Perceptrons and XOR

Austin Smith

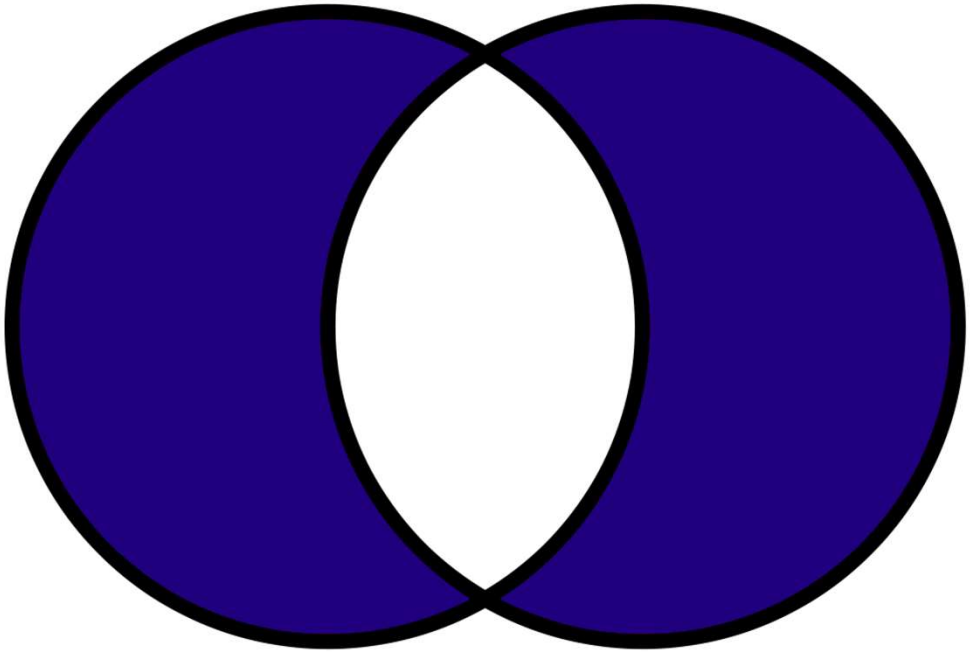
2/17/22

Overview

- Terms and review
- History
- Example

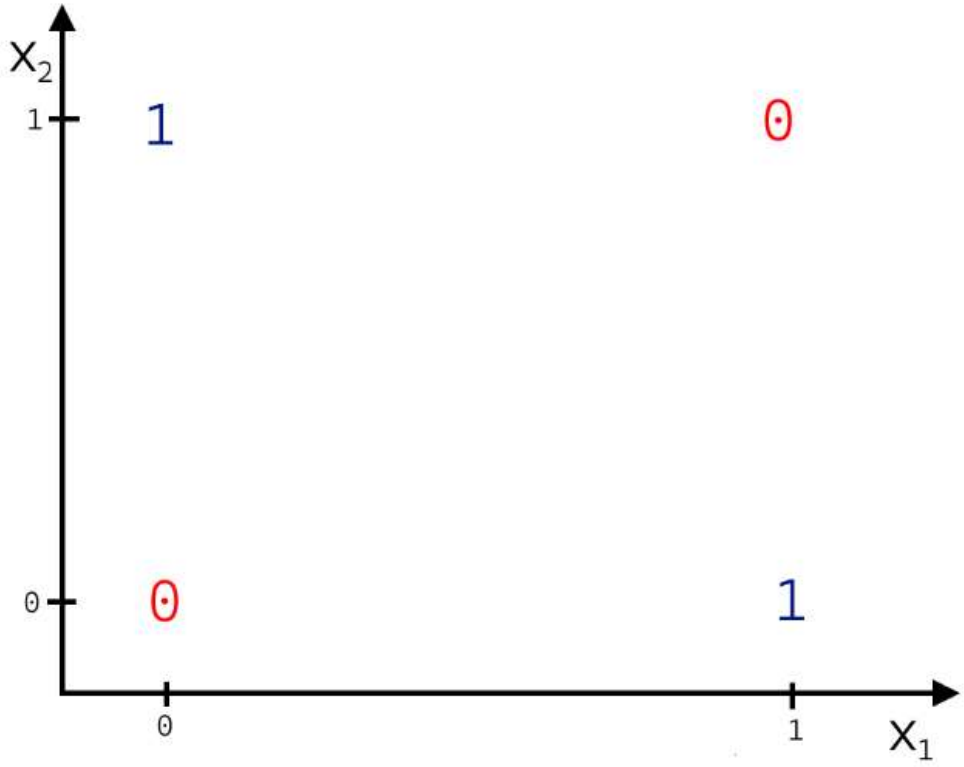
What is XOR?

