CS3244 Tutorial 3

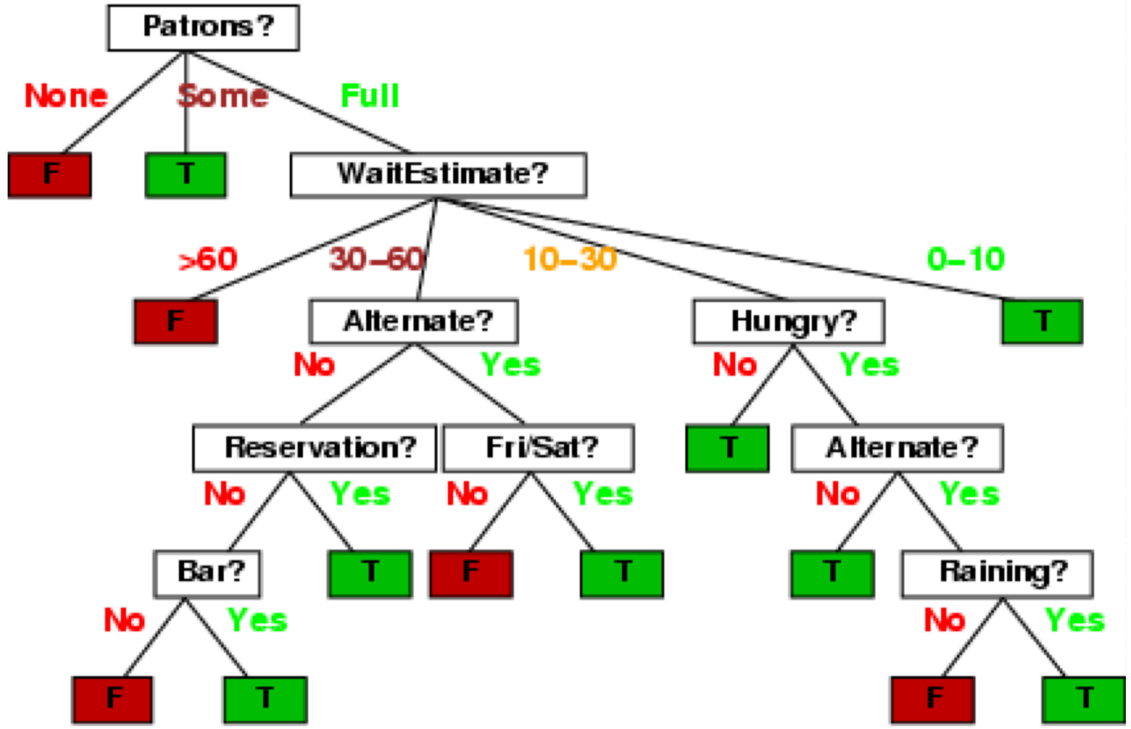
Wu Zhaoxuan
wu.zhaoxuan@u.nus.edu

2022.2.18

Decision Tree (DT) Learning

- Recall that in Concept Learning, we only learn concept with binary (0/1) outputs. The hypothesis space is often also restricted.

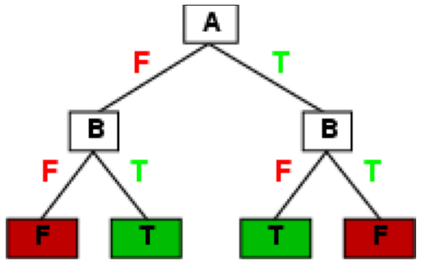
- DT is a new type of **representation for hypotheses**



Decision Tree (DT) Learning

- For Boolean functions (concepts), a truth table corresponds to a path to leaf.
- On page 8 of DT lecture slides, “A Boolean decision tree can be expressed in disjunctive normal form”
- E.g. A XOR $B \Leftrightarrow (\neg A \wedge B) \vee (A \wedge \neg B)$

A	B	A xor B
F	F	F
F	T	T
T	F	T
T	T	F



TM 3.1

*Give the **smallest** possible decision trees to represent the following boolean functions:*

