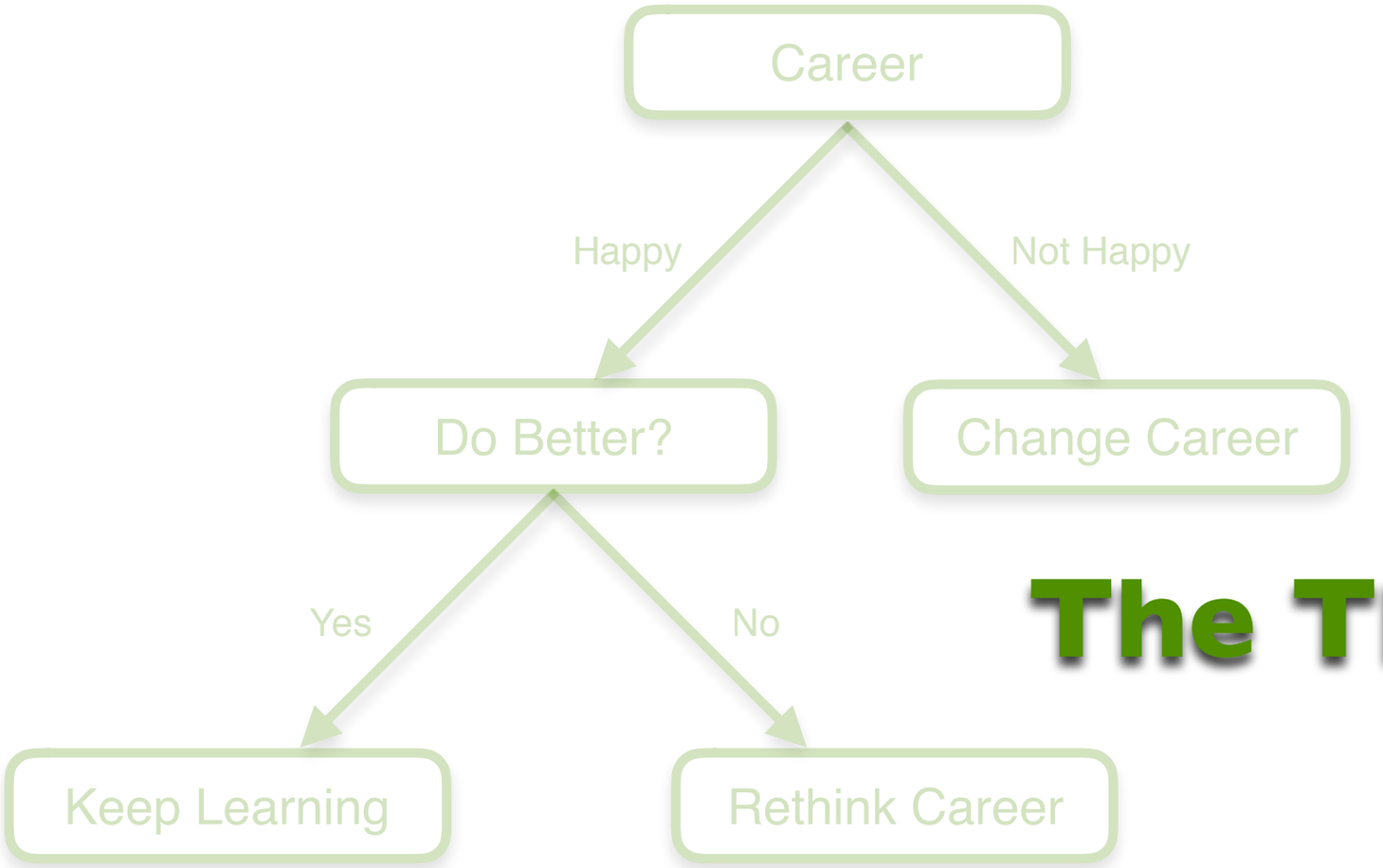
Decision Tree



The Thinking Tree

Instructor

Dr. Sourish Das
Post Doc, Duke University
Associate Professor,
Chennai Mathematical Institute

Information sourced from

“Prof Madhavan Mukund, Dean of Science and Deputy Director of CMI helped me to develop the following slides.” - Dr Sourish Das

