Lecture 13: Decision Trees

COMP90049
Introduction to Machine Learning
Semester 2, 2020

Lida Rashidi, CIS

© 2020 The University of Melbourne
Acknowledgement: Jeremy Nicholson, Tim Baldwin & Karin Verspoor

Lecture Outline

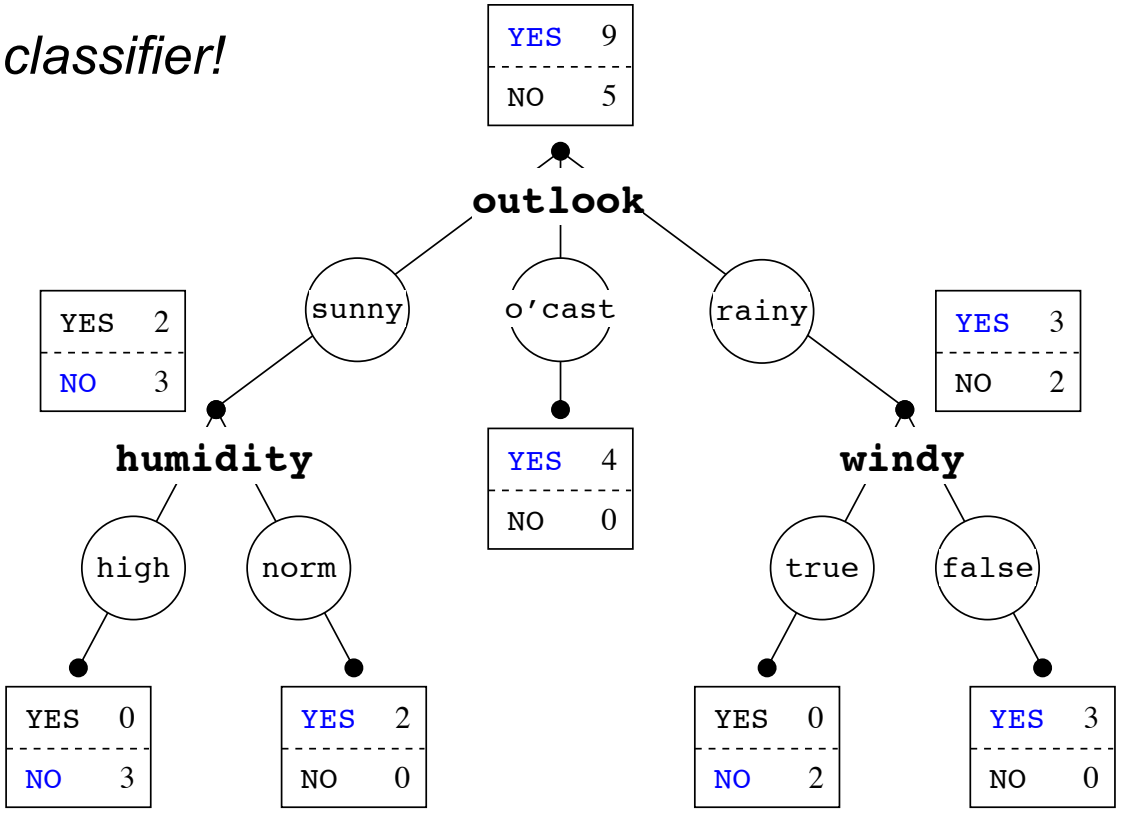
- Decision Trees
  - Introduction
- ID3 Algorithm
  - Algorithm
  - Attribute Criterion: Information Gain
  - Attribute Criterion: Gain Ratio
  - Stopping Criteria
- Discussion

Weather Nominal Dataset

	Outlook	Temperature	Humidity	Windy	Play
a	no	hot	high	FALSE	no
b	no	hot	high	TRUE	no
c	mild	hot	high	FALSE	yes
d	severe	mild	high	FALSE	yes
e	severe	cool	normal	FALSE	yes
f	severe	cool	normal	TRUE	no
g	mild	cool	normal	TRUE	yes
h	no	mild	high	FALSE	no
i	no	cool	normal	FALSE	yes
j	severe	mild	normal	FALSE	yes
k	no	mild	normal	TRUE	yes
l	mild	mild	high	TRUE	yes
m	mild	hot	normal	FALSE	yes
n	severe	mild	high	TRUE	no

Rule based classification

We need labelled data to build a classifier!

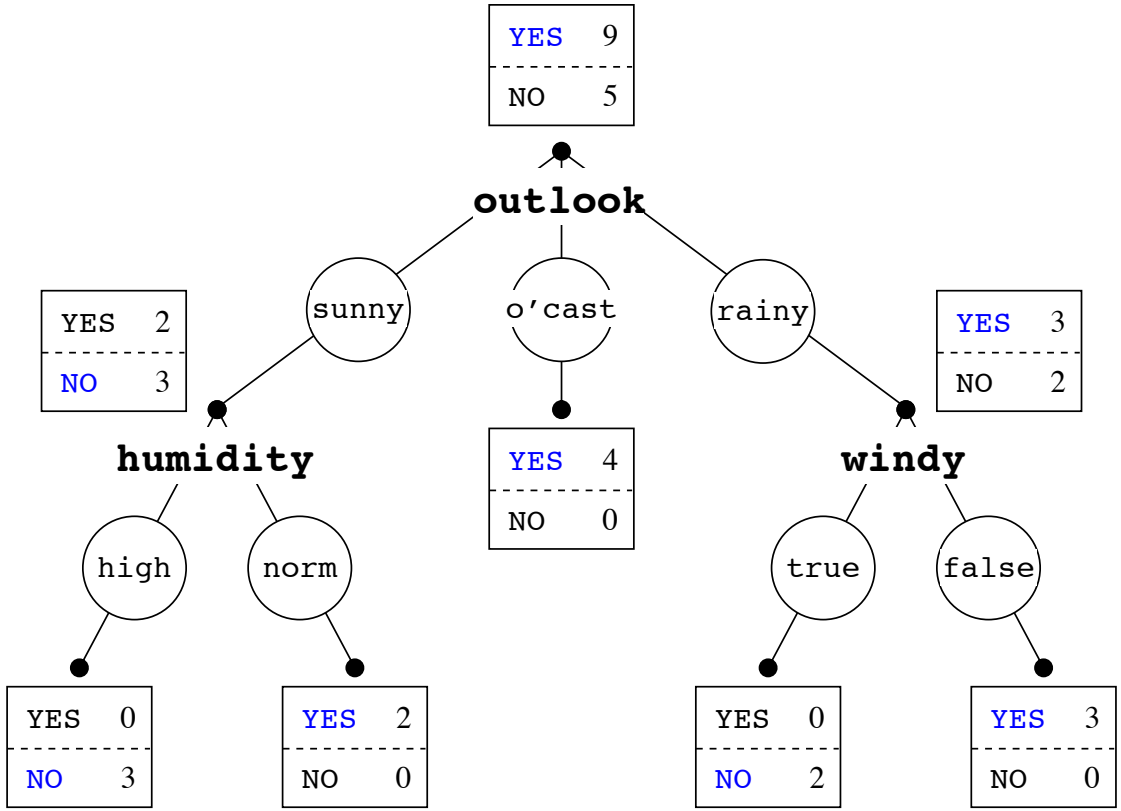


- A flow-chart-like tree structure:
- Internal node denotes a test on an attribute
- Branch represents an outcome of the test
- Leaf nodes represent class labels or class distribution

Rule based classification

One good approach is:

- Construct a decision tree
- Extract one rule for each leaf node



Rule1: if (outlook = o'cast) - > Yes

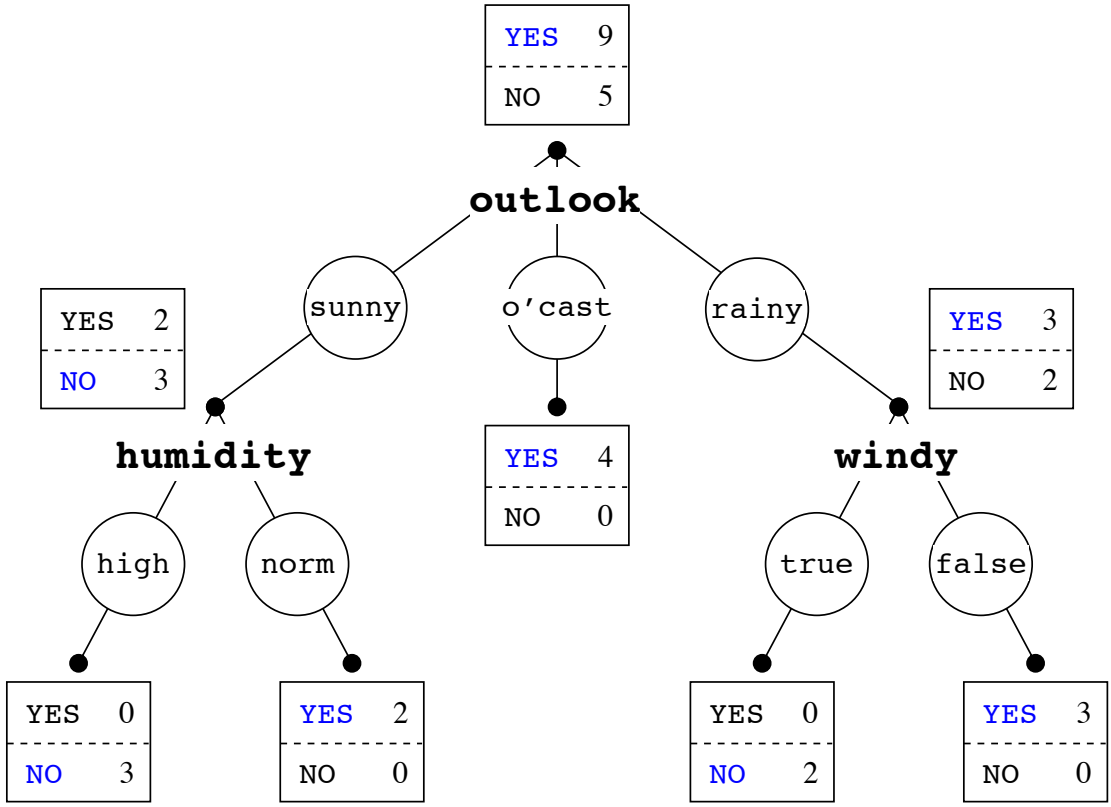
Rule2: if (outlook = sunny) and (humidity = normal) - > Yes

