Summary of previous content

0. The starting question (the origin of all discussion)

When can we trust that a model which performs well on training data will also perform well on real (unseen) data?

To answer this question, the discussion proceeds in the following order:

$$\text{PAC} \;\rightarrow\; \text{uniform convergence} \;\rightarrow\; \text{VC dimension} \;\rightarrow\; \text{non-uniform learning}$$

1. Structure of error concepts (basic framework)

The key error quantities with which we deal are the following:

- $L_S(h)$: training error of a fixed hypothesis h
- $L_D(h)$: true (population) error of a fixed hypothesis h (unobservable)
- $A(S)$: learning algorithm (selects a hypothesis based on data S)
- $L_S(A(S))$: the value we actually observe
- $L_D(A(S))$: the value we truly care about (but cannot observe)

The essence of the problem is:

$$\text{What we want to know: } L_D(A(S)), \qquad \text{What we can see: } L_S(A(S))$$

2. ERM and the nature of overfitting

ERM always holds:

$$L_S(A(S)) = \min_{h\in\mathcal{H}} L_S(h)$$

However, the following is generally false:

$$L_D(A(S)) = \min_{h\in\mathcal{H}} L_D(h)$$

That is,

$$\text{training-optimal} \;\neq\; \text{true-optimal}$$

and this gap is precisely what overfitting means.

3. Why is analysis difficult?

For a fixed hypothesis, the analysis is easy:

$$\Pr\big(|L_D(h) - L_S(h)| > \varepsilon\big) \;\le\; 2e^{-2m\varepsilon^2} \qquad \text{(pointwise)}$$

Learning, however, is different:

- the hypothesis is chosen *after* seeing the data,
- therefore, we need to control

$$\sup_{h \in \mathcal{H}} |L_D(h) - L_S(h)|$$

This requirement is exactly uniform convergence.

4. What PAC learning really requires

The goal of PAC learning is:

$$L_S(A(S)) \text{ is small } \Rightarrow \; L_D(A(S)) \text{ is small}$$

To achieve this, PAC requires:

$$\Pr\Big(\forall h \in \mathcal{H} : |L_D(h) - L_S(h)| \le \varepsilon\Big) \;\ge\; 1 - \delta$$

That is,

- all hypotheses must be controlled simultaneously,
- uniform convergence is essential.

5. Why VC dimension appears

The conclusion is:

$$\text{uniform convergence holds} \iff \mathrm{VCdim}(\mathcal{H}) < \infty$$

Hence,

$$\boxed{\text{PAC learnable} \iff \mathrm{VCdim}(\mathcal{H}) < \infty}$$

If the VC dimension is infinite, PAC learning is impossible.

6. The fundamental question that follows

Is uniform convergence truly necessary for learning? Do we control all hypotheses simultaneously?

This question leads to non-uniform learning.

7. The key shift in non-uniform learning

Non-uniform learning drops exactly one requirement of PAC.

PAC:

$$\exists\, m(\varepsilon, \delta) \quad \text{(common to all hypotheses)}$$

Non-uniform:

$$\forall h \in \mathcal{H},\ \exists\, m(h, \varepsilon, \delta)$$

That is,

- different hypotheses may require different sample sizes,
- uniform convergence is not required,
- pointwise control plus relative comparison is sufficient.

8. Absolute vs. relative performance

PAC learning provides absolute guarantees:

- reference: 0 or $\inf L_D$,
- a single global target,
- a common sample size.

Non-uniform learning provides relative guarantees:

- reference: a specific hypothesis h,
- comparison:
$$L_D(A(S)) \leq L_D(h) + \varepsilon,$$
- different targets for different hypotheses,
- hypothesis-dependent sample sizes.

9. What is the reference hypothesis h?

- not chosen by the learner,
- not revealed by the data,
- fixed *a posteriori* by the analyst as a benchmark.

Formally:

$$\forall h \in \mathcal{H},\ \Pr\left(L_D(A(S)) \leq L_D(h) + \varepsilon\right) \ \geq\ 1 - \delta$$

This guarantees: "the learner performs almost as well as this hypothesis."

10. Easy vs. hard hypotheses

- Easy hypotheses:
  - fast generalization,
  - small required $m(h)$.
- Hard hypotheses:
  - high overfitting risk,
  - large required $m(h)$.

"Easy" does not mean simple in form, but easy to generalize.

