CS 3/5780: Introduction to Machine Learning

Perceptron

Instructor: John Thickstun

Reading: UML 9.1.1-9.1.2

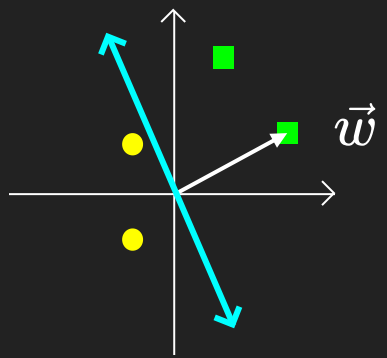
The Perceptron Algorithm (homogeneous & batch)

- **Input:** training data $\mathcal{S} = \{(\vec{x}_1, y_1), ..., (\vec{x}_m, y_m)\}$
- **Initialize** $\vec{w}^{(0)} = (0, ..., 0)$ and $t = 0$
- **While** there is $i \in [m]$ such that $y_i(\vec{w}^{(t)} \cdot \vec{x}_i) \leq 0$ then,
  - $\vec{w}^{(t+1)} = \vec{w}^{(t)} + y_i \vec{x}_i$
  - $t \leftarrow t + 1$
- **End While**
- **Output** $\vec{w}^{(t)}$

Interactive Demo: https://mlweb.loria.fr/book/en/perceptron.html

So far, we've seen...

- Terminates when $h_{\vec{w}}$ is consistent with $\mathcal{S}$
- Changing the data order
  - can lead to faster convergence
  - can lead to a different classifier
- In our examples, always converges in finite steps



The Perceptron Algorithm (homogeneous & batch)

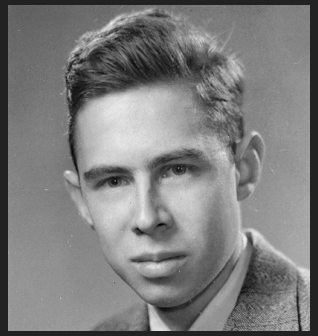
- **Input:** training data $\mathcal{S} = \{(\vec{x}_1, y_1), ..., (\vec{x}_m, y_m)\}$
- **Initialize** $\vec{w}^{(0)} = (0, ..., 0)$ and $t = 0$
- **While** there is $i \in [m]$ such that $y_i(\vec{w}^{(t)} \cdot \vec{x}_i) \leq 0$ then,
  - $\vec{w}^{(t+1)} = \vec{w}^{(t)} + y_i\vec{x}_i$
  - $t \leftarrow t + 1$
- **End While**
- **Output** $\vec{w}^{(t)}$

NEW NAVY DEVICE LEARNS BY DOING

Psychologist Shows Embryo of Computer Designed to Read and Grow Wiser

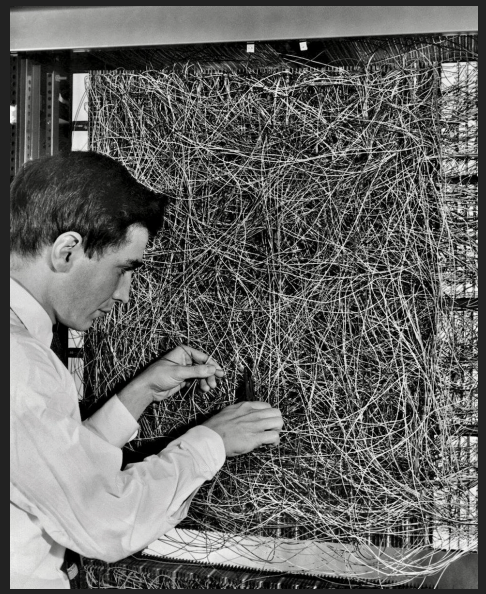
WASHINGTON, July 7 (UPI) —The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.

