The Perceptron Algorithm

Announcements

1. P2 (Perceptron) will be out tmr

Recap on PCA

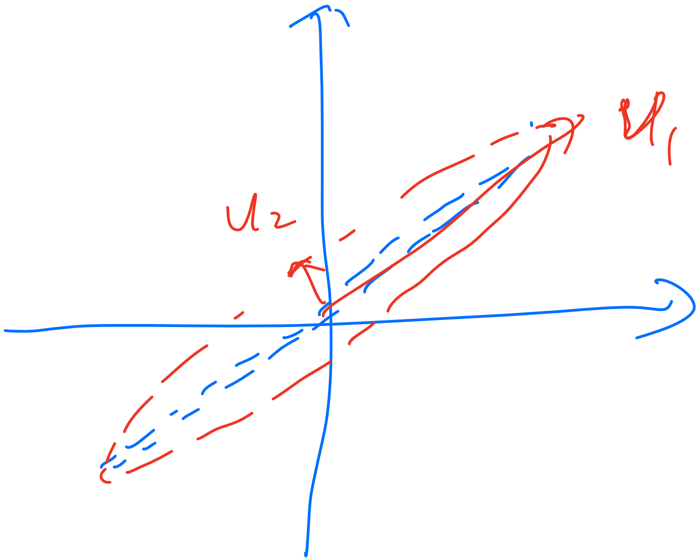
T/F: we need to center the dataset before we run PCA

Recap on PCA

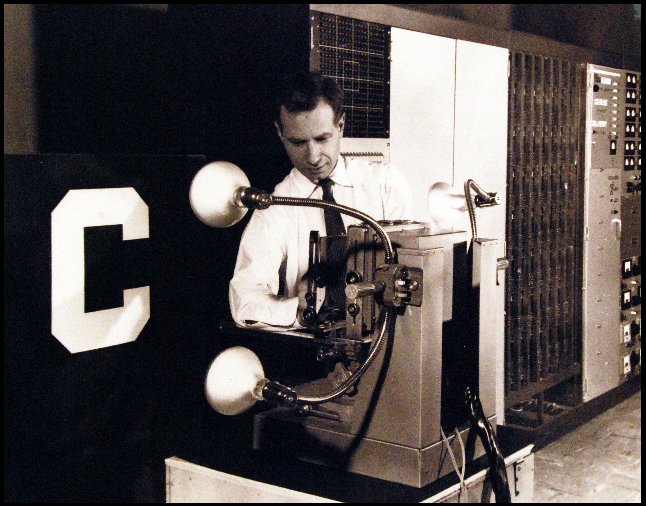
T/F: we need to center the dataset before we run PCA

Q: How to pick the parameter K in PCA?

$$XX^T = U \Lambda U^T$$



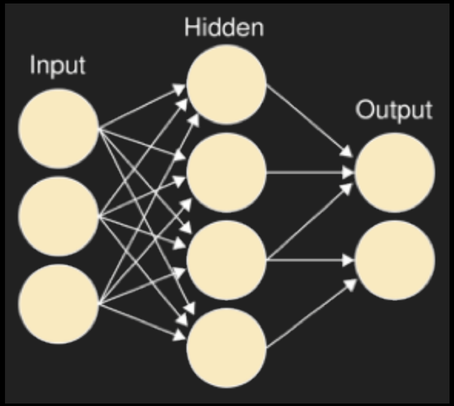
Perceptron, 1957

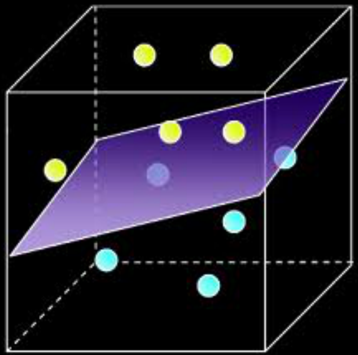


Frank Rosenblatt
@ Cornell!

Predecessor of deep networks.

Separating two classes of objects using a linear threshold classifier.





Perceptron, 1957

NEW NAVY DEVICE LEARNS BY DOING

Psychologist Shows Embryo of Computer Designed to Read and Grow Wiser

WASHINGTON, July 7 (UPI) —The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.

The embryo—the Weather Bureau's $2,000,000 "704" computer—learned to differentiate between right and left after fifty attempts in the Navy's demonstration for newsmen.

The service said it would use this principle to build the first of its Perceptron thinking machines that will be able to read and write. It is expected to be finished in about a year at a cost of $100,000.

Dr. Frank Rosenblatt, designer of the Perceptron, conducted the demonstration. He said the machine would be the first device to think as the human brain. As do human beings, Perceptron will make mistakes at first, but will grow wiser as it gains experience, he said.

Dr. Rosenblatt, a research psychologist at the Cornell Aeronautical Laboratory, Buffalo, said Perceptrons might be fired to the planets as mechanical space explorers.

Without Human Controls

The Navy said the perceptron would be the first non-living mechanism "capable of receiving, recognizing and identifying its surroundings without any human training or control."

The "brain" is designed to remember images and information it has perceived itself. Ordinary computers remember only what is fed into them on punch cards or magnetic tape.

Later Perceptrons will be able to recognize people and call out their names and instantly translate speech in one language to speech or writing in another language, it was predicted.

Mr. Rosenblatt said in principle it would be possible to build brains that could reproduce themselves on an assembly line and which would be conscious of their existence.

In today's demonstration, the "704" was fed two cards, one with squares marked on the left side and the other with squares on the right side.

Learns by Doing

In the first fifty trials, the machine made no distinction between them. It then started registering a "Q" for the left squares and "O" for the right squares.

Dr. Rosenblatt said he could explain why the machine learned only in highly technical terms. But he said the computer had undergone a "self-induced change in the wiring diagram."

The first Perceptron will have about 1,000 electronic "association cells" receiving electrical impulses from an eye-like scanning device with 400 photo-cells. The human brain has 10,000,000,000 responsive cells, including 100,000,000 connections with the eyes.

The New York Times

New Navy Device Learns by Doing

- The New York Times (July 8, 1958)

"Later perceptrons will be able to recognize people and call out their names and instantly translate speech in one language to speech or writing in another language, it was predicted."

https://news.cornell.edu/stories/2019/09/professors-perceptron-paved-way-ai-60-years-too-soon

Today

Objective: learn our first (binary) classification algorithm
and understand why it works

Outline

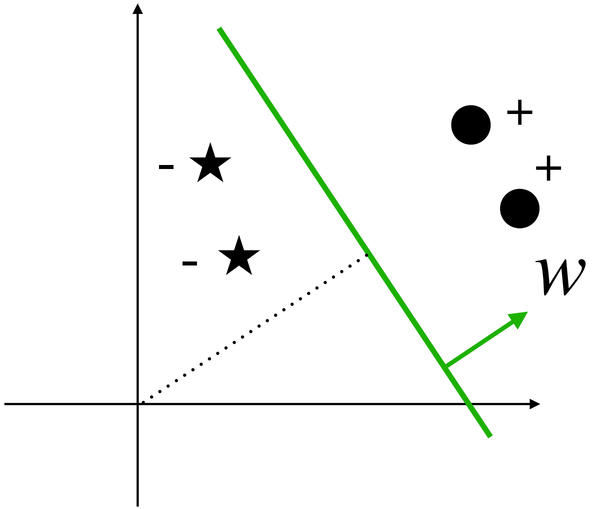
1. Linear binary Classifier

2. Algorithm

3. Proof of why it works

Linear classifier

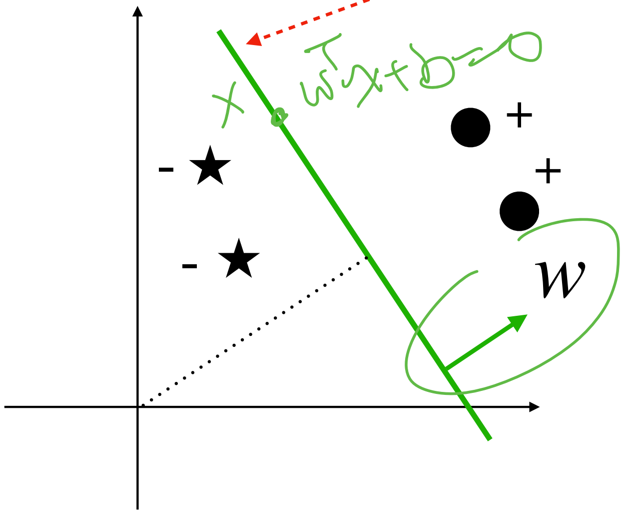
Binary classification setting: $x \in \mathbb{R}^d, y = \{-1, +1\}$