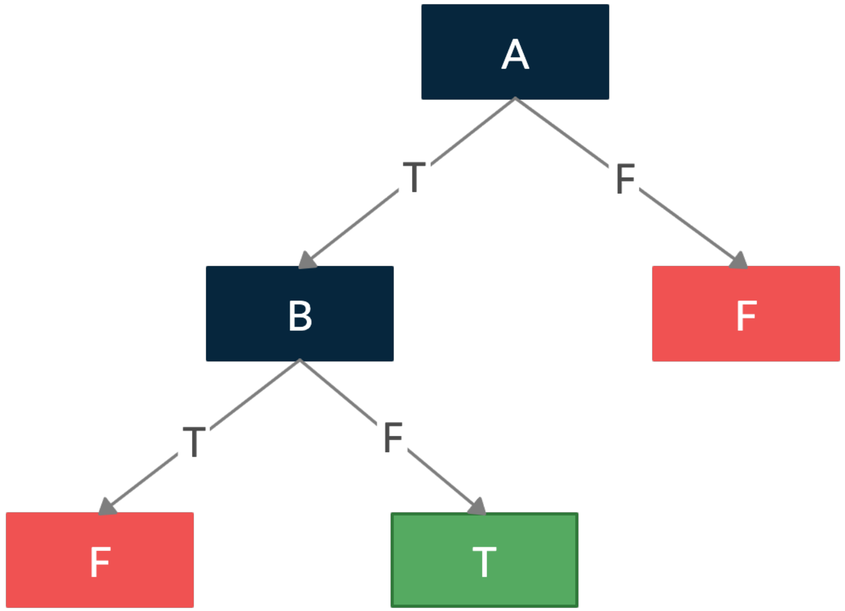
(a) $A \wedge \neg B$

(b) $A \vee (B \wedge C)$

(c) $(A \wedge B) \vee (C \wedge D)$

TM 3.1

(a) $A \wedge \neg B$



TM 3.1

(b) $A \vee (B \wedge C)$

TM 3.1

(b) $A \vee (B \wedge C)$

Distribution of conjunction over disjunction

$(P \wedge (Q \vee R)) \Leftrightarrow ((P \wedge Q) \vee (P \wedge R))$

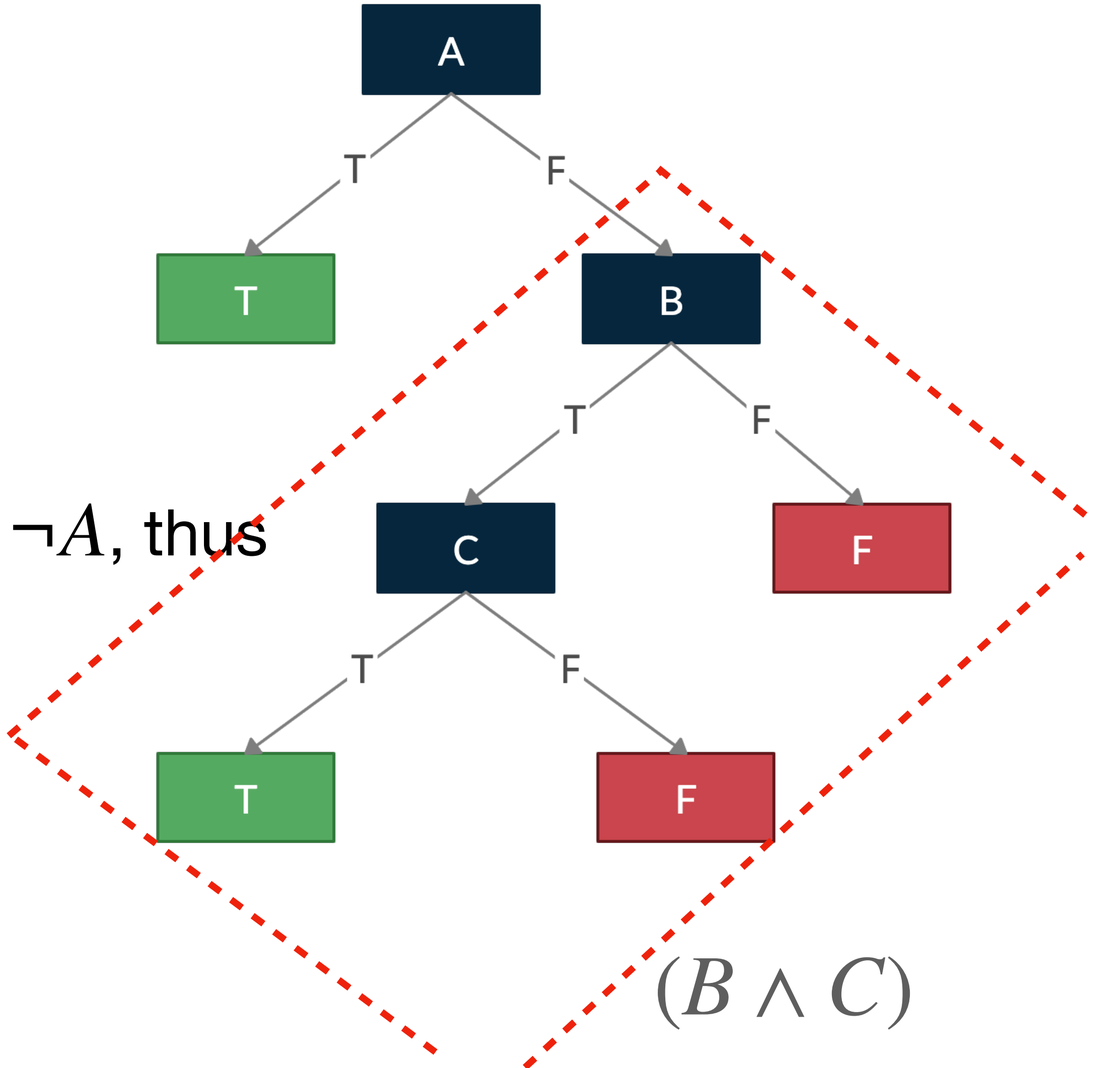
Distribution of disjunction over conjunction

