IA008: Computational Logic

5. Inductive Inference

Achim Blumensath

blumens@fi.muni.cz

Faculty of Informatics, Masaryk University, Brno

Basic Concepts

Induction

learning **general facts** from **examples:**

Induction is the process of forming of a **hypothesis** (about a **target concept/function**) based on **observed data.**

Example

What is the next number?

0, 1, 1, 2, 3, 5, 8,… $a_n = a_{n-2} + a_{n-1}$

0, 0, 0, 0, 0, 120, 720,… $a_n = n(n-1)(n-2)(n-3)(n-4)$

Fundamental Problem

From a strictly logical point of view, induction is **not possible:** there are always several possible explanations for the observed phenomena and there is no rational basis for choosing one over the others. Hence, a hypothesis can be **falsified** but never **verified.**

Consequently we need to make **additional a priori assumptions** (the so-called **inductive bias**) regarding the target concept.

Inductive Learning Hypothesis

A hypothesis that approximates the target concept well over a sufficiently large amount of training data will also approximate it well over unobserved examples.

Occam's Razor

Use the **simplest** hypothesis that matches the observations.

(What's simple depends on our formalism.)

Philosophy of Science

Scientific Method

In the 17th century, **Francis Bacon, René Descartes,** and **Isaac Newton** developed the **scientific method** based on induction.

Problem of Induction

David Hume was the first to point out that inductive inferences are unprovable and always subject to falsification.

Falsifiability

Karl Popper argued that induction does not exist. Instead science is based on **conjecture** and **criticism.** One should select hypotheses that are the easiest to falsify.

Paradigm Shift

Thomas Kuhn viewed science as a **social process.** He emphasised the role of **paradigms** and the way they are replaced when sufficiently many observations point to problems with the current paradigm.

Machine Learning

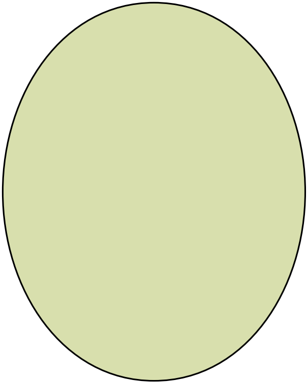
Induction (and learning in general) works best if it is **interactive:**

- form a hypothesis based on the **current data**
- test the hypothesis on **new data**
- repeat

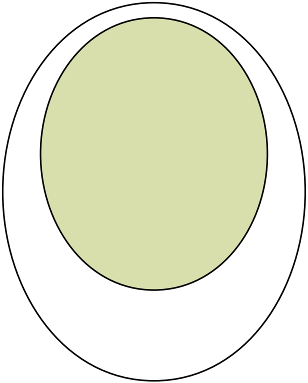
The question therefore is not whether the hypothesis is **true,** but **how well** it predicts observations.

Most decent algorithms for inference use **statistical methods** and fall outside the scope of this course.

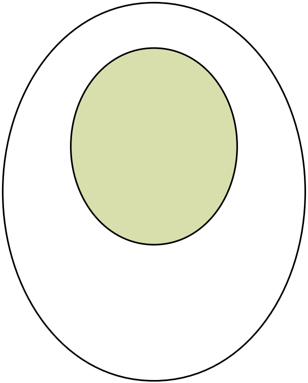
Hypothesis Space



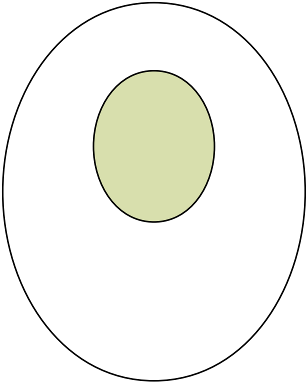
Hypothesis Space



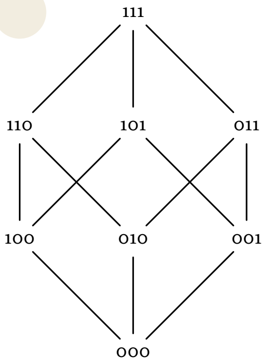
Hypothesis Space



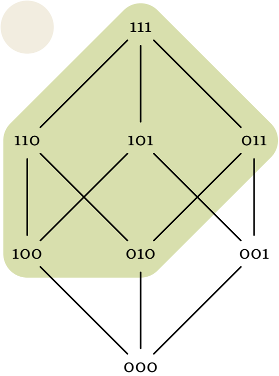
Hypothesis Space



Example

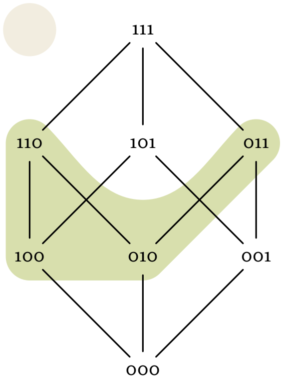


Example



$x \lor y$

Example



$x \vee y$

$\neg x \vee \neg z$

Boolean Functions

Boolean functions

In this lecture we will concentrate on learning **boolean functions**

$f : \{0,1\}^n \to \{0,1\}$

(which can be encoded as propositional formulae)