Linear classifier

Binary classification setting: $x \in \mathbb{R}^d, y = \{-1, +1\}$

Hyperplane $H = \{x : w^\top x + b = 0\}$



weight vector

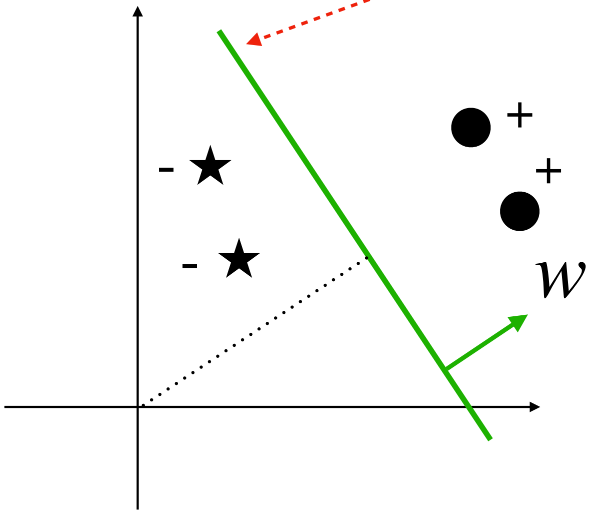
$b \in \mathbb{R}$

$w \in \mathbb{R}^d$

Linear classifier

Binary classification setting: $x \in \mathbb{R}^d, y = \{-1, +1\}$

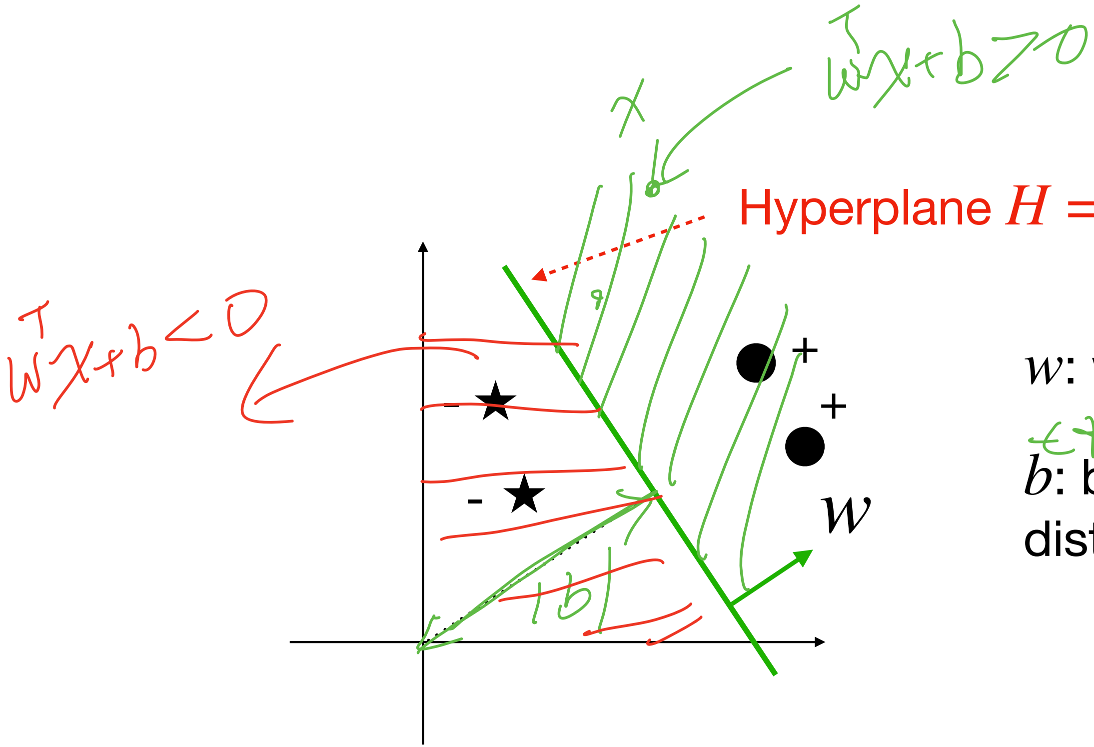
Hyperplane $H = \{x : w^\top x + b = 0\}$



w: weight vector, wlog assume $\|w\|_2 = 1$

Linear classifier

Binary classification setting: $x \in \mathbb{R}^d, y = \{-1, +1\}$



w: weight vector, wlog assume $\|w\|_2 = 1$

b: bias term; $|b|$ determines the distance of the hyperplane to origin

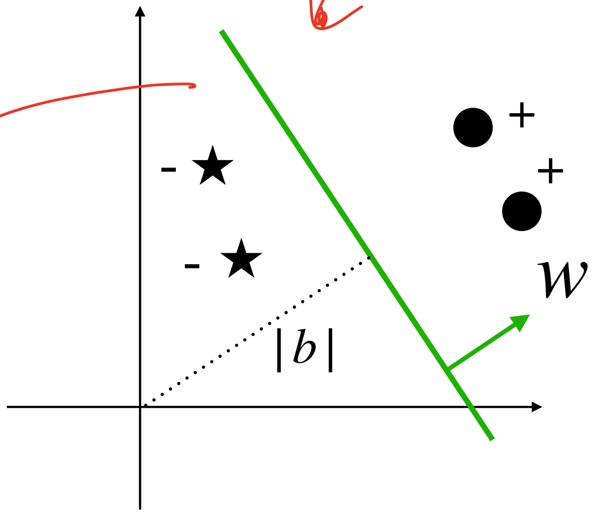
$\in \mathbb{R}$

Linear classifier

Binary classification setting: $x \in \mathbb{R}^d, y = \{-1, +1\}$

$w^T x + b > 0$

$w^T x + b < 0$



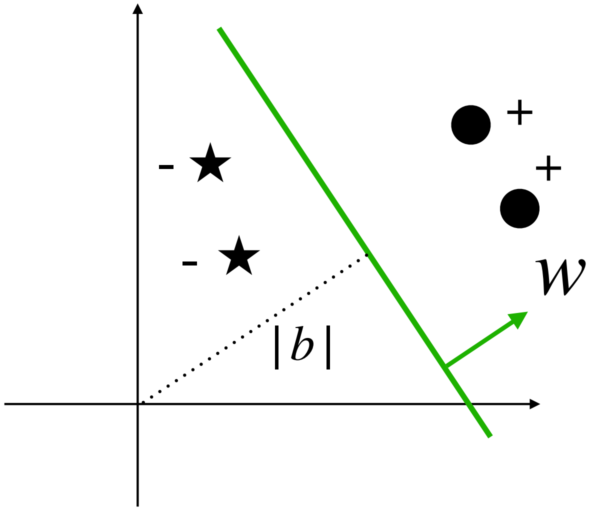
A Hyperplane defines a binary linear classifier