11. Final relationship

$$\boxed{\text{PAC learnable} \ \Rightarrow\ \text{non-uniform learnable}}$$

but

$$\text{non-uniform learnable} \ \not\Rightarrow\ \text{PAC learnable}$$

7.1 Non-uniform Learnability

Definition 7.1

(1) Intuition and purpose of the definition

non-uniform learnability:

- abandons the *uniform requirement* of PAC learning,
- allows different hypotheses to have different levels of difficulty.

(2) Precise meaning of the definition

Definition:

$$\exists A,\ \exists m_{\mathcal{H}}^{\mathrm{NUL}}(\varepsilon, \delta, h)$$

such that

$$\forall \varepsilon, \delta,\ \forall h \in \mathcal{H},\ \forall m \geq m_{\mathcal{H}}^{\mathrm{NUL}}(\varepsilon, \delta, h),$$

$$\Pr_{S \sim D^m} \Big(L_D(A(S)) \leq L_D(h) + \varepsilon \Big) \ \geq\ 1 - \delta.$$

Interpretation

- The learning algorithm A is *single and fixed*.
- The reference hypothesis h is fixed *by the analyst*, not by the learner.
- The learner only needs to perform within ε of that h.
- The required sample size may depend on h.

(3) Consequences of Definition 7.1

Non-uniform learning provides:

- relative performance guarantees,
- hypothesis-dependent sample complexity.

In particular:

- uniform convergence is *not* required,
- finite VC dimension is *not* required.

Theorem 7.1.1

(1) Main claim of the theorem

Non-uniform learning is always possible when the hypothesis class is countable.

Unlike PAC learning, which is characterized by VC dimension, non-uniform learning is characterized by the *structural size* of the hypothesis class.

(2) Statement (informal)

For binary classification, a hypothesis class $\mathcal{H}$ is non-uniformly learnable if and only if

- $\mathcal{H}$ is countable, or
- each hypothesis can be assigned a finite complexity measure (e.g., description length or complexity penalty).

(3) Why this works (key ideas)

Idea 1: Indexing hypotheses

Enumerate the hypothesis class:

$$\mathcal{H} = \{h_1, h_2, h_3, \dots\},$$

and assign each hypothesis a complexity value $c(h_i)$.

Idea 2: Distributing the union bound

Uniform control requires:

$$\Pr\left(\exists h \in \mathcal{H} : \text{bad event}\right) \;\leq\; |\mathcal{H}| \cdot (\cdot),$$

which explodes when $\mathcal{H}$ is infinite.

Non-uniform control instead uses:

$$\Pr(\text{bad event for fixed } h_i) \leq \delta_i,$$

with

$$\sum_i \delta_i \leq \delta.$$

Thus, hypotheses are controlled *one at a time*, rather than all at once.

(4) Conclusion of Theorem 7.1.1

- Non-uniform learning is possible even when VC dimension is infinite.
- Some hypothesis classes are non-uniformly learnable but not PAC learnable.

$$\boxed{\text{PAC} \subsetneq \text{Non-uniform}}$$

compare between PAC VS Non-uniform learnability

1. What uniform (PAC) learning requires

PAC learning (uniform convergence) always requires:

$$\Pr\left(\sup_{h\in\mathcal{H}} |L_D(h) - L_S(h)| \le \varepsilon\right) \ \ge\ 1-\delta.$$

Meaning

- Regardless of realizable or agnostic setting,
- not because of a single reference hypothesis,
- but because *all hypotheses* $h \in \mathcal{H}$
- must simultaneously generalize well.

Therefore:

- a union bound is required,
- finite VC dimension is required.

Non-uniform learning relaxes the requirement to the following:

$$\forall h \in \mathcal{H},\ \exists m(\varepsilon,\delta,h) \text{ s.t. } \Pr\left(L_D(A(S)) \le L_D(h)+\varepsilon\right) \ge 1-\delta.$$

Key difference

- $\times$ controlling all hypotheses simultaneously,
- ✓ controlling only one fixed reference hypothesis h.

As a result:

- the sample size may depend on h,
- uniform convergence is not required.

Theorem 7.2

A hypothesis class $\mathcal{H}$ is non-uniformly learnable if and only if it can be written as a countable union of agnostic PAC-learnable classes.

$$\mathcal{H} = \bigcup_{n=1}^{\infty} \mathcal{H}_n, \qquad \mathcal{H}_n \text{ is agnostic PAC learnable.}$$

Interpretation

- The entire class need not be PAC-learnable at once.
- It suffices that the class can be decomposed into "progressively harder but individually learnable" subclasses.

