Aalto University

Statistical machine learning | Intro

Introduction to machine learning

Francesco Corona

Chemical and Metallurgical Engineering
School of Chemical Engineering

The statistical learning framework

Intro

January 15, 2025
– FC –

Inputs to the learner

In the basic statistical learning framework, the learner can access to the following info

Domain set
This is the set of all *objects* that we might wish to label
Domain points are encoded as vectors of N_x features

$$\mathcal{X} = \{x : x \in \mathbb{R}^{N_x}\}$$
$$\mathcal{X} \subseteq \mathbb{R}^{N_x}$$

Label set
This is the set of N_y *labels* a domain point may take on
(We start with a two-label label set, $N_y = 2$)

$$\mathcal{Y} \subset \mathbb{N}_0$$
$$\mathcal{Y} = \{0, 1\}$$

In practice, the learner has access to a combination of the domain and the label set

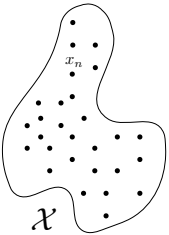
Training data
Some subset of pairs in $\mathcal{X} \times \mathcal{Y}$ of labeled domain points
The set of $N = |\mathcal{S}|$ *training examples*, the *training set*

$$\mathcal{S} = \{(x_n, y_n)\}_{n=1}^{N}$$
$$\text{with} \begin{cases} x_n \in \mathcal{X} \\ y_n \in \mathcal{Y} \end{cases}$$

January 15, 2025 – FC

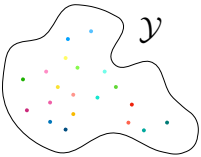
Inputs to the learner (cont.)

$\mathcal{X} \subset \mathbb{R}^{N_x}$



Domain set

$\mathcal{Y} \subset \mathbb{N}_0$

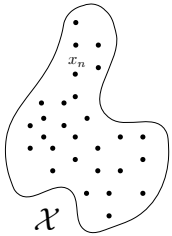


Label set

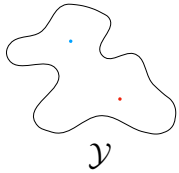
y 15, 2025
– FC –

Inputs to the learner (cont.)

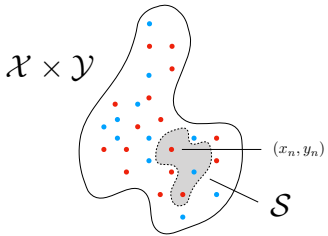
$\mathcal{X} \subset \mathbb{R}^{N_x}$



$\mathcal{Y} = \{0, 1\}$



$\mathcal{S} \subset \mathcal{X} \times \mathcal{Y}$

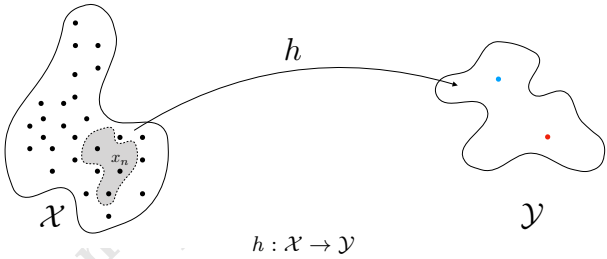


Training data

January 15, 2025

Outputs from the learner

The learner is asked to output a **prediction rule**, some function h from set $\mathcal{X}$ to set $\mathcal{Y}$

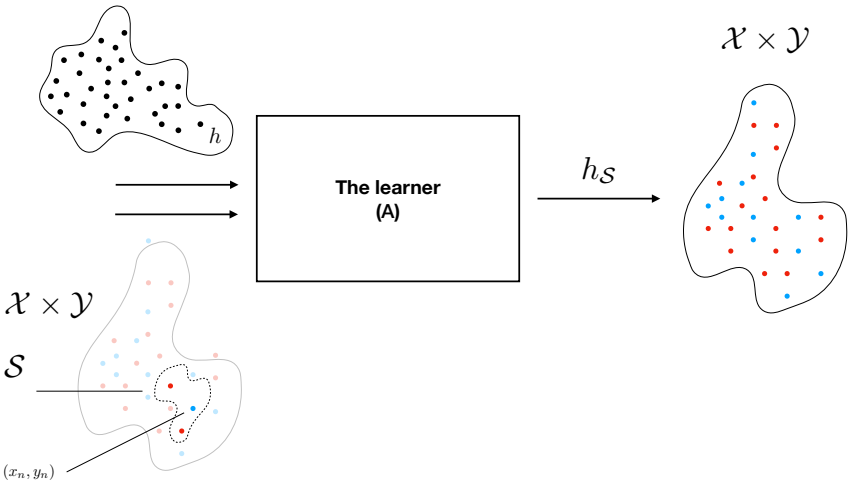


Oftentimes, the prediction rule is known as the *predictor*, or *hypothesis*, or *classifier*

- This is the function used to predict the label y of any (new) domain point x

The learner

$\{h : h : \mathcal{X} \to \mathcal{Y}\}$

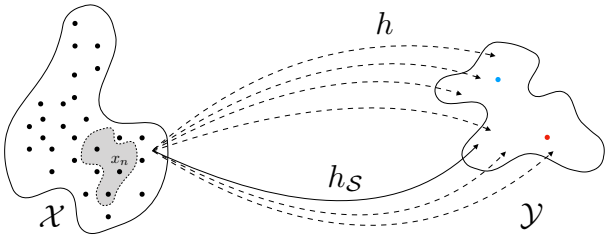


$h_{\mathcal{S}} = A(\mathcal{S})$ is the hypothesis that a **learning algorithm** A returns, given a training set $\mathcal{S}$

