Foundations for Machine Learning

L. Y. Stefanus

TU Dresden, June-July 2018

Slides 07

Reference

- Shai Shalev-Shwartz and Shai Ben-David. UNDERSTANDING MACHINE LEARNING: From Theory to Algorithms. Cambridge University Press, 2014.

VC-dimension

Definition

The VC-dimension of a hypothesis class H, denoted VCdim(H), is the maximal size of a set $C \subset X$ that can be shattered by H. If H can shatter sets of arbitrarily large size we say that H has infinite VC-dimension.

- Note: VC = Vapnik-Chervonenkis,

  from Vladimir Vapnik and Alexey Chervonenkis

How to compute VCdim

To show that VCdim(H) = d we need to show that

1. There exists a set C of size d that is shattered by H.
2. Every set C of size $d+1$ is not shattered by H.

Example 1

Threshold Functions

- Let H be the class of threshold functions over R. In previous example, we have shown that for an arbitrary set $C = \{c_1\}$, H shatters C; therefore $VCdim(H) \geq 1$.
- We have also shown that for an arbitrary set $C = \{c_1, c_2\}$ where $c1 \leq c2$, H does not shatter C; therefore $VCdim(H) \leq 1$.
- We conclude that $VCdim(H) = 1$.

Example 2

Axis Aligned Rectangles

- Let H be the class of axis aligned rectangles:

$$H = \{ h_{(a1,a2,b1,b2)} : a1 \le a2 \text{ and } b1 \le b2 \}$$

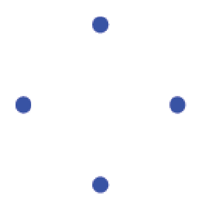
where

$$h_{(a1,a2,b1,b2)}(x, y) = \begin{cases} 1 & \text{if } a1 \le x \le a2 \text{ and } b1 \le y \le b2 \\ 0 & \text{otherwise} \end{cases}$$

- VCdim(H) = 4.

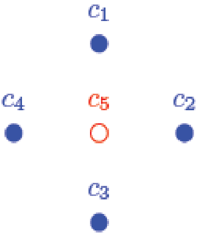
- Proof:

- We need to find a set of 4 points that are shattered by H, and show that no set of 5 points can be shattered by H.
- There is a set of 4 points that is shattered by H:



- Next consider any set C $\subset$ **R**^{2} of 5 points. In C, select a leftmost point, a rightmost point, a lowest point, and a highest point. Without loss of generality, denote C = { c1, c2, c3, c4, c5 } and let c5 be the point that was not selected.
- Now, define the labeling (1, 1, 1, 1, 0). It is impossible to obtain this labeling by an axis aligned rectangle. Indeed, such a rectangle must contain c1, c2, c3, c4; but it must also contain c5, because the coordinates of c5 are within the intervals defined by the selected points. So, C is not shattered by H, and therefore VCdim(H) = 4.

