One R Classifier.

One Rule Classifier

It is simple, accurate classification algorithm.

⇒ It generates one rule for each predictor in the data

⇒ Selects the rule with smallest total error as its "One rule"

⇒ To create a rule, construct a frequency table for each predictor against the target.

→ produces rules only slightly less accurate than state-of-art classification algorithms

Algorithm

For each predictor,

for each value of that predictor, make a rule as follows;

Count how often each value of target (class) appears

Find the most frequent class

Make the rule assign that class to this value of the predictor

Calculate total error of rules of each predictor

Choose the predictor with smallest total error.

Finding the best predictor with the smallest total error using OneR algorithm based on related frequency tables.

Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	yes
Sunny	Mild	High	False	yes
Sunny	Cool	Normal	False	yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	yes
Sunny	Mild	Normal	False	yes
Rainy	Mild	Normal	True	yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Which One is the best predictor?

		Play Golf	
		yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

		Play Golf	
		yes	No
Temp	Hot	2	2
	Mild	4	2
	Cool	3	1

		yes	No
Humidity	High	3	4
	Normal	6	1

		yes	No
Windy	False	6	2
	True	3	3

Frequeny Tables

The best predictor is :

IF Outlook = Sunny THEN PlayGolf = yes

IF Outlook = Overcast THEN playGolf = yes

IF Outlook = Rainy THEN PlayGolf = NO

Predictors Contribution

The total error calculated from the frequeny tables is the measure of predictor contribution

A low error means higher contribution to the predictability of the model

Model Evalution.

Confusion matrix shows significant predictability power

Confusion Matrix		Play Golf yes	No		
OneR	yes	7	2	Positive Predictive Value	0.78
	NO	2	3	Negative Predictive Value	0.60
		Sensitivity	Specificity		Accuracy =
		0.78	0.60		0.71

Confusion Matrix.

	Disorder	No Disorder
Positive Test Result	True Positive (TP)	False Positive (FP)
Negative Test Result	False Negative (FN)	True Negative (TN)

$$\text{Sensitivity} = TP / (TP + FN)$$

$$\text{Specificity} = TN / (TN + FP)$$

$$\text{Positive Predictive Value (PPV)} = TP / TP + FP$$

$$\text{Negative Predictive Value (NPV)} = TN / (FN + TN)$$

$$\text{Accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

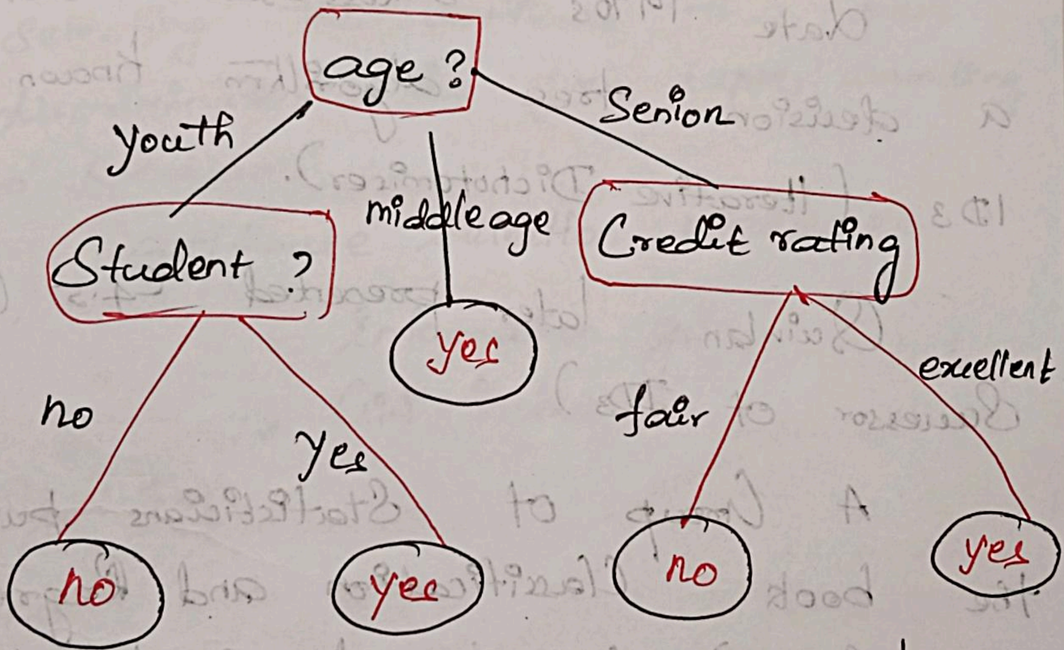
Decision Trees

Decision tree Induction is the learning of decision trees from class-labeled training tuples.

