House Price Prediction Using Structured Data and Satellite Imagery

Dheerajpreet Singh (23322012)

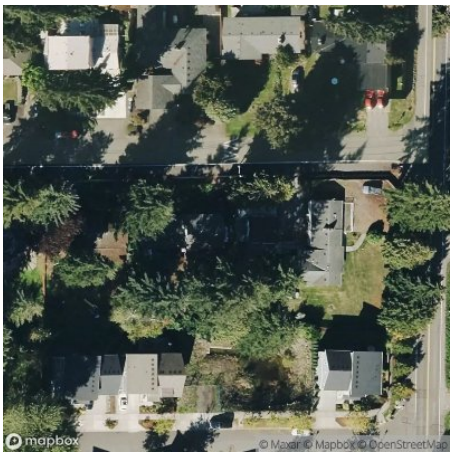
1. Introduction

We predict house prices using both tabular features and satellite imagery. Image features extracted with EfficientNet were compressed using PCA and combined with structured features to train an XGBoost model. Results show improved accuracy and interpretable insights through SHAP and Grad-CAM.

2. Dataset Description

The dataset consists of two complementary sources of information: structured tabular housing data and satellite imagery associated with each property. Combining these two views allows the model to capture both property-specific attributes and contextual information about the surrounding neighborhood and environment.

- *Structured Data* -
  The structured dataset contains numerical and categorical attributes describing each property. Each row represents one house, and each column corresponds to a specific feature. Key variables include:
    - *price* – target variable representing the sale price.
    - *sqft_living* – interior living area in square feet.
    - *bedrooms* – number of bedrooms.
    - *bathrooms* – number of bathrooms.
    - *sqft_lot* – lot size.
    - *floors* – number of stories.
    - *grade* / *condition* – qualitative measures of construction quality and overall state of the property.
    - *zipcode* – geographic region identifier.
    - *lat, long* – latitude and longitude coordinates for mapping and image retrieval.
    - These features provide direct and interpretable information about each property. Missing values, duplicates, and outliers were inspected and handled appropriately during preprocessing.
- *Satellite Images* -
  In addition to tabular features, satellite images were collected for each property using the **Mapbox Static Maps API**. Latitude and longitude from the structured dataset were used to download high-resolution aerial images centered on each house location.



Each image is linked to a property through its unique ID, ensuring consistent alignment between visual and tabular features. During image collection, some coordinates produced corrupted or invalid downloads. These were identified and recorded separately (bad image ID files) so they could be excluded or replaced without interrupting the pipeline. Duplicate property IDs with different timestamps were mapped to the same image to avoid unnecessary downloads and computation.

These satellite images introduce valuable spatial context — neighborhood characteristics, density, greenery, and surrounding infrastructure — which are often missing from structured real-estate datasets.