Rule3: if (outlook = rainy) and (windy = False) - > Yes

Decision Trees

Issues:

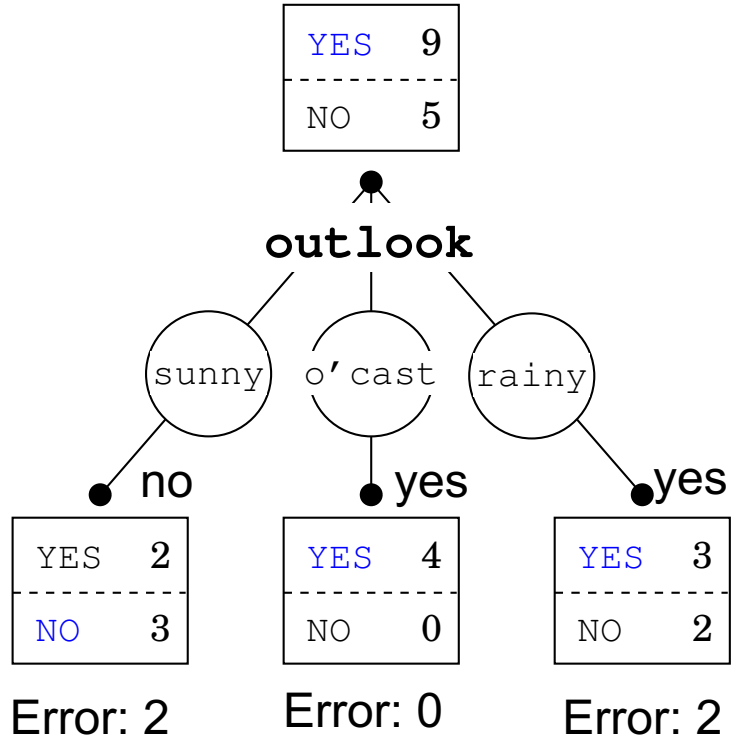
- How to build optimal Decision Tree?
- What does optimality mean?
- How to choose attribute values at each decision point (node)?
- When to stop the growth of the tree?



Decision Stump

	Outlook	Play
a	sunny	no
b	sunny	no
c	overcast	yes
d	rainy	yes
e	rainy	yes
f	rainy	no
g	overcast	yes
h	sunny	no
i	sunny	yes
j	rainy	yes
k	sunny	yes
l	overcast	yes
m	overcast	yes
n	rainy	no

- leaves: immediately connects to the root
- predictions: made based on one attribute



Majority voting

Total Error = 4/14

Decision Tree Algorithm

- Optimal construction of a Decision Tree is NP (non-deterministic polynomial) hard.
- So we use heuristics:
  - Choose an attribute to partition the data at the node such that each partition is as homogeneous (least impure) as possible. This means we would like to see most of the instances in each partition belonging to as few classes as possible and each partition should be as large as possible.
  - We can stop the growth of the tree if all the leaf nodes are largely dominated by a single class (that is the leaf nodes are nearly pure).

ID3 Algorithm

Basic method: construct decision trees in recursive divide-and-conquer fashion

FUNCTION ID3 (Root)

IF all instances at root have same class

THEN stop

ELSE

1. Select a new attribute to use in partitioning root node instances

2. Create a branch for each attribute value and partition up root node instances according to each value

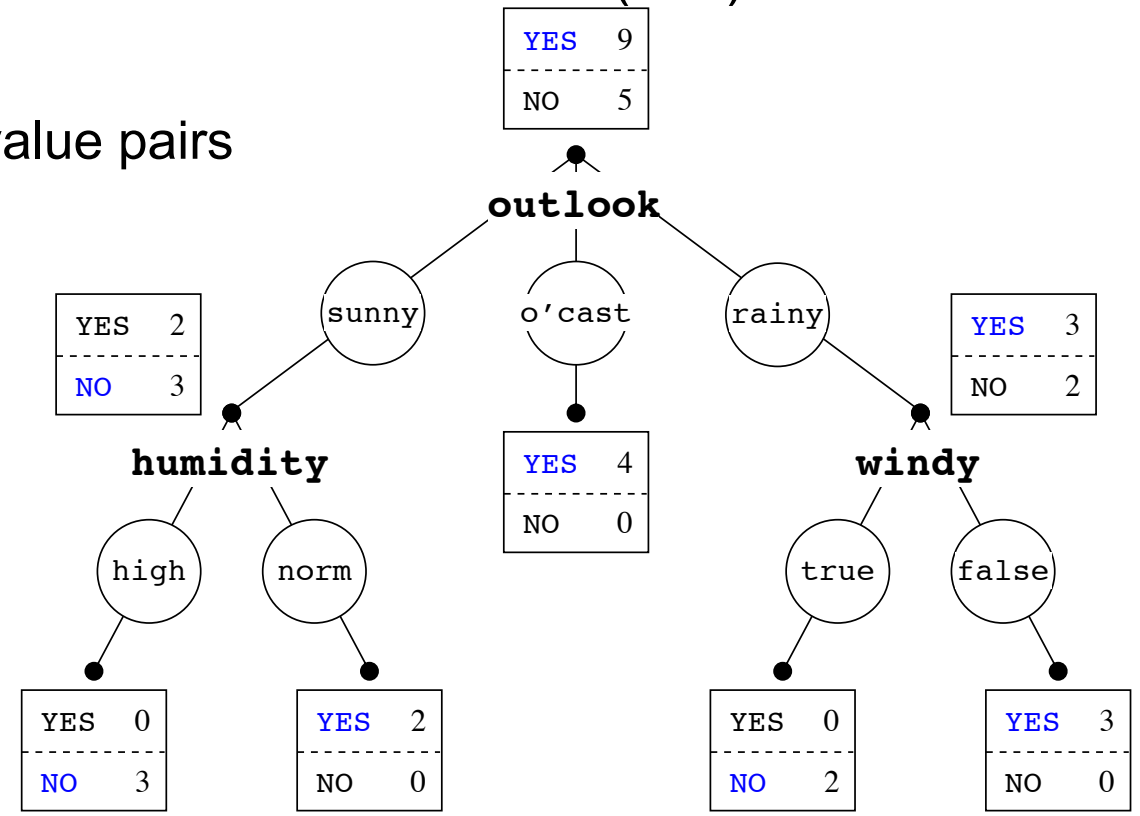
3. Call ID3($LEAF_i$) for each leaf node $LEAF_i$

Classification Example

- Having constructed the decision tree, we classify novel instances by traversing down the tree and classifying according to the label at the deepest reachable point in the tree structure (leaf)
- Complications:
  - unobserved attribute–value pairs
  - missing values

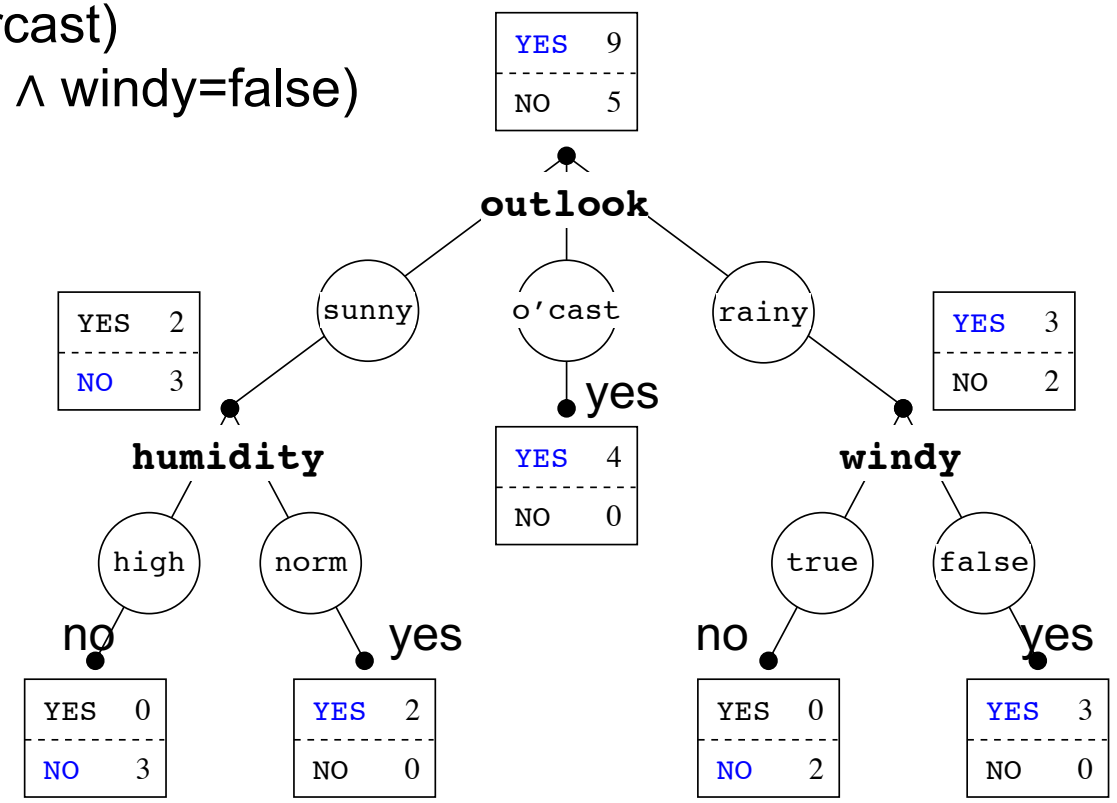
Try to classify these test instances:

(sunny, hot, normal, False)
(rainy, hot, low, False)
(?,cool, high, True)



Disjunctive descriptions

Decision Trees can be read as a disjunction; for example, yes:
(outlook=sunny ∧ humidity=normal)
∨ (outlook=overcast)
∨ (outlook=rainy ∧ windy=false)



Total Error = 0/14

Criterion for Attribute Selection

- How do we choose the attribute to partition the instances at a given node?
- We want to get the smallest tree. Prefer the shortest hypothesis that fits the data.
- In favor:
  - Fewer short hypotheses than long hypotheses
    - a short hyp. that fits the data unlikely to be a coincidence
    - a long hyp. that fits data might be a coincidence
- Against:
  - Many ways to define small sets of hypotheses

Criterion for Attribute Selection: Information Gain

Information Gain: 'Reduction of entropy before and after the data is partitioned using the attribute A'.

- Entropy: 'the expected (average) value of self-information'.
- Self-information: 'level of information (surprise) associated with one particular outcome (event) of a random variable'.
  - low probability:
    - event not likely to happen
    - if it happens, it is big news and there is
      - high information/surprise
  - high probability:
    - event is very likely to happen,
    - if it happens, it is not a news and there is
      - low information/surprise

$$\text{Self-information} \sim \frac{1}{\text{probability}}$$

Entropy

- Entropy: 'the expected (average) value of self-information'.
- Self-information: 'level of information (surprise) associated with one particular outcome (event) of a random variable'.

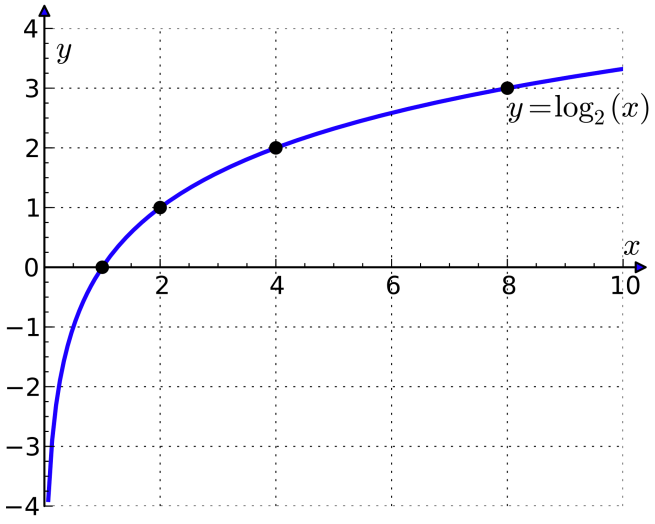
$$\text{Self-information} \sim \frac{1}{\text{probability}}$$

$\text{Self-information} = \frac{1}{\text{P}}$: not convenient for computation.

$$\text{Self-information} = log_2 \frac{1}{P} = -log_2 P$$

$$H(x) = \sum_{i=1}^{n} P(i)\,\text{Self-information}(i)$$

$$= -\sum_{i=1}^{n} P(i)\, log_2 P(i)$$



Entropy

Flip a normal coin 100 times:
52 head instances, 48 tail instances:

$$H = -\left[\frac{52}{100} log_2 \frac{52}{100} + \frac{48}{100} log_2 \frac{48}{100}\right] = 0.999$$

A special 2-headed coin
100 head instances, 0 tail instances:

$$H = ?$$

Entropy

Flip a normal coin 100 times:
52 head instances, 48 tail instances:

$$H(x) = -\sum_{i=1}^{n} P(i)\, log_2 P(i)$$

$$H = -\left[\frac{52}{100} log_2 \frac{52}{100} + \frac{48}{100} log_2 \frac{48}{100}\right] = 0.999$$

A special 2-headed coin
100 head instances, 0 tail instances:

$$H = -\left[\frac{100}{100} log_2 \frac{100}{100} + \frac{0}{100} log_2 \frac{0}{100}\right] = 0$$

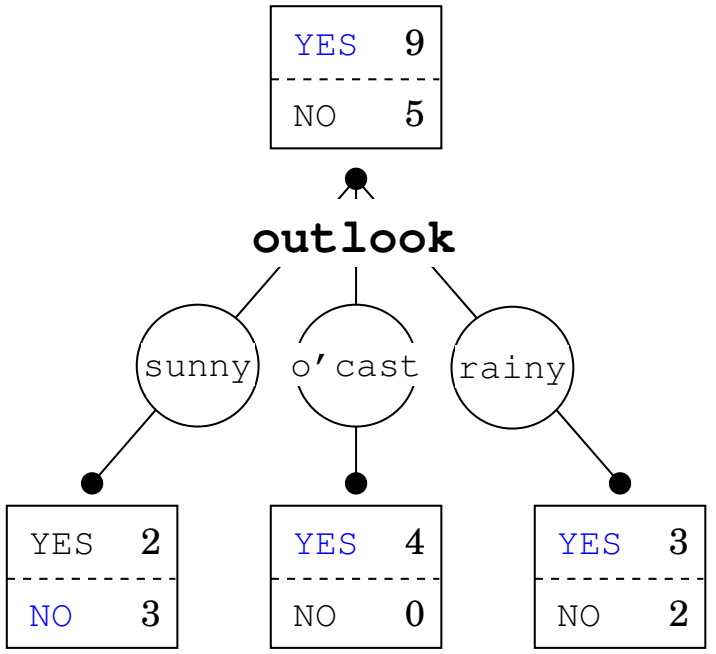
pure class: entropy=0
evenly distributed: entropy=1

$$H = -\left[\frac{50}{100} log_2 \frac{50}{100} + \frac{50}{100} log_2 \frac{50}{100}\right] = 1$$

Entropy

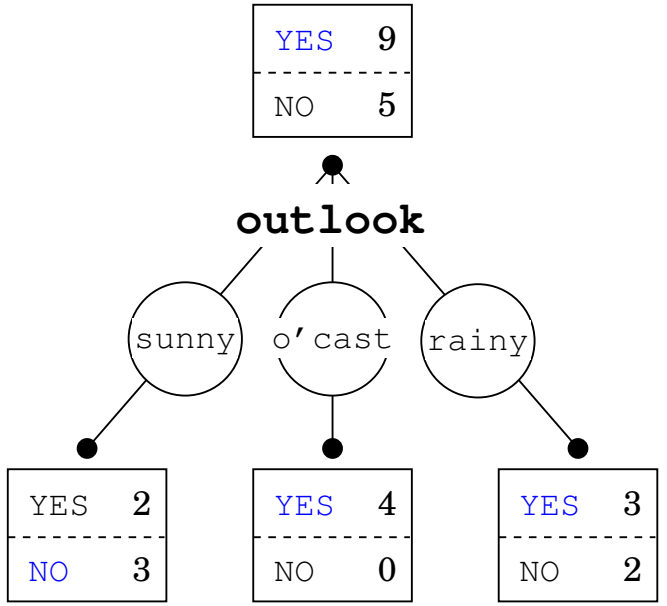
Entropy is a measure of unpredictability:

- If the probability of a single class is high:
  - the class is predictable
  - low entropy
- If the probability is evenly divided between multiple classes:
  - the class is unpredictable
  - high entropy



Criterion for Attribute Selection: Information Gain

- decision tree with low entropy: class is more predictable.
- Information Gain (reduction of entropy): measures how much uncertainty was reduced.
- Select the attribute that has largest information gain: the most entropy (uncertainty) is reduced, class is mostly predictable.



