Decision Trees

Important concepts

A tree has three types of nodes :

- A **root node** that has no incoming edges and zero or more outgoing edges.
- **Internal nodes**, each of which has exactly one incoming edge and two or more outgoing edges.
- **Leaf** or **terminal nodes**, each of which has exactly one incoming edge and no outgoing edges.

How to build a Decision tree

Greedy strategy

Hunt's Algorithm

In Hunt's algorithm, a decision tree is grown in a recursive fashion by partitioning the training records into successively purer subsets. Let D_t be the set of training records that are associated with node t and $y = \{y_1, y_2, \ldots, y_c\}$ be the class labels. The following is a recursive definition of Hunt's algorithm.

Step 1: If all the records in D_t belong to the same class y_t, then t is a leaf node labeled as y_t.

Step 2: If D_t contains records that belong to more than one class, an attribute test condition is selected to partition the records into smaller subsets. A child node is created for each outcome of the test condition and the records in D_t are distributed to the children based on the outcomes. The algorithm is then recursively applied to each child node.

Design

1. How should the training records be split?

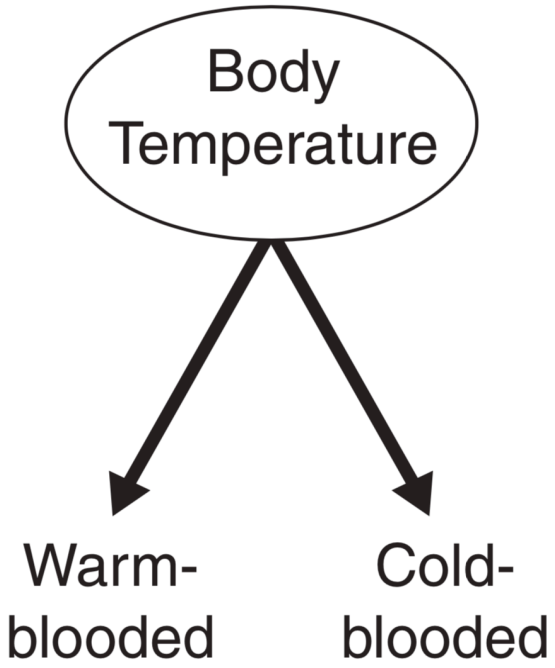
- select an attribute test condition to divide the records into smaller subsets

2. How should the splitting procedure stop?

- A possible strategy is to continue expanding a node until either all the records belong to the same class or all the records have identical attribute values.

