**USA Real Estate**

Kevin Hansen, Ryan Brown, Kelly Arsenau

**Overview**

This project utilizes a 2024 real estate dataset covering listings across the United States to develop a predictive pricing model. The model is designed to estimate home prices based on user-input variables, including zip code, house size, number of bedrooms, and number of bathrooms. The original dataset contains 1,048,575 listings with attributes such as price, number of bedrooms and bathrooms, house size, lot size, and location details (city, state, and zip code). Significant preprocessing was performed to ensure the data was clean, consistent, and suitable for predictive analysis.

**Data Exploration**

Figure 1 provides a distribution overview of key housing features in the dataset. Most homes are priced between $300,000 and $400,000, with the distribution skewed slightly right, indicating a higher frequency of moderately priced homes and fewer high-end listings. The majority of properties include three bedrooms and two bathrooms. House sizes cluster around 2,000 square feet, and most lots are close to a quarter of an acre. These distributions highlight the typical property profile in the dataset and help guide the selection of features for predictive modeling.

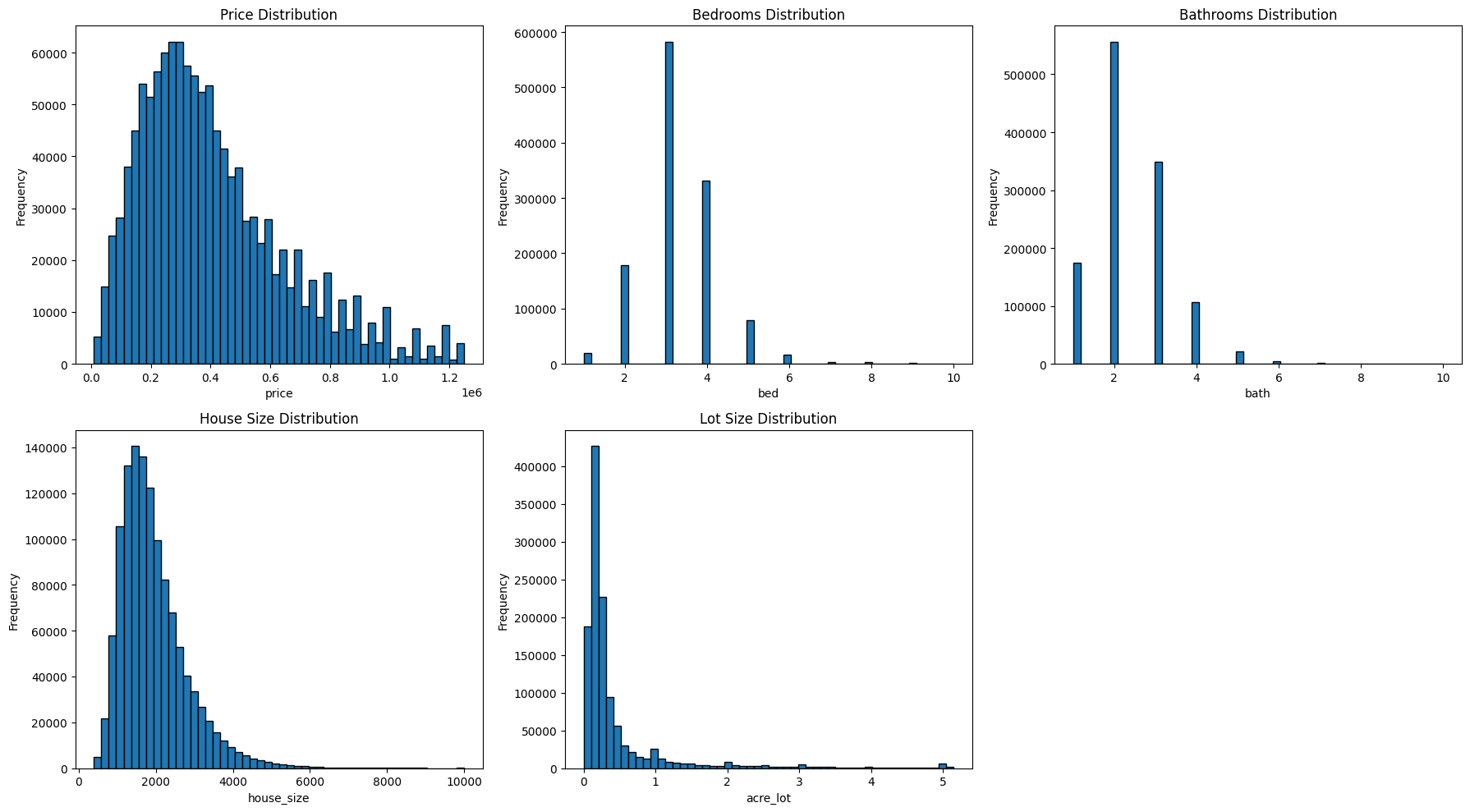


Figure 1. Feature Distribution