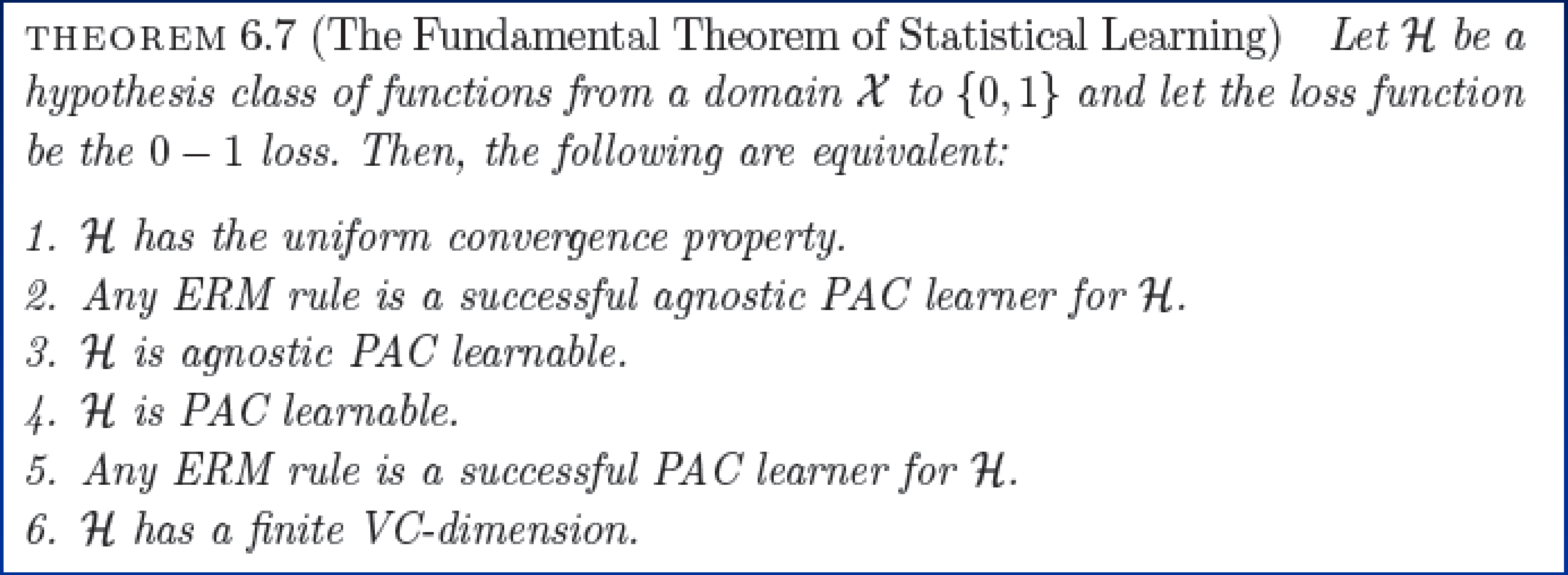
■



Example 3

Finite Classes

- Let H be a finite class.
- Then, for any set C we have $|H_C| \leq |H|$ and thus C cannot be shattered if $|H| < 2^{|C|}$, namely, if $|C| > \log_2(|H|)$.
- This implies that $\mathrm{VCdim}(H) \leq \log_2(|H|)$.
- However, the VC-dimension of a finite class H can be significantly smaller than $\log_2(|H|)$.
- For example, let $X = \{1, \ldots, k\}$, for some integer k, and consider the class of threshold functions. Then, $|H| = k$ but $\mathrm{VCdim}(H) = 1$.
- Since k can be arbitrarily large, the gap between $\log_2(|H|)$ and $\mathrm{VCdim}(H)$ can be arbitrarily large.



THEOREM 6.7 (The Fundamental Theorem of Statistical Learning) *Let $\mathcal{H}$ be a hypothesis class of functions from a domain $\mathcal{X}$ to $\{0,1\}$ and let the loss function be the $0-1$ loss. Then, the following are equivalent:*

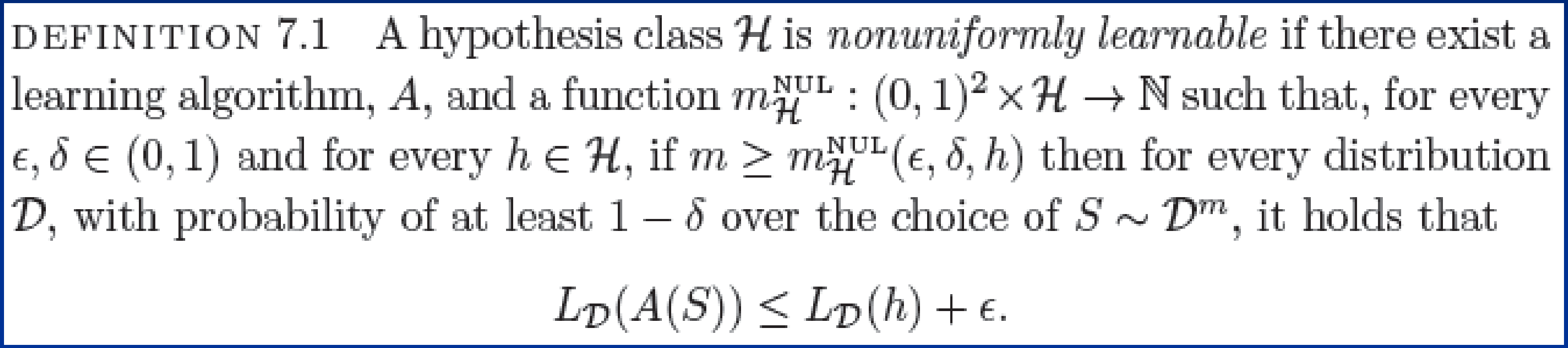
1. *$\mathcal{H}$ has the uniform convergence property.*
2. *Any ERM rule is a successful agnostic PAC learner for $\mathcal{H}$.*
3. *$\mathcal{H}$ is agnostic PAC learnable.*
4. *$\mathcal{H}$ is PAC learnable.*
5. *Any ERM rule is a successful PAC learner for $\mathcal{H}$.*
6. *$\mathcal{H}$ has a finite VC-dimension.*

Non-uniform Learnability

Non-uniform Learnability

- The notions of PAC learnability allow the sample sizes to depend on the accuracy and confidence parameters, but they are uniform with respect to the labeling rule and the underlying data distribution.
- Consequently, classes that are learnable in that respect are limited, namely, they must have a finite VC-dimension
- Next, we consider a strict relaxation of agnostic PAC learnability: non-uniform learnability, which allows the sample size to depend also on the hypothesis to which the learner is compared.

