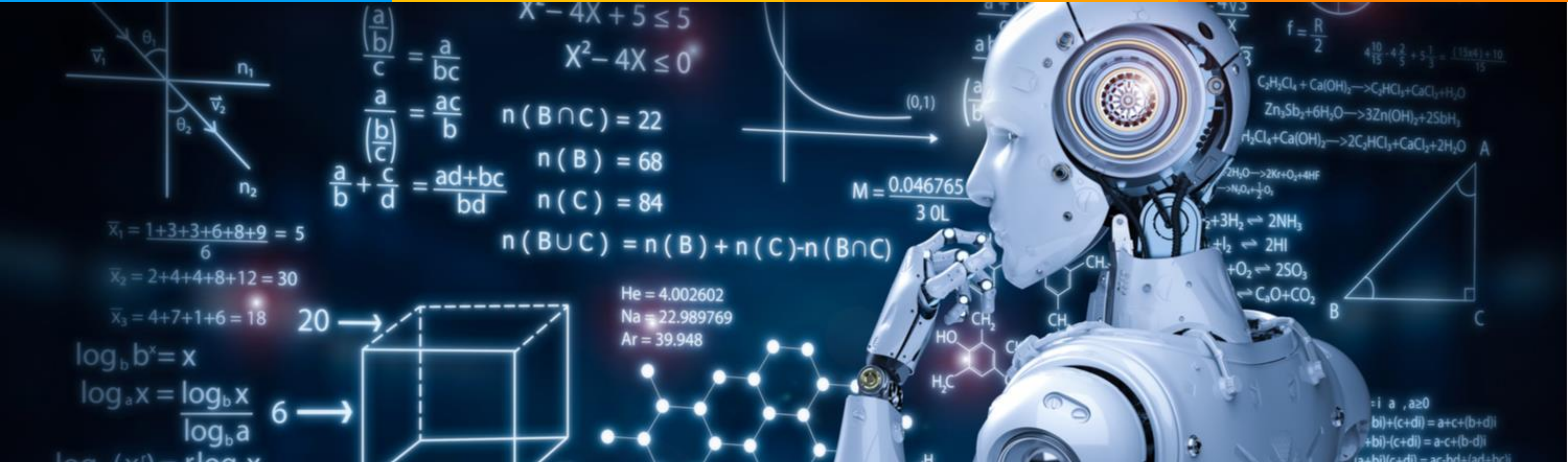
Duration: 60 min | Srikanth Shenoy | AML 5102 | M.E. AI & ML



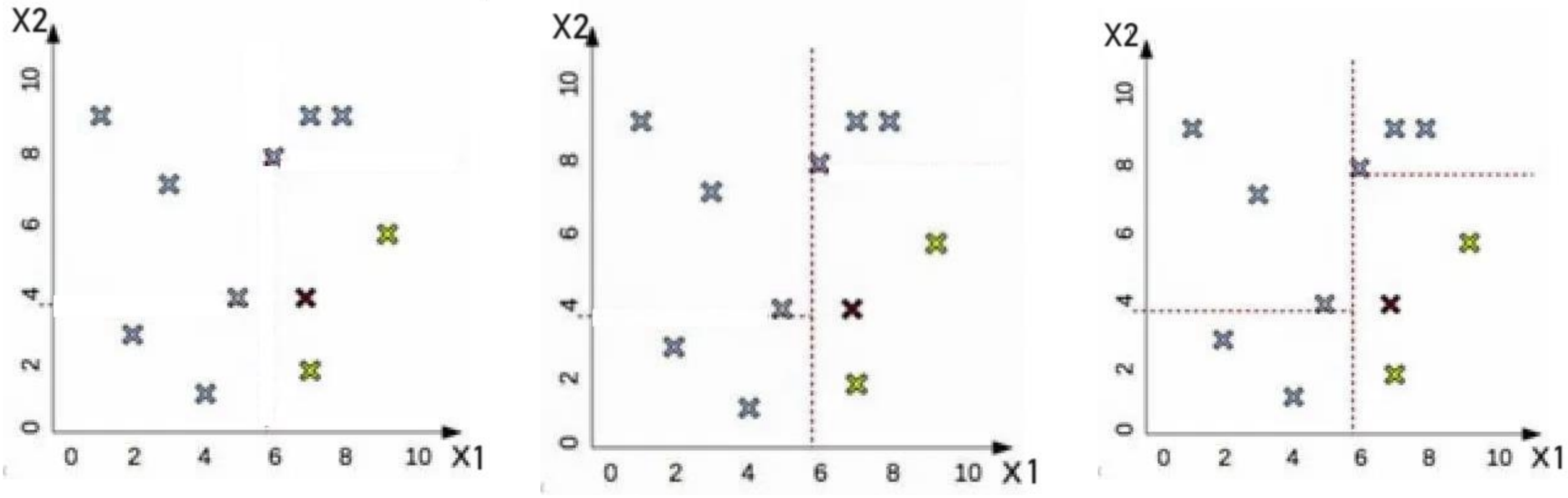
Lecture 17: KD Trees & Decision Trees

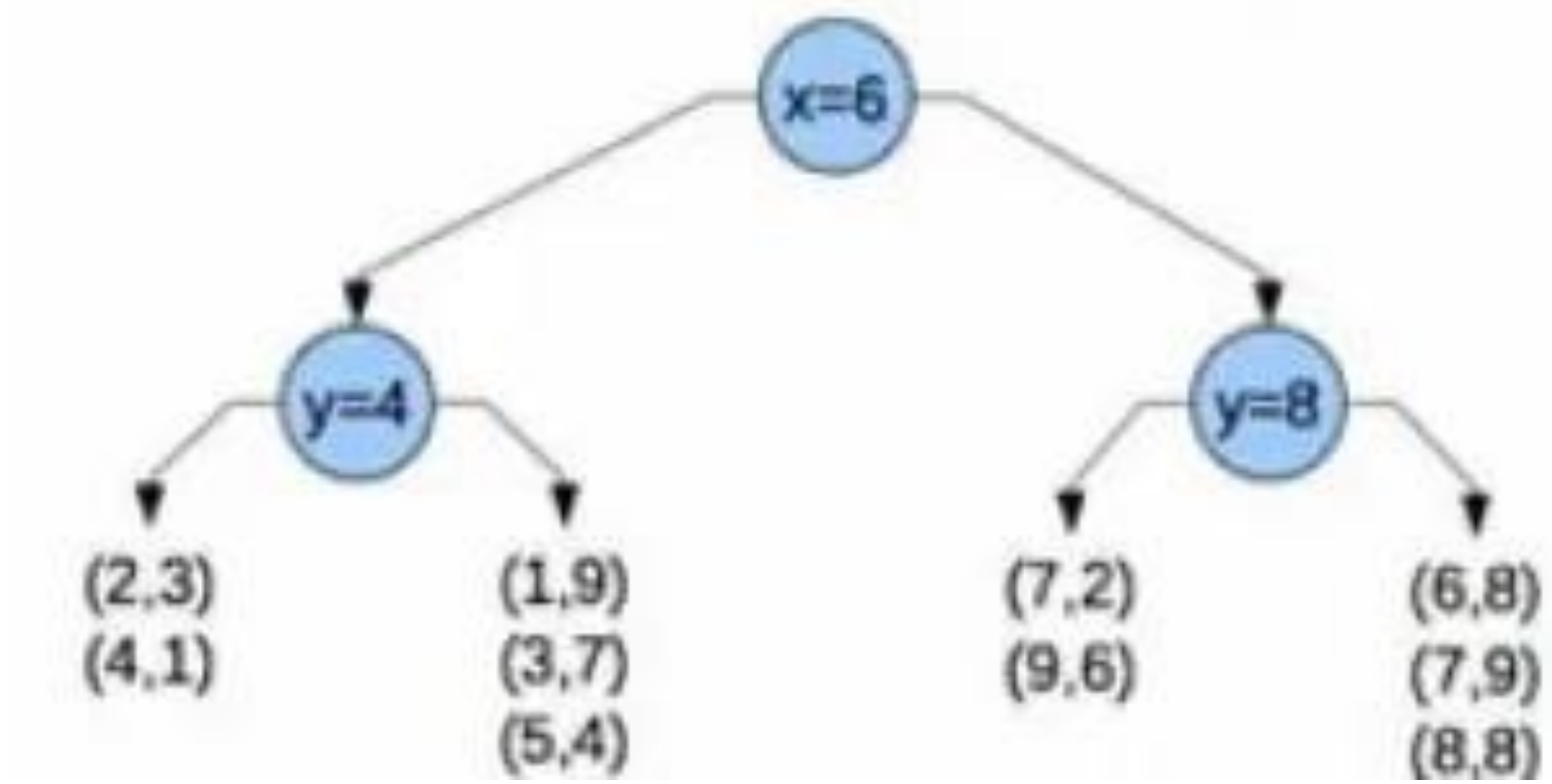
Recap

- Hierarchical Clustering
  - Divisive Clustering
  - Agglomerative Clustering
- Linkages
  - Simple, Complete, Average, Centroid, Ward
- Dendrogram


X²– 4X ≤ 0
n (B ∩ C) = 22
n (B) = 68
n (C) = 84
n (B ∪ C) = n (B) + n (C)-n (B ∩ C)
a/b + c/d = ad+bc/bd
x̄₂ = 2+4+4+8+12 = 30
x̄₃ = 4+7+1+6 = 18
log_b b^x = x
20 →
6 →
He = 4.002602
Na = 22.989769
Ar = 39.948
(0,1)

KD Trees

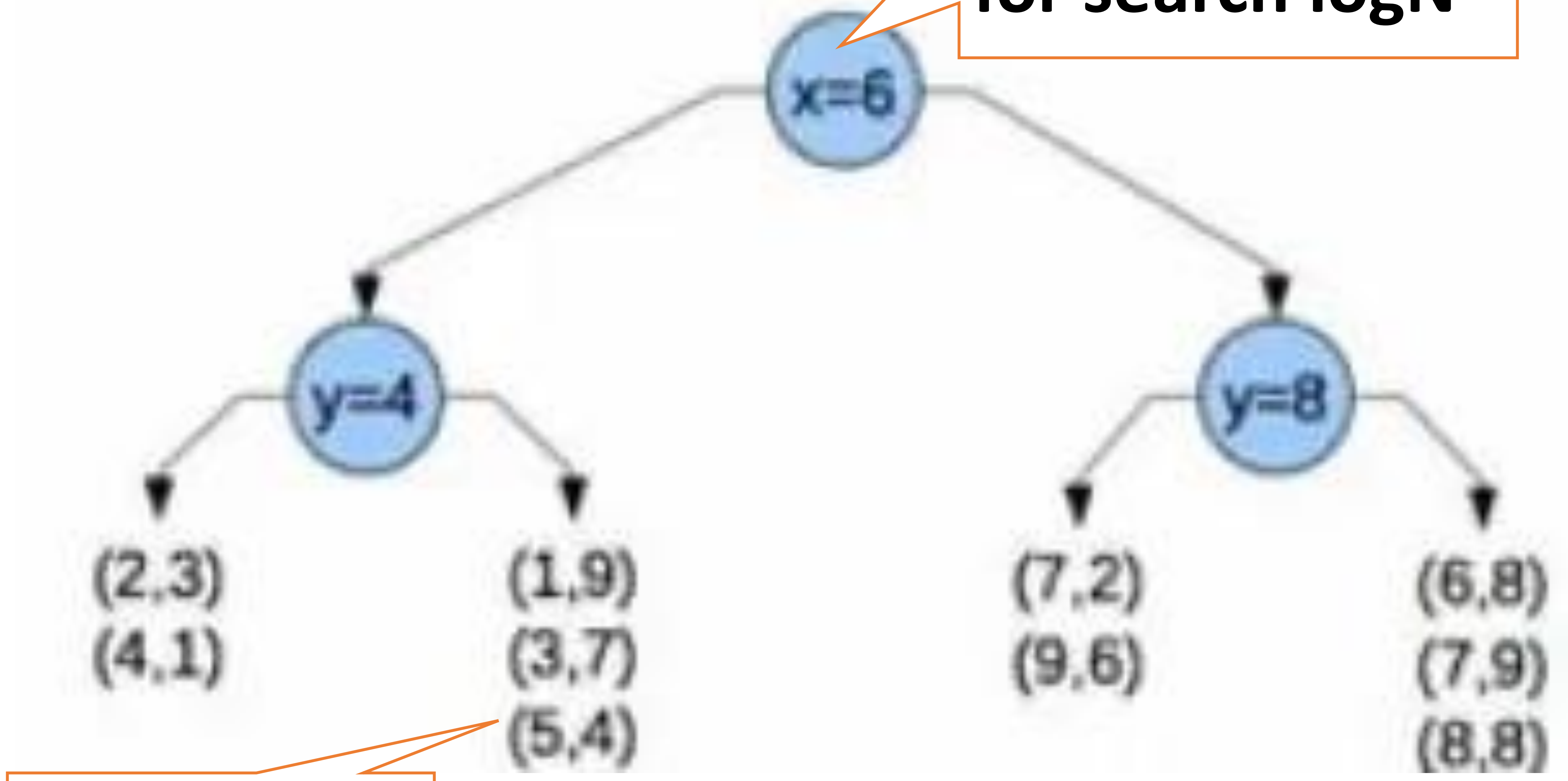

X2
X1
0
2
4
6
8
10


x=6
y=4
y=8
(2,3)
(4,1)
(1,9)
(3,7)
(5,4)
(7,2)
(9,6)
(6,8)
(7,9)
(8,8)

