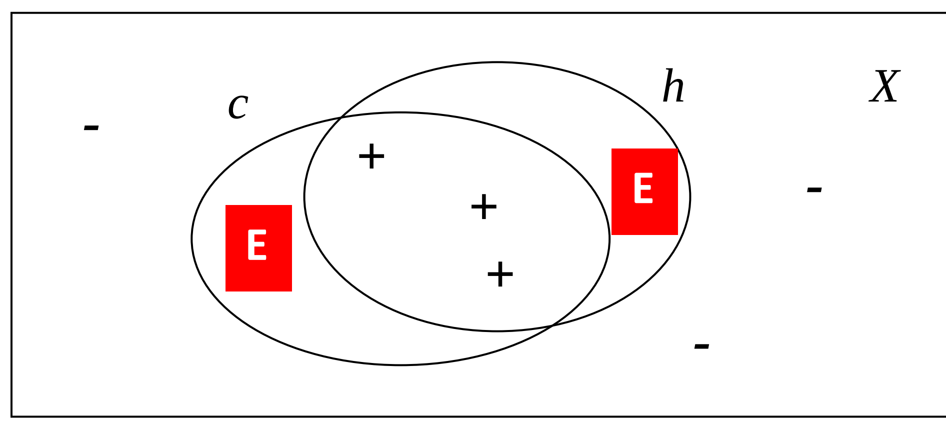
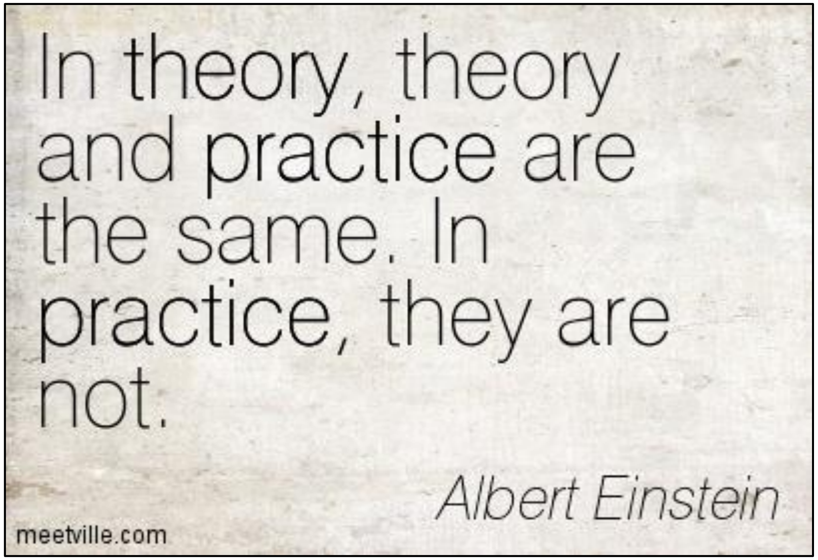
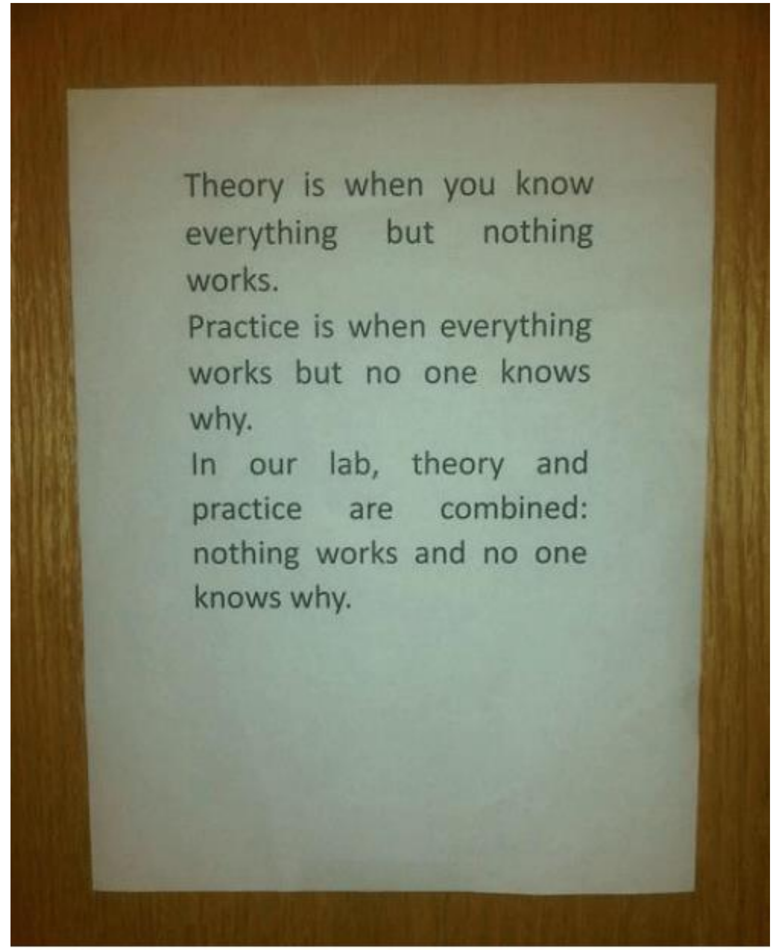
Learning Theory
Sample Complexity
PAC Learning

Ariel Shamir
Zohar Yakhini




In theory, theory and practice are the same. In practice, they are not.
Albert Einstein
meetville.com


Theory is when you know everything but nothing works.
Practice is when everything works but no one knows why.
In our lab, theory and practice are combined: nothing works and no one knows why.