Theorem 7.3

If $\mathcal{H} = \bigcup_n \mathcal{H}_n$ and each $\mathcal{H}_n$ satisfies uniform convergence, then $\mathcal{H}$ is non-uniformly learnable.

That is,

$$\text{uniform convergence (locally)} \;\Rightarrow\; \text{non-uniform learnability (globally)}.$$

VC dimension questions arising here

(A) Does finite VC dimension imply uniform convergence?

Yes. Finite VC dimension guarantees uniform convergence via union bounds.

(B) If each $\mathcal{H}_n$ has finite VC dimension, is the union finite?

No.

$$\mathrm{VCdim}\Big(\bigcup_n \mathcal{H}_n\Big) \;\neq\; \sup_n \mathrm{VCdim}(\mathcal{H}_n) \quad \text{in general.}$$

- If there is a common upper bound, the VC dimension is finite.
- Without such a bound, it can be infinite.

Key Example 1 (The interval class with VC dimension $2n$)

Hypothesis class

$$\mathcal{H}_n = \{\text{unions of at most } n \text{ intervals on the real line}\}.$$

(1) Why $\mathrm{VCdim}(\mathcal{H}_n) = 2n$?

Key observation

- One interval corresponds to one contiguous block of label 1.
- n intervals can represent at most n such blocks.

Lower bound: why $2n$ points can be shattered Consider $2n$ ordered points:

$$x_1 < x_2 < \cdots < x_{2n}.$$

Assign the alternating labeling:

$$1, 0, 1, 0, \ldots, 1, 0.$$

- The number of 1-blocks is exactly n.
- Each block can be covered by one interval.

Hence, shattering is possible.

Upper bound: why $2n+1$ points cannot be shattered Consider $2n+1$ points with labeling:

$$1, 0, 1, 0, \ldots, 1.$$

- The number of 1-blocks is $n+1$.
- n intervals are insufficient.

Thus, shattering fails.

Conclusion

$$\boxed{\mathrm{VCdim}(\mathcal{H}_n) = 2n.}$$

(2) Why does the union have infinite VC dimension?

Let

$$\mathcal{H} = \bigcup_{n=1}^{\infty} \mathcal{H}_n.$$

- Given any k points,
- choose $n = \lceil k/2 \rceil$,

- then $\mathcal{H}_n \subset \mathcal{H}$ can shatter them.

Therefore,

$$\mathrm{VCdim}(\mathcal{H}) = \infty.$$

Each subclass has finite VC dimension, but there is no uniform bound.

Key Example II 2 (Polynomial classifiers (Example 7.1))

Hypothesis class

$$\mathcal{H}_n = \{\mathrm{sign}(p(x)) : \deg p \le n\}, \qquad \mathcal{H} = \bigcup_n \mathcal{H}_n.$$

Facts

- $\mathrm{VCdim}(\mathcal{H}_n) = n + 1$,
- $\mathrm{VCdim}(\mathcal{H}) = \infty$.

Hence, $\mathcal{H}$ is not PAC learnable. However, this non-uniform learning still works. Suppose the true target is

$$h^*(x) = \mathrm{sign}(x^3 - x), \qquad h^* \in \mathcal{H}_3.$$

Non-uniform analysis

1. Fix the reference hypothesis h^*.
2. Since $h^* \in \mathcal{H}_3$,
3. and $\mathcal{H}_3$ has finite VC dimension,
4. uniform convergence holds within $\mathcal{H}_3$.

Thus, with sufficient samples,

$$L_D(A(S)) \le L_D(h^*) + \varepsilon.$$

This satisfies Definition 7.1.

- The degree 3 need not be known.
- The algorithm need not explicitly search by degree.
- Only the analysis fixes the reference hypothesis.

- Uniform learning:
  - guarantees hold for all h simultaneously.
- Non-uniform learning:
  - guarantees hold only for a fixed reference hypothesis h.

7.2 Structural Risk Minimization

$$\mathcal{H} \text{ is non-uniformly learnable} \iff \mathcal{H} = \bigcup_{n=1}^{\infty} \mathcal{H}_n, \quad \mathcal{H}_n \text{ is agnostic PAC learnable}$$

Interpretation:

- the entire hypothesis class cannot be controlled at once,
- but can be decomposed into controllable substructures,
- the true target hypothesis belongs to one of these structures.

