# ELG 5255 Applied Machine Learning Summer 2021

# Assignment 1 (Decision Tree and Ensemble Learning)s

## Submission

You must submit your assignment on-line with Bright Space. This is the only method by which we accept assignment submissions. We do not accept assignments sent via email, and we are not able to enter a mark if the assignment is not submitted on Bright Space! The deadline date is firm since you cannot submit a assignment passed the deadline. It is a student's responsibility to ensure that the assignment has been submitted properly. A mark of 0 will be assigned to any missing assignment.

## Part 1: Numerical Questions:

Part 1 is not programming questions and should be solved manually!

Let's assume that TAs would go hiking every weekend, and we would make final decisions (i.e., Yes/No) according to weather, temperature, humidity, and wind. Please create a decision tree to predict our decisions.

Weather (F1)	Temperature (F2)	Humidity (F3)	Wind (F4)	Hiking (Label)
Cloudy	Hot	High	Weak	No
Sunny	Hot	High	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Cloudy	Mild	High	Strong	Yes
Rainy	Mild	High	Strong	No
Rainy	Cool	Normal	Strong	No
Rainy	Mild	High	Weak	Yes

Weather (F1)	Temperature (F2)	Humidity (F3)	Wind (F4)	Hiking (Label)
Sunny	Hot	High	Strong	No
Cloudy	Hot	Normal	Weak	Yes
Rainy	Mild	High	Strong	No

Please show the whole process. You will not receive any marks if you only show the final results.

Q1 (20 Marks): Please build a decision tree by using Gini Index (i.e., $Gini = 1 - \sum_{i=1}^{N_C}(p_i)^2$, where N_C is the number classes)

Q2 (20 Marks): Please build a decision tree by using Information Gain (i.e., $IG(T,a) = Entropy(T) - Entropy(T|a)$). More information about IG

Q3 (10 Marks): Please compare the advantages and disadvantages between Gini Index and Information Gain

## Part 2: Programming Questions

### Dataset

To plot figures and understand how ensemble algorithms work, we will use a 2D dataset generated by make_circles.

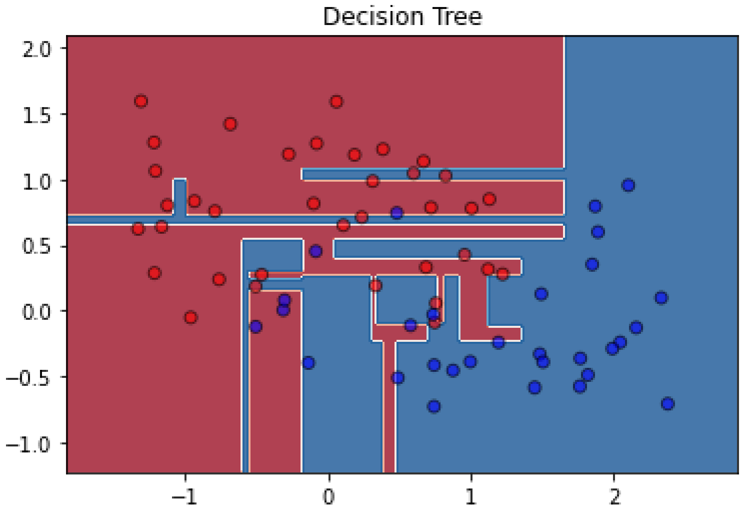
~~~~
``` python
from sklearn.datasets import make_circles
from sklearn.model_selection import train_test_split

rs = 0
X, y = make_circles(300, noise=0.1, random_state=rs)
trX, teX, trY, teY = train_test_split(X, y, test_size=0.33, random_state=rs)
```
~~~~

### Decision Tree

Q4 (5 Marks): Apply decision tree to classify testing set, get the accuracy of the result, and plot the decision boundary as we showed in the lab.

Decision boundary example:



### Bagging

Bagging is to generate a set of bootstrap datasets, create estimators for each bootstrap dataset, and finally utilize majority voting (soft or hard) to get the final decision.

Q4 (25 Marks): Using decision tree as base-estimator and write bagging algorithm from scratch, set the number of estimators as 2, 5, 15, 20 respectively, and generate the results accordingly (i.e., accuracy and decision boundary)

Q5 (5 Marks): Explain why bagging can reduce the variance and mitigate the overfitting problem

### Boosting

Q6 (15 Marks): There are 2 important hyperparameters in AdaBoost, i.e., the number of estimators (ne), and learning rate (lr). Please plot 12 subfigures as the following table's setup. Each figure should plot the decision boundary and each of their title should be the same format as `{n_estimaotrs}, {learning_rate}, {accuracy}`

ne=10; lr=0.1	ne=50; lr=0.1	ne=100; lr=0.1	ne=200; lr=0.1
ne=10; lr=1	ne=50; lr=1	ne=100; lr=1	ne=200; lr=1
ne=10; lr=2	ne=50; lr=2	ne=100; lr=2	ne=200; lr=2