Example

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$f(\bar{x})$
0	1	0	1	1	1	1	0	0	1	✓
1	0	1	0	0	0	0	1	1	1	✗
1	1	0	0	1	1	1	0	1	0	✗
0	0	0	0	1	0	0	0	1	0	✓
0	0	0	1	1	0	0	1	1	0	✓
0	1	1	1	0	1	1	0	1	1	✗
0	1	0	0	1	0	0	1	0	0	✓

Conjunctive Hypotheses

Setting

Learning a boolean function $f : \{0, 1\}^n \to \{0, 1\}$ using as hypotheses **conjunctions** $\eta := x_i \wedge \cdots \wedge \neg x_k$ of literals.

General-to-specific ordering

η is **more specific** than ζ if $\eta \models \zeta$.

Idea

Find the **most specific** hypothesis.

Conjunctive Hypotheses

Setting

Learning a boolean function $f : \{0,1\}^n \to \{0,1\}$ using as hypotheses **conjunctions** $\eta := x_i \wedge \cdots \wedge \neg x_k$ of literals.

General-to-specific ordering

η is **more specific** than ζ if $\eta \vDash \zeta$.

Idea

Find the **most specific** hypothesis.

Conjunctive Hypotheses

Setting

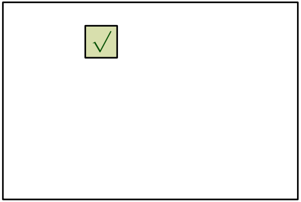
Learning a boolean function $f : \{0,1\}^n \to \{0,1\}$ using as hypotheses **conjunctions** $\eta := x_i \wedge \cdots \wedge \neg x_k$ of literals.

General-to-specific ordering

η is **more specific** than ζ if $\eta \models \zeta$.

Idea

Find the **most specific** hypothesis.



Conjunctive Hypotheses

Setting

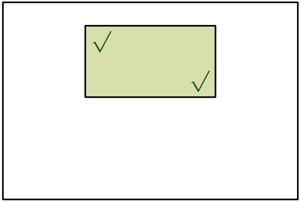
Learning a boolean function $f : \{0, 1\}^n \to \{0, 1\}$ using as hypotheses **conjunctions** $\eta := x_i \wedge \cdots \wedge \neg x_k$ of literals.

General-to-specific ordering

η is **more specific** than ζ if $\eta \models \zeta$.

Idea

Find the **most specific** hypothesis.



Conjunctive Hypotheses

Setting

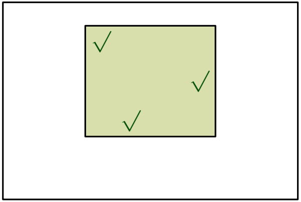
Learning a boolean function $f : \{0, 1\}^n \to \{0, 1\}$ using as hypotheses **conjunctions** $\eta := x_i \wedge \cdots \wedge \neg x_k$ of literals.

General-to-specific ordering

η is **more specific** than ζ if $\eta \vDash \zeta$.

Idea

Find the **most specific** hypothesis.



Conjunctive Hypotheses

Setting

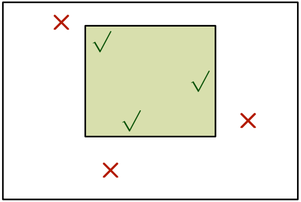
Learning a boolean function $f : \{0,1\}^n \to \{0,1\}$ using as hypotheses **conjunctions** $\eta := x_i \wedge \cdots \wedge \neg x_k$ of literals.

General-to-specific ordering

η is **more specific** than ζ if $\eta \vDash \zeta$.

Idea

Find the **most specific** hypothesis.



Conjunctive Hypotheses

Setting

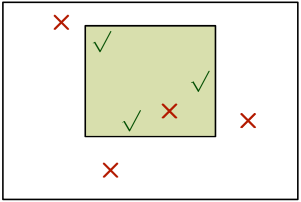
Learning a boolean function $f : \{0,1\}^n \to \{0,1\}$ using as hypotheses **conjunctions** $\eta := x_i \wedge \cdots \wedge \neg x_k$ of literals.

General-to-specific ordering

η is **more specific** than ζ if $\eta \vDash \zeta$.

Idea

Find the **most specific** hypothesis.



