Cairo University, Faculty of Computers
and Artificial Intelligence

FACULTY OF COMPUTERS AND AI, CAIRO UNIVERSITY

Supervised Learning
Year ٢٠٢٣-٢٠٢٤

Second Semester

Assignment #١

Supervised Learning
Lab ١

Cairo University, Faculty of Computers and Artificial Intelligence

Assignment #١

- In this assignment, you will practice using the **KNN** (k-Nearest Neighbors) algorithm to solve a classification problem.

- In this assignment, the Iris dataset will be used. The dataset was first introduced by statistician R. Fisher and consists of ٥٠ observations from each of three species Iris (*Iris* setosa, *Iris virginica* and *Iris versicolor*). For each sample, ٤ features are given: the sepal length and width, and the petal length and width.

- Download `iris.csv` file from the Data Folder. The Data Set description with the definitions of all the columns can be found on the dataset page. ***Iris Dataset***

- Load the data from the file (`iris.csv`) into the DataFrame. Set the names of columns according to the column definitions given in Data Description.

- **Data inspection.** Display the first ٥ rows of the dataset and use any relevant functions that can help you to understand the data. Prepare ٢ scatter plots - `sepal_width` vs `sepal_length` and `petal_width` vs `petal_length`. Scatter plots should show each class in a different color (`seaborn.lmplot` is recommended for plotting).

- **Prepare the data for classification**. Using the pandas operators prepare the feature variables `X` and the response `Y` for the fit.

- **Split** the data into `train` and `test` using `sklearn` `train_test_split` function.

- **Run the** `KNeighborsClassifier` that you have made from scratch. on the training set, using the ٣ distance measures(***create them from scratch***)

- Use the learning model to **predict the class from features**, run prediction on `X` from test part. Show the **accuracy score** of the prediction by comparing predicted iris classes and the `Y` values from the test, using 'accuracy_score()'. Also, comparing these two arrays (predicted classes and test Y) count the numbers of correct predictions and predictions that were wrong.