$(P \vee (Q \wedge R)) \Leftrightarrow ((P \vee Q) \wedge (P \vee R))$

For this question, $(B \wedge C)$ part is missing A or $\neg A$, thus

$$A \vee (B \wedge C) \equiv \underbrace{(A \vee \neg A)}_{\equiv 1} \wedge (A \vee (B \wedge C))$$

$$\equiv A \vee (\neg A \wedge (B \wedge C))$$



TM 3.1

(c) $(A \wedge B) \vee (C \wedge D)$

TM 3.1

(c) $(A \wedge B) \vee (C \wedge D)$

In this question, the path $C \wedge D$ is missing a A, $\neg A$, B or $\neg B$.

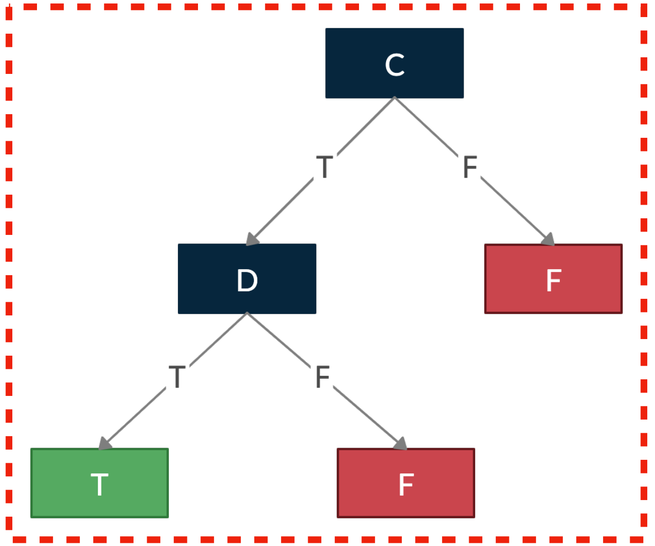
$$
\begin{aligned}
& (A \wedge B) \vee (C \wedge D) \\
\equiv\ & \underbrace{[(A \wedge B) \vee (\neg A \wedge B) \vee (\neg A \wedge \neg B) \vee (A \wedge \neg B)]}_{\equiv 1} \wedge [(A \wedge B) \vee (C \wedge D)] \\
\equiv\ & [(A \wedge B) \vee (\neg A \wedge (B \vee \neg B)) \vee (A \wedge \neg B)] \wedge [(A \wedge B) \vee (C \wedge D)] \\
\equiv\ & [(A \wedge B) \vee \neg A \vee (A \wedge \neg B)] \wedge [(A \wedge B) \vee (C \wedge D)] \\
\equiv\ & (A \wedge B) \vee \{[\neg A \vee (A \vee \neg B)] \wedge (C \wedge D)\} \\
\equiv\ & (A \wedge B) \vee (\neg A \wedge C \wedge D) \vee (A \wedge \neg B \wedge C \wedge D)
\end{aligned}
$$