THEORY
PRACTICE

Outline

- Generalization and evaluating the true error (again …)
- Complexity of learning
- Sample complexity
- Examples:
  - Finite spaces and Boolean expressions
  - Circles, rectangles
- VC dimension

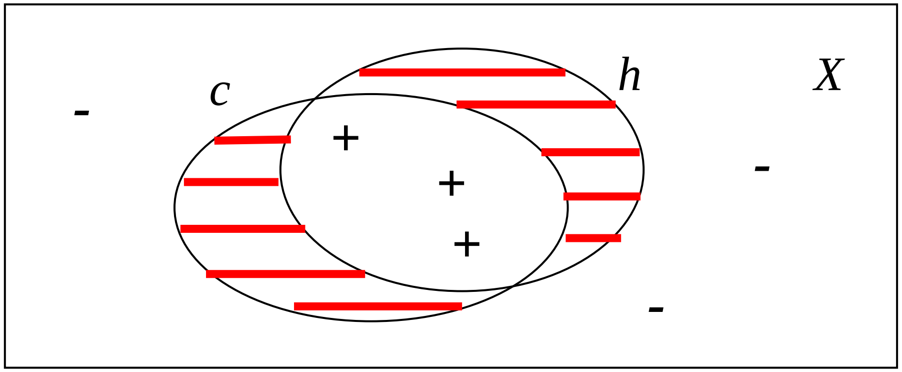
General setting

- Instances come from $\Omega = (X, Y, \pi)$
- Consider a concept $c \subseteq X$, which we are trying to learn. That is – provide a model (hypothesis) h that will be positive on c and negative otherwise
- The learning algorithm L takes training data $D \in \Omega^m$
- It works with some set of hypotheses, H
- It returns a hypothesis $L(D) = h \in H$

The True Error of h

Assume that we have a probability distribution over X (called π in the previous slide)
Then we can define:

$$TrueErr(h) = \ error_{\pi}(h) = \pi\left(x \in X \ : \ c(X) \neq h(X)\right)$$



Statistical Estimation – a detour into practice

- In practice, we use a test set to estimate the true error of a candidate hypothesis. It is a representative sample of data we have not used for learning.
- If the test set is all the rest of X then we know the true error!
  (of course this is unrealistic and we use sampling)

Statistical Estimation of the Classification Error

- Using a test set of size n, assume that we counted r errors.
- We estimate the generalization error by $\hat{p} = \frac{r}{n}$.
- From statistical sampling theory it follows that a 95% confidence interval for the generalization error is

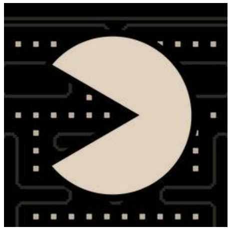
$$(\hat{p} - 2se, \hat{p} + 2se)$$

where

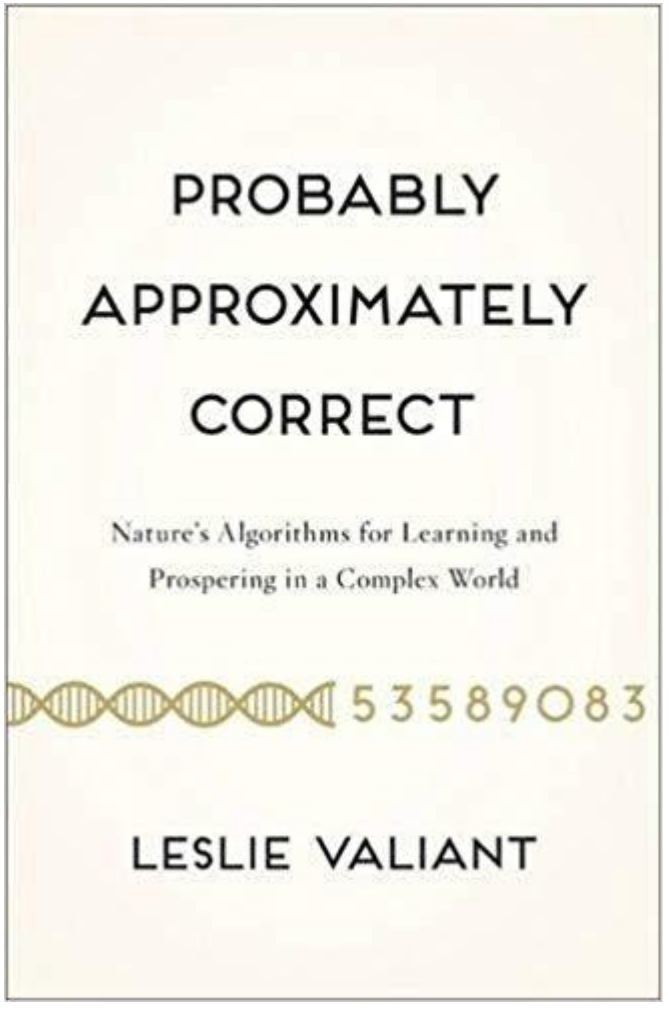
$$se = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

