Q1) calculate root node Entropy

entropy of root node = $Entropy_{root} = -\left(\frac{5}{10}\log_2\frac{5}{10} + \frac{5}{10}\log_2\frac{5}{10}\right)$

$Entropy_{root} = -(0.5 \cdot (\log_2 0.5)) \times 2 = 1$

* calculate IG for each feature

for long term debt

split into YES : 5 samples (A1, 4R)

5 samples (A4, R1)

Entropy for subset

$Entropy_{Yes} = -\left(\frac{1}{5}\log_2\frac{1}{5} + \frac{4}{5}\log_2\frac{4}{5}\right)$

weighted entropy or Entropy (S | long term debt)

$= P(LTD = Yes)\left(-\sum P_i \log_2 P_i\right) + P(LTD = No)\left(-\sum P_i \log_2 P_i\right)$

$= \frac{5}{10} \times \left(-\frac{1}{5}\log_2\frac{1}{5} - \frac{4}{5}\log_2\frac{4}{5}\right) + \frac{5}{10}\left(-\frac{4}{5}\log_2\frac{4}{5} - \frac{1}{5}\log_2\frac{1}{5}\right)$

$= 0.721$

for unemployment

Entropy(S|unemployment)

$= \frac{2}{10}\left(-\frac{2}{2}\log_2\frac{2}{2} - 0\right) + \frac{8}{10}\left(-\frac{5}{8}\log_2\frac{5}{8} - \frac{3}{8}\log_2\frac{3}{8}\right)$

$= 0.763$

for credit Rating

Entropy(S|Credit Rating)

$= \frac{3}{10}\left(-\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}\right) + \frac{7}{10}\left(-\frac{3}{7}\log_2\frac{3}{7} - \frac{4}{7}\log_2\frac{4}{7}\right)$

$\simeq 0.965$

for Downpayment < 20%

Entropy(S | DP<20%)

$= \frac{5}{10} \times \left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right)$

$+ \frac{5}{10}\left(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}\right)$

$\simeq 0.971$

Information gain

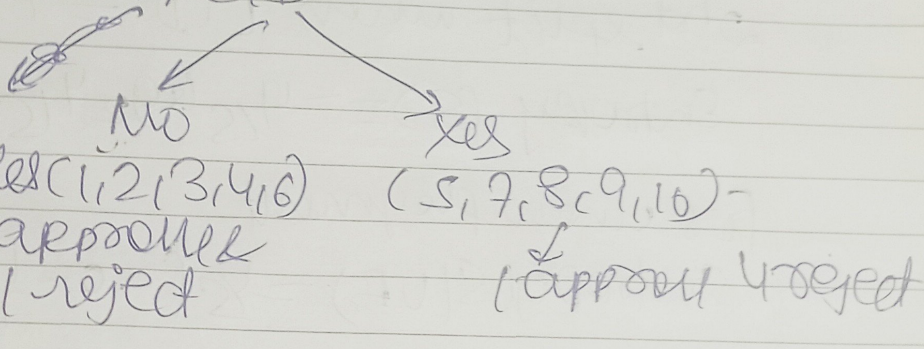
Gain(S, LTD) = 1 − 0.721 = 0.279 → highest IG

Gain (S, unemp) = 1 − 0763 = 0.237

G(S, CR) = 1 − 0.965 = 0.035

Gain (S, DP<20%) = 1 − 0.971 = 0.029

Splitting based on LTD



2nd split when LTD = 'NO'

$$Entropy(S) = -\frac{4}{5}\log_2\frac{4}{5} - \frac{1}{5}\log_2\frac{1}{5} = 0.721$$

for unemployment

$$Entropy(S|UE) = 4/5\left(-4/5\log_2 1/2 - \frac{1}{2}\log_2 1/2\right) + \frac{3}{5}\left(-\frac{3}{5}\log_2 3/3 - 0\right) = 2/5 = 0.4$$

for DP<20%

$$Entropy(S|DP<20\%) = 3/5\left[-\frac{2}{3}\log\frac{2}{3} - \frac{1}{3}\times \log_2 1/3\right) + \frac{2}{5}\left(-\frac{2}{2}\log\frac{2}{2} - 0\right) = 0.55$$

IG(S, LTD = NO) →

IG(S, unemp) = 0.721 - 0 = 0.721 → highest

IG(S, CR) = 0.721 - 0.4 = 0.321

IG(S, DR 20%) = 0.721 - 0.55 = 0.171

Splitting on left side (LTD = No)

based on unemployed

2nd split when LTD = Yes

Entropy (S) = $-4/5 \log_2 4/5 - 1/5 \log_2 1/5 = 0.721$

for unemployment

Entropy(S|UE) = $1/5(-0 - 1/1 \log_2 1/1) - 4/5(-1/4 \log_2 1/4$

$-3/4 \log_2 3/4) \approx 0.649$

<u>for credit rangt</u>

Entropy (S|CR) = $1/5(-1 \log_2 1/1 - 0) + 4/5(0 - 4/4 \log_2 4/4)$

$= 0$

for DP<20%.

