Last time: Specific noise models: toy scenario
- malicious noise (hard!) :(
- random classification noise (easy!) :) → RCN

→
- Learning with malicious noise is hard :(
- Sad, sad positive result for
- Start discussion of learning with RCN :)

Today: Handling RCN.
- Revisit old PAC alg (for mon. conj.) that's not RCN-tolerant, adapt it to handle RCN
- Motivates new learning model: learning from STATISTICAL QUERIES (SQ)
- Any SQ learning alg. automatically yields a PAC alg. that can handle RCN.

Questions?

---

Let's consider... PAC learning mon. conj. over $\{0,1\}^n$, with RCN.

Old alg:
- Start w/ hyp. $h(x) = x_1 \wedge x_2 \wedge \ldots \wedge x_n$
  Draw data set from $EX(c,\mathcal{D})$
- For each pos ex $x$ in data set,
  if $x_i = 0$, remove $x_i$ from $h$. (elim alg.)
- Works! CCHF thm, given large enough data set.

This will fail badly under $EX^{\eta}(c,\mathcal{D})$: terrible idea to "trust" any given "pos. ex." in data set (might be noisy...)

Hi level idea of fix: use statistical properties of whole data set.

---

① • New version of alg, uses only $EX(c,\mathcal{D})$, works w/ no noise

② • adapt ① to setting of $EX^{\eta}(c,\mathcal{D})$.

---

① : Consider (for $i \in [n]$)

$$p_i := \Pr_{(x,c(x)) \sim EX(c,\mathcal{D})}[c(x)=1 \wedge x_i = 0]$$

(Note: if $x_i$ in target $c(x)$, $p_i = 0$.)

Intuitively: if $x_i$ not in $c(x)$, having $x_i$ in hyp. $h$ contrib. $\leq p_i$ to $err_{\mathcal{D}}(h,c)$.

Goal: identify all $x_i$ s.t. $p_i \geq \varepsilon/n$.

If we cross all these off from $h$: tot. error of resulting $h$ would be $\leq n \cdot \frac{\varepsilon}{n} = \varepsilon$.

---

So... to learn mon conj, suff. to get a $\pm \frac{\varepsilon}{2n}$ acc. est. of each $p_i$, $i \in [n]$.

In noiseless setting (we have $EX(c,\mathcal{D})$):
easy! Draw $m$ ex., take our estimate $\hat{p}_i$ of $p_i$ to be:

$\log(1/\delta)/\tau^2$

(frac. of the $m$ ex. s.t. $c(x)=1$, $x_i=0$)  $\tau = \frac{\varepsilon}{10n}$, $\delta' = \frac{\delta}{n}$

CB: take $m = O\left(\left(\frac{n}{\varepsilon}\right)^2 \log\left(\frac{n}{\delta}\right)\right)$ tosses:

for each $i$, $\Pr[\text{est wrong by} > \frac{\varepsilon}{10n}] \le \frac{\delta}{n}$

$\forall$ good $\smile$

---

②: Goal: as before, est. $p_i := \Pr_{(x,c(x))\sim EX(c,\vartheta)}[c(x)=1 \wedge x_i=0]$

(noiseless prob!), but we only have $EX^\eta(c,\vartheta)$.

$\Pr[A \wedge B] = \Pr[A|B] \cdot \Pr[B]$

Hmm... well, $p_i = \Pr_{(x,c(x))\sim EX(c,\vartheta)}[c(x)=1 \mid x_i=0] \cdot \Pr_{(x,c(x))\sim EX(c,\vartheta)}[x_i=0]$.

$= q_i$

- $\Pr_{(x,c(x))\sim EX(c,\vartheta)}[x_i=0]$ : event $x_i=0$ unaffected by noise $\smile$

  $= \Pr_{(x,c(x))\sim EX^\eta(c,\vartheta)}[x_i=0]$

  So can est. easily, given noisy data $\smile$

- If $\Pr_{(x,c(x))\sim EX(c,\vartheta)}[x_i=0]$ close to 0: good $\smile$ : $p_i$ also $\approx 0$. ($p_i \le$ this).

  So can assume that not too small; say, $\ge \frac{\varepsilon}{4n}$.

  Remaining goal: estimate $q_i$.

