



ID3

Algorithm By:

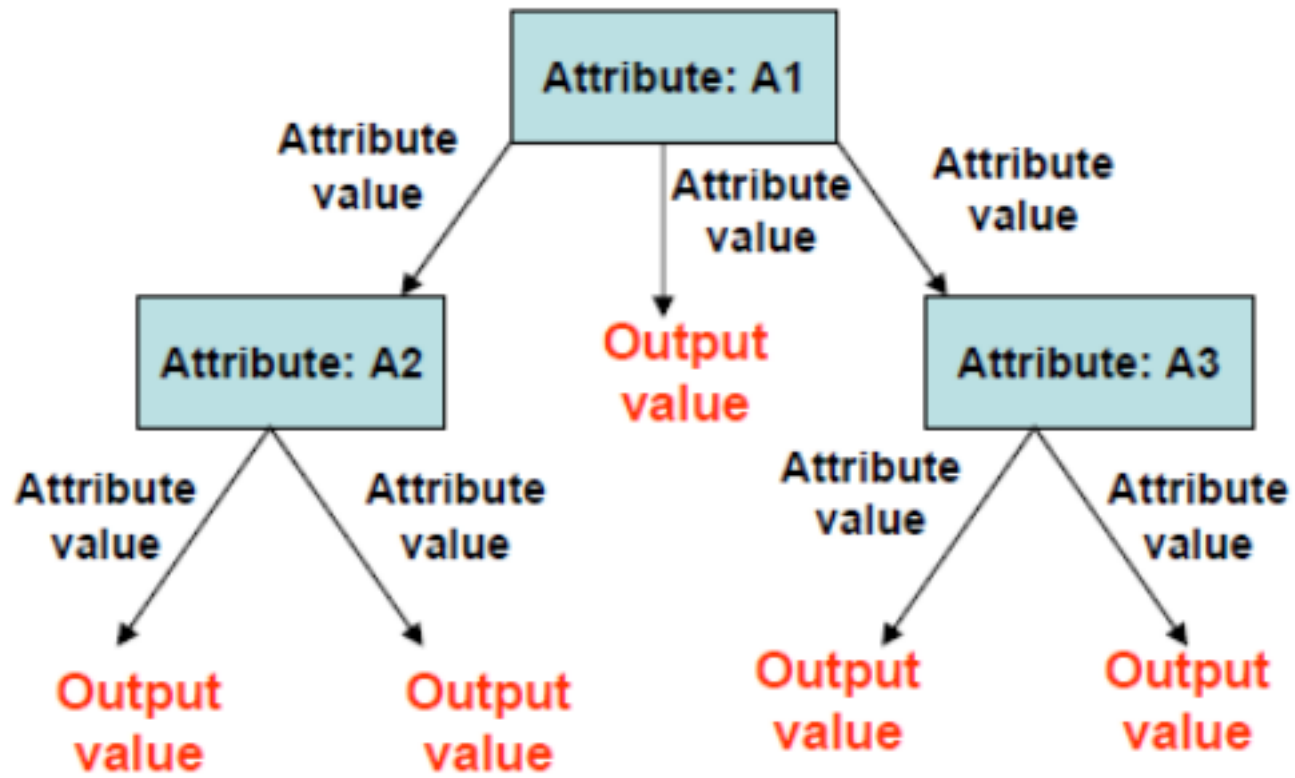
Dr. Sonali Patil

Agenda

- Decision Trees
- What is ID3?
- Entropy
- Calculating Entropy with Code
- Information Gain
- Advantages and Disadvantages
- Example

Decision Trees

- DT learning is a method for approximating **discrete value target functions**, in which learned function is represented by a decision tree
- Rules (if-then-else) for classifying data using attributes.
- The tree consists of decision nodes and leaf nodes
- A decision node has two or more branches, each representing values for the attribute tested •
- A leaf node attribute produces a homogeneous result (all in one class), which does not require additional classification testing.
- Most widely used approach for inductive inference