The success of the learner

It remains to define somehow the *quality of a prediction rule* h (how well it performs)

- This will define the strategy used by the learner to select the predictor $h_{\mathcal{S}}$



The quality of a rule should be determined with respect to the data-generating process

- (That is, it does not matter too much if a rule h fails on unlikely instances)

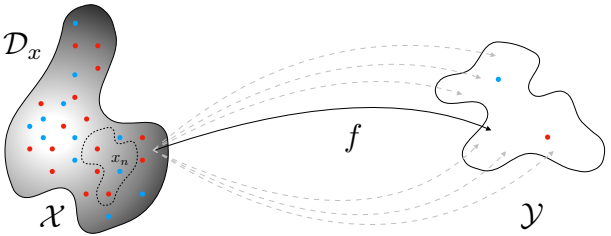
The success of the learner (cont.)

Assumptions

Training instances $\{x_n\}$ are assumed to be from a **probability distribution** $\mathcal{D}_x$ over $\mathcal{X}$

- For our learning tasks, we allow $\mathcal{D}_x$ to be an arbitrary distribution

Importantly, note that the learner has no information regarding the distribution $\mathcal{D}_x$



As for the labels, we start by assuming that there exists an exact **labelling function** f

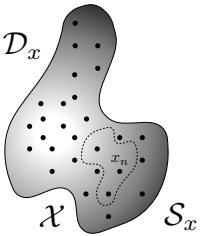
$$f : \mathcal{X} \to \mathcal{Y}$$

That is, we assume that the label $y \in \mathcal{Y}$ of all $x \in \mathcal{X}$ is fully determined as $y = f(x)$

- Note that also the labelling function f is unknown to the learner
- (This function is precisely what the learner tries to figure out)
- (As we proceed, such a strong assumption will be relaxed)

The success of the learner (cont.)

Formally, we are given a domain subset $\mathcal{S}_x \subset \mathcal{X}$ and a probability distribution $\mathcal{D}_x$ that assigns a number $\mathcal{D}_x(x)$ which determines how likely it is to observe any point $x \in \mathcal{X}$



- The set $\mathcal{S}_x$ is an event that can be realised using the function $p : \mathcal{X} \to \{0, 1\}$
- $\mathcal{S}_x = \{x_n; x_n \in \mathcal{X}, p(x_n) = 1\}_{n=1}^N$ occurs with probability $\mathbb{P}_{x \sim \mathcal{D}_x}[\{x_n\}_{n=1}^N]$

The success of the learner (cont.)

The **error** of a classifier h is defined as the probability that the label y of an instance x, randomly drawn from $\mathcal{X}$ according to $\mathcal{D}_x$, is predicted wrongly, or $h(x) \neq y = f(x)$

We can use this notion to define the error, or **loss**, L incurred by the predictor $h : \mathcal{X} \to \mathcal{Y}$

$$L_{\mathcal{D}_x,f}(h) \equiv \underbrace{\mathbb{P}_{x\sim\mathcal{D}_x}[x : x \in \mathcal{X}, h(x) \neq f(x)]}_{\mathcal{D}_x(\{x:x\in\mathcal{X},h(x)\neq f(x)\})}$$

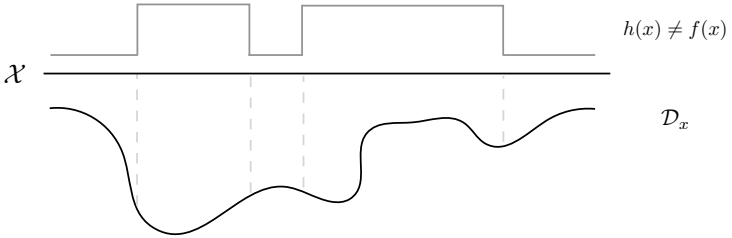
Thus, the error occurred by h is the probability of sampling a x for which $h(x) \neq f(x)$

- $(\mathcal{D}_x, f)$ indicates that error L of h is evaluated with respect to $\mathcal{D}_x$ and f

$L_{\mathcal{D}_x,f}(h)$ is often denoted as *generalisation error*, or *risk*, or *true error* of predictor h

---

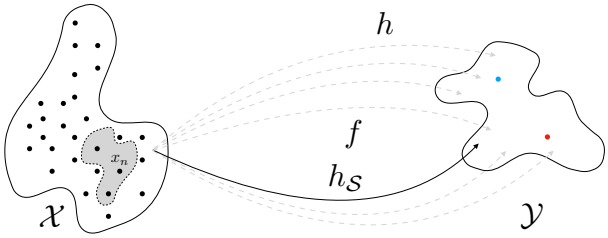
Graphically, $L_{\mathcal{D}_x,f}(h)$ is the volume under the portion of $\mathcal{D}_x$ associated to errors of h



The success of the learner (cont.)

In summary, pairs in $\mathcal{S}$ are generated by sampling x from $\mathcal{D}_x$ and labelling them by f

- Given $\mathcal{S}$, the goal of the learner is to return a predictor of smallest loss $L_{\mathcal{D}_x,f}$



In principle, a learning algorithm A is requested to return that predictor $h_{\mathcal{S}}$ that, given $\mathcal{S}$, minimises the loss $L_{\mathcal{D}_x,f}$ (with respect to distribution $\mathcal{D}_x$ and labelling function f)

- However, $L_{\mathcal{D}_x,f}$ cannot be directly calculated
- ($\mathcal{D}_x$ and f are unknown to the learner)

The success of the learner (cont.)

