CS440/ECE448: Intro to Artificial Intelligence

Lecture 21: Classification; Decision Trees

Prof. Julia Hockenmaier
juliahmr@illinois.edu

http://cs.illinois.edu/fa11/cs440

Supervised learning: classification

Supervised learning

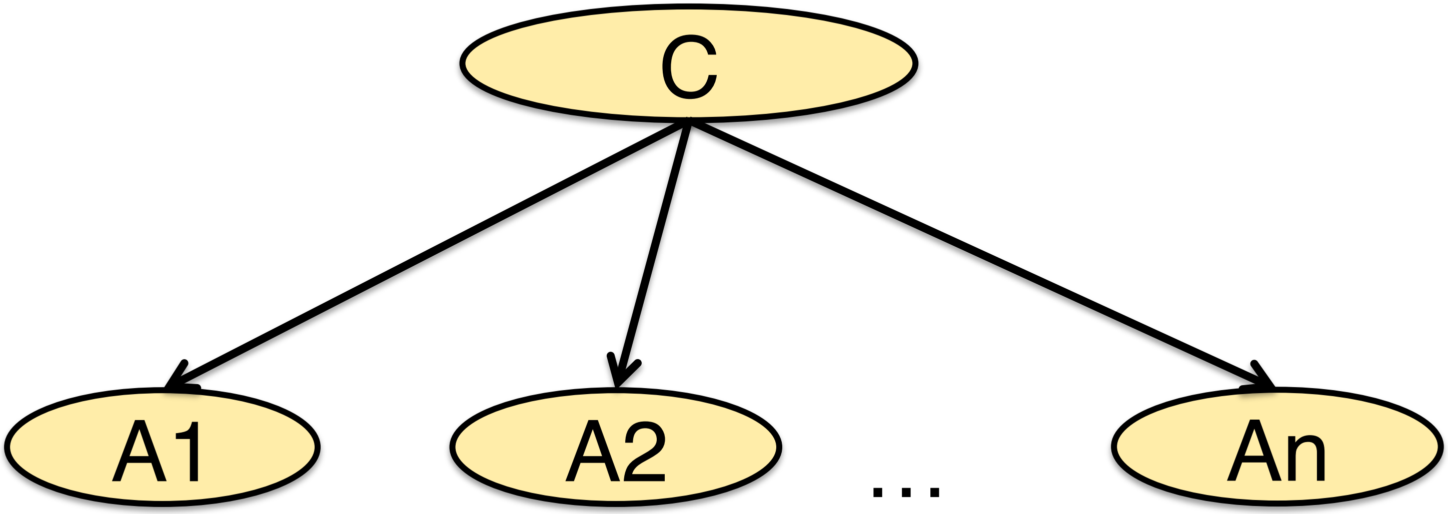
Given a set D of N items $\boldsymbol{x}_i$, each paired with an output value $y_i = f(\boldsymbol{x}_i)$, discover a function $h(\boldsymbol{x})$ which approximates $f(\boldsymbol{x})$

$D = \{(\boldsymbol{x}_1, y_1), \dots (\boldsymbol{x}_N, y_N)\}$

Typically, the **input** values $\boldsymbol{x}$ are (real-valued or boolean) vectors: $\boldsymbol{x}_i \in R^n$ or $\boldsymbol{x}_i \in \{0,1\}^n$

The **output** values y are either boolean *(binary classification)*, elements of a finite set *(multiclass classification)*, or real *(regression)*

The Naïve Bayes Classifier



Each item has a number of attributes

$A_1=a_1,\ldots,A_n=a_n$

We predict the class c based on

$c = \mathrm{argmax}_c \prod_i P(A_i = a_i \mid C=c)\, P(C=c)$

An example

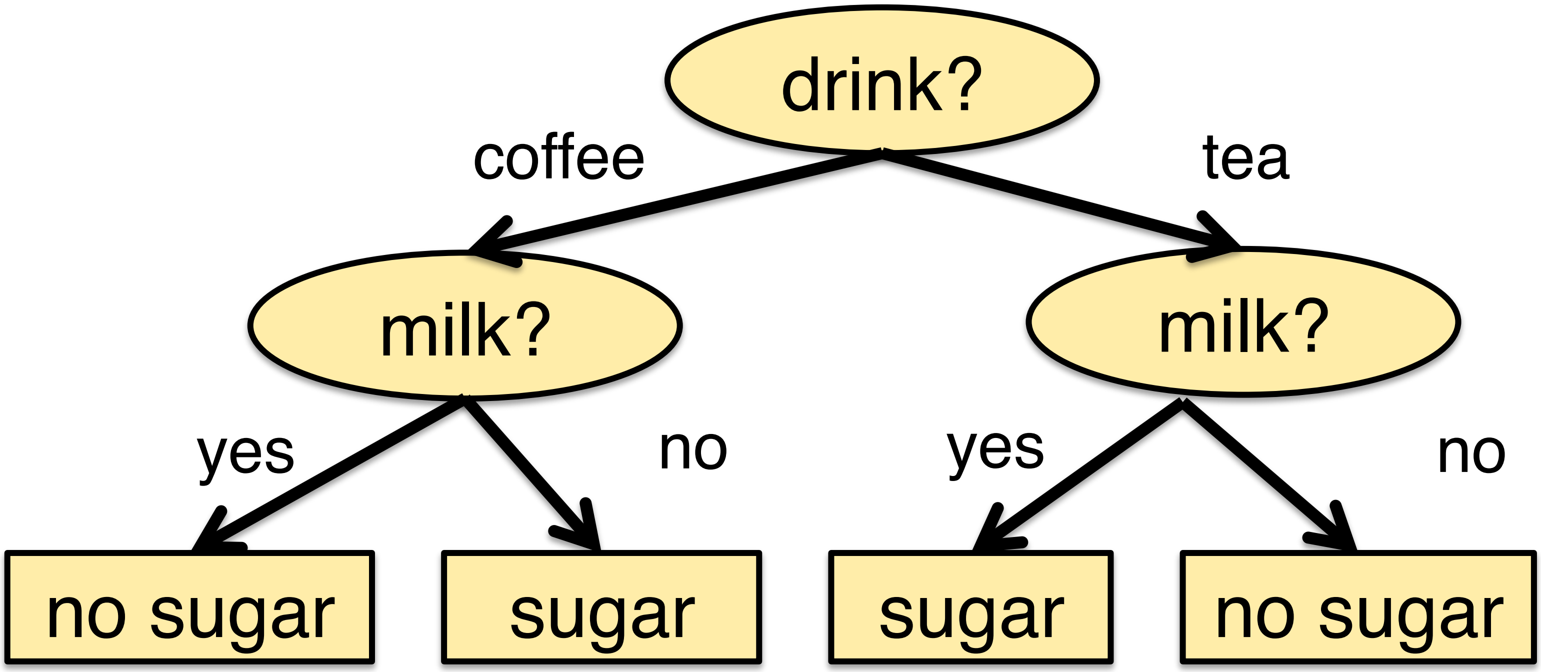
x1	x2	Y
A1: drink	**A2: milk?**	**C: sugar?**
coffee	no	yes
coffee	yes	no
tea	yes	yes
tea	no	no