Entropy (S|DP<20%) = $2/5(0 - 2/2 \log_2 2/2)$

$+ 3/5(-1/3 \log 1/3 - 2/3 \log_2 2/3)$

≈ 0.55

IG (S(LTD = 'yes') →

IG(S, unemploy) = 0.721 − 0.649 = 0.072

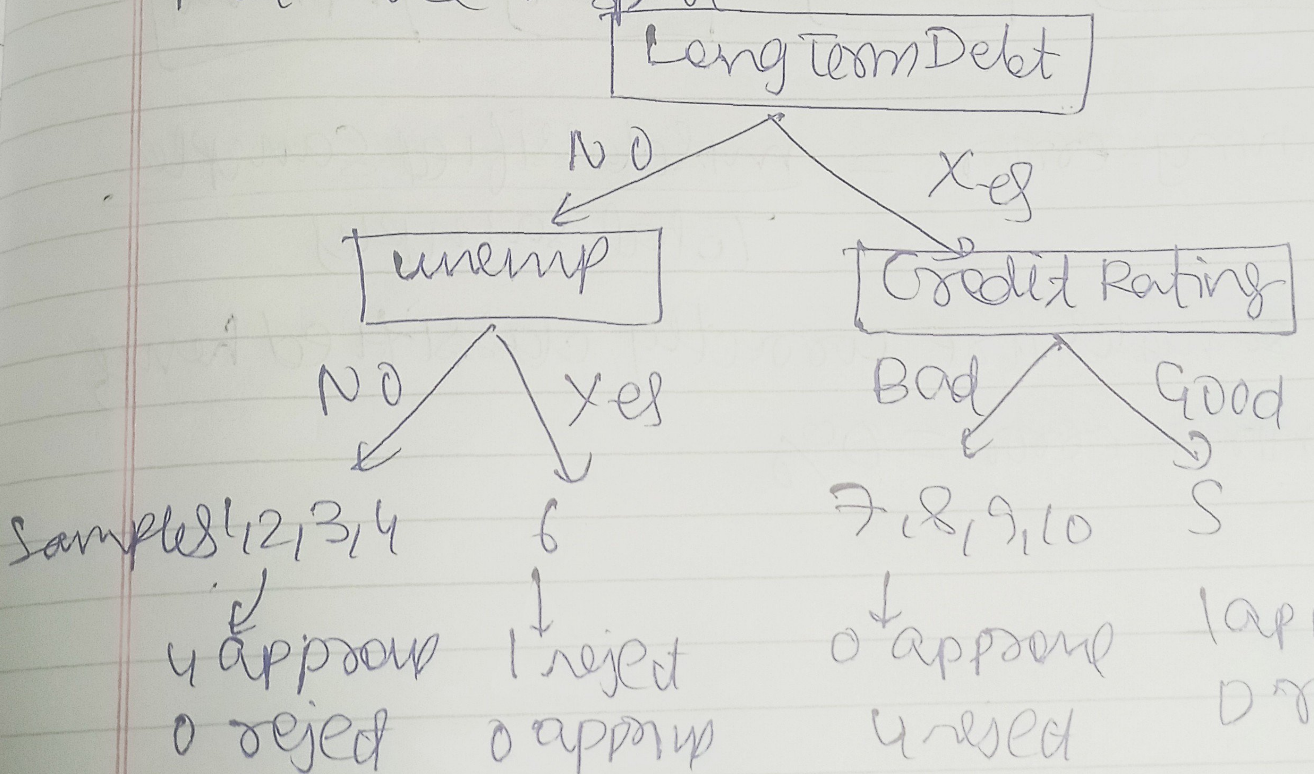
IG(S, CR) = 0.721 − 0 = 0.721

IG(S, DP<20%) = 0.721 − 0.55 = 0.171

→ highest IG

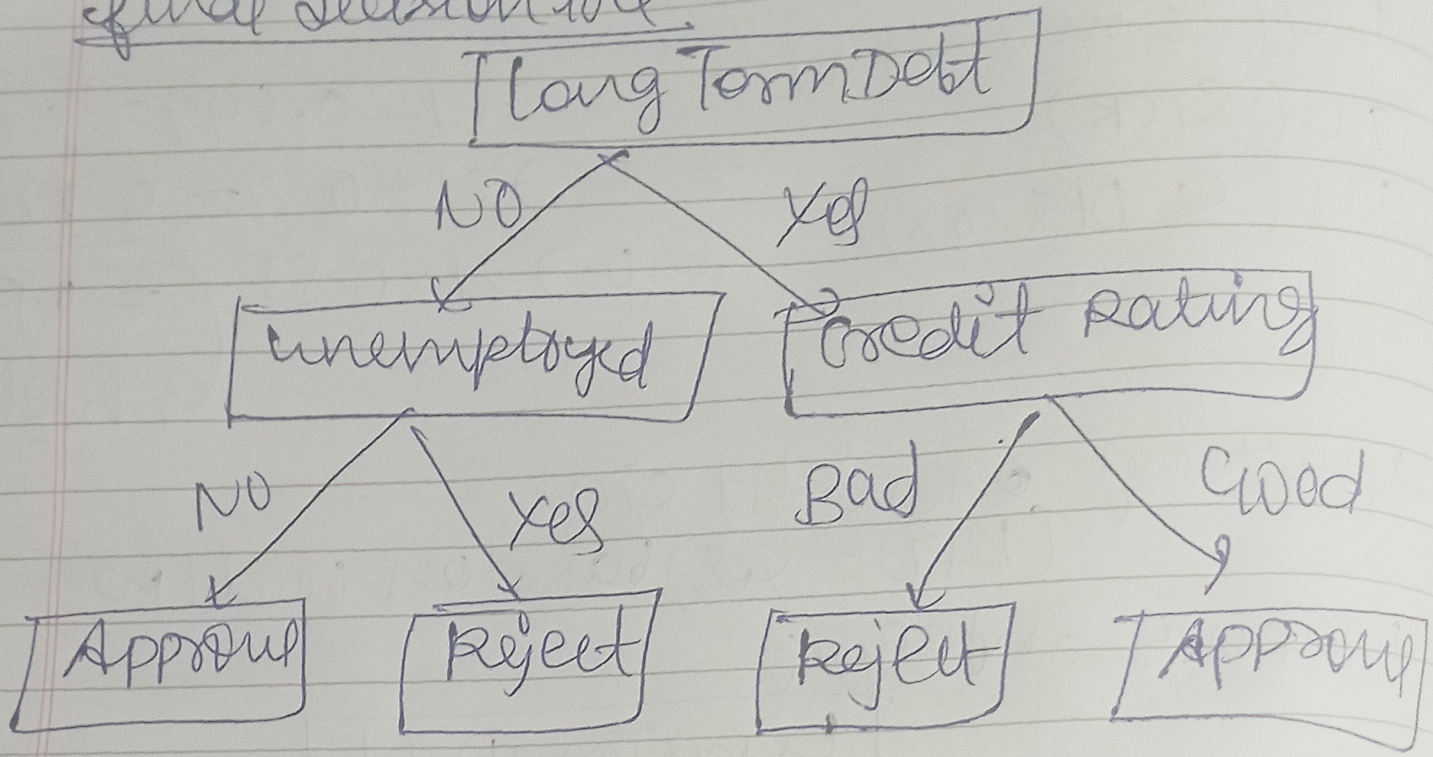
splitting on right side (LTD = yes) based on credit rating

making 2nd split



all in pure leaves, no impurities, all are same, homogeneous

final decision tree



$$\text{Training error} = \frac{\text{misclassified samples}}{\text{Total samples}}$$

All samples are correctly classified here,

training error = 0%

CODING PROBLEMS REPORT

Introduction

The objective of this project was to build a Decision Tree Regressor for predicting real estate price brackets. The task involved preprocessing the dataset, handling imbalance, training and optimizing the model, and evaluating its performance. Additionally, advanced techniques like SMOTE, ADASYN, and Random Forest were explored.

Task 1: Understanding the Dataset (2 Marks)

1.1 Dataset Overview

- **The dataset contained numerical features such as `Carpet_area`, `Buildup_area`, and `Per_sqft_price`, along with categorical attributes like `Furnishing` and `Possession`.**
- **Target Variable: `Price`, binned into four brackets: `Low`, `Medium`, `High`, and `Very High`.**

1.2 Statistical Summary of Numerical Columns

- **Key features (`Carpet_area`, `Per_sqft_price`) showed high variability, indicating their influence on `Price`.**

Figures to Include:

1. **Summary table of mean, standard deviation, min, and max for numerical features.**

```
Statistical Analysis of Numerical Columns:
              index  Buildup_area  Carpet_area    Bathrooms  Property_age  \
count  1564.000000   1564.000000  1564.000000  1564.000000   1564.000000
mean   4850.597826   1097.715378   850.997202     1.996071      7.279412
std    2752.198896    667.933500   531.584051     0.853918      6.553627
min       4.000000    225.000000   180.000000     1.000000      1.000000
25%    2530.000000    650.000000   461.663715     1.000000      2.000000
50%    4862.000000    910.000000   707.000000     2.000000      5.000000
75%    7222.000000   1320.000000  1050.000000     2.000000     10.000000
max    9543.000000  10000.000000  8000.000000     7.000000     50.000000

            Parking         Price     Brokerage        Floor  Per_sqft_price  \
count  1564.000000  1.564000e+03  1.564000e+03  1564.000000     1564.000000
mean      1.323529  2.961389e+07  1.067016e+07    20.109013    23347.163657
std       0.795177  3.419089e+07  2.843969e+07    14.001883    12734.140364
min       0.000000  1.800000e+06  0.000000e+00     2.000000     2550.000000
25%       1.000000  1.010000e+07  9.600000e+04    10.000000    15252.500000
50%       1.000000  1.945000e+07  2.400000e+05    16.000000    21685.000000
75%       2.000000  3.557500e+07  3.564006e+06    23.000000    29035.000000
max       8.000000  5.000000e+08  3.100000e+08    92.000000    88890.000000

               BHK  Total_bedrooms
count  1564.000000     1564.000000
mean      2.136509        2.177727
std       0.991682        0.956470
min       1.000000        1.000000
25%       1.000000        1.000000
50%       2.000000        2.000000
75%       3.000000        3.000000
max       6.000000        6.000000
```

Task 2: Drop Irrelevant Columns (1 Mark)

Columns Removed

- **`Possession`: Low correlation with `Price` (< |0.1|).**
- **`Total_bedrooms`: Minimal predictive value.**

Reasoning:

- **These features did not significantly contribute to the model's ability to predict `Price`.**

Task 3: Encoding Categorical Features (2 Marks)

Steps Taken:

- **Applied Label Encoding to transform categorical variables into numeric values.**

- **Addressed high cardinality issues in the `Address` column by assigning unseen labels a default value.**

Task 4: Feature Scaling (3 Marks)

Steps Taken:

- **Scaled numerical features using StandardScaler.**
- **Analysis showed scaling improved consistency but had minimal effect on Decision Tree performance due to its scale-invariant nature.**

