MBA Semester – IV

Research Project – Interim Report

Name	Sushma R
Project	House price prediction
Group	
Date of Submission	23-10-2025

A study on "House Price Prediction "

Research Project submitted to Jain Online

(Deemed-to-be University)

In partial fulfillment of the requirements

for the award of:

Master of Business Administration

Submitted by:

Sushma R

USN:

231VMBR04860

Under the guidance of:

Mention your Guide's Name

(Faculty-JAIN Online)

Jain Online (Deemed-to-be University)

Bangalore

2023-24

DECLARATION

I, *Sushma R,* hereby declare that the Research Project Report titled *"House Price Prediction" has been* prepared by me under the guidance of the *Faculty name.* I declare that this Project work is towards the partial fulfillment of the University Regulations for the award of the degree of Master of Business Administration by Jain University, Bengaluru. I have undergone a project for a period of Eight Weeks. I further declare that this Project is based on the original study undertaken by me and has not been submitted for the award of any degree/diploma from any other University / Institution.

Place: Bangalore

Date: 23-10-2025

Sushma R

USN:231VMBR04860

Contents

Executive Summary

The house price prediction model study develops a predSictive model to estimate residential property prices in **King County, Washington**, using statistical and machine learning techniques. The model leverages key housing attributes such as the number of bedrooms and bathrooms, total living area, lot size, location-based variables, and other structural or neighbourhood factors to forecast the selling price.

The analysis applies a systematic data preprocessing approach — including handling of extreme values through **winsorization** and removal of **highly correlated variables** — to improve model reliability. Multiple regression and ensemble methods, including **Linear Regression**, **Decision Tree Regressor**, **XGBoost**, **LightGBM**, and **CatBoost**, are implemented to evaluate performance and determine the most influential predictors of housing price. Model comparison is based on **Root Mean Square Error (RMSE)**, **R-squared**, **Adjusted R-squared**, and **cross-validation scores**. The model demonstrating the **lowest RMSE** and **highest goodness-of-fit** on the test dataset is identified as the most effective predictor. The findings indicate that **CatBoost** outperforms other models and provides superior predictive accuracy for the **King County housing market**.

Introduction

The main goal of this project is to predict house prices accurately. This is very important in the real estate industry because correct price estimates help buyers, sellers, and agents make better decisions. A reliable house price prediction model can help set fair listing prices, understand property values, find good investment opportunities, and support fair negotiations for everyone involved.

Defining problem statement

The goal is to build a model that can correctly predict the selling price of a house. Using details such as the house's location, size, number of rooms, and available facilities, the model should be able to estimate the price as accurately as possible.

Needs of study project

By addressing the house price prediction problem, we aim to provide value to both individual buyers and sellers as well as real estate professionals by empowering them with accurate price estimates and actionable insights based on data-driven predictions.

Understanding business

The housing market is influenced by a multitude of factors, including location, property size, number of bedrooms and bathrooms, amenities, proximity to schools and transportation, and economic indicators such as interest rates and market trends. Analyzing and understanding these factors can be complex and time-consuming, making it essential to employ advanced techniques such as machine learning to build an effective

predictive model

The objective is to leverage historical data on house attributes and their corresponding prices to develop a robust prediction model that can estimate the price of a house accurately. By utilizing

this model, prospective buyers can make informed decisions about affordability, and sellers can set competitive prices to attract potential buyers. Real estate professionals can also leverage this model to gain insights into market trends and assist clients in making well-informed investment decisions

Data Cleaning and Preprocessing:

1. **Importing Libraries:**

   The first step is to import all the required libraries for data handling, analysis, and visualization.

   - **pandas** and **numpy** are used for data manipulation and calculations.
   - **matplotlib** and **seaborn** are used for creating charts and graphs

2. **Libraries for Model Building:**

   To build and evaluate models, we import the following:

   - **Models:** Linear Regression, Ridge, Decision Tree Regressor, XGB Regressor, LGBM Regressor, and Cat Boost Regressor
   - **Encoding and Scaling:** Label Encoder, Min Max Scaler, StandardScaler
   - **Model Setup:** train_test_split, Grid Search CV
   - **Model Evaluation:** r2_score, mean_squared_error, mean_absolute_error, accuracy_score

3. **Raw unprocessed data**

	cid	dayhours	price	room_bed	room_bath	living_measure	lot_measure	ceil	coast	sight	condition	quality	ceil_measure	basement
0	3876100940	20150427T000000	600000	4.00	1.75	3050.00	9440.00	1	0	0.00	3	8.00	1800.00	1250.00
1	3145600250	20150317T000000	190000	2.00	1.00	670.00	3101.00	1	0	0.00	4	6.00	670.00	0.00
2	7129303070	20140820T000000	735000	4.00	2.75	3040.00	2415.00	2	1	4.00	3	8.00	3040.00	0.00
3	7338220280	20141010T000000	257000	3.00	2.50	1740.00	3721.00	2	0	0.00	3	8.00	1740.00	0.00
4	7950300670	20150218T000000	450000	2.00	1.00	1120.00	4590.00	1	0	0.00	3	7.00	1120.00	0.00

yr_built	yr_renovated	zipcode	lat	long	living_measure15	lot_measure15	furnished	total_area	City	County	Type
1966	0	98034	47.72	-122.183	2020.00	8660.00	0.00	12490	Kirkland	King	Standard
1948	0	98118	47.55	-122.274	1660.00	4100.00	0.00	3771	Seattle	King	Standard
1966	0	98118	47.52	-122.256	2620.00	2433.00	0.00	5455	Seattle	King	Standard
2009	0	98002	47.34	-122.213	2030.00	3794.00	0.00	5461	Auburn	King	Standard
1924	0	98118	47.57	-122.285	1120.00	5100.00	0.00	5710	Seattle	King	Standard

4. Removing Unwanted Columns

Removed columns cid and yr_renovated as they do not contribute meaningful information to the target variable (price).

5. Handling Missing and Invalid Values

Replaced missing values with appropriate measures (median for numerical columns).

- **Converted invalid entries (like '$' symbols)** into numerical form and handled conversion errors using pd. tonumeric(errors='coerce').
- The following variables were treated for missing or inconsistent values:
    - room_bed, room_bath → filled with median and rounded to an integer.
    - living_measure, lot_measure, ceil, coast, condition, basement, ceil_measure → replaced nulls with median after cleaning special characters ($, ,).
    - sight → replaced null values with **0**.
    - quality → replaced missing values with median and converted to an integer.
    - furnished → replaced missing values with median.
    - yr_built → rows with missing or invalid year were dropped.
    - total_area → missing or invalid ($) values replaced with sum of living_measure and lot_measure.

6. Date and Time Decomposition

- Decomposed the day hours column into:
    - sold_year
    - sold_month
    - sold_day

- sold_date (combined from the above three).

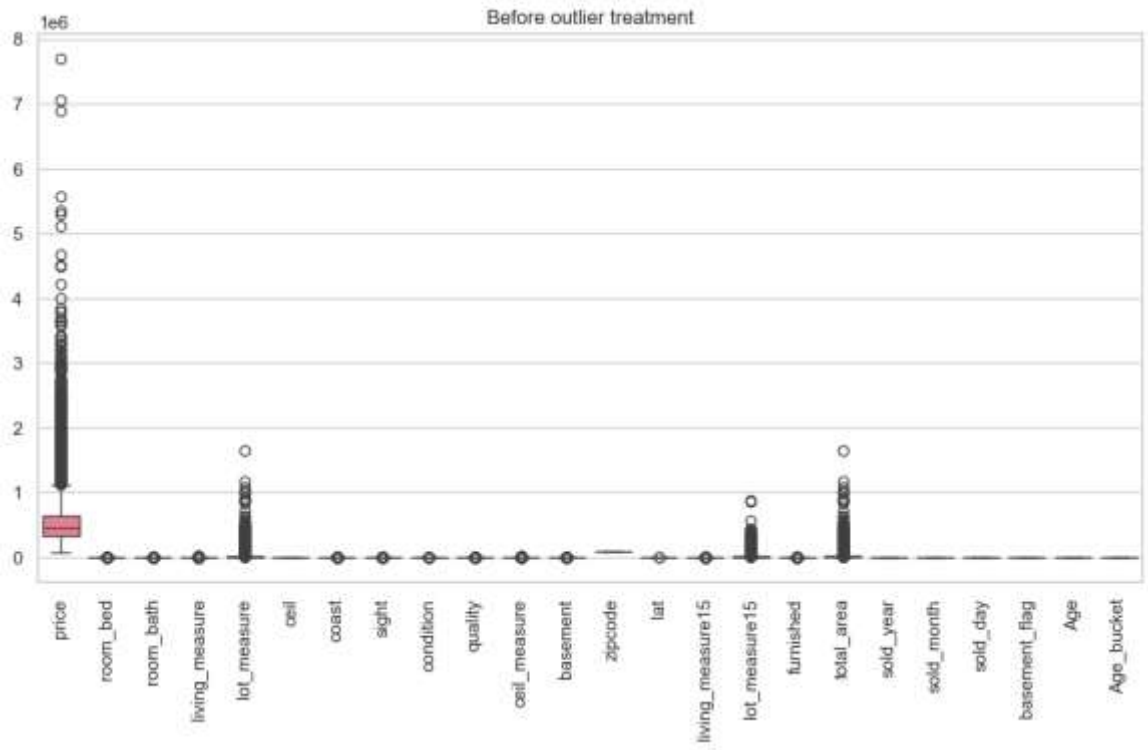
- Converted all date-related columns to integer type for analysis consistency.

7. **Feature Engineering**
   - **Basement Flag:** Created basement_flag — value 1 if basement > 0, else 0.
   - **Age:** Computed as 2015 - yr_built, representing the age of each property.
   - **Age Bucket:** Categorised properties into age ranges:
     - 0–10 years → 5
     - 10–20 years → 4
     - 20–30 years → 3
     - 30–40 years → 2
     - 40 years → 1

       This was stored in a categorical column, Age bucket.