January 15, 2025

What is a reasonable strategy for the learner to practically overcome such a limitation?

- ... knowing that the learner only has access to the sample $\mathcal{S}$

We could think of looking for a predictor h that works well with sample $\mathcal{S}$ and that would work well also with other points generated according to $\mathcal{D}_x$ and labelled with f

- A predictor $h_{\mathcal{S}}$ that works well also on other similar sets (say, a test set)
- More precisely, a prediction rule $h_{\mathcal{S}}$ such that $L_{\mathcal{D}_x,f}(h_{\mathcal{S}})$ is smallest

Empirical risk minimisation

Intro

January 15, 2025
– FC –

Empirical risk

Empirical risk minimisation

January 13, 2025
– FC –

Empirical risk

The algorithm A receives as input a training set $\mathcal{S}$ (whose x-elements are from some distribution $\mathcal{D}_x$ and labeled by some target function f) and outputs a rule $h : \mathcal{X} \to \mathcal{Y}$

- To be calculable, we need a notion of error of h that depends on sample $\mathcal{S}$

Pragmatically, it is then reasonable to search for that predictor h that works well on $\mathcal{S}$

The **training error** or **empirical risk** $L_{\mathcal{S}}$ is a notion of loss which can be calculated on $\mathcal{S}$

It is defined as the error that classifier h incurs over sample $\mathcal{S}$ with N labelled examples

$$L_{\mathcal{S}}(h) \equiv \frac{|\{x_n : h(x_n) \neq y_n\}_{n=1}^{N}|}{N}$$

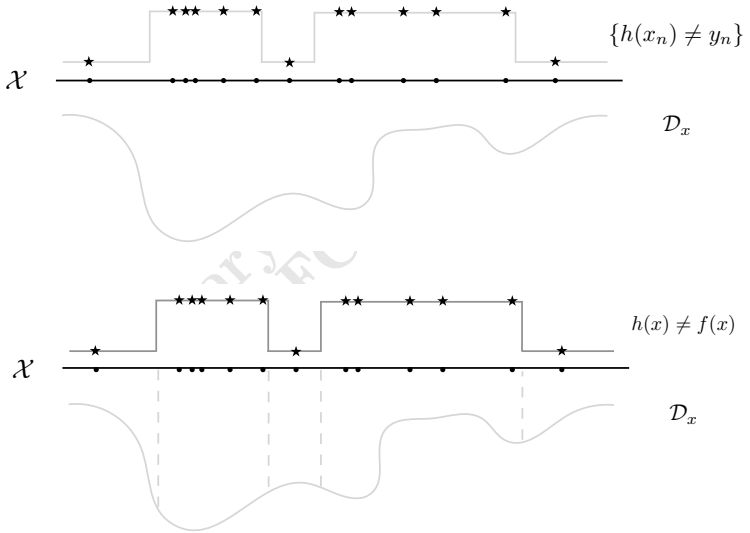
That is, $L_{\mathcal{S}}(h)$ is defined as the fraction of training examples mislabeled by the rule h

- As such, $L_{\mathcal{S}}(h)$ can be calculated without knowing anything about $\mathcal{D}_x$ and f

January 15, 2025 — NPC —

Empirical risk (cont.)

Graphically, $L_{\mathcal{S}}(h)$ is the number of instances $(x_n, y_n = f(x_n))$ mislabelled by a rule h

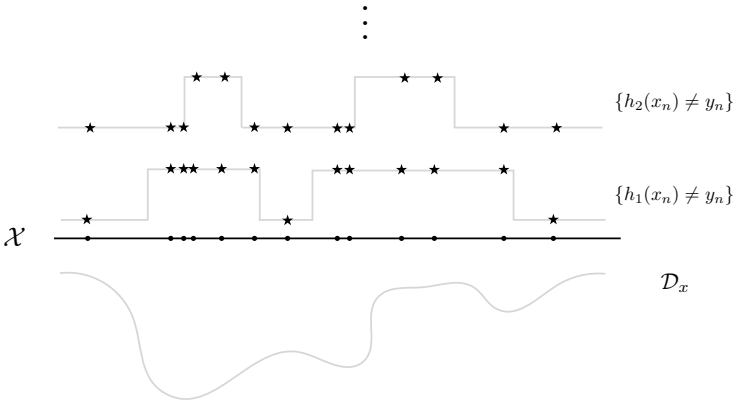


Empirical risk minimisation

A learning paradigm that returns a rule h that minimises $L_{\mathcal{S}}$ (that works well on the training data $\mathcal{S}$) is said to operate according to the **Empirical Risk Minimisation** (**ERM**)

$$\text{ERM}(\mathcal{S}) \in \arg\min_{h:\mathcal{X}\to\mathcal{Y}} \underbrace{\frac{|\{x_n : h(x_n) \neq y_n\}_{n=1}^{N}|}{N}}_{L_{\mathcal{S}}(h)}$$

$\arg\min_{h:\mathcal{X}\to\mathcal{Y}}$ is that subset $\{h_{\mathcal{S}}\}$ of predictors that minimise the empirical error $L_{\mathcal{S}}$



Empirical risk minimisation | What could possibly go wrong?

$$\mathrm{ERM}(\mathcal{S}) \in \arg\min_{h:\mathcal{X}\to\mathcal{Y}} L_{\mathcal{S}}(h)$$

Given that the learner has access to all possible functions $h \in \mathcal{X}^{\mathcal{Y}}$, why not just pick one that has zero error on the training sample $\mathcal{S}$ (or, equivalently such that $L_{\mathcal{S}}(h) = 0$)?

