**Boston Housing Price Prediction - Project Report**

**Introduction**

This project involves building regression models from scratch to predict house prices using the Boston Housing Dataset. The models implemented include Linear Regression and Random Forest. The dataset consists of 506 samples with 13 features related to housing conditions in Boston suburbs.

**Objective**

* Normalize numerical features and handle preprocessing
* Implement Linear Regression and Random Forest from scratch (excluding use of libraries like scikit-learn models)
* Evaluate performance using RMSE and R² metrics
* Visualize feature importance for tree-based models

**Dataset Overview**

The Boston Housing Dataset contains the following features:

* CRIM, ZN, INDUS, CHAS, NOX, RM, AGE, DIS, RAD, TAX, PTRATIO, B, LSTAT
* Target: PRICE (Median value of owner-occupied homes)

**Data Preprocessing**

* Normalized all numerical features using standard scaling
* No categorical variables needed encoding
* Data was split into training and testing sets using an 80-20 ratio

**Model Implementation**

**Linear Regression (from Scratch)**

* Implemented gradient descent for optimization
* Used a fixed learning rate and iteration count
* Learned weights and bias by minimizing the mean squared error

**Random Forest (from Scratch)**

* Built individual decision trees using variance reduction as the splitting criterion
* Bootstrapped samples for training each tree
* Aggregated predictions through averaging
* Calculated feature importance by summing variance reductions across all trees

**Personal Findings**

* Linear regression was easier to implement but had limitations in capturing non-linear patterns.
* Random forest significantly improved prediction performance due to its ensemble nature and better handling of variance.
* Feature importance visualization helped understand which features influenced housing prices the most — especially RM (average number of rooms), LSTAT (lower status population), and PTRATIO (pupil-teacher ratio).

**Problems Faced**

1. **Tree Splitting Logic**:
   * Challenge: Implementing efficient variance reduction and handling empty splits.
   * Solution: Skipped thresholds that resulted in invalid splits and added checks to ensure left/right branches had data.
2. **Feature Importance Calculation**:
   * Challenge: Aggregating importance from all trees and normalizing the values correctly.
   * Solution: Maintained a cumulative sum during training and divided by number of trees at the end.
3. **Numerical Instability in Gradient Descent**:
   * Challenge: Convergence issues in Linear Regression with high learning rate.
   * Solution: Reduced learning rate and increased epochs for more stable convergence.
4. **Model Evaluation Consistency**:
   * Challenge: Evaluating and comparing models on the same test set.
   * Solution: Fixed a random seed and used the same train-test split across all models.

**Counter Actions and Solutions**

* Implemented debugging logs and plotted learning curves to monitor model behavior
* Built modular functions to reuse decision tree logic for both Random Forest and later for XGBoost (planned)
* Careful documentation of importance gain for visualization

**Conclusion**

The project helped reinforce foundational understanding of regression and ensemble models. Building from scratch deepened insight into internal workings, limitations, and trade-offs in model design.