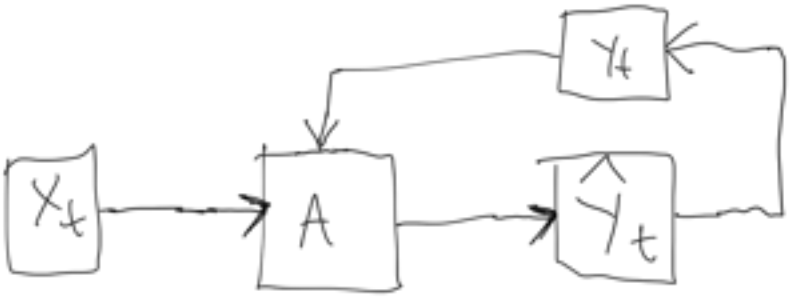
Online learning

- So far we have considered algorithms that process training data all at once.

- We consider online learning where predictions need to be done on-the-fly.



- Online learning protocol

For $t = 1, 2, 3, \ldots$

1) Algorithm receives feature vector $x_t \in X$

2) Algorithm makes a prediction

$\hat{y}_t \in Y$

$(Y = \{0,1\}$ or $Y = \{-1,+1\})$

3) Algorithm receives correct label $y_t \in Y$

- Algorithm makes a mistake in round t iff $\hat{y}_t \neq y_t$.
- The goal of the algorithm is to make as few mistakes as possible.
- We will make various assumptions on the sequence $(x_1, y_1), (x_2, y_2), \ldots$

Littlestone's model

- Online learning version of PAC model
- There exists $H \subseteq Y^X$ known to the algorithm

- There is a target function $h^* \in H$ such that
$y_1 = h^*(x_1)$, $y_2 = h^*(x_2)$, ...

- We do NOT make any probabilistic or statistical assumptions !

Halving algorithm

- Suppose H is finite.

- We know that h^* belongs to H.

- Maintain set $G \subseteq H$ of predictors that do not contradict any (x_t, y_t)

- Initially $G := H$

- When $h \in G$ disagrees

with (x_t, y_t) we remove h from G

Halving algorithm

- Initialize $G_1 = H$
- For $t = 1, 2, \ldots$
  1) Receive $x_t \in X$
  2) Split G_t into

$$G_t^0 = \{ h \in G_t : h(x_t) = 0 \}$$

$$G_t^1 = \{ h \in G_t : h(x_t) = 1 \}$$

  3) Predict

$$\hat{y}_t = \begin{cases} 1 & \text{if } |G_t^1| > |G_t^0| \\ 0 & \text{if } |G_t^0| \geq |G_t^1| \end{cases}$$

  4) Receive $y_t \in \{0, 1\}$

  5) Remove incosistent classifiers:

$$G_{t+1} = G_t^{Y_t}$$

Theorem: Halving algorithm makes at most $\lfloor \log_2 |H| \rfloor$ mistakes.

Proof:

- Clearly

$$G_1 \supseteq G_2 \supseteq G_3 \supseteq \cdots$$

- when $|G_t| = 1$ the algorithm cannot make mistakes anymore.

- If the algorithm makes a mistake in round t then $|G_{t+1}| \leq \frac{|G_t|}{2}$.

- $G_{t+1} = G_t^{Y_t}$

- $G_t^{Y_t} = G_t^{1-\hat{Y}_t}$

- $|G_t^{\hat{y}_t}| \geq |G_t^{1-\hat{y}_t}| = |G_t^{y_t}|$

  ↑ This how the algorithm chooses $\hat{y}_t$!

- $|G_t| = |G_t^0| + |G_t^1|$

  $= |G_t^{1-\hat{y}_t}| + |G_t^{\hat{y}_t}|$

  $\geq 2|G_t^{1-\hat{y}_t}|$

  $= 2|G_t^{y_t}|$

  $= 2|G_{t+1}|$

- That implies $|G_{t+1}| \leq \frac{|G_t|}{2}$.

- The number of halvings can be at most $L \log_2 |H|$.

∎

Littlestone's dimension

- Analogous to Vapnik-Chervonenkis dimension
- Let $M(A, h^*, x_1, x_2 \dots)$ be the number of mistakes
- $\text{Ldim}(H) = \min_{A} \max_{h^* \in H} \max_{x_1, x_2 \dots} M(A, h^*, x_1, x_2 \dots)$

best algorithm

worst target

worst input sequence

- It is analogous to the optimal worst-case time complexity for a problem.

(E.g. comparison-based sorting has optimal worst-case complexity $\Theta(N \log N)$.)

