Machine learning

Presented by : Dr. Hanaa Bayomi

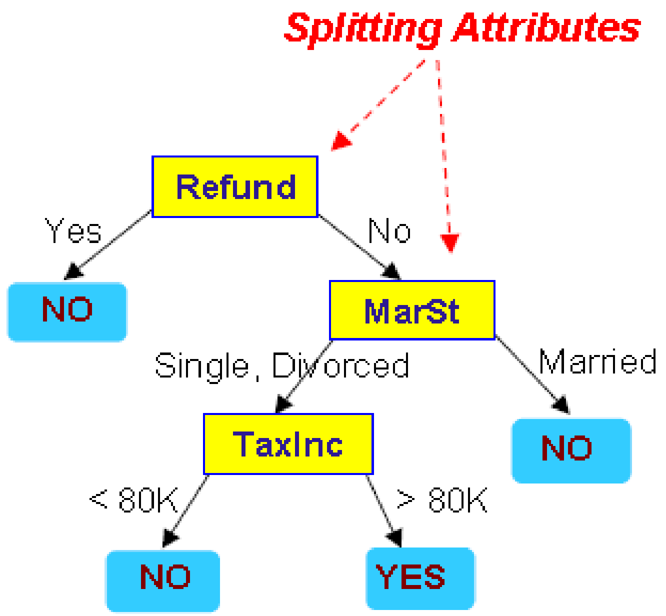
Lecture 6: Decision tree

DECISION TREE: EXAMPLE

categorical categorical continuous class

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Training Data

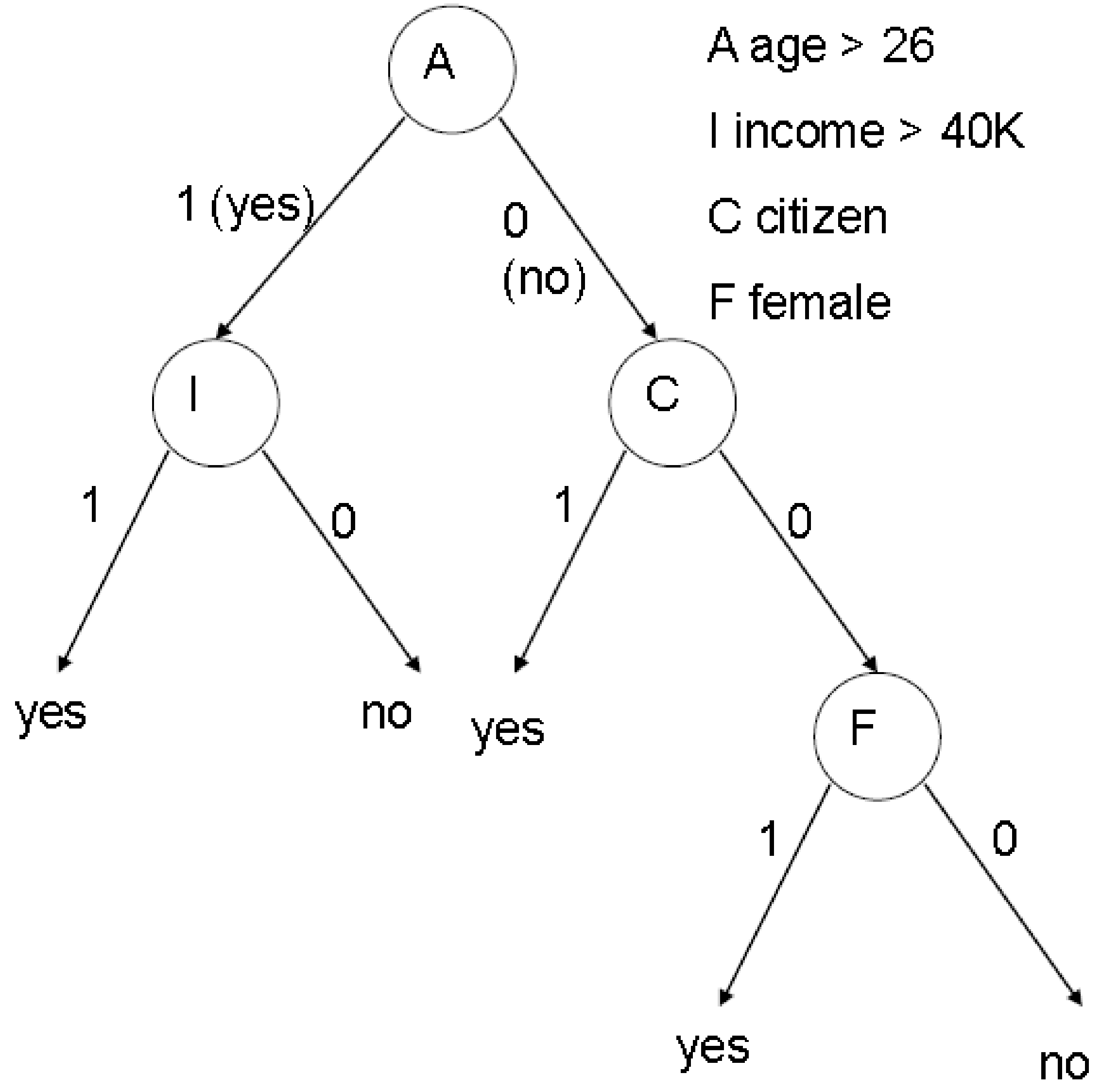


Model: Decision Tree

- There can be many different trees that all work equally well!

Structure of a decision tree

- Internal nodes correspond to attributes (features)
- Leafs correspond to classification outcome
- edges denote assignment



Predict if John will play tennis

- Hard to guess
- Divide & conquer:
  - split into subsets
  - are they pure? (all yes or all no)
  - if yes: stop
  - if not: repeat
- See which subset new data falls into

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	?