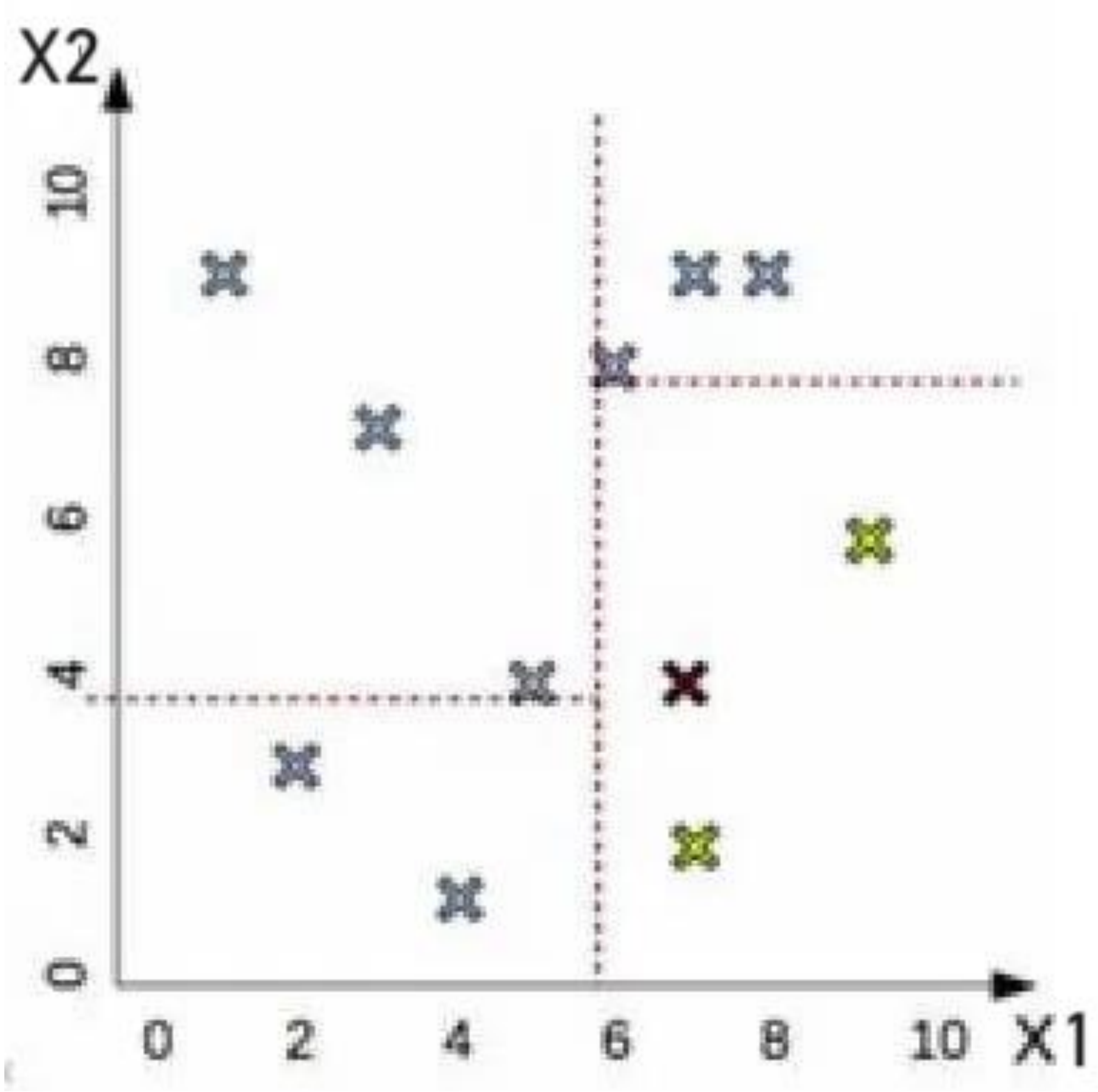
KD Trees - Prediction

- How to search?
  - DFS

Cost of going down the tree for search logN



kNN within a node



Decision Trees

Training examples: **9 yes / 5 no**

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

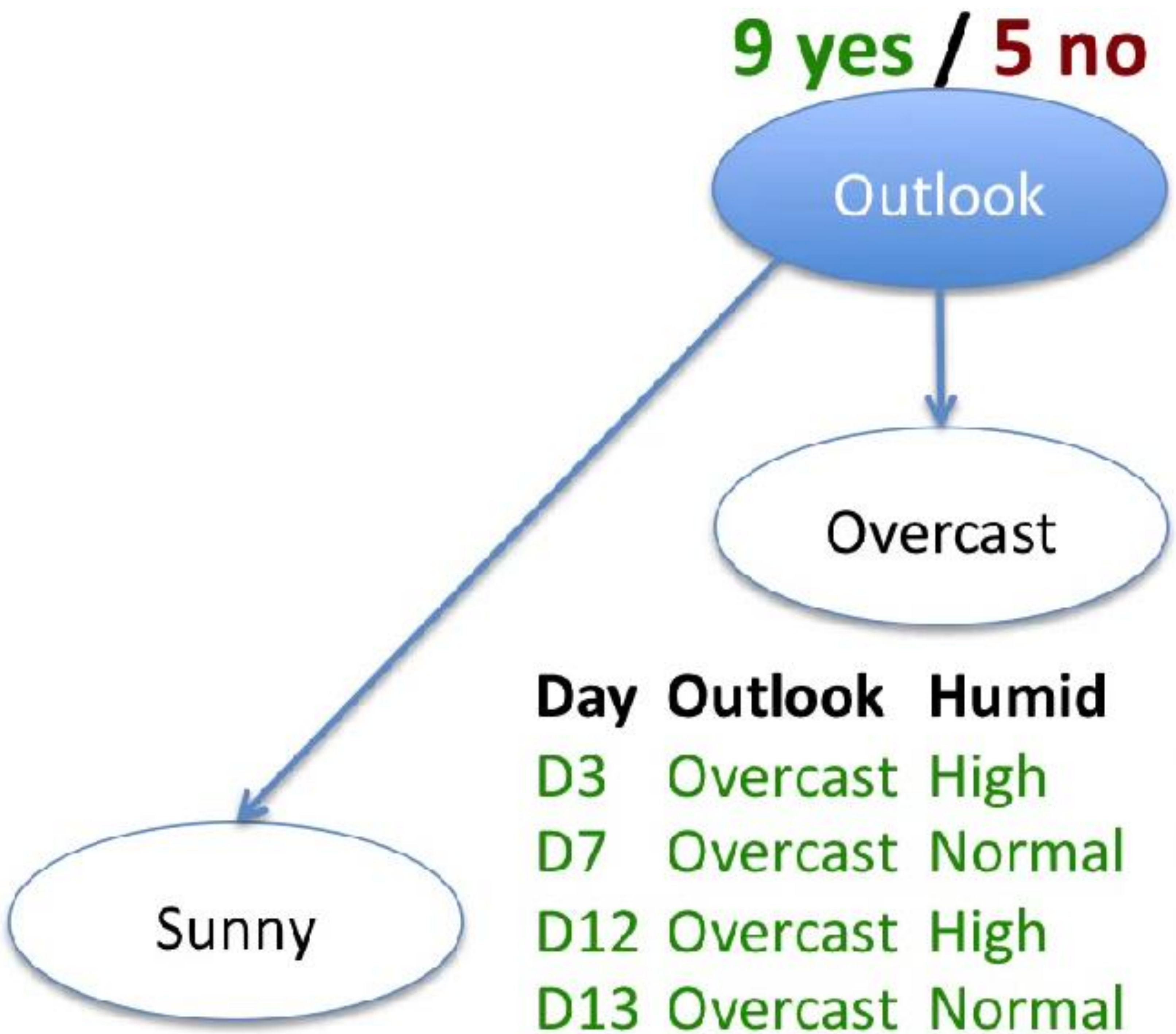
New data:

D15	Rain	High	Weak	?

- Training
  - Split on a feature
  - Are subsets pure?
    - If Yes, Stop
    - If no repeat
- Prediction


9 yes / 5 no
Outlook

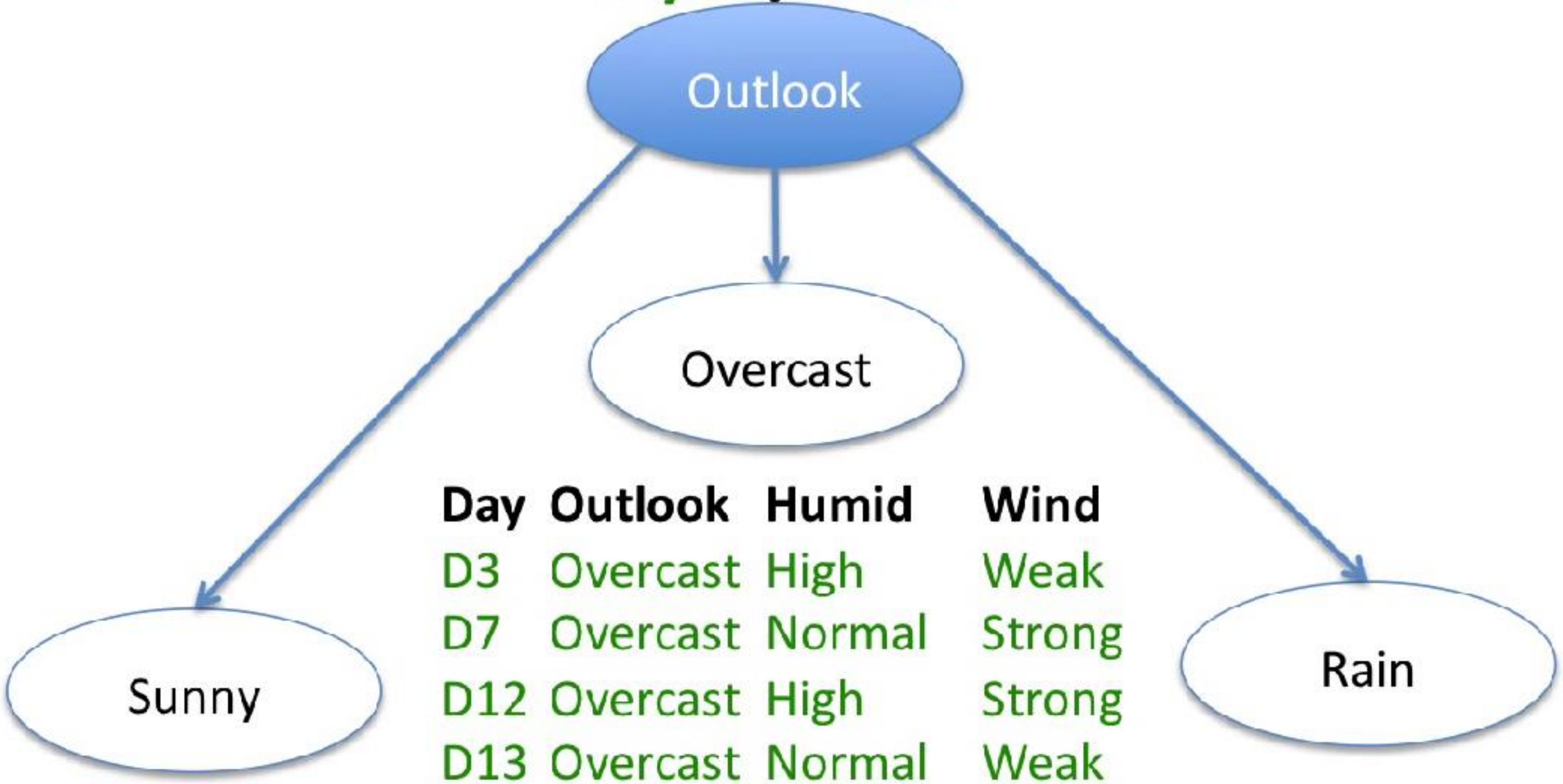
Day	Outlook	Humid	Wind
D1	Sunny	High	Weak
D2	Sunny	High	Strong
D8	Sunny	High	Weak
D9	Sunny	Normal	Weak
D11	Sunny	Normal	Strong



Day	Outlook	Humid	Wind
D3	Overcast	High	Weak
D7	Overcast	Normal	Strong
D12	Overcast	High	Strong
D13	Overcast	Normal	Weak

Day	Outlook	Humid	Wind
D1	Sunny	High	Weak
D2	Sunny	High	Strong
D8	Sunny	High	Weak
D9	Sunny	Normal	Weak
D11	Sunny	Normal	Strong

9 yes / 5 no



Day	Outlook	Humid	Wind
D3	Overcast	High	Weak
D7	Overcast	Normal	Strong
D12	Overcast	High	Strong
D13	Overcast	Normal	Weak

4 yes / 0 no

pure subset

Day	Outlook	Humid	Wind
D1	Sunny	High	Weak
D2	Sunny	High	Strong
D8	Sunny	High	Weak
D9	Sunny	Normal	Weak
D11	Sunny	Normal	Strong

2 yes / 3 no

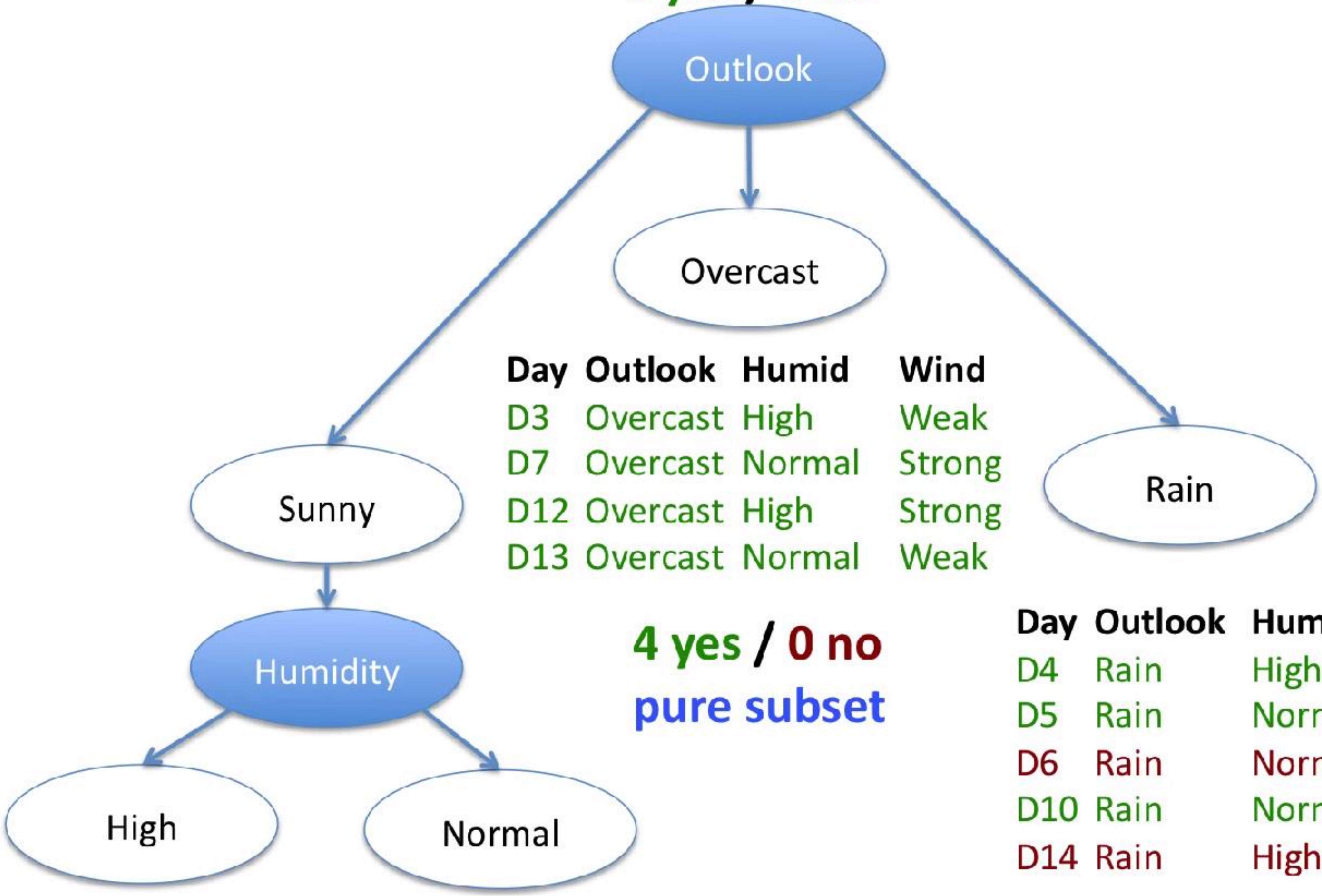
split further

Day	Outlook	Humid	Wind
D4	Rain	High	Weak
D5	Rain	Normal	Weak
D6	Rain	Normal	Strong
D10	Rain	Normal	Weak
D14	Rain	High	Strong

