Online Algorithms: Perceptron and Winnow

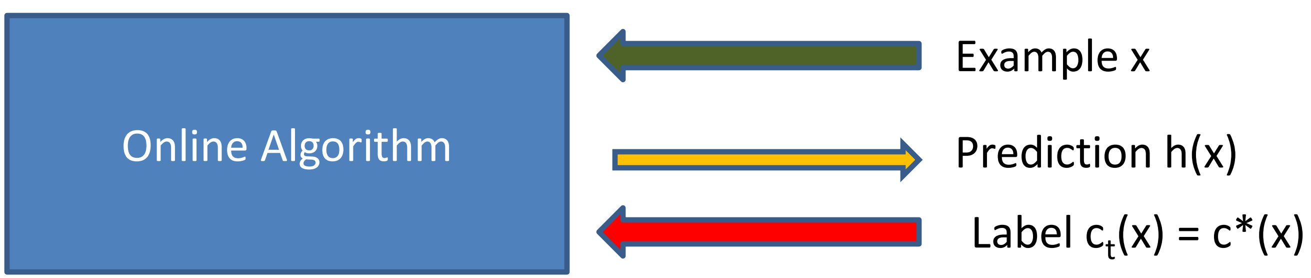
Outline

- Online Model
- Linear Separator
- Percpetron
  - Realizable case
  - Unrealizable case
- Winnow

Online Model

- Example arrive sequentially
- Need to make a prediction
  - Afterwards observe the outcome
- No distributional assumptions
- Goal: Minimize the number of mistakes

Phase t:



Example: Learning OR of literals

- Inputs: $(z_1, \dots, z_n)$
- Literals : $x_1, \bar{x}_1$
- OR functions:

  $x_1 \vee \bar{x}_4 \vee x_7$

- Realizable case:
  - $C^*(z)$ is an OR

- ELIM algorithm:
  - Initialize: $L = \{x_1, \bar{x}_1, \dots, x_n, \bar{x}_n\}$
  - Time t, receive
    - $z=(z_1, \dots, z_n)$
  - Predict OR(L,z)
  - Receive $c^*(z)$
    - If error (has to be negative)
    - delete from L the positive literals in z.

What is the MAXIMUM number of mistakes?

Learning Linear Separators

- Input $\{0,1\}^d$ or R^d
- Linear Separator
  - weights w in R^d and threshold θ
  - hypothesis $h(x)=+1$ iff

$$\langle w,x \rangle = \Sigma\, w_i\, x_i \geq \theta$$

- Simplifying assumptions:
  - $\theta=0$ (add coordinate x_0 such that $x_0=1$ always)
  - $||x||=1$

Perceptron - Algorithm

- Initialize $w_1=(0,\ldots,0)$
- Given example x_t,
  - predict positive iff $\langle w_t, x_t \rangle \geq 0$
- On a Mistake t: $w_{t+1} = w_t + c_t(x)\, x_t$,
  - Mistake on negative (i.e., $c^*(x)=+1$): $w_{t+1} = w_t + x_t$.
  - Mistake on positive (i.e., $c^*(x)=-1$): $w_{t+1} = w_t - x_t$.

Perceptron - motivation

- **False Negative**
  - $c_t(x) = +1$
  - $<w_t, x_t>$ negative
  - after update

$<w_{t+1}, x_t>$

$= \; <w_t, x_t> + <x_t, x_t>$

$= \; <w_t, x_t> + 1$

- **False Positive**
  - $c_t(x) = -1$
  - $<w_t, x_t>$ positive
  - after update