8. **Data Type Conversions**
   - Ensured all numeric features are in appropriate numerical format (float or int).
   - Converted categorical features such as Age_bucket to object type for future encoding.

Before outlier treatment:



Graph 1.1 – Before Outlier Treatment

The boxplot represents the distribution of the numerical features **before** performing outlier treatment. From the visualisation, it is evident that certain columns such as **price**, **lot_measure**, **lot_measure15**, and **room_bed** contain extreme values that lie far beyond the interquartile range (IQR).

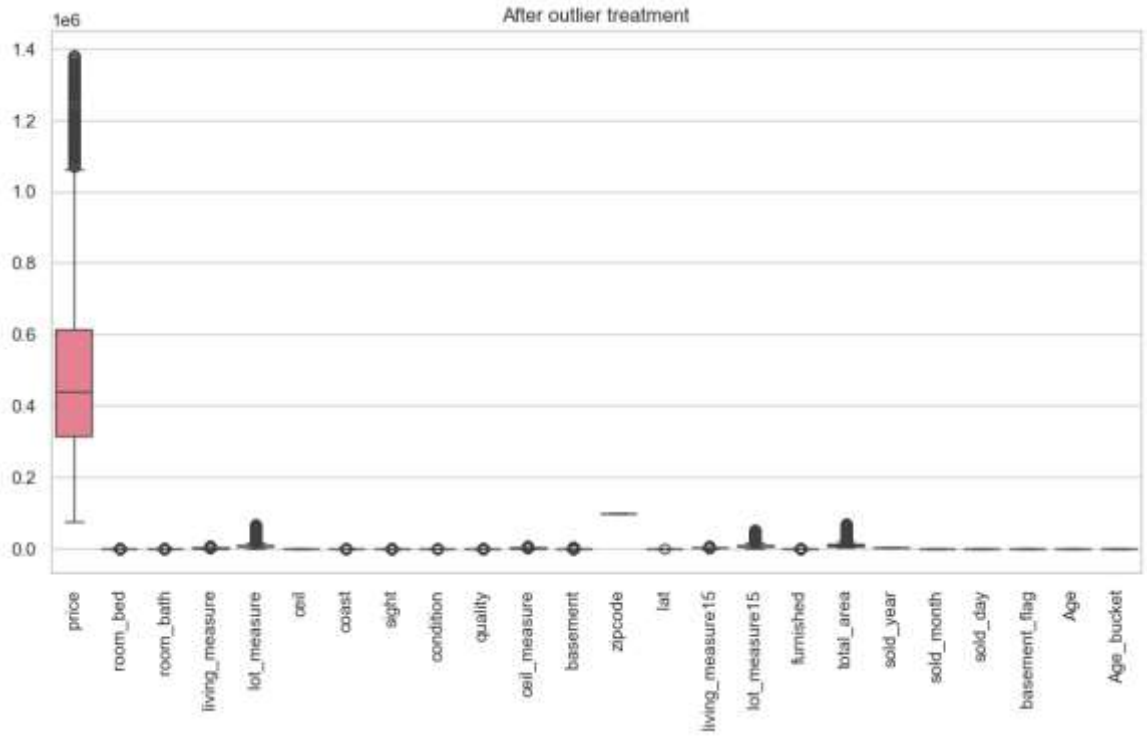
Graph 1.1

- `outliers = processing_data.quantile(0.97)`: Calculates the quantile range to exclude outliers.
- The code then filters the 'processing_data' DataFrame to remove outliers in the 'price', 'lot_measure', 'lot_measure15', and 'room_bed' columns using the calculated quantile range.
- Another boxplot is created to visualize the data after outlier treatment.

These extreme values indicate:

- Very high-priced houses compared to the majority of properties.
- Unusually large land sizes (lot_measure and lot_measure15).
- Unrealistic bedroom counts (room_bed) that may result from data entry errors.
- To mitigate the effect of these outliers, the **97th percentile** was used as a threshold for trimming extreme values.

After outlier treatment:



Graph 1.2 – After Outlier Treatment

The boxplot shown in *Graph 1.2* illustrates the distribution of numerical features after applying outlier treatment. From the visualisation, it is evident that the extreme values observed earlier in *Graph 1.1* have been significantly reduced. The data now appears more balanced and within a reasonable range.

Process Explanation:

- A **97th percentile quantile** (processing_data.quantile(0.97, numeric_only=True)) was calculated to determine the threshold beyond which values were considered outliers.
- Outliers were specifically removed from the columns:
    - price
    - lot_measure
    - lot_measure15
    - room_bed (values greater than 15 were excluded as unrealistic)
- The cleaned dataset was then reassigned to processed_data for further model training.

This process helped in maintaining the natural variability of the data while eliminating extreme distortions that could negatively affect the performance of regression models.

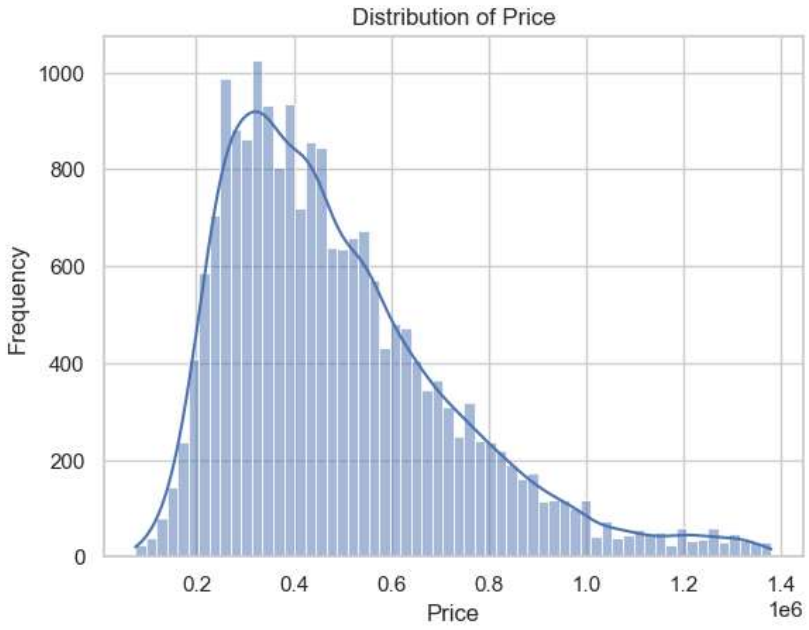
Observation:

- The **price** distribution is now more condensed, with most data points lying within the interquartile range (IQR).
- **lot_measure** and **lot_measure15** show a significant reduction in extreme values.
- Overall, the dataset appears smoother and more consistent, indicating that the outlier removal process was effective.

Exploratory data analysis:

Univariate Analysis:

1.Target variable analysis of its distribution:



Graph 2.1 – Distribution of Price

The histogram shown in *Graph 2.1* represents the **distribution of the target variable 'price'**.The plot is created using the sns.histplot() function with the kde=True parameter, which adds a **kernel density estimate (KDE)** curve to visualize the underlying probability density of house prices.

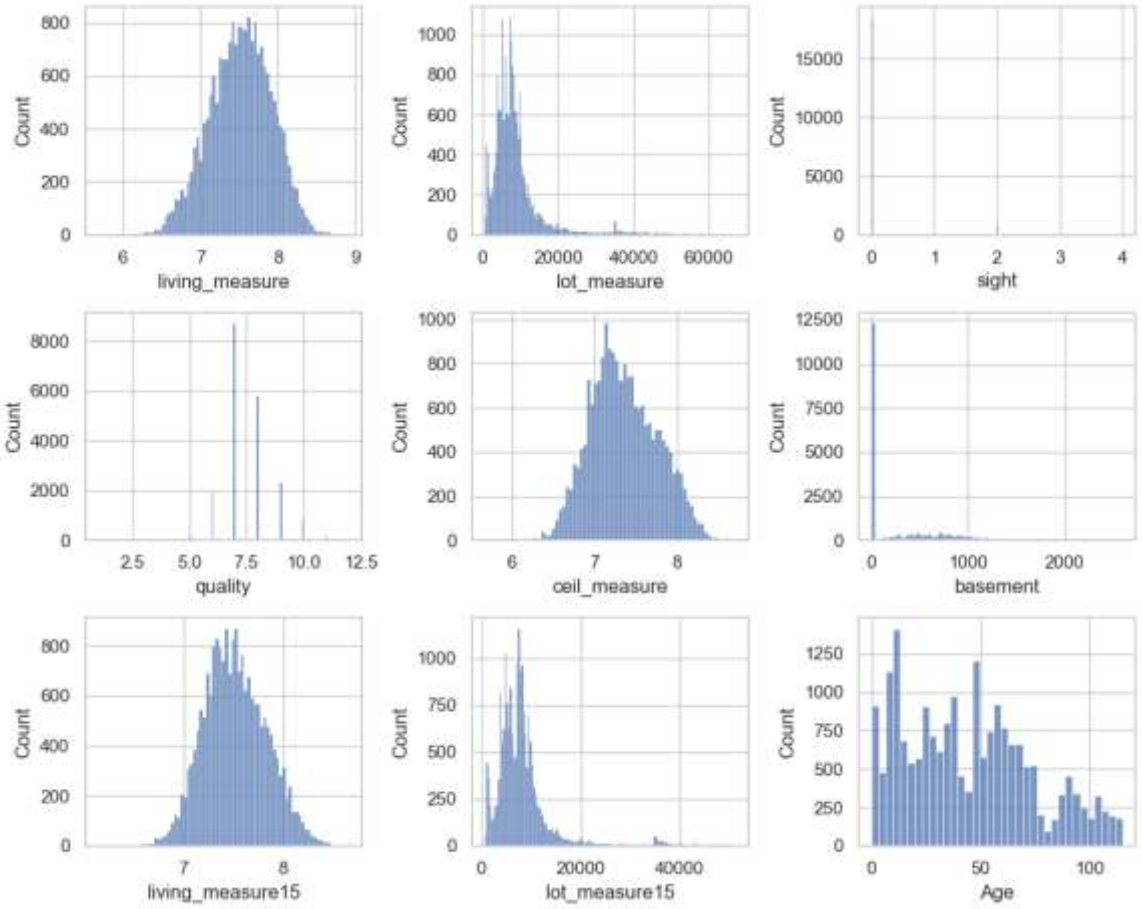
Observation:

- The distribution of house prices is **right-skewed**, indicating that most houses are priced on the lower end of the scale, while a few properties have very high prices.
- The peak of the curve lies between **$200,000 and $400,000**, suggesting that the majority of houses in King County fall within this price range.
- The long right tail shows that there are some **luxury properties** with much higher prices, which aligns with typical real estate market behaviour.

