Aalto University

Statistical machine learning | Learning models

Introduction to machine learning

Francesco Corona

Chemical and Metallurgical Engineering
School of Chemical Engineering

Recap

We studied how $L_{\mathcal{D}_x,f}(h)$ depends on the training sample $\mathcal{S}$, which is randomly picked

- Thus, also $L_{\mathcal{D}_x,f}(h_{\mathcal{S}})$ is a random variable

This fact has led us to recognise the randomness in the choice of $\mathrm{ERM}_{\mathcal{H}}$ predictors $h_{\mathcal{S}}$

We cannot always expect that sample $\mathcal{S}$ is sufficient to guarantee ε-good predictors $h_{\mathcal{S}}$

- At least, not with respect to $\mathcal{D}_x$ and f
- That is, that $L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) \leq \varepsilon$

There is always some probability that $\mathcal{S}$ is non-representative of the underlying data

If we accept the realisability and other assumptions, this probability is upperbounded

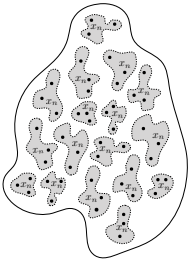
$$\mathcal{D}_x(\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}) \leq |\mathcal{H}| \exp(-|\mathcal{S}|\varepsilon)$$

Recap (cont.)

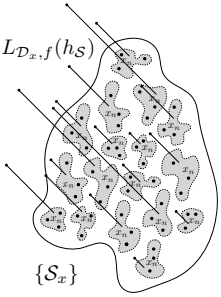
Before the data, we select a finite hypothesis class $\mathcal{H}$ including labelling function f and asked what is the probability that sample $\mathcal{S}$ is ε-wrong ($L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon$) for the ERM$_{\mathcal{H}}$

We considered all possible samples of size $|\mathcal{S}_x|$ and labelled them with function f

For each sample, the ERM$_{\mathcal{H}}$ would select a rule $h_{\mathcal{S}}$ whose true error is $L_{\mathcal{D}_x,f}(h_{\mathcal{S}})$



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

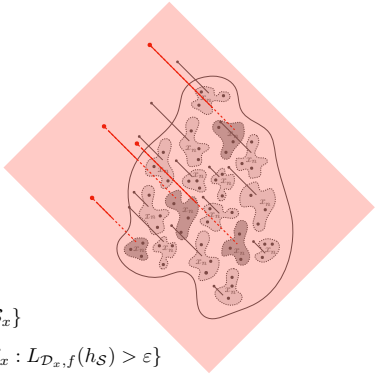


$\{\mathcal{S}_x\}$

The true risk cannot be calculated, as both the probability distribution $\mathcal{D}_x$ and the exact labelling function f are not accessible to the learner (here, based on the ERM$_{\mathcal{H}}$)

Recap (cont.)

We only know that certain samples $\mathcal{S}$ will lead to a failure of the learner ($L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon$)



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

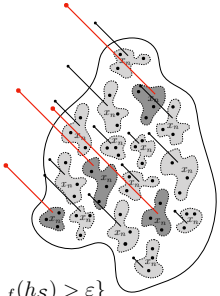
$\{\mathcal{S}_x : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\}$

This probability cannot be calculated and we ended up determining an upper bound, depending on $\mathcal{S}$ and $\mathcal{H}$

- A distribution-free quantity

Yet, we are interested in understanding how the probability of drawing such a sample depends on the learning set up

- Sample size $|\mathcal{S}|$
- Class size $|\mathcal{H}|$



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

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

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

PAC learning

Learning models

Accuracy and confidence

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

Let δ be the acceptable probability of drawing a non-representative sample $\mathcal{S}$ for $\text{ERM}_\mathcal{H}$

- We use $\delta \in [0,1]$ to define $(1-\delta)$ as the **confidence** on $h_\mathcal{S}$

An $\text{ERM}_\mathcal{H}$ hypothesis is then said to be **probably $(1-\delta)$ approximately (ε) correct**, **PAC**

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

The given notion of probably approximately correct contains two parameters (ε and δ)

- Accuracy ε indicates what is the tolerable magnitude of the true error
- Confidence $(1-\delta)$ shows how likely the solution is to meet accuracy

The two quantities are inevitable parameters under the typical data generating model

Informally, with PAC we are asking under which condition the learner (here, $\text{ERM}_\mathcal{H}$) is at least ε-correct most of the time, that is with a probability which is at least $(1-\delta)$

- The learner has only access to the hypothesis class $\mathcal{H}$ and the data $\mathcal{S}$
- This requirement is guaranteed when enough instances $|\mathcal{S}|$ are given

Accuracy and confidence (cont.)

For some fixed domain set $\mathcal{X}$, label set $\mathcal{Y}$, and hypothesis class $\mathcal{H}$, we have these steps

The user chooses the accuracy $\varepsilon \in (0, 1)$ and the confidence $(1 - \delta) \in (0, 1)$

- The pair (ε, δ) is accessible to the $\text{ERM}_{\mathcal{H}}$

To ensure success, the $\text{ERM}_{\mathcal{H}}$ requires a minimum amount of instances

- The requirement is independent of $\mathcal{D}_x$ and f
- They are unaccessible to the $\text{ERM}_{\mathcal{H}}$, anyway

The user provides the $\text{ERM}_{\mathcal{H}}$ with a sample $\mathcal{S}_x \sim \mathcal{D}_x$ labelled by $f \in \mathcal{H}$

The $\text{ERM}_{\mathcal{H}}$ returns a predictor $h_{\mathcal{S}}$

- The $\text{ERM}_{\mathcal{H}}$ may succeed ($L_{\mathcal{D}_x, f}(h_{\mathcal{S}}) < \varepsilon$)
- The $\text{ERM}_{\mathcal{H}}$ may fail ($L_{\mathcal{D}_x, f}(h_{\mathcal{S}}) > \varepsilon$)

If this is repeated many times, the $\text{ERM}_{\mathcal{H}}$ is guarantees with high probability, $(1 - \delta)$

Learner

Although given for the $\mathrm{ERM}_{\mathcal{H}}$, the notion of probably approximately correct is general

