School of Computer Science Engineering and Technology

Course-B.Tech.	**Type-** Specialization Core II
Course Code- CSET228	**Course Name-** Data Mining and Predictive Modelling (Lab)
Year- 2025	**Semester-** Even
Date- 15/02/2025	**Batch-**

CO-Mapping

Q(s)	CO1	CO2	CO3
Q1			√
Q2		√	

Objectives

1. Students will be able to gain a understanding of supervised learning (binary classification).
2. Students will be able to gain a deeper understanding of Simple Linear regression.

Questions:

1. Given a data for detecting credit card fraud based on two feature: distance_from_home and ratio_to_median_purchase_price. The third column is dependent column is having the binary value 0 for normal and 1 for fraud.

a. import the data and visualize it in scattered plot. Normal and fraud for different colors.

b. Divide the features and labels into x and y.

c. Split the train and test into 80:20 ratio.

d. Normalize the x_train, and x_test using standard scaler.

e. Fit KNeighborsClassifier on x_train and y_train.

f. Predict x_test.

g. Calculate accuracy using y_pred and y_test

link to download the dataset: https://www.kaggle.com/datasets/mlg-u...

2. Suppose you've trained a Simple Linear Regression model on the student performance dataset using gradient descent. You've experimented with different values of max_iterations and learning rates. However, you notice that for certain combinations of hyperparameters, the model converges very slowly and may not yield satisfactory results even after many iterations. 1. Follow the following steps.

I. Download the Students performance dataset available on UCI repository (https://archive.ics.uci.edu/ml/datasets/student+performance) which consists of a total of 32 attributes.

II. **Read** the dataset (use **read_csv()** from **pandas**) into some variable. Take the last two columns (G2 and G3) into XY.

III. Print the different statistical values of data contained in XY using **describe()** function from **pandas**.

IV. Divide XY into X consisting of G2 and Y consisting of G3. Print the shape of both.

V. Add a column at position 0 with all values=1.

VI. Print some of the rows from XY.

VII. Complete the following functions given in the provided Ipython Notebook to implement a Linear Regression model between X and Y (Y = mX + C).

- Write code to predict G3 for a given set of weights and input G2.
- Write a function to calculate the loss (mean squared error) for given set of weights, input G2 and actual output G3.
- Write a function to calculate the gradient for given set of weights, input G2 and actual output G3.
- Write a function to perform gradient decent for given set of input G2 and actual output G3.

VIII. Play with different values of max_iterations and the learning rate.

Additional fun (will not be evaluated)

IX. Split the data in X_train, X_test, Y_train, Y_test (sklearn.model_selection.train_test_split function)

X. Calculate mean squared error on both X_train and X_test.

XI. Generalize the code for multivariate(multiple) linear regression.

Helpful links

- Scikit-learn documentation for linear regression:
  https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html
- Read till where you feel comfortable:
  https://jakevdp.github.io/PythonDataScienceHandbook/05.06-linear-regression.html

3.