3. Data Preprocessing and EDA

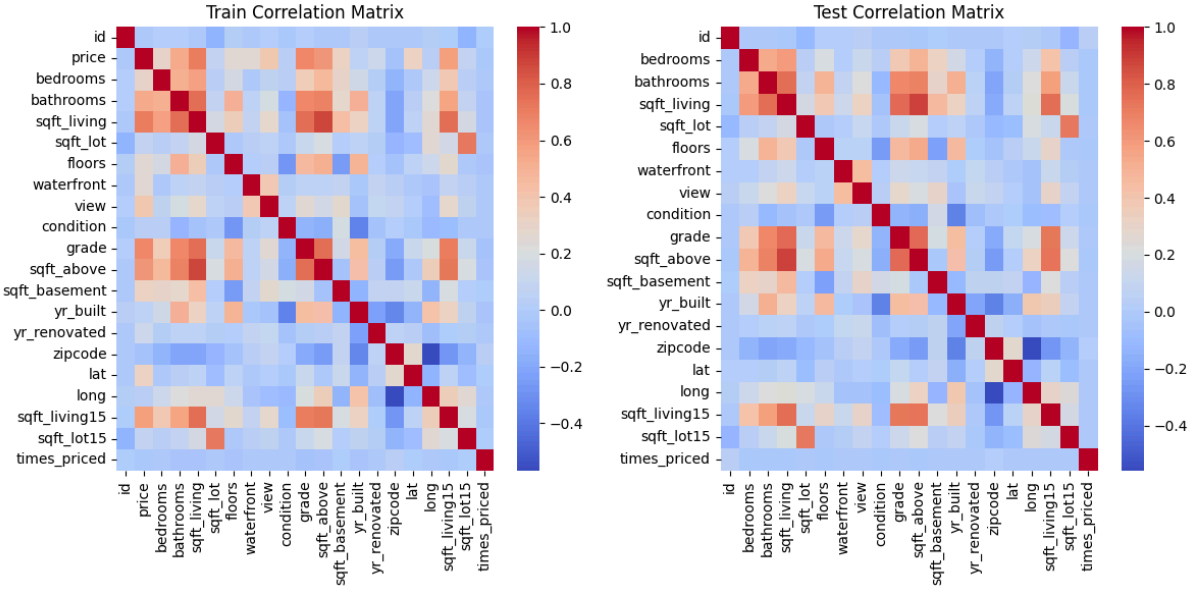
This section describes the preprocessing steps applied to the dataset and the exploratory data analysis (EDA) conducted to understand feature behavior, detect anomalies, and guide modeling decisions.

- *Missing Values and Duplicates*

Although the dataset did not contain missing values, we observed several duplicated property IDs in both the training and test data. These duplicates were not errors; instead, they represented the same house recorded at different dates, meaning the property had been priced multiple times.

Rather than removing these rows, we preserved them and introduced a new feature called times_priced, which indicates how many times a property has appeared in the dataset and which occurrence a specific record corresponds to. Single-occurrence properties were assigned a value of 0, while repeated listings were numbered sequentially based on date. Since the underlying property does not change, the same satellite image was reused for duplicate IDs, avoiding unnecessary downloads. This approach allowed us to retain valuable temporal information while keeping the dataset consistent and computationally efficient.

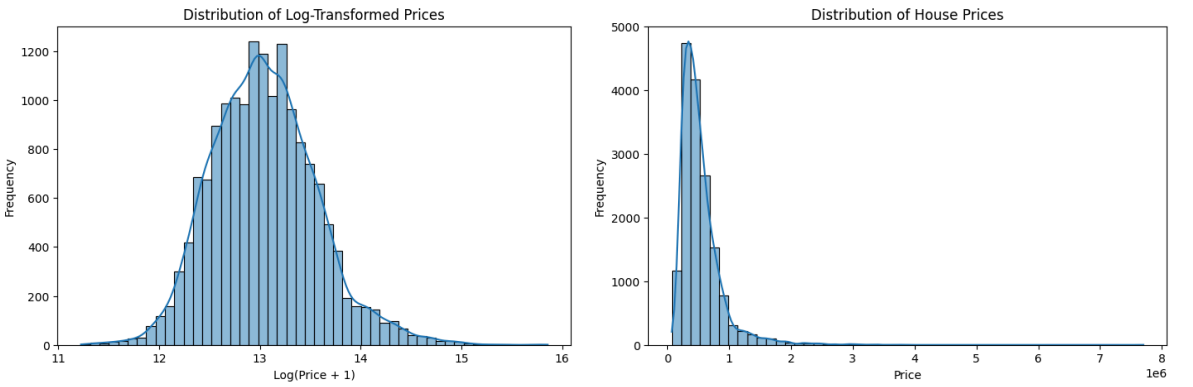
- *Correlation Heatmap*



The correlation analysis shows that interior size features — especially sqft_living and sqft_above — have the strongest positive relationship with price, indicating that bigger living areas drive value more than lot size. Quality indicators such as grade and condition also correlate well with price, while lot-related variables contribute much less.