3 yes / 2 no

split further

9 yes / 5 no



Day	Outlook	Humid	Wind
D3	Overcast	High	Weak
D7	Overcast	Normal	Strong
D12	Overcast	High	Strong
D13	Overcast	Normal	Weak

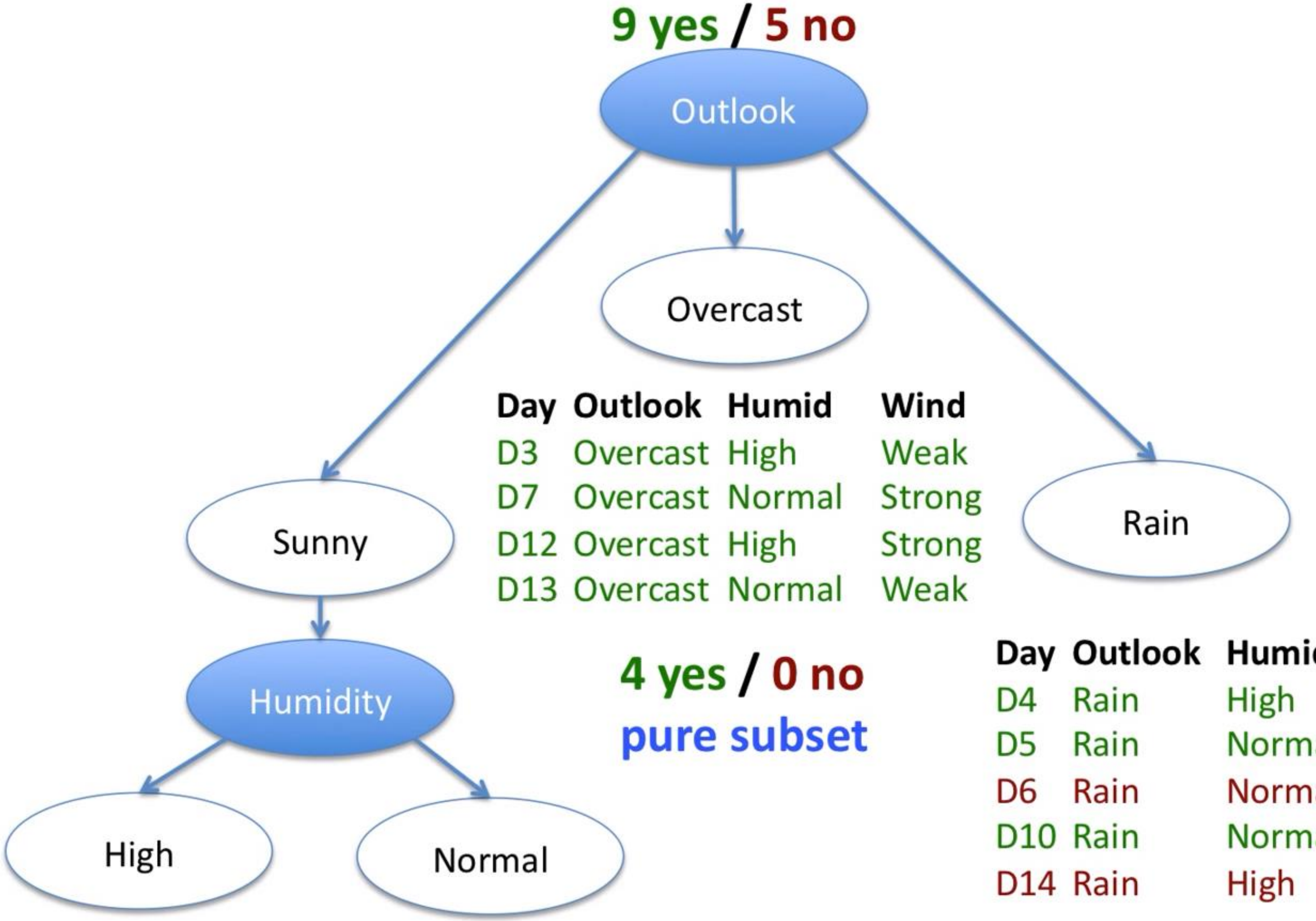
4 yes / 0 no
pure subset

Day	Outlook	Humid	Wind
D4	Rain	High	Weak
D5	Rain	Normal	Weak
D6	Rain	Normal	Strong
D10	Rain	Normal	Weak
D14	Rain	High	Strong

3 yes / 2 no
split further

Day	Humid	Wind
D1	High	Weak
D2	High	Strong
D8	High	Weak

Day	Humid	Wind
D9	Normal	Weak
D11	Normal	Strong



Day	Outlook	Humid	Wind
D3	Overcast	High	Weak
D7	Overcast	Normal	Strong
D12	Overcast	High	Strong
D13	Overcast	Normal	Weak

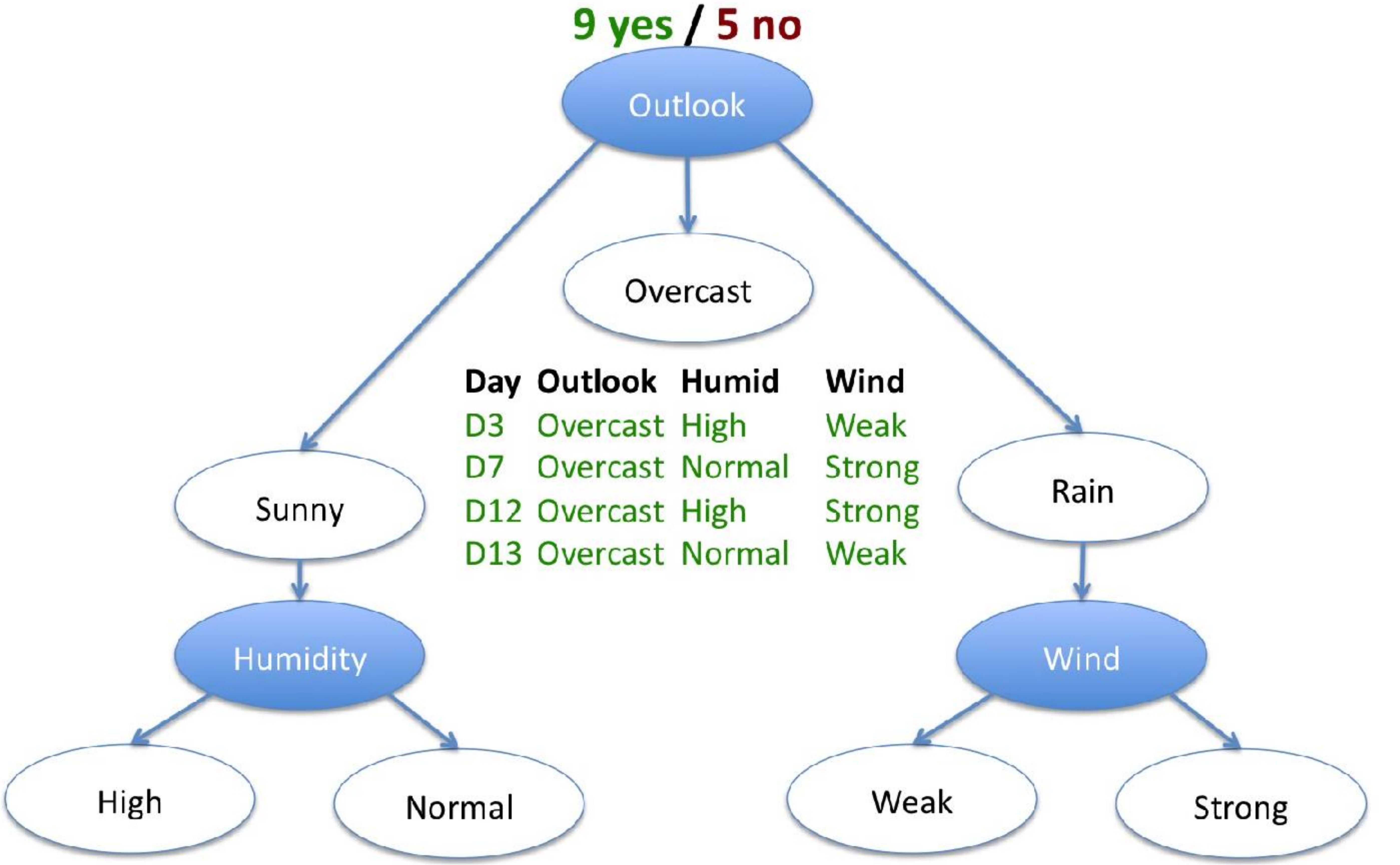
4 yes / 0 no
pure subset

Day	Outlook	Humid	Wind
D4	Rain	High	Weak
D5	Rain	Normal	Weak
D6	Rain	Normal	Strong
D10	Rain	Normal	Weak
D14	Rain	High	Strong

3 yes / 2 no
split further

Day	Humid	Wind
D1	High	Weak
D2	High	Strong
D8	High	Weak

Day	Humid	Wind
D9	Normal	Weak
D11	Normal	Strong



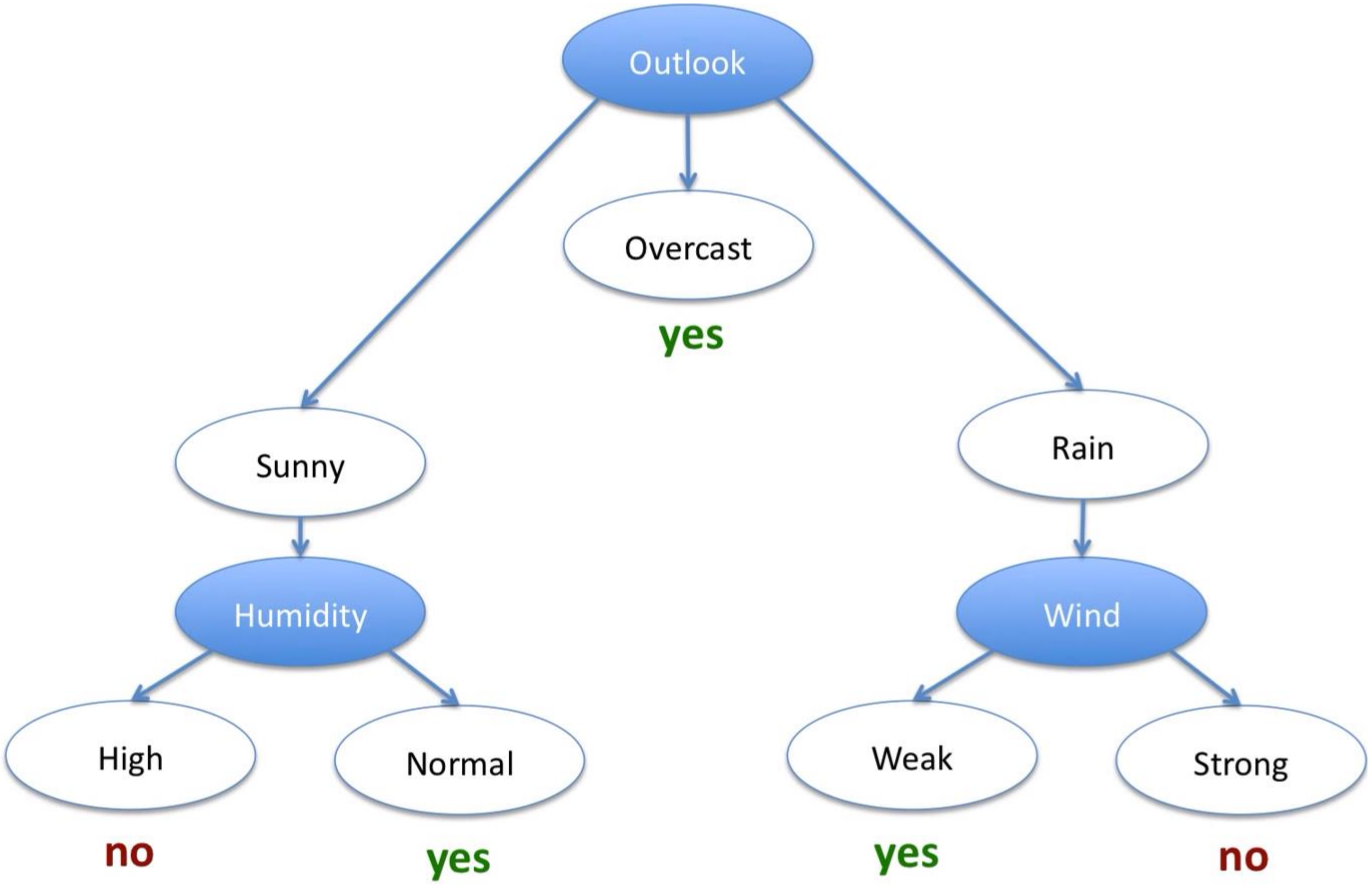
Day	Outlook	Humid	Wind
D3	Overcast	High	Weak
D7	Overcast	Normal	Strong
D12	Overcast	High	Strong
D13	Overcast	Normal	Weak