- Non-uniform learnability allows the sample size to be non-uniform with respect to the different hypotheses. It allows the sample size to be of the form $m_H(\epsilon, \delta, h)$, namely, it depends also on the hypothesis h.



DEFINITION 7.1 A hypothesis class $\mathcal{H}$ is *nonuniformly learnable* if there exist a learning algorithm, A, and a function $m^{\mathrm{NUL}}_{\mathcal{H}} : (0,1)^2 \times \mathcal{H} \to \mathbb{N}$ such that, for every $\epsilon, \delta \in (0,1)$ and for every $h \in \mathcal{H}$, if $m \geq m^{\mathrm{NUL}}_{\mathcal{H}}(\epsilon, \delta, h)$ then for every distribution $\mathcal{D}$, with probability of at least $1 - \delta$ over the choice of $S \sim \mathcal{D}^m$, it holds that

$$L_{\mathcal{D}}(A(S)) \leq L_{\mathcal{D}}(h) + \epsilon.$$

- Note that that non-uniform learnability is a relaxation of agnostic PAC learnability. That is, if a class is agnostic PAC learnable then it is also non-uniformly learnable.

Characterization of Non-uniform Learnability

Theorem 7.2

- A hypothesis class H of binary classifiers is non-uniformly learnable if and only if it is a countable union of agnostic PAC learnable hypothesis classes.

Example

- Consider a binary classification problem with the instance domain being $X = R$. For every $n \in N$ let H_n be the class of polynomial classifiers of degree n; namely, H_n is the set of all classifiers of the form $h(x) = \mathrm{sign}(p(x))$ where $p : R \to R$ is a polynomial of degree n.

$$p(x) = a_0 + a_1 x + \cdots + a_n x^n$$

- Let $H = \bigcup_{n\in\mathbb{N}} H_n$. Therefore, H is the class of all polynomial classifiers over R. VCdim(H) = ∞ while VCdim(H_n) = $n + 1$. Hence, H is not PAC learnable, but H is non-uniformly learnable.

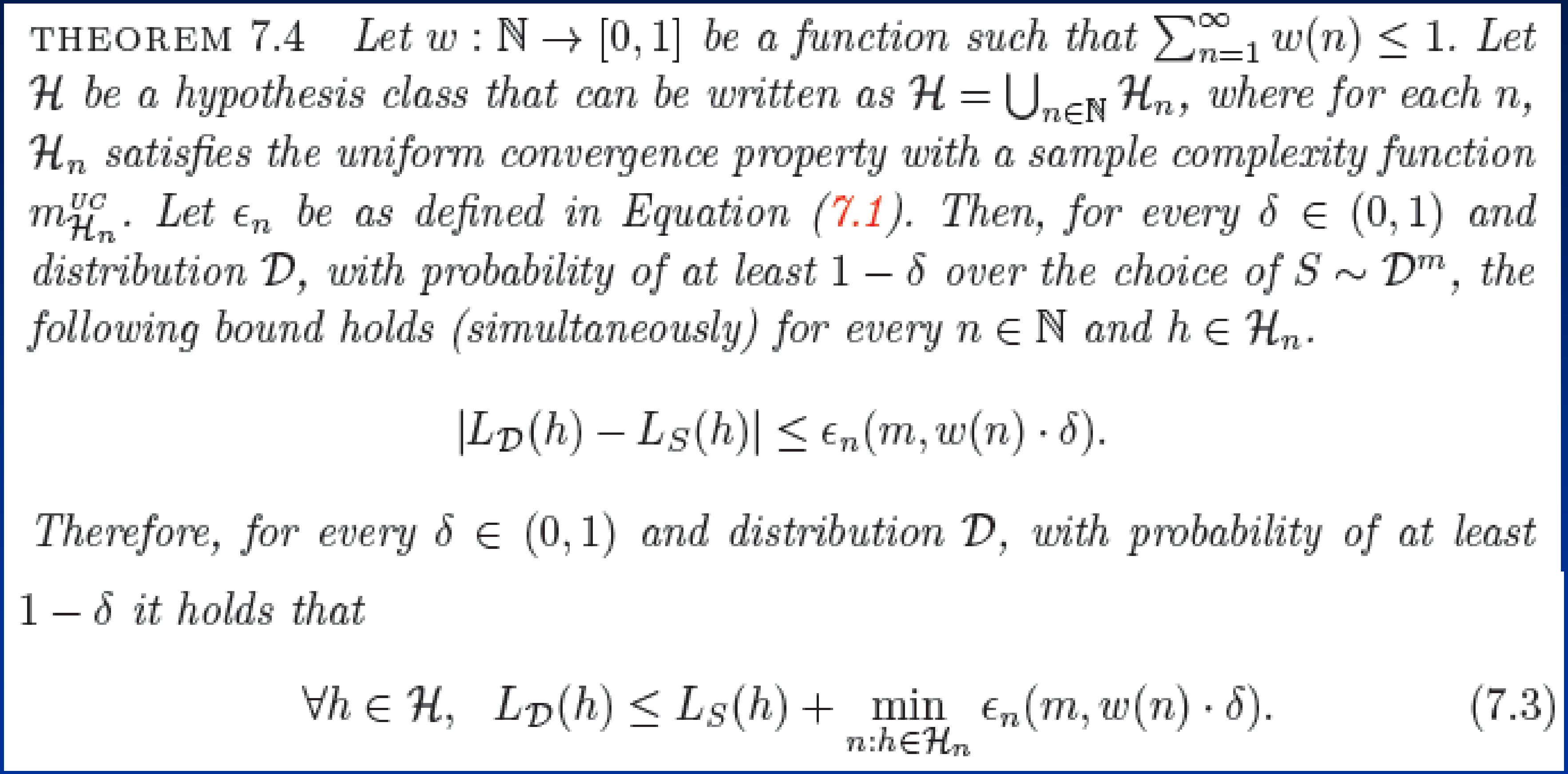
SRM (Structural Risk Minimization)

