Presentation: Predicting Housing Prices with XGBoost

Presented by Kewei He

Objective

- Predict housing prices using various features such as the number of bedrooms, bathrooms, square footage, and location coordinates.

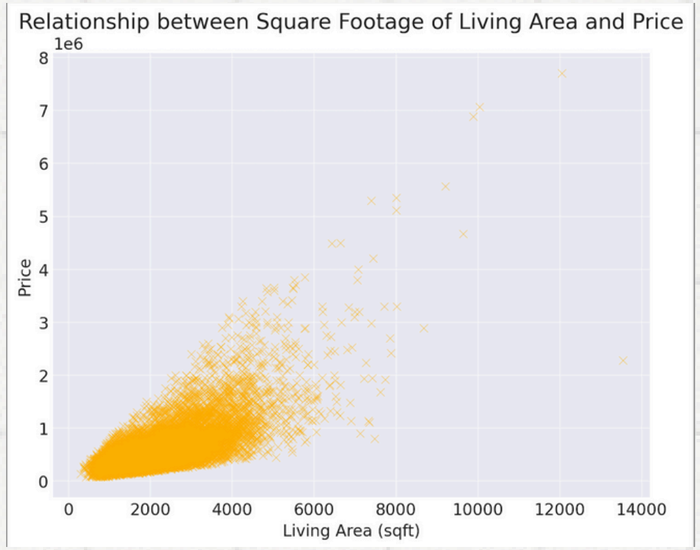
Dataset Overview

01. Data Source

- Sourced from **Kaggle's House Sales Prediction dataset**
- **Features**: Number of bedrooms, bathrooms, living area, lot size, floors, waterfront view, etc
- **Target Variable**: House price

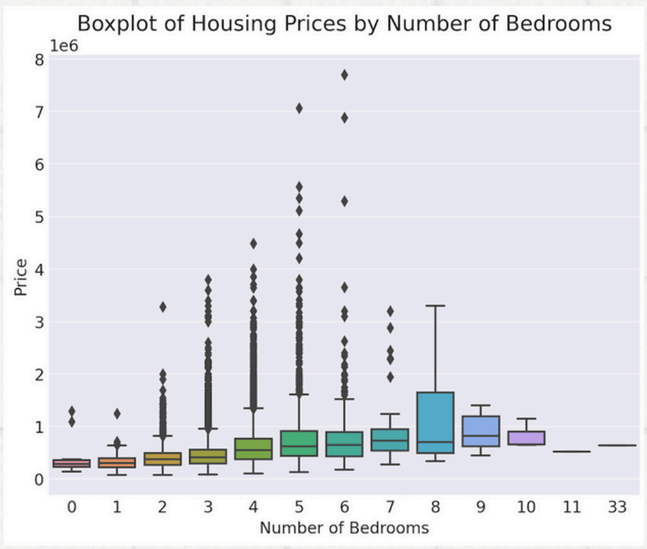
02. Data Preparation

- Handled missing values and outliers
- Applied log transformation to stabilize variance in price data.
- Split the data into **70% training** and **30% test sets**
- Removed outliers by filtering based on residuals greater than 3 standard deviations

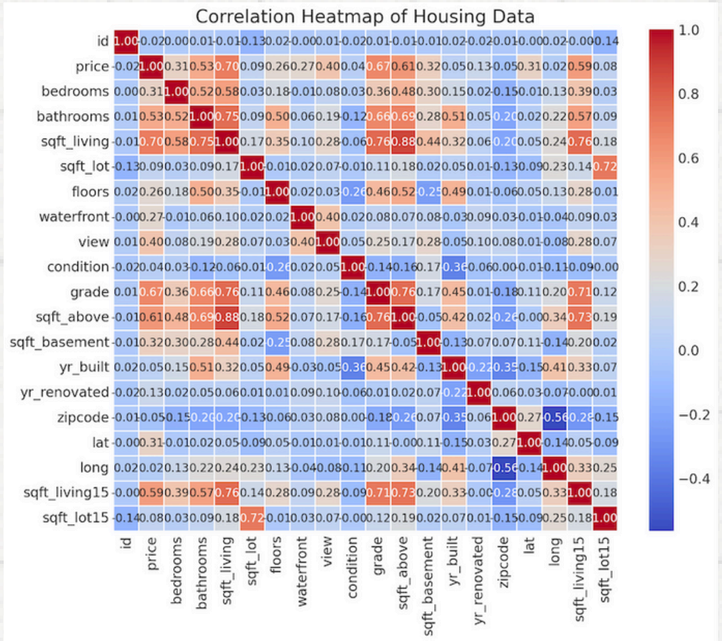


- **Positive correlation** observed between square footage and price

- **Prices are right-skewed**, with most houses priced under $1 million
- Applied **log transformation** to stabilize variance



- **Prices increase with the number of bedrooms**, but variability is higher for houses with more than 8 bedrooms



- **Strong correlations** found between:
  - sqft_living and price: 0.70
  - grade and price: 0.67

Key Results – Linear Regression Model

- Model Evaluation Metrics
  - **R^2: 0.695** (explains ~69.5% of the variance in prices).
  - **RMSE: 209,985** – showing the average prediction error

Add Interaction term and remove outlier

- Model Evaluation Metrics
  - **R^2: 0.669** (explains ~66.9% of the variance in prices).
  - **RMSE: 218,538** – showing the average prediction error

Perform Poor

Switch to XGBoost Models

The model’s performance suggested that certain relationships might be non-linear, warranting exploration with non-linear models.

Why XGBoost

- **Captures non-linear relationships**: Unlike linear models, XGBoost can model complex patterns in data
- **Handles missing values and outliers**: Makes it robust for real-world data.
- **Built-in regularization**: Prevents overfitting through L1/L2 regularization
- **Efficient and fast**: Boosted decision trees make it ideal for large datasets

Model Performance – XGBoost Results

- Initial Model:
  - **R²: 0.892** – 89.2% of the variance in prices explained.
  - **RMSE: 125,148** – Predictive error significantly lower than linear regression.
  -

Hyperparameter Tuning →

- After Hyperparameter Tuning:
  - **R²: 0.8997** – Nearly 90% of the variance in prices explained.
  - **RMSE: 120,305** – Model predictions improved considerably
  -

Model Comparison with Linear Regression

Metric	Linear Regression	XGBoost(Tuned)
R^2	0.695	0.8997
RMSE	209,985	120,305

Top 4 Influential Features

01
Grade

- Homes with higher construction quality have higher prices

02
Living Area (sqft)

- Larger homes command higher prices

03
Waterfront

- Properties with waterfront views are priced at a premium

04
Latitude

- Location directly affects prices, with some areas being more desirable

Next Steps with XGBoost

- **Deploy in Production**: Use the tuned model for real-time predictions.
- **Monitor Performance**: Track new data points for retraining or further fine-tuning.
- **Explore SHAP values**: Use SHAP (SHapley Additive exPlanations) to explain individual predictions and improve interpretability.
  -

Thank you very much!