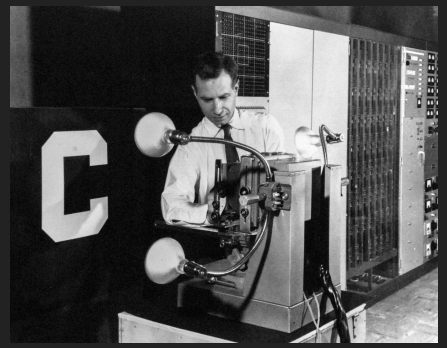
New York Times, 1958



Frank Rosenblatt

IBM 704





MARK I Perceptron, 1960

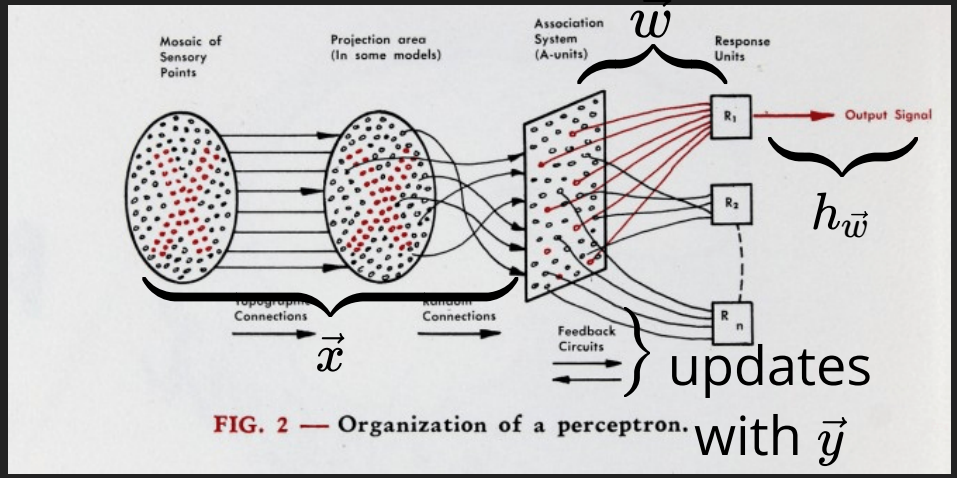


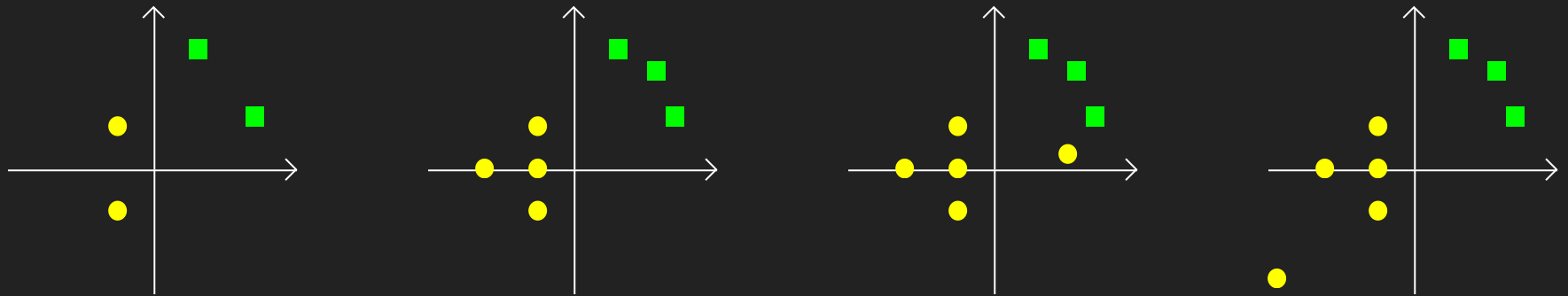
FIG. 2 — Organization of a perceptron.

