Admin: Last lecture! 🎭
Pick up PS sol 1-4 after class.

Last time:
- rel. betw. SQ learning, PAC learn,
  PAC learn. w/noise
- Inherent unpred. / crypto hard. of
  learning.

Today:
- AMA about course.
- topics you may be interested in.

PS 5, problem #2:
using AdaBoost to learn certain CTFs
using $\mathbf{x}_i, -\mathbf{x}_i$ as WH’s. $\rightarrow$ Ada. Final H
is a CTF!

Comp. Hard. of Learning.

- Rep. dependent HOL (3-term O(NF using):
  "complexity-theoretic" assumptions.
  (worst-case assumptions.)

- Rep. independent HOL (no poly(n)-time alg,
  can eff learn $R = \forall$ poly(n)-size circuits): 
  "cryptographic" assumptions.
  (average-case assume.)

 Stranger
Ex of WC hardness:

"There is no poly(n)-time rand. alg. which, given any n-node graph, correctly determines if it's 3-colorable." 

Ex of Avg-case hardness:

"There is no poly(polylog)-time rand. alg. which, given as input \( N = p \cdot q \) where \( p, q \) are unit. random n-bit primes, correctly factors \( N \) with succ. prob. \( \gt \frac{1}{100} \)."

H0C

Book: can establish based on existence of public-key cryptosystems (trapdoor permutations)

"Decryption functions in are hard to learn."

Today: pseudorandomness.

"Oracle access to function \( f \): black-box access to \( f \).

\( x \xrightarrow{\$} f(x) \rightarrow f \)

"Membership every access"
alg $A$: write $A^f$ to indicate $A$ has oracle access to $f$.

"Pseudorandom functions": back up.

vs

Random functions

"$f$ is a (truly) random fn":
$f$ is chosen uniformly from all $f$s $\{0,1\}^n \rightarrow \{0,1\}$.

$2^n$ poss.

$2^n$ times to gen.

$TT$

"$f \sim \text{RAND}$" means $f$ is truly random fn.

like this.

Alg $A$: $A^f$: every query to $f$ is answered w/ fresh coin toss.
**Def:** Let $\mathcal{F}$ be a set of $2^n$ BF's

$$\{f_s : s \in \{0,1\}^n\}$$

Each $f_s : \{0,1\}^n \rightarrow \{0,1\}$.

We say $\mathcal{F}$ is a **PRFF** (pseudo random fn family) if

1. **(eff. computability):** There is a poly (n) time alg. which, given $s, x$, outputs $f_s(x)$.

2. **(indistinguishability):** Let $O$ be any poly (n) time alg. (maybe random) which gets oracle access to a BF & outputs "R" or "PR". Then

$$\Pr[f \sim \text{Rand}, \text{outputs "R"}] \leq \frac{1}{p(n)}$$

for every poly. $p(n)$.

- **Super-useful in crypto.**

- **Fact:** Under (insert suitable average-case hardness assumption),
If one-way functions exist, so do PRF.

Suppose \( g \) is a PRF. Then there is no poly (1/\( \epsilon \))-time PAC learning algorithm \( A \) for \( g \) under \( \epsilon \)-error. For \( g \), even if \( \epsilon \) only needs to succeed with \( \epsilon \) probability \( 1/2 \), the learner is given oracle access to \( g \).

Then can use \( A \) to dist. truly and \( F \) from PRF in poly-time, as follows: Given \( g \), run learning algo \( A \) using \( \hat{E}(g) \) to dist. truly and \( F \) from PRF sketch. So \( A \) is an eff. PAC learning algo.

Suppose \( F \) is a PRF. Then

1. **Construction:** We construct a PRF \( F \) from a given PRF \( g \).
2. **Oracle Access:** We have oracle access to \( g \).
3. **Poly-time:** We can learn \( F \) in poly-time.
4. **Error Bound:** We achieve error \( \epsilon \) with high probability.

Thus, if one-way functions exist, so do PRF.
oracle for \( f \) to get \( f(x) \). Eval. \( h \) on \( x \) to get \( h(x) \). If \( h(x) = f(x) \), output "PR". If \( h(x) \neq f(x) \), output "R".

\[
\text{If } f = f_S, s \sim \{0,1\}: \quad (\text{PR case}) \quad \text{get } \quad \text{w.p. } > 1 - 0.01 - 0.01 = 98\%.
\]

\[
\text{If } f \sim \text{RAND}: \quad \text{w.p. } > 1 - \frac{\text{poly}(n)}{2^n} \quad \text{training,}
\]

\( z \) is a "fresh" new pt \( x \) and \( f(z) \) is \( \emptyset \), so \( \Pr[h(z) \neq f(z)] = \frac{1}{3} \).

Overall \( \Pr[\text{output } \text{"PR"}] \leq \frac{1}{2} + \frac{\text{poly}(n)}{2^n} \).

This yields, via suitable crypto, that

depth-5 LTF circuits is inherently unlearnable.