Back to the sample complexity theory

PAC Learning



Leslie Valiant
Harvard Univ
Born 1949
Turing Award Laureate in 2010
American computer scientist

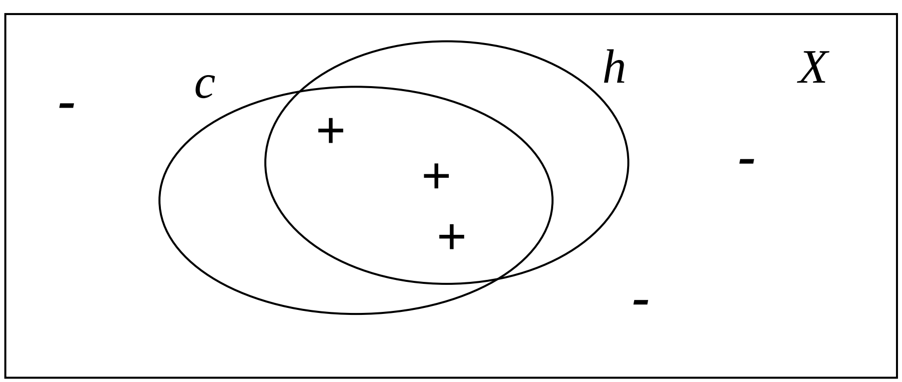


Probably Approximately Correct Framework

1. **Probability**
   (guarantees given with $1 - \delta < 1$ certainty)
2. **Approximation**
   (a desired bound, $\varepsilon > 0$, on the true error will be specified)
3. We will study the use of **resources**
   - Size of training set (<u>sample complexity</u>)
   - Time/space of learning
     (learnability in polynomial time per training instance)

PAC framework, cont

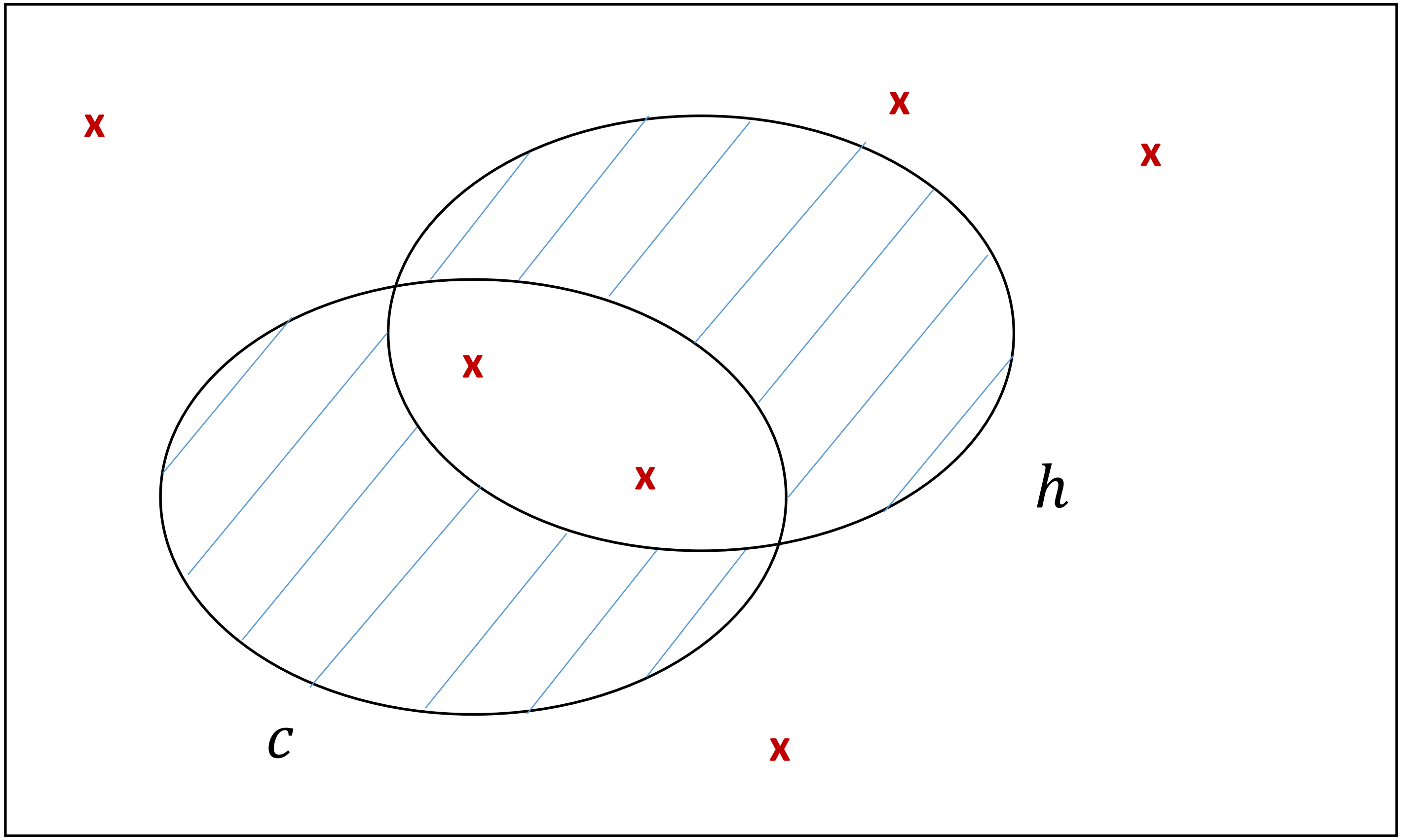
- We are trying to learn a concept c (in this framework, a subset of X)
- We have a training set in the instance space X, drawn at random according to Ω^m and labeled without errors
- Working with a hypothesis space H we seek a hypothesis h that is learned based on our data
- We want to guarantee that with probability $1-\delta$ it will not have a true error of more than ε