Since we assumed that labels are deterministically set, $y = f(x)$, we can design such h

$$h_{\mathcal{S}}(x) = \begin{cases} y_n, & \text{if } x \in \{x_n\} \\ 0 \text{ (or 1)}, & \text{otherwise} \end{cases}, \qquad \mathcal{S} = \{(x_n \in \mathcal{X}, y_n \in \{0,1\})\}$$

Such a predictor will always achieve a perfect empirical error, regardless of sample $\mathcal{S}$

- As such, it can be chosen when using the ERM learning strategy
- Clearly, no rule can achieve a smaller loss on $\mathcal{S}$, as $L_{\mathcal{S}}(h_{\mathcal{S}}) = 0$

Predictor $h_{\mathcal{S}}$ has an excellent performance on $\mathcal{S}$, yet its true performance is very poor

- This phenomenon is the infamous **overfitting**

January 15, 2025
PCW

Empirical risk minimisation | Overfitting

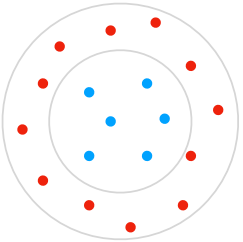
Example

Consider the problem of labelling a set of points x uniformly distributed inside a circle

Consider some labelling function f

- label $y = (\cdot)$ to points x that are within the inner circle
- label $y = (\cdot)$ to other points

Let the area of the outer circle be 2 and that of the inner circle be 1



January 15, 202
FC

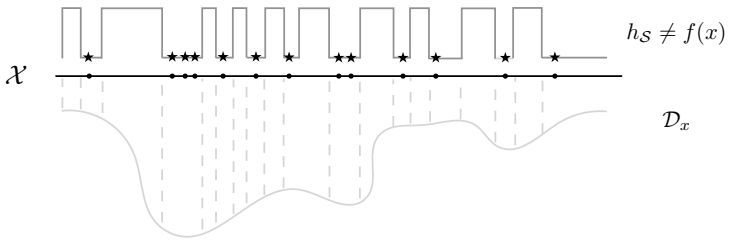
We are given sample $\mathcal{S} = \{(x_n, y_n)\}$ and now consider the following prediction rule $h_{\mathcal{S}}$

$$h_{\mathcal{S}}(x) = \begin{cases} y_n, & \text{if } x \in \{x_n\} \\ (\cdot), & \text{otherwise} \end{cases} \qquad \rightsquigarrow \qquad L_{\mathcal{S}}(h_{\mathcal{S}}) = 0$$

The true error of any classifier that predicts the label $(\cdot)$ only on finite sample $\mathcal{S}$ is $1/2$

$$L_{\mathcal{D}_x, f}(h_{\mathcal{S}}) = 1/2$$

Empirical risk minimisation | Overfitting (cont.)



How to amend the ERM($\mathcal{S}$) in a way that the learner is protected against overfitting?

- ... considering that all the learner has access to is the sample $\mathcal{S}$

We will discuss certain conditions under which the ERM is unlikely to overfit the data

- We ask how to find a predictor h with good performance with respect to $\mathcal{S}$
- and, good performance over the (unknown) distribution $\mathcal{D}_x$ and function f

Inductive bias

Empirical risk minimisation

January 13, 2025 – FC

ERM | Inductive bias

One strategy to fix the ERM($\mathcal{S}$) would be to apply it over some restricted search space

Before seeing the data $\mathcal{S}$, the learner picks a class of predictors, the **hypothesis class** $\mathcal{H}$

- Each prediction rule $h \in \mathcal{H}$ must be a function which maps $\mathcal{X}$ to $\mathcal{Y}$

$$\mathcal{H} \subset \{h : h \in \mathcal{X}^{\mathcal{Y}}\}$$

- We might also assume that the target function f is in the set $\mathcal{H}$

By selecting $\mathcal{H}$, we are including a form of **prior knowledge** into the learning paradigm

- The choice of $\mathcal{H}$ should be based on some knowledge about the learning task
- (Say, we assume that same-class instances are bounded to certain regions)

We point the learner towards a class of prediction rules (**inductive bias**) by restricting it to pick only predictors from a hypothesis class $\mathcal{H}$, chosen before seeing a sample $\mathcal{S}$

- We have shown that without prior knowledge, ERM learners cannot learn

January 15, 2025

ERM | Inductive bias (cont.)

For the chosen hypothesis class $\mathcal{H}$ and given some training set $\mathcal{S}$, an $\text{ERM}_{\mathcal{H}}(\mathcal{S})$ learner uses $\text{ERM}(\mathcal{S})$ strategy to pick rules $\{h_{\mathcal{S}}\}$ in $\mathcal{H}$ with smallest loss $L_{\mathcal{S}}$ over that sample

$$\text{ERM}_{\mathcal{H}}(\mathcal{S}) \in \arg\min_{h \in \mathcal{H}} \underbrace{\frac{|\{x_n : h(x_n) \neq y_n\}_{n=1}^{|\mathcal{S}|}|}{|\mathcal{S}|}}_{L_{\mathcal{S}}(h)}$$