The distributions of the log-transformed variables (log\_price, log\_house\_size, log\_acre\_lot) are now more symmetric and approximately normal, reducing the effect of outliers. log\_price and log\_house\_size exhibit clear bell-shaped curves, indicating a stable distribution suitable for modeling. However, log\_acre\_lot shows some left-skewness and a broader distribution, suggesting more variability and potential outliers still present. This is all seen in Figure 2.

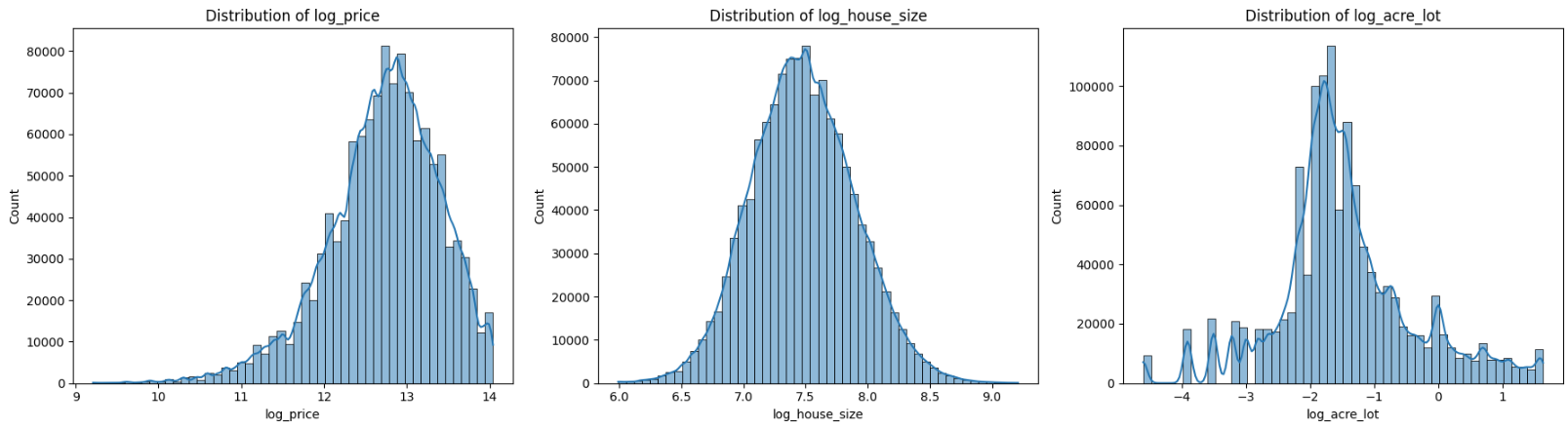


Figure 2. Log of Feature Distribution

**Data Cleaning**

Data cleaning and preprocessing involved a series of detailed steps to enhance dataset quality and ensure readiness for modeling. Non-U.S. territories such as Puerto Rico and the Virgin Islands were excluded, and irrelevant columns like brokered\_by and “street” were removed to preserve privacy and simplify analysis. Extreme outliers were capped to stabilize trends, and non-residential properties were filtered out based on unrealistic lot sizes and missing core housing attributes.

Additional corrections included standardizing zip codes with missing leading zeros and removing placeholder entries such as “99999.” Missing values were imputed using either state-level or dataset-wide medians, and city names were filled using the most frequent value within each zip code. Figure 3 illustrates the extent of missing data across key features prior to cleaning.

Feature engineering steps included categorizing house sizes to support structured comparisons. Finally, a small number of rows with missing city data were dropped. After cleaning, the dataset was reduced to 680,399 complete and usable listings—ready for feature selection and predictive modeling.