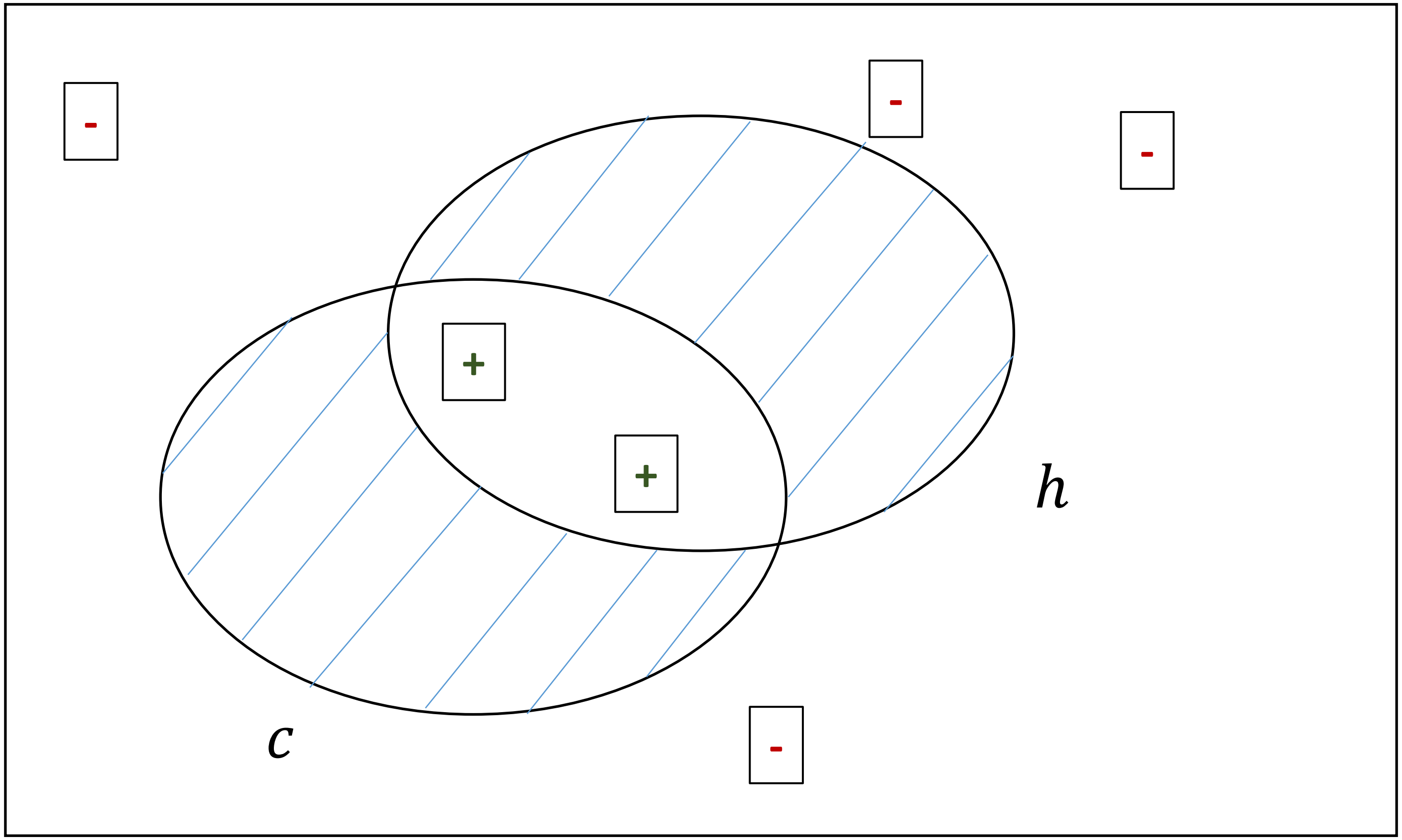


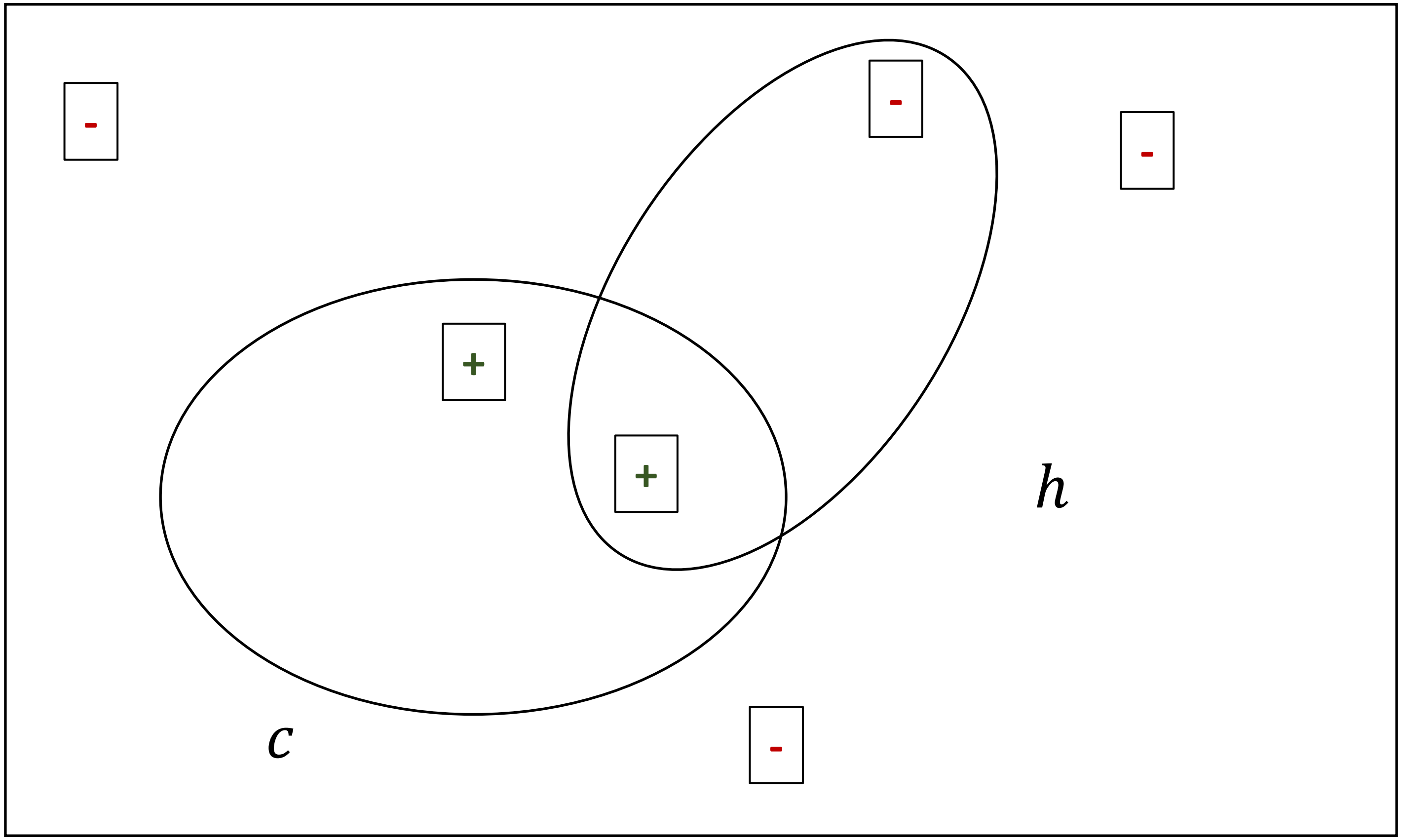
Consistent hypotheses

A hypothesis h is D-*consistent* with respect to a concept c and training data D if

