Bayes & Decision Tree Classifiers

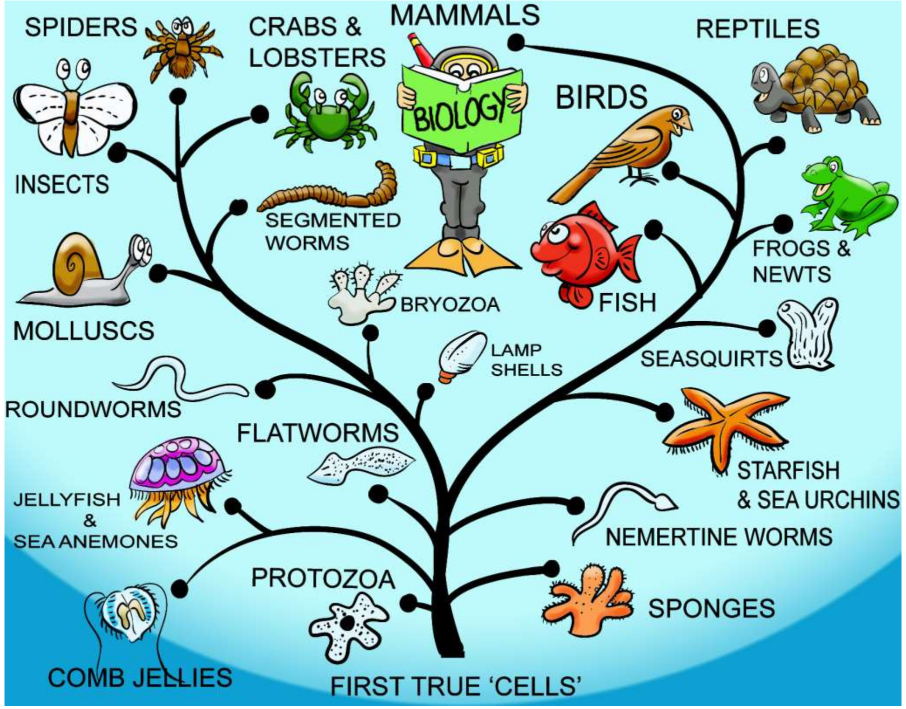
Lecturer: Dr. Bo Yuan

E-mail: yuanb@sz.tsinghua.edu.cn

Overview

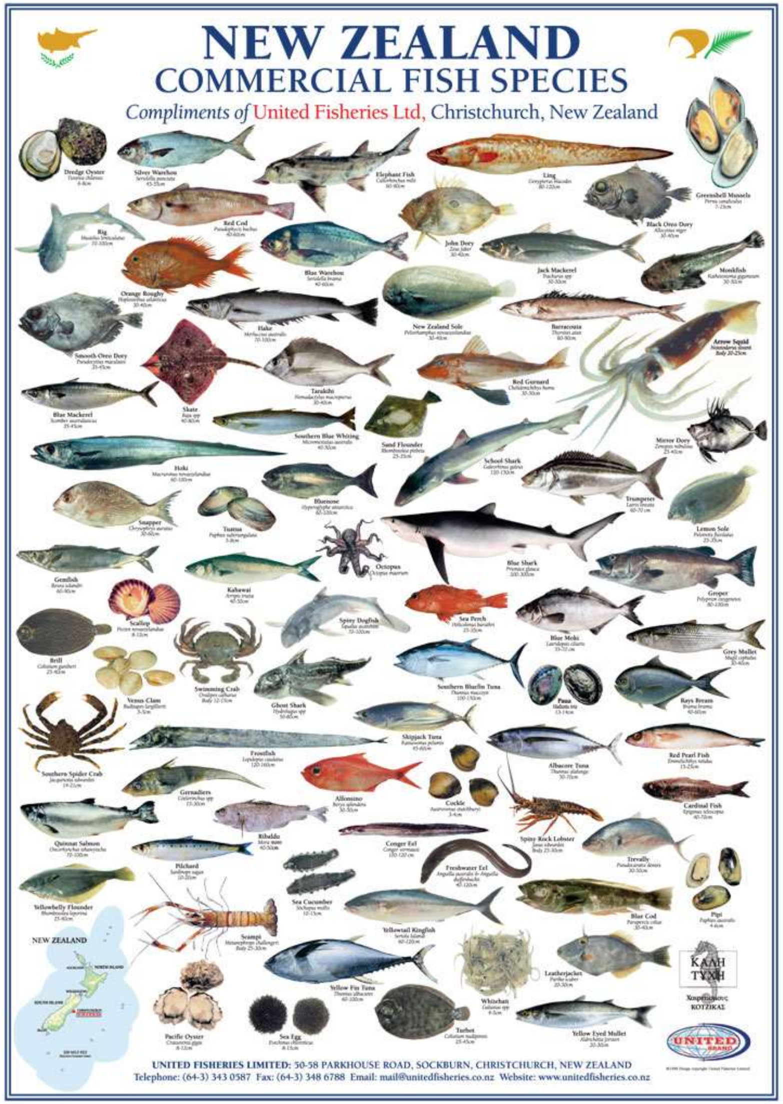
- Naïve Bayes Classifier
- Decision Tree Model

