Iterative Dichotomiser 3 (ID3) Decision Tree Algorithm

- Uses entropy as the measure of impurity

ID3 uses this information gain measure to select among the candidate attributes at each step while growing the tree.

S is a collection of 14 examples of a Boolean concept, including 9 positive and 5 negative examples [9+, 5].

Then the entropy of S relative to this Boolean classification is:

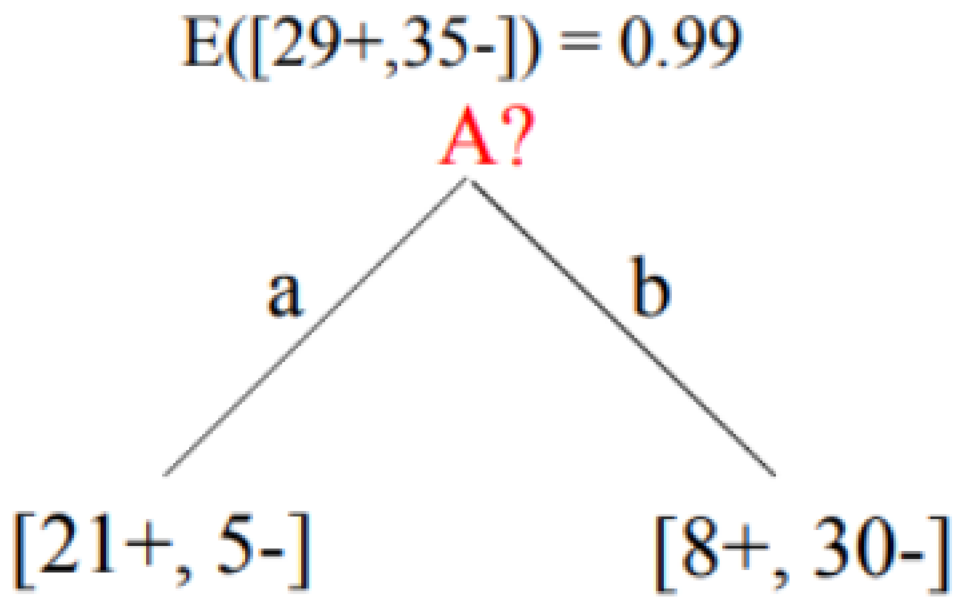
$$\begin{aligned} Entropy([9+, 5-]) &= -(9/14)\log_2(9/14) - (5/14)\log_2(5/14) \\ &= 0.940 \end{aligned}$$

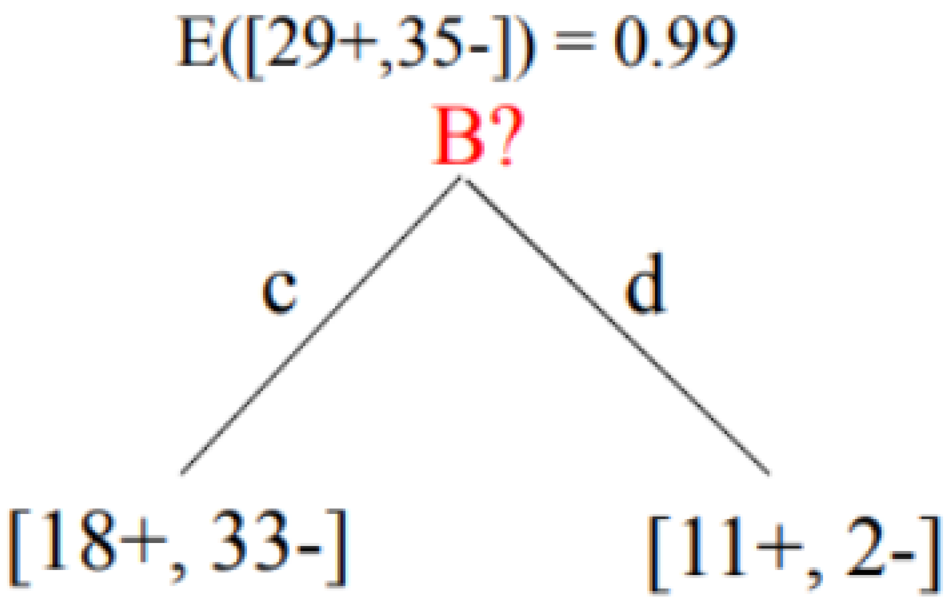
ID3($Examples$, $Target\_attribute$, $Attributes$)

Examples are the training examples. Target_attribute is the attribute whose value is to be predicted by the tree. Attributes is a list of other attributes that may be tested by the learned decision tree. Returns a decision tree that correctly classifies the given Examples.