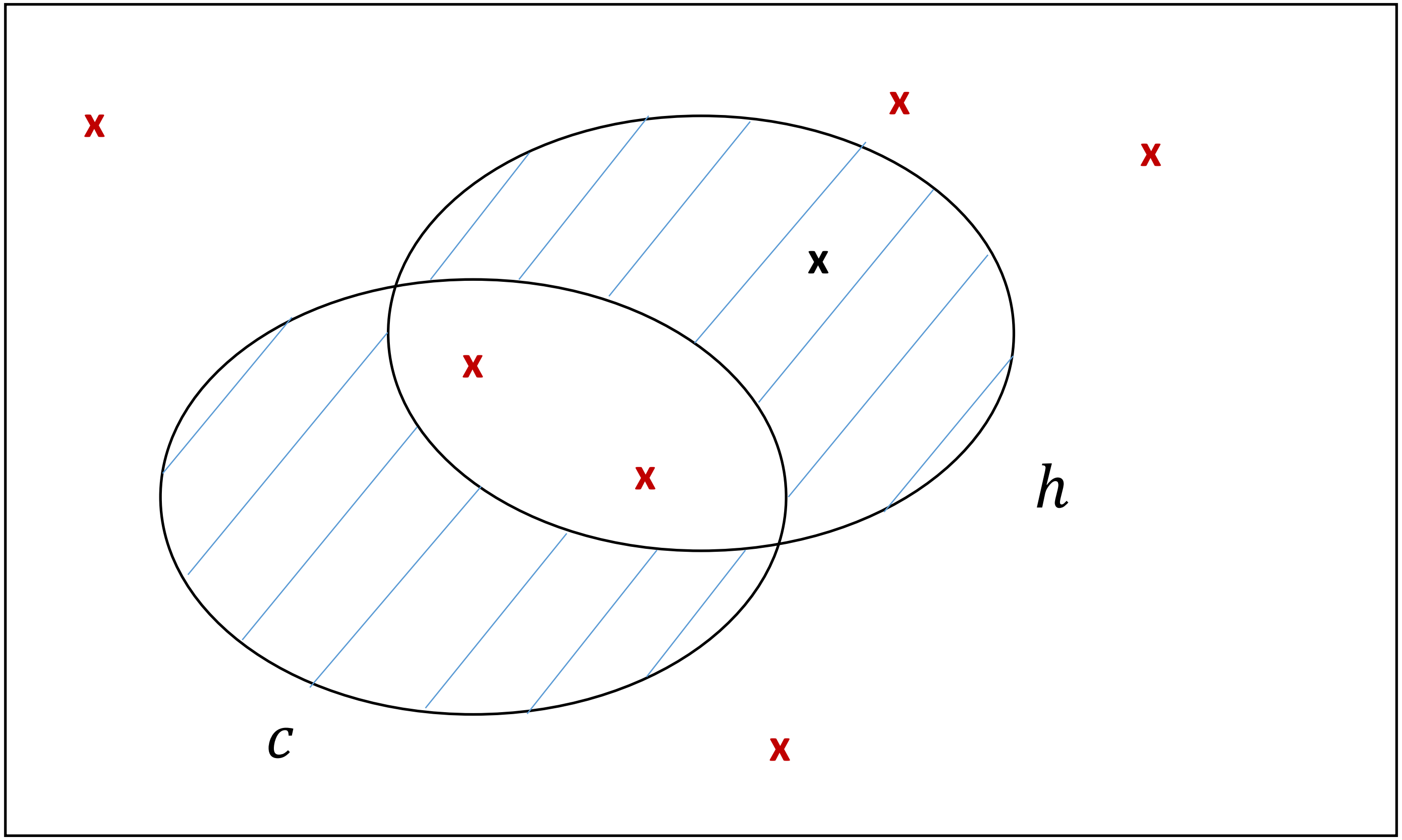
$$\forall d \in D \;\; h(d) = c(d)$$


x
x
x
x
x
x
h
c


-
-
-
+
+
h
c
-



-
-
-
+
+
h
c
-

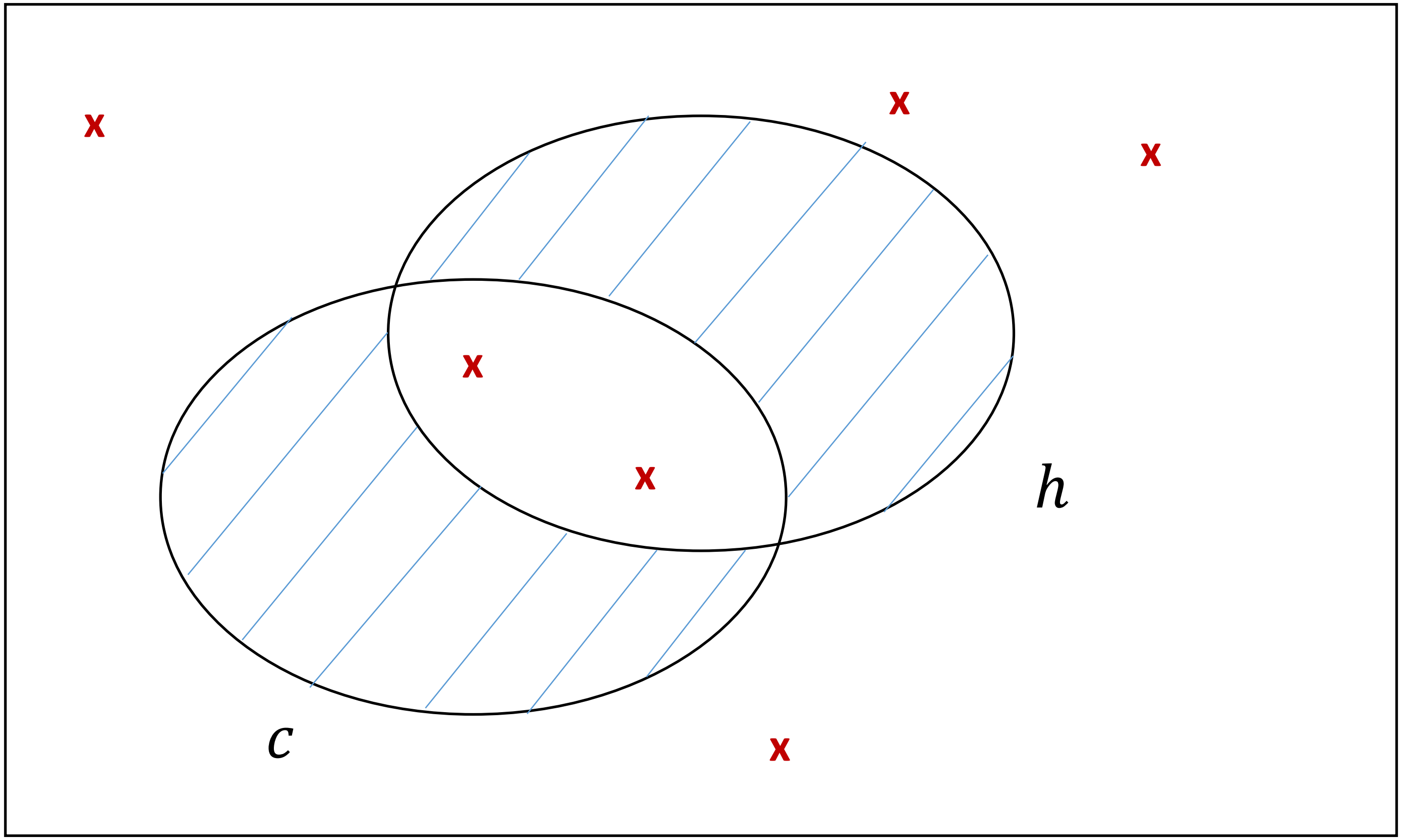

x
x
x
x
x
x
x
h
c

Consistent learners

A learning algorithm L , operating on training data produced by concepts from C and using a hypothesis space H is said to be a <u>consistent learner</u> if for any training data D and for any $c \in C$ the output $h = L(D)$
is D-*consistent* with respect to c .

