Machine Learning

Computer Engineering

Fabio Vandin

October 2nd, 2023

Machine Learning

Chiave x Moodle : MLearn23

Disponibili materiali degli anni passati

~ 6 corsi in laboratorio
max 3 homeworks 0/3 pts (not mandatory)
→ 2 weeks → in tutto

Exam : 30 mins multiple choice question (fail / pass) } same day → no split
1.5/2 hrs questions and exercises → Possible oral exam

Grade : 2° part grade + Homeworks

23/01/24
13/02/24
02/06/24
10/09/24

"Understanding machine learning" → free to download

python (scikit-learn , numpy, ...)

jupiter lab (through Anaconda)

Homework 0 fabio.vandin@ unipd.it

Formal Model

A learner (us / machine) has access to:

- Domain set $x \in X$: set of all possible objects to make predictions about
  → instance space
  instance (vector of features)
- Label set Y
- Training data $S = ((x_1, y_1), \dots, (x_m, y_m))$ as the input
  → finite
  → Called training set
- Output : $h : X \to Y$
  → prediction rule / predictor / hypotesis / classifier
- Data generation model : instances generated and labeled accordingly to a function
  - probability distribution that generates the instances (not known)
  - labeling function $f : X \to Y$ (not known)
  - label y_i for each instance x_i
  - each point in the training set
- Measure of success : error of a classifier = probability it doesn't predict well

A Formal Model (Statistical Learning)

We have a *learner* (us, or the machine) has access to:

1. **Domain set** $\mathcal{X}$: set of all possible objects to make predictions about
   - domain point $x \in \mathcal{X}$ = *instance*, usually represented by a vector of *features*
   - $\mathcal{X}$ is the *instance space*
2. **Label set** $\mathcal{Y}$: set of possible labels.
   - often two labels, e.g $\{-1, +1\}$ or $\{0, 1\}$
3. **Training data** $S = ((x_1, y_1), \dots, (x_m, y_m))$: finite sequence of labeled domain points, i.e. pairs in $\mathcal{X} \times \mathcal{Y}$
   - this is the learner's **input**
   - S: *training example* or *training set*

A Formal Model

4. **Learner's output** h: prediction rule $h : \mathcal{X} \to \mathcal{Y}$
   - also called *predictor, hypothesis,* or *classifier*
   - $A(S)$: prediction rule produced by learning algorithm A when training set S is given to it
   - sometimes $\hat{f}$ used instead of h
5. **Data-generation model**: instances are generated by some probability distribution and labeled according to a function
   - $\mathcal{D}$: probability distribution over $\mathcal{X}$ (**NOT KNOWN TO THE LEARNER!**)
   - labeling function f: $\mathcal{X} \to \mathcal{Y}$ (**NOT KNOWN TO THE LEARNER!**)
   - label y_i of instance x_i: $y_i = f(x_i)$, for all $i = 1, \ldots, m$
   - each point in training set S: first sample x_i according to $\mathcal{D}$, then label it as $y_i = f(x_i)$
6. **Measures of success**: *error of a classifier* = probability it does not predict the correct label on a random data point generate by distribution $\mathcal{D}$

Loss

Given domain subset $A \subset \mathcal{X}$, $\mathcal{D}(A)$ = probability of observing a point $x \in A$.

In many cases, we refer to A as *event* and express it using a function $\pi : \mathcal{X} \to \{0,1\}$, that is:

$$A = \{x \in \mathcal{X} : \pi(x) = 1\}$$

In this case we have $\mathbb{P}_{x \sim \mathcal{D}}[\pi(x)] = \mathcal{D}(A)$

Error of prediction rule $h : \mathcal{X} \to \mathcal{Y}$ is

true label

Loss

$$L_{\mathcal{D},f}(h) \stackrel{def}{=} \mathbb{P}_{x \sim \mathcal{D}}[h(x) \neq f(x)] \stackrel{def}{=} \mathcal{D}(\{x : h(x) \neq f(x)\})$$