It is a flow chart like structure, each internal node denotes a test on an attribute

each branch represents an outcome of the test

leaf node holds a class label.



A decision tree for the concept buys-computer

"How are decision tree used for classification?"

Decision trees can be easily be converted to classification rules.

Why are decision tree classifier So popular?

- It does not require any domain knowledge
- handle multidimensional data.
- Simple and fast
- It have good accuracy

Decision Tree Induction

Late 1970s J. Ross Quinlan developed a decision tree algorithm known as ID3 (Iterative Dichotomiser).

Quinlan later presented C4.5 (a Successor of ID3)

A Group of Statisticians published the book Classification and Regression Trees (CART), which described the generation of binary decision trees.

ID3, C4.5, and CART adopt a greedy (ie nonbacktracking) approach in which decision trees are constructed in top-down recursive divide and conquer manner.

Strategy of algorithm as follows

→ Three parameters

1. D :- Data partition (set of training tuples and associated class labels.

2. attribute_list :- list of attributes describing tuple.

3. Attribute_Selection_method :

heuristic procedure for selecting the attribute that best discriminates the given tuples according to class.

attribute selection measures

* information gain
* Gini index

Algorithm

Method :

Create a Node N ;

if tuples in D are all of the same class C, then
 return N as leaf labeled with class C ;

if attribute_list is empty then
 return N as a leaf node labeled with majority class in D

apply Attribute_Selection_method (D, attribute_list)
 to find "best" splitting_criterion;

label node N with Splitting_criterion;
if splitting_attribute is discrete-valued and
multiway splits allowed then
attribute_list ← attribute_list − splitting_attribute;
for each outcome j of splitting_criterion
let D_j be the set of data tuples in D
if D_j is empty then
attach a leaf labeled with class in D to Node N
else attach the node returned by Generate_decision_tree(D_j, attribute_list)
end for
return N.

Example

Entropy: Common way to measure impurity in dataset

Information Gain: ~~decrease~~ decline in entropy after the dataset is split also called as Entropy Reduction.

Day Outlook Temp Humidity Wind Play Tennis

Information Gain:

ID3 uses information gain as its attribute selection measure

The attribute with highest information gain is choosen as splitting attribute for node N.

It minimizes the expected number of tests needed to classify a given tuple

The expected information needed to classify a tuple in D is given by

$$\boxed{Info(D) = -\sum_{i=1}^{m} P_i \log_2(P_i)}$$

P_i - nonzero probability estimated by $|C_{i,D}| / |D|$

$\log_2$ is used because information is encoded in bits.

Info(D) is also known as entropy of D.

How much more information would we still need to arrive at an exact classification?

$$\boxed{Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j)}$$

The term $\frac{|D_j|}{|D|}$ acts as the weight of the j^{th} partition.

The smaller the expected information required, the greater the purity of partition

Information Gain is the difference between the original information requirement and the new requirement.
It

i.e.

$$\boxed{Gain(A) = Info(D) - Info_A(D).}$$

It tells how much would be gained by branching on A.

Example.

S NO	Outlook	Temperature	Humidity	Windy	Play Tennis
1.	Sunny	Hot	High	Weak	No
2.	Sunny	Hot	High	Strong	No
3.	Overcast	Hot	High	Weak	yes
4.	Rainy	Mild	High	Weak	yes
5.	Rainy	Cool	Normal	Weak	yes
6.	Rainy	Cool	Normal	Strong	No
7.	Overcast	Cool	Normal	Strong	yes
8.	Sunny	Mild	High	Weak	No
9.	Sunny	Cool	Normal	Weak	yes
10.	Rainy	Mild	Normal	Weak	yes
11.	Sunny	Mild	Normal	Strong	yes
12.	Overcast	Mild	High	Strong	yes
13.	Overcast	Hot	Normal	Weak	yes
14.	Rainy	Mild	High	Strong	No.