- Create a *Root* node for the tree
- If all $Examples$ are positive, Return the single-node tree $Root$, with label = +
- If all $Examples$ are negative, Return the single-node tree $Root$, with label = −
- If $Attributes$ is empty, Return the single-node tree $Root$, with label = most common value of *Target_attribute* in *Examples*
- Otherwise Begin
    - $A \leftarrow$ the attribute from *Attributes* that best* classifies $Examples$
    - The decision attribute for $Root \leftarrow A$
    - For each possible value, v_i, of A,
        - Add a new tree branch below $Root$, corresponding to the test $A = v_i$
        - Let $Examples_{v_i}$ be the subset of $Examples$ that have value v_i for A
        - If $Examples_{v_i}$ is empty
            - Then below this new branch add a leaf node with label = most common value of $Target\_attribute$ in $Examples$
            - Else below this new branch add the subtree
              ID3($Examples_{v_i}$, $Target\_attribute$, $Attributes - \{A\}$))
- End
- Return *Root*

Which attribute is the best classifier?

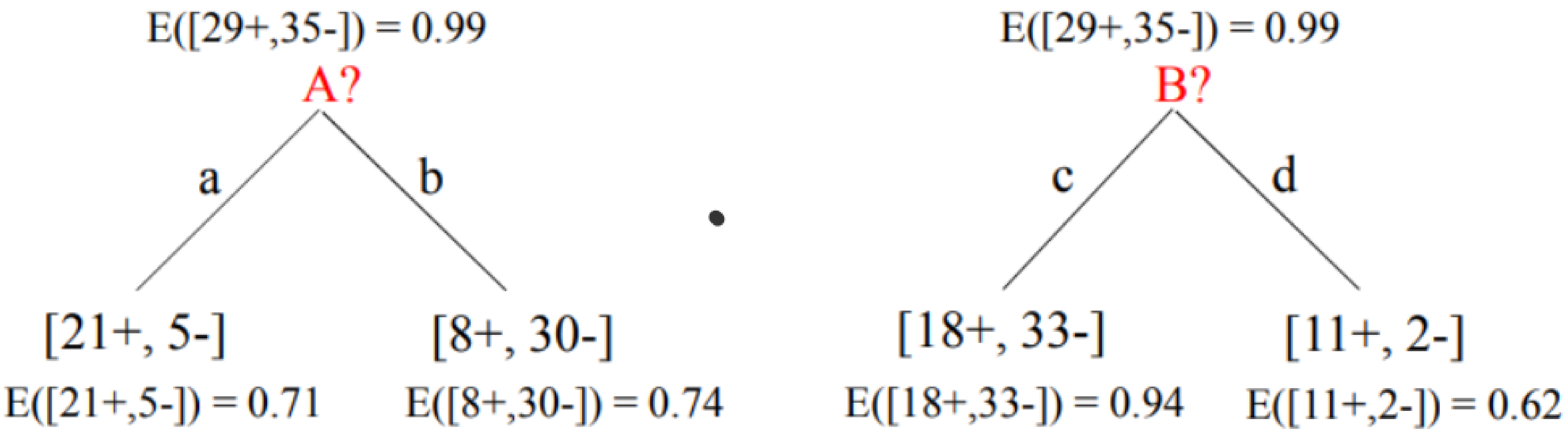




Gain(S,A) = Entropy(S)
-26/64*Entropy([21+,5-])
-38/64*Entropy([8+,30-])

Gain(S,B) = Entropy(S)
-51/64*Entropy([18+,33-])
-13/64*Entropy([11+,2-])

Which attribute is the best classifier?



Gain(S,A) = Entropy(S)
-26/64*Entropy([21+,5-])
-38/64*Entropy([8+,30-])
= **0.27**

Gain(S,B) = Entropy(S)
-51/64*Entropy([18+,33-])
-13/64*Entropy([11+,2-])
= **0.12**

A provides greater information gain than B.
A is a better classifier than B.

