Concept Learning

Concept Learning

Concept Learning

Learning of a boolean function from examples

Categories

- "Nice weather"
- "Dog"
- "Motor vehicle"
- "Criminal offence"

Subsets of a superset X

Terminology

c The concept to learn

$$c(x) \to 0/1, \quad x \in X$$

h Hypothesis, Result of the learning ("guessed c")

$$h(x) \to 0/1, \quad x \in X$$

H Hypotheses space, All conceivable hypotheses (before data arrives)

$$h \in H$$

D Set of available training data

$$D \subseteq X$$

Terminology

Two kinds of training examples

Positive example:

$$x : c(x) = 1, \quad x \in D$$

Negative example:

$$x : c(x) = 0, \quad x \in D$$

Example of a *concept*

"Nice Weather"

Let each "weather instance" x_i be composed of four attributes:

$x_1 =$ <Sunny, Warm, Windy, Dry>
$x_2 =$ <Cloudy, Warm, Calm, Dry>
$x_3 = \cdots$

Generally: *Sky* × *Temperature* × *Wind* × *Humidity*

Assume that the attributes can only take on certain discrete values:

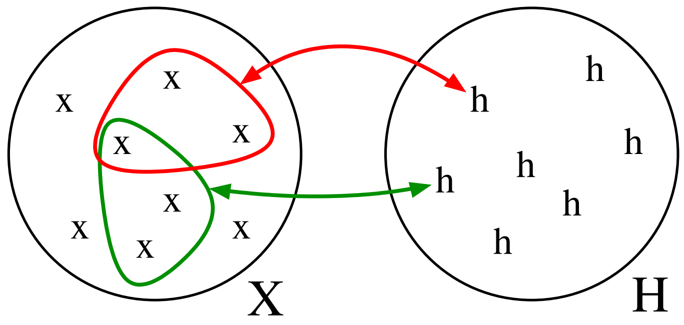
Sky ∈ { Sunny, Cloudy, Rainy }
Temp ∈ { Warm, Cold }
Wind ∈ { Windy, Calm }
Humid ∈ { Humid, Dry }

Number of possible weathers: $|X| = 3 \cdot 2 \cdot 2 \cdot 2 = 24$

Typical training samples

$x_1 =$ <Sunny, Warm, Windy, Dry> $\rightarrow$ Nice
$x_2 =$ <Sunny, Warm, Windy, Humid> $\rightarrow$ Nice
$x_3 =$ <Rainy, Cold, Windy, Humid> $\rightarrow$ Bad
$x_4 =$ <Sunny, Warm, Calm, Humid> $\rightarrow$ Nice

What does the hypotheses space H look like?



Each hypothesis h corresponds to one subset of X

How many hypotheses can we choose from?
How many subsets does X have?

$$|H| = 2^{|X|}$$

$$|H| = 2^{24} = 16777216$$

It is necessary to make restrictions!

Example of a Restriction

Assume that the concept is always a conjunction of attribute values

Examples of concepts c of this kind

Sunny & Warm
Cold & Calm & Dry

How many hypotheses do we have now?

Sky	Temperature	Wind	Humidity
Sunny			
Cloudy	Warm	Windy	Dry
Rainy	Cold	Calm	Humid
$\star$	$\star$	$\star$	$\star$

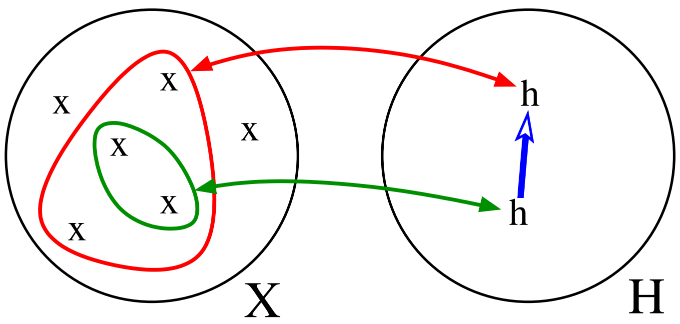
$4 \cdot 3 \cdot 3 \cdot 3 = 108$

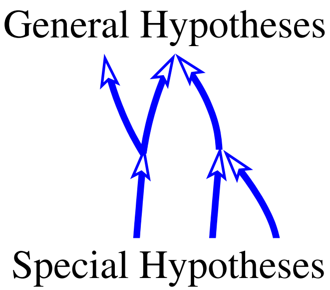
Learning $\equiv$ search for a hypothesis which matches all examples