9 yes / 5 no

Outlook

Overcast

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

Sunny

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

Rain

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

Outlook

Overcast

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

Sunny

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

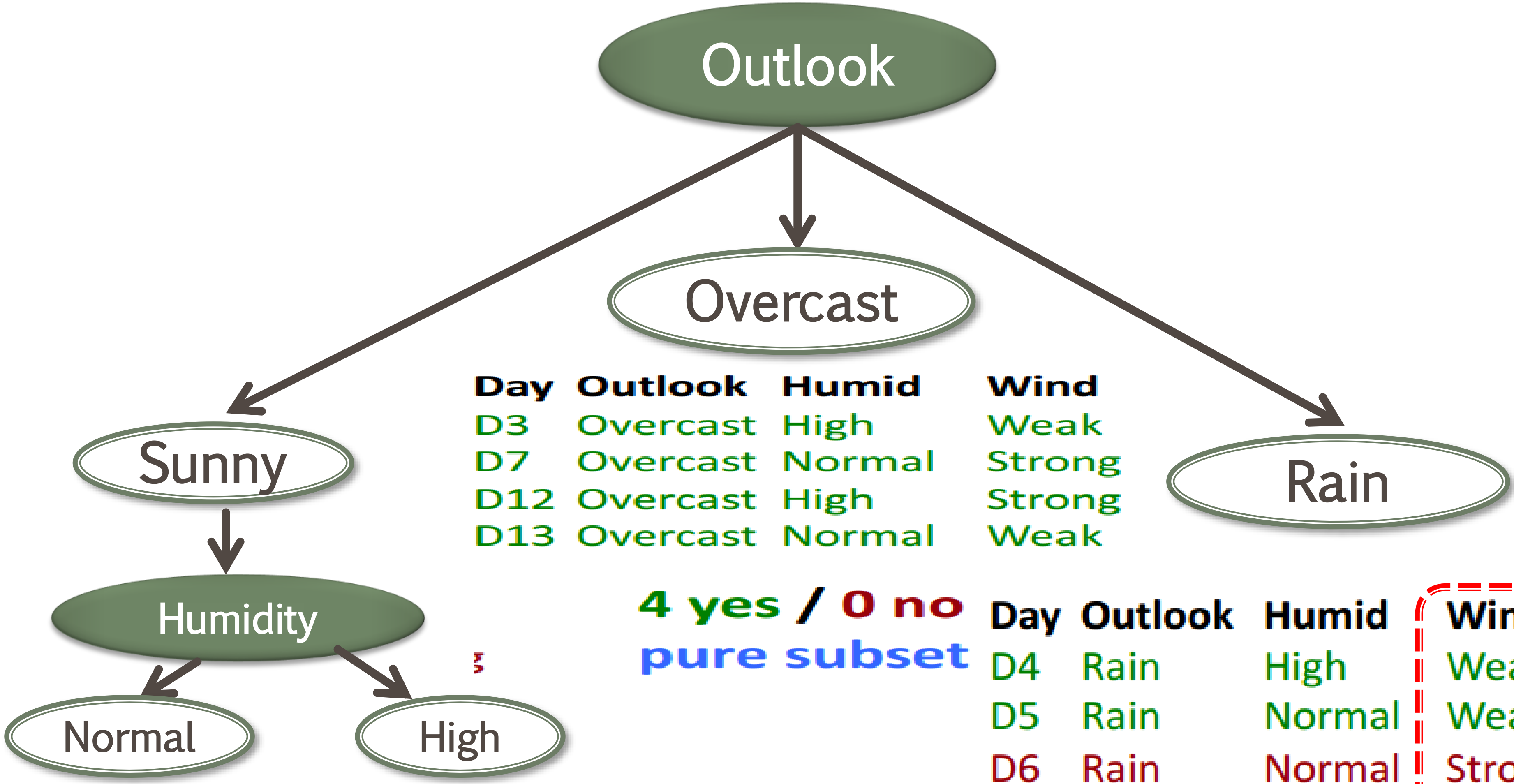
2 yes / 3 no
split further

Rain

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

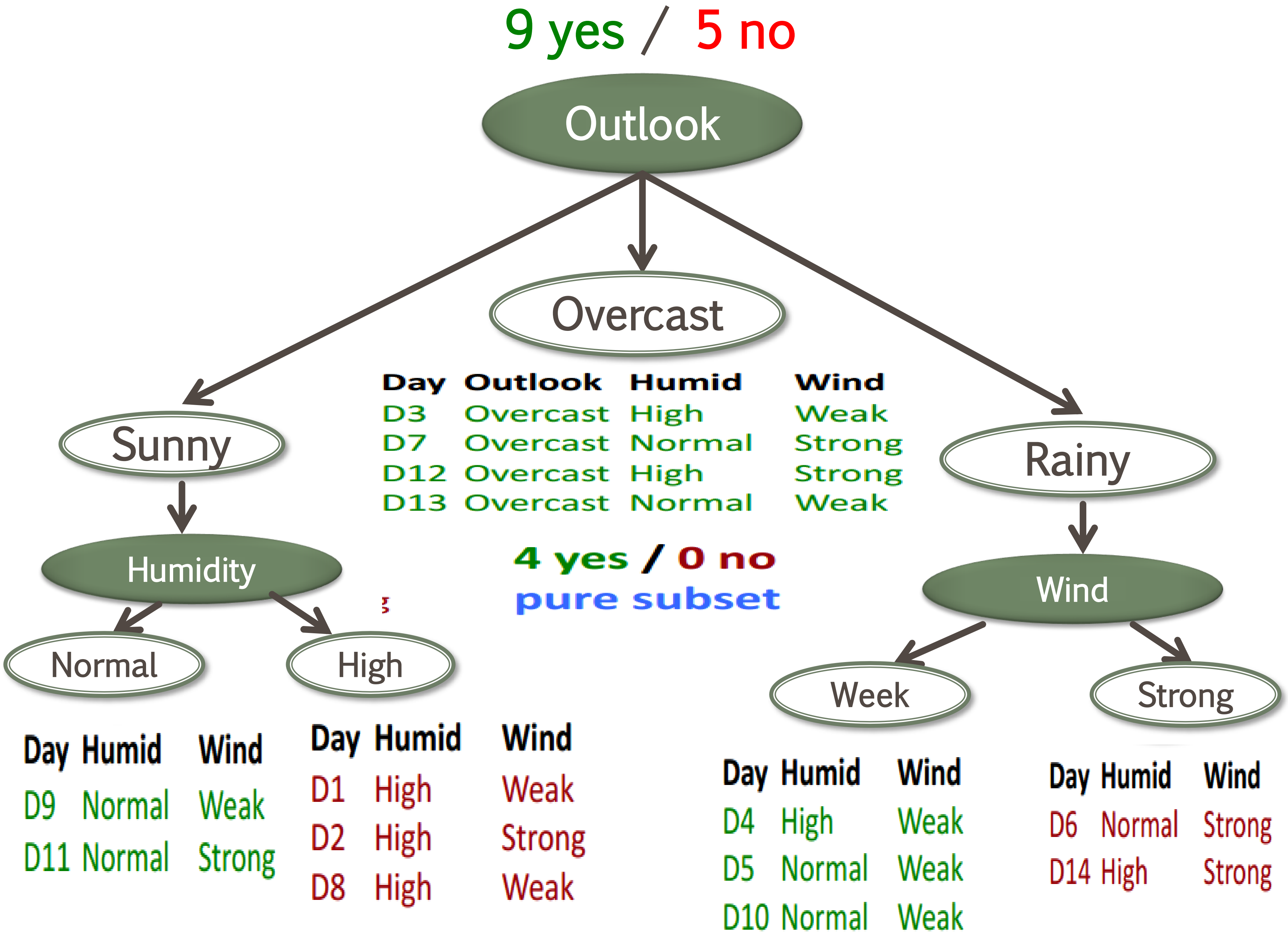
4 yes / 0 no
pure subset