- Optimal algorithm
  - $G_1 = H$
  - For $t = 1, 2, \dots$
    1) Receive $x_t \in X$

2) Split G_t into

$$G_t^0 = \{h \in G_t : h(x_t) = 0\}$$
$$G_t^1 = \{h \in G_t : h(x_t) = 1\}$$

3) Predict $\hat{Y}_t \in \{0,1\}$

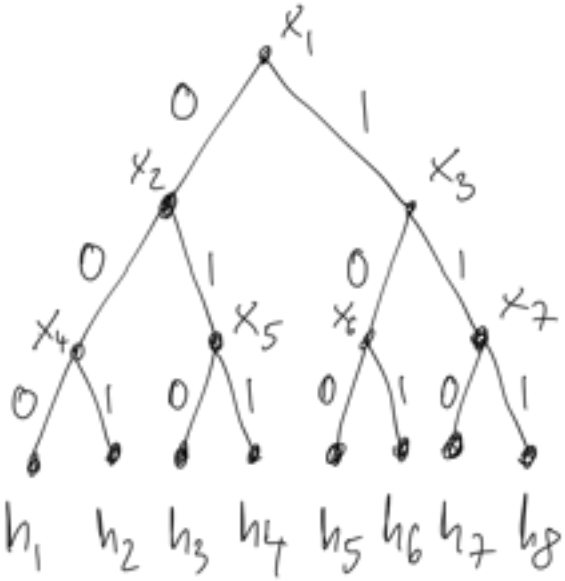
$$\hat{Y}_t = \begin{cases} 1 & \text{if } Ldim(G_t^1) > Ldim(G_t^0) \\ 0 & \text{if } Ldim(G_t^0) \geq Ldim(G_t^1) \end{cases}$$

4) Receive $Y_t \in \{0,1\}$

5) Update

$$G_{t+1} = G_t^{Y_t}$$

- Shattered tree



Complete binary tree

Where internal nodes are

labeled with elements of X (not necessarily distinct), left edges are labeled 0, right edges are labeled 1, and leaves are labeled by predictors from H such that root-to-leaf path is consistent with predictor in the leaf.

- Theorem:
  $Ldim(H)$ = depth of largest shattered tree.

Proof:

- Let $DST(H)$ = depth of largest shattered tree.
- We need to show that
  $$Ldim(H) = DST(H)$$
- For any $x \in X$ define

$$H_x = \{h \in H : \ h(x) = 0\}$$

$$H'_x = \{h \in H : \ h(x) = 1\}$$

- We claim that both Ldim and DST satisfy the same recurrence:

$$Ldim(\{h\}) = 0$$

$$DST(\{h\}) = 0$$

- Define

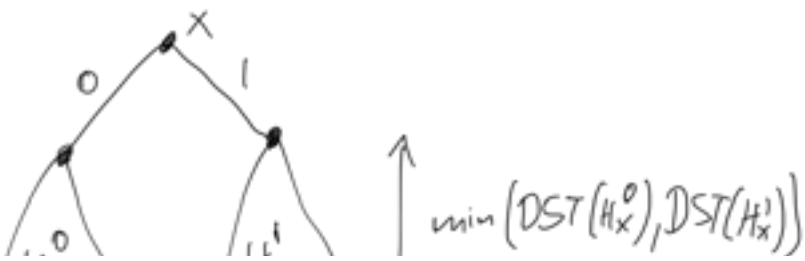
$$\phi(a,b) = \begin{cases} 1+a & \text{if } a = b \\ \max(a,b) & \text{if } a \neq b \end{cases}$$

$$Ldim(H) = \max_{x \in X} \left(\phi\left(Ldim(H_x^0), Ldim(H_x')\right)\right)$$

$$DST(H) = \max_{x \in X} \left(\phi\left(DST(H_x^0), DST(H_x')\right)\right)$$

- The recurrence for Ldim follows from the optimal algorithm

- The recurrence for DST follows from this picture:



Example #1 (Singletons):

- X is any non-empty set
- For any $x \in X$,

$$h_x(x') = \begin{cases} 1 & \text{if } x = x' \\ 0 & \text{otherwise} \end{cases}$$

- $H_{singleton} = \{ h_x : x \in X \}$
- $Ldim(H_{singleton}) = 1$
- Optimal algorithm predicts 0 all the time until it makes a mistake

---

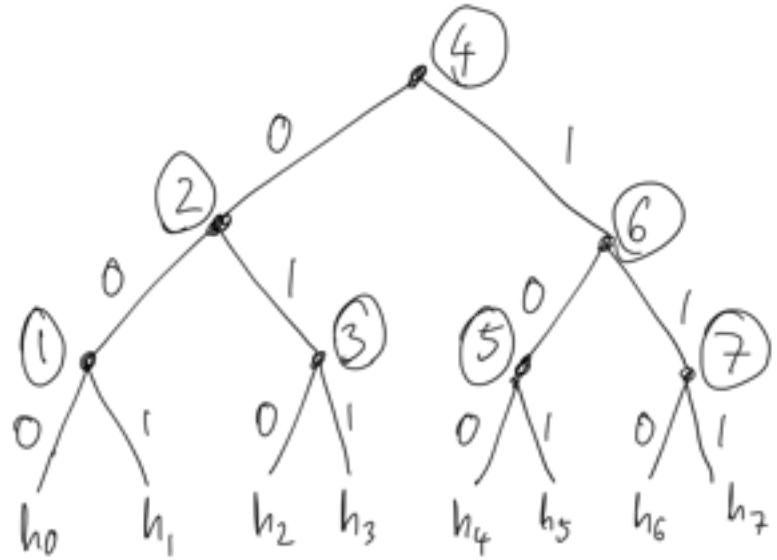
Example #2 (Thresholds):

