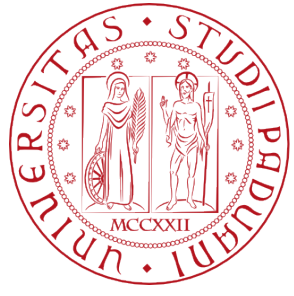
Introduction to Machine Learning

Alessandro Sperduti



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Outline

- Introduction to some basic notions of Machine Learning
- Paradigms, learning tasks, and structured domains
- Statistical Learning Theory
- Kernel Methods
- Neural Networks and Deep Learning
- Links with specific areas of Computer Science
- Software resources and application examples
- Future Directions

Introduction to Machine Learning

- *When* and *Why* to use Machine Learning
- Paradigms
- Fundamental Ingredients
- Statistical Learning Theory

When is Machine Learning necessary ?

When the considered system should...

- adapt to the surrounding environment (also automatic customization)
- improve its performance with respect to a specific computational task
- discover regularities and new information (knowledge) from empirical data
- acquire new computational capabilities

Why should we use Machine Learning ?

Why not to use a traditional algorithmic approach ?

- impossible to exactly formalize the problem to be solved (and so to give an algorithmic solution)
- presence of noise and/or uncertainty
- high complexity in formulating a solution: cannot be done manually
- lack of compiled knowledge with respect to the problem to be solved

Data

Typically...

- data is available
  - obtained once for all (batch learning)
  - acquired incrementally by interacting with the environment (on-line learning)
- (maybe) knowledge of the application domain is available, however
  - incomplete
  - imprecise (noise, ambiguity, uncertainty, errors, ...)