Can you train a Naïve Bayes classifier to predict whether the customer wants sugar or not?

What is P(coffee I sugar)?

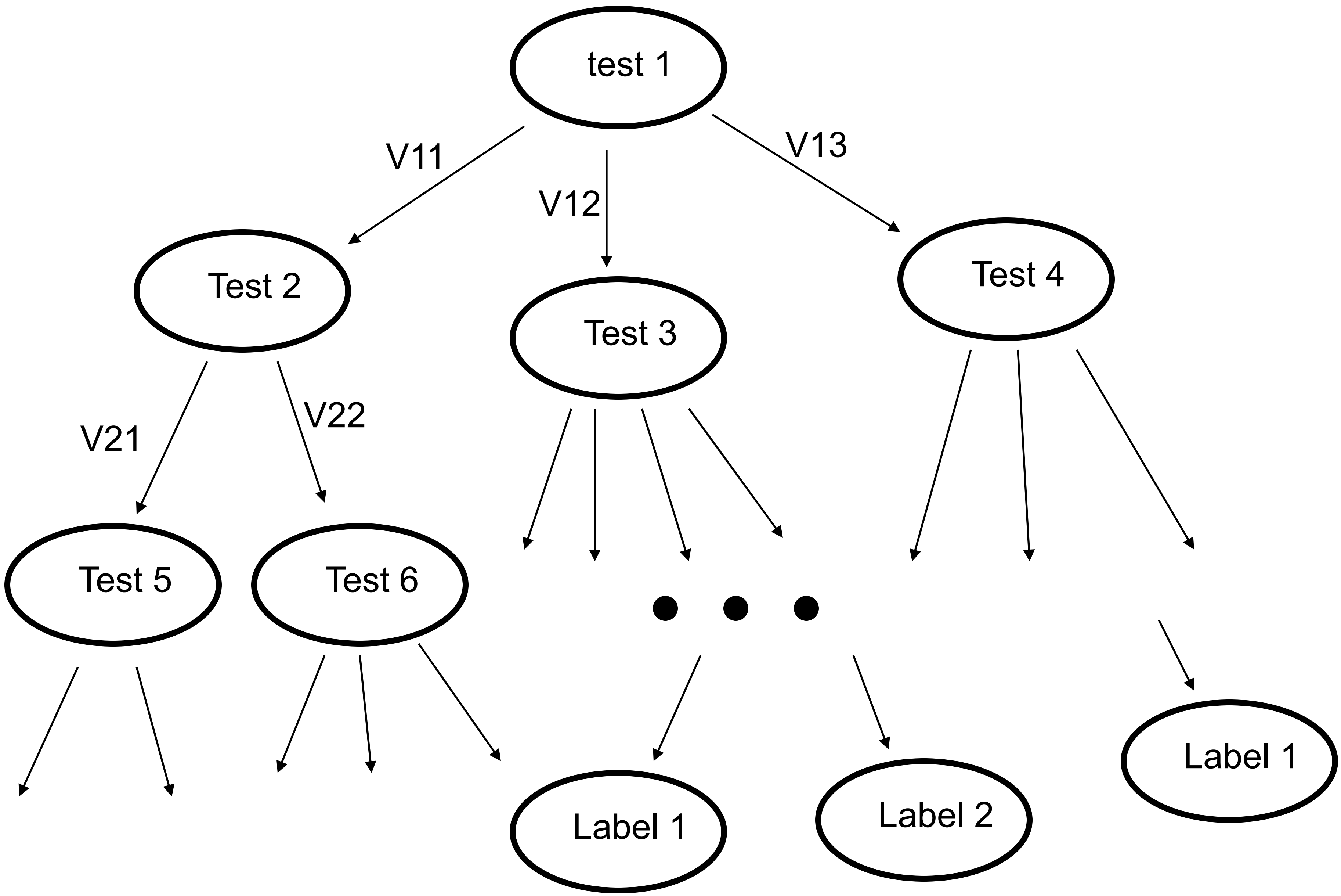
Decision trees

Decision trees



In this example, the attributes (drink; milk?) are not conditionally independent given the class ('sugar')

What is a decision tree?

