CS 540 Introduction to Artificial Intelligence

Neural Networks (I): Perceptron

Yudong Chen
University of Wisconsin-Madison

Oct 19, 2021

Slides created by Sharon Li [modified by Yudong Chen]

Announcement

- HW6 released today, due Nov 4 (Thursday)
- Midterm: Oct 28 (Thursday)
  - Online. 90 min within 24 hours.
  - Cover NN III (next Tuesday's lecture).
  - Will post sample questions.

Today's outline

- HW5 Review
- Recap: Bayes and Naive Bayes Classifiers
- Single-layer Neural Network (Perceptron)

Part I: Bayes and Naïve Bayes (Recap)

Bayesian classifier

$$\hat{y} = \arg\max_{y} p(y \mid X_1, \ldots, X_k) \quad \text{(Posterior)}$$

(Prediction)

Bayesian classifier

$$\hat{y} = \arg\max_y p(y \mid X_1, \ldots, X_k) \quad \text{(Posterior)}$$

(Prediction)

$$= \arg\max_y \frac{p(X_1, \ldots, X_k \mid y) \cdot p(y)}{p(X_1, \ldots, X_k)} \quad \text{(by Bayes' rule)}$$

Independent of y

Bayesian classifier

$$\hat{y} = \arg\max_{y} p(y \mid X_1, \ldots, X_k)$$ (Posterior)

(Prediction)

$$= \arg\max_{y} \frac{p(X_1, \ldots, X_k \mid y) \cdot p(y)}{p(X_1, \ldots, X_k)}$$ (by Bayes' rule)

$$= \arg\max_{y} p(X_1, \ldots, X_k \mid y)\, p(y)$$

Class conditional likelihood

Class prior

Naïve Bayes Assumption

Conditional independence of feature attributes

$$p(X_1, \ldots, X_k \,|\, y)p(y) = \Pi_{i=1}^{k} p(X_i \,|\, y)p(y)$$

Easier to estimate

(using MLE!)

or Histogram / counting

Example 1: Play outside or not?

- If weather is sunny, would you likely to play outside?

Posterior probability p(Yes | ☀) vs. p(No | ☀)

- Weather = {Sunny, Rainy, Overcast}
- Play = {Yes, No}
- Observed data {Weather, play on day *m*}, m={1,2,…,N}

$$p(\text{Play} \mid \text{☀}) = \frac{p(\text{☀} \mid \text{Play})\; p(\text{Play})}{p(\text{☀})}$$

Bayes rule

Example 1: Play outside or not?

Step 1: Convert the data to a frequency table of Weather and Play

Step 2: Based on the frequency table, calculate **likelihoods** and **priors**

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

p(Play = Yes) = 0.64

p( | Yes) = 3/9 = 0.33

https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

Example 1: Play outside or not?

Step 3: Based on the likelihoods and priors, calculate posteriors

P(Yes| ☀)
=P(☀ |Yes) * P(Yes) / P(☀)
=0.33 * 0.64 / 0.36
=0.6

P(No| ☀)
=P(☀ |No) * P(No) / P(☀)
=0.4 * 0.36 / 0.36
=0.4

P(Yes| ☀) > P(No| ☀) go outside and play!

Quiz break

Q1-3: Consider the following dataset showing the result whether a person has passed or failed the exam based on various factors. Suppose the factors are independent to each other. We want to classify a new instance with Confident=Yes, Studied=Yes, and Sick=No.

Confident	Studied	Sick	Result
Yes	No	No	Fail
Yes	No	Yes	Pass
No	Yes	Yes	Fail
No	Yes	No	Pass
Yes	Yes	Yes	Pass

- A Pass
- B Fail

Quiz break

Q1-3: Consider the following dataset showing the result whether a person has passed or failed the exam based on various factors. Suppose the factors are independent to each other. We want to classify a new instance with Confident=Yes, Studied=Yes, and Sick=No.

Confident	Studied	Sick	Result
Yes	No	No	Fail
Yes	No	Yes	Pass
No	Yes	Yes	Fail
No	Yes	No	Pass
Yes	Yes	Yes	Pass

- A Pass
- B Fail

Q1-3: classify Confident=Yes, Studied=Yes, and Sick=No.

Confident (X_1)	Studied (X_2)	Sick (X_3)	Result (Y)
Yes	No	No	Fail
Yes	No	Yes	Pass
No	Yes	Yes	Fail
No	Yes	No	Pass
Yes	Yes	Yes	Pass

$$P(Y=1 \mid X_1=1, X_2=1, X_3=0)$$

$$= P(X_1=1 \mid Y=1)\, P(X_2=1 \mid Y=1)\, P(X_3=0 \mid Y=1)\, P(Y=1) / \cdots$$

$$= \frac{2}{3} \cdot \frac{2}{3} \cdot \frac{1}{3} \cdot \frac{3}{5} / \cdots$$

$$P(Y=0 \mid X_1=1, X_2=1, X_3=0)$$

$$= \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{2}{5} / \cdots$$

predict "pass"

Q1-3: classify Confident=Yes, Studied=Yes, and Sick=No.

Confident	Studied	Sick	Result
Yes	No	No	Fail
Yes	No	Yes	Pass
No	Yes	Yes	Fail
No	Yes	No	Pass
Yes	Yes	Yes	Pass

$$P(Y = \text{pass} \mid X_1 = 1, X_2 = 1, X_3 = 0)$$

$$\propto P(X_1 = 1 \mid Y = \text{pass}) \cdot P(X_2 = 1 \mid Y = \text{pass}) \cdot P(X_3 = 0 \mid Y = \text{pass}) \cdot P(Y = \text{pass})$$

$$= \frac{2}{3} \cdot \frac{2}{3} \cdot \frac{2}{3} \cdot \frac{3}{5}$$

$$P(Y = \text{fail} \mid X_1 = 1, X_2 = 1, X_3 = 0)$$

$$\propto P(X_1 = 1 \mid Y = \text{fail}) \cdot P(X_2 = 1 \mid Y = \text{fail}) \cdot P(X_3 = 0 \mid Y = \text{fail}) \cdot P(Y = \text{fail})$$

$$= \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{2}{5}$$

Part I: Single-layer Neural Network

How to classify

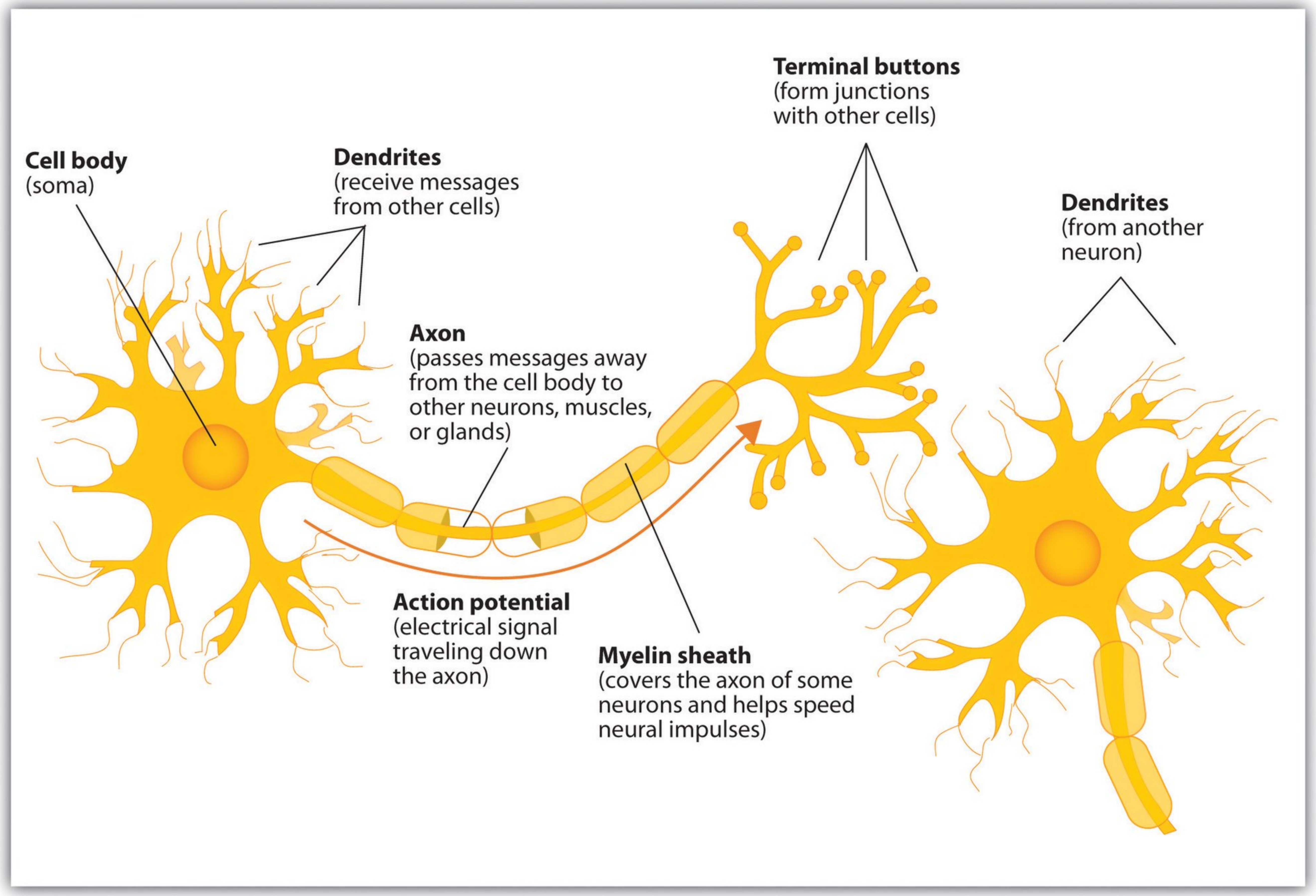
Cats vs. dogs?

Inspiration from neuroscience

- Inspirations from human brains
- Networks of simple and homogenous units

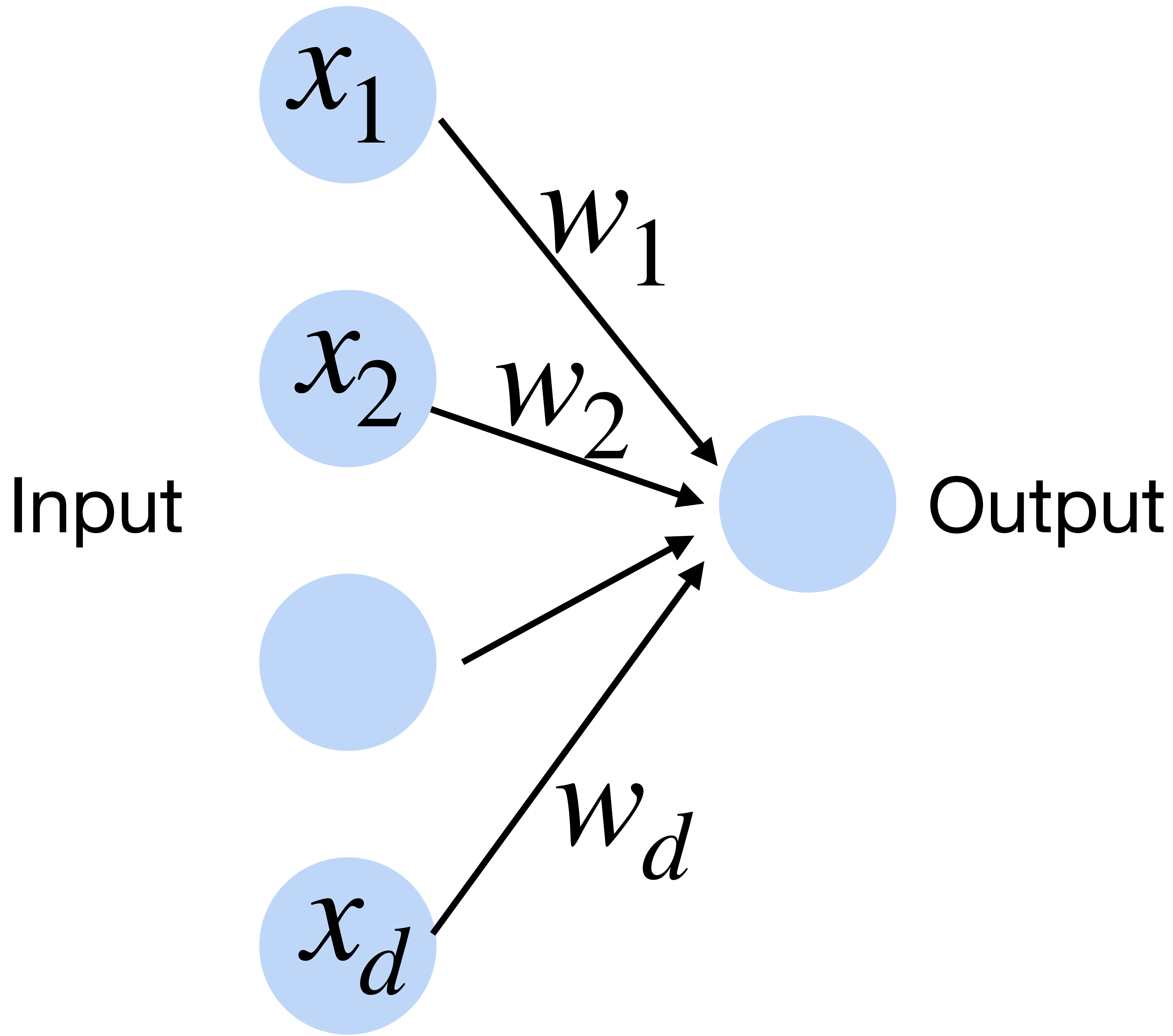
many.

(wikipedia)



Perceptron

Cats vs. dogs?