Decision Tree Example



Example

Outlook

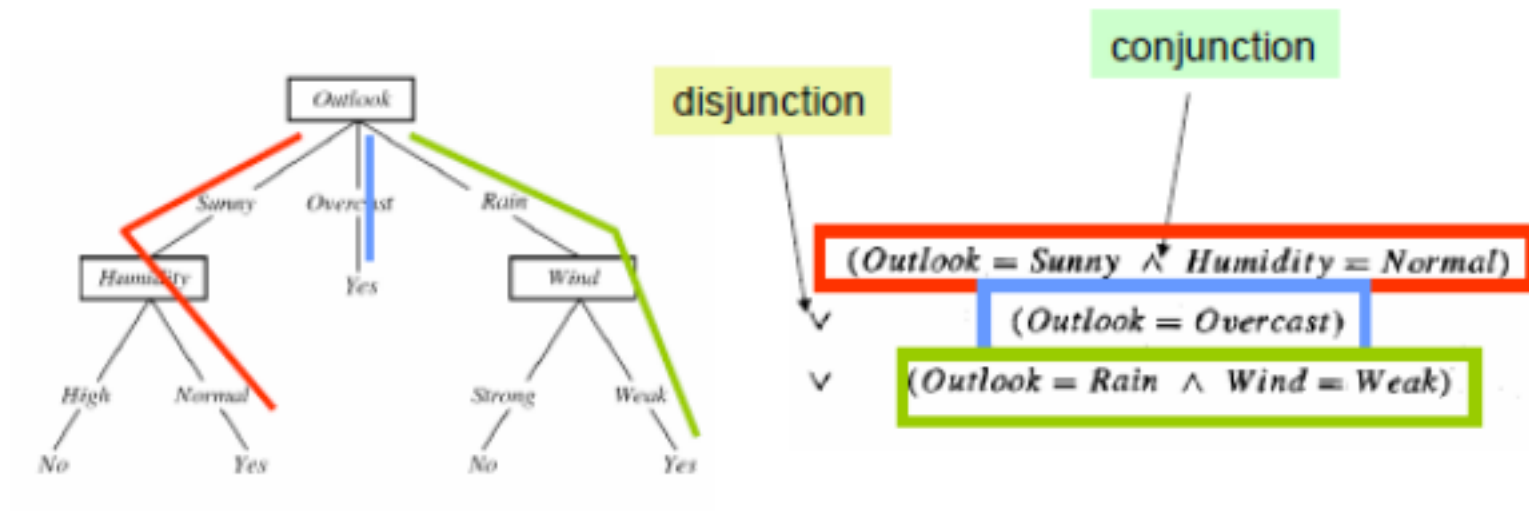
sunny rain

overcast

Yes

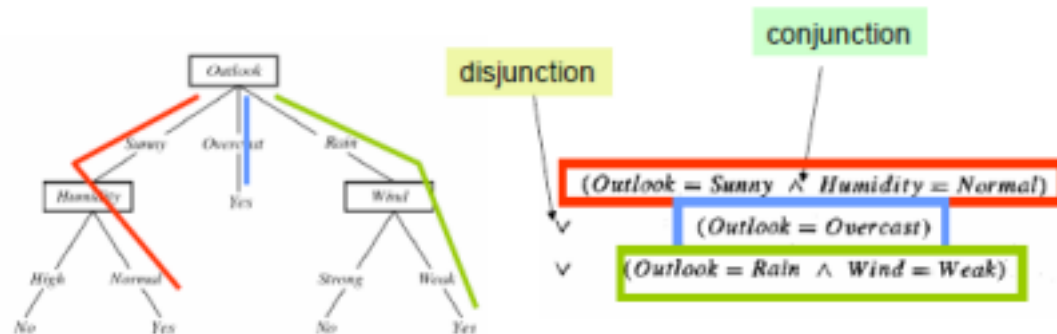
Humidity Windy high normal true false

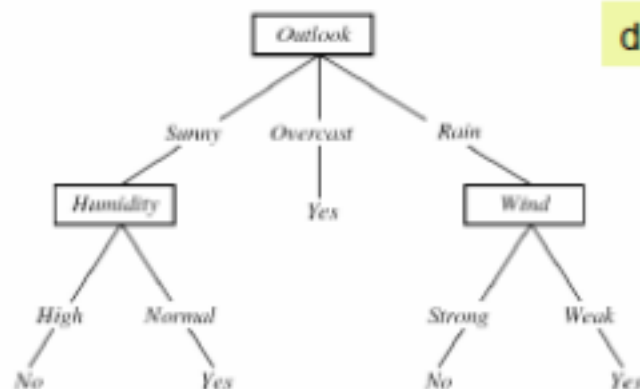
No Yes No Yes



Decision Tree

Representation





disjunction

conjunction

$(\text{Outlook} = \text{Sunny} \wedge \text{Humidity} = \text{Normal})$
 $(\text{Outlook} = \text{Overcast})$
 $(\text{Outlook} = \text{Rain} \wedge \text{Wind} = \text{Weak})$

- If (Outlook = Sunny AND humidity = Normal) then PlayTennis = Yes
- If (Outlook = Overcast) then PlayTennis = Yes
- If (Outlook = Rain AND Wind = Weak) then PlayTennis = Yes

Decision Tree as If-then-else rule

What is ID3?

- A mathematical algorithm for building the decision tree.
- Invented by J. Ross Quinlan in 1979.
- Uses Information Theory invented by Shannon in 1948.
- Builds the tree from the top down, with no backtracking.
- Information Gain is used to select the most useful attribute for classification.



Entropy



- A completely homogeneous sample has entropy of 0.

- An equally divided sample has entropy of 1.
- A

formula to calculate the homogeneity of a sample.

$$\text{Entropy}(s) = - (p+) \log_2(p+) - (p-) \log_2(p-)$$



Entropy Example

Entropy(S) =

$$- (9/14) \log_2(9/14) - (5/14) \log_2(5/14)$$

$$= 0.940$$



Information Gain (IG)

- The information gain is based on the decrease in entropy after a dataset is split on an attribute.
- Which attribute creates the most homogeneous branches?
- First the entropy of the total dataset is calculated. • The dataset is then split on the different attributes. • The entropy for each branch is calculated. Then it is added proportionally, to get total entropy for the split.