- So far, we have encoded our prior knowledge by specifying a hypothesis class H, which we believe includes a good predictor for our learning task.
- Another way to express our prior knowledge is by specifying preferences over hypotheses within H.
- In the Structural Risk Minimization (SRM) paradigm, 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, $w\colon \mathbb{N} \to [0,1]$, which assigns a weight to each hypothesis class, H_n, such that a higher weight reflects a stronger preference for the hypothesis class.

- Let *H* be a hypothesis class that can be written as $H = \bigcup_{n \in \mathbb{N}} H_n$. Assume that for each n, the class H_n has the uniform convergence property with a sample complexity function $m_{Hn}^{\text{UC}}(\epsilon, \delta)$. Let us also define the function $\epsilon_n \colon \mathbb{N} \times (0,1) \to (0,1)$ by

$$\epsilon_n(m, \delta) = \min\{\epsilon \in (0,1) \colon m_{Hn}^{\text{UC}}(\epsilon, \delta) \leq m\}. \quad (7.1)$$

- In words, we have a fixed sample size *m*, and we are interested in the lowest possible upper bound on the gap between empirical and true risks achievable by using a sample of *m* examples.
- The goal of the SRM paradigm is to find a hypothesis that minimizes a certain upper bound on the true risk. The bound that the SRM rule wishes to minimize is given in the following theorem.

