USA_Real_Estate Price Prediction
Project Documentation

This dataset contains Real Estate listings in the US broken by State and zip code.

Inspiration:

- Can we predict housing prices based on the features?
- How are housing price and location attributes correlated?
- What is the overall picture of the USA housing prices w.r.t. locations?
- Do house attributes (bedroom, bathroom count) strongly correlate with the price? Are there any hidden patterns?

1. LOADIND & SAMPLING DATA

- The dataset was loaded using ***pandas*** for analysis.
- The original dataset is large-scale; therefore, sampling was used to enable efficient iteration without compromising statistical validity.
- Initially, a sample size of 50K was used. To assess representativeness, different sample sizes were tested. Increasing the sample size did not materially change the observed patterns, indicating that the sampled data adequately represents the underlying distribution.

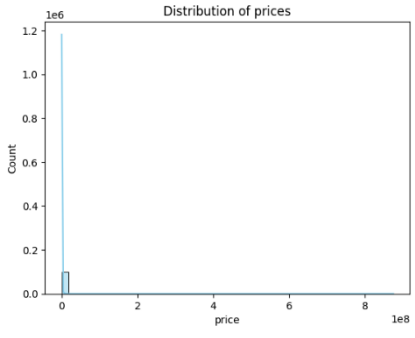
2. UNDERSTANDING DATA

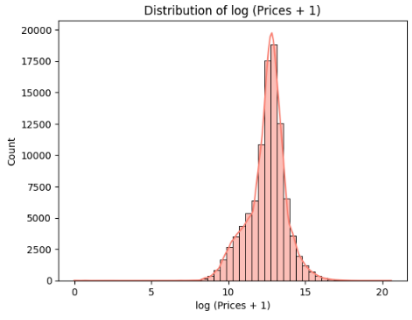
In this step, the dataset was examined to understand its structure, feature types, summary statistics, and overall composition using ***NUMPY & PANDA*** libraries. Missing values and their percentages were analyzed across all columns, revealing that some features contain substantial missing data. This assessment helped identify potential data quality issues and informed decisions for exploratory data analysis and subsequent preprocessing steps.

3. Exploratory Data Analysis (EDA)

Exploratory data analysis was conducted using ***SEABORN & MATPLOTLIB*** libraries to understand the distribution of the target variable, examine relationships between features and price, and identify patterns, trends, and outliers that may influence modeling decisions.

Checking distributing of target column (price) and Log Price:





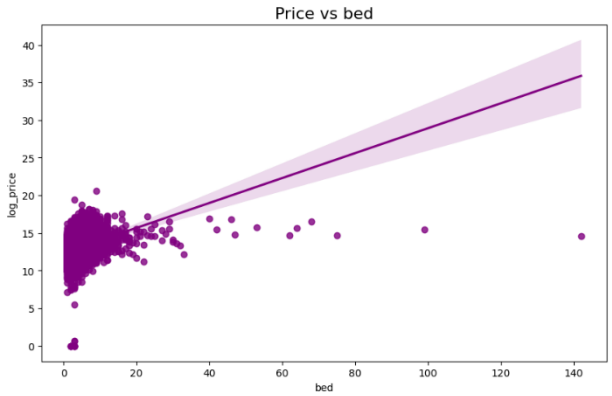
➢ **Analysis:**

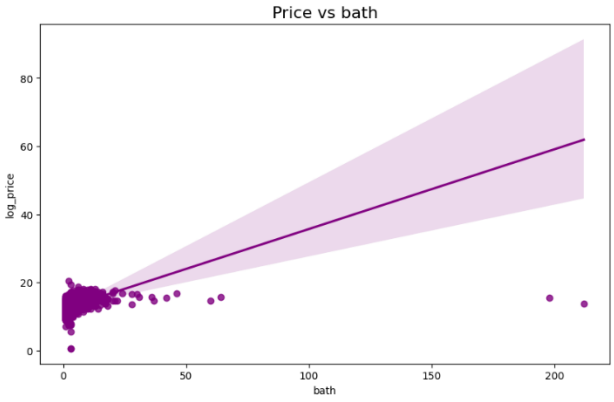
The original price distribution was heavily right-skewed. Applying a log transformation reduced skewness and produced a more symmetric distribution, with most observations concentrated between 10 and 15 on the log scale.