This is an **existence theorem**.

From Uniform Convergence to $\varepsilon_n(m, \delta)$

Each structure $\mathcal{H}_n$ satisfies uniform convergence with sample complexity

$$m^{UC}_{\mathcal{H}_n}(\varepsilon, \delta).$$

Define its inverse:

$$\boxed{\varepsilon_n(m, \delta) = \min \left\{ \varepsilon : m^{UC}_{\mathcal{H}_n}(\varepsilon, \delta) \le m \right\}} \tag{7.1}$$

Precise Meaning

> The tightest generalization error bound that is already guaranteed for structure $\mathcal{H}_n$ given m samples.

- Not an asymptotic limit,
- Not a future guarantee,
- A **present-time** bound.

Why Inequality (7.2) Holds Automatically

By the definition of uniform convergence and ε_n:

$$\Pr\Big(\forall h \in \mathcal{H}_n,\ |L_D(h) - L_S(h)| \le \varepsilon_n(m,\delta)\Big) \ge 1 - \delta$$

Suppressing probability notation:

$$\forall h \in \mathcal{H}_n, \quad |L_D(h) - L_S(h)| \le \varepsilon_n(m,\delta) \tag{7.2}$$

This follows directly from the definition and is not a new theorem.

Key Properties of ε_n

(1) Increasing Structural Complexity

$$n\uparrow \quad \Rightarrow \quad \varepsilon_n(m,\delta)\uparrow$$

More complex structures yield looser guarantees.

(2) Increasing Sample Size

$$m\uparrow \quad \Rightarrow \quad \varepsilon_n(m,\delta)\downarrow$$

More data yields tighter guarantees.

(3) Is It Always the Tightest?

- Yes, within a fixed structure $\mathcal{H}_n$,
- No, across different structures.

The Role of the Weight Function $w(n)$

$$w : \mathbb{N} \to [0,1], \quad \sum_{n=1}^{\infty} w(n) \le 1$$

Common Misconceptions

- Not a probability distribution,
- Not related to data generation,
- Not a posterior.

Correct Interpretation

> A bookkeeping device for allocating failure probabilities across structures.

Each structure $\mathcal{H}_n$ receives failure probability $w(n)\delta$, and the union bound ensures total failure probability is at most δ.

What SRM Actually Minimizes

SRM does **not** minimize true risk directly.

Instead, it minimizes the bound:

$$\boxed{L_S(h) + \varepsilon_n\big(m, w(n)\delta\big)} \qquad (h \in \mathcal{H}_n)$$

- first term: data fit,
- second term: structural reliability penalty.

fit + trust

Summary Comparison

	PAC / Uniform	Non-uniform + SRM
Guarantee Target	All hypotheses	Fixed structure
Uniform Convergence	Required	Not required
VC Dimension	Must be finite	Can be infinite
Meaning of ε	Preset target	Resulting bound
Primary Goal	Learnability	Model selection

Theorem 7.4

1. Why this theorem is needed

- The full hypothesis space $\mathcal{H}$ has infinite VC dimension, so uniform convergence over $\mathcal{H}$ is impossible.
- Therefore, plain ERM
$$\arg\min_h L_S(h)$$
is prone to overfitting.

At the same time, assume:

$$\mathcal{H} = \bigcup_{n=1}^{\infty} \mathcal{H}_n,$$

where:

- each $\mathcal{H}_n$ has finite VC dimension,
- each $\mathcal{H}_n$ satisfies uniform convergence,
- but the convergence rates ε_n differ across structures.

The problem is therefore:

> "Each structure can be controlled individually, but how can we control *all of them simultaneously*?"

2. Core setup of Theorem 7.4

(1) Structure-wise generalization bounds

For each structure $\mathcal{H}_n$,

$$\forall h \in \mathcal{H}_n : \quad |L_D(h) - L_S(h)| \le \varepsilon_n(m, \delta_n) \quad \text{with probability } \ge 1 - \delta_n.$$

(2) Allocation of failure probabilities

$$\delta_n := w(n)\delta, \qquad \sum_{n=1}^{\infty} w(n) \le 1.$$

Important clarifications:

- $w(n)$ is *not* a probability,
- it has nothing to do with data generation,
- it is purely an analysis device for managing infinitely many events.

