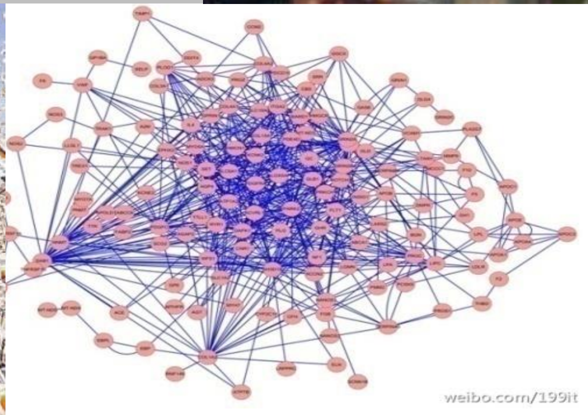




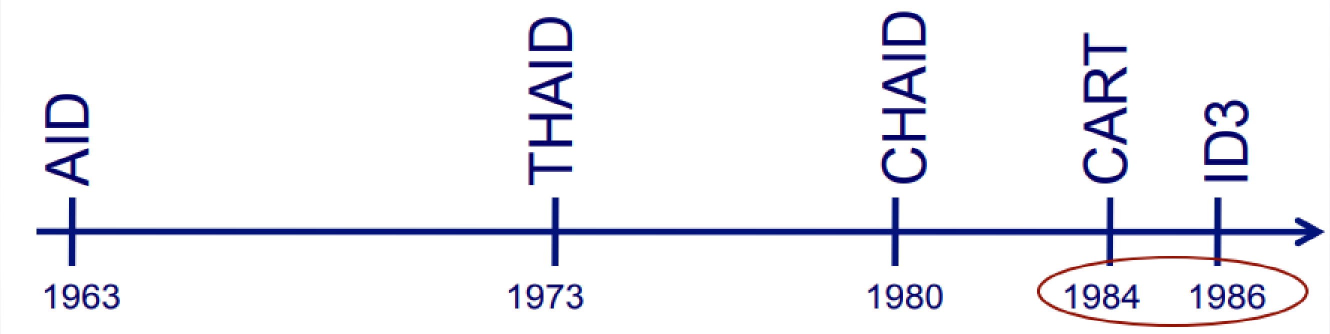
Chapter 2 Decision Tree

第二章 决策树

2025 Autumn

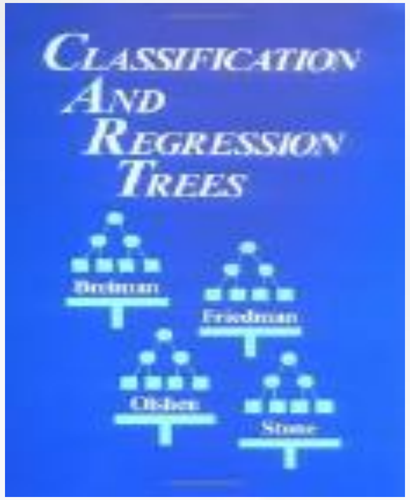
Lei Sun

决策树的学习历史

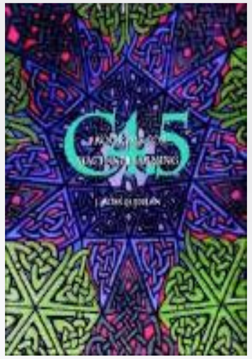


many DT variants have been developed since CART and ID3

CART developed by Leo Breiman, Jerome Friedman, Charles Olshen, R.A. Stone



ID3, C4.5, C5.0 developed by Ross Quinlan



http://www.rulequest.com/Personal

Season	Weather	A_Control	Airline	DelayOrNot
Summer	Sunny	no	CZ	no
Summer	Sunny	no	CA	no
Autumn	Sunny	no	CZ	yes
WinterSpring	RainyOrSnowy	no	SH	yes
WinterSpring	Cloudy	Yes	CZ	yes
WinterSpring	Cloudy	yes	CA	no
Autumn	Cloudy	yes	CA	yes
Summer	RainyOrSnowy	no	SH	no
Summer	Cloudy	Yes	CZ	yes
WinterSpring	RainyOrSnowy	yes	CZ	yes
Summer	RainyOrSnowy	yes	CA	yes
Autumn	RainyOrSnowy	No	SH	yes
Autumn	Sunny	Yes	CZ	yes
WinterSpring	RainyOrSnowy	no	CA	no

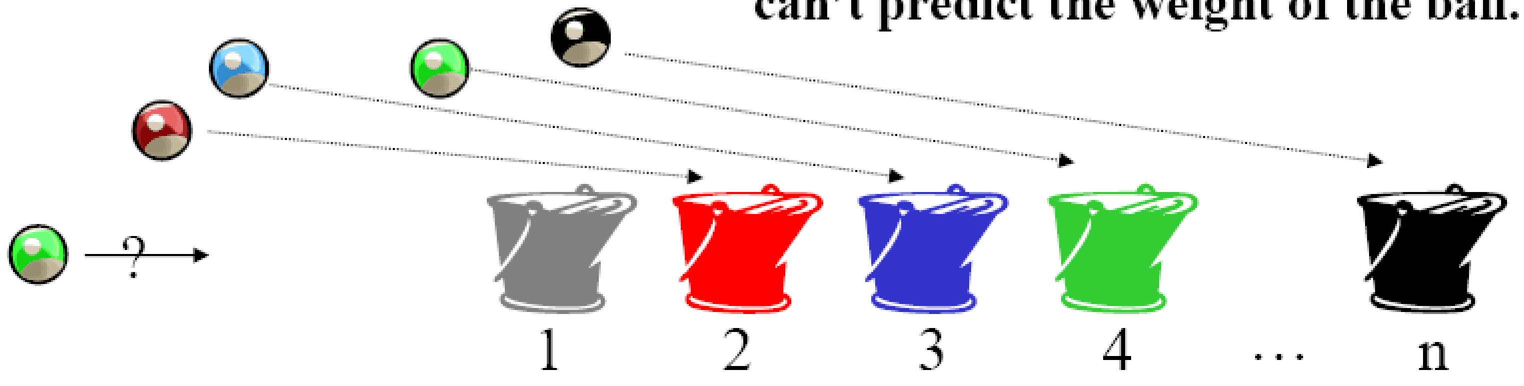
01

Classification

- ✓ The goal of data classification is to organize and categorize data in distinct classes
  分类的目的是对数据进行有序整理和分类

-- A model is first created based on the data distribution(training data).
首先根据训练数据的数据分布构建模型
-- The model is then used to classify testing data for evaluation.
随后模型在测试集上进行评估
-- Given the model, a class can be predicted for new data
根据该模型可以对新的数据进行预测

With classification, I can predict in which bucket to put the ball, but I can't predict the weight of the ball.



01

Classification

three-step process

① Model construction (learning)
模型的构建

每个元组被假定属于预定义的类别，这个类别是由其中的一个名为类标号的属性决定的

Each tuple(元组）is assumed to belong to a predefined class, as determined by one of the attributes, called the class label(类标号).

用于构建模型的所有元组叫做训练集

The set of all tuples used for construction of the model is called training set. （训练集）

Classification

three-step process

② Model Evaluation
模型评估

基于测试集对模型的准确率进行估算

Estimate accuracy rate (准确率） of the model based on a test set（测试集）

将测试样本的标签与分类得到的结果进行比较

The known label of test sample is compared with the classified result from the model

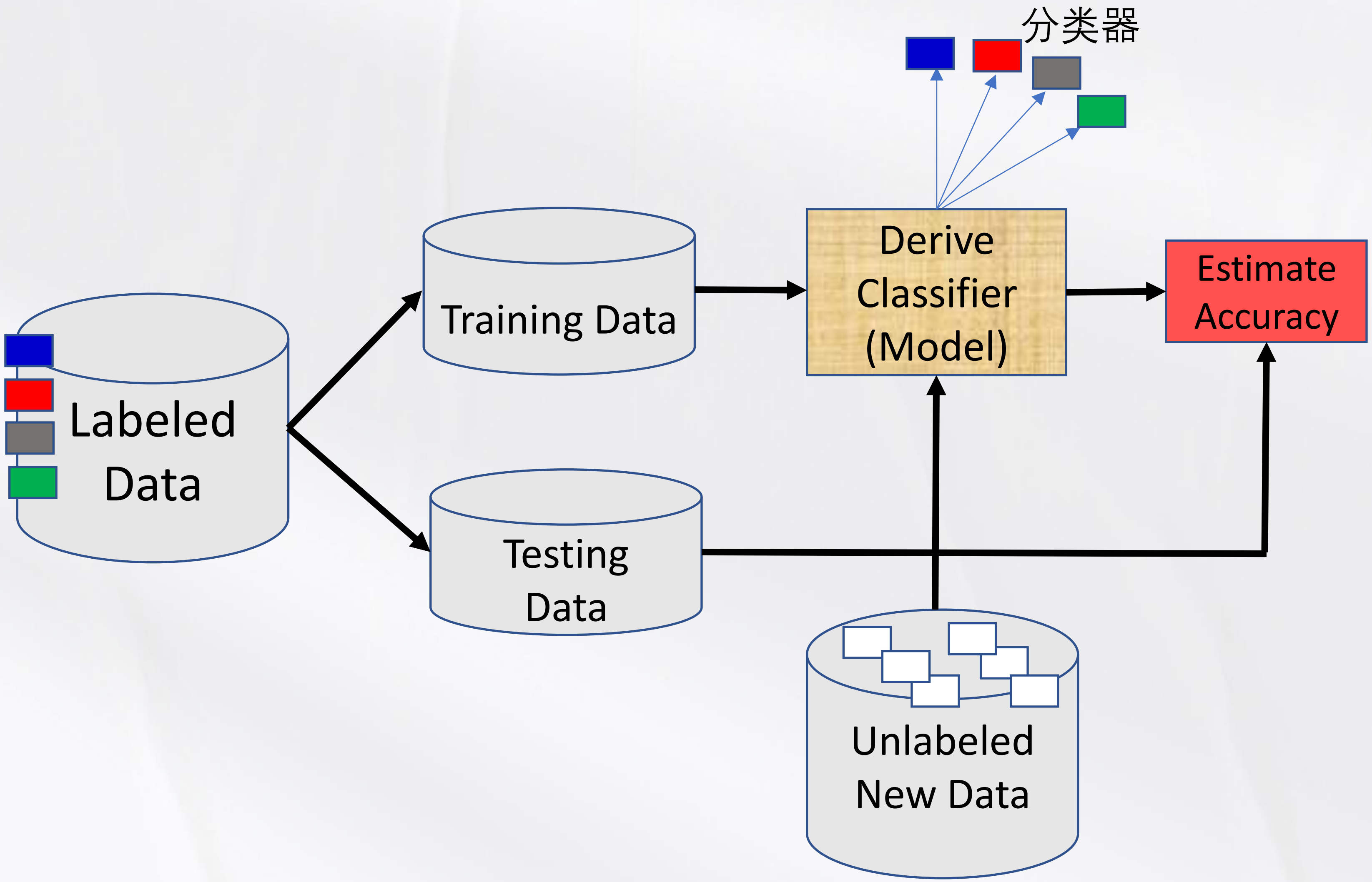
准确率是指模型对测试集样本进行正确分类的样本所占的百分比。

Accuracy rate is the percentage of test set samples that are correctly classified by the model

Test set is independent of training set otherwise over-fitting（过度拟合）will occur

③ Prediction: predict the class label of new data.

