**Lab 4**

**School of Computer Science Engineering and Technology**

|  |  |  |  |
| --- | --- | --- | --- |
| Course | B. Tech. | Type | Core |
| Course Code | CSET301 | Course Name | Artificial Intelligence and Machine Learning |
| Year | 2025 | Semester | Odd |
| Date | 04/08/2025 | Batch | 2023–2027 |

**CO-Mapping**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** | **CO6** |
| Q1 |  | √ | √ |  |  |  |

**AI/ML Lab – Linear Regression with scikit-learn**

### **Objective: Total Marks: 0.5**

This lab aims to introduce students to building a simple linear regression model using Python and scikit-learn. Students will explore how to train, evaluate, and interpret a regression model on real-world data.

**Problem Statement:**

You are given a dataset containing information about housing prices in California. Your task is to build a linear regression model to predict **median house value** based on selected numerical features.

You can load the dataset directly from sklearn.datasets.

Link for the ref: <https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_california_housing.html>

A screenshot of a computer

Description automatically generated

**Instructions:**

Perform the following tasks in your Colab:

1. Load the California Housing dataset from sklearn.datasets into your Python environment using the fetch\_california\_housing() function.
2. Convert the dataset into a Pandas DataFrame to facilitate exploration and manipulation of the features and target variable.
3. Perform exploratory data analysis (EDA) by visualizing the distribution of key variables and checking for any correlations using tools like seaborn, matplotlib, and pandas.
4. From the available features, select only the relevant numerical attributes that are expected to influence the housing price prediction (e.g., average number of rooms, median income).
5. Divide the dataset into training and testing subsets using train\_test\_split() from sklearn.model\_selection, typically with an 80-20 split for training and testing respectively.
6. Initialize and train a **Linear Regression** model from sklearn.linear\_model using the training set's input features and target values.
7. Evaluate the trained model on the test set using performance metrics such as the R² score and Mean Squared Error (MSE) from sklearn.metrics to quantify prediction accuracy.
8. Generate a scatter plot comparing predicted versus actual median house values to visually assess the model’s prediction capability and identify any patterns or biases.
9. Interpret the evaluation metrics and comment on the model's strengths and weaknesses, and discuss possible limitations such as linearity assumptions, outliers, or feature multicollinearity.

You may use tools like pandas, numpy, matplotlib, seaborn, sklearn.