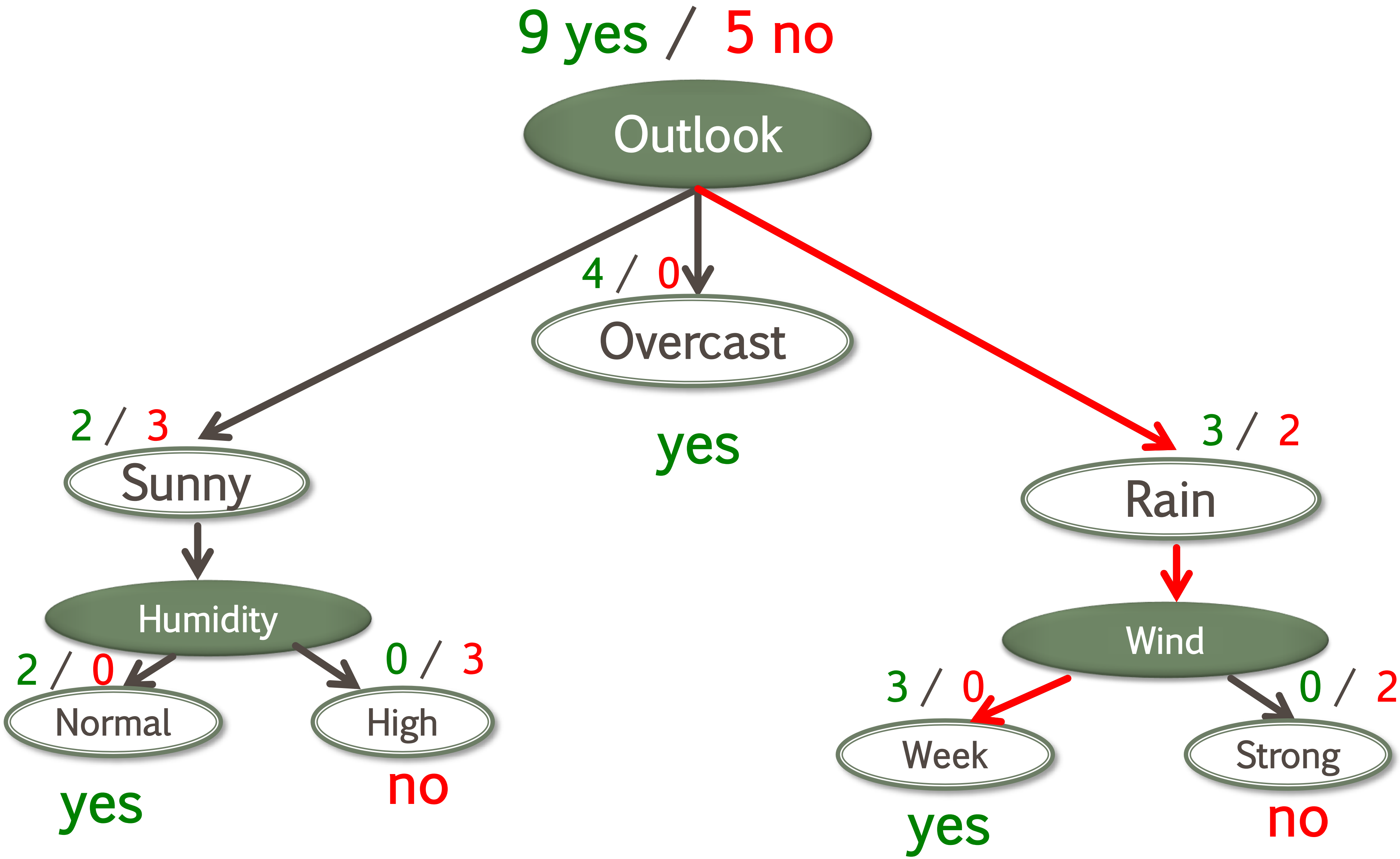
Day	Humid	Wind
D9	Normal	Weak
D11	Normal	Strong

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

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
Outlook
Overcast
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
Sunny
Humidity
Normal
High
Day Humid Wind
D9 Normal Weak
D11 Normal Strong
Day Humid Wind
D1 High Weak
D2 High Strong
D8 High Weak
Rainy
Wind
Week
Strong
Day Humid Wind
D4 High Weak
D5 Normal Weak
D10 Normal Weak
Day Humid Wind
D6 Normal Strong
D14 High Strong



New data:

Day	**Outlook**	**Humid**	**Wind**	
D15	Rain	High	Weak	$\boxed{\text{yes}}$

ID3 Algorithm

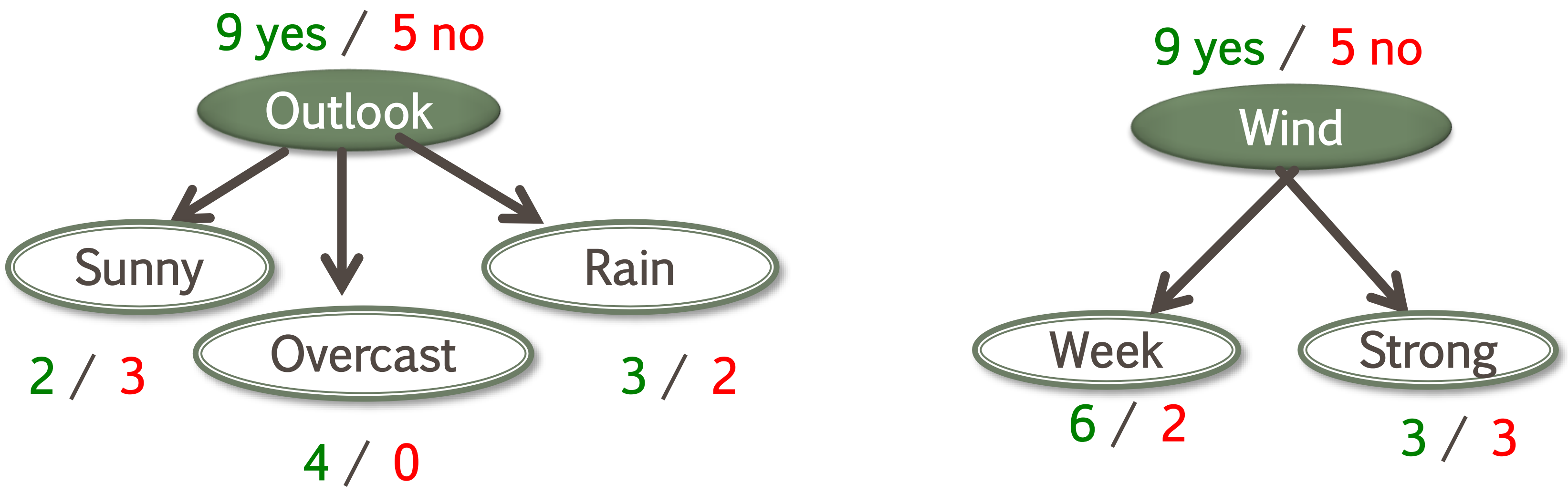
- Split (node, {examples}):
  1. A ← the best attribute for splitting the {examples}
  2. Decision attribute for this node ← A
  3. For each value of A, create new child node
  4. Split training {examples} to child nodes
  5. If examples perfectly classified: STOP
     else: iterate over new child nodes

     Split (child_node, {subset of examples})
- Ross Quinlan (ID3: 1986), (C4.5: 1993)
- Breimanetal (CaRT: 1984) from statistics

Identifying ‘bestAttribute’

- There are many possible ways to select the best attribute for a given set.
- We will discuss one possible way which is based on information theory and generalizes well to non binary variables

Which attribute to split on?



- Want to measure "purity" of the split
  - more certain about Yes/No after the split
    - pure set (4 yes / 0 no) => completely certain (100%)
    - impure (3 yes / 3 no) => completely uncertain (50%)

Entropy

- Entropy: $H(S) = - p_{(+)} \log_2 p_{(+)} - p_{(-)} \log_2 p_{(-)}$ bits
  - S … subset of training examples
  - $p_{(+)}$ / $p_{(-)}$ … % of positive / negative examples in S

• The Entropy is 1 when the collection contains *an equal number of positive and negative examples.*

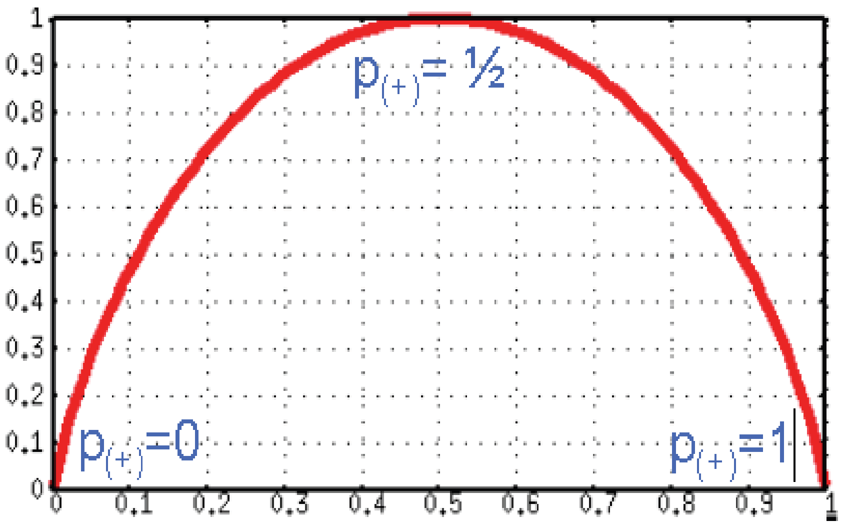
• The Entropy is 0 if all members of S belong to *the same class*

- impure (3 yes / 3 no):

$$H(S) = -\frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2\frac{3}{6} = 1 \text{ bits}$$

- pure set (4 yes / 0 no):

$$H(S) = -\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4} = 0 \text{ bits}$$



INFORMATION GAIN

- We want to determine which attribute in a given set of training feature vectors is most useful for discriminating between the classes to be learned.

- Information gain tells us how important a given attribute of the feature vectors is.

- We will use it to decide the ordering of attributes in the nodes of a decision tree.

INFORMATION GAIN

Information Gain = entropy(parent) – [average entropy(children)]

- Want many items in pure sets
- Expected drop in entropy after split:

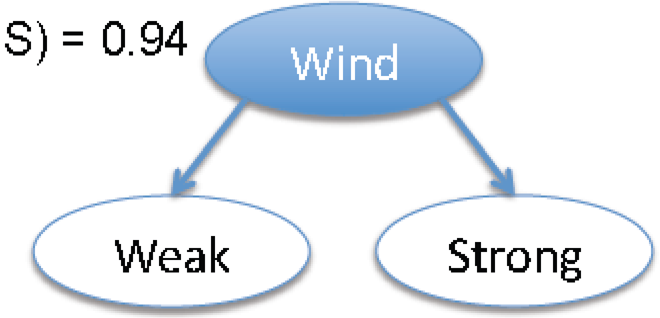
$$Gain(S, A) = H(S) - \sum_{V \in Values(A)} \frac{|S_V|}{|S|} H(S_V)$$