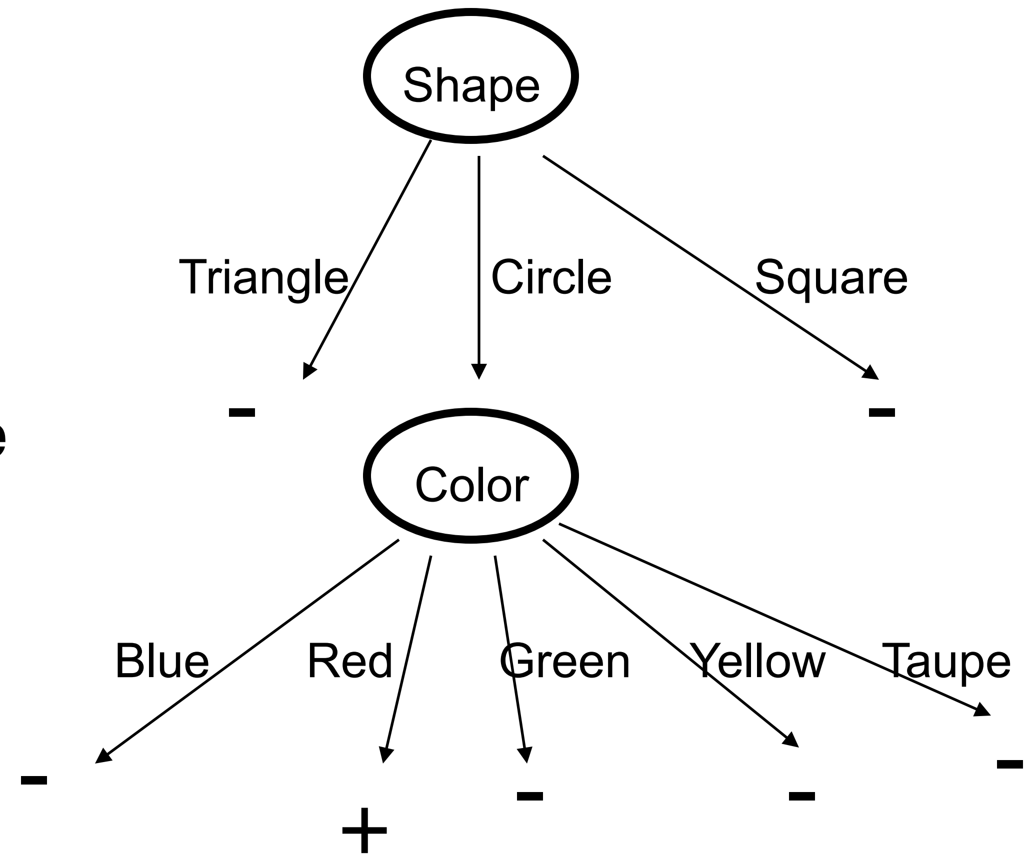


Suppose I like circles that are red

(I might not be aware of the rule)

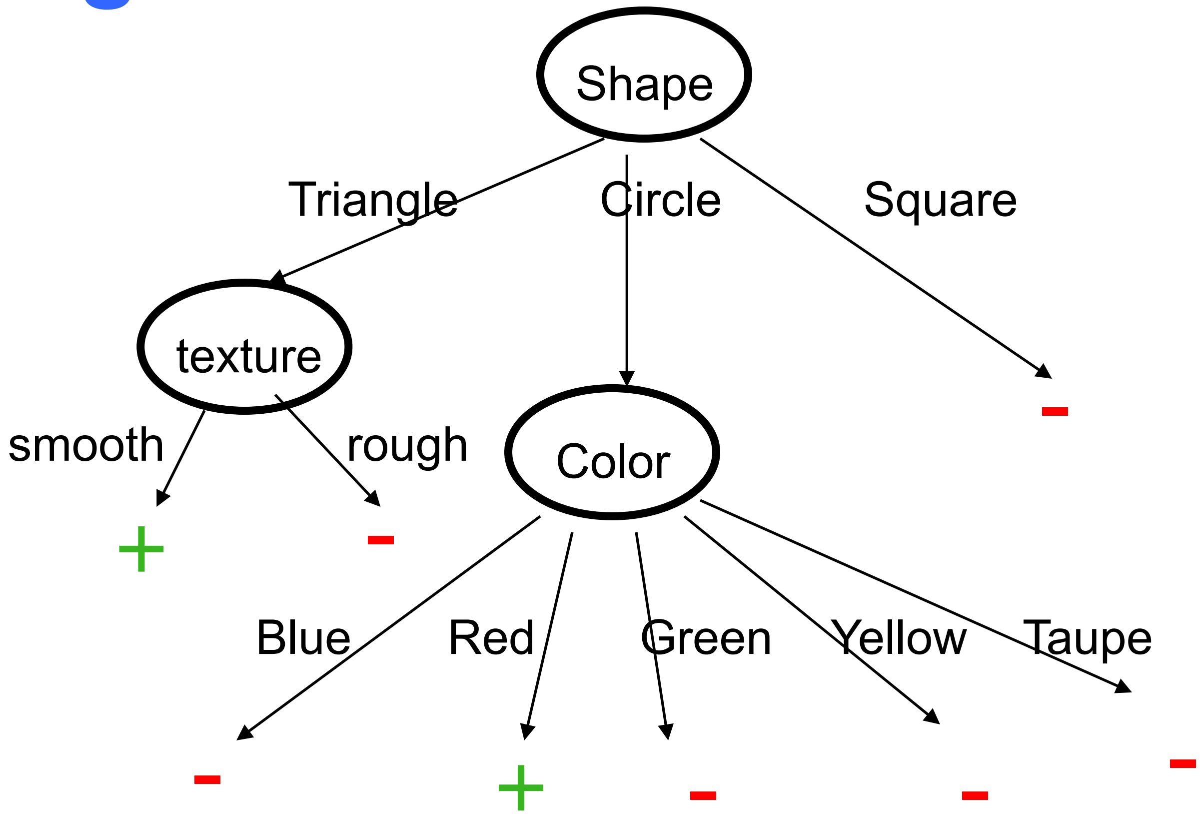
Features:

- **Owner:** John, Mary, Sam
- **Size:** Large, Small
- **Shape**: Triangle, Circle, Square
- **Texture:** Rough, Smooth
- **Color:** Blue, Red, Green, Yellow, Taupe



$$\forall x\ [Like(x) \Leftrightarrow (Circle(x) \wedge Red(x))]$$

Suppose I like circles that are red and triangles that are smooth



$$\forall x\ [Like(x) \Leftrightarrow ((Circle(x) \wedge Red(x))$$

$$\vee\ (Triangle(x) \wedge Smooth(x))]$$

Expressiveness of decision trees

Consider binary classification (y=*true,false*) where the items have Boolean attributes.

In the decision tree, each path from the root to a leaf node is a conjunction of propositions.

