Machine learning & deep learning basics

Lecture 2

Changho Suh

September 27, 2021

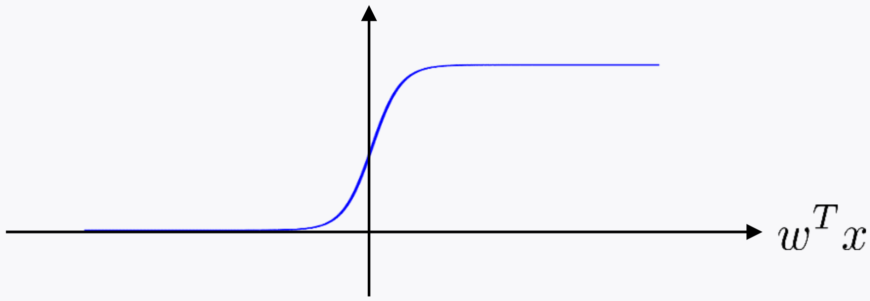
Gradient descent and DNNs

Logistic regression

$$\min_{w} \sum_{i=1}^{m} \ell(y^{(i)}, f_w(x^{(i)}))$$

Employ: Perceptron w/ logistic function

$$f_w(x) = \frac{1}{1 + e^{-w^T x}}$$

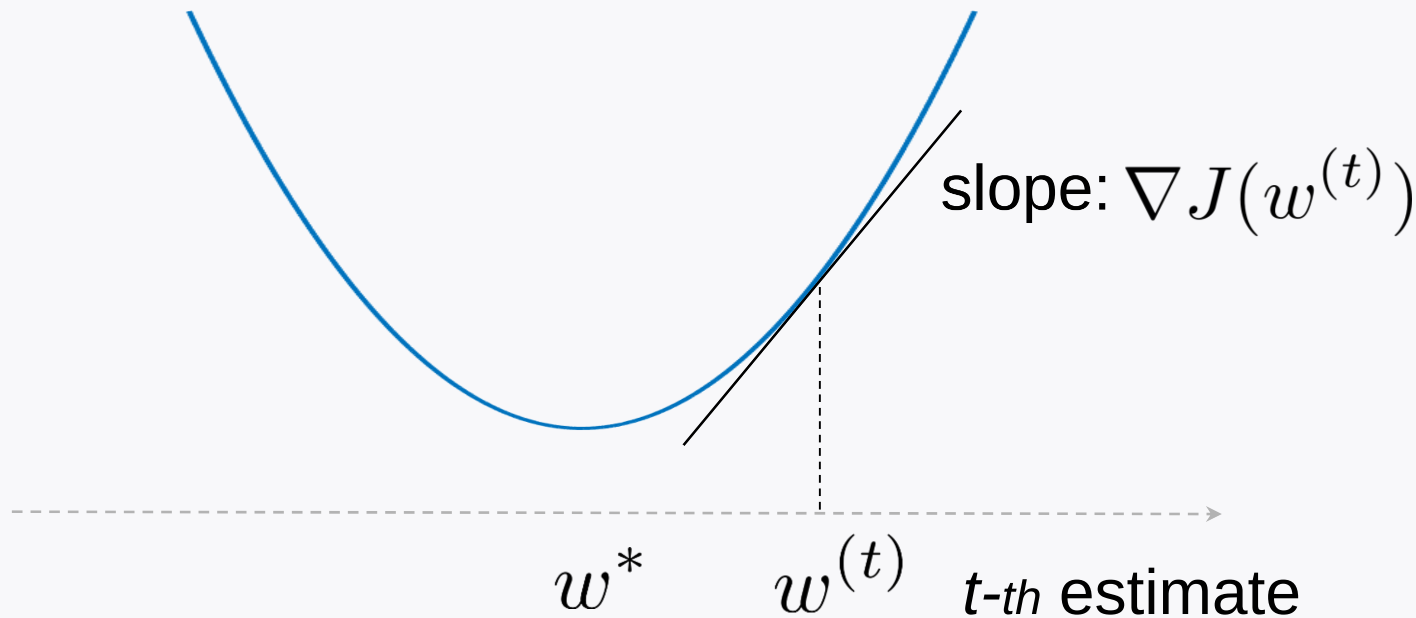


CE loss:

$$\ell(y, \hat{y}) = -y \log \hat{y} - (1 - y) \log(1 - \hat{y})$$

Gradient descent

Iterative algorithm



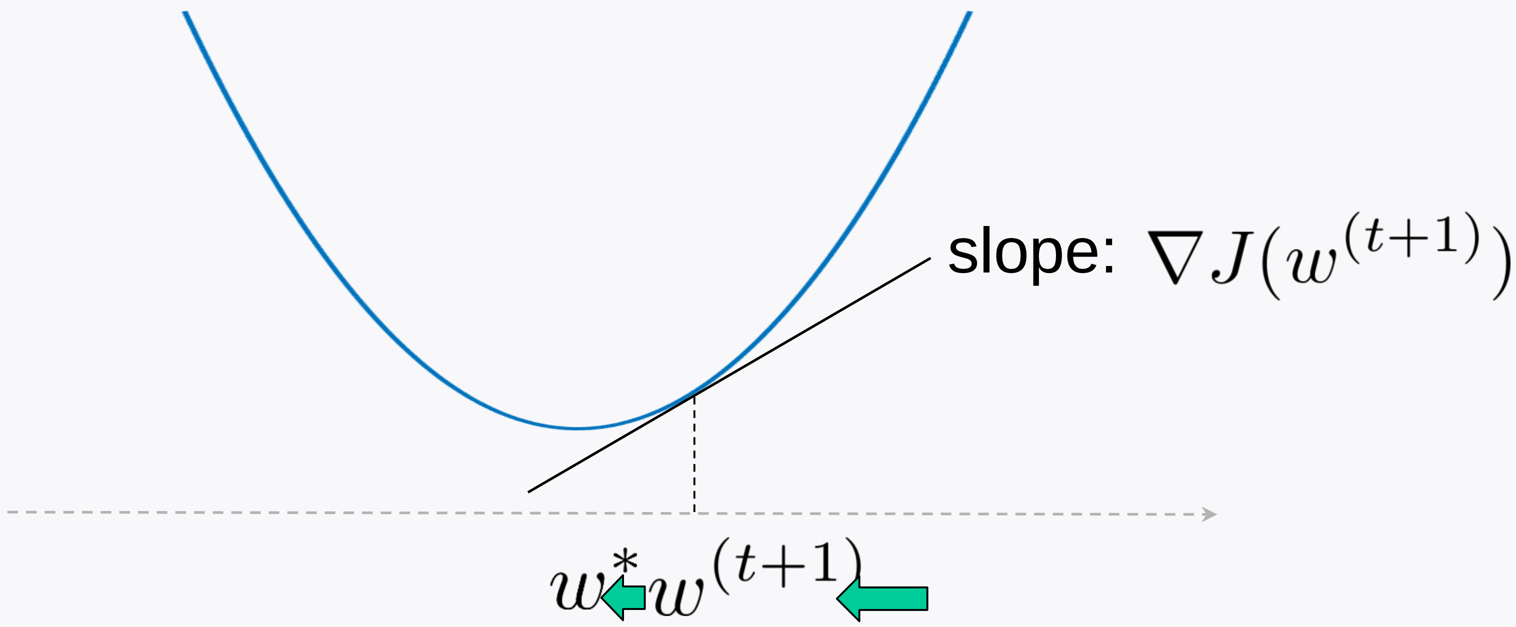
$$w^{(t+1)} \leftarrow w^{(t)} - \alpha^{(t)} \nabla J(w^{(t)})$$

move to the left!

learning rate (>0)