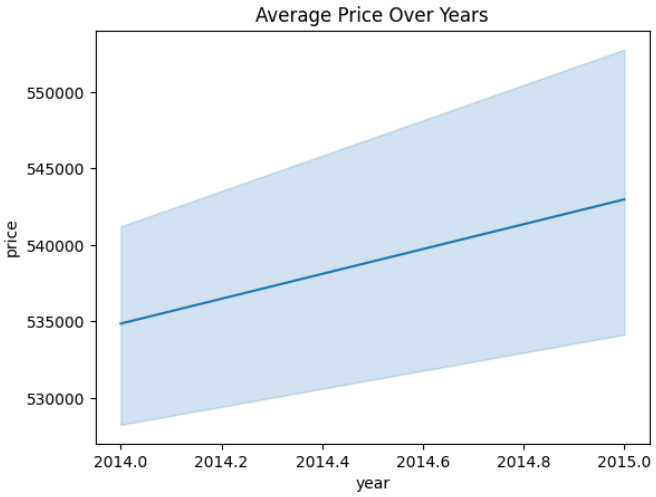
Geographic attributes, particularly latitude, show moderate correlation, reflecting price differences across neighborhoods. Many space-related variables are highly inter-correlated, revealing redundancy in the dataset. Importantly, the correlation structure is consistent across train and test sets, suggesting no distribution mismatch, and the id field behaves strictly as an identifier with no predictive influence.Missing Values and Duplicates
Although the dataset did not contain missing values, we observed several duplicated property IDs in both the training and test data. These duplicates were not errors; instead, they represented the s

- *Log Transformation of Prices*



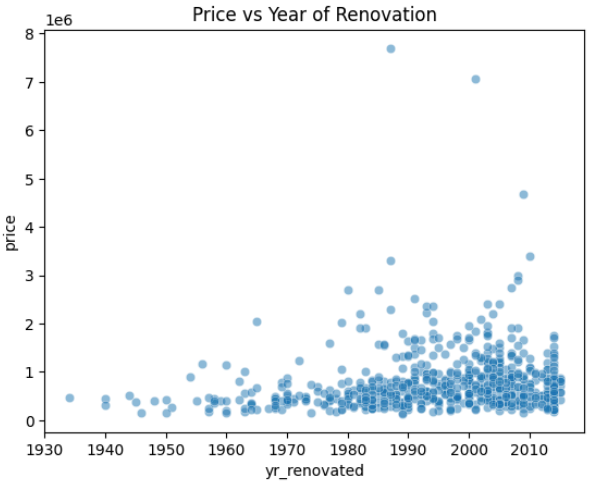
The histogram of house prices shows a strong right-skewed distribution, with most houses clustered at lower prices and a long tail of very expensive properties. Such skewed data can make regression models unstable and bias the learning process. To normalize the distribution and stabilize variance, we apply a log transformation to the target variable price before training the model.

- *Average House Price Trend Over Time*



The line plot shows the trend in average house prices over the available years. There is a clear upward movement, indicating that housing prices were gradually increasing over time. The shaded region represents the variation around the mean, showing that although individual prices fluctuate, the overall market trend is consistently rising. This suggests that time-related features (such as year or date of sale) can provide useful information for the model.