Inference:

- Since the target variable is not normally distributed, applying certain machine learning algorithms that assume normality (like Linear Regression) may result in biased predictions.
- To improve model performance, a **log transformation** of the 'price' variable can be considered to reduce skewness and stabilise variance.

Univariate analysis of continuous variables:



Graph 2.2 – Distribution of Numerical Features

The set of histograms shown in *Graph 2.2* represents the distribution of key **numerical variables** in the dataset. This analysis helps in understanding the spread, skewness, and general characteristics of each numeric feature used in the model.

Observation and Interpretation:

- **living_measure and living_measure15:**
  Both variables follow an approximately **normal distribution**, indicating that most houses have a moderate living area with fewer extremely small or large properties.
- **lot_measure and lot_measure15:**
  These features show a **right-skewed distribution**, meaning that while most properties have small to medium lot sizes, a few have exceptionally large land areas.

- **sight:**

  The distribution is highly **concentrated around zero**, showing that most properties have little or no scenic view advantage, with very few having high sight scores.
- **quality:**

  The data for house quality is **discrete and moderately skewed**, clustered around average-to-good quality ratings (5–8 range), with few very high or very low values.
- **ceil_measure:**

  This variable also approximates a **normal distribution**, suggesting that the ceiling height measurements are fairly consistent across properties.
- **basement:**

  The basement area distribution is **heavily skewed toward zero**, indicating that many houses have small or no basements, while a few have significantly larger ones.
- **Age:**

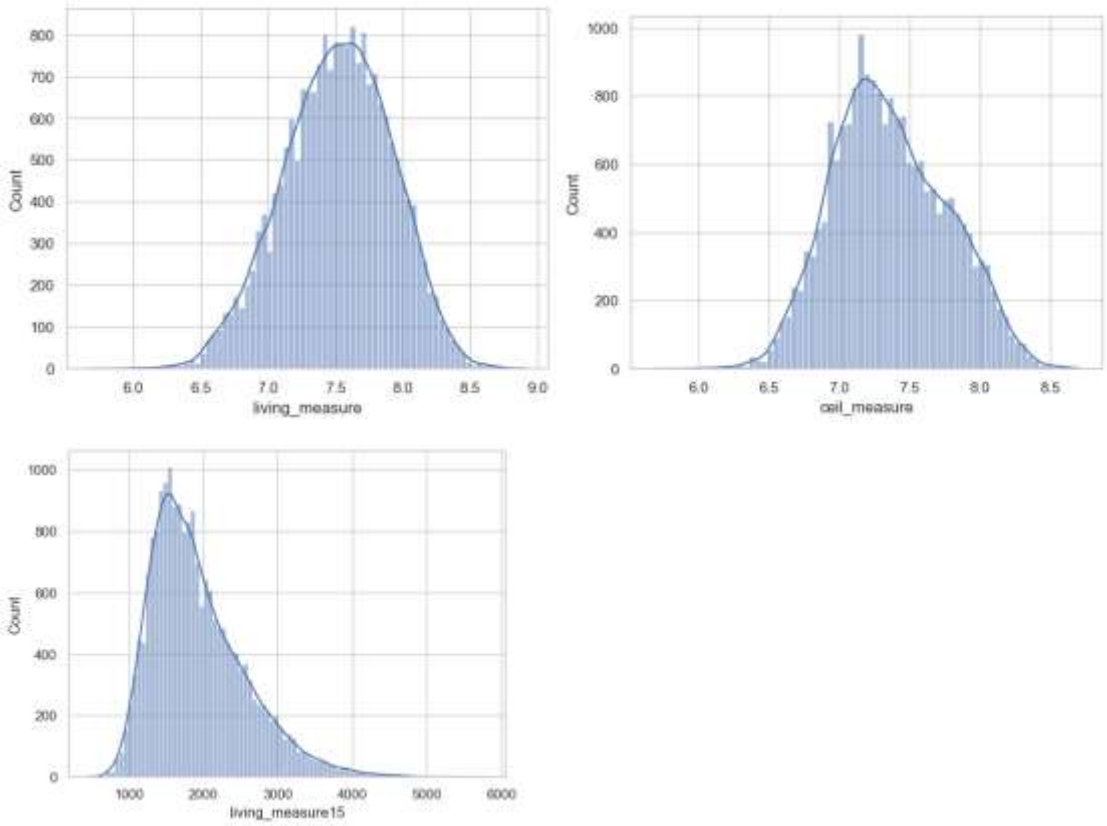
  The house age distribution is **slightly right-skewed**, showing that the dataset contains a larger number of newer homes compared to older ones.

Inference:

Most numerical features are not perfectly normally distributed — several are **positively skewed**, which may influence regression-based models. These distributions provide insights into feature variability and highlight where transformation techniques (like log scaling or standardization) might improve model accuracy.

3. Log transformation of variables:

Graph 2.3



Before Log Transformation

Before applying the log transformation, features such as **living_measure**, **ceil_measure**, and **living_measure15** displayed **right-skewed distributions**. This means most observations were concentrated on the lower end, while a few extreme values extended toward the higher side. Such skewness can negatively impact model training, especially for regression models that assume normality.

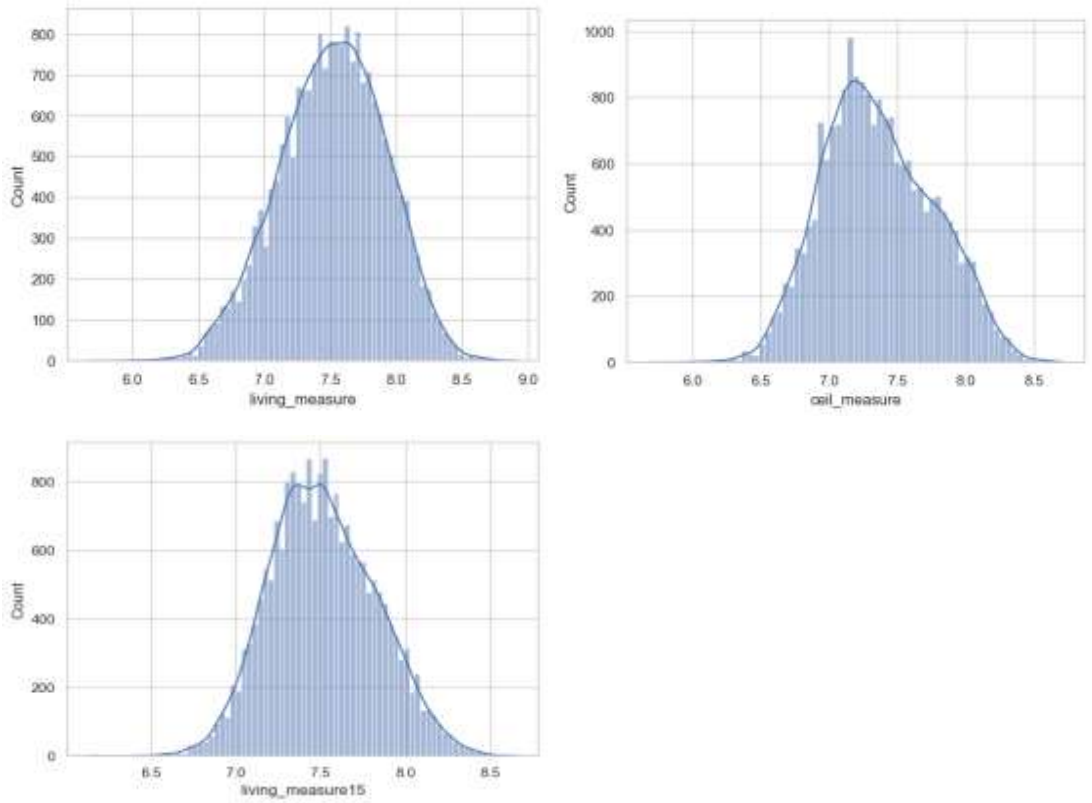
Key Observations:

- **living_measure**: Slight right skew, with most houses having moderate living area and few extremely large ones.
- **ceil_measure**: Slightly right-skewed distribution; most ceiling measurements are around the average, but a few homes have significantly higher ceilings.

- **living_measure15**: Highly right-skewed; few houses have much larger living areas compared to others in the neighbourhood.

This skewed pattern could lead to biased model predictions, as the regression algorithm may give disproportionate weight to these extreme values.

After the log transformation, the histograms for these variables are plotted again using `sns.histplot()` to show the improved distribution.



Graph 2.4

To reduce the skewness and improve model interpretability, a **logarithmic transformation** (np.log1p()) was applied to the above variables.

After transformation, the distributions became **more symmetrical and bell-shaped**, closely resembling a **normal distribution**.

