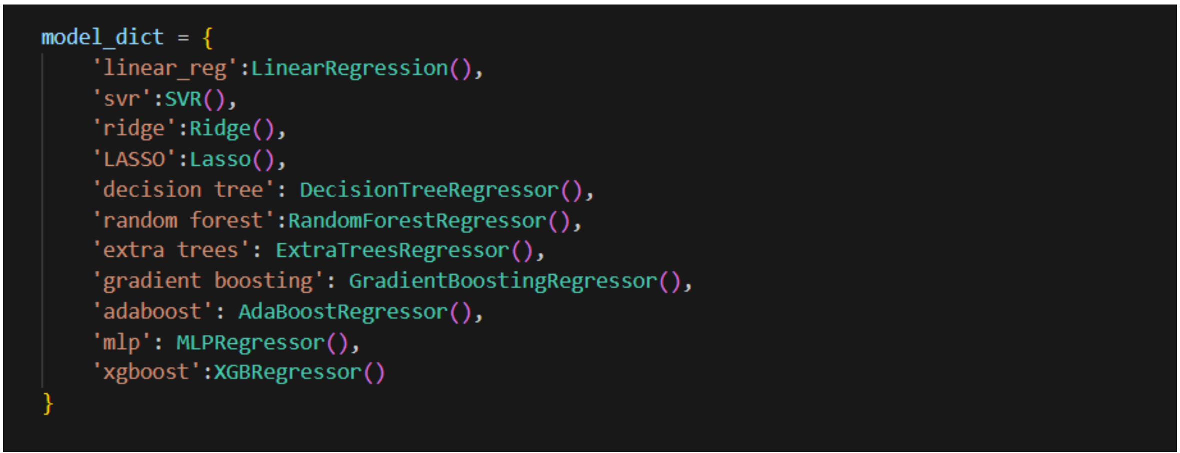
```
array([[-0.23622517, -0.487417  , -0.19258125, ..., -0.22595308,
        -0.81774956, -0.11891154],
       [-0.23622517, -0.487417  , -0.19258125, ..., -0.22595308,
        -0.81774956, -0.11891154],
       [-0.23622517, -0.487417  , -0.19258125, ..., -0.22595308,
        -0.81774956, -0.11891154],
       ...,
       [-0.23622517,  0.06876048, -0.19258125, ..., -0.22595308,
         1.22286828, -0.11891154],
       [-0.23622517, -0.487417  , -0.19258125, ..., -0.22595308,
        -0.81774956, -0.11891154],
       [-0.23622517,  1.45920419, -0.19258125, ...,  0.53247549,
         1.22286828, -0.11891154]])
```

5. Model Selection and Training

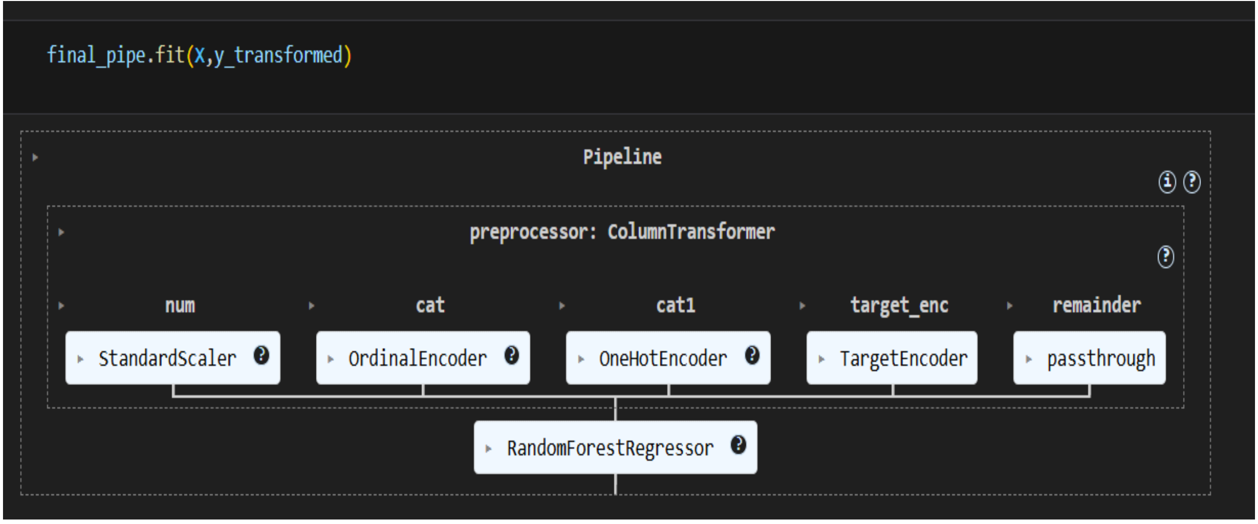
Multiple regression algorithms are tested to find the best-performing model for predicting house prices. Common techniques include:

- Linear Regression
- Ridge/Lasso Regression
- Random Forest Regressor
- Gradient Boosting Regressor

The selected models are trained on the processed dataset to learn the patterns between features and prices.



```
model_dict = {
    'linear_reg':LinearRegression(),
    'svr':SVR(),
    'ridge':Ridge(),
    'LASSO':Lasso(),
    'decision tree': DecisionTreeRegressor(),
    'random forest':RandomForestRegressor(),
    'extra trees': ExtraTreesRegressor(),
    'gradient boosting': GradientBoostingRegressor(),
    'adaboost': AdaBoostRegressor(),
    'mlp': MLPRegressor(),
    'xgboost':XGBRegressor()
}
```



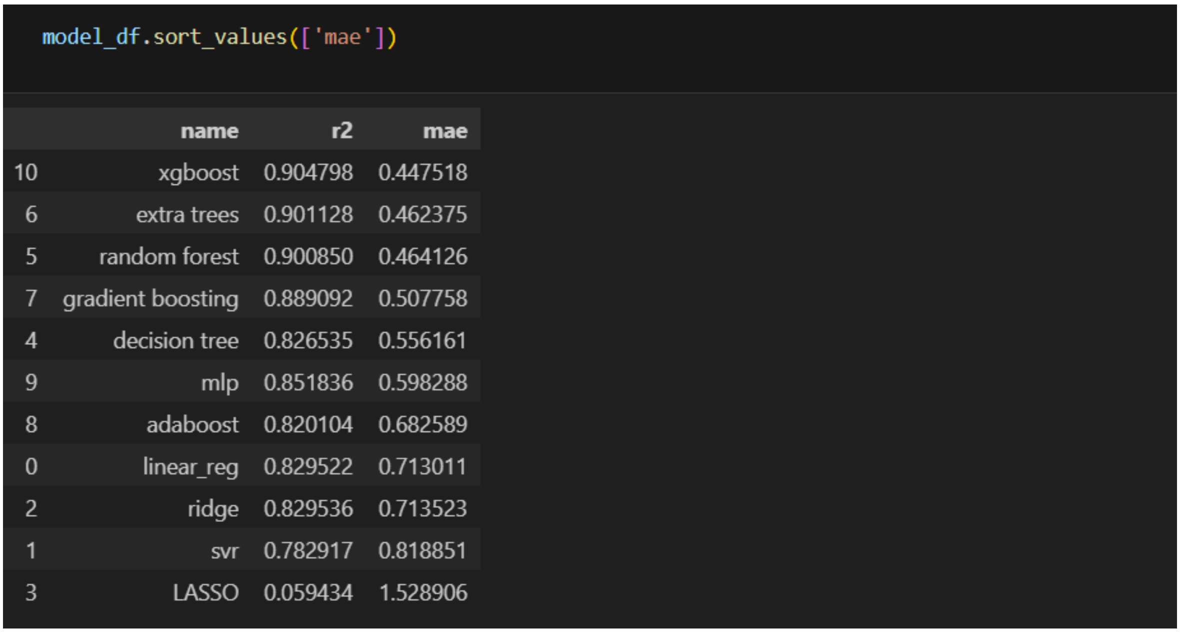
```
final_pipe.fit(X,y_transformed)
```

6. Model Evaluation

The trained models are evaluated using metrics such as:

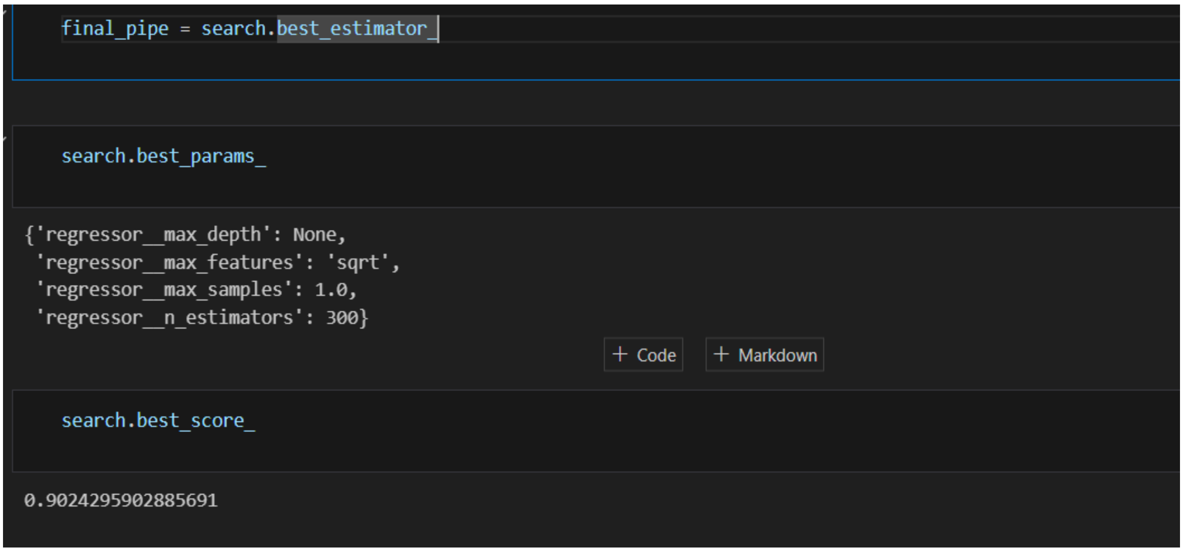
- **R² Score**
- **Mean Absolute Error (MAE)**
- **Root Mean Squared Error (RMSE)**

This step helps in comparing different models and choosing the most accurate and reliable one for final use.



```
model_df.sort_values(['mae'])
```

	name	r2	mae
10	xgboost	0.904798	0.447518
6	extra trees	0.901128	0.462375
5	random forest	0.900850	0.464126
7	gradient boosting	0.889092	0.507758
4	decision tree	0.826535	0.556161
9	mlp	0.851836	0.598288
8	adaboost	0.820104	0.682589
0	linear_reg	0.829522	0.713011
2	ridge	0.829536	0.713523
1	svr	0.782917	0.818851
3	LASSO	0.059434	1.528906



```
final_pipe = search.best_estimator_
```

```
search.best_params_
```

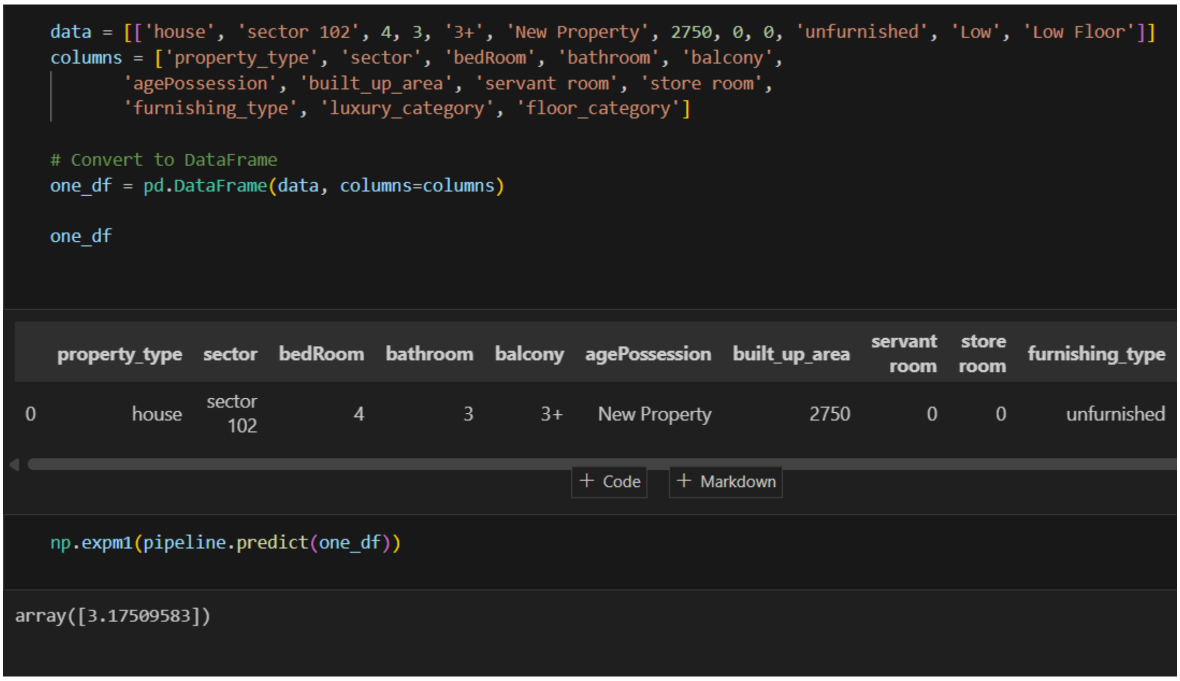
```
{'regressor__max_depth': None,
 'regressor__max_features': 'sqrt',
 'regressor__max_samples': 1.0,
 'regressor__n_estimators': 300}
```

```
search.best_score_
```

```
0.9024295902885691
```

