CSC 561: Neural Networks and Deep Learning

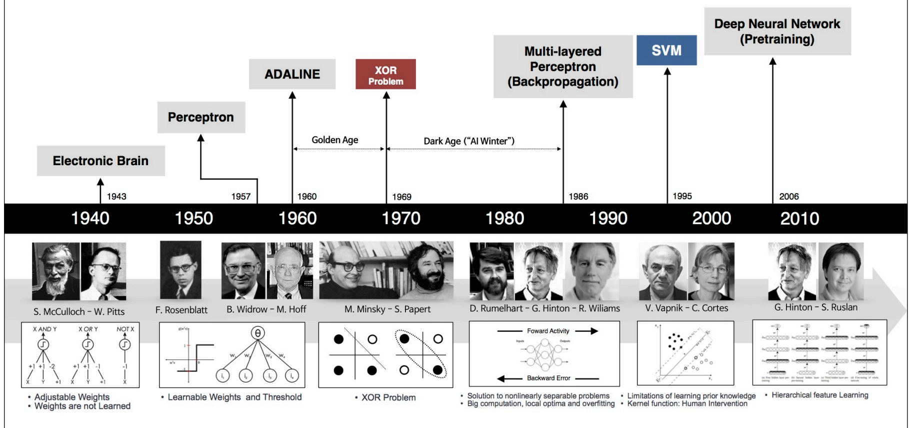
Perceptron

Marco Alvarez

Department of Computer Science and Statistics
University of Rhode Island

Spring 2025

THINK BIG WE DO℠


Deep Neural Network (Pretraining)
SVM
Multi-layered Perceptron (Backpropagation)
XOR Problem
ADALINE
Perceptron
Electronic Brain
Golden Age
Dark Age ("AI Winter")
1943
1957
1960
1969
1986
1995
2006
1940
1950
1960
1970
1980
1990
2000
2010
S. McCulloch – W. Pitts
F. Rosenblatt
B. Widrow – M. Hoff
M. Minsky – S. Papert
D. Rumelhart – G. Hinton – R. Wiliams
V. Vapnik – C. Cortes
G. Hinton – S. Ruslan
X AND Y
X OR Y
NOT X
Foward Activity
Backward Error
• Adjustable Weights
• Weights are not Learned
• Learnable Weights and Threshold
• XOR Problem
• Solution to nonlinearly separable problems
• Big computation, local optima and overfitting
• Limitations of learning prior knowledge
• Kernel function: Human Intervention
• Hierarchical feature Learning

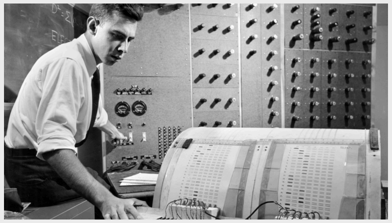
http://beamandrew.github.io/deeplearning/2017/02/23/deep_learning_101_part1.html

Rosenblatt (1958)

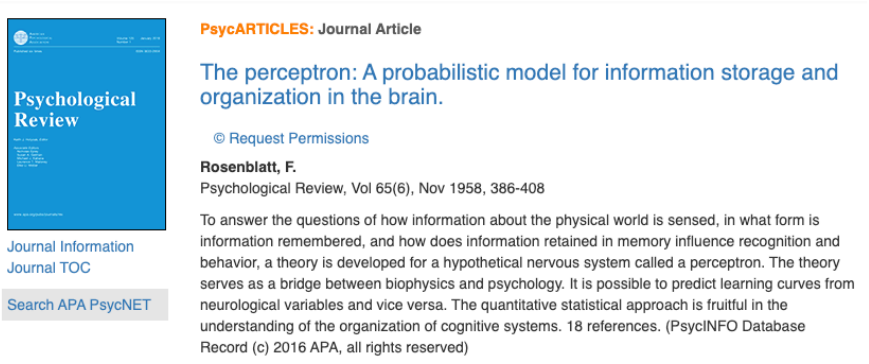
- Perceptron introduced by Frank Rosenblatt (psychologist, logician)
  - ✓ first trainable neural network
  - ✓ built upon McCulloch-Pitts artificial neurons
  - ✓ very powerful **learning** algorithm with high expectations

NEW NAVY DEVICE LEARNS BY DOING; Psychologist Shows Embryo of Computer Designed to Read and Grow Wiser

WASHINGTON, July 7, 1958 (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.