or, no. of bits saved

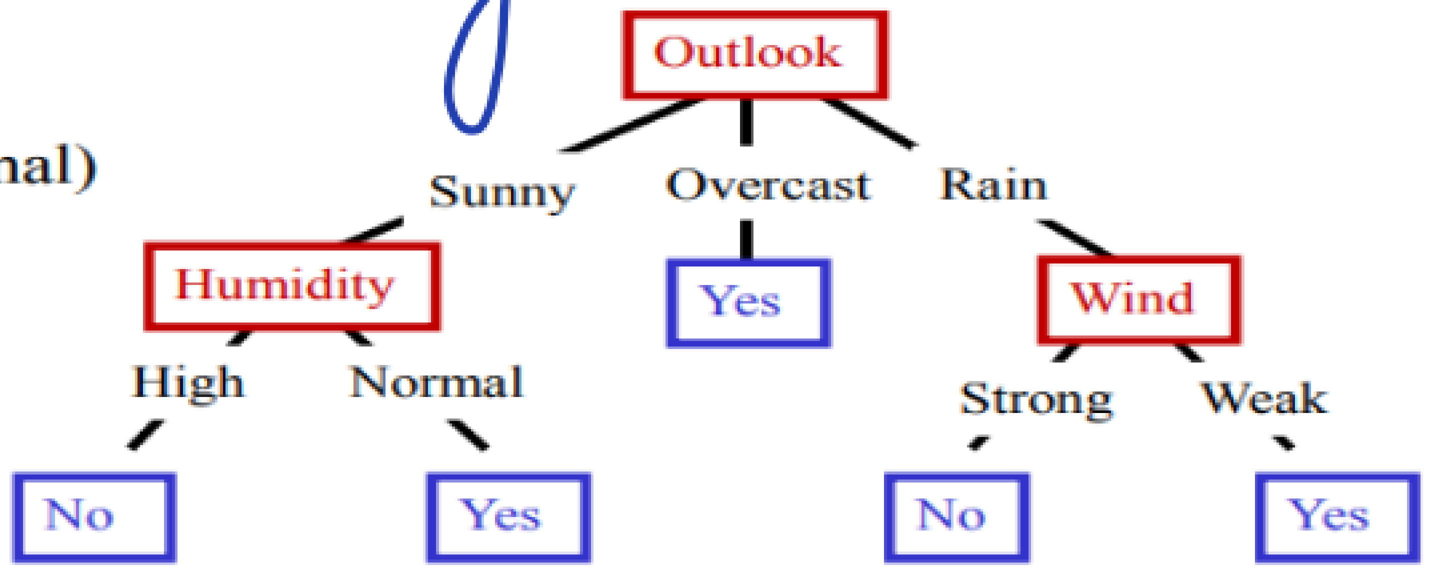
Gain: Expected reduction in entropy caused by knowing the value of attribute A

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

DT represents a disjunction of conjunction of constraints on the attribute values of instances.

Hypothesis

$$
\begin{aligned}
&(Outlook = Sunny \wedge Humidity = Normal) \\
\vee\ &(Outlook = Overcast) \\
\vee\ &(Outlook = Rain \wedge Wind = Weak)
\end{aligned}
$$