Guideline

“Please refer to the corresponding video lecture to understand the slides better.” - Dr Sourish Das

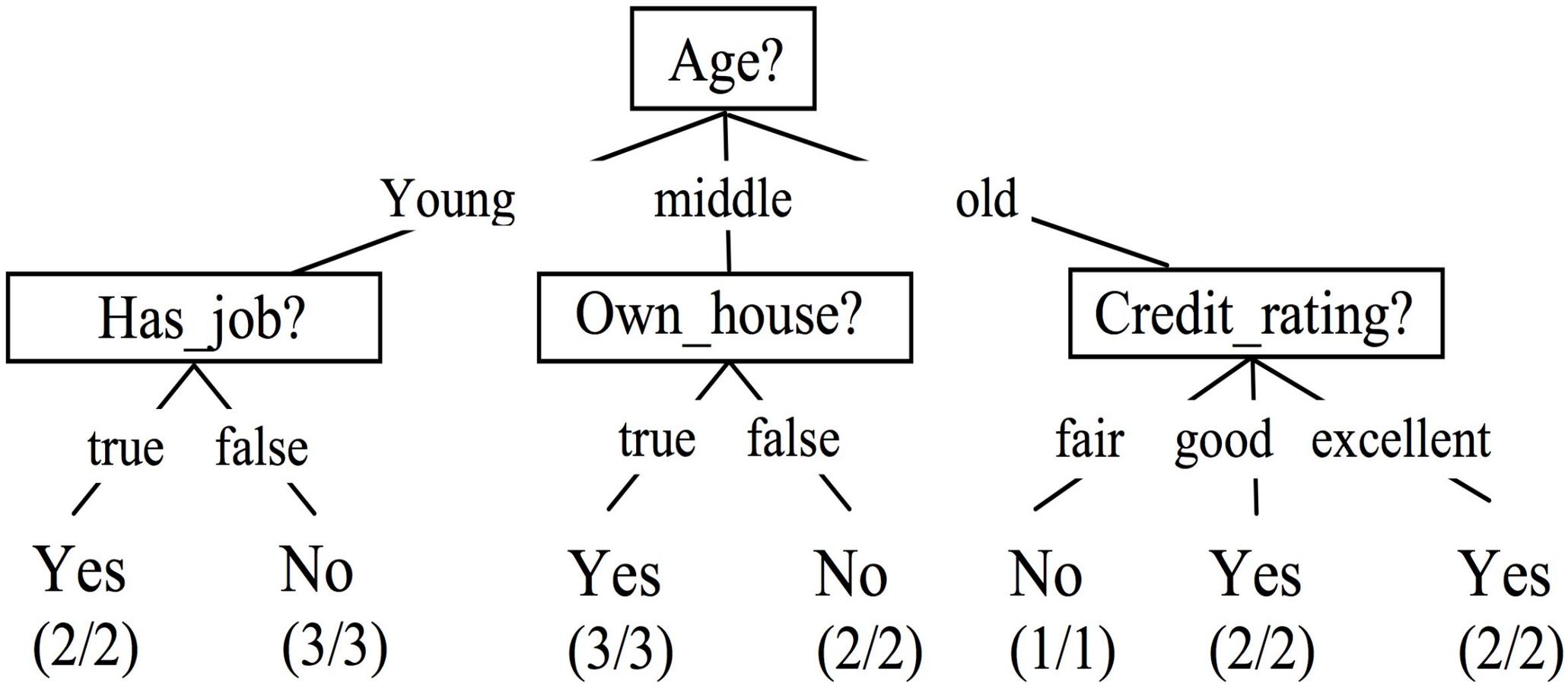
Decision trees

- Examine features one at a time
- Branch according to value of features
- (Adaptively) choose next features to examine

