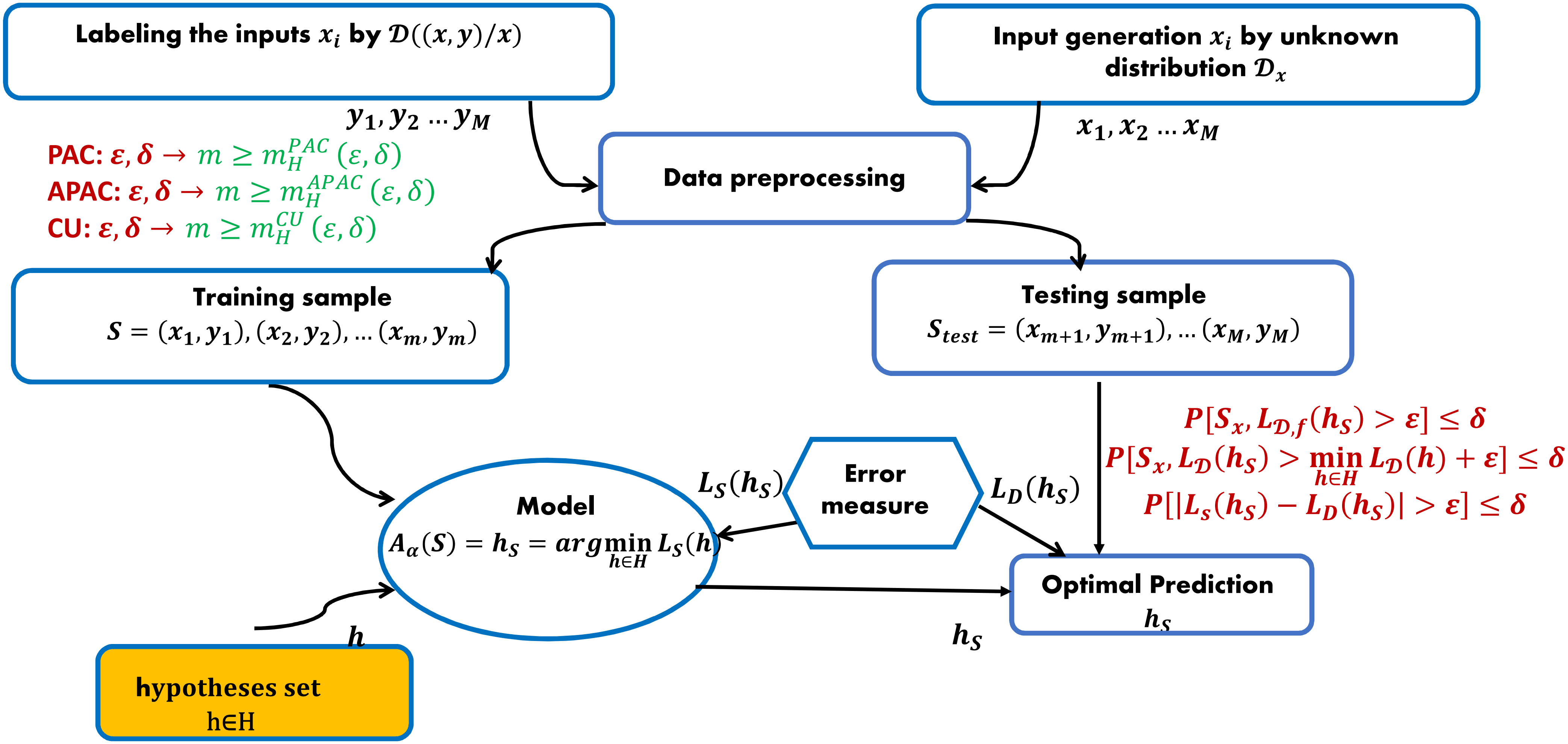
Part 1: Machine learning theory

1. Learning framework
2. Uniform convergence
3. Learnability of infinite size hypotheses set
4. Tradeoff Bias/Variance
5. Non-Uniform Learning.
    1. Non Uniform learning.
    2. Structural risk minimization.
    3. Minimum Description length
    4. Occam's Rasor
    5. Consistency

Supervised Learning Passive Offline Algorithm (SLPOA)



Labeling the inputs x_i by $\mathcal{D}((x,y)/x)$

Input generation x_i by unknown distribution $\mathcal{D}_x$

$y_1, y_2 \dots y_M$

$x_1, x_2 \dots x_M$

Data preprocessing

PAC: $\varepsilon, \delta \rightarrow m \geq m_H^{PAC}(\varepsilon, \delta)$

APAC: $\varepsilon, \delta \rightarrow m \geq m_H^{APAC}(\varepsilon, \delta)$

CU: $\varepsilon, \delta \rightarrow m \geq m_H^{CU}(\varepsilon, \delta)$

Training sample

$S = (x_1, y_1), (x_2, y_2), \dots (x_m, y_m)$

Testing sample

$S_{test} = (x_{m+1}, y_{m+1}), \dots (x_M, y_M)$

$P[S_x, L_{\mathcal{D},f}(h_S) > \varepsilon] \leq \delta$

$P[S_x, L_{\mathcal{D}}(h_S) > \min_{h \in H} L_{\mathcal{D}}(h) + \varepsilon] \leq \delta$