When no info about any of the attribute is present

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

14 cases

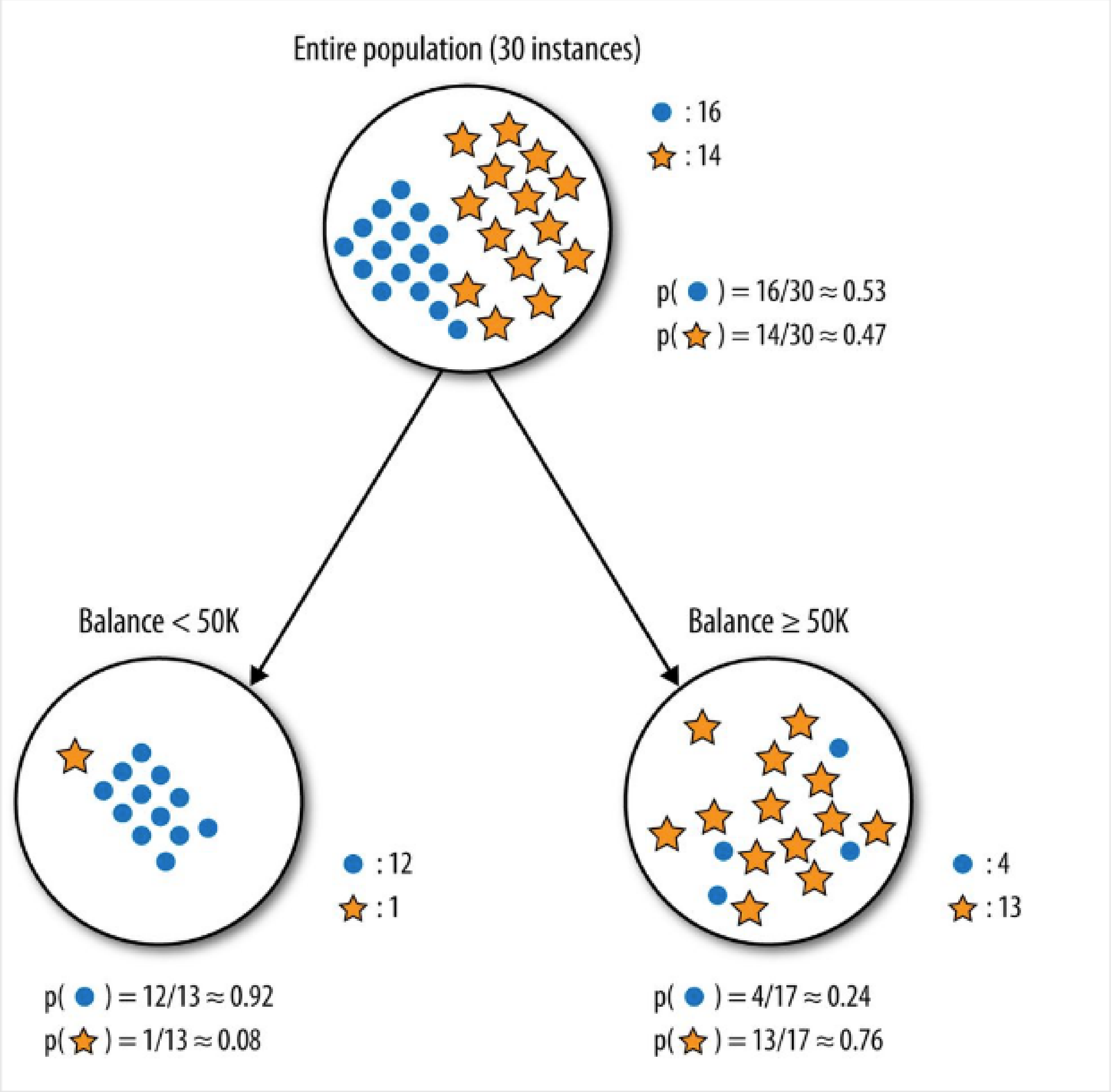
9 positive cases

- Step 1: *Calculate* ***entropy*** for all cases:

$N_{Pos} = 9$ $N_{Neg} = 5$ $N_{Tot} = 14$

entropy → $H(S) = -(9/14)*\log_2(9/14) - (5/14)*\log_2(5/14) = 0.940$

Yes : 9
No : 5



Idea :

Choose that attribute for splitting which results in purer nodes

1 2 3 4

Yes : 9
No : 5

$E(P) = 0.940$

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Outlook

Sunny — Yes : 2, No : 3

overcast — Yes : 4, No : 0

Rain — Yes : 3, No : 2

ses

9 positive cases

$E_1 = 0.9$ $E_2 = 0$ $E_3 =$

$$\text{Gain} = E(P) - E(C)$$

$$E(C) = \frac{5}{14} E_1 + \frac{4}{14} E_2 + \frac{5}{14} E_3$$

Step 2: Loop over all attributes, calculate gain:

– Attribute = *Outlook*

- Loop over values of *Outlook*

  Outlook = Sunny

  $N_{Pos} = 2$ $N_{Neg} = 3$ $N_{Tot} = 5$

  $H(Sunny) = -(2/5)*\log_2(2/5) - (3/5)*\log_2(3/5) = 0.971$

  Outlook = Overcast

  $N_{Pos} = 4$ $N_{Neg} = 0$ $N_{Tot} = 4$

  $H(Sunny) = -(4/4)*\log_2 4/4) - (0/4)*\log_2(0/4) = 0.00$

  Outlook = Rain

  $N_{Pos} = 3$ $N_{Neg} = 2$ $N_{Tot} = 5$

  $H(Sunny) = -(3/5)*\log_2(3/5) - (2/5)*\log_2(2/5) = 0.971$

- Calculate **Information Gain** for attribute Outlook

$$Gain(S, Outlook) = H(S) - N_{Sunny}/N_{Tot}*H(Sunny) - N_{Over}/N_{Tot}*H(Overcast) - N_{Rain}/N_{Tot}*H(Rainy)$$

$$Gain(S, Outlook) = 9.40 - (5/14)*0.971 - (4/14)*0 - (5/14)*0.971$$

$$Gain(S, Outlook) = 0.246$$

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

– Attribute = *Temperature*

- (Repeat process looping over {Hot, Mild, Cool})

  Gain(S,*Temperature*) = 0.029

– Attribute = *Humidity*

- (Repeat process looping over {High, Normal})

  Gain(S,*Humidity*) = ~~0.029~~ 0.151

– Attribute = *Wind*

- (Repeat process looping over {Weak, Strong})

  Gain(S,*Wind*) = 0.048

Find attribute with greatest information gain:

Gain(S,*Outlook*) = 0.246, Gain(S,*Temperature*) = 0.029

Gain(S,*Humidity*) = ~~0.029~~, 0.151 Gain(S,*Wind*) = 0.048

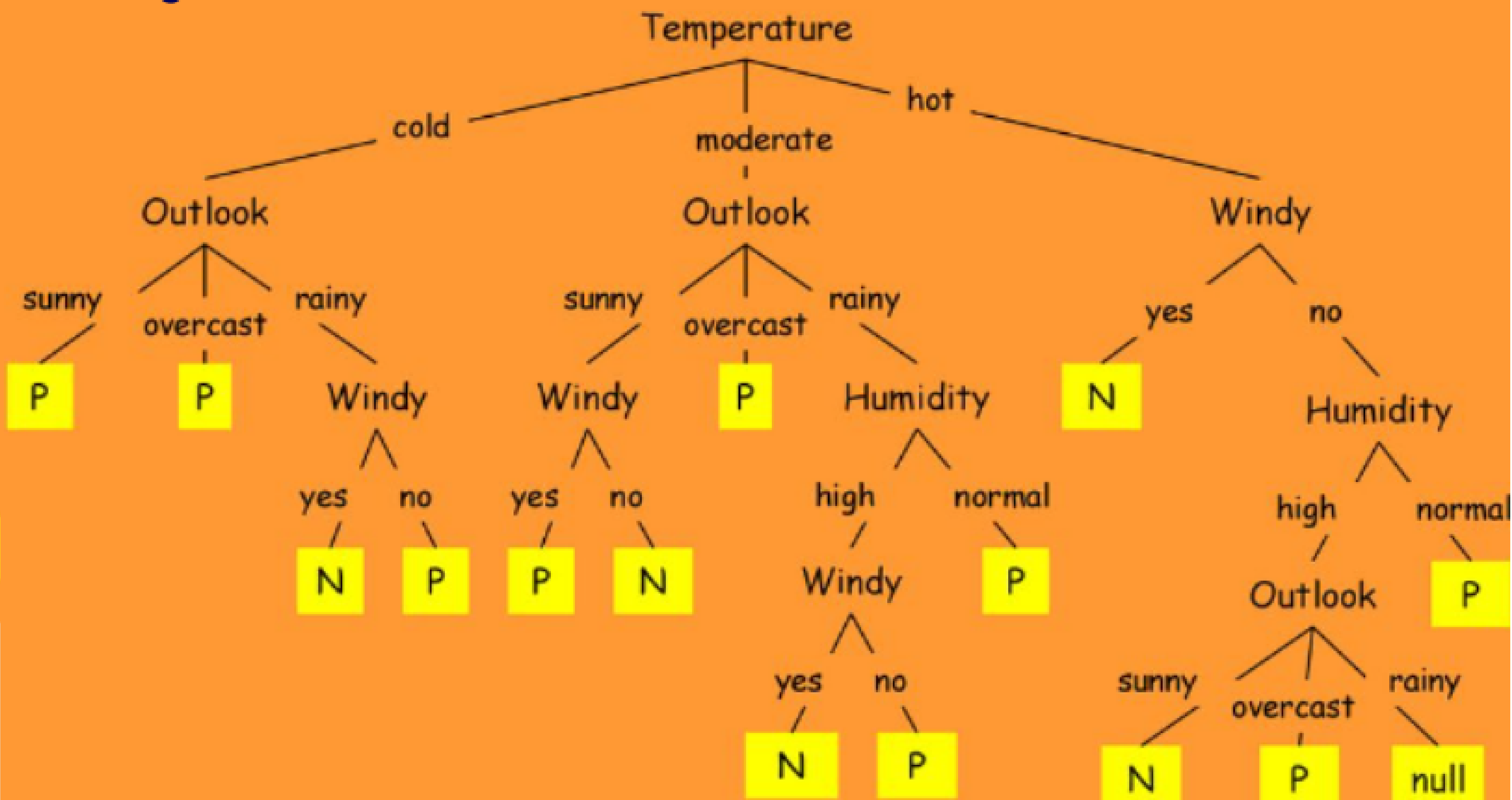
∴ Outlook is root node of tree

Outlook vs Temperature as candidates at "root" node

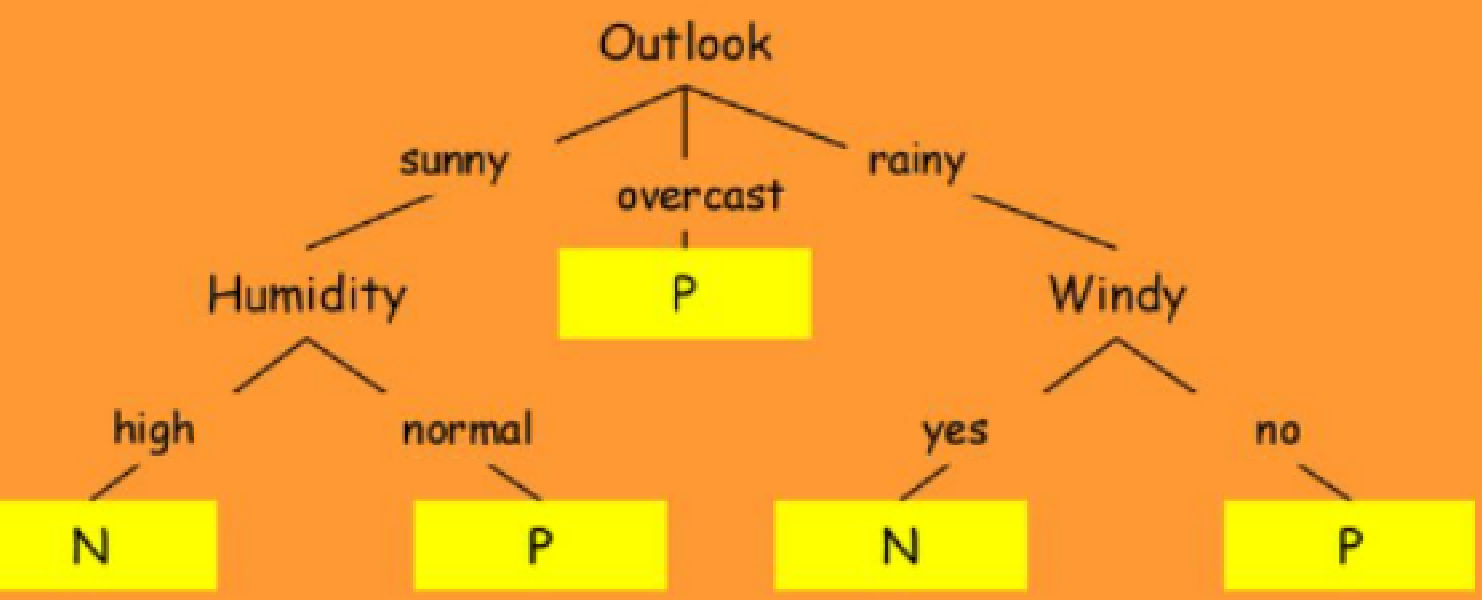
Day	Temp	Humidity	Wind	Play Tennis
D1	Hot	High	Weak	No