$$\text{sign}(w^\top x + b)$$

Setting

$y_i \in \{+1, -1\}$

We often assume data $\{x_i, y_i\}_{i=1}^n$ is linearly separable,

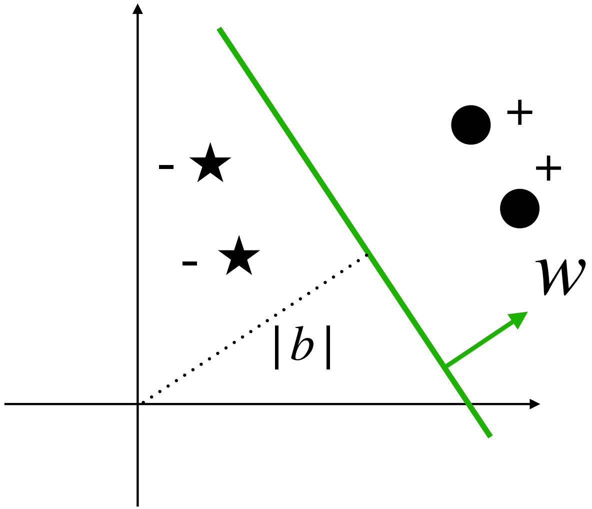


i.e., $\exists w^\star, b^\star$, such that

$$\mathrm{sign}((w^\star)^\top x_i + b^\star) = \mathrm{sign}(y_i), \forall i$$

Setting

We often assume data $\{x_i, y_i\}_{i=1}^n$ is linearly separable,



i.e., $\exists w^\star, b^\star$, such that

$$\mathrm{sign}((w^\star)^\top x_i + b^\star) = \mathrm{sign}(y_i), \forall i$$

Or equivalently,

$$y_i((w^\star)^\top x_i + b^\star) > 0, \forall i$$

$$y_i(w^{\star T} x_i + b^\star) < 0$$

Linear classifier

Absorbing the bias term into the feature vector

$$w^\top x + b = \begin{bmatrix} w \\ b \end{bmatrix}^\top \begin{bmatrix} x \\ 1 \end{bmatrix}$$

$\tilde{w}$ $\tilde{x}$ $\tilde{w}^T \tilde{x}$

$\text{sign}(w^T x + b) = \text{sign}(\tilde{w}^T \tilde{x})$

Linear classifier

Absorbing the bias term into the feature vector

$$w^\top x + b = \begin{bmatrix} w \\ b \end{bmatrix}^\top \begin{bmatrix} x \\ 1 \end{bmatrix}$$

Throughout the semester, we will assume feature x in default contains the constant 1

$\mathrm{sign}(w^\top x)$

Outline

1. Linear binary Classifier

2. Algorithm

3. Proof of why it works

The learning protocol

Consider the **online learning** setting where every iteration t, a pair (x_t, y_t) shows up

For $t = 0 \to \infty$

The learning protocol

Consider the **online learning** setting where every iteration t, a pair (x_t, y_t) shows up

For $t = 0 \to \infty$

New feature x_t shows up

The learning protocol

Consider the **online learning** setting where every iteration t, a pair (x_t, y_t) shows up

$w_t^\top x$

For $t = 0 \rightarrow \infty$

New feature x_t shows up

Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$

The learning protocol

Consider the **online learning** setting where every iteration t, a pair (x_t, y_t) shows up

For $t = 0 \to \infty$

New feature x_t shows up

Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$

Check if $\hat{y}_t = y_t$

The learning protocol

Consider the **online learning** setting where every iteration t, a pair (x_t, y_t) shows up

For $t = 0 \rightarrow \infty$

New feature x_t shows up

Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$

Check if $\hat{y}_t = y_t$

Alg updates w_{t+1}

The learning protocol

Consider the **online learning** setting where every iteration t, a pair (x_t, y_t) shows up

For $t = 0 \to \infty$

New feature x_t shows up

Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$

Check if $\hat{y}_t = y_t$

Alg updates w_{t+1}

Goal: make # of mistakes $\sum_{t=0}^{\infty} \mathbf{1}(\hat{y}_t \neq y_t)$ as small as possible

The learning protocol

Consider the **online learning** setting where every iteration t, a pair (x_t, y_t) shows up

For $t = 0 \to \infty$

New feature x_t shows up

Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$

Check if $\hat{y}_t = y_t$

Alg updates w_{t+1}

Perceptron tells us how to do this update!

Goal: make # of mistakes $\sum_{t=0}^{\infty} \mathbf{1}(\hat{y}_t \neq y_t)$ as small as possible

The Algorithm

Initialize $w_0 = \mathbf{0}$

For $t = 0 \rightarrow \infty$

New feature x_t shows up

Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$

Check if $\hat{y}_t = y_t$

The Algorithm

Initialize $w_0 = \mathbf{0}$

Initialization
the zero vector

For $t = 0 \rightarrow \infty$

New feature x_t shows up

Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$

Check if $\hat{y}_t = y_t$

Alg updates $w_{t+1} = w_t + \mathbf{1}(\hat{y}_t \neq y_t) y_t x_t$

The Algorithm

Initialize $w_0 = \mathbf{0}$

For $t = 0 \to \infty$

New feature x_t shows up

Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$

Check if $\hat{y}_t = y_t$

Alg updates $w_{t+1} = w_t + \mathbf{1}(\hat{y}_t \neq y_t) y_t x_t$

Case 1: $\hat{y}_t = y_t$, $w_{t+1} = w_t$

The Algorithm

Initialize $w_0 = \mathbf{0}$

For $t = 0 \rightarrow \infty$

> New feature x_t shows up
>
> Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$
>
> Check if $\hat{y}_t = y_t$
>
> Alg updates $w_{t+1} = w_t + \mathbf{1}(\hat{y}_t \neq y_t) y_t x_t$

$w_t^\top x_t$ was negative

Case 1: $\hat{y}_t = y_t$, $w_{t+1} = w_t$

Case 2: $\hat{y}_t \neq y_t$ (e.g., $\hat{y}_t = -1, y_t = 1$)

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

$$= w_t + x_t$$

$$w_{t+1}^\top x_t - w_t^\top x_t$$

$$= x_t^\top x_t > 0$$

The Algorithm

Initialize $w_0 = \mathbf{0}$

For $t = 0 \to \infty$

- New feature x_t shows up
- Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$
- Check if $\hat{y}_t = y_t$
- Alg updates $w_{t+1} = w_t + \mathbf{1}(\hat{y}_t \neq y_t) y_t x_t$

Case 1: $\hat{y}_t = y_t$, $w_{t+1} = w_t$

Case 2: $\hat{y}_t \neq y_t$ (e.g., $\hat{y}_t = -1, y_t = 1$)

$$w_{t+1}^\top x_t - w_t^\top x_t = (x_t^\top x_t)$$

The Algorithm

Initialize $w_0 = \mathbf{0}$

For $t = 0 \to \infty$

New feature x_t shows up

Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$

Check if $\hat{y}_t = y_t$

Alg updates $w_{t+1} = w_t + \mathbf{1}(\hat{y}_t \neq y_t) y_t x_t$

Case 1: $\hat{y}_t = y_t$, $w_{t+1} = w_t$

Case 2: $\hat{y}_t \neq y_t$ (e.g., $\hat{y}_t = -1, y_t = 1$)

$$w_{t+1}^\top x_t - w_t^\top x_t = (x_t^\top x_t)$$

Value of $w_{t+1}^\top x_t$ is increased
(the correct progress)

The Algorithm

Initialize $w_0 = \mathbf{0}$

For $t = 0 \to \infty$

New feature x_t shows up

Alg makes a prediction $\hat{y}_t = \text{sign}(w_t^\top x_t)$

Check if $\hat{y}_t = y_t$

Alg updates $w_{t+1} = w_t + \mathbf{1}(\hat{y}_t \neq y_t) y_t x_t$

Case 1: $\hat{y}_t = y_t$, $w_{t+1} = w_t$

Case 2: $\hat{y}_t \neq y_t$ (e.g., $\hat{y}_t = -1, y_t = 1$)

$$w_{t+1}^\top x_t - w_t^\top x_t = (x_t^\top x_t)$$

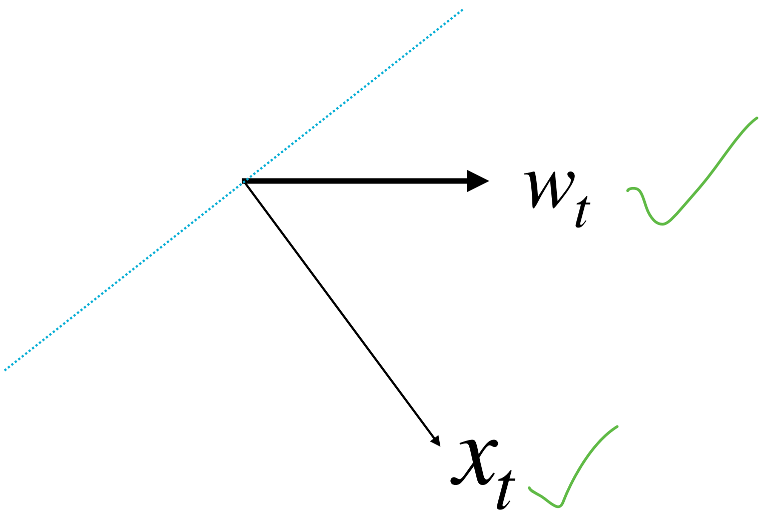
Value of $w_{t+1}^\top x_t$ is increased (the correct progress)