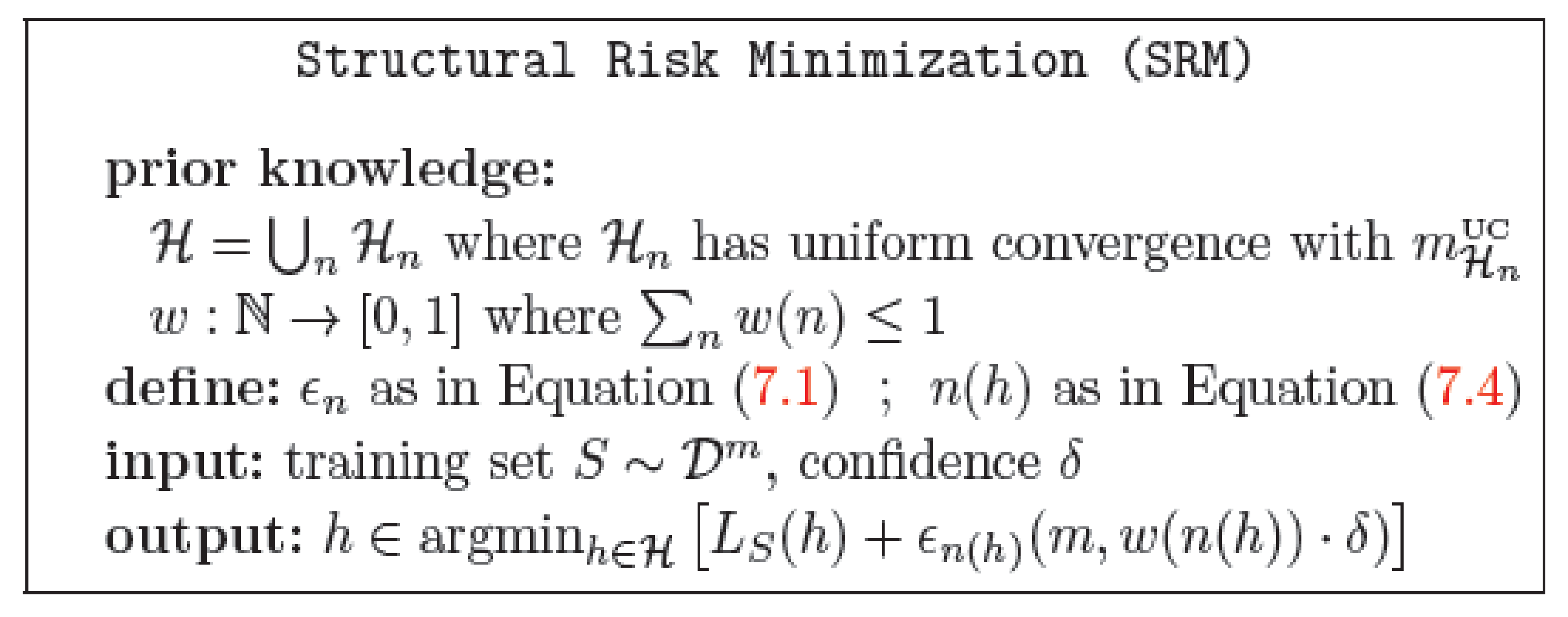


THEOREM 7.4 *Let $w : \mathbb{N} \rightarrow [0,1]$ be a function such that $\sum_{n=1}^{\infty} w(n) \leq 1$. Let $\mathcal{H}$ be a hypothesis class that can be written as $\mathcal{H} = \bigcup_{n \in \mathbb{N}} \mathcal{H}_n$, where for each n, $\mathcal{H}_n$ satisfies the uniform convergence property with a sample complexity function $m^{\mathrm{UC}}_{\mathcal{H}_n}$. Let ϵ_n be as defined in Equation (7.1). Then, for every $\delta \in (0,1)$ and distribution $\mathcal{D}$, with probability of at least $1-\delta$ over the choice of $S \sim \mathcal{D}^m$, the following bound holds (simultaneously) for every $n \in \mathbb{N}$ and $h \in \mathcal{H}_n$.*

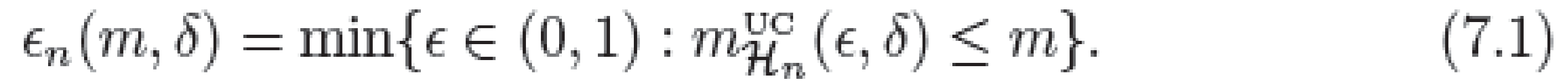
$$|L_{\mathcal{D}}(h) - L_S(h)| \leq \epsilon_n(m, w(n) \cdot \delta).$$

Therefore, for every $\delta \in (0,1)$ and distribution $\mathcal{D}$, with probability of at least $1-\delta$ it holds that

$$\forall h \in \mathcal{H}, \quad L_{\mathcal{D}}(h) \leq L_S(h) + \min_{n : h \in \mathcal{H}_n} \epsilon_n(m, w(n) \cdot \delta). \tag{7.3}$$