Key Results:

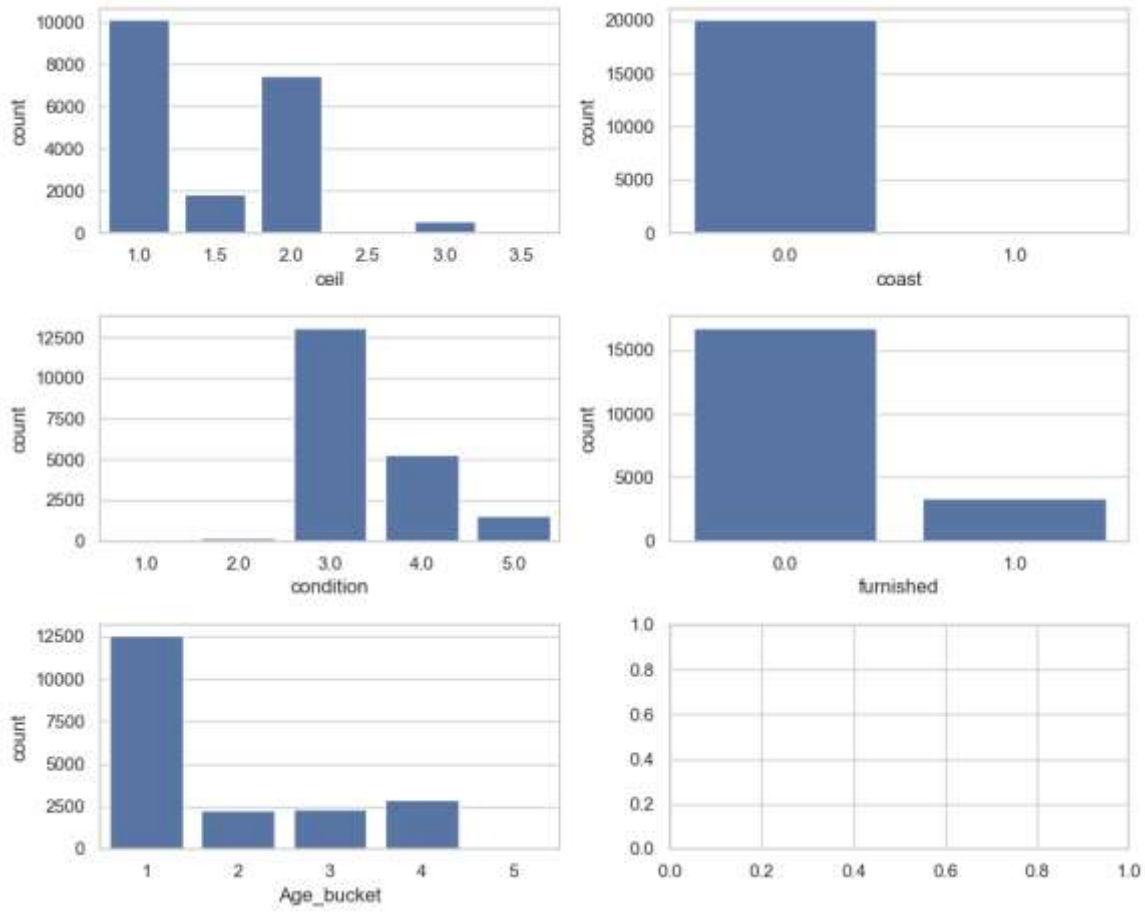
- The peaks of the curves are now centered, and the tails on both sides are balanced.
- The **living_measure**, **ceil_measure**, and **living_measure15** variables show reduced variance and improved scale uniformity.
- This normalization step makes the data more suitable for linear and ensemble regression models.

Inference:

The log transformation significantly improved the distribution of the data, reducing the effect of outliers and enhancing the model's ability to generalize well.

As a result, these normalized features now contribute more effectively to predictive accuracy.

1. Univariate analysis of categorical variables:



Graph 2.5 – Distribution of Categorical Variables

This section analyzes the distribution of categorical features in the dataset. The count plots visualize how data points are distributed across various categories such as ceiling type, proximity to coast, condition, furnishing status, and property age group.

Observations and Insights:

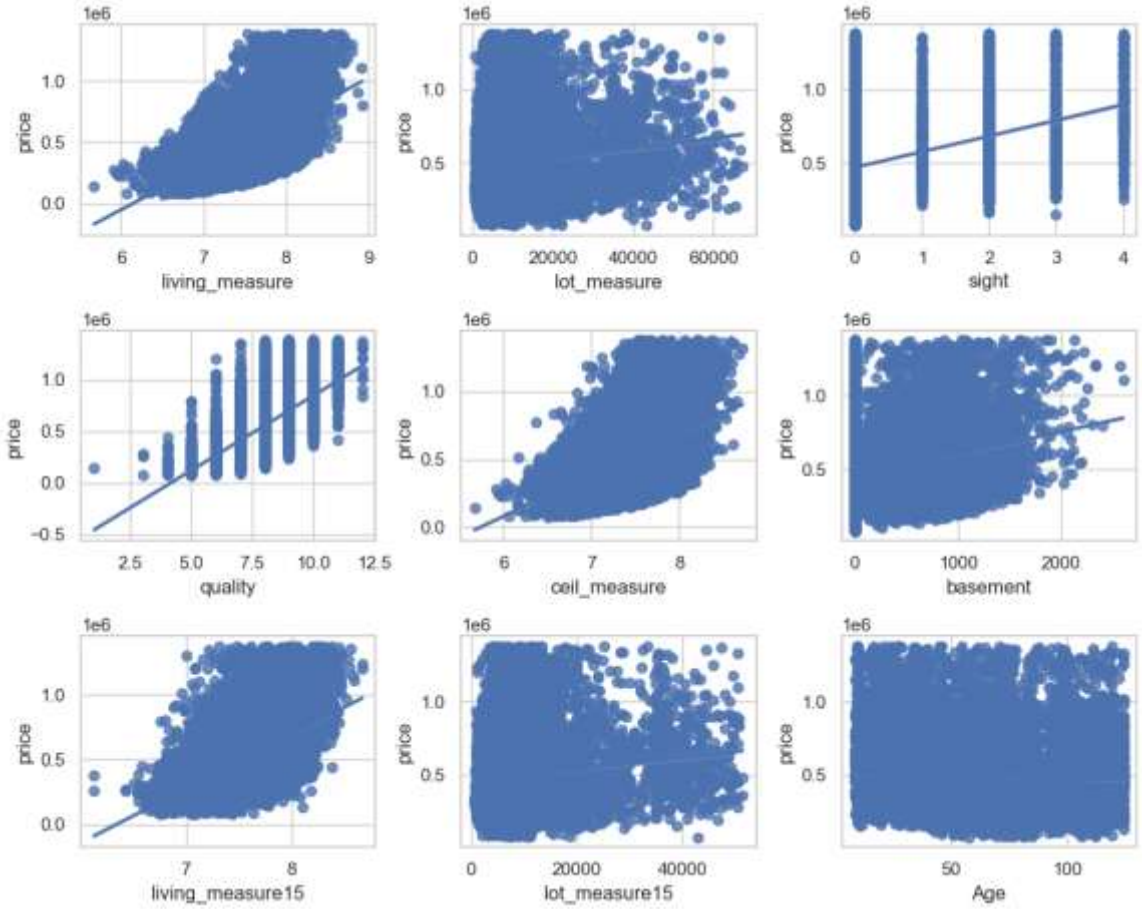
- **Ceil:**
  Most houses have the same type of ceiling, with **category 1** being the most common. Only a small number of properties have higher ceiling categories, suggesting standard construction patterns in the region.
- **Coast:**
  The majority of properties (**over 90%**) are located **away from the coast** (value 0). Only a small percentage are coastal homes (value 1), which could significantly influence property prices due to location desirability.
- **Condition:**
  Most properties are in **average to good condition** (category 3), followed by a smaller number in **above-average condition** (category 4). Very few houses are in poor (1–2) or excellent (5) condition.
  This indicates that the dataset is dominated by mid-range housing quality.
- **Furnished:**
  A large portion of houses are **unfurnished (0)**, with only a smaller group being **furnished (1)**. Furnished houses, though fewer, may reflect higher pricing segments or newer developments.
- **Age_bucket:**
  The **majority of homes fall into the youngest category (1)**, meaning they were built within the last decade. Older homes (categories 2–5) form smaller proportions, indicating newer developments dominate the market.

Inference:

The dataset shows a clear imbalance across several categorical variables. Most houses are non-coastal, unfurnished, and in average condition, reflecting typical middle-market housing trends. These categorical features will be critical in understanding how qualitative factors—such as location, condition, and furnishing—affect the overall house price prediction.

Bivariate analysis:

1.Bivariate analysis of numerical features:



Graph 2.6

Observations and Insights:

- **Living Measure vs. Price:**

  There is a **strong positive correlation** — as the living area increases, the price also rises significantly. Larger homes tend to command higher prices.

- **Lot Measure vs. Price:**

  A **slight positive correlation** is observed. Houses with bigger lot sizes are generally priced higher, though the relationship is weaker compared to living area.

- **Sight vs. Price:**
  The correlation is **weakly positive**. Houses with better views (higher sight scores) show a tendency toward higher prices, indicating that scenic views add marginal value.
- **Quality vs. Price:**
  A **strong linear relationship** is evident — as the quality rating of the house increases, so does the price. Quality appears to be one of the most influential predictors of price.
- **Ceil Measure vs. Price:**
  A **moderate positive correlation** exists, showing that homes with higher ceilings are often more expensive, likely reflecting luxury construction standards.
- **Basement vs. Price:**
  A **weak positive trend** is visible. Houses with basements have slightly higher prices, though the relationship is not very strong, possibly due to varying basement usability.
- **Living Measure 15 vs. Price:**
  A **clear positive trend** indicates that the living area of nearby properties (neighbors within the same block or zip code) also influences a property's price — suggesting a **neighborhood effect**.
- **Lot Measure 15 vs. Price:**
  Shows a **weak correlation**, implying that the average lot size of neighboring houses has minimal effect on price compared to living area.
- **Age vs. Price:**
  A **slight negative correlation** is seen. Newer houses (lower age) tend to be more expensive than older ones, which depreciate over time or require renovation.