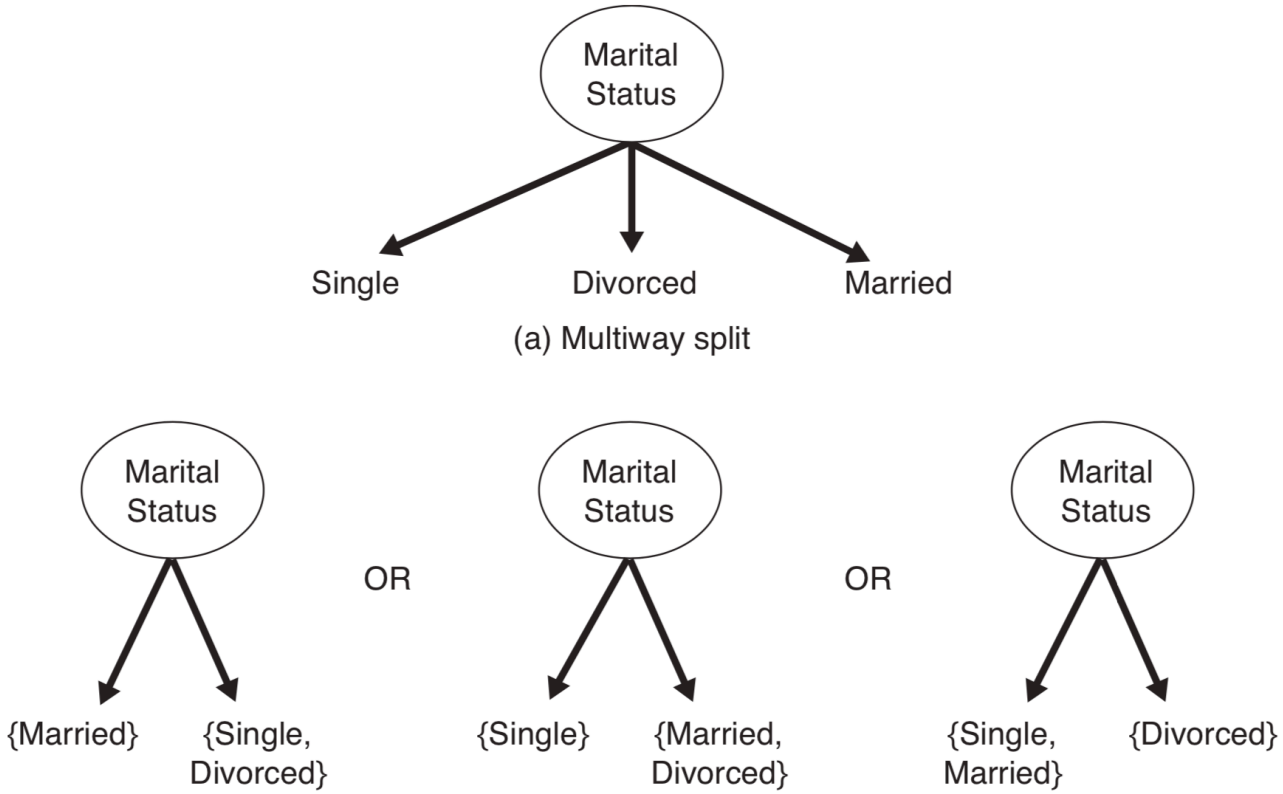
Expressing Attribute Test Conditions

Binary attributes



Expressing Attribute Test Conditions

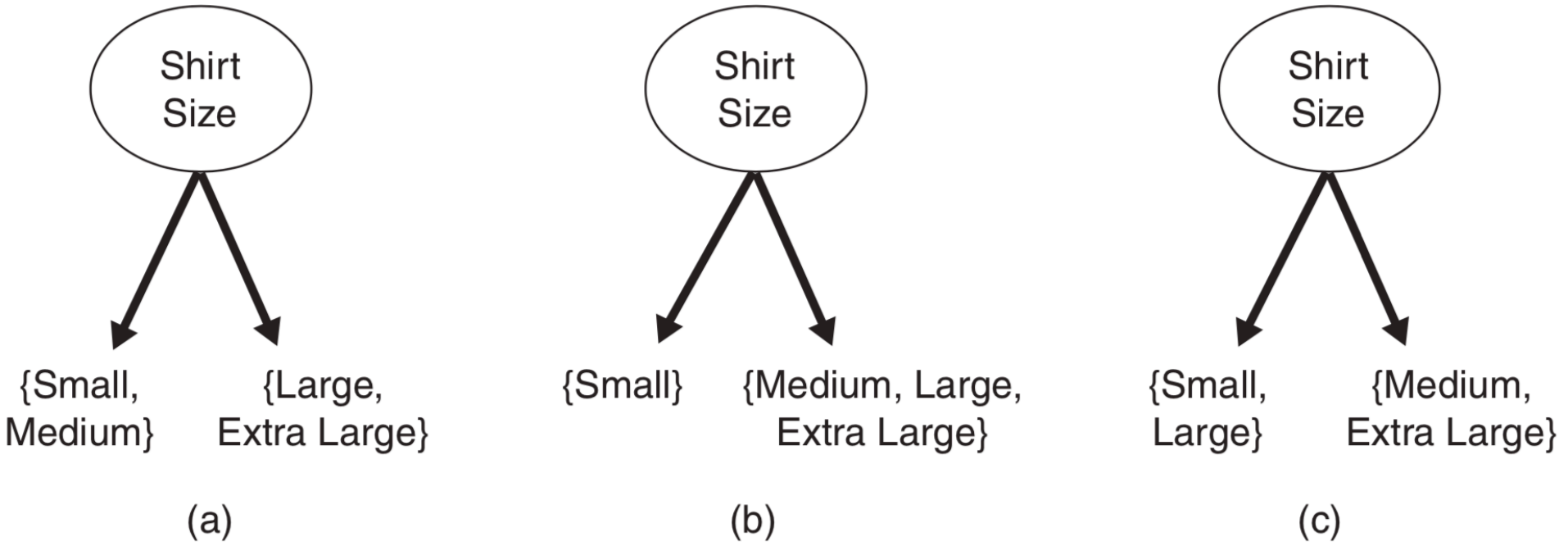
Nominal attributes



Expressing Attribute Test Conditions

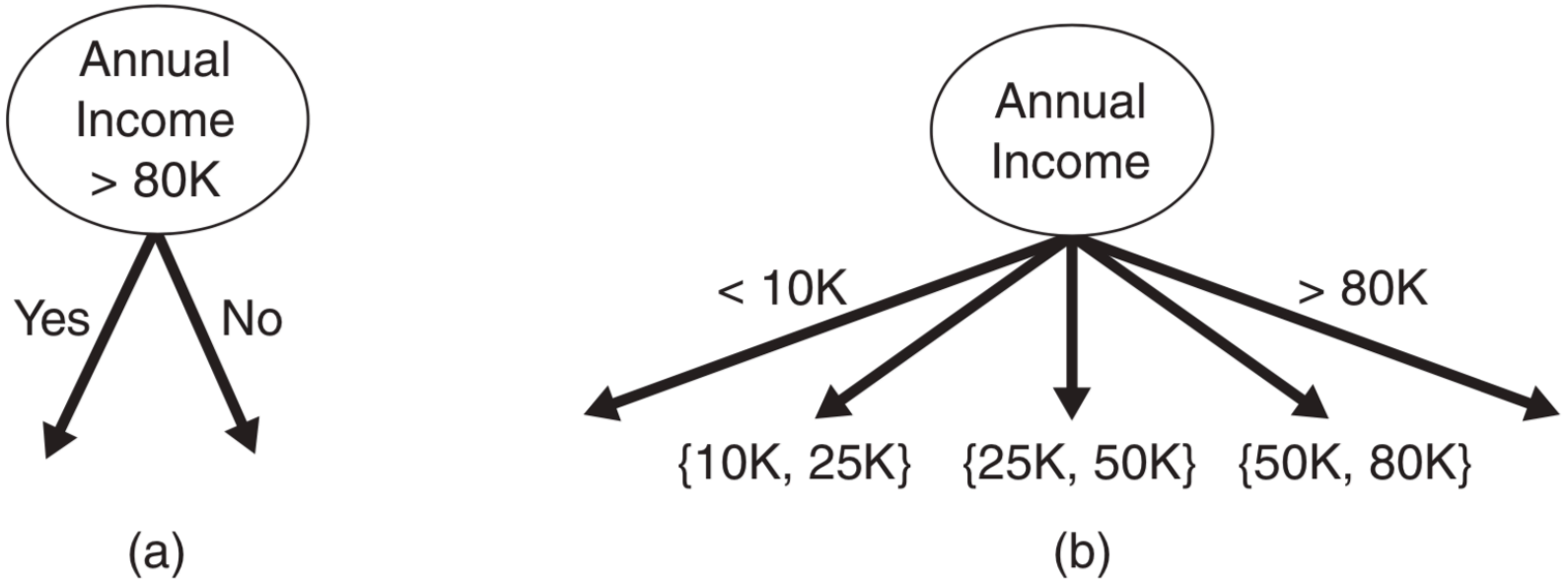
Ordinal attributes

Ordinal attribute values can be grouped as long as the grouping does not violate the order property of the attribute values.



Expressing Attribute Test Conditions

Continuous attributes



For continuous attributes, the test condition can be expressed as a comparison test ($A < v$) or ($A \geq v$) with binary outcomes, or a range query with outcomes of the form $v_i \leq A < v_{i+1}$, for $i = 1,\ldots,k$.

Measures for Selecting the best split

These measures are defined in terms of the class distribution of the records before and after splitting.

Let p(ilt) denote the fraction of records belonging to class i at a given node t.

We sometimes omit the reference to node t and express the fraction as p_i.

In a two-class problem, the class distribution at any node can be written as (p_0, p_1), where $p_1 = 1 - p_0$.