TM 3.1

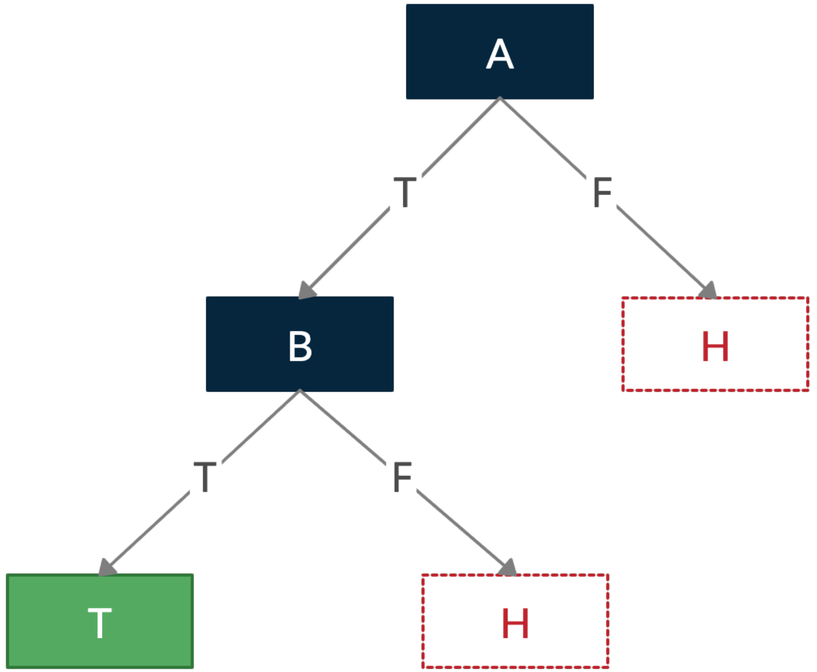
(c)

$$(A \wedge B) \vee (C \wedge D)$$

$$\equiv (A \wedge B) \vee (\neg A \wedge C \wedge D) \vee (A \wedge \neg B \wedge C \wedge D)$$



$$H \equiv (C \wedge D)$$



DT Learning Algorithm

- Aim. Find a small tree consistent with the training examples
- Idea. Greedily choose "most important" attribute as root of (sub)tree

function DECISION-TREE-LEARNING(*examples*, *attributes*, *parent_examples*) **returns** tree

if *examples* is empty **then return** PLURALITY-VALUE(*parent_examples*)
else if all *examples* have the same classification **then return** the classification
else if *attributes* is empty **then return** PLURALITY-VALUE(*examples*)
else
  $A \leftarrow \text{argmax}_{a \in attributes}$ IMPORTANCE(a, *examples*)
  tree $\leftarrow$ a new decision tree with root test A
  for each value v_k of A **do**
    $exs \leftarrow \{e : e \in examples \text{ and } e.A = v_k\}$
    subtree $\leftarrow$ DECISION-TREE-LEARNING(exs, $attributes - A$, *examples*)
    add a branch to *tree* with label $(A = v_k)$ and subtree *subtree*
  return *tree*

“Importance” Score

- Intuition: A good attribute splits the examples into subsets that are (ideally) “all +ve” or “all –ve”
- We can use the information theory concept of Entropy to measure uncertainty of a random variable
  $C \in \{c_1, \ldots, c_k\}, H(C) = -\sum_{i=1}^{k} P(c_i)\log P(c_i)$
- In binary case where $C \in \{p, n\}$, we have
  $H(C) = B(\frac{p}{p+n}) = -\frac{p}{p+n}\log\frac{p}{p+n} - \frac{n}{p+n}\log\frac{n}{p+n}$
- A chosen attribute A with d distinct values are split into subsets $E_1, \ldots, E_d$. Each subset E_i has p_i positive and n_i negative examples, we define $H(C|A) = \sum_{i=1}^{d}\frac{p_i+n_i}{p+n}B(\frac{p_i}{p_i+n_i})$, that is, ***a weighted sum of the entropy of each subset***.
- Lastly, our importance metric $Gain(C, A) = B(\frac{p}{p+n}) - H(C|A)$

BL 4

The loans department of a bank has the following past loan processing records, each of which contains an applicant's income, credit history, debt, and the final approval decision. These records can serve as training examples to build a decision tree for a loan advisory system.

(a) Construct a decision tree based on the above training examples.

Income	CreditHistory	Debt	Decision
$0-5$K	Bad	Low	Reject
$0-5$K	Good	Low	Approve
$0-5$K	Unknown	High	Reject
$0-5$K	Unknown	Low	Approve
$0-5$K	Unknown	Low	Approve
$0-5$K	Unknown	Low	Reject
$5-10$K	Bad	High	Reject
$5-10$K	Good	High	Approve
$5-10$K	Unknown	High	Approve
$5-10$K	Unknown	Low	Approve
Over 10K	Bad	Low	Reject
Over 10K	Good	Low	Approve