- $X = \{1, 2, \ldots, n\}$
- For any $x = 0, 1, \ldots, n$:

$$1_x(x') = \begin{cases} 1 & \text{if } x' \leq x \end{cases}$$

$h_x(x') = \begin{cases} 1 & \text{if } x' > x \\ 0 & \text{if } x' \le x \end{cases}$

- $H_n = \{h_0, h_1, h_2, \dots, h_n\}$
- $Ldim(H) = \lfloor \log_2(n+1) \rfloor$
- Shattered tree for $n = 7$:



- Note: $VC(H_n) = 1$
  $Ldim(H_n) = \lfloor \log_2 |H_n| \rfloor$

Theorem:

$H \subset Y^X$

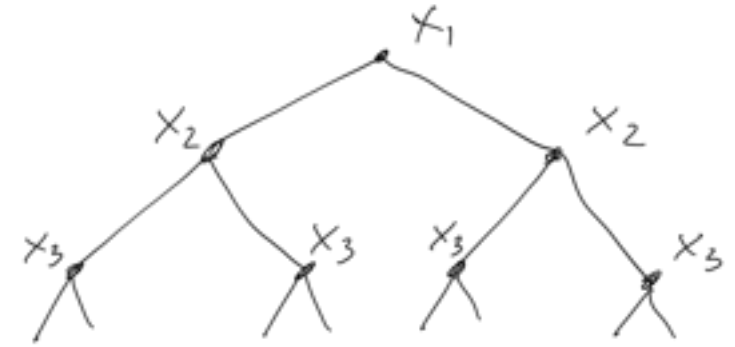
For any finite $|H| \geq 1$,

$$VC(H) \leq Ldim(H) \leq \lfloor \log_2 |H| \rfloor$$

Proof:

- First inequality:
  Shattered set $\Rightarrow$ shattered tree

- If $S = \{x_1, x_2, \ldots, x_n\}$ is shattered by H.



h_1 . . . . . $h_{2^{|S|}}$

- Second inequality follows from Halving algorithm.

∎

Winnow algorithm

. . d

- $X = \{0,1\}^d$
- A monotone disjunction is a function $h: X \to \{0,1\}$ of the form:

  $h(x_1 \dots x_d) = x_{i_1} \vee x_{i_2} \vee \cdots \vee x_{i_k}$

  for some indices

  $1 \le i_1 < i_2 < \cdots < i_k \le d$.

- k is called the number of <u>relevant</u> variables

- Winnow algorithm

Parameters: $\alpha > 1$, $\theta \ge 1/\alpha$

- $w_1 = (1, 1, \dots, 1) \in \mathbb{R}^d$
- For $t = 1, 2, \dots$
  1) Receive $x_t \in \{0,1\}^d$
  2) Predict $\hat{y}_t \in \{0,1\}$,

  $\hat{y}_t = \mathbb{1}\left[w_t^T \cdot x_t > \theta \right]$

3) Receive $y_t \in \{0,1\}$

4) If $\hat{y}_t = y_t$ set

$$w_{t+1} = w_t \quad \text{(no update)}$$

If $\hat{y}_t = 1$ and $y_t = 0$

$$w_{t+1,i} = \begin{cases} 0 & \text{if } x_{t,i} = 1 \\ w_{t,i} & \text{if } x_{t,i} = 0 \end{cases}$$

If $\hat{y}_t = 0$ and $y_t = 1$

$$w_{t+1,i} = \begin{cases} \alpha \cdot w_{t,i} & \text{if } x_{t,i} = 1 \\ w_{t,i} & \text{if } x_{t,i} = 0 \end{cases}$$

Theorem: If the target function is a monotone disjunction $h^* : \{0,1\}^d \to \{0,1\}$ with at most k relevant variables then Winnow makes at most

$$\alpha k (\log_\alpha \theta + 1) + \frac{d}{\theta}$$

mistakes.

In particular for $\alpha = 2$, $\theta = \frac{d}{2}$

Winnow makes at most

$$2k \log_2 d + 2$$

mistakes.

