National Institute of Technology Calicut
Department of Computer Science and Engineering
CS4038D DATA MINING - Assignment 2

Submission deadline (on or before):
12 th November 2023, 10:00:00 PM - There won't be further extension - Fixed Deadline.

Each group in this Google Sheet can work together to complete the following Assignment Question over the following dataset, and must submit your assignment in the moodle (Eduserver) course page, on or before the submission deadline. Only one member among your team should make the submission in Eduserver on behalf of the entire team. Include a README.PDF which contains the name and roll number of the group members. Total of 5 files (4 python files [task 1, 2-2files and 3-2files], RESULTS.PDF [task 4] and the README.PDF) is expected to be submitted as part of this assignment.

During evaluation, the genuinity of the submission and contribution of each member will be checked either through viva/quiz. The total marks for the assignment is 12. The marks awarded will be based on the uploaded documents, the viva/quiz and the relative performance of the teams. Any sort of copying/ plagiarism in the submission content will result in zero marks for the assignment.

Assignment Question

Dataset:
The dataset that should be used for the following tasks is available in this link. The dataset lists the passenger details of those who survived the Titanic disaster. There are three features - pclass (passenger class), age and gender; and the output to be predicted is survived (yes/no). Use the given labels for test data to calculate the evaluation measures.

1. Implement Decision Tree classification algorithm in Python from scratch (do not use libraries) using Information Gain as the splitting measure. Grow the complete tree for the given training dataset, and test the given dataset. Create the confusion matrix, and find out the accuracy, precision and recall. You can write the code for training, testing and calculating of measures in a single file.

2. Implement Naive Bayes classifier from scratch (do not use libraries). You should use Laplacian correction. After calculating the likelihood and prior probabilities in trainNBayes.py, save the probability values to a txtfile. The testNBayes.py should read the probabilities from the txtfile, and can use it for predicting the class labels for test dataset, and finally find accuracy, precision and recall by finding the confusion matrix.

3. Use the scikit-learn library to create Decision Tree (using Information Gain), Naive Bayes classifier in a single python file named compareModels.py, and compare with the results of task 1 and 2. Implement a Random Forest Classifier using scikit-learn in the python file named applyRandomForest.py, to improve the accuracy measure obtained using the single classification tree in compareModels.py. You are allowed to do any sort of tuning on the parameters of Random Forest to achieve the improvement in accuracy. The optimized model parameters and the performance of the model should be tabulated well in the document mentioned in task 4.

4. In a document, tabulate the evaluation measures obtained for tasks 1, 2 and 3, and write down your inference on the misclassified samples, and on the comparative performance of all the different classifier models implemented for the task 1, 2 and 3. Intentionally not specifying the content format in this document, you may order the contents in a good presentation format.