Data

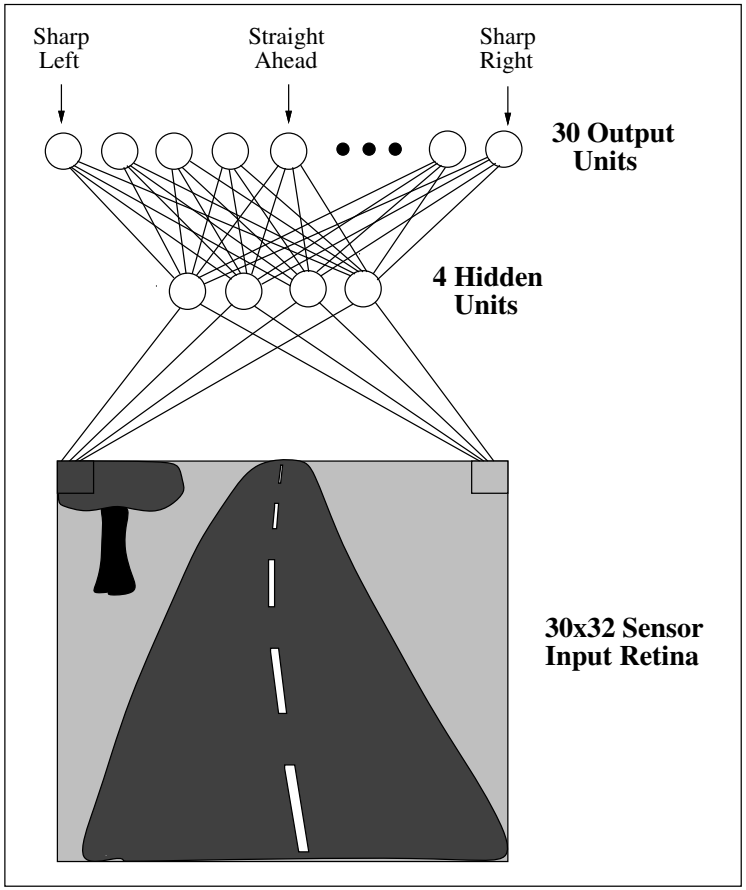
Desiderata: to use data for

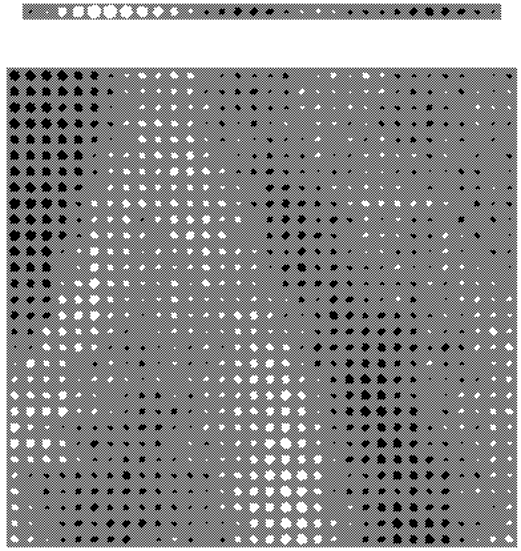
- acquiring new knowledge
- refining the already available knowledge
- correcting the already available knowledge

Example - Handwritten digit recognition

- impossible to exactly formalize the problem: only examples are available
- noise may be present and data may present ambiguities

Example - Autonomous Car





Example - Medical knowledge extraction from data

Patient103 time=1 → *Patient103* time=2 ··· → *Patient103* time=n

Age: 23
FirstPregnancy: no
Anemia: no
Diabetes: no
PreviousPrematureBirth: no
Ultrasound: ?
Elective C–Section: ?
Emergency C–Section: ?
...

Age: 23
FirstPregnancy: no
Anemia: no
Diabetes: YES
PreviousPrematureBirth: no
Ultrasound: abnormal
Elective C–Section: no
Emergency C–Section: ?
...

Age: 23
FirstPregnancy: no
Anemia: no
Diabetes: no
PreviousPrematureBirth: no
Ultrasound: ?
Elective C–Section: no
Emergency C–Section: **Yes**
...

(Some) Lines of research within Machine Learning

- Induction of Rules/Decision Trees from data,
- Neural Networks,
- Clustering & Discovery,
- Instance Based Learning
- Probabilistic (Bayesian) learning,
- Reinforcement learning,
- Genetic Algorithms,
- Inductive Logic Programming,
- ... (many more)

Main Learning Paradigms

Supervised Learning:

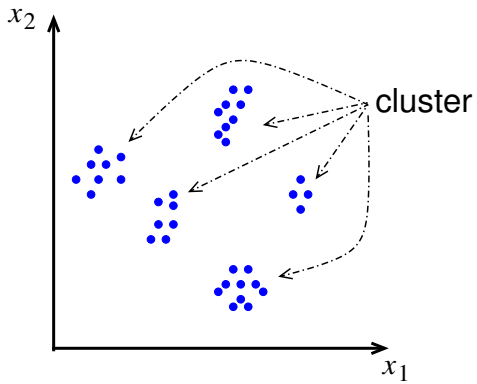
- given pre-classified examples, $Tr = \{(x^{(i)}, f(x^{(i)}))\}$, learn a general description which captures the information content of the examples (rules working for the whole input domain)
- it should be possible to use this description in a predictive way (given a new input $\tilde{x}$, predict the associated output $h(\tilde{x})$)
- it is assumed that an expert (or teacher) provides the supervision
  (i.e., the values of $h()$ corresponding to the training instances x)

Example of application: handwritten character recognition

Main Learning Paradigms

Unsupervised Learning:

- given a set of example $Tr = \{x^{(i)}\}$, discover regularities and/or patterns
  (true on the whole input domain)
- there is no expert (or teacher) to help us (i.e., no supervision!)



- Clustering
- Rule Discovery

Example of application: data mining on structured databases

Main Learning Paradigms

Reinforcement Learning:

- agent which may
  - be in state s, and
  - execute an action a (chosen among the ones admissible in the current state)
- and operates in an environment e, which in response to action a in the state s returns
  - the next state, and
  - a reward r, which can be positive (+), negative (-), or neutral (0).