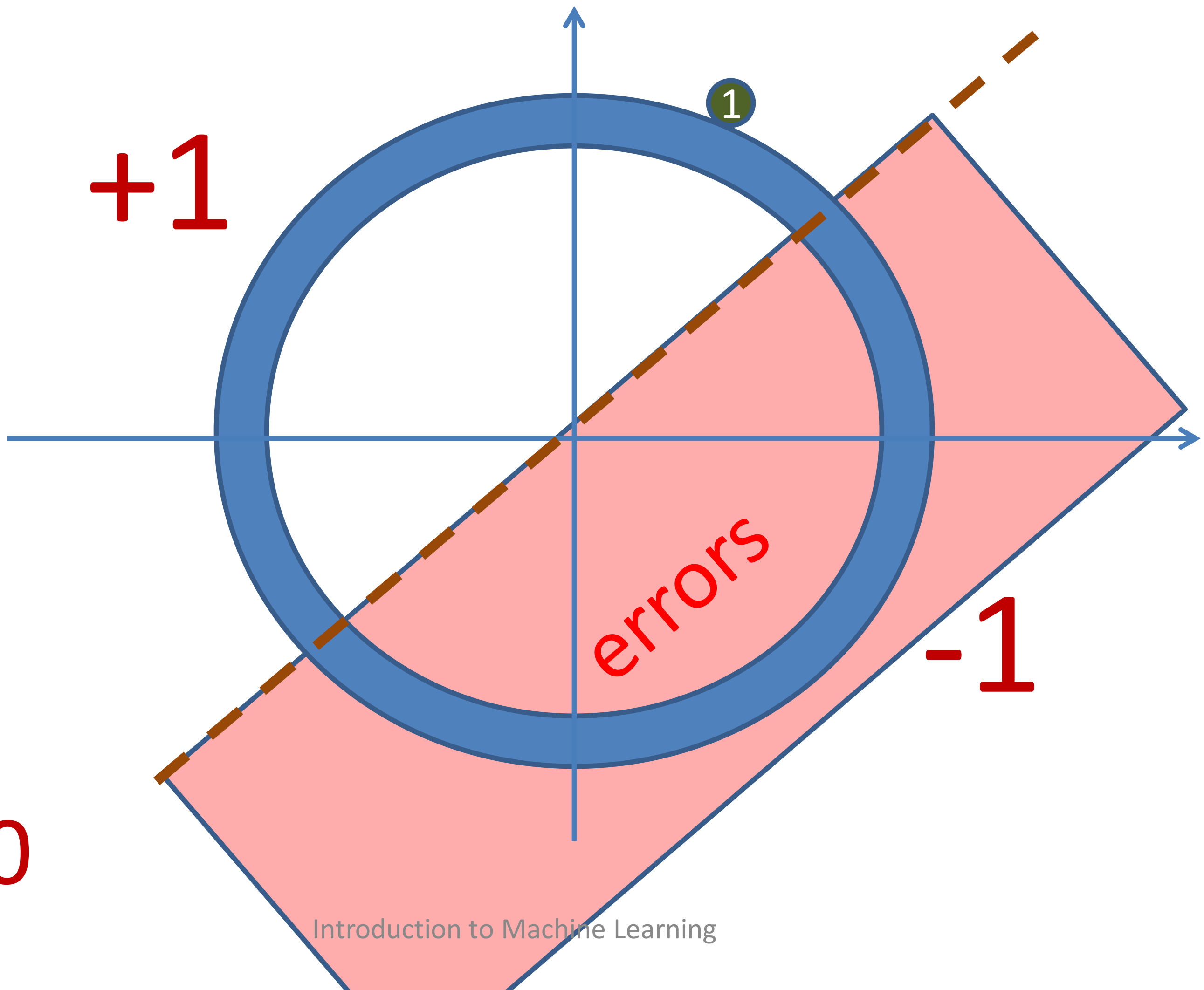
$<w_{t+1}, x_t>$

$= \; <w_t, x_t> - <x_t, x_t>$

$= \; <w_t, x_t> - 1$