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


PsycARTICLES: Journal Article

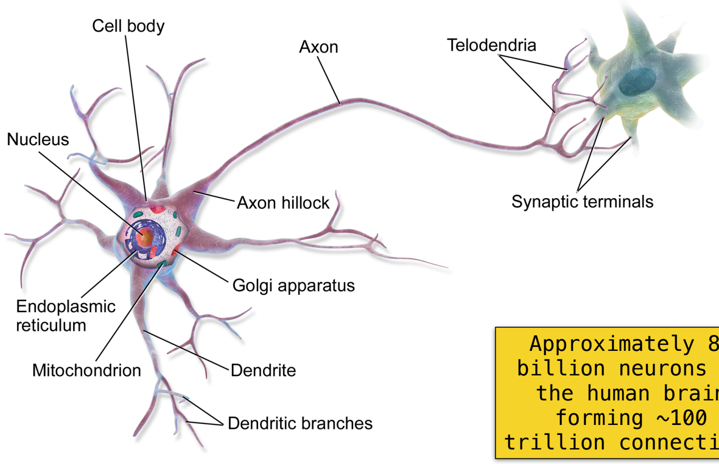
The perceptron: A probabilistic model for information storage and organization in the brain.

© Request Permissions

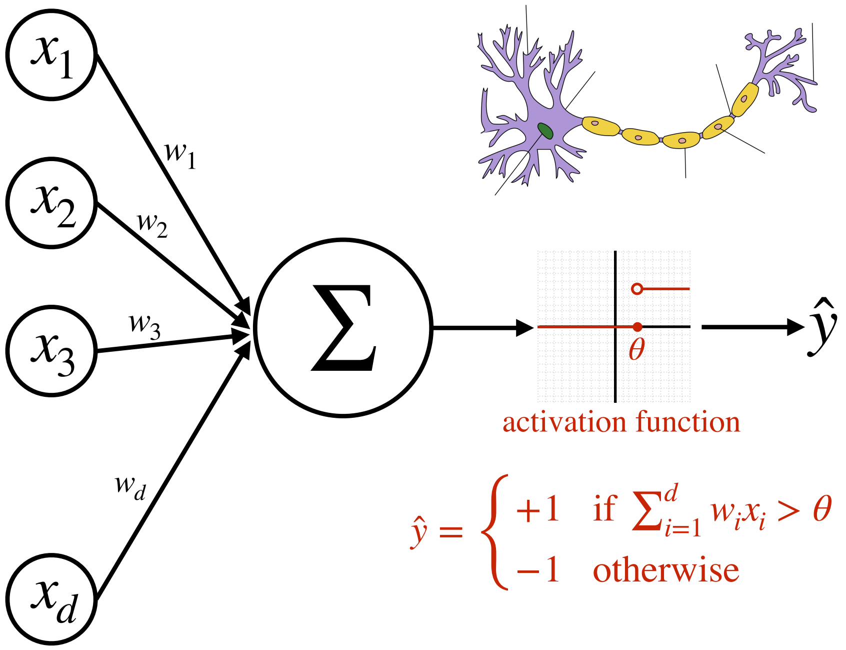
Rosenblatt, F.
Psychological Review, Vol 65(6), Nov 1958, 386-408

To answer the questions of how information about the physical world is sensed, in what form is information remembered, and how does information retained in memory influence recognition and behavior, a theory is developed for a hypothetical nervous system called a perceptron. The theory serves as a bridge between biophysics and psychology. It is possible to predict learning curves from neurological variables and vice versa. The quantitative statistical approach is fruitful in the understanding of the organization of cognitive systems. 18 references. (PsycINFO Database Record (c) 2016 APA, all rights reserved)