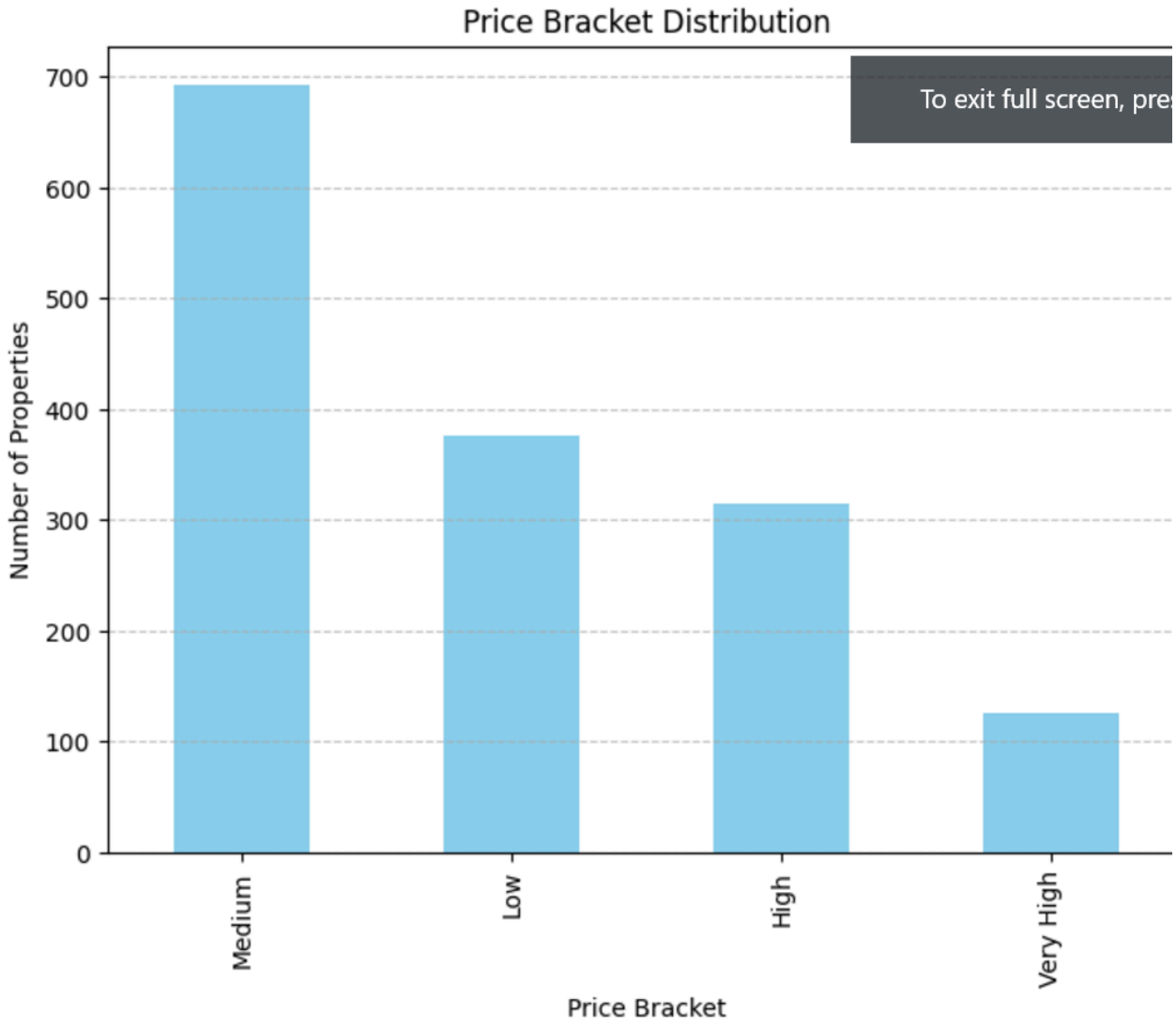
```
Scaled Training Data (First Few Rows):
    Address  Possesion  Furnishing  Buildup_area  Carpet_area  Bathrooms  \
0 -0.761694        0.0    1.081839     -0.165811    -0.378231   0.004603
1  1.654996        0.0   -0.358158      0.639916     0.706705  -1.166844
2 -0.714597        0.0    1.081839     -0.236200    -0.257797   0.004603
3 -1.129643        0.0   -1.798154     -0.520750     0.025043   0.004603
4  0.345097        0.0   -0.358158      0.527593     0.481940  -1.166844

    Parking     Price  Brokerage     Floor  Per_sqft_price       BHK  \
0 -0.406995  15800000  -0.369678 -0.150672       -0.576365 -0.137698
1 -0.406995  50000000   1.383363 -0.936532        0.741774  0.871013
2  0.850989  19500000  -0.368623 -0.364997       -0.204804 -0.137698
3 -0.406995  29000000   0.644722  0.349421        1.203673 -0.137698
4 -0.406995  42000000   1.101976 -0.936532        0.440912  0.871013

   Total_bedrooms
0       -0.185875
1        0.859971
2       -0.185875
3       -0.185875
4        0.859971

Summary Statistics of Scaled Numerical Columns:
            Address           BHK     Bathrooms     Brokerage  Buildup_area  \
count  1.564000e+03  1.564000e+03  1.564000e+03  1.564000e+03  1.564000e+03
mean   1.521943e-16  9.313380e-17 -2.294272e-16  8.518335e-17 -4.088801e-17
std    1.000320e+00  1.000320e+00  1.000320e+00  1.000320e+00  1.000320e+00
min   -1.674208e+00 -1.146409e+00 -1.166844e+00 -3.753055e-01 -1.307008e+00
25%   -8.742867e-01 -1.146409e+00 -1.166844e+00 -3.719289e-01 -6.705137e-01
50%   -4.198573e-02 -1.376980e-01  4.603109e-03 -3.668639e-01 -2.811289e-01
75%    8.462435e-01  8.710125e-01  4.603109e-03 -2.499474e-01  3.329010e-01
max    1.778627e+00  3.897144e+00  5.861838e+00  1.052844e+01  1.333236e+01

        Carpet_area         Floor    Furnishing       Parking  Per_sqft_price  \
count  1.564000e+03  1.564000e+03  1.564000e+03  1.564000e+03    1.564000e+03
mean  -4.997423e-17 -1.680951e-16 -9.994847e-17  6.133201e-17   -2.226125e-16
std    1.000320e+00  1.000320e+00  1.000320e+00  1.000320e+00    1.000320e+00
min   -1.262664e+00 -1.293741e+00 -1.798154e+00 -1.664979e+00   -1.633704e+00
25%   -7.326368e-01 -7.222062e-01 -3.581577e-01 -4.069950e-01   -6.358696e-01
50%   -2.709699e-01 -2.935553e-01 -3.581577e-01 -4.069950e-01   -1.305699e-01
75%    3.744778e-01  2.065373e-01  1.081839e+00  8.509895e-01    4.468033e-01
```

Task 5: Target Variable Imbalance Detection (4 Marks)

Analysis:

- **Price Distribution: `Medium` dominated the dataset, while `Very High` was underrepresented.**

Visualization:



1. **Bar chart showing the original class distribution.**

Task 6: Handling Imbalanced Data (3 Marks)

Techniques Used:

1. **Random Undersampling: Reduced the majority class to balance the dataset.**
2. **SMOTE: Generated synthetic samples for minority classes.**

Results:

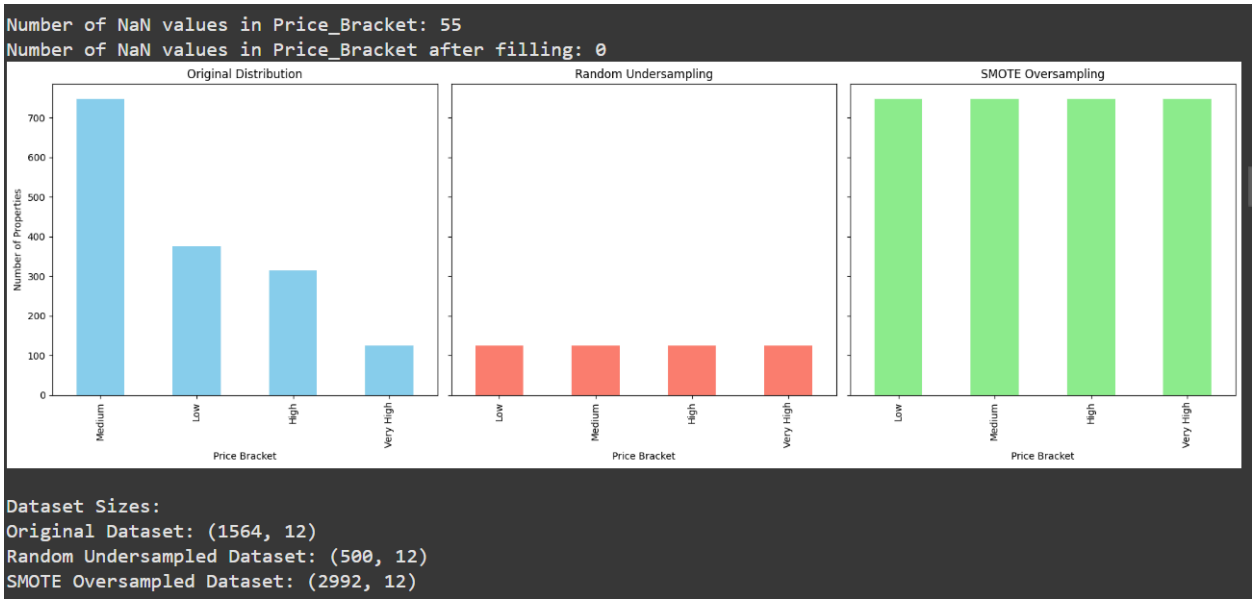
```
Dataset Sizes:
Original Dataset: (1564, 12)
```

```
Random Undersampled Dataset: (500, 12)
SMOTE Oversampled Dataset: (2992, 12)
```