Thomas Bayes



Evolution Tree

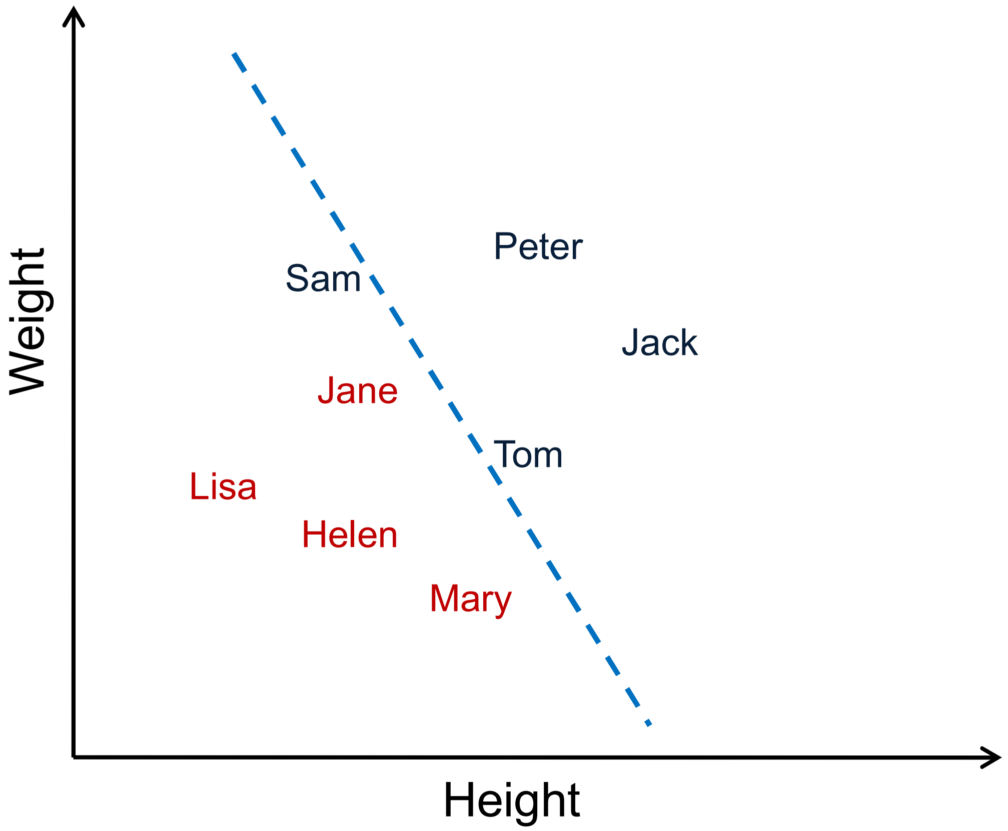
Classification

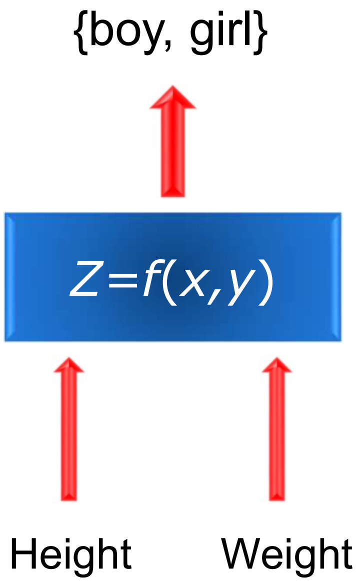


Definition

- Classification is one of the fundamental skills for survival.
  - Food vs. Predator
- A kind of **supervised** learning
  - Techniques for deducing a function from data
  - <Input, Output>
  - Input: a vector of features
  - Output: a Boolean value (binary classification) or integer (multiclass)
- “Supervised” means:
  - A teacher or oracle is needed to label each data sample.
- We will talk about **unsupervised** learning later.

Classifiers





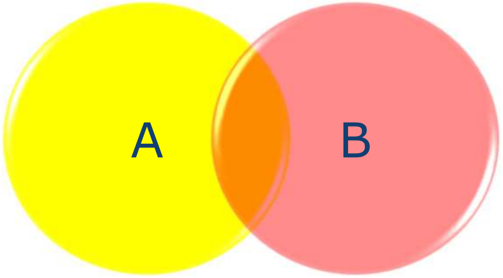
Training a Classifier

Learning

Bayes Theorem

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

$$P(A \cap B) = P(A|B)P(B) = P(B|A)P(A)$$



Likelihood of evidence B if A is true

Prior probability

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Posterior probability of A given the evidence B

Prior probability that evidence B is true

Fish Example

- Salmon vs. Tuna
- Grab a fish at random.
- $P(\omega_1)=P(\omega_2)$
- $P(\omega_1)>P(\omega_2)$
- Additional information

$$P(\omega_i \mid x)=\frac{P(x \mid \omega_i)P(\omega_i)}{P(x)}$$

Shooting Example

- Probability of Kill
  - P(A): 0.6
  - P(B): 0.5

- The target is killed with:
  - One shoot from A
  - One shoot from B

- What is the probability that it is shot down by A?
  - C: The target is killed.

$$P(A \mid C) = \frac{P(C \mid A)P(A)}{P(C)} = \frac{1 \times 0.6}{0.6 \times 0.5 + 0.4 \times 0.5 + 0.6 \times 0.5} = \frac{3}{4}$$

Cancel Example

- ω_1: Cancer; ω_2: Normal
- $P(\omega_1)=0.008$; $P(\omega_2)=0.992$
- Lab Test Outcomes: + vs. –

- $P(+|\omega_1)=0.98$; $P(-|\omega_1)=0.02$
- $P(+|\omega_2)=0.03$; $P(-|\omega_2)=0.97$
- Now someone has a positive test result…
- Is he/she doomed?

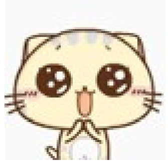
Cancel Example

$$P(\omega_1 \mid +) \propto P(+ \mid \omega_1)P(\omega_1) = 0.98 \times 0.008 = 0.0078$$

$$P(\omega_2 \mid +) \propto P(+ \mid \omega_2)P(\omega_2) = 0.03 \times 0.992 = 0.0298$$

$$P(\omega_1 \mid +) < P(\omega_2 \mid +)$$

$$P(\omega_1 \mid +) = \frac{0.0078}{0.0078 + 0.0298} = 0.21 >> P(\omega_1)$$



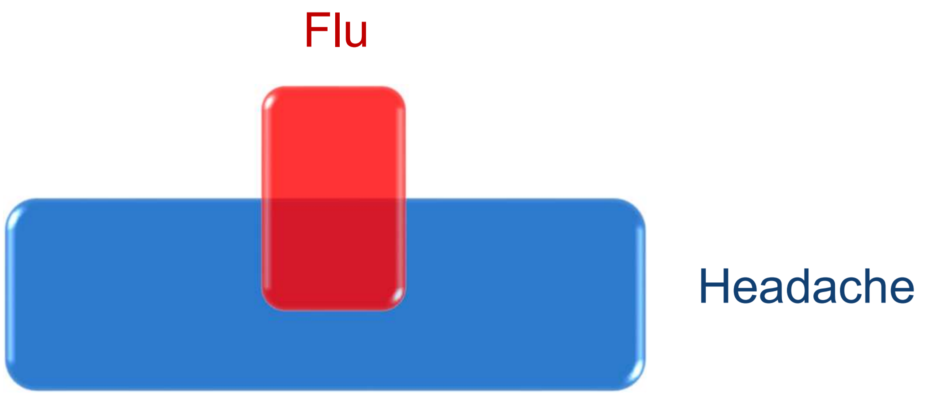
Headache & Flu Example

- H="Having a headache"
- F="Coming down with flu"
- P(H)=1/10; P(F)=1/40;   P(H|F)=1/2
- What does this mean?
- One day you wake up with a headache …
- Since 50% flu cases are associated with headaches …
- I must have a 50-50 chance of coming down with flu!