Perceptron Example

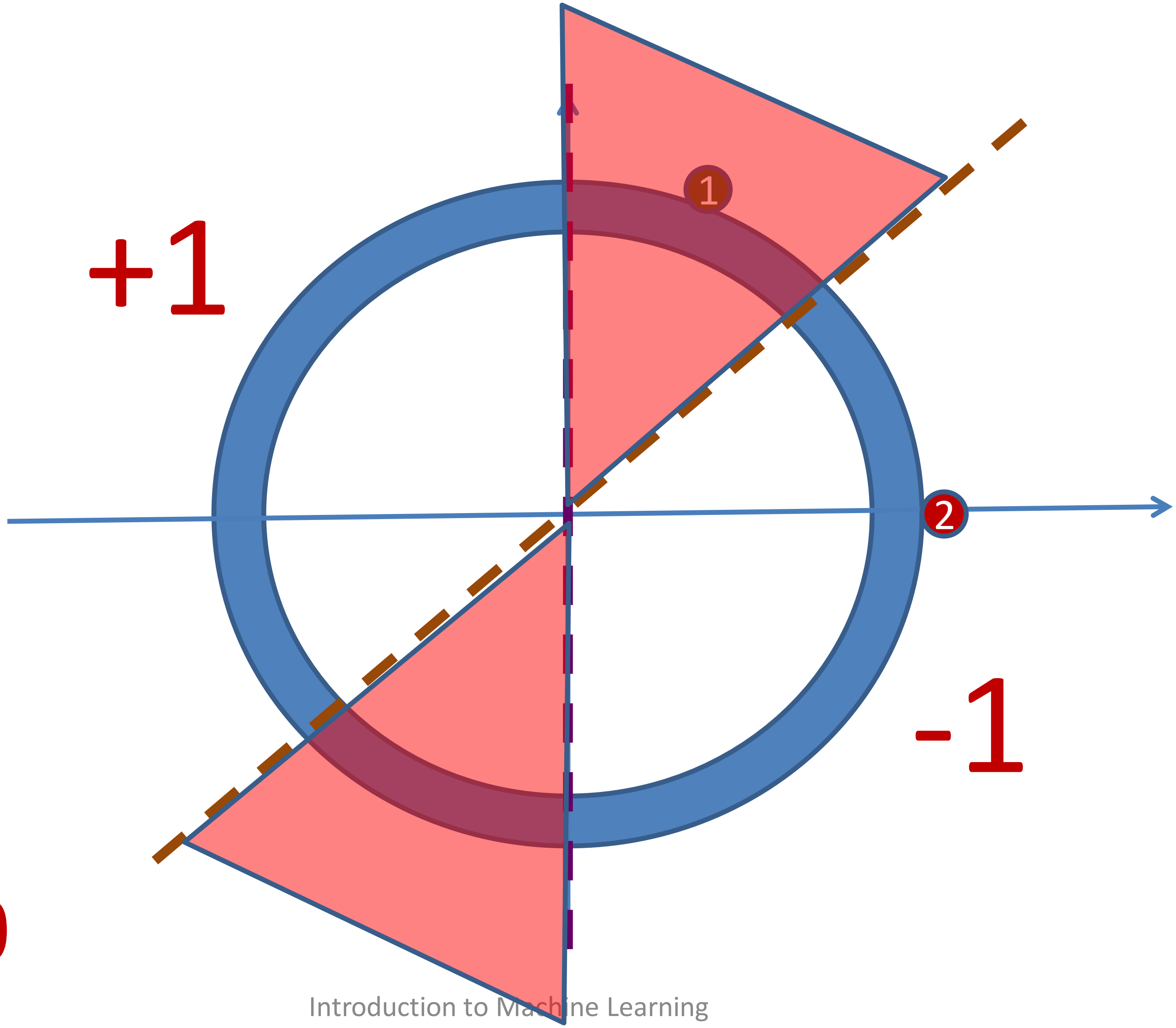
$w_1 = (0,0)$
$w_2 = (0,0)$



$x_1 - x_2 = 0$