3. First conclusion of Theorem 7.4 (probabilistic)

With probability at least $1-\delta$, for all structures $\mathcal{H}_n$ and all $h \in \mathcal{H}_n$, the corresponding structure-wise generalization bounds hold simultaneously.

$$\Pr\Big(\forall n,\ \forall h \in \mathcal{H}_n :\ |L_D(h) - L_S(h)| \le \varepsilon_n(m, w(n)\delta)\Big) \ge 1 - \delta.$$

Meaning:

- even infinitely many structures can be controlled,
- by distributing failure probabilities,
- all bounds hold at once with high probability.

This probabilistically legitimizes non-uniform learning.

4. Second conclusion: a bound for any hypothesis

On the same high-probability event, for any $h \in \mathcal{H}$:

$$L_D(h) \le L_S(h) + \min_{n:\, h \in \mathcal{H}_n} \varepsilon_n(m, w(n)\delta). \tag{7.3}$$

Precise meaning of (7.3)

- A hypothesis may belong to multiple structures.
- The tightest applicable bound is chosen automatically.
- No structure needs to be selected in advance.

That is:

> "Regardless of where a hypothesis comes from, it is evaluated using the most trustworthy structure available."

5. Why SRM follows naturally

Since $L_D(h)$ is unobservable, the rational strategy is to minimize a bound that always holds:

$$\arg\min_{h \in \mathcal{H}} \Big[L_S(h) + \varepsilon_{n(h)}\big(m, w(n(h))\delta\big) \Big].$$

Here:

- $L_S(h)$ measures data fit,
- $\varepsilon_{n(h)}$ penalizes structural complexity,
- $n(h)$ denotes the simplest structure containing h.

This is exactly Structural Risk Minimization.

6. Connecting ERM, uniform learning, non-uniform learning, and SRM

Uniform learning with finite VC dimension

- a single global ε,
- no distinction between structures,
- ERM is sufficient.

Non-uniform learning with infinite VC dimension

- different structures converge at different rates,
- ERM is unsafe,
- Theorem 7.4 guarantees all structure-wise bounds simultaneously,
- SRM emerges as the only principled selection rule.

7. The true role of Theorem 7.4

- not a new algorithm,
- not a heuristic regularization trick,
- but a probabilistic bridge from non-uniform learnability to a concrete model-selection principle.

It provides the rigorous justification for adding a structure-dependent complexity term.

Theorem 7.5

1. Why this theorem is needed

Consider the hypothesis space

$$\mathcal{H} = \bigcup_{n=1}^{\infty} \mathcal{H}_n.$$

- $\mathrm{VCdim}(\mathcal{H}) = \infty \Rightarrow$ uniform PAC learning is impossible.
- Each structure $\mathcal{H}_n$ has finite VC dimension $\Rightarrow$ uniform convergence holds within each structure.

The remaining question is:

> "Even if each structure generalizes well, who guarantees that the hypothesis chosen by SRM actually learns?"

Theorem 7.4 only constructed bounds. Theorem 7.5 fills this gap by proving learnability.

2. Ingredients used in Theorem 7.5

(1) A fact obtained from Theorem 7.4

With probability at least $1-\delta$, simultaneously:

$$\forall h' \in \mathcal{H}: \quad L_D(h') \leq L_S(h') + \varepsilon_{n(h')}\big(m, w(n(h'))\delta\big).$$

(2) Definition of the SRM algorithm

$$A(S) \in \arg\min_{h' \in \mathcal{H}} \Big[L_S(h') + \varepsilon_{n(h')}\big(m, w(n(h'))\delta\big)\Big].$$

3. What Theorem 7.5 proves

Core conclusion

For any comparison hypothesis $h \in \mathcal{H}$, and for sufficiently large sample size m,

$$\boxed{L_D(A(S)) \leq L_D(h) + \varepsilon} \qquad \text{with probability } \geq 1-\delta.$$

Interpretation

- The hypothesis selected by SRM
- is never worse than any reference hypothesis h
- by more than ε in terms of true risk.

In particular, choosing:

- $h = h^*$ (an optimal hypothesis), or
- h belonging to a simple structure,

yields a concrete performance guarantee.