Information Gain

$$IG(R_A|R) = H(R) - MeanInfo(x_1, \dots, x_m)$$

$$Mean\ Info(x_1, \dots, x_m) = \sum_{i=1}^{m} w(x_i)H(x_i)$$

$$H(x) = -\sum_{i=1}^{n} P(i)\ log_2 P(i) \qquad (0log_2 0 = 0)$$

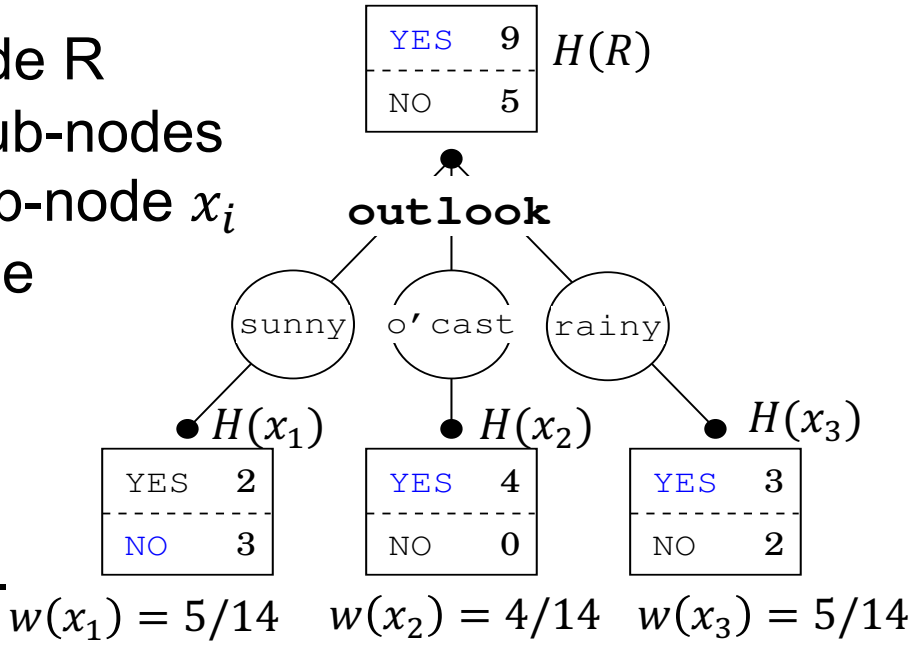
H(R): entropy of the instances in node R

Mean Info: weighted entropy of the sub-nodes

$H(x_i)$: entropy of the instances in sub-node x_i

$P(i)$: the probability of the i^{th} outcome

$w(x_i)$: weight of each sub-node (the proportion of the number of instances at sub-node x_i to the total number of instances).



Entropy (outlook)

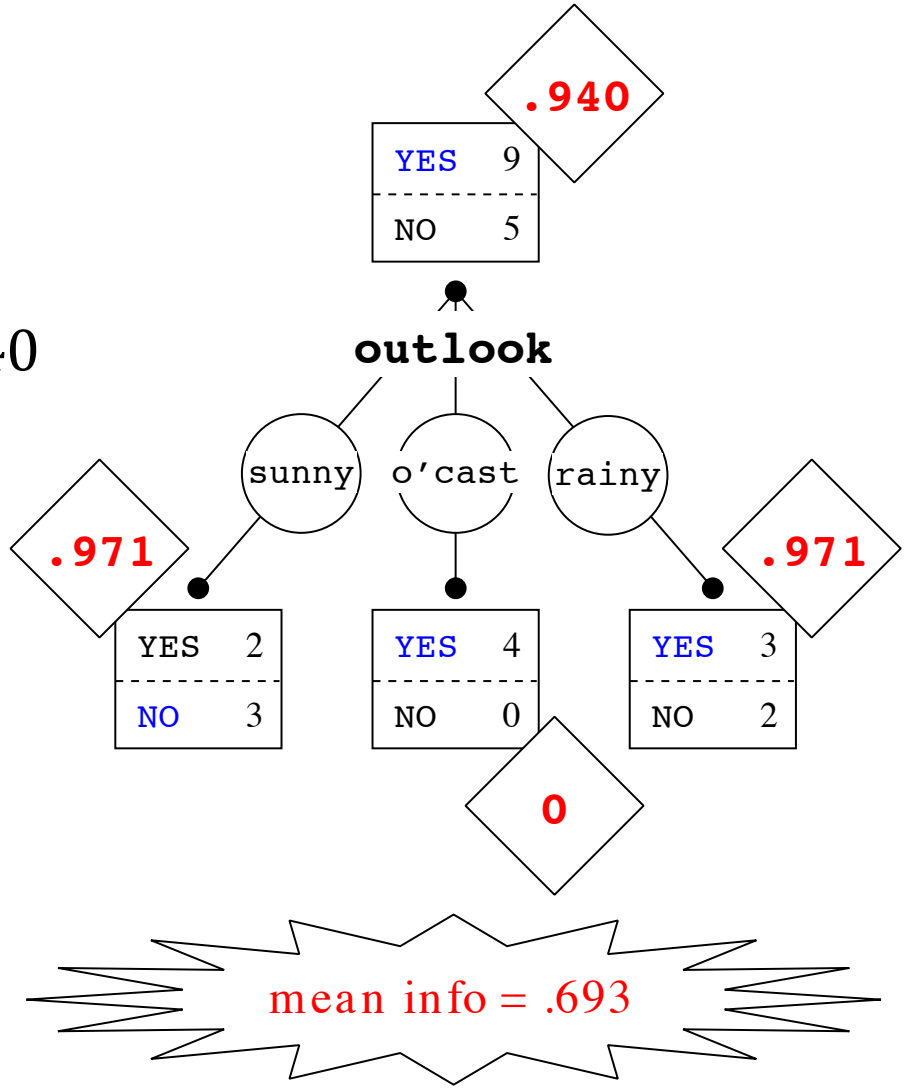
$$H(x) = -\sum_{i=1}^{n} P(i)\, log_2 P(i)$$

$$H(root) = -\left[\frac{9}{14} log_2 \frac{9}{14} + \frac{5}{14} log_2 \frac{5}{14}\right] = 0.940$$

$$H(sunny) = -\left[\frac{2}{5} log_2 \frac{2}{5} + \frac{3}{5} log_2 \frac{3}{5}\right] = 0.971$$

$$H(overcast) = -\left[\frac{4}{4} log_2 \frac{4}{4} + 0 log_2 0\right] = 0$$

$$H(rainy) = -\left[\frac{3}{5} log_2 \frac{3}{5} + \frac{2}{5} log_2 \frac{2}{5}\right] = 0.971$$



Information Gain (outlook)

$$IG(R_A|R) = H(R) - MeanInfo(x_1, \ldots, x_m)$$

$$Mean\ Info(x_1, \ldots, x_m) = \sum_{i=1}^{m} w(x_i)H(x_i)$$

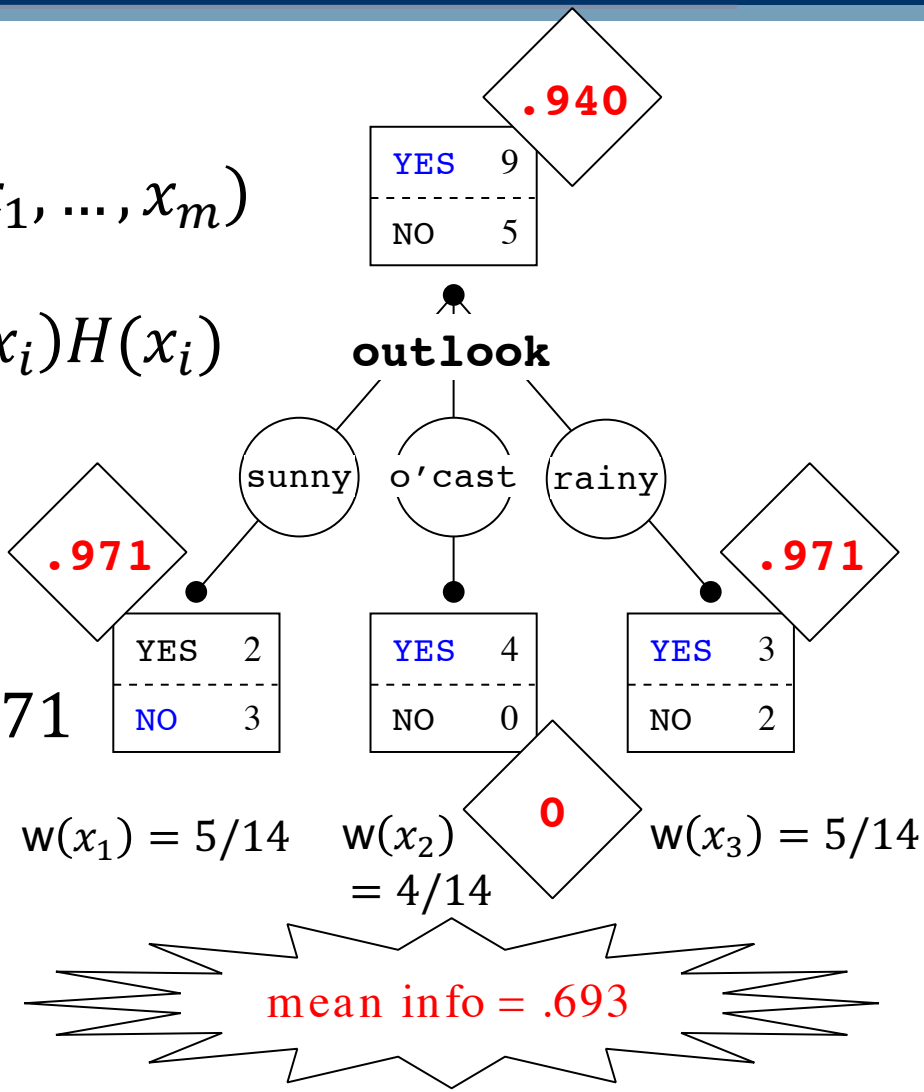
Mean_info(outlook) =

$$\frac{5}{14} \times 0.971 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0.971$$