- The resulting entropy is subtracted from the entropy before the split.
- The result is the Information Gain, or decrease in entropy.
- The attribute that yields the largest IG is chosen for the decision node.





Information Gain (IG)

- The information gain , $\text{Gain}(S,A)$ of an attribute A , relative to a collection of examples S , is defined as





Information Gain (IG)





Information Gain

(cont'd) • A branch set with entropy of 0

is a leaf node. • Otherwise, the branch needs further splitting to classify its dataset.

- The ID3 algorithm is run recursively on the non-leaf branches, until all data is classified.



Advantages of using ID3

- Understandable prediction rules are created from the training data.
- Builds the fastest tree.
- Builds a short tree.
- Only need to test enough attributes until all data is classified.
- Finding leaf nodes enables test data to be pruned, reducing number of tests.
- Whole dataset is searched to create tree.



Disadvantages of using ID3

- Data may be over-fitted or over-classified, if a small sample is tested.
- Only one attribute at a time is tested for making a decision.
- Classifying continuous data may be computationally expensive, as many trees must be generated to see where to break the continuum.



Example: PlayTennis





































































Example: The Simpsons

[illegible]

Per				
 Homer	0"	250	36	M
 Marge	10"	150	34	F
 Bart	2"	90	10	M
 Lisa	6"	78	8	F
 Maggie	4"	20	1	F
 Abe	1"	170	70	M
 Selma	8"	160	41	F
 Otto	10"	180	38	M

 Krusty	6"	200	45	M
---	----	-----	----	---

 Comic 8" 290 38 ?