- **Balanced datasets created using both techniques, with SMOTE generating 2992 samples.**

Visualization:

1. **Bar charts showing class distributions after SMOTE and undersampling.**



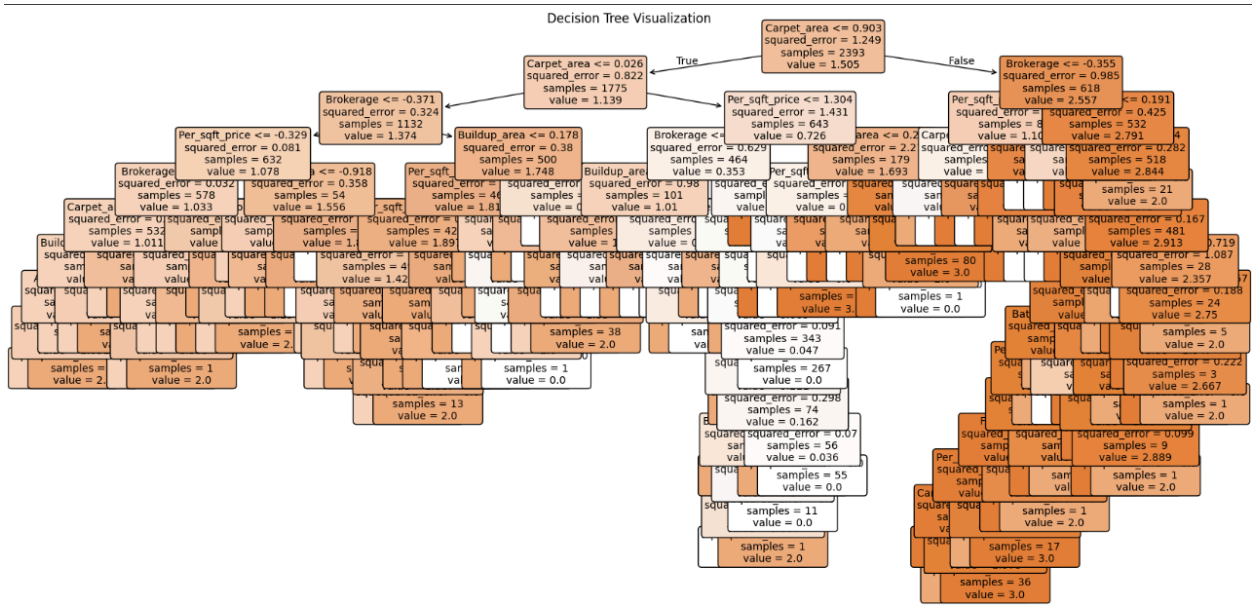
Task 7: Decision Tree Model Training (3 Marks)

Model Training:

- **Trained a Decision Tree Regressor using the SMOTE-balanced dataset.**

Visualization:

1. **Full Decision Tree structure showing depth, splits, and leaf values.**




```
Validation Set Results (Encoded Numeric Values):
Mean Absolute Error (MAE): 0.0651085141903172
Mean Squared Error (MSE): 0.11519198664440734
R-squared (R²): 0.9081823069164842

Decoded Predictions vs Actuals:
   Predicted     Actual
0       High     Medium
1  Very High  Very High
2     Medium     Medium
3       High       High
4       High       High
```

Task 8: Feature Importance and Hyperparameter Tuning (4 Marks)

Feature Importance:

- **Top features: `Carpet_area`, `Per_sqft_price`, and `Buildup_area`.**

Hyperparameter Tuning:

- **GridSearchCV found optimal parameters:**

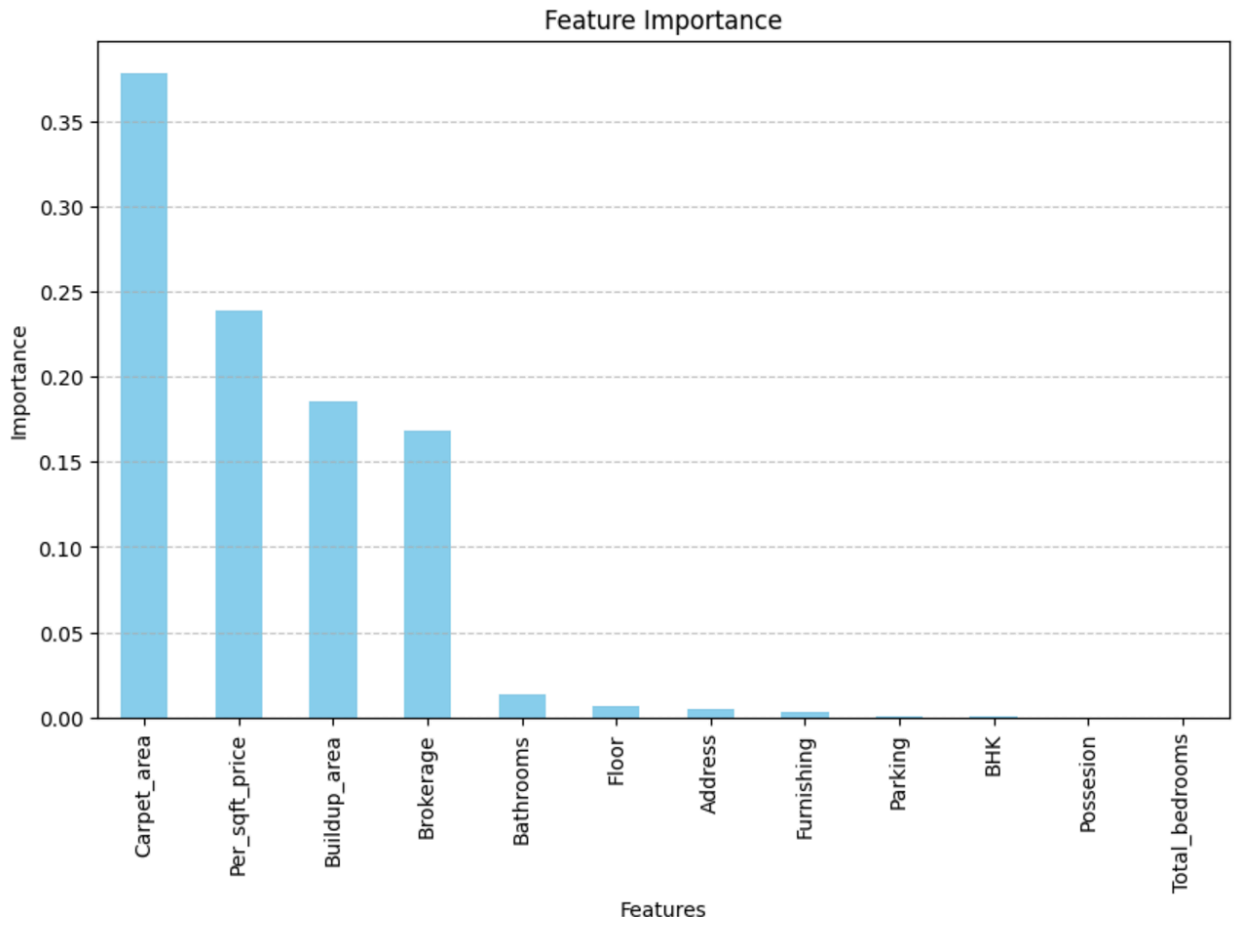
- `max_depth=10`
- `min_samples_split=5`
- `min_samples_leaf=2`