Classification



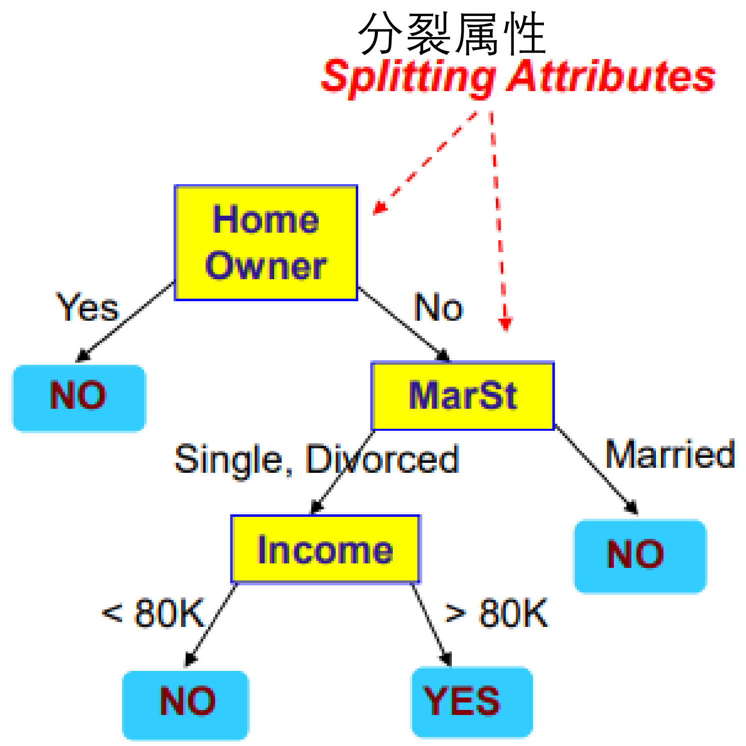
Decision Tree（决策树）

There could be more Than one tree based on the same data!!

Which is the best?

	categorical	categorical	continuous	class
ID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
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

Decision Tree（决策树）

Training examples 9yes/5no

- Try to understand when to play
- Divide & Conquer(分而治之)
  – Split into subsets
  – Are they pure (纯)?(all yes or no)
  – If yes: stop
  – If not: repeat
- See which subset the new data falls into

Day	Outlook	Humidity	Wind	Play
D1	Sunny	High	Weak	No
D2	Sunny	High	Strong	No
D3	Cloudy	High	Weak	Yes
D4	Rain	High	Weak	Yes
D5	Rain	Normal	Weak	Yes
D6	Rain	Normal	Strong	No
D7	Cloudy	Normal	Strong	Yes
D8	Sunny	High	Weak	No
D9	Sunny	Normal	Weak	Yes
D10	Rain	Normal	Weak	Yes
D11	Sunny	Normal	Strong	Yes
D12	Cloudy	High	Strong	Yes
D13	Cloudy	Normal	Weak	Yes
D14	Rain	High	Strong	No
D15	Rain	High	Week	?

Decision Tree

Outlook

9 yes / 5 no

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

Cloudy

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

4 yes / 0 no

Pure subset

Rain

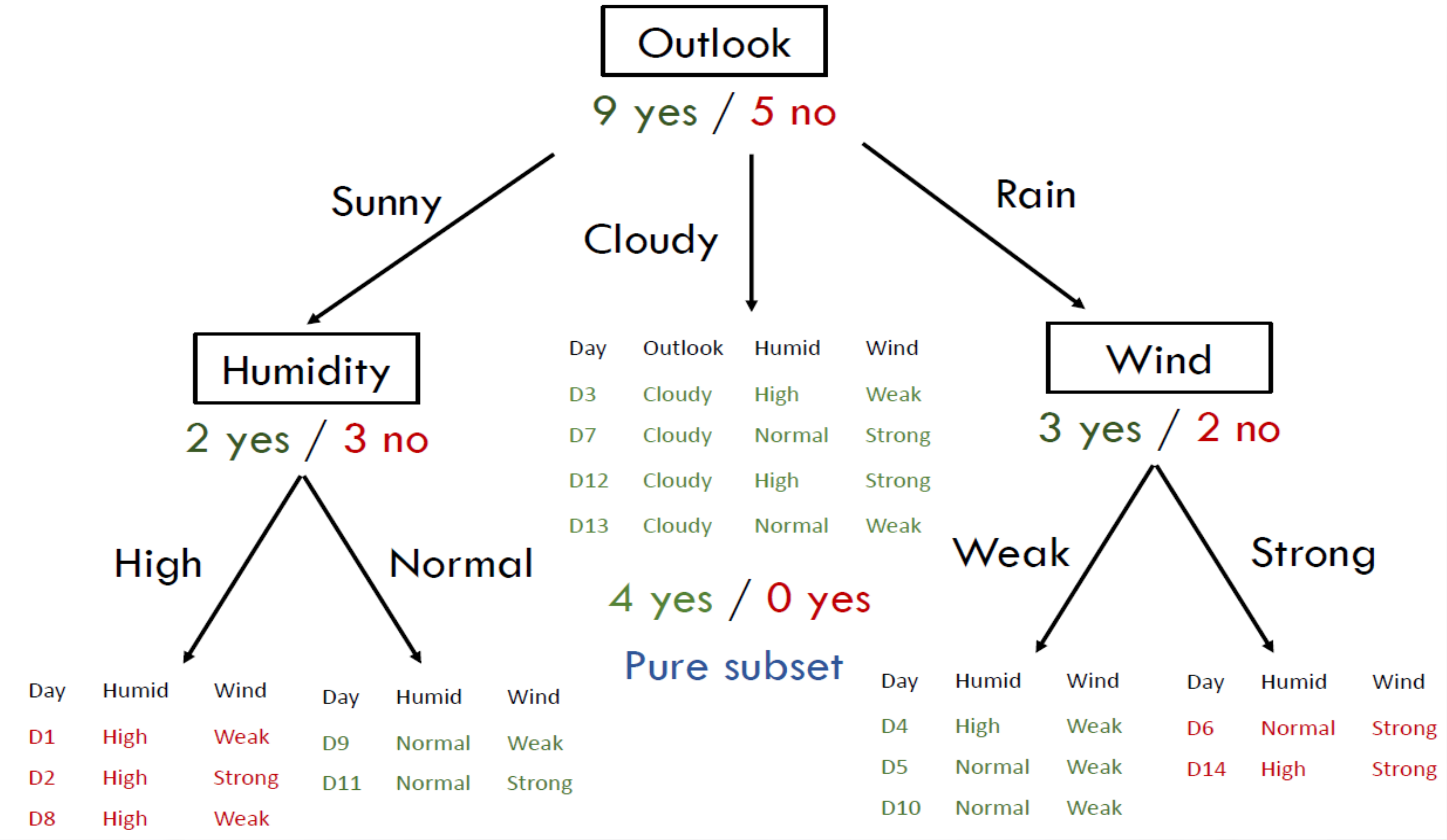
Day	Outlook	Humid	Wind
D4	Rain	High	Weak
D5	Rain	Normal	Weak
D6	Rain	Normal	Strong
D10	Rain	Normal	Weak
D14	Rain	High	Strong

3 yes / 2 no

Split further

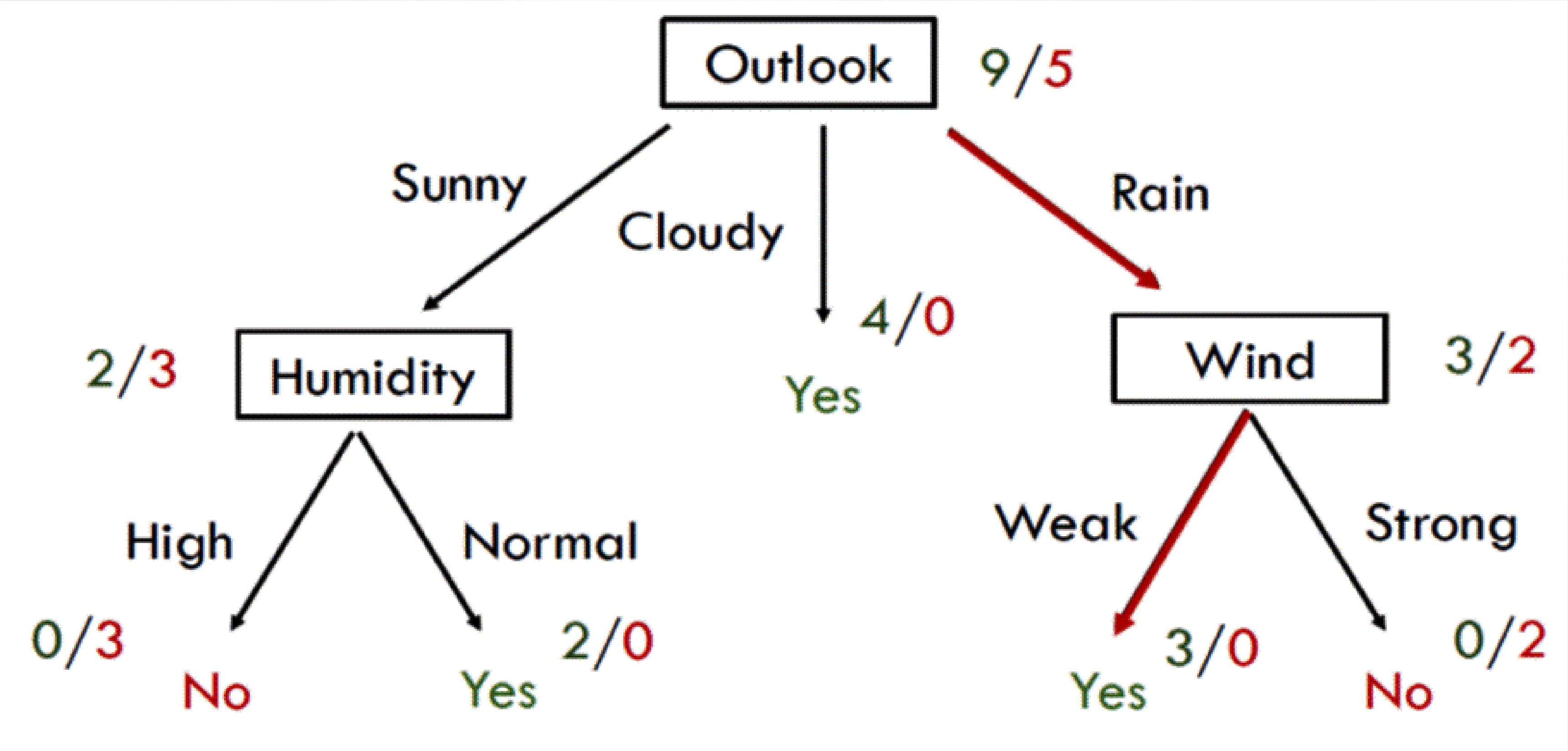
02

Decision Tree（决策树）



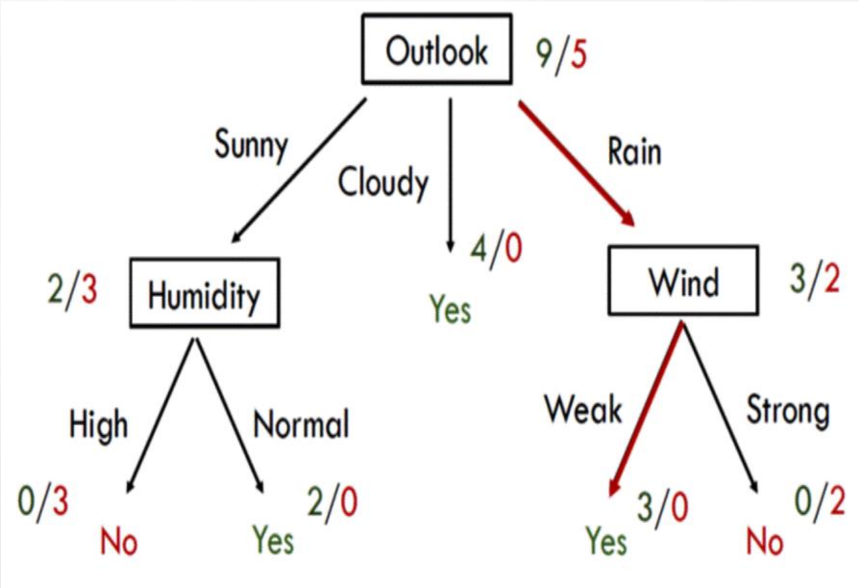
Decision Tree（决策树）

Final tree



Day	Outlook	Humidity	Wind	play
15	rain	high	weak	?