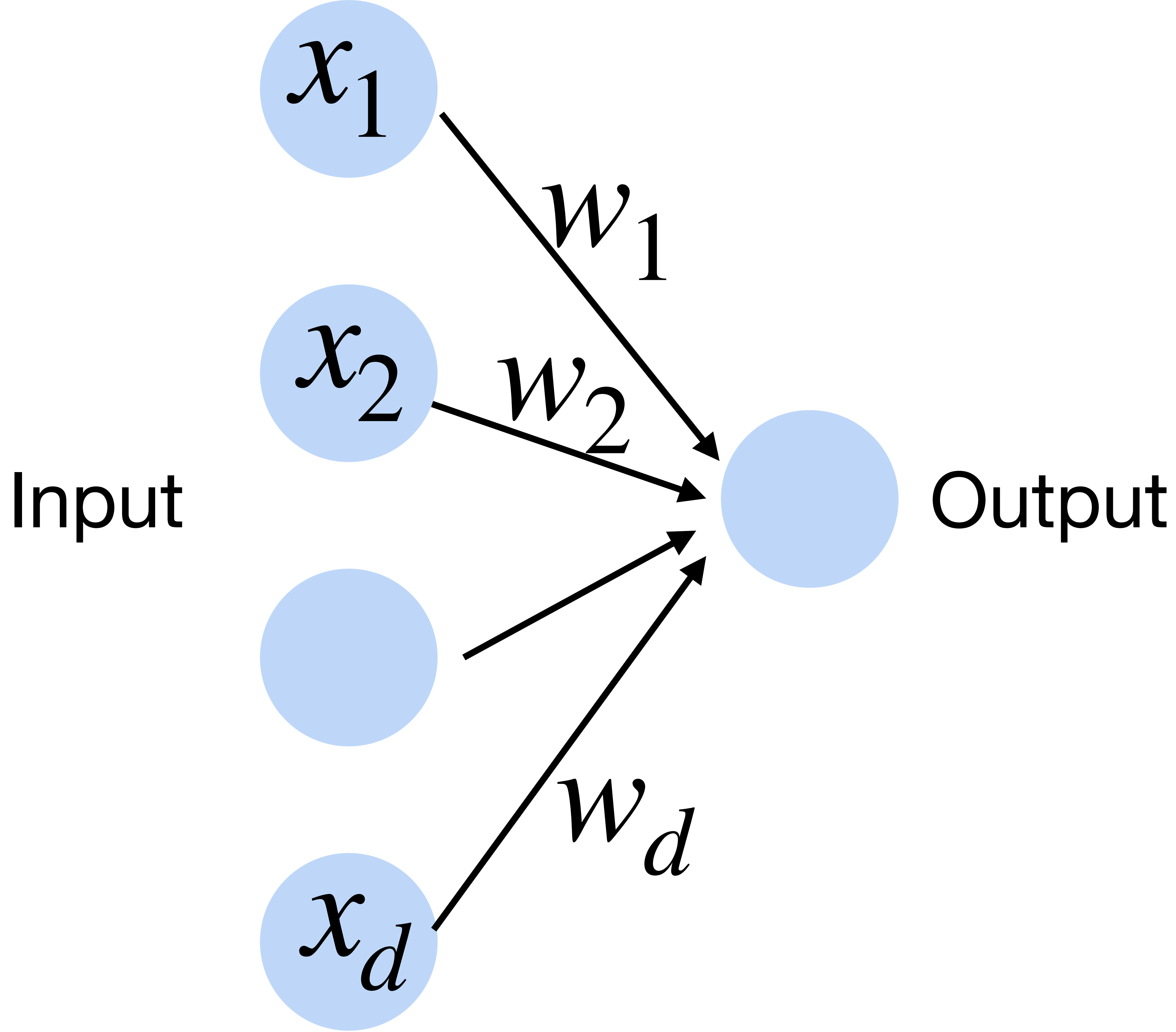
Linear Perceptron (=linear regression)

- Given input $\mathbf{x}$, weight $\mathbf{w}$ and bias b, perceptron outputs:

$$f_{\mathbf{w},b}(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b$$

$$= x_1 w_1 + x_2 w_2 + \cdots + x_d w_d + b.$$

Cats vs. dogs?



Perceptron

- Given input $\mathbf{x}$, weight $\mathbf{w}$ and bias b, perceptron outputs:

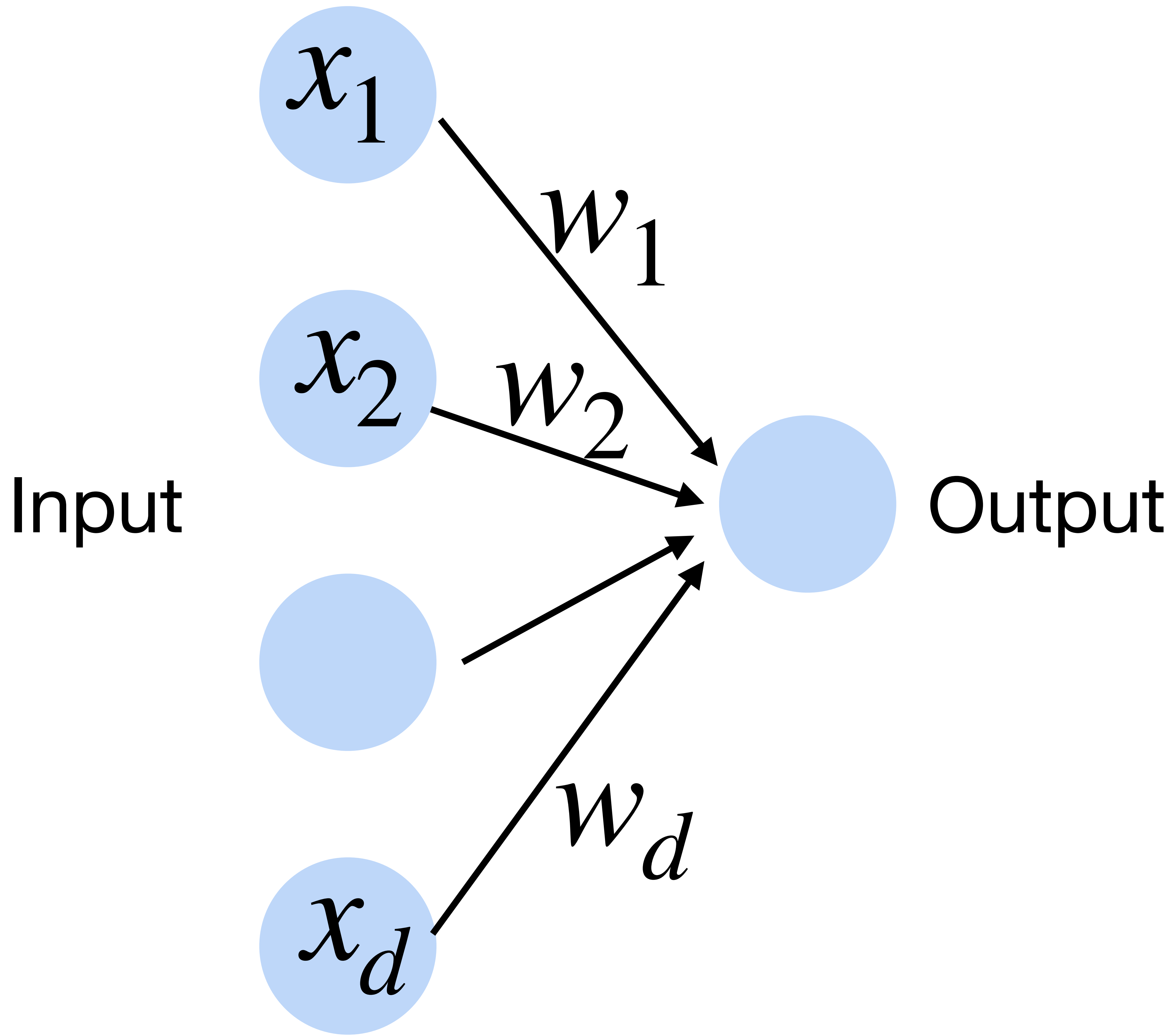
$$f_{\mathbf{w},b}(\mathbf{x}) = \sigma\left(\langle \mathbf{w}, \mathbf{x} \rangle + b\right)$$

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

Activation function

$$= \begin{cases} 1 & \langle w, x \rangle + b > 0 \\ 0 & \text{o.w.} \end{cases}$$

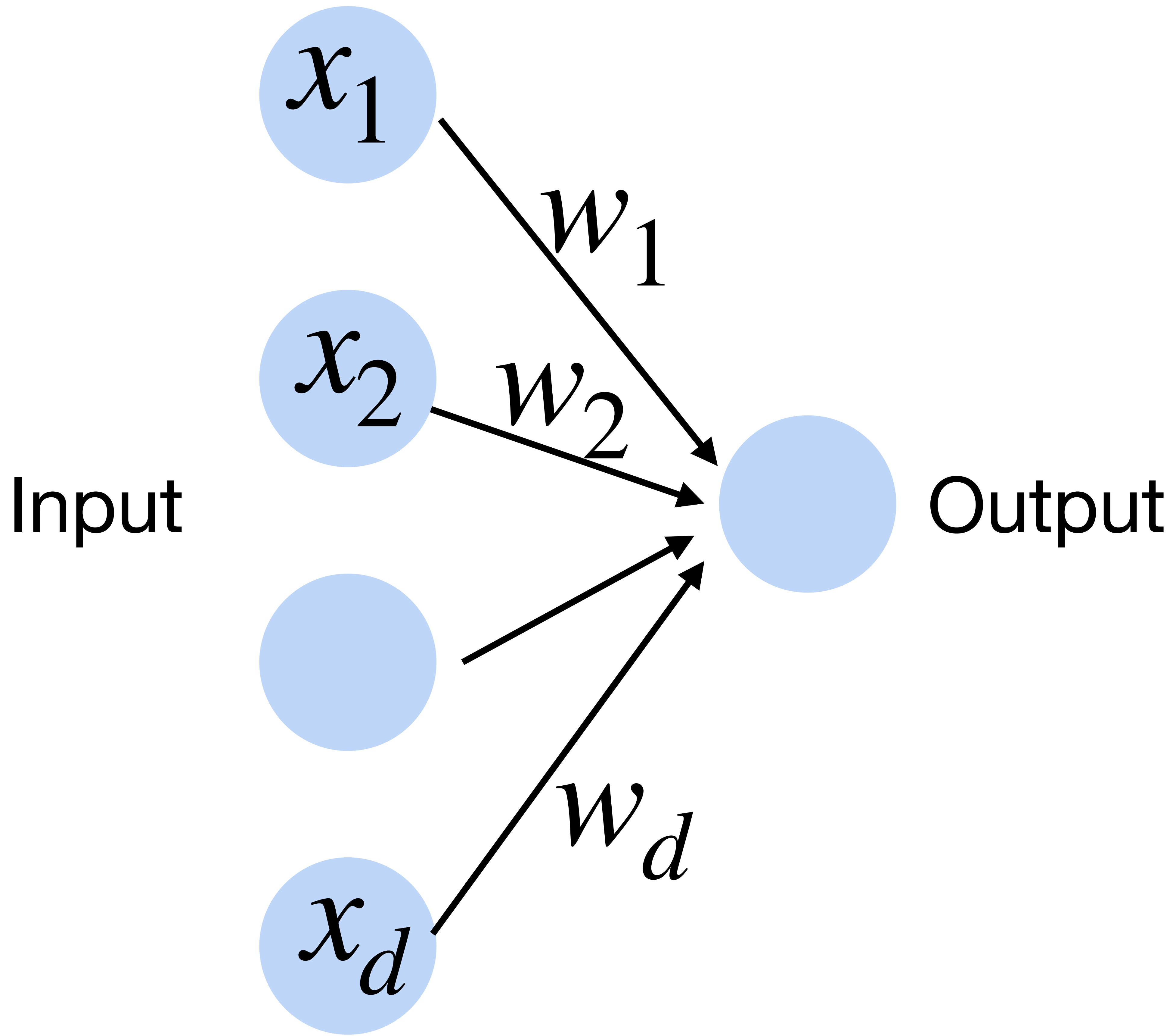
Cats vs. dogs?



Perceptron

- Goal: learn parameters $\mathbf{w} = \{w_1, w_2, \ldots, w_d\}$ and b to minimize the classification error

Cats vs. dogs?



Training the Perceptron

$y_i = 1$, $w^T x_i < 0$
$y_i = -1$, $w^T x_i > 0$
} classification error for i-th point.

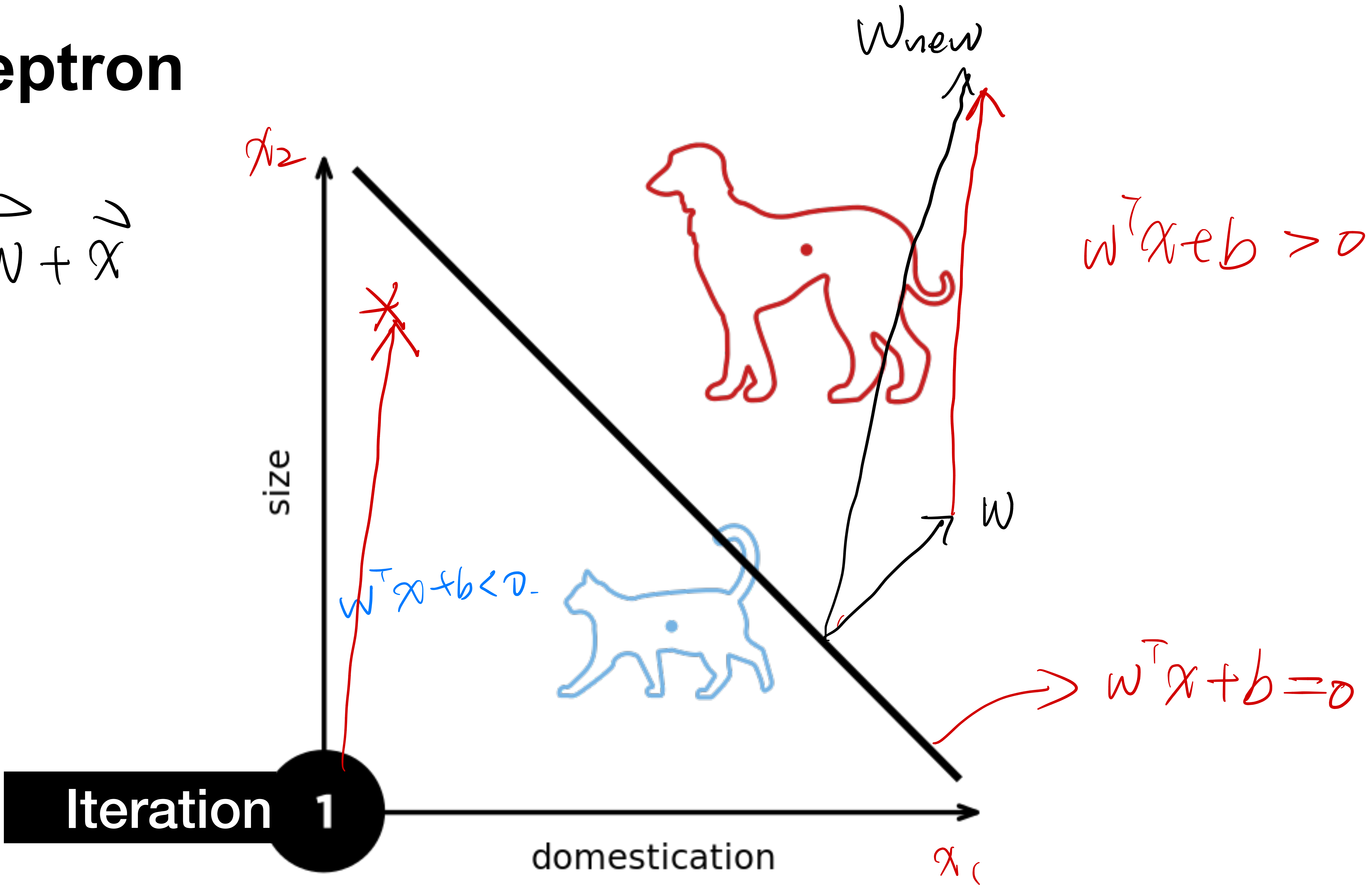
Perceptron Algorithm

Initialize $\vec{w} = \vec{0}$	// Initialize $\vec{w}$. $\vec{w} = \vec{0}$ misclassifies everything.
while TRUE **do**	// Keep looping
$m = 0$	// Count the number of misclassifications, m
for $(x_i, y_i) \in D$ **do**	// Loop over each (data, label) pair in the dataset, D
if $y_i(\vec{w}^T \cdot \vec{x}_i) \leq 0$ **then**	// If the pair $(\vec{x}_i, y_i)$ is misclassified
$\vec{w} \leftarrow \vec{w} + y\vec{x}$	// Update the weight vector $\vec{w}$
$m \leftarrow m + 1$	// Counter the number of misclassification
end if	
end for	
if $m = 0$ **then**	// If the most recent $\vec{w}$ gave 0 misclassifications
break	// Break out of the while-loop
end if	
end while	// Otherwise, keep looping!

