## Project: Predicting House Sale Prices

### Introduction

The goal of this project is to build a machine learning model that accurately predicts house prices using the Ames Housing dataset. This dataset contains detailed information about residential homes in Ames, and is widely used for regression modelling practice. By analysing and modelling this data, we aim to identify the most influential features that affect house prices and create a robust predictive model.

**Objectives**

* Clean and preprocess the Ames Housing dataset.
* Perform exploratory data analysis (EDA) to understand data distribution and relationships.
* Engineer meaningful features to improve model performance.
* Train and compare two machine learning models: **Linear Regression** and **Random Forest Regressor**.
* Evaluate and interpret model results using appropriate metrics.

**Tools**

* **Python** (Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn)

### Section 1: Data Cleaning & Preparation

This section details the preprocessing steps taken to prepare the dataset for modelling.

#### 1.1. Data Understanding

To begin, it was essential to familiarize ourselves with the Ames Housing dataset. This involved using Python's Panda’s library to perform an initial exploration, including:

* Obtaining a list of all column headers and understanding their meaning.
* Checking the dimensions of the dataset to know the number of rows and columns.
* Identifying the data types of each variable to distinguish between numerical and categorical features.
* Analysing the descriptive statistics (median, mean, etc.) for all numerical variables to gain preliminary insights into their distribution.

#### 1.2. Identification and Removal of Irrelevant Variables

To prepare a clean and efficient dataset for modelling, several irrelevant variables were identified and removed. This was done by inspecting columns for high numbers of missing values and assessing their potential importance to the model. The Following columns were removed :

* **Order, PID**: These are unique identifiers with no predictive value for the house price.
* **Alley, Pool QC, Fence, Misc Feature, 3Ssn Porch**: These columns were removed due to a high percentage of missing values, indicating that the features they represent are rare in the dataset.
* **Utilities, Lot Config, Roof Style, Roof Matl, Heating QC, Central Air, Electrical, Functional, Fireplace Qu, Sale Type**: These variables have low variance or importance, and their removal helps simplify the model.

#### 1.3. Handling Missing Values

A specific strategy was applied to handle the remaining missing values in columns deemed important for the model.

* **Numerical Variables**: For columns like “**Lot Frontage**” with missing values, a replacement strategy was used. The missing values in Lot Frontage were imputed with the median Lot Frontage of their respective neighbourhood, as lot sizes tend to be similar within the same area. Other numerical columns with missing values, such as **Mas Vnr Area, BsmtFin SF 1, and Garage Cars,** were filled with 0 to indicate the absence of that particular feature.
* **Categorical Variables**: For quality-based categorical variables with missing values, a 'None' category was created. This was applied to columns such as **Bsmt Qual** and **Garage Type**, where a missing value signifies that the house does not have a basement or a garage.

#### 1.4. Encoding Categorical Variables

As Machine Learning models require numerical input, all categorical variables were converted into a numerical format.

* **Nominal Variables**: For variables without a natural order, such as Neighbourhood and House Style, **One-Hot Encoding** was applied. This method creates new binary columns for each unique category, preventing the model from assuming a false hierarchy.
* **Ordinal Variables**: For variables with a clear ranking, such as Exter Qual and Bsmt Qual, **Label Encoding** was used. This technique assigns numerical values that reflect the inherent order of the categories, for example, Excellent = 5, Good = 4, etc.

#### 1.5. Feature Engineering

Following the initial data preparation, several new variables were engineered to enrich the dataset and potentially enhance model performance. This process involved creating combined, interaction, and binary features :