Q: what happens when $\hat{y}_t = 1, y_t = -1$

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

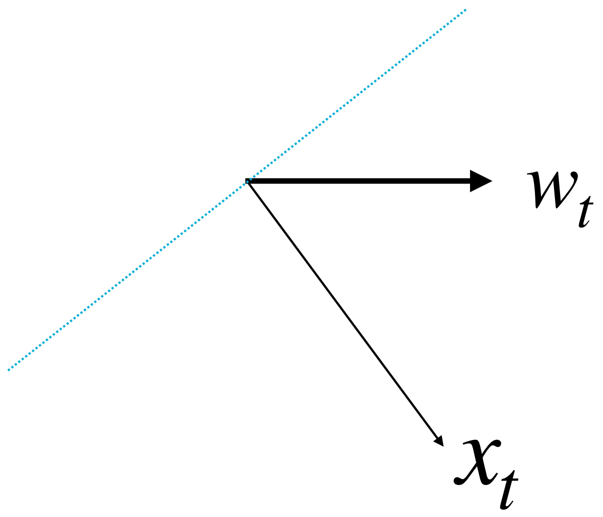
A Geometric explanation

When we make a mistake, i.e., $y_t(w_t^\top x_t) < 0$ (e.g., $y_t = -1, w_t^\top x_t > 0$)



A Geometric explanation

When we make a mistake, i.e., $y_t(w_t^\top x_t) < 0$ (e.g., $y_t = -1$, $w_t^\top x_t > 0$)



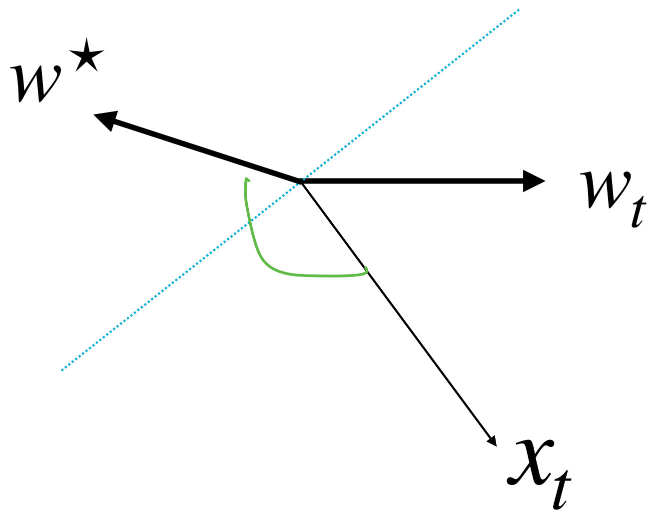
Q: What does $w^\star$ look like?

A Geometric explanation

When we make a mistake, i.e., $y_t(w_t^\top x_t) < 0$ (e.g., $y_t = -1$, $w_t^\top x_t > 0$)

$\text{sign}(w^{*\top} x_t) < 0$

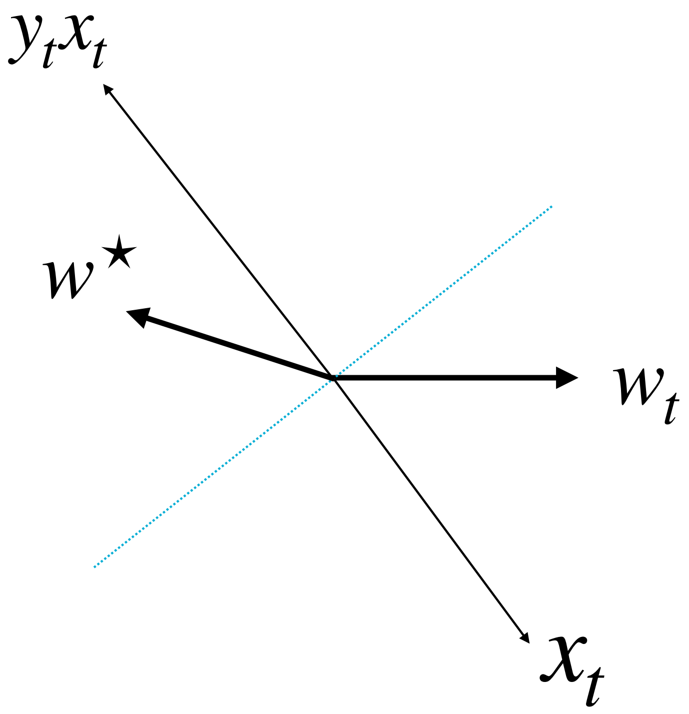
$w_{t+1} = w_t + y_t x_t$



Q: What does $w^\star$ look like?

A Geometric explanation

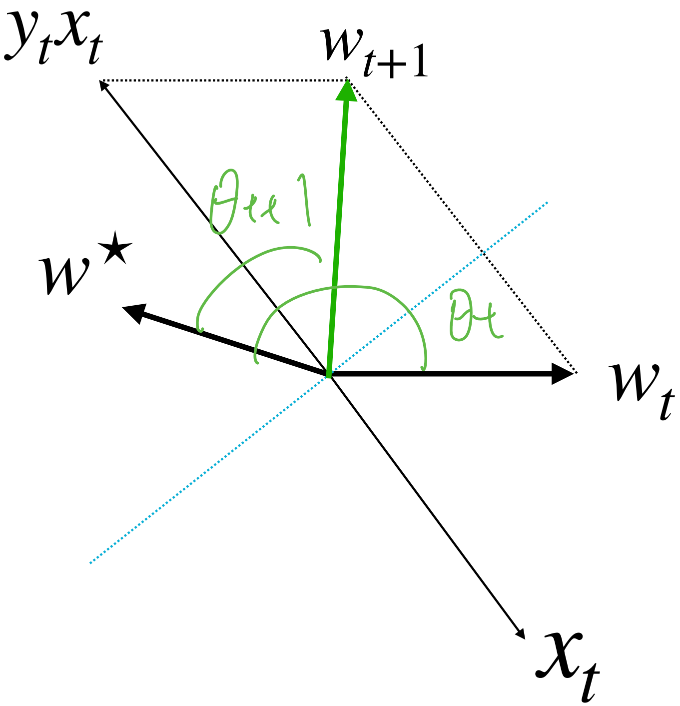
When we make a mistake, i.e., $y_t(w_t^\top x_t) < 0$ (e.g., $y_t = -1$, $w_t^\top x_t > 0$)



Q: What does $w^\star$ look like?