The goal (y=*true*) corresponds to a disjunction of such conjunctions (=all the paths from the root to a *true* leaf)

How many different decision trees are there?

With *n* Boolean attributes, there are 2^n possible kinds of examples.

One decision tree = assign *true* to *one subset* of these 2^n kinds of examples.
There are 2^{2^n} different subsets of examples.

There are 2^{2^n} possible decision trees!
(10 attributes: $2^{1024} \approx 10^{308}$ trees;
20 attributes $\approx 10^{300,000}$ trees)

Example space
and hypothesis space

Example space:

The set of all possible examples ***x***

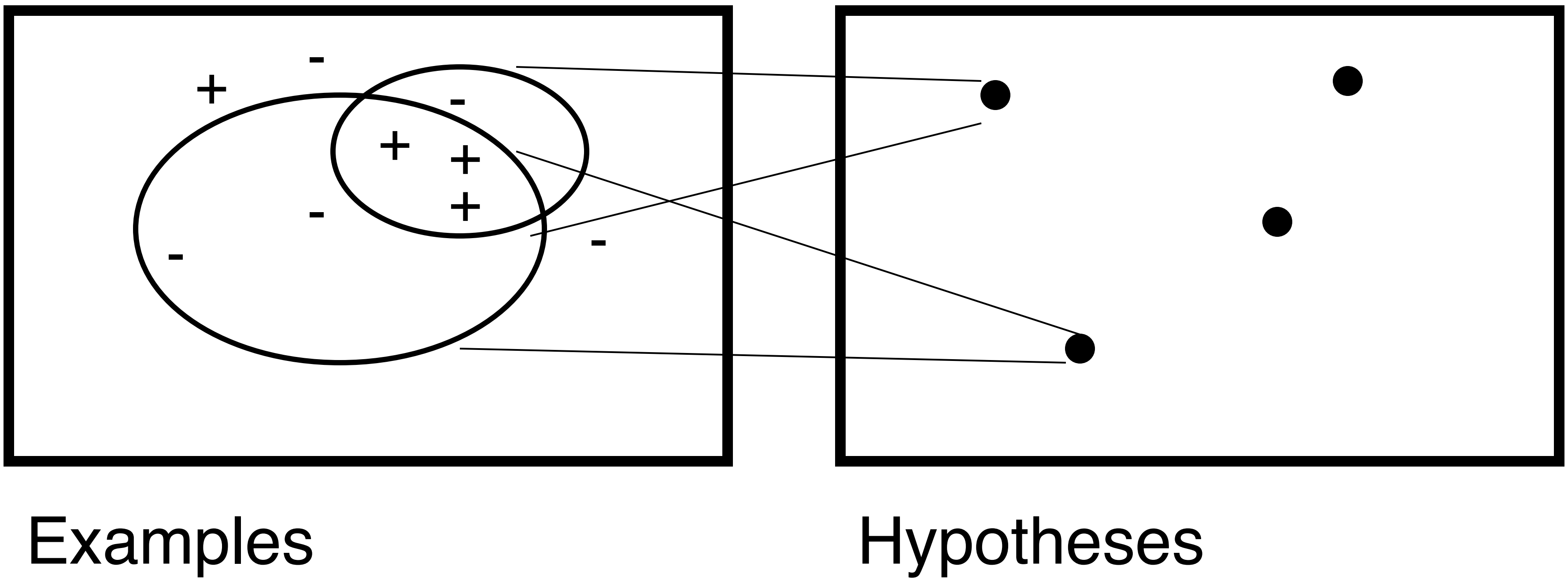
(this depends on our feature representation)

Hypothesis space:

The set of all possible hypotheses *h(**x**)*

that a particular classifier can express.