  Consider "noisy" version of $q_i$:

since ✓, can estimate this without too much slowdown!

$$\bigstar = \Pr_{(x,b)\sim EX^{\hat{}}(c,\eta)}[\, b=1 \mid x_i=0\,]$$

c=1, no noise  c=0, noise

$$= q_i \cdot (1-\eta) + (1-q_i)\cdot \eta$$

$$= \eta + q_i(1-2\eta)$$

So $q_i = \dfrac{\bigstar - \eta}{1-2\eta}$ : Know est. of ★, Know $\eta$, so can est. $q_i$. ☺

---

What happened?

- Convinced ourselves: to learn, enough to est. some probs. ← noiseless
- To est. desired prob. in noisy world: re-expressed ↓ in terms of
  - part unaffected by noise
  - | predictably affected ) .

---

We'll do this more generally.

First, generalize

---

# New Learning Model:

# SQ
# STATISTICAL QUERY MODEL

(in ↑ of itself... no noise in picture)

**SQ model:**
- no indiv. examples given to alg.
- alg. can estimate probabilities (of noiseless data).

**Def:** A predicate of a labeled ex $(x,y)$ ($x \in X$, $y \in \{0,1\}$): a statement about $(x,y)$ that's T or F.

$X = \{0,1\}^n$

Ex: "$(x,y)$ has $x_1 = 1$, $x_2 = 0$, & $y = 0$"

Formally, a pred. is a fn $\chi : X \times \{0,1\} \to \{0,1\}$

Given a dist. $\mathcal{D}$ over $X$, a concept $c$, & a pred. $\chi$, we write

$$P_\chi := \Pr_{(x,c(x)) \sim EX(c,\mathcal{D})}[\chi(x,c(x)) = 1]$$

---

**Def:** The oracle $STAT(c,\mathcal{D})$ takes 2 inputs from learner:
- a pred. $\chi$, &
- a tolerance parameter $\tau > 0$.

$STAT(c,\mathcal{D})$ returns an est. $\hat{P}_\chi$ of $P_\chi$ s.t.

$$|\hat{P}_x - P_x| \le \tau.$$

---

Ex: Learner calls $STAT(c, \mathcal{D})$ with
$x =$ "$x_1 = 1 \wedge y = 0$", $\tau = 0.01$.

STAT returns 0.36: means

$$0.35 \le \Pr_{(x, c(x)) \sim EX(c, \mathcal{D})}[c(x) = 0 \wedge x_1 = 1] \le 0.37$$

---

Note: if had $EX(c, \mathcal{D})$:
can simulate having $STAT(c, \mathcal{D})$:

- given $x, \tau$ (input to ↗):
  - draw $m$ ex $(x, c(x)) \sim EX(c, \mathcal{D})$.
  - evaluate $x$ on each
  - output $\hat{P}_x$ = frac. of the $m$ ex. that satisfy $x$.

Taking $m = O(\log(1/\delta) \cdot \frac{1}{\tau^2})$:

$$\Pr[|\hat{P}_x - P_x| > \tau] \le \delta \quad (CB).$$

---

Cost of simulation: $\approx \frac{1}{\tau^2}$

involve cxity of evaluating $x$ on $(x, c(x))$

So eff. SQ alg: doesn't call $STAT(c, \mathcal{D})$ with $\tau$'s that are too small, or with $\chi$'s that are too expensive to compute.

---

**Def**: Concept class $\mathcal{C}$ is **SQ-learnable** if there's a learning alg $L$ s.t.:

- $\forall c \in \mathcal{C}$,
- $\forall \varepsilon > 0$,
- $\forall$ dist $\mathcal{D}$,

if $L$ is given access to $STAT(c, \mathcal{D})$, then $L$ outputs an $h$ s.t. $err_{\mathcal{D}}(h, c) \leq \varepsilon$.

---

We say $\mathcal{C}$ is **efficiently SQ learnable by $L$** if

- for every query $(\chi, \tau)$ that $L$ makes, $\tau \geq \frac{1}{poly(n, \frac{1}{\varepsilon}, size(c))}$ & $\chi(x, y)$ is evaluable in time $poly(n, \frac{1}{\varepsilon}, size(c))$;

- $L$ runs in time $poly(n, \frac{1}{\varepsilon}, size(c))$, viewing each call to STAT as taking one time step.

---

We have:

---

**Thm**: Let $\mathcal{C}$ be eff. SQ learnable.
Then $\mathcal{C}$ is eff. PAC learnable.

---

## Interesting thing (next time):

**Thm:** Let $\mathcal{C}$ be eff. SQ learnable.

Then $\mathcal{C}$ is eff. PAC learnable even with RCN.