$P[|L_s(h_S) - L_D(h_S)| > \varepsilon] \leq \delta$

Model

$A_\alpha(S) = h_S = arg\min_{h \in H} L_S(h)$

$L_S(h_S)$

Error measure

$L_D(h_S)$

Optimal Prediction

h_S

h

h_S

hypotheses set

h∈H

Recall

Definition: Uniform Convergence

We say that H has the uniform convergence property with respect to (Z, l), if there exist:

- a function $m_H^{CU}(\varepsilon, \delta)$: $[0,1]^2 \longrightarrow \mathbb{N}$, such that: $\forall(\varepsilon, \delta) \in [0,1]^2 and \; \forall \; \mathcal{D}$ over Z.
- S is a sample of size $m \geq m_H^{CU}(\varepsilon, \delta)$, whose points are drawn $(i.i.d.)$ by $\mathcal{D}$, such that with probability of at least $(1 - \delta)$, S is ε-representative:

$$P[|L_s(h) - L_D(h)| \leq \varepsilon] \geq 1 - \delta \Leftrightarrow P[|L_s(h) - L_D(h)| > \varepsilon] \leq \delta$$

Recall

Definition: Agnostic PAC learning model

H follows agnostic PAC learning, if there exist m_H: $(0,1)^2 \rightarrow \mathbb{N}$ and A_α.

Having the following property: $\forall \varepsilon, \delta \in (0,1)$, $\forall\, \mathcal{D}$ on $X \times Y$.

Then, if we run A_α on $m \geq m_H(\varepsilon, \delta)$ generated $(i.i.d.)$ such that S is selected with a probability at least $(1-\delta)$, A_α will generate the hypothesis h_S such that:

$$L_{\mathcal{D}}(h_S) \leq \min_{h \in H} L_{\mathcal{D}}(h) + \varepsilon.$$

In other words:

$$P_{S \sim \mathcal{D}^m}\left[L_{\mathcal{D}}(h_S) > \min_{h \in H} L_{\mathcal{D}}(h) + \varepsilon\right] \leq \delta \;\; for\; all\; m \geq m_H(\varepsilon, \delta)$$

Notice:

- The generalization bound $\min\limits_{h \in H} L_D(h) + \varepsilon$ is uniform (the same) $\forall h \in H$ (complex or simple).
- The sample complexity is uniform (the same) $\forall h \in H$.

Motivation

Objectives:

In this chapter we consider more relaxed, weaker notions of learnability:

- Define the notion of Non Uniform Learnability(NUL).
- Show that NUL is a strict relaxation of APAC learnability.
- Give the characterization of Non Uniform Learnability.
- Show that the sufficient condition for NUL.
- Introduce a new learning paradigm SRM.
- Introduce MDL which is SRM specified for countable hypothesis classes.
- Introduce Consistency as an even weaker notion of learnability.

5.1. NonUniform learning

Definition : (ε, δ)-competitive

We say that h is (ε, δ)-competitive with h', if :

$$P_{S \sim D^m}[L_D(h) \leq L_D(h') + \varepsilon] \geq 1 - \delta \Leftrightarrow P_{S \sim D^m}[L_D(h) > L_D(h') + \varepsilon] \leq \delta$$

Notice:

In PAC learnability, the notion of competitiveness is not very useful, because we look for a hypothesis with an absolute low risk:

$$L_D(h_s) \leq \varepsilon$$

In APAC learnability, it is not useful too, because we look for a hypothesis with a low risk compared to the best hypothesis in H:

$$L_D(h_s) \leq \min_{h \in H} L_D(h) + \varepsilon$$

5.1. NonUniform learning

In PAC and APAC learnability, the sample size depends only on the accuracy and the confidence parameters.

In NUL learnability, we allow the sample size to be of the form:

$$m_H(\varepsilon,\delta) \longrightarrow m_H(\varepsilon,\delta,h)$$

Namely, it is nonuniform with respect to the different hypotheses with which the learner is competing.

Definition : Nonuniform Learning: NUL

H is nonuniformly learnable, if $\exists \boldsymbol{A_\alpha}$ and $m_H^{NUL}: [0,1]^2 \times H \longrightarrow \mathbb{N}$, such that $\forall \varepsilon, \delta \in [0,1], \forall h \in H$, if $m \geq m_H^{NUL}(\varepsilon,\delta,h)$, then $\forall D$ we have :

$$P_{S \sim D^m}[L_D(h_S) \leq L_D(h) + \varepsilon] \geq 1 - \delta \Leftrightarrow P_{S \sim D^m}[L_D(h_S) > L_D(h) + \varepsilon] \leq \delta$$

Notice:

NUL is a relaxation of APAC learnability.

5.1. NonUniform learning

Lemma:

If H is agnostic PAC learnable, then H is nonuniformly learnable.

(agnostic PAC learning is a special case of nonuniform learning).

Proof:

Note that:

$$L_D(h_S) \leq \min_{h \in H} L_D(h) + \varepsilon$$

So:
$(\forall h \in H)$:

$$L_D(h_S) \leq L_D(h) + \varepsilon$$

5.1. NonUniform learning

PAC learnability characterization:

A class of binary classifiers is APAC if and only if its VC dimension is finite.

Theorem 1: NUL characterization

A hypothesis class H of binary classifiers is Nonuniformly learnable if and only if it is a countable union of agnostic PAC learnable hypothesis classes:

$$H \text{ is nonuniformly } learnable \Leftrightarrow \begin{cases} H = \bigcup_{n\in\mathbb{N}} H_n \\ H_n \; is \; APAC \; learnable \end{cases}$$

Theorem 2: Sufficient condition for NUL

Let $H = \bigcup_{n\in\mathbb{N}} H_n$, such that H_n enjoys the uniform convergence property.