Headache & Flu Example

The truth is …

$$P(F \mid H) = \frac{P(H \mid F)P(F)}{P(H)} = \frac{1/2 \times 1/40}{1/10} = \frac{1}{8}$$



Naïve Bayes Classifier

$$\omega_{MAP} = \underset{\omega_i \in \omega}{\arg\max}\, P(\omega_i \mid a_1, a_2, \ldots, a_n)$$

$$\omega_{MAP} = \underset{\omega_i \in \omega}{\arg\max}\, \frac{P(a_1, a_2, \ldots, a_n \mid \omega_i)P(\omega_i)}{P(a_1, a_2, \ldots, a_n)}$$

$$\omega_{MAP} = \underset{\omega_i \in \omega}{\arg\max}\, \underline{P(a_1, a_2, \ldots, a_n \mid \omega_i)}P(\omega_i)$$

Conditionally Independent

$$\omega_{MAP} = \underset{\omega_i \in \omega}{\arg\max}\, P(\omega_i)\boxed{\prod_j P(a_j \mid \omega_i)}$$

MAP: Maximum A Posterior

Independence

$P(A \cap B) = P(A)P(B|A)$ $+$ $P(B|A) = P(B)$

$$\Downarrow$$

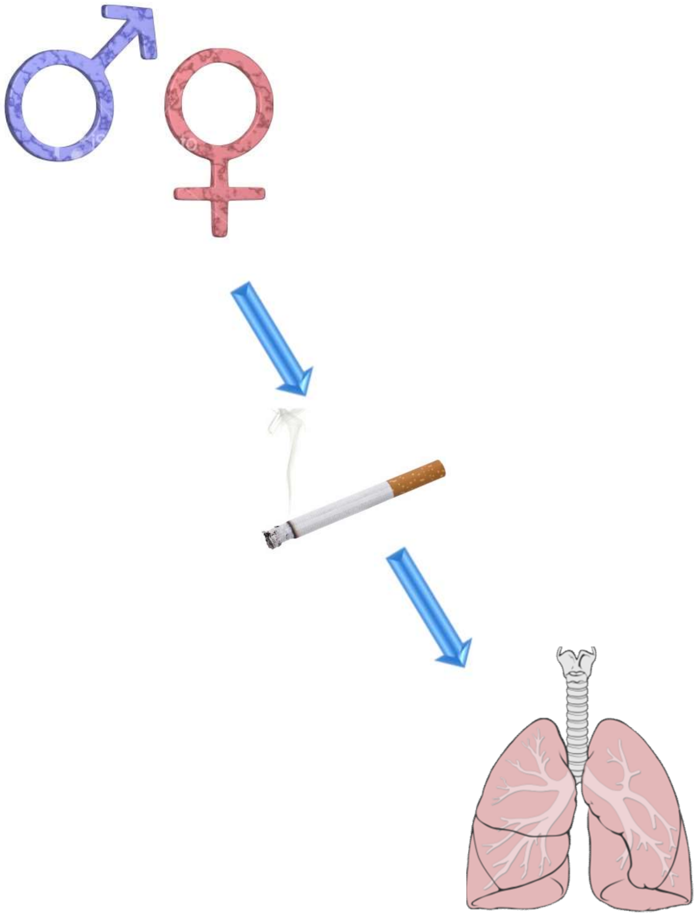
$$P(A \cap B) = P(A)P(B)$$

Conditionally Independent

$$P(A,B \mid G) = P(A \mid G)P(B \mid G) \Longleftrightarrow \underline{P(A \mid G,B)} = P(A \mid G)$$

$$\begin{aligned} P(A, B \mid G) &= P(A, B, G) / P(G) = P(A \mid B, G) \times P(B, G) / P(G) \\ &= \underline{P(A \mid B, G)} \times P(B \mid G) \end{aligned}$$

Conditional Independence



$$P(Cancer|Male) = 65/100{,}000$$

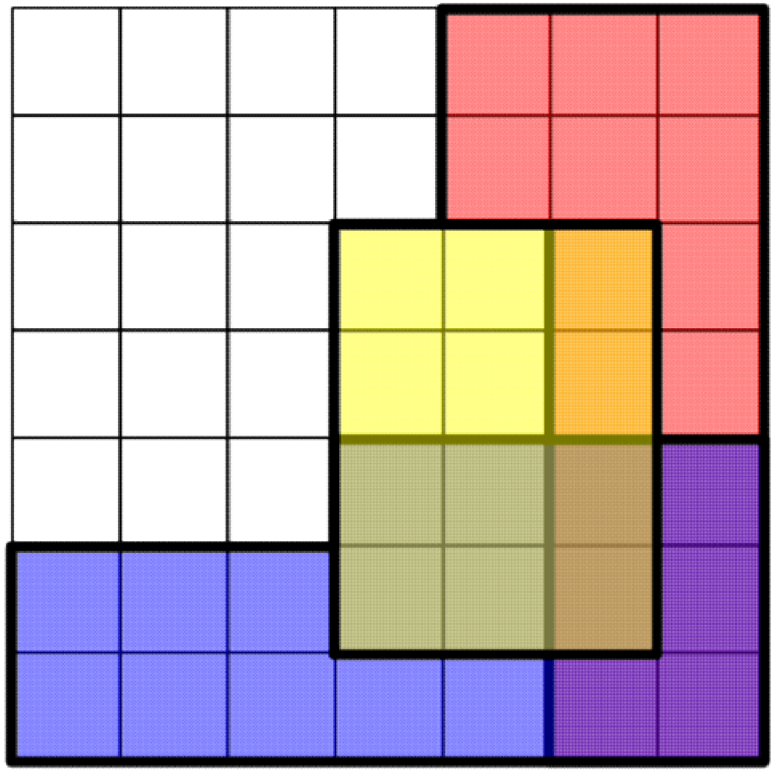
$$P(Cancer|Female) = 48/100{,}000$$

- Are the two events Male/Female and Cancer independent?
- Assume smoking is the sole contributing factor to cancer.

Conditionally Independent

$$P(Cancer|Male, Smoking) = P(Cancer|Smoking)$$

Conditional Independence



$P(R \cap B) = 6/49$

$P(R) = 16/49$

$P(B) = 18/49$

$$\Downarrow$$

$P(R \cap B) \neq P(R)P(B)$

Not Independent

$P(R \cap B|Y) = 1/6$

$P(R|Y) = 1/3$

$P(B|Y) = 1/2$

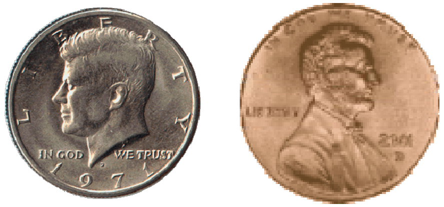
$$\Downarrow$$

$P(R \cap B|Y) = P(R|Y)P(B|Y)$

Conditionally Independent

Conditional Independence

- Two coins: fair vs. biased (two-headed)
- Select one coin at random and toss twice.
- A: First coin toss is head.
- B: Second coin toss is head.
- C: You selected the fair coin.



$$P(A) = P(B) = 0.5 \times 0.5 + 0.5 \times 1.0 = 0.75$$

$$P(C|A) = \frac{P(A|C)P(C)}{P(A|C)P(C)+P(A|\neg C)P(\neg C)} = \frac{0.5\times 0.5}{0.5\times 0.5+1\times 0.5} = \frac{1}{3}$$

$$P(B|A) = \frac{1}{3} \times 0.5 + \frac{2}{3} \times 1.0 = \frac{5}{6} \neq P(B)$$