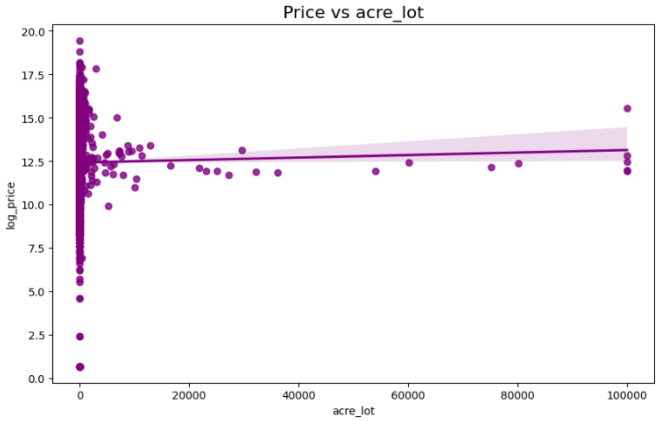
Relationship between target column and numeric features:

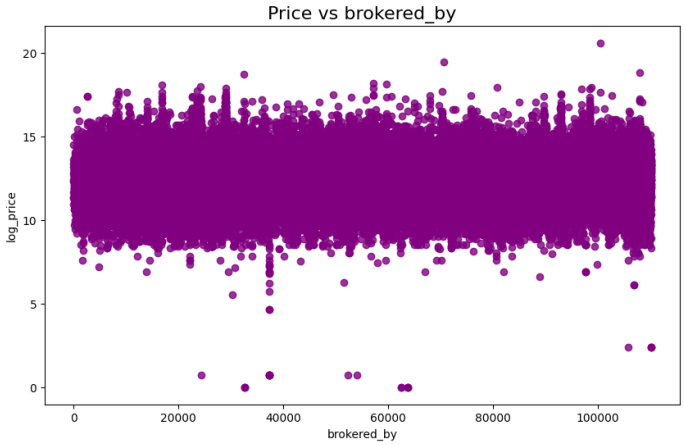
```
#list of all numeric features
numeric_features = ['house_size', 'bed', 'bath', 'acre_lot', 'brokered_by']
```

REGPLOT:

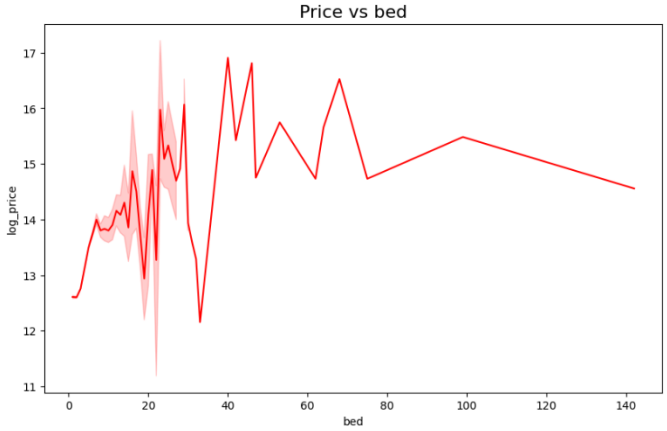


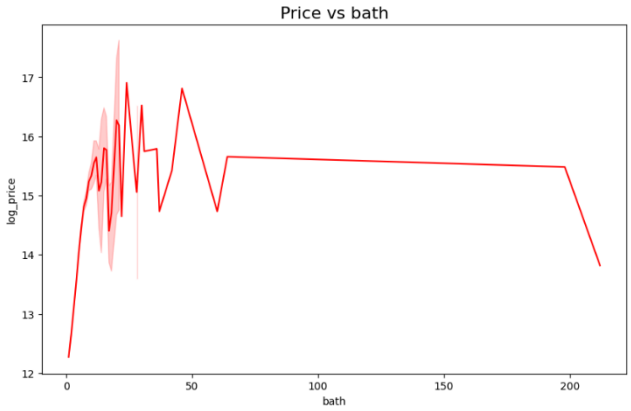


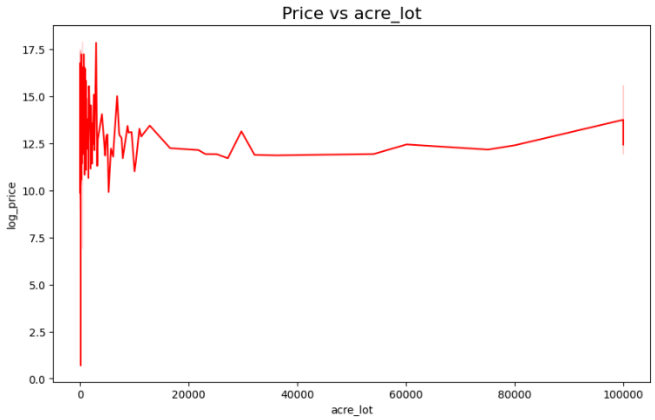




LINEPLOT:







- ***<u>Analysis</u>:***

  house size vs log price | House size shows a clear upward trend with log price

  bed vs log price | Bedrooms show a mild positive effect on price

  bath vs log price | Bathrooms show a stronger influence on price

  acre lot vs log price | Acre lot shows minimal impact within observed range

  brokered_by vs log price |Brokered_by shows no meaningful linear relationship with price

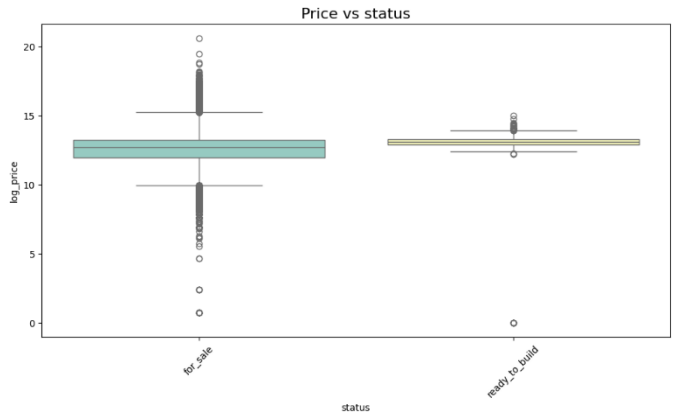
- ***<u>General Observation</u>:***

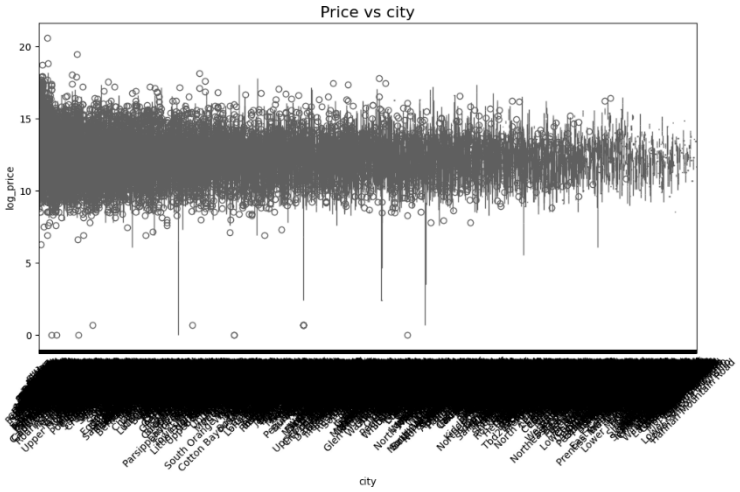
  Most numeric features are concentrated in lower ranges with a few outliers. House size and number of bathrooms show a clear upward trend with log price, bedrooms show a mild effect, while acre lot and brokered_by show minimal influence.