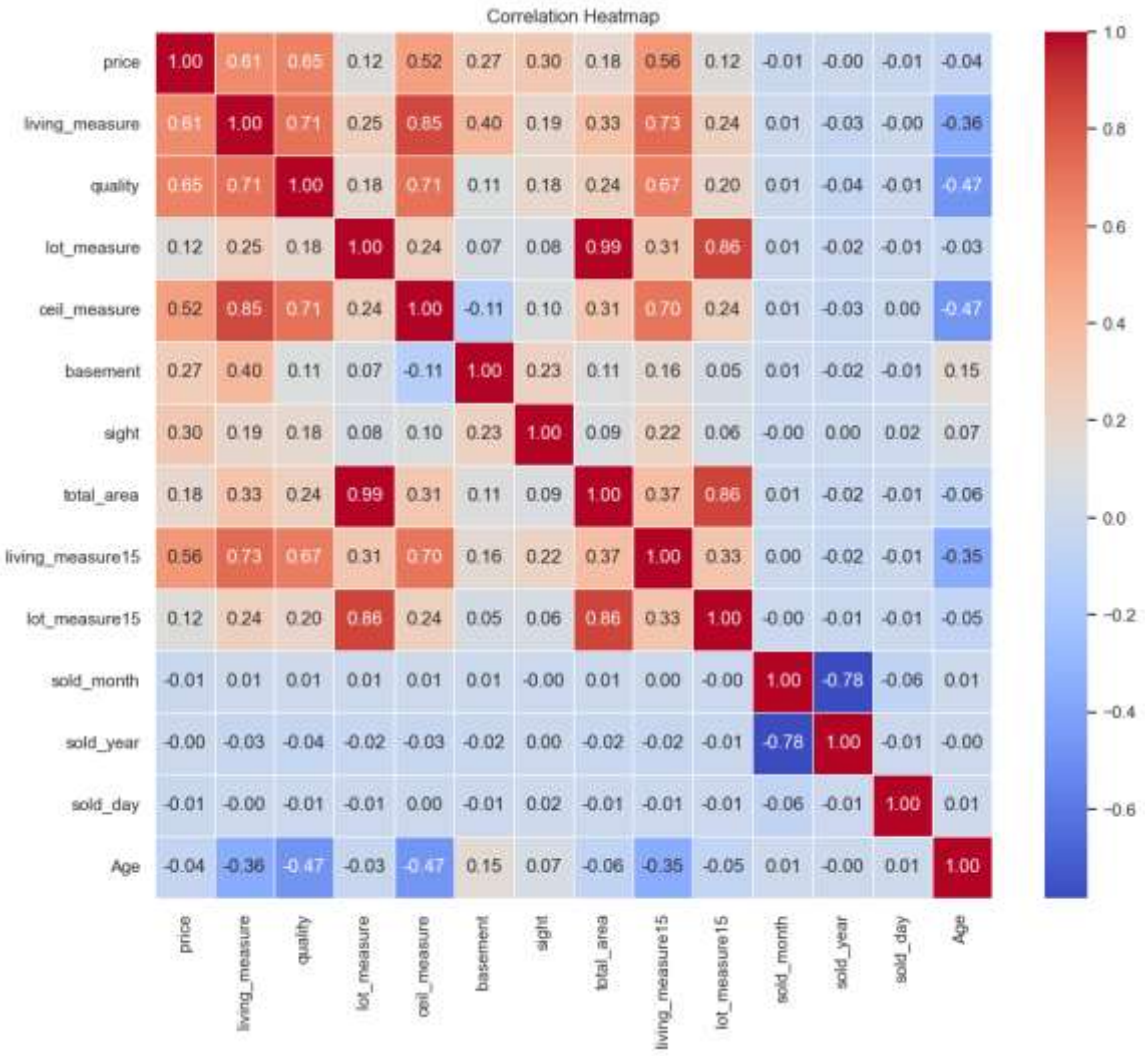
Inference:

From the analysis, **living_measure**, **quality**, and **living_measure15** are the most significant predictors of house price.

While **lot size** and **ceiling height** add some influence, variables like **age** and **basement** show relatively weaker correlations.

These findings will help in feature selection for building the regression and ensemble models during the modeling phase.

2. Heatmap of correlation:



Graph 2.7 – Correlation Heatmap

Observations and Insights:

- **Strong Positive Correlations with Price:**

- **Quality (0.65):**
  Indicates a strong linear relationship — as the quality of the house increases, the price rises significantly.
- **Living Measure (0.61):**
  Houses with larger living areas have higher prices, making it one of the most important predictive features.
- **Living Measure 15 (0.56):**
  Reflects the effect of neighboring houses' living areas, showing that price is also influenced by the size of nearby homes.
- **Ceil Measure (0.52):**
  Higher ceilings are associated with higher prices, though the correlation is moderate.

- **Moderate to Weak Correlations:**
  - **Basement (0.27):** Slightly positive — houses with basements tend to cost more, but the effect is limited.
  - **Sight (0.30):** Indicates that scenic views have a mild positive impact on property prices.
  - **Lot Measure (0.12)** and **Total Area (0.18):** Show weak relationships, suggesting that lot size alone doesn't determine price strongly.
- **Negative Correlations:**
  - **Age (-0.04):** Older houses tend to have slightly lower prices, reflecting depreciation over time.
  - **Sold Month (-0.01)** and **Sold Year (-0.03):** These time-based features show almost no correlation, meaning seasonal or yearly variations are minimal.
- **Multicollinearity Among Independent Variables:**
  - **Total Area and Lot Measure (0.99):** Extremely high correlation, indicating potential redundancy.
  - **Living Measure and Living Measure 15 (0.73):** High correlation, suggesting these two variables may convey similar information.

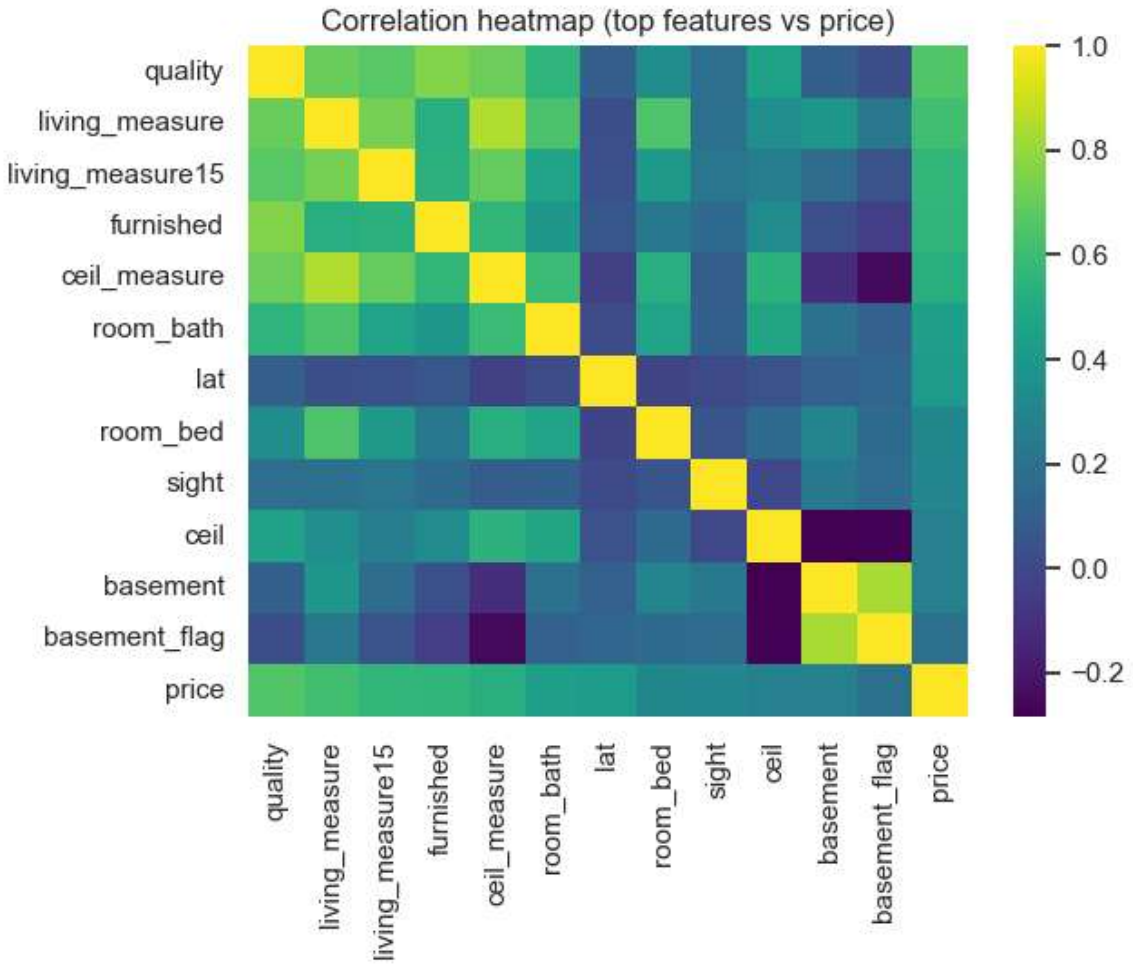
- **Lot Measure and Lot Measure 15 (0.86):** Strongly correlated, which should be handled carefully during feature selection to prevent multicollinearity.

Inference:

The heatmap reveals that **Quality**, **Living Measure**, **Ceil Measure**, and **Living Measure 15** are the most influential predictors of house price.

However, due to high correlation among some variables (like *lot_measure* and *total_area*), feature reduction or regularization techniques such as **Ridge** or **Lasso Regression** may be required to improve model performance and prevent overfitting.