Neuron



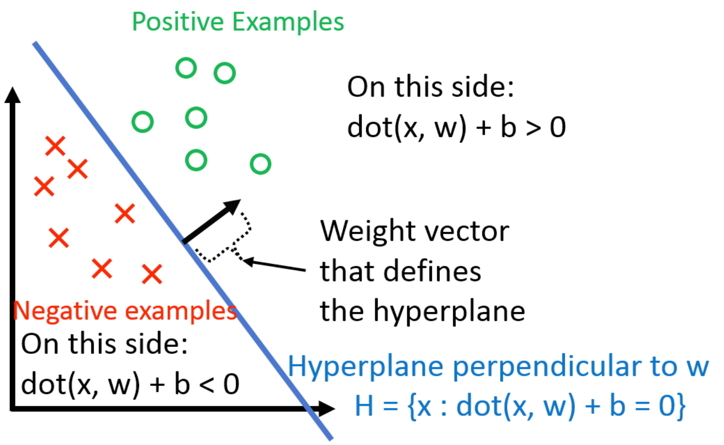
Approximately 86 billion neurons in the human brain forming ~100 trillion connections



$$\hat{y} = \begin{cases} +1 & \text{if } \sum_{i=1}^{d} w_i x_i > \theta \\ -1 & \text{otherwise} \end{cases}$$

Interpretation

b is the *bias* term $(-\theta)$



Positive Examples

On this side:
dot(x, w) + b > 0

Weight vector
that defines
the hyperplane

Negative examples
On this side:
dot(x, w) + b < 0

Hyperplane perpendicular to w
H = {x : dot(x, w) + b = 0}

https://www.cs.cornell.edu/courses/cs4780/2024sp/lectures/lecturenote03.html

Absorbing the threshold/bias

$$\hat{y} = \begin{cases} +1 & \text{if } \sum_{i=1}^{d} w_i x_i > \theta \\ -1 & \text{otherwise} \end{cases}$$

$$\Big\downarrow \quad x_0 = +1, \quad w_0 = -\theta$$

For convenience we can 'absorb' the threshold into the weight vector by adding an extra dimension

$$\hat{y} = \begin{cases} +1 & \text{if } \sum_{i=0}^{d} w_i x_i > 0 \\ -1 & \text{otherwise} \end{cases}$$

Summary

$$h_{\mathbf{w}}(\mathbf{x}) = \sigma\left(\sum_{i=0}^{d} w_i x_i\right) = \sigma(\mathbf{w}^T\mathbf{x})$$

$$\sigma(z) = \begin{cases} +1 & \text{if } z > 0 \\ -1 & \text{if } z \leq 0 \end{cases}$$

$$\mathscr{H} = \{h_w : \mathbf{w} \in \mathbb{R}^{d+1}\}$$

set of all functions
$h_{\mathbf{w}} : \mathbb{R}^{d+1} \mapsto \{-1, +1\}$
defined by **w**

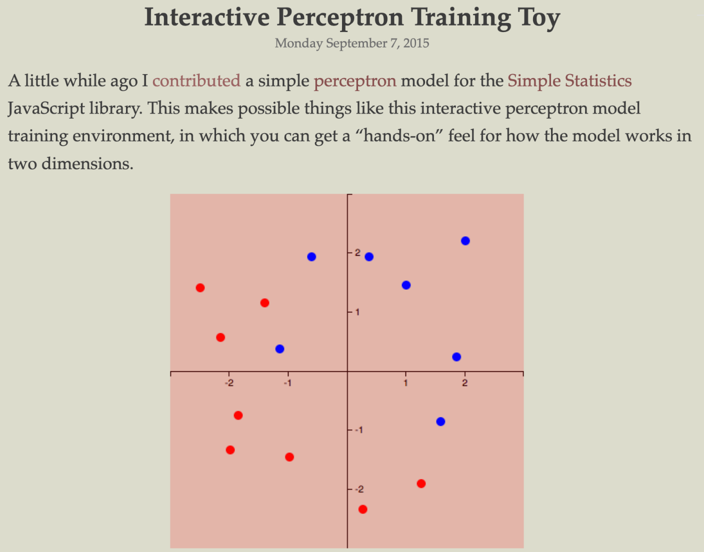
The algorithm

- Start with a null vector **w**
- Repeat for **T** epochs
  - shuffle the data instances
  - for all examples **($\mathbf{x_i}$, $\mathbf{y_i}$)** in training data
    - if $\mathbf{x_i}$ misclassified
      - if $\mathbf{y_i}$ equals +1
        - update **w** by adding $\mathbf{x_i}$ to **w**
      - else
        - update **w** by subtracting $\mathbf{x_i}$ from **w**
- Return **w**

Practice

- Write the pseudocode or python for the perceptron algorithm?

