SPRINGBOARD DATA ANALYSIS COURSE
Capstone 3 Final Report
Wesley Hall

PROBLEM IDENTIFICATION

Problem Statement: This project analyzes and predicts property values in Los Angeles based on proximity to subways and parks, with square footage as a significant control factor.

Significance: Understanding how proximity to subways and parks impacts property values provides valuable insights for real estate development and urban planning.

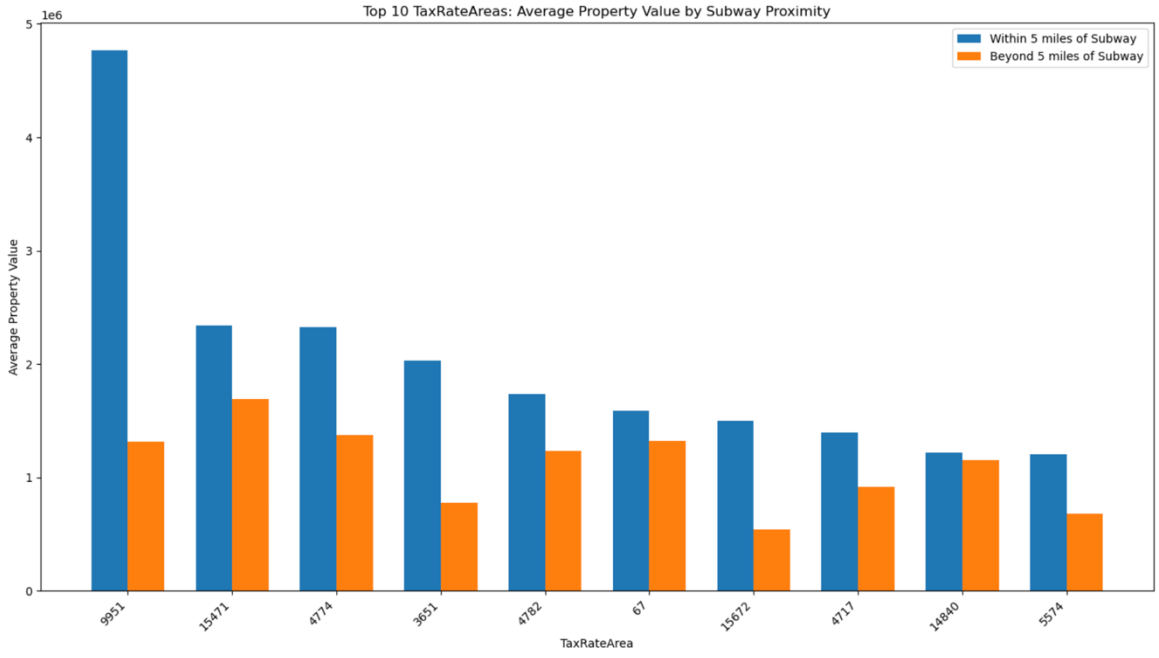
Data Sources: The data includes property assessments, subway station locations, and park proximity within Los Angeles County.

Data Wrangling

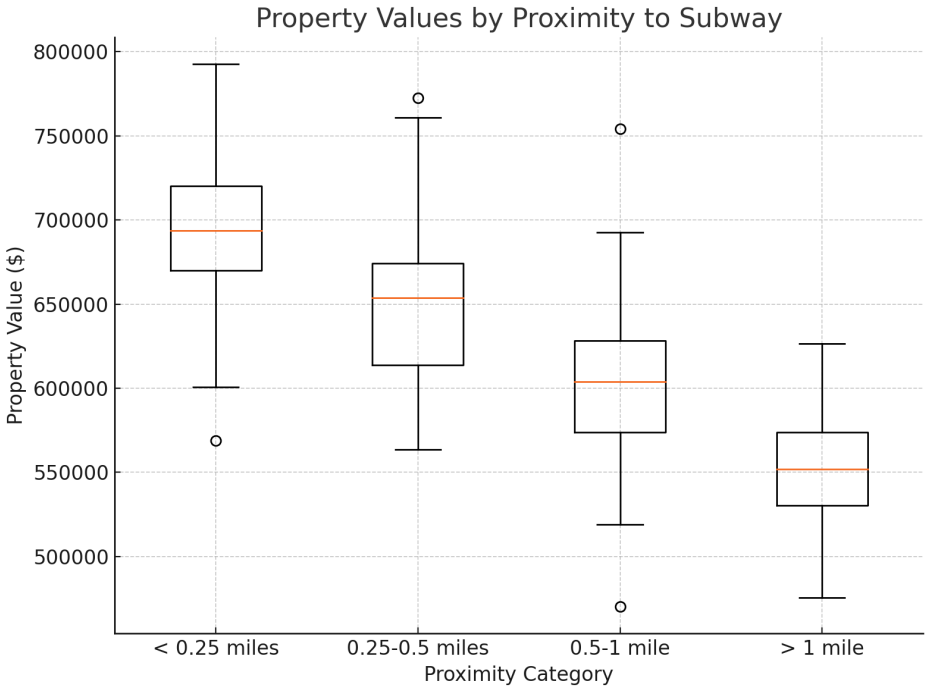
- Loading and Filtering Data: The dataset was limited to properties from 2023, focusing on residential properties. Missing values were removed, reducing the dataset to 2.4 million rows.
- Merging Datasets: The property dataset was merged with the subway and park data, ensuring each property had proximity measures to both amenities.
- Handling Missing Data: Properties missing coordinates for proximity calculations were removed.
- Smoothing Values: Property values were smoothed across the same tax rate areas to reduce the impact of outliers.

