Online Learning ; Mistake-bounded Learning

- So far we've assumed that there is a distribution over some instance space X, and the learner gets data from such a distribution.
- This is often not realistic. Data typically does not come from a stationary distribution
- Is there a "meaningful" model of learning which doesn't require any fixed distribution.
- We will think of an online/sequential learning framework.
  We have discrete time-steps
  (a) At time t an example $x_t \in X$ is given to the learning algorithm.
  (b) Learning algorithm predicts $\hat{y}_t \in \{0,1\}$
  (using all possible historical data)
  (c) The true label $y_t \in \{0,1\}$ is revealed..

what's the notion of performance?

For time t, $\text{MISTAKES}(t) = \sum_{s=1}^{t} \mathbb{1}(\hat{y}_s \neq y_s)$.

#MISTAKES remains finite even as $t \to \infty$.

---

§ Assume that there is a class of functions C from $X \to \{0,1\}$ and that $\exists\, c \in C$, s.t. $y_t = c(x_t)$ $\forall\, t$.

§ Formally, at time t

$$(x_1, y_1, \ldots, x_{t-1}, y_{t-1}, x_t) \xrightarrow{\text{Learner}} \hat{y}_t .$$

Learning CONJUNCTIONS,

- start with $h_0 \equiv x_1 \wedge \bar{x}_1 \wedge x_2 \wedge \bar{x}_2 \ldots \wedge x_n \wedge \bar{x}_n$
- For $t = 1, 2, \ldots,$

```
ŷ_t = h_{t-1}(x_t)
if (y_t = 1 and ŷ_t = 0) {
    remove all literals that were false/0 in x_t
}
```

Theorem : $\forall t$ MISTAKES$(t) \leq 2n$

Proof: Suppose c is the target conjunction.

Maintain the invariant that $c(x)=0 \Rightarrow h_t(x)=0 \quad \forall t$

Every time we make a mistake, we remove at least one literal from h_t.

literals $\leq 2n$. ∎

MISTAKES $\leq n+1$.

Exercise: Show that $n+1$ is tight. i.e. $\exists\, c$, and a sequence $x_1, x_2, \ldots$ s.t. ~~any~~ this algorithm would make at least $n+1$ mistakes

---

Def: MISTAKE-BOUNDED LEARNING: We will say that a concept class C is learnable with a mistake bound M, if there exists a learning algorithm L, that when input any sequence $(x_t, y_t)_{t=1}^{\infty}$ satisfies $\forall t$, $\text{MISTAKES}(L, t) \leq M$.

Efficiency:

- Most generous: At time t, running time is $\text{poly}(t, n, \text{size}(c))$ where n is the size of examples in X_n. & $\text{size}(c)$ is the rep. size of the target.
- Space-bounded algorithms: Learning algorithm keeps some "sketch" S_t of data up to time t, $|S_t| \leq \text{poly}(n, \text{size}(c))$.
- Want that $S_{t+1} = f(S_t, x_t, y_t)$ and f is efficiently computable.
- Want that $\hat{y}_t = g(S_t, x_t)$ and g is efficiently computable.

PAC model: $EX(c, D)$ - example oracle

MQ (Membership Queries): $x \longrightarrow$ receive $c(x)$ x is chosen by L

EQ (Equivalence Queries): L produces h, $\longrightarrow$ get success if $h \equiv c$ or counterexample x s.t. $h(x) \neq c(x)$.

- Mistake bound of M, implies that C is learnable using at most $M+1$ equivalence queries only
- $g(S_t, x_t) = \hat{y}_t$: $g(S_t, \cdot) \equiv h$ (hypothesis)
-

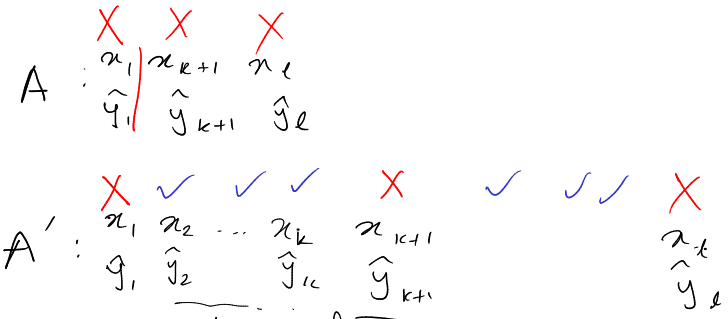
If C is learnable using q equivalence queries only then, C is learnable with mistake bound of at most q.