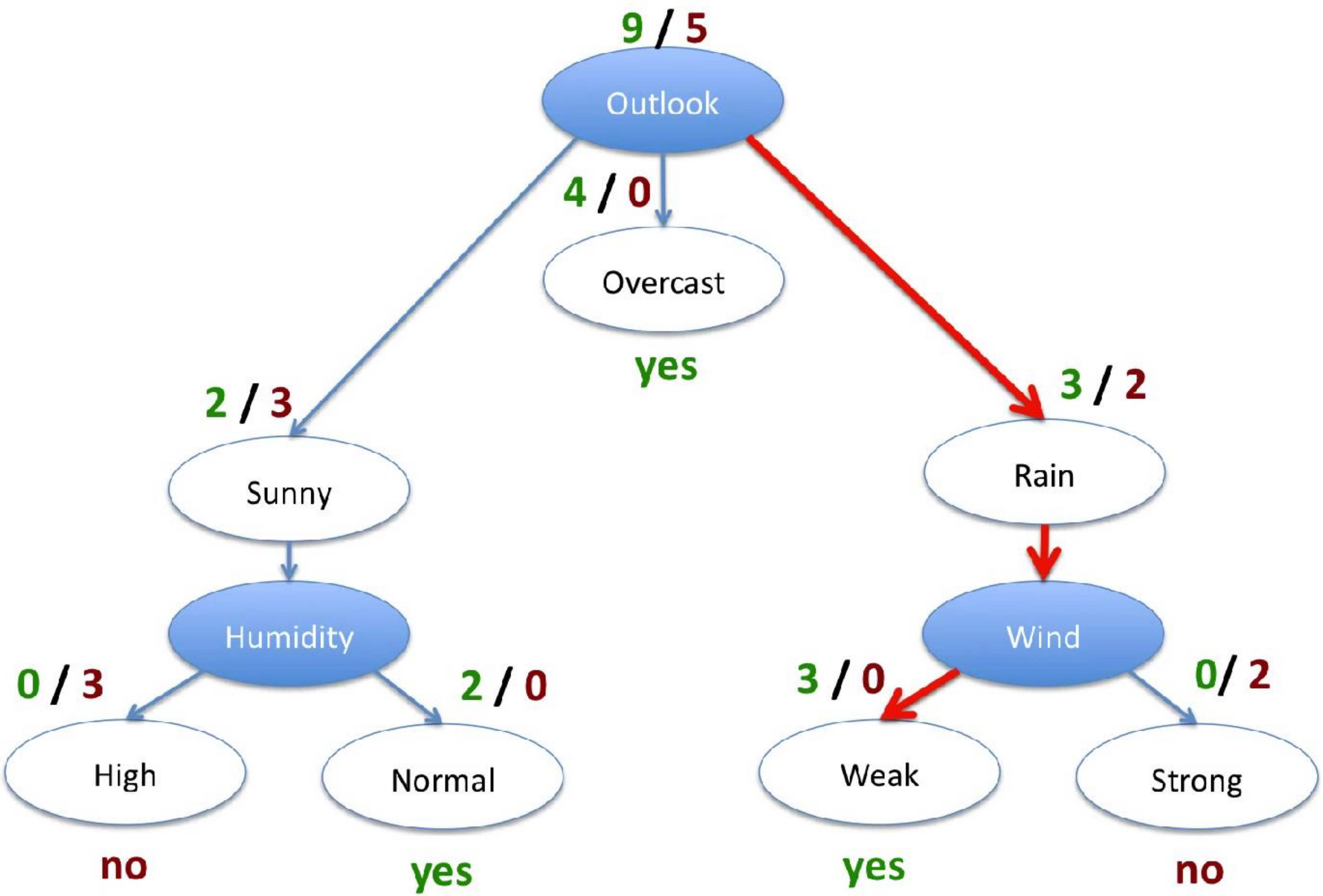
Day	Humid	Wind
D1	High	Weak
D2	High	Strong
D8	High	Weak

Day	Humid	Wind
D9	Normal	Weak
D11	Normal	Strong

Day	Humid	Wind
D4	High	Weak
D5	Normal	Weak
D10	Normal	Weak

Day	Humid	Wind
D6	Normal	Strong
D14	High	Strong


Outlook
Overcast
yes
Sunny
Rain
Humidity
Wind
High
Normal
Weak
Strong
no
yes
yes
no



New data:

Day	Outlook	Humid	Wind	
D15	Rain	High	Weak	→ Yes

Decision Tree – ID3 algorithm

ID3 algorithm

- Recursive algorithm
- Operates on node
- Split(data):
  - Pick the best feature xi to split data (how)
  - xi becomes decision attribute for this node
  - Create a node
  - Branch on all possible values of xi (categorical)
  - Split data for each categorical value of xi
  - Send to each branch
    - If subset is pure, Stop
    - If impure split(data-subset)

Begin with all data

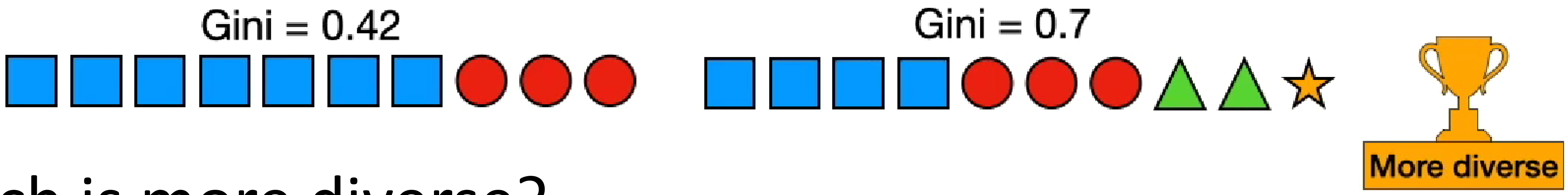
Discovered independently in 1983/84 by Quinlan and Breimanetal

Performance depends on attribute picked for split

Gini Impurity

Gini Impurity Intuition

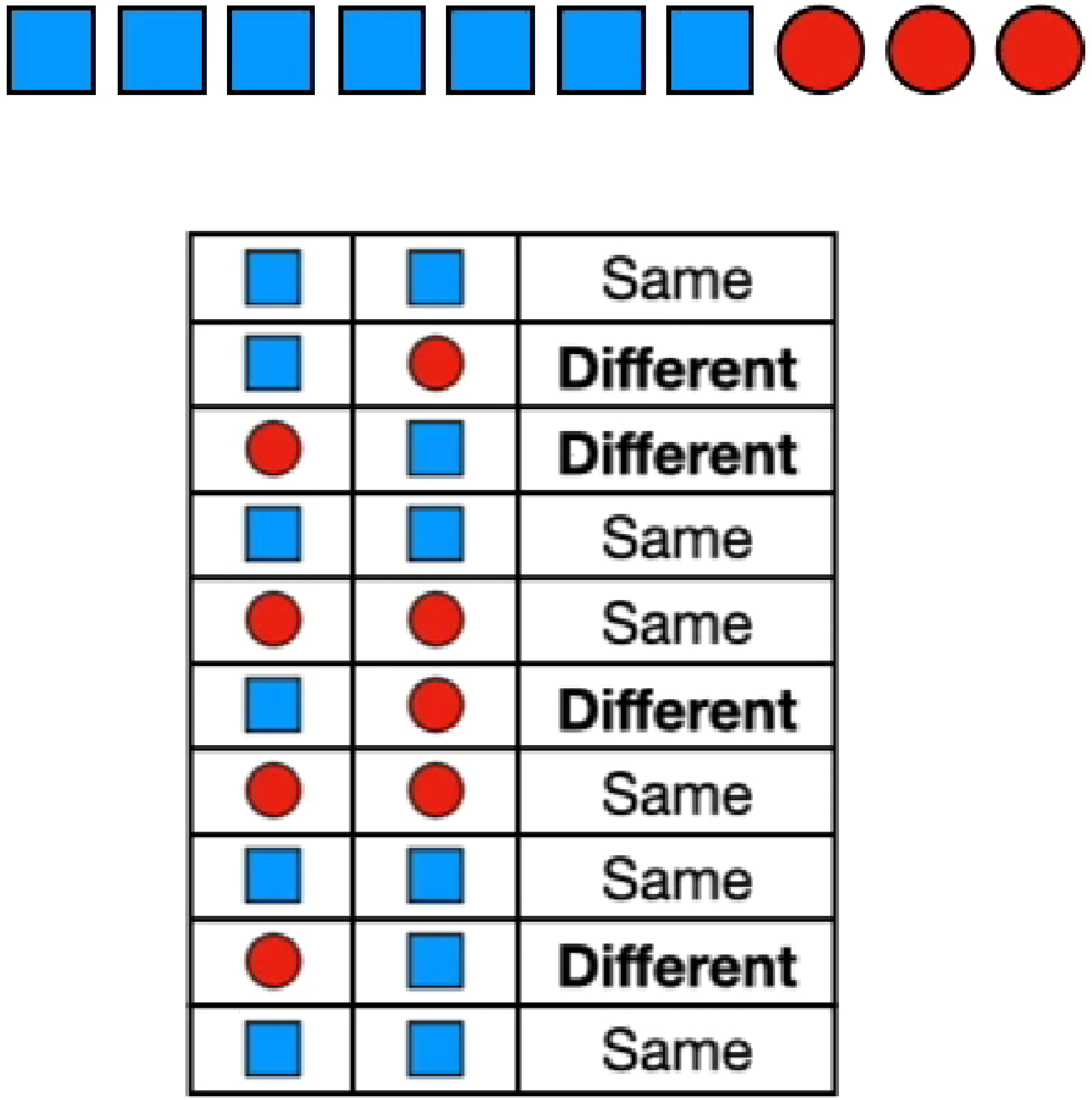
- Measures the mixture of categorical variable



- Which is more diverse?
- How to measure diversity/impurity?