$w \leftarrow w + \vec{x}_i$ if $y = 1$
$w \leftarrow w - \vec{x}_i$ if $y = -1$

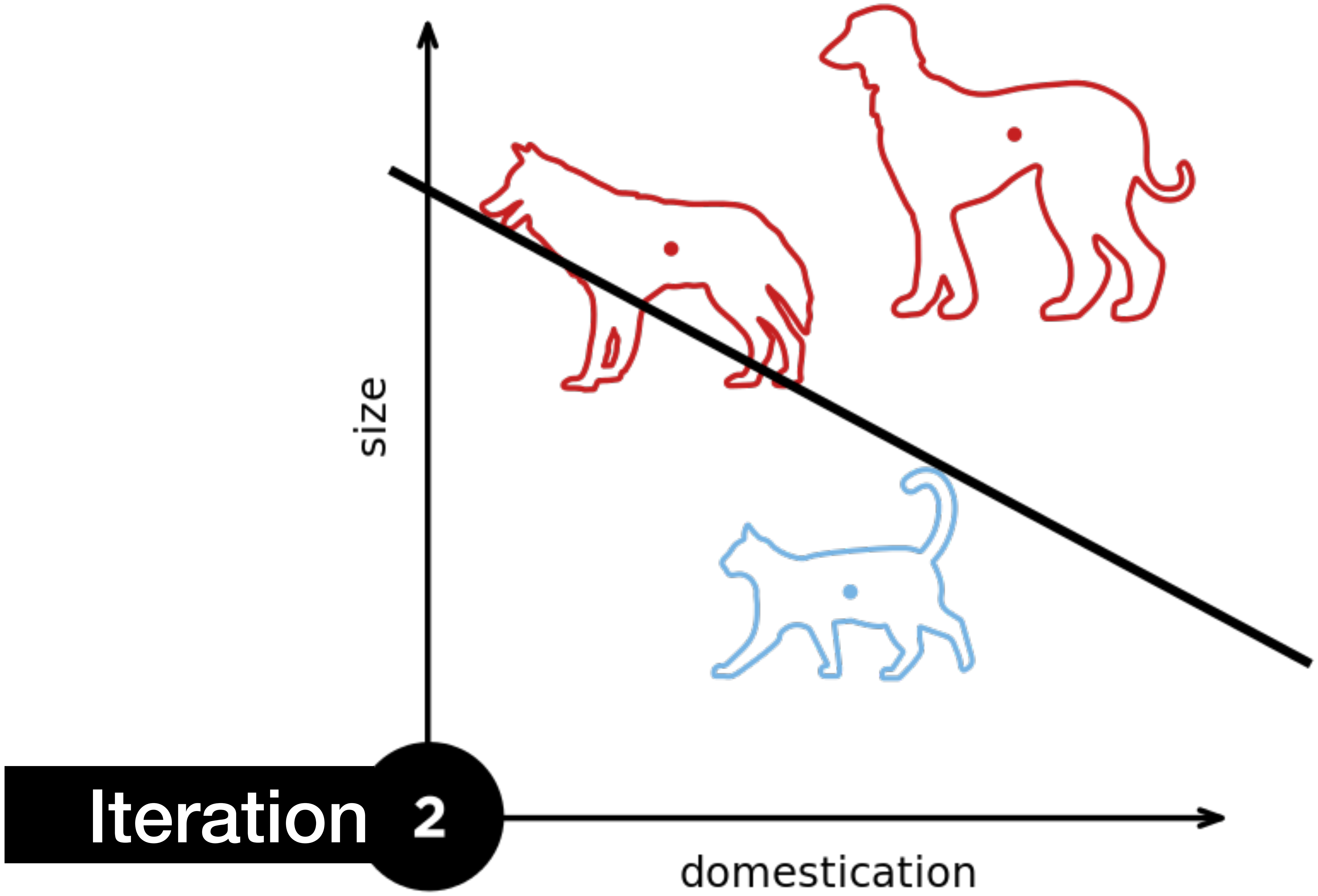
Perceptron

$\vec{w} \leftarrow \vec{w} + \vec{x}$



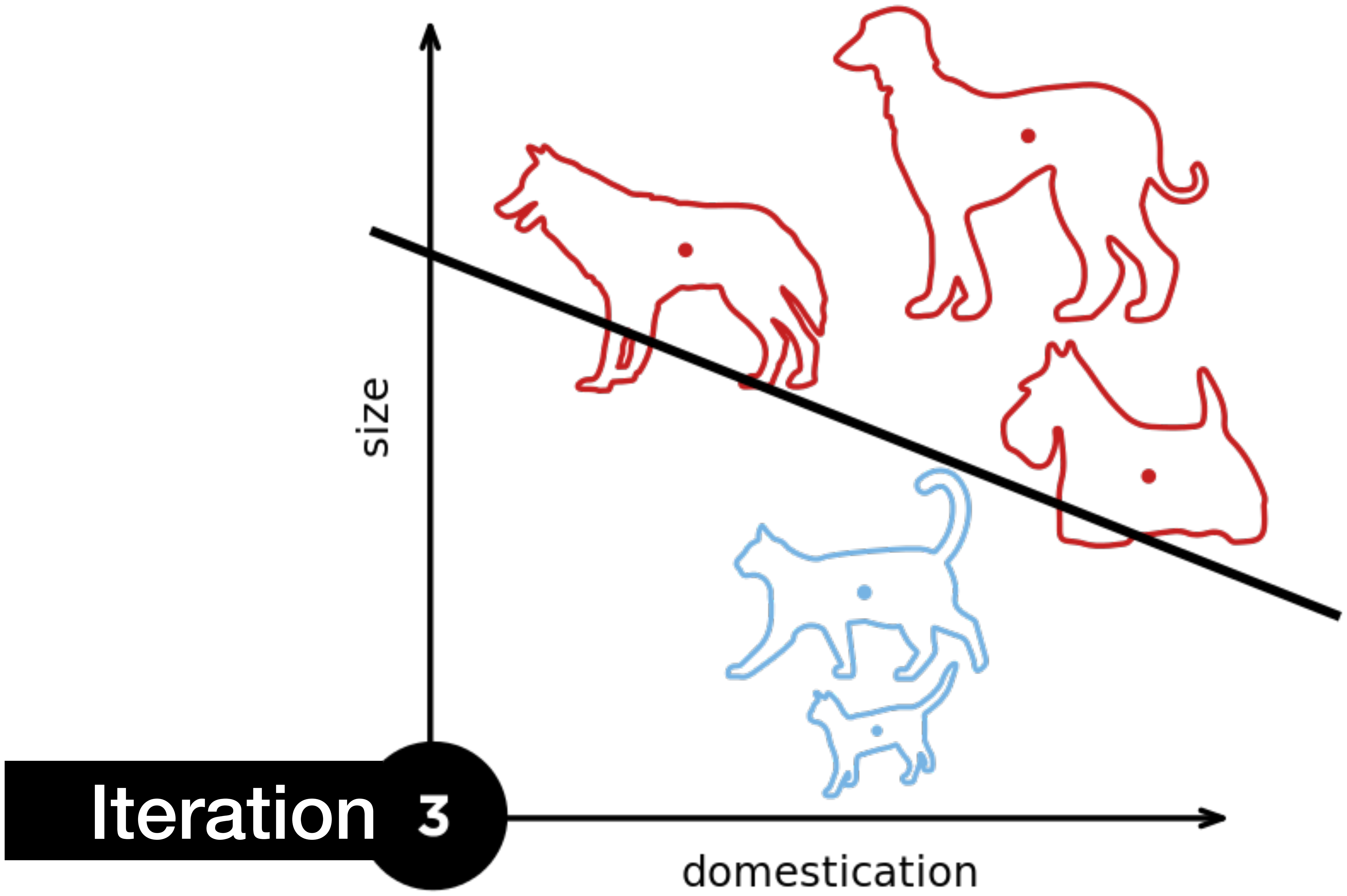
From wikipedia

Perceptron



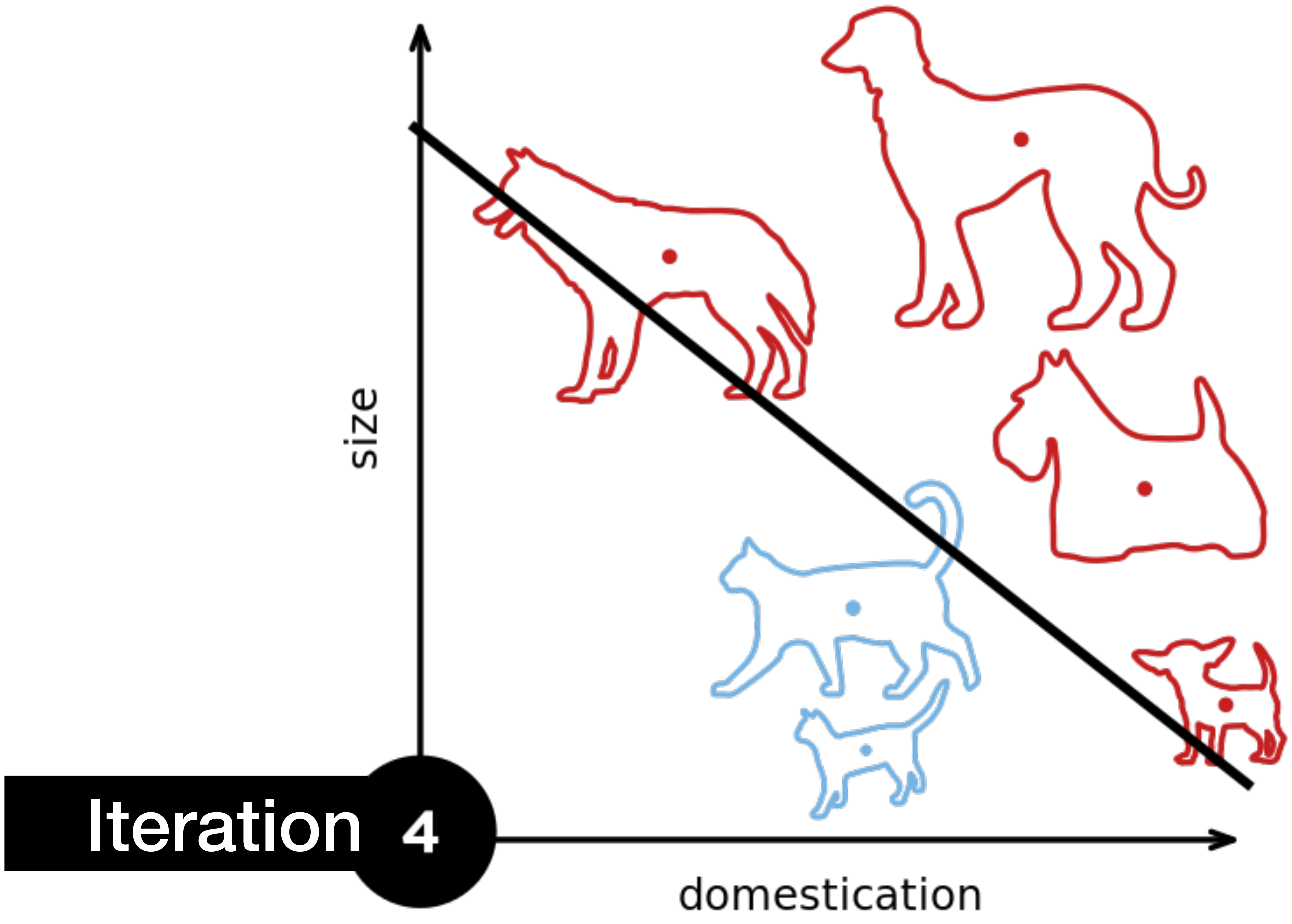
From wikipedia

Perceptron



From wikipedia

Perceptron



From wikipedia

Learning AND function using perceptron

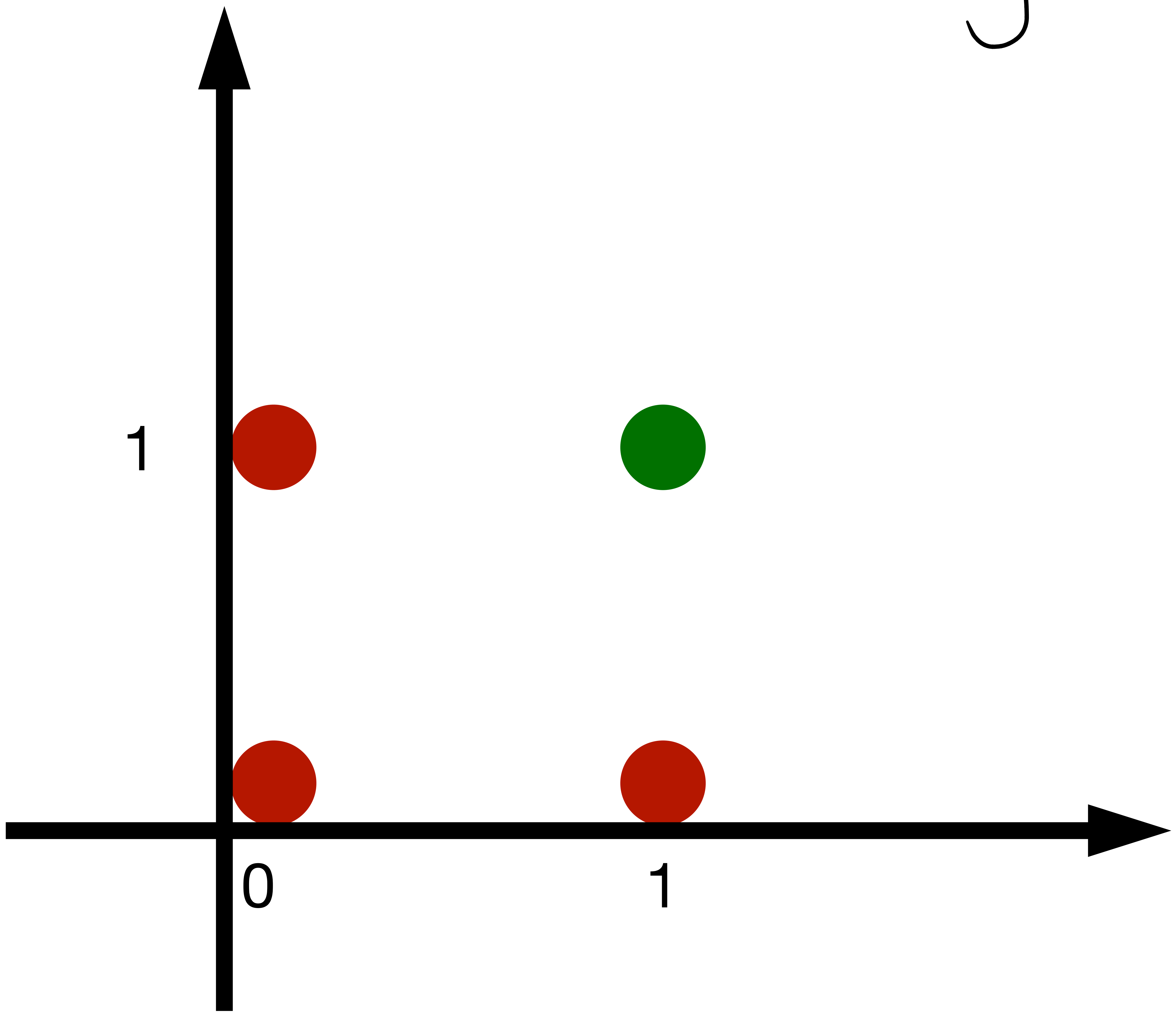
The perceptron can learn an AND function $y = x_1 \wedge x_2$

