Project 1: Classification Algorithms

Introduction:

In this project, you are asked to use and compare different classification algorithms. Each team (1~2 people) should submit codes and a report via Canvas.

Dataset Description:

Two datasets (***project1_dataset1***, ***project1_dataset2***). Here is a short description of the datasets:

Each line represents one data sample.
The last column of each line is class label, either 0 or 1.
The rest columns are feature values, each of them can be a real-value (continuous type) or a string (nominal type).
project2_dataset1: 569 observations, 31 attributes
project2_dataset2: 462 observations, 10 attributes

Complete the following tasks (both project1_dataset1, project1_dataset2):

- Apply four classification algorithms: **Nearest Neighbor**, **Decision Tree**, **Naïve Bayes**, and **SVM**. (Normalize the data to avoid scaling issue, and/or apply regularization to avoid overfitting if needed.)
- Apply **AdaBoost** based on your Decision Tree.
- Train a **neural network** for classification, the architecture and hyperparameters can be designed and tuned by yourself, try to achieve better results.
- Adopt **10-fold Cross Validation** to evaluate the performance of all methods on the provided two datasets in terms of **Accuracy, Precision, Recall, and F-1 measure**.
- **Note: You may use existing packages (e.g., scikit-learn, Pytorch, Numpy)**

Project Submission:

• Prepare your submission. **One team only needs to provide one submission**. Make a zipped folder named "**CaseID_CaseID_Proj1.zip**", where "CaseID" refers to your group members' Case IDs. In the folder, you should include:

1. **Report**: A pdf file named ***report.pdf***. Describe the flow of all the methods, and briefly describe the choice you make (such as parameter setting, pre-processing, post-processing, how to deal with over-fitting, etc.). Compare their performance, and state their pros and cons based on your findings. **Each report should NOT exceed 4 pages**. Suggested template is attached, please use double column.
2. **Code**: A zipped folder named ***code.zip***, which contains all codes used in this part (preferably, each algorithm has a separate .py file with informative file name). Inside the folder, please also provide a **README file** which describes how to run your code.

Note that copying code/results/report from another group or source is not allowed and may result in an F in the grades of all the team members.

Grades will be given based on the following criterion:

- Report (60 pts):
  - (12 pts) Algorithm description: brief description of the workflow of each of the 6 algorithms
  - (25 pts) Result evaluation and hyperparameter tuning: cross validation, and sufficient discussion on how your group conducts pre-/post- processing, parameter selection
  - (15 pts) Performance comparison and sensitivity analysis: explanation of the performance differences and how the performances vary with hyperparameter selection
  - (8 pts) Overall coherence and clarity
- Code (40 pts)
  - (35 pts) Correctness and Reproducibility of the results in the report
  - (5) Readme: Readability and clarity