

# Assignment 1

## Instructions:

- 1) Use **Python** programming. You may use **numpy**, **pandas** and **matplotlib** libraries.
- 2) Handle missing data as and when required using any approach.
- 3) There are two questions, each of 50 marks. You will be submitting two python code files named as “**q1.py**” and “**q2.py**”.
- 4) You will prepare a **README** file to explain how to execute your code.
- 5) You will print the outputs in a “**.txt**” file and also provide the plots.
- 6) All source code files, results files and documents should be kept in a folder named “**roll1\_and\_roll2\_a1**”. **Zip the folder and upload it on Moodle.**
- 7) **Use Dataset C for both Decision Tree and Bayesian. Consider the last column of your dataset as label.**

## Question 1: Decision Tree (50)

- 1) Split Dataset C into 80%-20% to form training and testing sets, respectively. Build a **Decision Tree Classifier** using ID3 algorithm. Train the classifier **using Information Gain (IG) measure (no packages to be used for Decision Tree Classifier)**.
- 2) Repeat (1) for 10 random splits. Print the best test accuracy and the depth of that tree.
- 3) Perform **reduced error pruning** operation over the tree obtained in (2). Plot a graph showing the variation in test accuracy with varying depths. Print the pruned tree obtained in hierarchical fashion with the attributes clearly shown at each level.
- 4) Prepare a **report** including all your results.

[20+5+20+5]

## Question 2: Bayesian (Naïve Bayes) Classifier (50)

- 1) Randomly divide Dataset C into 80% for training and 20% for testing. Encode categorical variables using appropriate encoding method (**in-built function allowed**).
- 2) A feature value is considered as an outlier if its value is greater than mean + 3 x standard deviation ( $\mu + 3 \times \sigma$ ). A sample having maximum such outlier features must be dropped. Print the final set of features formed. Normalise the features as required.
- 3) Train the Naïve Bayes Classifier using 10-fold cross validation (**no packages to be used for Naïve Bayes Classifier**). Print the final accuracy.
- 4) Train the Naïve Bayes Classifier using **Laplace correction** on the same train and test split. Print the final accuracy.
- 5) Prepare a **report** including all your results.

[5+10+20+10+5]