Comparison:

- **Tuned model improved generalization compared to the default tree.**

Visualization:

1. **Bar chart of feature importances.**



```
Feature Importances:
Carpet_area        0.378291
Per_sqft_price     0.239163
Buildup_area       0.185290
Brokerage          0.168301
Bathrooms          0.013475
Floor              0.006275
Address            0.005119
Furnishing         0.003000
Parking            0.000861
BHK                0.000223
Possesion          0.000000
Total_bedrooms     0.000000
dtype: float64
```

```
Fitting 5 folds for each of 108 candidates, totalling 540 fits

Best Hyperparameters:
{'max_depth': 10, 'max_features': None, 'min_samples_leaf': 2, 'min_samples_split': 5}

Tuned Model Validation Set Results:
Mean Absolute Error (MAE): 0.08333709603425153
Mean Squared Error (MSE): 0.12050095811508825
R-squared (R²): 0.9039506105347829

Default Model Validation Set Results (for comparison):
Mean Absolute Error (MAE): 0.0651085141903172
Mean Squared Error (MSE): 0.11519198664440734
R-squared (R²): 0.9081823069164842
/usr/local/lib/python3.10/dist-packages/numpy/ma/core.py:2820: RuntimeWarning: invalid value encountered in
  _data = np.array(data, dtype=dtype, copy=copy,
```

Task 9: Pruning Decision Tree (4 Marks)

Cost-Complexity Pruning:

- **Pruned the Decision Tree using `ccp_alpha` to improve generalization.**

Results:

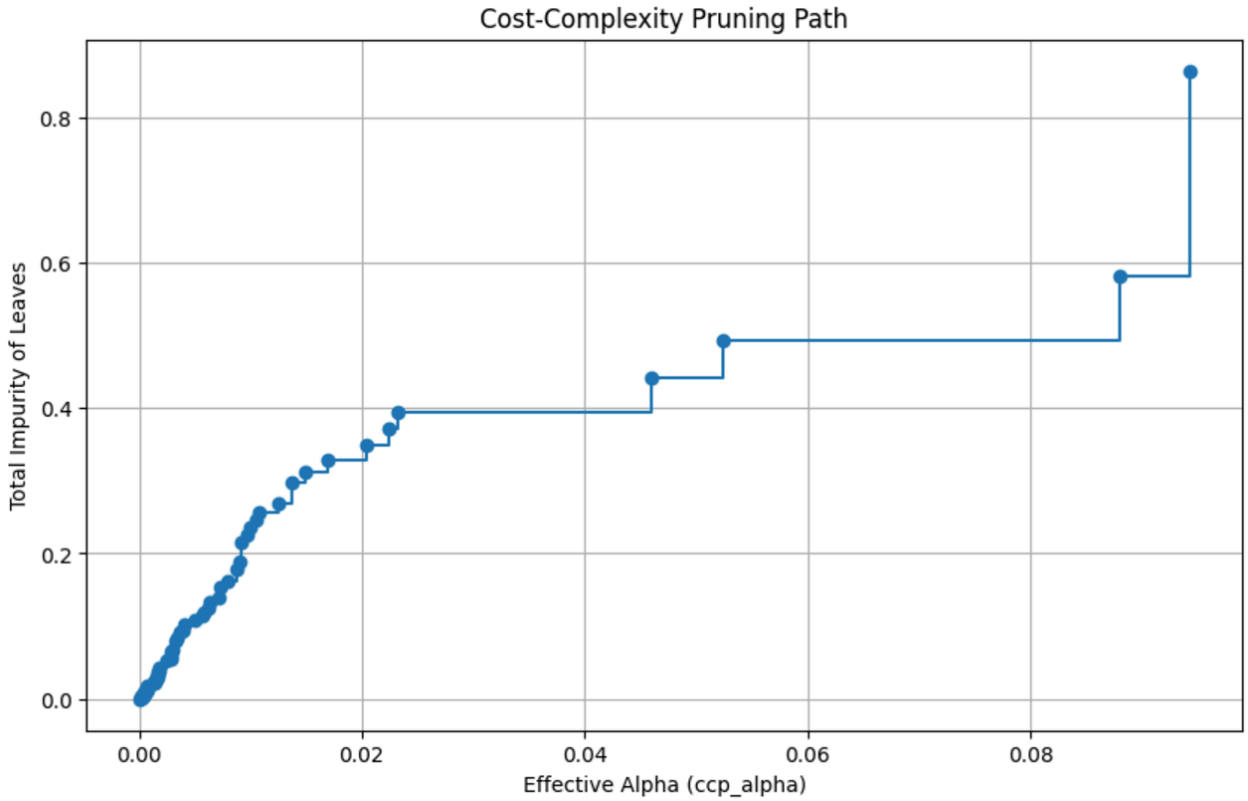
- **Pruned tree achieved better generalization:**

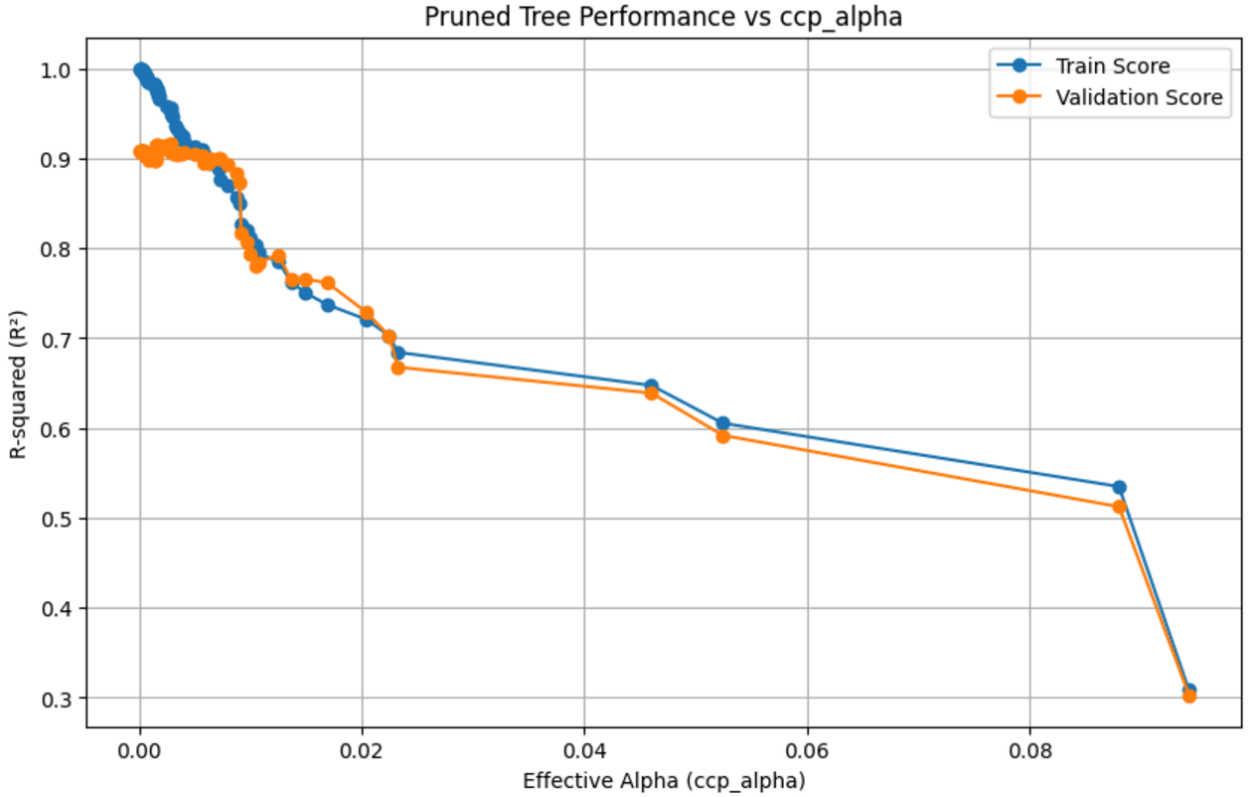
```
Pruned Tree Validation Set Results:
Mean Absolute Error (MAE): 0.09983620880496988
Mean Squared Error (MSE): 0.10543984939229471
R-squared (R²): 0.915955579790811

Default Tree Validation Set Results (for comparison):
Mean Absolute Error (MAE): 0.0651085141903172
Mean Squared Error (MSE): 0.11519198664440734
R-squared (R²): 0.9081823069164842
```