Perceptron Example

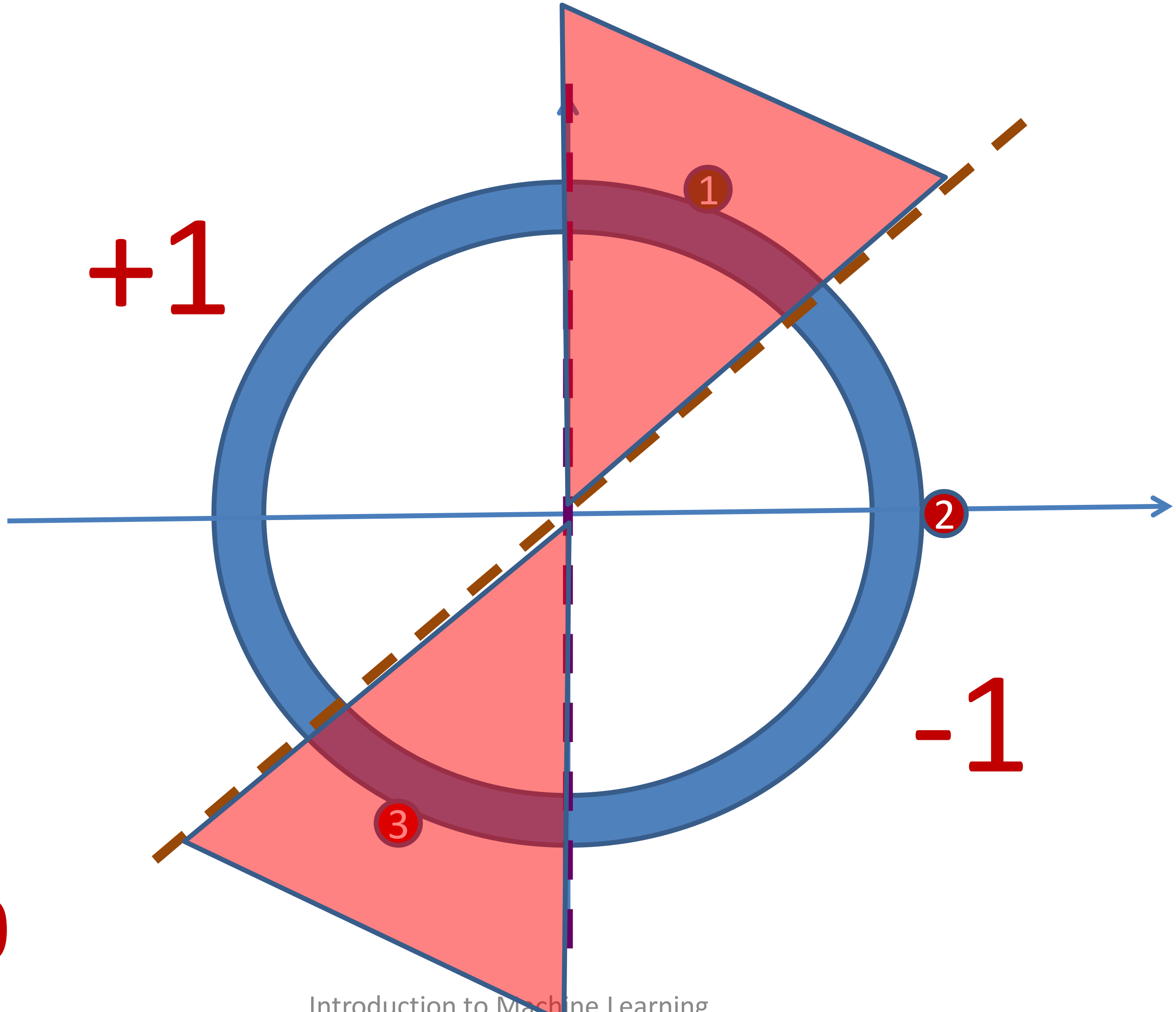
$w_1 = (0,0)$
$w_2 = (0,0)$
$w_3 = (-1,0)$