Then H is nonuniformly learnable.

5.1. NonUniform learning

Example: NUL doesn't imply APAC learnability

Consider a binary classification problem with the instance domain being $X = \mathbb{R}$.

For every $n \in \mathbb{N}$ let H_n be the class of polynomial classifiers of degree n; namely H_n is the set of all classifiers of the form:

$h(x) = sign(p(x))$ where $p: \mathbb{R} \rightarrow \mathbb{R}$ is a polynomial of degree n.

Let:

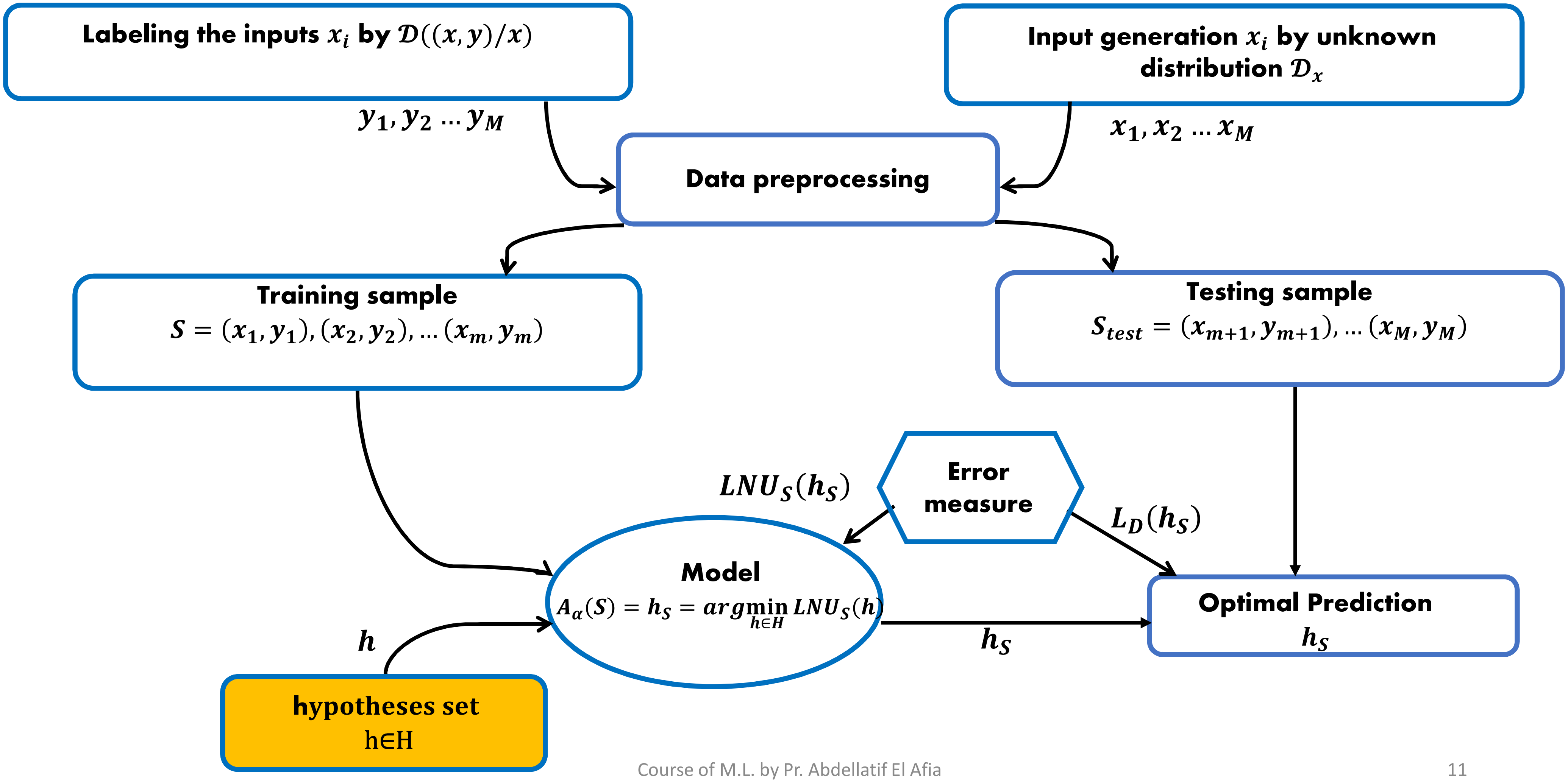
$$H = \bigcup_{n \in \mathbb{N}} H_n$$

Therefore, H is the class of all polynomial classifiers over $\mathbb{R}$,

It is easy to verify that $d_{VC}(H) = \infty$ while $d_{VC}(H_n) = n + 1$.

Hence, H is not APAC learnable, while on the basis of theorem 2, H is nonuniformly learnable.