V … possible values of A
S … set of examples {X}
S_V … subset where X_A = V

- Mutual Information
  - between attribute A and class labels of S

$-\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14}$ 9 yes / 5 no

H(S) = 0.94



6 yes / 2 no

$-\frac{6}{8}\log_2\frac{6}{8} - \frac{2}{8}\log_2\frac{2}{8}$

$H(S_{weak}) = 0.81$

3 yes / 3 no

$-\frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2\frac{3}{6}$

$H(S_{strong}) = 1.0$

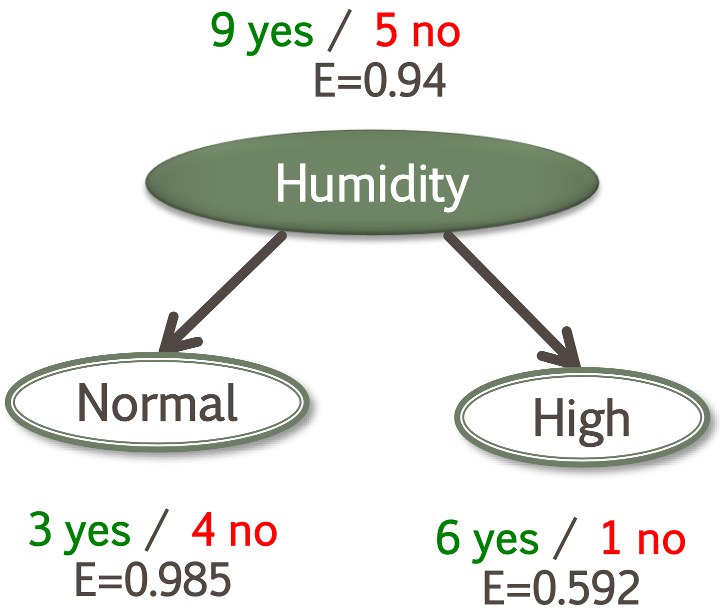
Gain (S, Wind)

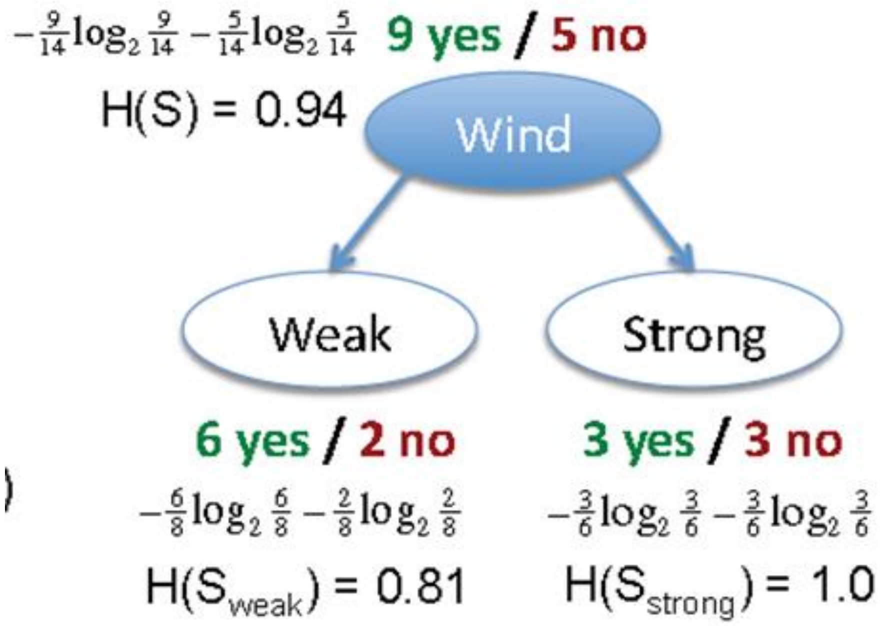
$= H(S) - {}^{8}/_{14}\, H(S_{weak}) - {}^{6}/_{14}\, H(S_{strong})$

$= 0.94 - {}^{8}/_{14} * 0.81 - {}^{6}/_{14} * 1.0$

$= 0.049$

Which attribute is the best classifier? Example





Gain (S,Humidity)=0.94-(7/14)*0.985-(7/14)*0.592
=0.151

Gain (S,Wind)=0.94-(8/14)*0.81-(6/14)*1
=0.048

- ***Humidity provides greater information gain than Wind, relative to the target classification.***

E stands for entropy and

S ***the original collection of examples. Given an initial collection S of 9*** positive and 5 negative examples,.