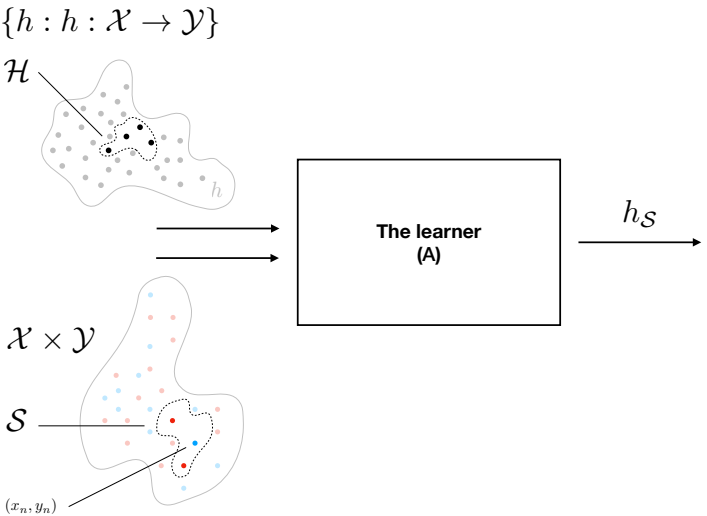
Again, $\arg\min_{h \in \mathcal{H}}$ is the subset of rules $h_{\mathcal{S}} \in \mathcal{H}$ which minimise the empirical loss $L_{\mathcal{S}}$

In many cases (under certain assumptions), $\text{ERM}_{\mathcal{H}}(\mathcal{S})$ is a successful learning strategy

- That is, it leads to picking hypothesis $h_{\mathcal{S}}$ with small generalisation error $L_{\mathcal{D}_x, f}$

January 15, 2025 — FC —

ERM | Inductive bias (cont.)



Notice that a fundamental question in statistical learning theory is 'over which class $\mathcal{H}$ of hypothesis functions, the learning paradigm $\text{ERM}_{\mathcal{H}}$ will not lead to overfitting?'

Finite hypothesis

Empirical risk minimisation | Inductive bias

January 15, 2025
– FC –

ERM | Inductive bias | Finite hypothesis

The simplest restriction on class $\mathcal{H}$ is obtained by imposing an upper bound on its size

$$\mathcal{H} = \{h_{n_h} : h_{n_h} \in \mathcal{X}^{\mathcal{Y}}\}_{n_h=1}^{N_h} \qquad (\text{with } N_h < \infty)$$

That is, we select a hypothesis class $\mathcal{H}$ whose number $N_h = |\mathcal{H}|$ of predictors h is finite

Theorem
It can be shown that, if $\mathcal{H}$ is a finite hypothesis class ($|\mathcal{H}| < \infty$) and a sufficiently large training sample $\mathcal{S}$ ($|\mathcal{S}| > \text{const}(|\mathcal{H}|)$) is available, then $\text{ERM}_{\mathcal{H}}(\mathcal{S})$ is unlikely to overfit

- **Assumption:** The x-elements of $\mathcal{S}$ are independent draws from $\mathcal{D}_x$
- **Assumption:** The correct labelling function f is also in class $\mathcal{H}$

We show that, without limiting $\mathcal{H}$ to be finite, the $\text{ERM}_{\mathcal{H}}(\mathcal{S})$ learner would always have a large probability of error on new data from $(\mathcal{D}_x, f)$, regardless of how large $\mathcal{S}$ is

January 13, 2025

ERM | Inductive bias | Finite hypothesis (cont.)

January 13, 2025
FC

This theorem highlights how learning refers to a different notion from hypothesis testing

- In hypothesis testing, we come up with an hypothesis before seeing the data
- Conversely, in machine learning we select an hypothesis based on the data

$$\underbrace{h_{\mathcal{S}} \in \arg\min_{h \in \mathcal{H}} L_{\mathcal{S}}(h)}_{\text{via } \mathrm{ERM}_{\mathcal{H}}}$$

Learning is about theories developed from data, not about theories chosen before data

- (Though, we still picked a hypothesis class rather than a single hypothesis)

ERM | Inductive bias | Finite hypothesis | Accuracy

January 15, 2025

For an algorithm A that has access only to sample $\mathcal{S}$, any guarantee on the error with respect to the underlying distribution must depend on the relation between $\mathcal{D}_x$ and $\mathcal{S}$

The sample $\mathcal{S}$ is the window through which the learner gets information about $(\mathcal{D}_x, f)$

- Intuitively, the larger $\mathcal{S}$ is the more representative it is of $\mathcal{D}_x$ and f
- We saw how a non representative sample leads the ERM to overfit

That is, we are interested in (avoiding) those samples that confuse the $\text{ERM}_{\mathcal{H}}$ learner

Inductive bias | Finite hypothesis | Accuracy (cont.)

January 15, 2025
_PC_

More precisely, for some fixed labelling function $f \in \mathcal{X}^{\mathcal{Y}}$, we want to determine what is the maximum probability to sample N instances that leads to a true failure of $\mathrm{ERM}_{\mathcal{H}}$

We quantify failure by introducing a fixed **accuracy parameter** $\varepsilon \in [0, 1]$ of the prediction

- Parameter ε permits us to interpret the event $\underbrace{L_{\mathcal{D}_x,f}(h_{\mathcal{S}})}_{\text{true risk}} > \varepsilon$ as $\mathrm{ERM}_{\mathcal{H}}$ failure
- Conversely, $L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) \leq \varepsilon$ defines an *approximately correct* $\mathrm{ERM}_{\mathcal{H}}$ predictor

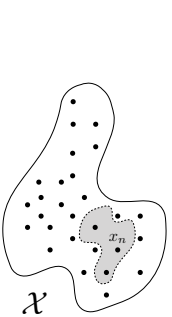
To identify failing samples, let $\mathcal{S}_x = \{x_n\}_{n=1}^{N}$ be the domain points in any training set