Supervised Learning Passive Offline Algorithm (SLPOA)


Labeling the inputs x_i by $\mathcal{D}((x, y)/x)$
Input generation x_i by unknown distribution $\mathcal{D}_x$
$y_1, y_2 \dots y_M$
$x_1, x_2 \dots x_M$
Data preprocessing
Training sample
$S = (x_1, y_1), (x_2, y_2), \dots (x_m, y_m)$
Testing sample
$S_{test} = (x_{m+1}, y_{m+1}), \dots (x_M, y_M)$
$LNU_S(h_S)$
Error measure
$L_D(h_S)$
Model
$A_\alpha(S) = h_S = \underset{h \in H}{argmin}\, LNU_S(h)$
h
h_S
Optimal Prediction
h_S
hypotheses set
h∈H

5.2. Structural Risk minimization

So far, we have encoded our prior knowledge by specifying a hypothesis class H, which we believe includes a good predictor for the learning task at hand.

Another way to express our prior knowledge.

Definition : New prior knowledge New learning Paradigm

We specify preferences over hypotheses within H. We do so by first assuming that H can be written as $H = \bigcup_{n\in\mathbb{N}} H_n$ and then specifying a weight function $w: \mathbb{N} \to [0,1]$, which assigns a weight to each hypothesis class, H_n, such that a higher weight reflects a stronger preference for the hypothesis class.

Objective :

In this section we discuss how to learn with such prior knowledge.

5.2. Structural Risk Minimization

NUL context: (ERM don't accepted)

Let H be a hypothesis class that can be written as:

$$H = \bigcup_{n \in \mathbb{N}} H_n$$

Let's assume that for each $\boldsymbol{n}$, the class $\boldsymbol{H_n}$ enjoys the uniform convergence property with a sample complexity $m_{H_n}^{UC}(\varepsilon_n, \delta)$.

Definition : Accuracy function $\boldsymbol{\varepsilon_n(m, \delta)}$

Given a fixed sample size m, the function ε_n looks for the smallest value of ε that defines the lowest possible upper bound on the gap between the empirical and the general error obtained by using a sample of size m:

$$\varepsilon_n : \mathbb{N} \times [0,1] \to [0,1]$$

$$\varepsilon_n(m, \delta) = \min\{\varepsilon \in [0,1] : m_{H_n}^{UC}(\varepsilon, \delta) \leq m\}$$

5.2. Structural Risk minimization

As a result, it follows that for every m and δ, with pprobability of at least $1-\delta$ over the choice of $S \sim D^m$ we have that:

$$(\forall h \in H_n) \qquad\qquad |L_D(h) - L_S(h)| \leq \varepsilon_n(m, \delta)$$

Definition : Weight function $\boldsymbol{w(n)}$

The function w refers to the weight function over the hypothesis classes $H_1, H_2, \ldots$

It is defined as:

$$w: \mathbb{N} \longrightarrow [0,1]$$

Such that:

$$\sum_{n=1}^{\infty} w(n) \leq 1$$

This function reflects the importance that an algorithm assigns to each hypothesis or some measure of the complexity of different hypothesis classes.

5.2. Structural Risk Minimization

Theorem: The generalization bound of nonuniform learning

- **Weight function $\boldsymbol{w(n)}$** Let $w: \mathbb{N} \longrightarrow [0,1]$, such that : $\sum_{n=1}^{\infty} w(n) \leq 1$.
- $\boldsymbol{H} = \bigcup_{\boldsymbol{n} \in \mathbb{N}} \boldsymbol{H_n}$ **,** $\forall \boldsymbol{n} \in \mathbb{N}$, H_n enjoys the uniform convergence property, with a sample complexity function $m_{H_n}^{UC}$.
- $(\boldsymbol{m}, \boldsymbol{\delta})$
- **Accuracy function:** Let the $\varepsilon_n: \mathbb{N} \times [0,1] \rightarrow [0,1]$, such that :

$$\varepsilon_n(m, \delta) = \min\{\varepsilon \in [0,1]: m_{H_n}^{CU}(\varepsilon, \delta) \leq m\}.$$

Then, $\forall \delta \in [0,1]$, $\forall D$, with probability of at least $1 - \delta$ over the choice of $S \sim D^m$ the following bound holds (simultaneously) for every $n \in \mathbb{N}$ and $\boldsymbol{h} \in \boldsymbol{H_n}$:

$$P_{S \sim D^m}[|L_D(h) - L_S(h)| \leq \varepsilon_n(m, w(n). \delta)\ ] \geq 1 - \delta$$

5.2. Structural Risk minimization

This implies that, $\forall\delta \in [0,1]$, $\forall D$ with probability of at least $1-\delta$ over the choice of $S \sim D^m$ it holds that:

$\forall \boldsymbol{h} \in \boldsymbol{H}$

$$L_D(h) \leq L_S(h) + \min_{n\in\mathbb{N}:h\in H_n} \varepsilon_n(m, w(n)\delta)$$

If we denote:

$$n(h) = \min\{n: h \in H_n\}$$

The bound will become:

$$L_D(h) \leq L_S(h) + \varepsilon_{n(h)}\big(m, w\big(n(h)\big)\delta\big) = \boldsymbol{LNU_S}(\boldsymbol{h})$$

5.2. Structural Risk Minimization

SRM Algorithm

Prior knowledge:

- $H = \bigcup_{n \in \mathbb{N}} H_n$ where H_n has the uniform convergence property with sample complexity $m_{H_n}^{UC}$
- $w \colon \mathbb{N} \longrightarrow [0,1]$, such that: $\sum_{n=1}^{\infty} w(n) \leq 1$.
- (m, δ)

Define :

- $\varepsilon_n(m,\delta) = \min\{\varepsilon \in [0,1] \colon m_{H_n}^{UC}(\varepsilon,\delta) \leq m\}$
- $n(h) = \min\{n \colon h \in H_n\}$