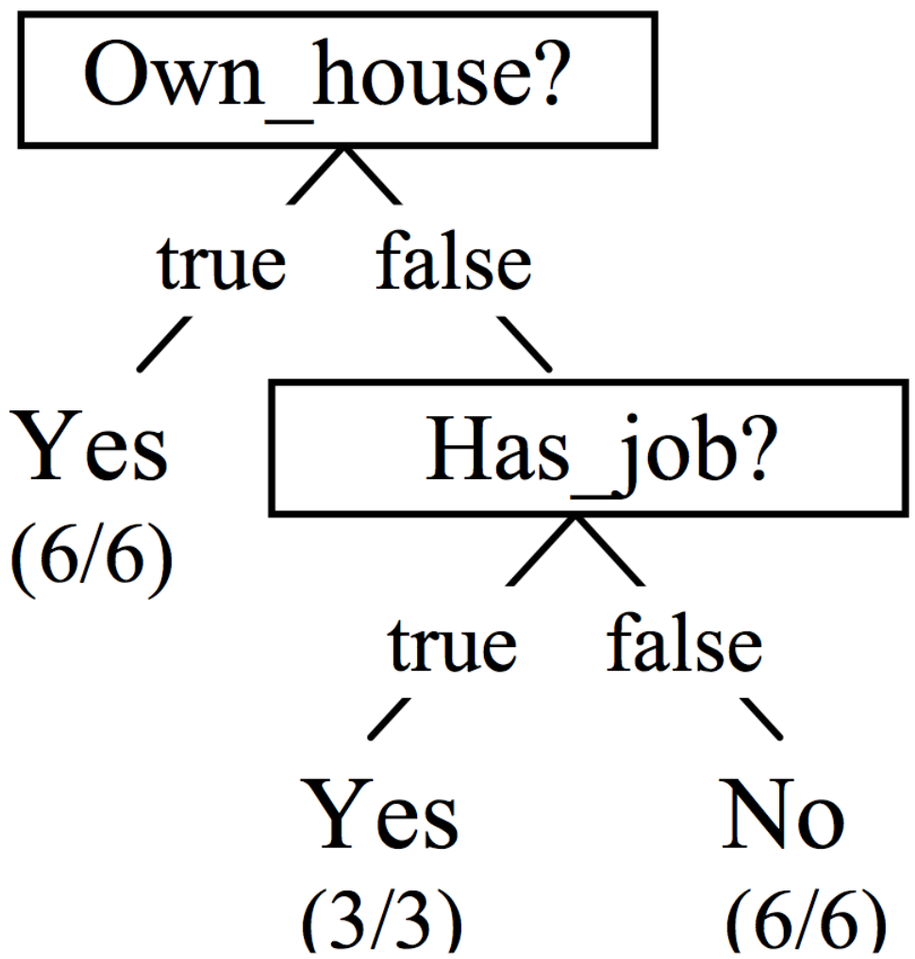
Example: Loan application data set

ID	Age	Has_job	Own_house	Credit_rating	Class
1	young	false	false	fair	**No**
2	young	false	false	good	**No**
3	young	true	false	good	**Yes**
4	young	true	true	fair	**Yes**
5	young	false	false	fair	**No**
6	middle	false	false	fair	**No**
7	middle	false	false	good	**No**
8	middle	true	true	good	**Yes**
9	middle	false	true	excellent	**Yes**
10	middle	false	true	excellent	**Yes**
11	old	false	true	excellent	**Yes**
12	old	false	true	good	**Yes**
13	old	true	false	good	**Yes**
14	old	true	false	excellent	**Yes**
15	old	false	false	fair	**No**

