Last time: - VCDIM(C): examples, l.b. on m.b. of any OLMB alg.
- Prediction with Expert Advice: WMA ← "noise-tolerant HA"

$$M \leq \frac{m \log 1/\beta + \log N}{\log \frac{2}{1+\beta}}$$

many mistakes when N experts, best expert makes m mistakes

Today: Randomized WMA "noise-tolerant RHA"
Start next unit:

Probably Approximately Correct (PAC) Learning

- motivation
- basic definition
- learning C = intervals of $\mathbb{R}$

Questions?

RWMA

Motiv.: consider WM on a trial where vote q_0 vs q_1 is 51% vs 49%

WM: predicts 0 for sure.

- Predictability : adv. can maybe exploit?

What if on WM predicted 1?
would be Almost as good...

Natural variant of WM in this setting:

- flip coin - ok...

What if 55 - 45 ?

flip biased coin?  .

RWM alg: As before, N expert pool.

- Initialize w_i (expert i's wt) to 1.
- At each trial, expert i predicts z_i.
- // // //
- RWM outputs z_i w. prob. $\frac{w_i}{W}$, $W = \sum_{i=1}^{N} w_i$
- Given correct value/outcome ℓ of trial, for each i s.t. $z_i \neq \ell$, set $w_i \leftarrow w_i \cdot \beta$.

(Note: z_i's need not be binary!)

Thm: Assume oblivious adv. model (i.e., seq of trial outcomes is fixed in advance before RWM starts). If best expert in pool makes m mistakes, then $\mathbb{E}[\#\text{mist. RWM makes}] = M$, where

$$M \le \frac{m \cdot \ln(1/\beta) + \ln N}{1-\beta}.$$

$\beta = \frac{1}{2}$: $1.39m + 2\ln N$

$\beta = 3/4$: $1.15m + 4\ln N$

$\beta = 1-\varepsilon$: $\varepsilon \to 0$, $m + \frac{1}{\varepsilon}\cdot \ln N$

Pf: Consider any seq of T trials.

Let F_i = frac. of tot. wt. at trial i that's on wrong prediction.

Hence $\Pr[\text{RWM alg makes mist. on trial } i] = F_i$.

Let $M = \Phi$; then $M = \sum_{i=1}^{T} F_i$.

Before i^{th} trial: tot wt $W = \underbrace{(1-F_i)W}_{\text{will be right; same}} + \underbrace{F_i W}_{\text{will be wrong; } \Phi \text{ by } \beta}$

After i^{th} trial, new W is $(1-F_i)W + \beta F_i W$.

$= (1-(1-\beta)F_i)W$.

Init. $w = N$; so final wt $w = N \cdot \prod_{i=1}^{T} (1-(1-\beta)F_i)$.

As before, best expert made m mist., so also

$W \geq$ wt of best expert $\geq \beta^m$. Done!

Here's why:

$$N \cdot \prod_{i=1}^{T} (1-(1-\beta)F_i) \geq \beta^m .$$

ln:

$$\ln N + \sum_{i=1}^{T} \ln\left(1-(1-\beta)F_i\right) \geq m \cdot \ln \beta$$

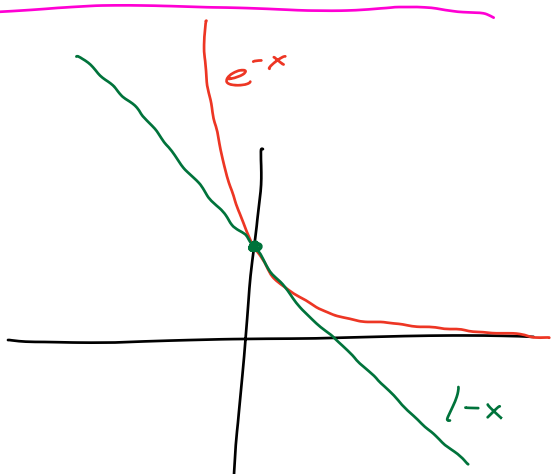
Negate:

$$-\ln N - \sum_{i=1}^{T} \ln\left(1-(1-\beta)F_i\right) \leq m \cdot \ln \tfrac{1}{\beta}$$

Recall from calculus:

$1-x \leq e^{-x}$ $\forall x$, hence

$\ln(1-x) \leq -x$, so $x \leq -\ln(1-x)$



Apply to each summand, with $x = (1-\beta)F_i$

$$-\ln N + (1-\beta)\sum_{i=1}^{T} F_i \leq m \ln \tfrac{1}{\beta}, \text{ so}$$

$$M = \sum_{i=1}^{T} F_i \leq \frac{m \ln 1/\beta + \ln N}{1-\beta}.$$ ■

---

next model: **PAC Learning**

Motivation:

Online learning has drawbacks:

- worst-case assump. on ex. seq. (like adversarial data)
- MB crit. : looks at whole process from beginning - too harsh?
- Guarantee may be unsatisfying: if haven't saturated m.b., no guarantee on next ex.

In PAC learning:
(independent identically distributed)

- Assume all examples $x \in X$ that learner gets are i.i.d. draws from some fixed, unknown, arbitrary distribution $\mathcal{D}$ over X.