The goal of the agent is to maximize a function of the rewards (e.g. expected discounted sum of rewards: $\sum_{t=0}^{\infty} \gamma^t r_{t+1}$ where $0 \leq \gamma < 1$)

Example of application: surfing the Web looking for focused information

Fundamental Ingredients

- Training Data (drawn from the Instance Space, X)
- Hypothesis Space, $\mathcal{H}$
  - it constitutes the set of functions which can be implemented by the machine learning system;
  - it is assumed that the function to be learned f may be represented by a hypothesis $h \in \mathcal{H}$... (the actual h is selected via the training data)
  - or that at least a hypothesis $h \in \mathcal{H}$ is "similar" to f (approximation);
- Search Algorithm into the Hypothesis Space, learning algorithm

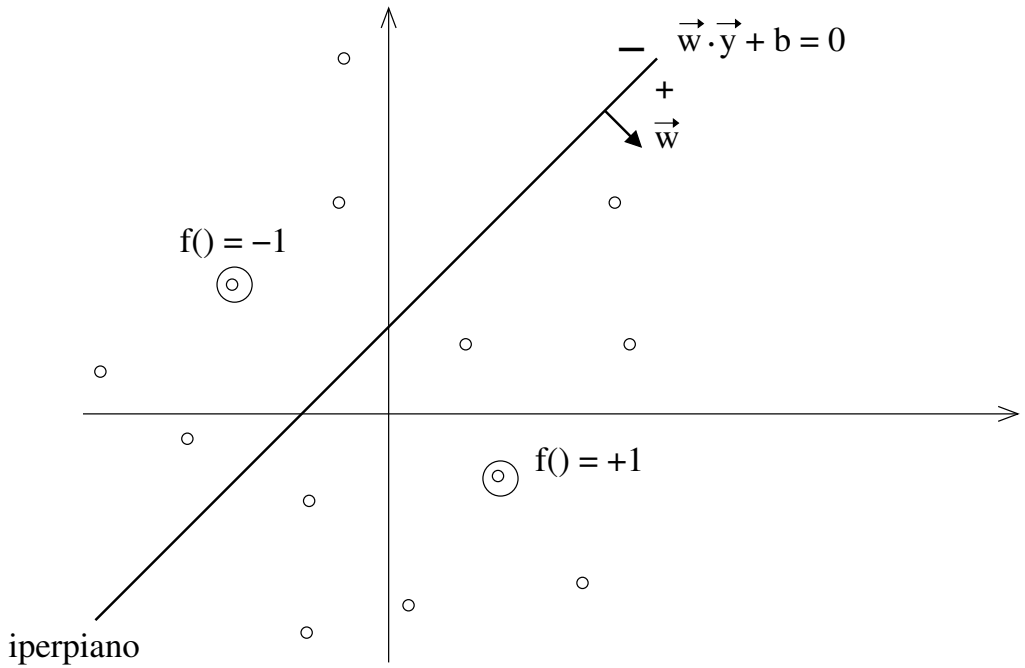
WARNING: $\mathcal{H}$ cannot coincide with the set of all possible functions and the search (into $\mathcal{H}$) to be exhaustive $\rightarrow$ Learning is useless!!!

Inductive Bias: on the representation ($\mathcal{H}$) and/or on the search (learning algorithm)