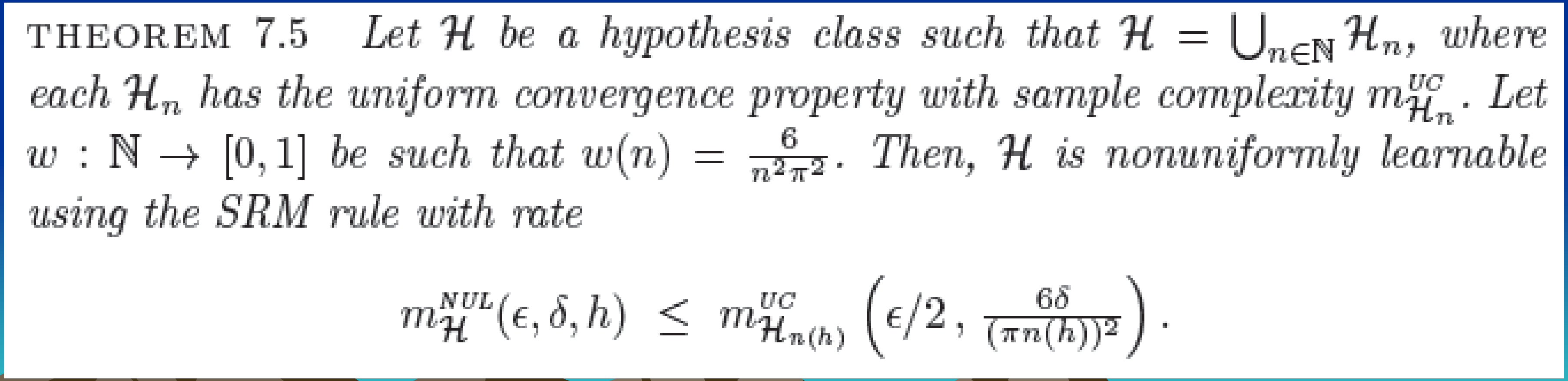
```
Structural Risk Minimization (SRM)

prior knowledge:
    $\mathcal{H} = \bigcup_n \mathcal{H}_n$ where $\mathcal{H}_n$ has uniform convergence with $m^{\mathrm{UC}}_{\mathcal{H}_n}$
    $w : \mathbb{N} \to [0,1]$ where $\sum_n w(n) \le 1$
define: $\epsilon_n$ as in Equation (7.1) ; $n(h)$ as in Equation (7.4)
input: training set $S \sim \mathcal{D}^m$, confidence $\delta$
output: $h \in \mathrm{argmin}_{h \in \mathcal{H}} \left[ L_S(h) + \epsilon_{n(h)}(m, w(n(h)) \cdot \delta) \right]$
```



$$\epsilon_n(m,\delta) = \min\{\epsilon \in (0,1) : m^{\mathrm{UC}}_{\mathcal{H}_n}(\epsilon,\delta) \le m\}. \tag{7.1}$$

$$n(h) = \min\{n : h \in \mathcal{H}_n\}, \tag{7.4}$$

- Unlike the ERM paradigm, in SRM we no longer just care about the empirical risk, $L_S(h)$, but we are willing to trade some of our bias toward low empirical risk with a bias toward classes for which $\epsilon_{n(h)}(m, w(n(h)) \cdot \delta)$ is smaller, for the sake of a smaller estimation error.
- The next theorem shows that the SRM paradigm can be used for non-uniform learning of every class, which is a countable union of uniformly converging hypothesis classes.



THEOREM 7.5 *Let $\mathcal{H}$ be a hypothesis class such that $\mathcal{H} = \bigcup_{n\in\mathbb{N}} \mathcal{H}_n$, where each $\mathcal{H}_n$ has the uniform convergence property with sample complexity $m^{UC}_{\mathcal{H}_n}$. Let $w : \mathbb{N} \to [0,1]$ be such that $w(n) = \frac{6}{n^2\pi^2}$. Then, $\mathcal{H}$ is nonuniformly learnable using the SRM rule with rate*

$$m^{NUL}_{\mathcal{H}}(\epsilon, \delta, h) \le m^{UC}_{\mathcal{H}_{n(h)}}\left(\epsilon/2\,,\,\frac{6\delta}{(\pi n(h))^2}\right).$$

No-Free-Lunch Theorem for Non-uniform Learnability

- We have learned that any countable union of classes of finite VC-dimension is non-uniformly learnable.
- It turns out that, 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-dimension.
- Therefore, the no free lunch theorem holds for non-uniform learning as well: namely, whenever the domain is not finite, there exists no universal non-uniform learner with respect to the class of all deterministic binary classifiers (although for each such classifier there exists an algorithm that learns it).

