

## Assignment 1: Explore different Classification Models In Python

Objective: In This Assignment, You Will Practice Using The **KNN (K-Nearest Neighbors)**, **Logistic Regression**, **Decision Tree**, **SVM Algorithms**, and **Gaussian Naive Bayes** To Solve A Classification Problem.

The goal Is To train a machine learning model To distinguish the flower species from one another in the IRIS Dataset which Can Be Downloaded From UCI Machine Learning Repository: <https://archive.ics.uci.edu/ml/machine-learning-databases/iris/> (Links to an external site.)

The Data Set description with the definitions of all the columns can be found on the dataset page - <https://archive.ics.uci.edu/ml/datasets/Iris> (Links to an external site.)

- ✓ 1- Load the data from the file (`iris.data`) into the DataFrame. Set the names of columns according to the column definitions given in Data Description.
- ✓ 2- Data inspection. Display the first 5 rows of the dataset and use any relevant functions that can help you to understand the data. Prepare 2 scatter plots - `sepal\_width` vs `sepal\_length` and `petal\_width` vs `petal\_length`.
- 3- Check data for **outliers** and **missing values** then Prepare the data for **classification**. Using the pandas operators prepare the feature variables `X` and the response `Y` for the fit. Note that **`sklean` expects data as arrays, so convert extracted columns into arrays**. **Split the data into `train` and `test`** using `sklearn` `train\_test\_split` function.
- 4- Run the fit using **`KNeighborsClassifier`** from `sklearn.neighbors`. First, instantiate the model, Then, run the classifier on the training set.
- 5- Use 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. Comparing these two arrays (predicted classes and test `Y`), count the numbers of correct predictions and predictions that were wrong.

6-For the `KNeighborsClassifier`, we want to see how accuracy score and the number of correct predictions change with the number of neighbors `k`. We will use the following number of neighbors `k`: 1, 3, 5, 7, 10, 20, 30, 40, and 50

7-Generate 10 random train/test splits for and Fit the model for each split and generate predictions Average the accuracy score for each `k` Calculate the average number of correct predictions for each `k` as well Plot the accuracy score for different values of `k`. What conclusion can you make based on the graph?

**8- Repeat steps 4-7 Using the different ML models changing the model parameters**

What conclusion can you make based on the results?