Predicting Housing Prices Using Zillow Dataset: A Data-Driven Approach

Contributors:

Devakinandan Palla
Sneha Narayanan
Sai Swaminathan
Francisco Cheung

Business understanding

The Zillow dataset offers significant business opportunities for the real estate industry, particularly for buyers, sellers, and investors. It provides detailed information about real estate transactions across various locations in the United States, including attributes such as sale price, property size, location and the year of construction. It also offers opportunities to uncover trends and insights at both local and national levels, benefiting a wide range of stakeholders, such as real estate agents, investors, property buyers, and city planners.

Business Opportunities

- **Market Value Prediction Across Regions:** Analyze the factors influencing property prices across regions to develop precise pricing strategies tailored to different markets.
- **Buyer Guidance by Region:** Help buyers identify ideal properties based on their preferences (e.g., size, price range) across multiple cities or towns.
- **Regional Investment Decision Support:** Identify areas with high potential for property value appreciation, helping investors target lucrative opportunities.
- **Geographic Trends Analysis:** Understand how property prices vary across regions and analyze the influence of location-based factors such as proximity to universities, population density, or local amenities.
- **Impact of Home Characteristics:** Assess how home attributes, such as lot size, square footage, and number of bedrooms and bathrooms, contribute to variations in sale prices across different regions.
- **Urban Planning Insights:** Provide city planners and policymakers with data-driven insights to address housing affordability, optimize zoning policies, and identify potential areas for infrastructure investment.

Business Questions:

1. What are the key factors influencing property prices in different regions across the U.S.?
2. Can we predict the sale price of a property based on its features (e.g., size, lot, year built, number of bedrooms)?
3. How do property prices vary geographically, and what regional trends can be identified?
4. What is the order of priority in terms of a property's features that should be considered if a unit is planned to be built?
5. What strategies can we implement to balance sales contributions across different property types?
6. How do we segment the available features into distinct types which helps to strategize business action steps more efficiently.

Data understanding:

Based on the descriptive analysis of the dataset the following business insights were found:

Market Value and Variability:

- The **Sale Amount** has a wide range, from a low of $17,000 to a high of $9.5 million, indicating a high variability in property values. This presents an opportunity to **develop pricing strategies** by considering factors like home size, location, and year of construction.
- The **mean sale price** is much higher than the **median** ($337,556 vs. $249,000), suggesting that while most properties are priced around $250,000, XXC, the average valuation is much higher due to the impact of luxury properties.

Home and Lot Sizes:

- The **Square Footage of Homes (Sqft_home)** ranges widely, with a median of 1,826 sqft. The wide range of home sizes presents an opportunity for **targeted marketing** for both buyers (looking for smaller vs. larger homes) and investors (identifying high-value homes).
- **Lot sizes (Sqft_lot)** also vary greatly, with a significant mean of 20,469 sqft. For real estate investors, evaluating the **lot size** relative to the sale price could help identify undervalued properties with potential for appreciation.

Number of Bedrooms and Bathrooms:

- The **median number of bedrooms (3)** and bathrooms (2) reflects common preferences for mid-sized homes. Buyers and real estate agents can **target specific property configurations** (e.g., 3-bedroom homes) based on the most common configurations.
- A wider range of bathrooms (from 0.5 to 43) suggests some properties may be larger luxury homes with more bathrooms, influencing the sale price.

Age of Properties:

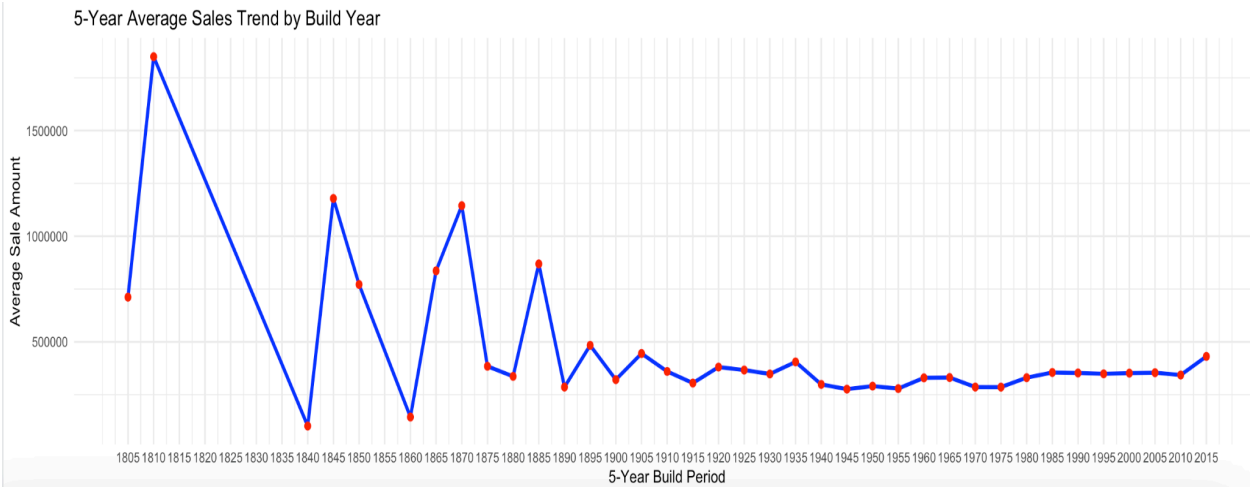
- The **mean build year** of 1968 suggests that many properties are relatively old but may have been renovated. Older homes may be **undervalued** relative to newer properties, presenting an opportunity for **investment** in renovations or flipping properties.

.

Exploratory Data analysis:

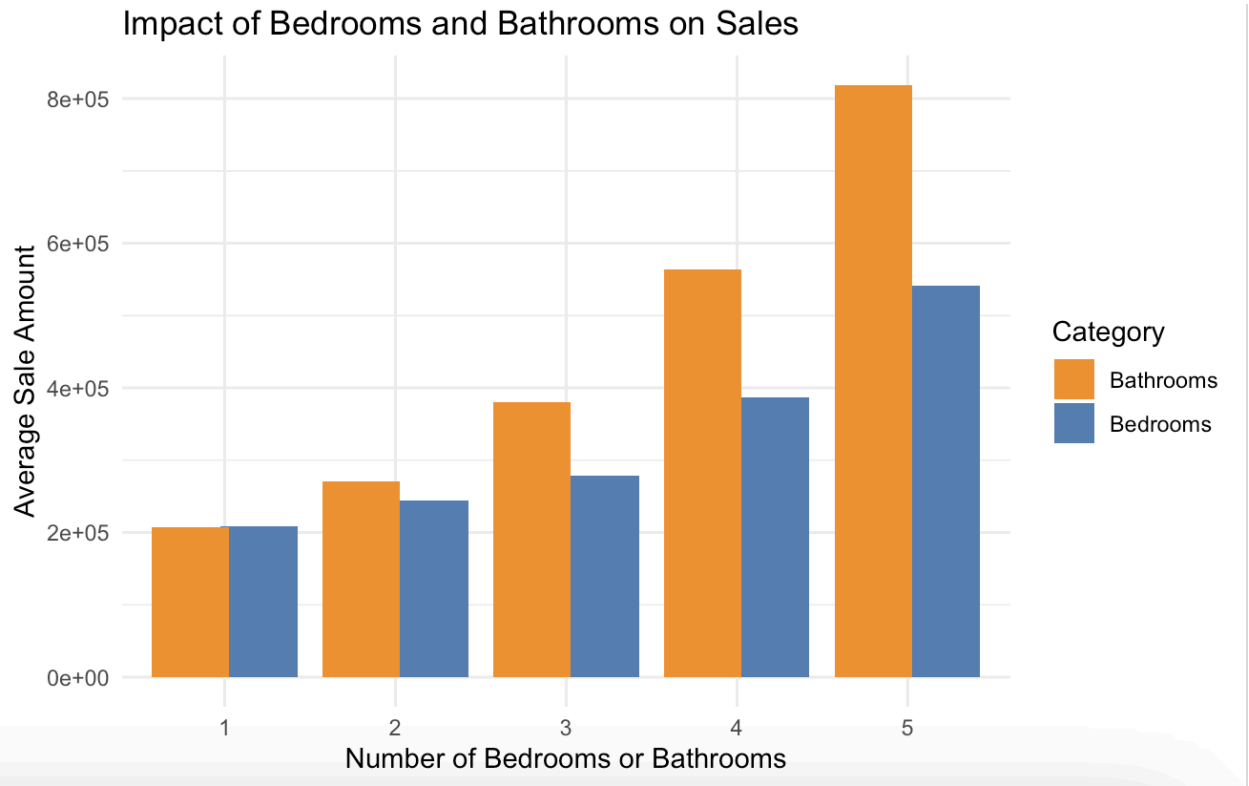
1. Average Sales Trend by Build Year

   The average sales price appears to be more volatile in the older built properties of the data set. This could be due to several unexplainable factors, and the year of construction does not explain much, although a small upward trend in demand can be noticed in the newer buildings.
   There may be some inconsistencies in the data set, which could be due to a number of factors, such as the sale of luxury homes or homes in highly desirable locations.



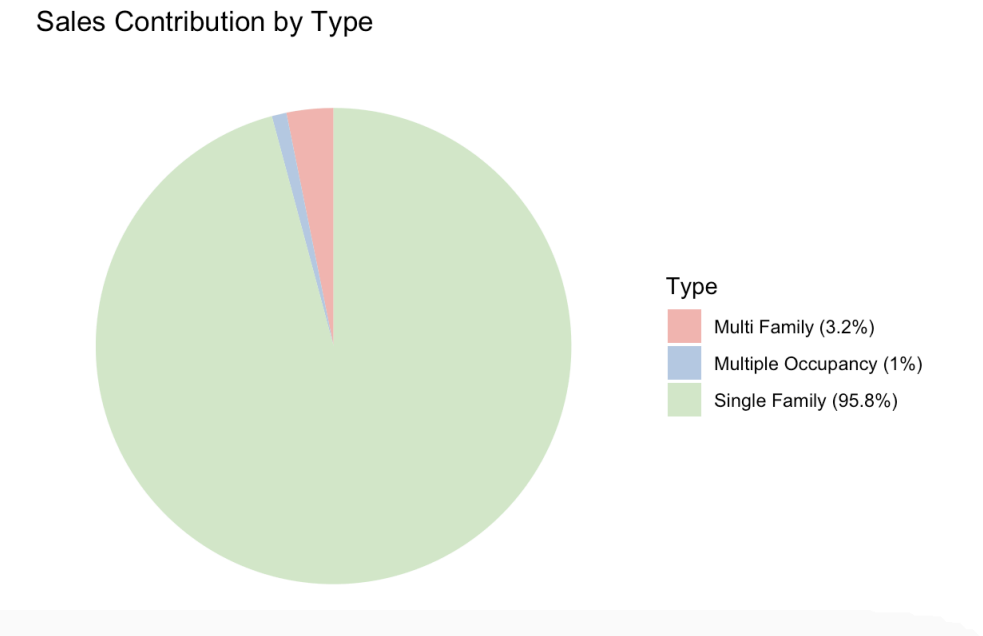
2. Impact of Bedrooms and Bathrooms on Sales

- Both bedrooms and bathrooms exhibit a positive correlation with average sale price. As the number of bedrooms or bathrooms increases, the average sale price tends to rise.
- The impact of bathrooms on sale price seems to be more pronounced than that of bedrooms. The average sale price increases more rapidly with each additional bathroom compared to an additional bedroom.
- While the trend is generally upward, there might be a point of diminishing returns, especially for bedrooms. Beyond a certain number of bedrooms, the increase in sale price per additional bedroom could become less significant.



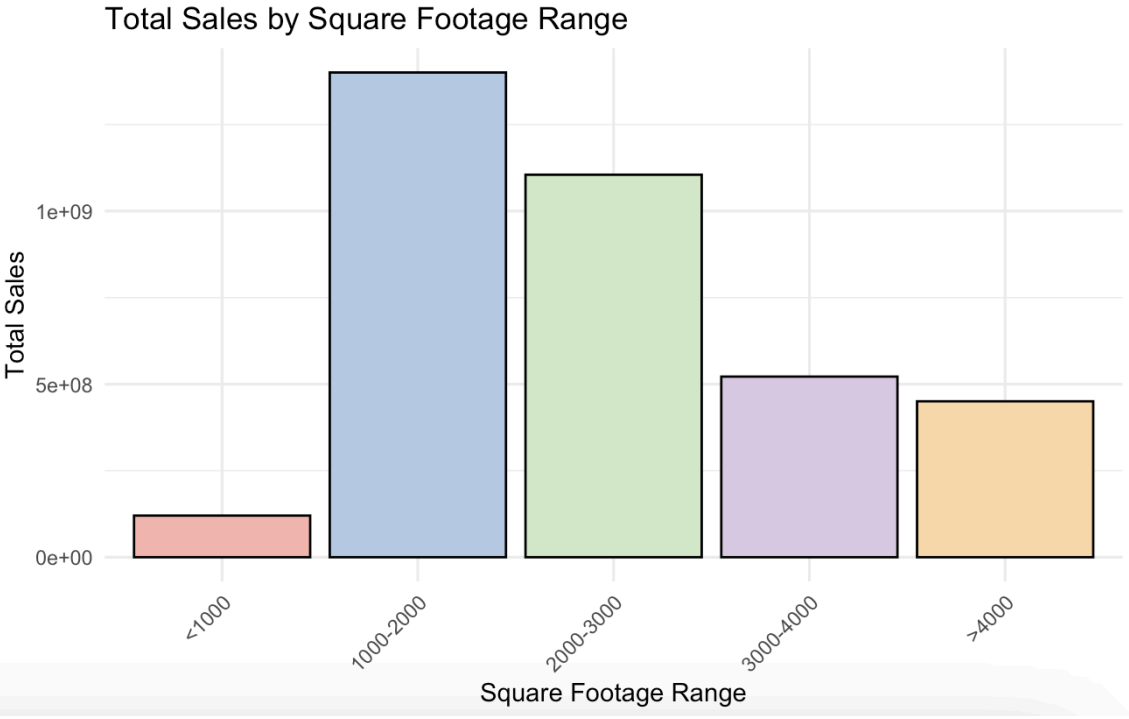
3. Sales Contribution by Type

The dominance of single-family homes suggests that the market primarily caters to individual buyers or families. It would be interesting to compare average sale prices across different property types to understand pricing trends and potential factors influencing them.
The smaller contributions of multi-family and multiple occupancy properties could indicate opportunities for investment or specific market niches.



