**Boston Housing Dataset Project Report**

**Dataset Selection**

For this project, we selected the **Boston Housing Dataset**, which includes various features about homes in Boston. The primary objective was to predict housing prices based on these features. This dataset is widely used for regression tasks, making it an ideal choice for our analysis and implementation of linear regression.

**Tasks and Methodology**

**1. Exploratory Data Analysis (EDA)**

**Objective:**

To gain insights into the dataset, identify patterns, and understand relationships between features and the target variable.

**Steps:**

1. **Dataset Inspection**:
   * Loaded the dataset and reviewed its structure using Python libraries.
   * Analyzed data types, column descriptions, and checked for missing values.
2. **Descriptive Statistics**:
   * Calculated key statistics such as mean, median, and standard deviation to understand the distribution of features.
3. **Visualizations**:
   * **Histograms** and **boxplots** were used to analyze feature distributions and detect outliers.
   * **Scatter plots** and a **correlation heatmap** helped identify relationships between features and the target variable (MEDV, median home price).
   * **Pair plots** and **violin plots** further highlighted interactions among features.

**2. Data Preprocessing**

**Objective:**

To prepare the data for modeling by handling missing values, scaling features, and splitting the dataset.

**Steps:**

1. **Handling Missing Values**:
   * Checked for and addressed missing values in the dataset. No significant issues were found.
2. **Feature Scaling**:
   * Standardized features using StandardScaler to ensure uniform scaling, as gradient descent is sensitive to feature magnitudes.
3. **Data Splitting**:
   * Divided the dataset into training and testing sets with an 80:20 ratio to evaluate model performance on unseen data.

**3. Model Implementation**

**Objective:**

To implement linear regression using gradient descent from scratch and compare it with scikit-learn's implementation.

**Steps:**

1. **Linear Regression with Gradient Descent**:
   * Initialized weights and biases randomly.
   * Implemented the hypothesis function to predict target values.
   * Computed the cost function (Mean Squared Error).
   * Updated weights iteratively using the gradient descent algorithm until convergence.
2. **Comparison with Scikit-learn**:
   * Used scikit-learn’s LinearRegression class for a benchmark model.
   * Compared coefficients and evaluated performance using MSE and R-squared metrics.

**4. Hyperparameter Tuning**

**Objective:**

To optimize the learning rate for gradient descent and analyze its impact on convergence.

**Steps:**

1. **Learning Rate Experimentation**:
   * Tested different learning rates (e.g., 0.01, 0.1, 0.001) to observe their effects on model convergence.
   * High learning rates caused divergence, while low rates led to slow convergence.
2. **Cost Function Visualization**:
   * Plotted the cost function values over iterations to illustrate convergence behavior for various learning rates.

**5. Model Evaluation**

**Objective:**

To assess the performance of the models using appropriate metrics.

**Steps:**

1. **Evaluation Metrics**:
   * **Mean Squared Error (MSE)**: Quantified the average squared difference between predicted and actual values.
   * **R-squared**: Measured the proportion of variance in the target variable explained by the model.
2. **Performance Comparison**:
   * Both the gradient descent implementation and scikit-learn's linear regression achieved comparable results, with minor differences in efficiency.
3. **Limitations**:
   * Discussed the challenges of gradient descent, including sensitivity to the learning rate, reliance on feature scaling, and potential for local minima in cost optimization.

**Conclusion**

* This project successfully demonstrated the implementation of linear regression using gradient descent from scratch and compared it with an optimized library-based solution.
* EDA provided valuable insights into the Boston Housing Dataset, while preprocessing ensured the data was ready for modeling.
* Hyperparameter tuning, particularly the learning rate, played a critical role in the effectiveness of gradient descent.
* The evaluation metrics showed that both implementations performed well, confirming the validity of the custom gradient descent approach.

This project highlights the practical application of machine learning concepts, from data analysis to model implementation and evaluation.