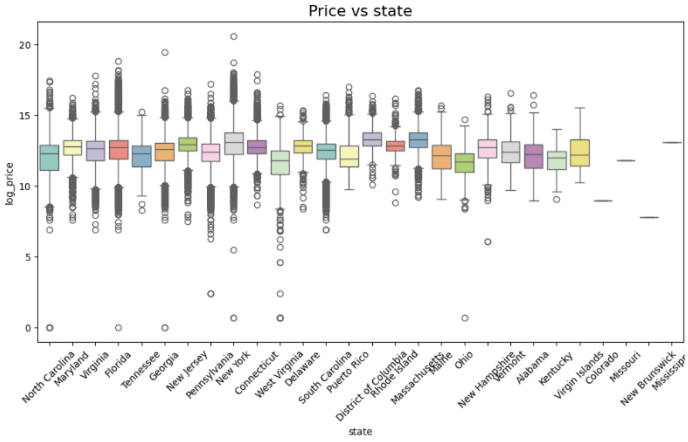
❖ Categorical features VS Price:

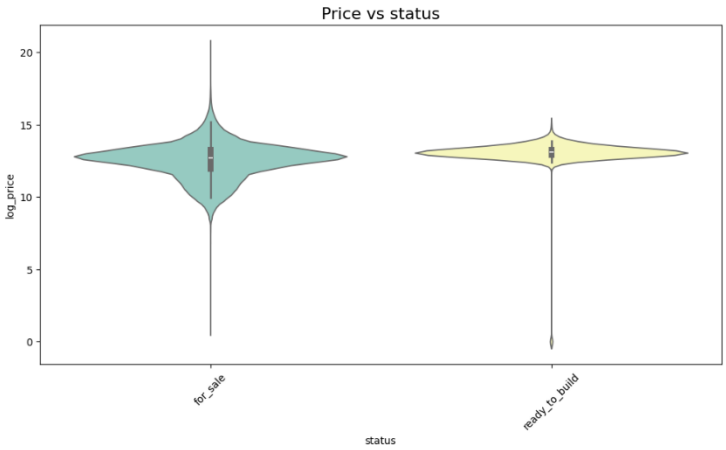
```
#List of categorical columns
cat_features = ["status", "city", "state"]
```

BOXPLOT









➢ Analysis:

Status vs log price:

The box plot and violin plot show that “For Sale” properties have a median logprice between 10 and 15 with a few high and low outliers, while “Ready to Build” properties have a slightly higher median around 13 to 15 with minimal outliers. This indicates that the property status has a modest impact on price.

City vs log price:

The box plot of price across cities is not clearly visible due to the very high number of unique cities. However, the range of log prices across cities is approximately 9 to 17, suggesting that city-level variation in price is relatively small within this dataset.

***state vs log price*:**

The box plot of price by state shows median log prices in a similar range (roughly 8.5–15.5) across states such as North Carolina, Maryland, and Virginia, indicating that state-level differences exist but are not extremely large in this sample.