$x_1 = 1, x_2 = 1, y = 1$

$x_1 = 1, x_2 = 0, y = 0$

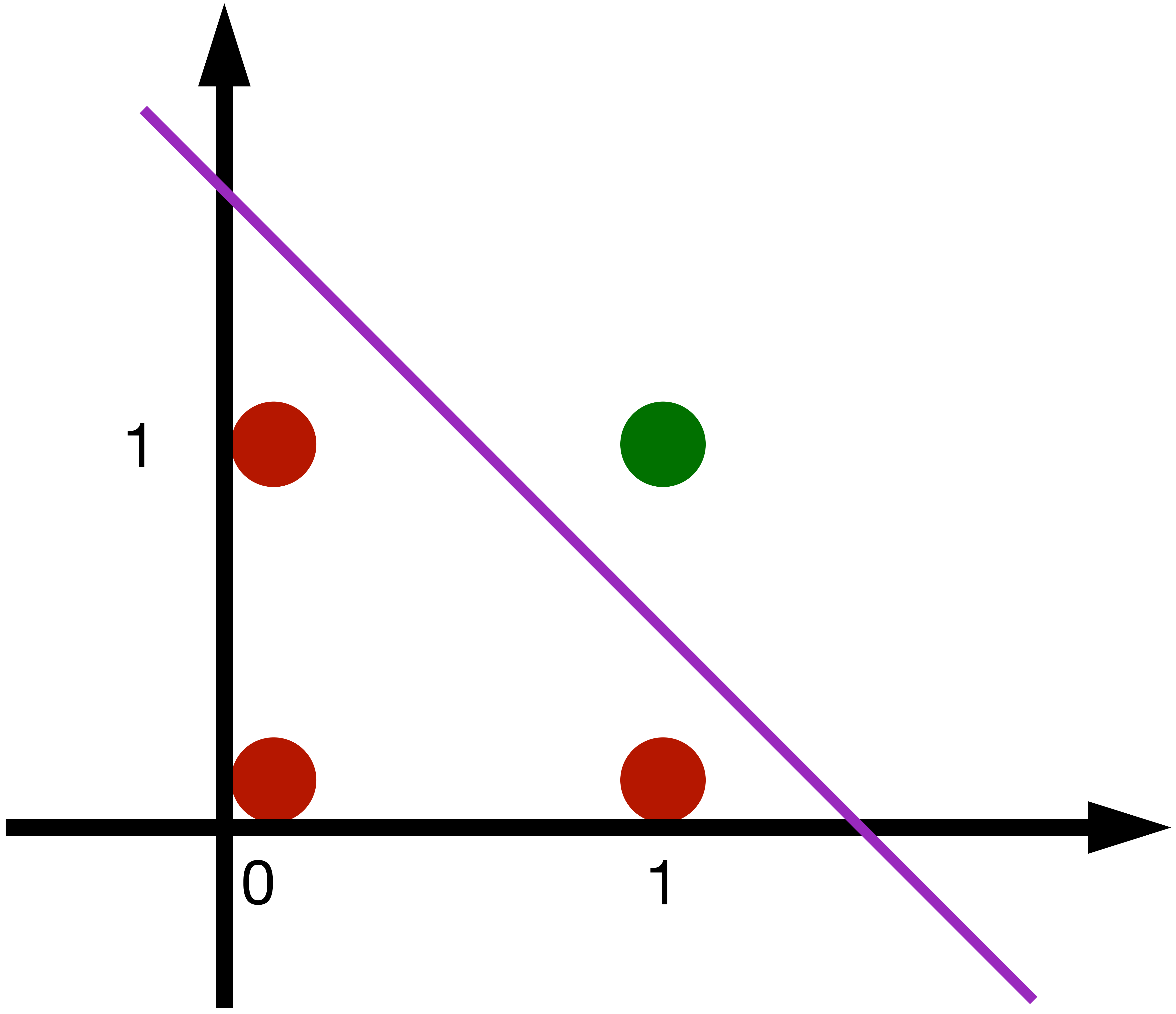
$x_1 = 0, x_2 = 1, y = 0$

$x_1 = 0, x_2 = 0, y = 0$



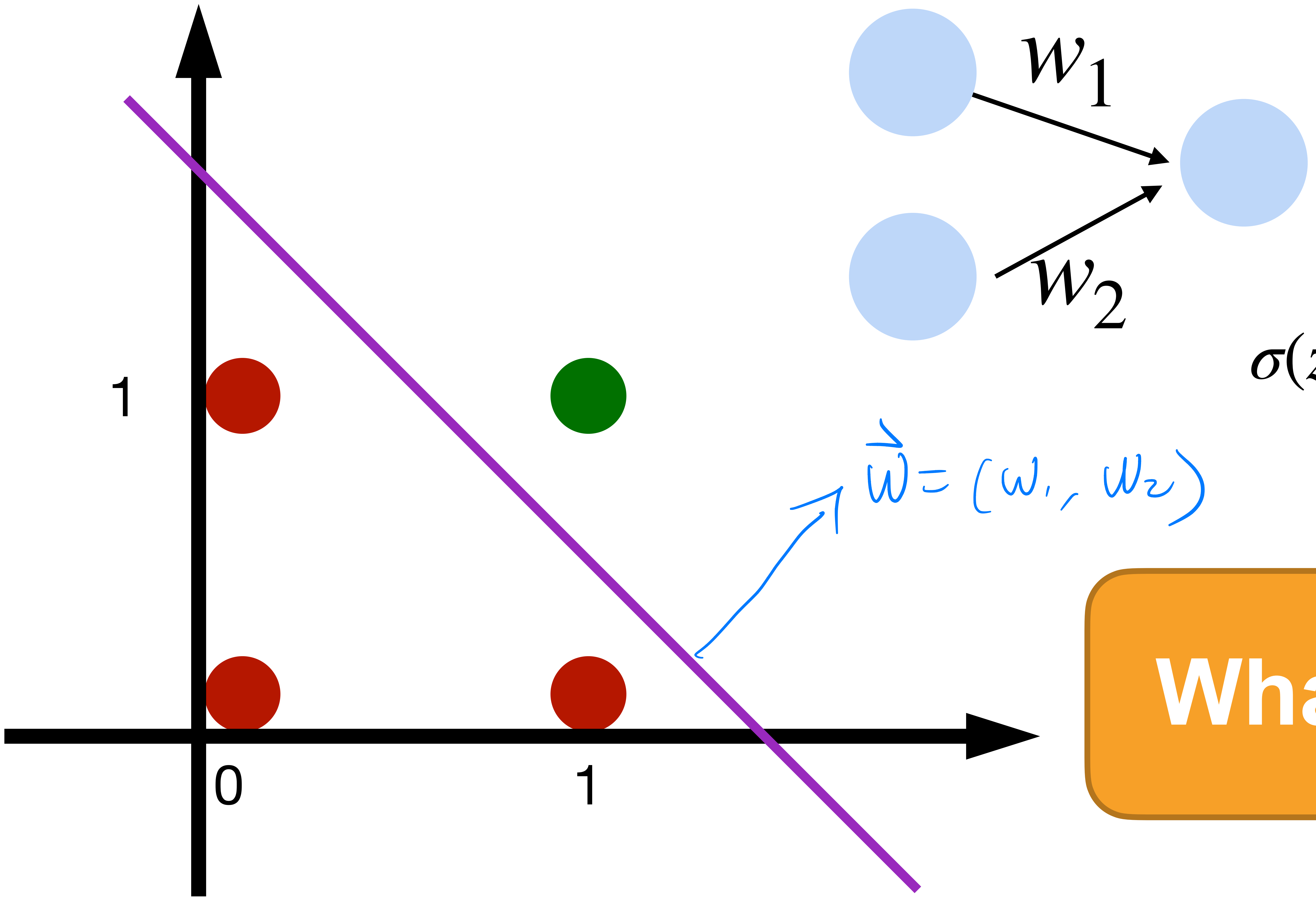
Learning AND function using perceptron

The perceptron can learn an AND function



Learning AND function using perceptron

The perceptron can learn an AND function



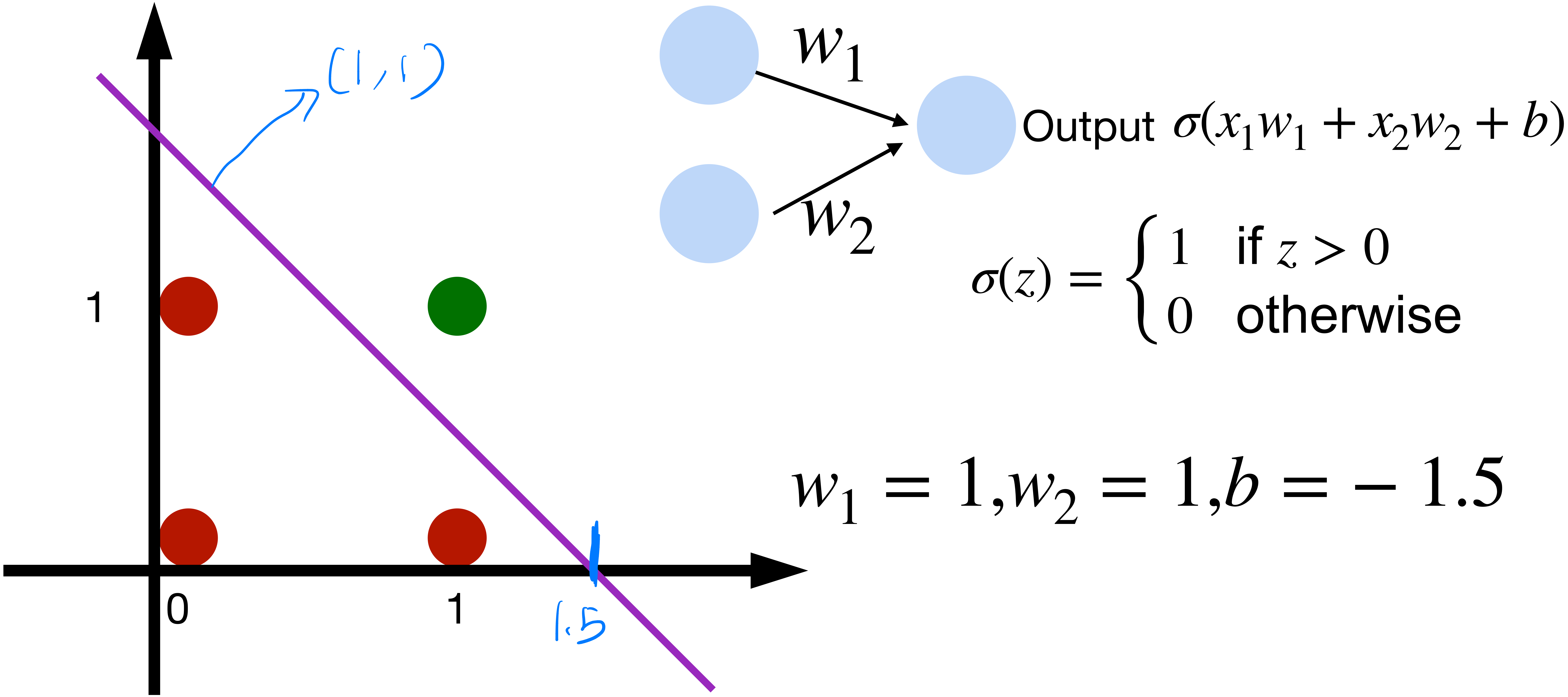
Output $\sigma(x_1 w_1 + x_2 w_2 + b)$

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

What's w and b?

Learning AND function using perceptron

The perceptron can learn an AND function



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

$$w_1 = 1, w_2 = 1, b = -1.5$$

Learning OR function using perceptron

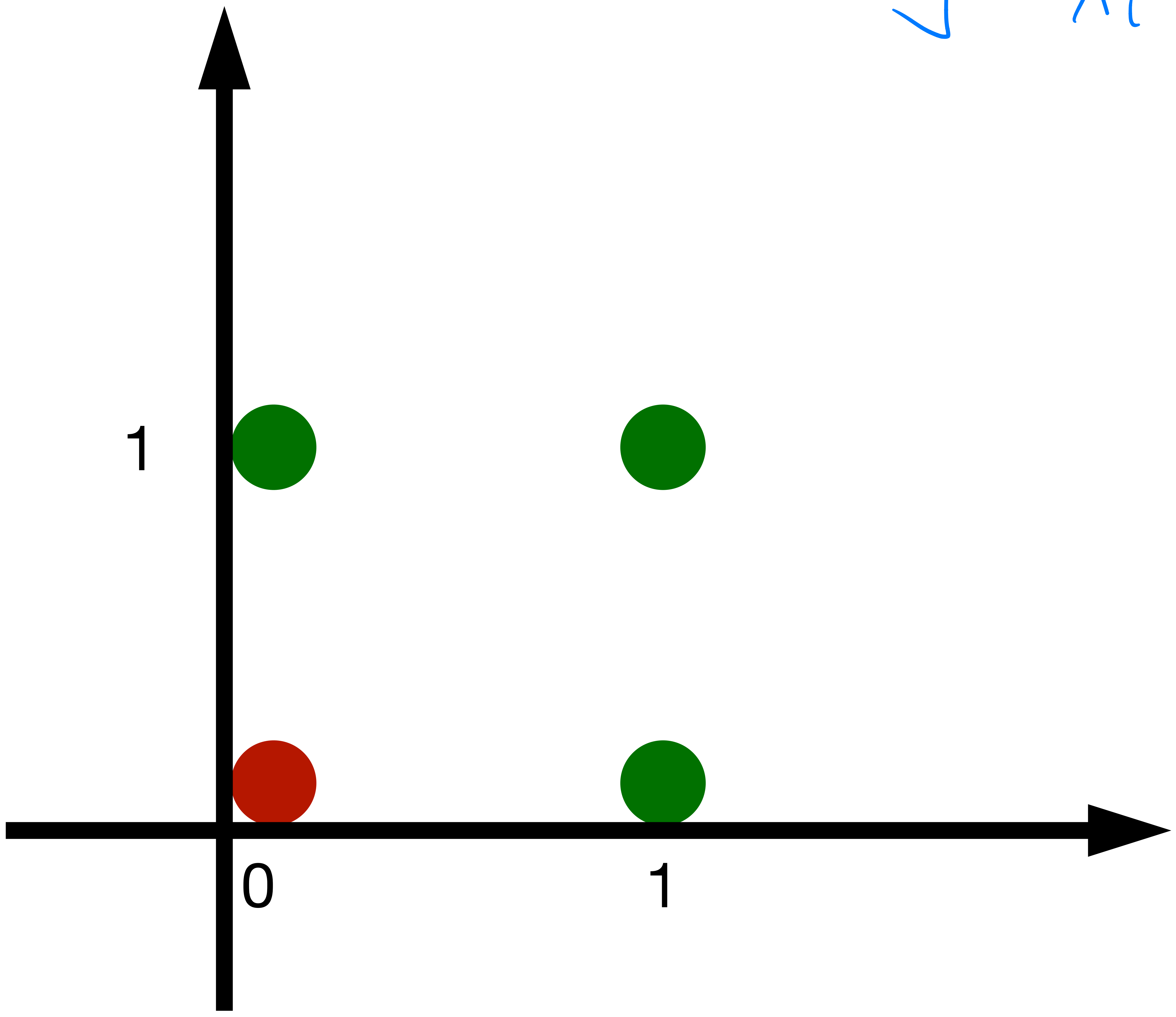
The perceptron can learn an OR function $y = x_1 \vee x_2$