Machine Learning as an Empirically Guided Search through the Hypothesis Space

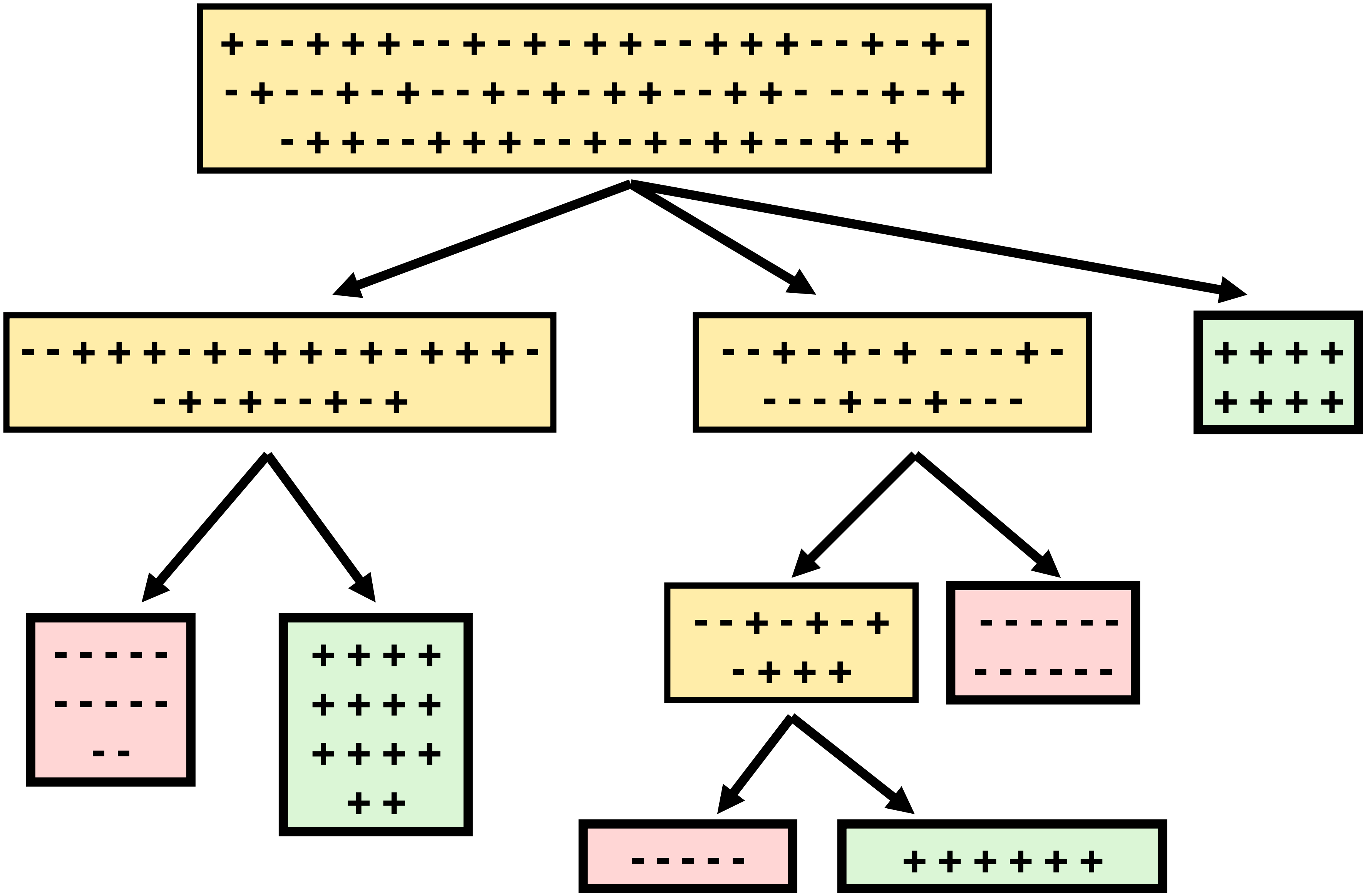


How should we find a good decision tree?

We cannot enumerate all trees.
We will need to do a greedy (local) search.

Tests (splits) need to be informative, i.e. after a split we need to be more certain about which label to assign to the items that meet the test.

Complete Training Data



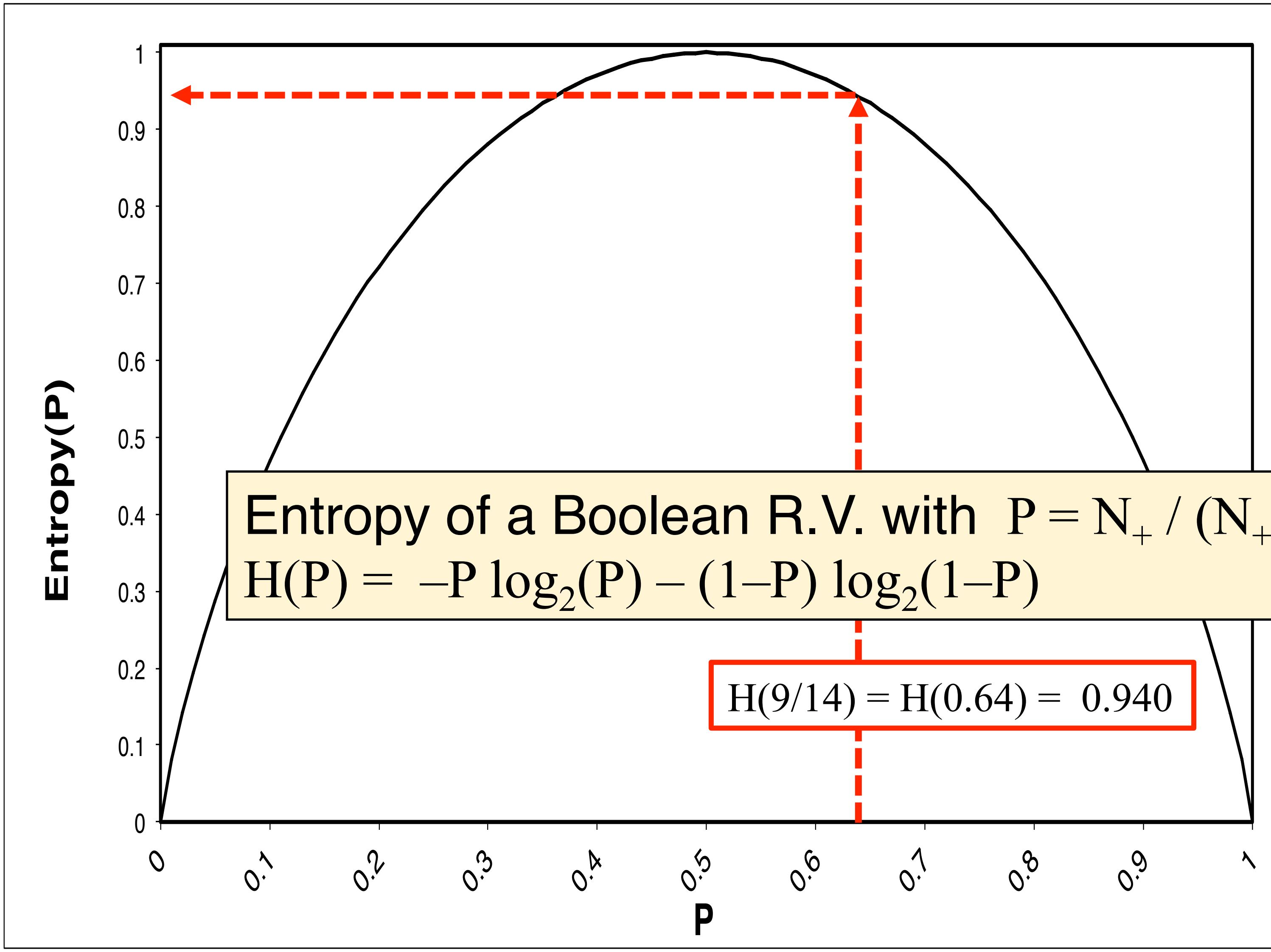
Entropy H(S)

The entropy $H(S)$ of a random variable S measures the uncertainty associated with S.