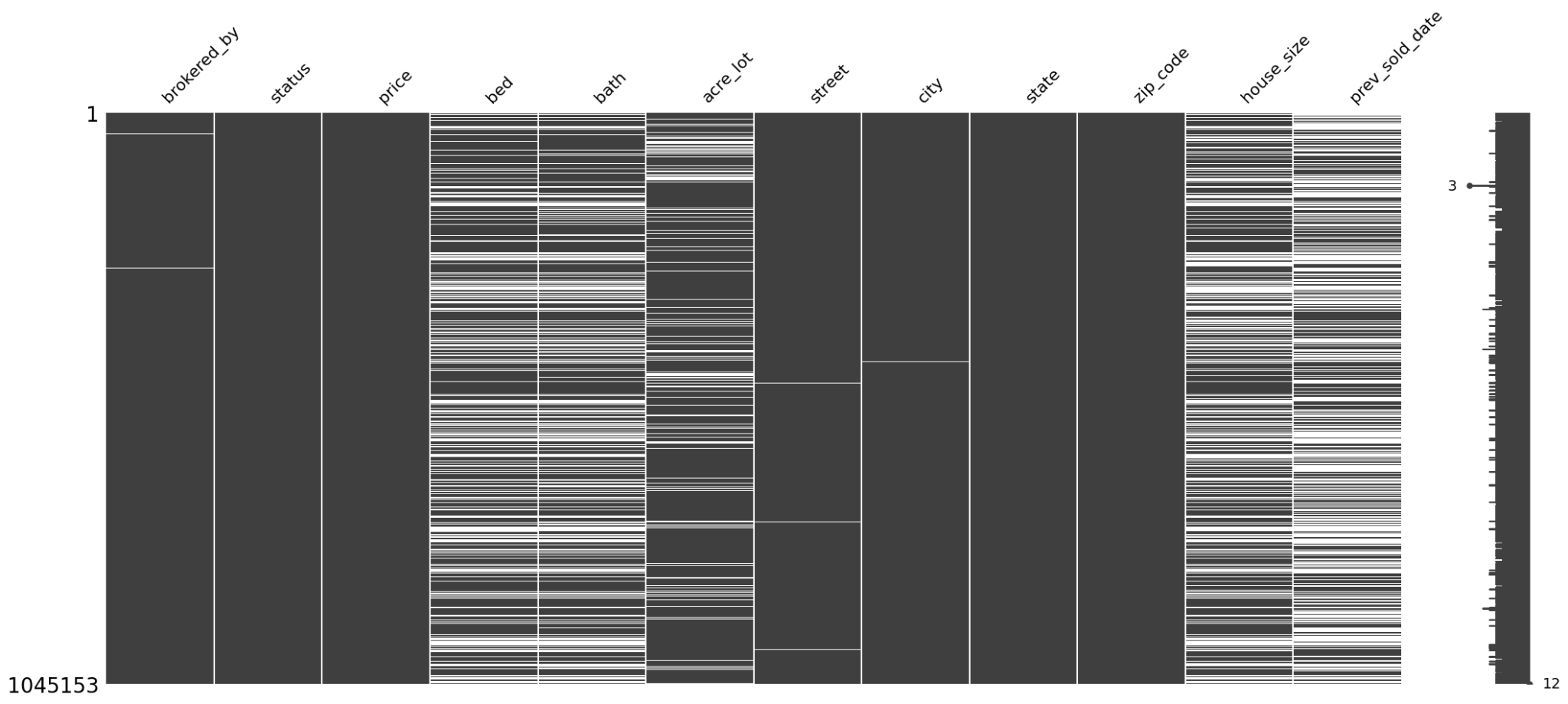


Figure 3. Missing Data Heatmap

### **Train-Test Split and Feature Preparation**

To prepare for modeling, the dataset was split into training and testing subsets using an 80/20 ratio. A Kolmogorov-Smirnov (KS) test was conducted to verify that the two subsets followed the same distribution of home prices. The test produced a KS statistic of 0.0026 and a p-value of 0.6859, indicating no statistically significant difference between the training and testing distributions. This confirmed that the split was balanced and free from selection bias.

Feature preparation included log-transforming skewed numerical variables such as price, house size, and lot size to normalize their distributions. The zip code feature was converted into a numerical format using encoding, allowing it to be used in model training. All selected features—bed, bath, log\_house\_size, log\_acre\_lot, and zip\_indexed—were combined into a single feature vector to support model input.

**Linear Regression**

Linear regression was used as a baseline model due to its simplicity and ease of interpretation. The model was trained on the log-transformed target variable, log\_price, using selected features: bed, bath, log\_house\_size, log\_acre\_lot, and zip\_indexed.

