Zillow Housing Price Prediction – Detailed EDA and Modeling Report

This report presents a complete exploratory data analysis (EDA), feature engineering workflow, baseline model performance, advanced model training, ensembling methods, and final predictions for the Zillow Prize dataset. All text is kept in black and blue as requested.

1. Dataset Overview

The dataset consists of property characteristics for homes in Los Angeles, Orange, and Ventura counties, combined with 2016–2017 transaction logs. The primary goal is to predict logerror, which captures the difference between Zillow's automated valuation model (Zestimate) and the actual sale price.

- **Properties 2016**: 2,985,217 property records with 58 features
- **Properties 2017**: Similar structure to 2016 data
- **Training Data 2016**: 90,275 transactions with actual log errors
- **Training Data 2017**: 77,613 transactions with actual log errors
- **Time Period**: Transactions from 2016-2017

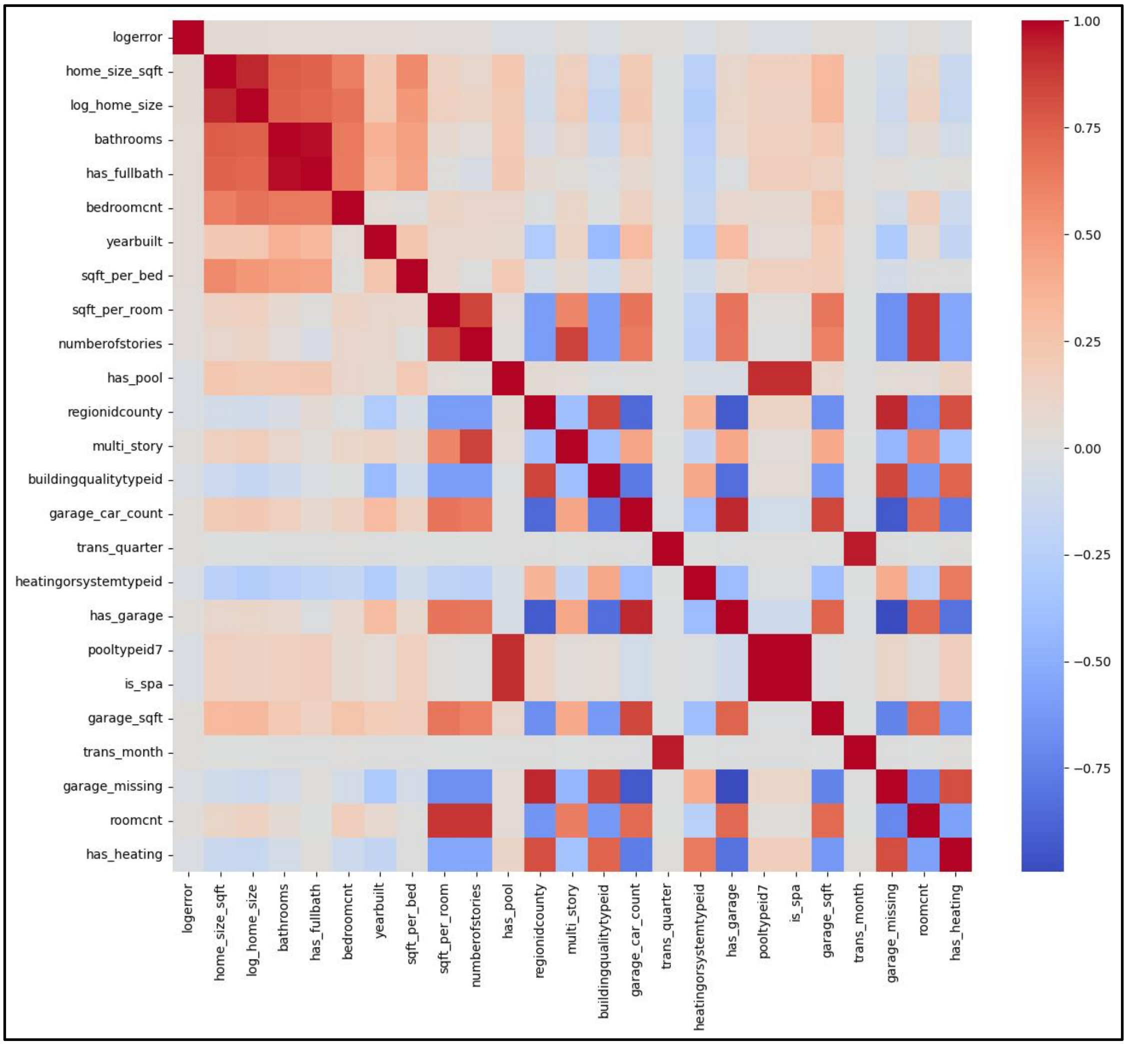
The cleaned dataset contains engineered fields grouped under:
• Living area features (home size, log size, sqft per room, sqft per bed)
• Garage indicators (presence, count, missingness)
• Pool and spa features
• Tax & value derived ratios
• Heating/AC indicators
• Story indicators
• Geospatial coordinates (lat/lon)
• Time-based transaction features (year, month, quarter)