$$p \log_2 p - \sum p \log_2 p$$

Entropy $S_{22}()$ log log

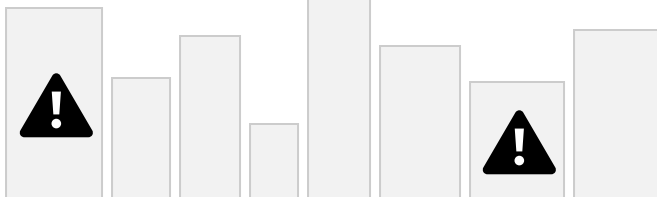
$$= - \sum p \log_2 p$$

$$= - \left(\frac{4}{9} \log_2 \frac{4}{9} + \frac{5}{9} \log_2 \frac{5}{9} \right)$$

$$\text{Entropy}(4\mathbf{F}, 5\mathbf{M}) = -(4/9)\log_2(4/9) - (5/9)\log_2(5/9)$$

$$= \mathbf{0.9911}$$

yes no
Hair Length <= 5?



child sets)

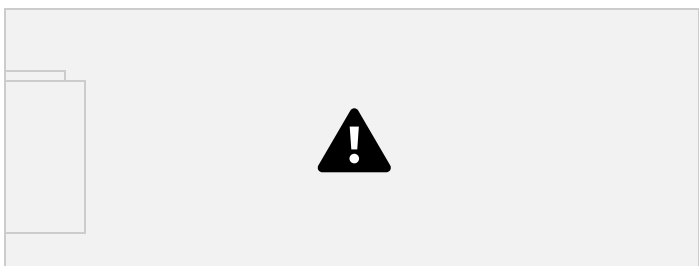


Let us try
splitting on
Hair length

$$Gain(A) = E(Current\ set) - \sum E(all$$

$$Gain(Hair\ Length \leq 5) = 0.9911 - (4/9 * 0.8113 + 5/9 * 0.9710) = 0.0911$$

$\frac{p}{n} \log \frac{p}{n} + \frac{n-p}{n} \log \frac{n-p}{n}$
Entropy $S_{2.2}() \log \log$



yes no
Weight <= 160?

$$= - \sum_{p,n} p_n \log_2 p_n$$

$$\begin{matrix} + & + \\ p & n \\ p & n \end{matrix} \quad \begin{matrix} + & + \end{matrix}$$

$$\begin{aligned} Entropy(4\text{F}, 5\text{M}) &= -(4/9)\log_2(4/9) - (5/9)\log_2(5/9) \\ &= 0.9911 \end{aligned}$$



child sets)

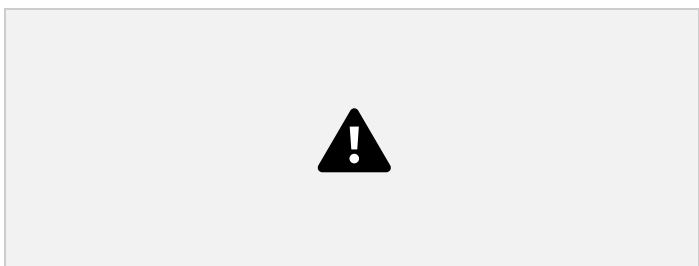


Let us try
splitting on
Weight

$$Gain(A) = E(Current\ set) - \sum E(all$$

$$Gain(Weight \leq 160) = 0.9911 - (5/9 * 0.7219 + 4/9 * 0) = 0.5900$$

$p \log_2 p - \sum p_i \log_2 p_i$
Entropy $S_{2.2}() \log \log$



$$= - \sum_{p,n} p \log p$$

$$\begin{matrix} + & + \\ p & n \\ p & n \end{matrix} \quad \begin{matrix} + & + \end{matrix}$$

$$\begin{aligned} \text{Entropy}(4\text{F}, 5\text{M}) &= -(4/9)\log_2(4/9) - (5/9)\log_2(5/9) \\ &= 0.9911 \end{aligned}$$

yes no
age <= 40?