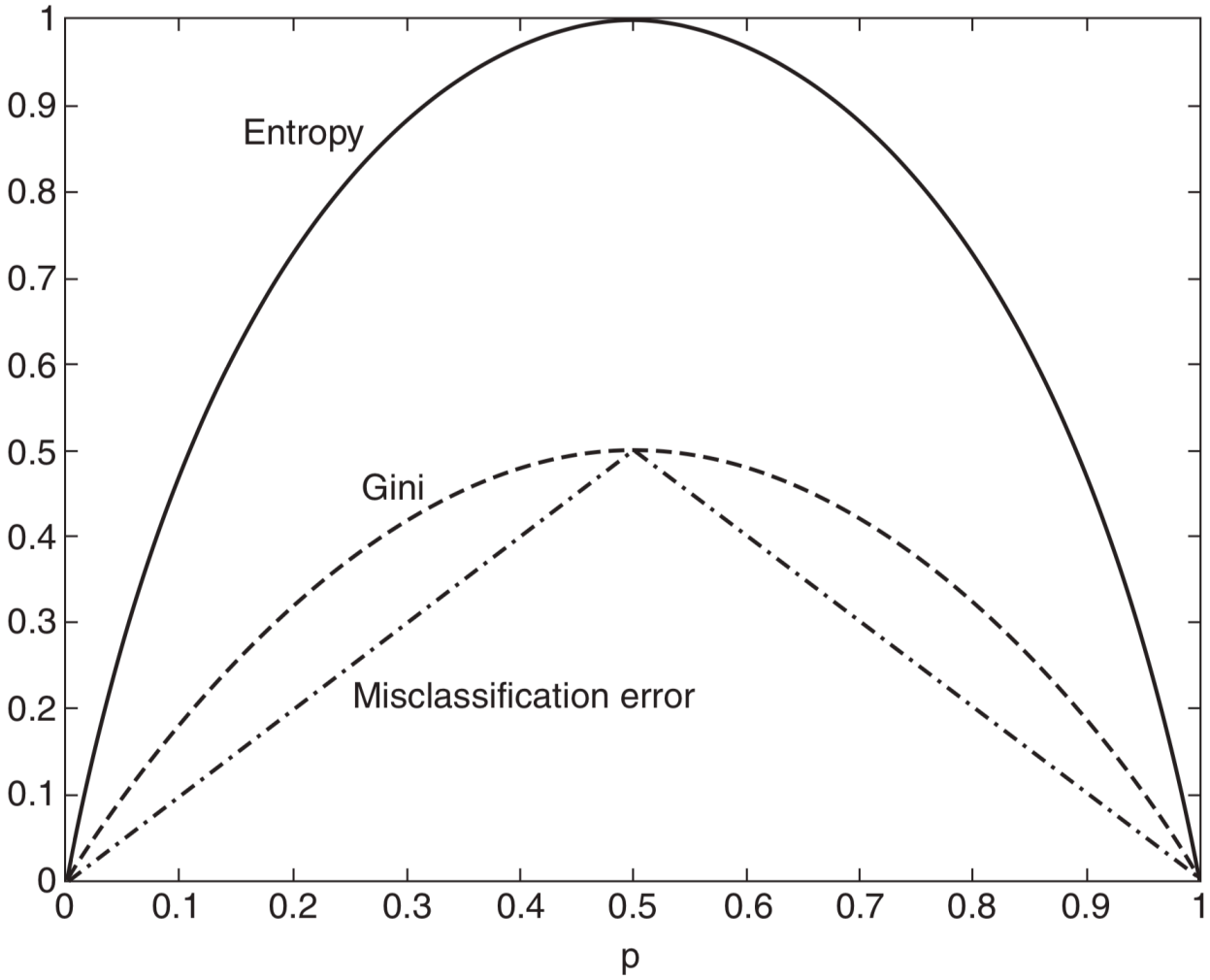
$$\text{Entropy}(t) = -\sum_{i=0}^{c-1} p(i|t) \log_2 p(i|t),$$

$$\text{Gini}(t) = 1 - \sum_{i=0}^{c-1} [p(i|t)]^2,$$

$$\text{Classification error}(t) = 1 - \max_i [p(i|t)],$$

where c is the number of classes and $0 \log_2 0 = 0$ in entropy calculations.