This initial model provided a useful benchmark but lacked the complexity needed to capture non-linear relationships between home attributes and price. It achieved an R² of 0.4346 and a root mean squared error (RMSE) of approximately $179,018.65 on the test set. These results indicated that the linear model struggled with pricing variability, especially for homes at the higher and lower ends of the price range.

Residual plots and predicted vs. actual price comparisons confirmed that the linear model consistently underpredicted high-priced homes and showed a wide spread in error. These limitations motivated the selection of a more advanced model capable of handling complex, non-linear interactions between features.

**Linear Regression Analysis**

The goal of this analysis was to develop a predictive model for housing prices using Linear Regression. The dataset included features such as the number of bedrooms, bathrooms, lot size, house size, and engineered features like price per acre and price per square foot.

The analysis aimed to create a predictive Linear Regression model for housing prices, utilizing features such as bedrooms, bathrooms, lot size, house size, and engineered metrics like price per acre and square foot. Data preparation involved excluding non-numeric features (e.g., city and zip code) and applying Min-Max scaling for uniform feature ranges. The linear regression model achieved an RMSE of 0.5176, suggesting an average prediction error of approximately ±0.52 for log-transformed housing prices, reflecting moderate accuracy. With an R² of 0.4580, the model explained about 45.8% of the variance in log\_price, highlighting limitations and indicating opportunities for further improvement in predictive performance. Figure 4 shows which features had the most influence on the model.

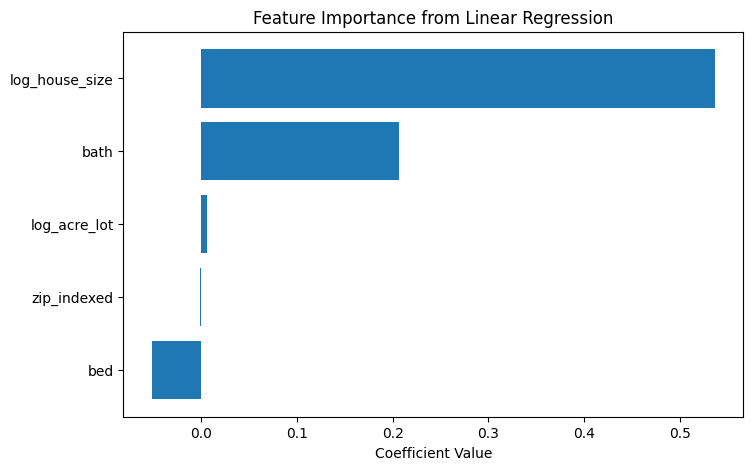


Figure 4. Feature importance from Linear Regression

Linear regression may oversimplify the complexities of real estate pricing, as it assumes linear relationships and fails to account for interactions between features like house size and lot size. Additionally, it excludes crucial location-based information like zip codes, which are pivotal in predicting housing prices. Incorporating zip codes as categorical variables can introduce complexity, but a model like Random Forest naturally evaluates feature importance and detects interactions.

**Gradient Boosted Trees**

Gradient Boosted Trees (GBT) were used to model log-transformed home prices due to their ability to handle non-linear relationships and feature interactions. The model was trained on five features: bed, bath, log\_house\_size, log\_acre\_lot, and zip\_indexed. Compared to the linear regression baseline, GBT provided a substantial improvement in predictive accuracy.

The model achieved an R² of 0.7465 on the training set and 0.7076 on the test set, indicating strong generalization to unseen data. The mean absolute error (MAE) was $80,347.01 for training and $85,965.94 for testing. Root mean squared error (RMSE) was $120,266.90 and $130,029.94, respectively. These metrics show the model captures pricing trends effectively, though individual predictions may deviate.

After removing the top 1% of home prices (extreme outliers), RMSE and MAE improved slightly, while R² remained consistent. This suggests that high-end properties had limited influence on overall model accuracy. However, a mean absolute percentage error (MAPE) of 30.58% indicates that predictions can still vary significantly for individual listings, particularly at higher price points.

The model performed best for mid-range properties. Accuracy declined for luxury homes, likely due to heteroscedasticity and missing high-end features. Future improvements could include adding neighborhood-level data, property condition, or segmented models for different market tiers.