BL 4

Income	CreditHistory	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

- For income,

Income	+ve	-ve	Total
0-5K	3	3	6
5-10K	3	1	4
Over 10K	1	1	2

$$Gain(Decision, Income) = B(\frac{7}{12}) - H(Decision\,|\,Income)$$

$$H(Decision\,|\,Income) = \frac{6}{12}(B(\frac{3}{6})) + \frac{4}{12}(B(\frac{3}{4})) + \frac{2}{12}(B(\frac{1}{2}))$$

$$= \frac{6}{12}(-\frac{3}{6}\log\frac{3}{6} - \frac{3}{6}\log\frac{3}{6}) + \frac{4}{12}(-\frac{3}{4}\log\frac{3}{4} - \frac{1}{4}\log\frac{1}{4}) + \frac{2}{12}(-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2})$$

$$= \frac{1}{2}(-\log 3 + \log 6) + \frac{1}{3}(-\frac{3}{4}(\log 3 - \log 4) - \frac{1}{4}(\log 1 - \log 4)) + \frac{1}{6}(-\log 1 + \log 2)$$

$$= \frac{1}{2}(-1.585 + 2.585) + \frac{1}{3}(-\frac{3}{4}(1.585 - 2) - \frac{1}{4}(-2)) + \frac{1}{6}(1)$$

$$\approx 0.937$$

$$Gain(Decision, Income) \approx 0.98 - 0.937 = 0.043$$

BL 4

Income	CreditHistory	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

- For CreditHistory,

CreditHis	+ve	-ve	Total
Bad	0	3	3
Good	3	0	3
Unknown	4	2	6

$$Gain(Decision, CreditHistory) = B(\frac{7}{12}) - H(Decision\,|\,CreditHistory)$$

$$H(Decision\,|\,CreditHistory) = \frac{3}{12}(B(\frac{0}{3})) + \frac{3}{12}(B(\frac{3}{3})) + \frac{6}{12}(B(\frac{4}{6}))$$

$$= \frac{3}{12}(0) + \frac{3}{12}(0) + \frac{6}{12}(-\frac{4}{6}\log\frac{4}{6} - \frac{2}{6}\log\frac{2}{6})$$

$$= \frac{1}{2}(-\frac{4}{6}(\log 4 - \log 6) - \frac{2}{6}(\log 2 - \log 6))$$

$$= \frac{1}{2}(-\frac{4}{6}(2 - 2.585) - \frac{2}{6}(1 - 2.585))$$

$$\approx 0.459$$

$$Gain(Decision, CreditHistory) \approx 0.98 - 0.459 = 0.521$$

BL 4

Income	CreditHistory	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

- For Debt,

Debt	+ve	-ve	Total
Low	5	3	8
High	2	2	4

$$Gain(Decision, Debt) = B(\frac{7}{12}) - H(Decision \mid Debt)$$

$$H(Decision \mid Debt) = \frac{8}{12}(B(\frac{5}{8})) + \frac{4}{12}(B(\frac{2}{4}))$$
$$\approx 0.970$$

$$Gain(Decision, Debt) \approx 0.98 - 0.970 = 0.01$$

Since *CreditHistory* has the highest *Gain*, choose it as the root. Since all examples for *Bad* are -ve and all examples for *Good* are +ve, both nodes have no further subtree. For unknown, we have the following:

Income	Debt	Decision
$0-5K$	High	Reject
$0-5K$	Low	Approve
$0-5K$	Low	Approve
$0-5K$	Low	Reject
$5-10K$	High	Approve
$5-10K$	Low	Approve

BL 4

Income	Debt	Decision
$0-5K$	High	Reject
$0-5K$	Low	Approve
$0-5K$	Low	Approve
$0-5K$	Low	Reject
$5-10K$	High	Approve
$5-10K$	Low	Approve

- For income,

Income	+ve	-ve	Total
0-5K	2	2	4
5-10K	2	0	2

$$H(Decision\,|\,Income) = \frac{4}{6}(B(\frac{2}{4})) + \frac{2}{6}(B(\frac{2}{2})) = \frac{4}{6} \approx 0.667$$

$$Gain(Decision, Income) = B(\frac{4}{6}) - \frac{4}{6} \approx 0.918 - 0.667 = 0.251$$