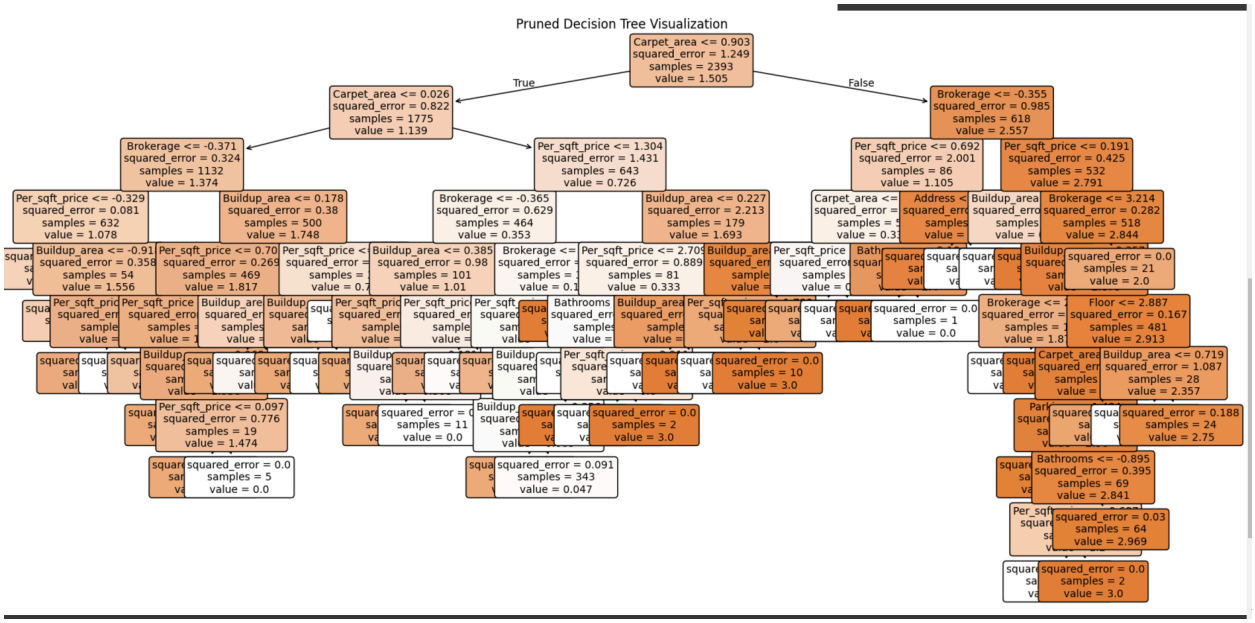
- **Validation R^2: 0.9160.**

Visualization:

1. **Pruned tree structure.**
2. **Pruning path (`ccp_alpha` vs. impurity).**







Task 10: Handling Overfitting (4 Marks)

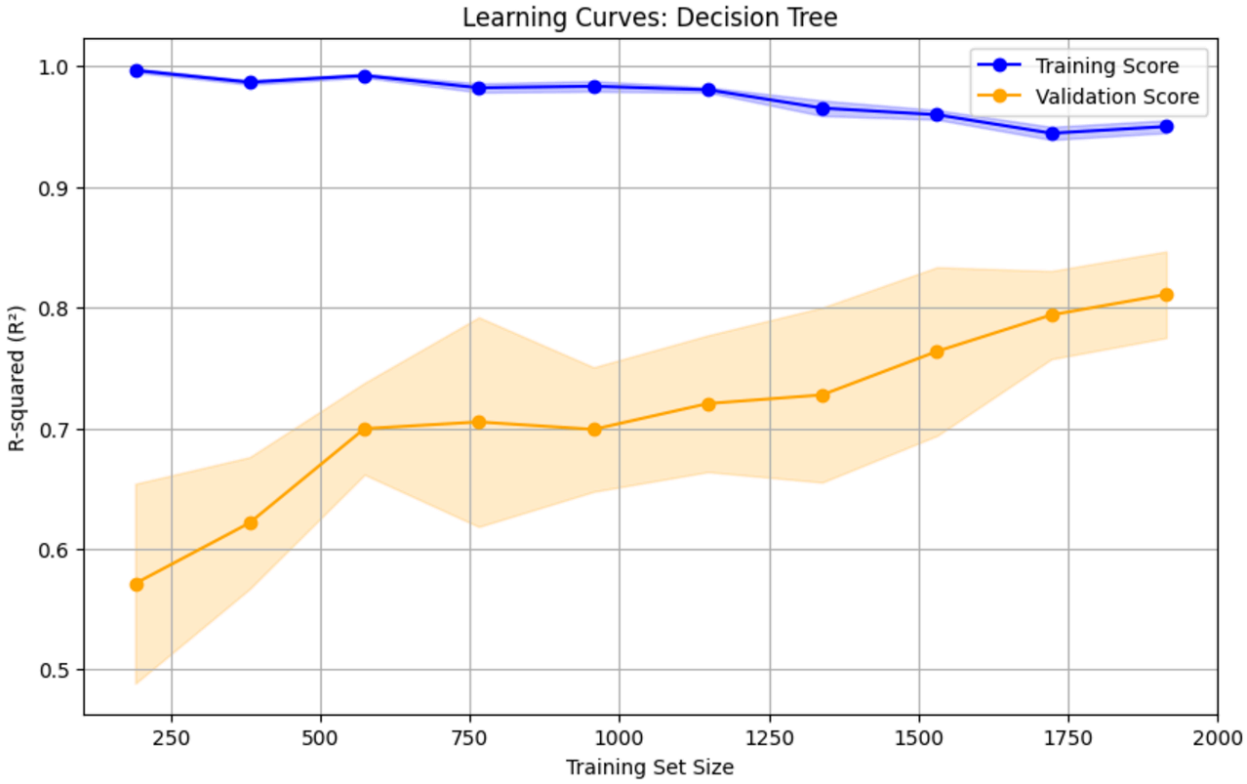
Cross-Validation:

```
Cross-Validation Results:
Cross-Validation R² Scores: [0.81435155 0.77922732 0.76478104 0.83170313 0.86399982]
Mean Cross-Validation R²: 0.8108125725002907
Standard Deviation of Cross-Validation R²: 0.03576129966305753
```

- **5-fold cross-validation:**
    - **Mean R^2: 0.918 (indicating good generalization).**

Learning Curves:

- **Showed convergence between training and validation scores, confirming reduced overfitting.**



Role of Cross-Validation:

- **Evaluated model stability and improved generalization by identifying overfitting risks.**

Visualization:

1. **Cross-validation scores plot.**
2. **Learning curves.**

Task 11: Model Evaluation and Error Analysis (10 Marks)

Evaluation on Test Data:

- **Pruned Tree Results:**
  - **MAE: 0.1065**
  - **MSE: 0.1084**
  - **R^2: 0.8688**

Residual Analysis:

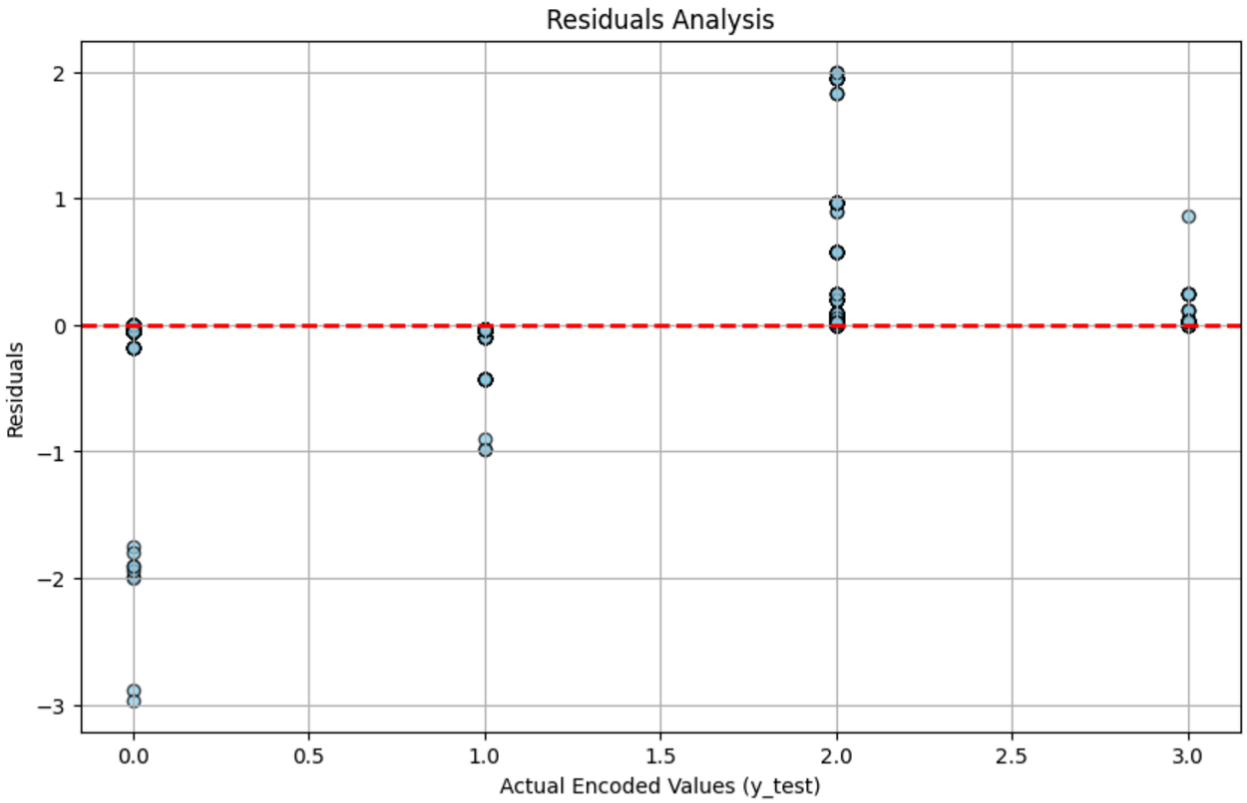
- **Residual mean: ~0.034 (indicating no significant bias).**
- **Scatter plot showed no major patterns.**

Feature Importance-Based Analysis:

- **Analyzed relationships between `Carpet_area`, `Per_sqft_price`, and predictions.**
- **RMSE for all features: ~0.329.**

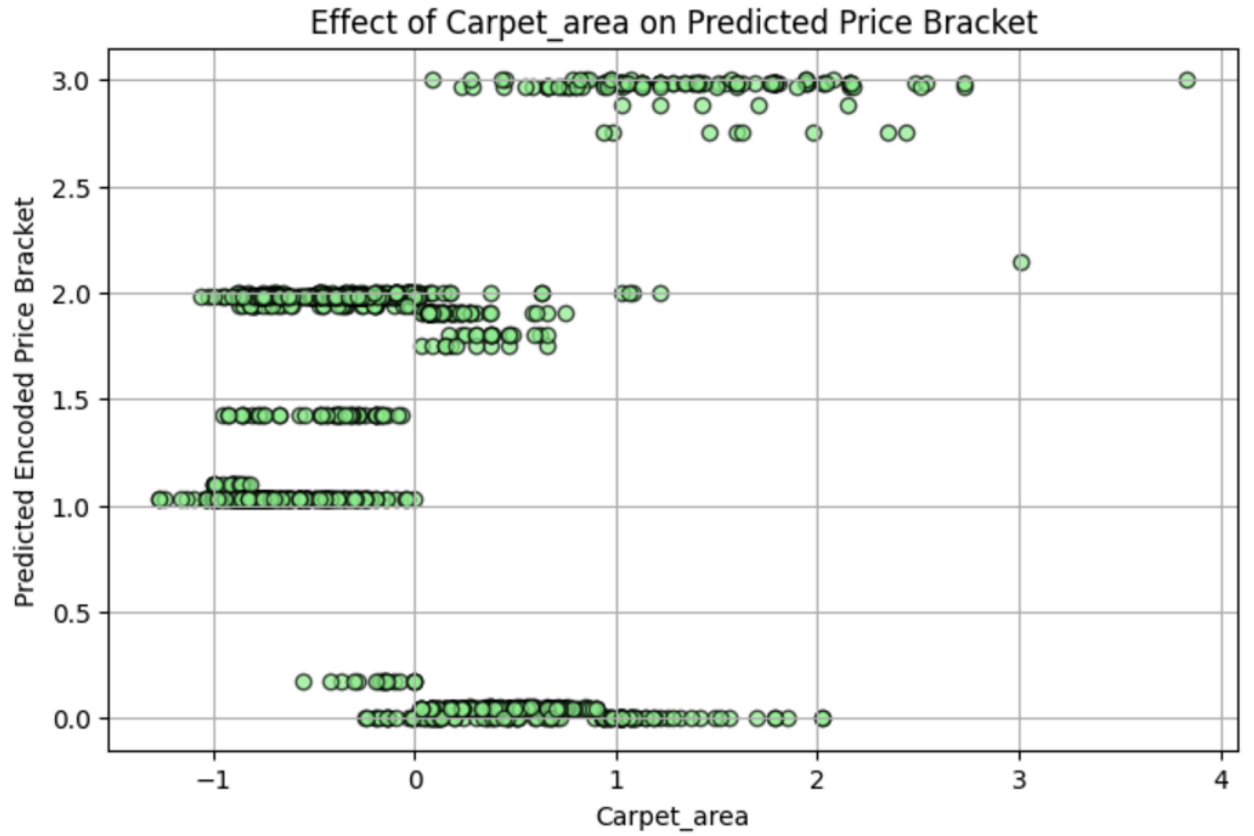
Visualization:

1. **Residual scatter plot.**

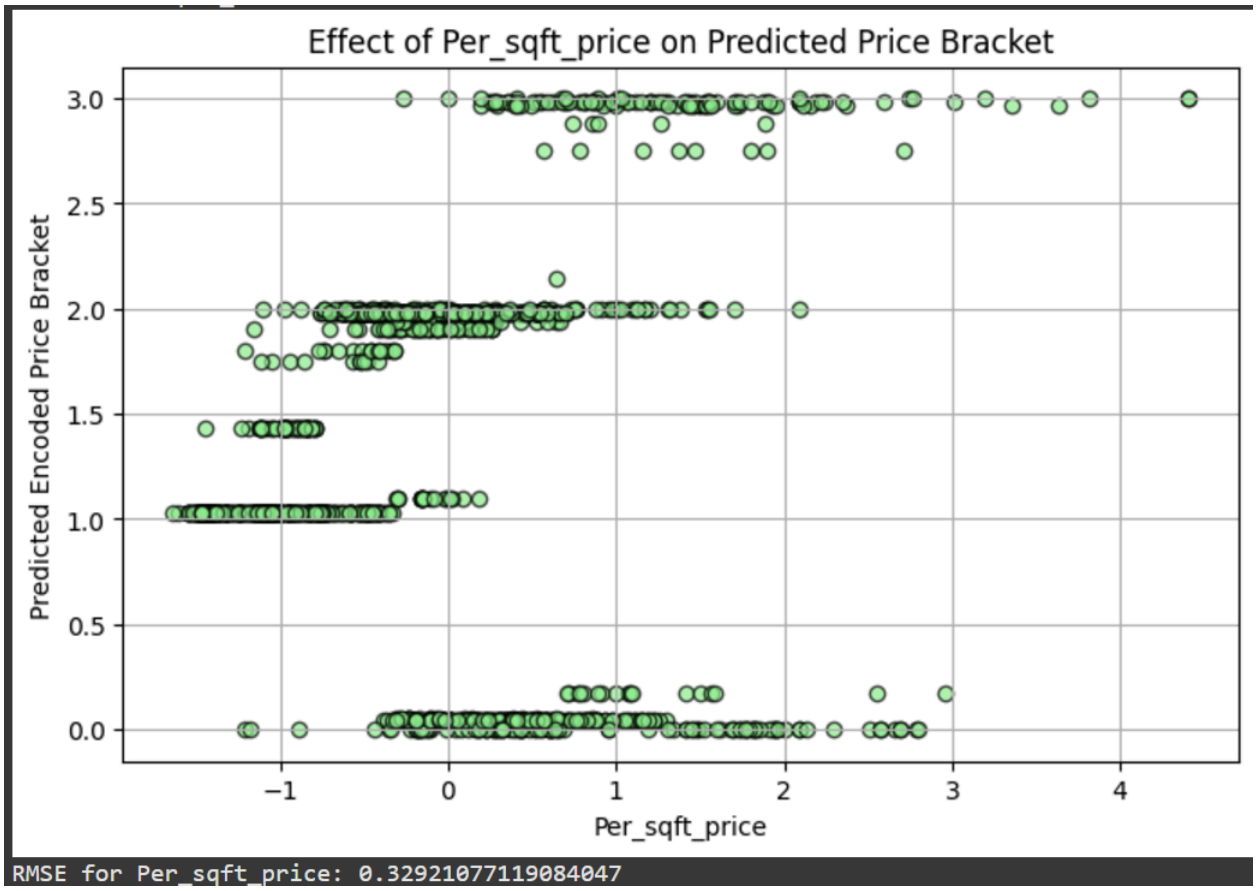



```
Residual Analysis Summary:
Mean of Residuals: 0.034085711530330354
Standard Deviation of Residuals: 0.32744143924912594
Residual Range (Min, Max): (-2.963855421686747, 2.0)
```