ILLUSTRATIVE EXAMPLE

Predict if John will play tennis

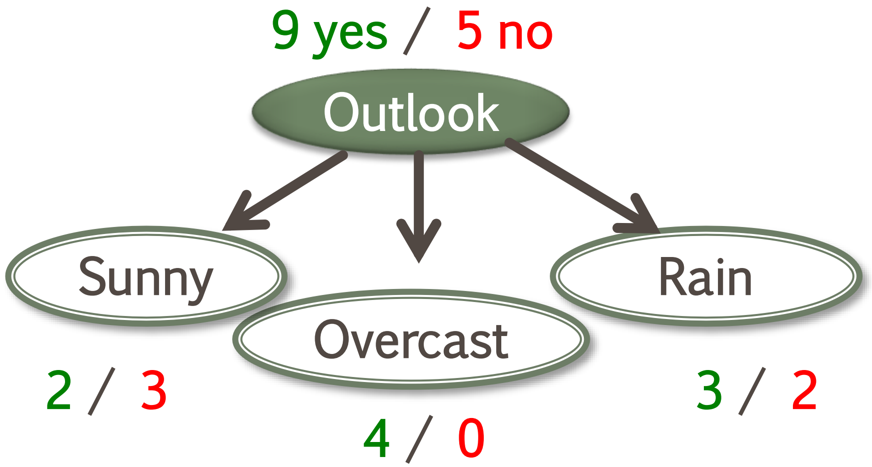
$$Gain(S, Outlook) = 0.246$$

$$Gain(S, Humidity) = 0.151$$

$$Gain(S, Wind) = 0.048$$

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



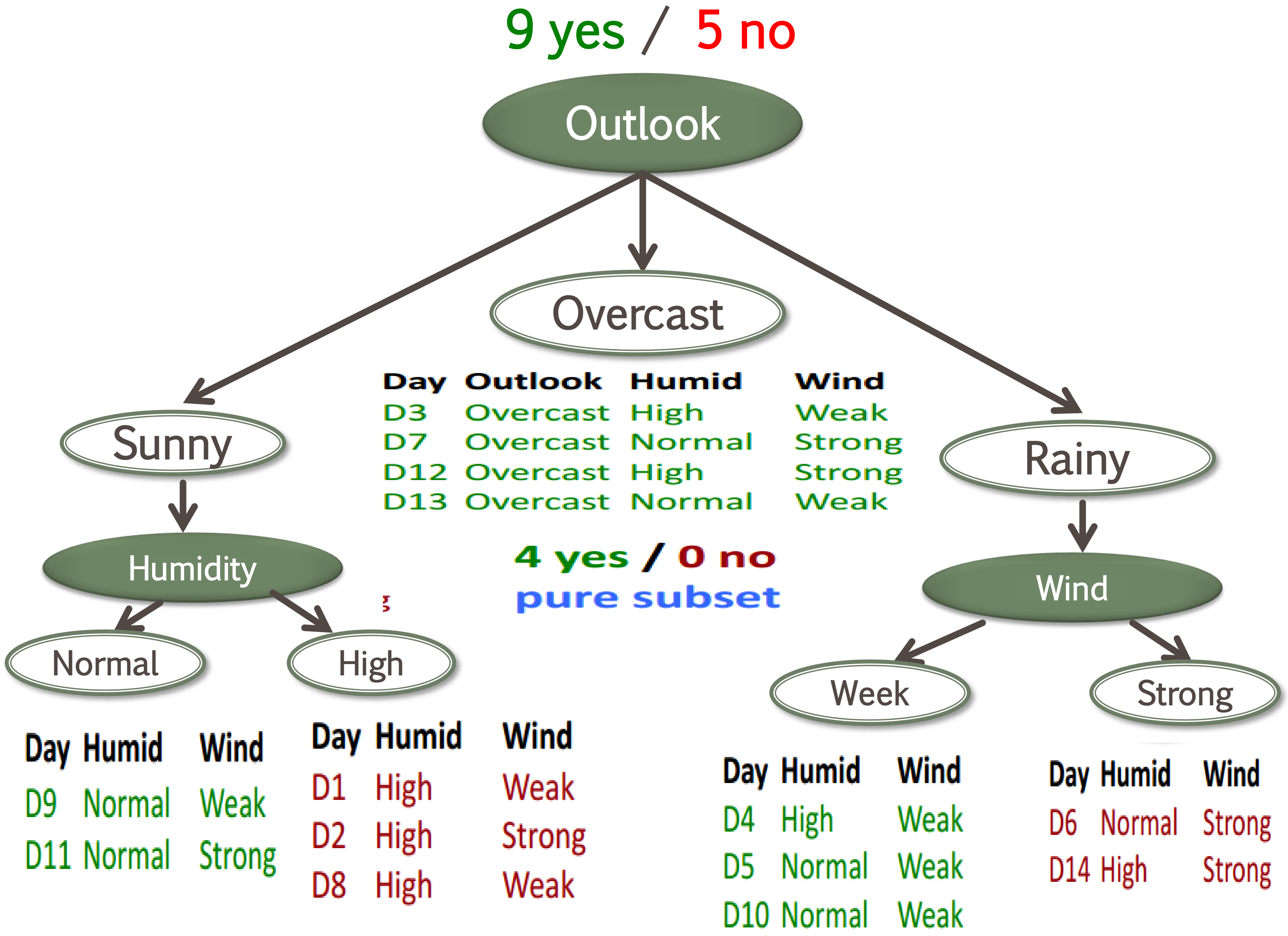
$$H(S, outlook) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0.94$$

$$H(S_{sunny}) = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5} = 0.97$$

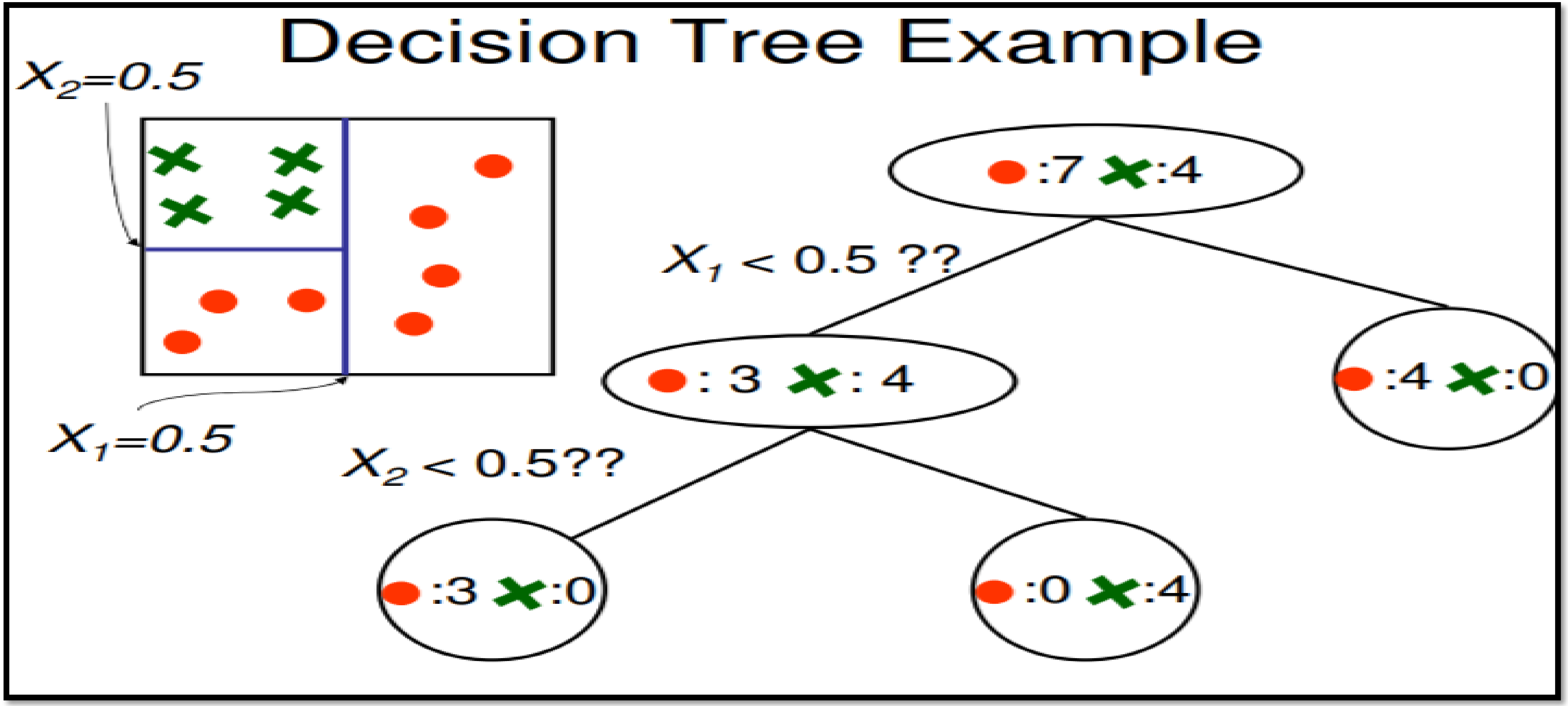
$$H(S_{overcast}) = -\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4} = 0$$

$$H(S_{rain}) = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5} = 0.97$$

$$Gain(S, Outlook) = 0.94 - \frac{5}{14}\times 0.97 - \frac{4}{14}\times 0 - \frac{5}{14}\times 0.97 = 0.247$$