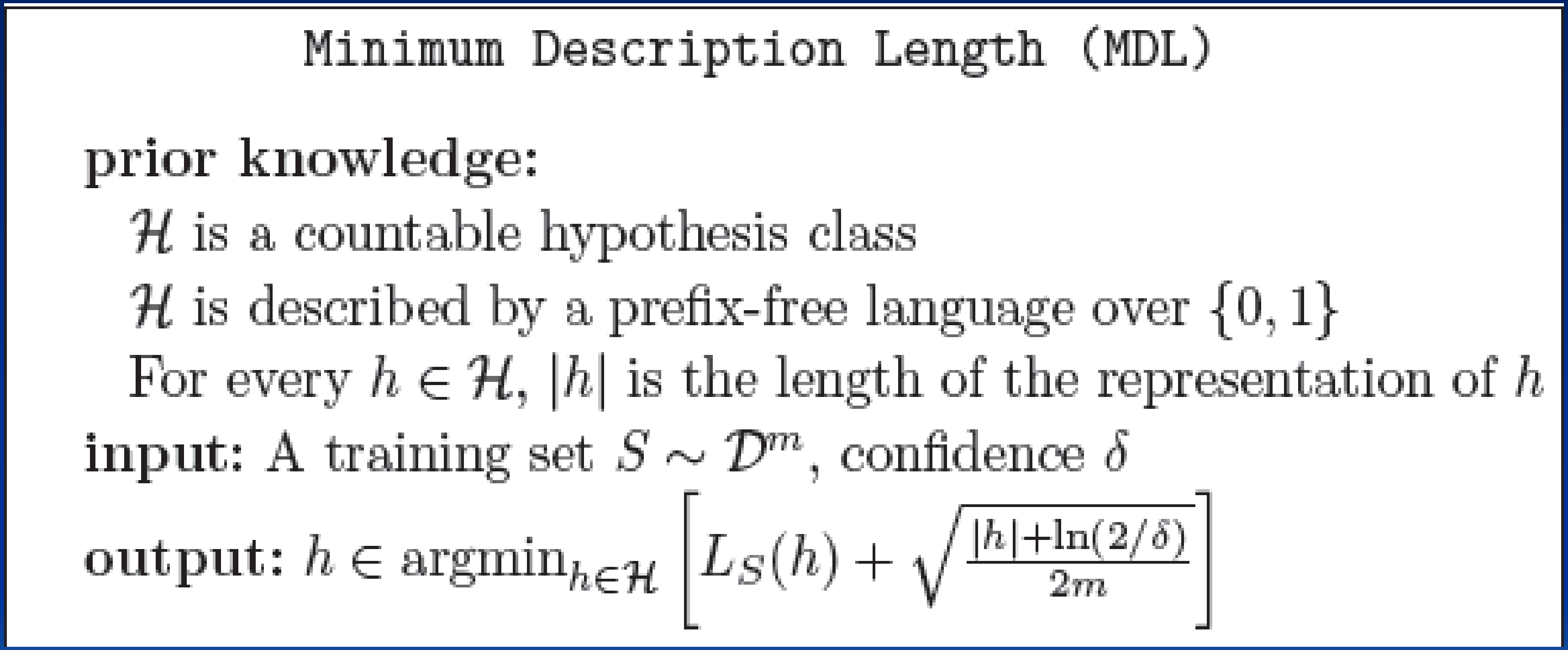
Minimum Description Length

- There is a convenient way to define a weight function w over H, which is derived from the length of descriptions given to hypotheses.
- Having a hypothesis class, one can wonder about how we describe, or represent, each hypothesis in the class.
- We fix some description language. This can be English, or a programming language, or some set of mathematical formulas.
- In any of these languages, a description consists of finite strings of symbols drawn from some fixed alphabet Σ.

Minimum Description Length

- The set of all finite length strings is denoted Σ^*.
- A description language for *H* is a function $d: H \to \Sigma^*$, mapping each member h of H to a string d(h), the description of h, and its length is denoted by |h|.
- We shall require that description languages be prefix-free; namely, for every distinct h, h', we do not allow that any string d(h) is exactly the first |h| symbols of any longer string d(h').

Minimum Description Length



Minimum Description Length (MDL)

prior knowledge:

$\mathcal{H}$ is a countable hypothesis class

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

For every $h \in \mathcal{H}$, $|h|$ is the length of the representation of h

input: A training set $S \sim \mathcal{D}^m$, confidence δ

output: $h \in \text{argmin}_{h \in \mathcal{H}} \left[L_S(h) + \sqrt{\frac{|h| + \ln(2/\delta)}{2m}} \right]$

Example of MDL

- Let H be the class of all predictors that can be implemented using some programming language, such as Python, Java or C++.
- Let us represent each program using the binary string obtained by running the gzip command on the program. This yields a prefix-free description language over the alphabet $\Sigma = \{0,1\}$. Then, |h| is simply the length (in bits) of the output of gzip when running on the program corresponding to h.

Occam’s Razor

- The MDL paradigm suggests that, having two hypotheses sharing the same empirical risk, the true risk of the one that has shorter description can be bounded by a lower value.
- This result corresponds to a philosophical message:

  A short explanation (that is, a hypothesis that has a short length) tends to be more valid than a long explanation.
- This is a well known principle, called Occam's razor, after William of Ockham, a 14th-century English logician.

Discussion : How to Learn? How to Express Prior Knowledge?