4. Total Sales by Square Footage Range

As the square footage range increases, the total sales also tend to increase. This indicates a positive correlation between house size and sales value.The largest square footage range (">4000 sqft") contributes significantly to the overall sales, suggesting a strong demand for larger homes. While larger homes generally command higher prices, there might be a point of diminishing returns. It's possible that beyond a certain size, the increase in sale price per additional square foot becomes less significant.



Data preparation:

1. Separation of 'Town' and 'State': The combined 'Town' column was split into two columns, "Town" and "State", to improve data structure. This allows for more focused analysis on location.

2. Converting 'Sale_date' to Date Format: The 'Sale_date' column was converted from a string to a Date format to enable accurate time-based analysis and trend identification.

3. Extracting Year from 'Sale_date': The year was extracted from the 'Sale_date' column to facilitate year-over-year analysis of property sales.

4. Calculating Average and Median Sales by Year: The data was grouped by year to calculate the average and median sales for each year, providing insights into property price trends over time.

5. Ensuring Data Quality: The dataset was checked for missing values and duplicates. No missing values or duplicate records were found, ensuring data integrity for analysis.

Modeling:

In this task, we are analyzing a housing dataset to predict the sale amount (Sale_amount) based on various features: Beds, Baths, Sqft_home, and Build_year. The goal is to identify which features (and their interactions) most significantly affect the sale price and to evaluate the performance of different regression models. We will explore 2 unsupervised models: K Median Clustering and DBSCAN, and three supervised models: Linear Regression, Random Forest, along with some additional respective variations with interactions.

K-Median Clustering is a partitioning algorithm that groups data points into kkk clusters by minimizing the sum of the distances between each point and the median of its assigned cluster. Unlike K-Means, which uses the mean as the cluster center, K-Median is more robust to outliers due to its reliance on medians, and this is more relevant to our dataset with a variety in the prices.

Initial K-Median Clustering: We start by clustering the data based on key numeric attributes (e.g., Beds, Baths, Sqft_home, and Build_year) to identify patterns in housing characteristics.

Extended K-Median Clustering: We enhance the clustering process by incorporating categorical attributes such as Town and State, converting them into numerical representations (e.g., one-hot encoding) to analyze whether including these features improves cluster cohesiveness and interpretability.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) identifies clusters as dense regions in the data, separating sparse points as noise. Unlike K-Means or K-Median, it doesn't require predefining the number of clusters and handles noise effectively.

Initial DBSCAN Clustering: Applied to scaled features (e.g., Beds, Baths, Sqft_home, Build_year) to detect dense regions, with sparse points labeled as noise.

Refined DBSCAN Clustering: Improved by tuning parameters (ϵ\epsilonϵ and MinPts) and incorporating encoded categorical features (e.g., Town, State) for more meaningful clusters.

Linear regression is a straightforward approach that is useful when we assume that there is a linear relationship between the predictors and the target variable (Sale_amount). It is essential for understanding how each predictor contributes to the target. We started by fitting a basic linear regression model and then explored models with interactions.

1. Basic Linear Regression: We first fit the model with Beds, Baths, Sqft_home, and Build_year as predictors.
2. Interaction Model: We extended the model to include interactions between Beds and Baths to capture potential synergies.
3. Full Interaction Model: A model with all possible interactions between the predictors was created to identify any non-linear relationships.

Random Forest is a non-parametric, ensemble learning method that can model complex, non-linear relationships. It builds multiple decision trees and averages their predictions to reduce variance and improve accuracy. Given its ability to capture non-linear interactions and its robustness to overfitting, Random Forest is particularly useful when dealing with a large number of predictors or complex relationships.

1. Initial Random Forest: We first fit a Random Forest model using the main predictors (Beds, Baths, Sqft_home, and Build_year) to predict Sale_amount.
2. Extended Random Forest: We then included categorical features such as Town and State to see if these additional predictors improve model performance.

Evaluation: Refocus on the business objectives of the project. Evaluate whether the model(s)

have properly achieved the business objectives outlined during the business understanding phase.

Formulate actionable recommendations based on the findings.

Silhouette Score Comparison

Model Type	Silhouette Score
K Median	0.4271
DBSCAN	0.53

Model Performance Comparison Table

Metric	Linear Regression	XGBoost	Random Forest	Decision Tree
RMSE	275669.4	160323.5076	173489.936	212473.6
MAE	160230.7	70548.63165	75768.69108	120242.4
MSE	75993640559	257036271.04	30098757900	45145013474
R-squared (%)	32.28%	78.59%	71.51%	57.62%

Factors Influencing Property Prices in Different RegionsThe key factors influencing property prices across different regions are reflected in the model evaluation and feature importance.

- **K Median Clustering Findings** :
    - **Key features**: Sqft_home, Beds, Baths, and Sale_amount emerge as key variables for clustering housing data. The clustering process reveals distinct property types based on these characteristics, with the median values providing a robust representation of each cluster.
    - Cluster 1 represents high-value properties with significantly larger square footage (median Sqft_home = 4013) and higher numbers of Beds and Baths, indicative of premium homes.
    - Cluster 2 includes mid-range properties with moderate square footage (median Sqft_home = 2688) and standard configurations of Beds and Baths.
    - Cluster 3 captures lower-value properties with smaller square footage (median Sqft_home = 1532), fewer Beds, and fewer Baths.
- **DBSCAN Findings**:
    - **Key features**: DBSCAN clustering applied to scaled data reveals three clusters (0, 1, and 2) with noise points excluded. Key features include Sqft_home, Beds, Baths, and other relevant property attributes.
    - Cluster 0 (Noise): Contains 6924 points labeled as noise, likely representing outliers or sparsely distributed data points that do not belong to any cluster.
    - Cluster 1: Contains 3035 points, representing a significant cluster of data points in dense regions of the feature space.
    - Cluster 2: Contains 700 points, indicating a smaller yet distinct cluster with high cohesion.