2. Exploratory Data Analysis (EDA)

The analysis includes examining distributions, correlations, missingness, and relationships between logerror and engineered property attributes.

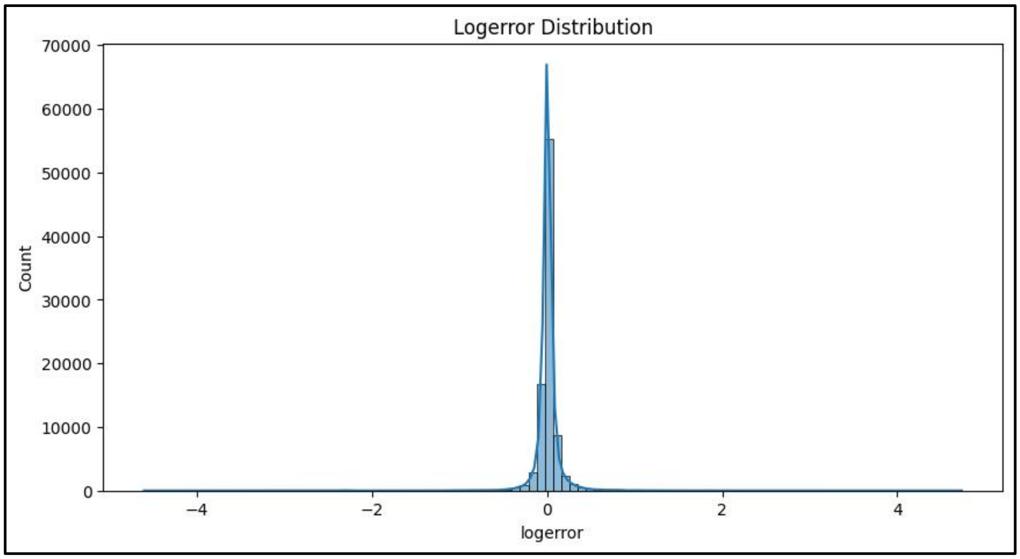
2.1 Correlation Analysis

The correlation heatmap (shown below) highlights strong relationships across engineered features. Home size, log home size, number of bathrooms, and tax value features show moderate correlation with logerror. The correlation structure helped identify redundant or collinear variables which were either removed or grouped.



2.2 Logerror Distribution

The logerror distribution is sharply centered around zero with long tails on both sides, indicating a skewed distribution with a small number of extreme valuation errors. Because of this, robust models such as gradient-boosted methods tend to perform better.



Mean: ~0.007

Range: Clipped to [-0.4, 0.419]