$$= 0.693$$

$$IG(outlook|R) = 0.940 - 0.693$$

$$= 0.247$$



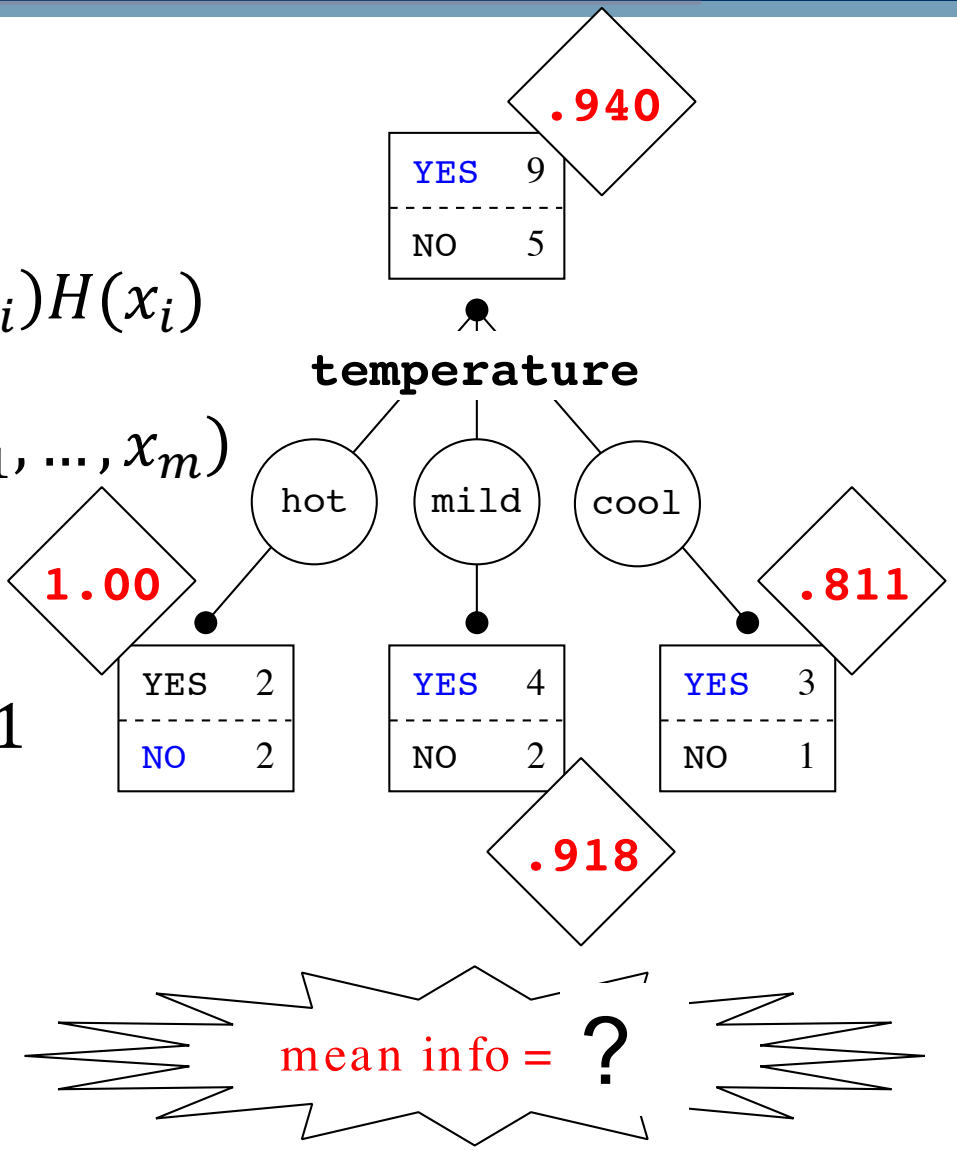
Information Gain (temperature)

$$H(x) = -\sum_{i=1}^{n} P(i)\, log_2 P(i)$$

$$Mean\ Info(x_1, \ldots, x_m) = \sum_{i=1}^{m} w(x_i) H(x_i)$$

$$IG(R_A|R) = H(R) - MeanInfo(x_1, \ldots, x_m)$$

$$H(hot) = -\left[\frac{2}{4} log_2 \frac{2}{4} + \frac{2}{4} log_2 \frac{2}{4}\right] = 1$$



Information Gain (temperature)

$$H(x) = -\sum_{i=1}^{n} P(i)\, log_2 P(i)$$

$$Mean\ Info(x_1, \ldots, x_m) = \sum_{i=1}^{m} w(x_i) H(x_i)$$

$$IG(R_A|R) = H(R) - MeanInfo(x_1, \ldots, x_m)$$

$$H(hot) = -\left[\frac{2}{4} log_2 \frac{2}{4} + \frac{2}{4} log_2 \frac{2}{4}\right] = 1$$

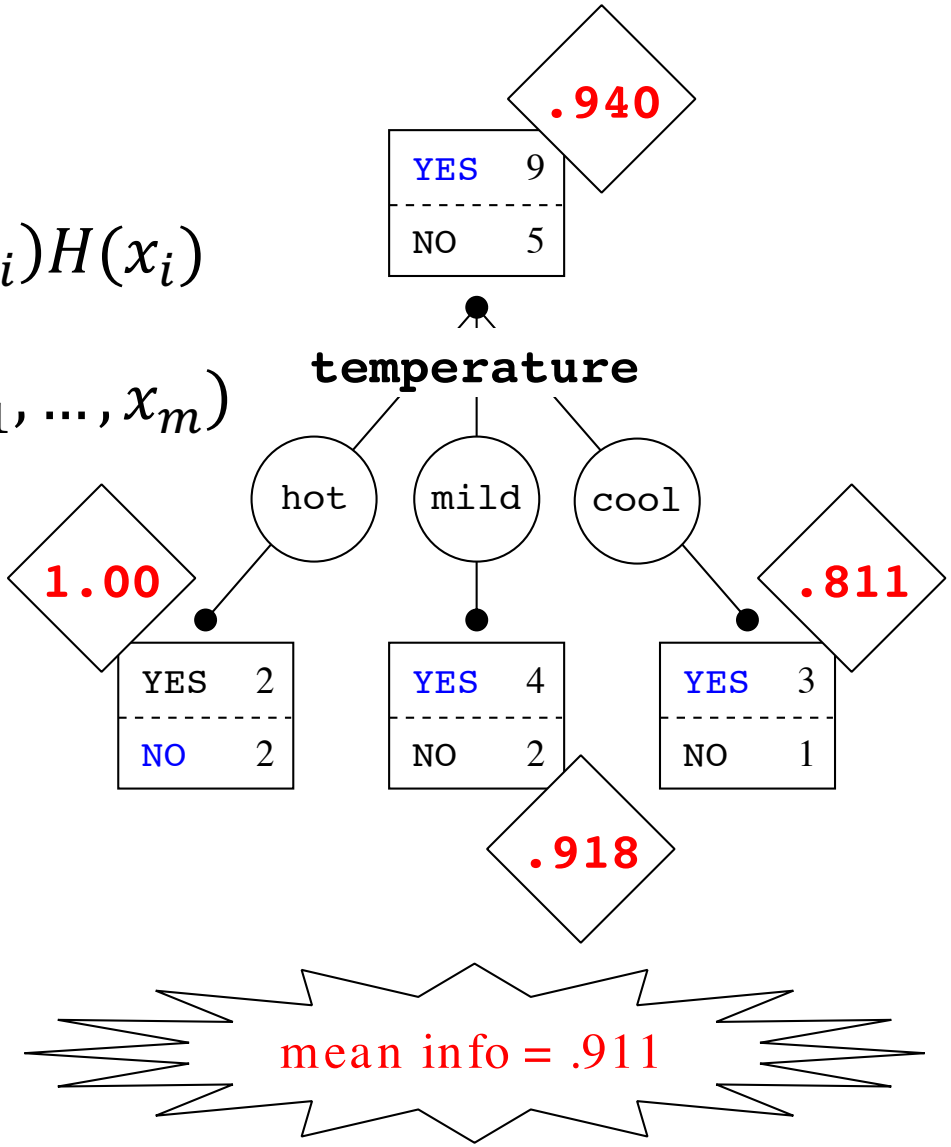
Mean_info(temperature) =

$$\frac{4}{14} \times 1 + \frac{6}{14} \times 0.918 + \frac{4}{14} \times 0.811$$