For Debt,

Debt	+ve	-ve	Total
Low	3	1	4
High	1	1	2

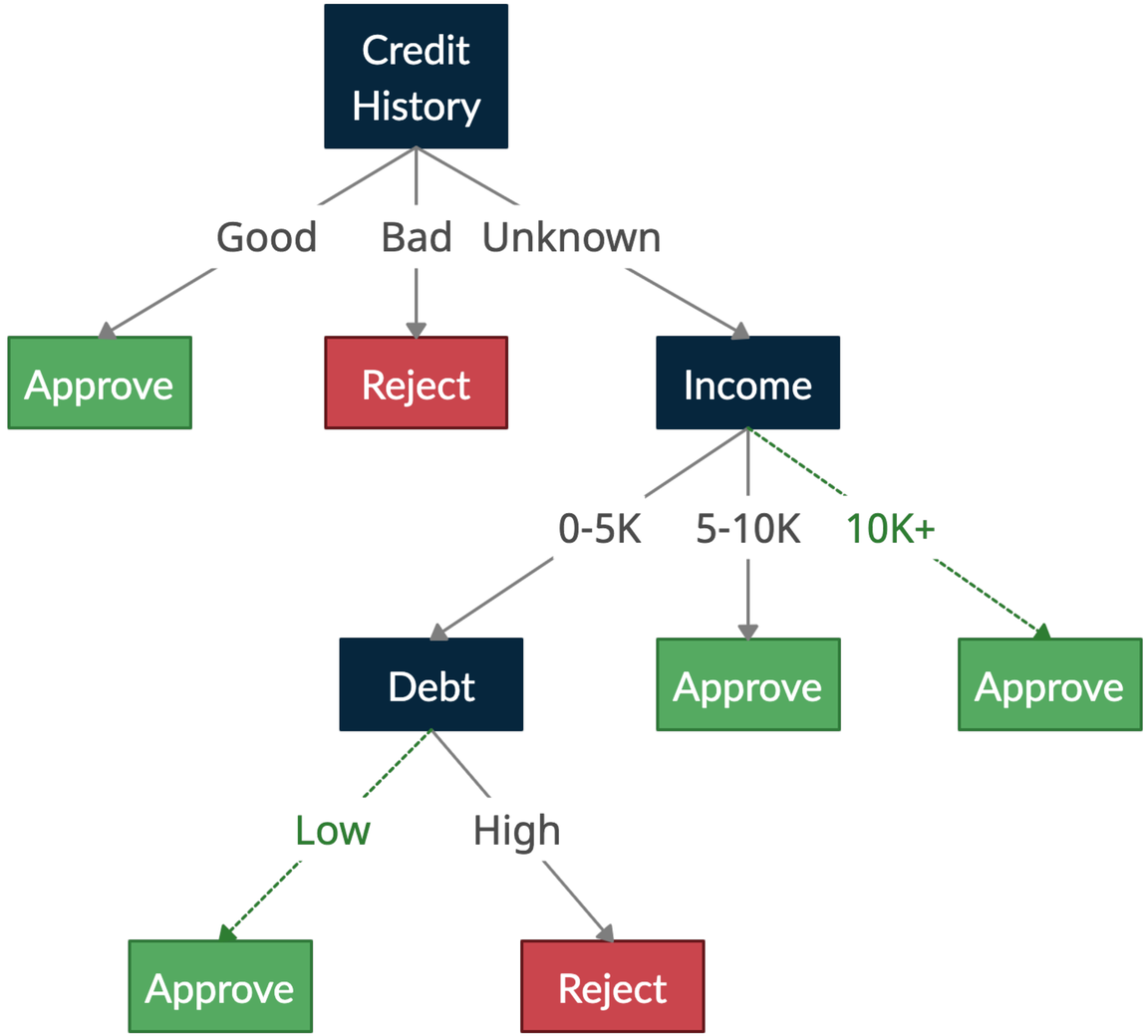
$$H(Decision\,|\,Debt) = \frac{4}{6}(B(\frac{3}{4})) + \frac{2}{6}(B(\frac{1}{2})) \approx 0.874$$

$$Gain(Decision, Debt) \approx 0.918 - 0.874 = 0.044$$

Then, $Income$ is chosen as the root of the subtree under $CreditHistory = Unknown$.

