Example 1: Information Gain (ID3)

- **Example 1**. Given training set D of class-labeled tuples randomly selected from *AllElectronics* customer database. Class label attribute, *buys_computer*, has two distinct values, namely, {*yes*, *no*}. Therefore, there are **two** classes (i.e., $m = 2$). Let class C_1 correspond to *yes* and class C_2 correspond to *no*. There are **nine** tuples of class *yes* and **five** tuples of class *no*. A (root) node N is created for tuples in D. To find the **splitting criterion** for these tuples, we must compute the **information gain** of **each** attribute.

Example 1: Information Gain (ID3)

- We first use Eq. (8.1) to compute the **expected information** (known as **entropy**) needed to classify a tuple in *D*:

$$Info(D) = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.940 \text{ (bits)}$$

Example 1: Information Gain (ID3)

• Next, we need to compute **expected information** requirement for each attribute.

• Let's start with attribute *age*. We need to look at the distribution of *yes* and *no* tuples for each category of *age*.

- For *age* category "youth," there are **two** *yes* tuples and **three** *no* tuples.

- For category "middle aged," there are **four** *yes* tuples and **zero** *no* tuples.

- For category "senior," there are **three** *yes* tuples and **two** *no* tuples.

Example 1: Information Gain (ID3)

TID	*age*	*income*	*student*	*credit_rating*	*Class: buy_computer*
9	youth	low	yes	fair	yes
11	youth	medium	yes	excellent	yes
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
8	youth	medium	no	fair	no
7	middle_aged	low	yes	excellent	yes
13	middle_aged	high	yes	fair	yes
3	middle_aged	high	no	fair	yes
12	middle_aged	medium	no	excellent	yes
5	senior	low	yes	fair	yes
10	senior	medium	yes	fair	yes
4	senior	medium	no	fair	yes
6	senior	low	yes	excellent	no
14	senior	medium	no	excellent	no

Example 1: Information Gain (ID3)

- Using Eq. (8.2), the **expected information** needed to classify a tuple in *D* if the tuples are partitioned according to *age* is

$$Info_A(D) = \frac{5}{14} \times \left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right) + \frac{4}{14} \times \left(-\frac{4}{4}\log_2\frac{4}{4}\right) + \frac{5}{14} \times \left(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}\right)$$

$$Info_A(D) = 0.694 \text{ (bits)}$$

Example 1: Information Gain (ID3)

- $Info_A(D) = 0.694 < Info(D) = 0.940$

$\rightarrow$ data homogeneity (purity) is improved.

- The **information gain** from such a partitioning is

$$Gain(A) = Info(D) - Info_{age}(D)$$
$$Gain(A) = 0.940 - 0.694 = 0.246 \text{ (bits)}$$

Example 1: Information Gain (ID3)

- Similarly, we can compute $Gain(income) = 0.029$ bits, $Gain(student) = 0.151$ bits, and $Gain(credit\_rating) = 0.048$ bits [Exercise].
- Because *age* has the **highest information gain** among the attributes, **it is selected as the splitting attribute**.

Example 1: Information Gain (ID3)

•

age?

youth

income	*student*	*credit_rating*	*class*
high	no	fair	no
high	no	excellent	no
medium	no	fair	no
low	yes	fair	yes
medium	yes	excellent	yes

middle_aged

income	*student*	*credit_rating*	*class*
high	no	fair	yes
low	yes	excellent	yes
medium	no	excellent	yes
high	yes	fair	yes

senior

income	*student*	*credit_rating*	*class*
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

Gain Ratio (C4.5)

- Information gain measure is biased toward tests with many outcomes. That is, it prefers to select attributes having a large number of values.
- C4.5, a successor of ID3, uses an extension to information gain known as *gain ratio*, which attempts to overcome this bias.