2. Correlation Heatmap (Top Features vs. Price)



Graph 2.8 – Top Correlation Heatmap

Observations and Insights:

- **Highly Correlated Features with Price:**
    - **Quality, Living Measure, and Living Measure15** again show strong positive relationships with price, confirming their predictive significance.
    - **Ceil Measure** and **Furnished** also demonstrate moderate positive associations, indicating that architectural height and furnished status contribute to higher property prices.
- **Moderate to Weak Correlations:**
    - **Basement and Basement Flag:** Houses with basements generally have higher prices, but the relationship remains moderate.
    - **Sight** and **Latitude (lat):** Both show weak but noticeable positive influence — properties with scenic views or located in prime latitude zones tend to be priced higher.
    - **Room_Bed and Room_Bath:** Display moderate correlation with price, suggesting that the number of rooms contributes but is not the sole price determinant.
- **Multicollinearity Observation:**
    - Features like **Living Measure**, **Living Measure15**, and **Quality** are strongly correlated among themselves, suggesting potential overlap in the information they provide.
    - Such multicollinearity needs to be managed during model development using **regularization** (Ridge/Lasso) or **dimensionality reduction** techniques to prevent overfitting.

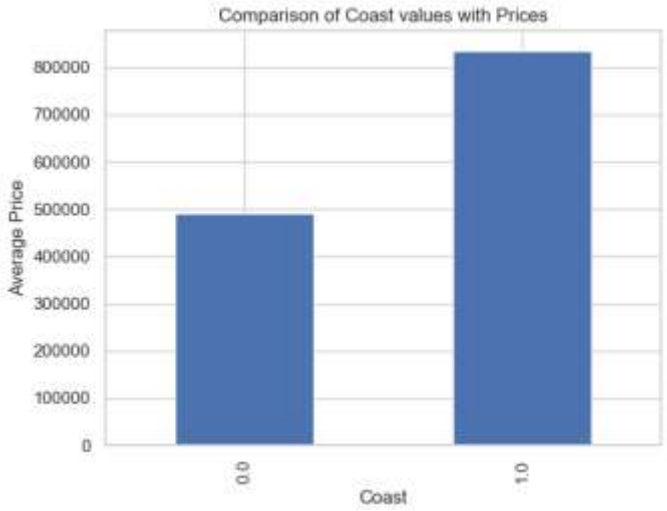
Inference:

This refined heatmap reaffirms that **Quality**, **Living Measure**, and **Ceil Measure** remain the dominant factors influencing house prices.

However, to build a robust and interpretable model, correlated features such as **Living Measure**

and **Living Measure15** must be carefully handled to avoid redundancy and ensure model stability.

3. Bivariate analysis of categorical variables:



Graph 2.9 – Cost Value with Price

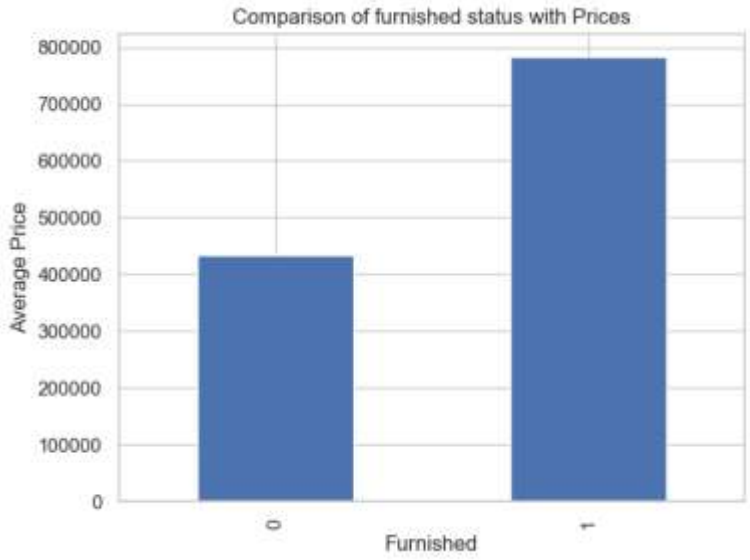
Observations and Insights:

- Properties located in **coastal regions (coast = 1)** have **significantly higher average prices** than those situated inland (**coast = 0**).
- The average price for coastal properties exceeds **₹800,000**, while non-coastal houses average around **₹480,000**.
- This indicates that **location advantage and scenic proximity to water bodies** contribute substantially to property value appreciation.
- Coastal properties are often associated with **premium views, lifestyle appeal, and limited availability**, which drives up demand and price.

Inference:

The coastal variable shows a **strong positive impact** on house prices, making it a valuable predictor in the regression model.

It highlights the **importance of geographic and locational features** in determining real estate value, reinforcing that properties near desirable natural landmarks command higher market prices.



Graph 2.10 – Furnished Status with Price

Observations and Insights:

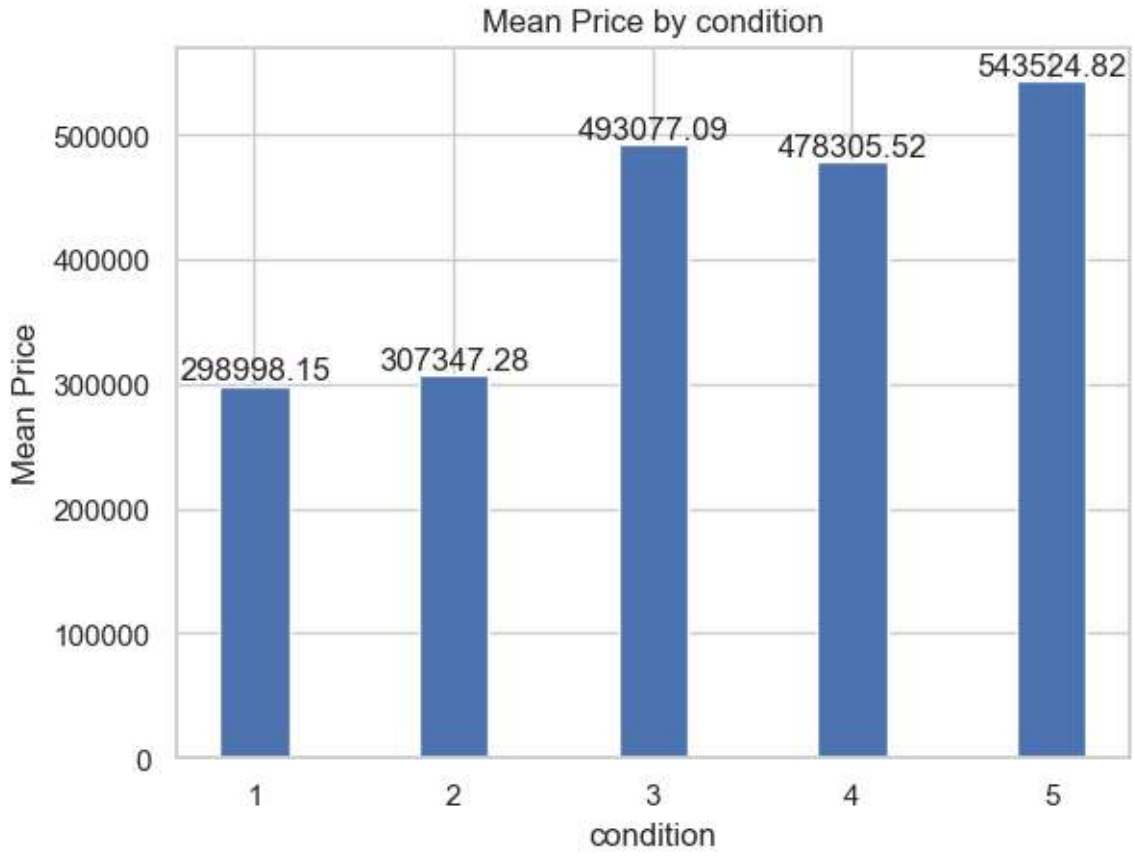
- **Furnished properties (furnished = 1)** have **substantially higher average prices** than **unfurnished properties (furnished = 0)**.
- The average selling price of furnished homes is approximately **₹780,000**, compared to **₹430,000** for unfurnished homes.
- This difference indicates that furnishing adds **perceived value and convenience**, leading to higher buyer willingness to pay.
- A fully furnished home often signifies better interior design, modern amenities, and readiness for occupancy, which increases its market appeal.

Inference:

Furnishing plays a **significant role** in influencing property prices.

This variable demonstrates a **positive correlation** with the target variable (**price**), highlighting

that **interior upgrades and ready-to-move conditions** are key value drivers in the housing market.



Graph 2.11 - Mean Price by House Condition

Observations and Insights:

- Properties in **better condition (condition = 5)** have the **highest average price**, around **₹540,000**, whereas houses with **poor condition (condition = 1)** have significantly lower prices, approximately **₹300,000**.
- A noticeable **upward trend** can be observed — as the **condition rating improves**, the **mean selling price increases**.
- Houses with **moderate condition (condition = 3)** are the most common and show an average price near **₹490,000**, indicating a mid-range valuation in the housing market.