Outline

1. ~~Recap & History~~
2. Convergence Theorem
3. Convergence Proof
4. Online Perceptron

Convergence of Perceptron

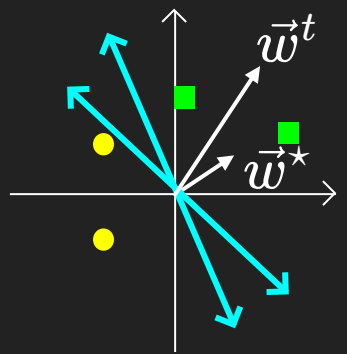
- **Theorem:** Given a dataset $\mathcal{S} = \{(\vec{x}_1, y_1), ..., (\vec{x}_m, y_m)\}$ and a radius R such that $\|\vec{x}_i\| \leq R$ for all $i \in [m]$. If $\mathcal{S}$ is linearly separable with (geometric) **margin** γ, then Perceptron makes at most R^2/γ^2 updates before finding a consistent linear classifier.
- *As long as the data is linearly separable with a margin, then the Perceptron algorithm terminates after finitely many updates*



Proof Idea

- Since $\mathcal{S}$ is linearly separable with **margin** γ, there is a $h_{\vec{w}^\star}$ with γ margin
  - this means $\|\vec{w}^\star\|_2 = 1$ and $y_i(\vec{w}^\star \cdot \vec{x}_i) \geq \gamma$ for all $i \in [m]$
- We will show that within $t = R^2/\gamma^2$ steps, $\vec{w}^{(t)}$ is close to $\vec{w}^\star$ in angle
- Recall that the angle is given by

$$\cos\left(\theta_{\vec{w}^\star, \vec{w}^{(t)}}\right) = \frac{\vec{w}^\star \cdot \vec{w}^{(t)}}{\|\vec{w}^\star\| \|\vec{w}^{(t)}\|}$$

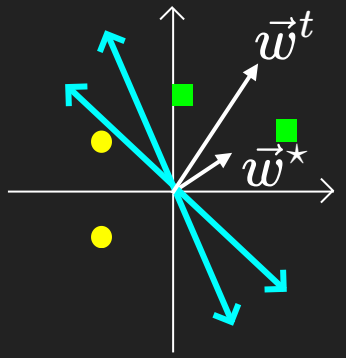


Outline

1. ~~Recap & History~~
2. ~~Convergence Theorem~~
3. Convergence Proof
4. Online Perceptron

Proof Outline

- Goal: Prove that Perceptron terminates after at most R^2/γ^2 steps
- Step 1: Show that $\vec{w}^\star \cdot \vec{w}^{(t)}$ is large (find a lower bound)
- Step 2: Show that $\|\vec{w}^{(t)}\|$ is not too large (find an upper bound)
- Step 3: combine 1 & 2 to show that if there are more than R^2/γ^2 updates, then the cosine becomes larger than one!
  - Proof by contradiction



$$\cos\left(\theta_{\vec{w}^\star,\vec{w}^{(t)}}\right) = \frac{\vec{w}^\star \cdot \vec{w}^{(t)}}{\|\vec{w}^\star\| \|\vec{w}^{(t)}\|}$$

Step 1: $\vec{w}^\star \cdot \vec{w}^{(t)}$ is large

- Goal: lower bound $\vec{w}^\star \cdot \vec{w}^{(t)}$ in terms of γ and t
- First, bound $\vec{w}^\star \cdot \vec{w}^{(t)}$ in terms of $\vec{w}^\star \cdot \vec{w}^{(t-1)}$ and γ
  - Let $(\vec{x}_i, y_i)$ be the mistake made at iteration t
  - $\vec{w}^\star \cdot \vec{w}^{(t)} = \vec{w}^\star \cdot (\vec{w}^{(t-1)} + y_i \vec{x}_i)$
    - *Definition of perceptron update*

      $= \vec{w}^\star \cdot \vec{w}^{(t-1)} + y_i \vec{w}^\star \cdot \vec{x}_i$
    - *Distributive property of dot products, commutative scalars*

      $\geq \vec{w}^\star \cdot \vec{w}^{(t-1)} + \gamma$
    - *Definition of $\vec{w}^\star$ and margin*
- Then, recurse to bound $\vec{w}^\star \cdot \vec{w}^{(t)}$ in terms of γ and t