Demo



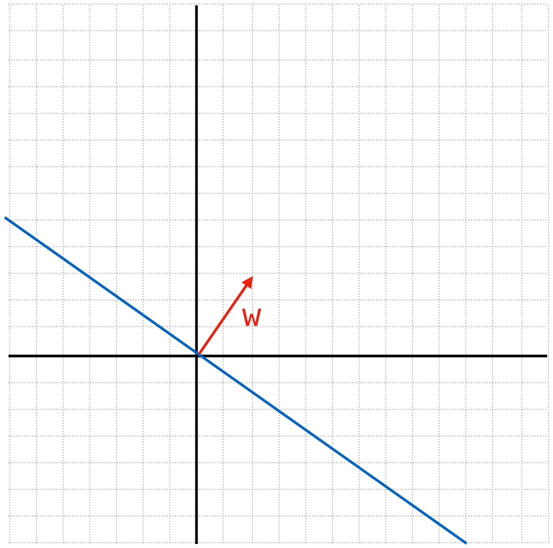
Interactive Perceptron Training Toy

Monday September 7, 2015

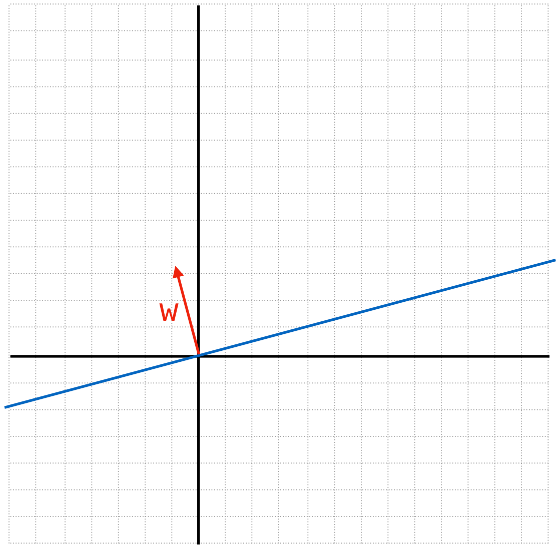
A little while ago I contributed a simple perceptron model for the Simple Statistics JavaScript library. This makes possible things like this interactive perceptron model training environment, in which you can get a "hands-on" feel for how the model works in two dimensions.

https://planspace.org/20150907-interactive_perceptron_training_toy/

Mistake on a positive (update)



Mistake on a negative (update)