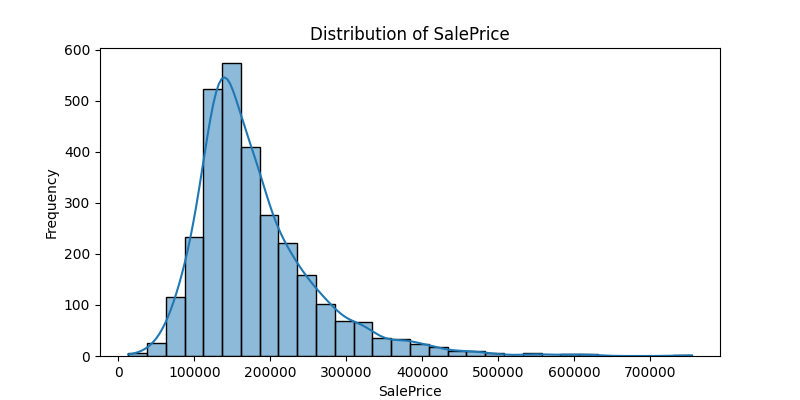
* **Combined Features**:
  + **TotalSF**: A combination of the living area, basement area, and second-floor area to represent the total surface area of the house.
  + **TotalBaths**: A weighted sum of all full and half bathrooms.
  + **TotalPorchSF**: The total square footage of all porches.
* **Time-based Features**:
  + **HouseAge**: The age of the house at the time of sale.
  + **SinceRemodel**: The number of years since the last remodel.
* **Interaction Features**:
  + **AreaQualInteraction**: An interaction feature created by multiplying the living area (Gr Liv Area) with the overall quality (Overall Qual) to capture the combined effect of size and quality.
  + **OverallScore**: A score combining the overall quality and overall condition.
* **Binary Features**:
  + New binary variables (HasPool, HasFireplace, HasGarage, HasBsmt, HasPorch) were created to indicate the presence or absence of a specific feature.
* **Ratios**:
  + **BsmtRatio** and **GarageRatio**: Ratios of the basement and garage area to the total living space.
  + **BathPerRoom**: The ratio of the total number of bathrooms to the total rooms above ground.

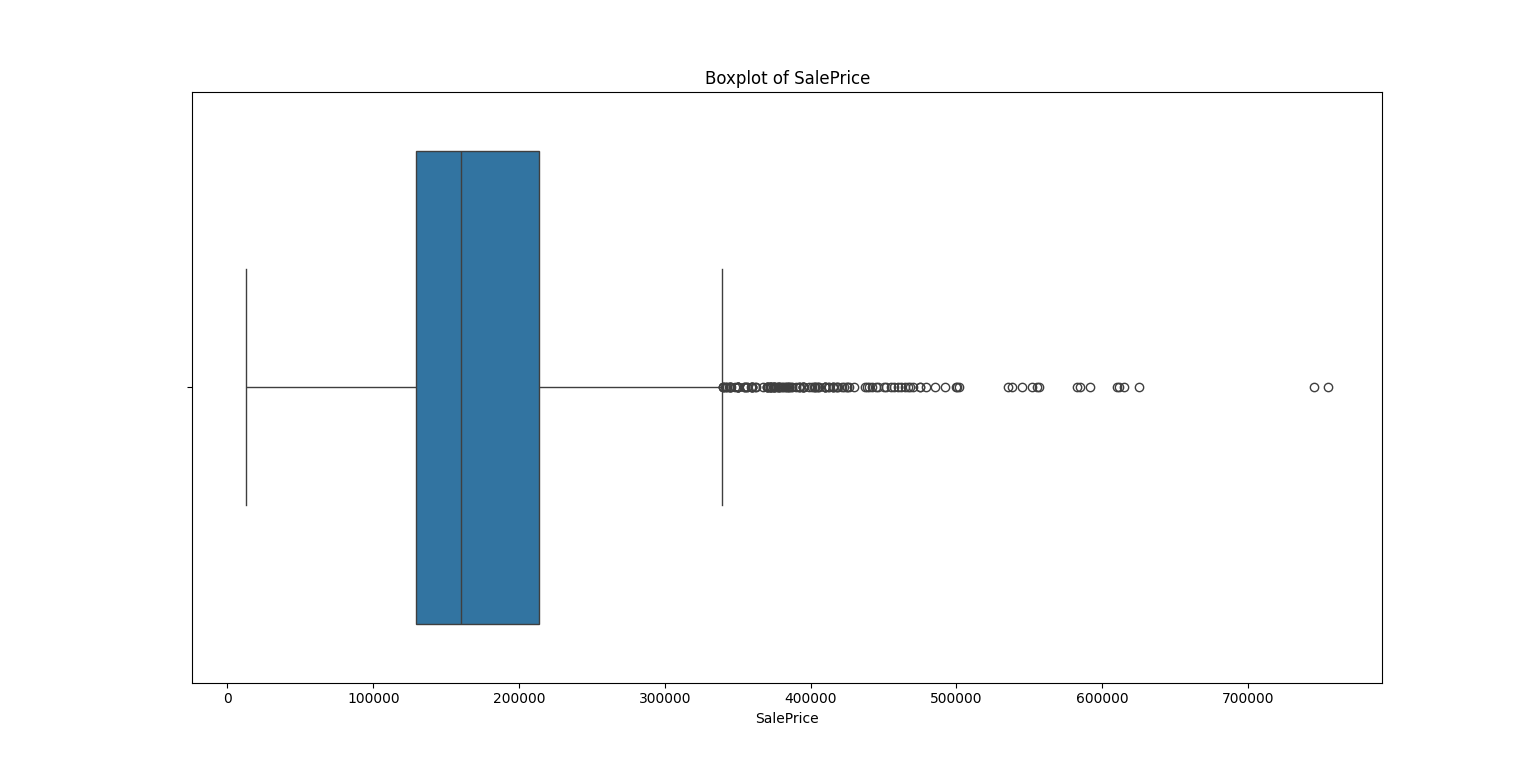
### Section 2: Exploratory Data Analysis (EDA)