Decision Tree（决策树）



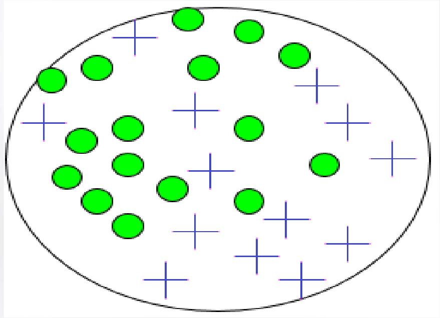
Structure of a Decision Tree

- ✓ **Root Node:** Represents the entire dataset and the initial decision to be made.
- ✓ **Internal Nodes:** A node that symbolizes a choice regarding an input feature. Each internal node has one or more branches.
- ✓ **Branches:** Represent the outcome of a decision or test (attribute value), leading to another node.
- ✓ **Leaf Nodes:** Represent the final decision or prediction(class label). No further splits occur at these nodes.

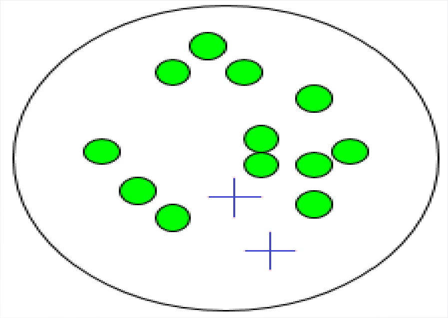
Decision Tree（决策树）

Impurity

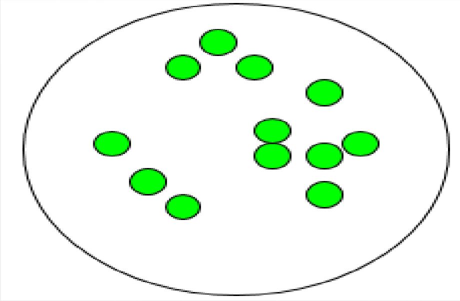
Very impure group



Less impure



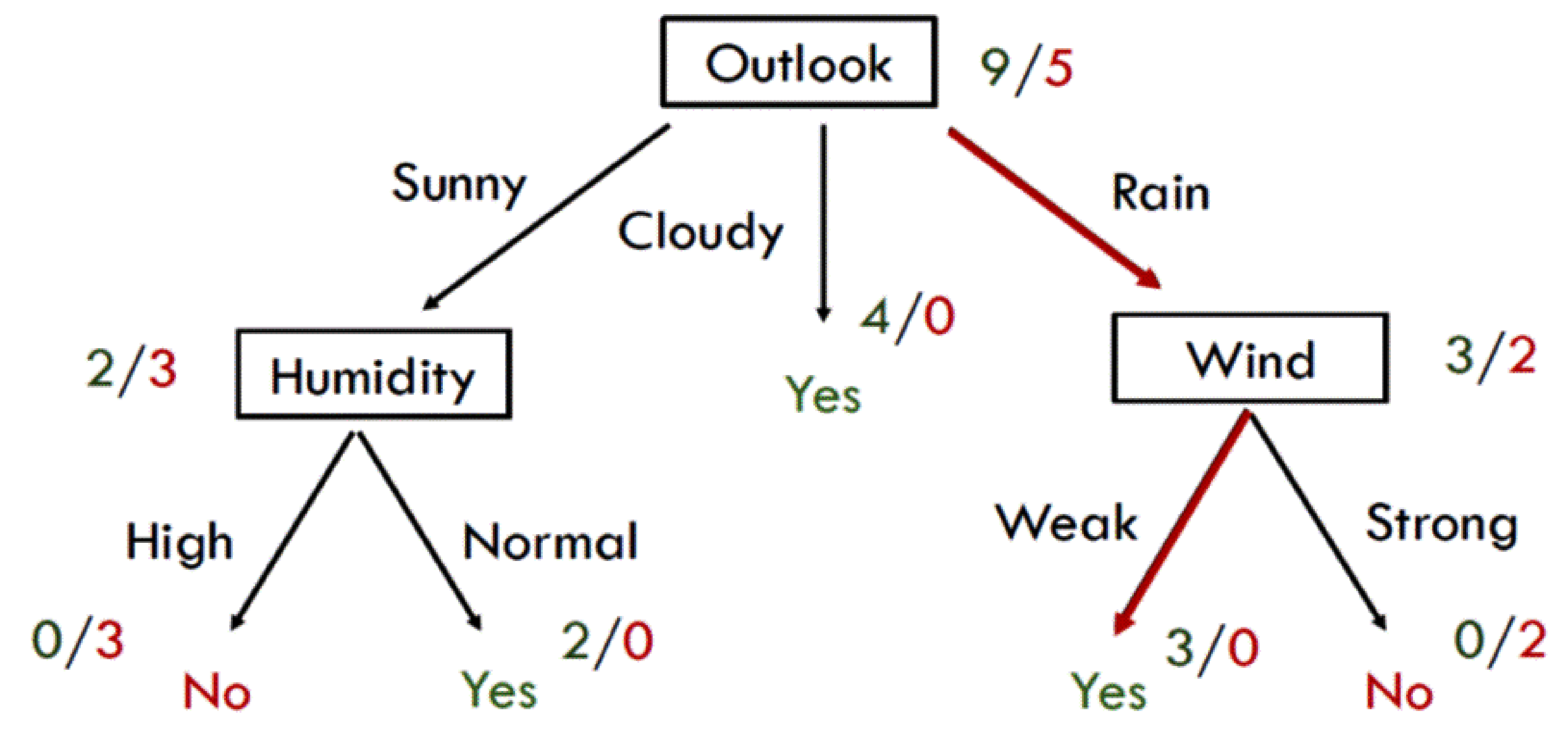
Minimum impure



- Want to measure "purity" of the split
  - – More certain about Yes/No after split
    - Pure set (4 yes/0 no or 0 yes/4 no) ⇒ completely certain (100%)
    - Impure (3 yes/3 no) ⇒ completely uncertain (50%)

Decision Tree（决策树）

Which attribute is spit on?



02

Decision Tree（决策树）

✓ **Impurity Measures (Metrics for Splitting):**

-- Information gain (Entropy)	ID3
-- Information Gain Ratio	C4.5
-- Gini Impurity	CART

three fundamental criteria to measure the quality of a split in DT.

03

Information Gain（信息增益）

Information Entropy（信息熵） (C.E.Shannon,1948)

- ✓ Information transfer is achieved by a transfer system:

information source
(send)

channel

information sink
(receive)

- ✓ P(U|V): Probability that a source sends a message *u* under the condition that the sink receives the message *v*. Information transfer has random error.

$$\begin{bmatrix} P(u_1 \mid v_1) & P(u_2 \mid v_1) & \ldots.P(u_r \mid v_1) \\ P(u_1 \mid v_2) & P(u_2 \mid v_2) & \ldots.P(u_r \mid v_2) \\ \ldots & \ldots & \ldots \quad . \\ P(u_1 \mid v_q) & P(u_2 \mid v_q) & \ldots P(u_r \mid v_q) \end{bmatrix}$$

$$\sum P(u_i \mid v_j) = 1 \quad (i = 1,2,\ldots,r)$$

information source
(send) U
$u_1, u_2, ..u_r$

channel
P(U|V)

information sink
(receive) V
$v_1, v_2, ..v_q$

Information Gain（信息增益）

- priori uncertainty(先验不确定性): Before transfer, receiver can not judge what status the sender is or what information the source will sent. P(U)
- posterior uncertainty (后验不确定性): After transfer, information sink received the information from source. Priori uncertainty is partly or thoroughly eliminated. P(U|V)
- P(U)= P(U|V): The sink did not received any information.
- P(U|V)=0: The information is thoroughly received by sink. Priori uncertainty is thoroughly eliminated.
- Generally, disturbance may damage information transfer. Priori uncertainty is impossibly thoroughly eliminated.
- Information is used to eliminate the random uncertainty.
- The information value:

$$I(u_i) = \log_2 \frac{1}{P(u_i)} = -\log_2 P(u_i)$$

Information Gain（信息增益）

- Information entropy: mathematical expectation of information quality.

  A measure of uncertainty associated with a random variable.

  mathematical expectation: sum of all the products(积) of all possible values of discrete(离散) random variables and their probabilities .

$$\mathbf{E(X)=Sum(p(x)*x)} \qquad Ent(U) = \sum_{i=1}^{C} \mathrm{P}(u_i)\log_2 \frac{1}{P(u_i)} = -\sum_{i=1}^{C} \mathrm{P}(u_i)\log_2 P(u_i)$$

Consider all possible values of the random variable, i.e. the expectation information quantity brought by all possible events.

average uncertainty before information is sent by source.

Information entropy

Information Gain（信息增益）

✓ Information gain

When the sink receives the information v_i, the probability of the source sending the information U is $P(U|v_i)$, and the average uncertainty of the source is as bellow: Posterior entropy(后验熵)

$$Ent(U|v_j)=\sum_i P(u_i|v_j)\log_2\frac{1}{P(u_i|v_j)}=-\sum_i P(u_i|v_j)\log_2 P(u_i|v_j)$$

mathematical expectation of posterior entropy(后验熵） (Conditional Entropy): after sink received information V(a random variable),

The average uncertainty of source U still existed .

$$Ent(U|V)=\sum_j P(v_j)\sum_i P(u_i|v_j)\log_2\frac{1}{P(u_i|v_j)}$$

$$=\sum_j P(v_j)(-\sum_i P(u_i|v_j)\log_2 P(u_i|v_j))$$

Information Gain（信息增益）

✓ Information gain: $Gains(U,V) = Ent(U) - Ent(U \mid V)$

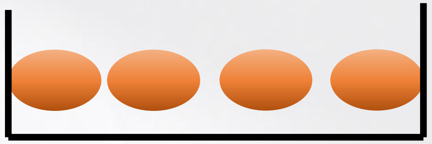
The degree to which information *V* eliminates uncertainty

反映的是信息V消除不确定性的程度

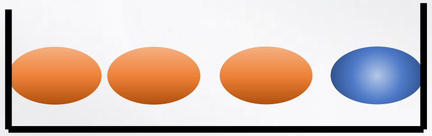
A small issue with this formula is that log(0) is undefined. Thus, when all samples belong to the same class, we would have trouble computing the Entropy. For this case, we assume $p_i \log p_i = 0$. This assumption makes sense since $lim_{x \to 0} x \log(x) = 0$.

03

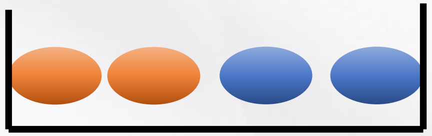
Information Gain（信息增益）



$$Ent[4(+),0(-)] = -(1 log_2 1 + 0 \log_2 0) = 0$$



$$Ent[3(+),1(-)] = -\left(\frac{3}{4} log_2 \frac{3}{4} + \frac{1}{4} \log_2 \frac{1}{4}\right) = 0.811$$



$$Ent[2(+),2(-)] = -\left(\frac{1}{2} log_2 \frac{1}{2} + \frac{1}{2} \log_2 \frac{1}{2}\right) = 1$$

Information entropy describes the uncertainty of random variables.
The smaller the information entropy, the higher the purity of information, and the less information there is.

A low Entropy indicates that the data labels are quite uniform.

A high Entropy means the labels are in chaos.

