

A Short Introduction to Statistical Learning

Frederic Koriche

LIRMM, Université Montpellier II

Parcours Intelligence Artificielle
Master Informatique 2ème Année
Septembre 2011

Outline

1 Prediction Tasks

2 The Learning Protocol

3 PAC Learning

4 Concept Classes

5 Version Spaces

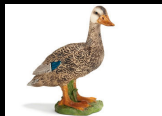
Prediction Tasks



Binary Classification

- Observations: horses and cows
- Decisions: 1 if horse and 0 if cow
- Attributes: $\text{Crest}(x)$, $\text{CrestColor}(x)$, $\text{Fur}(x)$, $\text{FurColor}(x)$, $\text{Hoof}(x)$, $\text{Horns}(x)$, . . .

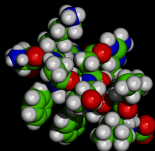
Prediction Tasks



Domain Classification

- Observations: farm animals
- Decisions: a category of animal
- Attributes: $Crest(x)$, $Fur(x)$, $Hoof(x)$, $Horns(x)$, $Scales(x)$, $Wings(x)$, . . .

Prediction Tasks



Regression

- Observations: molecules
- Decisions: a degree of toxicity
- Attributes: presence of groups, bio-physical properties, chemical properties, etc.

Prediction Tasks



Ranking

- Observations: a set of movies
- Decisions: a ranking of movies according to user's preferences
- Attributes: $\text{Director}(x)$, $\text{Genre}(x)$, $\text{Year}(x)$, ...

Prediction Tasks



Structured Prediction

- Observations: a scene observed by the NPC
- Decisions: a sequence of actions
- Attributes: NPC's attributes, actors' and objects' attributes, constraints, etc.

Decision Tasks

Observation Space: \mathcal{X}

- Boolean: $\{0, 1\}^n$
- Euclidean: \mathbb{R}^n

Decision Space: \mathcal{Y}

- Classification: $\{0, 1\}$ or $[m] = \{1, \dots, m\}$
- Regression: $[0, 1]$ or \mathbb{R}
- Ranking: \mathbb{S}_m
- Structured: $\{0, 1\}^m$ or \mathbb{R}^m

Hypothesis space: \mathcal{H}

Set of functions $h : X \rightarrow Y$ from observations to decisions

Loss Function ℓ

Map $Y \times Y \rightarrow \mathbb{R}_+$ from pairs of observations to penalties

Decision Tasks

Examples of simple loss functions

Discrete	$\ell_{dis}(\hat{y}, y) = I_{\hat{y}=y}$
Absolute	$\ell_{abs}(\hat{y}, y) = y - \hat{y} $
Square	$\ell_{sq}(\hat{y}, y) = (y - \hat{y})^2$

Outline

1 Prediction Tasks

2 The Learning Protocol

3 PAC Learning

4 Concept Classes

5 Version Spaces

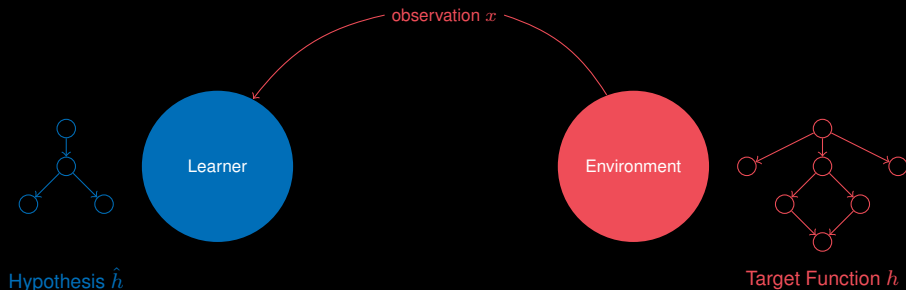
The Learning Protocol



Online Supervised Learning

- The environment supplies an observation x
- The learner makes a prediction \hat{y}
- The environment supplies the correct response y

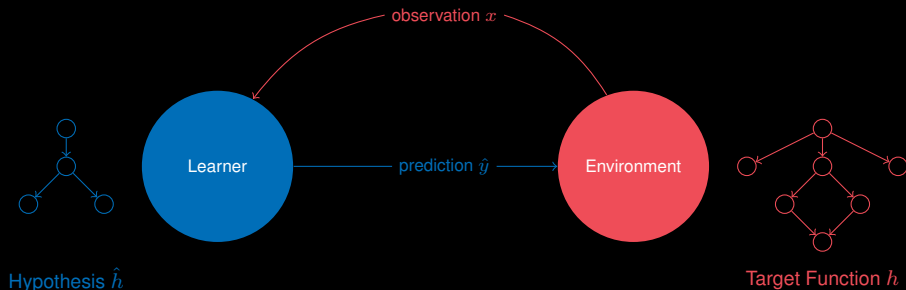
The Learning Protocol



Online Supervised Learning

- The environment supplies an observation x
- The learner makes a prediction \hat{y}
- The environment supplies the correct response y