Proof:

- #Mistakes = # Updates to w_t

- Two types of updates:
  - $\hat{y}_t = 1$ and $y_t = 0$ (elimination step)
  - $\hat{y}_t = 0$ and $y_t = 1$ (promotion step)

- Let u = # promotion steps
  v = # elimination steps
- We upper bound u, v separately
- Let T be the total number of rounds

Lemma: $V \leq \frac{d}{\theta} + (\alpha - 1)u$

Proof:

- Consider $\|w_t\|_1 = \sum_{i=1}^{d} w_{t,i}$
- Initially $\|w_1\|_1 = d$
- Promotion step increases $\|w_t\|_1$ by at most $(\alpha-1)\theta$
- Elimination step decrease $\|w_t\|_1$ by at least θ
- Therefore

$$\|w_{T+1}\|_1 \leq d + \theta(\alpha-1)u - \theta v$$

- Since $\|w_{T+1}\|_1 \geq 0$,

$$0 \leq d + \theta(\alpha-1)u - \theta v$$

∎

Lemma: For all t and all i,

$$w_{t,i} \leq \alpha \cdot \theta$$

Proof:

- $w_{1,i} = 1$, $\alpha \cdot \theta \geq 1$ ✓
- $w_{t,i}$ is increased only during promotion
- If promotion

$$w_{t+1,i} = \alpha \cdot w_{t,i}$$

happens then

$$w_{t,i} \leq w_t^T x_t \leq \theta$$

- So

$$w_{t+1,i} \leq \alpha \cdot \theta$$

∎

Lemma: There exists $i \in \{1, \ldots, d\}$

$$\log_\alpha w_{T+1,i} \geq u/k$$

Proof:

- Let $h : \{0,1\}^n \to \{0,1\}$ be the target function

$$h^*(x_1 \dots x_d) = x_{i_1} \vee x_{i_2} \vee \dots \vee x_{i_k}$$

where
$1 \le i_1 < i_2 < \dots < i_k \le d$.

- Let $R = \{i_1, i_2, \dots, i_k\}$ be the relevant coordinates

- Consider $p_t = \prod_{i \in R} w_{t,i}$

- p_t is unchanged in elimination step
- p_t increases by at least factor α in promotion step
- After u promotion steps

$$p_{T+1} \ge \alpha^u$$

- Thus

$$\log_\alpha p_{T+1} \ge u$$

- Thus

$$\sum_{i \in R} \log_\alpha w_{T+1,i} \geq u$$

- Since $|R|$, there exists $i \in R$ such that

$$\log_\alpha w_{T+1,i} \geq u/k$$

■

Proof of theorem:

There exists i such that

1) $u/k \leq \log_\alpha w_{T+1,i}$

2) $w_{T+1,i} \leq \alpha \theta$

3) $V \leq \frac{d}{\theta} + (\alpha - 1) u$

- 1) + 2) imply:

$$u/k \leq \log_\alpha w_{T+1,i} \leq \log_\alpha (\alpha \theta)$$

$$= 1 + \log_\alpha \theta$$

- Thus

$$u \leq k(1+\log_2 \theta) \qquad (*)$$

- 3) + (*) imply

$$V \leq \frac{d}{\theta} + (2-1)\,k(1+\log_2 \theta) \qquad (**)$$

- (*) + (**) imply

$$u+V \leq \frac{d}{\theta} + 2k(1+\log_2 \theta)$$

↑ Total number of mistakes

Theorem:

Let $H_{d,k}$ be the class of monotone disjunctions over $\{0,1\}^d$ with at most exactly k relevant variables.

Then,

$$VC(H_{d,k}) \geq k\left\lfloor \log_2 \frac{d}{k} \right\rfloor$$

Proof omitted. 

- The theorem shows that Winnow is nearly optimal

- Winnow can be easily extended to non-monotone disjunctions

$$x_1 \vee \overline{x}_7 \vee x_{13} \vee \overline{x}_{47}$$

- Treat negated literals as new variables.

  Map feature vector

  $$x \in \{0,1\}^d \longmapsto \{0,1\}^{2d}$$

  by adding negations

Weighted Majority Algorithm

- Suppose $H \subseteq Y^X$ is finite

- We drop the assumption that there exists a target function $h^* \in H$

(i.e. $h^*(x_t) = y_t$)

- In other words, $y_1, y_2, \cdots \in Y$ can be completely arbitrary
- We want to come up with an algorithm A such that $\#\text{mistakes}(A)$ is not much worse than $\#\text{mistakes}(h)$ for all $h \in H$
- Let $N = |H|$ and $H = \{h_1, h_2, \cdots, h_N\}$
- Weighted Majority Algorithm (WMA)