Hypothesis Space: Example 1

Hyperplanes in $\mathbb{R}^2$

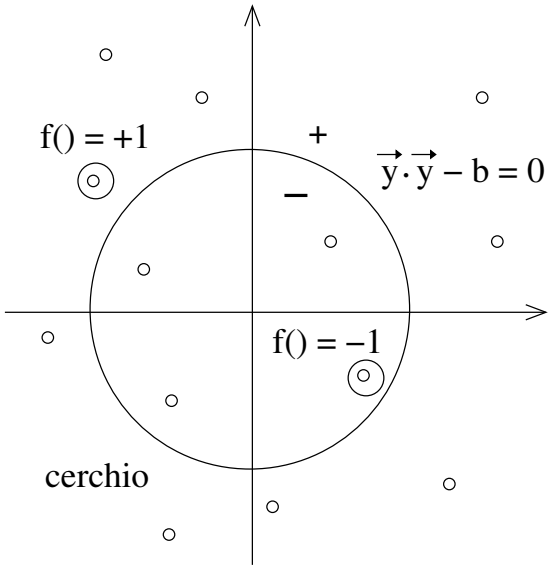
- Instance Space $\rightarrow$ points into the plane: $X = \{\vec{y} \in \mathbb{R}^2\}$
- Hypothesis Space $\rightarrow$ dichotomies induced by hyperplanes in $\mathbb{R}^2$: $\mathcal{H} = \{f_{(\vec{w},b)}(\vec{y}) | f_{(\vec{w},b)}(\vec{y}) = sign(\vec{w} \cdot \vec{y} + b), \vec{w} \in \mathbb{R}^2, b \in \mathbb{R}\}$



Hypothesis Space: Example 2

Disks in $\mathbb{R}^2$

- Instance Space $\rightarrow$ points in the plane: $X = \{\vec{y} \in \mathbb{R}^2\}$
- Hypothesis Space $\rightarrow$ dichotomies induced by disks in $\mathbb{R}^2$ centered into the origin: $\mathcal{H} = \{f_b(\vec{y}) | f_b(\vec{y}) = sign(\vec{y} \cdot \vec{y} - b), b \in \mathbb{R}\}$



Hypothesis Space: Example 3

Conjunctions of m positive literals

- Instance Space $\rightarrow$ strings of m bits: $X = \{s | s \in \{0,1\}^m\}$
- Hypothesis Space $\rightarrow$ all the logic sentences involving positive literals $l_1, \ldots, l_m$ (l_1 is true if the first bit is 1, l_2 is true if the second bit is 1, etc.) and just containing the operator $\wedge$ (**and**):

$$\mathcal{H} = \{f_{\{i_1,\ldots,i_j\}}(s) | f_{\{i_1,\ldots,i_j\}}(s) \equiv l_{i_1} \wedge l_{i_2} \wedge \cdots \wedge l_{i_j}, \{i_1, \ldots, i_j\} \subseteq \{1, \ldots, m\}\}$$

E.g. $m = 3$, $X = \{0,1\}^3$
Examples of instances $\rightarrow$ $s1 = 101$, $s2 = 001$, $s3 = 100$, $s4 = 111$
Examples of hypotheses $\rightarrow$ $h_1 \equiv l_2$, $h_2 \equiv l_1 \wedge l_2$, $h_3 \equiv true$, $h_4 \equiv l_1 \wedge l_3$, $h_5 \equiv l_1 \wedge l_2 \wedge l_3$
Notice that: h_1, h_2, and h_5 are false for $s1$, $s2$ and $s3$ and true for $s4$; h_3 is true for any instance; h_4 is true for $s1$ and $s4$ but false for $s2$ and $s3$

Hypothesis Space: Example 3

Conjunction of m positive literals

- Question 1: how many and which are the distinct hypotheses for $m = 3$?
- Question 2: how many distinct hypotheses there are as a function of m ?

Hypothesis Space: Example 3

Conjunction of m positive literals

- Question 1: how many and which are the distinct hypotheses for $m = 3$?
  - Ans.(which): *true*, l_1, l_2, l_3, $l_1 \wedge l_2$, $l_1 \wedge l_3$, $l_2 \wedge l_3$, $l_1 \wedge l_2 \wedge l_3$
  - Ans.(how many): 8
- Question 2: how many distinct hypotheses there are as a function of m ?