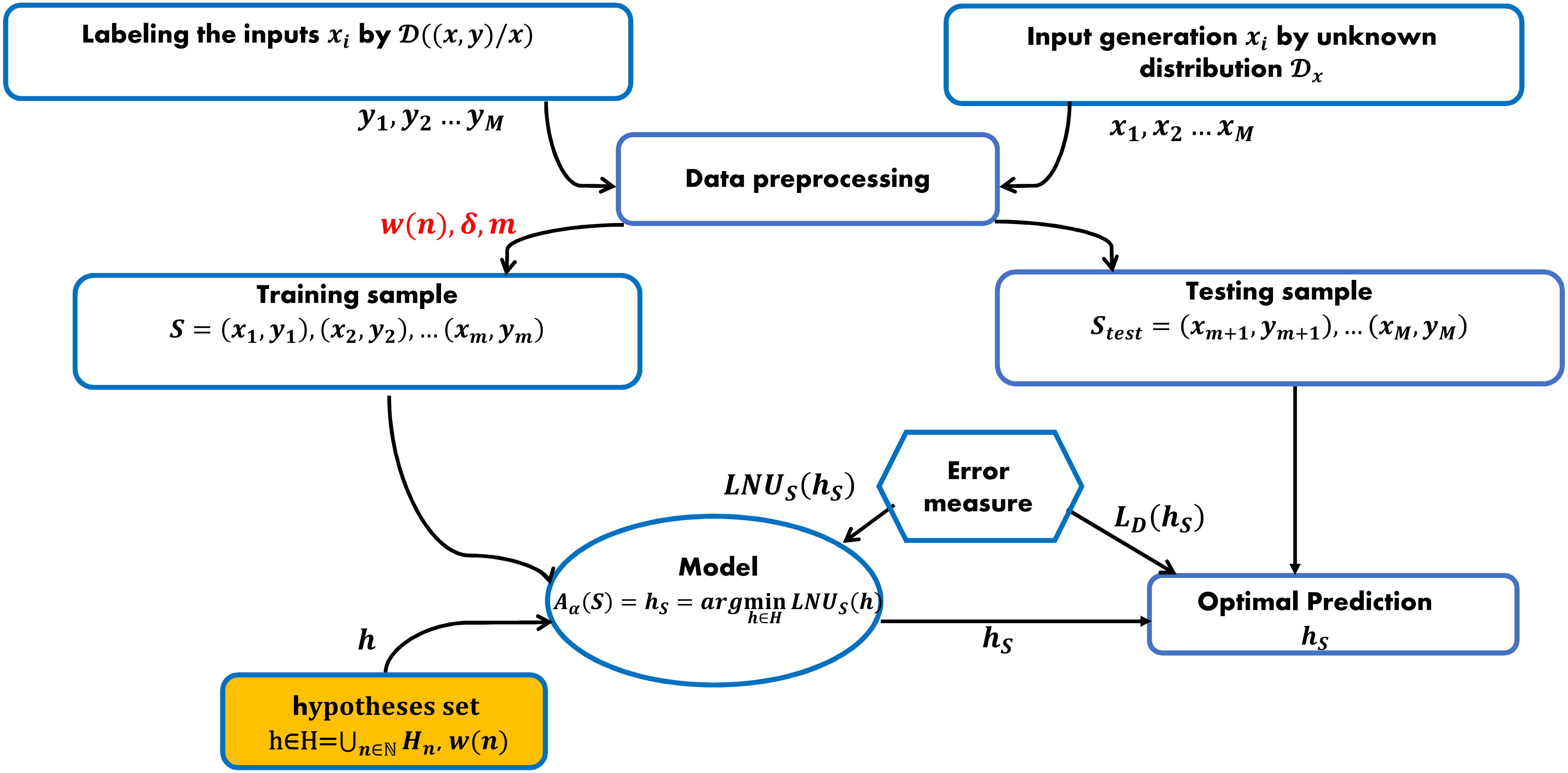
Input :

- Training set $S \rightsquigarrow D^m$ and the confidence parameter δ.

Output :

$$h_S \in \underset{h \in H}{\operatorname{argmin}}\{L_S(h) + \varepsilon_{n(h)}(m, w\big(n(h)\big).\delta)\} = \underset{h \in H}{\operatorname{argmin}}\{LNU_S(h)\}$$

Supervised Learning Passive Offline Algorithm (SLPOA)



Labeling the inputs x_i by $\mathcal{D}((x,y)/x)$

Input generation x_i by unknown distribution $\mathcal{D}_x$

$y_1, y_2 \dots y_M$

$x_1, x_2 \dots x_M$

Data preprocessing

$w(n), \delta, m$

Training sample

$S = (x_1, y_1), (x_2, y_2), \dots (x_m, y_m)$

Testing sample

$S_{test} = (x_{m+1}, y_{m+1}), \dots (x_M, y_M)$

Error measure

$LNU_S(h_S)$

$L_D(h_S)$

Model

$A_\alpha(S) = h_S = \underset{h \in H}{argmin}\, LNU_S(h)$

h

h_S

Optimal Prediction

h_S

hypotheses set

$h \in H = \bigcup_{n \in \mathbb{N}} H_n, w(n)$

5.2. Structural Risk Minimization

Theorem: Non Uniform Learning by SRM

Let H be a hypothesis class such that :

$$H = \bigcup_{n \in \mathbb{N}} H_n$$

Where each H_n enjoys the uniform convergence property with sample complexity $m_{H_n}^{UC}$.

