C S 519 Applied Machine Learning I
Compare classifiers in scikit-learn library

1. Objective

In this **individual** homework, you are required to understand and compare several classification algorithms that are provided by the Python scikit-learn library (https://scikit-learn.org/stable/).

2. Requirements

2.1 Tasks

(1) **[50 points]** Write classification code by utilizing several scikit-learn classifiers: (i) perceptron, (ii) logistic regression, (iii) linear support vector machine (SVM), (iv) non-linear SVM using Radial Basis Function (RBF) kernel, (v) decision tree, and (vi) KNN.

(2) **[15 points]** Each classifier needs to be tested using the digits dataset offered by scikit-learn library. Please partition the data to be split as training and testing (80:20) sets. The analysis should go to a report (detailed requirements see below). Note that you do not need to submit the digits dataset.

(3) **[15 points]** Run (i) logistic regression, (ii) SVM (non-linear using RBF kernel), and (iii) decision tree on a second dataset of your own choice and add the analysis to the report (detailed requirements see below). For the second dataset, please also use 80:20 training and testing split. You need to clearly denote the data source in the report. You also need to submit the second dataset.

(4) **[15 points]** Please create a report that properly analyzes the classifiers behavior by applying the knowledge discussed in this course.

(a) [5 points] For each classifier, you should report the accuracy of the model's performance on both the training and testing data. For the reported accuracy, the report should clearly denote the model setting (parameter values).

(b) [7 points] For each classifier, report performance when tuning one (one is sufficient) hyper-parameter. For example, you can tune **η** (eta) for the perceptron model; for logistic regression, you can tune the **C** value.

(c) [3 points] For each classifier, you should report both the training and testing time.

Please note that your analysis of the digits dataset will be worth 9 points (a. 3pts, b. 4pts, c. 2pts). While your analysis of the 2nd dataset will be worth 6 points (a. 2pts, b. 3 pts, c. 1pt).

(5) **[5 points]** Write a readme file readme.txt with detailed instructions to run your program.

2.2 Other requirements

- Your Python code should be written for Python version 3.10 or higher.
- Please write proper comments in your code to help the instructor and teaching assistants to understand it.

- Please properly organize your Python code (e.g., create proper classes, modules). You can submit your code as a Jupyter Notebook or .py files.

3. Submission instructions

Put all your files (Python code, readme file, report, datasets, etc.) to a zip file named **hw3_<YourName>.zip** and upload it to Canvas.

4. Grading criteria

- **ZERO point** will be given if your code does not work. Please do not submit code that you did not test and make sure it works.
- The score allocation has been put beside the questions.
- FIVE points will be deducted if files are not submitted in the required format.
- If the total points are more than 100. Your grades will be scaled to the range of [0,100].
- Please make sure that you test your code thoroughly by considering all possible test cases.