Not Independent

$$P(B|A,C) = P(B|C) = 0.5$$

Conditionally Independent

Independent ≠ Uncorrelated

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E\big((X-\mu_X)(Y-\mu_Y)\big)}{\sigma_X \sigma_Y}$$

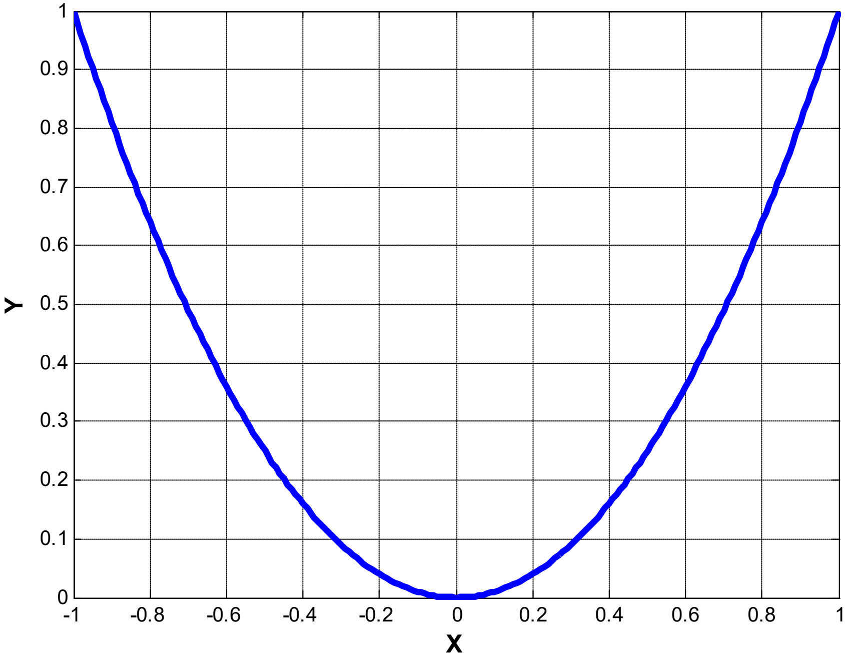
$$X \in [-1,\ 1]$$

$$Y = X^2$$

Cov (X,Y)=0 → X and Y are uncorrelated.

However, Y is completely determined by X.

X	Y
1	1
0.5	0.25
0.2	0.04
0	0
-0.2	0.04
-0.5	0.25
-1	1



Estimating $P(a_j|\omega_i)$

a_1	a_2	a_3	ω
	+		ω_1
			ω_2
	-		ω_1
	+		ω_1
			ω_2

$$P(\omega_1) = 3/5; \qquad P(\omega_2) = 2/5$$

$$P(a_2 = '+' \mid \omega_1) = 2/3$$

$$P(a_2 = '-' \mid \omega_1) = 1/3$$

Laplace Smoothing

$$P(a_{jk} \mid \omega_i) = \frac{|a_j = a_{jk} \wedge \omega = \omega_i| + 1}{|\omega = \omega_i| + |a_j|}$$

How about continuous variables?

Tennis Example

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
Day1	Sunny	Hot	High	Weak	*No*
Day2	Sunny	Hot	High	Strong	*No*
Day3	Overcast	Hot	High	Weak	Yes
Day4	Rain	Mild	High	Weak	Yes
Day5	Rain	Cool	Normal	Weak	Yes
Day6	Rain	Cool	Normal	Strong	*No*
Day7	Overcast	Cool	Normal	Strong	Yes
Day8	Sunny	Mild	High	Weak	*No*
Day9	Sunny	Cool	Normal	Weak	Yes
Day10	Rain	Mild	Normal	Weak	Yes
Day11	Sunny	Mild	Normal	Strong	Yes
Day12	Overcast	Mild	High	Strong	Yes
Day13	Overcast	Hot	Normal	Weak	Yes
Day14	Rain	Mild	High	Strong	*No*

Tennis Example

Given:

$<\text{Outlook} = \mathit{sunny}, \text{Temperatue} = \mathit{cool}, \text{Humidity} = \mathit{high}, \text{Wind} = \mathit{strong}>$

Predict:

$\mathit{PlayTennis}\,(\mathit{yes\ or\ no})$

Bayes Solution :

$$P(PlayTennis = yes) = 9/14$$
$$P(PlayTennis = no) = 5/14$$
$$P(Wind = strong \mid PlayTennis = yes) = 3/9$$
$$P(Wind = strong \mid PlayTennis = no) = 3/5$$

...

$$P(yes)P(sunny \mid yes)P(cool \mid yes)P(high \mid yes)P(strong \mid yes) = 0.0053$$
$$P(no)P(sunny \mid no)P(cool \mid no)P(high \mid no)P(strong \mid no) = 0.0206$$

The conclusion is not to play tennis with probability: $\dfrac{0.0206}{0.0206 + 0.0053} = 0.795$

Text Classification Example

Interesting? Boring?

Politics? Entertainment? Sports?

Text Representation

a_1	a_2	a_3	a_4	…	a_n	ω
Long	long	ago	there	…	king	1
New	sanctions	will	be	…	Iran	0
Hidden	Markov	models	are	…	method	0
The	Federal	Court	today	…	investigate	0

We need to estimate probabilities such as $P(a_2 = king|\omega = 1)$.

However, there are 2×n×|Vocabulary| terms in total. For n=100 and a vocabulary of 50,000 distinct words, it adds up to 10 million terms!

Text Representation

- By only considering the probability of encountering a specific word instead of the specific word position, we can reduce the number of probabilities to be estimated.

- We only count the frequency of each word.

- Now, 2×50,000=100,000 terms need to be estimated.

$$P(V_K \mid \omega = \omega_i) = \frac{n_k + 1}{n + |Vocabulary|}$$

- *n*: the total number of word positions in all training samples whose target value is ω_i.

- n_k: the number of times word V_k is found among these *n* positions.