- eXclusive OR



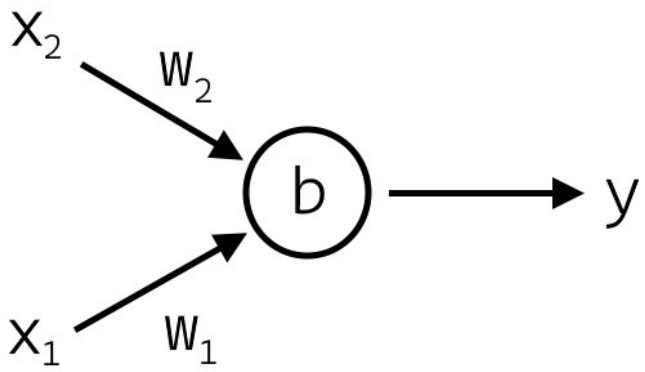
What is XOR?

x_1	x_2	XOR(x_1,x_2)
0	0	0
0	1	1
1	0	1
1	1	0



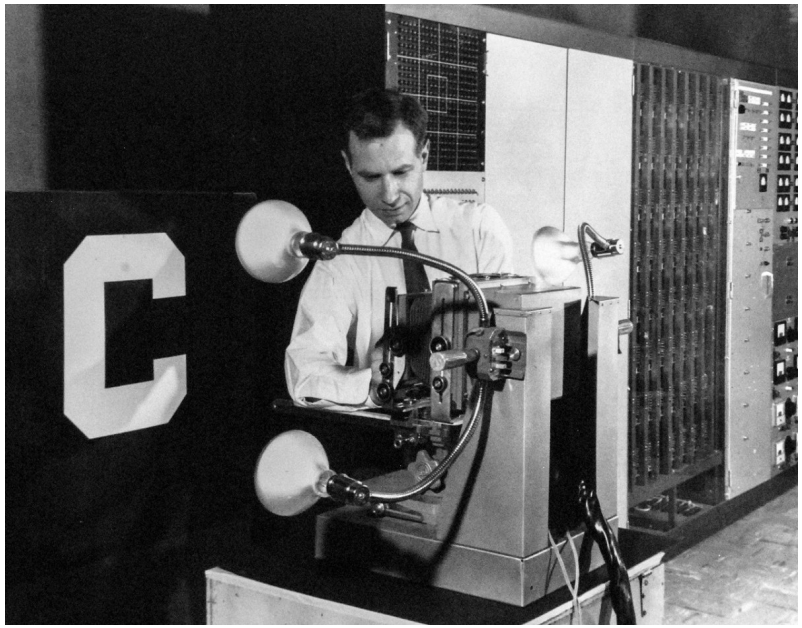
What is a perceptron?

