Learning with Classification Noise
The Model

Same as PAC, but now we see \( \langle x, \tilde{y} \rangle \), \( x \in P \) and:

\[
\tilde{y} = \begin{cases} 
    c(x) \text{ with prob. } 1-\eta \\
    \neg c(x) \text{ with prob. } \eta
\end{cases}
\]

\( \eta \in [0, \frac{1}{2}) \) noise rate

- Goal: Still want \( \mathbb{E}(h) = P[h(x) \neq c(x)] \leq \delta \)
Say \( \mathcal{C} \) is \( \text{CN-learnable} \) by \( \mathcal{H} \) if PAC criteria met by algo using noisy \( y \)'s.

- What \( \mathcal{C} \) are \( \text{CN-learnable} \)?
- How much harder is \( \text{CN} \)?
- Are there "general principles" for \( \text{CN} \)?

**Note:** Can't use consistency any more...
An Observation

\[ Pr \left[ h(x) = y \right] \geq \mathbb{I} \frac{3}{2} \frac{\pi}{2} \left( \frac{h(y)}{3} \right) \]

= \left( 1 - \pi \right) + \pi \left( \frac{3}{2} - 1 \right) \left( \frac{h(y)}{3} \right)

\[ \geq \pi + \frac{\pi}{2} \left( \frac{3}{2} - 1 \right) \left( \frac{h(y)}{3} \right) \]

fixed \quad \text{increasing with } \frac{h(y)}{3}
ordering by $\bar{E}(h) \equiv E(h)$ ordering by $E(h)$

$\bar{E}(h)$ minimization $\equiv E(h)$ minimization

Note: $\bar{E}(h_1) - \bar{E}(h_2) = (1-2\pi)(E(h_1) - E(h_2))$

shrinks need resolution $\bar{E}$

$$\Rightarrow \text{sample sizes } \geq \frac{1}{(1-2\pi)^3}$$
CN learning (monotone) conjuncts

Original algo:
- start with \( h = x_1x_2 \cdots x_n \)
- \( \langle x_i \rangle \rightarrow \text{delete any } x_i \)

\( \text{Won't work on } \hat{y} \)

Algo too brittle/sensitive.

Define \( \forall x_i : \)
\( p_0(x_i) = P[x_i = 0] \)
\( \geq \) \( p_{01}(x_i) = P[x_i = 0, y = 1] \)

\( \text{true label } c(x) \)
Say \( x_i \) is significant: \( \text{pol}(x_i) \geq \frac{\varepsilon}{8n} \)

harmful: \( \text{pol}_1(x_i) \geq \frac{\varepsilon}{8n} \)

Let \( h \) contain all \( x_i \) that are sig. but not harmful.

\[
\Pr[h(x) = 1, y = 0]: \text{must be some } x_i \in C, x_i \notin h, x_i = 0
\]

\[
\text{not harmful} \Rightarrow \text{not sig.} \Rightarrow \text{pol}(x_i) \leq \frac{\varepsilon}{8n}
\]

\[
\therefore \Pr[h(x) = 1, y = 0] \leq n\left(\frac{\varepsilon}{8n}\right) = \frac{\varepsilon}{8}\]
\[ Pr[h(x) = 0, y = 1] \text{ must be some } x_i \in \text{c}, x_i \in \text{e}h, x_i = 0 \]
\[ p_0(x_i) \leq \frac{3}{8n} \]
\[ \therefore Pr[h(x) = 0, y = 1] \leq \left( \frac{3}{8n} \right)^n \]
\[ \frac{3}{8} \times \frac{3}{8} = \frac{9}{64} \]

\[ \therefore \exists (h) \leq \frac{3}{8} + \frac{9}{64} = \frac{1}{4}. \]

The \( p_0(x_i) \) are easy to estimate from \( x \in P \).

How can we estimate the \( p_0(x_i) \) from \( \langle x, \tilde{y} \rangle \)?
The Statistical Query (SQ) Learning Model

PAC:

\[ \langle x_i, y_i \rangle \rightarrow L \rightarrow h \in \mathcal{H} \]

SQ:

\[ \langle x_i, y_i \rangle \rightarrow SQ \rightarrow \left( \frac{x_i}{a_i}, \frac{x_i}{a_i}, \ldots \right) \rightarrow L \rightarrow h \in \mathcal{H} \]

“Statistical queries”
Queries are predicates $X$ over $\langle x, y \rangle$ (no noise):

$X(x, y) \in \{0, 1\}$

E.g., $X(x, y) = 1 \iff x_5 = 1, y = 0, (po_i(x_5))$

$X(x, y) = 1 \iff x_1 \leq 1.7, x_2 \geq 2.9, y = 1$

Assume $X$ "reasonable"/computable
View $X(x, y)$ as a request for

$$P_x \triangleq Pr[X | x, y] = 1$$

But SQ won't answer exactly, but with tolerance $\varepsilon$:

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

Full query: $X, \varepsilon$

Get back: $\hat{P}_x$
Say $C$ is $SQ$-learnable by $H$ if there exists an algorithm $L$ that for all $C \subseteq \mathcal{C}$, $\forall \varepsilon, \forall \mathcal{P}, \forall \Delta > 0$:

- $L$ makes $\text{poly}(\frac{1}{\varepsilon}, \mathcal{P}, \text{size}(C))$ queries, each of tolerance $\Delta$.

  $$\varepsilon \geq \frac{1}{\text{poly}(\cdot)} \quad (\text{e.g. } \varepsilon \geq \frac{1}{n^3})$$

- $L$ runs in poly time.

- $L$ outputs $h \in H$ s.t. $\Pr[h(\cdot) \leq 3] \geq \frac{3}{4}$. 
**Theorem**

C SQ learnable by $H \Rightarrow$

C PAC learnable by $H$.

**Proof.** Let $L$ be SQ algo. On each query $X, \xi$ of $L$, sample $m < X, y >$ pairs to get estimate $\hat{P}_X$. If $m = \frac{1}{\xi^2} \log \left( \frac{2}{\delta} \right)$ then

w.p. $\geq 1 - \frac{\delta}{2}$

$\hat{P}_X \in P_X + [-\epsilon_2, \epsilon_2]$. Now let $l = \# \text{queries of } L$

$\leq \text{runtime of } L$. 

More Interesting Theorem

\[ C \text{ SQ learnable by } \mathcal{H} \implies C \text{ CN learnable by } \mathcal{H}. \]

How can we simulate \text{ SQ} from \text{ CN}?
Decomposing $P_x$

- $P_x = P_r [x(x, y) = 1]
- $\tilde{P}_x = P_r [x(x, \tilde{y}) = 1]

Can estimate directly.

Now define disjoint $X_1, X_2$

$X_1 \cap X_2 = \emptyset, X_1 \cup X_2 = X$:

$X_1 = \{ x \in X : x(x, 0) \neq x(x, 1) \}$

$X_2 = X - X_1$

Can test $x$ for membership.
Let \( p_1 = P[\mathbf{x}_1], \ p_2 = P[\mathbf{x}_2] \)
\( p_2 = 1 - p_1 \)

*Induced distributions:*

\( P_1 \) over \( \mathbf{X}_1 \): \( P_1[\mathbf{x}] = P[\mathbf{x}]/p_1 \)

\( P_2 \) over \( \mathbf{X}_2 \): \( P_2[\mathbf{x}] = P[\mathbf{x}]/p_2 \)

*And finally:*

\( P_1' = P_1[\mathbf{x} | \mathbf{x}_0, y = 1] \)

\( P_2' = P_2[\mathbf{x} | \mathbf{x}_0, y = 1] \)
Note: The following can be estimated from $\langle x, y \rangle$:

$\hat{P}_x, P_x^2, P_1, P_2$

Goal: formula where

$P_x = \text{some function of } \hat{P}_x, P_x^2, P_1, P_2, \pi$
$P_x = (1 - \hat{y}) P_x +$

$\hat{y} = \begin{cases} 
1 & \text{if } p, p_1 \{x (x, \hat{y}) = 1 \} \\
0 & \text{otherwise}
\end{cases}$

$\hat{y} = \hat{y}$

$+ p_2 \ P_2 \{x (x, \hat{y}) = 1 \}$
\[
\tilde{P}_x = (1 - \theta) P_x + \theta (p, P, [x, x, y] = 1) + p_2 P_2 \]
\[
\tilde{P}_x = (1 - \eta) P_x + \eta \left( p_1 P_1 \left[ x_1(x, y) = 1 \right] + p_2 P_2 \left[ x_1(x, y) = 1 \right] \right)
\]

\[
= p_1 \left[ x_1(x, y) = 0 \right]
\]

\[
= 1 - P_x'
\]
\[ \tilde{P}_x = (1 - \bar{\eta}) P_x + \eta \left( p \cdot Q_1 \left[ x \mid (x, \bar{y}) = 1 \right] + p_2 \cdot Q_2 \left[ x \mid (x, \bar{y}) = 1 \right] \right) \]

\[ = P_2 \left[ x \mid (x, \bar{y}) = 1 \right] \]

\[ = \bar{p} \cdot P_x \]
\[ \hat{P}_x = (1-\eta) P_x + \eta (p, P_{x|x,y}^1) \]
\[ \hat{y} = y + p_2 P_{x|x,y}^1 \]
\[ \hat{y} = y \]
\[ \Rightarrow (1-\eta) (p_1 P_{x|x} + p_2 P_{x|x}^2) \]
\[ + \eta (p_1 (1-P_{x|x}^1) + p_2 P_{x|x}^2) \]
\[ P_x = (1 - \hat{\eta}) P_x + \eta (p_1 P_x' + p_2 P_x^2) \]