Make a Decision tree that predicts whether Tennis will be played on the day?

Sol: Total = 14

Play Tennis yes = 9 No = 5.

Calculate $\text{Info}(D) = -\sum_{i=1}^{m} P_i \log_2(P_i)$

$$\text{Info}(D) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14}$$

$$\boxed{\text{Info}(D) = 0.940}$$

Attribute "Outlook"

	Yes	No
Sunny	2	3
Rainy	3	2
Overcast	4	0

$\text{Info}_{Outlook}(D)$

$$\text{Info}_{outlook}(D) = \frac{5}{14}\left(-\frac{2}{5}\log_2 \frac{2}{5} - \frac{3}{5}\log_2 \frac{3}{5}\right)$$

$$+ \frac{15}{14}\left(-\frac{3}{5}\log_2 \frac{3}{5} - \frac{2}{5}\log_2 \frac{2}{5}\right)$$

$$+ \frac{4}{14}\left(-\frac{4}{4}\log_2 \frac{4}{4}\right)$$

$$= 0.693$$

Attribute "temperature"

	Y	N
Hot	2	2
Mild	4	2
Cool	3	1

$$\text{Info}_{temp}(D) = \frac{4}{14}\left(-\frac{2}{4}\log_2 \frac{2}{4} - \frac{2}{4}\log_2 \frac{2}{4}\right)$$

$$+ \frac{6}{14}\left(-\frac{4}{6}\log_2 \frac{4}{6} - \frac{2}{6}\log_2 \frac{2}{6}\right)$$

$$+ \frac{4}{14}\left(-\frac{3}{4}\log_2 \frac{3}{4} - \frac{1}{4}\log_2 \frac{1}{4}\right)$$

$$= 0.911$$

"Humidity"		Y	N
	Normal	6	1
	High	3	4

$$Info_{Humi}(D) = \frac{7}{14}\left(-\frac{6}{7}\log_2\frac{6}{7} - \frac{1}{7}\log_2\frac{1}{7}\right) + \frac{7}{14}\left(-\frac{3}{7}\log_2\frac{3}{7} - \frac{4}{7}\log_2\frac{4}{7}\right)$$

$$= 0.788$$

"Windy"		Y	N
	Weak	6	2
	Strong	3	3

$$Info_{Windy}(D) = \frac{8}{14}\left(-\frac{6}{8}\log_2\frac{6}{8} - \frac{2}{8}\log_2\frac{2}{8}\right) + \frac{6}{14}\left(-\frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2\frac{3}{6}\right)$$

$$= 0.892$$

$$Gain(Outlook) = Info(D) - Info_{outlook}(D)$$
$$= 0.940 - 0.693$$
$$= 0.247$$

$$Gain(temperature) = 0.940 - 0.911$$
$$= 0.029$$

$$Gain(Humidity) = 0.940 - 0.788$$
$$= 0.152$$

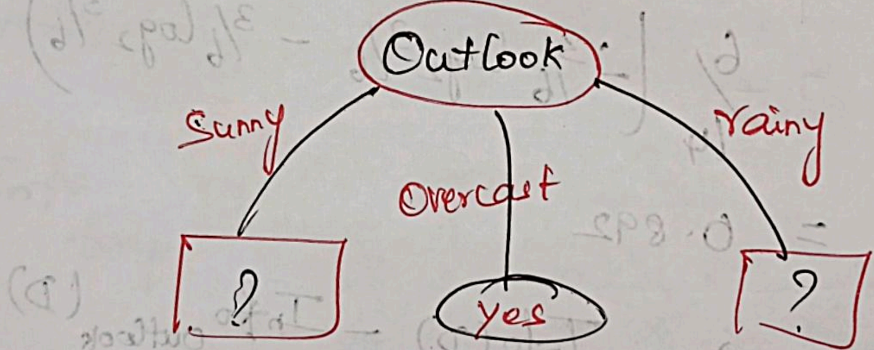
~~Inf~~

$$Gain\ (windy) = 0.940 - 0.892$$

$$= 0.048$$

Attribute	Gain
Outlook	0.247
Temperature	0.029
Humidity	0.152
Windy	0.048

Root Node : Outlook.