$x_1 - x_2 = 0$

Perceptron Example

$w_1 = (0,0)$
$w_2 = (0,0)$
$w_3 = (-1,0)$

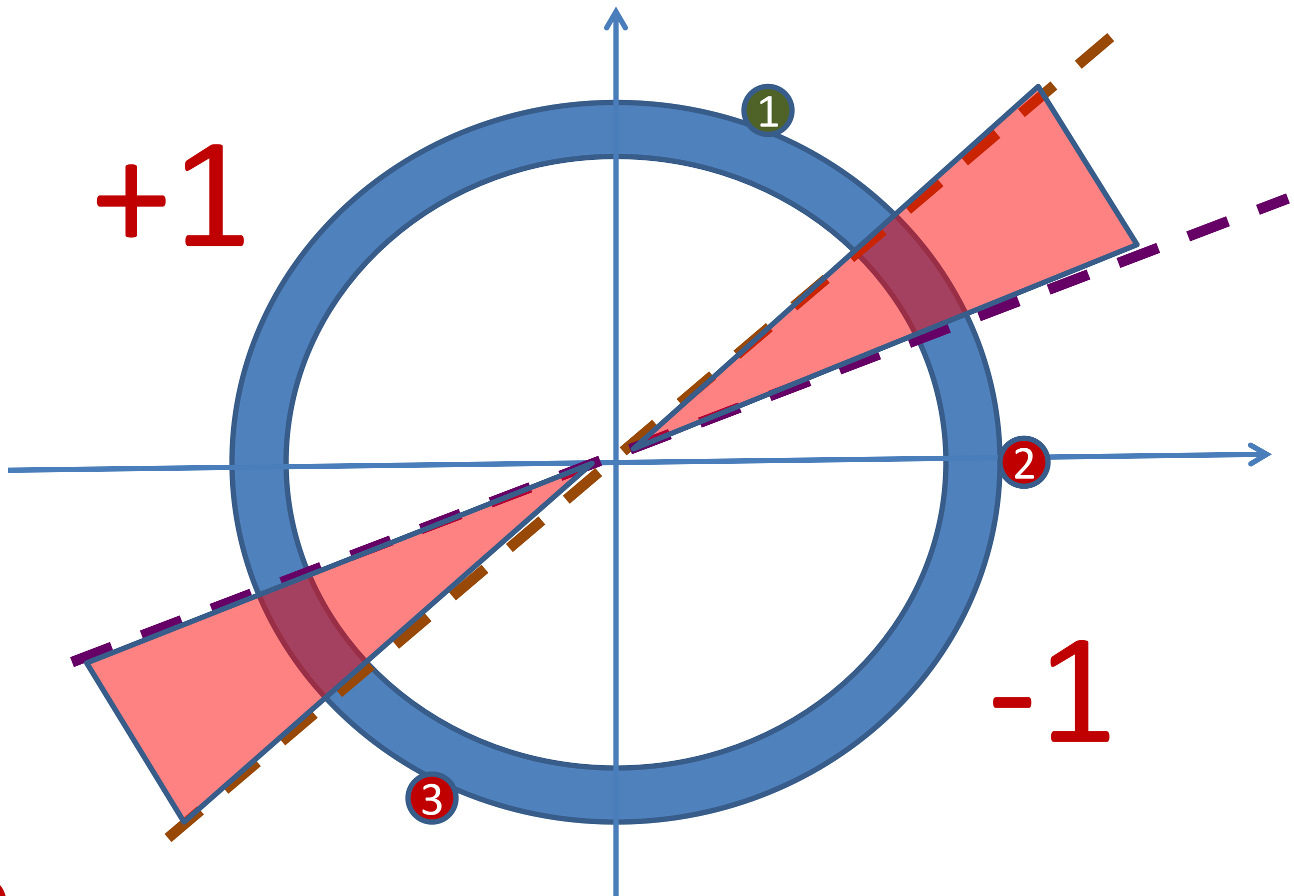


$x_1 - x_2 = 0$

Perceptron Example

$w_1 = (0,0)$
$w_2 = (0,0)$
$w_3 = (-1,0)$

$w_4 = (-0.2,+0.6)$

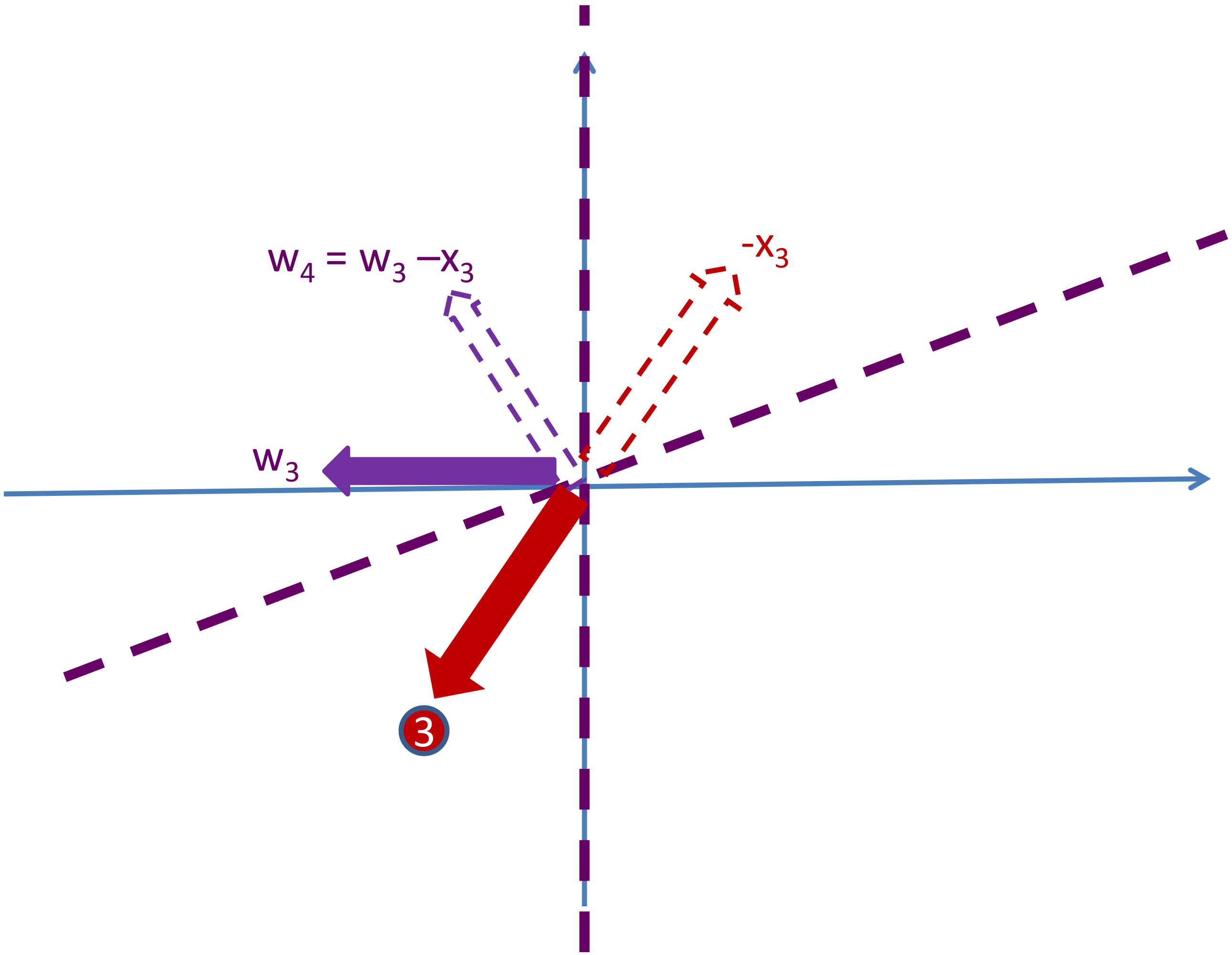


$x_1 - x_2 = 0$

Perceptron - Geometric Interpretation

$w_1 = (0,0)$
$w_2 = (0,0)$
$w_3 = (-1,0)$

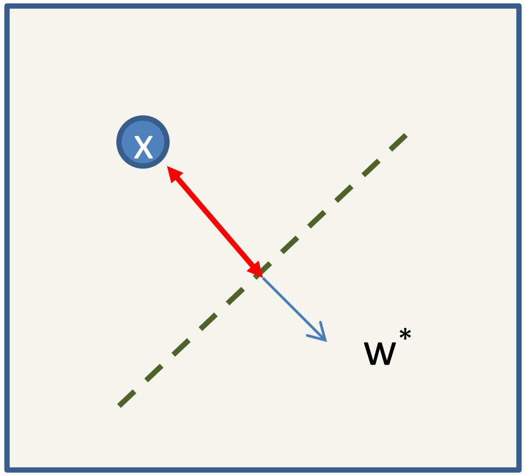
$w_4 = (-0.2,+0.6)$



Percdeptron - Analysis

- target concept $c^*(x)$ uses w^* and $||w^*||=1$
- Margin γ:
  - For any *x is S*

$$\gamma = \min_{x \in S} \frac{< x, w^* >}{\|x\|}$$



- Theorem: *Number of mistakes ≤ $1/\gamma^2$*

Perceptron - Performance

Claim 1:
$\langle w_{t+1}, w^* \rangle \geq \langle w_t, w^* \rangle + \gamma$

Assume $c^*(x)=+1$

$\langle w_{t+1}, w^* \rangle =$

$\langle (w_t + x), w^* \rangle =$

$\langle w_t, w^* \rangle + \langle x, w^* \rangle \geq$

$\langle w_t, w^* \rangle + \gamma$

Similar for $c^*(x)=-1$

Claim 2: $||w_{t+1}||^2 \leq ||w_t||^2 + 1$

Assume $c^*(x)=+1$

$||w_{t+1}||^2 =$

$||w_t + x||^2 =$

$||w_t||^2 + 2\langle w_t, x \rangle + ||x||^2 \leq$

$||w_t||^2 + 1$

Since x is a mistake $\langle w_t, x \rangle$ is negative.

