Assignment-4: AI Monsoon-2024

Amartya Singh (2022062)

December 2, 2024

1 Data Preprocessing and Exploratory Data Analysis

1.1 Task 1: Understanding the Dataset

1.1.1 Dataset Overview

The dataset contains the following features and their unique value counts:

Feature Name	**Unique Values**
Address	3233
Possession	1
Furnishing	3
Bathrooms	85
Buildup area	944
Carpet area	2520
Property age	46
Parking	10
Price	755
Brokerage	1517
Floor	125
Per sqft price	2501
BHK	9
Total bedrooms	27

1.1.2 Statistical Analysis of Numerical Columns

The statistics summary is in the provided notebook, and I have attached a picture of it here for reference.

1.2 Task 2: Drop Irrelevant Columns

1.2.1 Columns Dropped

The following columns were removed:

- **Index**: Redundant as it serves no predictive purpose.
- **Property Age**: Correlation with target variable (*Price*) is $< |0.1|$.

Summary Statistics:

	Buildup_area	Carpet_area	Bathrooms	Property_age	Parking
count	6256.000000	6256.000000	6256.000000	6256.000000	6256.000000
mean	1120.690537	864.869801	1.968057	7.519661	1.298593
std	735.147038	583.283918	0.911779	7.374092	0.797501
min	180.000000	150.000000	1.000000	1.000000	0.000000
25%	650.000000	475.000000	1.000000	2.000000	1.000000
50%	950.000000	708.315583	2.000000	5.000000	1.000000
75%	1325.000000	1050.000000	2.000000	10.000000	2.000000
max	15000.000000	14000.000000	10.000000	99.000000	9.000000

	Price	Brokerage	Floor	Per_sqft_price	BHK
count	6.256000e+03	6.256000e+03	6256.000000	6256.000000	6256.000000
mean	3.057852e+07	1.148133e+07	19.885595	23415.351551	2.159527
std	3.790301e+07	3.164281e+07	13.951480	13067.308580	1.002020
min	7.800000e+05	0.000000e+00	2.000000	1440.000000	1.000000
25%	1.050000e+07	1.000000e+05	10.000000	15657.500000	1.000000
50%	1.920000e+07	2.500000e+05	16.000000	21355.000000	2.000000
75%	3.500000e+07	1.100000e+07	23.000000	28792.500000	3.000000
max	5.000000e+08	5.000000e+08	99.000000	100000.000000	10.000000

	Total_bedrooms
count	6256.000000
mean	2.206878
std	0.985628
min	1.000000
25%	2.000000
50%	2.000000
75%	3.000000
max	10.000000

Figure 1: Statistics Summary

1.3 Task 3: Encoding Categorical Features

1.3.1 Label Encoding

Categorical features such as **Address**, **Possession**, and **Furnishing** were encoded using Label Encoding. High-cardinality features such as **Address** were mitigated by aggregating similar groups (e.g., grouping by locality).

1.4 Task 4: Feature Scaling

1.4.1 Standard Scaler Analysis

StandardScaler was applied to numerical columns. Scaling helps in stabilizing model training, but Decision Trees are invariant to scaling. The results of scaled and unscaled training showed minimal differences in the tree-based model performance.

1.5 Task 5: Target Variable Imbalance Detection

1.5.1 Target Variable Distribution

The target variable **Price** is heavily skewed, as shown in Figure 2.