- **Linear Regression Findings**:
    - **Key features**: Sqft_home, Beds, Baths, and Year Built emerge as important predictors.
    - The **interaction term** between Baths and Beds suggests that these two features together can have a significant effect on the price, which makes sense in the context of determining overall property size and layout.
- **Random Forest Findings**:
    - Features like **Sqft_home**, **Beds**, **State**, and **Build_year** play important roles in predicting property prices. This indicates that not only the physical characteristics of the property (size, bedrooms, and age) but also the geographical location (State) strongly influence the price.
- **Decision Tree Findings**:
    - Decision trees identify **State**, **Sqft_home**, and **Build_year** as the most impactful features for predicting property prices. The hierarchical splits highlight key thresholds, such as **Sqft_home < 2627**, that differentiate high- and low-value properties.
    - The tree structure reveals that properties with larger square footage and newer build years in specific states tend to command significantly higher prices, emphasizing the combined importance of physical and regional attributes

Our best model however turned out to be XGBoost with impeccable results when evaluated with metrics.

- XGBoost captures complex feature interactions while preventing overfitting, making it the most precise and reliable model for house price predictions, and that is validated when put up against other models too.

2. Predicting Sale Price Based on Property Features

The model has shown that **property features such as size (Sqft_home), number of bedrooms (Beds), bathrooms (Baths), and year built (Build_year)** are crucial for predicting sale prices. The significant improvements in predictive performance with the inclusion of geographical features (Town, State) demonstrate that models can accurately predict sale prices when regional context is considered.

XGBoost RMSE: 160,323.5 – the best-performing model, providing the most precise sale price predictions.

Random Forest RMSE: 173,489.9 – a strong alternative with slightly lower accuracy but robust predictions.

Decision Tree RMSE: 212,473.6 – useful for interpretability but less accurate compared to ensemble models.

Linear Regression RMSE: 275,669.4 – serves as a baseline model but struggles to capture non-linear interactions.

- Model Insights:
    - To optimize property pricing models, developers and real estate agents should prioritize key features like property size (Sqft_home), regional context (Town, State), and age (Build_year). Features such as the number of bedrooms and bathrooms, though relevant, have comparatively less influence on the predicted prices.
- Recommendation:
    - The analysis suggests that **property size (Sqft_home)** should be a primary focus for developers and real estate agents, as it has the most significant influence on sale prices. Additionally, leveraging **regional context (Town, State)** is crucial, as location plays a vital role in determining property value. Emphasizing the modernity of properties through **Build_year** can also attract higher prices, especially for newer homes. While **Beds** and **Baths** are less critical, optimizing property layouts with functional configurations can enhance buyer appeal. For older properties, targeting high-demand areas with historical or cultural significance can help maintain strong pricing potential.

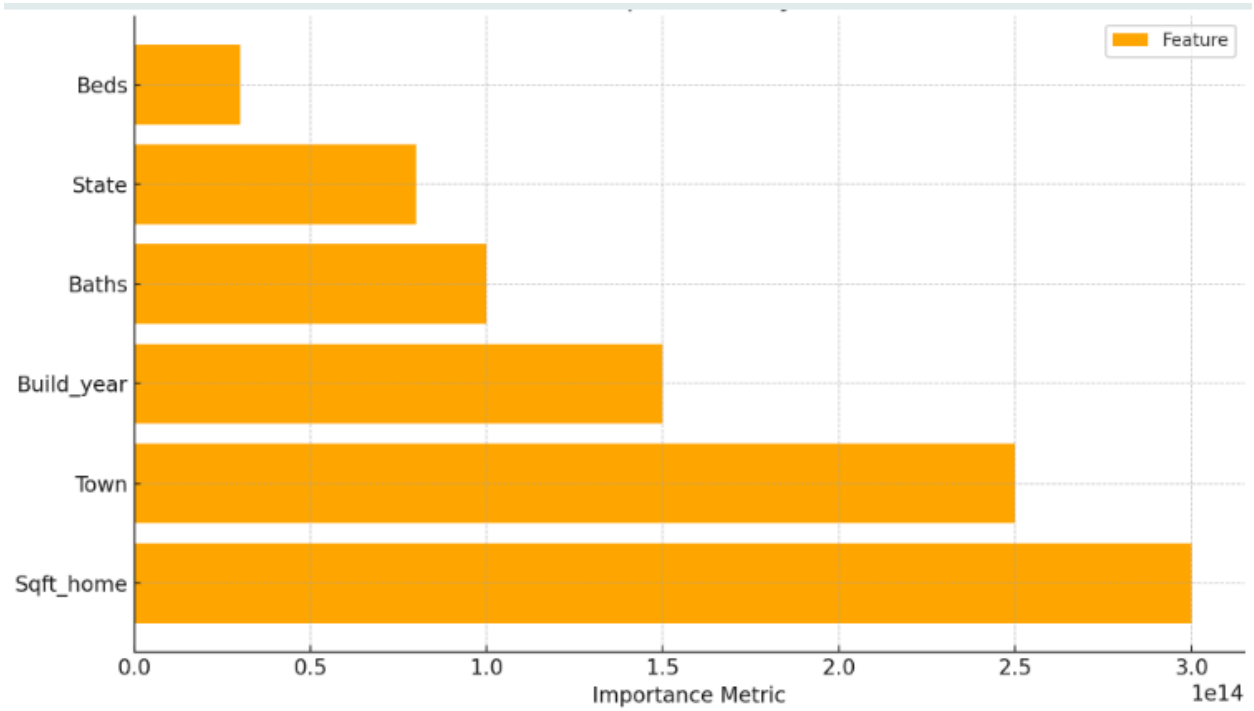
Deployment:

Actionable Recommendations:

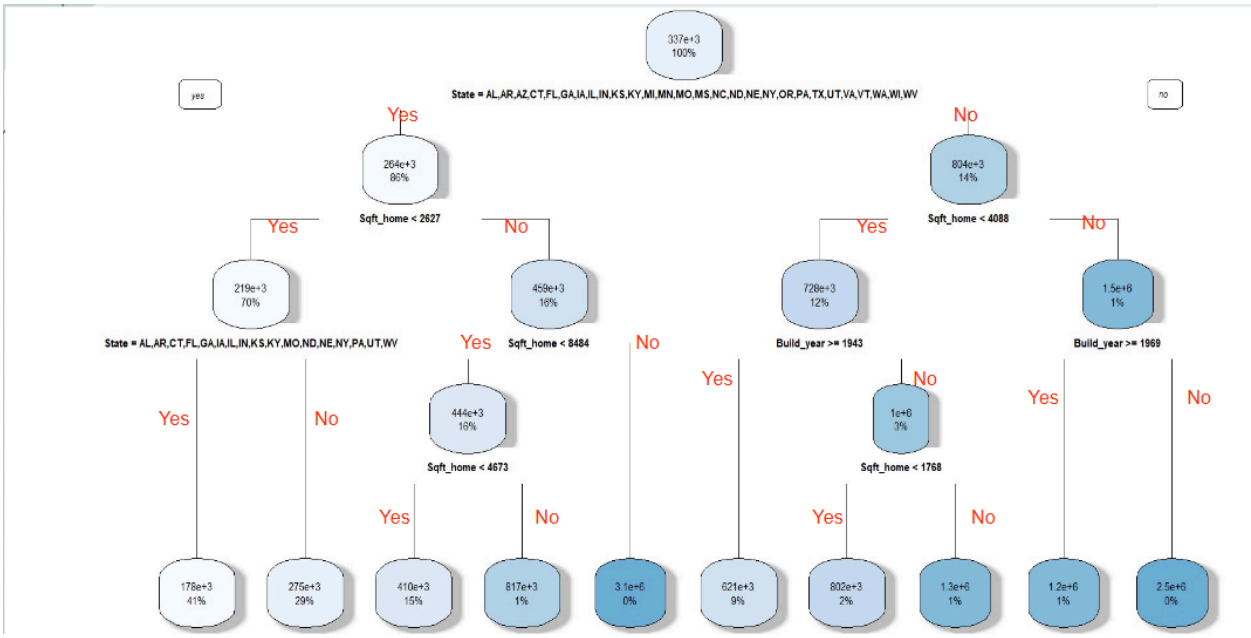
- **Leverage Predictive Models**: Real estate firms and developers should leverage these predictive models, especially the **XGBoost**, to make more informed decisions about pricing and investment.
- **Focus on Regional Analysis**: Pay particular attention to geographical factors—**state**, **town**, and **proximity to landmarks**—as they significantly influence property values.
- **Consider Age and Size**: Optimize property development strategies around features like **square footage**, **bedrooms/bathrooms**, and **build year**, which are proven to impact pricing.

- **Invest in Hotspots**: Look for emerging areas with potential for growth and development, where property values are currently undervalued.
- **Incorporate Time-Based Data**: Future models could benefit from integrating **time-series data** to better understand price trends over time.

RESULTS THAT FURTHER HELP TO SOLVE OUR BUSINESS PROBLEMS:



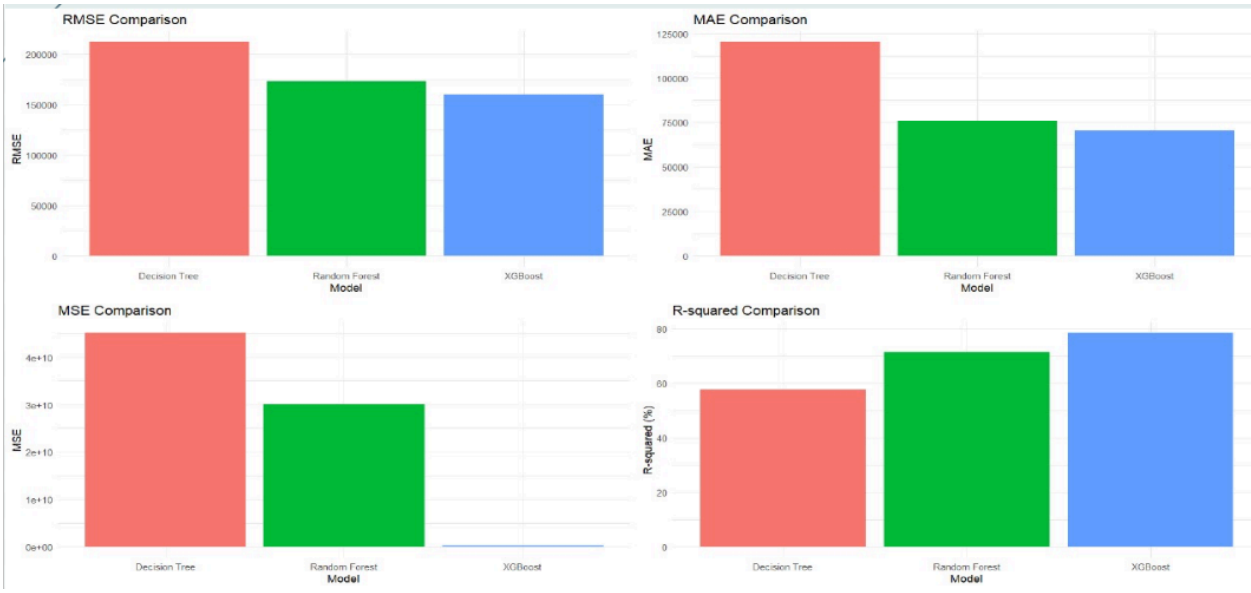
This chart shows the importance of each variable analyzed. Square footage and town are the most critical factors influencing property value. Baths and build year have a moderate impact, while the amount of beds and state the property located in play smaller roles. Realtors can prioritize listings emphasizing large homes in desirable towns.



It highlights how factors like square footage, build year, and state drive property value and sales potential. For example, homes under 2,627 sq ft in states like Alaska, Connecticut and Florida generate an average sales performance of $178K, while in other states, a similar 2627 sq ft house could yield around $275K. Similarly, it is fair to take a note on prices based on conditions at every level, which gives a much more detailed overview.

COMPARISON RESULTS:

Since all the below 3 models were judged on a common ground, we also made a comparison analyses.

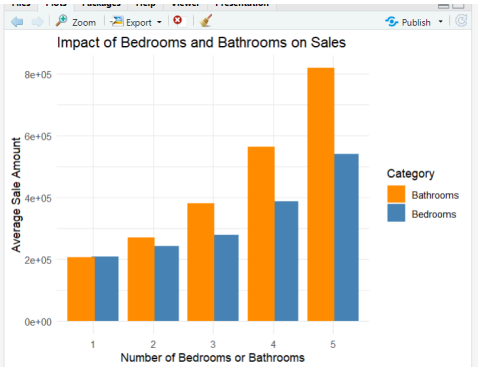


Our most impressive model, as highlighted earlier was the XG Boost algorithm in terms of metrics such as MSE, MAE and RMSE. It also had the most impressive R square of almost 79%, which is by far the best one and is very impressive considering the scale of the data we had to gather insights from.