  Parameter: $\beta \in (0,1)$

  Initialize $w_1 = (1, 1, \cdots, 1) \in \mathbb{R}^N$

  For $t = 1, 2, \cdots$

  1) Receive $x_t \in X$
  2) Compute

$$W_t^0 = \sum_{i:h_i(x_t)=0} w_{t,i}$$

$$W_t^1 = \sum_{j:h_i(x_t)=1} w_{t,i}$$

3) Predict $\hat{y}_t \in \{0,1\}$

$$\hat{y}_t = \begin{cases} 1 & \text{if } W_t^1 > W_t^0 \\ 0 & \text{if } W_t^0 \geq W_t^1 \end{cases}$$

4) Receive $y_t \in \{0,1\}$

5) For $i = 1, 2, \ldots, N$

$$w_{t+1,i} = \begin{cases} w_{t,i} & \text{if } h_i(x_t) = y_t \\ \beta \cdot w_{t,i} & \text{if } h_i(x_t) \neq y_t \end{cases}$$

(Decrease weight of h_i if h_i made a mistake)

- Let

m = #mistakes of WMA

$m(h)$ = #mistakes of h

Theorem: For any $h \in H$

$$m \leq \frac{\log(1/\beta)\, m(h) + \log N}{\log\left(\frac{2}{1+\beta}\right)}$$

Proof:

- Consider $\|w_t\|_1 = \sum_{i=1}^{N} w_{t,i}$

- If WMA makes a mistake

$$\begin{aligned}
\|w_{t+1}\| &= \beta w_t^{\hat{y}_t} + w_t^{y_t} \\
&= \beta\left(\|w_t\|_1 - w_t^{y_t}\right) + w_t^{y_t} \\
&= (1-\beta) w_t^{y_t} + \beta \|w_t\|_1 \\
&\leq (1-\beta)\frac{\|w_t\|_1}{2} + \beta \|w_t\|_1 \\
&= \frac{1+\beta}{2}\|w_t\|_1
\end{aligned}$$

- So after m mistakes

$$\|w_{T+1}\|_1 \leq \left(\frac{1+\beta}{2}\right)^m \|w_1\|_1 = \left(\frac{1+\beta}{2}\right)^m N$$

- For any $h_i \in H$

$$\|W_{T+1}\|_1 \geq W_{T+1,i} = \beta^{m(h_i)}$$

- Thus
$$\beta^{m(h_i)} \leq \|W_{T+1}\|_1 \leq \left(\frac{1+\beta}{2}\right)^m N$$

- Thus
$$\beta^{m(h_i)} \leq \left(\frac{1+\beta}{2}\right)^m N$$

- Solve for m !

■

Randomized Weighted Majority Algorithm

- Also called Hedge algorithm
- Also called Exponential Weights Algorithm
- We also slightly generalize the problem:
- Notice that WMA uses $h_1, h_2, \ldots, h_N$ as black

boxes. The only thing that matters is whether h_i makes a mistake or not.

- We consider slightly more general problem where there are N experts.

- Each experts makes a prediction (e.g. whether or not it will rain tomorrow)
- The algorithm will choose one of the expert's predictions.
- Then each experts is assigned a loss/cost/error ...
- <u>Prediction with Expert Advice</u>

  For $t=1,2,\dots$

  1) Choose expert $I_t \in \{1,2,\dots,N\}$
  2) Experts suffer losses

     $\ell_{t,1}, \ell_{t,2}, \dots, \ell_{t,N} \in [0,1]$

3) Algorithm suffer loss

$$\ell_{t,I_t}$$

- Randomized Weighted Majority Algorithm

Parameters:

N = number of experts

$\eta > 0$ (learning rate)

Initialize $W_1 = (1, 1, \ldots, 1) \in \mathbb{R}^N$

For $t = 1, 2, \ldots$

1) Compute for $i = 1, \ldots, N$

$$P_{t,i} = \frac{W_{t,i}}{\sum_{j=1}^{N} W_{t,j}}$$

2) Choose $I_t \in \{1, 2, \ldots, N\}$ at random from distribution $P_t = (P_{t,1}, \ldots, P_{t,N})$

That is

$$\Pr[I_t = i] = P_{t,i}$$

3) Observe the experts' losses

$\ell_{t,1}, \ell_{t,2}, \cdots, \ell_{t,N}$

(and "suffer" loss ℓ_{t,I_t})

4) Update for $i = 1, \ldots, N$

$$W_{t+1,i} = W_{t,i} \cdot e^{-\eta \ell_{t,i}}$$

- Note that when

$\ell_{t,i} \in \{0,1\}$ and $e^{-\eta} = \beta$

then the update rules of WMA and RWMA are identical

Theorem:

Suppose $\eta = \sqrt{\frac{8 \ln N}{T}}$ where T is a positive integer.