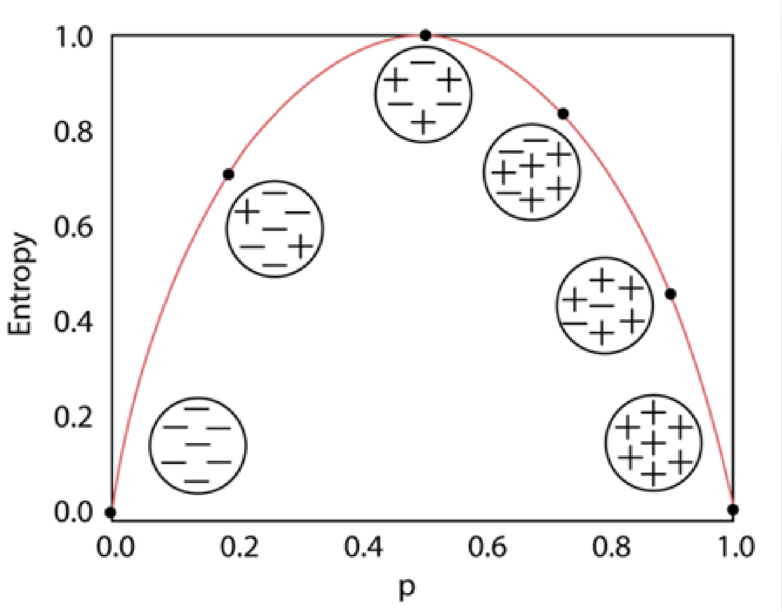
Information Gain（信息增益）

Entropy

- ✓ entropy is a measure of uncertainty (chaos)associated with a random variable
- ✓ defined as the expected number of bits required to communicate the value of the variable
- ✓ We are using base 2 here because the information is usually measured in bits.

$$H(Y) = -\sum_{y \in \text{values}(Y)} P(y)\log_2 P(y)$$

As the purity decreases,
entropy gradually increases.



entropy function for binary variable

03

Information Gain（信息增益）

Information gain
(a.k.a. mutual information 又名：互信息)

✓ Choosing splits in ID3 (Iterative Dichotomiser 3)

select the split S that most reduces the conditional entropy of Y for training set D

$$\text{InfoGain}(D,S) = H_D(Y) - H_D(Y \mid S)$$

D indicates that we're calculating probabilities using the specific sample *D*

S: attribute
Y: class label

Information Gain（信息增益）

① **Problem:** What is the entropy of the examples before we select a feature for the root node of the tree?

Solution: There are 14 examples: 9 positive, 5 negative. Applying the formula give

$$Ent(U) = -\frac{9}{14}log_2\frac{9}{14} - \frac{5}{14}log_2\frac{5}{14} = 0.940$$

higher entropy→ higher uncertainty
lower entropy→ lower uncertainty

Day	Outlook	Humidity	Wind	Play
D1	Sunny	High	Weak	No
D2	Sunny	High	Strong	No
D3	Cloudy	High	Weak	Yes
D4	Rain	High	Weak	Yes
D5	Rain	Normal	Weak	Yes
D6	Rain	Normal	Strong	No
D7	Cloudy	Normal	Strong	Yes
D8	Sunny	High	Weak	No
D9	Sunny	Normal	Weak	Yes
D10	Rain	Normal	Weak	Yes
D11	Sunny	Normal	Strong	Yes
D12	Cloudy	High	Strong	Yes
D13	Cloudy	Normal	Weak	Yes
D14	Rain	High	Strong	No
D15	Rain	High	Week	?

Information Gain（信息增益）

② **Problem:** What is the expected information gain if we select Outlook as the root node of the tree?

Solution: Testing Outlook yields three branches:

$$\text{Outlook} = \begin{cases} \text{Sunny} & 2+ \quad 3- \quad 5 \text{ total} \\ \text{Overcast} & 4+ \quad 0- \quad 4 \text{ total} \\ \text{Rain} & 3+ \quad 2- \quad 5 \text{ total} \end{cases}$$

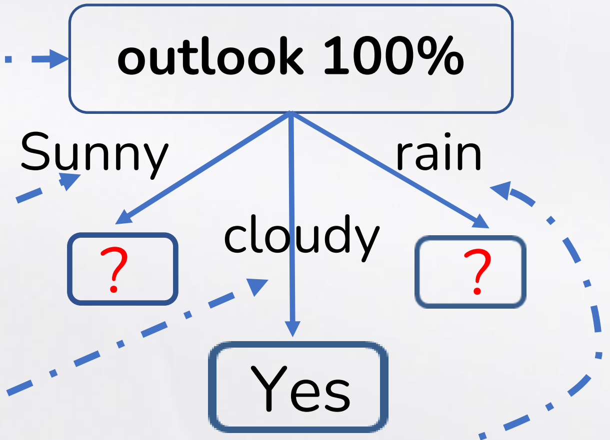
$$Ent(U|outlook = sunny) = \frac{5}{14}\left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right)$$

$$Ent(U|outlook = cloudy) = \frac{4}{14}\left(-\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4}\right)$$

$$Ent(U|outlook = rain) = \frac{5}{14}\left(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}\right)$$

Day	Outlook	Humidity	Wind	Play
D1	Sunny	High	Weak	No
D2	Sunny	High	Strong	No
D3	Cloudy	High	Weak	Yes
D4	Rain	High	Weak	Yes
D5	Rain	Normal	Weak	Yes
D6	Rain	Normal	Strong	No
D7	Cloudy	Normal	Strong	Yes
D8	Sunny	High	Weak	No
D9	Sunny	Normal	Weak	Yes
D10	Rain	Normal	Weak	Yes
D11	Sunny	Normal	Strong	Yes
D12	Cloudy	High	Strong	Yes
D13	Cloudy	Normal	Weak	Yes
D14	Rain	High	Strong	No
D15	Rain	High	Week	?

$$Ent(U) = -\frac{9}{14}log_2\frac{9}{14} - \frac{5}{14}log_2\frac{5}{14} = 0.940$$



$$Ent(U|outlook = sunny) = \frac{5}{14}(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5})$$

$$Ent(U|outlook = cloudy) = \frac{4}{14}(-\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4})$$

$$Ent(U|outlook = rain) = \frac{5}{14}(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5})$$

pick Outlook since it has the **greatest** expected information gain.

$$Gain(outlook) = \mathrm{Ent(U)} - \mathrm{Ent}(U_{outlook}) = 0.246bits$$

features	Entropy	Gain
outlook	0.694	0.246
humidity	0.789	0.151
windy	0.891	0.049

Information Gain=Entropy(parent)-[entropy(children)]

Entropy drops after a decision

The more the Entropy being reduced after splitting (that is, the more the dataset being clear after splitting), the more the Information Gain.

Case 1: Outlook = **Sunny**. $+: 9, 11$ $-: 1, 2, 8$

$$Humidity = \begin{cases} High & +: \quad -: 1,2,3 \\ Normal & +: 9,11 \quad -: \end{cases}$$

$$Ent(Usunny) = -\left(\frac{3}{5}log_2\frac{3}{5} + \frac{2}{5}log_2\frac{2}{5}\right) = 0.745$$

$$Ent(humidity) = -\frac{3}{5}\left(log_2\frac{0}{3} + \frac{3}{3}log_2\frac{3}{3}\right) = 0$$

Or, Ent(U)-Ent(humidity)

Gain(humidity)=0.745 **the highest**

$$\boldsymbol{Wind} = \begin{cases} Weak & +: 9 \quad -: 1,8 \\ Strong & +: 11 \quad -: 2 \end{cases}$$

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

Information Gain（信息增益）

Case 2: Outlook = **Cloudy.** $+: 3, 7, 12, 13$ $-: 0$

Already pure

Case 3: Outlook = **Rain.** $+: 4, 5, 10$ $-: 6, 14$

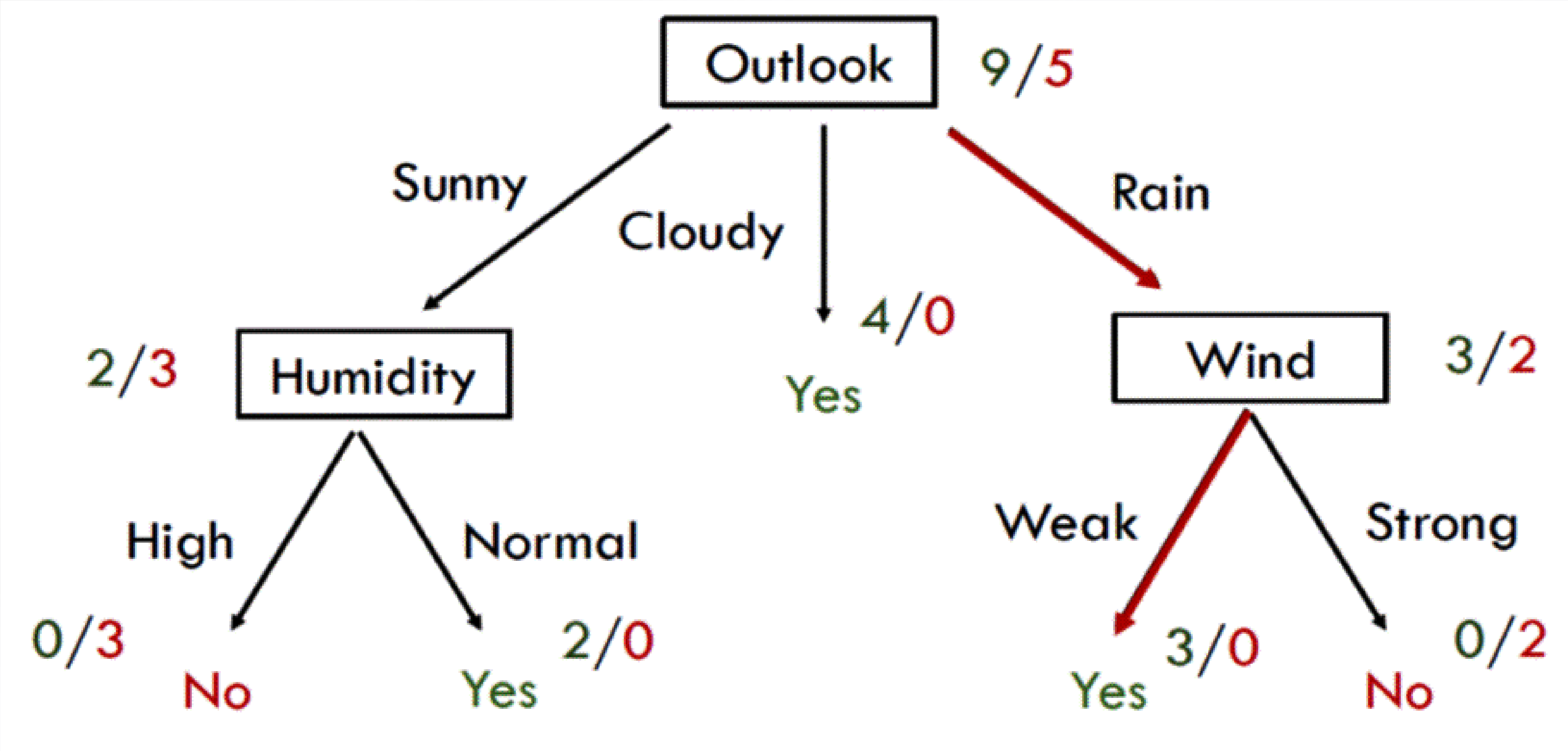
$$Humidity = \begin{Bmatrix} High & +:? & -:? \\ Normal & +:? & -:? \end{Bmatrix}$$