[learn from random ex, not a seq. of adversarially chosen examples]

- "Batch" model: train alg. on whole data set of $(x, c(x))$ pairs, each $x \sim \mathcal{D}$ i.i.d. Alg. outputs just one h based on .

- Performance guarantee: should "do well" on future $x \sim \mathcal{D}$.

More details:

PAC framework for learning conc. class $\mathcal{C}$ using a hypoth. class $\mathcal{H}$:

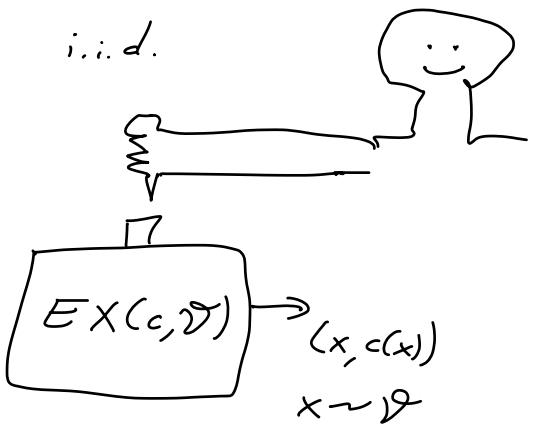
Like OLMB: unknown $c \in \mathcal{C}$ (target concept), $\mathcal{C}$ is known.

(new!) unknown dist $\mathcal{D}$ over X

Learner given a training set of m lab. ex.

$(x, c(x))$, each $x \sim \mathcal{D}$ i.i.d.

Equivalent POV:



learner given access to

presses button m times.

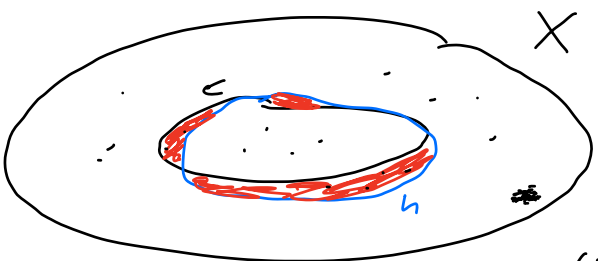
Learner does computation on m data pts, & outputs some

$$h \in \mathcal{H}, \quad h : X \to \{0,1\}.$$

Def: Let $h, c : X \to \{0,1\}$, let $\mathcal{D}$ be a distrib. on X.

The <u>error of h on c under $\mathcal{D}$</u> is

$$er_{\mathcal{D}}(h,c) := \Pr_{x \sim \mathcal{D}}[h(x) \neq c(x)].$$



$$er_{\mathcal{D}}(h,c) = \Pr_{x \sim \mathcal{D}}[x \in \text{[redacted]}]$$

Note: • Might not be poss. to achieve $er_{\mathcal{D}}(h,c) = 0$: could be very small prob. wt on some ex... ε

• Also can't expect <u>100%</u> guarantee δ even of moderate error: some chance of non-representative sample.

We <u>can</u> expect: low error with high confidence.

Success in PAC setting.

Def (Prelim. def.) "Algorithm A PAC learns $\mathcal{C}$ using $\mathcal{H}$" means:

- $\forall\, c \in \mathcal{C}$,
- $\forall$ dist $\mathcal{D}$ over X,
- $\forall\, \varepsilon, \delta > 0$ (accuracy, confidence),

over the m calls to $EX(c, \mathcal{D})$, $\forall$ any internal rand. of A

if A is given ε, δ, & access to $EX(c, \mathcal{D})$, then with prob. $\geq 1-\delta$ A outputs a hyp $h \in \mathcal{H}$ s.t.

$$err_{\mathcal{D}}[h, c] \leq \varepsilon.$$

Some notes:

- ε = error/acc param, δ = confidence param
- $m = \#$ ex. = sample complexity.

  $m = m(\varepsilon, \delta)$

- Running time: an efficient PAC learner runs in $poly(1/\varepsilon, 1/\delta)$. If $X = \{0,1\}^n$ or $\mathbb{R}^n$ or $[0,1]^n$,

should be $\text{poly}(n, \frac{1}{\varepsilon}, \frac{1}{\delta})$.

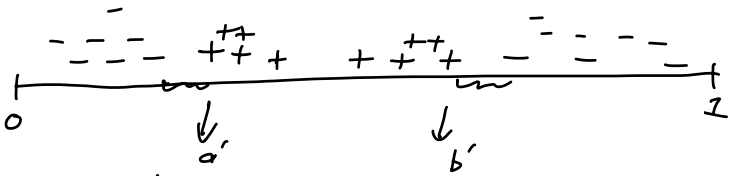
- If $\mathcal{H} = \mathcal{C}$, we say the alg is a proper learner.

Ex: learning $\mathcal{C}$ = intervals of $[0,1]$.

$X = [0,1]$ e.g. $c = [0.03, \frac{1}{\pi}]$

Target = $[a,b]$

The alg:



- draw m examples
- let a' = leftmost pos ex
  b' = rightmost " "
- output $[a', b'] = h$.

Next time: analyze this.