$$= 0.911$$

$$IG(\text{temperature}|R) = 0.940 - 0.911$$

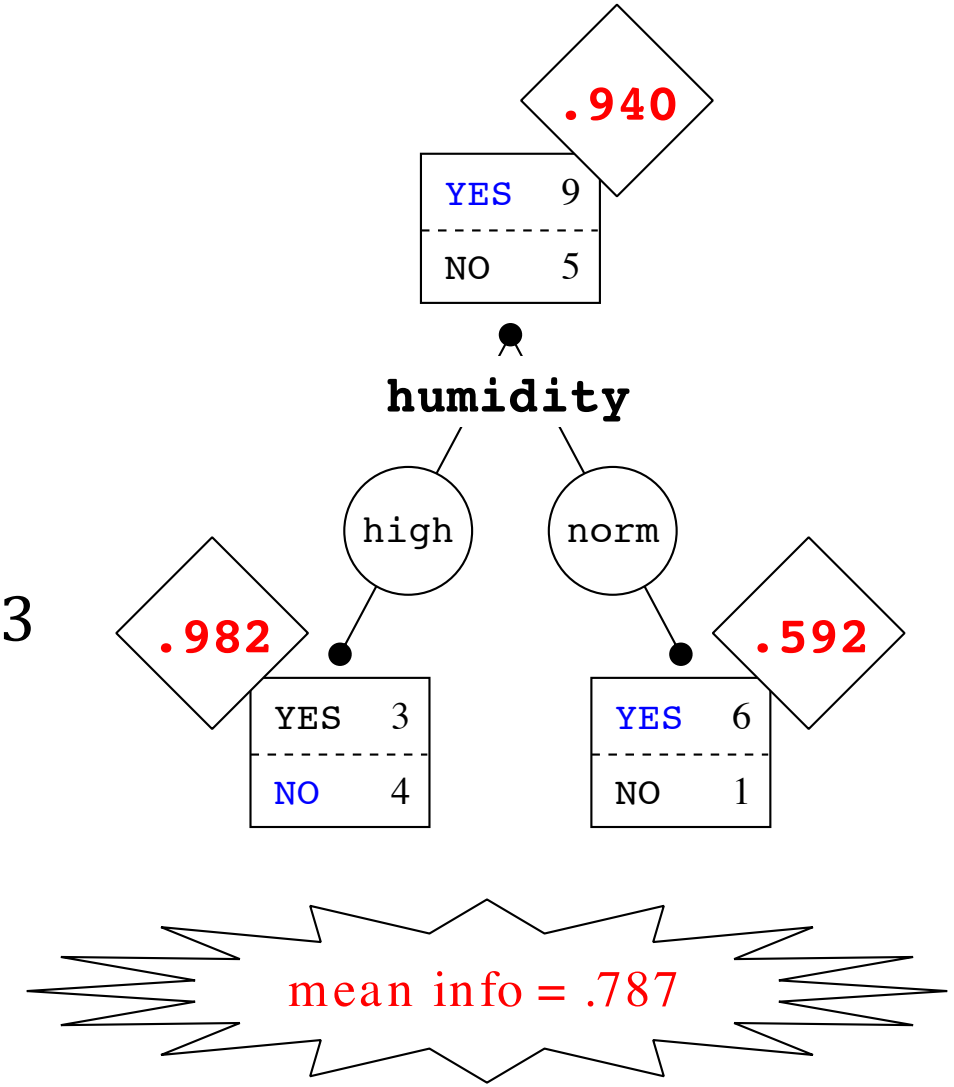
$$= 0.029$$



Information Gain (humidity)

$$H(R) = 0.94$$

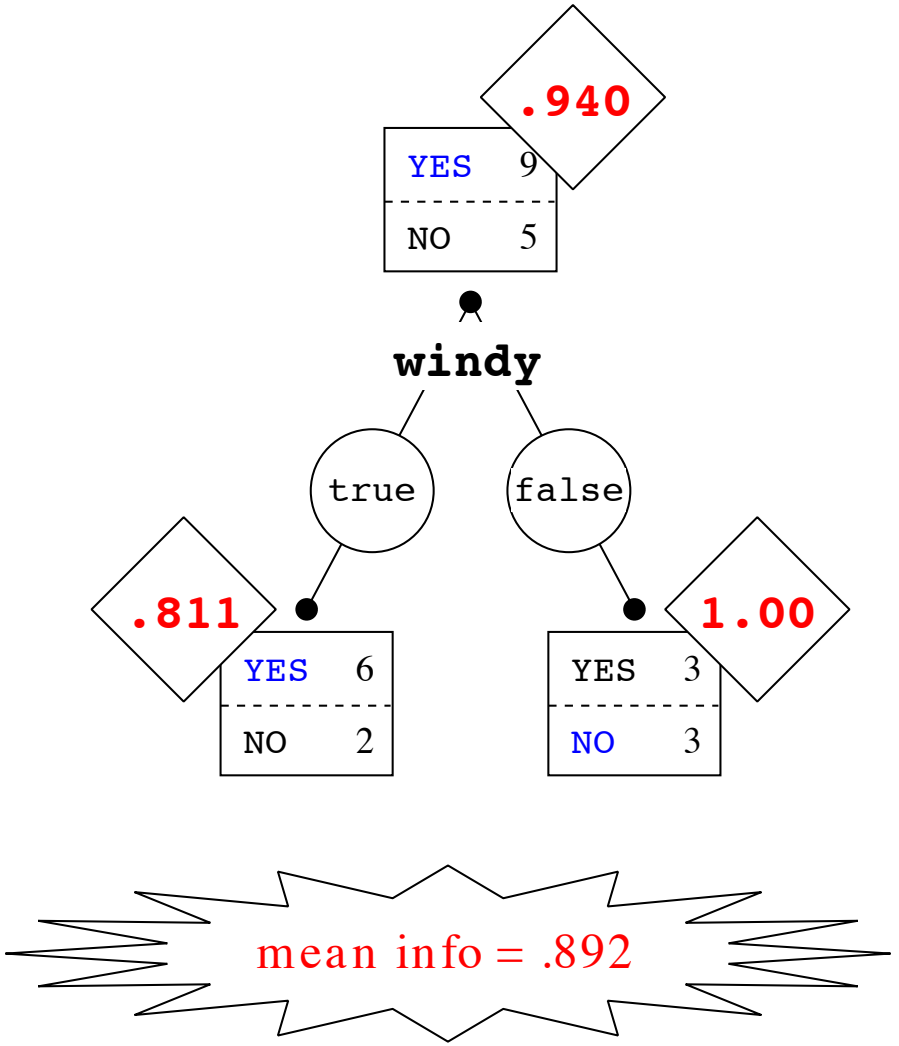
$$IG(humidity|R) = 0.94 - 0.787 = 0.153$$



Information Gain (windy)

$$H(R) = 0.94$$

$$IG(windy|R) = 0.94 - 0.892 = 0.048$$

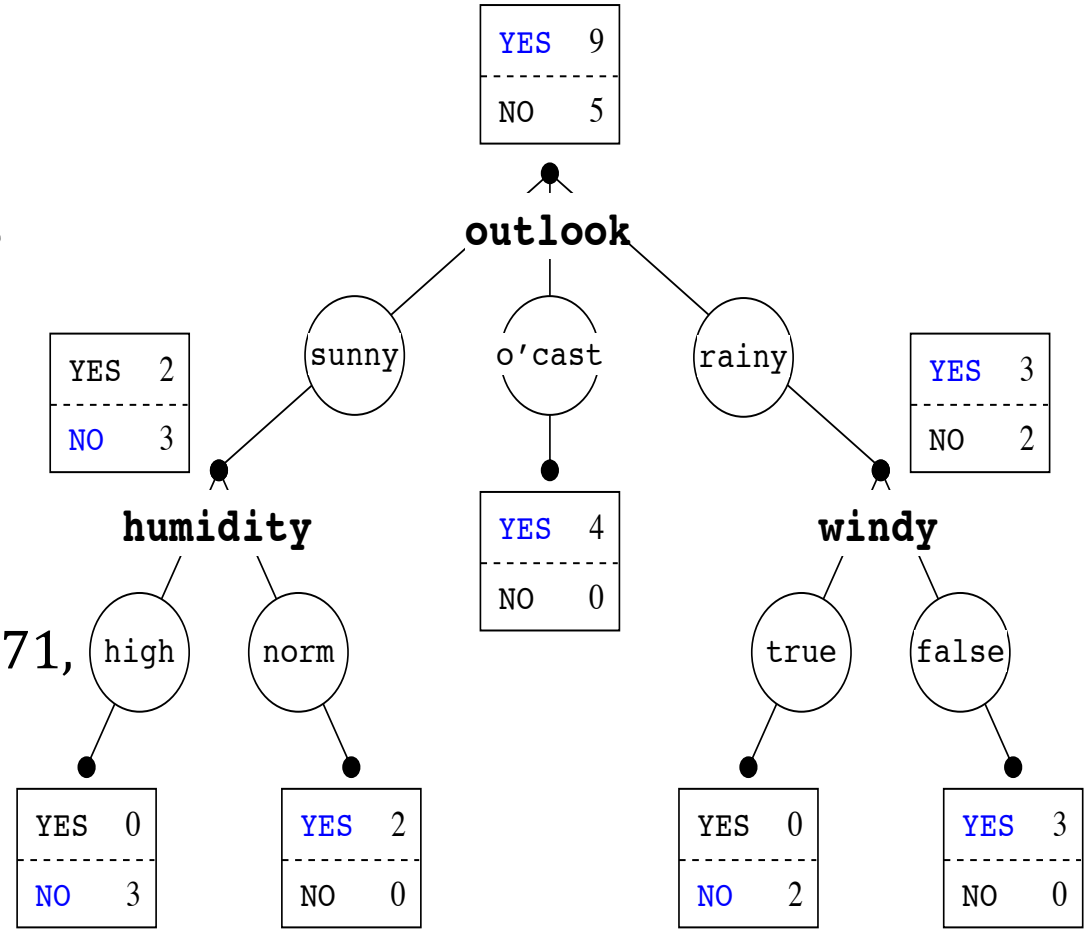


Attribute Selection: Information Gain

more IG: reduce more entropy & makes the class more predictable

$IG(outlook|R) = 0.247,$
$IG(temperature|R) = 0.029,$
$IG(humidity|R) = 0.153,$
$IG(windy|R) = 0.048$

$H(sunny) = 0.971$
$IG(temporature|sunny) = 0.571,$
$IG(humidity|sunny) = 0.971,$
$IG(windy|sunny) = 0.020,$