- A perceptron is a single artificial neuron
- Multilayered perceptron is a feed-forward neural net
- x_i inputs, w_i weights, b_i biases, y output



History

- Frank Rosenblatt (1928–1971)
  - Psychologist and Neurobiologist
  - Developed the Perceptron (1957)
- *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms* (1962)
  - Trial and error learning devices work!
  - Unlimited hidden layers can solve any classification problem



History

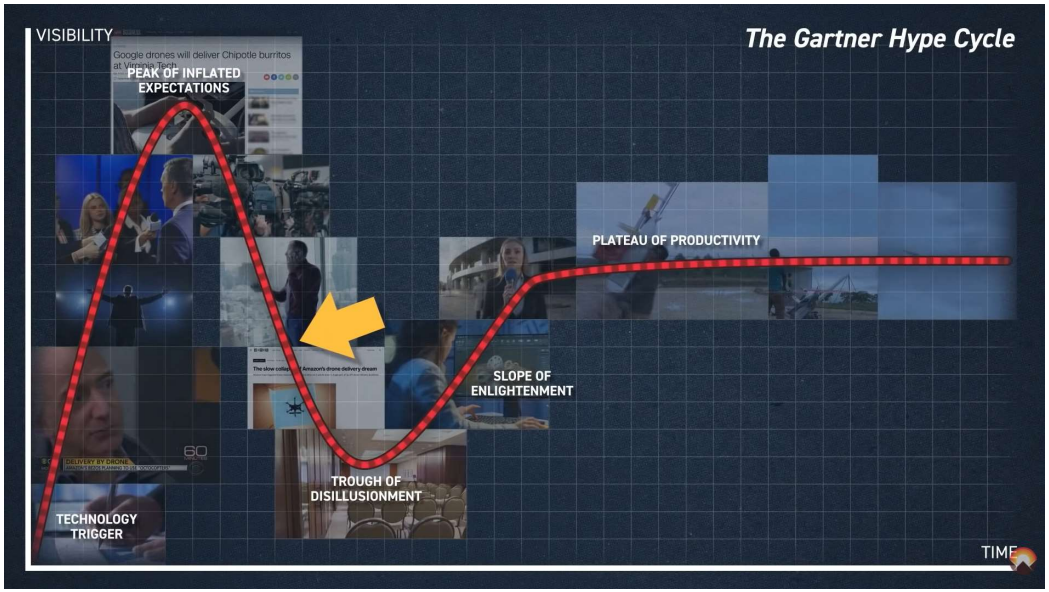
- The Mark 1 Perceptron
  - 400 input nodes
  - 512 layer 2 nodes
  - 8 output nodes
- Marvin Minsky (1927–2016)
  - Mathematician
- *Perceptrons: An Introduction to Computational Geometry (1969)*

Perceptrons: An Introduction to Computational Geometry (1969)

- Presented a pessimistic view on contemporary research
- Looked only at limited networks
- Local connected neurons
  - Physical hardware construction
  - Poor computing power
- Results of these restrictions prevent classifications including XOR

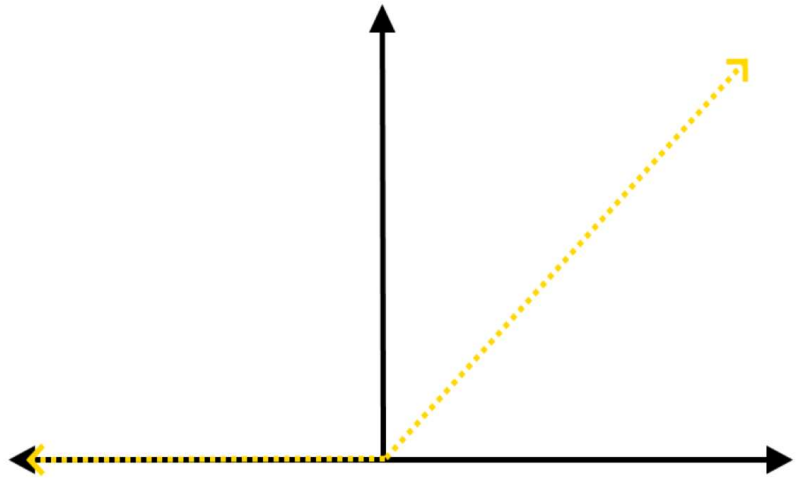
First Winter

- 1973–1980
- Possibly a result of *Perceptrons*
  - Non-linear functions can not be modeled by linear functions
  - Combinatorial explosion
- Other factors:
  - Funding cuts
  - Trough of disillusionment