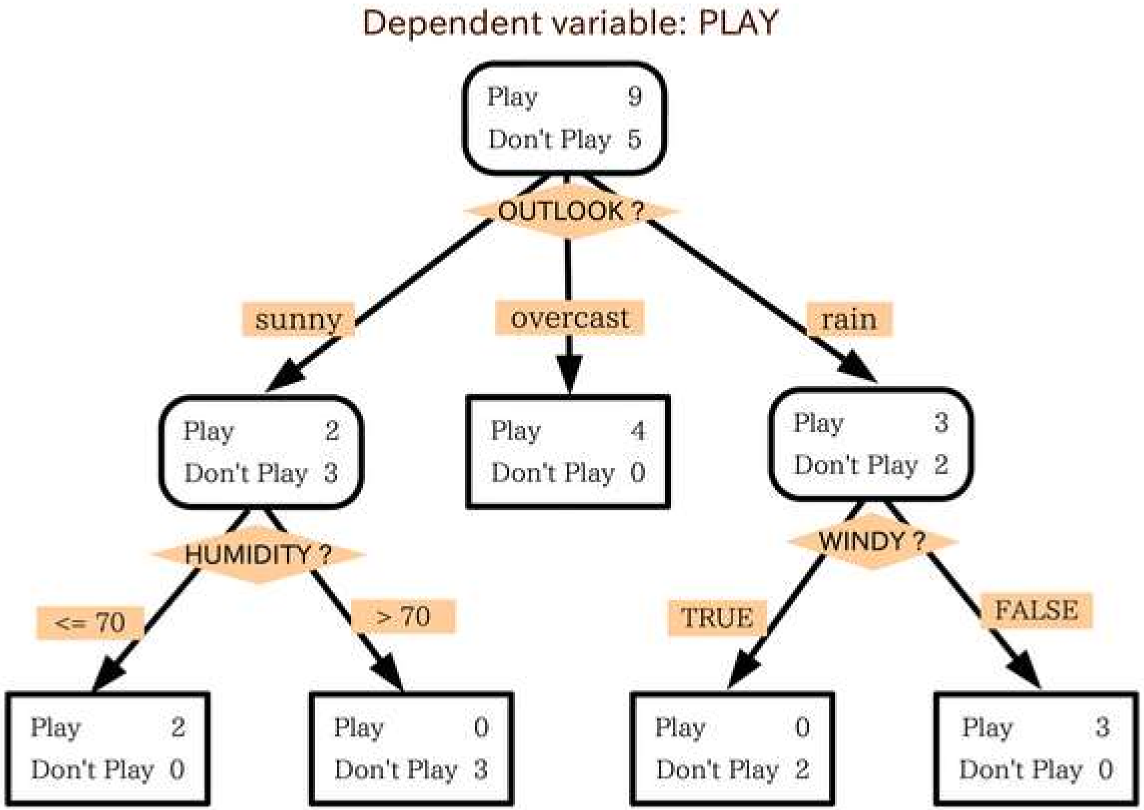
Case Study: Newsgroups

- Classification
  - Joachims, 1996
  - 20 newsgroups
  - 20,000 documents
  - Random Guess: 5%
  - NB: 89%

- Recommendation
  - Lang, 1995
  - *NewsWeeder*
  - User rated articles
  - Interesting vs. Uninteresting
  - Top 10% selected articles
  - 16% vs. 59%

comp.*
humanities.*
misc.*
news.*
alt.*
rec.*
sci.*
soc.*
talk.*

Decision Making



Dependent variable: PLAY

Play 9
Don't Play 5

OUTLOOK ?

sunny

overcast

rain

Play 2
Don't Play 3

Play 4
Don't Play 0

Play 3
Don't Play 2

HUMIDITY ?

<= 70

> 70

WINDY ?

TRUE

FALSE

Play 2
Don't Play 0

Play 0
Don't Play 3

Play 0
Don't Play 2

Play 3
Don't Play 0

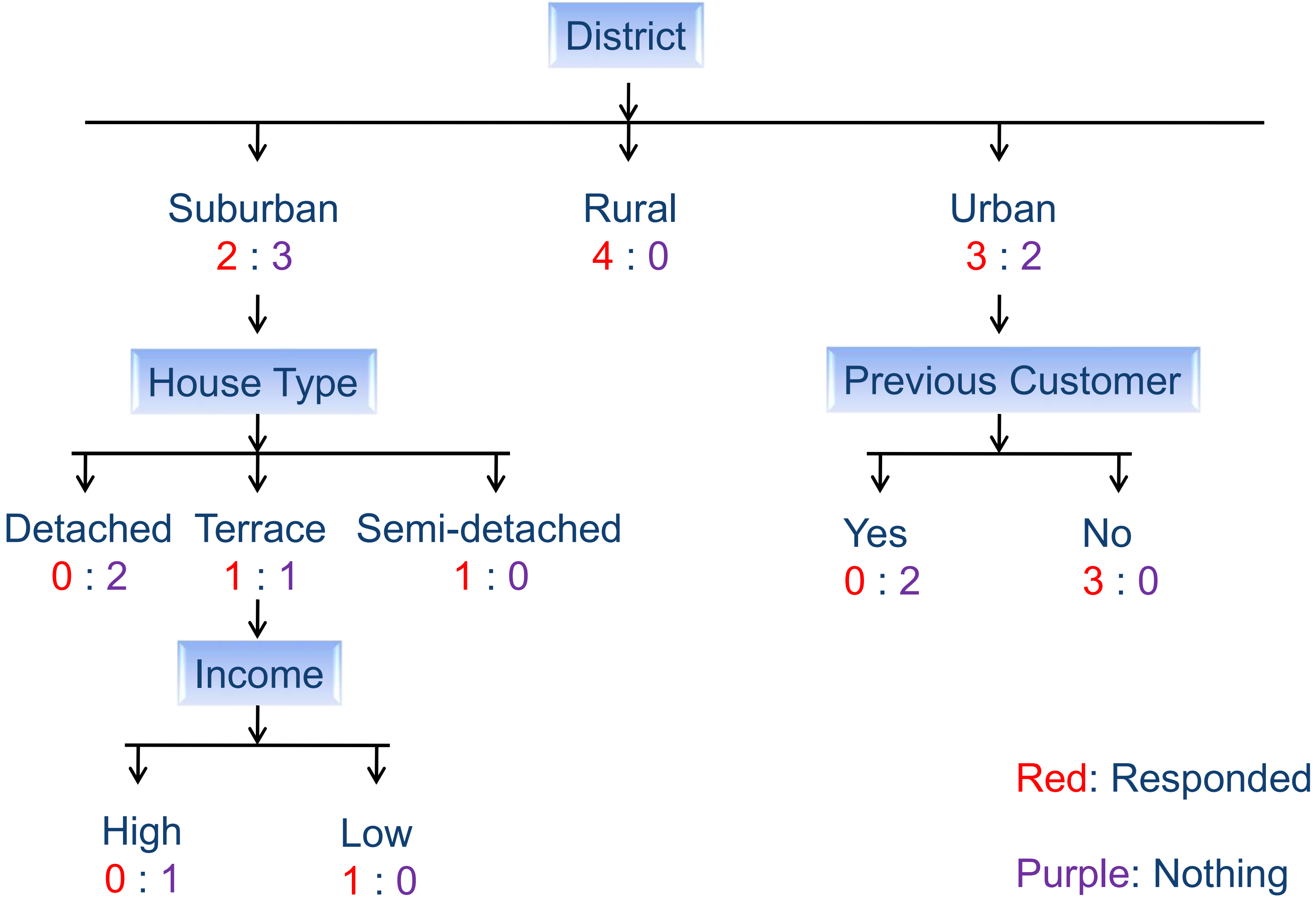
A Survey Dataset

- Given the data collected from a promotion activity.
  - Could be tens of thousands of such records.
- Can we find any interesting patterns?
  - All rural households responded …
- To find out which factors most strongly affect a household’s response to a promotion.
  - Better understanding of potential customers
- Need a classifier to examine the underlying relationships and make future predictions.
- Send promotion brochures to selected households next time.
  - Targeted Marketing