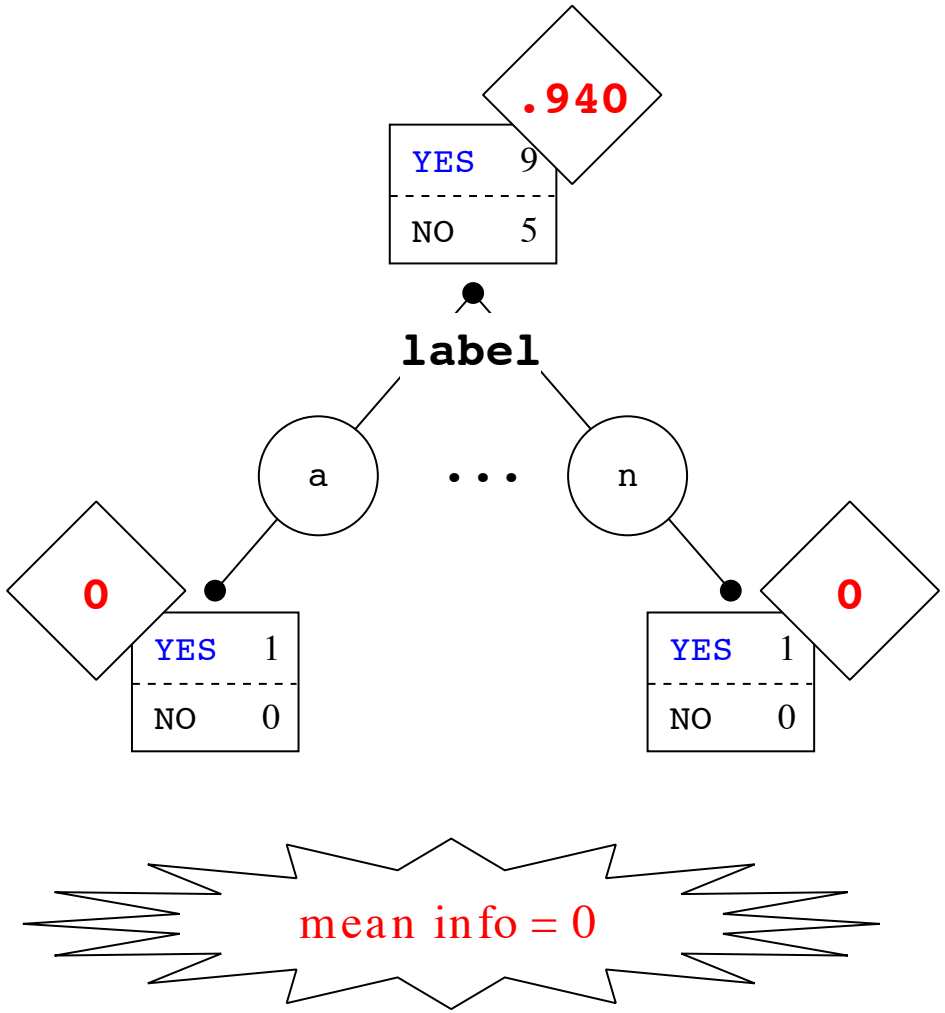


Shortcomings of Information Gain

- Information gain tends to prefer highly-branching attributes:
  - For an attribute with is a large number of values, subsets are more likely to be pure (above a purity threshold):
    - entropy is 0, mean information is 0.
    - information gain is large

- This may result in overfitting/fragmentation.

Shortcomings of Information Gain



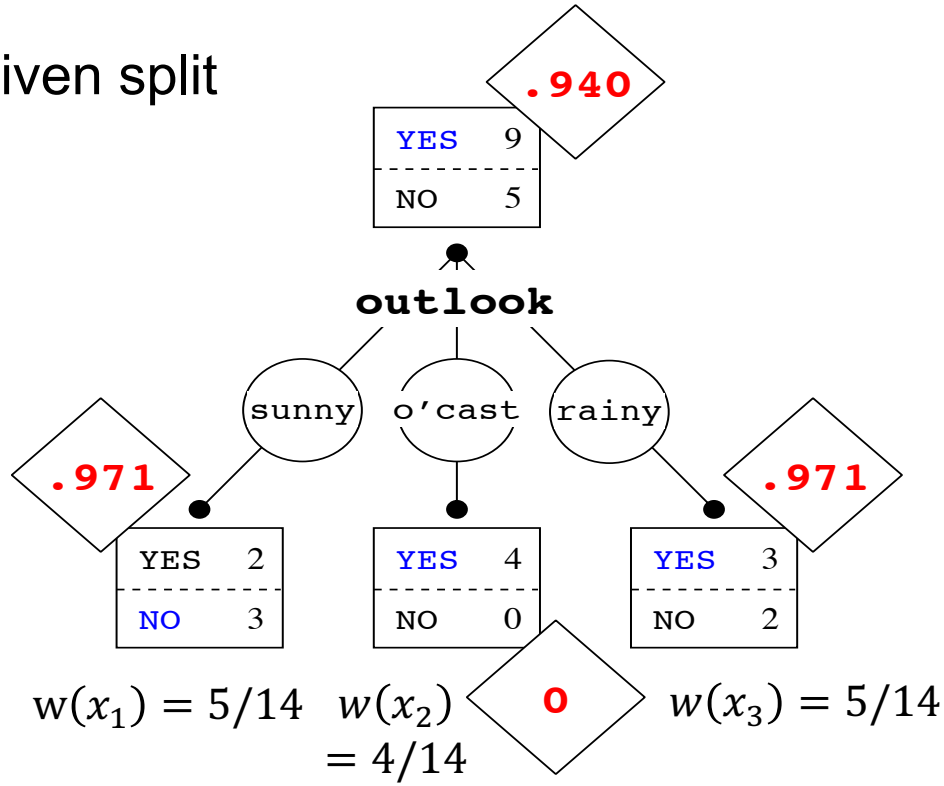
Solution: Gain Ratio

- Gain ratio (GR) reduces the bias for information gain towards highly-branching attributes by normalising relative to the split info
- Discourages the selection of attributes with many uniformly distributed values
- Split info (SI) is the entropy of a given split (evenness of split)

$$GR(R_A|R) = \frac{IG(R_A|R)}{SI(R_A|R)}$$

$$SI(R_A|R) = -\sum_{i=1}^{m} w(x_i)\, log_2 w(x_i)$$

$w(x_i)$: weight of each sub-node (the proportion of the number of instances at sub-node x_i to the total number of instances).



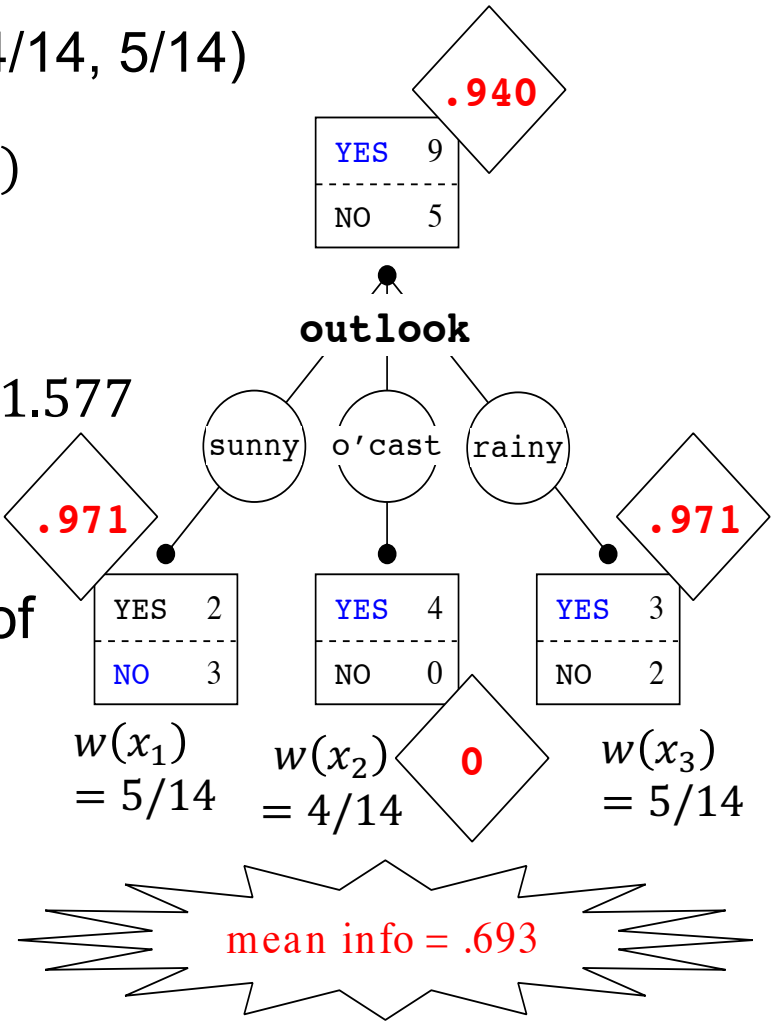
Split Info

- Entropy of the split distribution (5/14, 4/14, 5/14)

$$SI(R_A|R) = -\sum_{i=1}^{m} w(x_i)\, log_2 w(x_i)$$

$$SI(outlook|R) = -[\frac{5}{14} log_2 \frac{5}{14} + \frac{4}{14} log_2 \frac{4}{14} + \frac{5}{14} log_2 \frac{5}{14}] = 1.577$$

- Different from mean information: mean information needs to calculate entropy of the class distribution of each split



$$Mean\ Info(x_1, \ldots, x_m) = \sum_{i=1}^{m} w(x_i) H(x_i)$$

Mean_info(outlook) =

$$\frac{5}{14} \times 0.971 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0.971 = 0.693$$