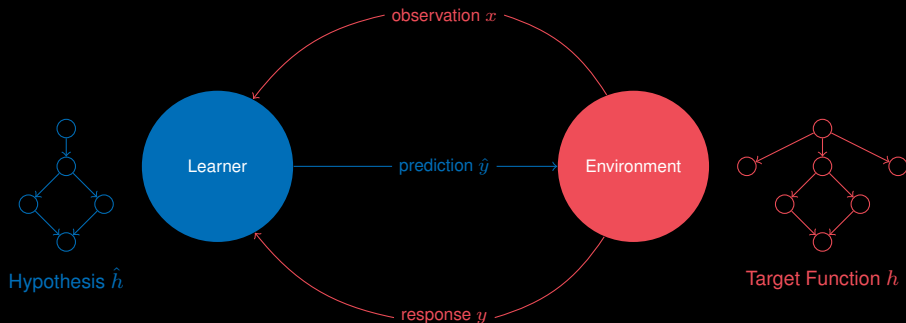
The Learning Protocol



Online Supervised Learning

- The environment supplies an observation x
- The learner makes a prediction \hat{y}
- The environment supplies the correct response y

The Learning Protocol



Online Supervised Learning

- The environment supplies an observation x
- The learner makes a prediction \hat{y}
- The environment supplies the correct response y

Outline

1 Prediction Tasks

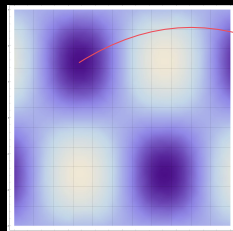
2 The Learning Protocol

3 PAC Learning

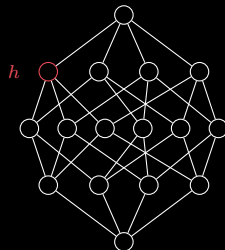
4 Concept Classes

5 Version Spaces

PAC Learning: The Realizable Case



Distribution \mathcal{D}



Hypothesis Class \mathcal{H}

Assumptions

- Each observation x is drawn at random according to a fixed but unknown probability distribution \mathcal{D} .
- Each decision y is determined by a target function h .
- The target function h is an hidden hypothesis in some known hypothesis class \mathcal{H} .

PAC Learning: The Realizable Case

Parameters

- Prediction task: $(\mathcal{X}, \mathcal{Y}, \ell)$
- Hypothesis class: \mathcal{H}

Risk

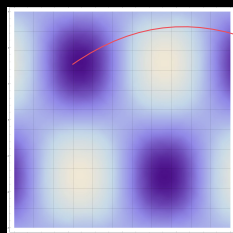
For a target function h and a distribution \mathcal{D} over $(\mathcal{X}, \mathcal{Y})$, the *risk* of any hypothesis \hat{h} is

$$risk(\hat{h}) = \mathbb{E}_{x \sim \mathcal{D}}[\ell(\hat{h}(x), h(x))]$$

PAC Learning

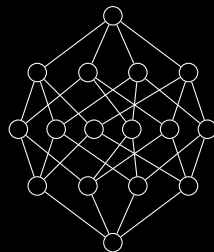
An algorithm A is a PAC-learning algorithm for a hypothesis class \mathcal{H} with respect to a class of prediction tasks $(\mathcal{X}, \mathcal{Y}, \ell)$ if for every parameters $\delta, \epsilon \in (0, 1)$, every target function $h \in \mathcal{H}$ and every target distribution \mathcal{D} on $(\mathcal{X}, \mathcal{Y})$, after seeing a polynomial number of examples drawn at random according to \mathcal{D} , the algorithm will output a hypothesis \hat{h} such that $risk(\hat{h}) \leq \epsilon$ with probability $1 - \delta$.

PAC Learning: The Agnostic Case



Distribution \mathcal{D}

(x, y)



Hypothesis Class \mathcal{H}

Assumptions

- Each example (x, y) is drawn at random according to a fixed but unknown probability distribution \mathcal{D} .
- There is *no* assumption concerning the target function

PAC Learning: The Agnostic Case

Parameters

- Prediction task: $(\mathcal{X}, \mathcal{Y}, \ell)$
- Hypothesis class: \mathcal{H}

Risk

For a distribution \mathcal{D} over $(\mathcal{X}, \mathcal{Y})$, the *risk* of any hypothesis \hat{h} is

$$\text{risk}(\hat{h}) = \mathbb{E}_{(x,y) \sim \mathcal{D}}[\ell(\hat{h}(x), y)]$$

Agnostic PAC Learning

An algorithm A is an agnostic PAC-learning algorithm for a hypothesis class \mathcal{H} with respect to a class of prediction tasks $(\mathcal{X}, \mathcal{Y}, \ell)$ if for every parameters $\delta, \epsilon \in (0, 1)$ and any target distribution \mathcal{D} on $(\mathcal{X}, \mathcal{Y})$, after seeing a polynomial number of examples drawn at random according to \mathcal{D} , the algorithm will output a hypothesis \hat{h} such that $\text{risk}(\hat{h}) \leq \epsilon$ with probability $1 - \delta$.