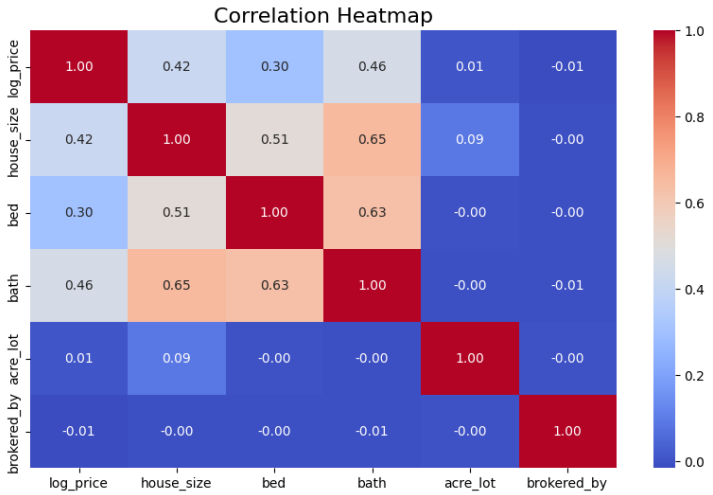
Violin Plots Note:

Violin plots for city and state could not be rendered due to high cardinality, but the price distributions inferred from the box plots are consistent with the expected spread observed in the dataset.

- ***GENERAL OBSERVATION:***

Overall, categorical features show modest variation in log price, with status having a slight effect, while city and state do not exhibit large differences in the majority of listings.

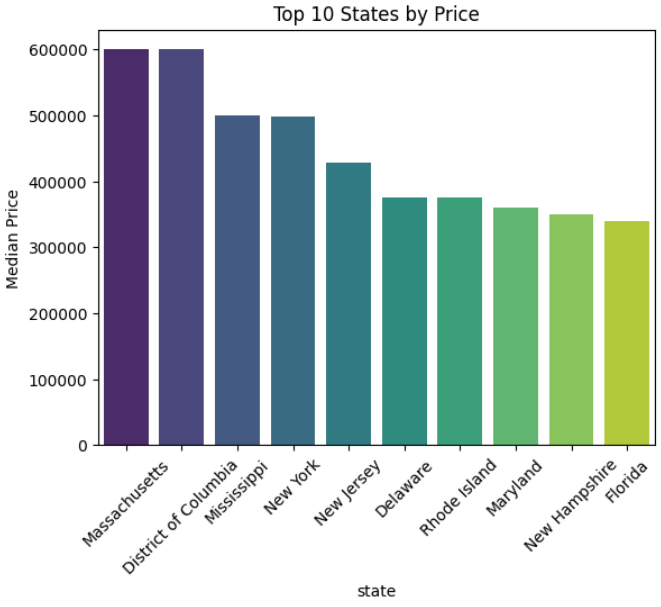
Heatmap correlation

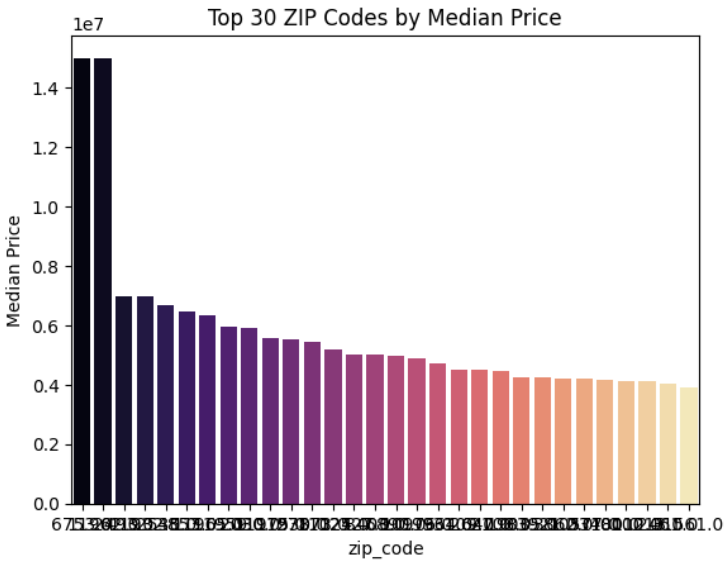


- **Finding:**

Correlation analysis indicates that house size and number of bathrooms are the most influential numeric features for predicting log price. Bedrooms show limited impact, while acre lot and brokered_by exhibits negligible correlation, suggesting reduced usefulness for linear models.