EDA helped us to better understand the data distribution and the relationships between variables, guiding our modeling decisions.

#### 2.1. Analysis of the Target Variable (SalePrice)

Using libraries like matplotlib and seaborn, we visualized the distribution of our target variable, SalePrice, with a histogram and a box plot as shown below.

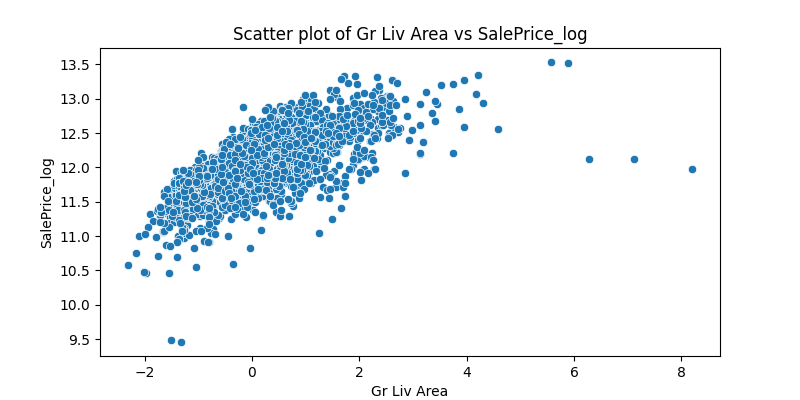




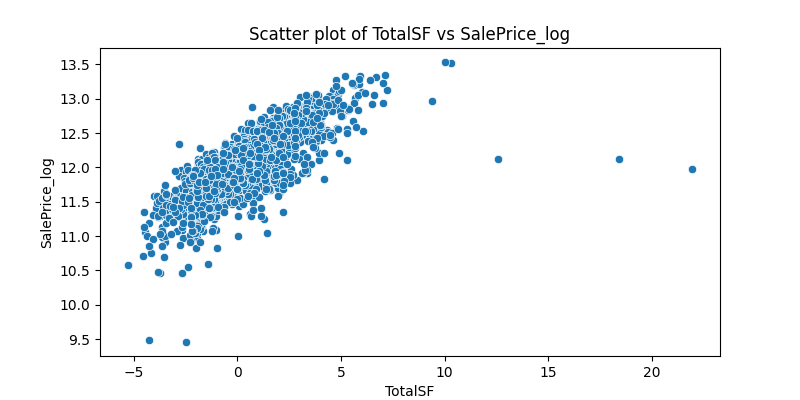
* **Histogram Analysis**: The histogram of SalePrice clearly showed a **strong positive skew**, with the majority of houses having lower prices and a long tail extending to the right due to a small number of very expensive properties. This skewed distribution is a common challenge for linear models, which assume a normal distribution of errors.
* **Box Plot Analysis**: The box plot for SalePrice reinforced the histogram's findings. The plot's long right whisker and numerous individual data points confirmed the presence of **outliers** with unusually high prices. The median line was also shifted to the left within the box, further indicating the positive skew.
* **Conclusion**: Based on these visualizations, a **logarithmic transformation** of SalePrice was a necessary step. This transformation makes the distribution more symmetrical (closer to a normal distribution), which improves the performance and reliability of regression models.