Find-S algorithm

- Start with $\eta := \bot$
- Consider the next positive example $\bar{b}$
- If $\eta(\bar{b})$ is true, continue.
- Otherwise, find the most specific ζ such that $\eta \vDash \zeta$ and $\zeta(\bar{b})$ is true.
- Continue with $\eta := \zeta$.

This algorithm computes find the least conjunction with respect to the $\vDash$-ordering that covers all positive examples.
If any of the negative examples is also covered, the training data cannot be described by a conjunction.

Example

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$f(\bar{x})$
0	1	0	1	1	1	1	0	0	1	✓
1	0	1	0	0	0	0	1	1	1	✗
1	1	0	0	1	1	1	0	1	0	✗
0	0	0	0	1	0	0	0	1	0	✓
0	0	0	1	1	0	0	1	1	0	✓
0	1	1	1	0	1	1	0	1	1	✗
0	1	0	0	1	0	0	1	0	0	✓

$\eta_0 := \bot$

$\eta_1 := \neg x_1 \wedge x_2 \wedge \neg x_3 \wedge x_4 \wedge x_5 \wedge x_6 \wedge x_7 \wedge \neg x_8 \wedge \neg x_9 \wedge x_{10}$

$\eta_2 := \neg x_1 \wedge \neg x_3 \wedge x_5 \wedge \neg x_8$

$\eta_3 := \neg x_1 \wedge \neg x_3 \wedge x_5$

$\eta_4 := \neg x_1 \wedge \neg x_3 \wedge x_5$

Hypothesis space

Goal Compute **all** hypotheses consistent with the data.

Let $D \subseteq \{0,1\}^n \times \{0,1\}$ be the observed data and H the set of all hypotheses consistent with every datum in D.

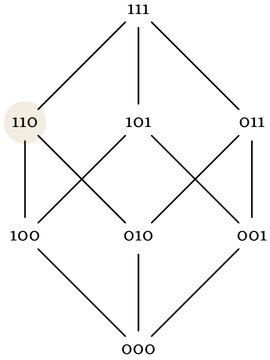
We compute the sets H^+ and H^- of maximal/minimal elements of H (with respect to the general-to-specific order $\models$).

Candidate-Elimination Algorithm

- Start with $H^+ := \{\top\}$ and $H^- := \{\bot\}$.
- For each positive $d \in D$:
  - Delete from H^+ every hypothesis η with $\eta(d) = 0$.
  - Replace every $\eta \in H^-$ with $\eta(d) = 0$ by the set of all minimal ζ such that
    $$\eta \models \zeta\,, \quad \zeta(d) = 1\,, \quad \text{and} \quad \zeta \models \eta'\,, \quad \text{for some } \eta' \in H^+.$$
  - Remove from H^- all elements that are not minimal.
- For each negative $d \in D$: proceed analogously with H^+ and H^- interchanged.

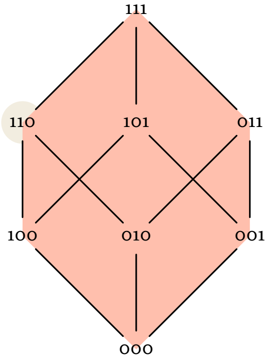
Example

x_1	x_2	x_3	$f(\bar{x})$
1	1	0	√
0	0	1	✗
1	0	0	√
1	0	1	✗



Example

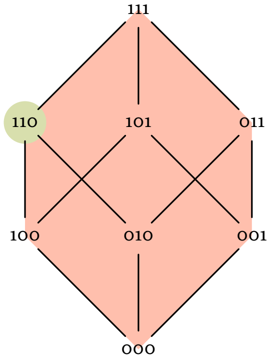
x_1	x_2	x_3	$f(\bar{x})$
1	1	0	$\surd$
0	0	1	×
1	0	0	$\surd$
1	0	1	×



Step 0. $H^- = \{\bot\}$ $H^+ = \{\top\}$

Example

x_1	x_2	x_3	$f(\bar{x})$
1	1	0	√
0	0	1	✕
1	0	0	√
1	0	1	✕

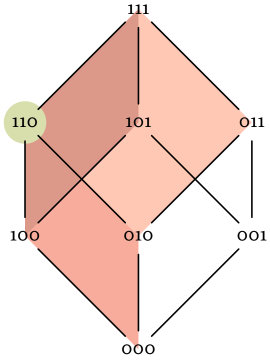


Step 0. $H^- = \{\bot\}$ $H^+ = \{\top\}$

Step 1. $H^- = \{x_1 \wedge x_2 \wedge \neg x_3\}$ $H^+ = \{\top\}$

Example

x_1	x_2	x_3	$f(\bar{x})$
1	1	0	$\surd$
0	0	1	$\times$
1	0	0	$\surd$
1	0	1	$\times$