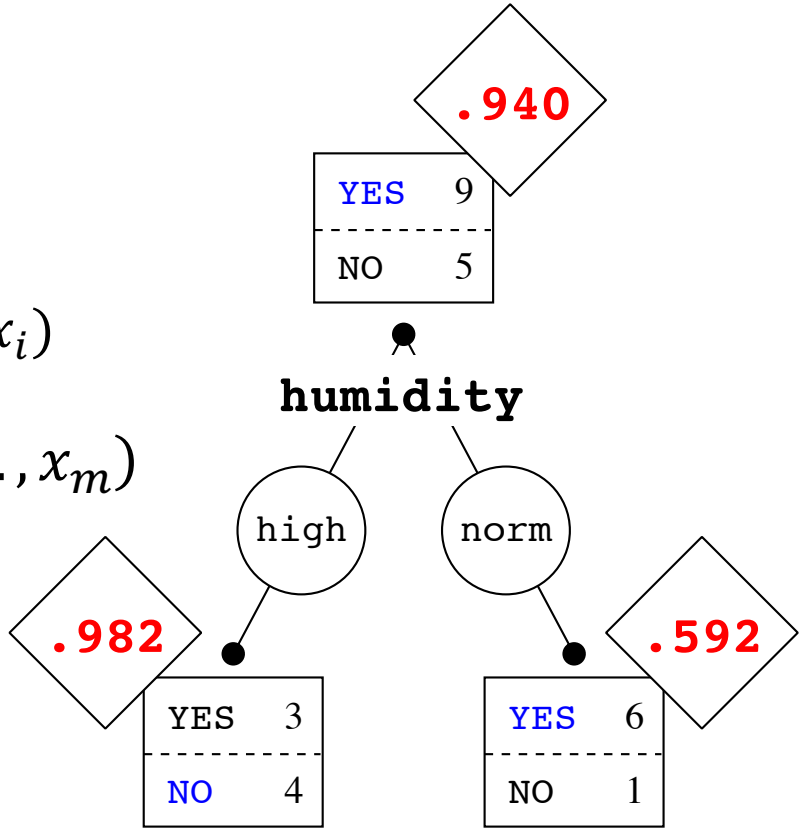
Gain Ratio: Example

$$SI(R_A|R) = -\sum_{i=1}^{m} w(x_i)\, log_2 w(x_i)$$

$$Mean\ Info(x_1, \ldots, x_m) = \sum_{i=1}^{m} w(x_i) H(x_i)$$

$$IG(R_A|R) = H(R) - MeanInfo(x_1, \ldots, x_m)$$

$$GR(R_A|R) = \frac{IG(R_A|R)}{SI(R_A|R)}$$



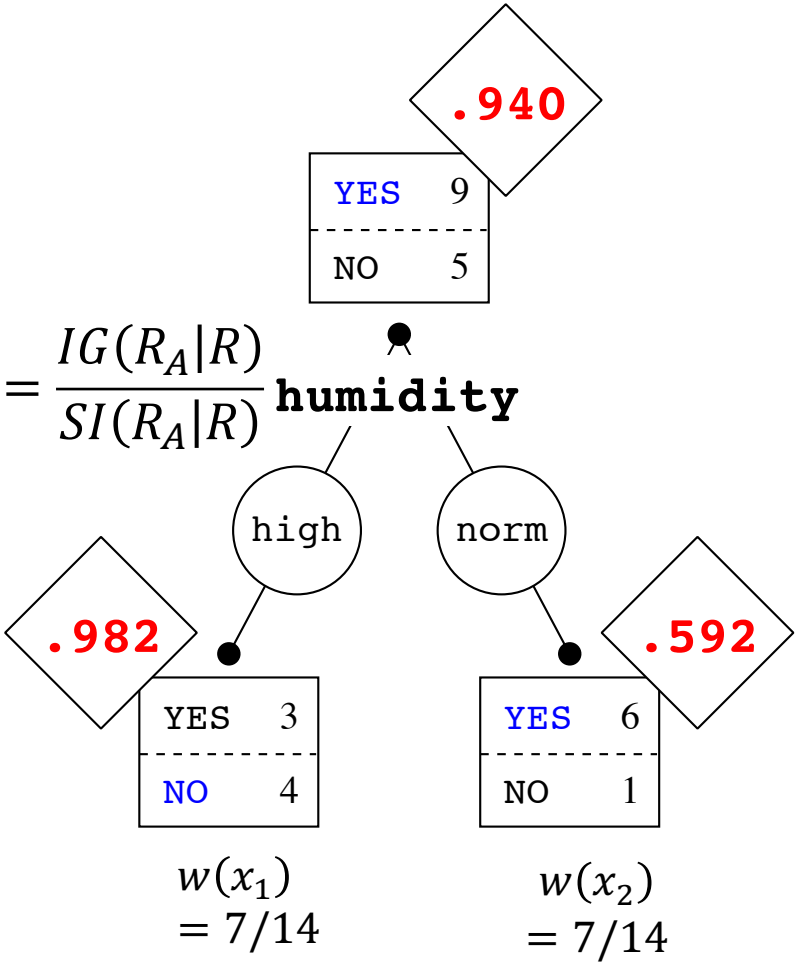
$$GR(humidity|R) = ?$$

Gain Ratio: Example

$$Mean\ Info(x_1, \ldots, x_m) = \sum_{i=1}^{m} w(x_i) H(x_i)$$

$$IG(R_A|R) = H(R) - MeanInfo(x_1, \ldots, x_m)$$

$$SI(R_A|R) = -\sum_{i=1}^{m} w(x_i)\ log_2 w(x_i) \qquad GR(R_A|R) = \frac{IG(R_A|R)}{SI(R_A|R)}$$



$$SI(humidity|R) = -\left[\frac{7}{14} log_2 \frac{7}{14} + \frac{7}{14} log_2 \frac{7}{14}\right] = 1$$

Mean_info(humidity) =

$$\frac{7}{14} \times 0.982 + \frac{7}{14} \times 0.592 = 0.787$$

$$IG(humidity|R) = 0.94 - 0.787 = 0.153$$

$$GR(humidity|R) = 0.153$$

Gain Ratio: Example

$$IG(outlook|R) = 0.247$$
$$SI(outlook|R) = 1.577$$
$$GR(outlook|R) = 0.157$$

$$IG(temperature|R) = 0.029$$
$$SI(temperature|R) = 1.557$$
$$GR(temperature|R) = 0.019$$

$$IG(humidity|R) = 0.153$$
$$SI(humidity|R) = 1$$
$$GR(humidity|R) = 0.153$$

$$IG(windy|R) = 0.048$$
$$SI(windy|R) = 0.985$$
$$GR(windy|R) = 0.049$$

$$IG(label|R) = 0.940$$
$$SI(label|R) = 3.807$$
$$GR(label|R) = 0.247$$

Stopping Criteria I

- The definition of ID3 above suggests that:
  - We recurse until the instances at a node are of the same class
  - This is consistent with our usage of entropy: if all of the instances are of a single class, the entropy of the distribution is 0
  - Considering other attributes cannot “improve” an entropy of 0 — the Info Gain is 0 by definition
- This helps to ensure that the tree remains compact

Stopping Criteria II

- The definition of ID3 above suggests that:
  - The Info Gain/Gain Ratio allows us to choose the (seemingly) best attribute at a given node
  - However, it is also an approximate indication of how much absolute improvement we expect from partitioning the data according to the values of a given attribute
  - An Info Gain of 0 means that there is no improvement; a very small improvement is often unjustifiable
  - Typical modification of ID3: choose best attribute only if IG/GR is greater than some threshold
  - Other similar approaches use pruning — post-process the tree to remove undesirable branches (with few instances, or small IG/GR improvements)

Stopping Criteria III

- The definition of ID3 above suggests that:
  - We might observe improvement through every layer of the tree
  - We then run out of attributes, even though one or more leaves could be improved further
  - Fall back to majority class label for instances at a leaf with a mixed distribution — unclear what to do with ties
  - Possibly can be taken as evidence that the given attributes are insufficient for solving the problem

Hypothesis Space Search in ID3

- ID3 performs a simple-to-complex, hill-climbing search through hypothesis space
- Beginning with the empty tree, then considering progressively more elaborate hypotheses in search of a decision tree that correctly classifies the training data.
- It does not do back tracking on selected attribute. Can get stuck in local optimal.

Disadvantages of Decision Tree

- Overfitting
- Loss information for continuous variables
- Complex calculation if there are many classes
- No guarantee to return the globally optimal decision
- Information gain: Bias for attributes with greater no. of values.

Summary

- Describe the basic decision tree induction method used in ID3
- What is information gain, how is it calculated and what is its primary shortcoming?
- What is gain ratio, and how does it attempt to overcome the shortcoming of information gain?
- What are the properties of ID3-style decision trees?

- *Next lecture: Feature Selection*

References

- *Some slides are from:*
- *Mitchell, Tom (1997). Machine Learning. Chapter 3: Decision Tree Learning.*
- *Tan et al (2006) Introduction to Data Mining. Section 4.3, pp 150-171.*
- *Quinlan, J. R. 1986. Induction of Decision Trees. Mach. Learn. 1, 1 (Mar. 1986), 81–106*
- https://www.datacamp.com/community/tutorials/decision-tree-classification-python
- Jeremy Nicholson & Tim Baldwin & Karin Verspoor: Machine Learning
- https://smallbusiness.chron.com/advantages-decision-trees-75226.html
- https://en.wikipedia.org/wiki/Entropy_(information_theory)