

Computational Learning Theory

Computational learning theory is interested in theoretical analyses of the following issues.

What is needed to learn effectively?

Sample complexity. How many examples?

Computational complexity. How much computation?

Mistake bound. How many mistakes?

Important attributes of learning problems are:

Size/complexity of the hypothesis space

ϵ : Desired accuracy of the hypotheses

δ : Probability of success (confidence)

Presentation of examples to the learner

Definitions

Examples are drawn from $X \times Y$ according to probability distribution D .

Each $\mathbf{x} \in X$ has length n , and $Y = \{0, 1\}$.

The target concept $c \in C$ correctly classifies all examples (or c is Bayes optimal).

The learner L can obtain any number of labeled examples.

L returns a hypothesis $h \in H$ (possibly $C \neq H$).

The *true error* of h is:

$$\text{error}_D(h) = \Pr_{(\mathbf{x}, y) \in D} (h(\mathbf{x}) \neq y)$$

Probably Approximately Correct

C is *PAC-learnable* by L using H ,
if for all $c \in C$, D , $\epsilon \in (0, 1/2)$, $\delta \in (0, 1/2)$,
where $error_D(c) = 0$, then, with probability $\geq 1 - \delta$,
 L will output an $h \in H$ such that $error_D(h) \leq \epsilon$

PAC-learnable in polynomial time means the computation is polynomial in $1/\epsilon$, $1/\delta$, and n . A key requirement is for the sample complexity to be polynomial.

A variation is to relax consistency, e.g.,
where $error_D(c) = \alpha$, then, with probability $\geq 1 - \delta$,
 L will output an $h \in H$ such that $error_D(h) \leq \epsilon + \alpha$

Sample Complexity of Finite H for Consistent Hypotheses

Theorem: If $c \in H$, and L finds a consistent h after m examples, where

$$m \geq \frac{\ln |H| + \ln(1/\delta)}{\epsilon}$$

then H is PAC-learnable by L using H ,
that is, $error_D(h) \leq \epsilon$ with prob. $\geq 1 - \delta$

Proof: If h' is bad, i.e., $error_D(h') > \epsilon$, then the prob. that h' is correct on all m exs. is $< (1 - \epsilon)^m$.

The number of bad hyps. is $< |H|$, so the prob. that some bad hyp. is correct on all m exs. is $< |H|(1 - \epsilon)^m$.

Proof Continued

We want $|H|(1 - \epsilon)^m \leq \delta$.

Because $1 - \epsilon \leq e^{-\epsilon}$,

$$|H|(1 - \epsilon)^m \leq |H|(e^{-\epsilon})^m = |H|(e^{-\epsilon m})$$

$$|H|(e^{-\epsilon m}) \leq \delta \text{ is equivalent to } -\epsilon m \leq \ln \frac{\delta}{|H|}$$

$$-\epsilon m \leq \ln \frac{\delta}{|H|} \text{ is equivalent to } \epsilon m \geq \ln \frac{|H|}{\delta}$$

$$\epsilon m \geq \ln \frac{|H|}{\delta} \text{ is equivalent to } m \geq \frac{\ln(|H|/\delta)}{\epsilon}$$

Sample Complexity of Finite H for Inconsistent Hypotheses

Th: If L finds an h with sample error α on m exs. where

$$m \geq \frac{\ln |H| + \ln(1/\delta)}{2\epsilon^2}$$

then $error_D(h) \leq \alpha + \epsilon$ with prob. $\geq 1 - \delta$

Proof: By Hoeffding's inequality, sample error of a single hyp. is off by more than ϵ with prob. $e^{-2m\epsilon^2}$.

To satisfy $|H|(e^{-2m\epsilon^2}) \leq \delta$

m needs to satisfy $m \geq \frac{\ln(|H|/\delta)}{2\epsilon^2}$

Examples of Polytime PAC-learnable Hypothesis Spaces

Boolean conjunctions and disjunctions: A conjunction or disjunction of Boolean literals.

k -CNF: k -CNF is a conjunction (AND) of disjunctions (ORs); each disjunction has k or fewer literals.

k -term DNF: k -term DNF is a disjunction of conjunctions. Each conjunction has at most k disjuncts.

Decision list: A decision list is a recursive structure of the form: if literal then outcome else sub-decision-list.

Parity: Even or odd parity of a subset of the literals.

The VC Dimension

Any H with real-number parameters/weights has an infinite number of hypotheses. The *Vapnik-Chervonenkis* dimension describes how H is limited.

A dataset S is *shattered* by H iff for every subset S' of S , there is an $h \in H$ that classifies S' as positive and $S - S'$ as negative.

The VC Dimension of H over X is the size of the largest subset of X shattered by H .