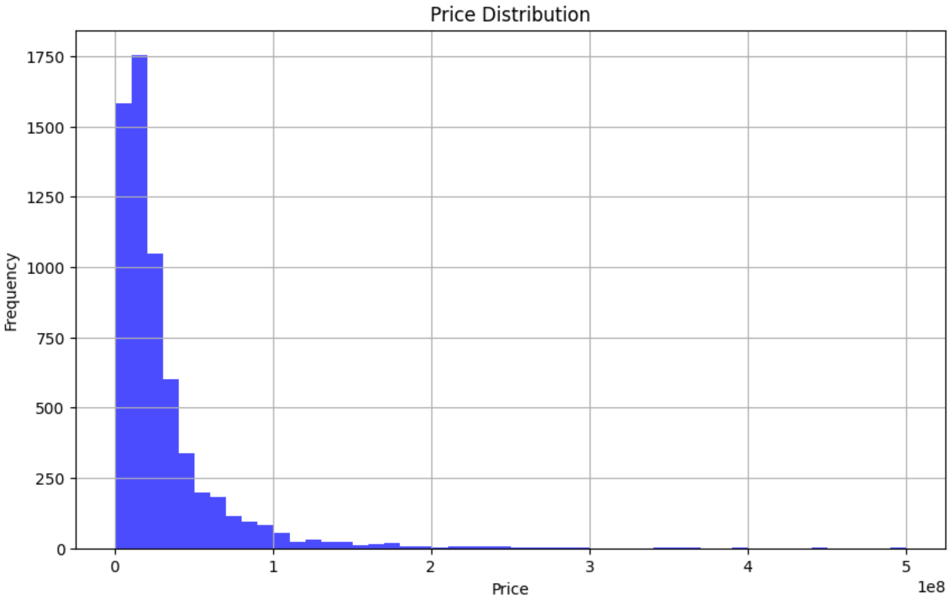

Figure 2: Price Distribution

1.6 Task 6: Handling Imbalanced Data

1.6.1 Random Oversampling and Undersampling

Random oversampling and undersampling were applied. While oversampling mitigates class imbalance effectively, it may lead to overfitting. Undersampling reduces dataset size, potentially discarding useful information.

2 Building Decision Tree Model)

2.1 Task 1: Model Training

A Decision Tree Regressor was trained, achieving a maximum depth of X. Figure 3 visualizes the tree structure.

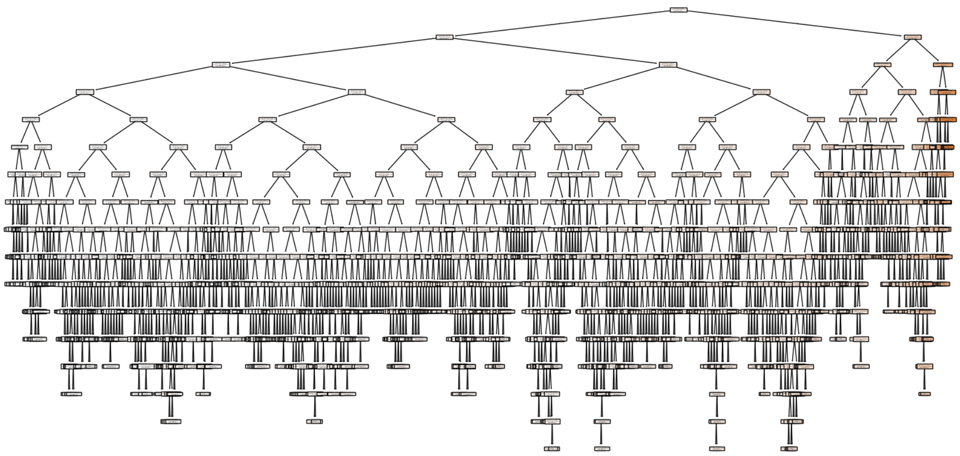

Figure 3: Visualized Decision Tree

2.2 Task 2: Feature Importance and Hyperparameter Tuning)

The important features have been mentioned below.

2.2.1 Hyperparameter Tuning

Grid Search was performed over:

- max_depth: [3, 5, 10, None]
- min_samples_split: [2, 10, 20]

Optimal Parameters: **max_depth=None**, **min_samples_split=2**, **min_samples_leaf=5**.