Example: Decision tree



Example: A smaller decision tree



Constructing decision trees

Smaller trees are better

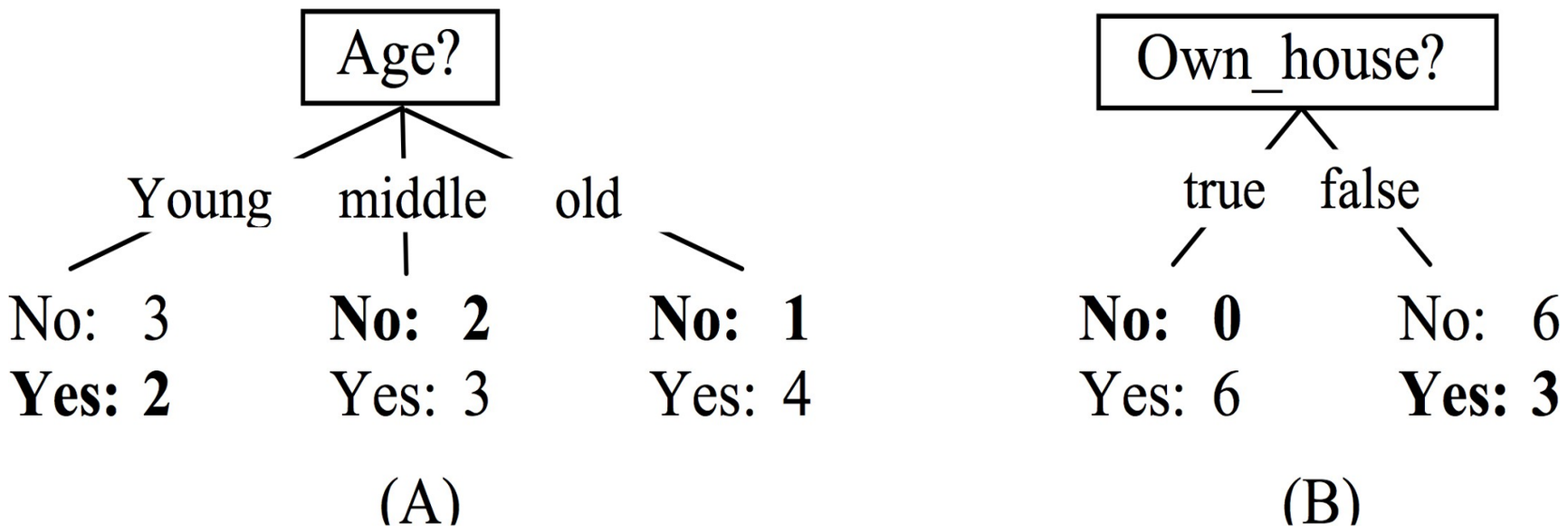
- Tend to be more accurate
- Easier to understand
    - Explaining the classification can be important
    - Disease diagnosis from symptoms and test results

Finding the best tree that fits the data is expensive

- NP-complete
- Need heuristics that can incrementally construct a good tree

Constructing decision trees: impurity

A reasonable heuristic is to minimize impurity



(B) is a better initial choice than (A) — resulting subgroups are more homogeneous, more pure

Can we quantify this notion?

Constructing decision trees: entropy

Information theory

- Data set D, classes C = {c_1, c_2, . . . , c_k}
- Pr(c_j) = fraction of samples in D classified as c_j

$$entropy(D) = -\sum_{i=1}^{k} \Pr(c_i) log_2 \Pr(c_i)$$

$$\sum_{i=1}^{k} \Pr(c_i) = 1$$

Constructing decision trees: entropy

Entropy measures disorder or impurity of data

Entropy is maximized when all entries in C are equiprobable

Suppose $C = \{y, n\}$

$\Pr(y) = 0, \Pr(n) = 1 \Rightarrow$

$\text{entropy}(D) = -(0 \log 0 + 1 \log 1) = 0$

Note We define $0 \log 0 = 0$

$\Pr(y) = 0.2, \Pr(n) = 0.8 \Rightarrow \text{entropy}(D) = -(0.2 \log 0.2 + 0.8 \log 0.8)$

$= 0.722$

$\Pr(y) = \Pr(n) = 0.5 \Rightarrow \text{entropy}(D) = -(0.5 \log 0.5 + 0.5 \log 0.5) = 1$

Information theoretically, entropy describes the minimum number of bits required to transmit values

Constructing decision trees: information gain

At each step, choose the attribute that maximally reduces the entropy — maximizes information gain

- Current data is D, attribute A has Avalues
- Choosing A as next node in the tree partitions D as $\{D_1, D_2, \ldots, D_A\}$

$$Define\ entropy_A(D) = \sum_{1}^{l} \frac{|D|_j}{|D|} entropy\left(|D|_j\right)$$

- Define gain(D, A) = entropy(D) – $entropy_A$(D)
- Choose A_j such that gain(D, A_j) is maximized

Constructing decision trees: information gain

- Attributes with unique values (Aadhar ID, passport number) produce pure partitions of size 1
- Zero entropy, but not useful as a classifier!
- Normalize the information gain by the entropy of the partition

$$gainration(D, A) = \frac{gain(D,\ A)}{-\sum_1^l \frac{|D|_j}{|D|} log \frac{|D|_j}{|D|}}$$

- Maximize normalized information gain—information gain ratio

Thank you.

Email us your questions, that you may have on this topic, at learn@spotle.ai

We will be happy to help you.