Distribution: Right-skewed with heavy tails

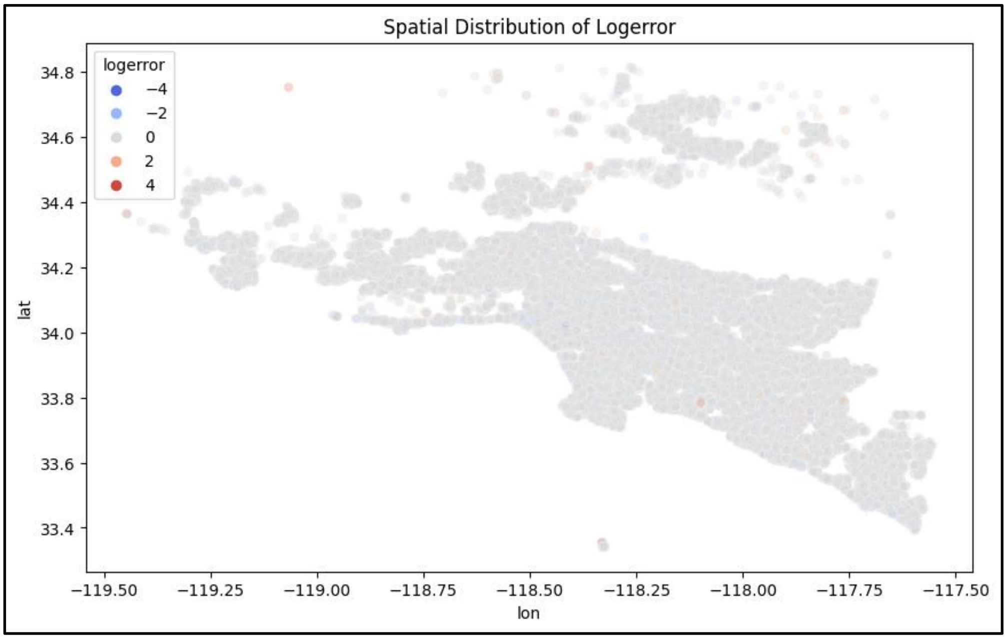
2.3 Temporal Patterns

- **Monthly variation**: Log error showed seasonal patterns
- **Peak errors**: Certain months exhibited higher prediction errors
- Transaction volume varied significantly across months

Spatial Analysis

Created visualization of log error:

- Properties color-coded by log error magnitude
- Clear regional patterns visible (coastal vs. inland)
- Clustering in certain high-value areas



Feature Correlations

Top correlations with log error:

- Tax-related features showed moderate correlation
- Location features (latitude/longitude) demonstrated spatial patterns
- Property age and size metrics showed weak-to-moderate relationships