- Simulate L that only uses EQs.
  - at any point it has a hypothesis h.

Defn: Conservative Online Learner: We say that an online learning algorithm is conservative if it only changes its prediction rule after making a mistake.

Lemma: If A learns a concept class C with mistake-bound M, then there exists a conservative algorithm A' which also learns C with mistake bound M.

Proof:



A' copies A's behaviour.
However it only further simulation of A if a mistake occurs

A' is clearly conservative.
mistakes made by A' = # mistakes made by A

∎

Theorem: If C is (efficiently) online learnable with mistake bound M, where $M \leq \text{poly}(\text{size}(c), n)$, then C is (efficiently) PAC learnable.

Proof: Feed examples drawn from $EX(c, D)$ to the online algorithm (conservative).

h_1 ... h_2 ... h_3 ... ... h_M

if we go for $\frac{1}{\varepsilon}\log(M/\delta)$ examples without making a mistake, then the h_t being used on this run has $err(h_t) \leq \varepsilon$ w.p. $1-\delta$.

if $err(h_b) > \varepsilon$, then $\mathbb{P}$(no mistake seen in $\frac{1}{\varepsilon}\log\left(\frac{M}{\delta}\right)$ examples) $\leq \frac{\delta}{M}$.

Alg will always output one of $h_1, \dots, h_M$, over all failure probs $\leq M \cdot \frac{\delta}{M} = \delta$.

General Upper Bound:

- Let C be a finite concept class, then the HALVING ALGORITHM has MISTAKE bound at most $\log|C|$.

HALVING ALGORITHM:

- $C_1 = C$

  for $t = 1, 2, \dots$

  - Given x_t, computes $c(x_t)\ \forall\ c \in C_t$ and sets $\hat{y}_t$ to be the majority of $(c(x_t) \mid c \in C_t)$.
  - if $\hat{y}_t \neq \hat{y}_t$, $C_{t+1} = C_t \setminus \{c \in C_t \text{ s.t. } c(x_t) = \hat{y}_t\}$.

    else $C_{t+1} = C_t$

- After every mistake $|C_{t+1}| \leq |C_t|/2$.
- $\therefore$ #MISTAKES $\leq \log_2 |C|$.

Lower bound: VC-DIM $(C) \leq \max_t$ MISTAKES (t).

Examples: $X = \{(1,0,\dots,0), (0,1,\dots,0), \dots, (0,0,\dots,0,1)\}$.

C = DICTATORS.

$c_i(\underline{x}) = x_i$.

MISTAKE-BOUND(HALVING ALG) = 1

Exercise: $X = \{0,1\}^n$, then mistake bound is $\lfloor \log n \rfloor$ for DICTATORS.

Examples: Binary Search.

$X = \{1, \dots, 2^n\} \subseteq \mathbb{N}$.

C is half intervals: $c_j(x) = \begin{cases} 1 & \text{if } x \geq j \\ 0 & \text{o/w} \end{cases}$

VC$(C) = 1$.

MISTAKE-BOUND $\approx n = \log|C|$.

$err_D(h) \geq \varepsilon$

$X \sim D \qquad P(h(x) \neq c(x)) \geq \varepsilon$

$x_i \sim D$, $\quad z_i = 1$ if $h(x_i) \neq c(x_i)$

$= 0$ o/w.

$x_1, \ldots, x_n, \qquad P\left(\sum_i z_i = 0\right) = \prod_{i=1}^{n} P(z_i = 0)$ $\qquad 1 - \varepsilon \leq e^{-\varepsilon}$

$\leq (1-\varepsilon)^n \leq e^{-\varepsilon n} \leq \frac{\delta}{M}$.

$n = \frac{1}{\varepsilon} \log\left(\frac{M}{\delta}\right)$.