7. Prediction Output

Random Forest Regressor is finalized, it is used to predict house prices based on user input. The model provides an approximate market price by analysing the property's features, making the system useful for buyers, sellers, and researchers.



```
data = [['house', 'sector 102', 4, 3, '3+', 'New Property', 2750, 0, 0, 'unfurnished', 'Low', 'Low Floor']]
columns = ['property_type', 'sector', 'bedRoom', 'bathroom', 'balcony',
        'agePossession', 'built_up_area', 'servant room', 'store room',
        'furnishing_type', 'luxury_category', 'floor_category']

# Convert to DataFrame
one_df = pd.DataFrame(data, columns=columns)

one_df
```

	property_type	sector	bedRoom	bathroom	balcony	agePossession	built_up_area	servant room	store room	furnishing_type
0	house	sector 102	4	3	3+	New Property	2750	0	0	unfurnished

```
np.expm1(pipeline.predict(one_df))
```

```
array([3.17509583])
```

Results

After completing data preprocessing and model training, several regression algorithms were evaluated to identify the most accurate model for predicting house prices in Gurugram. Models such as **Linear Regression**, **Ridge Regression**, **Random Forest Regressor**, and **Gradient Boosting Regressor** were tested on the dataset. Each model's performance was measured using metrics like **R^2 Score**, **Mean Absolute Error (MAE)**, and **Root Mean Squared Error (RMSE)** to understand how closely the predictions matched the actual prices.

Among all the models, the **Random Forest Regressor** delivered the best performance. Initially, its results were good, but after applying **hyperparameter tuning** (adjusting parameters such as the number of estimators, maximum depth, and minimum samples split), the model showed a significant improvement. The tuned Random Forest model achieved an **R^2 score of approximately 0.90**, meaning it was able to explain 90% of the variation in house prices based on the selected features. This level of accuracy indicates strong predictive capability and aligns well with the non-linear nature of real estate data.

The **MAE and RMSE** values for the tuned Random Forest model were also lower compared to other models, showing that the predicted prices stayed very close to the actual values on average. This suggests that the model generalizes well and does not overfit the training data.

Visual evaluation further supported the numerical results. The **actual vs. predicted price graph** showed that most data points were close to the ideal diagonal line, reflecting high accuracy. The **feature importance analysis** revealed that attributes such as **built-up area, sector location, BHK, and number of bathrooms** played the most important roles in determining prices—consistent with real market behaviour.

Overall, the results clearly show that the tuned Random Forest Regressor is the most suitable model for this project. With an R^2 score of 0.90, it provides reliable, stable, and realistic price estimates for residential properties across different sectors of Gurugram

Conclusion

The *Gurugram House Price Prediction* project demonstrates how Machine Learning can be effectively used to estimate real estate prices in a rapidly developing city like Gurugram. By collecting real data through web scraping, cleaning it, and analysing the important features, the project highlights the complete process of building a data-driven prediction system. The model learns how different factors—such as built-up area, sector location, number of bedrooms and bathrooms, and property type—contribute to the final market value of a house.

After testing multiple regression algorithms, the Random Forest Regressor with tuned hyperparameters proved to be the most accurate model, achieving an R^2 score of around 0.90. This shows that the model is able to capture complex patterns in the dataset and make predictions that are close to real market values. The results also align well with actual real estate trends observed in Gurugram, where location and property size significantly impact pricing.

The project successfully meets its primary goal: providing a reliable, data-driven method for estimating house prices. While no model can perfectly predict real estate values due to market changes, builder differences, and economic factors, this system offers a strong analytical foundation. It can assist students, homebuyers, and researchers in understanding price behaviour across different sectors of the city.

Overall, this project demonstrates the practical application of Machine Learning in solving real-world problems and shows how data science techniques can bring clarity to complex markets like real estate. The insights gained from this work can be extended further to create more robust tools in the future.

GitHub Link

LINK : **https://github.com/vinayak910/Data-Visualization-Assignments-/tree/main/Gurugram%20Price%20Prediction**