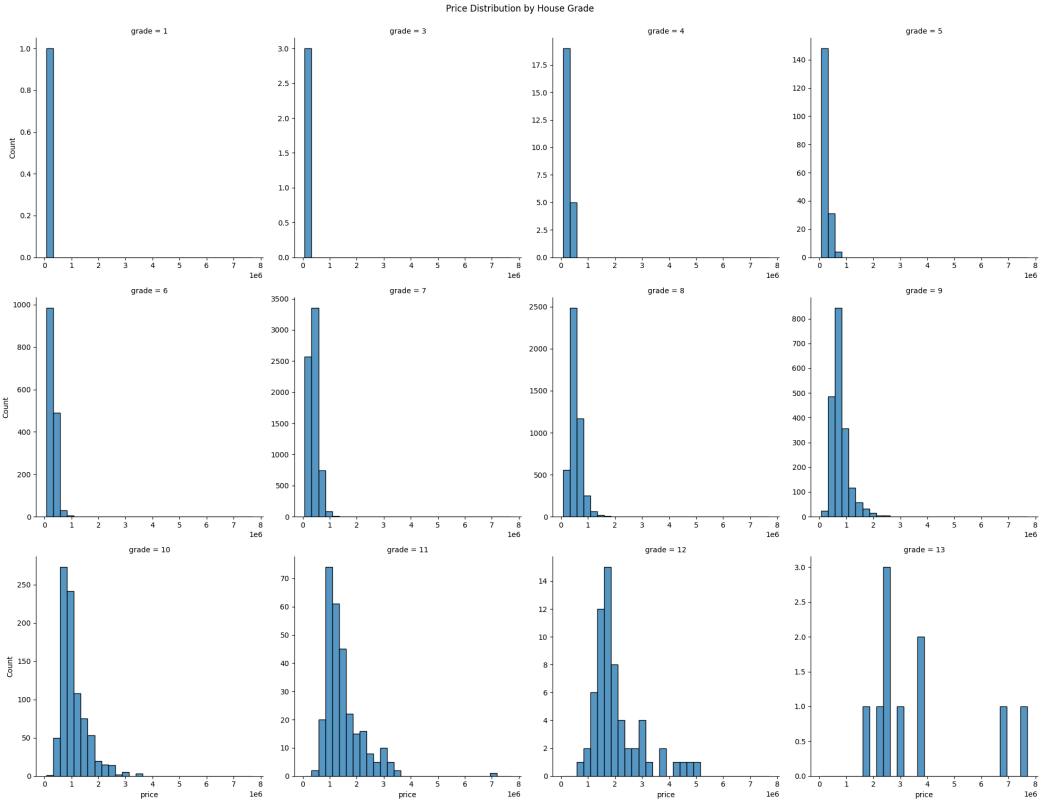
- *Price vs Year of Renovation*



This scatter plot shows the relationship between renovation year and house price. More recently renovated homes tend to cluster at higher price levels, while older or never-renovated properties generally sell for less. However, the spread is wide,

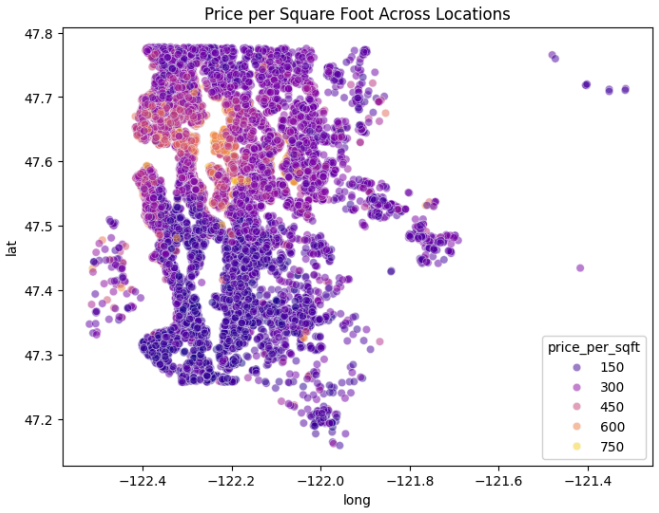
indicating that renovation year alone does not determine price but contributes as one of several price-influencing factors.

- *Price Distribution Across Different House Grades*



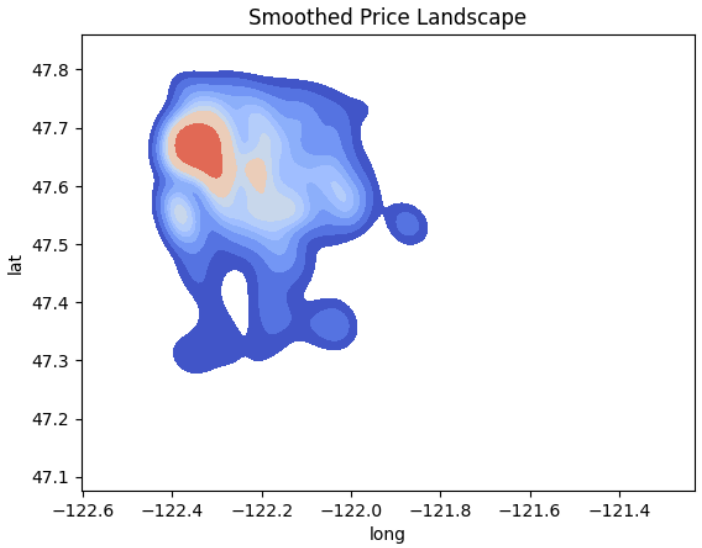
This grid of histograms shows how house prices are distributed within each grade category. Lower-grade homes are concentrated at the lower end of the price range, while higher-grade homes exhibit progressively higher price levels and wider spreads. The shift in distributions across grades confirms that the grading system strongly reflects overall property quality and is highly relevant for predicting price.