$x_1 = 1, x_2 = 1, y = 1$

$x_1 = 1, x_2 = 0, y = 1$

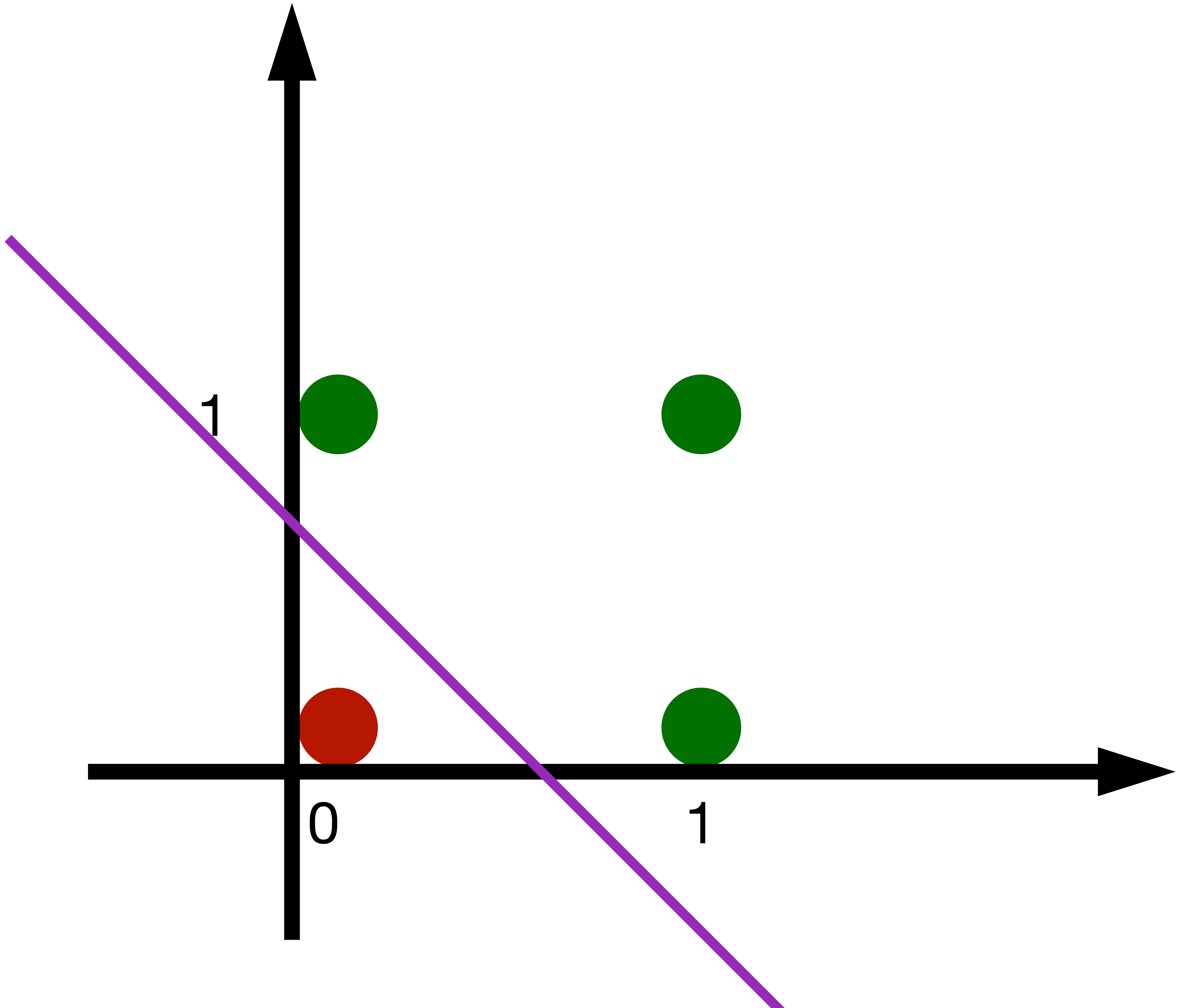
$x_1 = 0, x_2 = 1, y = 1$

$x_1 = 0, x_2 = 0, y = 0$



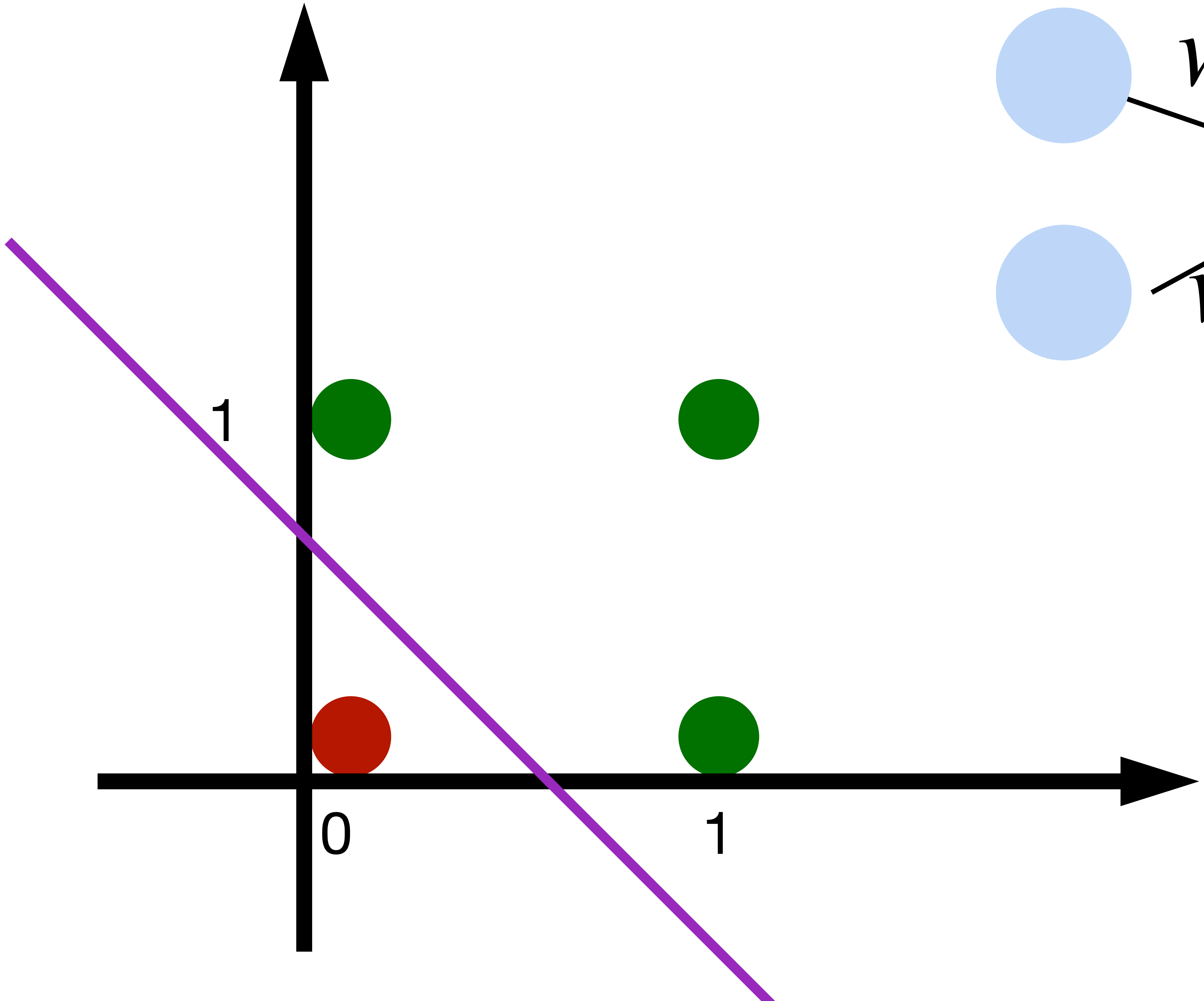
Learning OR function using perceptron

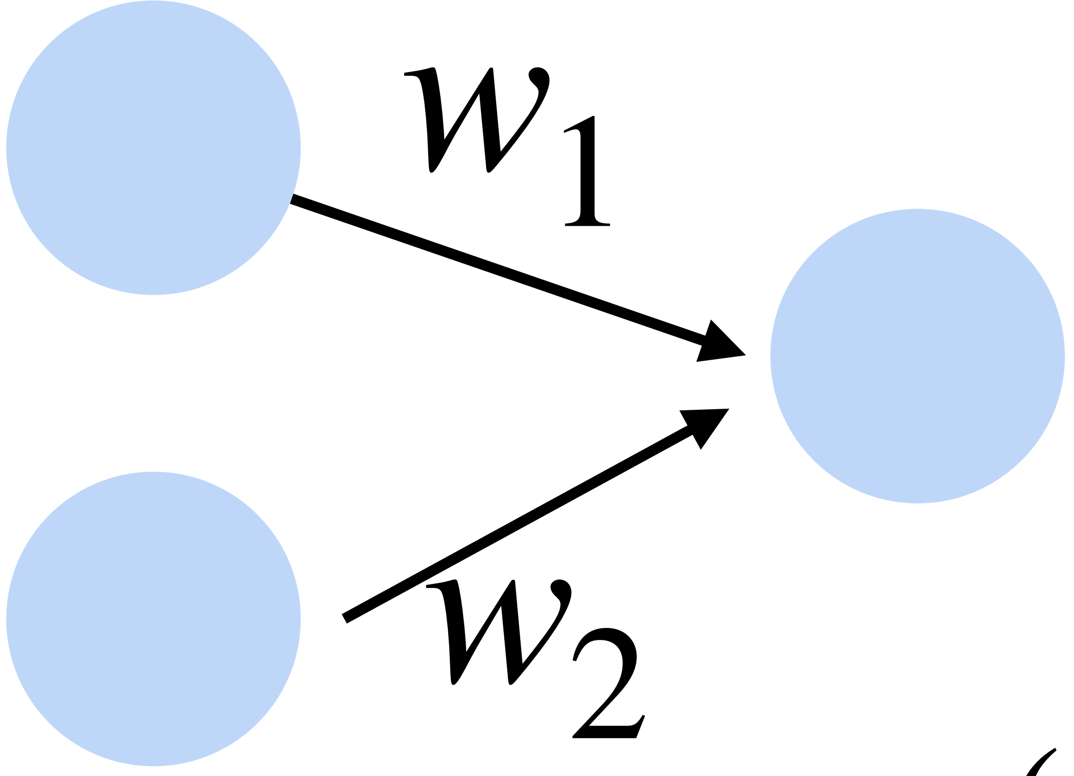
The perceptron can learn an OR function



Learning OR function using perceptron

The perceptron can learn an OR function





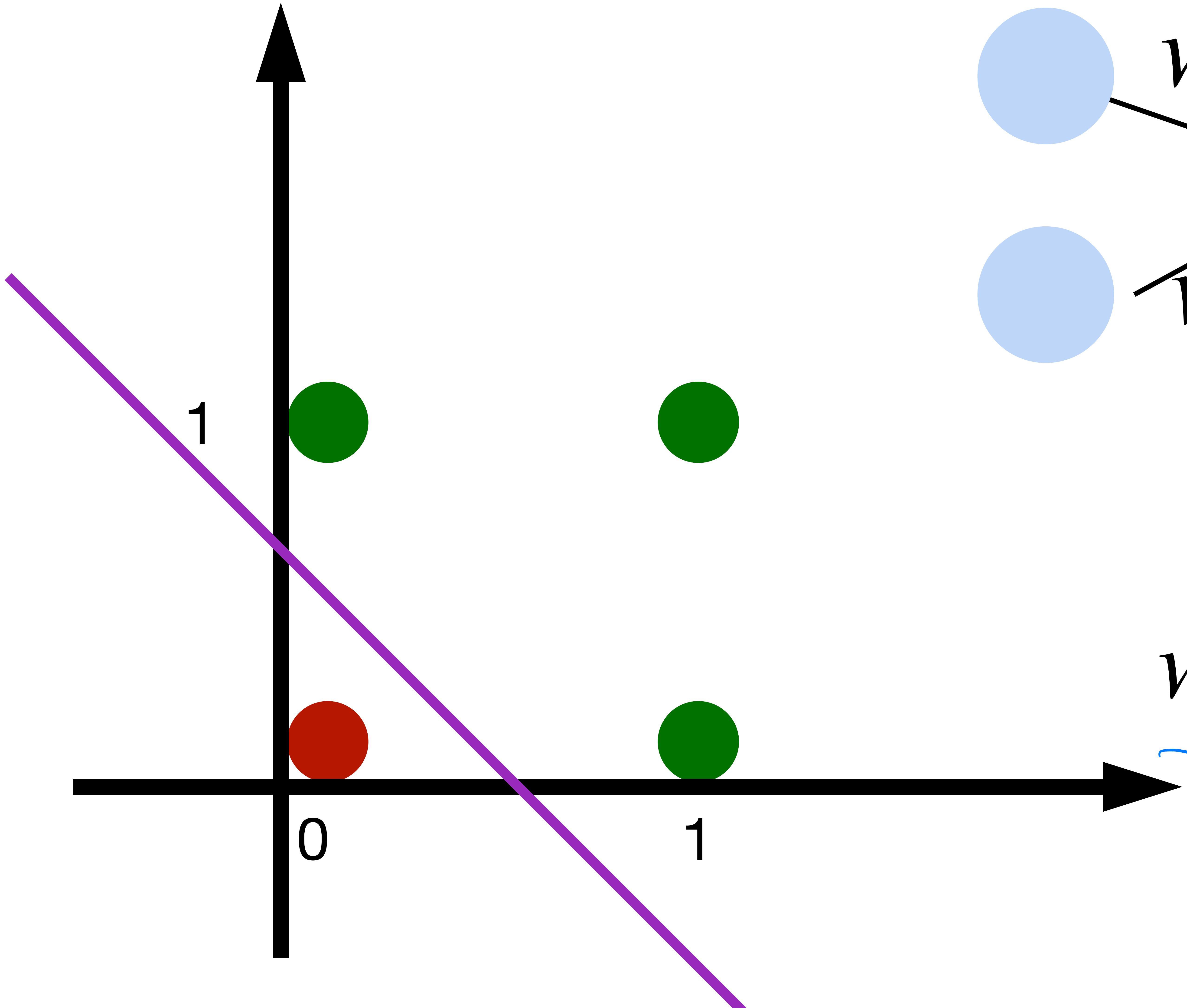
Output $\sigma(x_1 w_1 + x_2 w_2 + b)$

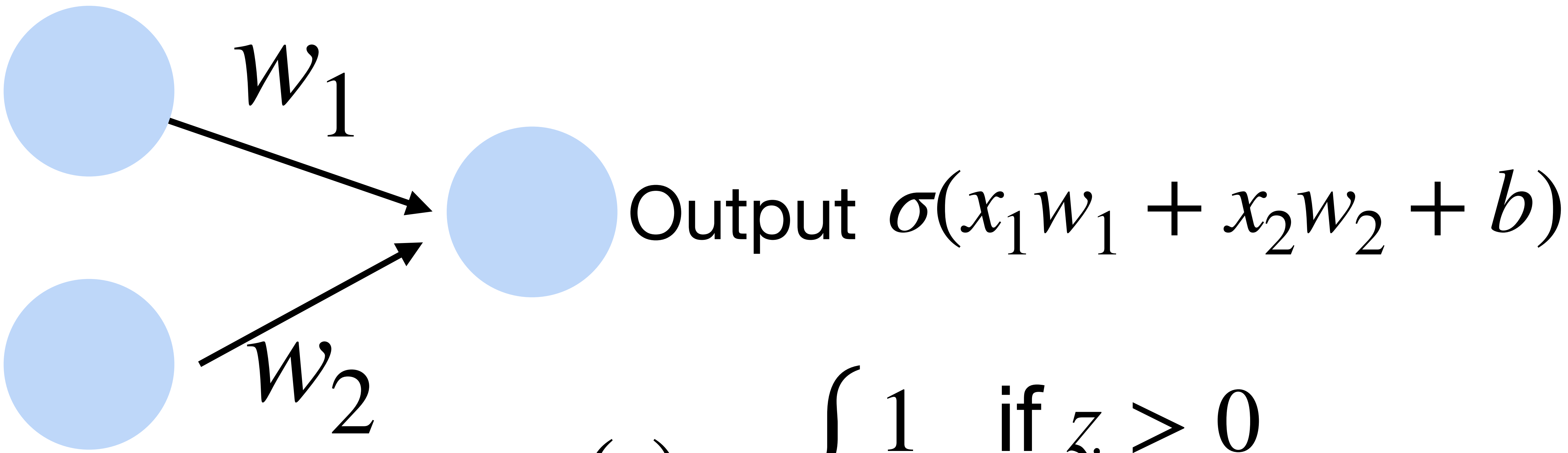
$$\sigma(z) = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{otherwise} \end{cases}$$

What's w and b?

Learning OR function using perceptron

The perceptron can learn an OR function





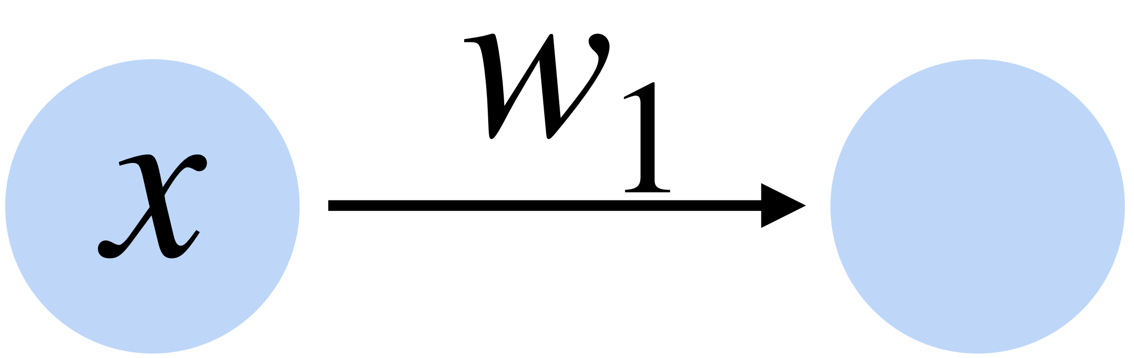
$$\sigma(z) = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$w_1 = 1, w_2 = 1, b = -0.5$$

Learning NOT function using perceptron

The perceptron can learn NOT function (single input)

$y = \neg x_1$



Output $\sigma(xw_1 + b)$

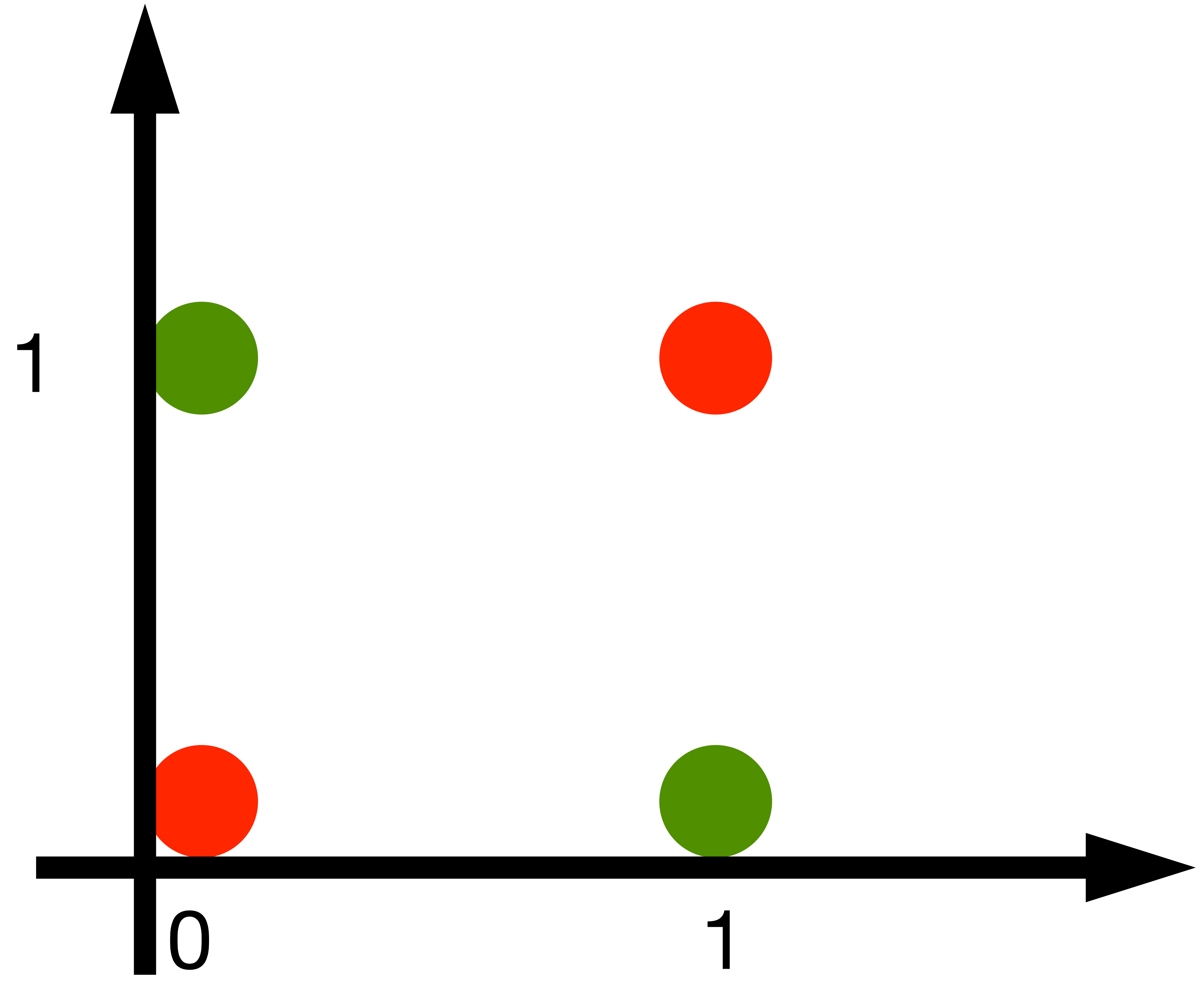
$$\sigma(z) = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$w_1 = -1, b = 0.5$$