A Geometric explanation

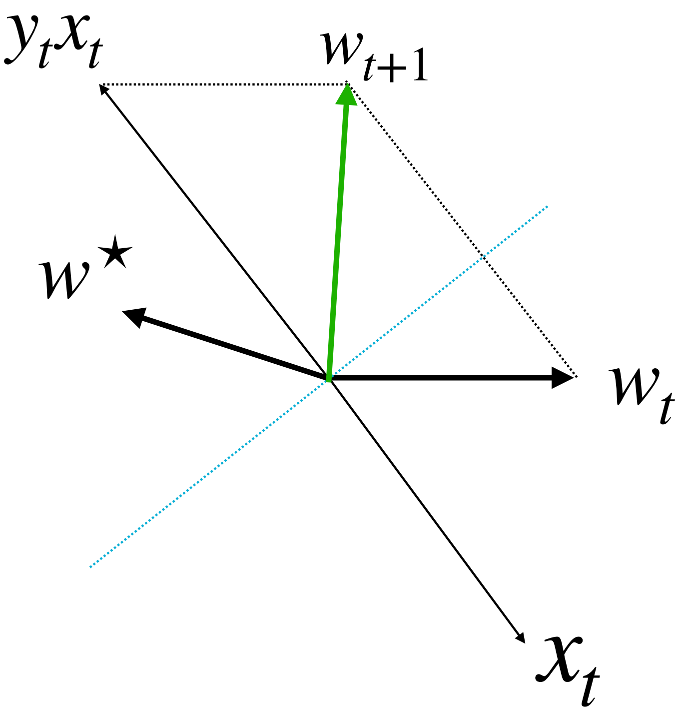
When we make a mistake, i.e., $y_t(w_t^\top x_t) < 0$ (e.g., $y_t = -1$, $w_t^\top x_t > 0$)



Q: What does $w^\star$ look like?

A Geometric explanation

When we make a mistake, i.e., $y_t(w_t^\top x_t) < 0$ (e.g., $y_t = -1$, $w_t^\top x_t > 0$)



We should track how the $\cos(\theta_t)$ is changing:

$$\cos(\theta_t) = \frac{w_t^\top w^\star}{\|w_t\|_2}$$

$= 1$

$w_t^\top w^\star = \cos(\theta_t) \|w_t\|_2 \|w^\star\|_2$

Q: What does $w^\star$ look like?

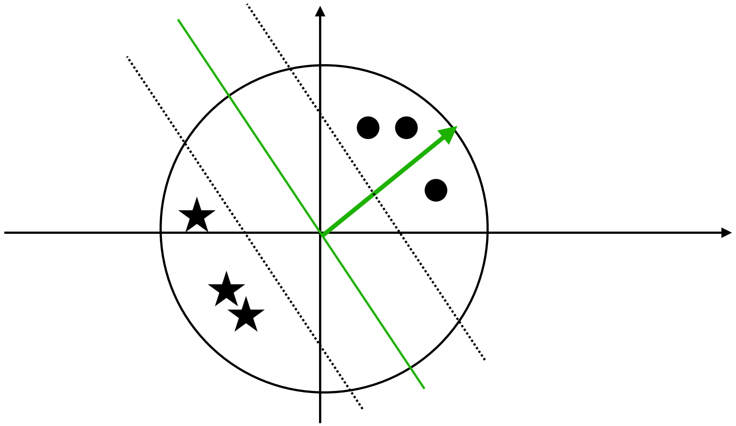
Outline

1. Linear binary Classifier

2. Algorithm

3. Proof of why it works

Main theorem

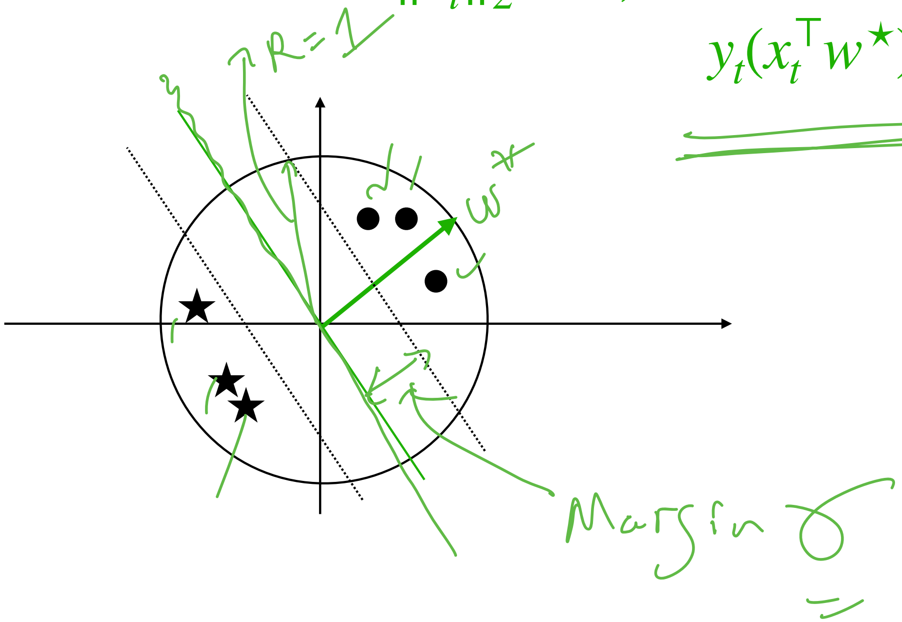


Main theorem

Theorem of Perceptron:

Assume $\|x_t\|_2 \leq 1, \forall t$. If there exists $w^\star$ with $\|w^\star\|_2 = 1$, such that

$$y_t(x_t^\top w^\star) \geq \gamma > 0, \forall t,$$



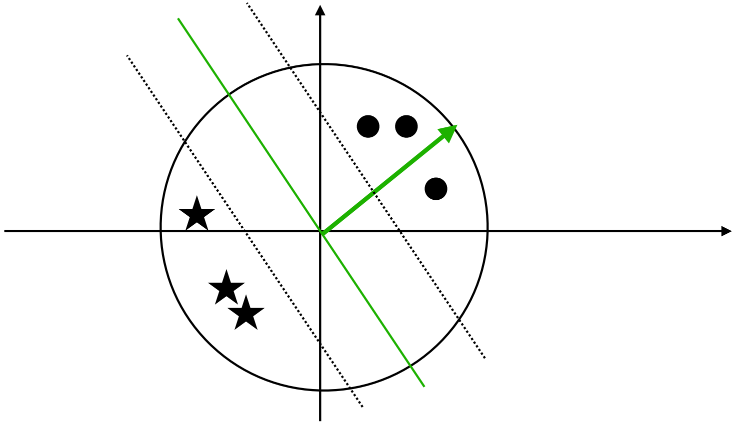
$\gamma \in \mathbb{R}^+$

Main theorem

Theorem of Perceptron:

Assume $\|x_t\|_2 \leq 1, \forall t$. If there exists $w^\star$ with $\|w^\star\|_2 = 1$, such that

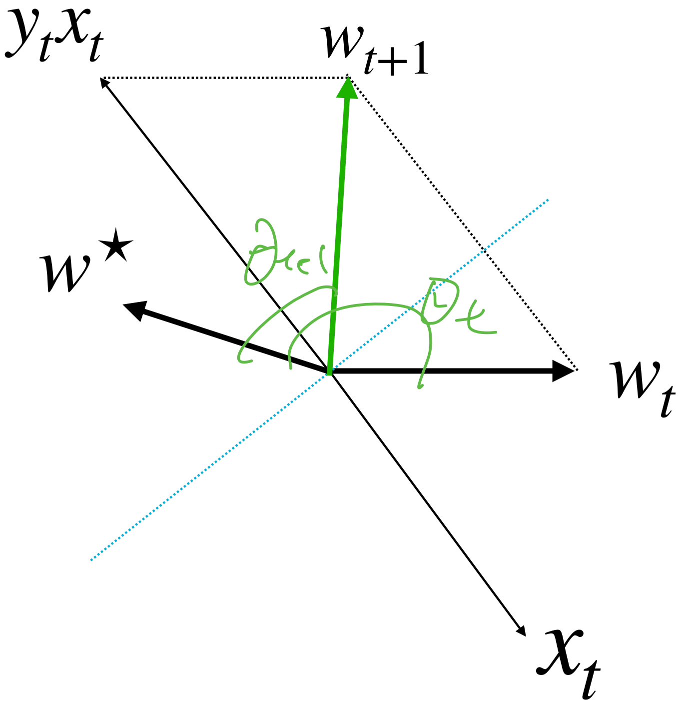
$$y_t(x_t^\top w^\star) \geq \gamma > 0, \forall t,$$



then:

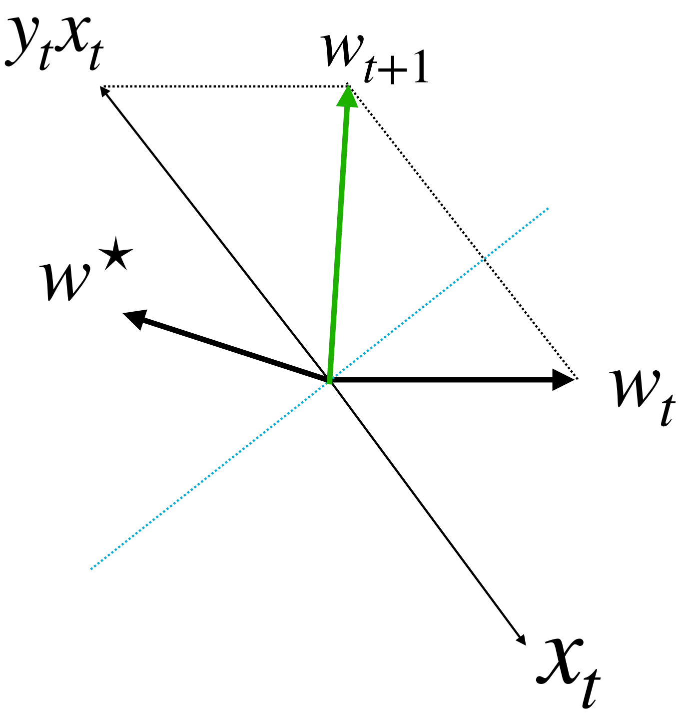
$$\sum_{t=0}^{\infty} \mathbf{1}(\hat{y}_t \neq y_t) \leq 1/\gamma^2$$

Proof of the theorem



$$\cos(\theta_t) = \frac{w_t^\top w^\star}{\|w_t\|_2}$$

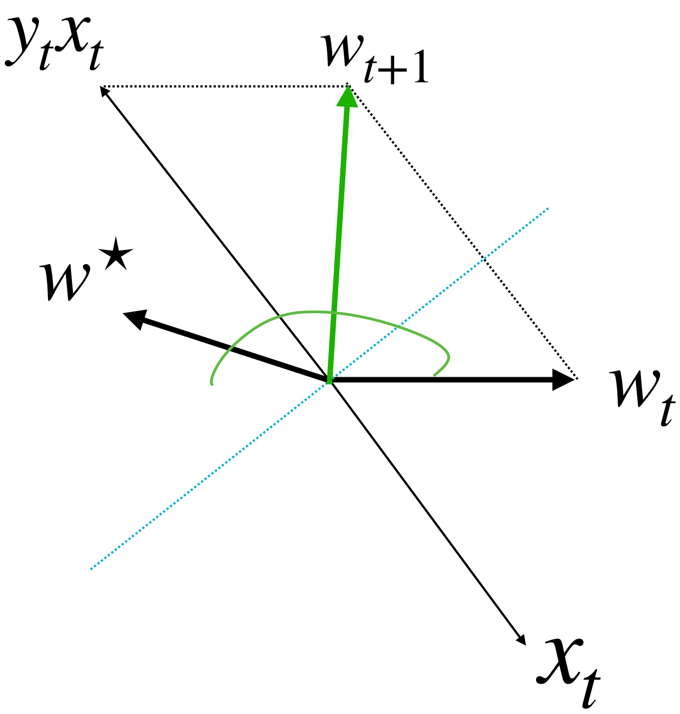
Proof of the theorem



$$\cos(\theta_t) = \frac{w_t^\top w^\star}{\|w_t\|_2}$$

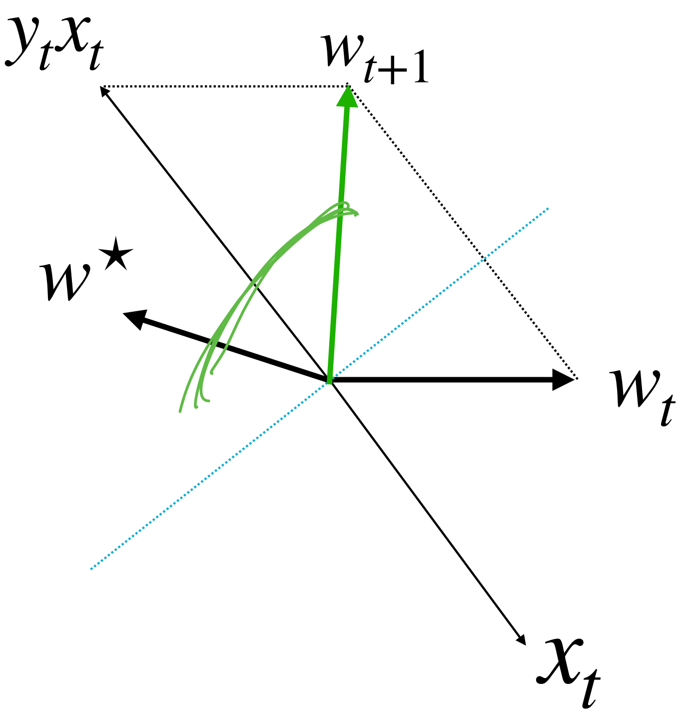
Assume we make a mistake at x_t, track how the denominator and numerator change