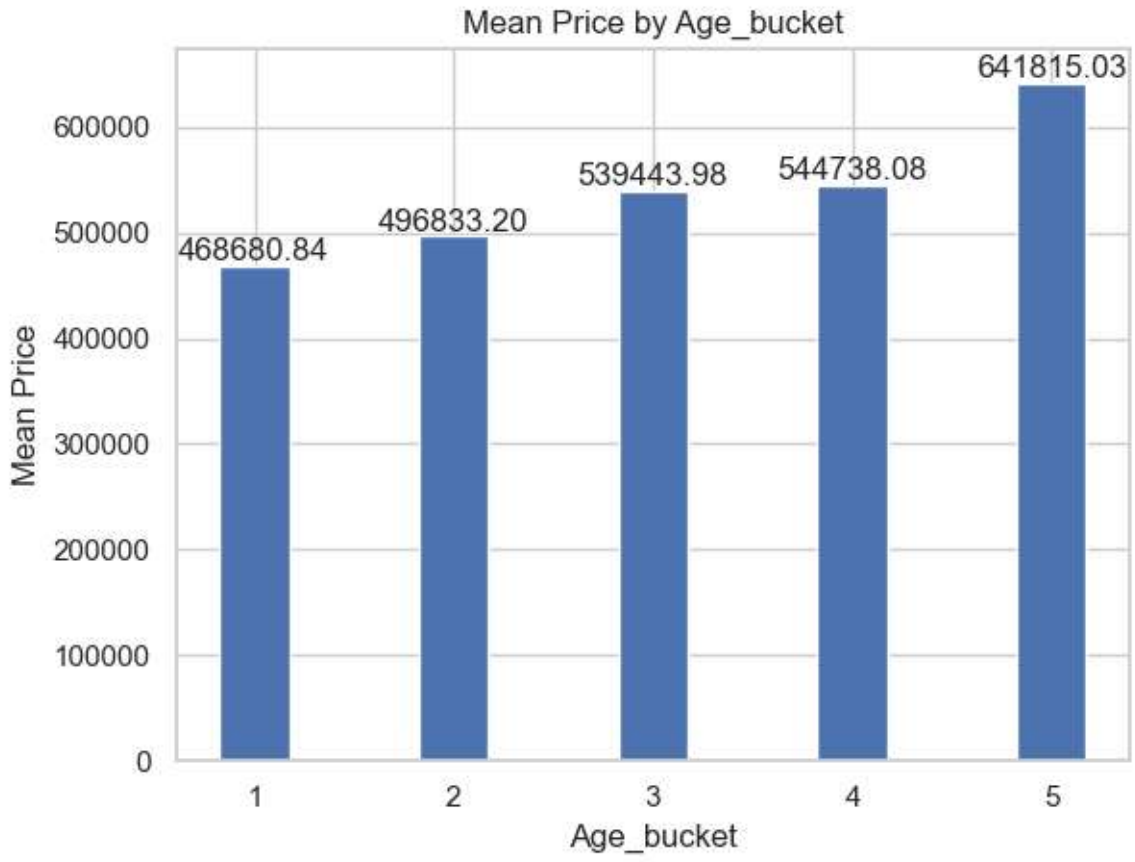
- The slight dip between conditions **3 and 4** might suggest variability due to other interacting factors such as location or interior features.

Inference:

The **condition of the house** plays an essential role in determining its **market value**.

Better-maintained houses command higher prices, reflecting buyers' preference for properties requiring **less renovation and upkeep**.

Thus, this feature contributes **positively and consistently** to the house price prediction model.



Graph 2.12 – Mean Price by Age Bucket

Observations and Insights:

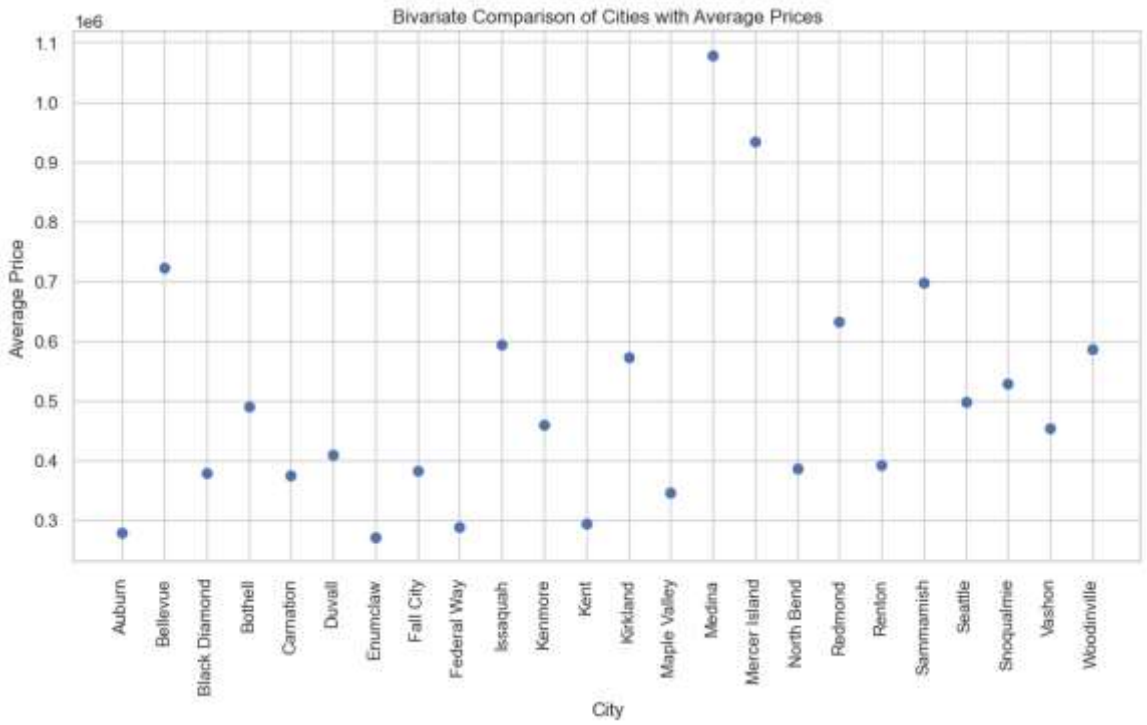
- Properties in the **oldest age group (Age_bucket = 5)** have the **highest mean price**, averaging around **₹640,000**.
- Conversely, **newer houses (Age_bucket = 1)** tend to have **lower average prices**, around **₹470,000**.
- A **steady increase** in price is observed as the age bucket rises, suggesting that **older houses may be located in prime or established areas** with higher land value.
- This trend might also indicate that **heritage or well-built older homes** continue to command strong market demand, depending on their location and build quality.

Inference:

The **Age_bucket** variable demonstrates a **positive correlation with price**, contrary to the usual expectation that newer houses are more expensive.

This implies that in the **King County region**, **location and neighborhood maturity** likely play a stronger role in determining property value than the physical age of the building.

Hence, **age-related features** can still be a **significant predictor** in the house price prediction model when combined with geographical attributes.



Graph 2.13 - Comparison of Cities with Average Prices

Observations and Insights:

- Cities such as **Medina** and **Mercer Island** exhibit **significantly higher average house prices**, exceeding **₹1,000,000**, indicating these are **premium or luxury residential zones**.
- On the other hand, cities like **Auburn**, **Enumclaw**, and **Federal Way** show **lower average prices**, generally below **₹300,000**, representing **affordable housing markets**.
- Mid-tier cities including **Bellevue**, **Issaquah**, and **Redmond** have moderately high prices ranging between **₹600,000 – ₹800,000**, reflecting balanced urban development and accessibility to key areas.
- Overall, the plot demonstrates a **strong geographic influence** on property prices — proximity to major employment hubs, quality of amenities, and coastal access play a major role in price variation.

Inference:

City-level analysis reveals that location is one of the most influential determinants of property value.

Urban centers and coastal cities with superior infrastructure, schools, and economic opportunities command higher property prices.

Thus, incorporating geospatial data (city or zipcode) into the model significantly improves prediction accuracy and helps capture regional market dynamics effectively.

Business Insights from Exploratory Data Analysis

1.1 Data Skewness and Its Business Implications

Based on the Exploratory Data Analysis (EDA), it is observed that the **target variable — house price — exhibits a right-skewed distribution**, as shown in *Graph 1*.

This means that most houses in the dataset are priced around or below the median value, while a smaller proportion of houses are extremely expensive, resulting in a **long right tail** in the distribution.

Business Implications of Right Skewness

Challenges:

- **Model Performance:**
  Many statistical and machine learning algorithms assume a normally distributed target variable.
  A right-skewed target (like price) can lead to biased model coefficients or less accurate predictions if not corrected (e.g., using logarithmic transformation).
- **Overestimation Risk:**
  High-priced properties can disproportionately influence the model, causing it to **overpredict** prices for average homes, which may mislead buyers and sellers.

Opportunities:

- **Luxury Segment Analysis:**

  The long right tail represents the **luxury real estate market** — a niche but high-margin segment.

  By analyzing these outliers, businesses can identify **unique characteristics** (e.g., premium locations, architectural features) that contribute to luxury property pricing.
- **Feature Engineering Opportunities:**

  Understanding which factors drive extreme prices can guide the creation of new predictive features, such as **proximity to amenities**, **view quality**, or **property upgrades**, improving overall model performance.

1.2 Other Key Business Insights

- **Feature Correlations:**

  Strong correlations were observed between price and certain features such as **living_measure**, **quality**, **location (latitude/zipcode)**, **ceil_measure**, and **living_measure15** (*refer Graph 1*).

  These variables act as **key price drivers** in the King County housing market.
- **Price Prediction Trends:**

  Houses with **larger living area, better quality ratings, and desirable locations** consistently command higher prices.

  Conversely, lower values in these features are associated with **reduced property prices**, confirming that **size, quality, and location** remain the most influential factors in real estate valuation.

Model Building and Interpretation

A **Machine Learning model** represents the mathematical relationship learned from training data to predict outcomes on unseen data.

In this project, several supervised learning algorithms were developed to predict house prices based on multiple features derived from the King County dataset.

1.1 Model Building

Various regression models such as **Linear Regression**, **Ridge Regression**, **Decision Tree Regressor**, **XGBoost**, **LightGBM**, and **CatBoost Regressor** were built.

Each model was trained using the pre-processed dataset, applying techniques like scaling, encoding, and cross-validation to ensure robustness.

1.2 Model Testing

The models were evaluated on a test dataset using key performance metrics including:

- **R^2 Score (Coefficient of Determination):** Measures how well the model explains the variance in house prices.
- **Adjusted R^2 Score:** Accounts for the number of features used, ensuring the model is not overfitted.
- **RMSE (Root Mean Squared Error):** Evaluates the model's prediction error in the same units as the target variable.

1.3 Model Interpretation

Among all tested models, **CatBoost** demonstrated the **highest R^2 and Adjusted R^2 scores** with the **lowest RMSE**, indicating superior predictive accuracy.

Feature importance analysis further highlighted that **latitude, living area, quality, and age** are the most significant predictors of property price.

Building various models:

Let's dive deeper into the models that have been employed and implemented in this project.