Let $w: \mathbb{N} \longrightarrow [0,1]$, such that:

$$w(n) = \frac{6}{n^2 \pi^2}$$

Then H is uniformly learnable using SRM rule with sample complexity:

$$m_H^{NUL}(\varepsilon, \delta, h) \leq m_{H_{n(h)}}^{UC}(\frac{\varepsilon}{2}, \frac{6\delta}{(\pi . n(h))^2})$$

5.2. Structural Risk Minimization

Theorem: No-Free-Lunch for Non Uniform Learnability

For any infinite domain set X, the class of all binary valued functions over X is not a countable union of classes of finite VC dimensions.

Remark:

The NFL holds for the Non Uniform Learning as well.

Whenever the domaine set is not finite, there exists no Non Uniform Learner for binary classifiers. However, for each such classifier there exists a trivial algorithm that learns it, which is ERM with respect to the hypothesis class that contains only this classifier.

5.2. Structural Risk Minimization

Remark: NUL Vs APAC learnability

The prior knowledge of NUL is weaker than the one of APAC learning.

The cost of this weakning in this prior knowledge introduces a cost in terms of sample complexity neede to compete with any specific $h \in H_n$.

Let's consider a task of binary classification with the zero-one loss.

Assume that for all n:

$$d_{VC}(H_n) = n$$

Since:

$$m_{H_n}^{UC}(\varepsilon, \delta) = C.\frac{n + \log(1/\delta)}{\varepsilon^2}$$

Where C is a constant. So:

$$m_{H}^{NUL}(\varepsilon, \delta, h) - m_{H_n}^{UC}(\varepsilon/2, \delta) \leq 4C.\frac{2\log(2n)}{\varepsilon^2}$$

5.2. Structural Risk Minimization

Let H be a countable hypothesis class. Then, we can write H as a countable union of singleton classes, namely:

$$H = \bigcup_{n \in \mathbb{N}} \{h_n\}$$

By Hoeffding's inequality, each singleton class has the uniform convergence property with rate:

$$m^{UC}(\varepsilon, \delta) = \frac{\log(\frac{2}{\delta})}{2\varepsilon^2}$$

Therefore, we will have:

$$\varepsilon_n(m, \delta) = \sqrt{\frac{\log\left(\frac{2}{\delta}\right)}{2m}}$$

And the SRM rule becomes:

$$\underset{h \in H}{\operatorname{argmin}} \left\{ L_S(h) + \sqrt{\frac{-\log(w(h)) + \log\left(\frac{2}{\delta}\right)}{2m}} \right\}$$

5.3. Minimum Description Length

How one can describe or represent each hypothesis in the class?

We naturally fix some description language (English, programming language, some set of mathematical formulas).

Definition: Description

In any language, a description consists of finite strings of symbols (or characters) drawn from some fixed alphabet.

In this section, we will define a weight function over H based on the length of descriptions given to hypotheses.

5.3. Minimum Description Length

Let H be a hypothesis class we wish to describe. Fix some finite set Σ of symbols (or characters), which we call the alphabet.

For concreteness, we let $\Sigma = \{0,1\}$.

A string is a finite sequence of symbols from Σ. For example $\sigma = (0,1,1,1,0)$ is a string of length $|\sigma| = 5$.

Let's consider Σ^* the set of all finite length strings. $\mathrm{d}(\sigma) = 5$

Definition: Description language

A description language for H is a function $d: H \to \Sigma^*$ mapping each member h of H to a string $d(h)$.

$$\begin{cases} d: & H & \longrightarrow & \Sigma^* \\ & h & \longrightarrow & d(h) \end{cases}$$

$d(h)$ is called « the description of h » and the length of the description is denoted by $|h|$.

5.3. Minimum Description Length

Definition : Prefix-free description language

The description language is **Prefix-free**, if $\forall h, h' \in H$ such that $h \neq h'$, $d(h)$ is not a Prefix for $d(h')$.

In other words, we don't allow the string $d(h)$ to be exactly the first $|h|$ symbols of any longer string $d(h')$.

Example : $\mathbf{(1, 0) \neq (1, 1)}$

Let two different hypotheses $\boldsymbol{h}$ and $\boldsymbol{h'}$ such that :

$\boldsymbol{d(h) = (0, 1, 1)}$ **and** $\boldsymbol{d(h') = (0, 1, 1, 1, 0, 0, 0)}$

In that case, $\boldsymbol{d(h)}$ is a prefix of $\boldsymbol{d(h')}$.

Remark :

Note that the description languages should be Prefix-free. These types of functions enjoy the following combinatrial property.

5.3. Minimum Description Length

Lemma : Kraft Inequality

If $\mathcal{S} \subseteq \{0,1\}^*$ is a prefix-free set of strings, then:

$$\sum_{\sigma \in \mathcal{S}} \frac{1}{2^{|\sigma|}} \leq 1$$

Such that σ is a string.

Proof:

For each $\sigma \in \mathcal{S}$ let $P(\sigma)$ be the probability of obtaining a string σ from a process of tossing an unbiased coin. So: $P(\sigma) = \frac{1}{2^{|\sigma|}}$.