Then RWMA satisfies for any expert $i \in \{1, 2, \ldots, N\}$

$$\underbrace{\mathbb{E}\left[\sum_{t=1}^{T} \ell_{t,I_t}\right]}_{\text{expected loss of RWMA}} \leq \underbrace{\sum_{t=1}^{T} \ell_{t,i}}_{\text{loss of expert } i} + \underbrace{\sqrt{\tfrac{1}{2} T \ln N}}_{\text{this is much less than } T}.$$

Proof:

- If $T \leq 2 \ln N$ then inequality is trivially true.
- We upper and lower bound
$$\ln\left(\sum_{j=1}^{N} w_{T+1,i}\right)$$
- Let $Z_t = \sum_{j=1}^{N} w_{t,j} = \|w_t\|_1$
- Lower bound $\ln Z_{T+1}$:

$$\ln Z_{T+1} = \ln\left(\sum_{j=1}^{N} w_{T+1,j}\right)$$

$$= \ln\left(\sum_{j=1}^{N} e^{-\eta \sum_{t=1}^{T} \ell_{t,j}}\right)$$

$$\geq \ln\left(e^{-\eta \sum_{t=1}^{T} \ell_{t,i}}\right)$$

$$= -\eta \underbrace{\sum_{t=1}^{T} \ell_{t,i}}_{\text{loss of expert } i}$$

- Upper bound $\ln Z_{T+1}$:

$$\ln Z_{T+1} = \ln Z_1 + \sum_{t=1}^{T} \ln\left(\frac{Z_{t+1}}{Z_t}\right)$$

$$= \ln N + \sum_{t=1}^{T} \ln\left(\frac{Z_{t+1}}{Z_t}\right)$$

- We upper bound $\ln\left(\frac{Z_{t+1}}{Z_t}\right)$:

$$\ln\left(\frac{Z_{t+1}}{Z_t}\right) = \ln\left(\frac{\sum_{j=1}^{N} w_{t+1,j}}{Z_t}\right)$$

$$= \ln\left(\sum_{j=1}^{N} \frac{w_{t,j} \cdot e^{-\eta \ell_{t,j}}}{Z_t}\right)$$

$$= \ln\left(\sum_{j=1}^{N} p_{t,j} \cdot e^{-\eta \ell_{t,j}}\right)$$

- Think of random variable X such that $X = \ell_{t,r}$

- Note that

$$\sum_{j=1}^{N} p_{t,j} \cdot e^{-\eta \ell_{t,j}} = \mathbb{E}\left[e^{-\eta X}\right]$$

- Let $Y = X - \mathbb{E}[X]$

- Then

$$\mathbb{E}\left[e^{-\eta X}\right] = \mathbb{E}\left[e^{-\eta(Y+\mathbb{E}X)}\right]$$

$$= e^{-\eta \mathbb{E}X} \cdot \mathbb{E}\left[e^{-\eta Y}\right]$$

- Y lies in an interval of length <u>one</u>. We apply Hoeffding's lemma

$$\mathbb{E}\left[e^{-\eta Y}\right] \leq e^{\eta^2/8}$$

- So

$$\ln\left(\frac{Z_{t+1}}{Z_t}\right) = \ln\left(e^{-\eta \mathbb{E}X} \cdot \mathbb{E}\left[e^{-\eta Y}\right]\right)$$

$$= -\eta \mathbb{E}X + \ln \mathbb{E}\left[e^{-\eta Y}\right]$$

$$\leq -\eta \mathbb{E}X + \eta^2/8$$

$$= -\eta \mathbb{E}\left[\ell_{t,I_t}\right] + \eta^2/8$$

- So

$$\ln Z_{T+1} = \ln N + \sum_{t=1}^{T} \ln\left(\frac{Z_{t+1}}{Z_t}\right)$$

$$\leq \ln N - \eta \sum_{t=1}^{T} \mathbb{E}\left[\ell_{t,I_t}\right] + T\eta^2/8$$

- Combining the upper and lower bounds on $\ln Z_{T+1}$

$$-\eta \sum_{t=1}^{T} \ell_{t,i} \leq \ln N - \eta \sum_{t=1}^{T} \mathbb{E}\left[\ell_{t,I_t}\right] + T\eta^2/8$$

- Equivalently

$$\sum_{t=1}^{T} \mathbb{E}\left[\ell_{t,I_t}\right] \leq \sum_{t=1}^{T} \ell_{t,i} + \frac{\ln N}{\eta} + \frac{\eta T}{8}$$

optimize over $\eta > 0$

- Choose $\eta = \sqrt{\frac{8 \ln N}{8}}$

$$\sum_{t=1}^{T} \mathbb{E}[\ell_{t,I_t}] \leq \sum_{t=1}^{T} \ell_{t,i} + \sqrt{\frac{1}{2} T \ln N}$$

■

- Theorem upper bounds <u>expected</u> loss of the algorithm
- It is possible to prove a high probability bound: with probability at least $1-\delta$

$$\sum_{t=1}^{T} \ell_{t,I_t} \leq \sum_{t=1}^{T} \ell_{t,i} + \sqrt{T \ln\left(\frac{N}{\delta}\right)}$$