It also corresponds to the average number of bits required to specify S.

$$H(S) = -\sum_{i=1}^{N} P(s_i)\log_2 P(s_i)$$

Entropy of Boolean R.V.s



Entropy of a Boolean R.V. with $P = N_+ / (N_+ + N_-)$:
$H(P) = -P \log_2(P) - (1-P) \log_2(1-P)$

Measuring Information

H(S) = bits required to label some $x \in S$
What is the upper bound if label $\in \{+,-\}$
What is $H(S_1)$?

$$S_1 = \quad + + +$$

Measuring Information

$H(S)$ = bits required to label some $x \in S$

What is the upper bound if label $\in \{+,-\}$

What is $H(S_1)$?

What is $H(S_2)$?

$S_2 =$ - - -
-

Measuring Information

H(S) = bits required to label some $x \in S$

What is the upper bound if label $\in \{+,-\}$

What is $H(S_1)$?

What is $H(S_2)$?

What is $H(S_3)$?

$$S_3 = \begin{array}{cccccccccccc} + & + & + & + & + & + & + & + & + & + & + & + \\ + & + & + & + & + & + & + & + & + & + & + & + \\ + & + & + & + & + & + & + & + & + & + & + & + \\ + & + & + & + & + & + & + & + & + & + & + & + \end{array}$$

Measuring Information

$H(S)$ = bits required to label some $x \in S$

What is the upper bound if label $\in \{+,-\}$

What is $H(S_1)$?

What is $H(S_2)$?

What is $H(S_3)$?

What is $H(S_4)$?

$S_4 =$ + -

Measuring Information

$H(S)$ = bits required to label some $x \in S$

What is the upper bound if label $\in \{+,-\}$

What is $H(S_1)$?

What is $H(S_2)$?

What is $H(S_3)$?

What is $H(S_4)$?

What is $H(S_5)$?

$$S_5 = \begin{array}{l} + + + + + + + + + + + + \\ + + + + + + + + + + + + \\ - - - - - - - - - - - - \\ - - - - - - - - - - - - \end{array}$$

Measuring Information

H(S) = bits required to label some $x \in S$

What is the upper bound if label $\in \{+,-\}$

What is $H(S_1)$?

What is $H(S_2)$?

What is $H(S_3)$?

What is $H(S_4)$?

What is $H(S_5)$?

What is $H(S_6)$?

S_6 =

\+ + + + + + + + + + + +
\+ + + + + + + + + + + +
\+ + + + + + + + + + + +
\+ + + + + + + + + + + -

Think of *expected* number of bits

$H(S_6)$ should be closer to 0 than to 1

Measuring Information

H(S) = bits required to label some $x \in S$

Label $\in$ {A,B,C,D,E,F}, Upper bound now?

What is $H(S_7)$?

S_7 =

F A B B A A B A
D A A A D A B E
A F A A B B A C
A E B A A A B C

=

A A A A A A A A
A A A A A A A A
B B B B B B B B
C C D D E E F F

16
8
2 2 2 2

FOR	SAY
A	1
B	01
C	0000
D	0001
E	0010
F	0011

Sometimes needs 4 bits / label (worse than 3)

Measuring Information

What is the expected number of bits?

- 16/32 use 1 bit
- 8/32 use 2 bits
- 4 x 2/32 use 4 bits

S_7 =

A A A A A A A A A A A A A A A A	16
B B B B B B B B	8
C C D D E E F F	2 2 2 2

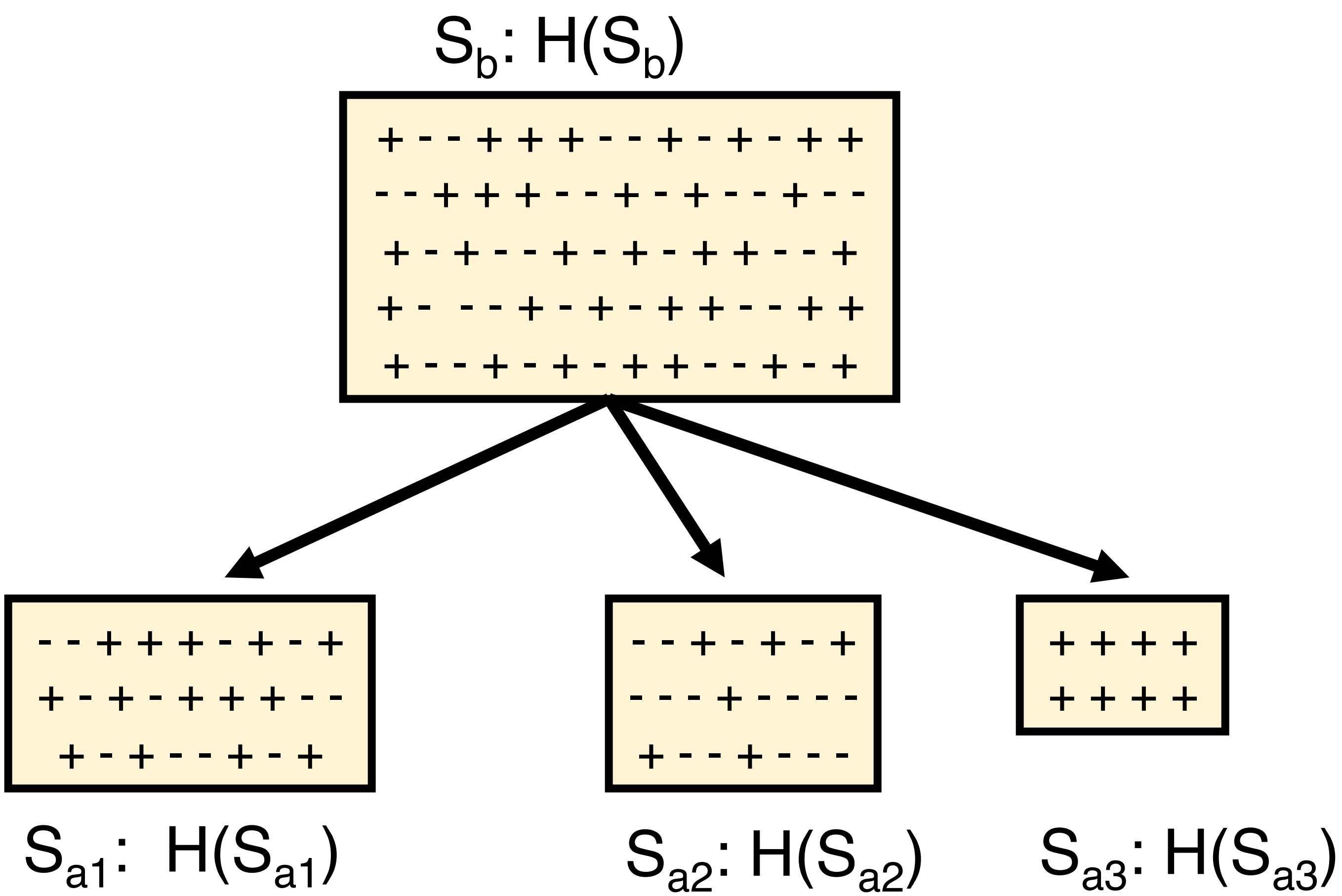
0.5(1) + 0.25(2) + 0.0625(4) +
0.0625(4) + 0.0625(4) + 0.0625(4)

