Admin: No class Thurs; Rocco OH today 11-1.

Last time: times: \( \Delta Pf \) of AdaBoost acc. bd.
\[
\frac{\#ex \text{ in } m\text{-elf train set}}{\text{on which Ada. wrong}} \leq \prod_{t=1}^{T} \sqrt{1 - \gamma_t^2} 
\]

- Noise framework: \( \eta = \text{noise rate} \)
- Mal noise model "
- RCN " " "

Today: introduce SQ model (Statistical Queries)

\( \exists \text{ SQ-learnable } \Rightarrow \exists \text{ PAC learnable in presence of RCN} \)

KV chapter 5

New Model: Learning from Statistical Queries

\( \rightarrow \) Idea:

Learn algorithm doesn't get \((x, c(x))\). Can access a "STAT(c,y)" oracle estimates prob. having to do with \((x, c(x)) \text{ x } \sim \mathcal{D} \).

A predicate of a lab. ex. \((x, c(x))\): a statement about \( x \) that's either \( T \) or \( F \).

Ex: " \((x, c(x)) \text{ has } c(x) = 1 \text{ + has } x_1 = x_2 = x_3 = 0\)".

Formally, a pred. of a lab. ex. is
Given a pred. $\pi$, define

$$P_\pi := \Pr \left[ \pi(x, c(x)) = 1 \right].$$

**Def:** The oracle $\text{STAT}(c, \theta)$ takes 2 inputs:

1) pred. $\pi$; 2) a tolerance parameter

$0 < \tau < \frac{1}{\sqrt{2}}$

It outputs a value $\hat{P}_\pi$ s.t. $\hat{P}_\pi$ is $(\pm \tau)$-add. est. of $P_\pi$.

$$P_\pi - \tau \leq \hat{P}_\pi \leq P_\pi + \tau$$

**Ex:** maybe $\pi$ is "$c(x) = 1, x_2 = 0$"

"$\tau = 0.01$.

If $\text{STAT}$ returns "0.36": know $P_\pi \equiv \hat{P}_\pi$ is in $[0.35, 0.377]$. 
Note: If you have EX(c, \( \theta \)) (noiseless),

you can simulate STAT(c, \( \theta \)) w/ failure prob. \( \delta \)
in straightforward way:

- Given \( x, z \): draw \( m \) ex. from EX(c, \( \theta \)).
  \((x, c(x))\): For each, eval. \( z(x, c(x)) \) & see if \( z = 1 \).

Your \( \hat{P}_x \) est. of \( P_x \): frac of \( m \) ex. s.t. \( z = 1 \).

CB: taking \( m = \frac{o(1)}{\varepsilon^2} \cdot \ln \left( \frac{1}{\delta} \right) \), w. p. \( \geq 1 - \delta \)

\[ \hat{P}_x \in [P_x - \varepsilon, P_x + \varepsilon] \]

Build intuition/consider efficiency:

- Should view "cost" of a call to STAT(c, \( \theta \))
  with \((x, z)\) as scaling with

  - \( m \): amount of time required to eval. \( z(x, c(x)) \)
  - \( \frac{1}{\varepsilon} \) (need this many ex to est. \( P_x \) to \( \pm \varepsilon - \varepsilon \))

Making model official: \( (\text{SQ-learnable}) \)

Def: Class \( C \) is learnable from SQ's if:

there is a learning alg \( \mathcal{L} \) s.t.:
\[ \forall c \in \mathcal{E}, \]
\[ \forall \varepsilon > 0, \]
\[ \forall \mathcal{D} \text{ distr. over } \mathcal{X}, \]

If \( L \) is given \( \varepsilon \) + access to \( \text{STAT}(c, \mathcal{D}) \), then \( L \) outputs an \( \varepsilon \)-acc hyp \( h \), i.e.
\[ \Pr_{x \sim \mathcal{D}}[h(x) \neq c(x)] \leq \varepsilon. \]

We say \( \mathcal{E} \) is efficiently SQ-learnable by \( L \) if

- For every query \((x, c)\) provided to \( \text{STAT}(c, \mathcal{D}) \) in execution of \( L \),\( \\{x \in \mathcal{X} \mid x = 0 \text{ or } 1 \} \) or \( \mathcal{X} \)
  - \( x(x, c(x)) \) can be evaluated \( \text{eff. in} \)
  - \( \text{time } \text{poly} \left( \frac{\varepsilon}{\varepsilon}, \text{size}(c), n \right) \)
  - \( L \geq \frac{1}{\text{poly} \left( \frac{\varepsilon}{\varepsilon}, \text{size}(c), n \right)} \)

- \( L \) runs in time \( \text{poly} \left( \frac{\varepsilon}{\varepsilon}, \text{size}(c), n \right) \) when we view each call to \( \text{STAT} \) as taking 1 time step.

Directly get:

**Theorem (easy):** If \( \mathcal{E} \) is efficiently SQ-learnable, then \( \mathcal{E} \) is also PAC learnable.
**Proof sketch:** The SQ alg makes \( M \) queries to STAT, each \( \text{poly}(\ldots) \) with tol. \( \varepsilon > \varepsilon_0 \), each w/ efficiently evaulatable \( \mathcal{X} \). Just saw: can sim. each execution of STAT \((c, \theta)\) on such a \((\mathcal{X}, \varepsilon)\) pair w/ succ. prob. \( \geq 1 - \frac{\delta}{M} \).

Next time: real the:

\[ C \xrightarrow{\text{eff.}} \text{SQ learnable} \Rightarrow C \xrightarrow{\text{eff.}} \text{PAC learnable in presence of RCN.} \]

Reduces RCN PAC learning to SQ learning.

Most PAC learnable \( C \)'s also have SQ alg's.