#### 2.2. Visualization of Relationships with Log-Transformed SalePrice

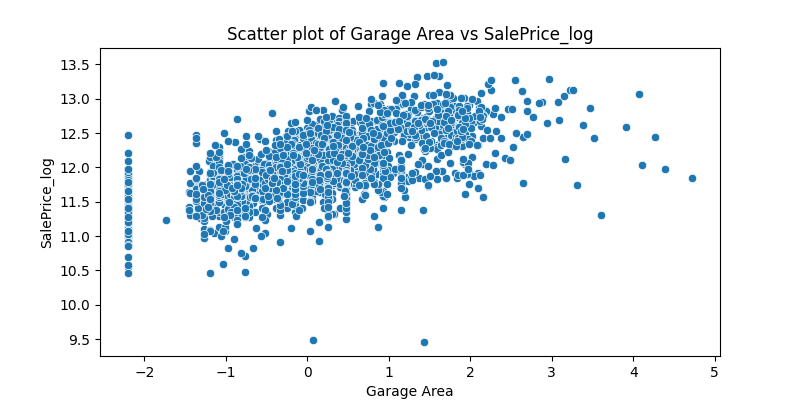
We then created scatter plots to analyse the relationships between key features and the log-transformed SalePrice.



* **Gr Liv Area vs. Log of SalePrice**: This plot showed a **strong, positive, and linear relationship**. The points are tightly clustered around a clear upward trend, demonstrating that Gr Liv Area is an excellent predictor of a house's price.



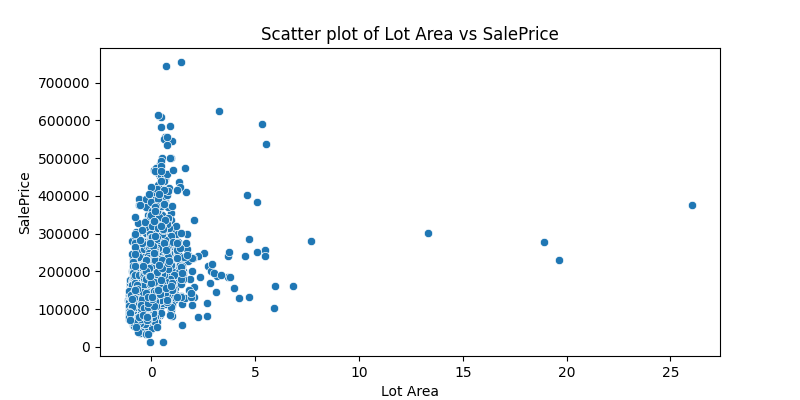
* **TotalSF vs. Log of SalePrice**: The scatter plot for the engineered TotalSF feature (total square footage including the basement) revealed an **exceptionally strong linear correlation**. This confirms that combining Gr Liv Area and Total Bsmt SF created a more powerful and consistent predictor than the individual variables alone.



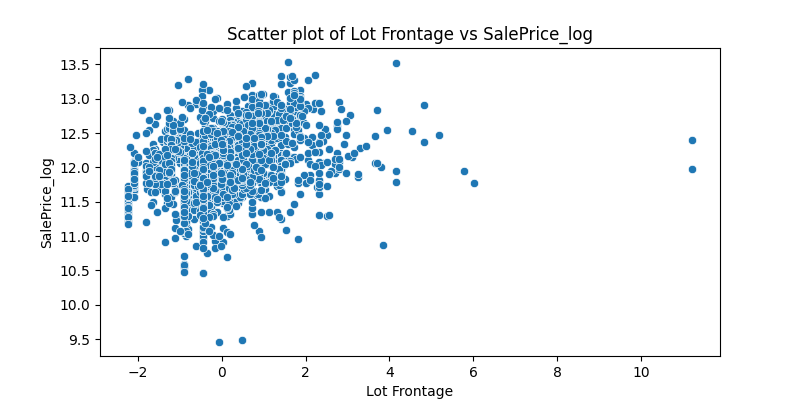
* **Garage Area vs. Log of SalePrice**: This plot also displayed a **strong positive correlation**, showing that a larger garage area is consistently associated with a higher log-transformed sale price.



* **Total Bsmt SF vs. Log of SalePrice**: The relationship here was a **clear positive correlation**, indicating that houses with larger basements generally sell for more.



* **Lot Area vs. Log of SalePrice**: In contrast, this plot showed a **weak positive relationship**. The data points were widely scattered, suggesting that while larger lots tend to increase value, Lot Area is not as reliable a single predictor as other variables.



* **Lot Frontage vs. Log of SalePrice**: Similar to Lot Area, this plot showed a **weak positive relationship** with significant scatter, confirming its lower predictive power.