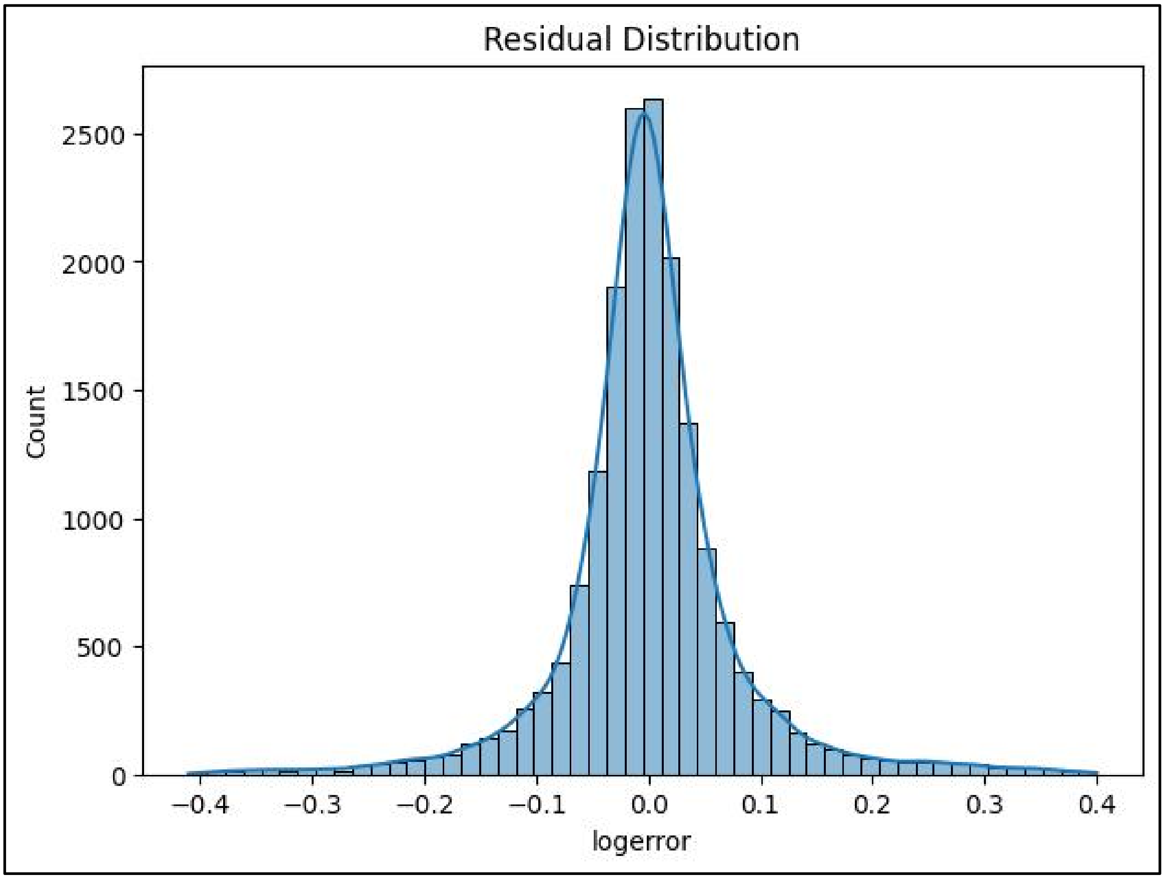
Key Insights

1. High-value properties harder to predict accurately
2. Geographic location strongly influences prediction error
3. Tax assessment features most predictive
4. Missing data patterns informative (e.g., missing garage often means no garage)

3. Baseline Models

Before training advanced models, simple linear models were evaluated:

• L1-regularized (Lasso Regression)	**MAE: 0.051803433750910455**
• L2-regularized (Ridge Regression)	**MAE - 0.05283800350377594**
• ElasticNet	**MAE− 05180384480777387**



These models served as sanity checks and confirmed that linear relations alone cannot capture the complexity of the Zillow feature space, producing MAE values significantly worse than tree-based methods.

4. Advanced Model Training

Given the high dimensionality, non-linearity, and categorical structure of the data, modern gradient boosting methods were selected:

- **LightGBM**: Fast, GPU-accelerated, leaf-wise tree growth, supports categorical splits.
- **TabNet** (Deep Learning): Sequential attention-based feature learning with sparse masks.

4.1.1 LightGBM - Model 1

LightGBM was used for faster convergance + strong predictive performance.
Key hyperparameters:

Hyperparameter	**Value**
num_leaves	64
learning_rate	0.05
n_estimators	1000
max_depth	8
subsample	0.85
colsample_bytree	0.65
reg_alpha	0.1
reg_lambda	0.1
early_stopping_rounds	50

Training Strategy:
Moderately sized tree depth, Faster learning rate, Early stopping at 50 rounds, Limited number of estimators. Early stopping allowed the model to halt significantly before reaching 1000 rounds, preventing overfitting. The model was not retrained on the full dataset to avoid unnecessary compute, since its purpose was primarily validation and comparison

Local MAE:
0.60660

4.1.2 LightGBM - Model 2

High capacity, high depth, slow trained model to improve predictive accuracy.
Key hyperparameter :

Hyperparameter	**Value**
num_leaves	96
learning_rate	0.009
n_estimators	20000
max_depth	-1
subsample	0.82
colsample_bytree	0.68
reg_alpha	0.3
reg_lambda	0.1
early_stopping_rounds	250

Training Strategy:
Very large boosting round limit (20,000 trees), Small learning rate, Deeper and higher-capacity trees, Early stopping at 250 rounds. Even with 20,000 estimators, LightGBM stopped at **iteration 100**, showing stable convergence. The model was then retrained on the **full dataset** using the optimal iteration count.

Local MAE:
0.055802

4.2 TabNet

TabNet introduces deep learning with sequential attention mechanisms. It captures interactions that tree models miss. Although training is heavier, it improves ensemble robustness.

Early stopping occurred at epoch 110 with best_epoch = 50 and **best_valid_mae = 0.06055**

TabNet Local MAE:
0.060549

5. Ensemble Strategy

To combine the strengths of all models, predictions were blended using weighted averaging:
Final Prediction = 0.40 * LightGBM_1 + 0.50 * TabNet + 0.10 * LightGBM_2
The blend leverages stability from LightGBM and representation power from TabNet.

6. Final Results

The final ensemble achieved a competitive MAE on local validation, and predictions were averaged for all required 2016–2017 forecast months. The final submission contains identical predictions for all six required months as per competition rules.