Similar for $c^*(x)=-1$

Perceptron - performance

Claim 3: $<w_t, w^*> \;\leq\; ||w_t||$

$$< w_t, w^* > \leq \; < w_t, \frac{w_t}{\|w_t\|} > = \|w_t\|$$

Completing the proof

- After M mistakes:

$<w_{M+1}, w^*> \;\geq\; \gamma M$ (claim1)

$||w_{M+1}||^2 \;\leq\; M$ (claim 2)

$$\gamma M \leq \; < w_{M+1}, w^* > \leq \|w_{M+1}\| \leq \sqrt{M}$$

$$M \leq \frac{1}{\gamma^2}$$

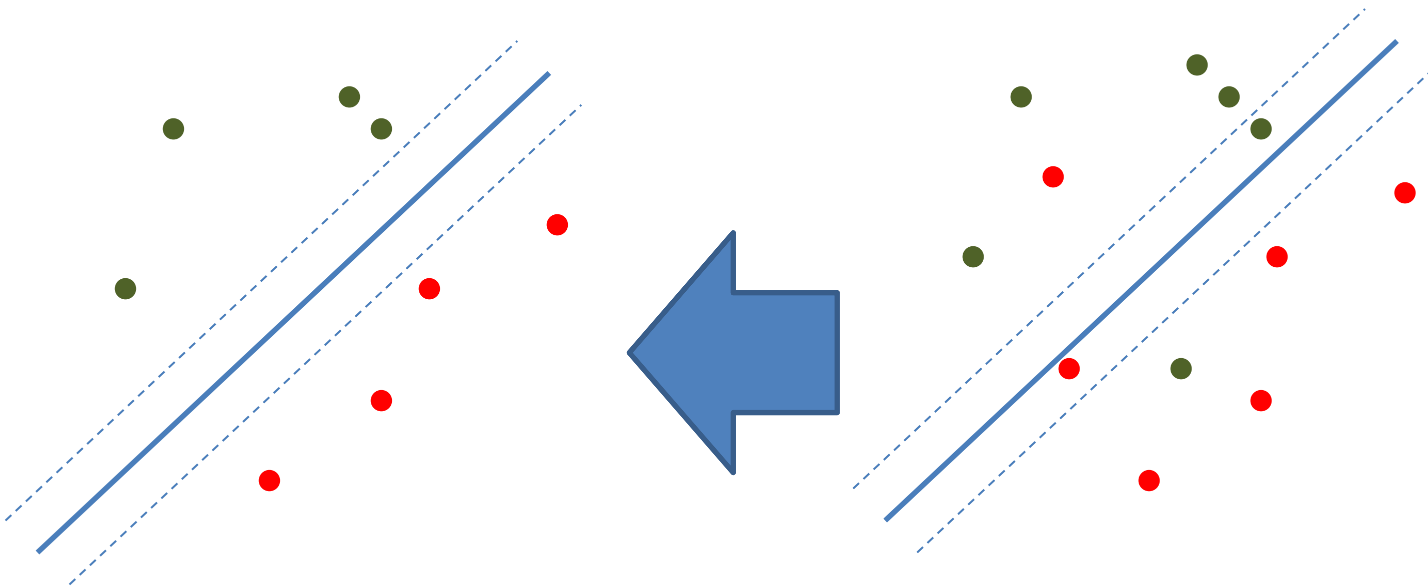
Perceptron

- Guaranteed convergence
  - realizable case
- Can be very slow (even for $\{0,1\}^d$)
- Additive increases:
  - problematic with large weights
- Still, a simple benchmark

Perceptron – Unrealizable case

Motivation

Realizable case **Unrelizable case**



Hinge Loss

Motivation

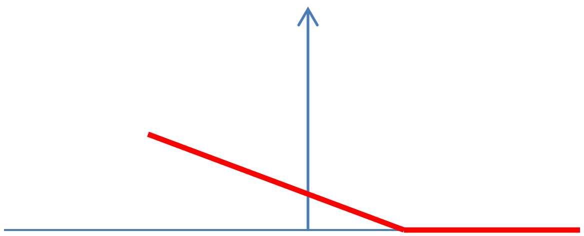
- "Move" points to be realizable
  - with margin γ
- correct points
  - both classification and margin
  - zero loss
- mistake points
  - even just margin
  - loss is the distance

Definition

- Assume <*x*,*w*> = β
- Hinge Loss with margin γ:

$$\max\{0, 1 - \frac{c^*(x)\beta}{\gamma}\}$$

- Error: $c^*(x)\beta < 0$
- correct margin: zero loss



Perceptron - Performance

- Let TD_γ=total distance

  $\Sigma_i \max\{0, \gamma - c^*(x)\beta_i\}$, where $\beta_i = \langle x_i, w^* \rangle$
- Claim 1': $\langle w_{M+1}, w^* \rangle \geq \gamma M - TD_\gamma$
- Claim 2: $||w_{t+1}||^2 \leq ||w_t||^2 + 1$
- Bounding the mistakes:

$$\sqrt{M} \geq \gamma M - TD_\gamma \quad \Rightarrow \quad M \leq \frac{1}{\gamma^2} + \frac{2}{\gamma} TD_\gamma$$