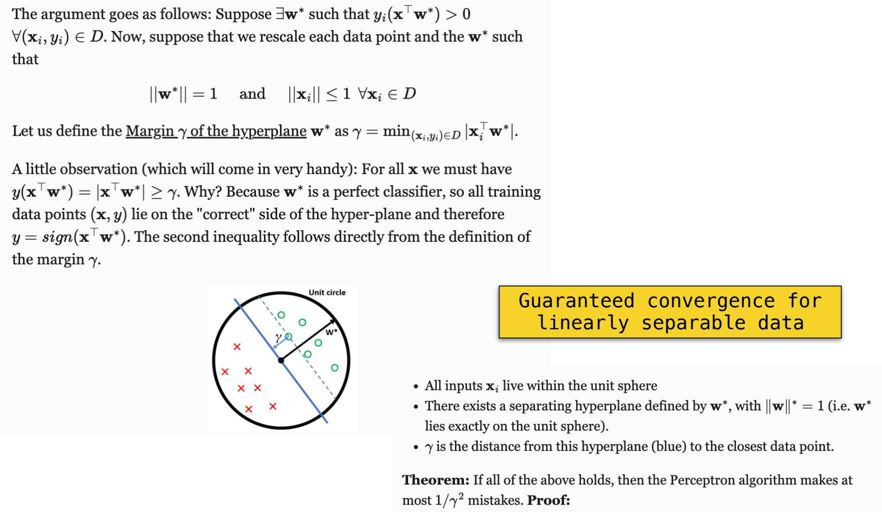


Intuition

- Suppose a mistake on the positive side:
  - ✓ $y = +1 \qquad \mathbf{w}^T x \leq 0$
- After 1 update the new weight vector will be:
  - ✓ $\mathbf{w}_{t+1} = \mathbf{w}_t + \mathbf{x}$
- Classifying the same datapoint with the new weight vector:
  - ✓ $\mathbf{w}_{t+1}{}^T\mathbf{x} = (\mathbf{w}_t + \mathbf{x})^T\mathbf{x} = \mathbf{w}_t^T\mathbf{x} + \mathbf{x}^T\mathbf{x} \geq \mathbf{w}_t^T\mathbf{x}$

can use the same idea for mistakes on the negative side

Convergence theorem



The argument goes as follows: Suppose $\exists \mathbf{w}^*$ such that $y_i(\mathbf{x}^\top \mathbf{w}^*) > 0$ $\forall (\mathbf{x}_i, y_i) \in D$. Now, suppose that we rescale each data point and the $\mathbf{w}^*$ such that

$$||\mathbf{w}^*|| = 1 \quad \text{and} \quad ||\mathbf{x}_i|| \leq 1 \ \forall \mathbf{x}_i \in D$$

Let us define the Margin γ of the hyperplane $\mathbf{w}^*$ as $\gamma = \min_{(\mathbf{x}_i, y_i) \in D} |\mathbf{x}_i^\top \mathbf{w}^*|$.

A little observation (which will come in very handy): For all $\mathbf{x}$ we must have $y(\mathbf{x}^\top \mathbf{w}^*) = |\mathbf{x}^\top \mathbf{w}^*| \geq \gamma$. Why? Because $\mathbf{w}^*$ is a perfect classifier, so all training data points $(\mathbf{x}, y)$ lie on the "correct" side of the hyper-plane and therefore $y = sign(\mathbf{x}^\top \mathbf{w}^*)$. The second inequality follows directly from the definition of the margin γ.

Guaranteed convergence for linearly separable data

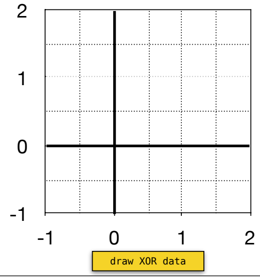
- All inputs $\mathbf{x}_i$ live within the unit sphere
- There exists a separating hyperplane defined by $\mathbf{w}^*$, with $\|\mathbf{w}\|^* = 1$ (i.e. $\mathbf{w}^*$ lies exactly on the unit sphere).
- γ is the distance from this hyperplane (blue) to the closest data point.

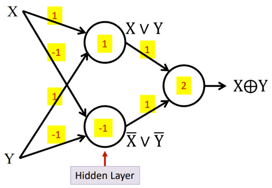
Theorem: If all of the above holds, then the Perceptron algorithm makes at most $1/\gamma^2$ mistakes. **Proof:**

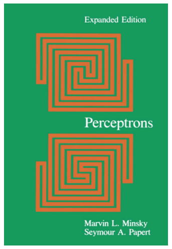
https://www.cs.cornell.edu/courses/cs4780/2024sp/lectures/lecturenote03.html

Minsky & Papert (1969)

- Limitations
  - ✓ cannot learn XOR function
  - ✓ limited to linear decision boundaries
  - ✓ led to first AI winter — significant impact, shifting focus on symbolic approaches







Remarks

- Assumes data is **linearly separable**
  - ✓ does not converge if classes are not linearly separable
- Different **correct** solutions can be found
  - ✓ most are not optimal in terms of **generalization**
- Modern extensions
  - ✓ averaged perceptron — returns a **weighted average** of earlier hypotheses
  - ✓ kernel perceptron — overcome XOR limitation
  - ✓ multilayer perceptrons