Now solve for \( P_x' \)
\[ P_x' = \frac{\hat{P}_x - p_2 P_x^2 - n \hat{P}_1}{(1-2\pi)p_1} \]

Plug into \( P_x = p_1 P_x' + p_2 P_x^2 \):

\[ P_x = \frac{\hat{P}_x - p_2 P_x^2 - n \hat{P}_1}{1-2\pi} + p_2 P_x^2 \]

\[ = \frac{\hat{P}_x}{1-2\pi} + (1 - \frac{1}{1-2\pi}) p_2 P_x^2 \]
Sensitivity Analysis (informal)

If we estimate \( \hat{P}_x \) within \( \pm \rho \), error

\[
\ln \frac{\hat{P}_x}{1-2\eta} \quad \text{is} \quad \pm \frac{\sigma}{1-2\eta}
\]

So better make

\[
\frac{\rho}{1-2\eta} < 2, \quad \rho > \frac{\eta(1-2\eta)}{2}
\]
Sensitivity Analysis (informal)

If error in $p_z$ is $\pm a$, $P_x$ is $\pm b$, error in $p_z P_x \leq a + b + a b$, make $a, b < \frac{1}{2} (1 - 2n)$ etc.
Guessing $N$ (informal)

- Suppose we are only given an upper bound: $\eta \leq \hat{\eta} < 1/2$
- Divide $[0, \hat{\eta}]$ into values $0, \Delta, 2\Delta, \ldots, \Delta \frac{\eta}{\hat{\eta}}$
• Run simulation for each guess \( \Pi = \Pi \cdot \Delta \)

• Verify hypotheses has \( E(y) \leq 3 \)

• One guess will be \( \pm \Delta \) of true \( \Pi \)

• Make \( \Delta \) small wrt \( \tau, \delta, \frac{1}{1-2\pi} \)

• Poly. overhead
Say $x_i$ is significant: $p(x_i) \geq \varepsilon/8n$ 

harmful: $p_1(x_i) \geq \varepsilon/8n$

Let $h$ contain all $x_i$ that are sig. but not harmful.

Pr [$h(x) = 1, y = 0$]: must be some $x_i \in C, x_i \notin h, x_i = 0$

not harmful $\Rightarrow$ not sig. $\Rightarrow p(x_i) \leq \varepsilon/8n$

$\therefore$ Pr [$h(x) = 1, y = 0$] $\leq n(\varepsilon/8n) = \varepsilon/8$

estimate all these (non-adaptive)
Claim:

"Almost" all \( C \) in PAC are also in SQ (with a different algo), and thus in CN.

- Conjunctions,
- Rectangles,
- Decision lists,
- Linear class, etc.
Claim

- Virtually every "practical" ML algo has an SQ variant.
- Backprop, dec. tree algo, gradient des., boosting, etc.
So what's hard in SQ?
Parity Functions

- $X = \{0, 1\}^n$
- For set $s \subseteq \{1, 2, \ldots, n\}$ define:

$$f_s(x) = \sum_{i \in s} x_i \mod 2$$

$$= \bigoplus_{i \in s} x_i$$

- So $f_s(x) = 1 \iff \# \text{bits} = 1$ in $x$ if $s$ is odd.

- Let $\text{PARITY}$ denote the class of all $(2^n)$ $f_s$. 
**PARITY is PAC-learnable**

- Note: \( f_s(x) = \sum x_i \mod 2 \)
- where \( x, y \in \{0, 1\}^n \) and \( x_i = 1 \iff i \in S \)
- Inner product \( \alpha \cdot x \) in \( \text{GF}^n(2) \)
- Given \( \langle x_1, y_1 \rangle, \ldots, \langle x_m, y_m \rangle \):

\[
\begin{bmatrix}
-x_1 \\
-x_2 \\
\vdots \\
-x_m
\end{bmatrix}
\begin{bmatrix}
\alpha_1 \\
\alpha_2 \\
\vdots \\
\alpha_n
\end{bmatrix}
= 
\begin{bmatrix}
y_1 \\
y_2 \\
\vdots \\
y_m
\end{bmatrix}
\]

Solve sys. of linear eqns!
Why is PARITY hard for SQ?