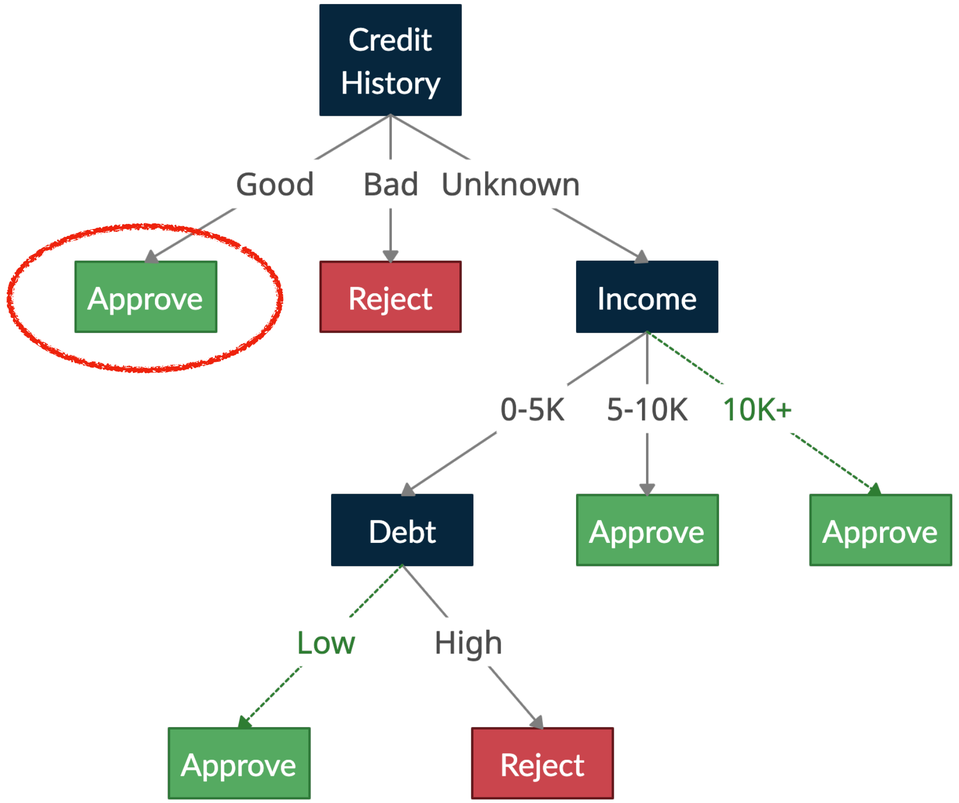
BL 4

- Finally…
- The dashed arrows are obtained through ***Plurality-Value*** Function (or majority vote)
  - For $Income = 10K+$, we have ***no examples left***.
  - Similarly, when we are at $Debt = Low$, ***no more attribute left***.
- Why do we have such a node where there seems to be ambiguity?
  - We are missing some attribute from the data that will allow us to differentiate at this last node
  - There is some non-determinism in the underlying function that we are trying to approximate with this decision tree
  - Noise/error in data



BL 4

(b) What is decision tree classifier's decision for a person who has 4K yearly income, a good credit history, and a high amount of debt?



BL 5

Given the following training examples about exotic dishes, we want to predict whether or not a dish is appealing based on the input attributes 'Temperature', 'Taste', and 'Size'.

(a) What is the information gain $Gain(Appealing, Taste)$ *associated with choosing the 'Taste' attribute at the root of the decision tree?*

ID	Temperature	Taste	Size	Appealing
1	Hot	Salty	Small	No
2	Cold	Sweet	Large	No
3	Cold	Sweet	Large	No
4	Cold	Sour	Small	Yes
5	Hot	Sour	Small	Yes
6	Hot	Salty	Large	No
7	Hot	Sour	Large	Yes
8	Cold	Sweet	Small	Yes
9	Cold	Sweet	Small	Yes
10	Hot	Salty	Large	No

BL 5

Taste	+ve	-ve	Total
Salty	0	3	3
Sweet	2	2	4
Sour	3	0	3

ID	Temperature	Taste	Size	Appealing
1	Hot	Salty	Small	No
2	Cold	Sweet	Large	No
3	Cold	Sweet	Large	No
4	Cold	Sour	Small	Yes
5	Hot	Sour	Small	Yes
6	Hot	Salty	Large	No
7	Hot	Sour	Large	Yes
8	Cold	Sweet	Small	Yes
9	Cold	Sweet	Small	Yes
10	Hot	Salty	Large	No

$$H(Appealing \mid Taste)$$

$$= \frac{3}{10}B(\frac{0}{3}) + \frac{4}{10}B(\frac{2}{4}) + \frac{3}{10}B(\frac{3}{3})$$