To extend and utilise it beyond the $\mathrm{ERM}_{\mathcal{H}}$, firstly we revise our definition of a learner

Definition
Formally, we redefine the **learner** as a function A that takes any possible sample $\mathcal{S} = \{(x_n, y_n)\}_{n=1}^{N}$ of any possible size $N = |\mathcal{S}|$ as input, and outputs a hypothesis $h \in \mathcal{X}^{\mathcal{Y}}$

$$A : \bigcup_{n=1}^{\infty} \{(x_n, y_n)\}_{n=1}^{|\mathcal{S}|} \to \{h : \mathcal{X} \to \mathcal{Y}\}, \qquad \text{with } (x_n, y_n) \in \mathcal{X} \times \underbrace{\mathcal{Y}}_{\{0,1\}}$$

The weighty bit is that the output $h = A(\mathcal{S})$ is no longer required to be in a class $\mathcal{H}$

- Moreover, function f no longer need be in class $\mathcal{H}$
- (For now, we still hold on to this assumption)

Sample complexity

Definition
For any hypothesis class $\mathcal{H}$ and $\varepsilon, \delta \in (0,1)$, we define a function $n_{\mathcal{H}} : (0,1) \times (0,1) \to \mathbb{N}$

Such a function determines the **sample complexity** of learning from hypothesis class $\mathcal{H}$

- It is a function of the accuracy and the confidence

It quantifies how many examples guarantee a probably approximately correct solution

PAC learnability

PAC learning

PAC learning | Learnability

We use the definition of sample complexity to introduce the notion of **PAC learnability**

- We provide a definition which is valid for any learning algorithm A

Definition
Class $\mathcal{H}$ is said to be **PAC learneable**, if there exists a function $n_{\mathcal{H}}$ such that, for all $|\mathcal{S}| \geq n_{\mathcal{H}}(\varepsilon, \delta)$, for every $\mathcal{D}_x$, and every $f : \mathcal{X} \to \mathcal{Y}$, there is a learner A for which the probability over samples that the error $L_{\mathcal{D}_x, f}(A(\mathcal{S}))$ is larger than ε is smaller than δ

$$\mathbb{P}_{\mathcal{S} \sim \mathcal{D}_x^N, f}\left[\mathcal{S} : L_{\mathcal{D}_x, f}(A(\mathcal{S})) > \varepsilon\right] < \delta \qquad \begin{cases} \text{For all sample sizes } N \geq n_{\mathcal{H}}(\varepsilon, \delta) \\ \text{For all data distributions } \mathcal{D}_x \\ \text{For all labelling functions } f \in \mathcal{X}^{\mathcal{Y}} \end{cases}$$

It follows that $n_{\mathcal{H}}(\varepsilon, \delta)$ is the minimum number of data needed for learning through $\mathcal{H}$

- ... with confidence $(1 - \delta)$ and accuracy ε, at least
- ... given that f exists and belongs to that class $\mathcal{H}$

PAC learning | Learnability (cont.)

For some fixed domain set $\mathcal{X}$, label set $\mathcal{Y}$, and hypothesis class $\mathcal{H}$, we have these steps

The user chooses the accuracy $\varepsilon \in (0, 1)$ and the confidence $(1 - \delta) \in (0, 1)$

- The pair (ε, δ) is accessible to the learner A

To ensure success, the learner A requires a minimum amount $n_{\mathcal{H}}$ of instances

- The requirement is independent of $\mathcal{D}_x$ and f
- They are unaccessible to A, anyway

The user provides the learner A with a sample $\mathcal{S}_x \sim \mathcal{D}_x$ labelled by $f \in \mathcal{H}$

The learner A returns a predictor $A(\mathcal{S})$

- A may succeed $(L_{\mathcal{D}_x, f}(A(\mathcal{S})) < \varepsilon)$
- A may fail $(L_{\mathcal{D}_x, f}(A(\mathcal{S})) > \varepsilon)$

If this is repeated many times, the learner A is guarantees with high probability, $(1-\delta)$

If there exists at least one learner A such that this is satisfied, class $\mathcal{H}$ is PAC learnable

PAC learning | Learnability (cont.)

Example

We can establish whether any finite class $\mathcal{H}$ is PAC learneable under the $\mathrm{ERM}_{\mathcal{H}}$ trick

$$\mathbb{P}_{\mathcal{S}\sim\mathcal{D}_x^N,f}\left[\mathcal{S} : L_{\mathcal{D}_x,f}(h_{\mathcal{S}}) > \varepsilon\right] \leq \underbrace{|\mathcal{H}|\exp\left(-|\mathcal{S}|\varepsilon\right) < \delta}$$

By taking the natural log of the second inequality, we get

$$\ln\left(|\mathcal{H}|\right) - \varepsilon|\mathcal{S}| < \ln\left(\delta\right)$$

Rearranging to get $|\mathcal{S}|$ and rounding up, we have

$$|\mathcal{S}| \geq \underbrace{\left\lceil \frac{\ln\left(|\mathcal{H}|/\delta\right)}{\varepsilon} \right\rceil}_{n_{\mathcal{H}}(\varepsilon,\delta)}$$

PAC learning | Learnability (cont.)

If $\mathcal{H}$ is PAC-learnable, there may be several functions $n_{\mathcal{H}}$ that satisfy the requirements

This motivates a definition of the sample complexity of learning $\mathcal{H}$ as the minimal $n_{\mathcal{H}}$

- From all triplets $(\varepsilon, \delta, n_{\mathcal{H}})$, we choose the one returning the smallest integer

---

We will show that what determines the PAC-learnability of a class is not its finiteness

- Rather, it is a combinatorial measure called the VC-dimension

We will also show that there are infinite classes of hypothesis which are PAC-learnable

A formal model of learning

Learning models

Formal model

We extend the learning model discussed so far to account for cases in which the realisability and exact labelling assumption are relaxed, and then to other learning tasks

Realisability assumption

Requiring that the learning algorithm succeeds on $\mathcal{D}_x$ and f when the realisability assumption is satisfied may be too strong an assumption for practical learning tasks

- We assumed that there exists a $h^* \in \mathcal{H}$ such that $\underbrace{\mathbb{P}_{x\sim\mathcal{D}_x}[x : h^*(x) = f(x)]}_{1-L_{\mathcal{D}_x,f}(h^*)} = 1$
- This assumption does not hold for the majority of real-world problems

Deterministic labelling

Unrealistically, we also assumed that label attribution is deterministic, given features

- We assumed that labels are fully determined by the features via $y = f(x)$

These are the main limitations of the PAC definition that we are interested to overcome

- How far can we go without assuming that f exists?
- How far can we go without assuming that $f \in \mathcal{H}$?