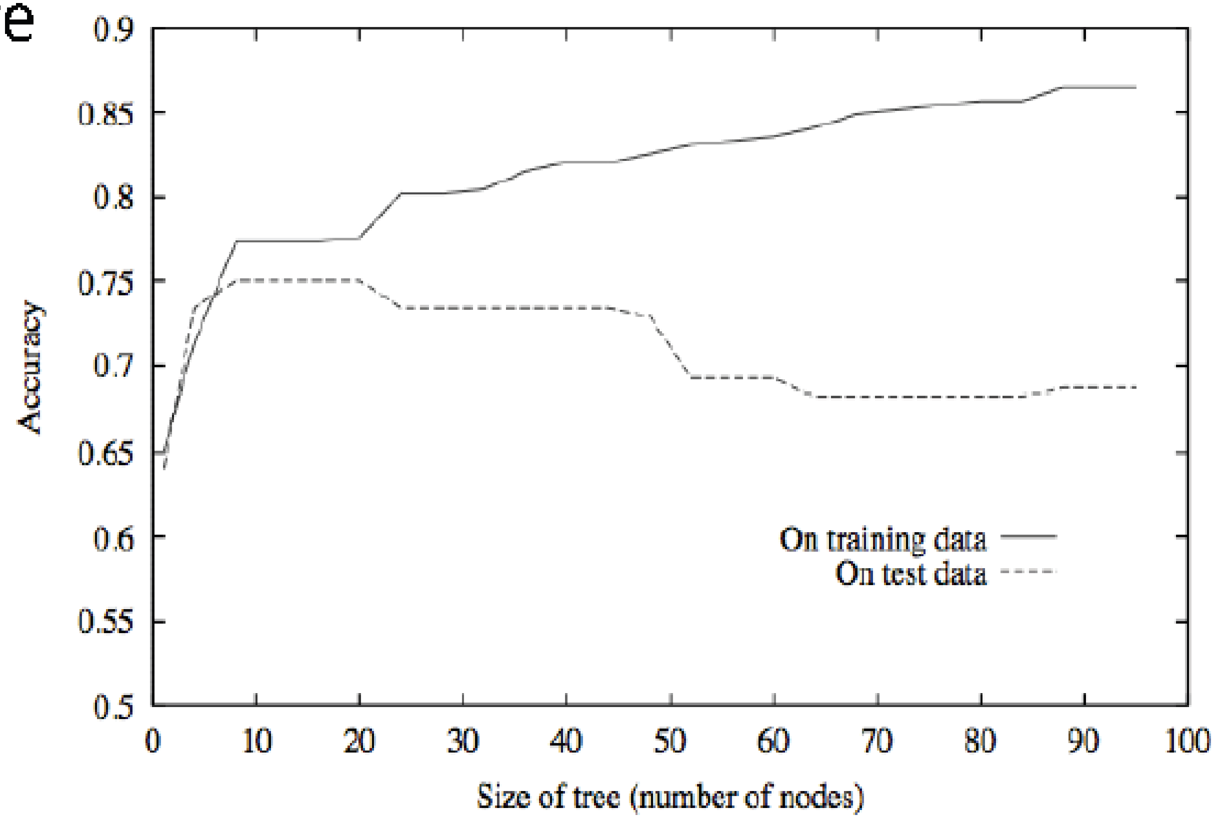

9 yes / 5 no
Outlook
Overcast
Day Outlook Humid Wind
D3 Overcast High Weak
D7 Overcast Normal Strong
D12 Overcast High Strong
D13 Overcast Normal Weak
4 yes / 0 no
pure subset
Sunny
Humidity
Normal
Day Humid Wind
D9 Normal Weak
D11 Normal Strong
High
Day Humid Wind
D1 High Weak
D2 High Strong
D8 High Weak
Rainy
Wind
Week
Day Humid Wind
D4 High Weak
D5 Normal Weak
D10 Normal Weak
Strong
Day Humid Wind
D6 Normal Strong
D14 High Strong

CONTINUOUS VALUE



Overfitting in Decision Trees

- Can always classify training examples perfectly
  - keep splitting until each node contains 1 example
  - singleton = pure
- Doesn't work on new data



How to Deal with Overfitting?

- Stop growing the tree when the data split is not statistically significant
- Grow the full tree, then prune
  - Do we really needs all the “small” leaves with perfect coverage?

Decision Tree Pre-Pruning

- Stop the algorithm before it becomes a fully-grown tree
- Typical stopping conditions for a node
  - Stop if all instances belong to the same class
  - Stop if all the feature values are the same
- More restrictive conditions
  - Stop if the number of instances is less than some use-specified threshold
  - Stop if the class distribution of instances are independent of the available features
    - Stop if expanding the current node does not improve impurity.