A Survey Dataset

District	House Type	Income	Previous Customer	Outcome
Suburban	Detached	High	No	Nothing
Suburban	Detached	High	Yes	Nothing
Rural	Detached	High	No	Responded
Urban	Semi-detached	High	No	Responded
Urban	Semi-detached	Low	No	Responded
Urban	Semi-detached	Low	Yes	Nothing
Rural	Semi-detached	Low	Yes	Responded
Suburban	Terrace	High	No	Nothing
Suburban	Semi-detached	Low	No	Responded
Urban	Terrace	Low	No	Responded
Suburban	Terrace	Low	Yes	Responded
Rural	Terrace	High	Yes	Responded
Rural	Detached	Low	No	Responded
Urban	Terrace	High	Yes	Nothing

A Tree Model



District

Suburban 2 : 3

Rural 4 : 0

Urban 3 : 2

House Type

Detached 0 : 2

Terrace 1 : 1

Semi-detached 1 : 0

Income

High 0 : 1

Low 1 : 0

Previous Customer

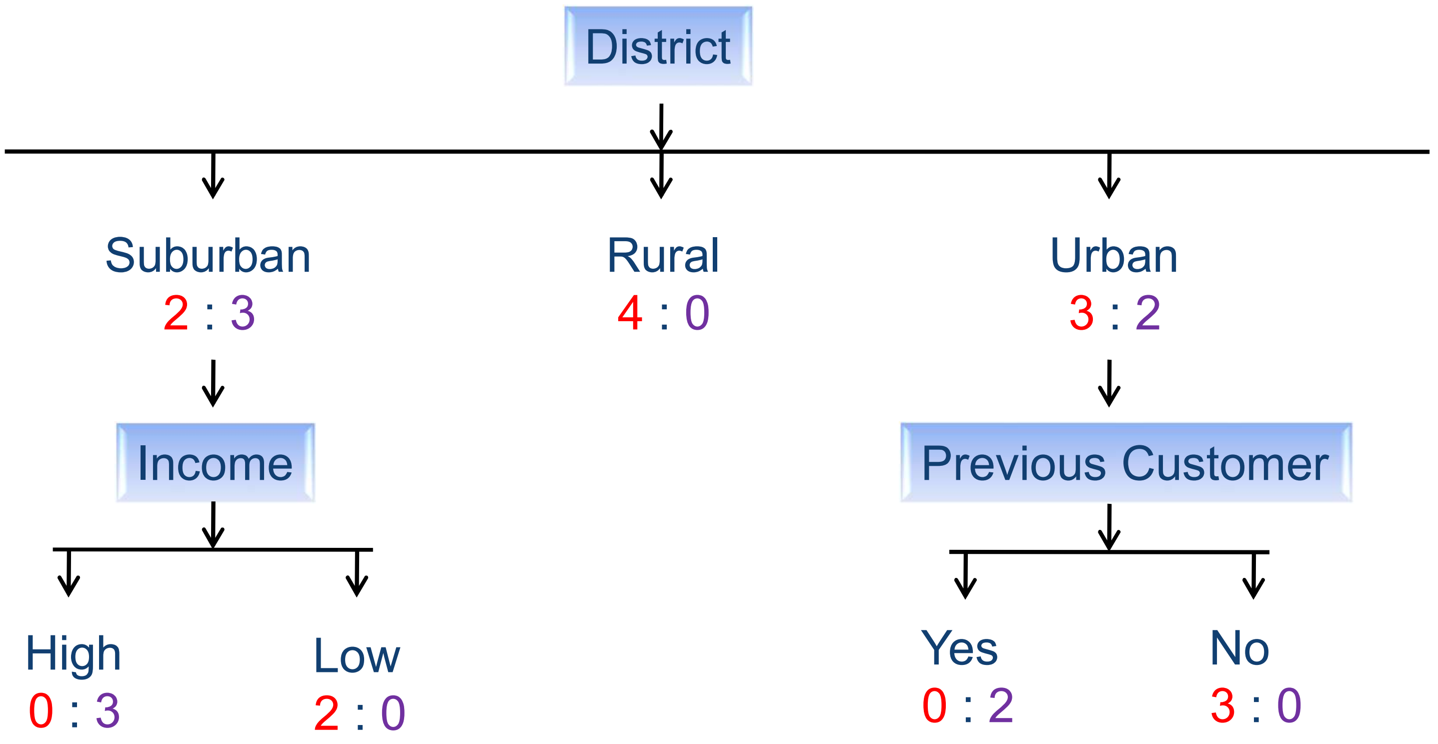
Yes 0 : 2

No 3 : 0

Red: Responded

Purple: Nothing

Another Tree Model



Red: Responded

Purple: Nothing

Some Notes …

- Rules can be easily extracted from the built tree.
  - (District = Rural) → (Outcome = Responded)
  - (District = Urban) AND (Previous Customer = Yes) → (Outcome = Nothing)

- One dataset, many possible trees

- Occam's Razor
  - The term *razor* refers to the act of shaving away unnecessary assumptions to get to the simplest explanation.
  - "When you have two competing theories that make exactly the same predictions, the simpler one is the better."
  - "The explanation of any phenomenon should make as few assumptions as possible, eliminating those making no difference in the observable predictions of the explanatory hypothesis or theory."

- Simpler trees are generally preferred.

ID3

- How to build a shortest tree from a dataset?
- Iterative Dichotomizer 3
- **Ross Quinlan**: http://www.rulequest.com/
- One of the most influential Decision Trees models
- Top-down, greedy search through the space of possible decision trees
- Since we want to construct short trees …
- It is better to put certain attributes higher up the tree.
- Some attributes split the data more purely than others.
- Their values correspond more consistently with the class labels.
- Need to have some sort of measure to compare candidate attributes.

Entropy

$$Entropy(S) = -\sum_{i=1}^{C} p_i \log(p_i)$$

p_i: the proportion of instances in the dataset that take the i th target value

$$S = [9/14\,(responses), 5/14\,(no\ responses)]$$

$$Entropy(S) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0.940$$

$$Gain(S, A) = Entropy(S) - \sum_{v \in A} \frac{|S_v|}{|S|} Entropy(S_v)$$

S_v: the subset of S where attribute A takes the value v.