4. Explicit connection to non-uniform learnability

Theorem 7.5 shows that

$$m_{\mathcal{H}}^{\mathrm{NUL}}(\varepsilon, \delta, h) \;\le\; m_{\mathcal{H}_{n(h)}}^{\mathrm{UC}}\Big(\varepsilon/2,\; w(n(h))\delta\Big).$$

Meaning:

- if h lies in a simple structure, fewer samples suffice,
- if h lies in a complex structure, more samples are needed,
- hypothesis-dependent sample complexity is allowed.

This is exactly the definition of non-uniform PAC learnability.

5. Logical structure of the proof (at a glance)

$$\begin{gathered}
\text{(7.4) Simultaneous generalization bounds for all } h \\
\Downarrow \\
\text{SRM selects } h \text{ minimizing the bound} \\
\Downarrow \\
\text{Bound of } A(S) \text{ is no worse than bound of any } h \\
\Downarrow \\
\epsilon/2 + \epsilon/2 \text{ decomposition} \\
\Downarrow \\
L_D(A(S)) \le L_D(h) + \epsilon
\end{gathered}$$

6. Relationship between Theorem 7.4 and Theorem 7.5

	Theorem 7.4	Theorem 7.5
Nature	Probabilistic tool	Learnability theorem
Role	Construct bounds	Guarantee SRM performance
Algorithm	None	SRM explicitly defined
Conclusion	“Bounds exist”	“Learning succeeds”

7. Common misconceptions

- "SRM finds the optimal hypothesis" — False. It tracks the optimal one within ε.
- "This proves uniform PAC learning" — False. The result is non-uniform PAC.
- "This is a structure selection theorem" — False. It is a justification of model selection.

7.3 Minimum Description Length and Occam's Razor

1. Why Section 7.3 is needed

From the previous results (Theorems 7.4 and 7.5), we already know that:

- if the hypothesis space $\mathcal{H}$ is countable,
- and if weights $w(h)$ satisfy $\sum_h w(h) \leq 1$,

then Structural Risk Minimization yields non-uniform PAC learning.
However, a crucial question remains:

> "How should the weights $w(h)$ be chosen in practice? Why should some hypotheses be trusted more than others?"

The unique mathematically clean answer is: *description length*.

2. Core idea: hypotheses as strings

(1) Describing a hypothesis

A hypothesis can be represented as a finite binary string in some description language (English, formulas, programs, etc.).

Formally, define a description map:

$$d : \mathcal{H} \to \{0,1\}^*.$$

- $d(h)$: the description of hypothesis h,
- $|h| := |d(h)|$: the description length (number of bits).

(2) The key requirement: prefix-free

The description language must be *prefix-free*:

- for any distinct $h \neq h'$,
- $d(h)$ is not a prefix of $d(h')$.

Without this condition:

- description length would not correspond to information content,
- description-based weights could not be treated like probabilities.

Theorem 7.6

Statement (meaning-centered)

Given a prefix-free description language, the following choice of weights is always valid:

$$\boxed{w(h) = 2^{-|h|}}$$

and it satisfies:

$$\sum_{h \in \mathcal{H}} w(h) = \sum_{h \in \mathcal{H}} 2^{-|h|} \leq 1.$$

Why this holds

- prefix-free descriptions imply the Kraft inequality,
- the Kraft inequality guarantees the sum is at most 1.

Thus, description lengths can be used as probability mass without mathematical inconsistency.

Role of Theorem 7.6

- It formally justifies using description length as weights.
- Without it, MDL would remain a heuristic.

Theorem 7.7

Now all ingredients are available:

- the general SRM bound (Theorem 7.4),
- the validity of description-length weights (Theorem 7.6).

Statement

Assume:

- $\mathcal{H}$ is countable,
- $d : \mathcal{H} \to \{0,1\}^*$ is prefix-free,
- $|h|$ denotes description length.

Then for any distribution D and any $m, \delta > 0$, with probability at least $1 - \delta$, the following holds simultaneously for all $h \in \mathcal{H}$:

$$\boxed{L_D(h) \leq L_S(h) + \sqrt{\frac{|h| + \ln(2/\delta)}{2m}}}$$

Proof structure (essential)

1. choose $w(h) = 2^{-|h|}$,
2. apply Theorem 7.4,
3. absorb $-\ln w(h) = |h| \ln 2 \leq |h|$,
4. conclude.

No new probability arguments are required; this is a specialization of SRM.

The MDL learning rule

Theorem 7.7 induces the following learning rule:

$$\boxed{h \in \arg\min_{h \in \mathcal{H}} \left[L_S(h) + \sqrt{\tfrac{|h| + \ln(2/\delta)}{2m}} \right].}$$

Interpretation:

- minimize empirical loss,
- minimize description length.