Remark:

Any prefix-free description language of a hypothesis class gives rise to a weighting function w over that hypothesis class, we will simply set:

$$(2) w(h) = \frac{1}{2^{|h|}}$$

5.3. Minimum Description Length

Theorem : Generaliztion bound using MDL

Let H be a hypothesis class, and let $d: H \rightarrow \{0,1\}^*$ be a prefix-free description language for H.

Then, for every sample size m, every confidence parameter $\delta > 0$, and every probability distribution D, with probabilit greater than $1 - \delta$ over the choice of $S \sim D^m$ we have that:

$\forall h \in H$:

$$L_D(h) \leq L_S(h) + \sqrt{\frac{|h| + \log(\frac{2}{\delta})}{2m}} \Longrightarrow MDL_S(h) = L_S(h) + \sqrt{\frac{|h| + \log(\frac{2}{\delta})}{2m}}$$

Where $|h|$ is the length of $d(h)$.

Proof:

Let's take $w(h) = \frac{1}{2^{|h|}}$, according to the previous theorem, we have $\varepsilon_n(m,\delta) = \sqrt{\frac{\ln(\frac{2}{\delta})}{2m}}$ and note that $\ln\left(2^{|h|}\right) = |h| \ln(2) < |h| \quad (|h| = -log_2\left(w(h)\right))$

- $P_{S\sim D^m}\left[L_D(h) \leq L_S(h) + \sqrt{\frac{|h|+\mathrm{lo}\ \left(\frac{2}{\delta}\right)}{2m}}\right] \geq 1-\delta$
- $P_{S\sim D^m}\left[L_D(h) > L_S(h) + \sqrt{\frac{|h|+\log\left(\frac{2}{\delta}\right)}{2m}}\right] \leq \delta$
- $w(h) = \frac{1}{2^{|h|}} \Longrightarrow |h| = -Log_2(w(h)$
- $|L_S(\boldsymbol{h}^*) - \boldsymbol{L_D}(\boldsymbol{h}^*)| \approx \mathbf{0}$
- ERM: $d_{Vc}(H) < \infty \rightarrow |L_S(\boldsymbol{h}^*) - \boldsymbol{L_D}(\boldsymbol{h}^*)| < \varepsilon(d_{Vc}(H), \mathrm{m}, \delta) \approx 0$
- $d_{Vc}(H) \approx \infty(\varepsilon(d_{Vc}(H), \mathrm{m}, \delta) \in V(0)) \rightarrow if\ H = \bigcup_{n\in\mathbb{N}} H_n\ \ such\ that$:

$$d_{Vc}(H_n) < \infty\ or\ |L_S(\boldsymbol{h}_{\boldsymbol{n}}^*) - \boldsymbol{L_D}(\boldsymbol{h}_{\boldsymbol{n}}^*)| < \boldsymbol{\varepsilon}(\boldsymbol{n})\ \forall \boldsymbol{n}\ \exists \boldsymbol{h}_{\boldsymbol{n}}^* \in H_n$$

  - SRM: $|NUL_S(\boldsymbol{h}^*) - \boldsymbol{L_D}(\boldsymbol{h}^*)| < \varepsilon$
  - MDL: $|MDL_S(\boldsymbol{h}^*) - \boldsymbol{L_D}(\boldsymbol{h}^*)| < \varepsilon$

5.3. Minimum Description Length

MDL Algorithm

Prior knowledge:

H is a countable hypothesis class.

H is described by a prefix-free language over $\{0,1\}$.

For every $h \in H$, $|h|$ is the length of the representation of h.