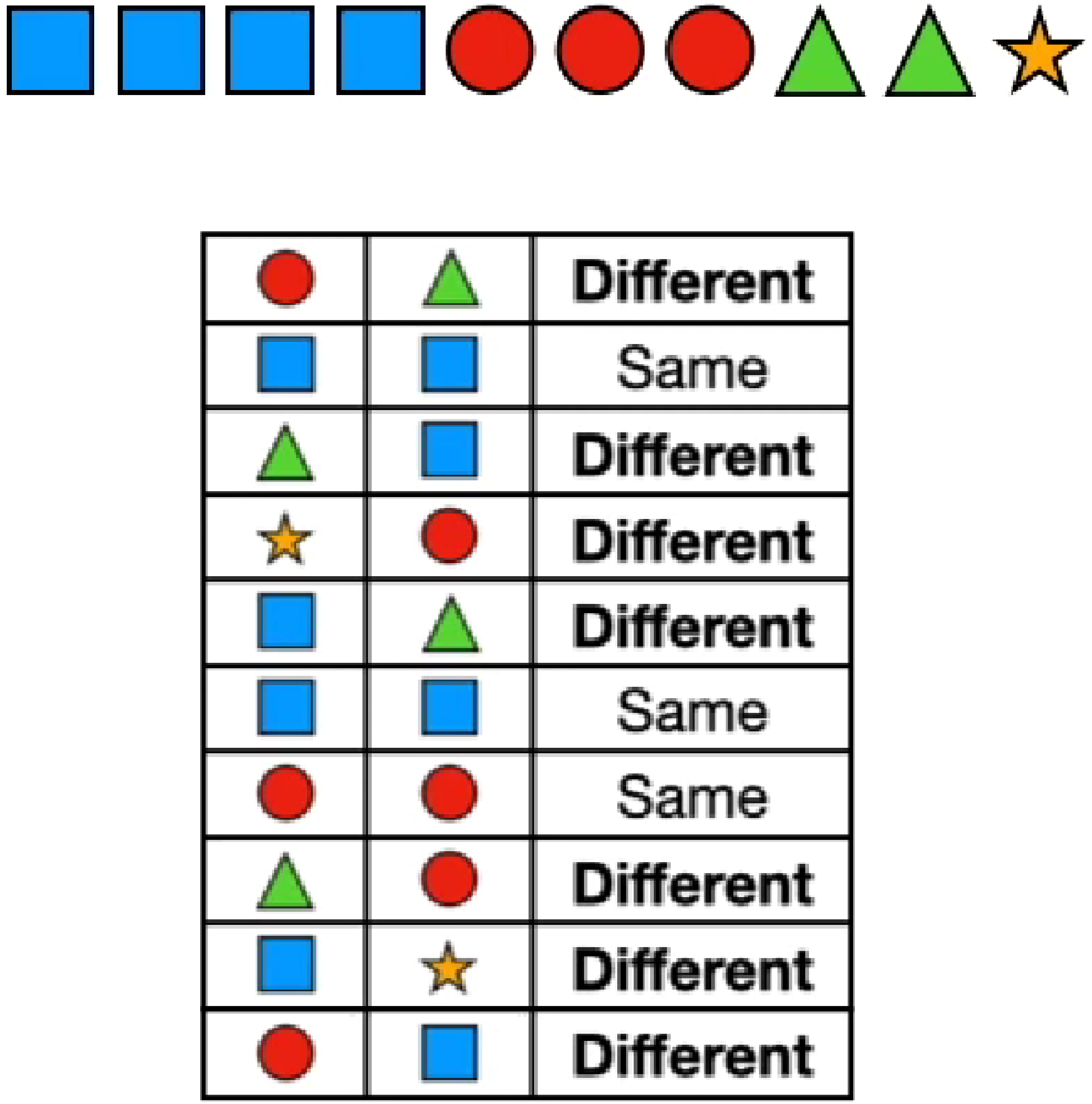
Gini Impurity Intuition - 2

Gini = 0.42



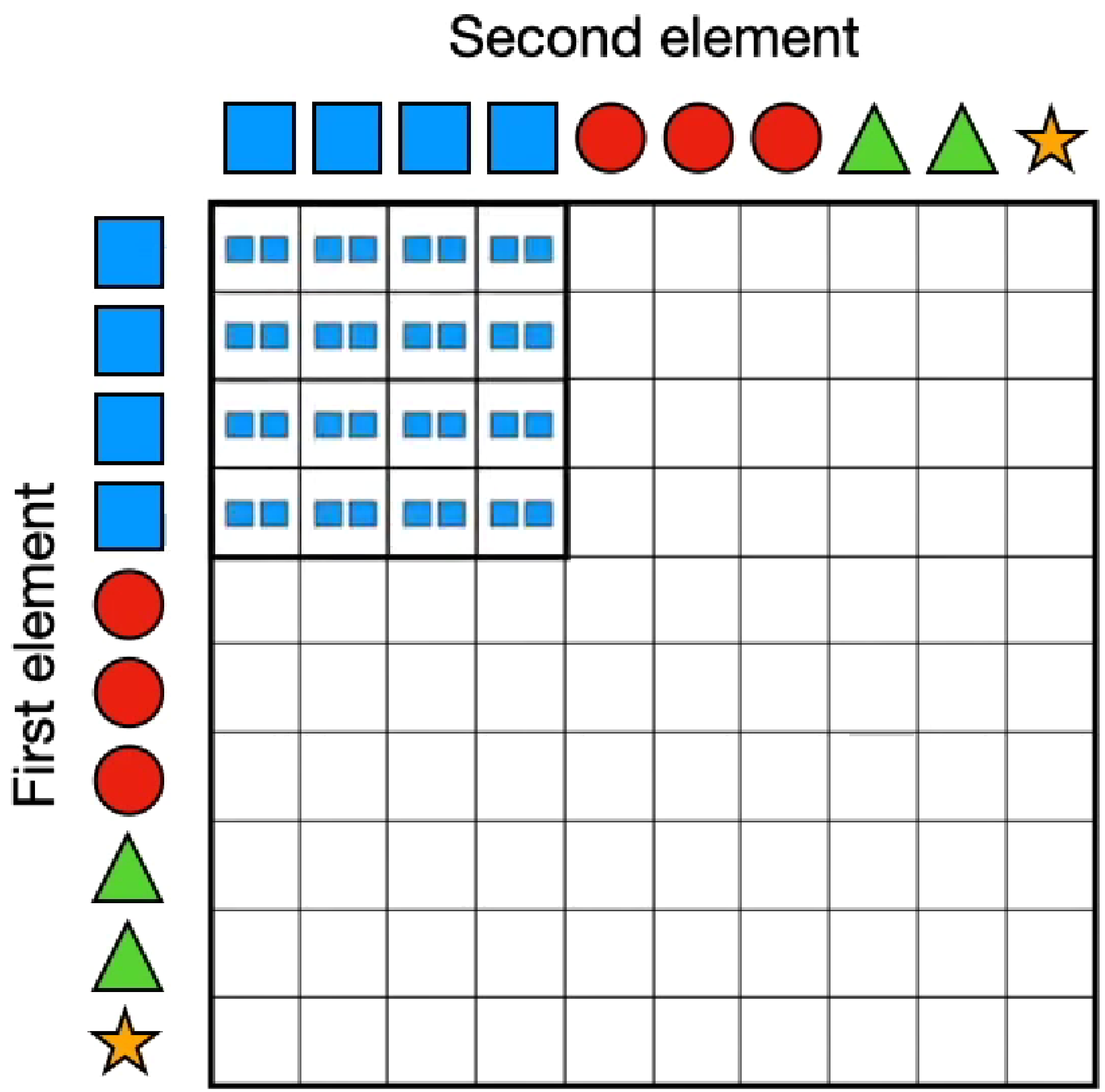
Different:
4 out of 10

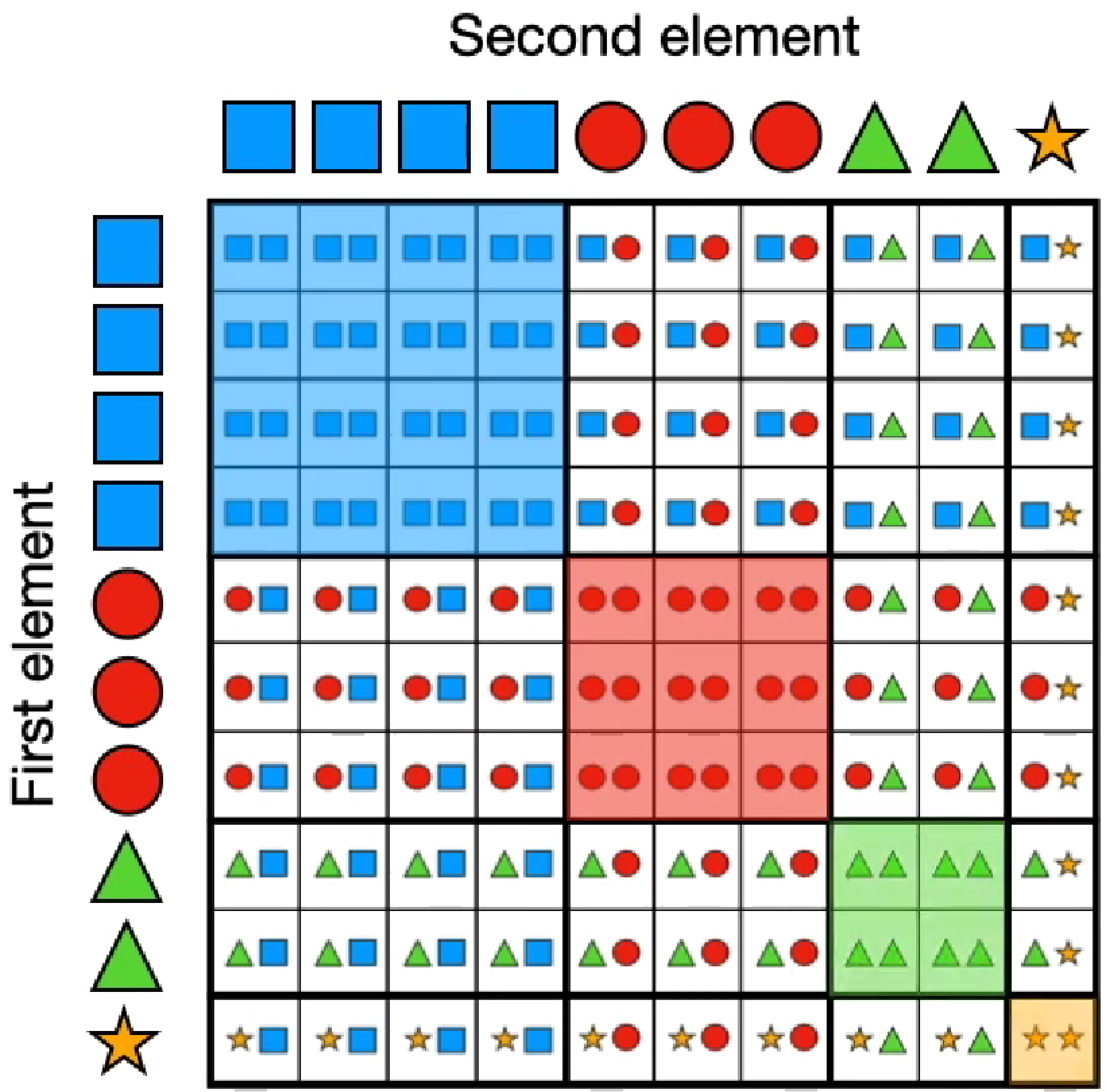
Gini = 0.7



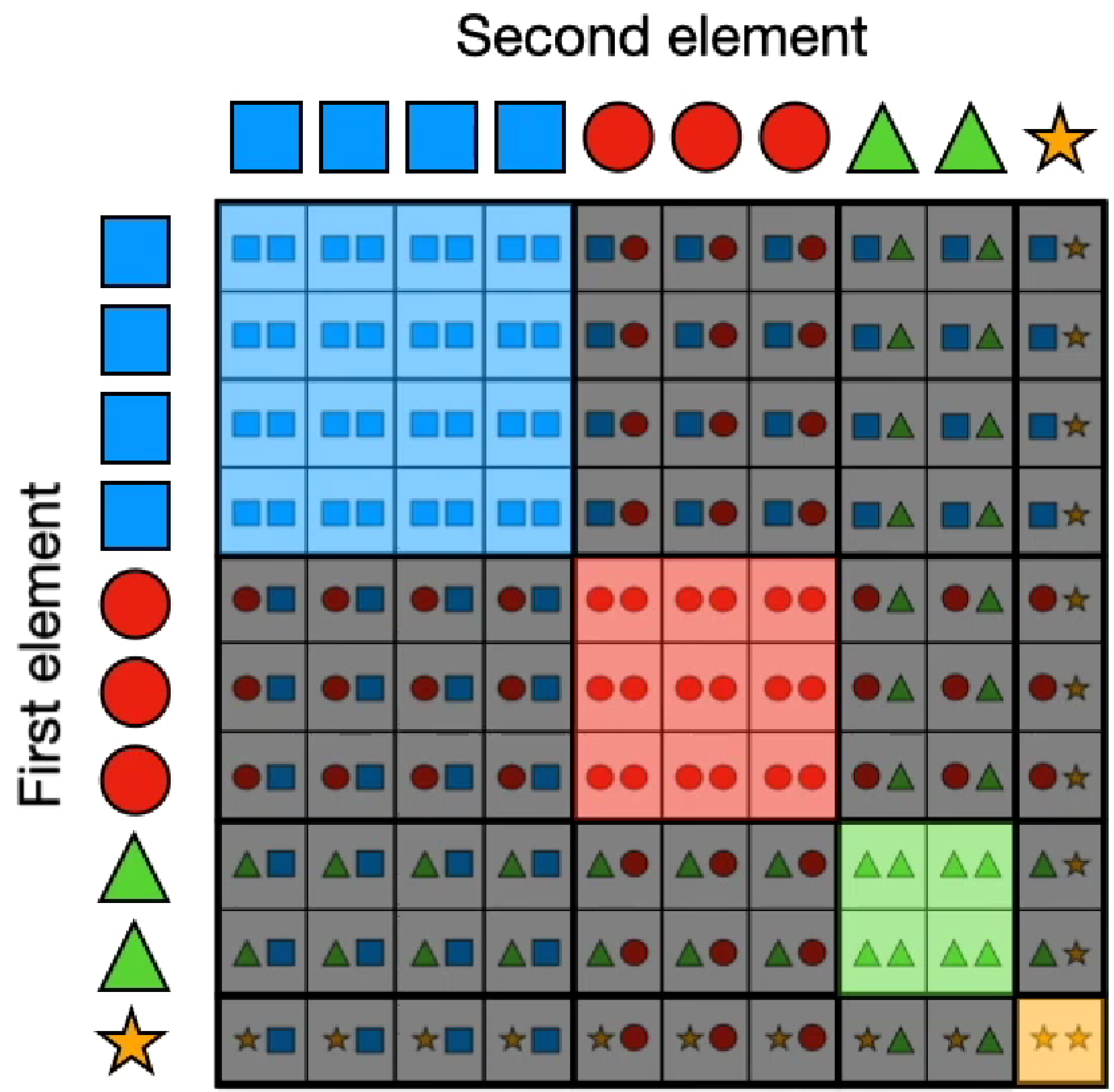
Different:
7 out of 10

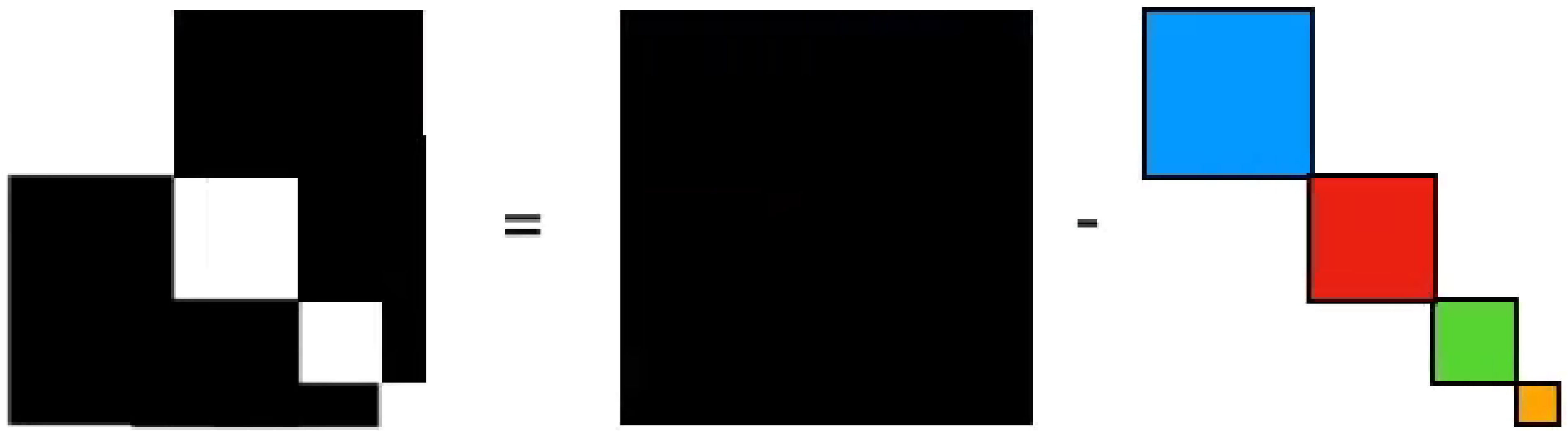
Gini Impurity Intuition - 3





Gini Impurity Intuition - 4

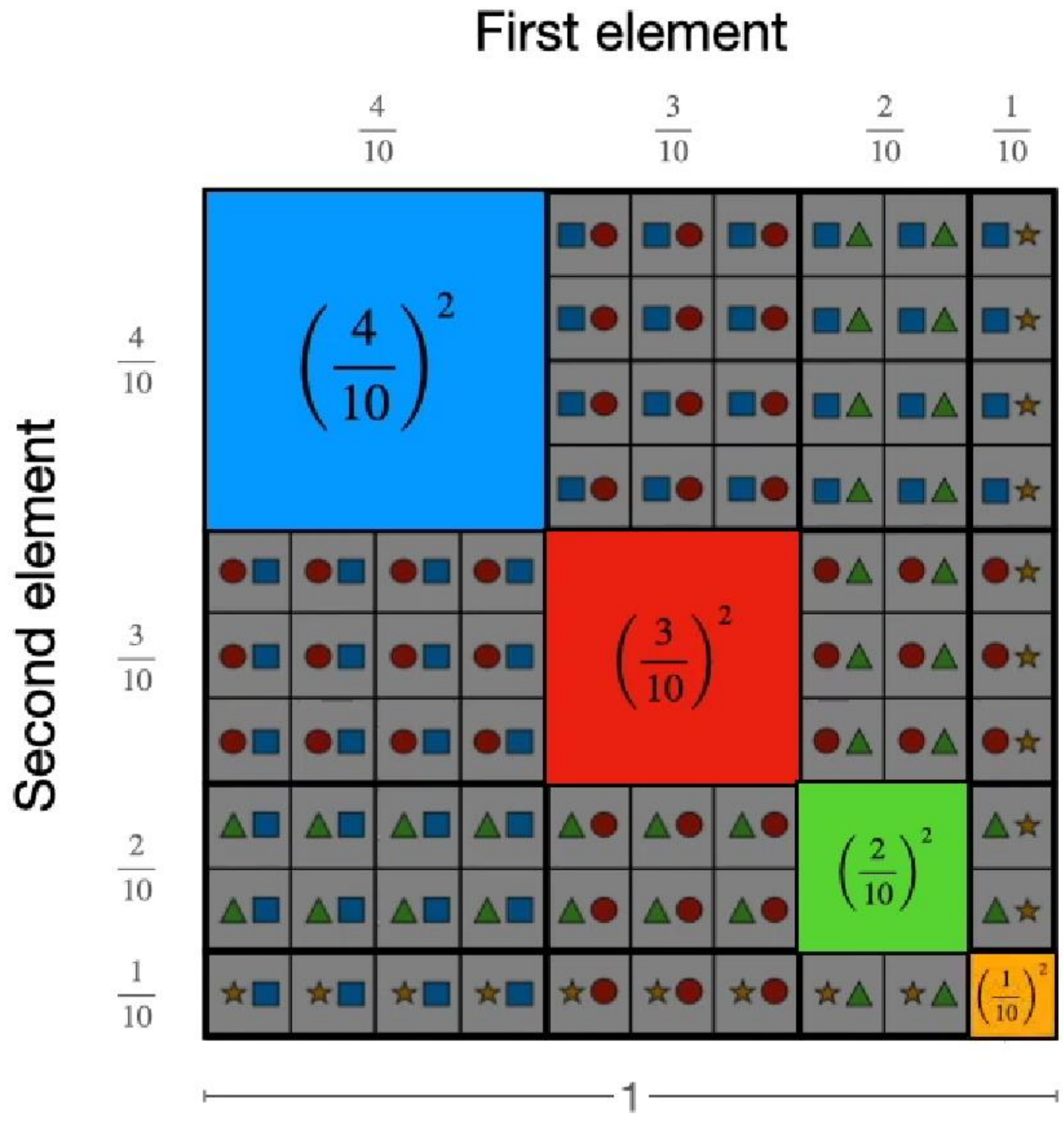




P(Both different) = P(Anything) - P(Both equal)