Use the structure of H to search faster

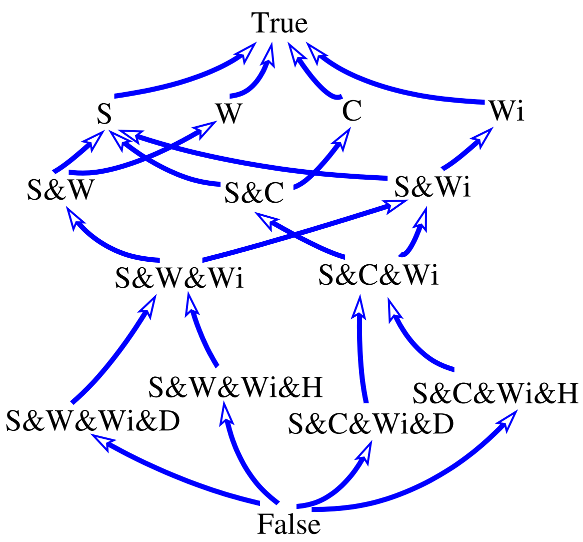
Some hypotheses are more general than others

Partial order between pairs of hypotheses





Most General in our example: "All weathers are nice"
Most Special in our example: "No weather is nice" (!)



Find-S algorithm

Start from the Most Special hypothesis and generalize when necessary.

$\hat{h} \leftarrow$ most special hypothesis in H
for $e \leftarrow$ next example:
  if positive example:
    generalize $\hat{h}$ to cover e too

Returns the most special hypothesis which is consistent with all examples.

Concrete example: "Nice Weather" assuming that this concept is a conjunction of attributes.

Initial Hypothesis: Current Hypothesis: $< \emptyset, \emptyset, \emptyset, \emptyset >$ (Maximally pessimistic) <Sunny, Warm, Windy, Dry> <Sunny, Warm, Windy, $\star >$ <Sunny, Warm, $\star, \star >$

Training examples:

$x_1 =$ <Sunny, Warm, Windy, Dry> $\rightarrow$ Nice
$x_2 =$ <Sunny, Warm, Windy, Humid> $\rightarrow$ Nice
$x_3 =$ <Rainy, Cold, Windy, Humid> $\rightarrow$ Bad
$x_4 =$ <Sunny, Warm, Calm, Humid> $\rightarrow$ Nice

Final hypothesis: "Nice Weather" $\equiv$ Sunny $\wedge$ Warm

Problems with Find-S

- Impossible to know if only one unique hypothesis remains.
- Why should we prefer the most specific hypothesis?
- We will not detect inconsistent data since all negative examples are ignored.
- What happens if there are more equally specific hypotheses?

Version Space (VS)

The set of all hypotheses consistent with the examples seen so far.

- $VS \subseteq H$
- $|VS| = 1$ One unique solution
- $VS = \emptyset$ Inconsistent examples

The List-then-Eliminate algorithm

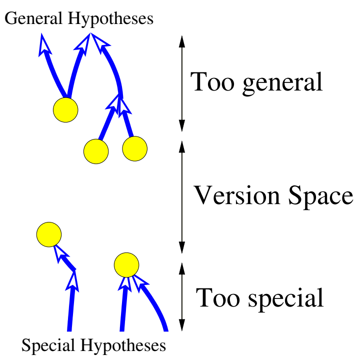
Direct representation of the Version Space (VS)

$VS \leftarrow H$
for $e \leftarrow$ next example:
  remove all hypotheses from VS which
  are not consistent with e

Problem: H is normally too large!

Candidate Elimination

- Efficient representation of the Version Space
- Utilizes the partial ordering between hypotheses.



Bias

Induction Bias

The choice of learning algorithm influences the result

Restriction Bias: Restriction of which hypotheses are allowed

Preference Bias: Tendency to prefer certain hypotheses before others

Unbiased Learner: A learning algorithm without bias
All hypotheses are treated equally

Is an *Unbiased Learner* better?

All subsets of X are equally likely.

Knowledge about $x_1, x_2, \ldots, x_n$ will reveal nothing about x_{n+1}

Without bias it becomes impossible to generalize to unseen examples $x \notin D$.