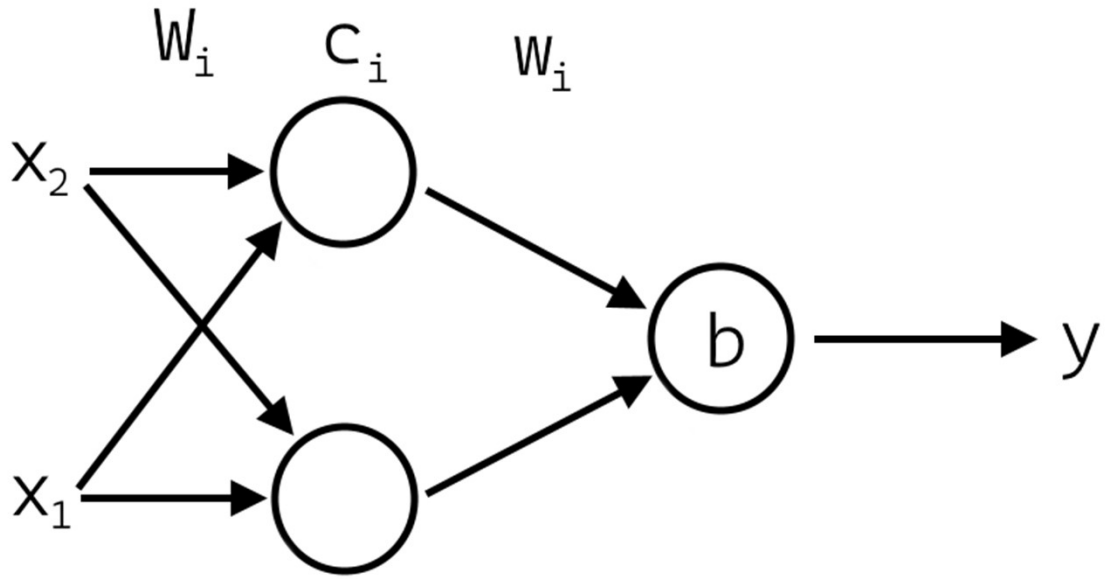
Quick History of ReLU

- Unnamed in 1980
  - Fukushima
- 1986 threshold function
  - Hinton
- 2000 ReLU
  - Hahnloser
  - Biological motivations
  - mathematical justifications
- 2010 ReLU
  - Nair and Hinton
  - Popularized the usage of ReLU



How to do XOR

- Add another layer, with non-local connections!
- Hidden layer with weights W_i and biases c_i



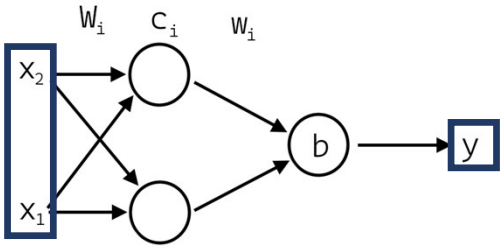
Example

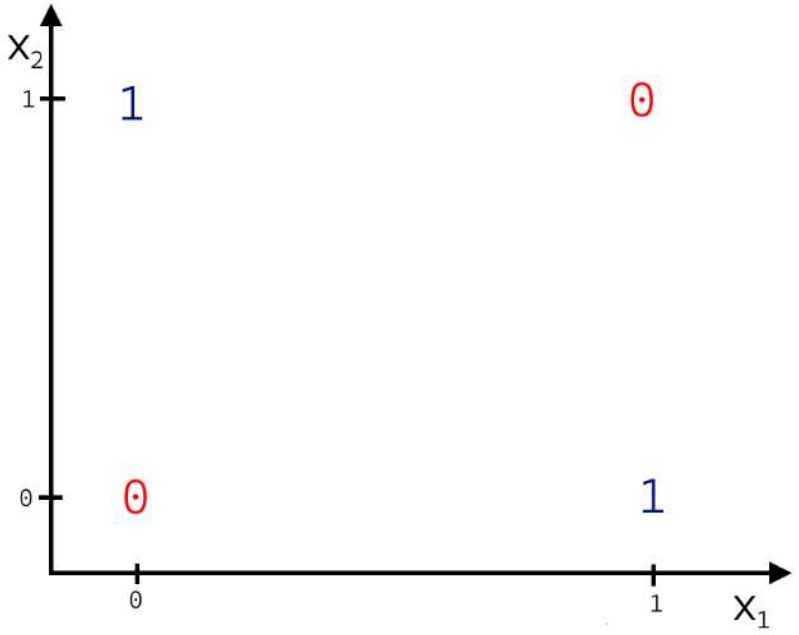
- Input:

$$\boldsymbol{X} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$$

- Goal:

$$\boldsymbol{y} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

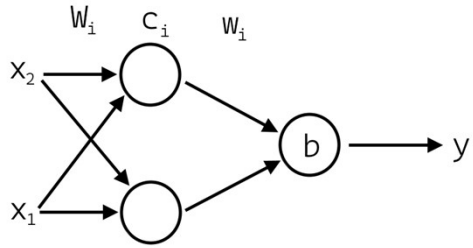




Example

- "Known" weights and Biases:

$$\boldsymbol{w} = \begin{bmatrix} 1 \\ -2 \end{bmatrix} \quad \boldsymbol{c} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \quad \boldsymbol{W} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad b = 0$$



- Function defining the network:

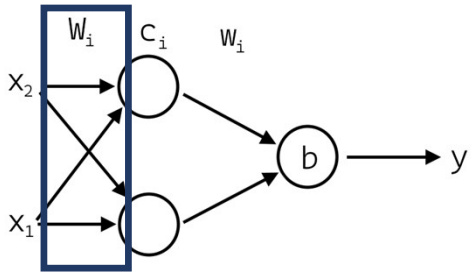
$$y = f(x; \boldsymbol{W}, \boldsymbol{c}, \boldsymbol{w}, b) = \boldsymbol{w}^{\top} \max\{0, \boldsymbol{W}^{\top}\boldsymbol{x} + \boldsymbol{c}\} + b$$

- ReLU activation function comes in the form of the $\max\{\}$

Example

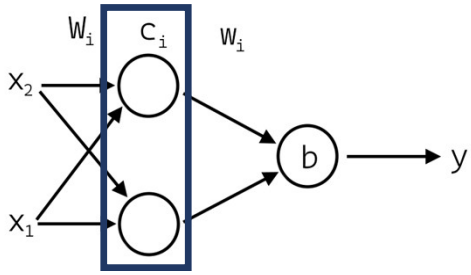
$$\boldsymbol{XW} = \begin{bmatrix} 0 & 0 \\ 1 & 1 \\ 1 & 1 \\ 2 & 2 \end{bmatrix}$$

Apply weights



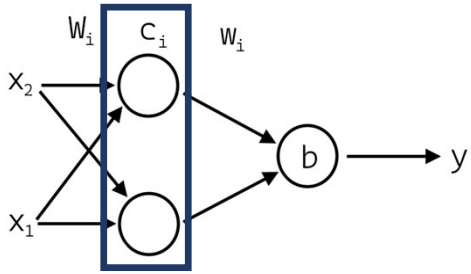
$$\boldsymbol{XW} + \boldsymbol{c} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \\ 1 & 0 \\ 2 & 1 \end{bmatrix}$$

Add Biases



$$\max\{0, \boldsymbol{XW} + \boldsymbol{c}\} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 1 & 0 \\ 2 & 1 \end{bmatrix}$$