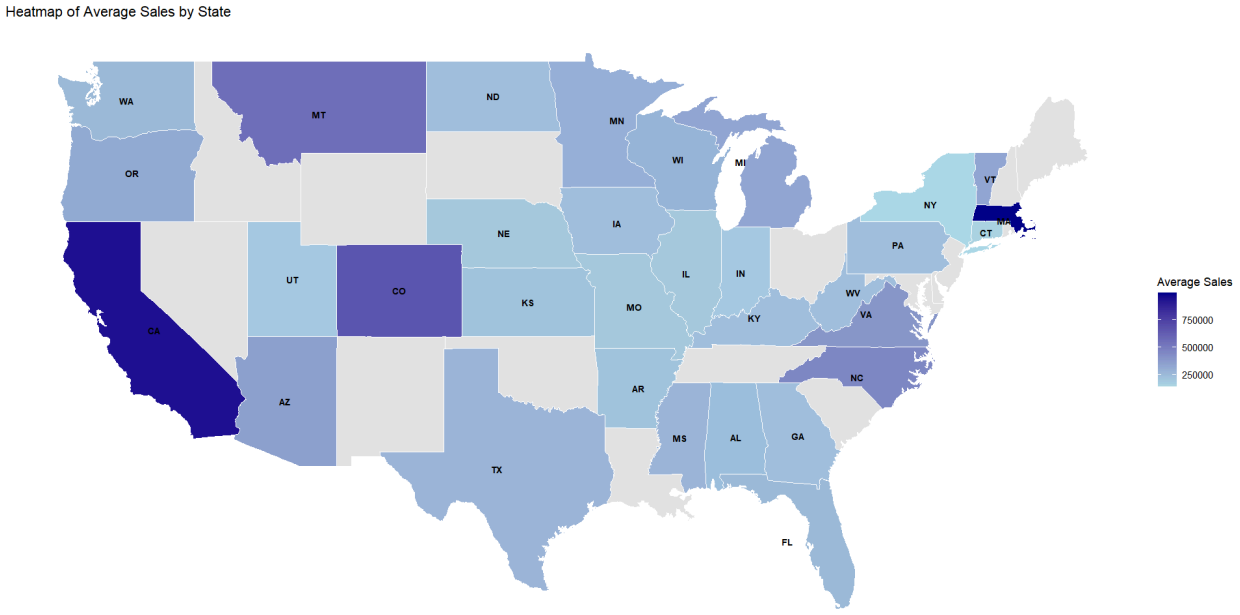
APPENDIX:

```
> house_data<-read_excel('C:/Users/Devakinandan/Downloads/Big_Data_Files.xlsx', sheet="House_Price")
> total_missing<-sum(is.na(house_data))
> total_missing
[1] 0
> #Interp: No missing values found in the datset
> # Check for duplicates:
> num_duplicates<-sum(duplicated(house_data))
> num_duplicates
[1] 0
> #There are no duplicates in the dataset as well
> #Different approach to outliers , reason to be stated
> summary(house_data$Sale_amount)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  17000  165975  249000  337556  375000 9500000
> nrow(house_data)
[1] 10659
> ncol(house_data)
[1] 11
> house_data <- house_data %>%
+   separate(Town, into = c("Town", "State"), sep = ", ", remove = FALSE)
> # View the updated dataset
> head(house_data)
# A tibble: 6 × 12
  Record Sale_amount Sale_date           Beds Baths Sqft_home Sqft_lot Type          Build_year Town  State
   <dbl>       <dbl> <dttm>             <dbl> <dbl>     <dbl>    <dbl> <chr>              <dbl> <chr> <chr>
1      1      295000 2016-05-31 00:00:00     5     3      2020    38333. Single Family       1976 Ames  IA
2      2      240000 2016-06-20 00:00:00     4     2      1498    54014. Single Family       2002 Ames  IA
3      3      385000 2016-05-31 00:00:00     5     4      4000    85813. Single Family       2001 Ames  IA
4      4      268000 2016-04-12 00:00:00     3     2.5    2283   118919. Single Family       1972 Ames  IA
5      5      186000 2016-04-05 00:00:00     3     1.25   1527    15682. Single Family       1975 Ames  IA
6      6      302500 2016-03-02 00:00:00     4     3      3117    33106. Single Family       1976 Ames  IA
# i 1 more variable: University <chr>
>
```

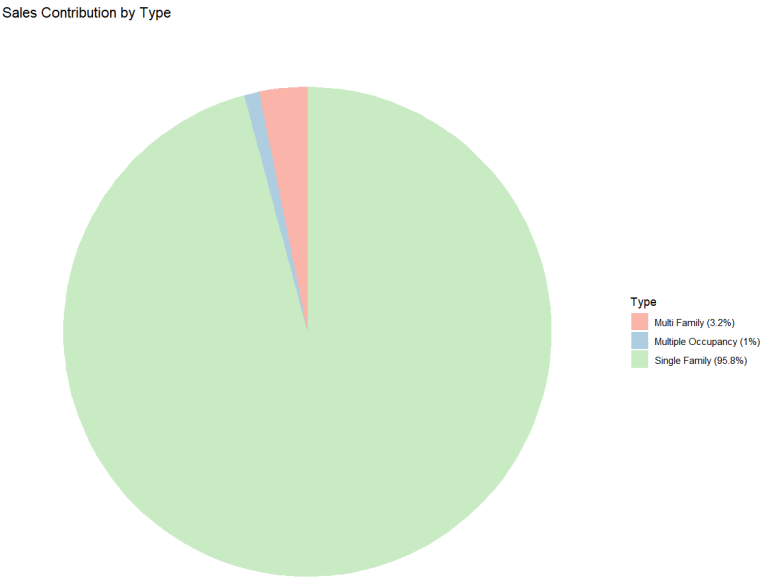
```
+   group_by(Beds) %>%
+   summarise(Average_Sales = mean(Sale_amount, na.rm = TRUE)) %>%
+   mutate(Category = "Bedrooms")
> # Calculate average sales grouped by Bathrooms
> bathroom_sales <- house_data %>%
+   group_by(Baths) %>%
+   summarise(Average_Sales = mean(Sale_amount, na.rm = TRUE)) %>%
+   mutate(Category = "Bathrooms")
> # Combine the data into a single dataset
> combined_sales <- bind_rows(
+   bedroom_sales %>% rename(Value = Beds),
+   bathroom_sales %>% rename(Value = Baths)
+ ) %>%
+   filter(Value %in% c(1, 2, 3, 4, 5))  # Keep only values 1 to 5
> # Plot grouped bar plot
> ggplot(combined_sales, aes(x = as.factor(Value), y = Average_Sales, fill = Category)) +
+   geom_bar(stat = "identity", position = position_dodge(width = 0.8)) +
+   labs(
+     title = "Impact of Bedrooms and Bathrooms on Sales",
+     x = "Number of Bedrooms or Bathrooms",
+     y = "Average Sale Amount",
+     fill = "Category"
+   ) +
+   scale_x_discrete(limits = c("1", "2", "3", "4", "5")) +  # Set x-axis scale to 1-5
+   scale_fill_manual(values = c("Bedrooms" = "steelblue", "Bathrooms" = "darkorange")) +
+   theme_minimal()
>
```



