भारतीय सूचना प्रौद्योगिकी संस्थान गुवाहाटी
INDIAN INSTITUTE OF INFORMATION TECHNOLOGY GUWAHATI

CS 360: Machine Learning Lab
Evaluation Assignment 4
Total Marks: 30

Instructions: This assignment is for evaluation and marks will be awarded. You need to complete execution by 12 PM. The saved results folder (consisting .csv files, plots, word file) has to be submitted through a Google form, which will be shared by Teaching Assistant.

1. Download the "Iris" dataset from mail.
   Write a program to do the followings:
   (a) Read the dataset.
   (b) Normalize the features, if needed.
   (c) Split the datasets into **training: testing** sets in the ratio of 80:20.
   (d) Apply K-NN classification algorithm (without using in-built Python packages) to predict the species-types. Consider 5 different K-values, randomly where$K \leq \sqrt{n}$ and n= number of observations in the dataset, and implement the algorithm, separately for each K-value.
   (e) Compute precision, recall and overall accuracy w.r.t. all the K-values and, report and visualize the results using different plots.
   (f) Apply distance-weighted K-NN algorithm for the above dataset. Consider the weight function as $\frac{1}{d}$ and $\frac{1}{d^2}$, where d= distance between the test observations and the training observations.
   Consider Euclidean distance as the distance measure, and the K-values as 1(d).
   (g) Compute precision, recall and overall accuracy w.r.t. both the weight functions, and all the K-values, report and visualize the results using different plots.
   (h) Compare and interpret the results from 1(e) and 1(g) w.r.t. the best K-values.

2. Write a program to do the followings:
   (a) Create a synthetic binary dataset with 300 observations and 2 features. The observations should have numeric values for the two features, and binary values for the target.
   (b) Split the datasets into **training: validation: testing** sets in the ratio of 70:10:20.
   (c) Apply Perceptron learning algorithm (without using in-built Python packages) to predict the labels ($\hat{y}$).
   Consider the followings to implement the algorithm:
   - range of initial weight values : {-0.5, +0.5}
   - epoch : {50, 100, 150}
   - bias, x_0=1.

Hint: Weight update rule:

- w= w+ x_i; if $w.x \leq 0, y = 1$.
- w= w- x_i; if $w.x \geq 0, y = 0$.

Stopping criteria: Attaining maximum number of epochs, or zero misclassifications.

(d) Compute precision, recall and overall accuracy for three different weight initialization scenarios, and report and visualize the results, considering all the epochs under the three scenarios.