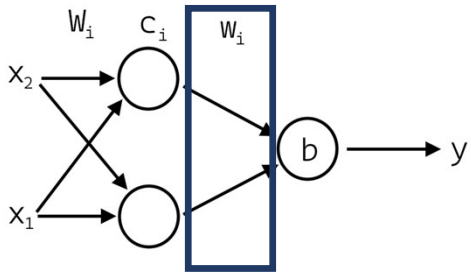
Apply ReLU



Example

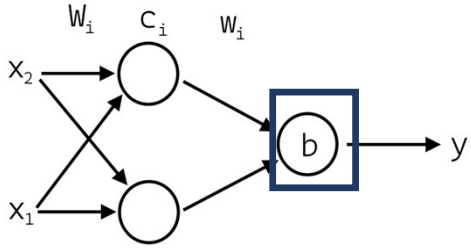
$$\boldsymbol{w}^{\top}\max\{0, \boldsymbol{XW} + \boldsymbol{c}\} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

Apply weights



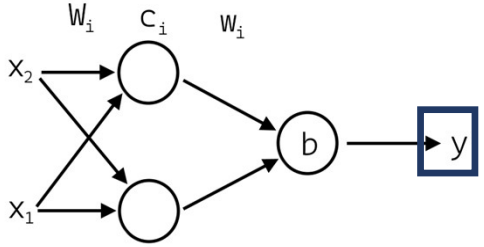
$$\boldsymbol{w}^{\top}\max\{0, \boldsymbol{XW} + \boldsymbol{c}\} + 0 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

Add Biases



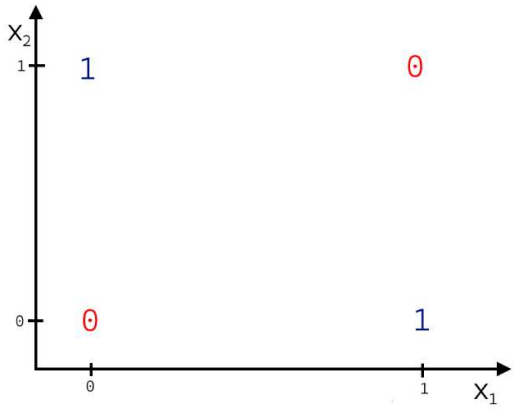
$$\boldsymbol{w}^{\top}\max\{0, \boldsymbol{W}^{\top}\boldsymbol{x} + \boldsymbol{c}\} + b = y = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

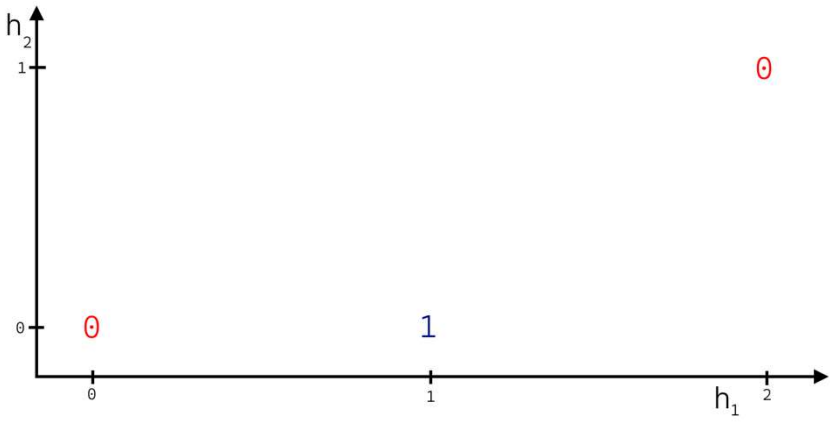
No need to apply an activation function, we have our solution!



Example

- In the hidden layer, the input space is transformed to a space that can be solved linearly





Sources:

- Slides on Frank Rosenblatt
- *Perceptrons: An Introduction to Computational Geometry*
- *Deep Learning*
  - Chapter 6.1

Further resources:

- Towards Data Science Python Example
  - Implemented example worked though here in python with training
- "A Sociological Study of the Official History of the Perceptrons Controversy"
  - Further discussion of how *Perceptrons* influenced the field of ML and AI
- Report detailing The Perceptron (1957)