Decision Tree Post-Pruning

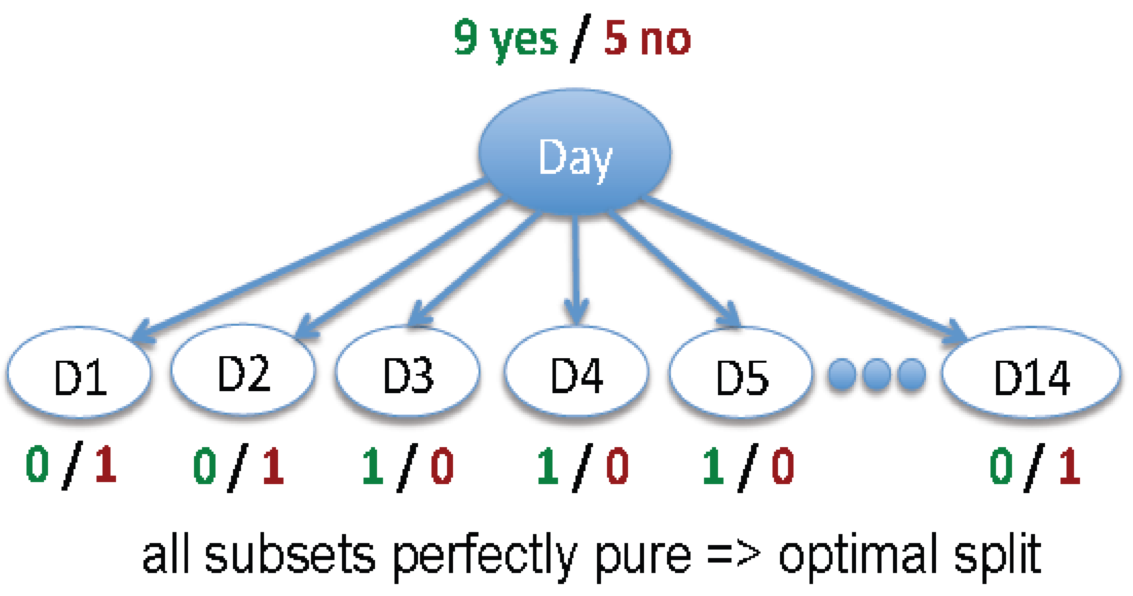
- Grow decision tree to its entirety
- Trim the nodes of the decision tree in a bottom-up fashion
- If generalization error improves after trimming, replace sub-tree by a leaf node
  - Class label of leaf node is determined from majority class of instances in the sub-tree

Decision Tree Post-Pruning

- Reduced Error Pruning
  - Split data into training and validation set
  - Remove one node at a time and evaluate the performance on the validation data
  - Remove the one that decreases the error
  - Usually produces the smallest version of a tree
  - But always requires a validation set

Problems with Information Gain

- Biased towards attributes with many values
- Won't work for new data: D15 Rain High Weak
- Use GainRatio:



$$SplitEntropy(S,A) = -\sum_{V \in Values(A)} \frac{|S_V|}{|S|} \log \frac{|S_V|}{|S|}$$

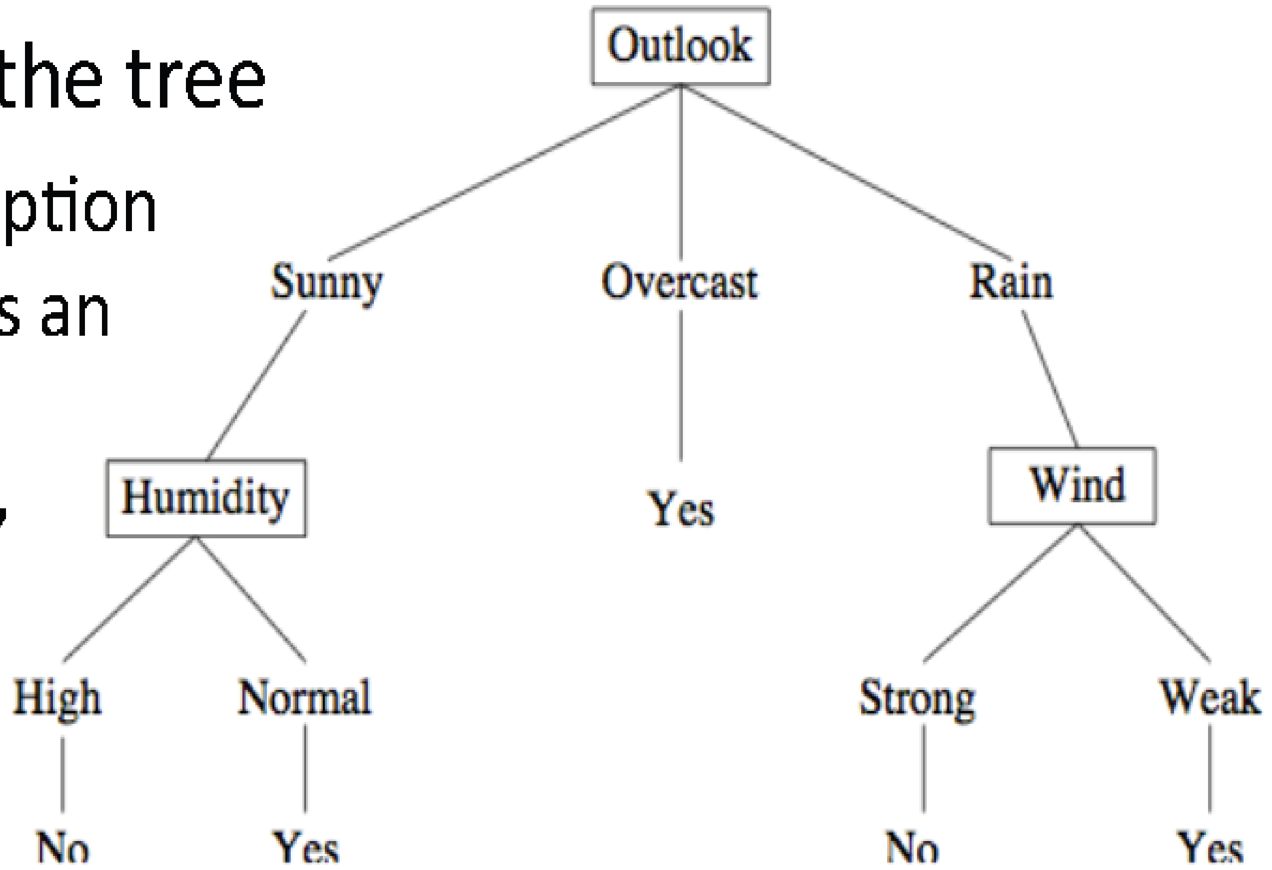
A ... candidate attribute
V ... possible values of A
S ... set of examples {X}
S_V ... subset where $X_A = V$

$$GainRatio(S,A) = \frac{Gain(S,A)}{SplitEntropy(S,A)}$$

penalizes attributes with many values

Trees are interpretable

- Read rules off the tree
  - concise description of what makes an item positive
- No “black box”
  - important for users



Rule: (Outlook = Overcast) ∨
(Outlook = Rain ∧ Wind = Weak) ∨
(Outlook = Sunny ∧ Humidity = Normal)