Formal model (cont.)

Realisability can be, naturally, relaxed by replacing the exact labelling process f assumed so far with a certain (conditional) probability distribution over the label set $\mathcal{Y}$

- We assume that labels are stochastically determined, given the features
- Implicitly, we also relax the assumption that labelling is deterministic

Though arbitrary and unknown to the leaner A, this distribution over $\mathcal{Y}$, like the distribution over $\mathcal{X}$, characterises how the domain-label generating process is modelled

- We are assuming that both domain instances and labels are picked randomly

$$\begin{aligned}\mathcal{D}_{xy} &= \mathcal{D}_{x|y}\mathcal{D}_y \\ &= \mathcal{D}_{y|x}\mathcal{D}_x\end{aligned}$$

Pragmatically, we introduce a more realistic data (domain-label) generating model and we are interested in the best prediction rule $h = A(\mathcal{S})$ that the learner could output

- (Keeping in mind that A has access to a finite sample $\mathcal{S}$ only)

For binary classification, that rule is a h that predicts the y for which $\mathcal{D}_{y|x}(y|\bar{x}) \geq 1/2$

- We discuss how to construct such an intuition
- We discuss why this rule cannot not used

Data model

Formal model

Formal model | Data model

Data model

Firstly, we will introduce an arbitrary (and marginal) distribution $\mathcal{D}_y$ over label set $\mathcal{Y}$

- This distribution allows us to introduce a joint domain-label distribution $\mathcal{D}_{xy}$
- The joint distribution $\mathcal{D}_{xy}$ is over $\mathcal{X} \times \mathcal{Y}$ and it is assumed to be arbitrary

$$\begin{aligned}\mathcal{D}_{xy} &= \mathcal{D}_{x|y}\mathcal{D}_y \\ &= \mathcal{D}_{y|x}\mathcal{D}_x\end{aligned}$$

Probability distribution $\mathcal{D}_{xy}$, though unknown, models the **data generating mechanism**

The data model $\mathcal{D}_{xy}$ implies the existence of two probability distributions over set $\mathcal{Y}$

$$\begin{aligned}\mathcal{D}_{xy} &= \mathcal{D}_{x|y}\underbrace{\mathcal{D}_y} \\ &= \underbrace{\mathcal{D}_{y|x}}\mathcal{D}_x\end{aligned}$$

From the factorisation of the joint distribution $\mathcal{D}_{xy}$

- A conditional distribution, $\mathcal{D}_{y|x}$
- A marginal distribution, $\mathcal{D}_y$

We characterise them both, from $\mathcal{D}_y$ towards $\mathcal{D}_{y|x}$

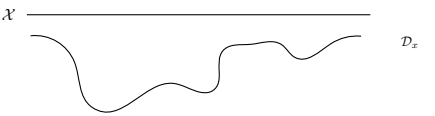
Formal model | Data model (cont.)

The marginal distribution $\mathcal{D}_x$ over $\mathcal{X}$

We already assumed the existence of a probability distribution $\mathcal{D}_x$ over $\mathcal{X}$

$$\mathcal{D}_{xy} = \mathcal{D}_{y|x}\,\underbrace{\mathcal{D}_x}$$

Training instances $\{x_n\}$ are random and independent draws from $\mathcal{D}_x^N$



The marginal distribution $\mathcal{D}_y$ over $\mathcal{Y}$

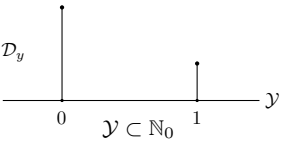
For presentation, we consider an arbitrary probability distribution $\mathcal{D}_y$ over $\mathcal{Y}$

$$\mathcal{D}_{xy} = \mathcal{D}_{x|y}\,\underbrace{\mathcal{D}_y}$$

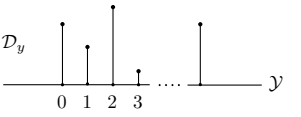
Again, $\mathcal{D}_y$ is unknown to the learner

This probability distribution is not directly relevant for our learning purposes

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

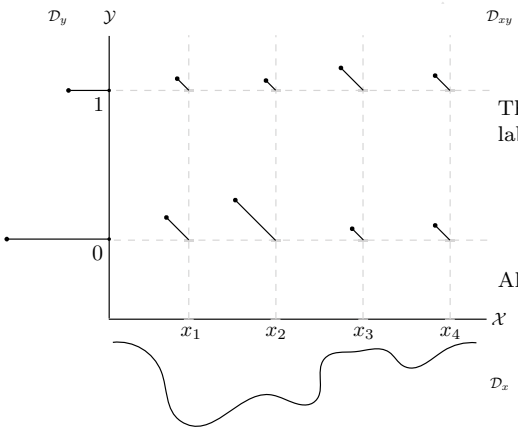


$\mathcal{Y} \subset \mathbb{N}_0$



Formal model | Data model (cont.)

The joint probability distribution $\mathcal{D}_{xy}$ (or $\mathcal{D}$) over $\mathcal{X} \times \mathcal{Y}$



The data model is a joint domain-label probability distribution $\mathcal{D}_{xy}$

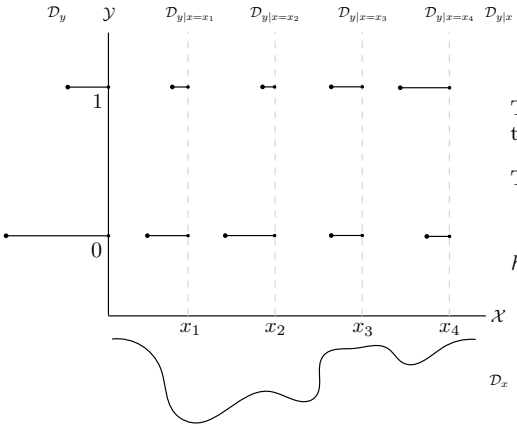
$$\underbrace{\mathcal{D}_{xy}} = \mathcal{D}_{y|x}\mathcal{D}_x$$
$$= \mathcal{D}_{x|y}\mathcal{D}_y$$

Also $\mathcal{D}_{xy}$ is unknown to the learner

Formal model | Data model (cont.)