- *Price per Square Foot Across Locations*



This map colors each home by price per square foot. We observe pockets of significantly higher per-square-foot values, highlighting premium neighborhoods even when overall house sizes differ.

- *Smoothed Price Landscape*



This density heatmap smooths house prices across geography. Hotter (red/orange) zones correspond to consistently high-priced regions, clearly showing premium market pockets.

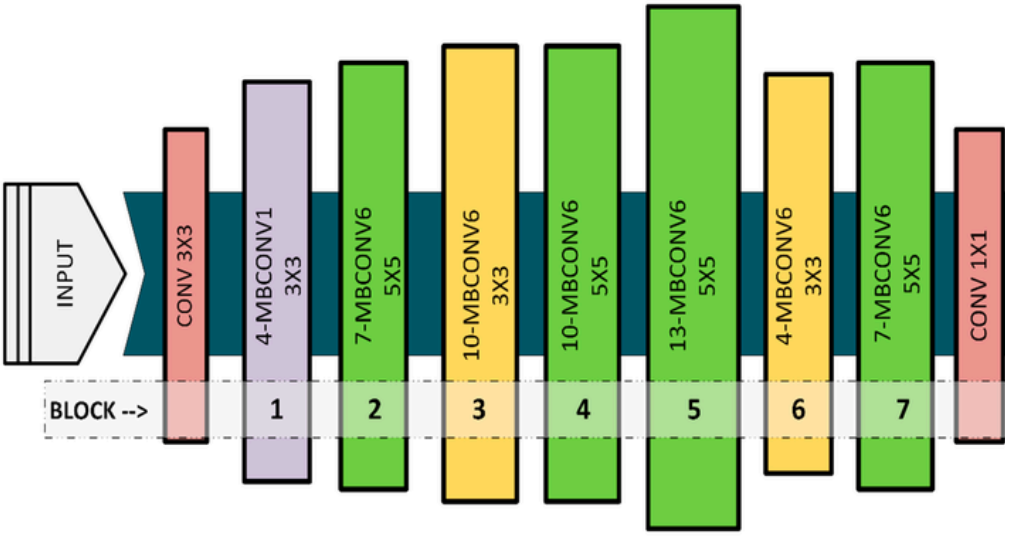
- *More such interesting finds..*

I've included additional EDA insights in preprocessing.ipynb—feel free to explore that notebook for a deeper dive into the data.

4. Image Feature Extraction

In addition to structured property attributes, this project incorporates satellite imagery for every house location. The goal is to capture neighborhood and visual context that may not be explicitly present in tabular variables — such as surroundings, density, greenery, roads, and construction style.

Each property image is downloaded using the Mapbox Static API, centered at the property's latitude/longitude coordinates. After removing failed or corrupted downloads, every valid image is processed using a pretrained EfficientNet-B7 convolutional neural network. Instead of classifying the image, the network is used as a feature extractor: the final fully connected classification layer is removed, and the model outputs a high-dimensional embedding that summarizes the visual content of the image.