Attribute Selection

$$Gain(S, District) = Entropy(S) - \frac{5}{14} Entropy(S_{District=Suburban})$$

$$-\frac{5}{14} Entropy(S_{District=Ur} \quad) - \frac{4}{14} Entropy(S_{District=Rural})$$

$$= 0.940 - \frac{5}{14} \cdot 0.971 - \frac{5}{14} \cdot 0.971 - \frac{4}{14} \cdot 0 = 0.247$$

$$Gain(S, Income) = Entropy(S) - \frac{7}{14} Entropy\left(S_{Income=High}\right)$$

$$-\frac{7}{14} Entropy(S_{Income=Lo} \quad)$$

$$= 0.940 - \frac{7}{14} \cdot 0.9852 - \frac{7}{14} \cdot 0.5917 = 0.152$$

ID3 Framework

- **ID3(Examples, Target_attribute, Attributes)**

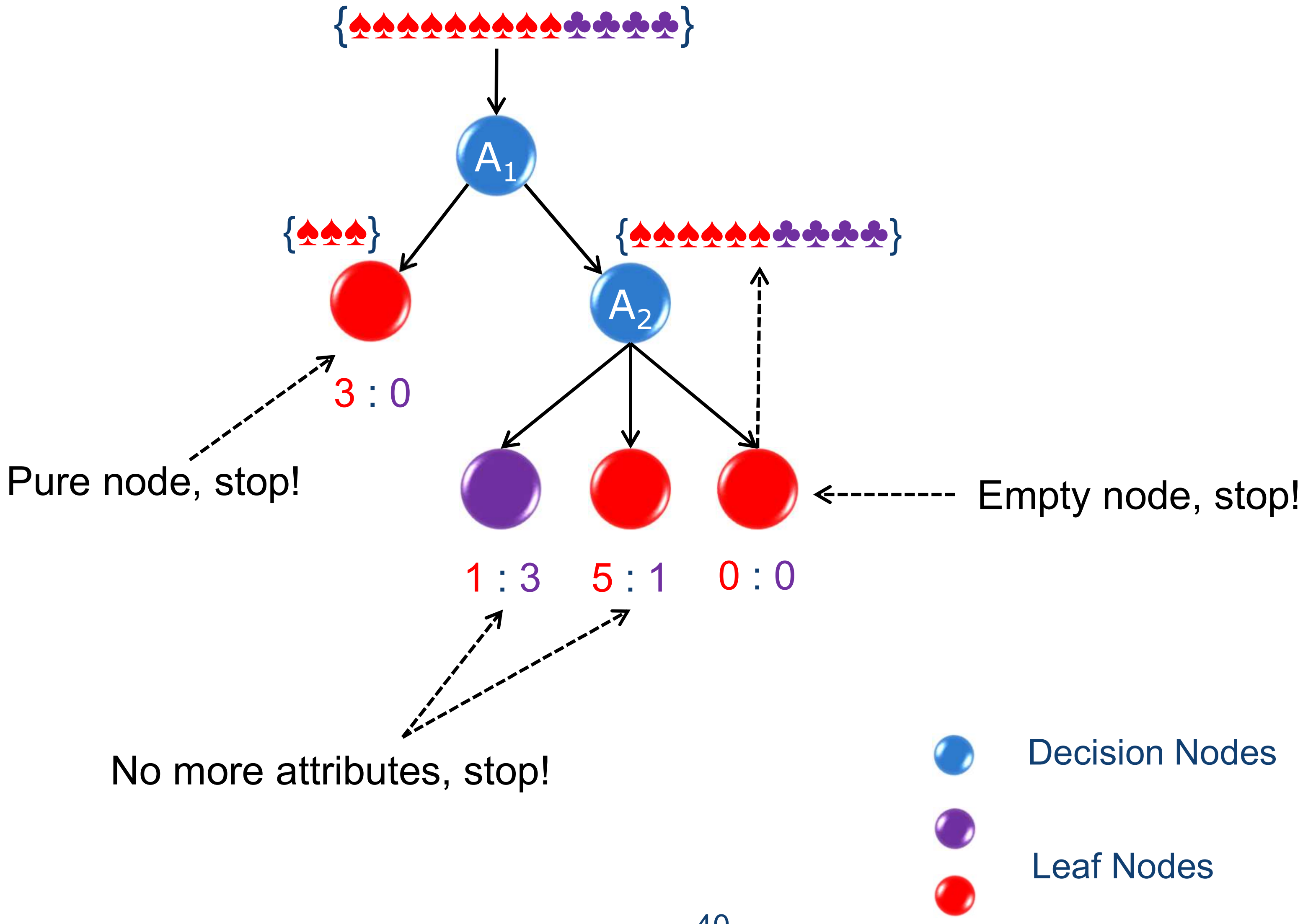
- Create a *Root* node for the tree.
- If *Examples* have the same target attribute T, return *Root* with label=T.
- If *Attributes* is empty, return *Root* with label=the most common value of *Target_attribute* in *Examples*.

- A ← the attribute from *Attributes* that best classifies *Examples*.
- The decision attribute for *Root* ← A.

- For each possible value v_i of A
  - Add a new tree branch below *Root*, corresponding to A= v_i.
  - Let *Examples* (v_i) be the subset of Examples that have value v_i for A.
  - If *Examples* (v_i) is empty
    - Below this new branch add a leaf node with label=the most common value of *Target_attribute* in *Examples*.
  - Else below this new branch add the subtree
    - **ID3(Examples(v_i), Target_attribute, Attributes-{A})**

- Return *Root*

ID3 Framework



Pure node, stop!

Empty node, stop!

No more attributes, stop!