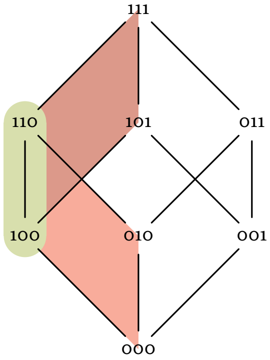
Step 0. $H^- = \{\bot\}$ $H^+ = \{\top\}$

Step 1. $H^- = \{x_1 \wedge x_2 \wedge \neg x_3\}$ $H^+ = \{\top\}$

Step 2. $H^- = \{x_1 \wedge x_2 \wedge \neg x_3\}$ $H^+ = \{x_1,\ x_2,\ \neg x_3\}$

Example

x_1	x_2	x_3	$f(\bar{x})$
1	1	0	√
0	0	1	×
1	0	0	√
1	0	1	×



Step 0. $H^- = \{\bot\}$ $\quad H^+ = \{\top\}$

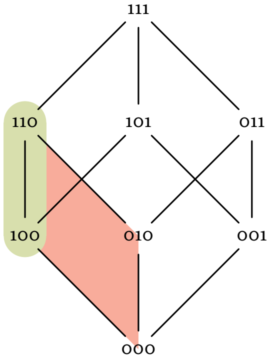
Step 1. $H^- = \{x_1 \wedge x_2 \wedge \neg x_3\}$ $\quad H^+ = \{\top\}$

Step 2. $H^- = \{x_1 \wedge x_2 \wedge \neg x_3\}$ $\quad H^+ = \{x_1,\ x_2,\ \neg x_3\}$

Step 3. $H^- = \{x_1 \wedge \neg x_3\}$ $\quad H^+ = \{x_1,\ \neg x_3\}$

Example

x_1	x_2	x_3	$f(\bar{x})$
1	1	0	√
0	0	1	×
1	0	0	√
1	0	1	×



Step 0. $H^- = \{\bot\} \quad H^+ = \{\top\}$

Step 1. $H^- = \{x_1 \wedge x_2 \wedge \neg x_3\} \quad H^+ = \{\top\}$

Step 2. $H^- = \{x_1 \wedge x_2 \wedge \neg x_3\} \quad H^+ = \{x_1,\ x_2,\ \neg x_3\}$

Step 3. $H^- = \{x_1 \wedge \neg x_3\} \quad H^+ = \{x_1,\ \neg x_3\}$

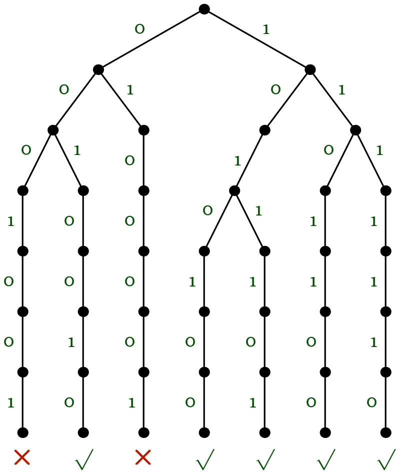
Step 4. $H^- = \{x_1 \wedge \neg x_3\} \quad H^+ = \{\neg x_3\}$

Decision Trees

Decision Trees

Organise the function to be learned as a tree.

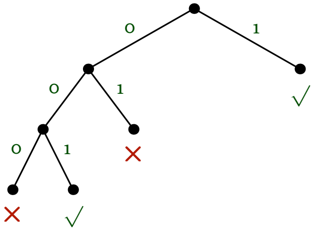
x_1	x_2	x_3	x_4	x_5	x_6	x_7	$f(\bar{x})$
1	0	1	1	1	0	1	✓
0	1	0	0	0	1	1	✗
1	1	1	1	1	1	0	✓
0	0	1	0	0	1	0	✓
0	0	0	1	1	0	1	✗
1	1	0	1	1	0	0	✓
1	0	1	0	1	0	0	✓



Decision Trees

Organise the function to be learned as a tree.

x_1	x_2	x_3	x_4	x_5	x_6	x_7	$f(\bar{x})$
1	0	1	1	1	0	1	√
0	1	0	0	0	1	1	×
1	1	1	1	1	1	0	√
0	0	1	0	0	1	0	√
0	0	0	1	1	0	1	×
1	1	0	1	1	0	0	√
1	0	1	0	1	0	0	√



The order of the variables x_i matters. Which one do we choose?

Ordered Binary Decision Diagrams (OBDDs)

- data structure to compactly represent a boolean function
- the arguments are **ordered** $x_1, \ldots, x_n$
- the graph is **reduced:** merge isomorphic subgraphs and eliminate unneeded vertices

$$(x_1 \wedge x_3) \vee (x_2 \wedge x_3) \vee \neg(x_1 \vee x_2 \vee x_3)$$