Outlook
Sunny

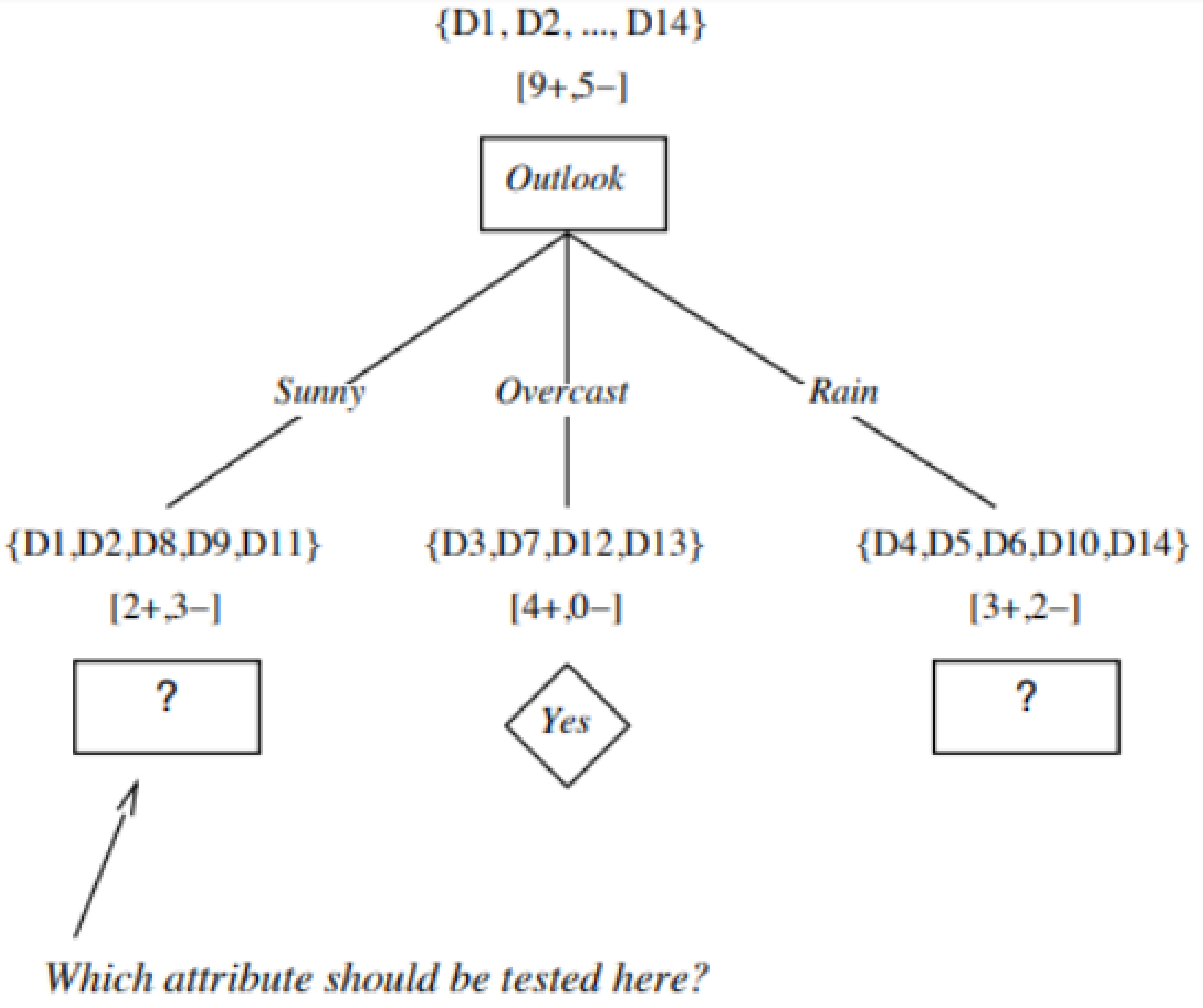
Complicated Tree



Simple Tree



Day	Temp	Humidity	Wind	Play Tennis
D1	Hot	High	Weak	No
D2	Hot	High	Strong	No
D8	Mild	High	Weak	No
D9	Cool	Normal	Weak	Yes
D11	Mild	Normal	Strong	Yes



Day	Temp	Humidity	Wind	Play Tennis