Overfitting

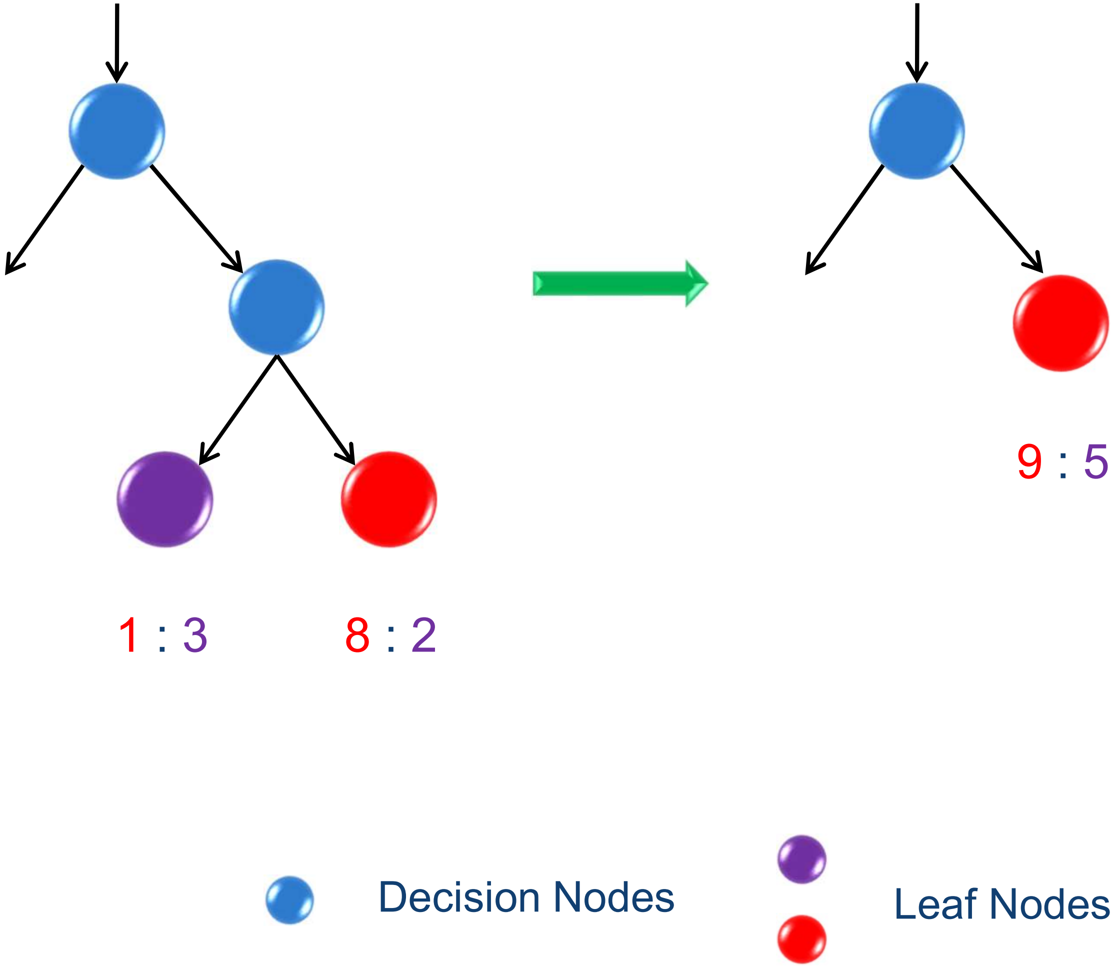
- It is possible to create a separate rule for each training sample.
  - Perfect Training Accuracy vs. Overfitting
  - Random Noise, Insufficient Samples

- We want to capture the general underlying functions or trends.

- Definition
  - Given a hypothesis space *H*, a hypothesis *h* ∈ *H* is said to overfit the training data if there exists some alternative hypothesis *h*' ∈ *H*, such as *h* has smaller error than *h*' over the training samples, but *h*' has a smaller error than *h* over the entire distribution of instances.

- Solutions
  - Stop growing the tree earlier.
  - Allow the tree to overfit the data and then post-prune the tree.

Pruning



Training Set

Validation Set

Test Set

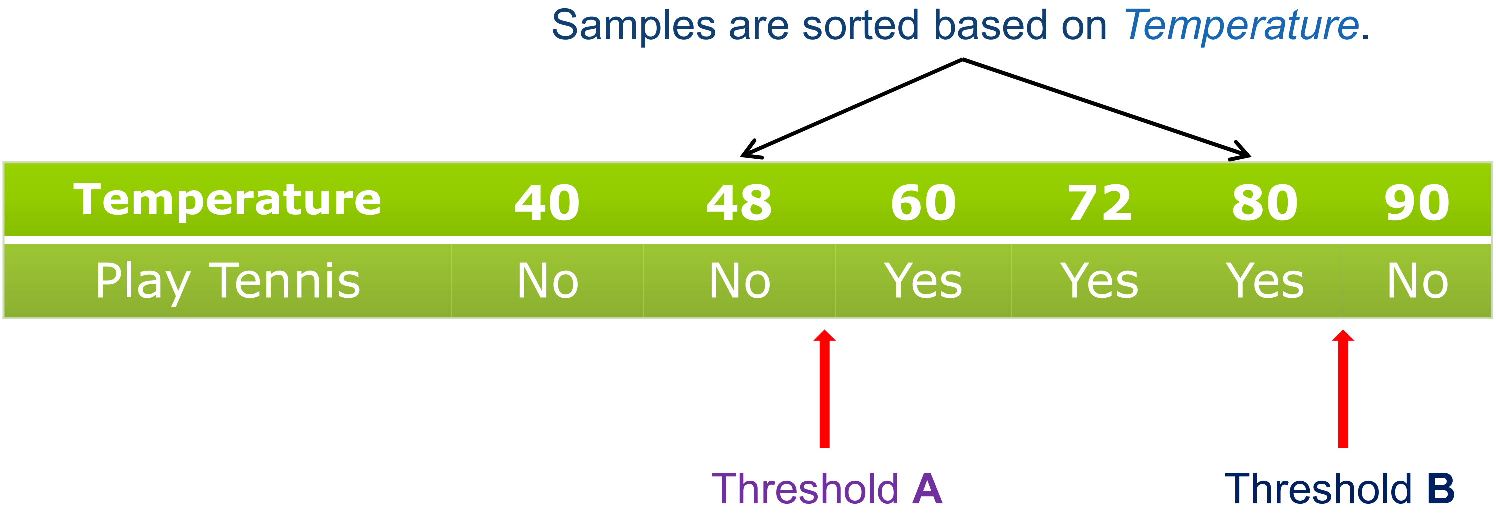
Entropy Bias

- The entropy measure guides the entire tree building process.
- There is a natural bias that favours attributes with many values.
- Consider the attribute “Birth Date”
  - Separate the training data into very small subsets.
  - Very high information gain
  - A very poor predicator of the target function over unseen instances.
- Such attributes need to be penalized!

$$SplitInformation(S,A) = -\sum_{i=1}^{C} \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|}$$

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

Continuous Attributes



Samples are sorted based on *Temperature*.

Temperature	40	48	60	72	80	90
Play Tennis	No	No	Yes	Yes	Yes	No

Threshold **A**

Threshold **B**

$$Gain(S,A) = Entropy(S) - \frac{1}{3}\cdot 0 - \frac{2}{3}\cdot(-\frac{3}{4}\cdot \log_2 \frac{3}{4} - \frac{1}{4}\cdot \log_2 \frac{1}{4}) = 1 - 0.541 = 0.459$$

$$Gain(S,B) = Entropy(S) - \frac{1}{6}\cdot 0 - \frac{5}{6}\cdot(-\frac{3}{5}\cdot \log_2 \frac{3}{5} - \frac{2}{5}\cdot \log_2 \frac{2}{5}) = 1 - 0.809 = 0.191$$

Reading Materials

- Online Tutorial
  - http://www.decisiontrees.net/node/21 (with interactive demos)
  - http://www.autonlab.org/tutorials/dtree18.pdf
  - http://people.revoledu.com/kardi/tutorial/DecisionTree/index.html
  - http://www.public.asu.edu/~kirkwood/DAStuff/decisiontrees/index.html
- Tom Mitchell, *Machine Learning,* Chapters 3&6, McGraw-Hill.
- Additional reading about Naïve Bayes Classifier
  - http://www-2.cs.cmu.edu/~tom/NewChapters.html
- Software for text classification using Naïve Bayes Classifier
  - http://www-2.cs.cmu.edu/afs/cs/project/theo-11/www/naive-bayes.html