This expresses a precise trade-off between data fit and model simplicity.

Exact connection to Occam's Razor

Occam's Razor states:

> "Do not use unnecessarily complex explanations."

Section 7.3 translates this into learning theory as:

> "Complex hypotheses are penalized because they generalize worse."

Thus:

- simplicity is not aesthetic,
- simplicity yields tighter generalization guarantees.

Structure of Section 7.3 at a glance

$$\begin{gathered}\text{SRM (Theorems 7.4, 7.5)}\\ \Downarrow \\ \text{Need weights } w(h) \\ \Downarrow \\ \text{Introduce description length } |h| \\ \Downarrow \\ \text{Prefix-free} \Rightarrow \text{Kraft inequality (Theorem 7.6)} \\ \Downarrow \\ \text{Justify } w(h) = 2^{-|h|} \\ \Downarrow \\ \text{MDL bound (Theorem 7.7)}\end{gathered}$$

7.3.1

The Exact Status of MDL

1. Why Section 7.3.1 is needed

By the end of Section 7.3, we have established that:

- Structural Risk Minimization yields non-uniform PAC learnability,
- MDL and Occam's Razor are rigorously justified via description length, prefix-free codes, and the Kraft inequality.

This naturally raises the question:

> "Is MDL stronger than PAC learning? Does it define a new notion of learnability?"

2. The core answer of Section 7.3.1

- **No.** MDL does not define a new notion of learnability.
- **Yes.** MDL is simply a concrete implementation of non-uniform PAC learning.

More precisely:

- MDL corresponds to a specific SRM strategy,
- using the weight choice $w(h) = 2^{-|h|}$,
- whose success is already guaranteed by non-uniform PAC theory.

3. The exact role of MDL

What MDL provides

- A principled answer to "which hypotheses should be trusted more?"
- A natural notion of complexity via description length.

What MDL does not provide

- No guarantees stronger than non-uniform PAC,
- No relaxation beyond what non-uniform PAC already allows.

Thus, the guarantee level of MDL is *exactly* non-uniform PAC.

7.4 Other Notions of Learnability - Consistency

1. Why consistency is introduced

Both PAC and non-uniform PAC learning share a strong requirement:

- they are distribution-free,
- the required sample size m does not depend on the data distribution D.

This leads to the question:

> "Is it necessary to require distribution-independent sample complexity? What if we allow the sample size to depend on the distribution?"

The answer is the notion of *consistency*.

2. Core intuition of consistency

"It is enough to eventually perform well."

- the rate of convergence may depend on the distribution,
- different distributions may require different sample sizes,
- but asymptotically the learner must match the optimal hypothesis.

3. Definition 7.8 — Consistency

The logical structure of the definition is:

$$\forall \varepsilon, \delta,\ \forall h,\ \forall D,\ \exists m(\varepsilon, \delta, h, D)$$

such that, for all $m \geq m(\varepsilon, \delta, h, D)$,

$$L_D(A(S)) \leq L_D(h) + \varepsilon.$$

Key difference:

- the sample size may depend on both h and D.

4. Universal consistency

A learner is *universally consistent* if:

- it is consistent for the class of *all* distributions.

That is:

> "For any distribution, given enough data, the learner converges to optimal performance."

5. Non-uniform PAC vs. Consistency

Concept	m depends on h	m depends on D
PAC	No	No
Non-uniform PAC	Yes	No
Consistency	Yes	Yes

Consistency is therefore a genuine relaxation of non-uniform PAC learning.

6. Relationship between notions

The book explicitly states:

- non-uniform PAC learnability $\Rightarrow$ universal consistency,
- the converse does not hold.

Hence:

$$\text{Consistency} \supsetneq \text{Non-uniform PAC}.$$

7. Example 7.4

Algorithm

- predict correctly on previously seen points,
- predict a default label on unseen points.

Result

- for countable input spaces,
- the algorithm is universally consistent.

However

- there is no distribution-independent sample complexity,
- hence it is not non-uniformly PAC learnable.

This example demonstrates how weak consistency can be.

8. The true message of Section 7.4

The meaning of "learnable" depends entirely on the type of guarantee required.

- PAC / non-uniform PAC guarantee *when* learning happens,
- consistency only guarantees that learning happens *eventually*.