- For the collection of all confusing training samples $\mathcal{S}_x$, we have

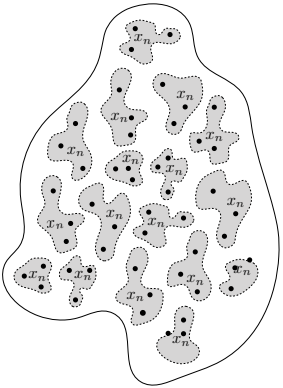
$$\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}$$

This set of sets cannot be determined because, though we could identify for each a $h_{\mathcal{S}}$ via $\mathrm{ERM}_{\mathcal{H}}$, we are not able to establish what its true risk is (as $\mathcal{D}_x$ and f are unknown)

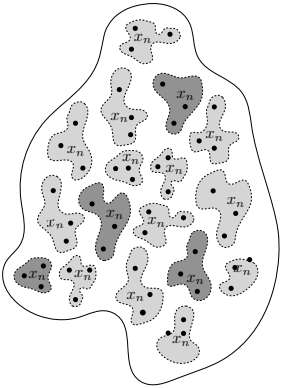
Inductive bias | Finite hypothesis | Accuracy (cont.)



$\mathcal{S}_x$



$\{\mathcal{S}_x\}$



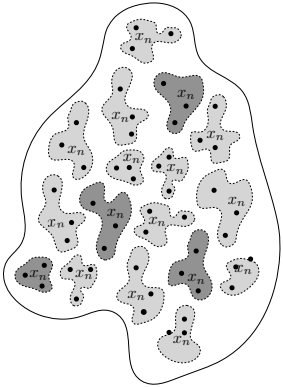
$\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}$

For each possible sample $\mathcal{S}_x$, the ERM$_{\mathcal{H}}$ strategy determines an optimal predictor $h_{\mathcal{S}}$

- Yet, the true risk of these predictors is not accessible to the learner

Inductive bias | Finite hypothesis | Accuracy (cont.)

We want to determine under which conditions, for the assumed mechanism for generating samples $\mathcal{S}$, the probability of observing a non-representative sample is very small



For each possible sample $\mathcal{S}_x$, $\mathrm{ERM}_{\mathcal{H}}$ strategy determines the optimal prediction rule $h_{\mathcal{S}}$

We are interested in upper bounding the probability of drawing a confusing sample

$$\mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\})$$

As $\mathcal{D}_x$ is the probability of drawing a single x, we let $\mathcal{D}_x^N$ be that of N i.i.d. copies of x

$$\mathbb{P}_{\mathcal{S}_x \sim \mathcal{D}_x^N}[\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon]$$

Inductive bias | Finite hypothesis | Accuracy (cont.)

Now, we let $\mathcal{H}_B$ be the set of all hypothesis rules in $\mathcal{H}$ which are not ε-correct, at least

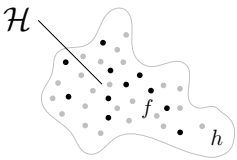
- $\mathcal{H}_B$ is the set of hypotheses which should be avoided by the learner

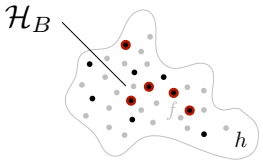
$$\mathcal{H}_B = \{h \in \mathcal{H} : \underbrace{L_{\mathcal{D}_x,f}(h)}_{\text{true risk}} > \varepsilon\}$$

Because $f \in \mathcal{H}$ and $L_{\mathcal{D}_x,f}(f) = 0$, we have that $\mathcal{H}_B \subset \mathcal{H}$ and thus also that $|\mathcal{H}_B| < |\mathcal{H}|$

- Set $\mathcal{H}_\mathcal{B}$ is stated regardless of the $\text{ERM}_\mathcal{H}$ strategy, as it only pertains $\{\mathcal{S}_x\}$

Also notice how also this set of functions cannot be established (unknown $\mathcal{D}_x$ and f)





Inductive bias | Finite hypothesis | Accuracy (cont.)

The state of things, up to this point

- The set of all possible samples that can be generated

$$\{\mathcal{S}_x\}$$

- The set of all truly bad predictors, regardless of the sample

$$\mathcal{H}_B = \{h \in \mathcal{H} : L_{\mathcal{D}_x,f}(h) > \varepsilon\}$$

- The set of samples that are truly bad, if accessed by the $\text{ERM}_{\mathcal{H}}$

$$\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}$$

January 15, 2025 — IC

How to combine all the info into something that can be practically analysed and used?

What we want to avoid are those samples that, though they lead to a good $\text{ERM}_{\mathcal{H}}$ performance (small $L_{\mathcal{S}}(h_{\mathcal{S}})$), would still perform badly in a true sense ($L_{\mathcal{D}_x,f}(h) > \varepsilon$)

Before we can proceed with such a set, we need to introduce an additional assumption

ERM | Inductive bias | Finite hypothesis | Realisability

Assumption
Let us assume (**realisability assumption**) there exists one $h^* \in \mathcal{H}$ such that $L_{\mathcal{D}_x,f}(h^*) = 0$

$$L_{\mathcal{D}_x,f}(h^*) = \underbrace{\mathbb{P}_{x\sim\mathcal{D}_x}[h^*(x) \neq f(x)]}_{\text{true risk}}$$