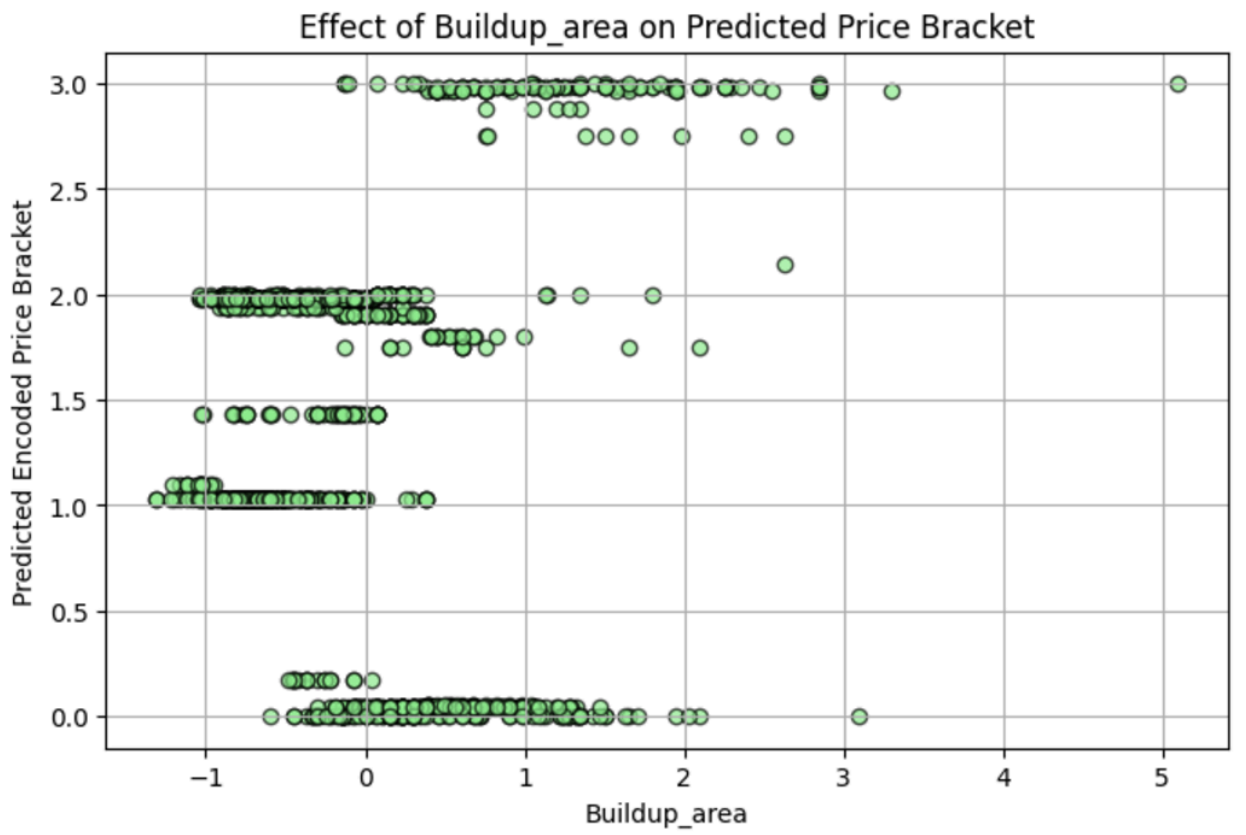
2. **Scatter plots for feature importance analysis.**




```
RMSE for Carpet_area: 0.32921077119084047
```




```
RMSE for Per_sqft_price: 0.32921077119084047
```




```
RMSE for Buildup_area: 0.32921077119084047
```

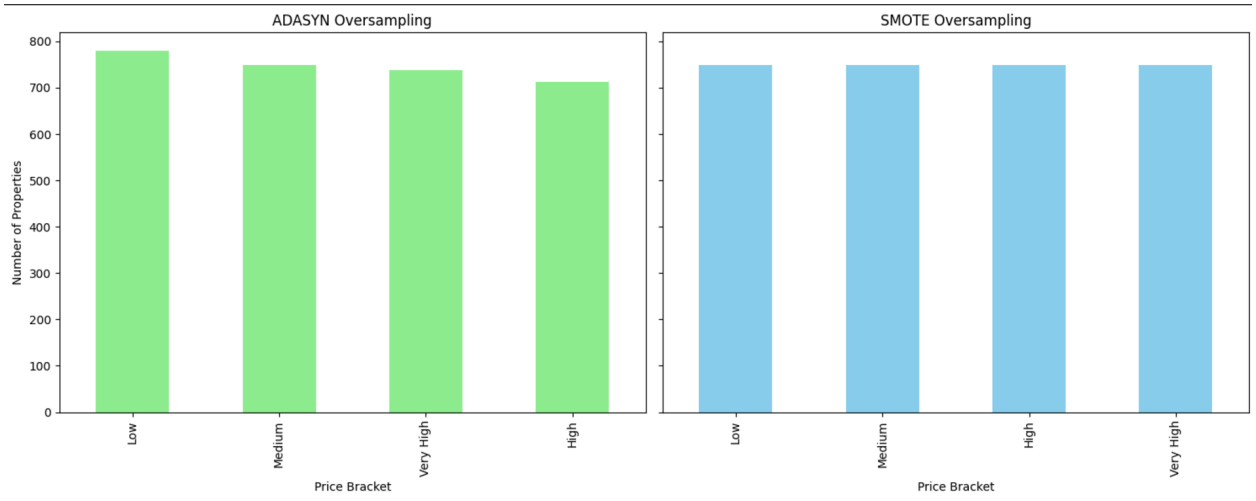
Bonus Task 1: Advanced Imbalance Handling (3 Marks)

ADASYN vs. SMOTE:

- **ADASYN prioritized harder-to-learn samples, creating 2979 samples.**
- **SMOTE created a more uniform dataset with 2992 samples.**

Visualization:

1. **Bar charts comparing ADASYN and SMOTE distributions.**




```
Dataset Sizes After Oversampling:
Original Dataset: (1564, 12)
ADASYN Oversampled Dataset: (2979, 12)
SMOTE Oversampled Dataset: (2992, 12)
```

Bonus Task 2: Ensemble Learning with Random Forest (3 Marks)

Random Forest Performance:

- **Validation Results:**
  - **MAE: 0.0939**
  - **MSE: 0.0602**
  - **R^2: 0.9520**
- **Comparison with Pruned Tree:**
  - **Random Forest outperformed the pruned tree, achieving better generalization and lower errors.**

Visualization:

1. **Random Forest performance comparison table.**

```
Random Forest Validation Set Results:
Mean Absolute Error (MAE): 0.093973288814691 14
Mean Squared Error (MSE): 0.06019749582637 73
R-squared (R²): 0.9520175373548792

Comparison with Pruned Decision Tree:
Pruned Tree MAE: 0.09983620880496988
Pruned Tree MSE: 0.10543984939229471
Pruned Tree R²: 0.915955579790811
```