D4	Mild	High	Weak	Yes
D5	Cool	Normal	Weak	Yes
D6	Cool	Normal	Strong	No
D10	Mild	Normal	Weak	Yes
D14	Mild	High	Strong	No

ID3 - S_{sunny}

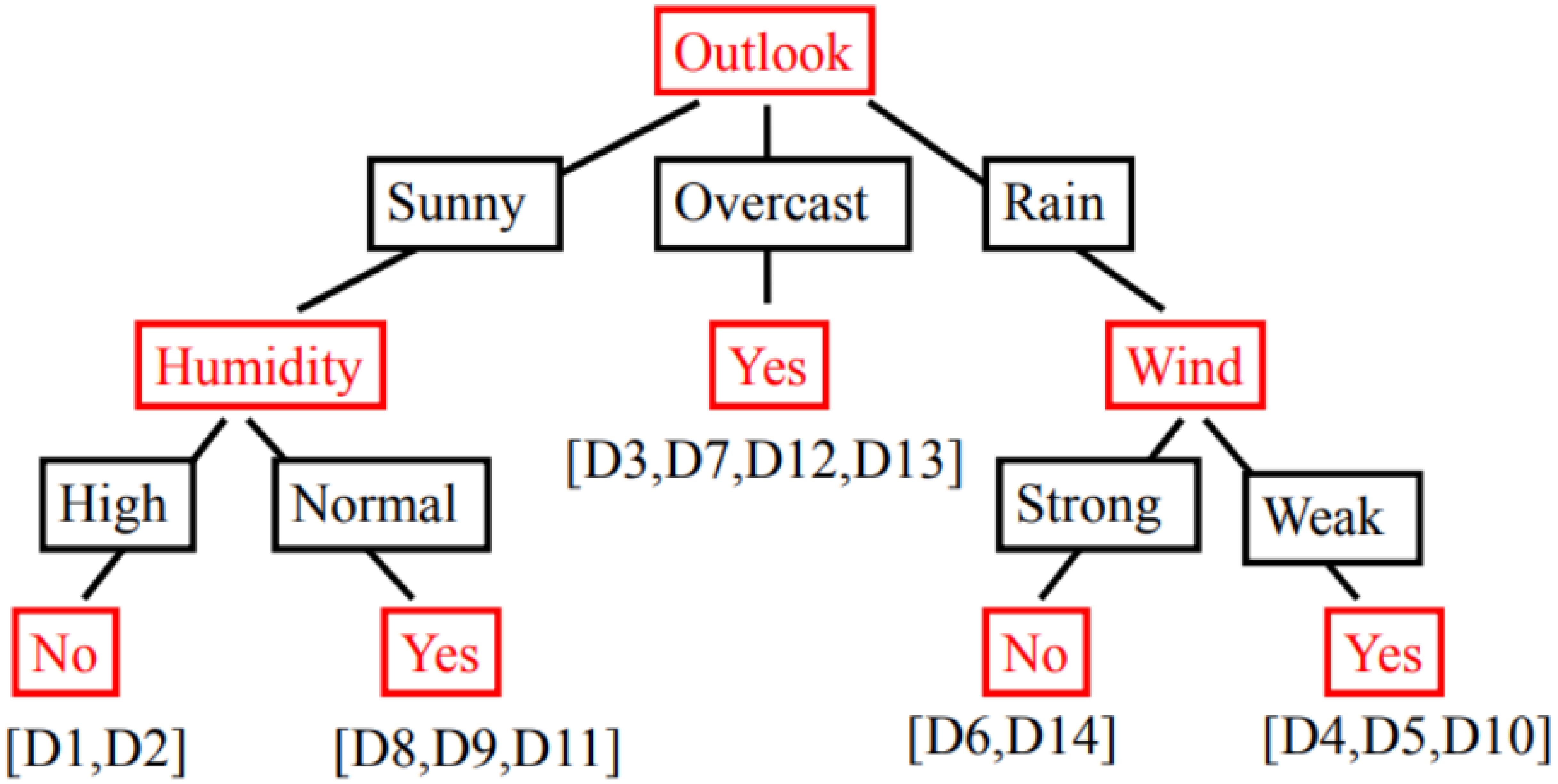
Gain(S_{sunny} , Humidity) = 0.970-(3/5)0.0 – 2/5(0.0) = **0.970**