Step 1: $\vec{w}^\star \cdot \vec{w}^{(t)}$ is large

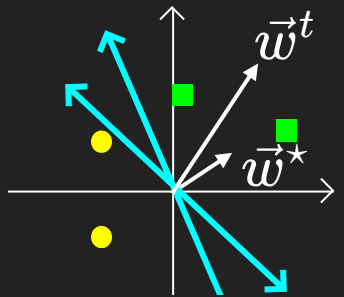
- Goal: lower bound $\vec{w}^\star \cdot \vec{w}^{(t)}$ in terms of γ and t
- First, bound $\vec{w}^\star \cdot \vec{w}^{(t)}$ in terms of $\vec{w}^\star \cdot \vec{w}^{(t-1)}$ and γ
  - $\vec{w}^\star \cdot \vec{w}^{(t)} \geq \vec{w}^\star \cdot \vec{w}^{(t-1)} + \gamma$
- Then, recurse to bound $\vec{w}^\star \cdot \vec{w}^{(t)}$ in terms of γ and t
  - $\vec{w}^\star \cdot \vec{w}^{(t)} \geq \vec{w}^\star \cdot \vec{w}^{(t-2)} + 2\gamma$
  - $\geq \vec{w}^\star \cdot \vec{w}^{(t-3)} + 3\gamma$

    $\ldots$

    $\geq \vec{w}^\star \cdot \vec{w}^{(0)} + t\gamma$

Proof Outline

- Goal: Prove that Perceptron terminates after at most R^2/γ^2 steps
- Step 1: $\vec{w}^\star \cdot \vec{w}^{(t)} \geq t\gamma$
- Step 2: Show that $\|\vec{w}^{(t)}\|$ is not too large (find an upper bound)
- Step 3: combine 1 & 2 to show that if there are more than R^2/γ^2 updates, then the cosine becomes larger than one!
  - Proof by contradiction



$$\cos\left(\theta_{\vec{w}^\star,\vec{w}^{(t)}}\right) = \frac{\vec{w}^\star \cdot \vec{w}^{(t)}}{\|\vec{w}^\star\|\|\vec{w}^{(t)}\|}$$

Step 2: $\|\vec{w}^{(t)}\|$ is large

- Goal: upper bound $\|\vec{w}^{(t)}\|$ in terms of R and t
- First, bound $\|\vec{w}^{(t)}\|^2$ in terms of $\|\vec{w}^{(t-1)}\|^2$and R

Let $(\vec{x}_i, y_i)$ be the mistake made at iteration t

1. $\|\vec{w}^{(t)}\|^2 = \|\vec{w}^{(t-1)} + y_i\vec{x}_i\|^2$
2. $= (\vec{w}^{(t-1)} + y_i\vec{x}_i) \cdot (\vec{w}^{(t-1)} + y_i\vec{x}_i)$
3. $= \vec{w}^{(t-1)} \cdot \vec{w}^{(t-1)} + 2y_i(\vec{w}^{(t-1)} \cdot \vec{x}_i) + (y_i)^2\vec{x}_i \cdot \vec{x}_i$
4. $\leq \|\vec{w}^{(t-1)}\|^2 + 0 + R^2$

1. *Definition of perceptron update*
2. *Definition of norm/dot product*
3. *Multiplying and combining terms*
4. *Mistake + radius assumption*

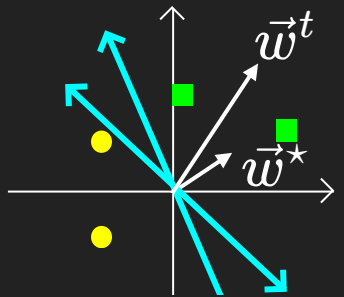
- Then, recurse to bound $\|\vec{w}^{(t)}\|^2$ in terms of R and t

Step 2: $\|\vec{w}^{(t)}\|$ is large

- Goal: upper bound $\|\vec{w}^{(t)}\|$ in terms of R and t
- First, bound $\|\vec{w}^{(t)}\|^2$ in terms of $\|\vec{w}^{(t-1)}\|^2$and R
  - $\|\vec{w}^{(t)}\|^2 \leq \|\vec{w}^{(t-1)}\|^2 + R^2$
- Then, recurse to bound $\|\vec{w}^{(t)}\|^2$ in terms of R and t
  - $\|\vec{w}^{(t)}\|^2 \leq \|\vec{w}^{(t-2)}\|^2 + 2R^2$
  - ...
  - $\|\vec{w}^{(t)}\|^2 \leq \|\vec{w}^{(0)}\|^2 + tR^2$