Geographical price variation by state & ZIP Code





➢ FINDINGS

geographical price variation by state

Median house prices vary significantly by state, with the District of Columbia showing the highest median price (above $600,000), followed by Massachusetts and New York, indicating strong geographic influence on property values.

geographical price variation by ZIP

A small number of ZIP codes exhibit extremely high median prices (above 2 million), while most ZIP codes in the top 30 cluster around $500,000, highlighting localized premium micro-markets.

geographical price variation by city

Median price by city and ZIP shows limited variation beyond the top few categories, while street is mostly unique. Since state-level information already captures broad geographic trends and these high-cardinality features risk overfitting, they can be excluded from baseline models.

5. PREPROCESSING

Based on insights from exploratory data analysis, the following preprocessing steps were applied to prepare the data for machine learning models.

➢ Observations and actions

- City, ZIP, street are high-cardinality columns in our dataset. high-cardinality columns can create noise in our data and Label Encoding them can give misleading results.
- Prev_sold_date is not needed but the history of ever sold or never sold could be useful so that column is encoded as 1/0.
- city, street, ZIP code columns can be used for deep analysis but encoding them using LabelEncoder/OneHotEncoder would not be useful due to high cardinality. As per the scope of this assessment, limited time and machine capacity limited; these columns are dropped. Moreover, these columns also didn't show any strong pattern with our target (price) in EDA. However, in case of compulsion by client; we can use HASHING or TARGET ENCODING to include these columns in our Model Training

6. EXPLORATORY DATA ANALYSIS AFTER PREPROCESSING

- After handling missing values, encoding categorical variables, and transforming the target variable, the dataset is now clean and ready for modeling.
- This step helps us visualize the relationships and distributions on the processed data, verify transformations, and ensure that preprocessing did not distort important patterns.
- Graphs and plots were recreated to confirm data integrity and feature-target relationships.

➢ Findings from EDA on Processed Data

- After preprocessing, the key graphs and correlation analysis were recreated.
- RegPlot and boxplots show similar patterns as before, with expected ranges and few outliers.
- Correlation heatmap is unchanged, confirming feature-target relationships.

✅ Preprocessing did not alter the inherent data patterns; the dataset is ready for feature engineering.

7. FEATURE ENGINEERING

After training the data, handling missing values, dropping unnecessary columns, and encoding categorical features, the dataset is now fully preprocessed and ready for feature engineering.
In this section, new features and transformation was done on existing ones to improve model performance.

```
#Features
x = df_sample[['brokered_by', 'bed', 'bath', 'acre_lot', 'house_size', 'ever_sold', 'status_encoded', 'state_encoded']]
#target
y = df_sample['log_price']
```

➢ Feature Engineering Summary

- New features were introduced to improve training of model. Converting lot size from acres to square feet provided a more interpretable scale, while the bedroom–bathroom interaction feature captured combined housing utility.
- Data splited into train and test using ***train_test_split***. 80% of data will be used for training and 20% for testing.
- Numeric features were standardized after the train–test split to avoid data leakage and to support efficient model training.

8. MODEL FITTING, TRAINING & TESTING

Based on insights from exploratory data analysis, both linear and non-linear relationships were observed between the features and the log-transformed target variable (LogPrice).
In this section, multiple linear and tree-based regression models were trained and evaluated to predict property prices.

For linear models, pipelines were used to integrate feature scaling and model training into a single workflow, ensuring consistent preprocessing and preventing data leakage during cross-validation. Tree-based models were trained separately without pipelines, as they are not sensitive to feature scaling.

➢ **FINDINGS:**

The results show that tree-based models outperform linear models on this dataset.
The Random Forest model achieved the highest **R^2 score (≈0.63)** along with the lowest MAE and RMSE, indicating its strong ability to capture non-linear relationships in housing prices.