$$Wind = \begin{Bmatrix} Weak & +:? & -:? \\ Strong & +:? & -:? \end{Bmatrix}$$

Gain=? repeat the operation with the other nodes

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

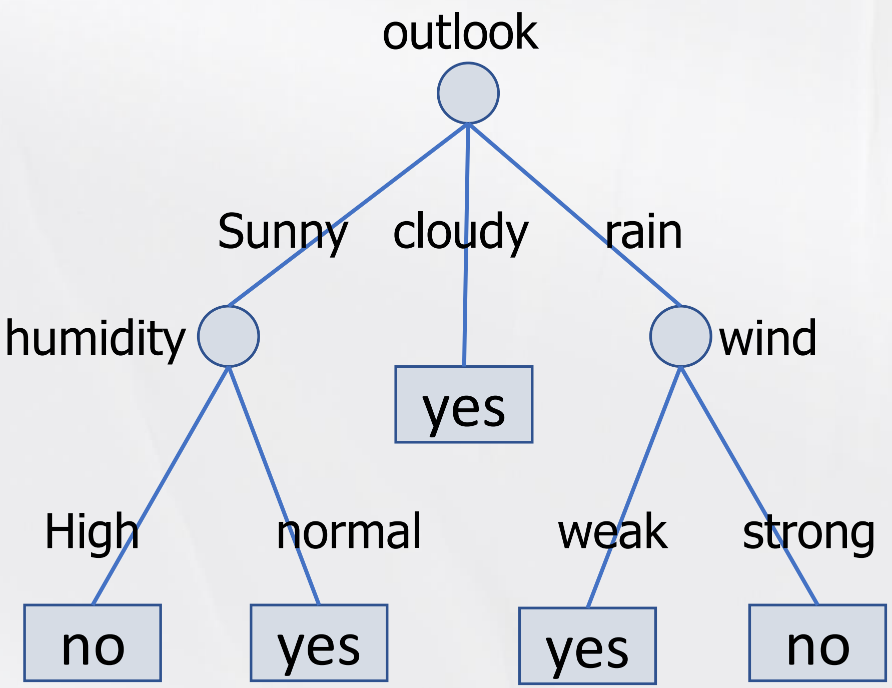
Information Gain（信息增益）



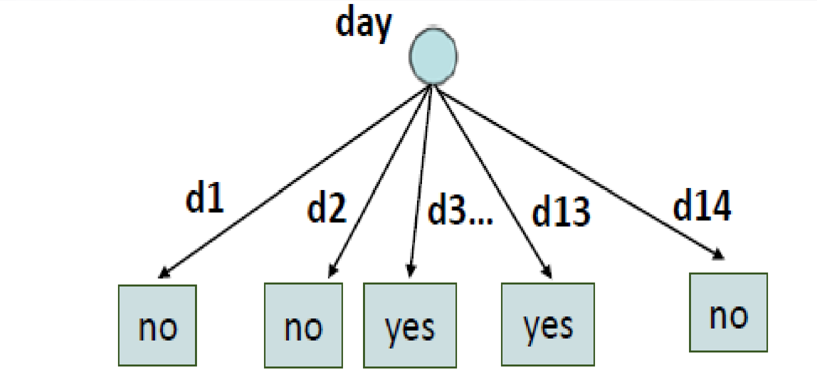
Day	Outlook	Humidity	Wind	play
15	rain	high	weak	?

Information Gain（信息增益）

✓ After using a feature to partition the data set, the purity of each data subset is higher than the purity of the data set D before partitioning (the uncertainty is lower than that of the data set before partitioning.)



Information Gain（信息增益）



Limitation of Information Gain

Consider a feature that uniquely identifies each training instance: ID number

– splitting on this feature would result in many branches, each of which is "pure" (has instances of only one class)

– maximal information gain based on ID number

Entropy of split = 0 (since each leaf node is "pure", having only one case). Therefore, the information gain is maximal.

Features with many unique values may appear to provide high Information Gain simply due to their granularity(粒度), potentially leading to **overfitting.**

03

Information Gain（信息增益）

Limitation of Information Gain

✓ **Limitation：** information gain is biased towards attributes with many distinct values

Age(T1)	B	A1	A2	C	B	B	C	C	C	A1	B	A1	A2	C
Sex(T2)	1	1	0	1	0	0	0	0	1	0	1	1	0	0
Buy(Y)	yes	yes	yes	no	yes	yes	yes	yes	no	no	yes	no	no	yes

$$Ent(U|T1)=\frac{3}{14}\left(-\frac{1}{3}log_2\frac{1}{3}-\frac{2}{3}log_2\frac{2}{3}\right)+\frac{2}{14}\left(-\frac{1}{2}log_2\frac{1}{2}-\frac{1}{2}log_2\frac{1}{2}\right)$$
$$+\frac{4}{14}\left(-\frac{4}{4}log_2\frac{4}{4}-\frac{0}{4}log_2\frac{0}{4}\right)+\frac{5}{14}\left(-\frac{3}{5}log_2\frac{3}{5}-\frac{2}{5}log_2\frac{2}{5}\right)=0.686867$$

$$Gain(U,T1)=\mathrm{Ent(U)}-\mathrm{Ent(U|T1)}$$
$$=0.940-0.686867=0.253133$$

Age(T1)	B	A	A	C	B	B	C	C	C	A	B	A	A	C
Sex(T2)	1	1	0	1	0	0	0	0	1	0	1	1	0	0
Buy(Y)	yes	yes	yes	no	yes	yes	yes	yes	no	no	yes	no	no	yes

0.253>0.246

03

Information Gain（信息增益）

- ✓ The smaller the entropy, the purer the distribution of the sample to the target attribute.

  A. TRUE　　　　B. FALSE

- ✓ A low Entropy indicates that the data labels are quite uniform. E.g. suppose a dataset has 100 samples. Among those, there are 1 Positive and 99 Negative labeled data points. In this case, the Entropy is very ________.

  A. low　　　　B. high

Information Gain（信息增益）

Disadvantages of ID3

- ✓ ID3 cannot handle continuous values
- ✓ ID3 cannot deal with missing values
- ✓ Produce a deep tree, easy to overfit
- ✓ Biased Towards Features with Many Categories: Features with a large number of categories may have higher IG simply due to their granularity, leading to such features have more chance to be selected than the input variables with less distinct values.

Gain Ratio（信息增益率）

- ✓ Gain Ratio attempts to lessen the bias of Information Gain on highly branched predictors.
- ✓ Gain ratio adjusts information gain by taking into account the number of branches (intrinsic information 内在信息)in the split, helping to mitigate bias towards features with many distinct values.
- ✓ It is calculated by dividing information gain by the intrinsic information of the feature, which reflects its potential for creating splits.
- ✓ The solution to this problem is to somehow penalize the attributes that lead to a very high number of branches.
- ✓ C4.5 uses a splitting criterion called gain ratio

Gain Ratio（信息增益率）

Gain Ratio is used to normalize the information gain of an attribute against how much entropy that attribute has. Formula :

$$\text{SplitInfo}(D,S) = -\sum_{k \in \text{outcomes}(S)} \frac{|D_k|}{|D|} \log_2\left(\frac{D_k}{D}\right) \longrightarrow \text{Entropy}$$

S: attribute， and k is one of the value of S.

use this to adjust information gain

$$\text{GainRatio}(D,S) = \frac{\text{InfoGain}(D,S)}{\text{SplitInfo}(D,S)}$$

Gain Ratio（信息增益率）

04

Outlook		Temperature		Humidity		Windy	
Info	0.693	Info	0.911	Info	0.788	Info	0.892
Gain	0.247	Gain	0.029	Gain	0.152	Gain	0.048
Split info	1.577	Split info	1.362	Split info	1.000	Split info	0.985
Gain ratio	0.157	Gain ratio	0.019	Gain ratio	0.152	Gain ratio	0.049

$$Splitinfo_{outlook} = -\frac{5}{14}log_2\frac{5}{14} - \frac{4}{14}log_2\frac{4}{14} - \frac{5}{14}log_2\frac{5}{14} = 1.577$$

How to select split attribute by Gain ratio?

① Find the attributes with higher information gain than the average level from the candidate partition attributes

② Select the attribute with the highest gain rate

04

Gain Ratio（信息增益率）

About missing values

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜	权重
1	---	蜷缩	浊响	清晰	凹陷	硬滑	是	1
2	乌黑	蜷缩	沉闷	清晰	凹陷	--	是	1
3	乌黑	蜷缩	---	清晰	凹陷	硬滑	是	1
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是	1
5	---	蜷缩	浊响	清晰	凹陷	硬滑	是	1
6	青绿	稍蜷	浊响	清晰	--	软黏	是	1
7	乌黑	稍蜷	浊响	稍糊	稍凹	软黏	是	1
8	乌黑	稍蜷	浊响	--	稍凹	硬滑	是	1
9	乌黑	---	沉闷	稍糊	稍凹	硬滑	否	1
10	青绿	硬挺	清脆	--	平坦	软粘	否	1
11	浅白	硬挺	清脆	模糊	平坦	---	否	1
12	浅白	蜷缩	---	模糊	平坦	软粘	否	1
13	---	稍蜷	浊响	稍糊	凹陷	硬滑	否	1
14	浅白	稍缩	沉闷	稍糊	凹陷	硬滑	否	1
15	乌黑	稍蜷	浊响	清晰	--	软粘	否	1
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否	1
17	青绿	---	沉闷	稍糊	稍凹	硬滑	否	1

Gain Ratio（信息增益率）

About missing values

① Select optimal attribute to be head node

Regarding “色泽”, ignore samples for which no values were obtained. So we have 14 observations.

$$Ent(\tilde{D}, \text{色泽}) = -\left(\frac{6}{14}log_2\frac{6}{14} + \frac{8}{14}log_2\frac{8}{14}\right) = 0.985$$

$$Ent(\tilde{D}|\text{青绿}) = -\frac{4}{14}\left(\frac{2}{4}log_2\frac{2}{4} + \frac{2}{4}log_2\frac{2}{4}\right) = 0.286$$

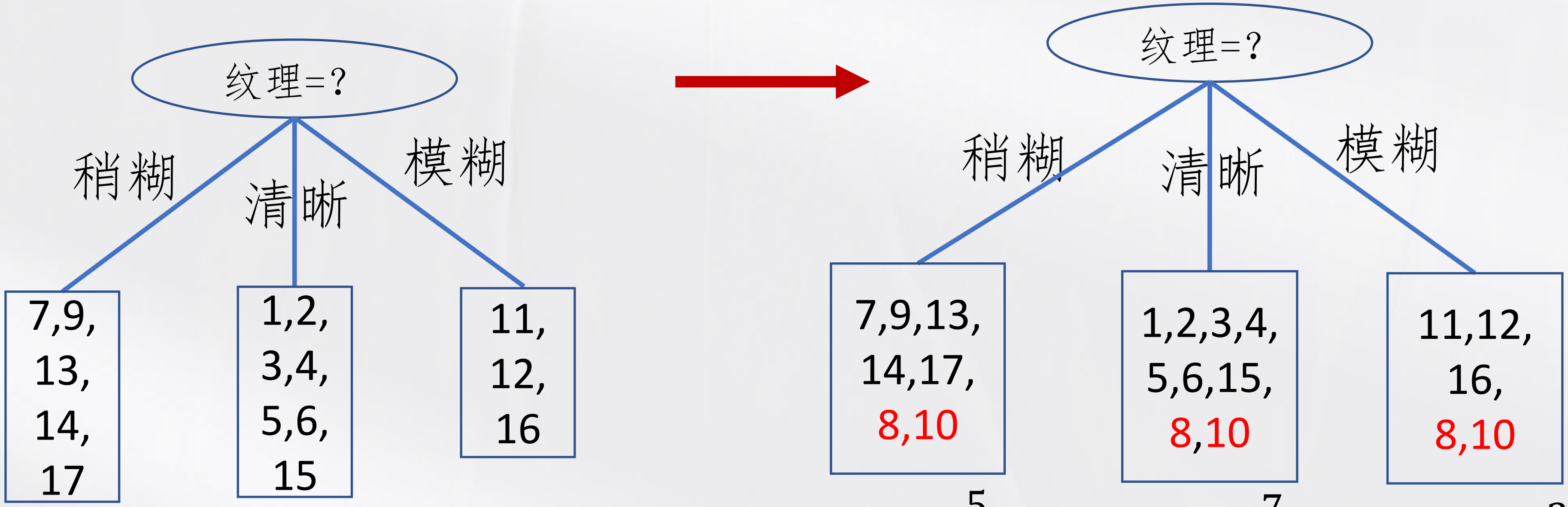
$$Ent(\tilde{D}|\text{乌黑}) = -\frac{6}{14}\left(\frac{4}{6}log_2\frac{4}{6} + \frac{2}{6}log_2\frac{2}{6}\right) = 0.394$$

$$Ent(\tilde{D}|\text{浅白}) = -\frac{4}{14}\left(\frac{0}{4}log_2\frac{0}{4} + \frac{4}{4}log_2\frac{4}{4}\right) = 0$$

$$Gain(D, \text{色泽}) = -\underbrace{\frac{14}{17}}_{\text{weight}}(0.985 - 0.286 - 0.394) = 0.252$$

属性	色泽	根蒂	敲声	纹理	脐部	触感
Gain	0.252	0.171	0.145	0.424	0.289	0.006

Gain Ratio（信息增益率）



$w_8, w_{10} = \frac{5}{15}$ $w_8, w_{10} = \frac{7}{15}$ $w_8, w_{10} = \frac{3}{15}$

For No.8 and No.10:

Go to 3 branches with weights: 7/15、5/15、3/15.