$$\sigma(-x + 0.5) = \begin{cases} 1 & \text{if } -x + 0.5 > 0 \iff x = 0 \\ 0 & \text{if } -x + 0.5 < 0 \iff x = 1 \end{cases}$$

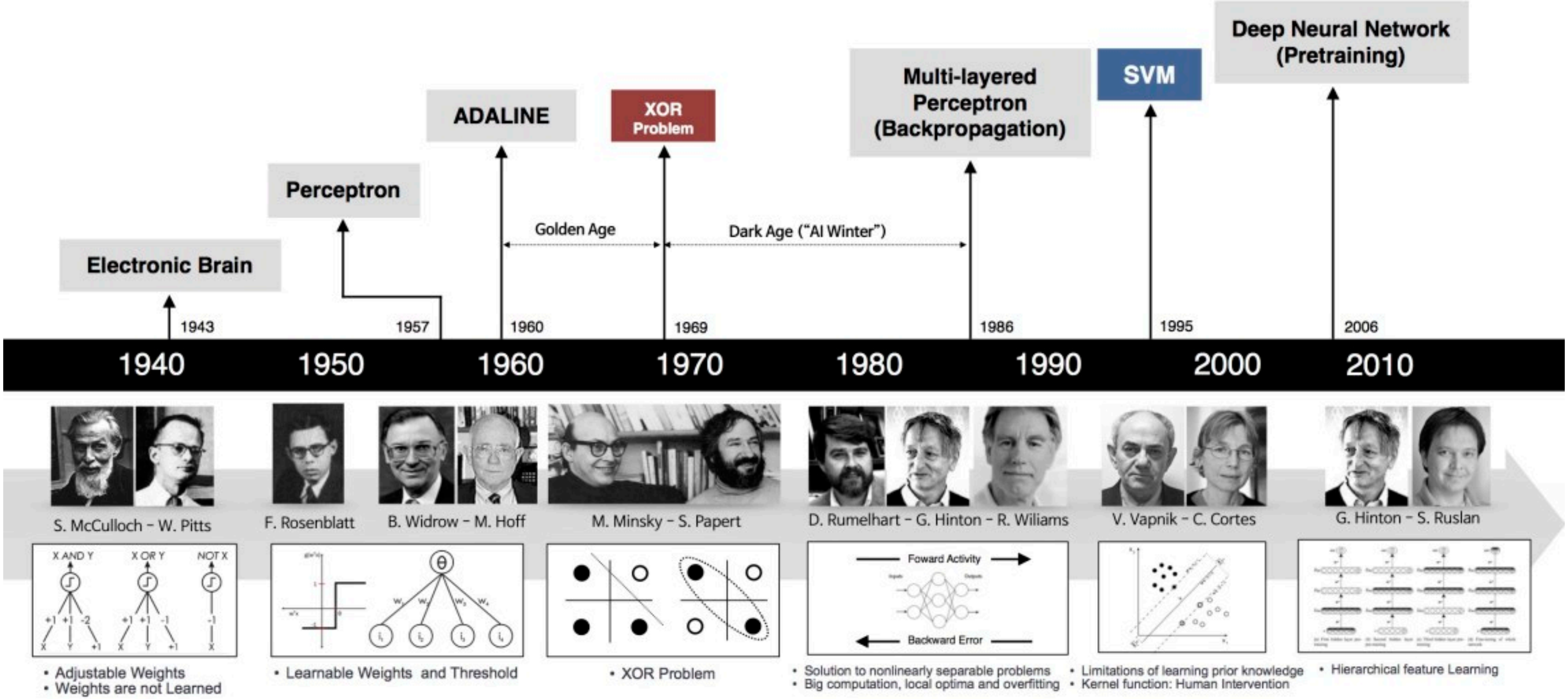
XOR Problem (Minsky & Papert, 1969)

The perceptron cannot learn an XOR function (neurons can only generate linear separators)



This contributed to the first AI winter

Brief history of neural networks



Quiz Break

Consider the linear perceptron with x as the input. Which function can the linear perceptron compute?

(1) $y = \text{ax} + \text{b}$

(2) $y = \text{ax}^2 + bx + c$

A. (1)

B. (2)

C. (1)(2)

D. None of the above

Quiz Break

Consider the linear perceptron with x as the input. Which function can the linear perceptron compute?

(1) $y = ax + b$

(2) $y = ax^2 + bx + c$

A. (1)

B. (2)

C. (1)(2)

D. None of the above

Answer: A. All units in a linear perceptron are linear. Thus, the model can not present non-linear functions.

Quiz Break

Perceptron can be used for representing:

A. AND function
B. OR function
C. XOR function
D. Both AND and OR function

Quiz Break

Perceptron can be used for representing:

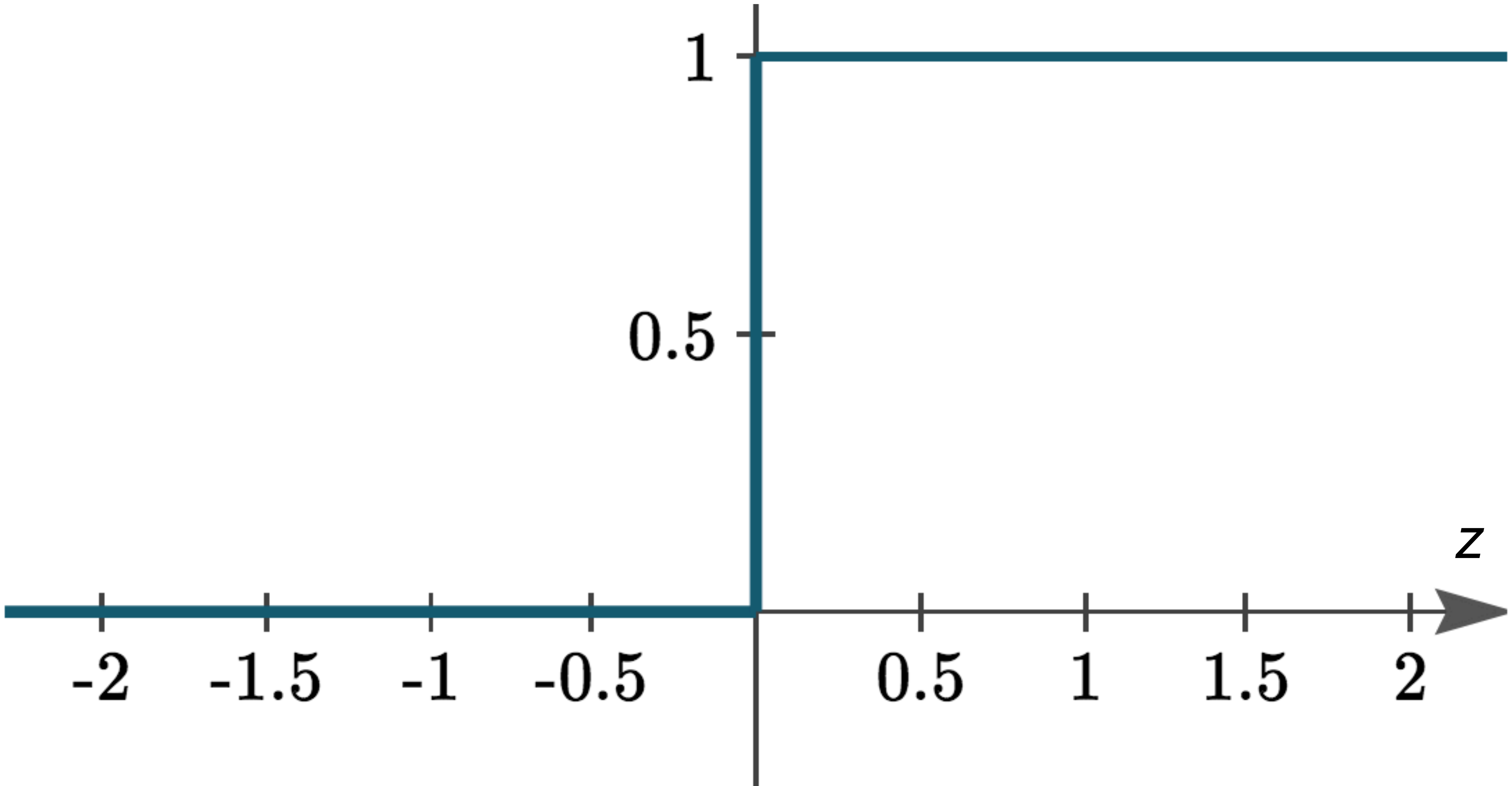
A. AND function
B. OR function
C. XOR function
D. Both AND and OR function