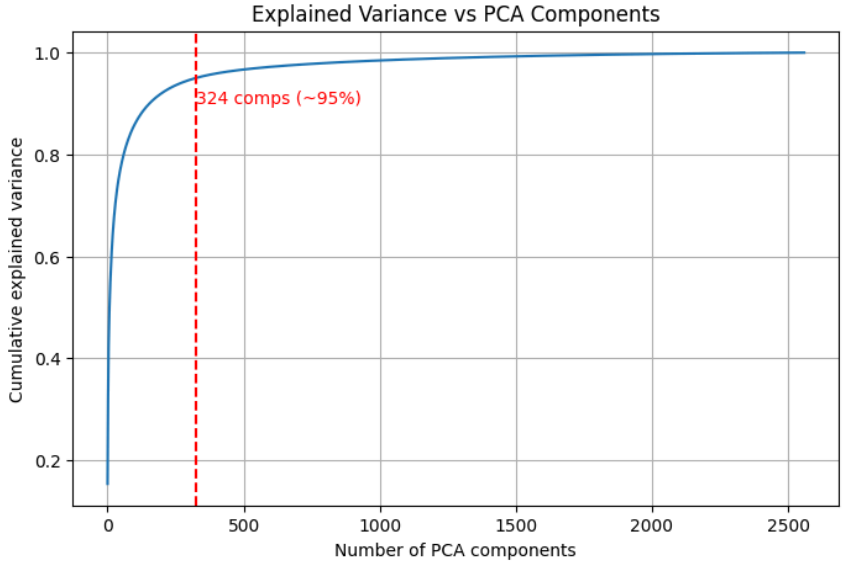


These extracted embeddings become new numerical features which are later combined with the structured dataset for modeling. However, since the resulting feature vectors are very high-dimensional (thousands of columns), Principal Component Analysis (PCA) is applied to compress them while retaining most of their variance. This allows the model to benefit from image information without significantly increasing dimensionality or computation cost.

- *PCA on the extracted data from Images*

The extracted image/derived features contained a very large number of columns, many of which were highly correlated or contributed very little unique information. Training models directly on such high-dimensional data can lead to: slower training, higher memory usage, and increased risk of overfitting.

To address this, we applied Principal Component Analysis (PCA) to the extracted feature matrix in the modelling notebook. PCA transforms the original features into a smaller set of principal components that retain most of the total variance (information) in the data while removing redundancy.



The plot above shows how much total variance is captured as we increase the number of PCA components. Each point represents the cumulative variance explained by the first k components.

We also draw a vertical dashed red line at the point where cumulative variance first exceeds ~95%. This helps us visually choose a good dimensionality: we keep enough components to retain most of the information, while discarding redundant or noisy ones. In our case, around ~324 components capture roughly 95% of the variance, which dramatically reduces dimensionality compared to the original image feature space.

After fitting PCA, we project both the train and test image feature matrices into the reduced 325-dimensional space

5. Modeling Methodology

We begin by integrating the extracted features with our structured data to prepare the final dataset for model training.

- The dataset was split into **80% training** and 20% validation.
- Shuffling and a fixed random seed ensured **reproducible results**.
- The validation set was used only for evaluation, preventing leakage.
- A modeling pipeline was created using:
    - ***StandardScaler*** to normalize numeric features.
    - ***XGBoost Regressor,*** chosen for strong performance on structured data.
- Scaling ensured that **features with larger ranges** did not dominate learning.
- Hyperparameter tuning was performed using **RandomizedSearchCV**:
    - Randomly samples parameter combinations instead of testing all possibilities.
    - Provides a good balance between performance and computation time.
    - Uses **3-fold cross-validation** for reliable evaluation.
    - Optimizes the model using **negative RMSE** (lower error = better model).
- Tuned parameters included:
    - Tree depth, learning rate, and number of trees.
    - Subsampling ratios to prevent overfitting.
    - Regularization terms to improve generalization.
- The search evaluated 40 candidate configurations in parallel.
- The best-performing configuration was selected automatically.
- This process produced a model that is:
    - Well-regularized
    - Evaluated on unseen data

The tuning process achieved a best cross-validated RMSE of ~126,184, indicating the typical average difference between predicted and actual house prices on unseen folds. Because this value is based on cross-validation, it provides a trustworthy estimate of real-world performance.

