National University of Singapore
School of Computing
CS3244: Machine Learning
Tutorial 2

Decision Trees and Ensemble Methods

1. **Introduction to Decision Trees.**

The data in table 1 represents the three states (S_1, S_2, S_3) which contribute to the lighting of a bulb (the final state F). Each state takes value from the set $\{0, 1\}$.

S_1	S_2	S_3	F
0	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Table 1: States and the final outcome

(a) Construct a decision tree to classify the final outcome (F) from the three initial states S_1, S_2 and S_3. Follow a greedy way to construct a decision tree with feature order S_1, S_2 and S_3 which can fit the dataset with 0 training error.

(b) Comment on the tree from part (a). Is the tree optimal? If it is not optimal construct an optimal decision tree. (Here optimality is decided from the depth of a DT.)

(c) Alice tries to implement a XOR function which has d inputs using decision tree (DT). Why using DT is not a scalable solution? Explain your answer. If we implement AND or OR function with d inputs, do we get any advantage over the XOR function?

2. **Bank on Decision Trees.** The loans department of DBN (Development Bank of NUS) has the following past loan processing records each containing an applicant's income, credit history, debt, and the final approval decision. Details are shown in Table 2.

(a) Construct a decision tree based on the above training examples. (Note: $\log_2 \frac{x}{y}$ = $\log_2$ x - $\log_2$ y, $\log_2 1 = 0$, $\log_2 2 = 1$, $\log_2 3 = 1.585$, $\log_2 4 = 2$, $\log_2 5 = 2.322$, $\log_2 6 = 2.585$, $\log_2 7 = 2.807$, $\log_2 8 = 3$, $\log_2 9 = 3.170$, $\log_2 10 = 3.322$, $\log_2 11 = 3.459$, and $\log_2 12 = 3.585$)

(b) Construct 3 different DTs, where each of the three DTs is fully grown from two of the three attributes, again based on the same set of examples: {Income, Credit History}, {Credit History, Debt} and {Debt, Income}.

(c) What is the DT classifier's (part (a)) decision for a person who has 4K yearly income, a good credit history and a high amount of debt? Is your result different if we use 3 DTs in part (b) to make a decision?

Income	Credit History	Debt	Decision
0 – 5K	Bad	Low	Reject
0 – 5K	Good	Low	Approve
0 – 5K	Unknown	High	Reject
0 – 5K	Unknown	Low	Approve
0 – 5K	Unknown	Low	Approve
0 – 5K	Unknown	Low	Reject
5 – 10K	Bad	High	Reject
5 – 10K	Good	High	Approve
5 – 10K	Unknown	High	Approve
5 – 10K	Unknown	Low	Approve
Over 10K	Bad	Low	Reject
Over 10K	Good	Low	Approve

Table 2: Loan processing records

(Optional) How could the decisions (possibly different) given by the 3 DTs be collated together?

3. **Discretizing Continuous Features.**

Bob suspects the passion a student has for Machine Learning (represented by a real number) and his midterm grade contribute to the student's final grade for CS3244. He collected some data, as shown in table 3. Construct a decision tree based on the given graph 1 and table 3. You need to discretize both the MidtermGrade and PassionForML features. (Hint: Discretize MidtermGrade first.)

Midterm Grade	Passion For ML	Final Grade Is A
5	1.5	T
6	-4	T
7.5	0	F
8	-3	F
8	3.5	T
9	2	F
12	1.2	F
12	4	T
13	-2	T
14	-1	T

Table 3: Students' Information

He then plotted these information in a graph:

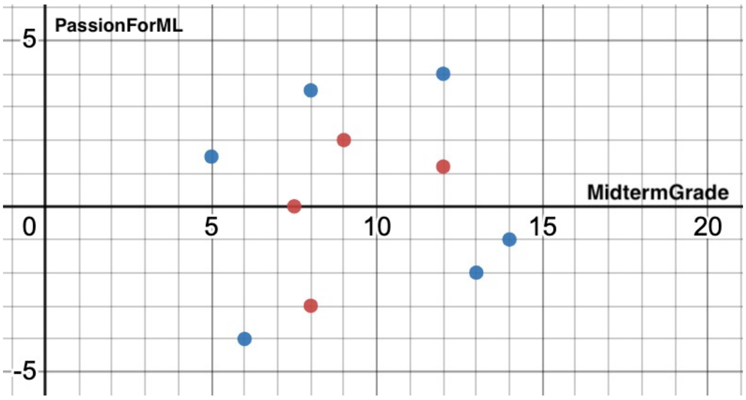

Figure 1: Blue dots denote A and red dots denote not an A

4. [1*] **Uniform Blending (UB).**

One of the simplest ensemble methods is UB. Given a set of hypothesis: $h_1, h_2, h_3, ..., h_T$, UB makes predictions simply by mixing the predictions given by $h_1, h_2, h_3, ..., h_T$ uniformly. Concretely, for binary classification, UB predicts by:

$$H(x) = \text{sign}(\sum_{t=1}^{T} h_t(x)), \tag{1}$$

and for regression, UB predicts by:

$$H(x) = \frac{1}{T}\sum_{t=1}^{T} h_t(x) \tag{2}$$

Taking regression as an example, show that the performance (measured by out-of-sample error) of UB is no worse than the average performance over $h_1, h_2, h_3, ..., h_T$; i.e.:

$$\frac{1}{T}\sum_{t=1}^{T} L_{\text{test}}(h_t(x)) \geq L_{\text{test}}(H(x)) \tag{3}$$

Assume we evaluate the testing error by mean square error.

Hint: Start by calculating the average error over $h_1, h_2, h_3, ..., h_T$ for one fixed data point x.

Proving Equation 3 can give us an intuition on why ensembling can help to reduce the out-of-sample error.

[1] * question is harder than other questions in this tutorial.