NOT

Step Function activation

Step function is discontinuous, which cannot be used for gradient descent

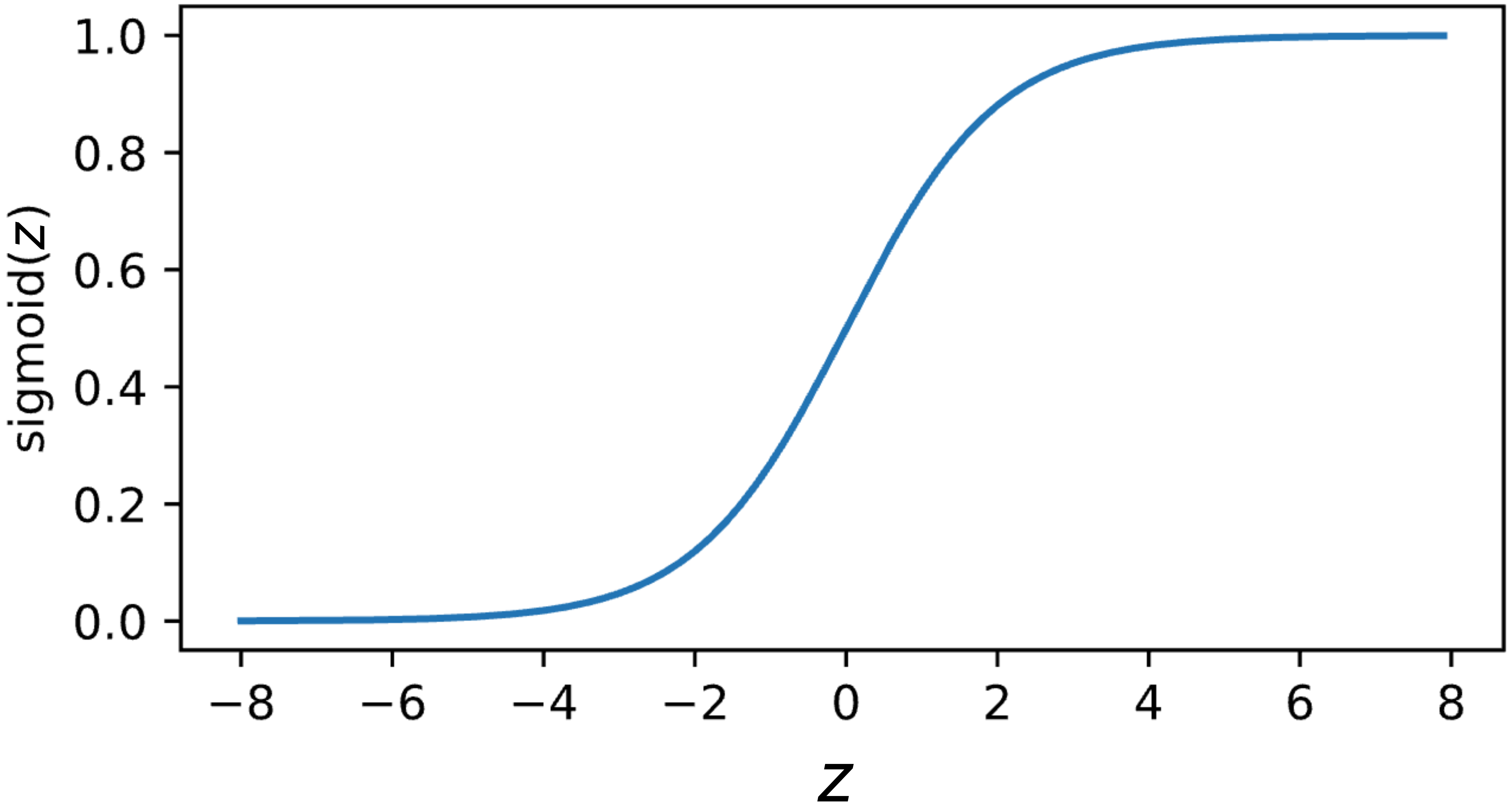
$$\sigma(z) = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{otherwise} \end{cases}$$



Sigmoid/Logistic Activation

Map input into [0, 1], a **soft** version of $\sigma(z) = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{otherwise} \end{cases}$

$$\text{sigmoid}(z) = \frac{1}{1 + \exp(-z)} = \begin{cases} \to 1 & z \to \infty \\ \to 0 & z \to -\infty \end{cases}$$



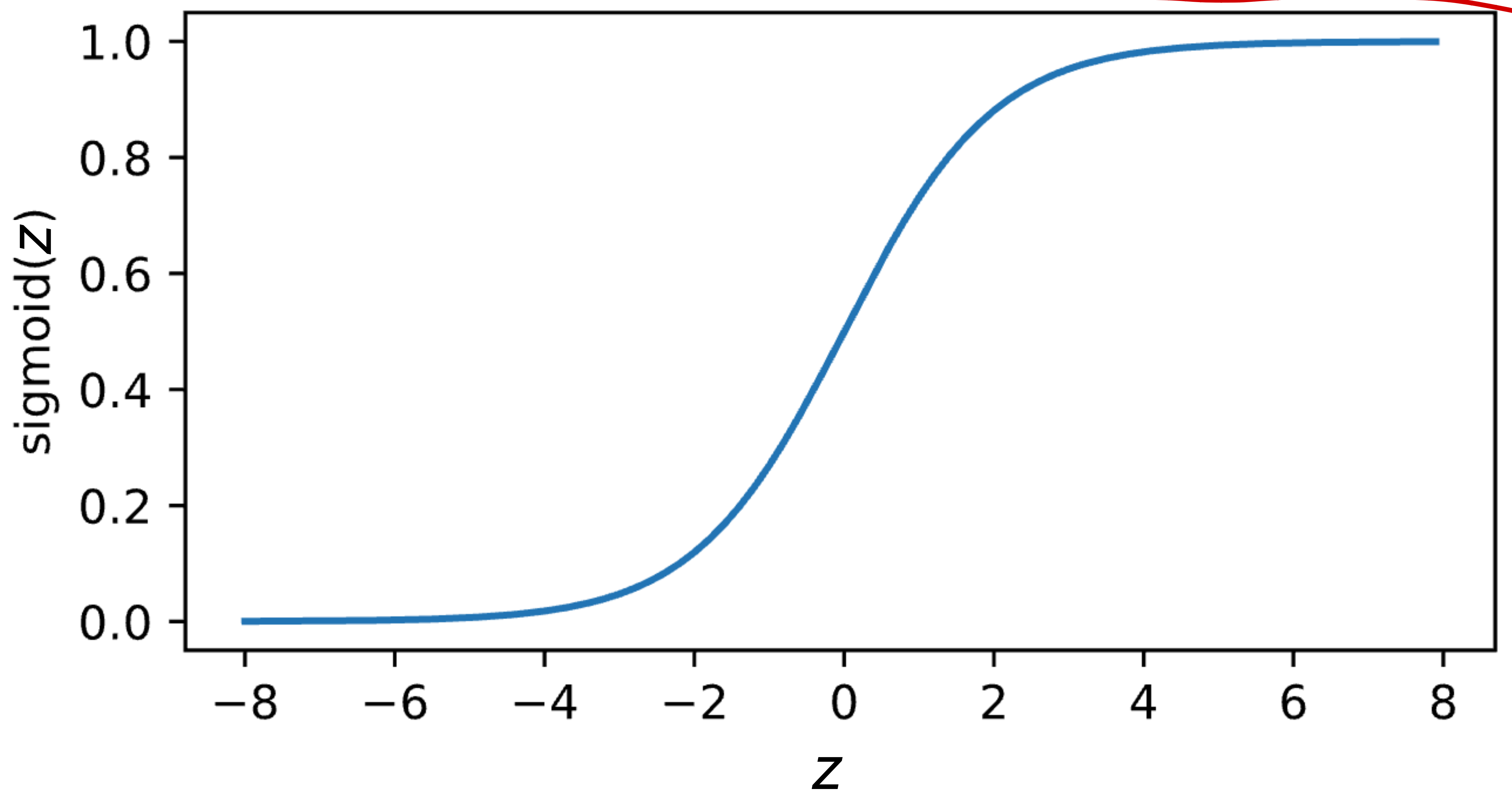
Logistic regression

$p(y|x) = \frac{1}{1+\exp(-y \cdot w^T x)}$

$\mathbf{x} \in \mathbb{R}^d, y = \{-1, +1\}$

$$p(y = 1 | \mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

$$p(y = -1 | \mathbf{x}) = 1 - \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(\mathbf{w}^T\mathbf{x})}$$



Logistic regression

Given: $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$

Training: maximize the likelihood (the conditional probability)

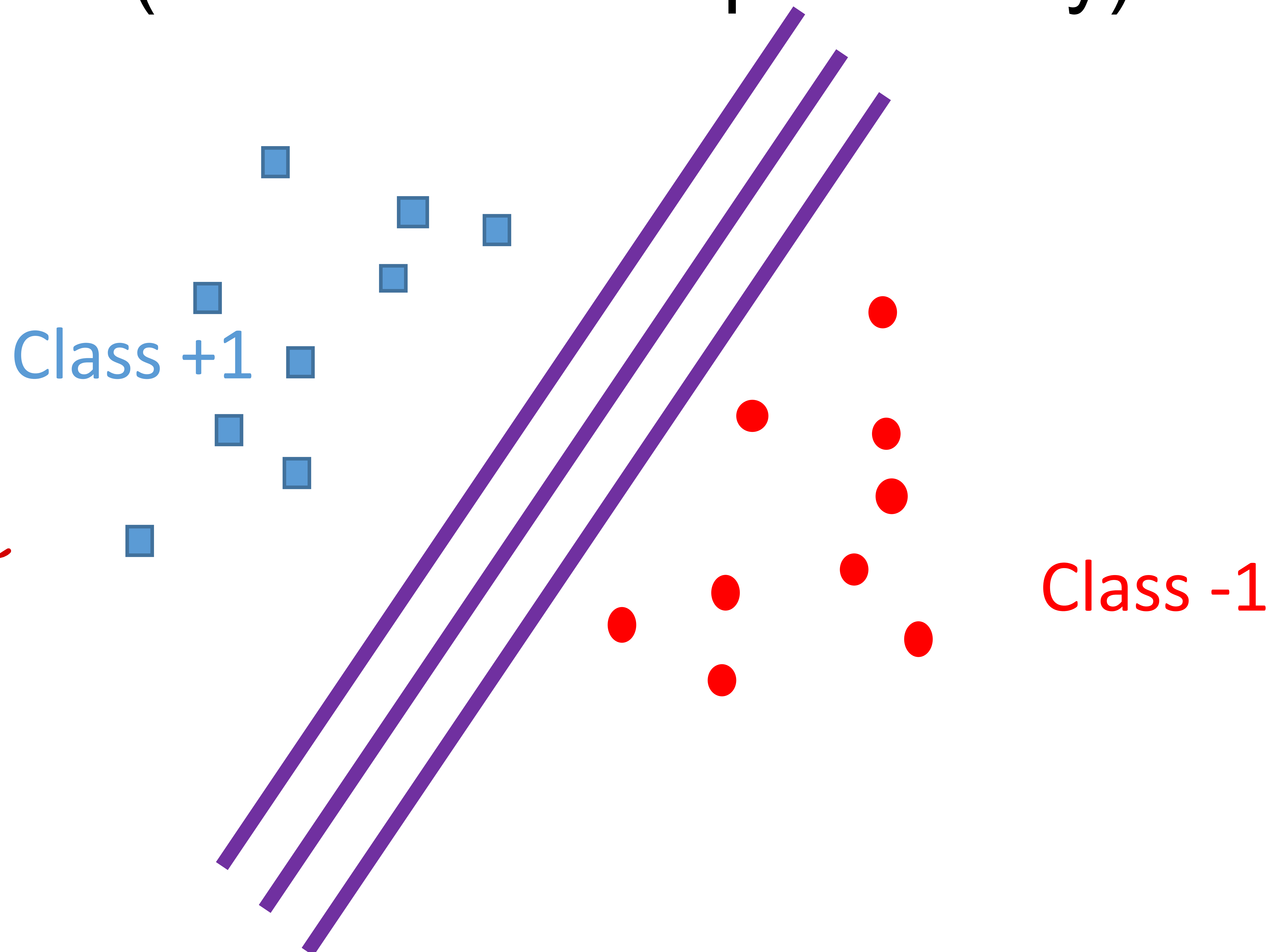
$$\max_{\mathbf{w}} \sum_i \log \frac{1}{1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)}$$

$$\max \log \prod_{i=1}^{n} p(y_i \mid x_i) = \max \sum_{i=1}^{n} \frac{1}{1+\exp(-y_i w^T x_i)}$$

Logistic regression

Given: $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$

Training: maximize the likelihood (the conditional probability)