The weight is equal to the proportion of samples in each branch.

② Select optimal attribute to be internal nodes

$$\rho = \frac{4+\frac{2}{3}}{5+\frac{2}{3}} = \frac{14}{17}$$

Proportion of samples without missing values
色泽上没有缺失值的比例(权值)

$$\tilde{p}_1 = \frac{1+\frac{1}{3}}{4+2\times\frac{1}{3}} = \frac{4}{14}$$ 正样本比例

$$\tilde{p}_2 = \frac{3+\frac{1}{3}}{4+\frac{2}{3}} = \frac{10}{14}$$ 负样本比例

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜	权重
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是	1
8	乌黑	稍蜷	浊响	--	稍凹	硬滑	是	5/15
9	乌黑	---	沉闷	稍糊	稍凹	硬滑	否	1
10	青绿	硬挺	清脆	--	平坦	软粘	否	5/15
13	---	稍蜷	浊响	稍糊	凹陷	硬滑	否	1
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否	1
17	青绿	---	沉闷	稍糊	稍凹	硬滑	否	1

$$Ent(\tilde{D}) = -\sum_{k=1}^{2} \tilde{p}_k \, log_2 \tilde{p}_k = -\left(\frac{4}{14} log_2 \frac{4}{14} + \frac{10}{14} log_2 \frac{10}{14}\right) = 0.863$$

$\tilde{D}^1$ {乌黑，7，8，9}

$$\tilde{r}_1 = \frac{2+\frac{1}{3}}{4+\frac{2}{3}} = \frac{7}{14}$$

$\tilde{D}^2$ {青绿，10，17}

$$\tilde{r}_2 = \frac{1+\frac{1}{3}}{4+\frac{2}{3}} = \frac{4}{14}$$

$\tilde{D}^3$ {浅白，14}

$$\tilde{r}_3 = \frac{1}{4+\frac{2}{3}} = \frac{3}{14}$$

weight

$$Ent(\tilde{D}^1) = -\left(\frac{1+\frac{1}{3}}{2+\frac{1}{3}} log_2 \frac{1+\frac{1}{3}}{2+\frac{1}{3}} + \frac{1}{2+\frac{1}{3}} log_2 \frac{1}{2+\frac{1}{3}}\right) = 0.985$$

$$Ent(\tilde{D}^3) = -\left(\frac{0}{1} log_2 \frac{0}{1} + \frac{1}{1} log_2 \frac{1}{1}\right) = 0$$

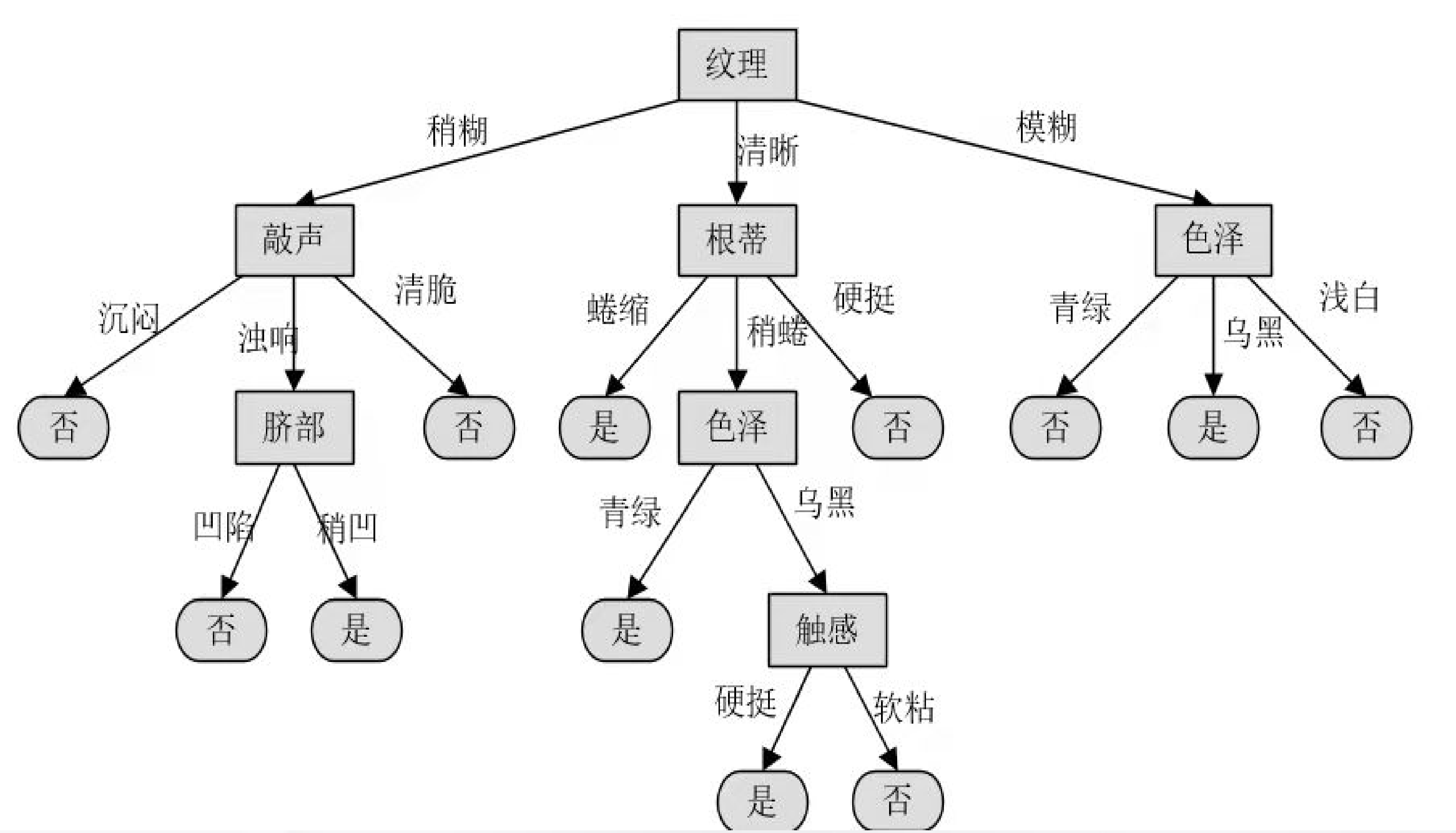
$$Ent(\tilde{D}^2) = -\left(\frac{0}{1+\frac{1}{3}} log_2 \frac{0}{1+\frac{1}{3}} + \frac{1+\frac{1}{3}}{1+\frac{1}{3}} log_2 \frac{1+\frac{1}{3}}{1+\frac{1}{3}}\right) = 0$$

$$Gain(\tilde{D}, 色泽) = Ent(\tilde{D}) - \sum_{v=1}^{3} \tilde{r}_v \, Ent(\tilde{D}^v) = 0.863 - \left(\frac{7}{14} * 0.985 + 0 + 0\right) = 0.371$$

$$Gain(D, 色泽) = \rho \times Gain(\tilde{D}, 色泽) = \frac{14}{17} * 0.371 = 0.305$$

Gain Ratio（信息增益率）

About missing values



Gain Ratio（信息增益率）

About continuous values

✓ Discretization（离散化）：

- The discretized features are very robust to abnormal data（异常数据）.

e.g. Age=300, if discretization, age>50, be good to modeling. Otherwise disturb modeling

- After discretization, there will be information loss. On the other hand it simplifies models and reduces the risk of model overfitting.
- After the feature is discretized, the model will be more stable.

Gain Ratio（信息增益率）

About continuous values

✓ Information Gain

- Sort according to continuous attribute values
- Dynamically divide the data set into two parts, one part is more than a certain value, another part is less than the certain value.
- Calculate the information gain according to the division, and the largest value is used as the final division.

23	25	27	30	39	41	43	45	46	47	62	63	66	66	68

24, 26, 28.5, 34.5, 40, 42, 44, 45.5, 46.5, 54.5, 62.5, 64.5, 66, 67

04

Gain Ratio（信息增益率）

ID	Age	Career	Gender	Income	Churn
1	23	Teacher	F	lower	N
2	25	Teacher	F	general	N
3	27	Engineer	F	general	N
4	30	Engineer	M	low	Y
5	39	Teacher	F	lower	N
6	41	Teacher	F	low	N
7	43	Teacher	F	High	N
8	45	Doctor	M	High	Y
9	46	Teacher	M	High	Y
10	47	Teacher	M	High	Y
11	62	Teacher	M	Higher	Y
12	63	Teacher	M	High	Y
13	66	Doctor	F	High	Y
14	66	Doctor	F	Higher	Y
15	68	Engineer	F	general	N

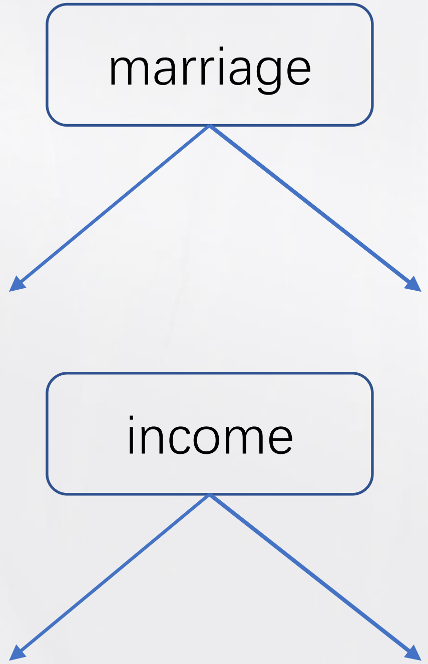
Split	Gain	split	Gain
{A|0<A<=24} {A|A>24	0.077	{A|0<A<=45.5} {A|A>45.5}	0.288
{A|0<A<=26} {A|A>26}	0.164	{A|0<A<=46.5} {A|A>46.5}	0.168
{A|0<A<=28.5} {A|A>28.5}	0.262	{A|0<A<=54.5} {A|A>54.5}	0.109
{A|0<A<=34.5} {A|A>34.5}	0.088	{A|0<A<=62.5 {A|A>62.4}	0.052
{A|0<A<=40} {A|A>40}	0.169	{A|0<A<=64.5} {A|A>64.5}	0.013
{A|0<A<=42} {A|A>42}	0.278	{A|0<A<=67} {A|A>67}	0.077
{A|0<A<=44} {A|A>44}	0.431		

ID	Age	Career	Gender	Income	Churn
1	<24	Teacher	F	lower	N
2	>24	Teacher	F	general	N
3	>24	Engineer	F	general	N
4	>24	Engineer	M	low	Y
5	>24	Teacher	F	lower	N
6	>24	Teacher	F	low	N
7	>24	Teacher	F	High	N
8	>24	Doctor	M	High	Y
9	>24	Teacher	M	High	Y
10	>24	Teacher	M	High	Y
11	>24	Teacher	M	Higher	Y
12	>24	Teacher	M	High	Y
13	>24	Doctor	F	High	Y
14	>24	Doctor	F	Higher	Y
15	>24	Engineer	F	general	N

$$SplitInfor_A(Z) = -\sum_{j=1}^{n} \frac{|Z_j|}{|Z|} \times \log_2\left(\frac{|Z_j|}{|Z|}\right) = -\frac{7}{15} \times \log_2 \frac{7}{15} - \frac{8}{15} \times \log_2 \frac{8}{15} = 0.997$$

Gini Index

No.	house	marriage	Income(k)	default
1	Yes	single	125	No
2	No	married	100	No
3	No	single	70	No
4	Yes	married	120	No
5	No	divorced	95	Yes
6	No	married	60	No
7	Yes	divorced	220	No
8	No	single	85	Yes
9	No	married	75	No
10	No	single	90	Yes



how can we quantify that?