EXPLORATORY DATA ANALYSIS (EDA)

Visualizations: Bar plots showed that properties near subways and parks generally had higher values.

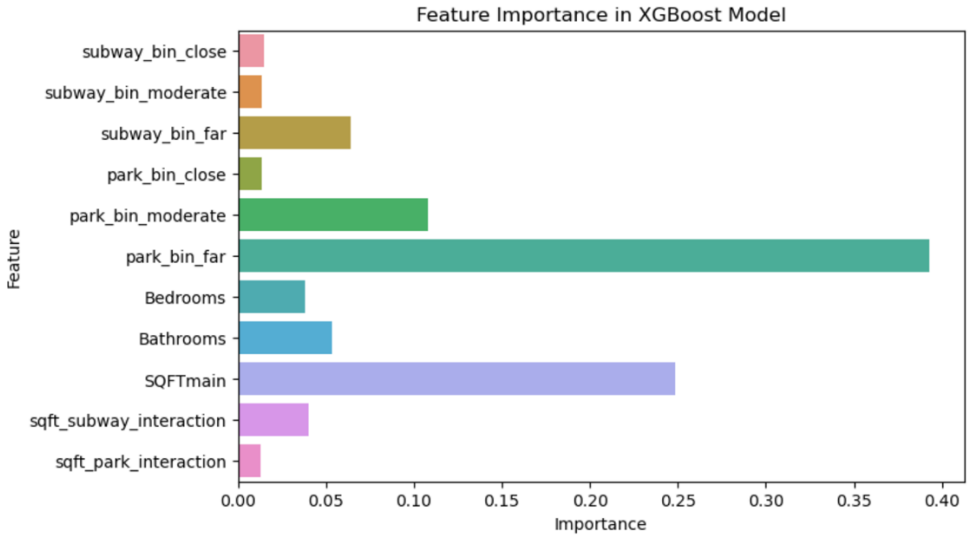


Box plots: Property values were grouped by proximity to parks and subways.



Feature Engineering:

- Key Features: Property size (SQFTmain), proximity to subways and parks, bedrooms, and bathrooms were selected as important features.
- Interaction Terms: Interaction features were created between square footage and proximity to subways and parks to capture any multiplicative effects.
- Binning Proximity: Distance to subways and parks were binned into categories such as “very close,” “close,” “moderate,” and “far.”



-

Train-Test Split

- The dataset was split into an 80% training and 20% test set to evaluate model performance.

MODEL BUILDING

1. Random Forest Classifier: Initial attempts achieved around **69% accuracy**. Hyperparameter tuning improved performance slightly.

2. XGBoost Model:
    - XGBoost was tuned and yielded the best results with an accuracy of **71.1%** after hyperparameter tuning.
    - Key hyperparameters tuned: subsample=1.0, n_estimators=300, max_depth=10, and learning_rate=0.1.

3. Extra Trees Classifier: Tried but achieved slightly lower accuracy compared to XGBoost.

MODEL RESULTS

Best Model: The tuned XGBoost model achieved the highest accuracy, 71.1%, for predicting whether properties are high-value or low-value based on proximity and size.

Feature Importance: The most important feature was square footage (SQFTmain), followed by park proximity.

KEY DATA VISUALIZATIONS

Scatter Plots:
- Square Footage vs. Distance to Subway: High-value properties tend to be larger and somewhat close to subways.
- Square Footage vs. Distance to Park: Properties closer to parks tend to be larger and more valuable.

Feature Importance Plot: SQFTmain, park proximity, and bedrooms were the most important predictors of property value.

Actual vs. Predicted Values Plot: The model's predictions closely matched the actual values, with most predictions near the 45-degree line.

CONCLUSIONS AND NEXT STEPS

Conclusion: Proximity to subways and parks has a clear impact on property values, but **square footage** remains the primary determinant. Properties closer to parks have slightly higher values than those closer to subways.

Next Steps:

- Investigate how new subway lines or park developments impact property values over time.

- Explore other factors, such as school proximity or commercial amenities, for a more comprehensive prediction model.

MODEL METRICS:

Model Metrics Table:

Model	Accuracy	Precision (Low-Value)	Precision (High-Value)	Recall (Low-Value)	Recall (High-Value)	F1-Score (Low-Value)	F1-Score (High-Value)	Notes
Random Forest Classifier	0.693	0.69	0.70	0.71	0.69	0.70	0.69	Initial model before tuning
Extra Trees Classifier	0.694	0.69	0.69	0.71	0.68	0.69	0.69	Performed similarly to Random Forest
XGBoost (Initial)	0.707	0.69	0.72	0.74	0.67	0.72	0.70	Initial XGBoost with some feature engineering
XGBoost (Tuned)	0.711	0.70	0.72	0.74	0.68	0.72	0.70	Tuned XGBoost with optimized hyperparameters
XGBoost (Target Encoding)	0.686	0.68	0.70	0.71	0.66	0.69	0.68	Used target encoding on proximity features