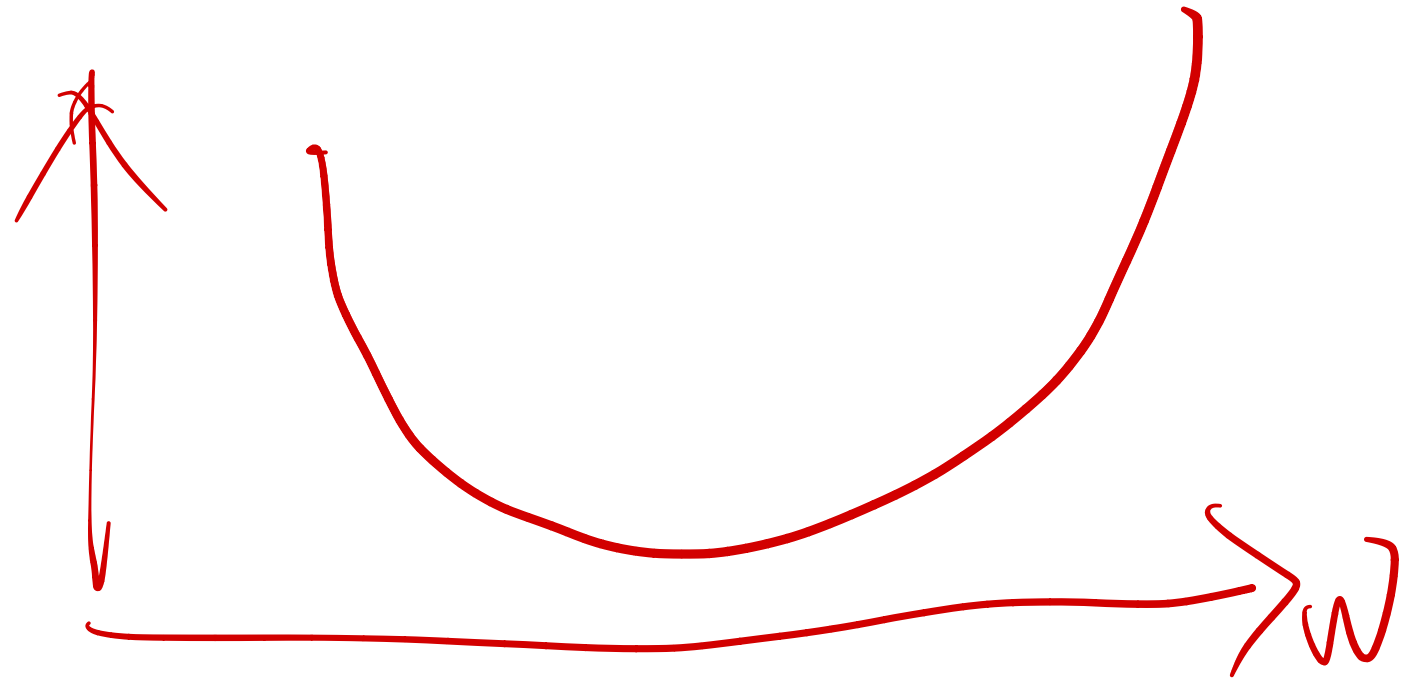
When training data is linearly separable, many solutions

Logistic regression

$\max_{w}$ log-likelihood $- \frac{\lambda}{2} \|w\|^2$

Given: $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$

Training: maximize the regularized likelihood



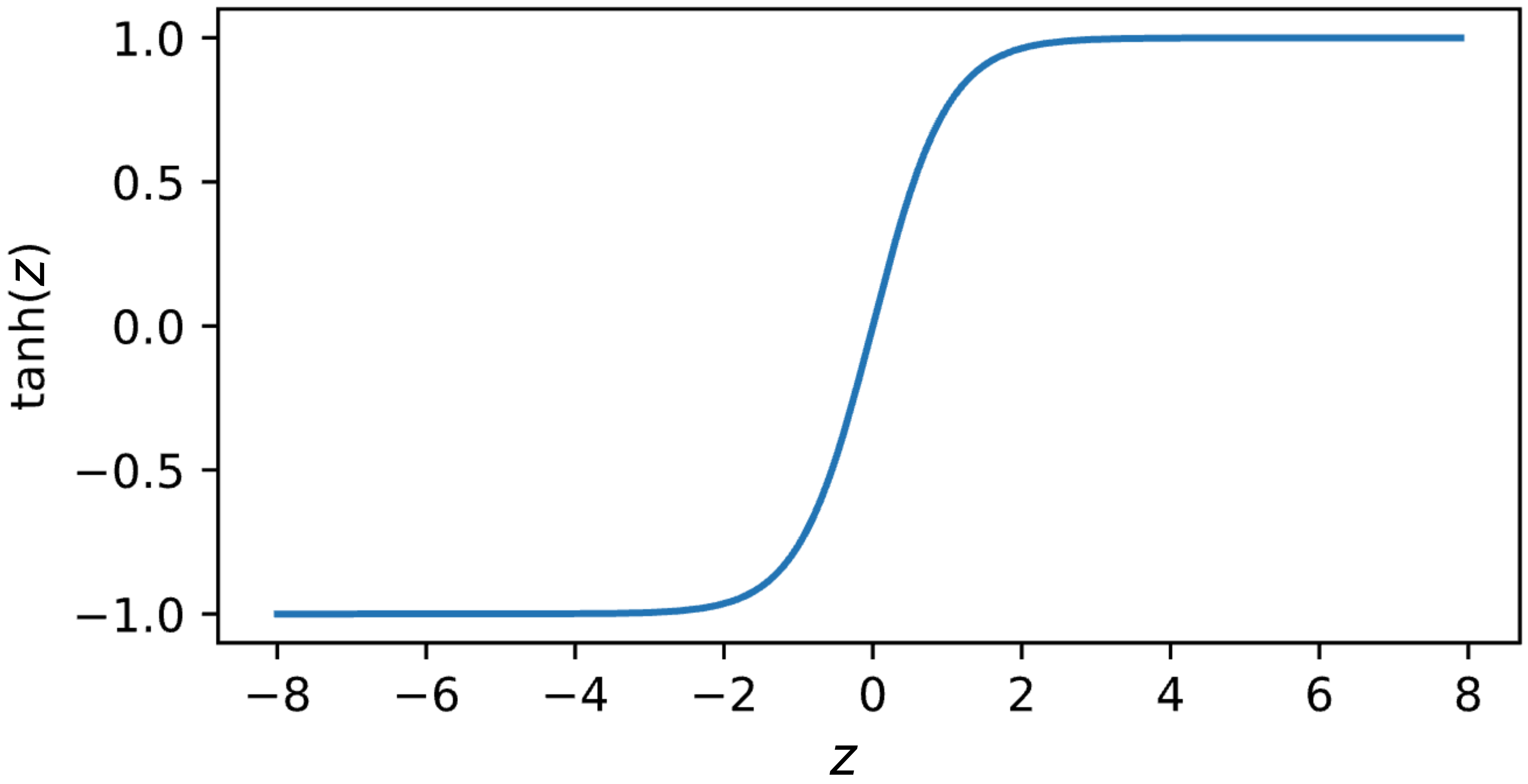
$$\min_{\mathbf{w}} \sum_i \left(-\log \frac{1}{1+\exp(-y_i \mathbf{w}^T \mathbf{x}_i)} \right) + \frac{\lambda}{2} \|\mathbf{w}\|_2^2$$

- Convex optimization
- Solve via (stochastic) gradient descent
- Related to *maximum A posteriori (MAP)* estimate

Tanh Activation

Map inputs into (-1, 1)

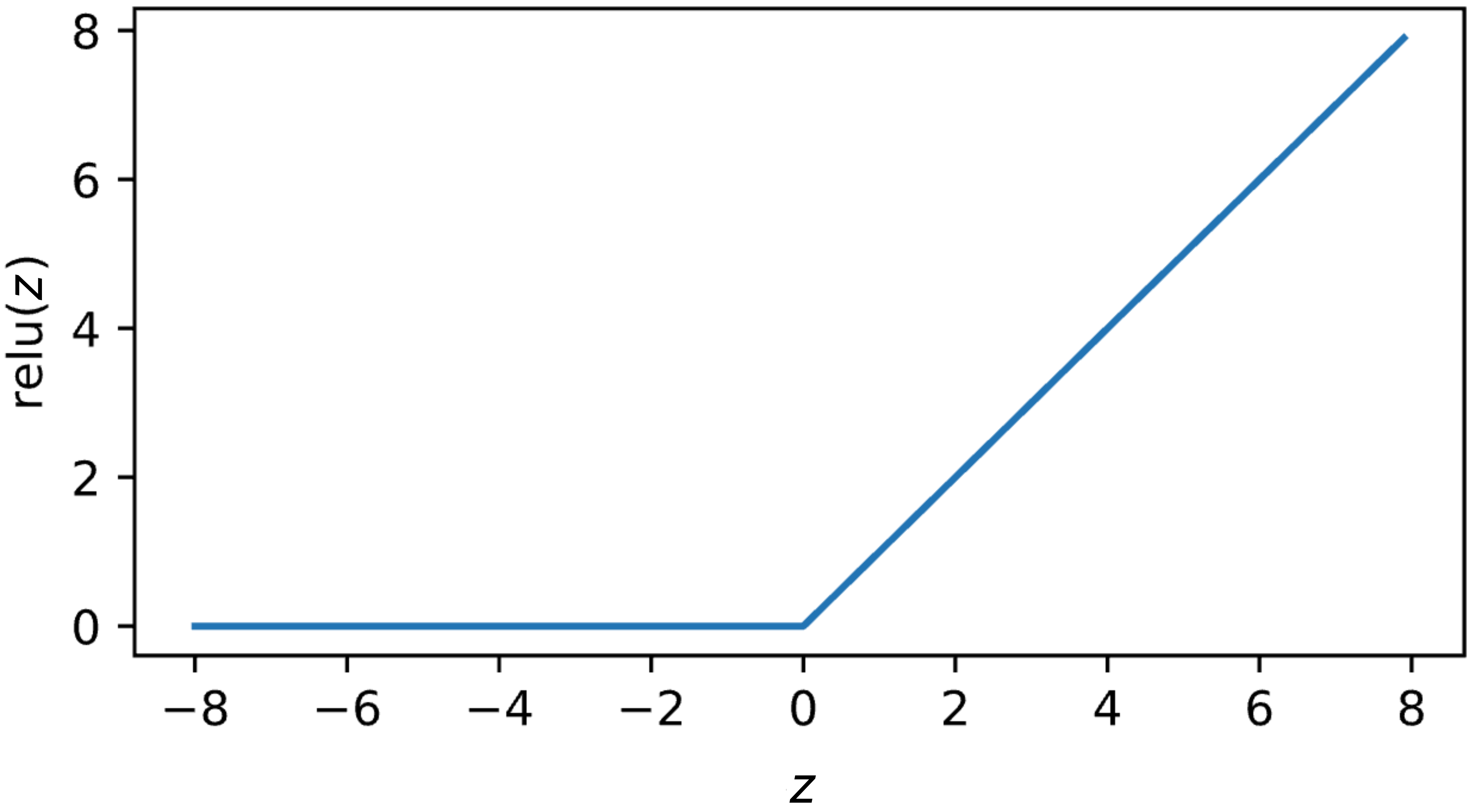
$$\tanh(z) = \frac{1 - \exp(-2z)}{1 + \exp(-2z)} = \left(\text{sigmoid}(z) - \frac{1}{2}\right) \cdot 2$$



ReLU Activation

ReLU: Rectified Linear Unit (commonly used in modern neural networks)

$$\mathrm{ReLU}(z) = \max(z, 0)$$



Quiz Break

Which one of the following is valid activation function

a) Step function

b) Sigmoid function

C) ReLU function

D) all of above

Quiz Break

Which one of the following is valid activation function

a) Step function

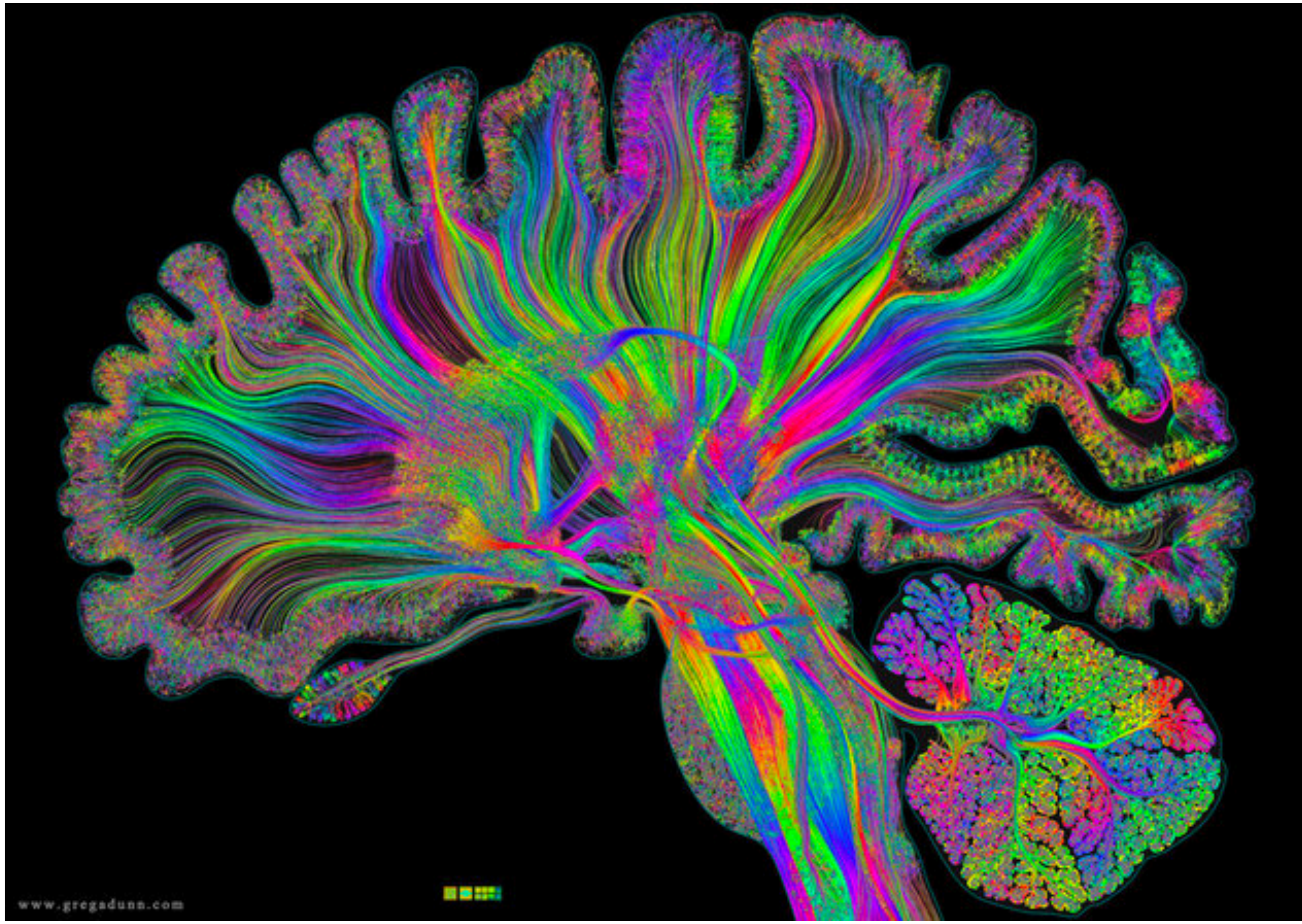
b) Sigmoid function

C) ReLU function

D) all of above

Coming Next:

Multi-layer Perceptron



Thanks!

Based on slides from Sharon Li, Xiaojin (Jerry) Zhu and Yingyu Liang, and Alex Smola: https://courses.d2l.ai/berkeley-stat-157/units/mlp.html

Thanks!

Based on slides from Sharon Li, Xiaojin (Jerry) Zhu and Yingyu Liang, and Alex Smola: https://courses.d2l.ai/berkeley-stat-157/units/mlp.html