Consider the Outlook table.

Outlook	Temp	Humi	Windy	Play Tennis
	Hot	High	Weak	No
	Hot	High	Strong	No
Sunny	Mild	High	Weak	No
	Cool	Normal	Weak	yes
	Mild	Normal	Strong	yes

Outlook	Temp	Humi	Windy	Play Tennis
Rainy	Mild	High	Weak	Yes
	Cool	Normal	Weak	Yes
	Cool	Normal	Strong	No
	Mild	Normal	Weak	Yes
	Mild	High	Strong	No

Sunny Yes 2 No 3

$$\text{Info}(\text{Sunny}) = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}$$

$$\Rightarrow 0.971$$

$$\cancel{\text{Info}_{\text{Humi}}(\text{Sunny}) = \frac{5}{5}\left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right)}$$

Y N
2 3

Humidy

	Y	N
H	0	3
N	2	0

$$= \frac{3}{5}\left(-\frac{3}{3}\log_2\frac{3}{3}\right) + \frac{2}{5}\left(-\frac{2}{2}\log_2\frac{2}{2}\right)$$

$$\text{Info}_{\text{Hum}}(\text{Sunny}) = 0.$$

Windy

	Y	N
Weak	1	2
Strong	1	1

$$\text{Info}_{\text{Windy}}(\text{Sunny}) = \frac{3}{5}\left(-\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}\right)$$

$$+\frac{2}{5}\left(-\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}\right)$$

$$= 0.951$$

	Hot	Cool	mild
Y	0	1	1
N	2	0	1

Info (Sunny) temp

$$\Rightarrow \frac{2}{5}\left(-\frac{2}{2}\log_2\frac{2}{2}\right)+\frac{1}{5}\left(-\frac{1}{1}\log_2\frac{1}{1}\right)$$

$$+\frac{2}{5}\left(-\frac{1}{2}\log_2\frac{1}{2}-\frac{1}{2}\log_2\frac{1}{2}\right)$$

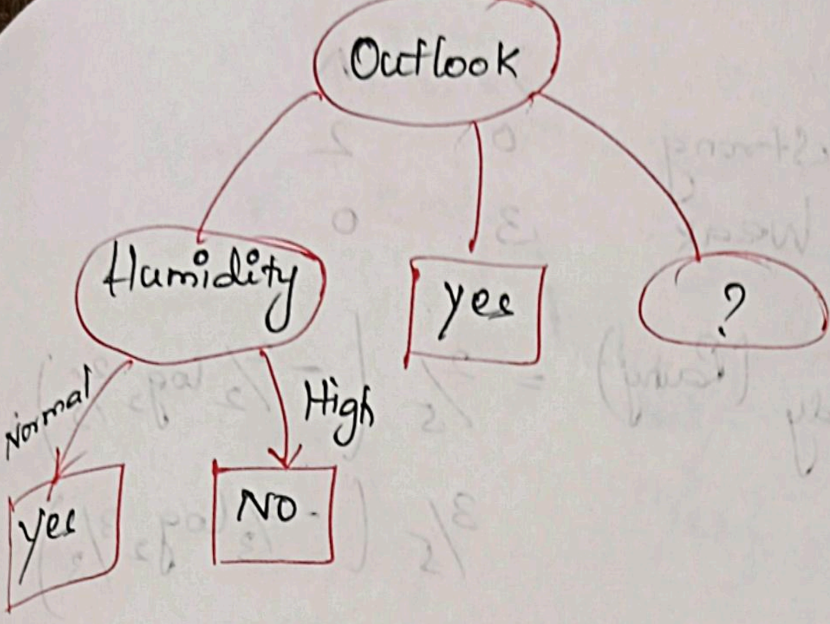
$\Rightarrow 0.4$

Gain (Temp) $= 0.971 - 0.4$
$= 0.571$

Gain (Humi) $= 0.971 - 0$
$= 0.971$

Gain (Windy) $= 0.971 - 0.951$
$= 0.02.$

Attribute	Gain
Temperature	0.571
Humidity	$\boxed{0.971}$
Windy	0.02



Rainy. Yes 3 No 2

$$\text{Info}(\text{Rainy}) = \left(-\tfrac{3}{5}\log_2 \tfrac{3}{5}\right) - \tfrac{2}{5}\log_2 \tfrac{2}{5}$$