In contrast, linear models such as Linear Regression, Ridge, Lasso, and ElasticNet demonstrated comparatively weaker performance. This confirms the presence of limited linear relationships, as previously observed during exploratory data analysis (EDA), and highlights the suitability of ensemble-based methods for this problem.

Evaluation Metric Note: Removal of MAPE

Mean Absolute Percentage Error (MAPE) was initially considered to evaluate the average percentage deviation between actual and predicted prices. However, because the target variable was log-transformed, some values after back-transformation approached zero, causing MAPE to produce infinite or misleading results due to division by zero.

As a result, MAPE was excluded from the final model comparison. Greater emphasis was placed on RMSE (log scale), RMSE in the original dollar scale, and R^2 score, as these metrics provide a more reliable and interpretable assessment of model performance for this dataset.

9. IMPROVING MODEL ACCURACY

- Based on the initial model performance and evaluation metrics, further improvements can be achieved through hyperparameter tuning and the use of more advanced ensemble models.
- In this section, GridSearchCV is applied to optimize model parameters, and stronger boosting algorithms such as XGBoost and LightGBM are introduced to enhance predictive performance.

10. FINAL MODEL FINDINGS

After applying hyperparameter tuning and advanced ensemble models, the predictive performance of different models was evaluated using R^2 and RMSE metrics (both in log scale and actual price in USD).

➢ Observations:

- Among the tested models, LightGBM and XGBoost performed slightly better than Random Forest in terms of R^2 and RMSE.
- All three models have similar performance in terms of RMSE in actual price scale (≈2.16 million USD), indicating stable predictions.
- The final model can be chosen based on the slight edge in accuracy, computational efficiency, or interpretability depending on the use case.

These results demonstrate that ensemble learning and hyperparameter tuning can meaningfully improve predictive performance over baseline models.

11. ROBUSTNESS AND DEPLOYMENT READINESS

The robustness of the machine learning solution was assessed by analyzing price behavior across multiple geographic sub-markets, including different states, to ensure consistent model performance. Log transformation was applied to the target variable to address skewness and improve model stability, which is a common and justified assumption for real estate price modeling. State-level analysis also helped ensure pricing fairness by confirming that predictions were not biased toward any specific region.

From a deployment perspective, the complete preprocessing and modeling workflow was organized into a clean and reproducible pipeline, including missing value handling, feature encoding, scaling, and model

evaluation. The final pipeline is structured in a way that allows new property data to be processed efficiently and price predictions to be generated, making the solution suitable for real-world implementation.

12. FINAL CONCLUSION

In this project, a comprehensive machine learning workflow was developed to predict **residential property prices using a large-scale U.S. real estate dataset**. Exploratory data analysis revealed significant right skewness in property prices, which was effectively addressed through log transformation of the target variable, leading to improved data stability and more reliable model training. The analysis identified house size and number of bathrooms as the most influential predictors of property prices, while features such as number of bedrooms and lot size showed comparatively weaker linear relationships.

A range of linear and tree-based regression models were trained and evaluated using standard regression metrics, including MAE, RMSE, and R^2 score. Pipelines were applied to linear models to integrate feature scaling and model training into a single workflow, ensuring consistent preprocessing and preventing data leakage. Tree-based models were trained without pipelines, as they are not sensitive to feature scaling.

The results demonstrated that tree-based ensemble models, particularly Random Forest, outperformed linear models, **achieving an R^2 score of approximately 0.63 along with the lowest RMSE**. This indicates that non-linear models are better suited for capturing the complex relationships present in real estate pricing data. Linear models showed weaker performance, confirming the limited linear structure observed during exploratory analysis.

Overall, the structured preprocessing approach, informed feature engineering, and systematic model evaluation support the reliability and robustness of the proposed solution. While the model does not capture all sources of price variation—reflecting the inherent complexity of the housing market—it provides a realistic, scalable, and interpretable framework that can serve as a strong foundation for real-world real estate price prediction systems.