Comparison among the impurity measures for binary classification problems



Examples

entropy = -(0/6) log_2(0/6) - (6/6)log_2(1)=0

gini = 1 - ((0/6)^2 +(6/6)^2) = 0

Error =1 - max[0/6,6/6] = 0

Node N_1	Count
Class=0	0
Class=1	6

entropy =0.65
gini = 0.278
error = 0.167

Node N_2	Count
Class=0	1
Class=1	5

entropy =1
gini = 0.5
error = 0.5

Node N_3	Count
Class=0	3
Class=1	3

Gain

To determine how well a test condition performs, we need to compare the degree of impurity of the parent node (before splitting) with the degree of impurity of the child nodes (after splitting).

$$\Delta = I(\text{parent}) - \sum_{j=1}^{k} \frac{N(v_j)}{N} I(v_j),$$

where I(·) is the impurity measure of a given node, N is the total number of records at the parent node, k is the number of attribute values, and N(v_j) is the number of records associated with the child node, v_j.

Information gain, I(·) = entropy

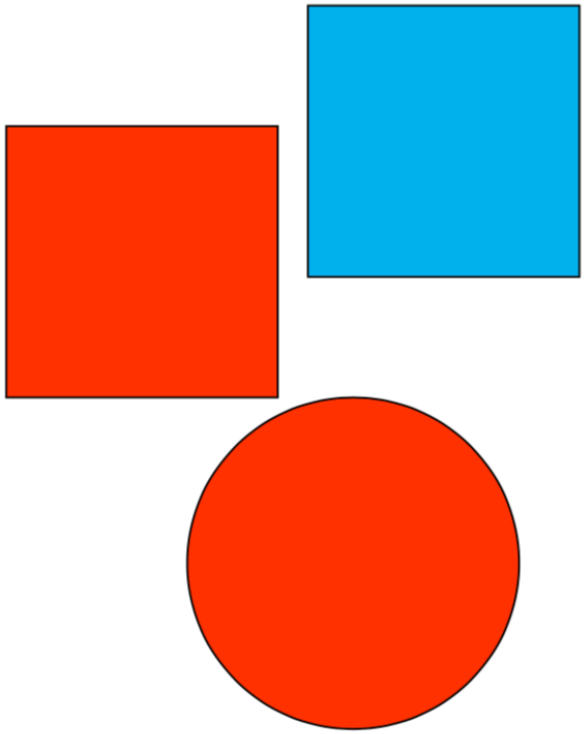
Entropy = -3/6 log_2(3/6) - 3/6 log_2(3/6) = 1

Example

- Features: color, shape, size
- What's the best question at root?

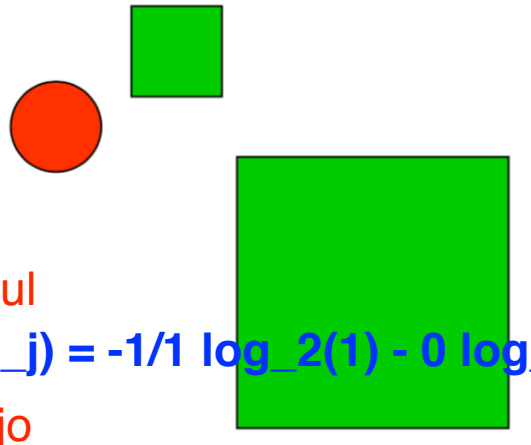
Entropy
Gini
Information Gain

+



-

Total de elementos = 6
clases = 2



azul
I(v_j) = -1/1 log_2(1) - 0 log_2(0) = 0

rojo
I(v_j) = -2/3 log_2(2/3)-1/3 log_2(1/3) =0.918295

verde

I(v_j) = -2/2 log_2(1)-0/2 log_2(0/2) = 0

Que pasa si usan Gini?, obtiene la misma conclusi'on?

Delta = 1 - [1/6 * 0 + 3/6 *0.918295 + 2/6 * 0] = 0.59

Revisar los ejemplos de la clase pasada
y hacer las cuentas correspondientes a los primeros dos splits.

Cual es el valor de Delta?

Gain ratio

Impurity measures such as entropy and Gini index tend to favor attributes that have a large number of distinct values.

There are two strategies for overcoming this problem.

1. restrict the test conditions to binary splits only (CART)
2. Gain ratio

$$\text{Gain ratio} = \frac{\Delta_{\text{info}}}{\text{Split Info}}.$$

$$\text{Here, Split Info} = -\sum_{i=1}^{k} P(v_i) \log_2 P(v_i) \text{ and } k \text{ is the total number of splits.}$$