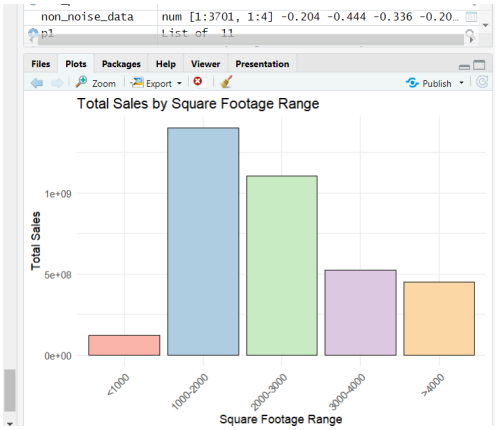
```
> ##viz 3-updated:
> ggplot(map_data_with_sales, aes(x = long, y = lat, group = group, fill = Average_Sales)) +
+   geom_polygon(color = "white") +
+   geom_text(
+     data = state_centers,
+     aes(x = center_long, y = center_lat, label = State),
+     inherit.aes = FALSE,  # do not inherit the group aesthetic from the main plot
+     color = "black",
+     size = 3,
+     fontface = "bold"
+   ) +
+   scale_fill_gradient(low = "lightblue", high = "darkblue", na.value = "grey90") +
+   labs(
+     title = "Heatmap of Average Sales by State",
+     fill = "Average Sales"
+   ) +
+   theme_minimal() +
+   theme(
+     panel.grid = element_blank(),
+     axis.title = element_blank(),
+     axis.text = element_blank(),
+     axis.ticks = element_blank()
+   )
Warning message:
Removed 17 rows containing missing values or values outside the scale range (`geom_text()`).
> #viz 4:
> # Aggregate sales data by Type
> type_sales <- house_data %>%
+   group_by(Type) %>%
+   summarise(Total_Sales = sum(Sale_amount, na.rm = TRUE)) %>%
+   mutate(Percentage = Total_Sales / sum(Total_Sales) * 100)
>
```



```
> #viz 4:
> # Aggregate sales data by Type
> type_sales <- house_data %>%
+   group_by(Type) %>%
+   summarise(Total_Sales = sum(Sale_amount, na.rm = TRUE)) %>%
+   mutate(Percentage = Total_Sales / sum(Total_Sales) * 100)
> # Update the Type labels to include percentages
> type_sales <- type_sales %>%
+   mutate(Type = paste0(Type, " (", round(Percentage, 1), "%)"))
> # Plot the pie chart with updated legend labels
> ggplot(type_sales, aes(x = "", y = Total_Sales, fill = Type)) +
+   geom_bar(stat = "identity", width = 1) +
+   coord_polar("y", start = 0) +
+   labs(
+     title = "Sales Contribution by Type",
+     x = NULL,
+     y = NULL,
+     fill = "Type"
+   ) +
+   scale_fill_brewer(palette = "Pastel1") +  # Optional color scheme
+   theme_minimal() +
+   theme(
+     axis.text = element_blank(),
+     axis.ticks = element_blank(),
+     panel.grid = element_blank()
+   )
```



```
> # viz 5:
> # Load necessary libraries
> library(dplyr)
> library(ggplot2)
> # Bin square footage into categories using Sqft_home
> house_data <- house_data %>%
+   mutate(SqFt_Bins = cut(
+     Sqft_home,
+     breaks = c(0, 1000, 2000, 3000, 4000, Inf),
+     labels = c("<1000", "1000-2000", "2000-3000", "3000-4000", ">4000"),
+     include.lowest = TRUE
+   ))
> # Aggregate sales data by Sq Ft bins
> sqft_sales <- house_data %>%
+   group_by(SqFt_Bins) %>%
+   summarise(Total_Sales = sum(Sale_amount, na.rm = TRUE)) %>%
+   mutate(Percentage = Total_Sales / sum(Total_Sales) * 100)
> # Plot the bar chart
> ggplot(sqft_sales, aes(x = SqFt_Bins, y = Total_Sales, fill = SqFt_Bins)) +
+   geom_bar(stat = "identity", color = "black") +
+   labs(
+     title = "Total Sales by Square Footage Range",
+     x = "Square Footage Range",
+     y = "Total Sales",
+     fill = "Sq Ft Range"
+   ) +
+   scale_fill_brewer(palette = "Pastel1") +  # Optional color scheme
+   theme_minimal() +
+   theme(
+     legend.position = "none",  # Remove legend since categories are on the x-axis
+     axis.text.x = element_text(angle = 45, hjust = 1)  # Rotate x-axis labels for readability
+   )
```




```
> #Adding interactions to check R sq:
> lm_model_interaction <- lm(Sale_amount ~ Baths * Beds + Sqft_home + Build_year, data = house_data)
> summary(lm_model_interaction)

Call:
lm(formula = Sale_amount ~ Baths * Beds + Sqft_home + Build_year,
    data = house_data)

Residuals:
     Min       1Q   Median       3Q      Max
-2206780  -129950   -42051    47961  4462991

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.922e+06  1.772e+05  22.139   <2e-16 ***
Baths        1.307e+05  4.352e+03  30.029   <2e-16 ***
Beds         5.360e+03  3.501e+03   1.531    0.126
Sqft_home    7.670e+01  3.467e+00  22.122   <2e-16 ***
Build_year  -2.050e+03  9.095e+01 -22.543   <2e-16 ***
Baths:Beds  -3.762e+03  4.041e+02  -9.310   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 279000 on 10653 degrees of freedom
Multiple R-squared:  0.3321,    Adjusted R-squared:  0.3318
F-statistic:  1059 on 5 and 10653 DF,  p-value: < 2.2e-16
```

```
> db <- dbscan(scaled_data, eps = eps_value, minPts = minPts_value)
> table(db$cluster)

   0    1    2
6958 3032  669
> # Evaluate using Silhouette (internal validation)
> # First, exclude noise points from silhouette calculation
> non_noise_indices <- which(db$cluster != 0)
> non_noise_data <- scaled_data[non_noise_indices, ]
> non_noise_clusters <- db$cluster[non_noise_indices]
> # Compute distance matrix
> dist_matrix <- dist(non_noise_data, method = "euclidean")
> # Compute silhouette object
> sil_obj <- silhouette(non_noise_clusters, dist_matrix)
> summary_sil <- summary(sil_obj)
> cat("Average silhouette width:", summary_sil$avg.width, "\n")
Average silhouette width: 0.540151
```

```
Decision Tree RMSE: 212473.6
> cat("Decision Tree MAE:", dt_mae, "\n")
Decision Tree MAE: 120242.4
> cat("Decision Tree MSE:", dt_mse, "\n")
Decision Tree MSE: 45145013474
> # Calculate R-squared
> dt_sst <- sum((test_target - mean(test_target
> dt_sse <- sum((test_target - dt_predictions)^
> dt_r_squared <- 1 - (dt_sse / dt_sst)
> cat("Variance Explained (R-squared):", dt_r_s
Variance Explained (R-squared): 57.62095 %
```