Proof Outline

- Goal: Prove that Perceptron terminates after at most R^2/γ^2 steps
- Step 1: $\vec{w}^\star \cdot \vec{w}^{(t)} \geq t\gamma$
- Step 2: $\|\vec{w}^{(t)}\|^2 \leq tR^2$
- Step 3: combine 1 & 2 to show that if there are more than R^2/γ^2 updates, then the cosine becomes larger than one!
  - Proof by contradiction



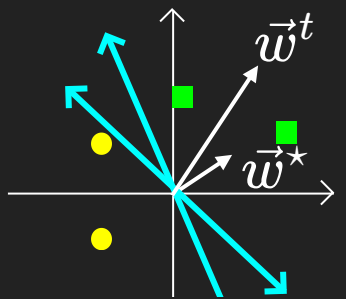
$$\cos\left(\theta_{\vec{w}^\star,\vec{w}^{(t)}}\right) = \frac{\vec{w}^\star \cdot \vec{w}^{(t)}}{\|\vec{w}^\star\| \|\vec{w}^{(t)}\|}$$

Step 3: Contradiction

- Goal: show that if $t > R^2/\gamma^2$, there is a contradiction
- We will use step 1 ($\vec{w}^\star \cdot \vec{w}^{(t)} \geq t\gamma$) & step 2 ($\|\vec{w}^{(t)}\|^2 \leq tR^2$)
- Starting with our expression for cosine,
  - $$\begin{aligned}\cos\left(\theta_{\vec{w}^\star,\vec{w}^{(t)}}\right) &= \frac{\vec{w}^\star \cdot \vec{w}^{(t)}}{\|\vec{w}^\star\|\|\vec{w}^{(t)}\|} \\ &= \frac{\vec{w}^\star \cdot \vec{w}^{(t)}}{\|\vec{w}^{(t)}\|} \\ &\geq \frac{t\gamma}{\|\vec{w}^{(t)}\|} \\ &\geq \frac{t\gamma}{\sqrt{tR^2}} \\ &\geq \frac{\sqrt{t}\gamma}{R}\end{aligned}$$
    - *Definition of $\vec{w}^\star$*
    - *Step 1*
    - *Step 2*
    - *division*
- If $t > R^2/\gamma^2$,
  - $\cos\left(\theta_{\vec{w}^\star,\vec{w}^{(t)}}\right) > \frac{\sqrt{R^2/\gamma^2}\gamma}{R} = 1 \implies$ contradiction!

Proof Outline

- Goal: Prove that Perceptron terminates after at most R^2/γ^2 steps
- Step 1: $\vec{w}^\star \cdot \vec{w}^{(t)} \geq t\gamma$
- Step 2: $\|\vec{w}^{(t)}\|^2 \leq tR^2$
- Step 3: If $t > R^2/\gamma^2$, $\cos\left(\theta_{\vec{w}^\star, \vec{w}^{(t)}}\right) > 1$
  - This is impossible, so it must be that $t \leq R^2/\gamma^2$



$$\cos\left(\theta_{\vec{w}^\star, \vec{w}^{(t)}}\right) = \frac{\vec{w}^\star \cdot \vec{w}^{(t)}}{\|\vec{w}^\star\| \|\vec{w}^{(t)}\|}$$

Remarks on Convergence

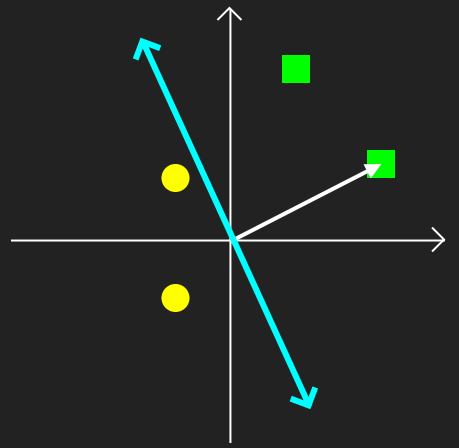
- **Theorem:** Given a dataset $\mathcal{S} = \{(\vec{x}_1, y_1), ..., (\vec{x}_m, y_m)\}$ and a radius R such that $\|\vec{x}_i\| \leq R$ for all $i \in [m]$. If $\mathcal{S}$ is linearly separable with (geometric) **margin** γ, then Perceptron makes at most R^2/γ^2 updates before finding a consistent linear classifier.
- This upper bound holds even if
  - We scale every instance in the training set
  - We shuffle the training set
  - We don't know the value of γ
- Actual number of updates can vary, but we know it cannot be more than R^2/γ^2