The conditional probability distribution over $\mathcal{Y}$, given (the elements in) $\mathcal{X}$

It determines the probability of all the labels for all possible values of the domain set



$$\mathcal{D}_{xy} = \underbrace{\mathcal{D}_{y|x}}\, \mathcal{D}_x$$

The probability distribution $\mathcal{D}_{y|x}$ is the principal objective of learning

The best possible prediction rule h

- The **Bayes rule**

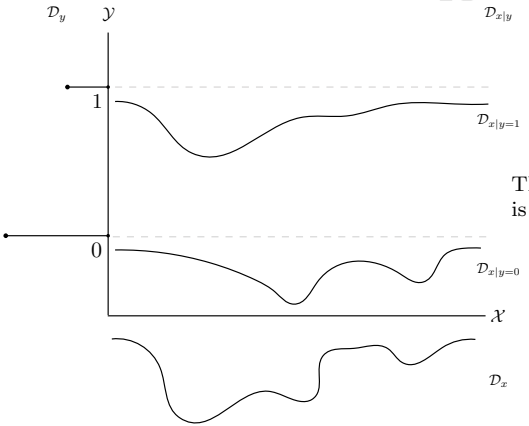
$$h^*(\bar{x}) = \begin{cases} 1, & \text{if } \mathcal{D}_{y|x}(1|\bar{x}) \geq 1/2 \\ 0, & \text{otherwise} \end{cases}$$

The Bayes rule h^* is optimal in the sense that no other rule $h \in \mathcal{X}^{\{0,1\}}$ has lower error

Formal model | Data model (cont.)

The conditional probability distribution over $\mathcal{X}$, given (the elements in) $\mathcal{Y}$

It determines the probability of all the domain instances for all values of the label set



$$\mathcal{D}_{xy} = \mathcal{D}_y \underbrace{\mathcal{D}_{x|y}}$$

The probability distribution $\mathcal{D}_{x|y}$ is not the objective of learning

Agnostic PAC

Formal model

Formal model | Agnostic PAC

Restating our assumptions and goals

Training instances $\{(x_n, y_n)\}$ are assumed to be from a joint distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$

- For our learning tasks, we again allow $\mathcal{D}$ to be an arbitrary distribution
- Again, a learner has no access to the probability distribution $\mathcal{D}$

For a probability distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, we are interested in determining how likely hypothesis h is to make an error when labelled points are drawn from such a $\mathcal{D}$

- That is, we need to refine what we mean by successful learning

Definition

We redefine the generalisation error, or risk, or loss, of the classifier $h \in \mathcal{X}^{\mathcal{Y}}$ as the probability that the label y of a pair (x, y) drawn according to $\mathcal{D}$, is predicted wrongly

$$L_{\mathcal{D}}(h) \equiv \underbrace{\mathbb{P}_{(x,y)\sim\mathcal{D}}\left[h(x) \neq y\right]}_{\mathcal{D}(\{(x,y):h(x)\neq y\})}$$

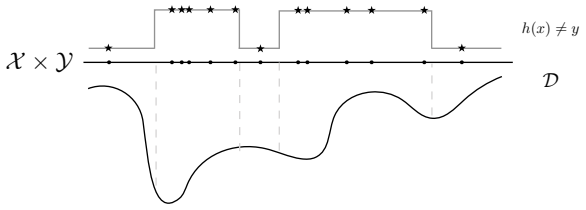
The error occurred by h is the probability of sampling a pair (x, y) for which $h(x) \neq y$

- $\mathcal{D}$ indicates that the risk L of $h = A(\mathcal{S})$ is evaluated with respect to $\mathcal{D}$

Formal model | Agnostic PAC (cont.)

$$L_{\mathcal{D}}(h) \equiv \mathbb{P}_{(x,y)\sim\mathcal{D}}\left[h(x) \neq y\right]$$

Graphically $L_{\mathcal{D}}(h)$ is the *volume* under the portion of $\mathcal{D}$ associated to errors of h on $\mathcal{Y}$



We are interested in a predictor h that, without knowing $\mathcal{D}$, minimises such an error

- Again, the learner has access to the training sample $\mathcal{S}$ only
- More importantly, there is no fixed f to compare against

The **empirical risk** remains unchanged and computable for any function $h : \mathcal{X} \to \{0, 1\}$

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

Formal model | Agnostic PAC | Learnability

No algorithm can find a hypothesis whose risk is smaller than the minimal possible one

It can be shown that, without prior assumption about the data-generating distribution, no algorithm is guaranteed to find a predictor that matches the minimal possible risk

Agnostic PAC learnability

A hypothesis class $\mathcal{H}$ is said to be agnostic PAC learnable if there exists some function $n_{\mathcal{H}} : (0,1) \times (0,1) \to \mathbb{N}$ and a learning algorithm A such that, for the hypothesis $A(\mathcal{S})$ from $|\mathcal{S}|$ independent examples from $\mathcal{D}$, the following upper-bound can be satisfied

$$\mathbb{P}_{\mathcal{S}\sim\mathcal{D}}\left[\mathcal{S} : L_{\mathcal{D}}(A(\mathcal{S})) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \varepsilon\right] < \delta, \qquad \begin{cases}\text{For all sample sizes } |\mathcal{S}| \geq n_{\mathcal{H}}(\varepsilon,\delta)\\ \text{For all data distributions } \mathcal{D}\end{cases}$$

That is, the leaner A is guaranteed to succeed if its error is at worst ε-worse than the best $h \in \mathcal{H}$, if at least $n_{\mathcal{H}}(\varepsilon,\delta)$ examples are available for learning and regardless of $\mathcal{D}$

Hypothesis $A(\mathcal{S})$ is said to be **probably $(1-\delta)$ approximately $(\min_{h\in\mathcal{H}} L_{\mathcal{D}}(h)+\varepsilon)$ correct**

If realisability holds ($f \in \mathcal{H}$), agnostic PAC learning and PAC learning provide the same guarantee, thus rendering agnostic PAC learning a more general notion of learnability

$$\min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) = 0$$

Formal model | Agnostic PAC | Learnability (cont.)