$$= 0 + \frac{2}{5}(1) + 0$$

$$= \frac{2}{5}$$

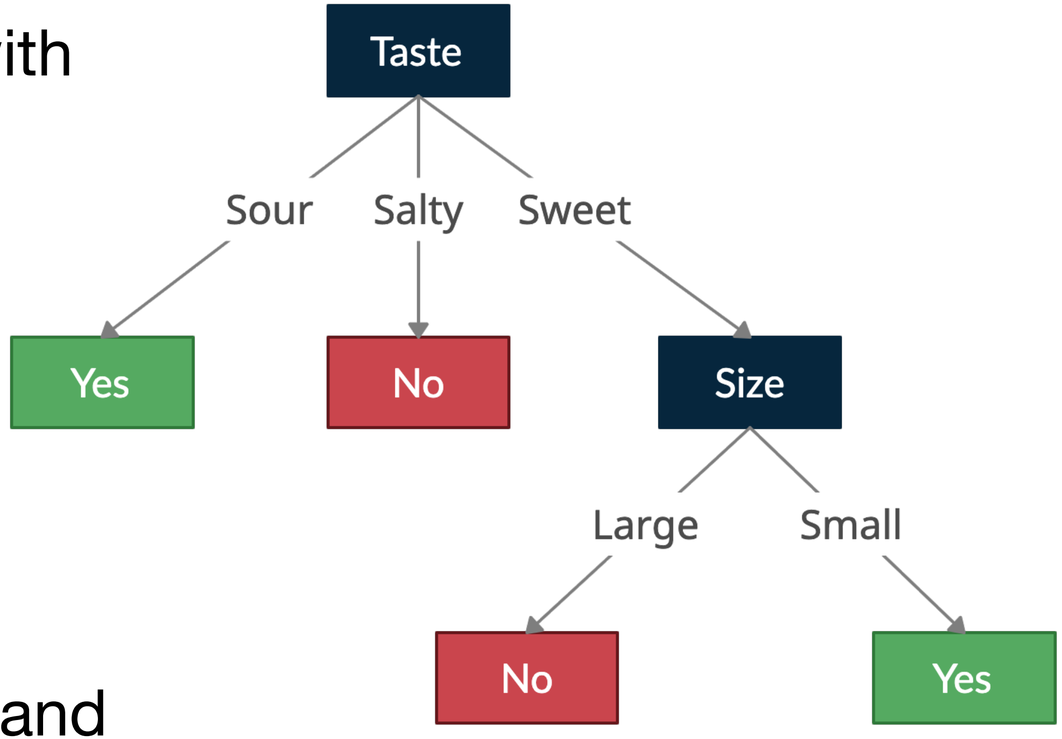
$$Gain(Appealing, Taste) = B(\frac{5}{10}) - H(Appealing \mid Taste) = 1 - \frac{2}{5} = \frac{3}{5}$$

BL 5

(b) Draw a decision tree with ‘Taste’ as the root.

After placing ‘Taste’ as the root, we are left with

ID	Temp.	Taste	Size	Appealing
2	Cold	Sweet	Large	No
3	Cold	Sweet	Large	No
8	Cold	Sweet	Small	Yes
9	Cold	Sweet	Small	Yes



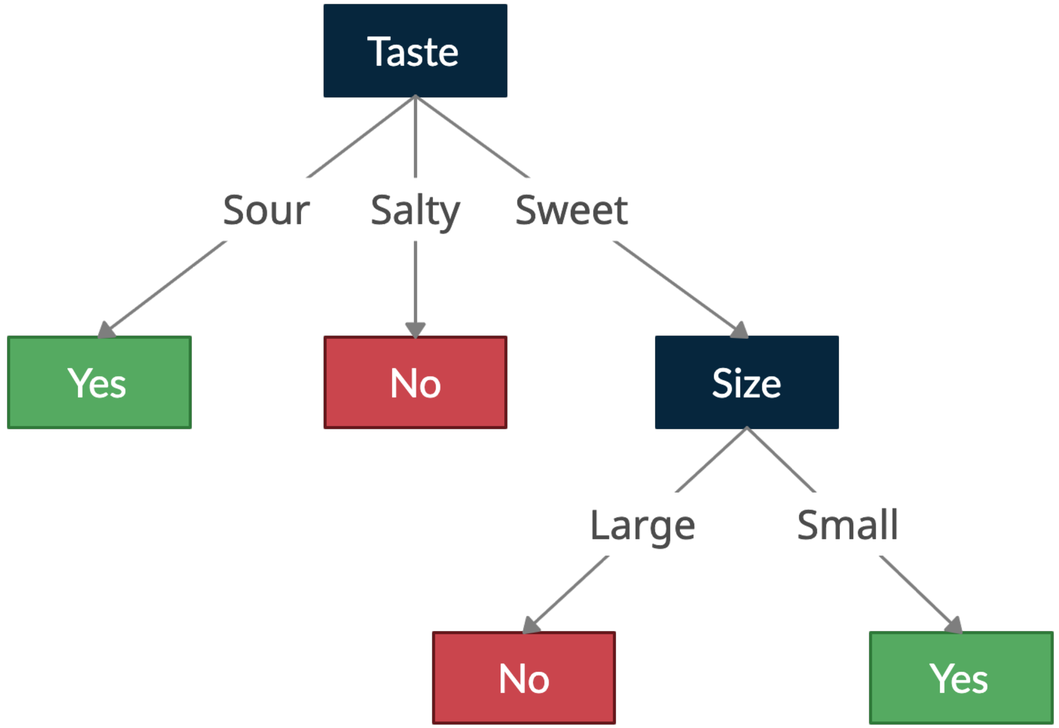
Note: $Gain(Appealing, Temperature) = 0$ and

$Gain(Appleaing, Size)$ is maximum.

BL 5

(c) Use the decision tree to predict the class value for the record given by

ID	Temperature	Taste	Size
11	Hot	Salty	Small
12	Cold	Sweet	Large



For ID=11, predict $Appealing = No$.

For ID=12, predict $Appealing = No$

Thank you!

- Any questions?