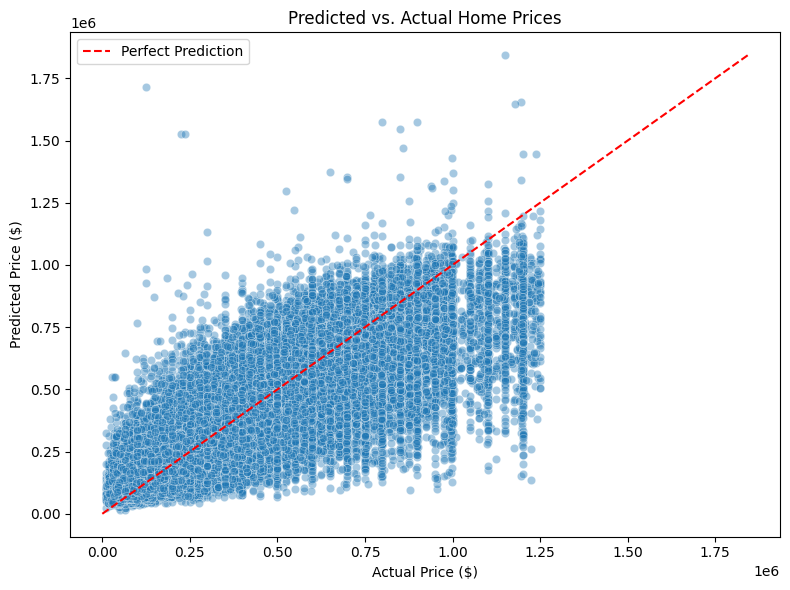


Figure 5. GBT Predictions vs. Actual Home Prices

This visualization confirms that GBT performs well on mid-priced homes but struggles more with luxury listings, consistent with the observed MAPE and RMSE. The prediction spread increases with price, indicating heteroscedasticity, which is typical in housing data.

### **Interesting/Surprising Results**

Several insights emerged during modeling. The log transformation of price, house size, and lot size dramatically improved distribution symmetry and model performance. Among features, zip code encoding (zip\_indexed) had the highest feature importance in the Gradient Boosted Trees model, even surpassing structural attributes like bedroom count. Conversely, the number of bedrooms contributed less than expected to price prediction.

Removing the top 1% of most expensive homes had only a minor impact on R² but slightly reduced RMSE, confirming that extreme outliers were not significantly skewing results. Finally, while the GBT model achieved strong overall accuracy, the predicted vs. actual plot revealed that the model performed best for mid-priced homes and underperformed on high-end properties, likely due to unmodeled luxury-specific variables.

### **Problems Encountered**

Several challenges emerged during model development. The initial dataset included missing values in critical columns such as house size, bed, and bath. These were handled through filtering and minimal imputation. The skewness of price and lot size required log transformation to reduce the influence of extreme values.

Encoding high-cardinality categorical variables like zip code introduced computational challenges, particularly with models like GBT, which required increasing the maxBins parameter to accommodate the number of unique regions. Lastly, the model demonstrated limited precision for high-priced homes, possibly due to a lack of luxury-specific features such as waterfront location, custom features, or neighborhood desirability.

### **Summary of How Well the Goals Were Achieved**

The project successfully met its objective of predicting home prices based on features commonly available in real estate searches: number of bedrooms, bathrooms, house size (sqft), lot size, and zip code. The final Gradient Boosted Trees model achieved an R² of 0.7076 and an RMSE of $130,029.94 on test data. After filtering out the top 1% of high-end homes, model performance improved slightly.

The model is particularly effective at capturing price trends and general value estimates for mid-range properties. While individual price predictions vary—particularly for high-value homes—the results show strong potential for real-world application, especially in tools where users input property specs to estimate market value. Additional data or model segmentation could further improve performance for niche markets such as luxury homes.

### **Conclusion and Next Steps**

The Gradient Boosted Trees (GBT) model demonstrated strong predictive performance with an R² of approximately 0.71 on test data, capturing key pricing trends across mid-range homes. However, there are several opportunities to improve accuracy, usability, and interpretability.

To better model location-based pricing variation, future iterations could incorporate features such as city, neighborhood, school quality ratings, and proximity to amenities. Spatial modeling using latitude and longitude (e.g., geohashing) may also enhance geographic resolution. Splitting house size into finished vs. unfinished square footage and including lot or garage size could further refine predictions.

To address underperformance on high-end homes, segmented modeling by price tiers or a dedicated model for luxury properties may be beneficial. Advanced methods such as XGBoost, LightGBM, or neural networks could uncover more complex interactions between features. Temporal features like build year or market conditions could help the model adapt to evolving real estate trends.

Deployment enhancements might include a user-friendly interface, confidence intervals for predicted prices, and model interpretability tools such as SHAP values. Expanding the dataset through external APIs or improving coverage of underrepresented regions would also support broader generalization.