Hypothesis Space: Example 3

Conjunction of m positive literals

- Question 1: how many and which are the distinct hypotheses for $m = 3$?
  - Ans.(which): *true*, l_1, l_2, l_3, $l_1 \wedge l_2$, $l_1 \wedge l_3$, $l_2 \wedge l_3$, $l_1 \wedge l_2 \wedge l_3$
  - Ans.(how many): 8
- Question 2: how many distinct hypotheses there are as a function of m ?
  - Ans.: 2^m, in fact for each possible bit of the input string the corresponding literal may occur or not in the logic formula, so:

$$\underbrace{2 \cdot 2 \cdot 2 \cdots 2}_{m \text{ times}} = 2^m$$

Hypothesis Space: Example 4

Conjunction of m literals

- Instance Space $\rightarrow$ strings of m bits: $X = \{s | s \in \{0,1\}^m\}$
- Hypothesis Space $\rightarrow$ all the logic sentences involving literals $l_1, \ldots, l_m$ (any boolean variable l_i or its negation $\neg l_i$) and just containing the operator $\wedge$ (**and**):
  $\mathcal{H} = \{f_{\{i_1,\ldots,i_j\}}(s) | f_{\{i_1,\ldots,i_j\}}(s) \equiv L_{i_1} \wedge L_{i_2} \wedge \cdots \wedge L_{i_j}$, where $L_{i_k} = l_{i_k}$ or $\neg l_{i_k}, \{i_1, \ldots, i_j\} \subseteq \{1, \ldots, 2m\}\}$

Notice that if in a formula a literal occurs together with its negation, then the formula is always *false* (unsatisfiable formula)
So, all the formulas containing a literal and its negation, are equivalent to *false*

Hypothesis Space: Example 4

Conjunctions of m literals

E.g. $m = 3$, $X = \{0,1\}^3$
Examples of instances $\rightarrow$ $s1 = 101$, $s2 = 001$, $s3 = 100$, $s4 = 111$, $s5 = 000$
Examples of hypotheses $\rightarrow$ $h_1 \equiv \neg l_2$, $h_2 \equiv \neg l_1 \wedge l_3$, $h_3 \equiv true$, $h_4 \equiv \neg l_1 \wedge \neg l_2 \wedge \neg l_3$
Notice that:

- h_1, is false for $s4$, and true for $s1$, $s2$, $s3$ and $s5$;
- h_2 is false for $s1$, $s3$, $s4$ and $s5$ and true for $s2$;
- h_3 is true for every instance;
- h_4 is false for $s1$, $s2$, $s3$, $s4$ and true for $s5$;

Question: how many distinct hypotheses there are as a function of m ?

Hypothesis Space: Example 4

Conjunction of m literals
Question: how many distinct hypotheses there are as a function of m ?

Answer: considering that all the unsatisfiable formulas are equivalent to *false*, we do not consider formulas where a literal occurs together with its negation.
So, for each possible bit of the input string the corresponding literal may not be present in the logic formula or, if it appears, it is either asserted or negated:

$$\underbrace{3 \cdot 3 \cdot 3 \cdots 3}_{m \text{ times}} = 3^m$$

And considering the always false formula, we get $3^m + 1$

Hypothesis Space: Example 5

Lookup Table

- Instance Space $\rightarrow$ strings of m bits: $X = \{s | s \in \{0,1\}^m\}$
- Hypothesis Space $\rightarrow$ all the possible truth tables which map input instances to *true* and *false*: $\mathcal{H} = \{f(s) | f : X \rightarrow \{true, false\}\}$

  Ex.

l_1	l_2	$\cdots$	l_m	$f(s)$
0	0	$\cdots$	0	1
0	0	$\cdots$	1	0
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
0	1	$\cdots$	0	0
0	1	$\cdots$	1	1
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
1	0	$\cdots$	0	1
1	0	$\cdots$	1	1
1	1	$\cdots$	0	0
1	1	$\cdots$	1	1
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$

Hypothesis Space: Example 5

Conjunctions of m literals

Question: how many distinct hypotheses there are as a function of m ?

Hypothesis Space: Example 5

Conjunctions of m literals
Question: how many distinct hypotheses there are as a function of m ?