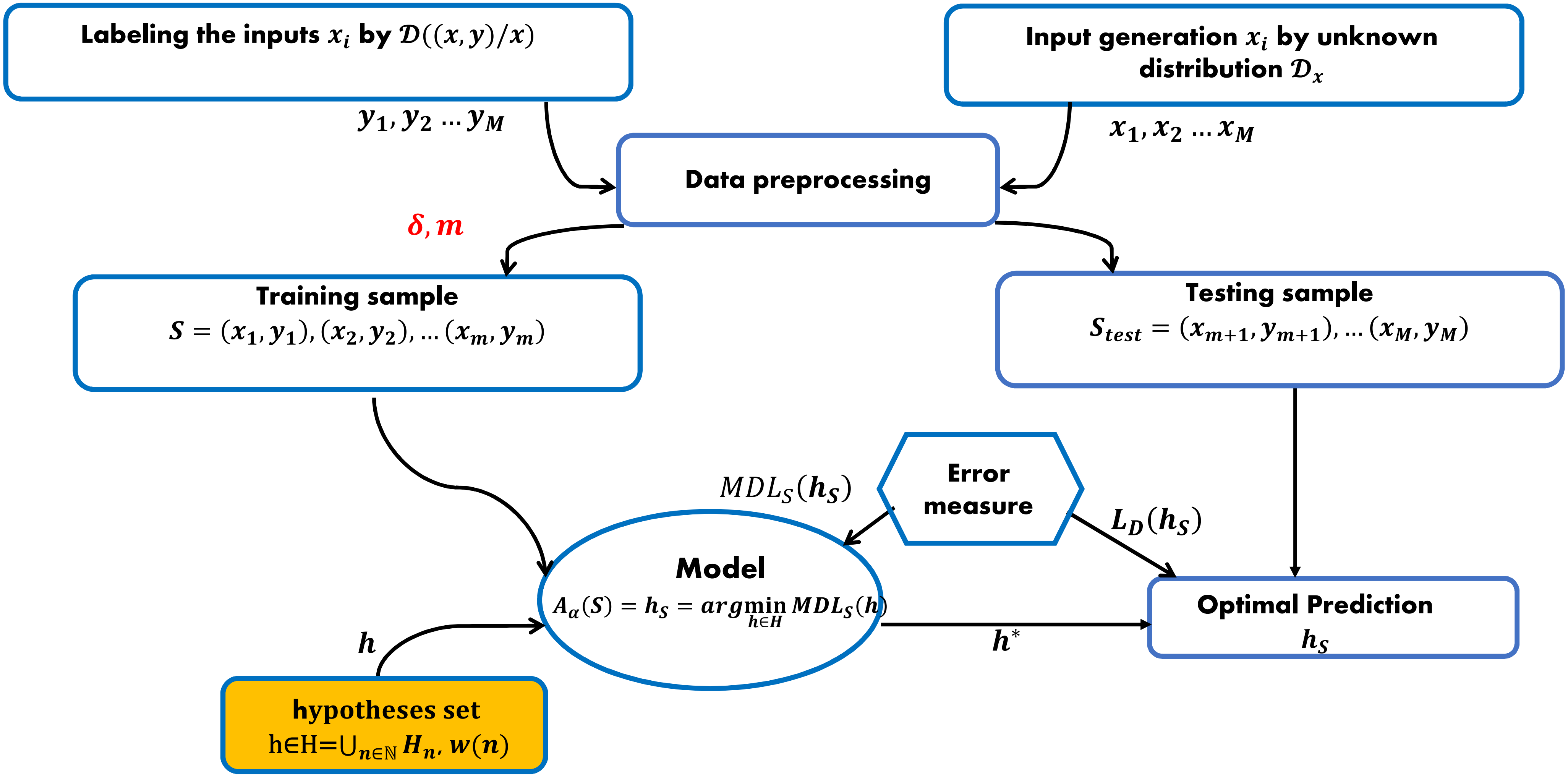
Input :

Training set $S \sim D^m$ and the confidence parameter δ.

Output :

$$h_S \in \underset{h \in H}{\operatorname{argmin}}\{L_S(h) + \sqrt{\frac{|h| + \log(\frac{2}{\delta})}{2m}}\} = \underset{h \in H}{\operatorname{argmin}}\{MDL_S(h)\}$$

Supervised Learning Passive Offline Algorithm (SLPOA)



Labeling the inputs x_i by $\mathcal{D}((x,y)/x)$

Input generation x_i by unknown distribution $\mathcal{D}_x$

$y_1, y_2 \dots y_M$

$x_1, x_2 \dots x_M$

Data preprocessing

δ, m

Training sample

$S = (x_1, y_1), (x_2, y_2), \dots (x_m, y_m)$

Testing sample

$S_{test} = (x_{m+1}, y_{m+1}), \dots (x_M, y_M)$

$MDL_S(h_S)$

Error measure

$L_D(h_S)$

Model

$A_\alpha(S) = h_S = \underset{h\in H}{argmin}\, MDL_S(h)$

h

h^*

Optimal Prediction

h_S

hypotheses set

$h \in H = \bigcup_{n\in\mathbb{N}} H_n, w(n)$

5.4. Occam’s Razor

William Ockham is a 14th-century English logician.

Claim : **Principle of Occam’s Razor**

A shorter explanation (a hypothesis that has a short length) tends to be more valid than a long explanation.

Assume that we have two hypotheses h and h' such that $|h'|$ is much smaller than $|h|$.

If both have the same error on a given training set, S, then the generalization error of h may be much higher than the generalization error of h'.

5.5. Consistency

Definition: Principle of Occam's Razor

Let Z be a domain set, let P be a set of probability distributions over Z, and let H be a hypothesis class.

A **learning rule $\boldsymbol{A_\alpha}$** is **consistent** with respect to H and P if exists a function $m_H^{CON}:[0,1]^2 \times H \times P \longrightarrow \mathbb{N}$, such that $\forall \varepsilon, \delta \in [0,1], \forall h \in H$, and for every $D \in P$, if $m \geq m_H^{CON}(\varepsilon, \delta, h, D)$, then, with probability of at least $1-\delta$ over the choice of $S \sim D^m$ we have that:

$\forall h \in H$:

$$L_D(\boldsymbol{A_\alpha}(S)) \leq L_D(h) + \varepsilon$$

Remarks:

If $\boldsymbol{P}$ is the set of all distributions, we say that $\boldsymbol{A_\alpha}$ is universally consistent with respect to H.

NUL implies Consistency.

Consistency doesn't imply NUL.

- $m_H^{CU}: [0,1]^2 \longrightarrow \mathbb{N}$, such that: $\forall \varepsilon, \delta \in [0,1]$, if $m \geq m_H^{CU}(\varepsilon, \delta)$

- $m_H^{NUL}: [0,1]^2 \times H \longrightarrow \mathbb{N}$, such that: $\forall \varepsilon, \delta \in [0,1], \forall h \in H$ if $m \geq m_H^{CON}(\varepsilon, \delta, h)$

- $m_H^{CON}: [0,1]^2 \times H \times P \longrightarrow \mathbb{N}$, such that:
- $\forall \varepsilon, \delta \in [0,1], \forall h \in H$, and $\forall D \in P$, if $m \geq m_H^{CON}(\varepsilon, \delta, h, D),$