Outline

1 Prediction Tasks

2 The Learning Protocol

3 PAC Learning

4 Concept Classes

5 Version Spaces

Concept Classes

Simple concepts

- Atom: boolean variable x_i (or pair attribute-value $a = v$)
- Literal: atom or its negation
- Monotone term: conjunction of atoms
- Monotone clause: disjunction of atoms
- Term: conjunction of literals
- Clause: disjunction of literals

Logical concepts

- Monotone DNF formula: disjunction of monotone terms
- Monotone CNF formula: conjunction of monotone clauses
- k -DNF formula: disjunction of terms containing at most k literals
- k -CNF formula: conjunction of clauses containing at most k literals
- k -term DNF formula: disjunction containing at most k terms
- k -term CNF formula: conjunction containing at most k clauses

Concept Classes

Cover Relation

An example x is *covered* by a logical concept f iff and only if x is a model of f :

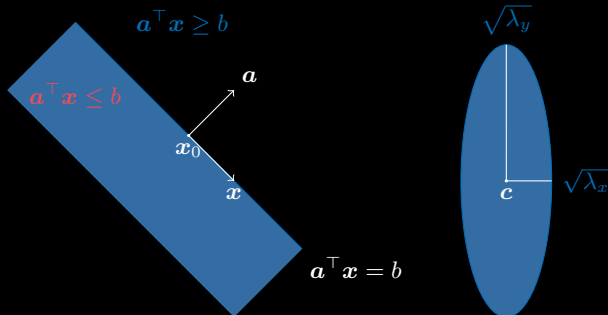
$$f(x) = 1 \text{ iff } x \models f$$

Example

The concept $(tails = 2) \wedge (nuclei = 2)$ covers every positive example and no negative example.

tails	nuclei	color	wall	class
2	2	dark	thin	+
2	2	light	thin	+
2	1	light	thin	-
1	2	dark	thick	-

Concept Classes



Linear formulas

The concept f consists in a vector a of weights in \mathbb{R}^n and a threshold value b in \mathbb{R} .

$$f(x) = \begin{cases} 1 & \text{if } a^\top x = \sum_{i=1}^n a_i x_i \geq b \\ 0 & \text{otherwise} \end{cases}$$

Ellipsoid formulas

The concept f consists in a center $c \in \mathbb{R}^n$ and a positive semi-definite matrix $P \in \mathbb{R}^{n \times n}$.

$$f(x) = \begin{cases} 1 & \text{if } (x - c)^\top P^{-1} (x - c) \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

Outline

1 Prediction Tasks

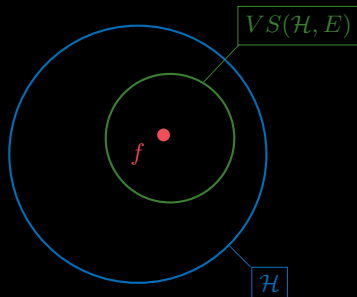
2 The Learning Protocol

3 PAC Learning

4 Concept Classes

5 Version Spaces

Version Spaces



Consistency

An hypothesis \hat{h} is consistent with an example $(x, h(x))$ if $\hat{h}(x) = h(x)$.

Version Space

The version space of a training set $E = \{(x_i, h(x_i))\}$ with respect to an hypothesis space \mathcal{H} is the set of all hypotheses in \mathcal{H} that are consistent with every example in E .

$$VS(\mathcal{H}, E) = \{\hat{h} \in \mathcal{H} : \forall (x_i, h(x_i)) \in E : \hat{h}(x_i) = h(x_i)\}$$

Version Spaces

ϵ -exhausting

The version space $VS(\mathcal{H}, E)$ is said to be ϵ -exhausted if every hypothesis in the version space has (discrete) risk less than ϵ .

$$\forall h \in VS(\mathcal{H}, E) \text{ risk}(h) < \epsilon$$

Hausser Theorem

If the hypothesis space \mathcal{H} is finite, then for any $0 \leq \epsilon \leq 1$, the probability that the version space $VS(\mathcal{H}, E)$ is not ϵ -exhausted is less than or equal to

$$|\mathcal{H}|e^{-\epsilon|E|}$$

Sample Complexity

The number of training examples needed to assure that any consistent hypothesis will be PAC must be greater than or equal to

$$\frac{1}{\epsilon} \left(\ln |\mathcal{H}| + \ln \left(\frac{1}{\delta} \right) \right)$$

Version Spaces

PAC-Learnable

- Simple concepts
- k -DNF and k -CNF formulas
- linear threshold formulas and ellipsoid formulas

Not PAC-Learnable

- polysize DNF formulas
- polysize CNF formulas
- boolean circuits