#### 2.3. Correlation Matrix Analysis

To quantify these relationships, we computed a correlation matrix. The results confirmed our visual analysis and provided a more objective basis for selecting features. The following table shows the correlation of the log-transformed SalePrice with the most relevant variables, demonstrating the effectiveness of the logarithmic transformation.

|  |  |
| --- | --- |
| **Feature Name** | **Correlation with Log(SalePrice)** |
| Overall Qual | 0.8256 |
| TotalSF | 0.7822 |
| Gr Liv Area | 0.6959 |
| Garage Cars | 0.6749 |
| Garage Area | 0.6508 |
| TotalBaths | 0.6485 |
| Total Bsmt SF | 0.6256 |
| Year Built | 0.6155 |
| 1st Flr SF | 0.6026 |
| Year Remod/Add | 0.5862 |
| Full Bath | 0.5773 |
| Foundation\_PConc | 0.5443 |
| HasGarage | 0.5208 |
| HasBsmt | 0.5185 |
| HasFireplace | 0.5092 |
| Exter Qual\_Gd | 0.5001 |
| TotRms AbvGrd | 0.4926 |
| Fireplaces | 0.4889 |
| BsmtFin Type 1\_GLQ | 0.4574 |
| Mas Vnr Area | 0.4430 |
| Garage Type\_Attchd | 0.4130 |
| BsmtFin SF 1 | 0.4111 |

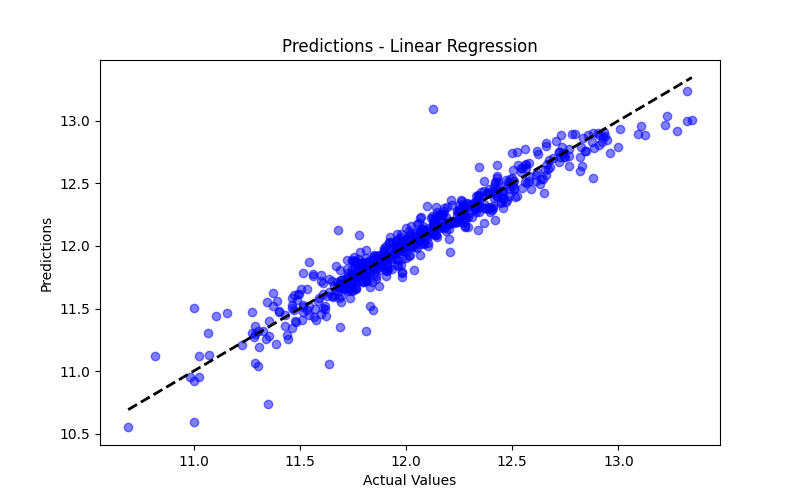
### Section 3: Model Building & Training

Two regression models were built to compare their performance. The following steps were taken to accomplish this:

* **Data Splitting**: The dataset was divided into training and testing sets, typically with an 80/20 split. The training set is used to train the model, while the testing set is used to evaluate its performance on unseen data.
* **Model Instantiation**: The two models, a Linear Regression model and a Random Forest Regressor model, were instantiated using the scikit-learn library.
* **Model Training**: Both models were trained using the fit() method on the training data.
* **Prediction**: The trained models were then used to make predictions on the test set's feature data using the predict() method.

#### 3.1. Model 1: Linear Regression

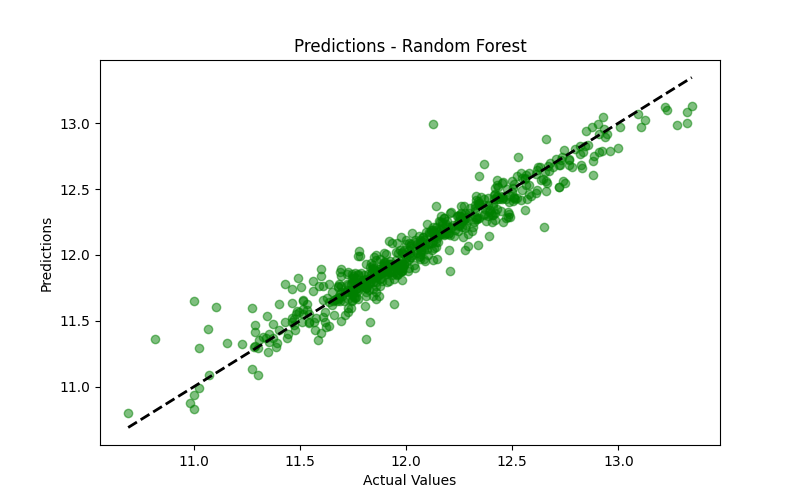
**Linear Regression** is a simple model that seeks to find a linear relationship between features and the target variable. It serves as an excellent baseline model due to its speed and ease of interpretation.