Outline

1. ~~Recap & History~~
2. ~~Convergence Theorem~~
3. ~~Convergence Proof~~
4. Online Perceptron

Online Perceptron

- The Perceptron algorithm can easily be used even when the data set is an incoming stream
- Online prediction setting:
  - our data comes as a sequence $(\vec{x}_1, y_1), (\vec{x}_2, \vec{y}_2), \ldots$
  - for each time step, use the current $\vec{w}$ to predict label of $\vec{x}_i$
  - observe the true y_i and update $\vec{w}$ if needed



- **Theorem:** Suppose incoming data has radius R and is linearly separable with margin γ. Then, the number of mistakes is $\leq R^2/\gamma^2$

Proof Outline

- Goal: Prove that the number of mistakes is at most R^2/γ^2
  - Key insight: number of mistakes $=$ number of updates
  - Therefore, t (from previous proof) $=$ number of mistakes
- Step 1: $\vec{w}^\star \cdot \vec{w}^{(t)} \geq t\gamma$
- Step 2: $\|\vec{w}^{(t)}\|^2 \leq tR^2$
- Step 3: If $t > R^2/\gamma^2$, $\cos\left(\theta_{\vec{w}^\star, \vec{w}^{(t)}}\right) > \frac{\sqrt{R^2/\gamma^2}\gamma}{R} = 1$
  - Proof by contradiction

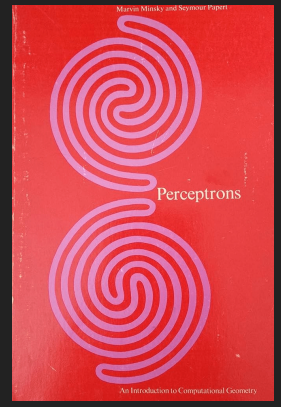
The Perceptron Algorithm

- **Initialize** $\vec{w}^{(0)} = (0, ..., 0)$ and $t = 0$
- **For** data $(\vec{x}_1, y_1), (\vec{x}_2, y_2), ...$
  - **If** $y_i(\vec{w}^{(t)} \cdot \vec{x}_i) \leq 0$ **then:**
    - $\vec{w}^{(t+1)} = \vec{w}^{(t)} + y_i \vec{x}_i$
    - $t \leftarrow t + 1$
- **End For**
- **Output** $\vec{w}^{(t)}$

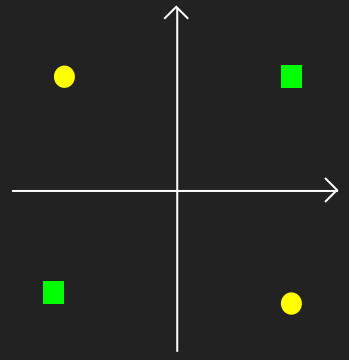
The Perceptron Algorithm

- **Initialize** $\vec{w}^{(0)} = (0, ..., 0)$ and $t = 0$
- **For** data $(\vec{x}_1, y_1), (\vec{x}_2, y_2), ...$
  - **If** $y_i(\vec{w}^{(t)} \cdot \vec{x}_i) \leq 0$ **then:**
    - $\vec{w}^{(t+1)} = \vec{w}^{(t)} + y_i \vec{x}_i$
    - $t \leftarrow t + 1$
- **End For**
- **Output** $\vec{w}^{(t)}$

Minsky & Papert



Perceptrons, 1969



Despite the early hype, perceptrons fell out of favor by the end of the 1960s

- limitations of linear classifiers
- no good algorithm for multiple layers

Next time: new algorithms for linear classifiers