$$\Rightarrow 0.971$$

Temp		Y	N
	Mild	2	1
	Cool	1	1

$$\text{Info}_{temp}(\text{Rainy}) = \tfrac{3}{5}\left(-\tfrac{2}{3}\log_2 \tfrac{2}{3} - \tfrac{1}{3}\log_2 \tfrac{1}{3}\right) + \tfrac{2}{5}\left(-\tfrac{1}{2}\log_2 \tfrac{1}{2} - \tfrac{1}{2}\log_2 \tfrac{1}{2}\right)$$

$$= 0.951$$

Humidity	Y	N
High	1	1
Normal	2	1

$$\text{Info}_{Humi}(\text{Rainy}) = \tfrac{2}{5}\left(-\tfrac{1}{2}\log_2 \tfrac{1}{2} - \tfrac{1}{2}\log_2 \tfrac{1}{2}\right) + \tfrac{3}{5}\left(-\tfrac{2}{3}\log_2 \tfrac{2}{3} - \tfrac{1}{3}\log_2 \tfrac{1}{3}\right)$$

$$= 0.951$$

Windy	Y	N
Strong	0	2
Weak	3	0

$$\text{Info}_{\text{Windy}}(\text{Rainy}) = \frac{2}{5}\left(-\frac{2}{2}\log_2\frac{2}{2}\right) + \frac{3}{5}\left(-\frac{3}{3}\log_2\frac{3}{3}\right)$$

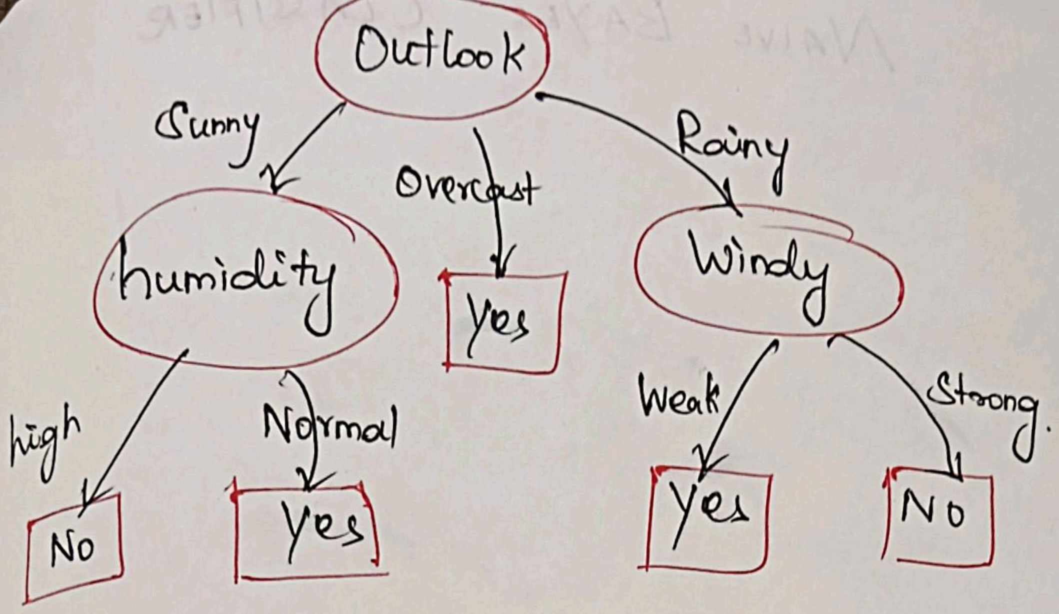
$$= 0.$$

$$\text{Gain}(\text{Windy}) = 0.971 - 0$$
$$= 0.971$$

$$\text{Gain}(\text{Temp}) = 0.971 - 0.951$$
$$= 0.020$$

$$\text{Gain}(\text{Humidy}) = 0.971 - 0.951$$
$$= 0.020.$$

Attribute	Gain
Humidity	0.02
Windy	0.971
Temperature	0.02


Outlook
Sunny
Overcast
Rainy
humidity
Yes
Windy
high
Normal
Weak
Strong.
No
Yes
Yes
NO