For some fixed domain set $\mathcal{X}$, label set $\mathcal{Y}$, and hypothesis class $\mathcal{H}$, we have these steps

The user chooses the accuracy $\varepsilon \in (0, 1)$ and the confidence $(1 - \delta) \in (0, 1)$

- The pair (ε, δ) is accessible to the learner A

To ensure success, the learner requires a minimum number $n_{\mathcal{H}}$ of instances

- The requirement is independent of the distribution $\mathcal{D}$
- $\mathcal{D}$ is not accessible to the learner, anyway

The user provides the learner A with a sample $\mathcal{S} \sim \mathcal{D}$

The learner A returns a predictor $A(\mathcal{S})$

- The learner may succeed ($L_{\mathcal{D}}(A(\mathcal{S})) < \varepsilon + \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h)$)
- The learner may fail ($L_{\mathcal{D}}(A(\mathcal{S})) > \varepsilon + \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h)$)

If this is repeated many times, the learner is guaranteed to succeed with high probability

$$(1 - \delta)$$

If there exists at least a learner A such that this holds, class $\mathcal{H}$ is agnostic PAC learnable

Formal model | Agnostic PAC | Learnability (cont.)

$$\mathbb{P}_{\mathcal{S}\sim\mathcal{D}}\left[\mathcal{S}: \underbrace{L_{\mathcal{D}}(A(\mathcal{S}))}_{\mathbb{P}_{(x,y)\sim\mathcal{D}}[A(\mathcal{S})\neq y]} > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h)+\varepsilon\right] < \delta \qquad \begin{cases}\text{For all } |\mathcal{S}| \geq n_{\mathcal{H}}(\varepsilon,\delta)\\ \text{For all } \mathcal{D}\end{cases}$$

For binary classification, the Bayes rule $h^\star$ would be the best possible prediction rule

- If it were available in some form and if it that form were included in $\mathcal{H}$

$$h^*(\bar{x}) = \begin{cases}1, & \text{if } \mathcal{D}_{y|x}(1|\bar{x}) \geq 1/2\\ 0, & \text{otherwise}\end{cases}$$

Agonistic PAC learnability would provide a relative guarantee of success for a learner A on $\mathcal{S}$

The learning algorithm A, which does not necessarily outputs a hypothesis $A(\mathcal{S}) \in \mathcal{H}$, is asked to compete against the best predictor in some benchmark hypothesis class $\mathcal{H}$

- We can see that classes $\mathcal{H}_a \subset \mathcal{H}_b$ are easier to compete against (to learn)

It is also important to notice that the learning algorithm A has access to $\mathcal{H}$ and to $\mathcal{S}$

Formal model | Agnostic PAC | Learnability (cont.)

A weaker notion of success $L_{\mathcal{D}}(A(\mathcal{S})) \leq \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \varepsilon$ is a much more modest one

- Though, it is also a much more realistic one

We started getting an intuition as for why larger hypothesis classes $\mathcal{H}$ harder to learn

- It can be shown that all finite hypothesis classes are agnostic PAC learnable
- It can be shown that there are infinite classes that are agnostic PAC learnable
- It can be shown that there exist classes that are not agnostic PAC learnable

General loss functions

Formal model

General losses

We are interested in extending the formal learning model to a wider variety of problems

Classification (multi-class)

- The domain set $\mathcal{X} = \{x : x \in \mathcal{R}^{N_y}\}$
- The label set $\mathcal{Y} \subset \mathbb{N}_0$

The training set $\mathcal{S} = \{(x_n, y_n)\}_{n=1}^{N}$ with $x_n \in \mathcal{X}$ and $y_n \in \mathcal{Y}$

Regression

- The domain set $\mathcal{X} = \{x : x \in \mathcal{R}^{N_y}\}$
- The label set $\mathcal{Y} \subseteq \mathcal{R}$

The training set $\mathcal{S} = \{(x_n, y_n)\}_{n=1}^{N}$ with $x_n \in \mathcal{X}$ and $y_n \in \mathcal{Y}$

It is easy to convince ourselves that what makes regression and multi-class classification different from the perspective of learning is the notion of error, or failure of the learner

Though we are protected against this discrepancy, at least terminology-wise, by the notion of loss, we need a more general set to discuss these (supervised) learning tasks

- (As well as other, unsupervised, ones)

General losses | Loss functions

For discussing these learning tasks on a general setup, we introduce the set $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$

Loss functions

We let l be any function from $\mathcal{Z} \times \mathcal{H}$ to the set of non-negative reals, $l : \mathcal{Z} \times \mathcal{H} \to \mathcal{R}_+$

We define the **risk function** to be the expected loss of rule h, with respect to $\mathcal{D}$ over $\mathcal{Z}$

$$L_{\mathcal{D}}(h) \equiv \mathrm{E}_{z \sim \mathcal{D}} \left[l(h, z) \right]$$

This is the expected loss hypothesis $h \in \mathcal{H}$ over instances $z \in \mathcal{Z}$ drawn according to $\mathcal{D}$

Similarly, we define the **empirical risk** to be the expected loss over sample $\mathcal{S} = \{z_n\}_{n=1}^{N}$

$$L_{\mathcal{S}}(h) \equiv \frac{1}{|\mathcal{S}|} \sum_{n=1}^{N} l(h, z_n)$$

Because of the law of large numbers, the empirical risk tends to the true risk as $N \to \infty$

General losses | Loss functions | Classification

$$L_{\mathcal{D}}(h) \equiv \mathrm{E}_{z\sim\mathcal{D}}\left[l(h,z)\right]$$

$$L_{\mathcal{S}}(h) \equiv \frac{1}{|\mathcal{S}|}\sum_{n=1}^{N} l(h,z_n)$$

In classification, no matter whether binary of multi-class, a common way to evaluate the quality of the hypothesis function $h : \mathcal{X} \to \mathcal{Y}$ is obtained by considering a **0 − 1 loss**