9. Final structural summary of Chapter 7

$$\begin{aligned} \text{PAC} &\longrightarrow \text{uniform over } h \text{ and } D \\ \text{Non-uniform PAC} &\longrightarrow \text{non-uniform over } h,\ \text{uniform over } D \\ \text{MDL / SRM} &\longrightarrow \text{implementations of non-uniform PAC} \\ \text{Consistency} &\longrightarrow \text{non-uniform over } h \text{ and } D \end{aligned}$$

7.5 Discussing the Different Notions of Learnability

Final Integrated Summary

1. Question 1: "How large is the risk of the hypothesis I learned now?"

PAC and Non-uniform PAC

- For a finite sample size m, we obtain explicit bounds:

$$L_D(\hat{h}) \le L_S(\hat{h}) + (\text{explicit bound}).$$

- Training error is directly linked to true error.
- These frameworks provide a theoretical answer to "Can I trust the current output?"

Consistency

- No such finite-sample bound exists.
- Consistency only states eventual convergence to the Bayes-optimal hypothesis.
- The risk of the current output hypothesis cannot be evaluated theoretically.
- One must rely on validation or empirical estimation instead.

Conclusion:

> If we care about the reliability of the current learned hypothesis, consistency is useless; only PAC and non-uniform PAC are meaningful.

2. Question 2: "How many samples are needed to match the optimal hypothesis?"

PAC

- The sample complexity $m(\varepsilon, \delta, \mathcal{H})$ can be computed in advance.
- Provides a clear criterion: "collect at least this many samples."

Non-uniform PAC

- Sample complexity $m(\varepsilon, \delta, h)$ depends on the optimal hypothesis.
- Since the optimal h is unknown, this cannot be determined a priori.

Consistency

- Sample complexity $m(\varepsilon, \delta, h, D)$ depends on both the hypothesis and the distribution.
- Completely unpredictable in advance.

Conclusion:

> Only PAC learning can provide a distribution-independent, a priori sample complexity guarantee.

Even PAC cannot control approximation error, highlighting the importance of prior knowledge through hypothesis class selection.

3. Question 3: "How should we learn? How is prior knowledge expressed?"

PAC

- Prior knowledge is encoded via the hypothesis class $\mathcal{H}$.
- The learning rule is ERM.
- The No-Free-Lunch theorem clearly shows that learning without prior knowledge is impossible.

Non-uniform PAC

- Prior knowledge is expressed via hypothesis weights or structures.
- The learning rule is SRM.
- MDL and Occam's Razor are concrete implementations of SRM.
- Particularly effective for model selection, balancing complexity and data fit.

Consistency

- Provides no principled way to encode prior knowledge.
- Offers no natural learning paradigm.
- Even intuitively "non-learning" algorithms, such as memorization, qualify as learners.

Conclusion:

PAC and non-uniform PAC prescribe how learning should be performed, while consistency provides no such guidance.

4. Question 4: "Is a consistent algorithm therefore better?"

Superficial argument

"If an algorithm is consistent, it eventually reaches Bayes optimality, so it must be better."

The book's rebuttal

1. **Practicality:**
   - For some distributions, the required sample size is unrealistically large.
   - "Eventually" has little practical meaning.
2. **Consistency is easy to obtain:**
   - Combine a non-uniform learner with a risk bound.
   - Fall back to memorization when performance is poor.
   - Almost any algorithm can be made consistent.

Conclusion:

Consistency is too weak and too easily satisfied to serve as a meaningful criterion for algorithm selection.

7.5.1

Why there is no contradiction

The key lies in the order of quantifiers.

No-Free-Lunch

$$\forall m,\ \exists (D, h^*) \quad \text{such that algorithm } A \text{ fails.}$$

- The sample size is fixed first.
- A distribution and target that defeat the algorithm are chosen afterward.

Consistency

$$\forall (D, h^*),\ \exists m \quad \text{such that algorithm } A \text{ succeeds.}$$

- The distribution and target are fixed first.
- A suitable sample size is chosen afterward.

These statements are logically different and therefore not contradictory.

6. One-table summary of Section 7.5

Question	PAC	Non-uniform	Consistency
Finite-sample risk bound	Yes	Yes	No
A priori sample complexity	Yes	No	No
Learning principle	ERM	SRM	None
Prior knowledge encoding	$\mathcal{H}$	Weights / structure	None
Practical usefulness	High	Very high	Low