= 0.5 + 0.5 + 0.25 + 0.25 + 0.25 + 0.25
= 2

FOR	SAY
A	1
B	01
C	0000
D	0001
E	0010
F	0011

$$H(S) = \sum_{v \in Labels} -\mathrm{P}(v) \cdot \log_2(\mathrm{P}(v))$$

Information Gain



Information Gain

How much information are we gaining by splitting node S on attribute A with values V(A)?

Information required before the split:

$$H(S_{parent})$$

Information required after the split:

$$\sum_{i \in V(A)} P(S_{child\_i}) H(S_{child\_i})$$

$$Gain(S_{parent}, A) = H(S_{parent}) - \sum_{i \in V(A)}^{N} H(S_{child_i}) \frac{|S_{child_i}|}{|S_{parent}|}$$

An example

Will I Play Tennis?

Features:

- Outlook: Sun, Overcast, Rain
- Temperature: Hot, Mild, Cool
- Humidity: High, Normal, Low
- Wind: Strong, Weak
- Label: +, -

Features are evaluated in the morning

Tennis is played in the afternoon

Training Set

1.	S H H W	-
2.	S H H S	-
3.	O H H W	+
4.	R M H W	+
5.	R C N W	+
6.	R C N S	-
7.	O C N S	+
8.	S M H W	-
9.	S C N W	+
10.	R M N W	+
11.	S M N S	+
12.	O M H S	+
13.	O H N W	+
14.	R M H S	-

Outlook: S, O, R

Temp: H, M, C

Humidity: H, N, L

Wind: S, W

9 + 5 - examples

Current entropy:

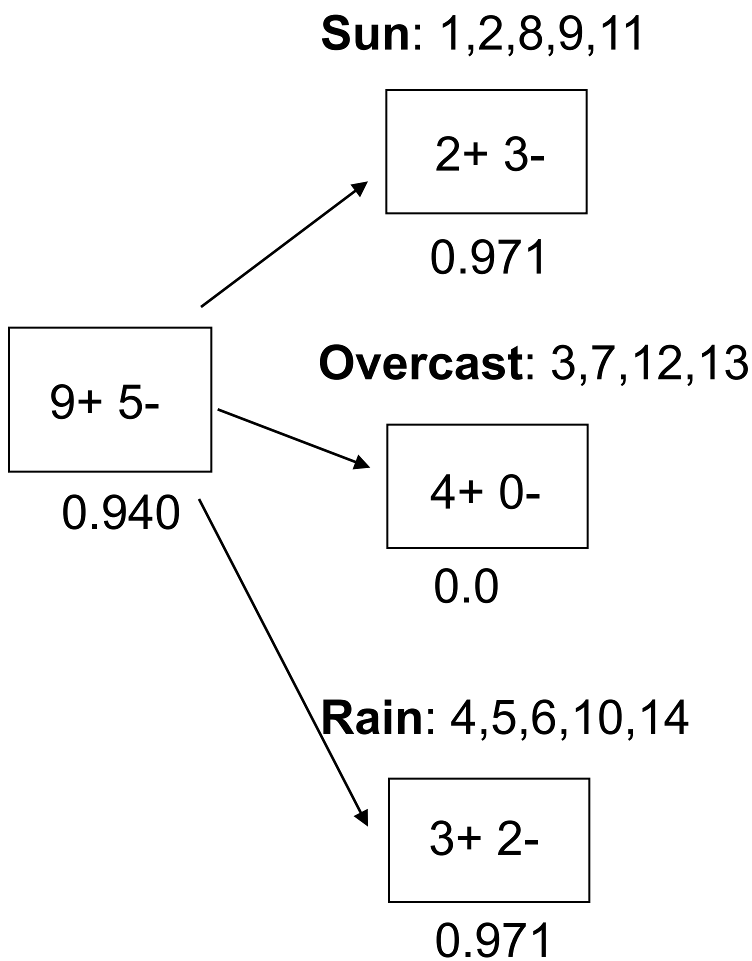
H(9/14)

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

≈ 0.94

Outlook Gain = 0.246

1.	S	H	H	W	-
2.	S	H	H	S	-
3.	O	H	H	W	+
4.	R	M	H	W	+
5.	R	C	N	W	+
6.	R	C	N	S	-
7.	O	C	N	S	+
8.	S	M	H	W	-
9.	S	C	N	W	+
10.	R	M	N	W	+
11.	S	M	N	S	+
12.	O	M	H	S	+
13.	O	H	N	W	+
14.	R	M	H	S	-



Information After:

0.971 * 5/14

+ 0.0 * 4/14

+ 0.971 * 5/14

= 0.694

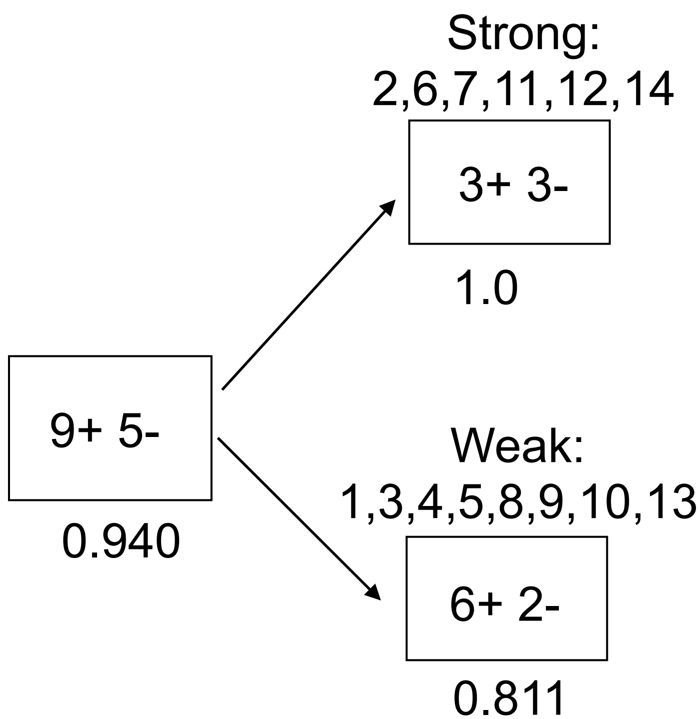
Information Gain:

0.940 – 0.694

= 0.246

Wind Gain = 0.048

1.	S H H	W	-
2.	S H H	S	-
3.	O H H	W	+
4.	R M H	W	+
5.	R C N	W	+
6.	R C N	S	-
7.	O C N	S	+
8.	S M H	W	-
9.	S C N	W	+
10.	R M N	W	+
11.	S M N	S	+
12.	O M H	S	+
13.	O H N	W	+
14.	R M H	S	-



Information After:

1.0 * 6/14

+ 0.811 * 8/14

= 0.892

Information Gain:

0.940 – 0.892

=0.048

Information Gain

• Outlook	0.25
• Temperature	0.03
• Humidity	0.15
• Wind	0.05

Outlook provides greatest local gain

Split on Outlook

S H H W	-	R C N W	+	S C N W	+	O H N W	+
S H H S	-	R C N S	-	R M N W	+	R M H S	-
O H H W	+	O C N S	+	S M N S	+		
R M H W	+	S M H W	-	O M H S	+		

Sunny

S H H W	-
S H H S	-
S M H W	-
S C N W	+
S M N S	+

Overcast

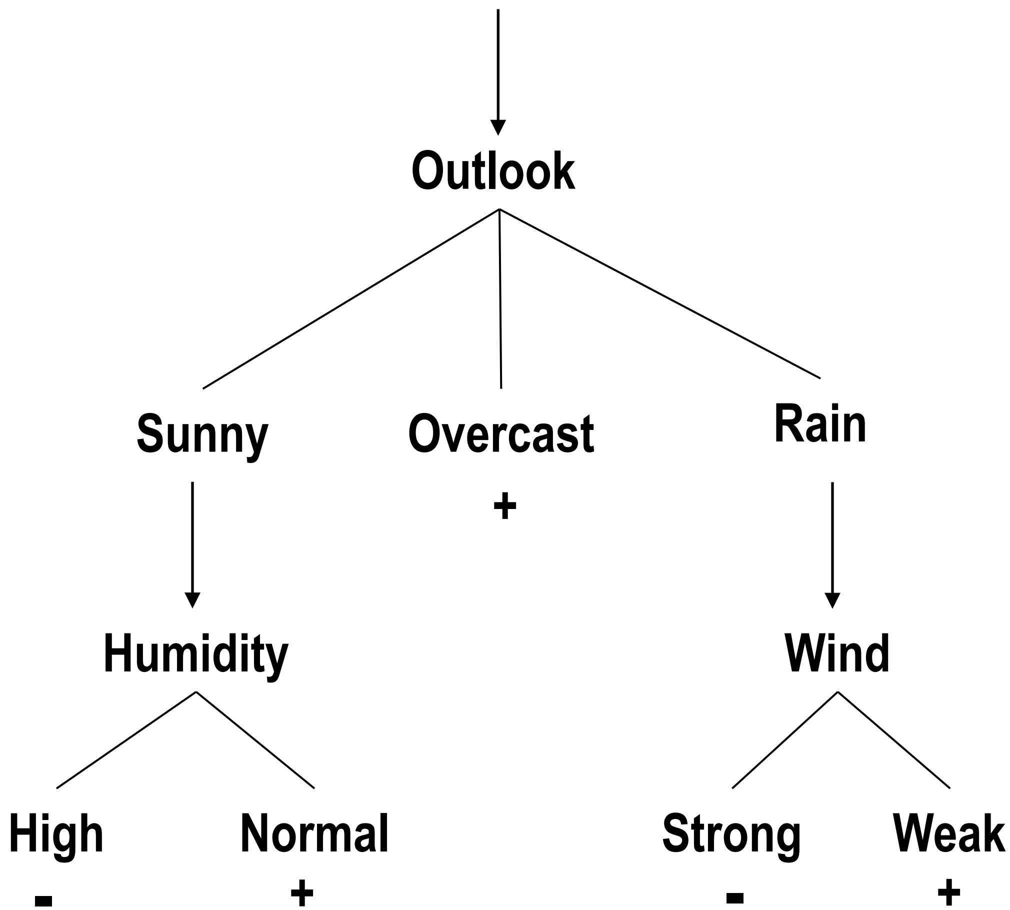
O H H W	+
O C N S	+
O M H S	+
O H N W	+

Rain

R M H W	+
R C N W	+
R C N S	-
R M N W	+
R M H S	-

Now recurse on each smaller set

Final Decision Tree



Suppose under Sunny we split on Outlook (again) instead of Humidity?

What can we say about entropy as we measure additional features?

Learning Decision Trees for Classification

- Ross Quinlan
  - ID3
  - C4.5
  - C5.0 (commercial product)
  - AI / ML
- Breiman, Friedman, Olshen, & Stone
  - CART
  - Statistics

Today's reading

Chapter 18.3