$$= 0$$

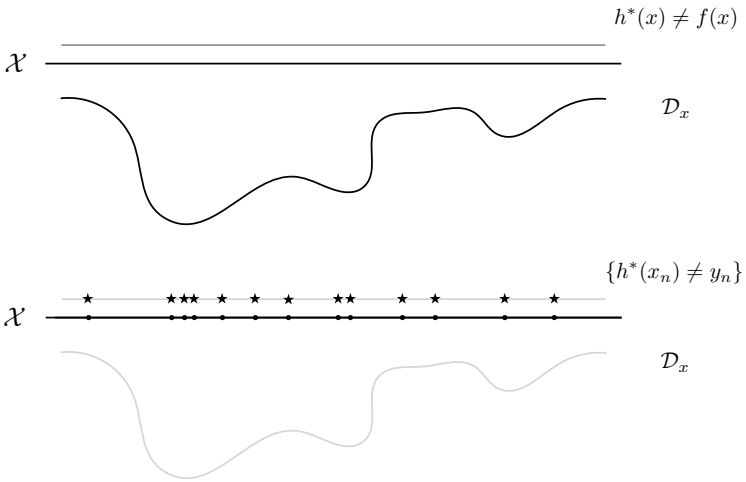
The assumption implies that with probability 1 over samples $\mathcal{S}$, chosen according to $\mathcal{D}_x^N$ and then labeled by f, there is at least one rule $h^* \in \mathcal{H}$ such that also $L_{\mathcal{S}}(h^*) = 0$

$$L_{\mathcal{S}}(h^*) = \underbrace{\frac{|\{x_n : h^*(x_n) \neq y_n\}_{n=1}^N|}{N}}_{\text{empirical risk}}$$

$$= 0$$

January 15, 2025

ERM | Inductive bias | Finite hypothesis | Realisability (cont.)



When we assumed that $f \in \mathcal{H}$, we have implicitly satisfied the realisability assumption

ERM | Inductive bias | Finite hypothesis | Accuracy (cont.)

Because of the realisability assumption, we have that the event $L_{\mathcal{D}_x,f}(h_S) > \varepsilon$ can only occur whenever, for some $h \in \mathcal{H}_B$, we draw a *misleading sample* such that $L_S(h) = 0$

- That is, for a truly bad hypothesis h which empirically performs as well as f

Let $\mathcal{M}_x$ denote the subset of those samples $\mathcal{S}_x$ for which there exists a truly bad rule ($h \in \mathcal{H}_B$) which would still lead to a good performance, empirically (in ERM$_\mathcal{H}$-sense)

$$\mathcal{M}_x = \{\mathcal{S}_x : \exists h \in \mathcal{H}_B, \quad \underbrace{L_S(h)}_{\text{empirical risk}} = 0\}$$

$$= \{\mathcal{S}_x : \exists h \in \{h \in \mathcal{H} : \underbrace{L_{\mathcal{D}_x,f}(h)}_{\text{true risk}} > \varepsilon\}_{\text{is bad}}, \quad \underbrace{L_S(h)}_{\text{empirical risk}} = 0\}_{\text{looks good}}$$

January 15, 2025

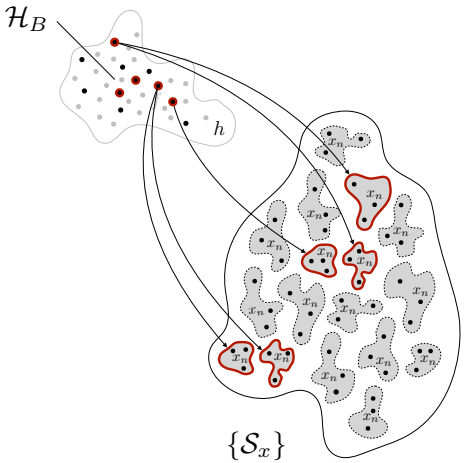
These samples are misleading, because they make the truly bad hypotheses look good

- Thus $\mathcal{M}_x$ is the set of samples for which there exists a truly bad hypothesis

Inductive bias | Finite hypothesis | Accuracy (cont.)

One way to construct set $\mathcal{M}_x$ is to consider each truly bad predictor $h \in \mathcal{H}_B$ and then identify all samples $\mathcal{S}_x$ such that, when h is given to $\text{ERM}_{\mathcal{H}}$, its empirical loss is zero

$$\mathcal{M}_x = \{\mathcal{S}_x : \exists h \in \mathcal{H}_B, L_{\mathcal{S}}(h) = 0\}$$
$$= \Big\{ \bigcup_{h \in \mathcal{H}_B} \{\mathcal{S}_x : L_{\mathcal{S}}(h) = 0\} \Big\}$$



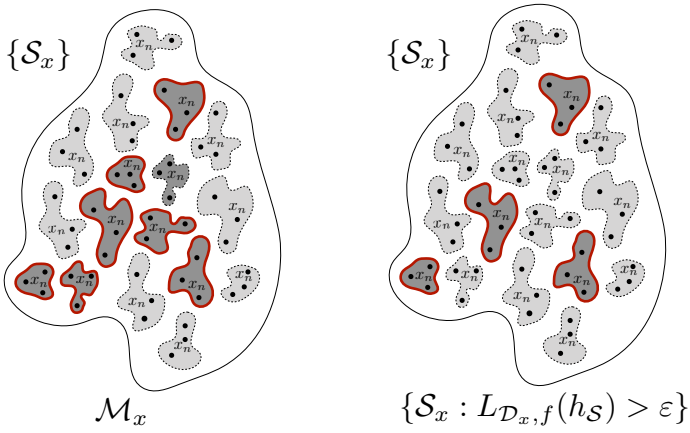
Does $\mathcal{M}_x$ relate to the target set $\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}$?

- $\mathcal{M}_x$ is about the existence of a bad hypothesis
- $\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}$ is about selecting it

Thus, we have

$$\{S_x : L_{\mathcal{D},f}(h_S) > \varepsilon\} \subseteq \mathcal{M}_x$$

Inductive bias | Finite hypothesis | Accuracy (cont.)