Binary Tree 二叉树

Gini Index

- ✓ Gini Index or Gini impurity measures the degree or probability of a particular variable being wrongly classified when it is randomly chosen.

- ✓ Gini index (Gini Impurity) assists the CART algorithm in identifying the most suitable feature for node splitting. It derives its name from the Italian mathematician Corrado Gini.

Gini Index

How to calculate Gini

✓ Consider a dataset $\boldsymbol{D}$ that contains samples from $\boldsymbol{k}$ **classes**. The probability of samples belonging to class $\boldsymbol{i}$ at a given node can be denoted as $\boldsymbol{p_i}$. Then the Gini Impurity of D is defined as:

$$Gini == 1 - \sum_{i=1}^{k} p_i^2 = 1 - \sum_{i=1}^{k} \left(\frac{|C_i|}{|D|}\right)^2$$

Where, k is the total number of classes and p_i is the probability of picking the data point with the class i. (The proportion of the class "i" is in the dataset.)

The more impure the dataset, the higher is the Gini index.

the lower the value of the Gini Index, the more pure is the dataset, and the lower is the entropy.

05

Gini Index

$$Gini == 1 - \sum_{i=1}^{k} p_i^2 = 1 - \sum_{i=1}^{k} \left(\frac{|C_i|}{|D|} \right)^2$$

The degree of Gini Index varies between 0 and 1

‘0’: denotes that all elements belong to a certain class or there exists only one class (pure)

‘1’: denotes that the elements are randomly distributed across various classes (impure).

‘0.5’ : denotes equally distributed elements into some classes.

Gini Index

$$Gini(Case1) = 1 - (\frac{4}{10})^2 - (\frac{6}{10})^2$$
$$= 0.48$$

$$Gini(Case2) = 1 - (\frac{3}{10})^2 - (\frac{2}{10})^2 - (\frac{1}{10})^2 - (\frac{4}{10})^2$$
$$= 0.7$$

Gini randomness $\uparrow$, Uncertainty $\uparrow$

05

Gini Index

To calculate the Gini index in a decision tree, follow these steps:

①**Calculate Gini Impurity for Each Node:**

$$Gini(D) = 1 - \sum_{i=1}^{k} p_i^2$$

②**Calculate Weighted Gini Impurity for Each Split:**

If a data set D is split on an attribute A into two subsets $D1$ and $D2$ with sizes $n1$ and $n2$, respectively, the Gini Impurity can be defined as:

$$Gini_A(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

Gini Index

③ Select the Split with the Lowest Gini Index:

In order to obtain Gini gain for an attribute, the weighted impurities of the branches is subtracted from the original impurity. The best split can also be chosen by maximizing the Gini gain. Gini gain is calculated as follows:

$$Gini_A(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$\triangle Gini(A) = Gini(D) - Gini_A(D)$$

05

Gini Index

✓ If a data set is split on A into two subsets, the standard Gini index /△G(t) of dataset D is defined as:

smallest ← $$Gini(D, A) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$

biggest ← $$\Delta G(A) = Gini(D) - Gini(D, A)$$

✓ The attribute providing smallest Gini index (greatest Gini gain) is chosen to split the node. (need to enumerate(枚举) all the possible splitting points for each attribute).

✓ Gini Impurity can be understood as a criterion to minimize the probability of **misclassification**

Gini Index

No.	house	marriage	Income(k)	default
1	Yes	single	125	No
2	No	married	100	No
3	No	single	70	No
4	Yes	married	120	No
5	No	divorced	95	Yes
6	No	married	60	No
7	Yes	divorced	220	No
8	No	single	85	Yes
9	No	married	75	No
10	No	single	90	Yes

$$Gini(D) = 1 - \left(\frac{3}{10}\right)^2 - \left(\frac{7}{10}\right)^2 = 0.42$$

$$Gini(D, house = Yes) = 1 - \left(\frac{0}{3}\right)^2 - \left(\frac{3}{3}\right)^2 = 0$$

$$Gini(D, house = No) = 1 - \left(\frac{3}{7}\right)^2 - \left(\frac{4}{7}\right)^2 = 0.4898$$

$$Gini(D, house) = \frac{7}{10} \times 0.4898 + \frac{3}{10} \times 0 = \boxed{0.343}$$

$$Gini(D, \{single, \{m, d\}) = \frac{4}{10} \times 0.5 + \frac{6}{10} \times$$

$$[1 - \left(\frac{1}{6}\right)^2 - \left(\frac{5}{6}\right)^2]=0.367$$

$$Gini(D, \{married\}, \{s, d\}) = \frac{4}{10} \times 0 + \frac{6}{10} \times$$

$$[1 - \left(\frac{3}{6}\right)^2 - \left(\frac{3}{6}\right)^2]\boxed{=0.3}$$

$$Gini(D, \{divoiced, \{m, s\}) = \frac{2}{10} \times 0.5 + \frac{8}{10} \times$$

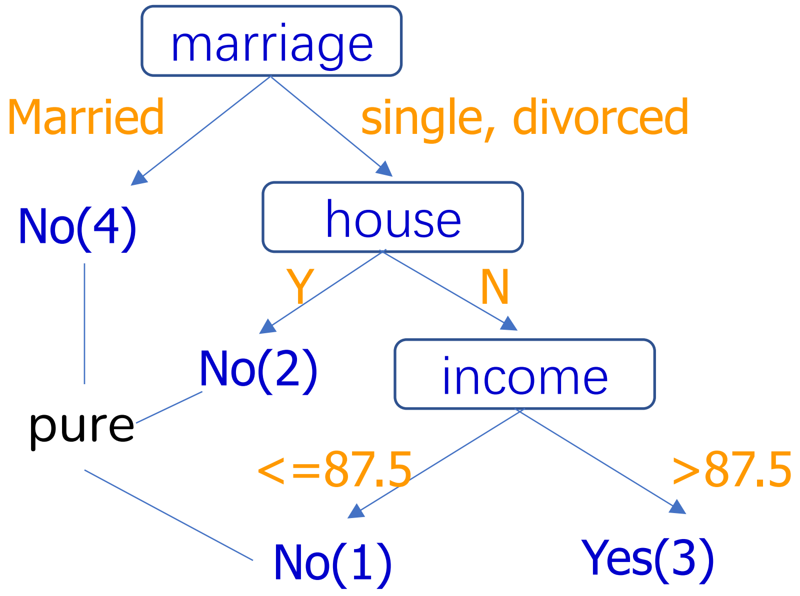
$$[1 - \left(\frac{2}{8}\right)^2 - \left(\frac{6}{8}\right)^2]=0.4$$

default	no	no	no	yes	yes	yes	no	no	no	no
income	60	70	75	85	90	95	100	120	125	220
average	65	72.5	80	87.7	92.5	97.5	110	122.5	172.5	
Gini	0.4	0.375	0.343	0.417	0.4	0.3	0.343	0.375	0.4	

$$Gini(D, income < 97.5) = 1 - \left(\frac{3}{6}\right)^2 - \left(\frac{3}{6}\right)^2 = \frac{1}{2}$$

$$Gini(D, income > 97.5) = 1 - \left(\frac{4}{4}\right)^2 - \left(\frac{0}{4}\right)^2 = 0$$

$$\boxed{Gini(D, income) = \frac{6}{10} \times \frac{1}{2} + \frac{4}{10} \times 0 = 0.3}$$



default	no	yes	yes	yes	no	no
income	70	85	90	95	125	220
average	77.5	87.5	92.5	110	172.5	
Gini	0.4	0.375	0.343	0.417	0.4	

$$Gini(default\ or\ not) = 1 - \left(\frac{3}{6}\right)^2 - \left(\frac{3}{6}\right)^2 = 0.5$$

$$Gini(house\ or\ not) = \frac{4}{6} \times \left[1 - \left(\frac{3}{4}\right)^2 - \left(\frac{1}{4}\right)^2\right] - \frac{2}{6} \times 0 = 0.25$$

Gini Index

Difference between Gini Index and Entropy

Gini Index	Entropy
It is the probability of **misclassifying** a randomly chosen element in a set.	While entropy measures the amount of **uncertainty or randomness** in a set.
Gini index is a **linear** measure.	Entropy is a **logarithmic** measure.
It can be interpreted as the expected **error rate** in a classifier.	It can be interpreted as the **average amount** of information needed to specify the class of an instance.
The range of the Gini index is **[0, 1],** where 0 indicates perfect purity and 1 indicates maximum impurity.	The range of entropy is **[0, log(c)],** where c is the number of classes.
Gini index is typically used in **CART** (Classification and Regression Trees) algorithms	Entropy is typically used in **ID3 and C4.5** .

Gini Index

最大熵：当所有类别出现的概率相等时，此时熵达到最大值。对于 个类别，每个类别发生的概率为 1/c。根据熵的定义公式：

$$Ent = -\sum_{i=1}^{c} p_i \log(p_i)$$

当 $p_i = \frac{1}{c}$

$$Ent = -c\left(\frac{1}{c}\right)\log\left(\frac{1}{c}\right) = -1 \times (-\log(c)) = \log(c)$$

最小熵：当只有一个类别发生而其他类别都不发生时，熵达到最小值。

?

05

Gini Index

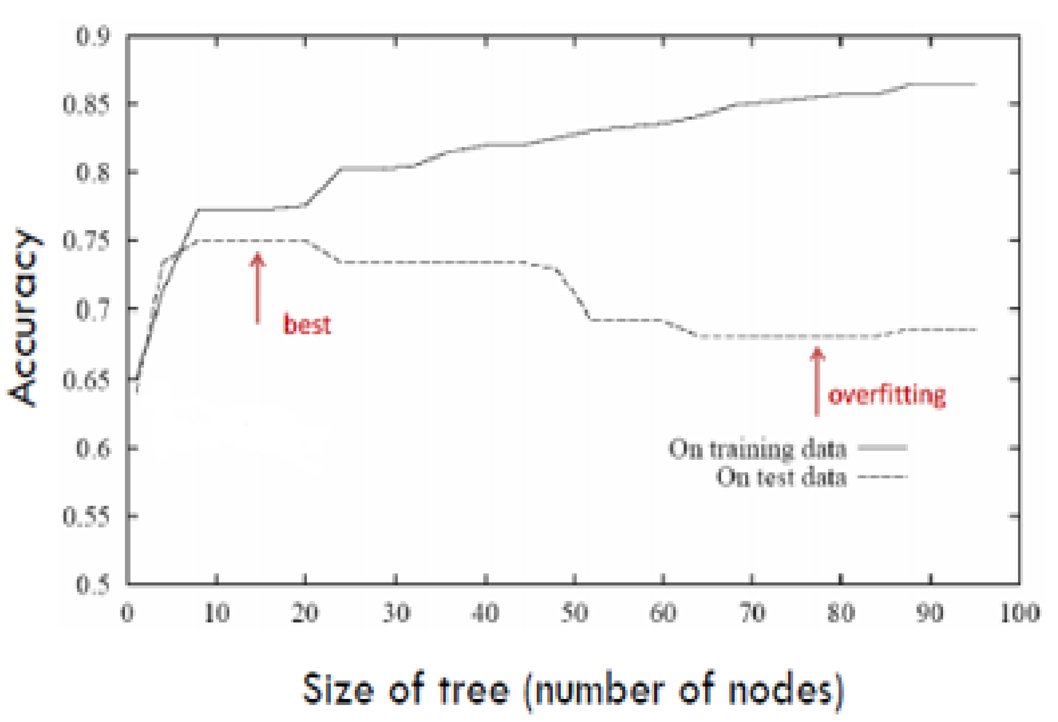
$$-\left(\sum p_i \log_2 p_i\right)$$

Above is the formula of (   ).

A. Information Gain of a split.
B. Intrinsic Information of a split.
C. Gini-index of a dataset.
D. Information Entropy of a dataset.