2.3 Task 3: Pruning Decision Tree

The training data is often overfitted by unpruned trees, which leads to a low training error but a greater validation error. Although they have a somewhat greater training error, pruned trees improve generalization and validation accuracy by reducing complexity. The trade-off between model complexity and performance is successfully balanced by pruning. Figure 5 shows the unpruned tree and then the pruned tree.

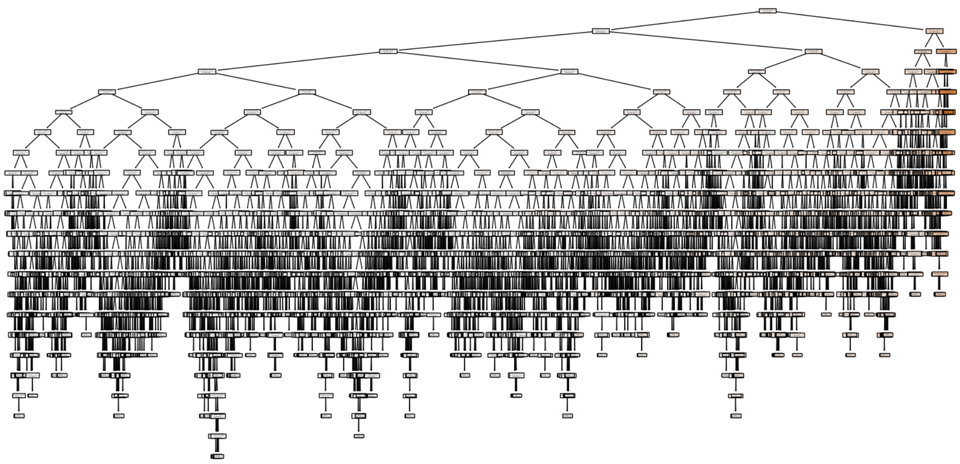

Figure 4: Unpruned Decision Tree

Pruning significantly improved model generalization:

$$\text{Node Reduction} = \frac{235 \text{ nodes} - 89 \text{ nodes}}{235 \text{ nodes}} \times 100\% = 62.13\% \tag{1}$$