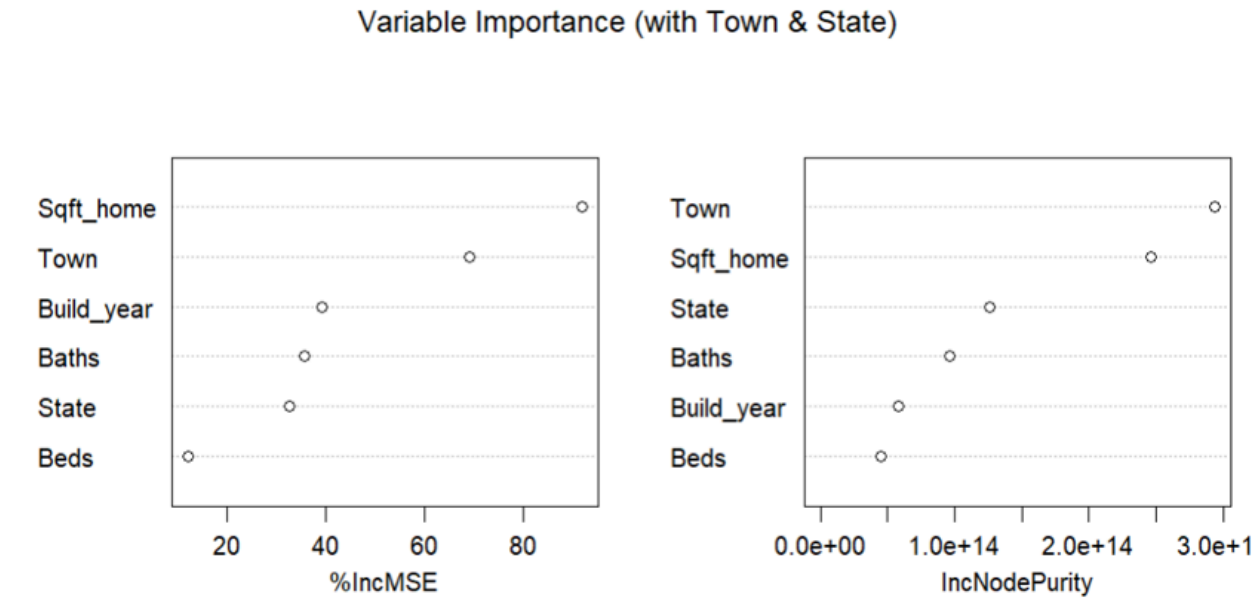
```
Random Forest RMSE (with Town & State): 173489.936
> cat("Random Forest MAE (with Town & State):", mae, "\n")
Random Forest MAE (with Town & State): 75768.69108
> cat("Random Forest MSE (with Town & State):", mse, "\n")
Random Forest MSE (with Town & State): 30098757900
> # Perform variable importance analysis
> varImpPlot(rf_model, main = "Variable Importance (with Town & State)")
```

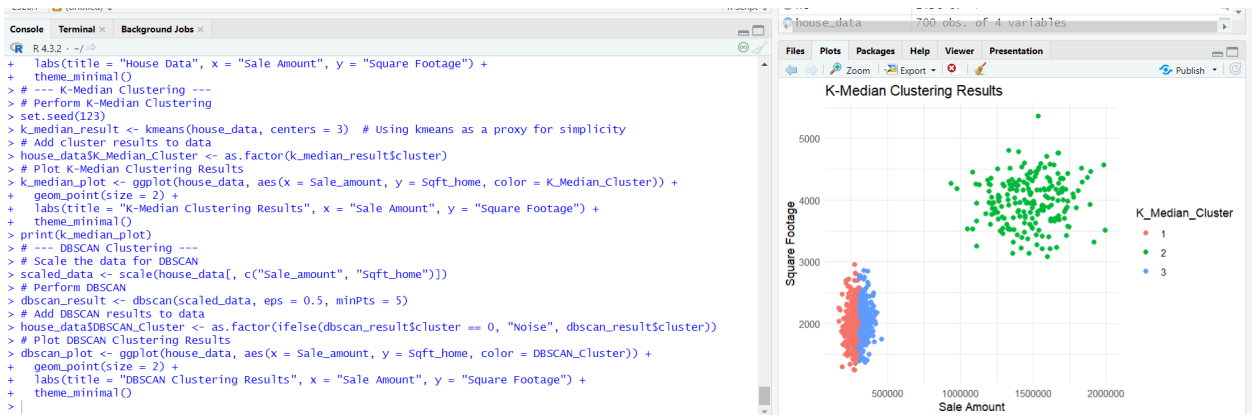
```
Call:
 randomForest(formula = Sale_amount ~ Beds + Baths + Sqft_home +      Build_year + Town + State, data = train_data, ntree
1000,      mtry = 3, importance = TRUE)
               Type of random forest: regression
                     Number of trees: 1000
No. of variables tried at each split: 3

          Mean of squared residuals: 32742253908
                    % Var explained: 71.51
```


```
Decision Tree RMSE: 212473.6
> cat("Decision Tree MAE:", dt_mae, "\n")
Decision Tree MAE: 120242.4
> cat("Decision Tree MSE:", dt_mse, "\n")
Decision Tree MSE: 45145013474
> # Calculate R-squared
> dt_sst <- sum((test_target - mean(test_target
> dt_sse <- sum((test_target - dt_predictions)^
> dt_r_squared <- 1 - (dt_sse / dt_sst)
> cat("Variance Explained (R-squared):", dt_r_s
Variance Explained (R-squared): 57.62095 %
```

```
Random Forest RMSE (with Town & State): 173489.936
> cat("Random Forest MAE (with Town & State):", mae, "\n")
Random Forest MAE (with Town & State): 75768.69108
> cat("Random Forest MSE (with Town & State):", mse, "\n")
Random Forest MSE (with Town & State): 30098757900
> # Perform variable importance analysis
> varImpPlot(rf_model, main = "Variable Importance (with Town & State)")
```





```
+   theme_minimal()
> # --- K-Median Clustering ---
> # Perform K-Median Clustering
> set.seed(123)
> k_median_result <- kmeans(house_data, centers = 3)  # Using kmeans as a proxy for simplicity
> # Add cluster results to data
> house_data$K_Median_Cluster <- as.factor(k_median_result$cluster)
> # Plot K-Median Clustering Results
> k_median_plot <- ggplot(house_data, aes(x = Sale_amount, y = Sqft_home, color = K_Median_Cluster)) +
+   geom_point(size = 2) +
+   labs(title = "K-Median Clustering Results", x = "Sale Amount", y = "Square Footage") +
+   theme_minimal()
> print(k_median_plot)
> # --- DBSCAN Clustering ---
> # Scale the data for DBSCAN
> scaled_data <- scale(house_data[, c("Sale_amount", "Sqft_home")])
> # Perform DBSCAN
> dbscan_result <- dbscan(scaled_data, eps = 0.5, minPts = 5)
> # Add DBSCAN results to data
> house_data$DBSCAN_Cluster <- as.factor(ifelse(dbscan_result$cluster == 0, "Noise", dbscan_result$cluster))
> # Plot DBSCAN Clustering Results
> dbscan_plot <- ggplot(house_data, aes(x = Sale_amount, y = Sqft_home, color = DBSCAN_Cluster)) +
+   geom_point(size = 2) +
+   labs(title = "DBSCAN Clustering Results", x = "Sale Amount", y = "Square Footage") +
+   theme_minimal()
> print(dbscan_plot)
>
```