= 1 - P(Both blue) - P(Both red) - P(Both green) - P(Both yellow)

Gini Impurity Intuition - 5



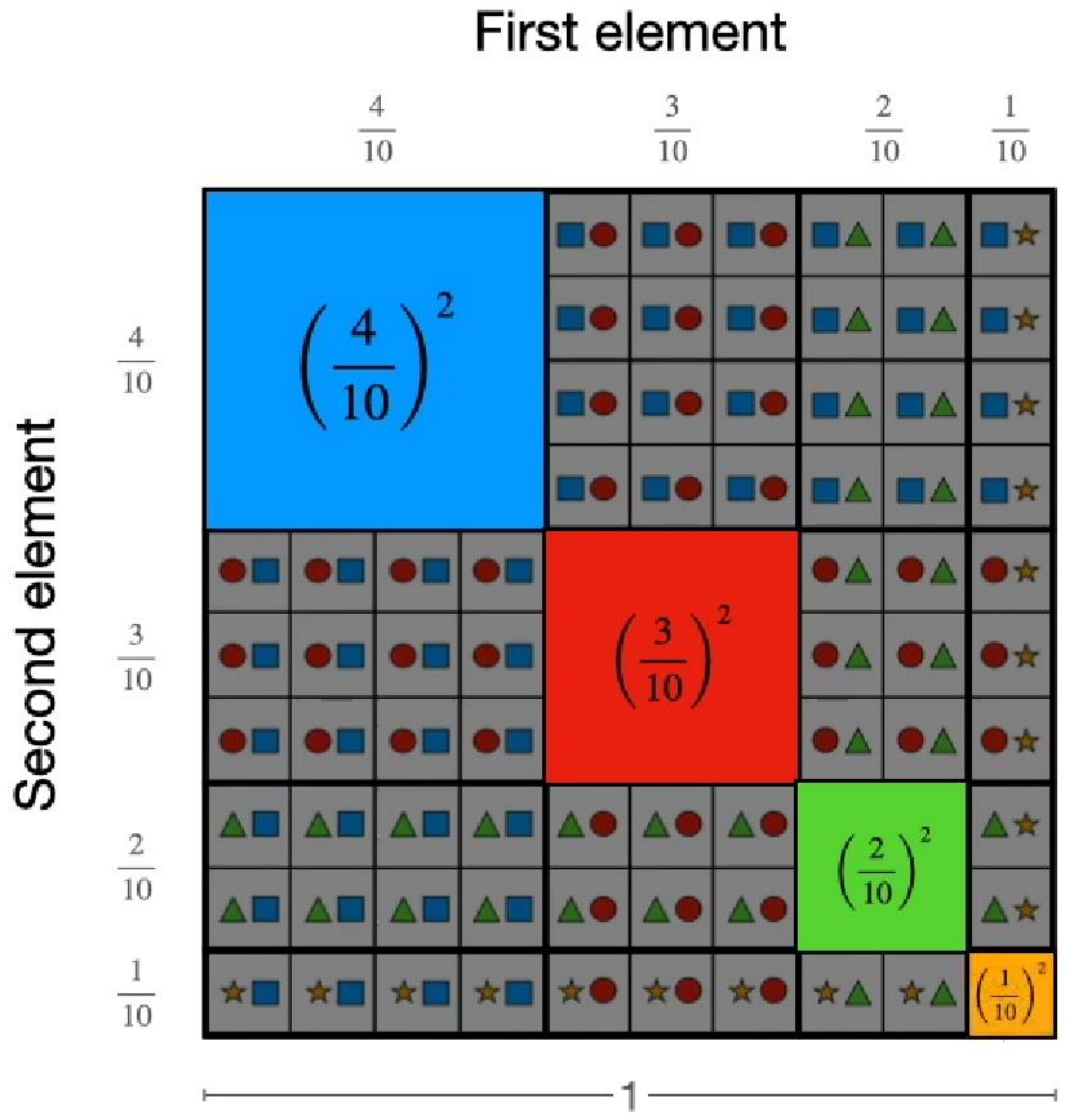
P(Both different) = P(Anything) - P(Both equal)

= 1 - P(Both blue) - P(Both red) - P(Both green) - P(Both yellow)

$$= 1 - \left(\frac{4}{10}\right)^2 - \left(\frac{3}{10}\right)^2 - \left(\frac{2}{10}\right)^2 - \left(\frac{1}{10}\right)^2$$

$$GiniImpurity = 1 - \sum_{i=1}^{k} p_i^2$$

Gini Impurity Formula



$$GiniImpurity = 1 - \sum_{i=1}^{k} p_i^2$$

$$p_1 +_p 2 + \ldots + p_k = 1$$

$$GiniImpurity = p_1 + p_2 \ldots + p_k - \sum_{i=1}^{k} p_i^2$$

$$= \sum_{i=1}^{k} p_i - \sum_{i=1}^{k} p_i^2 = \sum_{i=1}^{k} p_i - p_i^2$$

$$= \sum_{i=1}^{k} p_i(1 - p_i)$$

Gini Impurity of dataset

Training examples: **9 yes / 5 no**

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

- Calculate Gini Impurity of dataset target variable

$$1 - (P(play = Yes)^2 + P(play = No)^2)$$

$$1 - \left(\left(\frac{9}{14}\right)^2 + \left(\frac{5}{14}\right)^2 \right)$$

$$1 - (0.41 + 0.13) = 1 - 0.54 = 0.46$$

$$G(S) = 0.46$$

$$S = \{(x_1, y_1), ...(x_n, y_n)\} \quad y_i \in \{1, ...k\}$$

Gini Impurity of dataset

Training examples: **9 yes / 5 no**

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

- Gini Impurity of dataset target variable

$$1 - (P(play = Yes)^2 + P(play = No)^2)$$

$$G(S) = 0.46$$

$$S = \{(x_1, y_1), ...(x_n, y_n)\} \quad y_i \in \{1, ...k\}$$

- Gini Impurity of target variable when split on an attribute

$$G(S|Outlook = Sunny) = ?$$

$$G(S|Outlook = Overcast) = ?$$

$$G(S|Outlook = Rainy) = ?$$

Selecting attribute for split using Gini

Training examples: **9 yes / 5 no**

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

- Calculate Gini Impurity for each attribute split

$$G(S)_{Outlook-split}$$

$$G(S)_{Humidity-split}$$

$$G(S)_{Wind-split}$$

- Split on attribute with minimum resulting impurity

Gini Impurity after split

- Recall Total Probability – A and B are random variables
  - B takes two values 0 and 1

$$P(A) = P(A|B=0)P(B=0) + P(A|B=1)P(B=1)$$

- Recall Total Expectation – A and B are random variables

$$E(A) = E(A|B=0)P(B=0) + E(A|B=1)P(B=1)$$

- Gini Impurity can be interpreted similarly

Expected Value of G(S) when split on outlook

$$G(S)_{Outlook} =$$

$$G(S|Outlook = Sunny)P(Outlook = Sunny) +$$
$$G(S|Outlook = Overcast)P(Outlook = Overcast) +$$
$$G(S|Outlook = Rainy)P(Outlook = Rainy)$$

Gini Impurity for Outlook split

Training examples: **9 yes / 5 no**

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

- Expected Value of Outlook split

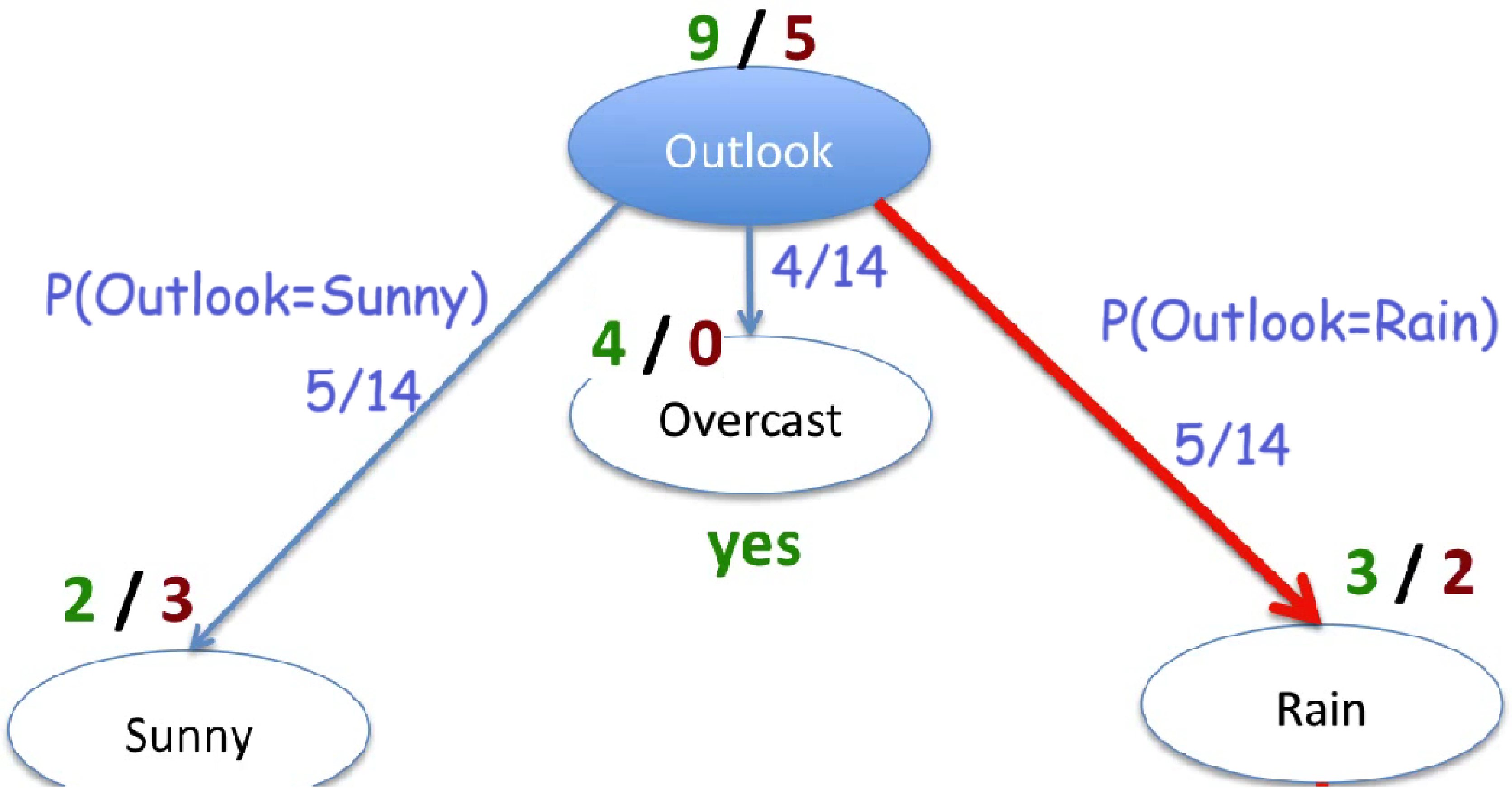
$$G(S)_{Outlook} =$$

$$G(S|O = Sunny)P(O = Sunny)+$$
$$G(S|O = Overcast)P(O = Overcast)+$$
$$G(S|O = Rainy)P(O = Rainy)$$

Day	Outlook	Humidity	Wind	Play
D1	Sunny	High	Weak	No
D2	Sunny	High	Strong	No
D8	Sunny	High	Weak	No
D9	Sunny	Normal	Weak	Yes
D11	Sunny	Normal	Strong	Yes