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

The **loss function**

$$l_{0-1}\left(h,(x,y)\right) \equiv \begin{cases} 0, & \text{if } h(x) = y \\ 1, & \text{if } h(x) \neq y \end{cases}$$

The **risk function**

$$L_{\mathcal{D}}(h) = \mathrm{E}_{(x,y)\sim\mathcal{D}}\left[l_{0-1}\left(h,(x,y)\right)\right]$$

Because $l_{0-1}(h,z)$ is a binomial variable, we have $\mathrm{E}_{l_{0-1}(h)\sim\mathcal{D}} = \mathbb{P}_{l_{0-1}(h)\sim\mathcal{D}}\left[l_{0-1}(h)=1\right]$

$$\rightsquigarrow \quad \mathrm{E}_{(x,y)\sim\mathcal{D}}\left[l_{0-1}\left(h,(x,y)\right)\right] = \underbrace{\mathbb{P}_{(x,y)\sim\mathcal{D}}\left[h(x)\neq y\right]}_{\mathcal{D}(\{(x,y):h(x)\neq y\})}$$

General losses | Loss functions | Regression

$$L_{\mathcal{D}}(h) \equiv \mathrm{E}_{z\sim\mathcal{D}}\left[l(h,z)\right]$$

$$L_{\mathcal{S}}(h) \equiv \frac{1}{|\mathcal{S}|}\sum_{n=1}^{N} l(h,z_n)$$

In regression, a common way to evaluate the quality of the hypothesis function $h : \mathcal{X} \to \mathcal{Y}$ is obtained by the using squared difference between true and predicted labels

$$\mathcal{Y} \subseteq \mathcal{R}$$

The **loss function**

$$l_2\left(h,(x,y)\right) \equiv \left(h(x)-y\right)^2$$

The **risk function**

$$L_{\mathcal{D}}(h) = \mathrm{E}_{(x,y)\sim\mathcal{D}}\left[l_2\left(h,(x,y)\right)\right]$$

General losses | Loss functions | Unsupervised learning

$$L_{\mathcal{D}}(h) \equiv \mathrm{E}_{z\sim\mathcal{D}}\left[l(h,z)\right]$$

$$L_{\mathcal{S}}(h) \equiv \frac{1}{|\mathcal{S}|}\sum_{n=1}^{N} l(h,z_n)$$

Unsupervised tasks likes like dimension reduction, density estimation, and clustering?

In clustering, we are interested in representing a collection of unlabelled domain points $\{x\}_{n=1}^{N}$, such that $x_n \in \mathcal{X}$, with a collection $K \ll N$ code-words $\{c_k\}_{k=1}^{K}$, with $c_k \in \mathcal{X}$

$$\mathcal{X} = \mathcal{R}^{N_x}$$
$$\mathcal{Y} = \mathcal{X}$$
$$\mathcal{Z} = \mathcal{X}\times\mathcal{X}$$
$$\mathcal{H} = \text{All possible } K\text{-tuples } h = \{c_k\}$$

The **loss function**

$$l_K\left(h,(x,c)\right) \equiv \min_{1\geq k\leq K} \|c_k - x\|^2$$

The **risk function**

$$L_{\mathcal{D}}(h) = \mathrm{E}_{(x,c)\sim\mathcal{D}}\left[l_K\left(h,(x,c)\right)\right]$$

Agnostic PAC

General loss functions

General losses | Agnostic PAC

Agnostic PAC learning with general loss functions

A hypothesis class $\mathcal{H}$ is agnostic PAC learnable with respect to $\mathcal{Z}$ and a loss function $l : \mathcal{H} \times \mathcal{Z} \to \mathcal{R}_+$, if there is a function $n_{\mathcal{H}} : (0,1) \times (0,1) \to \mathbb{N}$ and a learner A such that, for the hypothesis $A(\mathcal{S})$ from $|\mathcal{S}|$ independent examples from $\mathcal{D}$, the bound holds

$$\mathbb{P}_{\mathcal{S}\sim\mathcal{D}}\left[\mathcal{S} : L_{\mathcal{D}}(A(\mathcal{S})) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \varepsilon\right] < \delta,$$

$$\begin{cases} \text{For all sample sizes } |\mathcal{S}| \geq n_{\mathcal{H}}(\varepsilon,\delta) \\ \text{For all data distributions } \mathcal{D} \\ \text{For } L_{\mathcal{D}}(h) = \mathrm{E}_{z\sim\mathcal{D}}\left[l(h,z)\right] \end{cases}$$

Agnostic PAC learnability for general loss functions is the most important concept in the statistical theory of learning, most machine learning problems are built around it

General losses | Agnostic PAC (cont.)

The definitions of agnostic PAC learnability required that the learning algorithm returns a hypothesis $h = A(S)$ from $\mathcal{H}$, but we may also require $A(S) \in \mathcal{H}$ with $\mathcal{H} \subset \mathcal{H}'$

- The loss function needs to be extended to $\mathcal{H}'$ (that is, $l : \mathcal{H}' \times \mathcal{Z} \to \mathcal{R}_+$)

Allowing $\mathcal{H} \subset \mathcal{H}'$ is often denoted to as **representation independent** or **improper** learning

Uniform convergence

General loss functions

General losses | Uniform convergence

We are interested in showing that finite hypothesis classes $\mathcal{H}$ are agnostic PAC learnable

- That is, there is at least one learning strategy that be used to learn them

$$\mathbb{P}_{\mathcal{S}\sim\mathcal{D}}\left[\mathcal{S} : L_{\mathcal{D}}(A(\mathcal{S})) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \varepsilon\right] < \delta, \quad \begin{cases}\text{For all sample sizes } |\mathcal{S}| \geq n_{\mathcal{H}}(\varepsilon,\delta)\\ \text{For all data distributions } \mathcal{D}\end{cases}$$

To prove the claim, it would suffice to show that $\text{ERM}_{\mathcal{H}}$ can learn any finite class $\mathcal{H}$

- In an PAC agnostic sense and with general loss functions
- (For concreteness, we consider binary classification)