Winnow

Winnow –motivation

- Updates
  - multiplicative vs additive
- Domain
  - $\{0,1\}^d$ or $[0,1]^d$
    - we will use $\{0,1\}^d$
- Weights
  - non-negative
    - monotone function

- Separation
  - $c^*(x)=+1$: $\langle w^*,x\rangle \geq \theta$
  - $c^*(x)=-1$: $\langle w^*,x\rangle \leq \theta - \gamma$
  - $\theta > 1$
    - part of the input
- Remarks:
  - normalizing x in L_∞ to 1

Winnow - Algorithm

- parameter **$\beta > 1$**
  - we will use **$\beta = 1 + \gamma/2$**
- Initialize **$w = (1, \ldots, 1)$**
- predict **$h(x) = +1$** iff

  $\langle w, x \rangle \geq \theta$

- For a mistake:
- False Positive (**demotion**)
  - $c^*(x) = -1$, $h(x) = +1$
  - for every **$x_i = 1$: $w_i = w_i / \beta$**
- False Negative (**promotion**)
  - $c^*(x) = +1$, $h(x) = -1$
  - for every **$x_i = 1$: $w_i = \beta w_i$**

Winnow - intuition

- Demotion step
  - target negative
  - hypothesis positive
- Before update

  $\langle w,x \rangle = \alpha \geq \theta$
- After the update:

  $\langle w,x \rangle = \alpha/\beta < \alpha$
- Decrease in $\sum w_i$
  - at least $(1- \beta^{-1})\theta$

- Promotion step
  - target positive
  - hypothesis negative
- Before update

  $\langle w,x \rangle = \alpha < \theta$
- After the update:

  $\langle w,x \rangle = \alpha\beta > \alpha$
- Increase in $\sum w_i$
  - at most $(\beta-1)\theta$

Winnow - example

- Target function:
- $w^*=(2,2,0,0)$
- $\theta=2$, $\beta=2$

- What is the target function?
  - x_1 v x_2
  - monotone OR

- $w_0=(1,1,1,1)$
- $x_1=(0,0,1,1)$ $c_t(x_1)=-1$
  - $w_1=(1,1, ½, ½)$
- $x_2=(1,0,1,0)$ $c_t(x_2)=+1$
  - $w_2=(2,1, 1, ½)$
- $x_3=(0,1,0,1)$ $c_t(x_3)=+1$
  - $w_3=(2,2, 1, 1)$

Winnow - Theorem

- Theorem (realizable case)

Number of mistakes bounded by

$$O\left(\frac{1}{\gamma^2}\frac{d}{\theta}+\frac{\ln\theta}{\gamma^2}\sum_{i=1}^{d}w_i^*\right)$$

- Corollary: For θ=d we have $O\left(\frac{\ln d}{\gamma^2}\sum_{i=1}^{d}w_i^*\right)$

Winnow - Analysis

- Mistakes
  - u promotion steps
  - v demotion steps
  - mistakes = u+v
- Lemma 1:

$$v \leq \frac{\beta}{\beta - 1}\frac{d}{\theta} + \beta u$$

- Lemma 2: $w_i \leq \beta\theta$
- Lemma 3:
  after u prom.
  and v demo.
  exists *i*

$$\log w_i \geq \frac{\theta u - (\theta - \gamma)v}{\sum_{i=1}^{d} w_i^*} \log \beta$$

- Proof of theorem

Winnow vs Perceptron

Percetron

- Additive updates
  - slow for large d
  - slow large weights
- Non-monotone
  - natural
- Simple Algorithm
- Margin scale $L_2(w^*)L_2(x)$

Winnow

- Multiplicative updates
  - handles large d nicely
  - ok with large weights
- Non-monotone
  - need to make monotone
  - flip non-monotone attributes
- Simple Algorithm
- Margin scale $L_1(w^*)L_\infty(x)$
- Additional factor log d
  - for θ=d

Summary

Linear Separators

- Today: Perceptron and Winnow
- Next week: SVM
- 2 weeks: Kernels
- 3 weeks: Adaboost

Brief history:

- Perceptron
  - Rosenblatt 1957
- Fell out of favor in 70s
  - representation issues
- Reemerged with Neural nets
  - late 80s early 90s
- Linear separators:
  - Adaboost and SVM
- future ???