Distribution

model

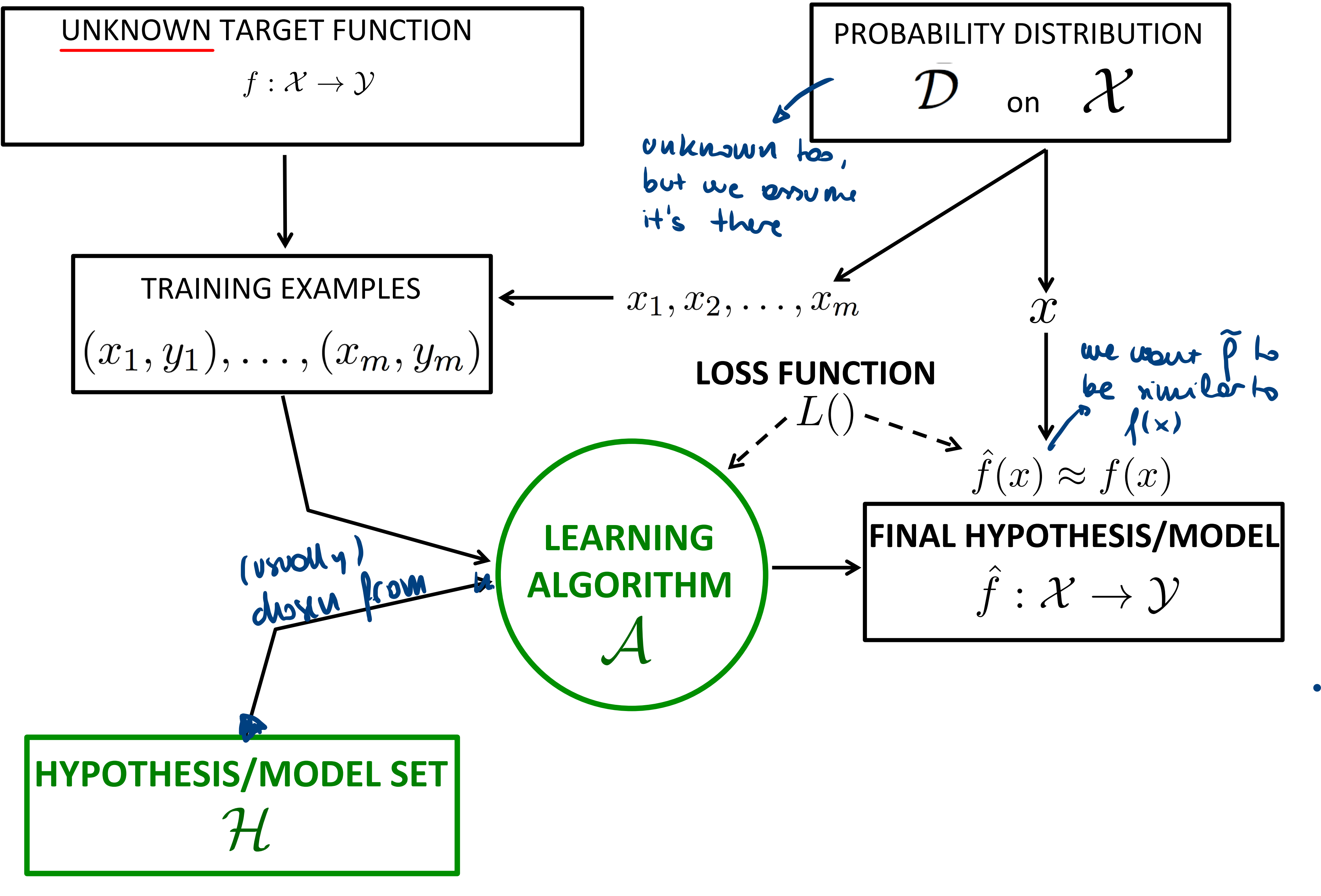
label predicted by h

true labeling

Notes:

- $L_{\mathcal{D},f}(h)$ has many different names: **generalization error**, *true error*, *risk*, **loss**, ...
- often f is obvious, so omitted: $L_{\mathcal{D}}(h)$

Learning Process (Simplified)



Types of Learning

y_i are known: **training set** $(x_1, y_1), \ldots, (x_m, y_m)$

supervised learning

Training set contains only $x_1, x_2, \ldots, x_m$

unsupervised learning

(for supervised learning)

There can be different types of output:

- $\mathcal{Y}$ is **discrete**
- $\mathcal{Y}$ is **continuous**

Notes: we will see a more general learning model soon, main ideas are the same!

Types of Learning

$\mathcal{Y}$ is …	y_i known ***Supervised Learning***	y_i not available ***Unsupervised Learning***
Discrete	classification	clustering dimensionality reduction …
Continuous	regression	