Answer: by a lookup table it is possible to implement any boolean function on the Instance Space
Since the number of possible instances is:

$$\underbrace{2 \cdot 2 \cdot 2 \cdots 2}_{m \text{ times}} = 2^m$$

the number of distinct functions is: 2^{2^m}

Observations on Examples 3, 4 and 5

Notice that for examples 3, 4 and 5 the Instance Space is the same. The Hypothesis Spaces, however (let denote with $\mathcal{H}_3$ the one referring to example 3, and so on) are different and for each given m the following relation holds: $\boxed{\mathcal{H}_3 \subset \mathcal{H}_4 \subset \mathcal{H}_5}$

E.g., given $m = 3$

- the boolean function $f(s)$, which is true only for instances 001 and 011, is contained in $\mathcal{H}_4$, in fact $f(s) \equiv \neg l_1 \wedge l_3 \in \mathcal{H}_4$, and in $\mathcal{H}_5$(it is easy to define a lookup table where the output column is 1 only for the rows corresponding to instances 001 and 011), but not in $\mathcal{H}_3$ because it is not possible to define $f(s)$ by using a conjunction of positive literals.
- the boolean function $f(s)$, which is true only for instances 001, 011 and 100 is contained in $\mathcal{H}_5$ (we can proceed as above), but not in $\mathcal{H}_3$ and $\mathcal{H}_4$, because it is not possible to define $f(s)$ using a conjunction of (positive) literals.

Specifically $\mathcal{H}_5$ coincides with the set of all possible boolean functions on X.

Hypothesis Space Complexity: VC-dimension

Def.: Shattering
Given $S \subset X$, S is shattered by the Hypothesis Space $\mathcal{H}$ if and only if

$$\forall S' \subseteq S,\ \exists h \in \mathcal{H},\ \text{such that } \forall x \in S,\ h(x) = 1 \Leftrightarrow x \in S'$$

($\mathcal{H}$ is able to implement all possible dichotomies of S)
Def.: VC-dimension
The VC-dimension of a Hypothesis Space $\mathcal{H}$ defined over an Instance Space X is the size of the largest finite subset of X shattered by $\mathcal{H}$:

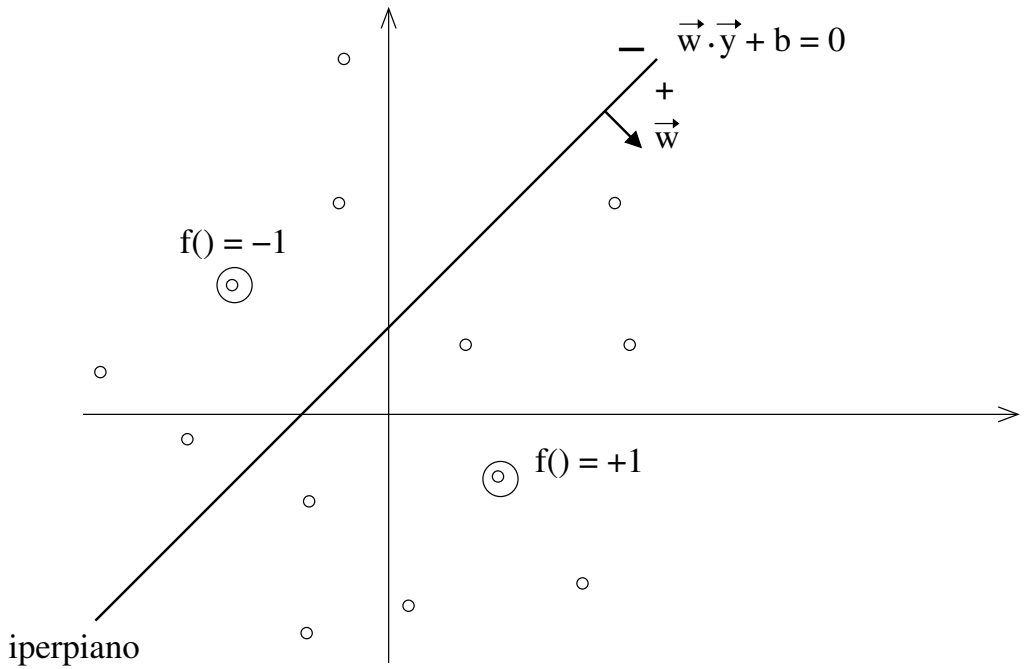
$$VC(\mathcal{H}) = \max_{S \subseteq X} |S| :\ S \text{ is shattered by } \mathcal{H}$$