Overfitting(过度拟合)

- ✓ Can classify training examples perfectly?
  - Yes, keep splitting until each node contains 1 sample
  - Singleton=pure
- ✓ Does not work well on new/testing data
  -- Overfitting



With the increasing complexity of trees, overfitting emerges

- ✓ Overfitting results in decision trees that are **more complex** than necessary
- ✓ **Less complex** trees can yield more **stable** models
- ✓ Avoid overfitting: **Pruning**

Overfitting(过度拟合)

A ML/DM model is said to be overfitted when the model does not make accurate predictions on testing data.

Why?

- ✓ **Complexity:** Decision trees become overly complex, fitting training data perfectly but struggling to generalize(泛化) to new data.
- ✓ **Memorizing Noise:** It can focus too much on specific data points or noise in the training data, hindering generalization.
- ✓ **Overly Specific Rules:** Might create rules that are too specific to the training data, leading to poor performance on new data.

In a nutshell(简而言之), Overfitting is a problem where the evaluation of classification algorithms on training data is different from testing /unseen data.

06

Overfitting(过度拟合)

Clues to prevent overfitting

- ✓ Involves **removing** parts of the decision tree that **do not contribute** significantly to its predictive power.
- ✓ Helps **simplify** the model and prevent it from memorizing noise in the training data.
- ✓ **Pruning** can be achieved through **techniques** such as cost-complexity pruning, which iteratively removes nodes with the least impact on performance.

06

Overfitting(过度拟合)

Common ways to prevent overfitting

Tolerate errors to a certain extent

Limiting Tree Depth

Setting a maximum depth for the decision tree restricts the number of levels or branches it can have.

Minimum Samples per Leaf Node

Specifying a minimum number of samples required to create a leaf node ensures that each leaf contains a sufficient amount of data to make meaningful predictions. This helps prevent the model from creating overly specific rules that only apply to a few instances in the training data, reducing overfitting.

Overfitting(过度拟合)

Feature Selection and Engineering

Feature selection involves choosing the most informative features that contribute to predictive power while discarding redundant or noisy ones. Feature engineering involves transforming or combining features to create new meaningful variables that improve model performance.

Ensemble Methods

Ensemble methods such as Random Forests combine multiple decision trees to reduce overfitting.

Pruning(剪枝)

Types of pruning

Pre-Pruning (Early Stopping)

тhe growth of the decision tree can be stopped before it gets too complex.
Prevent the overfitting of the training data, which results in a poor performance when exposed to new data.

Post-Pruning (Reducing Nodes)

After the tree is fully grown, post-pruning involves removing branches or nodes to improve the model’s ability to generalize

07

Pruning

✓ **Some common pre-pruning techniques**

- **Maximum Depth:** limits the maximum level of depth in a decision tree.
- **Minimum Samples per Leaf**: Set a minimum threshold for the number of samples in each leaf node.
- **Minimum Samples per Split:** Specify the minimal number of samples needed to break up a node.
- **Maximum Features:** Restrict the quantity of features considered for splitting.

By pruning early, we come to be with a simpler tree that is less likely to overfit the training facts.

Pruning

✓ Post-Pruning (Reducing Nodes)

•**Minimal Cost-Complexity Pruning (MCCP最小代价复杂度):** assigns a price to each subtree primarily based on its accuracy and complexity, then selects the subtree with the lowest fee.

•**Reduced Error Pruning:** Removes branches that do not significantly affect the overall accuracy.

•**Minimum Impurity Decrease:** Prunes nodes if the decrease in impurity (Gini impurity or entropy) is beneath a certain threshold.

•**Minimum Leaf Size:** Removes leaf nodes with fewer samples than a specified threshold.

Pruning

Minimal Cost-Complexity

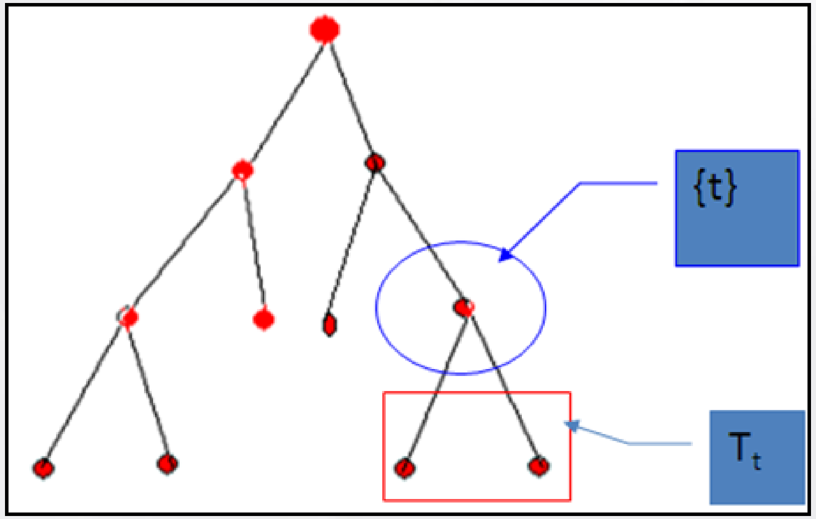
For any subtree $T<Tmax$, the cost-complexity measure $R\alpha(T)$ as:

CP(complexity parameter)

$$R_{\alpha}(\mathrm{T}) = R(T) + \alpha|\tilde{T}| \qquad \alpha \geqslant 0$$

error of DT

number of leaf nodes in T



$R_{\alpha}(\{t\})$ VS $R_{\alpha}(T_t)$

Which subtree is selected eventually depends on α .

If $\alpha=0$: the biggest tree will be chosen

As α approaches infinity: the tree has a single root node.

In general, given a pre-selected α , find the subtree $T(\alpha)$ that minimizes $R\alpha(T)$.

08

Cross Validation (交叉验证)

What is Cross-Validation?

Cross validation is a technique used in machine learning to evaluate the performance of a model on unseen data.

It involves dividing the available data into multiple folds or subsets, using one of these folds as a validation set, and training the model on the remaining folds.

Cross Validation (交叉验证)

What is cross-validation used for?

The main purpose of cross validation is to **prevent overfitting**, which occurs when a model is trained too well on the training data and performs poorly on testing, unseen data.

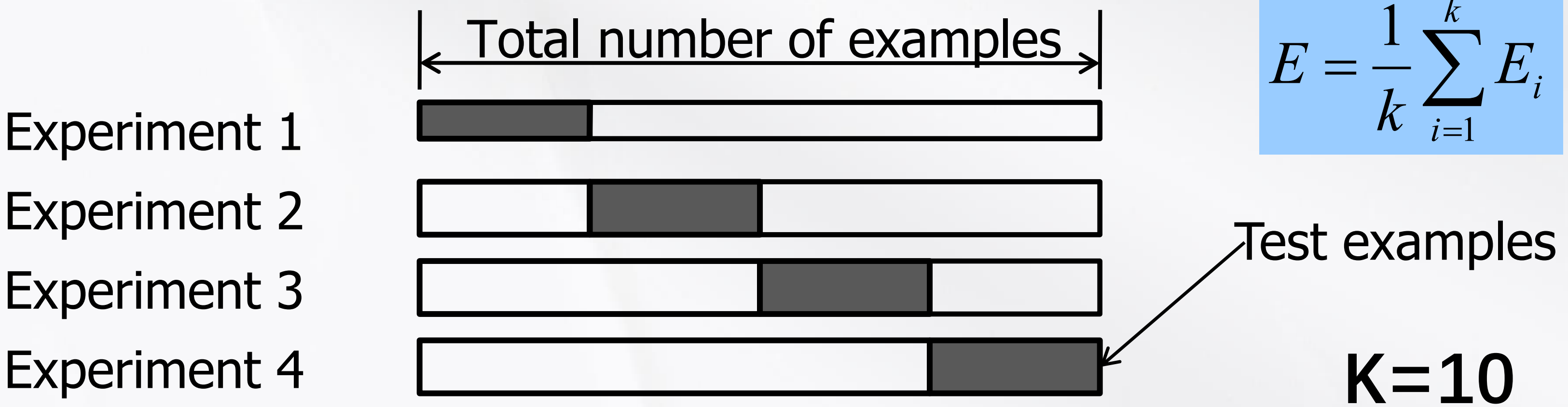
By evaluating the model on multiple validation sets, cross validation provides a more realistic estimate of the model’s generalization performance, i.e., its ability to perform well on testing, unseen data.

Cross Validation (交叉验证)

Types of Cross-Validation

k-fold cross validation

Split the dataset into k number of subsets (folds), perform training on the all the subsets but leave one(k-1) subset for the evaluation of the trained model. Iterate k times with a different subset reserved for testing purpose each time. All the samples are used for both training and testing.



$$E = \frac{1}{k}\sum_{i=1}^{k} E_i$$

K=10

- The true error is estimated as the average error rate

Cross Validation (交叉验证)

Types of Cross-Validation

LOOCV (Leave One Out)	train on the whole dataset but leaves only one data of the dataset,and then iterates for each data-point. the model is trained on samples and tested on the one omitted sample, repeating this process for each data point in the dataset.

Drawback:

- it leads to higher variation in the testing model as we are testing against one data point. If the data point is an outlier it can lead to higher variation.
- it takes a lot of execution time as it iterates over ‘the number of data points’ times.

Cross Validation (交叉验证)

Types of Cross-Validation

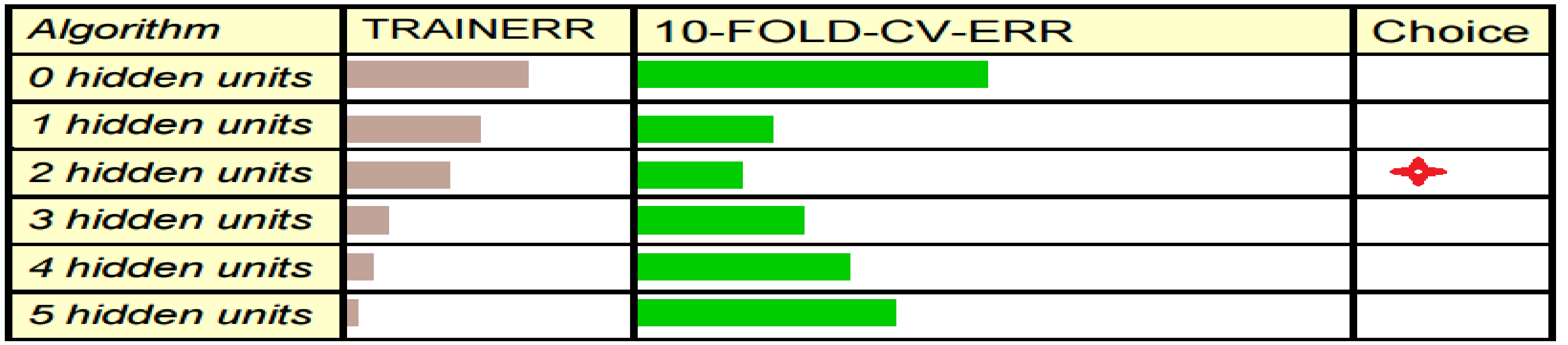
Hold Out	Perform training on the part of the given dataset and the rest is used for the testing purpose. It's a simple and quick way to evaluate a model.

The major drawback :

we perform training on the 50% of the dataset, it may possible that the remaining 50% of the data contains some important information which we are leaving while training our model i.e. higher bias(高偏差).

Cross Validation (交叉验证)

Cross-validation is often used for parameters tuning.



For example, choosing number of hidden units in a neural net can be done by cross validation.

Step1: For different values of the parameter compute 10-fold CV.

Step2: Pick whichever model's parameter that gave best CV score.

Notice: If you use CV for parameter tuning you need another independent sample to correctly measure the final model's performance.

09

Summary

Advantages

Easy to understand and interpret: can be easily understood and interpreted by humans, even those without a machine learning background.

No Need for Feature Scaling: do not require normalization or scaling of the data.

Handles Non-linear Relationships: Capable of capturing non-linear relationships between features and target variables.

Disadvantages

Overfitting: can easily overfit the training data, especially if they are deep with many nodes.

Bias towards Features with More Levels: towards features with more levels.

Instability: Small variations in the data can result in a completely different tree being generated.