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

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

General losses | Uniform convergence (cont.)

$$\mathbb{P}_{\mathcal{S}\sim\mathcal{D}}\left[\mathcal{S} : L_{\mathcal{D}}(h_{\mathcal{S}}) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \varepsilon\right] < \delta, \quad \begin{cases}\text{For all hypothesis } h_{\mathcal{S}} \in \mathrm{ERM}_{\mathcal{H}}(\mathcal{S}) \\ \text{For all sample sizes } |\mathcal{S}| \geq n_{\mathcal{H}}(\varepsilon,\delta) \\ \text{For all data distributions } \mathcal{D}\end{cases}$$

We are interested in showing that all the $h_{\mathcal{S}} \in \mathrm{ERM}_{\mathcal{H}}$ also minimise the true loss $L_{\mathcal{D}}$

One strategy is to show that it is true for samples such that $|L_{\mathcal{S}}(h_{\mathcal{S}}) - L_{\mathcal{D}}(h^\star)|$ is small

- That is, for samples $\mathcal{S}$ such that $h_{\mathcal{S}}$ is close to $h^\star$, the best $h \in \mathcal{H}$

Definition
A sample $\mathcal{S} \sim \mathcal{D}^N$ is said to be **ε-representative sample** of the distribution $\mathcal{D}$ over $\mathcal{Z}$, with respect to a hypothesis class $\mathcal{H}$ and a loss function $l(h, z)$, if the following holds

$$|L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \leq \varepsilon, \quad \text{for all } h \in \mathcal{H}$$

That implies that, if we draw such $\mathcal{S}$, minimising $L_{\mathcal{S}}$ also approximately minimises $L_{\mathcal{D}}$

- Then, for learnability, it only remains to guarantee that N is large enough
- (And, as always, how likely we are to be drawn one such $\mathcal{S}$ for training)

General losses | Uniform convergence (cont.)

We can prove the existence of an upper bound on the sample complexity of the $\mathrm{ERM}_{\mathcal{H}}$

- We did it for PAC learnability, we need to show it also for agnostic PAC
- $|\mathcal{S}| \geq \underbrace{\left\lceil \frac{\ln\left(|\mathcal{H}|/\delta\right)}{\varepsilon} \right\rceil}_{n_{\mathcal{H}}(\varepsilon,\delta)}$

General losses | Uniform convergence (cont.)

Theorem

It can be shown that if a sample $\mathcal{S}$ is ε-representative of $\mathcal{D}$ over $\mathcal{Z}$, with respect to class $\mathcal{H}$ and loss function $l(h, z)$, then the following bound holds for all $\mathrm{ERM}_{\mathcal{H}}(\mathcal{S})$ predictors

$$L_{\mathcal{D}}(h_{\mathcal{S}}) \leq \underbrace{\min_{h \in \mathcal{H}} L_{\mathcal{D}}(h)}_{L_{\mathcal{D}}(h^\star)} + 2\varepsilon$$

Proof

Because $\mathcal{S}$ is ε-representative, we know that $L_{\mathcal{D}}(h_{\mathcal{S}}) \leq L_{\mathcal{S}}(h_{\mathcal{S}}) + \varepsilon$ and, because we are considering the $\mathrm{ERM}_{\mathcal{H}}$ learner, we also know that $L_{\mathcal{S}}(h_{\mathcal{S}}) + \varepsilon \leq \min_{h \in \mathcal{H}} (L_{\mathcal{S}}(h) + \varepsilon)$

$$\begin{aligned} L_{\mathcal{D}}(h_{\mathcal{S}}) &\leq L_{\mathcal{S}}(h_{\mathcal{S}}) + \varepsilon \\ &\leq \underbrace{\min_{h \in \mathcal{H}} L_{\mathcal{S}}(h)}_{L_{\mathcal{S}}(h_{\mathcal{S}})} + \varepsilon \\ &\leq \underbrace{\underbrace{\min_{h \in \mathcal{H}} L_{\mathcal{D}}(h)}_{L_{\mathcal{D}}(h^\star)} + \varepsilon}_{L_{\mathcal{S}}(h_{\mathcal{S}})} + \varepsilon \end{aligned}$$

Because sample $\mathcal{S}$ is ε-representative, the last inequality is true, and this ends the proof

General losses | Uniform convergence (cont.)

$$L_{\mathcal{D}}(h_{\mathcal{S}}) \leq \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + 2\varepsilon$$

We proved that on a ε-representative sample $\mathcal{S}$, the true risk of ERM is upper bounded

To prove that the ERM can learn agnostically, we need to prove that large enough samples $\mathcal{S}$ are likely to be ε-representative (that is, with probability at least $(1-\delta)$)

Definition
We say that a hypothesis class $\mathcal{H}$ has the **uniform convergence**, with respect to $\mathcal{D}$ over $\mathcal{Z}$ and a loss function $l(h, z)$ if there exists a function $n_{\mathcal{H}}^{\mathrm{UC}} : (0,1) \times (0,1) \to \mathbb{N}$ such that samples $\mathcal{S} \sim \mathcal{D}^N$ are ε-representative with high probability (that is, at least $(1-\delta)$)

That is, the definition quantify the sample complexity needed for uniform convergence

- How large $\mathcal{S}$ must be to make it ε-representative, with probability at least $(1-\delta)$

General losses | Uniform convergence (cont.)