Remembering that we want to upper bound $\mathbb{P}_{\mathcal{S}_x \sim \mathcal{D}_x^N}[\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_\mathcal{S}) > \varepsilon]$, , it suffices to upper bound $\mathbb{P}_{\mathcal{S}_x \sim \mathcal{D}_x^N}[\mathcal{S}_x : \exists h \in \mathcal{H}_B, L_\mathcal{S}(h) = 0]$, because $\{S_x : L_{\mathcal{D},f}(h_\mathcal{S}) > \varepsilon\} \subseteq \mathcal{M}_x$

$$\mathbb{P}_{\mathcal{S}_x \sim \mathcal{D}_x^N}[\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_\mathcal{S}) > \varepsilon] \leq \mathbb{P}_{\mathcal{S}_x \sim \mathcal{D}_x^N}[\mathcal{S}_x : \exists h \in \mathcal{H}_B, L_\mathcal{S}(h) = 0]$$

Inductive bias | Finite hypothesis | Accuracy (cont.)

$$\mathbb{P}_{\mathcal{S}_x \sim \mathcal{D}_x^N}[\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon] \leq \mathbb{P}_{\mathcal{S}_x \sim \mathcal{D}_x^N}[\mathcal{S}_x : \exists h \in \mathcal{H}_B, L_{\mathcal{S}}(h) = 0]$$

That is, we have

$$\begin{aligned}\mathcal{D}_x^N(\{\mathcal{S}_x : L_{(\mathcal{D},f)}(h_{\mathcal{S}}) > \varepsilon\}) &\leq \mathcal{D}_x^N(\mathcal{M}_x) \\ &= \mathcal{D}_x^N(\bigcup_{h \in \mathcal{H}_B} \{\mathcal{S}_x : L_{\mathcal{S}}(h) = 0\})\end{aligned}$$

By using the usual union bound ($\mathcal{D}(A \cup B) \leq \mathcal{D}(A) + \mathcal{D}(B)$), we have the inequality

$$\mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}) \leq \sum_{h \in \mathcal{H}_B} \mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{S}}(h) = 0\})$$

The inequality allows us to bound each summand $\mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{S}}(h) = 0\})$ individually

January 15, 2025 — FC —

Inductive bias | Finite hypothesis | Accuracy (cont.)

$$\mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}) \leq \sum_{h \in \mathcal{H}_B} \mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{S}}(h) = 0\})$$

By fixing a $h \in \mathcal{H}_B$, we observe that $L_{\mathcal{S}}(h) = 0$ corresponds to $h(x_n) = f(x_n)$ for all n

$$\mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{S}}(h) = 0\}) = \mathcal{D}_x^N(\{\mathcal{S}_x : h(x_n) = f(x_n), \text{ for all } n\})$$

As the examples are independently sampled from the same distribution (i.i.d.), we get

$$\mathcal{D}_x^N(\{\mathcal{S}_x : h(x_n) = \underbrace{f(x_n)}_{y_n}, \text{ for all } n\}) = \prod_{n=1}^{N} \mathcal{D}_x(\{x_n : h(x_n) = \underbrace{f(x_n)}_{y_n}\})$$

For an individual possible draw of a training sample and a tolerable failure ε, we get

$$\begin{aligned}
\mathcal{D}_x(\{x_n : h(x_n) = y_n\}) &= 1 - \mathcal{D}_x(\{x_n : h(x_n) \neq y_n\}) \\
&= 1 - L_{\mathcal{D}_x,f}(h) \\
&\leq 1 - \varepsilon \\
&\leq \exp(-\varepsilon)
\end{aligned}$$

January 15, 2025
FC

Inductive bias | Finite hypothesis | Accuracy (cont.)

$$\mathcal{D}_x(\{x_n : h(x_n) = y_n\}) \leq \exp(-\varepsilon)$$

$$\mathcal{D}_x^N(\{\mathcal{S}_x : h(x_n) = y_n, \text{ for all } n\}) = \prod_{n=1}^{N} \mathcal{D}_x(\{x_n : h(x_n) = y_n\})$$

By combining the results relative to one $h \in \mathcal{H}_B$, we get

$$\begin{aligned}\mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{S}}(h) = 0\}) &\leq (1-\varepsilon)^N \\ &\leq \exp(-N\varepsilon)\end{aligned}$$

Remembering that we have $|\mathcal{H}_B|$ such hypothesis, we have

$$\sum_{h \in \mathcal{H}_B} \mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{S}}(h) = 0\}) \leq |\mathcal{H}_B| \exp(-N\varepsilon)$$

For the probability of drawing a non-representative (ε-wrong) sample of size $N = |\mathcal{S}|$,

$$\begin{aligned}\mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}) &\leq \sum_{h \in \mathcal{H}_B} \mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{S}}(h) = 0\}) \\ &\leq |\mathcal{H}_B| \exp(-|\mathcal{S}|\varepsilon) \\ &\leq |\mathcal{H}| \exp(-|\mathcal{S}|\varepsilon)\end{aligned}$$

January 15, 202
FC_

Inductive bias | Finite hypothesis | Accuracy (cont.)

$$\mathcal{D}_x^N(\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}) \leq |\mathcal{H}| \exp(-|\mathcal{S}|\varepsilon)$$

The bound decays exponentially with the number $|\mathcal{S}|$ of data and tolerable accuracy ε

- (The larger the training set the better)

Still, the bond grows linearly with the number $|\mathcal{H}|$ of hypotheses in the selected class

January 15, 2025
FCC