If arbitrarily large finite sets of X can be shattered by $\mathcal{H}$, then $VC(\mathcal{H}) = \infty$.

VC-dimension: Example

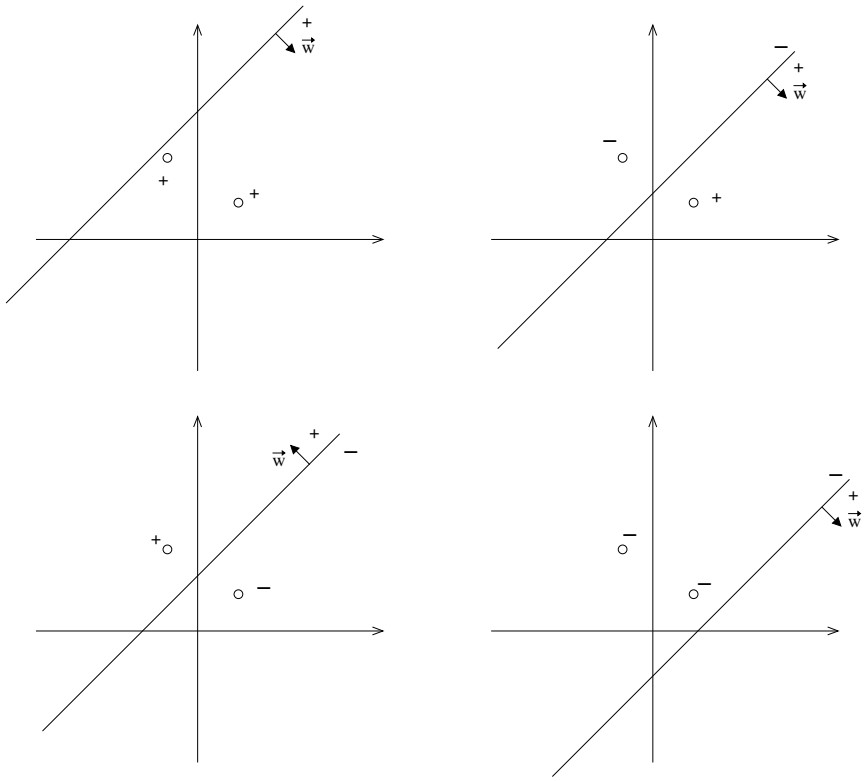
What is the VC-dimension of $\mathcal{H}_1$?

$\mathcal{H}_1 = \{f_{(\vec{w},b)}(\vec{y}) | f_{(\vec{w},b)}(\vec{y}) = sign(\vec{w} \cdot \vec{y} + b), \vec{w} \in \mathbb{R}^2, b \in \mathbb{R}\}$