Sample Complexity: Finite Hypothesis Spaces

- A hypothesis h is ε-bad if $error_{\pi}\ (h)\ >\ \varepsilon$ (e.g. 5%)
- A **consistent learner** using H must output some consistent h for any m samples
- Question: what is the probability that such a hypothesis h (an output of a consistent learner) will be ε-bad?
- In other words:
  what is the probability of obtaining a training dataset that may lead to such h, when our learner is consistent?



If h is ε bad then $\pi(\mathrm{err}) \geq \varepsilon$

To be the output of a consistent learning algorithm, all m training data points had to have avoided the blue region.

Otherwise h would not have been consistent

A bound on $Prob(h$ is ε-bad)

- Consider some ε bad hypothesis $h \in H$
- For h to be the output of a consistent learner the training data has to have avoided a region $E \subset X$ with $\pi(E) \geq \varepsilon$.
- Since all m samples are independent, the probability (in $\Omega^m = (X, \pi)^m$) of an ε-bad h to be the output of a consistent learner is therefore $\leq (1 - \varepsilon)^m$

That is:

If L is a consistent learner for C
then,
for any $c \in C$ and any $\varepsilon -$ bad hypothesis $h \in H$,
we have:

$$\pi^m(\underbrace{D: L(D) = h}) \leq (1-\varepsilon)^m$$

Denote this set of
datasets by $T(h)$

Cont …

Assume that H is finite.
Considering m sampled data points, $D \in X^m$, the probability that $L(D)$ is $\varepsilon -$ bad can now be bounded as follows:

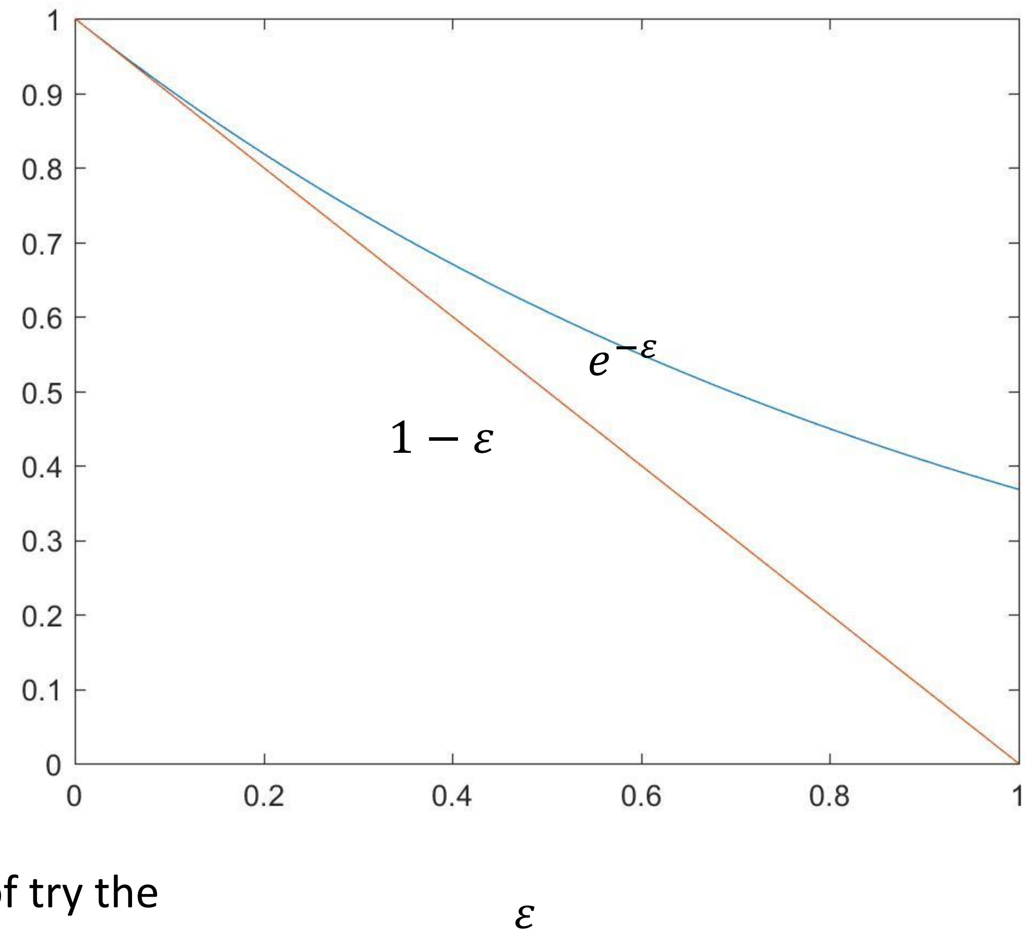
$$Prob(L(D) \text{ is } \varepsilon - \text{bad }) = \pi^m(D\colon L(D) \text{ is } \varepsilon - \text{bad})$$

$$= \pi^m\left(\bigcup_{h \text{ is } \varepsilon-\text{bad}} T(h)\right) \leq \sum_{h \text{ is } \varepsilon-\text{bad}} \pi^m(T(h))$$

$$\leq \sum_{h \text{ is } \varepsilon-\text{bad}} (1 - \varepsilon)^m \leq |H|(1 - \varepsilon)^m$$

$$\leq |H| \cdot \exp(-\varepsilon m)$$

$$1 - \varepsilon < e^{-\varepsilon}$$



For a formal proof try the Taylor/MacLaurin expansion of $\exp(-x)$ around 0.