If $VC(H) = d$ and $|S| = m > d$, then H can split S in at most $(em/d)^d$ ways. Note this is much less asymptotically than 2^m , the number of subsets of S .

Examples of VC Dimension

H = intervals on the real number line

H = linear threshold functions on the plane

H = rectangles on the plane

$VC(\text{linear threshold functions}) = n + 1$

n = number of attributes

$VC(\text{neural networks}) \leq 2ds \log_2(es)$

$d \geq VC(\text{any unit}), s$ = number of non-input units

$VC(\text{SVMs}) \leq 2 + D^2/M^2$

D = maximum distance between examples

M = width of margin

A heuristic is $VC(H) \approx$ number of parameters/weights

VC Bounds

Theorem: If $c \in H$, if L finds a consistent h after m examples, where

$$m \geq \frac{4}{\epsilon} \left(VC(H) \log_2 \frac{12}{\epsilon} + \log_2 \frac{2}{\delta} \right)$$

where $VC(H)$ is the VC Dimension of H ,

then H is PAC-learnable by L using H ,

that is, $error_D(h) \leq \epsilon$ with prob. $\geq 1 - \delta$.

Th: If L finds an h with sample error α on m exs. where

$$m \geq \frac{8}{\epsilon^2} \left(VC(H) \ln \frac{16}{\epsilon^2} + \ln \frac{6}{\delta} \right)$$

then $\alpha - \epsilon \leq error_D(h) \leq \alpha + \epsilon$ with prob. $\geq 1 - \delta$.

Practicality of Sample Complexity Bounds

All the above bounds are *distribution-free*. There are no assumptions on how the examples are distributed.

In practice, good results are achieved using much fewer examples than these bounds. One possible explanation is that “typical” datasets do not have worst-case distributions. Even so, the number of examples should be many times $\ln H$ or the VC dimension.

The key asymptote from all the bounds is that the sample complexity is logarithmic in the number of hypotheses. Efficient learning is possible with exponential-size hypothesis spaces.

Mistake Bounds

How many mistakes k will Find-S make before it converges to the target concept?

Find-S

$h \leftarrow x_1 \wedge \neg x_1 \wedge \dots \wedge x_n \wedge \neg x_n$

for each example \mathbf{x}

 predict outcome of \mathbf{x} using h

 receive outcome y for \mathbf{x}

 if y is positive

 then $h \leftarrow h - \text{literals inconsistent with } \mathbf{x}$

 else if prediction is wrong

 then no consistent hypothesis exists

return h

The Halving Algorithm

Halving(H : a finite set of hypotheses)

for each example \mathbf{x}

predict outcome of \mathbf{x} by majority vote of H

receive outcome y for \mathbf{x}

$H \leftarrow H - \text{hyps. inconsistent with } (\mathbf{x}, y)$

return H

If a mistake is made, at least $1/2$ of the remaining hyps. are eliminated. This implies $k \leq \log_2 |H|$. More mistakes would result in $|H| < 1$.

For Find-S algorithm, $|H| = 3^n$. For each x_i , a hypothesis has x_i , $\neg x_i$, or neither, so $k \leq n \log_2 3$ for Find-S.

Weighted Majority Algorithm

Weighted-Majority(β, L_1, \dots, L_n)

L_1 through L_n are n learning algs.

$0 < \beta < 1, y \in \{-1, 1\}$

for i from 1 to $n, w_i \leftarrow 1$

for each example \mathbf{x}

$sum \leftarrow 0$

for i from 1 to n

$sum \leftarrow sum + y w_i$

if $sum > 0$ then predict 1 else predict -1

receive outcome y for x

for i from 1 to n , if $L_i(x) \neq y$ then $w_i \leftarrow \beta w_i$

send outcome y to L_1 through L_n

Weighted Majority Analysis

If some learner L_j makes k mistakes, then Weighted-Majority with $0 < \beta < 1$ makes at most

$$K \leq \frac{k \ln \frac{1}{\beta} + \ln n}{\ln \frac{2}{1+\beta}}$$

mistakes (\ln is natural logarithm).

Because L_j makes k mistakes, then $w_j = \beta^k$.

The sum of the weights must be at least $w_j = \beta^k$.

Whenever Weighted-Majority makes a mistake, at least $1/2$ of the old sum of the weights is multiplied by β .

This implies that the new sum of the weights is at most $1/2 + \beta/2$ times the old sum.

If Weighted-Majority makes K mistakes, the sum of weights is at most

$$n \left(\frac{1 + \beta}{2} \right)^K$$

The initial sum of weights is n . Each mistake reduces the sum by a factor of $1/2 + \beta/2$ or more.

Thus:

$$\beta^k \leq n \left(\frac{1 + \beta}{2} \right)^K$$

which is equivalent to the above bound.

Graph of Weighted Majority Bound