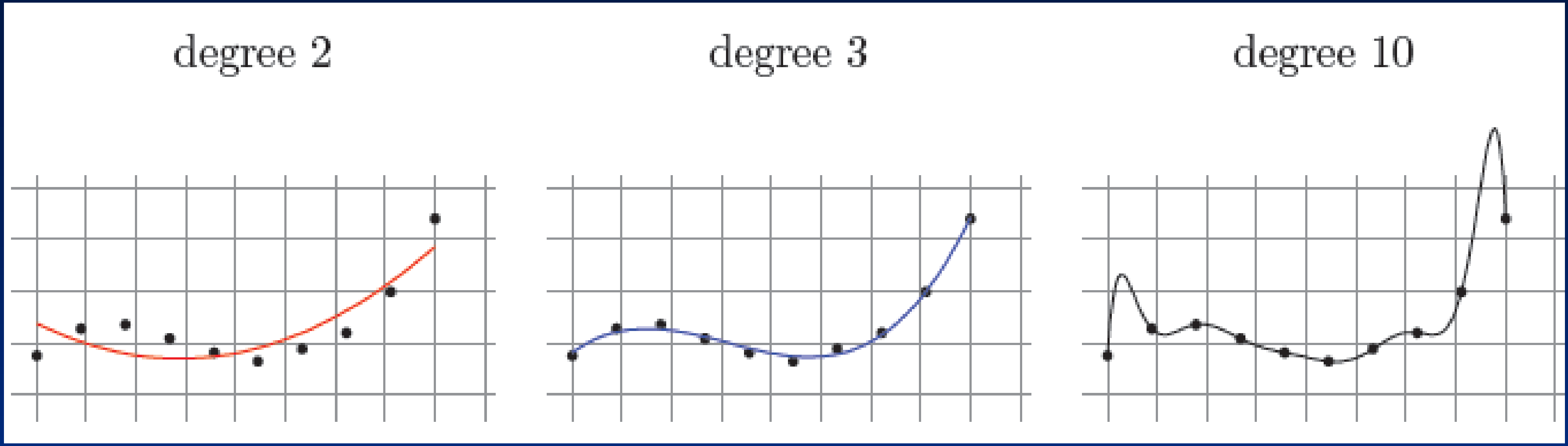
- Maybe the most useful aspect of the theory of machine learning is in providing an answer to the question of "how to learn."
- The definition of PAC learning yields the limitation of learning (via the No-Free-Lunch theorem) and the necessity of prior knowledge. It gives us a well-defined way to encode prior knowledge by choosing a hypothesis class, and once this choice is made, we have a generic learning rule, namely ERM.

Discussion : How to Learn? How to Express Prior Knowledge?

- The definition of non-uniform learnability also yields a well-defined way to encode prior knowledge by specifying weights over (subsets of) hypotheses of H. Once this choice is made, we again have a generic learning rule, namely SRM.

Discussion : How to Learn? How to Express Prior Knowledge?

- Consider the problem of fitting a one dimensional polynomial to data; namely, our goal is to learn a function, $h : R \to R$, and as prior knowledge we consider the hypothesis class of polynomials.
- However, we might be uncertain regarding which degree d would give the best results for our data set: A small degree might not fit the data well (i.e., it will have a large approximation error), whereas a high degree might lead to overfitting (i.e., it will have a large estimation error).
- The following picture shows the result of fitting a polynomial of degrees 2, 3, and 10 to the same training data set.



- Empirical risk decreases as the degree of the polynomial increases.
- Therefore, if we choose H to be the class of all polynomials up to degree 10 then the ERM rule with respect to this class would output a polynomial of degree 10 and would overfit.
- On the other hand, if we choose too small a hypothesis class, say, polynomials up to degree 2, then the ERM would suffer from underfitting (i.e., a large approximation error).
- In contrast, we can use the SRM rule on the set of all polynomials, while ordering subsets of H according to their degree, and this will yield a polynomial of degree 3 since the combination of its empirical risk and the bound on its estimation error is the smallest.

Computational Complexity

- Arguably, the class of all predictors that we can implement in a programming language such as C++ is a powerful class of functions and probably contains all that we can hope to learn in practice.
- The ability to learn this class is impressive, and, seemingly, our lectures should end here. This is not the case, because of the computational aspect of learning: that is, the runtime needed to apply the learning rule.

Computational Complexity

- For example, the implementation of the ERM paradigm w.r.t. all C++ programs of description length at most 1000 bits requires an exhaustive search over 2^{1000} hypotheses. While the sample complexity of learning this class is not too large, the runtime is $\geq 2^{1000}$. This is much larger than the number of atoms in the visible universe.
- Next we will study hypothesis classes for which the ERM or SRM schemes can be implemented efficiently.