First Bound on Sample Complexity

- Note that linear increase in the sample size reduces the chance of error in an exponential rate!
- In order to reduce the failure probability below some desired level δ we can require:

$$|H|e^{-\varepsilon m} \leq \delta \;\text{ or}$$

$$m \geq \tfrac{1}{\varepsilon}\ln\tfrac{|H|}{\delta} = \tfrac{1}{\varepsilon}\left(\ln|H| + \ln\tfrac{1}{\delta}\right)$$

- This is not a tight bound (mainly due to replacing the number of ε-bad hypotheses by the size of H)

Example: Disjunctions of Boolean Literals

- The instance space X is n dimensional Boolean vectors.
- The hypotheses space, H, and the concept space, C, both consist of disjunctions of n Boolean literals of the form:

$$x_1 \vee \bar{x}_5 \vee \bar{x}_{22}$$

- No further limitations on the hypotheses/concept space:

  For each Boolean variable x_i our hypothesis may contain either x_i or $\overline{x_i}$ (but not both) or none of them. The latter will mean that we get F for x_i

- Can we construct a consistent learner?

A Consistent Learner

Begin with $h = x_1 \lor \bar{x}_1 \lor x_2 \lor \bar{x}_2 \lor \cdots \lor x_n \lor \bar{x}_n$

For each negative training instance $\vec{x}$ (c$(\vec{x})$ $=$ false):

Remove all literals l_i from h, which are consistent with $\vec{x}$

We end up with $h = l_{i_1} \lor l_{i_2} \lor \cdots \lor l_{i_k}$
that is consistent also with the positive instances,

since one of their literals must be in h .

Thus we end up with h that is consistent with all training examples

Example

start: you start with all literals. x1 is positive x1_bar is negative. from the last page, you need to remove each iteration(sample) the literals that are consistent(true), hence, in example 1, you remove x1_bar, x2_bar, x3, x4_bar, then in sample2 you remove x1_bar(already removed), x2, x3_bar, x4_bar(already removed), and you are left with a consistent h

1. Start: $x_1 \vee \overline{x_1} \vee x_2 \vee \overline{x_2} \vee x_3 \vee \overline{x_3} \vee x_4 \vee \overline{x_4}$
2. Instance 1: $x_1 \vee x_2 \vee \overline{x_3} \vee x_4$
3. Instance 2: $x_1 \vee x_4$

- Consistent with Instances 3 & 4

Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1

0 x x x

x 0 x x

x x 1 x

x x x 1

How many instances are sufficient?

- Assume we have 10 attributes
- In our example, using conjunctions of features (or empty conjunctions at some is) we get $|H| = 3^{10} = 59{,}049$
- We want to ensure with 95% certainty that our hypothesis will have error < 10% .
- We then need $m > \frac{1}{0.1}(\ln 59049 + \ln \frac{1}{0.05}) = 10(11+3) = 140$ instances
- Note that we have 1024 instances in total (all of X)

Another example

- 20 attributes in the same setting
- We get $|H| = 3^{20} \sim 3.5 * 10^9$
- In this case, to have 95% certainty that our hypothesis will have error < 10% , we need $m > \frac{1}{0.1}(\ln 3.5*10^9 + \ln \frac{1}{0.05}) = 10(22+3) = 250$ instances
- In this case we have about 10^6 possible instances

When |H| increases exponentially with the number of features then sample complexity increases linearly.
The required fraction of the full population decreases.

PAC Learnability

- Consider a class **C** of possible target concepts defined over a space of instances **X**, and a learning algorithm **L** using hypothesis space **H**.

- <u>Definition</u>

הגדרה שלא באמת חייב להבין... (זוהר אמר)

<u>C is PAC-learnable by L using H</u>

<u>if</u> for all $0 < \varepsilon < ½$, $0 < \delta < ½$, and for all $c \in$ **C** and distributions π over **X**, the following holds:

with data drawn independently according to π, **L** will output, with probability at least $(1-\delta)$, a hypothesis $h \in$ **H** such that $\text{error}_{\pi}(h) \leq \varepsilon$,

L operates in time (and sample) complexity that is polynomial in $1/\varepsilon$, $1/\delta$ (and in other possible parameters).

Summary and next steps

- Generalization errors
- PAC learning framework
- Consistent learners for finite hypotheses spaces
- Directly calculating bounds on the sample complexity of consistent learners
- Next week – complete our theory section:
  - Concepts in $\mathbb{R}^n$
  - Agnostic learning
  - VC dimension
- After theory – unsupervised learning, special topics