Training examples: **9 yes / 5 no**

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



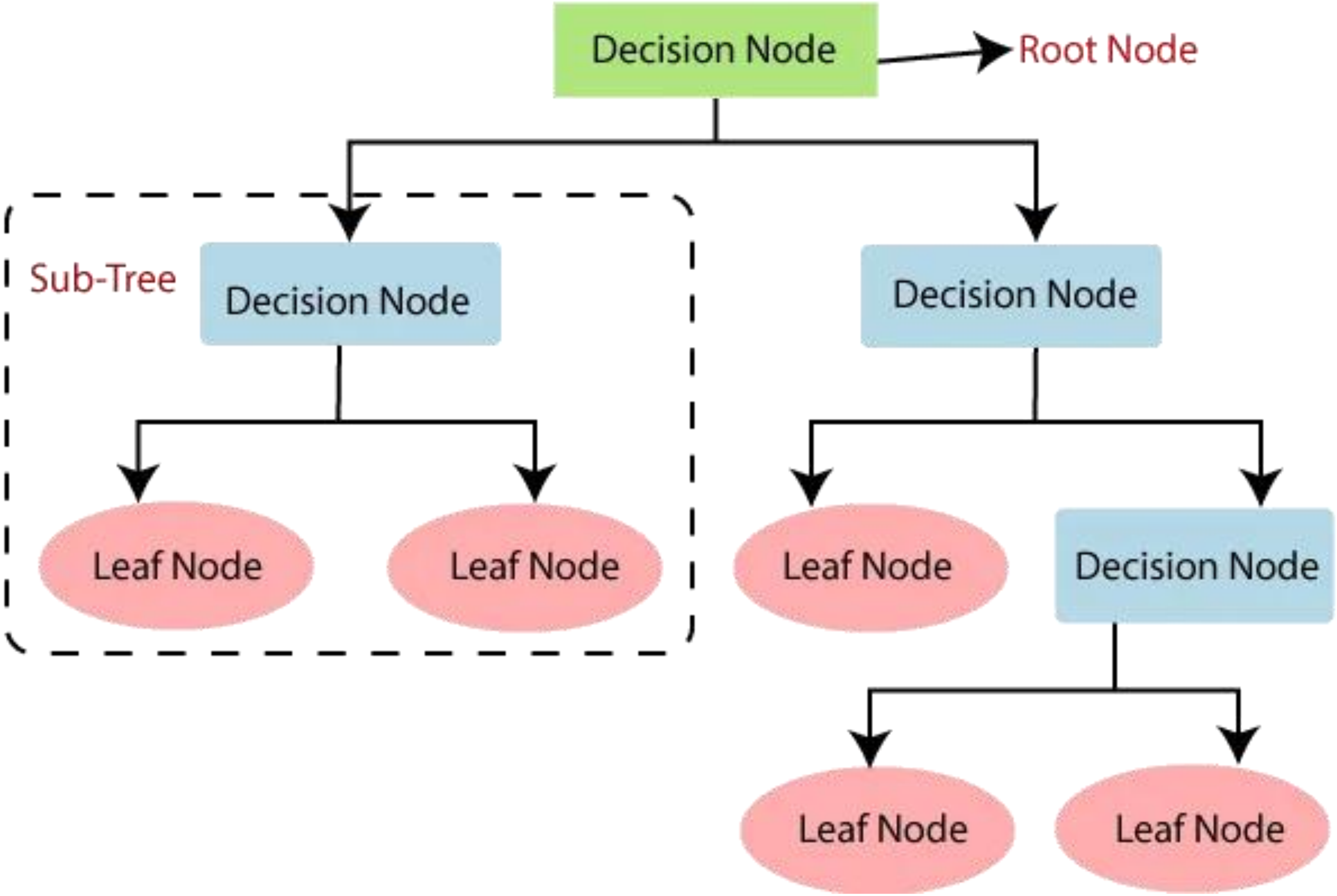
$$G(S|O = Sunny) = 1 - ((2/5)^2 + (3/5)^2)$$
$$= 1 - (0.16 + 0.36) = 0.48$$

$$G(S|O = Overcast) = 1 - ((4/4)^2 + (0/4)^2)$$
$$= 1 - (1 + 0) = 0$$

$$G(S|O = Rain) = 1 - ((2/5)^2 + (3/5)^2)$$
$$= 1 - (0.16 + 0.36) = 0.48$$

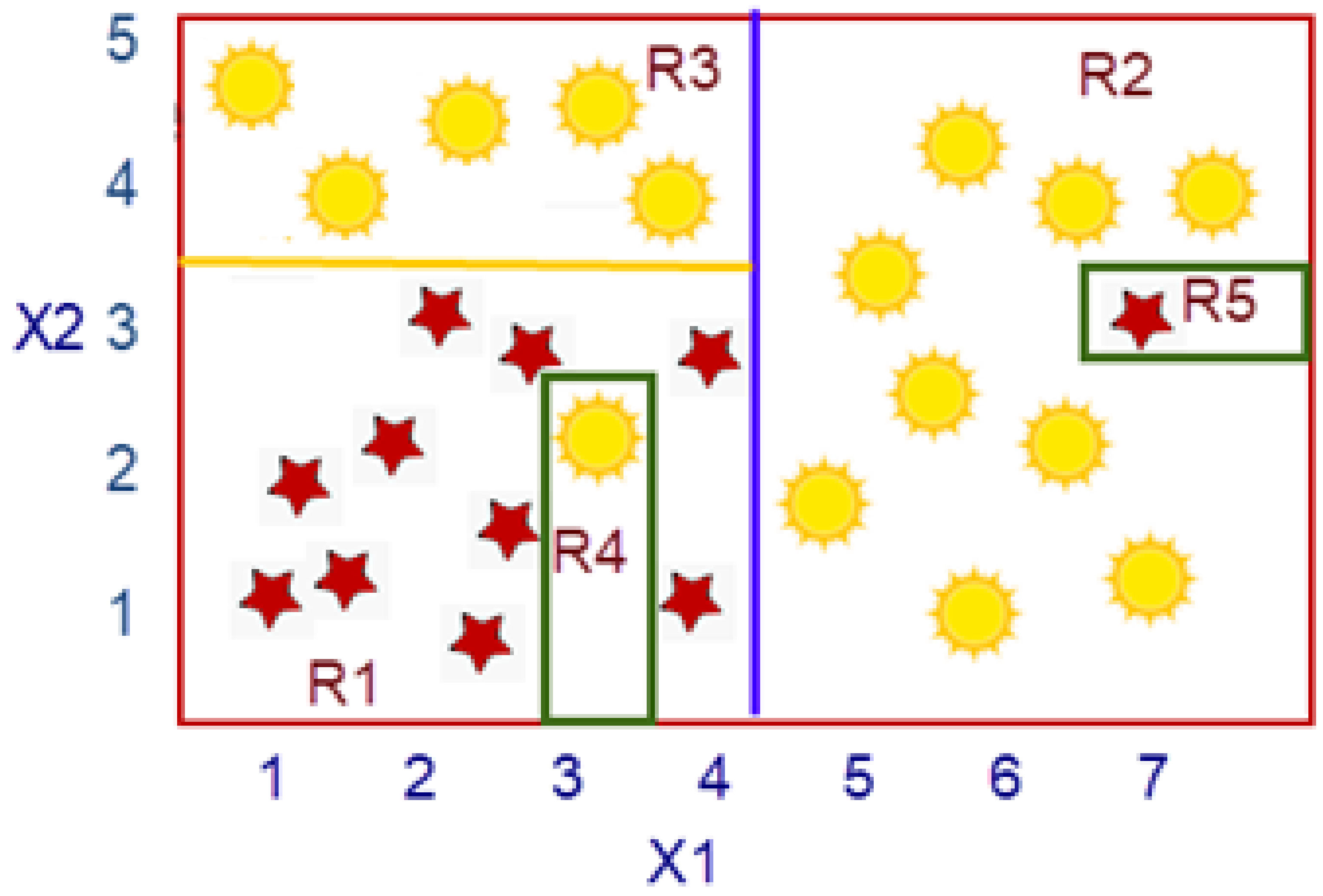
$$G(S)_{Outlook} = 0.48 * 5/14 + 0 * 4/14 + 0.48 * 5/14$$

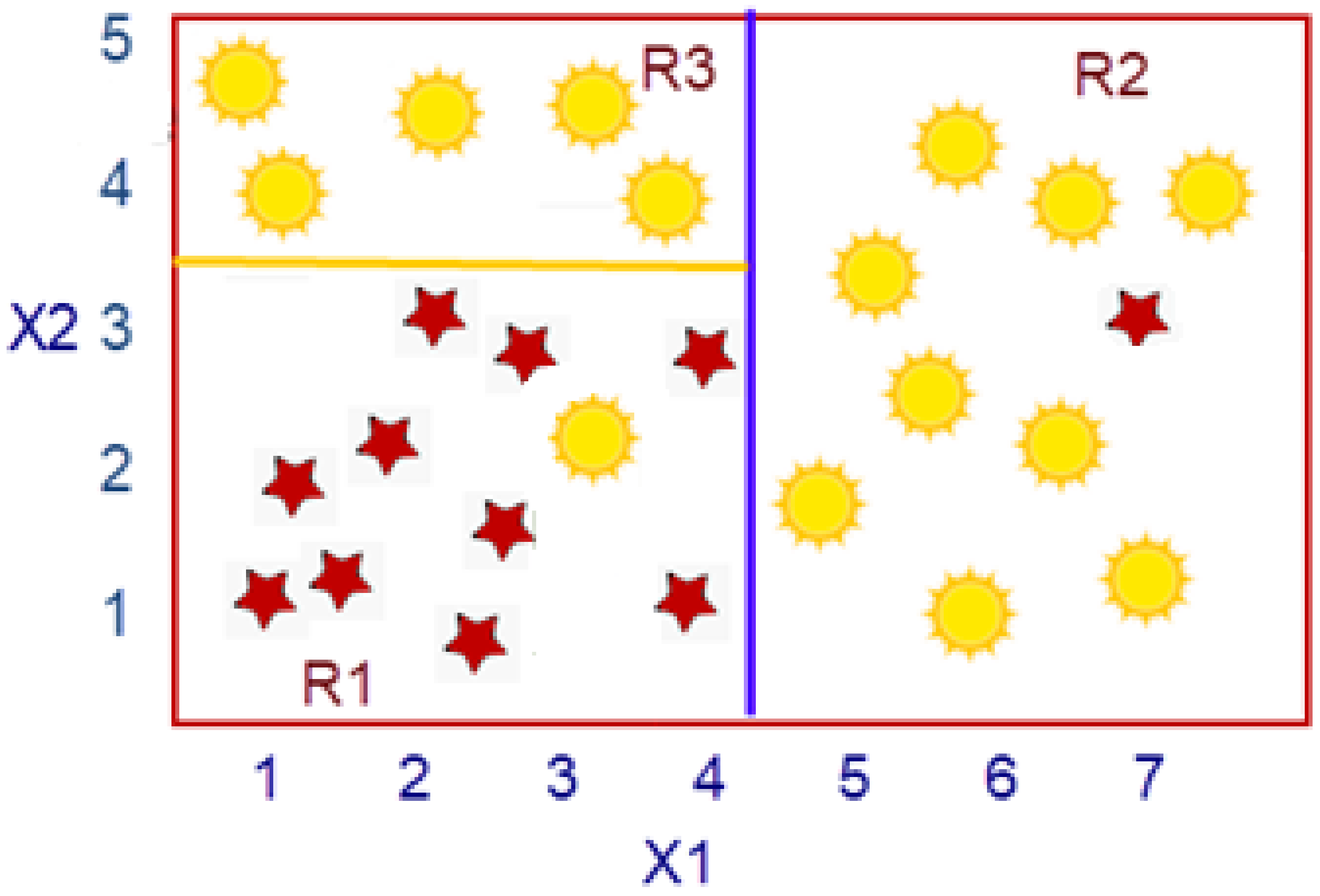
Decision Tree overfitting


Decision Node
Root Node
Sub-Tree
Decision Node
Decision Node
Leaf Node
Leaf Node
Leaf Node
Decision Node
Leaf Node
Leaf Node

How long should split be done?

- Split till all entries in a node are pure.
  - 0 Gini impurity
  - Or stop early?





Measuring Overfitting

- Low training error, falling & increasing validation error

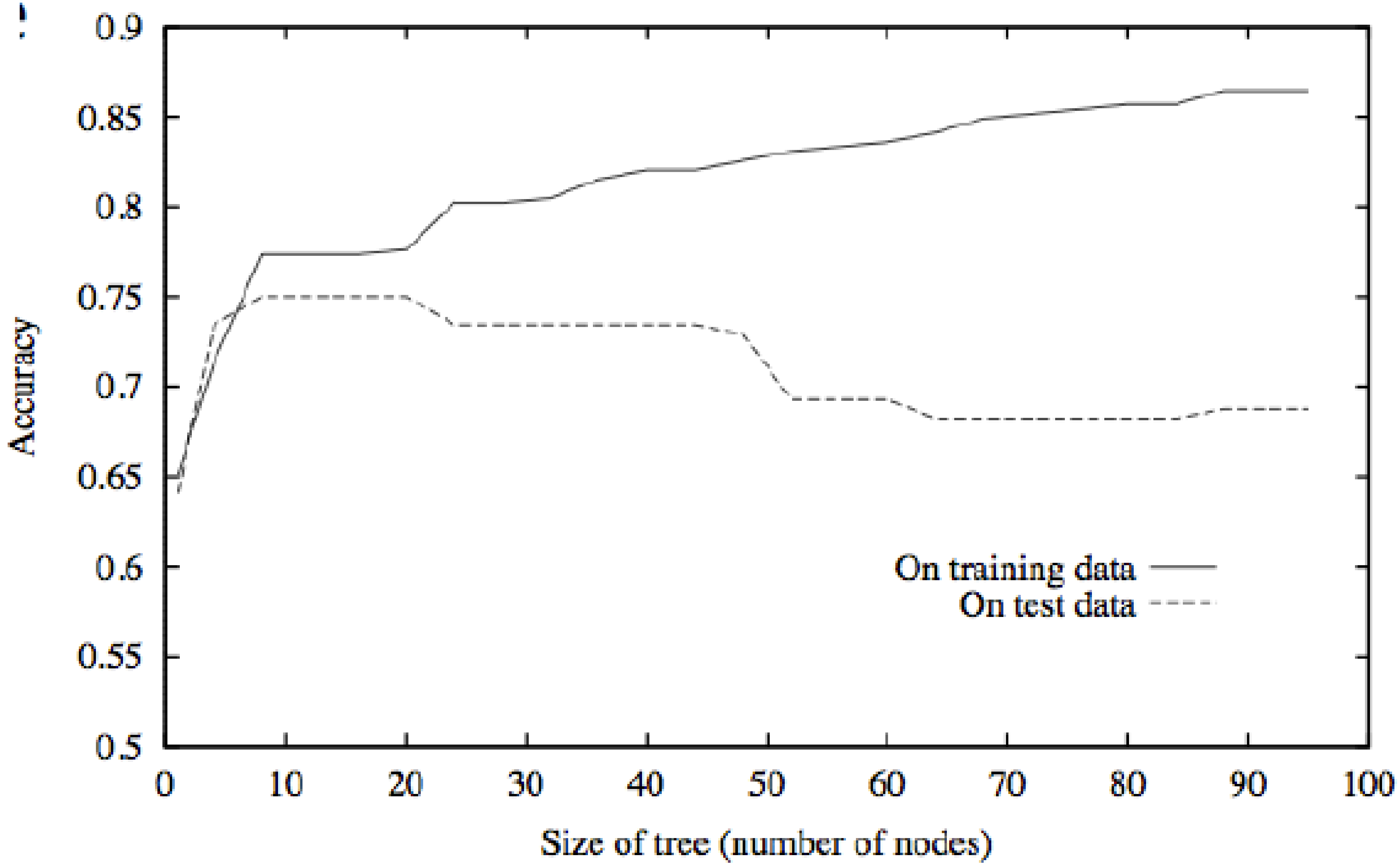


Figure credit: Tom Mitchell, 1997

Pre-pruning

- Done before training a decision tree
- Control Max-Depth of the tree
- Min samples required to split at a node
- Min samples that should be present at a leaf node