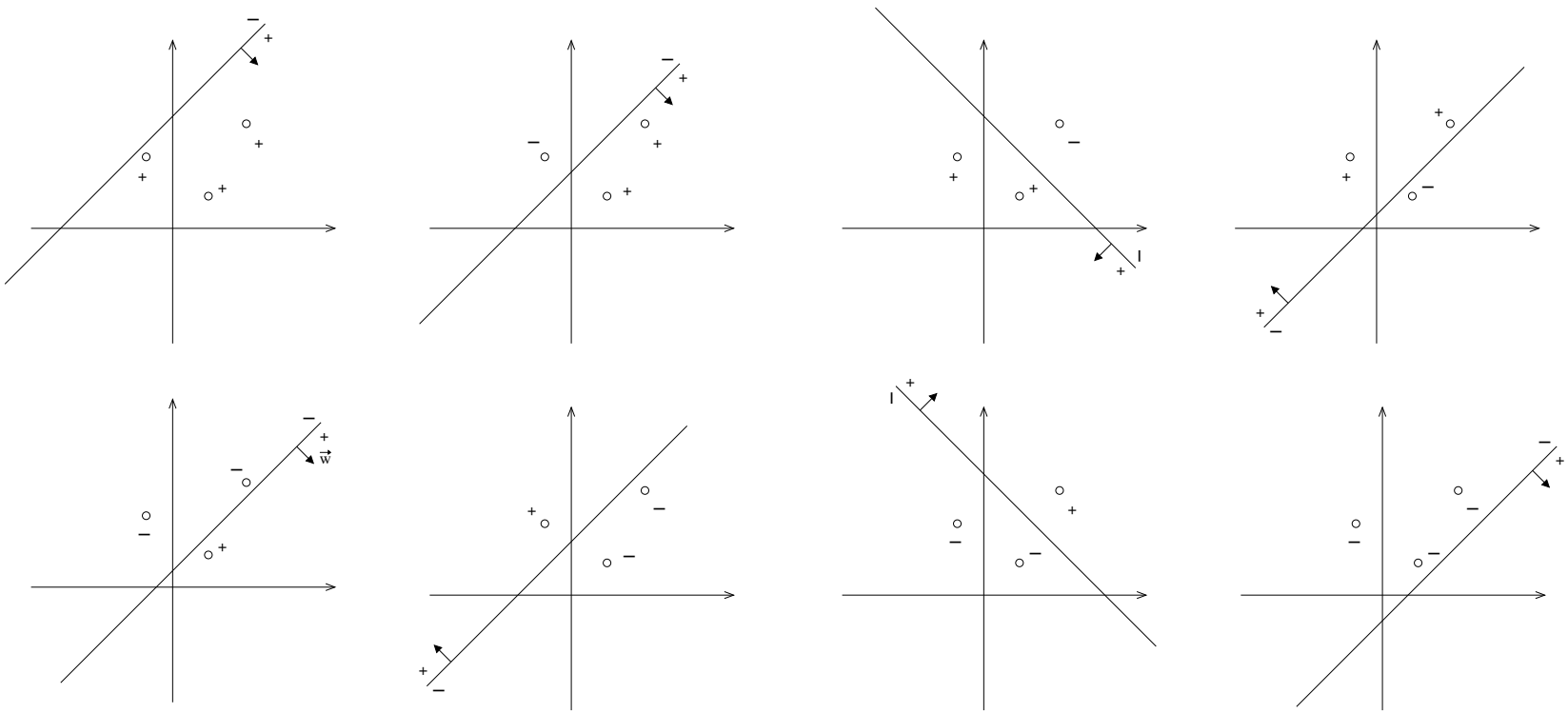
VC-dimension: Example

What is the VC-dimension of $\mathcal{H}_1$?
$VC(\mathcal{H}) \geq 1$ trivial. Let consider 2 points:



VC-dimension: Example

What is the VC-dimension of $\mathcal{H}_1$?
Thus $VC(\mathcal{H}) \geq 2$. Let consider 3 points:



VC-dimension: Example

What is the VC-dimension of $\mathcal{H}_1$?
Thus $VC(\mathcal{H}) \geq 3$. What does happen with 4 points ?

VC-dimension: Example

What is the VC-dimension of $\mathcal{H}_1$?
Thus $VC(\mathcal{H}) \geq 3$. What does happen with 4 points ? It is impossible to shatter 4 points!!
In fact there always exist two couples of points such that if we connect the two members by a segment, the two resulting segments will intersect. So, if we label the points of each couple with a different class, a curve is necessary to separate them! Thus $VC(\mathcal{H}) = 3$