<u>Perceptron</u>

- Suppose $X = \mathbb{R}^d$
- We play the online learning game from the beginning

of the lecture:

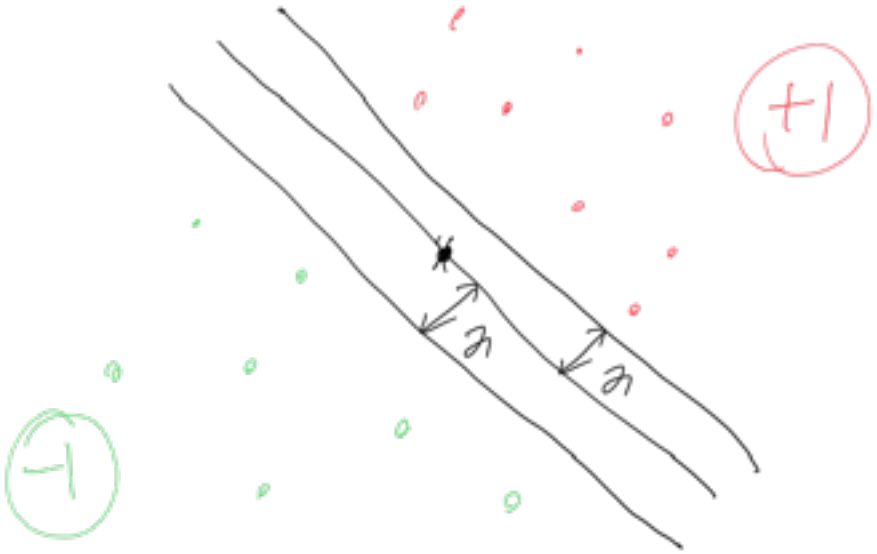
For $t = 1, 2, \ldots$

1) Receive $x_t \in X$
2) Predict $\hat{y}_t \in \{+1, -1\}$
3) Receive $y_t \in \{+1, -1\}$

- We assume that

$(x_1, y_1), (x_2, y_2) \ldots (x_T, y_T)$

are linearly separable with margin $\gamma > 0$:



- Formally there exists $w \in \mathbb{R}^d$, $\|w\|_2 = 1$ such

that for all $t=1,2,\dots$

$$y_t \underbrace{w^T \cdot x_t}_{\text{oriented distance between } x_t \text{ and separating hyperplane}} \geq \gamma$$

- **Perceptron algorithm**

Initialize $w_1 = (0, 0, \dots, 0) \in \mathbb{R}^d$

For $t=1,2,\dots$

1) Receive $x_t \in \mathbb{R}^d$

2) Predict $\hat{y}_t \in \{+1, -1\}$

$$\hat{y}_t = \text{sign}\left(w_t^T \cdot x_t\right)$$

3) Receive $y_t \in \{+1, -1\}$

4) If $\hat{y}_t = y_t$ then

$$w_{t+1} = w_t \quad \text{(no update)}$$

If $\hat{y}_t \neq y_t$ then

$$w_{t+1} = w_t + y_t x_t$$

Novikoff's theorem:

Assume that $(x_1, y_1), (x_2, y_2) \ldots$ are linearly separable with margin $\gamma > 0$.

Assume there exists $R \geq 0$ such that

$$\forall t \qquad \|x_t\|_2 \leq R.$$

Then, Perceptron makes at most

$$\left\lfloor \left(\frac{R}{\gamma}\right)^2 \right\rfloor$$

mistakes.

For proof we will need

Cauchy-Schwarz inequality

If $u, v \in \mathbb{R}^d$ then

$$|u^T v| \leq \|u\|_2 \cdot \|v\|_2$$

(Equality holds if and only if u, v are multiples of each other.)

$$u^T v = \|u\|_2 \cdot \|v\|_2 \cdot \cos\alpha$$

Proof of Novikoff's theorem:

- Let M be the number of mistakes
- Let T the # rounds
- Let $w \in \mathbb{R}$, $\|w\|_2 = 1$ be separating hyperplane with margin γ.

- We upper and lower bound $\|w_{T+1}\|_2^2$

- Suppose a mistake happens in round t:

$$w_{t+1} = w_t + y_t x_t$$

$$\|w_{t+1}\|_2^2 = \|w_t + y_t x_t\|_2^2$$

$$= (w_t + y_t x_t)^T (w_t + y_t x_t)$$

$$= \|w_t\|_2^2 + \|y_t x_t\|_2^2 + \underbrace{2 y_t w_t^T x_t}_{\leq 0}$$

$$\leq \|w_t\|_2^2 + \|y_t x_t\|_2^2$$

$$= \|w_t\|_2^2 + \|x_t\|_2^2$$

$$\leq \|w_t\|^2 + R^2$$

- Therefore

$$\|w_{\cdot}\|^2 \leq M \cdot R^2$$

$$\|w_{T+1}\|_2 = ?$$

- Lower bound

$$\|w_{T+1}\|_2 \geq w^T \cdot w_{T+1}$$

- We focus on $w^T \cdot w_{T+1}$

- Suppose mistake happens in round t:

$$w^T \cdot w_{t+1} = w^T(w_t + y_t x_t)$$

$$= w^T w_t + \underbrace{y_t w^T x_t}_{\geq \gamma}$$

$$\geq w^T w_t + \gamma$$

- Therefore

$$w^T w_{T+1} \geq M\gamma$$

- Thus

$$M^2\gamma^2 \le \|w_{T+1}\|_2^2 \le MR^2$$

- Thus
$$M^2\gamma^2 \le MR^2$$

- Thus
$$M \le \left(\frac{R}{\gamma}\right)^2$$

■

Online-to-Batch Conversions

- Suppose $(x_1, y_1), (x_2, y_2), \dots, (x_T, y_T)$ is an i.i.d. sample from a distribution D over $X \times Y$

- Can we use an online learning algorithm to learn a classifier with a small generalization error?

- Online-to-Batch conversion is a way to do it!

- Consider the online learning protocol:

  For $t = 1, 2, \ldots$
  1) Receive $x_t \in X$
  2) Predict $\hat{y}_t \in Y$
  3) Receive $y_t \in Y$

- The prediction $\hat{y}_t$ in step 2) is a function of x_t (and previous examples $(x_1, y_1) \ldots (x_{t-1}, y_{t-1})$)
- Let $h_t : X \to Y$ be that function
- We modify the protocol:

  For $t = 1, 2, \ldots$
  1) Choose $h_t \in Y^X$
  2) Receive $(x_t, y_t) \in X \times Y$

- So an online learning algorithm produces a sequence of classifiers

  $\in Y^X$

$h_1, h_2, \ldots \in Y$

- Is there a way to construct a single h with small generalization error?

- There are several options:

  1) Pick one at random from $h_1, h_2, \ldots, h_T$

  2) Construct a combined classifier e.g. using majority vote of $h_1, h_2, \ldots, h_T$

  3) Test $h_1, h_2, \ldots, h_T$ on an independent data set ("validation set") and pick the best one.

Pick one at random

- Suppose the online learning algorithm guarantees that for any sequence of examples $(x, y), (x, y), \ldots$ the

$(\hat{x}_1, \hat{y}_1), \ldots, (\hat{x}_T, \hat{y}_T)$ the algorithm makes at most M mistakes:

$$\sum_{t=1}^{T} \mathbb{1}[h_t(x_t) \neq y_t] \leq M$$

- Let τ be a random variable that is uniformly distributed over $\{1, 2, \ldots, T\}$
- Let $(X, Y) \sim D$
- Then

$$\mathbb{E}[\text{err}_D(h_\tau)] = \frac{1}{T} \sum_{t=1}^{T} \text{err}_D(h_t)$$

$$= \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[\mathbb{1}[h_t(X) \neq Y]]$$

h_t is independent of (x_t, y_t)!

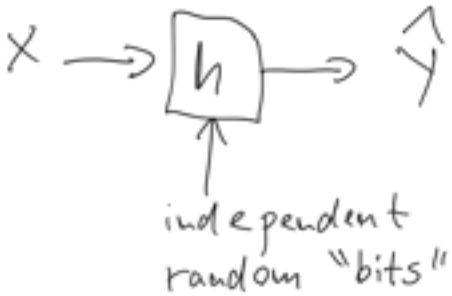
$$= \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[\mathbb{1}[h_t(x_t) \neq y_t]]$$

$$\leq \frac{M}{T}$$

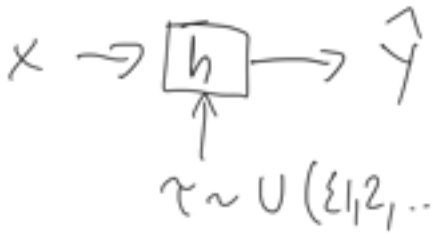
$\leq \frac{1}{T}$

Randomized Majority Classifier

- Randomized classifier:



-



$$h(x, \tau) = h_\tau(x)$$

$$err_D(h) = \Pr_{\substack{(x,y)\sim D \\ \tau \sim U(\cdots)}}[h(x,\tau) \neq Y]$$

$$= \mathbb{E}_\tau\left[\mathbb{1}[h(x,\tau) \neq Y]\right]$$

$$= \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}[h_t(x) \neq y]\right]$$

$$= \mathbb{E}\left[\overline{err}_D(h_T)\right]$$

$$\leq \frac{M}{T}$$

Other Online-to-Batch conversions

- Deterministic majority classifier
- Test $h_1, h_2, \ldots, h_T$ on an independent validation data set sampled from D