Post pruning

- Done after the Decision Tree is overfit
- Reduced Error Pruning
  - Remove a node and the subtree.
  - Check if the resulting tree performs better on validation set
  - If yes, the retain the pruning, else abstain from pruning
- Rule based pruning

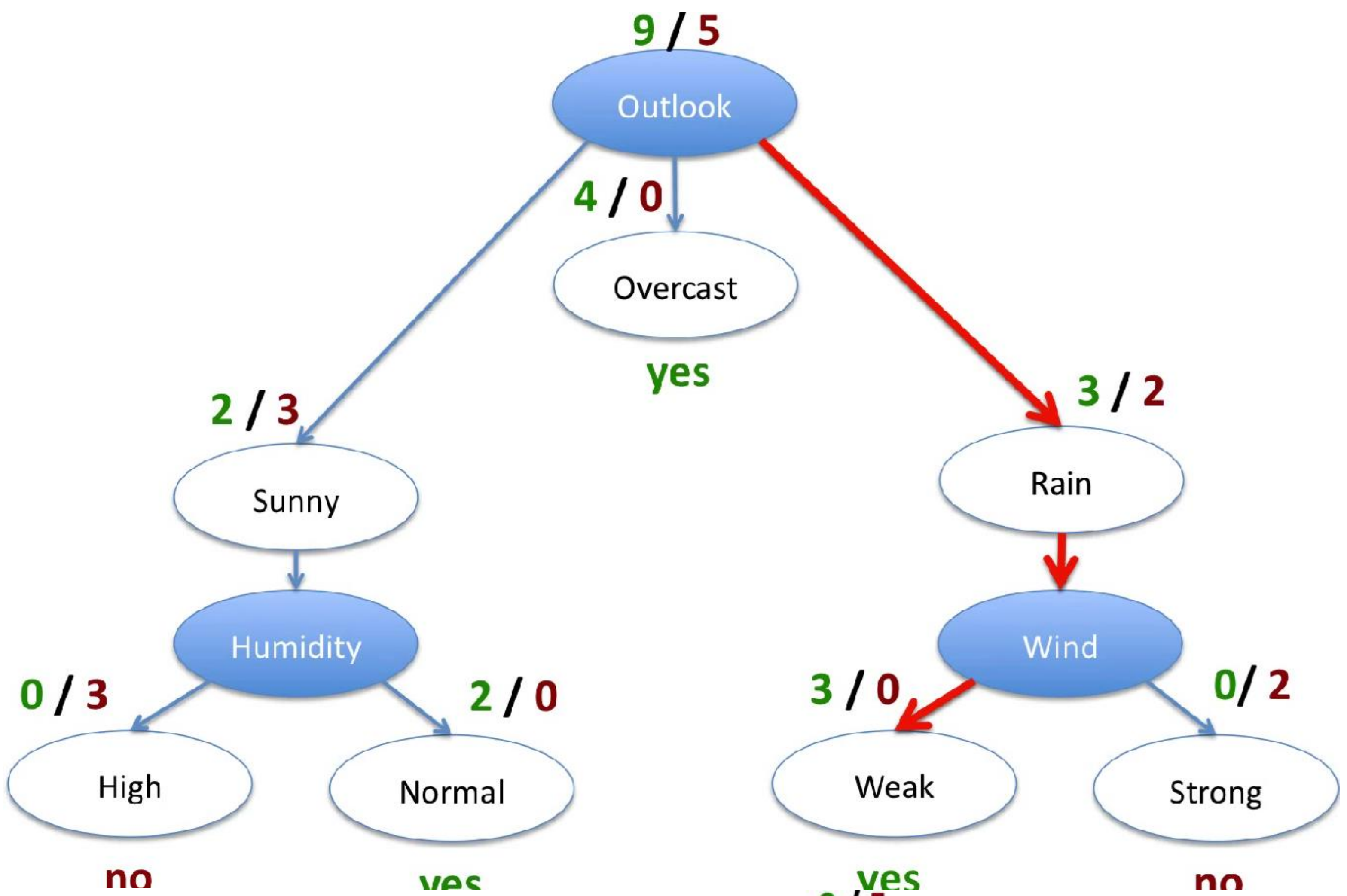
Post Pruning - Cost Complexity Pruning

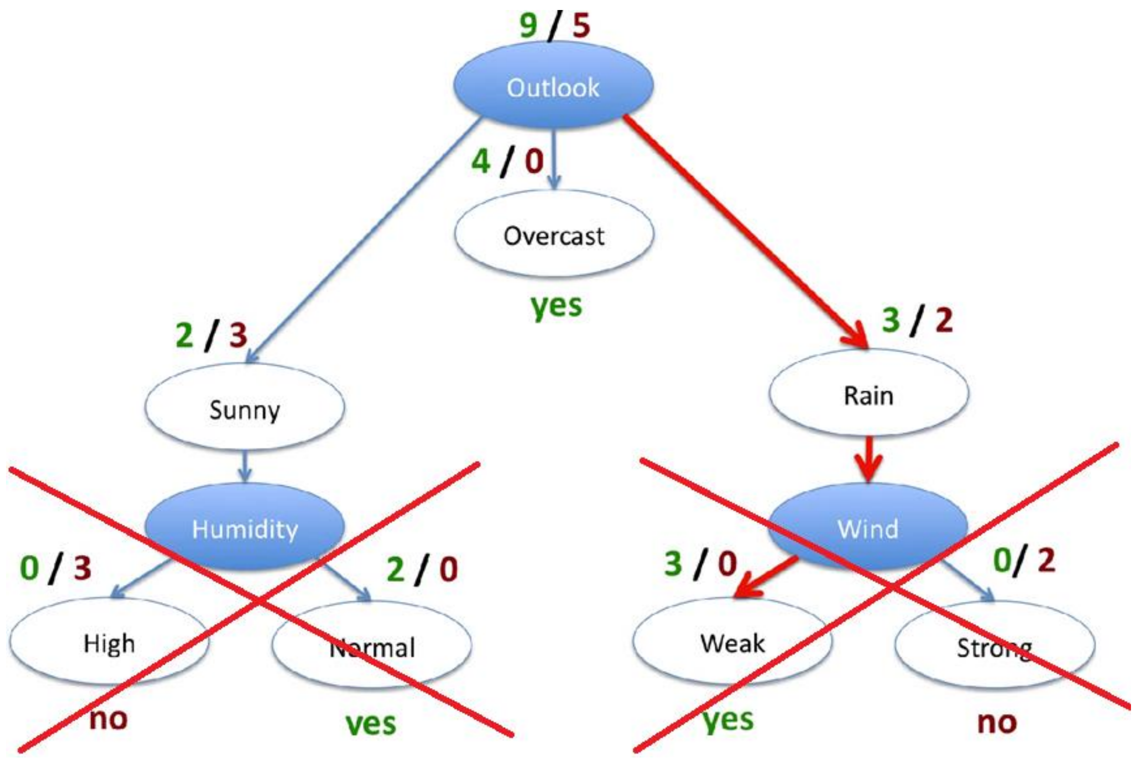
- Total Loss of a tree with ALL data

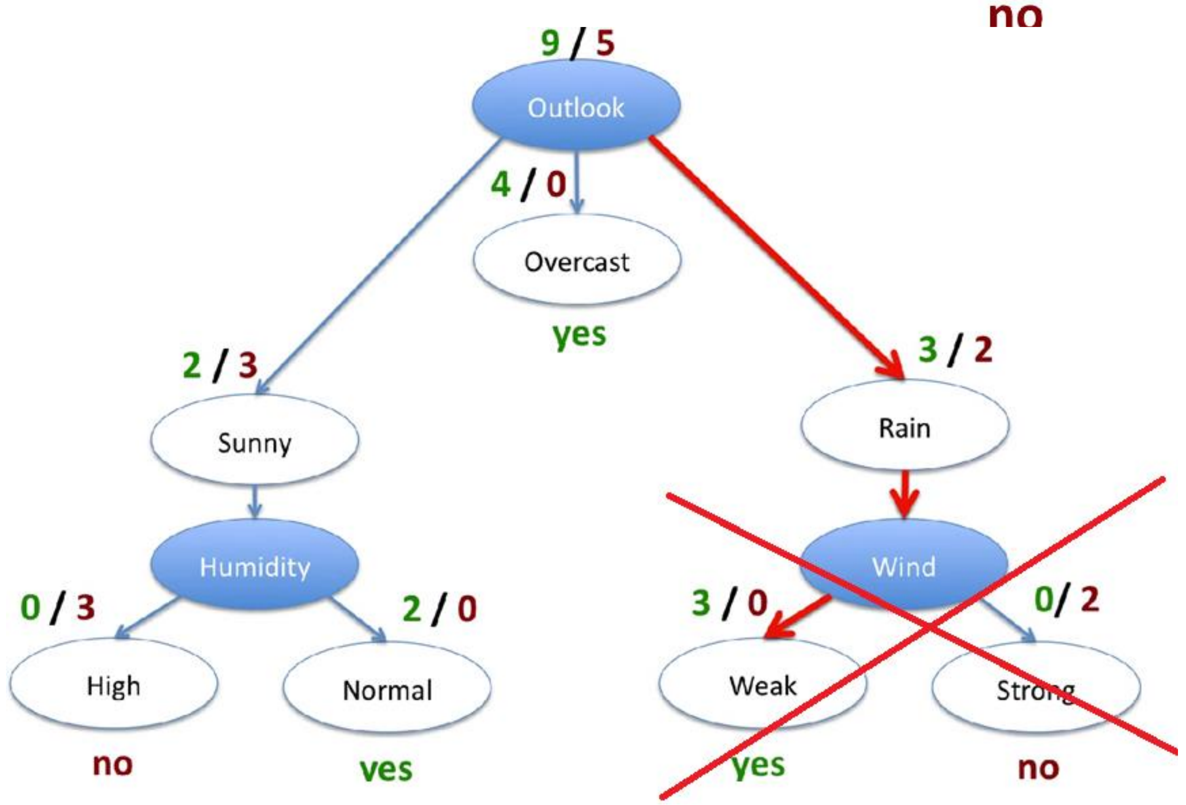
$$\mathcal{L} = \sum_{i \in features} G(S)_{feature_i}$$

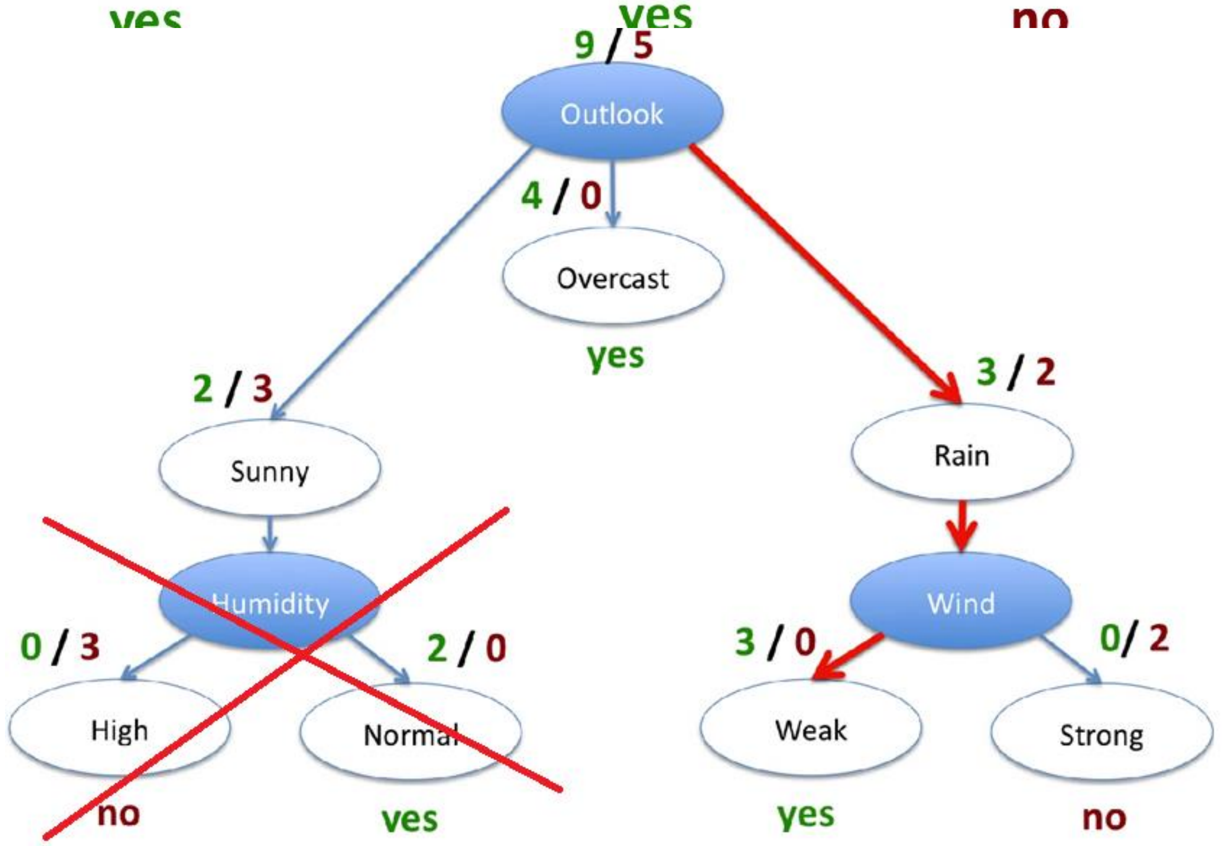
- Total loss after adding penalty

$$\mathcal{L} = \sum_{i \in features} G(S)_{feature_i} + \alpha|T|$$









Better methods than post pruning

- Post pruning is of theoretical interest these days
- Overfitting mitigated with RandomForest instead of pruning


QUESTIONS