The selected hyperparameters describe a model that is intentionally conservative and regularized. A low learning rate (0.03) and shallow trees (max depth = 4) help prevent overfitting, while subsample = 0.7 and colsample_bytree = 0.7 add randomness to make the model more robust. Regularization is mainly handled through L2 (lambda = 1), while L1 regularization was not needed. Altogether, this configuration balances accuracy and stability rather than pushing for an overly complex model.

6. Evaluation Metrics

Model	RMSE	R-Squared
XGBoost (with extracted features) (with PCA) (Without Log Transformation)	118328.7223289426	0.8884226679801941
XGBoost (with extracted features) (with PCA) (Log Transformation)	115537.87784099205	0.8936238288879395
XGBoost (without extracted features)(Without Log Transformation)	112330.71629790313	–
XGBoost (without extracted features)(Log Transformation)	116901.86462157051	0.8910973072052002
Elastic Net Regression (With PCA) (Log T)	15836.3377482	–
Elastic Net Regression (Without PCA) (Log T)	18038.28748486284	–

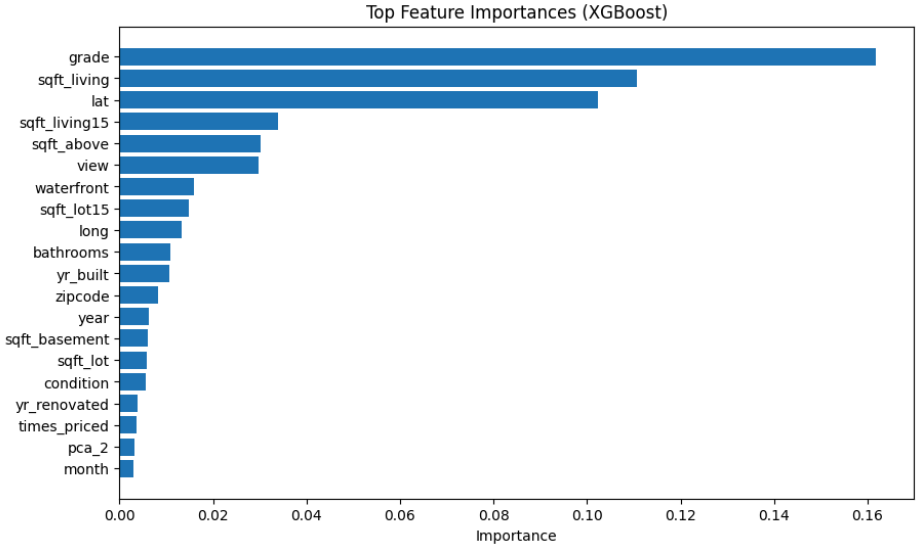
The model results are somewhat confusing because improvements are not consistent across configurations. Adding image features and PCA does not always reduce error, suggesting that the structured variables already capture most of the predictive signal. PCA may also remove useful information, since XGBoost can naturally handle high-dimensional inputs. Log-transformed models show better performance in some cases but not others, likely due to differences between log-scale and real-price RMSE. Finally, Elastic Net reports unusually low RMSE values, indicating that metrics may not be directly comparable across models.

Overall, these results show that multimodal modeling is sensitive to preprocessing choices, and fair comparison requires careful alignment of evaluation methods.

7. Model Explainability

After building and evaluating the predictive models, the next step was to understand why the models make certain predictions and which features influence house prices the most. Instead of treating the model as a black box, we use explainability techniques such as feature importance analysis, SHAP values, and Grad-CAM (for image features). These methods help reveal which aspects of a property — location, structure, or visual characteristics — drive the price predictions, allowing us to interpret, validate, and trust the results.