• Let $P$ be uniform on $\{0,1\}^n$

• Look at $X$’s of form:
  $$X(x,y) = 1 \iff x(T = z)$$
  and $y = 1$

• Draw $S \subseteq \{0,1\}^3$ randomly

• If $T$ is small ($|T| \leq n/2$) then whp over draw of $S$,
  $$S \cap T \neq \emptyset$$
  and thus
  $$P_T[\chi(x, f_s(x)) = 1] = \frac{1}{2}$$

• "no information" conveyed
Theorem. For \( \varepsilon = \frac{1}{2} \), and any polynomial \( p(n) \), the number of queries of tolerance \( \varepsilon = \frac{1}{p(n)} \) required to learn \( \text{PARITY} \) is exponential in \( n \).
So $\text{PARITY} \in \text{PAC}$
but $\text{PARITY} \notin \text{SQ}$.

Is $\text{PARITY} \in \text{CN}$?

Open problem,
but believed hard.

- "Nearby" NP-hardness
- Crypto proposals
Query complexity in SQ

- target class $C$, dist $P$
- view $f \in C$ as $\pm 1$-valued
- $\forall f_i, f_j \in C$, inner product:

$$<f_i, f_j>_P = \sum_x P(x)f_i(x)f_j(x)$$

$$= P[f_i = f_j] - P[f_i \neq f_j]$$

- For every $f, f' \in PARITY$, $f \neq f' \Rightarrow <f, f'>_P = 0$

where $P = \text{uniform}$.
Define

\[ \text{SQL}(C, P) = \max d \text{ s.t. } \]

\[ \forall f_i, f_j \in C, \forall 1 \leq i \neq j \leq d: \]

\[ \langle f_i, f_j \rangle_p \leq \frac{1}{d^3} \]

\text{Largest # of "almost orthogonal" fns/vectors}

in \( C \) w.r.t. \( P \)

\[ \text{SQL(PARITY, uniform)} = 2^n \]
Theorem. Let $\text{SQ}(C, P) = d$. Then at least $d^{1/3}/2$ queries with $\tau \geq \frac{1}{d^{1/3}}$ are needed to learn $C$ wrt $P$ in SQ model. (even "weakly")

- $d$ superpoly in $n \Rightarrow$ superpoly # queries
An Application

• Consider “small” parities, i.e. $f_S$ where $|S| = \log(n)$

  E.g. parity of first $\log(n)$:

\[ x_1, x_2, \ldots, x^{\log(n)} \mapsto f_S \]

\[
\begin{array}{cccc}
0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & \cdots & 1 \\
0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & \cdots & 1 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & 1 \\
\end{array}
\]

\[ \text{only } 2^{\log(n)} = n \text{ rows} \]
Encode as decision tree:

Size = $n$ nodes
Encode as DNF:

\[
\begin{align*}
\left( x_1 \land x_2 \land \ldots \land x_{\log(n)-1} \land x_n \right) & \lor \\
\left( x_1 \land x_2 \land \ldots \land x_{\log(n)-1} \land x_n \right) & \lor \\
& \vdots \\
& \left( x_1 \land x_2 \land \ldots \land x_{\log(n)-1} \land x_n \right) & \text{all terms where } f_s = 1
\end{align*}
\]

size \leq n terms

But there are \((n \choose \log(n)) \cdot n^{\log(n)}\) such small \(f_s\).
So decision trees & DNF are not learnable in the SQ model. 

For informational, not computational, reasons. 
No assumptions. 

Status in PAC: open.
The PAC Solar System

- Status known
- (partially) open

All poly. eval.

SQ

CN

PAC

finite automata

Parity

conjunctions
rectangles
dec. lists
linear seps

DNF

DT

SQ ⊆ CN ⊆ PAC ⊆ poly. eval

neural nets
boolean formulac

finite automata
What about the *?

There are functions that are easy to compute but hard to invert.

Example: multiplication.

\[ f(p, q) = p \cdot q \]

Over domain \( p, q \) prime, \( f^{-1} \) is factorization.
In fact, there are large families of such functions.

If primes $p, q$ define

$$f_{p,q}(x) = x^2 \mod p \cdot q$$

Inverting for unknown $p \cdot q \equiv \text{factor in } p \cdot q$.

Encode as PAC learning neural nets, finite automata, boolean formulae...
Note: Hardness holds even for weak PAC learning, where

$$\exists = \frac{1}{2} - \frac{1}{\text{poly}(n)}$$

Does weak PAC \( \Rightarrow \) PAC?
One final twist...

- Let's make PAC more powerful by adding membership queries.

- In addition to \( \langle x, y \rangle, x \in P \), algo can get close for any chosen \( x \).

- Goal remains same
The PAC Solar System

- status known
- (partially) open

All poly. eval.

all

eval.

finite automata

PAC + MQ

PAC

CN

SQ

all

poly.

eval.

conjunctions
rectangles
decl. lists
linear seps

weird

parity

DNF

DT

poly. eval.

parity

PAC + MQ

PAC

CN

SQ

neural nets*

boolean formulac*

* PAC
Next Up:
Boosting and Beyond