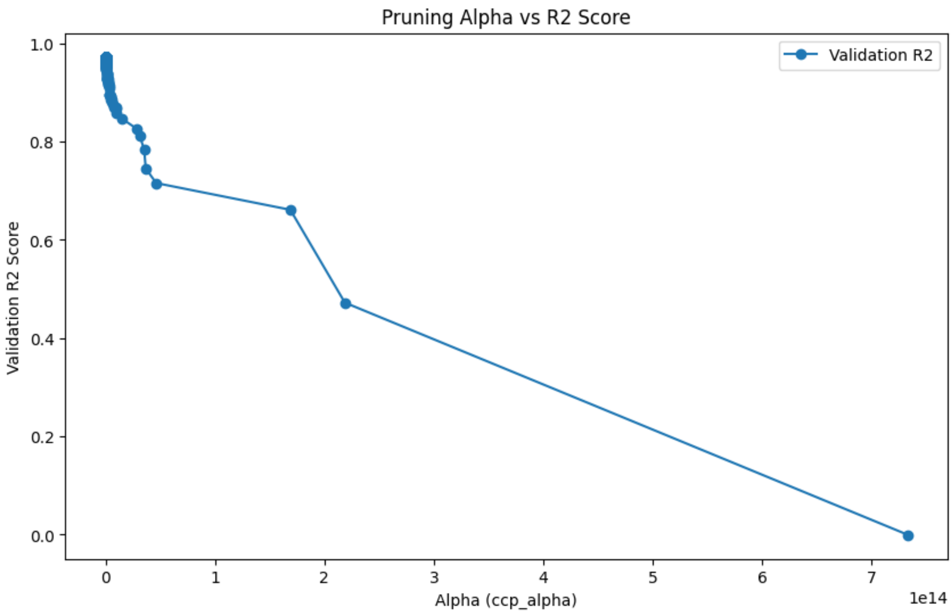

Figure 6: Alpha vs R2 score

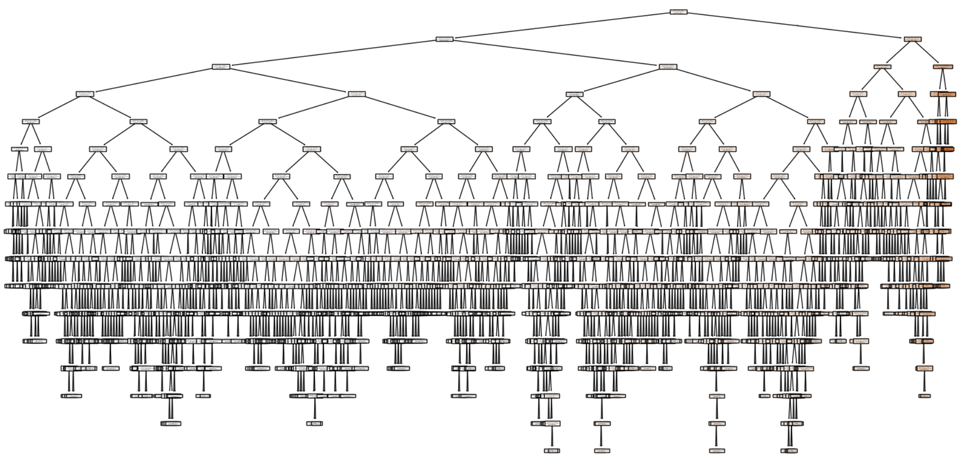

Figure 5: Pruned Decision Tree

2.3.1 Impact of Alpha on Model Performance

Following figure illustrates the effect of the regularization parameter α on the model's training and validation performance. As α increases, both training and validation scores decline. Smaller α values result in higher scores but can lead to overfitting, whereas larger α values reduce model complexity, leading to underfitting.

2.4 Task 4: Over-fitting

Cross-validation helps us solve the problem of over-fitting.
Cross-Validation Scores: [0.9736957 0.96025188 0.97554157 0.92392334 0.92079348]
Mean R2 Score: 0.9508411933020531

3 Model Evaluation and Error Analysis)

3.1 Task 1: Model Evaluation

The tuned model achieved the following:

- **Training R2**: 0.9783
- **Test R2**: 0.9650
- **Training MSE**: 31153040523866.4023
- **Test MSE**: 40902369190984.5312
- **Training MAE**: 1363621.9762
- **Test MAE**: 1645979.3192

3.2 Task 2: Residual and Error Analysis

Residuals analysis highlighted underperformance for high-price properties, as shown in Figure 7.

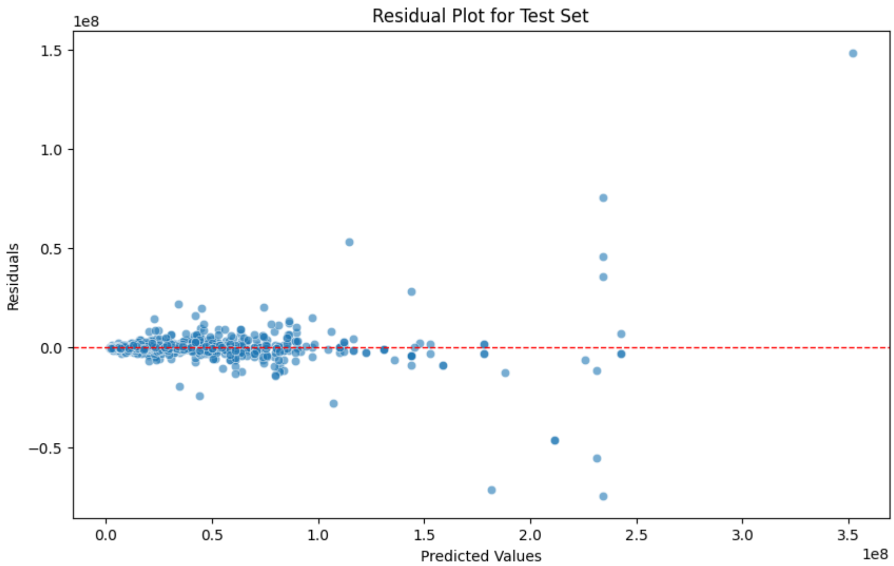

Figure 7: Residual Analysis

3.3 Task 3: Feature Importance Based Analysis

The top 3 features were analyzed individually for their impact on **Price**. RMSE for each feature:

- Feature 1: Carpet Area
- Feature 2: Brokerage
- Feature 3: Square feet price

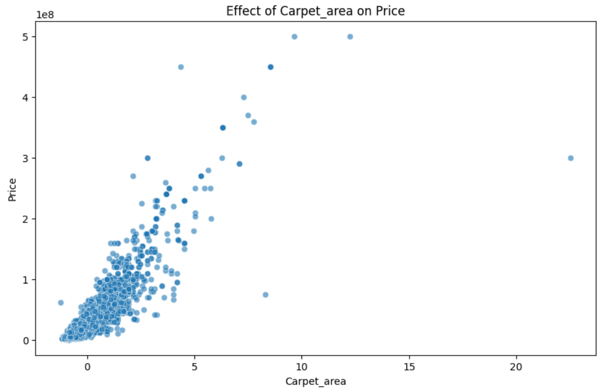

Figure 8: Price vs Carpet area

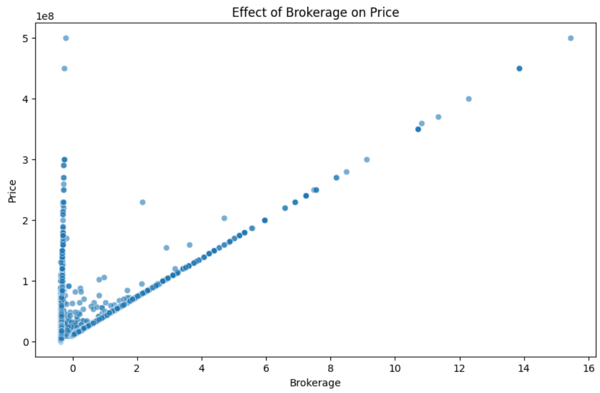

Figure 9: Brokerage vs Price

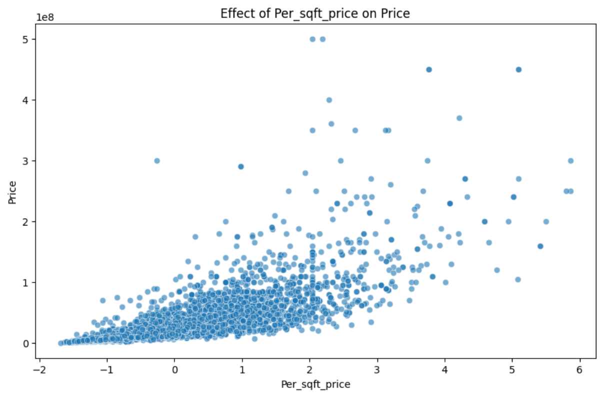

Figure 10: Price vs Sqrt. Area price

4 Bonus Challenge

4.1 Task 1: Advanced Imbalance Handling

ADASYN performed better than SMOTE for handling imbalance. ADASYN focuses on generating synthetic samples for minority classes.

4.2 Task 2: Ensemble Learning: Random Forest

Random Forest - Training Set:

MSE: 5129886353872.8750, R2: 0.9964
Random Forest - Test Set:
MSE: 9797526109555.6172, R2: 0.9916

Model Comparison:
Decision Tree Test R2: 0.9650
Random Forest Test R2: 0.9916
Tradeoffs: Decision Tree is simpler and interpretable but prone to overfitting. Random Forest reduces overfitting but requires more computation.