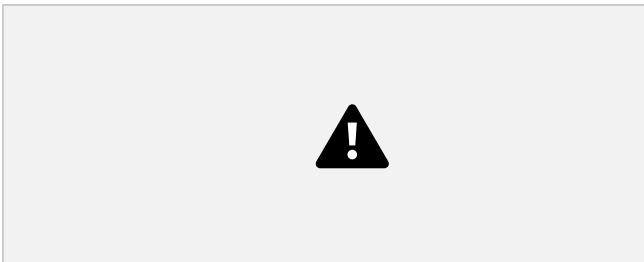
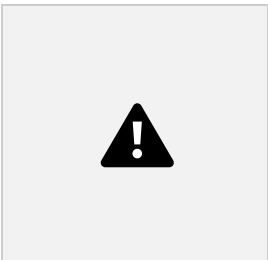
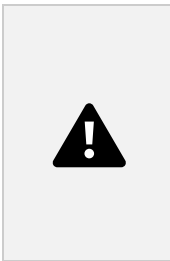
child sets)



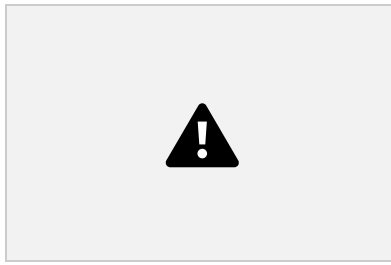
Let us try
splitting on
Age

$$Gain(A) = E(Current\ set) - \sum E(all$$

$$Gain(Age \leq 40) = 0.9911 - (6/9 * 1 + 3/9 * 0.9183) = 0.0183$$



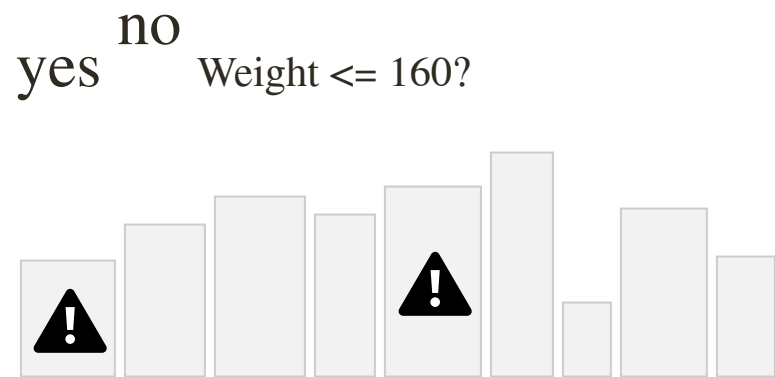


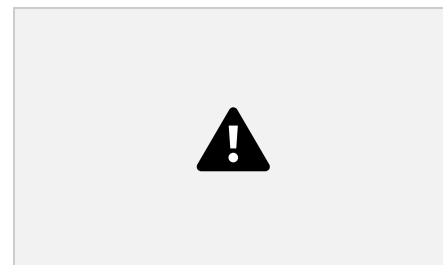
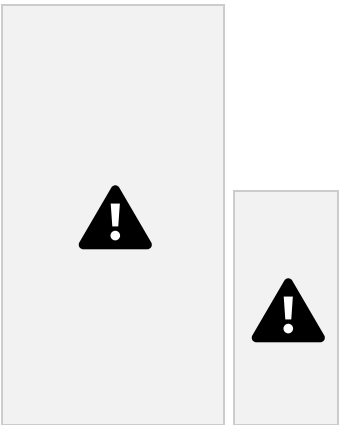


Of the 3 features we had, *Weight* was best. But while people who weigh over 160 are perfectly classified (as males), the under 160 people are not perfectly classified... So we simply recurse!

This time we find that we can split on *Hair length*, and we are done!

yes no Hair Length ≤ 2 ?









We need don't need to keep the data

around, just the test conditions. **Weight ≤ 160 ?** yes no

How would

these people be
classified?

Hair Length ≤ 2 ?

Male **Female**

Male

yes no

It is trivial to convert Decision

Trees to rules... **Weight ≤ 160 ?** yes no

Female



Rules to Classify Males/Females

If *Weight* **greater than** 160, classify as **Male**

Elseif *Hair Length* **less than or equal to** 2, classify as **Male** Else classify as **Female**



Male