Gain(S_{sunny} , Temp.) = 0.970-(2/5)0.0 –2/5(1.0)-(1/5)0.0 = 0.570

Gain(S_{sunny} , Wind) = 0.970= -(2/5)1.0 – 3/5(0.918) = 0.019

So, Hummudity will be selected

ID3 - Result


Outlook
Sunny
Overcast
Rain
Humidity
Yes
Wind
[D3,D7,D12,D13]
High
Normal
Strong
Weak
No
Yes
No
Yes
[D1,D2]
[D8,D9,D11]
[D6,D14]
[D4,D5,D10]

Inductive Bias of ID3:

- Shorter trees are preferred over longer trees.
- Trees that place high information gain attributes close to the root are preferred over those that do not.

→ Demonstrates preference bias (or search bias)

Preference for certain hypothesis over others

Inductive Bias in ID3 - Occam's Razor

OCCAM'S RAZOR: Prefer the simplest hypothesis that fits the data.

The answer that requires the fewest assumptions is generally the correct one.

Why prefer short hypotheses?

Argument in favor:

- Fewer short hypotheses than long hypotheses
- A short hypothesis that fits the data is unlikely to be a coincidence
- A long hypothesis that fits the data might be a coincidence

Argument opposed:

- There are many ways to define small sets of hypotheses
- What is so special about small sets based on *size* of hypothesis

Choose the explanation that is most Simple & straightforward

Decision Tree Advantages

- Inexpensive to construct
- Extremely fast at classifying unknown records
  - O(*d*) where *d* is the depth of the tree
- Presence of redundant attributes does not adversely affect the accuracy of decision trees
  - One of the two redundant attributes will not be used for splitting once the other attribute is chosen
- Nonparametric approach
  - Does not require any prior assumptions regarding probability distributions, means, variances, etc.

–Easy to interpret for small-sized trees → Inherently interpretable
–Robust to noise (especially when methods to avoid overfitting are employed)
–Can easily handle redundant or irrelevant attributes (unless the attributes are interacting)

Disadvantages

1. Overfitting (High Variance)

→ Not able to generalize

→ Small changes in data changes the tree.

2. Single attribute based decision boundaries are rectilinear.

Handling interactions

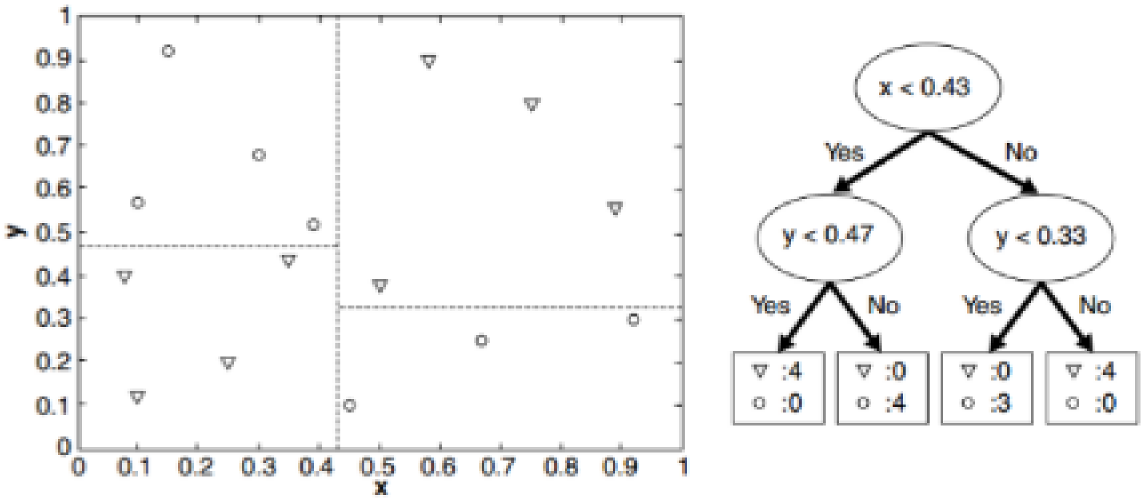


Figure 4.20. Example of a decision tree and its decision boundaries for a two-dimensional data set.