By combining $L_{\mathcal{D}}(h_{\mathcal{S}}) \leq \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + 2\varepsilon$ and the definition of uniform convergence with function $n_{\mathcal{H}}^{\mathrm{UC}}$, we can show that any finite class $\mathcal{H}$ is PAC learnable agnostically with sample complexity $n_{\mathcal{H}}(\varepsilon, \delta) \leq n_{\mathcal{H}}^{\mathrm{UC}}(\varepsilon/2, \delta)$ and that $\mathrm{ERM}_{\mathcal{H}}$ is a successful learner

Informally, we have the two following steps

1. For some (ε, δ), we need to determine the sample size N which guarantees with probability $(1 - \delta)$ that for a $\mathcal{S} \sim \mathcal{D}^N$, regardless of $\mathcal{D}$, all $h \in \mathcal{H}$ are such that

$$|L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \leq \varepsilon$$

   For all $h \in \mathcal{H}$, the probability of drawing a ε-representative $\mathcal{S}$ is at least $(1 - \delta)$
2. We need that for any $h \in \mathcal{H}$, $|L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)|$ is small enough if N is large enough

We followed similar steps when we discussed $\mathrm{ERM}_{\mathcal{H}}$ for finite classes under realisability

General losses | Uniform convergence | Step 1

Step 1

For some (ε, δ), we need to determine the sample size N which guarantees with probability $(1-\delta)$ that for a $\mathcal{S} \sim \mathcal{D}^N$, whatever $\mathcal{D}$, all $h \in \mathcal{H}$ with $|\mathcal{H}| < \infty$ are such that

$$|L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \leq \varepsilon$$

Equivalently, we have

$$\mathcal{D}^N(\{\mathcal{S} : \exists h \in \mathcal{H}, |L_{\mathcal{D}}(h) - L_{\mathcal{S}(h)}| \leq \varepsilon\}) \geq 1 - \delta$$

Or,

$$\mathcal{D}^N(\{\mathcal{S} : \exists h \in \mathcal{H}, |L_{\mathcal{D}}(h) - L_{\mathcal{S}(h)}| > \varepsilon\}) < \delta$$

We also have,

$$\{\mathcal{S} : \exists h \in \mathcal{H}, |L_{\mathcal{D}}(h) - L_{\mathcal{S}(h)}| > \varepsilon\} = \bigcup_{h \in \mathcal{H}} \{\mathcal{S} : |L_{\mathcal{D}}(h) - L_{\mathcal{S}(h)}| > \varepsilon\}$$

Using the union bound, we get

$$\mathcal{D}^N(\{\mathcal{S} : \exists h \in \mathcal{H}, |L_{\mathcal{D}}(h) - L_{\mathcal{S}(h)}| > \varepsilon\}) \leq \sum_{h \in \mathcal{H}} \mathcal{D}^N\{\mathcal{S} : |L_{\mathcal{D}}(h) - L_{\mathcal{S}(h)}| > \varepsilon\}$$

General losses | Uniform convergence | Step 2

$$\mathcal{D}^N(\{\mathcal{S} : \exists h \in \mathcal{H}, |L_{\mathcal{D}}(h) - L_{\mathcal{S}(h)}| > \varepsilon\}) \leq \sum_{h \in \mathcal{H}} \mathcal{D}^N\{\mathcal{S} : |L_{\mathcal{D}}(h) - L_{\mathcal{S}(h)}| > \varepsilon\}$$

Step 2

We want to show that each of the summands is small enough when N is large enough

- For a single h, the true risk $L_{\mathcal{D}}(h)$ need be ε-close to the empirical risk $L_{\mathcal{S}}(h)$
- With high probability, for samples $\mathcal{S} \sim \mathcal{D}^N$ whose size N is large enough

We have the notions of true and empirical risk, an expectation and empirical estimate

$$L_{\mathcal{D}}(h) = \mathrm{E}_{z \sim \mathcal{D}}[l(h, z)]$$

$$L_{\mathcal{S}}(h) = N^{-1} \sum_{n=1}^{N} l(h, z_n)$$

However, the expectation of the loss function $l(h, z)$ is not accessible, as $\mathcal{D}$ is unknown

- This implies that we cannot determine how far apart its sample average is

General losses | Uniform convergence | Step 2 (cont.)

$$L_{\mathcal{D}}(h) = \mathrm{E}_{z\sim\mathcal{D}}[l(h,z)]$$

$$L_{\mathcal{S}}(h) = N^{-1}\sum_{n=1}^{N} l(h,z_n)$$

We need a statistical tool, a concentration inequality, to quantify how close the expectation $L_{\mathcal{D}}(h)$ is to its empirical estimate $L_{\mathcal{S}}(h)$ is calculated using samples $\mathcal{S}\sim\mathcal{D}^N$

- The **Hoeffding's inequality** is the statistical tool for this purpose

For a single h and for $l(h,z)\in[0,1]$, we have

$$\underbrace{\mathbb{P}\left[\underbrace{|\frac{1}{N}\sum_{n=1}^{N} l(h,z_n)}_{L_{\mathcal{S}}(h)} - L_{\mathcal{D}}(h)| > \varepsilon\right]}_{\mathcal{D}^N\{\mathcal{S}:|L_{\mathcal{D}}(h)-L_{\mathcal{S}(h)}|>\varepsilon\}} \leq 2\exp\left(2N\varepsilon^2\right)$$

Over all the $h\in\mathcal{H}$, we have

$$\begin{aligned}\mathcal{D}^N(\{\mathcal{S}:\exists h\in\mathcal{H}, |L_{\mathcal{D}}(h) - L_{\mathcal{S}(h)}| > \varepsilon\}) &\leq \sum_{h\in]\mathcal{H}} 2\exp\left(2N\varepsilon^2\right)\\ &= 2|\mathcal{H}|\exp\left(2N\varepsilon^2\right)\end{aligned}$$

General losses | Uniform convergence (cont.)

By choosing for $n_{\mathcal{H}} \geq \dfrac{\log(2|\mathcal{H}|/\delta)}{2\varepsilon^2}$, we have

$$\mathcal{D}^N\left(\{\mathcal{S} : \exists h \in \mathcal{H}, |L_{\mathcal{D}}(h) - L_{\mathcal{S}(h)}| > \varepsilon\}\right) \leq \delta$$

Uniform convergence property is satisfied

$$n_{\mathcal{H}}^{\mathrm{UC}}(\varepsilon, \delta) \leq \left\lceil \frac{\log(2|\mathcal{H}|/\delta)}{2\varepsilon^2} \right\rceil$$

The sample complexity for the $\mathrm{ERM}_{\mathcal{H}}$

$$\begin{aligned} n_{\mathcal{H}}(\varepsilon, \delta) &\leq n_{\mathcal{H}}^{\mathrm{UC}}(2/\varepsilon, \delta) \\ &\leq \left\lceil \frac{2\log(2|\mathcal{H}|/\delta)}{\varepsilon^2} \right\rceil \end{aligned}$$