1. Multiple Linear Regression model:

```
▾ LinearRegression
LinearRegression()
```

- Multiple Linear Regression is a statistical technique used to predict the value of a dependent variable based on two or more independent variables.
- It assumes a linear relationship between the dependent variable (price) and the independent variables (features).
- This model provides interpretability and serves as a baseline to compare more complex models.

2. Decision tree regression model:

```
▾      DecisionTreeRegressor
DecisionTreeRegressor(max_depth=4)
```

- A Decision Tree Regressor works by splitting the dataset into smaller subsets based on feature values, creating a tree-like structure of decisions.
- In this project, it helps capture **non-linear relationships** between the features and the target variable.
- However, if the tree depth (e.g., max_depth = 4) is set too high, the model can **overfit**, learning noise and fine details from the training data rather than general patterns.

3. RIDGE REGRESSION:

```
▾ Ridge
Ridge()
```

- Ridge Regression is a linear model that incorporates **L2 regularization** to handle **multicollinearity** among predictors.
- When multicollinearity exists, the coefficients estimated by simple linear regression can become unstable and produce high variance.
- Ridge regression addresses this by adding a penalty term to the cost function, shrinking coefficient values and improving model generalization.

4. X Gradient Boosting model:

```
▸ XGBRegressor
```

- XGBoost (Extreme Gradient Boosting) is an ensemble machine learning method that builds multiple decision trees sequentially, where each new tree corrects the errors made by the previous ones.
- This technique is based on the **boosting** approach, where weak learners are combined to form a strong predictive model.
- XGBoost is known for its **speed, accuracy, and regularization features**, making it one of the most powerful algorithms for structured data.

5. Light GBM model:

```
▾ LGBMRegressor
LGBMRegressor()
```

- LightGBM (Light Gradient Boosting Machine) is another gradient boosting framework designed for **high efficiency and scalability**.
- It uses two novel techniques to improve performance:

- **Gradient-based One-Side Sampling (GOSS):** Retains instances with large gradients to focus on harder-to-predict samples.
- **Exclusive Feature Bundling (EFB):** Combines mutually exclusive features to reduce dimensionality.
- LightGBM provides faster training and lower memory usage compared to other gradient boosting frameworks.

6. Cat boost regression model:

- CatBoost (Categorical Boosting) is a gradient boosting algorithm specifically designed to handle **categorical features efficiently**.
- It builds an ensemble of decision trees sequentially, where each tree learns from the mistakes of the previous trees, gradually reducing errors.
- The algorithm applies **ordered boosting** to prevent target leakage and improve model stability.
- CatBoost is robust, requires minimal data preprocessing, and often performs well out-of-the-box on tabular datasets.

Testing models:

To evaluate model performance and ensure generalization, several supervised regression algorithms were tested on the dataset.

Each model was assessed using **R^2 score**, **Adjusted R^2 score**, **Root Mean Squared Error (RMSE)**, and **Cross-Validation (CV) scores**.

These metrics collectively provide insights into how well each model explains the variance in the target variable and predicts unseen data.

2.1 Linear regression:

Metric	**Value**
R^2 Test Score	**0.84**

Metric	Value
Adjusted R^2 Score	**0.84**
RMSE	**95,068.17**
Cross-Validation Scores	[0.8370, 0.8326, 0.8379, 0.8281, 0.8370]
Average CV Score	**0.83**

Observation:

Linear Regression performs fairly well, explaining about **84% of the variance** in house prices. It serves as a reliable baseline model with balanced bias-variance trade-off.

2.2 Decision Tree regression:

Metric	Value
R^2 Test Score	**0.68**
Adjusted R^2 Score	**0.67**
RMSE	**133,956.71**
Cross-Validation Scores	[0.6931, 0.6798, 0.6588, 0.6561, 0.6774]
Average CV Score	**0.67**

Observation:

The Decision Tree model tends to **overfit the training data** when depth is not controlled. Although it captures non-linear relationships, its predictive performance is weaker compared to ensemble methods.

2.3 Ridge Regression:

Metric	Value
R^2 Test Score	**0.84**
Adjusted R^2 Score	**0.83**
RMSE	**95,264.14**

Metric	Value
Cross-Validation Scores	[0.8366, 0.8322, 0.8371, 0.8309, 0.8365]
Average CV Score	**0.83**

Observation:

Ridge Regression performs similarly to Linear Regression but with **better coefficient stability** due to regularization. It handles multicollinearity effectively.

2.4 XGBoost Regressor

Metric	Value
R^2 Test Score	**0.88**
Adjusted R^2 Score	**0.88**
RMSE	**81,671.63**
Cross-Validation Scores	[0.8854, 0.8735, 0.8730, 0.8678, 0.8770]
Average CV Score	**0.88**

Observation:

XGBoost shows strong performance with high accuracy and generalization. The boosting approach effectively reduces bias and variance.

2.5 Light GBM Regressor

Metric	Value
R^2 Test Score	**0.89**
Adjusted R^2 Score	**0.88**
RMSE	**79,713.01**
Cross-Validation Scores	[0.8919, 0.8817, 0.8830, 0.8760, 0.8833]
Average CV Score	**0.88**

Observation:

LightGBM outperforms previous models with **higher R^2** and lower RMSE. It demonstrates better efficiency and faster convergence due to optimized sampling and feature bundling.

2.6 Cat Boost Regressor

Metric	**Value**
R^2 Test Score	**0.90**
Adjusted R^2 Score	**0.90**
RMSE	**75,374.57**
Cross-Validation Scores	[0.9028, 0.8899, 0.8948, 0.8882, 0.8911]
Average CV Score	**0.89**

Observation:

CatBoost achieved the **best overall performance** with the highest R^2 and lowest RMSE. Its ability to handle categorical features and prevent overfitting makes it the **most robust and accurate model** in this study.

Summary of Model Comparison

Model	**R^2**	**Adjusted R^2**	**RMSE**	**Avg. CV Score**
Linear Regression	0.84	0.84	95,068	0.83
Decision Tree	0.68	0.67	133,957	0.67
Ridge Regression	0.84	0.83	95,264	0.83
XGBoost	0.88	0.88	81,672	0.88
LightGBM	0.89	0.88	79,713	0.88
CatBoost	**0.90**	**0.90**	**75,375**	**0.89**

S

Overall Insight:

CatBoost emerged as the most effective model, followed closely by LightGBM and XGBoost. Linear and Ridge regression performed adequately as baseline models, while Decision Tree showed limited generalization capability.

Interpretation of Models

Model	Train R^2	Test R^2	Train RMSE	Test RMSE	Train MAE	Test MAE
CatBoost	0.9540	0.9001	50,433.87	75,374.55	36,876.92	50,752.74
LightGBM	0.9231	0.8892	65,233.66	79,403.64	45,974.92	53,921.83
XGBoost	0.9548	0.8826	49,993.09	81,714.21	35,815.30	55,227.10
Linear Regression	0.8380	0.8411	94,661.41	95,067.94	67,412.94	68,231.57
Ridge Regression	0.8380	0.8410	94,683.66	95,127.31	67,410.21	68,256.11
Decision Tree	0.6815	0.6846	132,735.80	133,956.70	94,838.54	95,966.93

Table 6.1: Performance Summary of Regression Models

Interpretation of Results

- Each model was evaluated using **R^2**, **Adjusted R^2**, **Root Mean Squared Error (RMSE)**, and **Mean Absolute Error (MAE)** across both the training and testing datasets to assess generalization capability.
- Additionally, **K-Fold Cross-Validation** was applied to verify that the models perform consistently across different data subsets.
- Among the evaluated models, ensemble learners (CatBoost, LightGBM, and XGBoost) significantly outperformed linear models, demonstrating their ability to capture non-linear and complex relationships in the housing dataset.

Model Tuning and Business Implications

Ensemble Modelling

- **Ensemble modelling** combines predictions from multiple base learners to build a more robust and accurate final model.
- In this project, a **Stacking Regressor** was constructed with:
    - **Base Models:** CatBoost, LightGBM, and Linear Regression
    - **Meta Model:** Ridge Regression

Performance of Stacked Model:

Metric	Value
R^2 (Test)	0.8686
Adjusted R^2 (Test)	0.8646
RMSE	86,456.48

Interpretation:

- The stacked model achieved a test R^2 of **0.8686**, which indicates that approximately **87% of the variance** in house prices can be explained by the model.
- Although its performance is slightly below that of the CatBoost model, it demonstrates **strong generalization** and **model stability**, validating that combining multiple learners yields reliable predictions.
- The results confirm that ensemble learning effectively reduces bias and variance, balancing interpretability and predictive power.

Selection of the Optimum Model

- Based on all evaluation metrics, **CatBoost Regressor** remains the **best-performing model**.
- It achieved the **highest R^2 (0.90)** and **lowest RMSE (≈ 75 k)**, outperforming all other individual and ensemble models.
- CatBoost's built-in handling of categorical features, efficient gradient boosting, and strong regularization enable superior predictive accuracy without overfitting.
- Thus, **CatBoost** is selected as the **final model** for house-price prediction, providing both **precision and reliability** for business decision-making.

Business Implication

The optimized CatBoost model empowers real-estate analysts and investors to:

- **Accurately estimate property prices**, reducing pricing uncertainty.
- **Identify key price drivers** such as living area, quality, and location.
- **Support data-driven investment strategies** and better property valuation decisions.

Appendix

Attachments:

House_price_prediction.ipynb (python code file)

link -> https://drive.google.com/file/d/1-cU6XZ6f-LiYM5QznFiYVEkaPGM4H9HY/view?usp=sharing

innerCity.csv -> https://docs.google.com/spreadsheets/d/1hBoFFirY1D0zH_4mv_-ZdwGxOT1JvCp7/edit?usp=sharing&ouid=115577863227913964973&rtpof=true&sd=true

Precision-Property-Dataset

https://drive.google.com/file/d/1_3ak2XJaN1QQDoo20WCu67Xwjr8tpcNr/view?usp=sharing