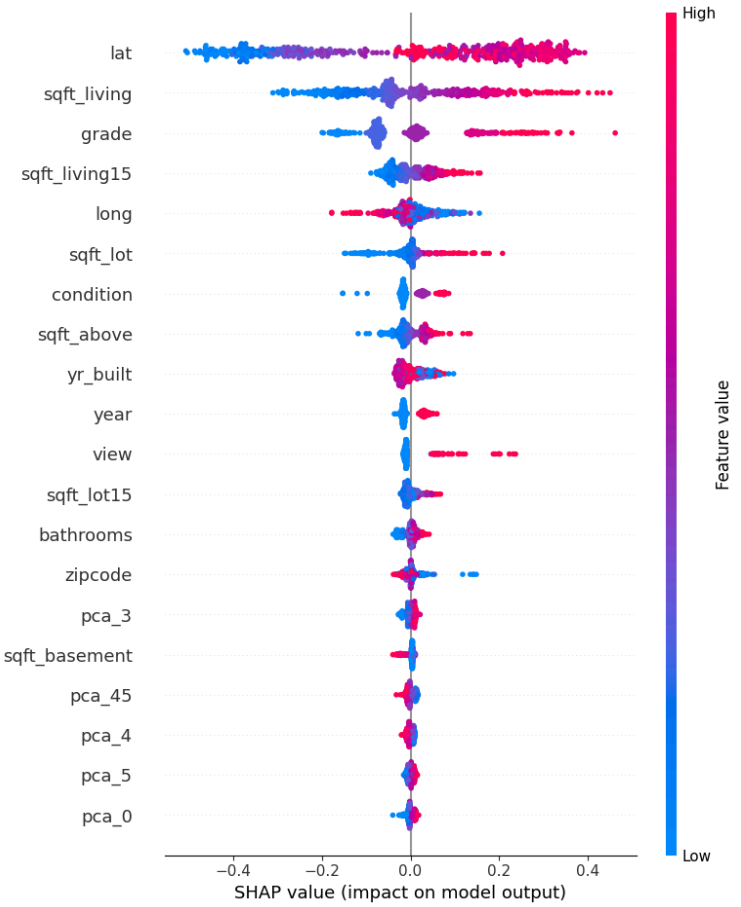
- *Feature Importance/ SHAP*



This plot shows which features contributed the most to the XGBoost model's predictions. Features such as grade, sqft_living, and location coordinates (lat/long) have the highest importance, meaning the model relies on them heavily when estimating prices.

Lower-ranked features still contribute, but removing them would likely have less impact on performance.

Overall, this helps verify that the model is learning meaningful relationships (larger homes, better quality, and desirable locations generally predict higher prices).



This SHAP summary plot explains how each feature pushes the predicted price up or down. Each dot represents a single house, and the color indicates whether the feature value is low (blue) or high (red).

- Features like latitude, living area (sqft_living), and grade show strong influence higher values generally push predictions higher.
- Some features have both positive and negative effects depending on context, showing non-linear relationships.
- PCA components appear, but their effect is indirect (capturing image-based patterns).

Overall, this visualization confirms that the model is learning intuitive behavior: larger, higher-grade homes in desirable locations are predicted to cost more.

- *Grad-CAM*

To understand what parts of the satellite images the CNN considers important, we applied Grad-CAM on the EfficientNet-B7 feature extractor. Each column in the figure shows a property image (top) and its corresponding Grad-CAM heatmap (bottom). Warmer colors (red/yellow) indicate regions that contribute more strongly to the predicted price, while cooler regions (blue) have little influence.

Across the examples, the network consistently focuses on built structures, roof density, and surrounding road networks, rather than vegetation or empty land. For suburban neighborhoods, areas with tightly clustered houses and visible infrastructure receive higher attention. Conversely, properties surrounded by open land or forest show weaker activations, suggesting the model implicitly associates development density with higher property value. These visualizations demonstrate that the model is not merely memorizing pixels but is learning meaningful spatial cues related to urbanization and housing characteristics.

8. Conclusion

This project shows that combining structured housing data with satellite-image features can improve house price prediction, while also offering richer insight into neighborhood context. Core variables such as living area, grade, and latitude remained the strongest drivers, but image features added complementary information about density, surroundings, and urban layout. PCA proved essential for compressing high-dimensional image embeddings, and XGBoost — especially with log-transformed prices — delivered the best balance of accuracy and stability. Model explainability using SHAP and Grad-CAM confirmed that the models focused on meaningful patterns rather than artifacts. Overall, satellite images enhance prediction performance, but they act best as a supporting signal alongside traditional real-estate features, suggesting future work should refine multimodal integration and expand testing across broader geographies.