Proof of the theorem



1. Track $w_t^\top w^\star$

Proof of the theorem

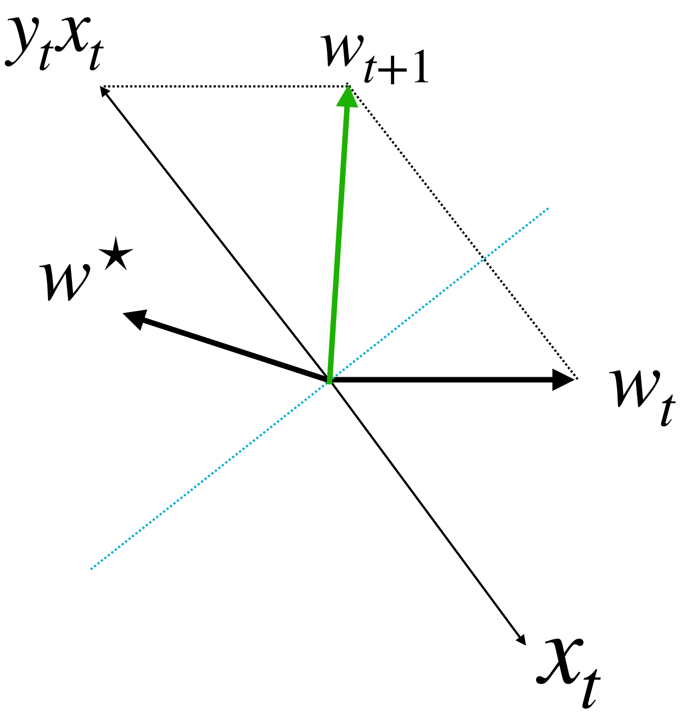


1. Track $w_t^\top w^\star$

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

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

Proof of the theorem



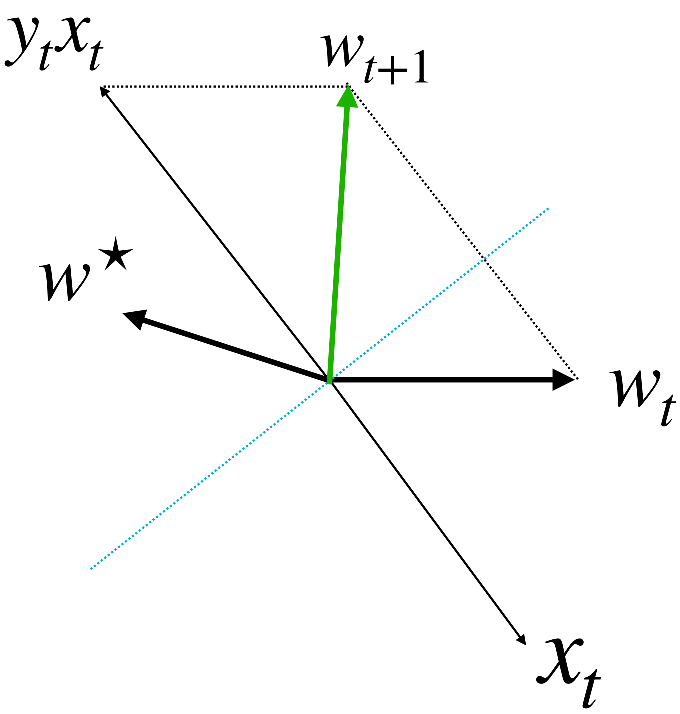
1. Track $w_t^\top w^\star$

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

$$= w_t^\top w^\star + y_t x_t^\top w^\star$$

$$y_t (w^{*T} x_t) \geq \gamma$$

Proof of the theorem



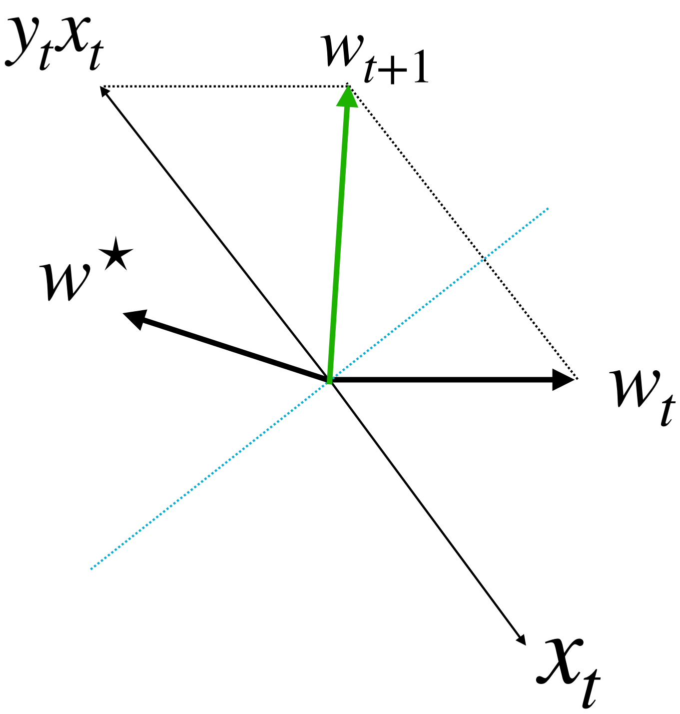
1. Track $w_t^\top w^\star$

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

$$= w_t^\top w^\star + y_t x_t^\top w^\star$$

$$\geq w_t^\top w^\star + \gamma$$

Proof of the theorem



1. Track $w_t^\top w^\star$

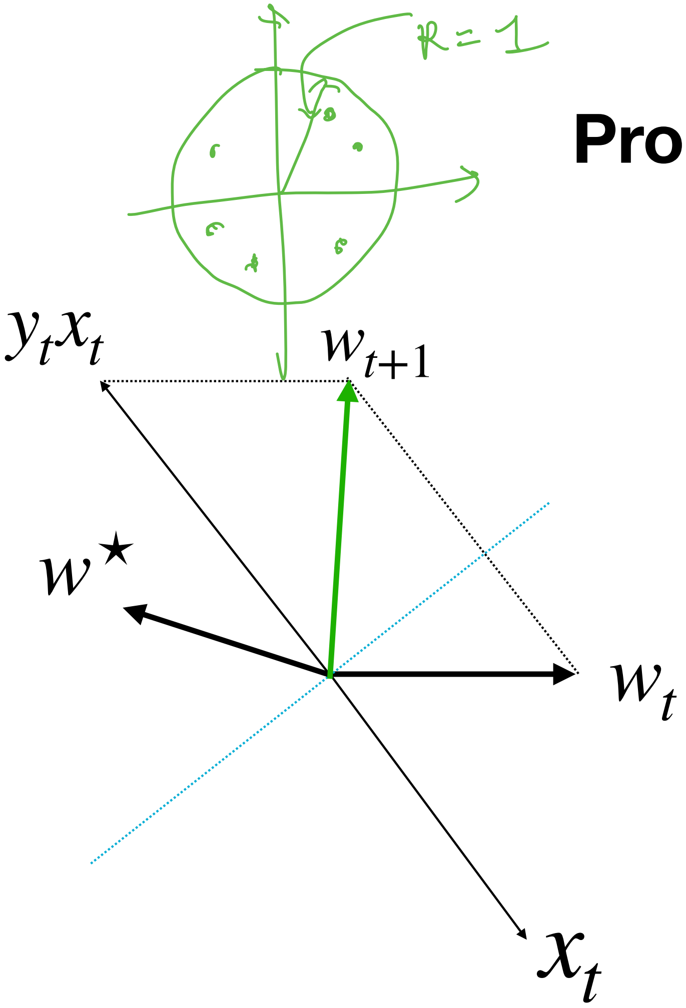
$$w_{t+1}^\top w^\star = (w_t + y_t x_t)^\top w^\star$$

$$= w_t^\top w^\star + y_t x_t^\top w^\star$$

$$\geq w_t^\top w^\star + \gamma$$

Whenever we make a mistake, $w_t^\top w^\star$ at least increased by γ

Proof of the theorem



update rule:

$W_{t+1} = W_t + y_t x_t$

2. Track $w_t^\top w_t$

$$w_{t+1}^\top w_{t+1} = (w_t + y_t x_t)^\top (w_t + y_t x_t)$$

$y_t^2 x_t^\top x_t$

$= 1$

$$= w_t^\top w_t + 2 w_t^\top (x_t y_t) + x_t^\top x_t$$

< 0

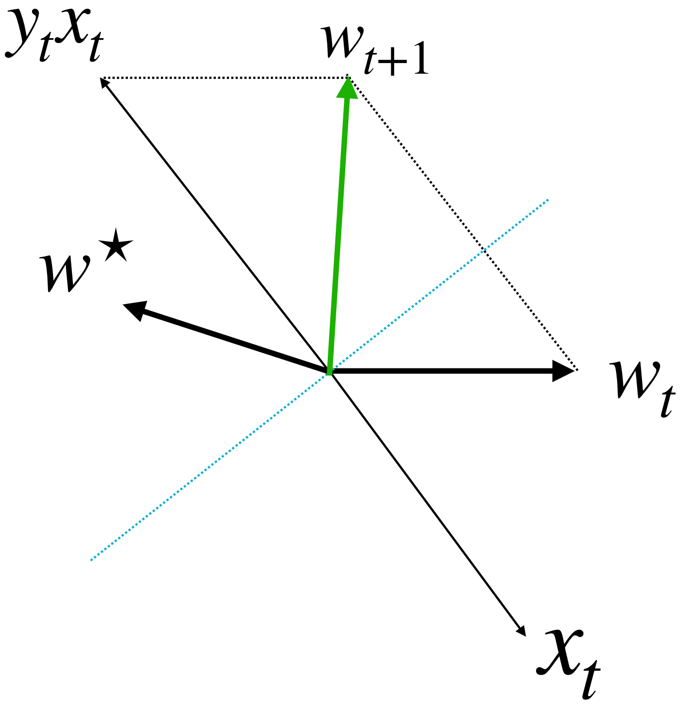
$\|x_t\|_2 \le 1$

$<$

$$\le w_t^\top w_t + 1$$

Discuss this derivation in small group for 5 minutes!