The plot above shows the predictions of the Linear Regression model against the actual values. The closer the points lie to the dashed black line, the more accurate the predictions are. The dense clustering of points around the line indicates that the model has a strong fit, which is further evidenced by its high R² score.

#### 3.2. Model 2: Random Forest Regressor

**Random Forest** is an ensemble model that builds multiple decision trees and aggregates their predictions. It is robust, capable of capturing non-linear relationships, and resistant to overfitting.



The scatter plot for the Random Forest Regressor's predictions shows an even tighter clustering of points along the dashed black line compared to the linear model. This visual analysis confirms that the Random Forest model is highly accurate, with its predictions closely matching the actual sale prices and slightly outperforming the simpler Linear Regression model.

### Section 4: Model Evaluation & Interpretation

Both models were evaluated on the test set using **RMSE** and the **R² score**.

#### 4.1. Results Obtained

|  |  |  |
| --- | --- | --- |
| **Model** | **RMSE (Root Mean Squared Error)** | **R² Score (Coefficient of Determination)** |
| Linear Regression | 0.1199 | 0.9223 |
| Random Forest | 0.1188 | 0.9237 |

#### 4.2. Interpretation of Results

* **Overall Performance**: Both models showed excellent performance, with R² scores above 0.92, which means they explain over 92% of the variance in sale prices.
* **Model Comparison**: The **Random Forest Regressor** slightly outperformed Linear Regression on both metrics. Its RMSE is lower and its R² score is higher. This result was expected, as Random Forest is better suited to capture the complex, non-linear relationships inherent in real estate pricing data.

#### 4.3. Analysis of Top Important Features

Analyzing the top features for each model provides insight into how each model makes its predictions.

**Top 15 Important Features (Linear Regression):**

|  |  |
| --- | --- |
| **Feature Name** | **Coefficient/Weight** |
| MS Zoning\_I (all) | 1.142461 |
| MS Zoning\_RH | 0.923025 |
| MS Zoning\_FV | 0.911790 |
| MS Zoning\_RL | 0.894828 |
| MS Zoning\_RM | 0.852550 |
| MS Zoning\_C (all) | 0.674567 |
| Exterior 1st\_CBlock | 0.610077 |
| Garage Qual\_Po | 0.584152 |
| Neighborhood\_GrnHill | 0.557756 |
| Garage Qual\_Fa | 0.536805 |
| Garage Qual\_TA | 0.520193 |
| Garage Qual\_Gd | 0.418879 |
| Mas Vnr Type\_CBlock | 0.368754 |
| House Style\_1.5Unf | 0.358469 |
| Garage Cond\_TA | 0.343067 |

The top features for the Linear Regression model are dominated by **categorical dummy variables**, particularly related to **zoning (MS Zoning) and garage quality**. The large positive coefficients for features like MS Zoning\_I (all) indicate that being in that specific zoning classification has a significant positive impact on the predicted sale price. This shows that the linear model gives high importance to specific categories.

**Top 15 Important Features (Random Forest):**

|  |  |
| --- | --- |
| **Feature Name** | **Importance Score** |
| Overall Qual | 0.438587 |
| TotalSF | 0.257729 |
| AreaQualInteraction | 0.075172 |
| TotalBaths | 0.023386 |
| OverallScore | 0.013499 |
| Year Built | 0.012105 |
| Lot Area | 0.011702 |
| Garage Cars | 0.010432 |
| Year Remod/Add | 0.010253 |
| Garage Yr Blt | 0.010171 |
| BsmtFin SF 1 | 0.008910 |
| Garage Area | 0.008763 |
| Gr Liv Area | 0.007316 |
| Bsmt Unf SF | 0.005445 |
| Total Bsmt SF | 0.004694 |

In contrast, the **Random Forest** model's most important features are primarily **numerical and ordinal variables**, such as Overall Qual and the engineered TotalSF. This is expected, as tree-based models excel at capturing complex, non-linear relationships. The importance scores are based on how much each feature contributes to the reduction of impurities in the decision trees. The high scores for Overall Qual and TotalSF indicate these two variables are by far the most significant drivers of the final house price prediction for the Random Forest model.