Fundamentals of Machine Learning (2022 Spring)

Programming Assignment: Classification of Titanic Data Set

1. **Benchmark Dataset**: This is the problem of predicting survivals based on the information of the people on board the Titanic. You should evaluate the performance of each model using the machine learning models presented in the assignment. You can download the dataset from the following website: https://www.kaggle.com/c/titanic

   In this assignment, the "**train.csv**" file will be used for both model training and testing.

2. **Preprocessing**

   A. You will split the data in the "train.cvs" to training and test data as follows:

      - Training data: PassengerId 1-600
      - Test data: PassengerId 601-891

   B. Do **NOT** use the following features for training your model:

      PassengerId, Name, Ticket, Cabin

   C. **Remove** the data with **missing values** from the training and test data.

3. **Machine Learning Models**: Use scikit-learn to implement the following three machine learning models and analyze their performance using the evaluation methods (See Section 4)

   A. **K-Nearest Neighbors(KNN) (sklearn.neighbors.KNeighborsClassifier)**

      - Analyze how the performance in the test data is changed while changing K to **[2-5]**

   B. **Logistic Regression (sklearn.linear_model.LogisticRegression)**

      - Analyze how the results change in the test data while changing the number of iterations (**max_iter**) to **20, 40, 60, 80, 100**.
      - After fixing the number of iterations to **100**, change the inverse of regularization strength (**C** in scikit-learn) by **1** in the range of **[1 to 5]**. Analyze how the results change in the test data.

C. **Decision Tree (sklearn.tree.DecisionTreeClassifier)**

- Analyze the separation criteria of the **first** and **second** depths in the decision tree with information gain (by handwriting or typing it).
- When **max_depth**=None, use an appropriate tool to visualize the tree to know the condition and gain values at each depth.
- Analyze how the results change in the test data when **max_depth** is changed to [1~3, None].

D. **Compare the performance of A, B and C.**

- Which model shows the best performance? Also, worst performance?
- Why do you think the model show the best or the worst performance?

4. **Evaluation Methods**: Show the performance according to each model through **Accuracy** and **F1-Score**.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}, \quad Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}, \quad F_1 = \frac{2 \cdot Precision \cdot Recall}{Precison + Recall},$$

$$TP = true\ positive, \quad TN = true\ negative, \quad FP = false\ positive, \quad FN = false\ negative.$$

5. **Submission Form**

A. **Submit a zip file** including **a report and python files**.

B. TA will copy the original train.csv file to your folder and run your python codes. Then, your python files should be run without errors and print out the performance results in your report.

C. **The file name**: name.zip (e.g., 2020714950_Hong_Gil-dong.zip).

D. You can submit .ipynb files instead of .py files.