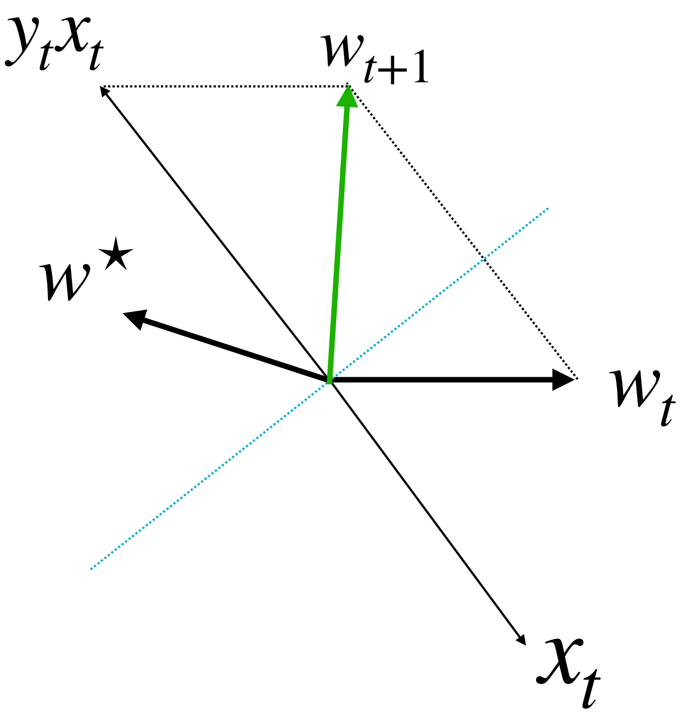
Proof of the theorem



3. What is $cos(\theta_t) = w_t^\top w^\star / \sqrt{w_t^\top w_t}$ if we have made M mistakes?

After make M mistakes:

Proof of the theorem

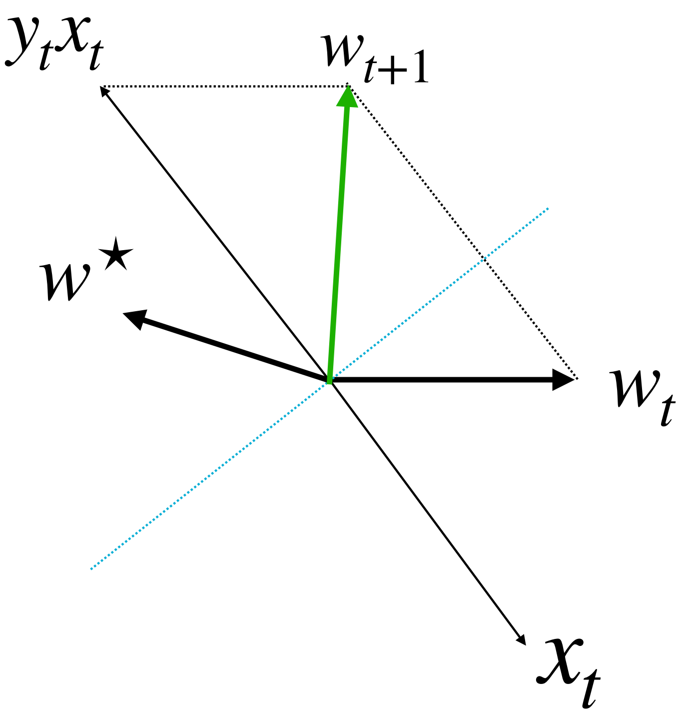


3. What is $cos(\theta_t) = w_t^\top w^\star / \sqrt{w_t^\top w_t}$ if we have made M mistakes?

After make M mistakes:

$$w_t^\top w^\star \geq M\gamma$$

Proof of the theorem



3. What is $cos(\theta_t) = w_t^\top w^\star / \sqrt{w_t^\top w_t}$ if we have made M mistakes?

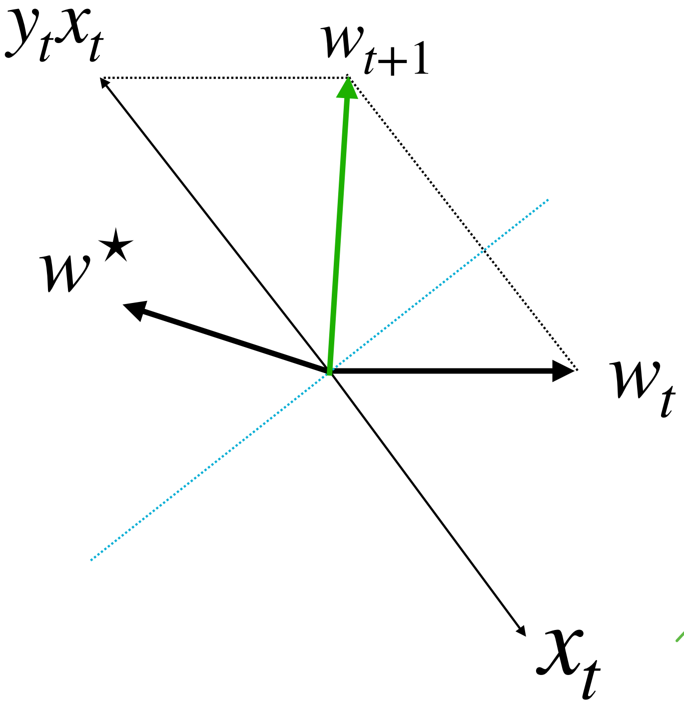
After make M mistakes:

$$w_t^\top w^\star \geq M\gamma$$

$$w_t^\top w_t \leq M$$

$$1 \geq \cos(\theta_t) = \frac{w_t^\top w^\star}{\|w_t\|_2} \geq \frac{M\gamma}{\sqrt{M}}$$

Proof of the theorem



3. What is $cos(\theta_t) = w_t^\top w^\star / \sqrt{w_t^\top w_t}$ if we have made M mistakes?

After make M mistakes:

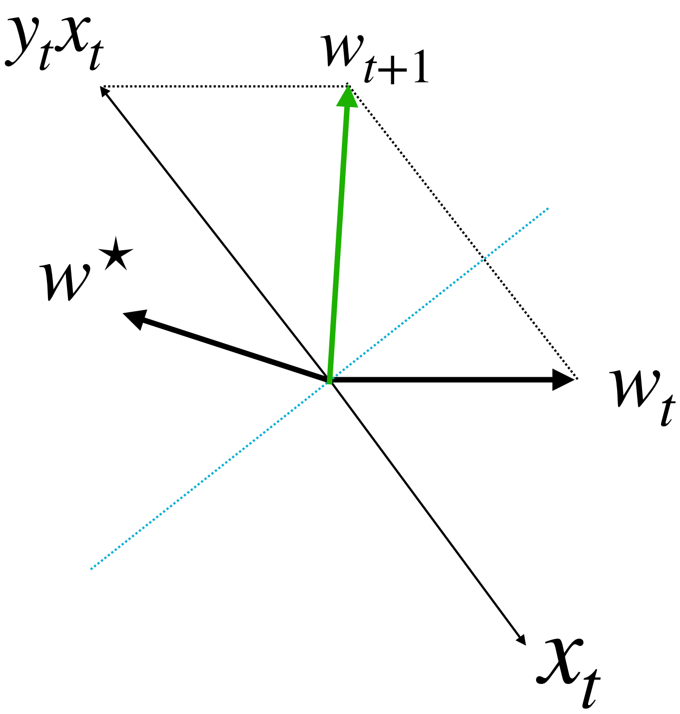
$$w_t^\top w^\star \geq M\gamma$$

$$w_t^\top w_t \leq M$$

$$1 \geq \cos(\theta_t) \geq (M\gamma)/\sqrt{M} = \sqrt{M}\gamma$$

$1 \geq \sqrt{m} \cdot \gamma$

Proof of the theorem



3. What is $cos(\theta_t) = w_t^\top w^\star / \sqrt{w_t^\top w_t}$ if we have made M mistakes?

After make M mistakes:

$$w_t^\top w^\star \geq M\gamma$$

$$w_t^\top w_t \leq M$$

$$1 \geq \cos(\theta_t) \geq (M\gamma)/\sqrt{M} = \sqrt{M}\gamma$$

$$\Rightarrow M \leq 1/\gamma^2$$

Summary

$(x+y)^T(x+y)$

$= x^Tx + y^Ty + 2x^Ty$

$= x^Tx + y^Ty + x^Ty + y^Tx$

The Perceptron algorithm:

1. Binary classification algorithm, runs in online mode, makes update when makes a mistake

(See lecture note for how to apply Perceptron on a static dataset)

2. Total # of mistakes is bounded by a constant $(1/\gamma^2)$