Gradient descent

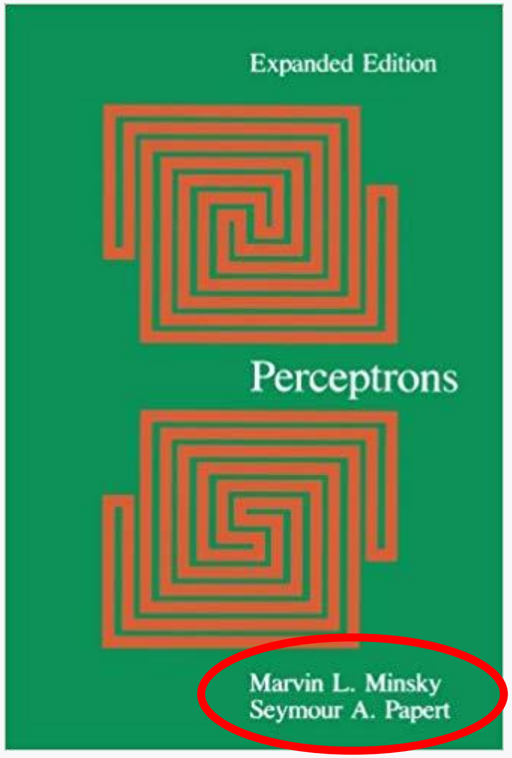
(t+1)-*th* step:

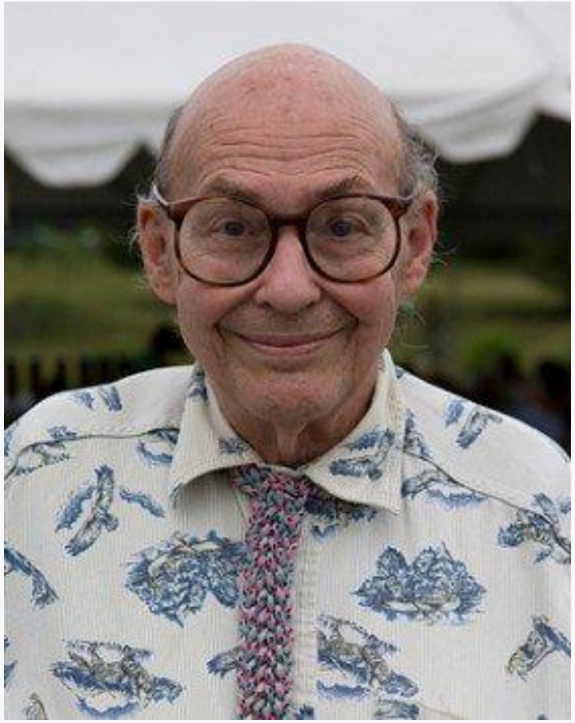


$$w^{(t+2)} \leftarrow w^{(t+1)} - \alpha^{(t+1)} \nabla J(w^{(t+1)})$$

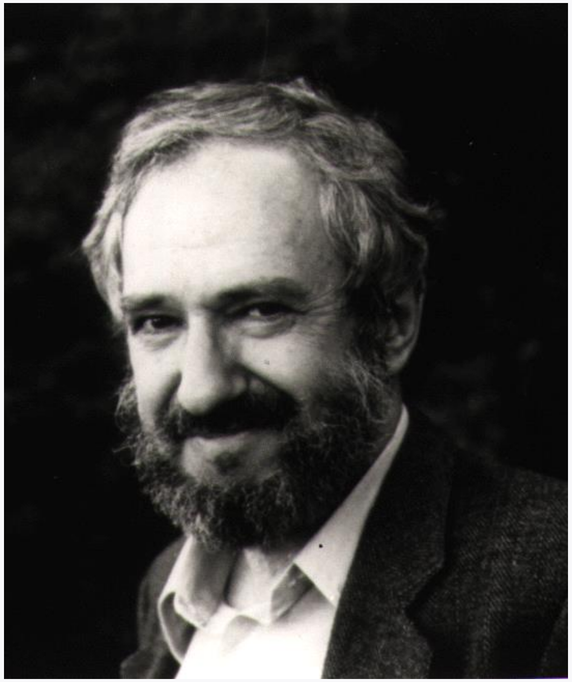
Turns out: $w^{(t)} \longrightarrow w^* \quad \text{as } t \rightarrow \infty$

AI boomed in 1960s but …





Marvin Minsky



Seymour Papert ‘69

Demonstrated limitations of the perceptron architecture.

→ Led to the AI winter!

AI revived in 2012

Alex Krizhevsky

Ilya Sutskever

Geoffrey Hinton

Won the ImageNet competition in 2012!

Demonstrated: Deep neural networks can achieve *human-level recognition performances*!

Anchored the start of deep learning revolution!

Deep neural networks

Say: A neural network is deep if it has **at least one layer** btw input/output layers.

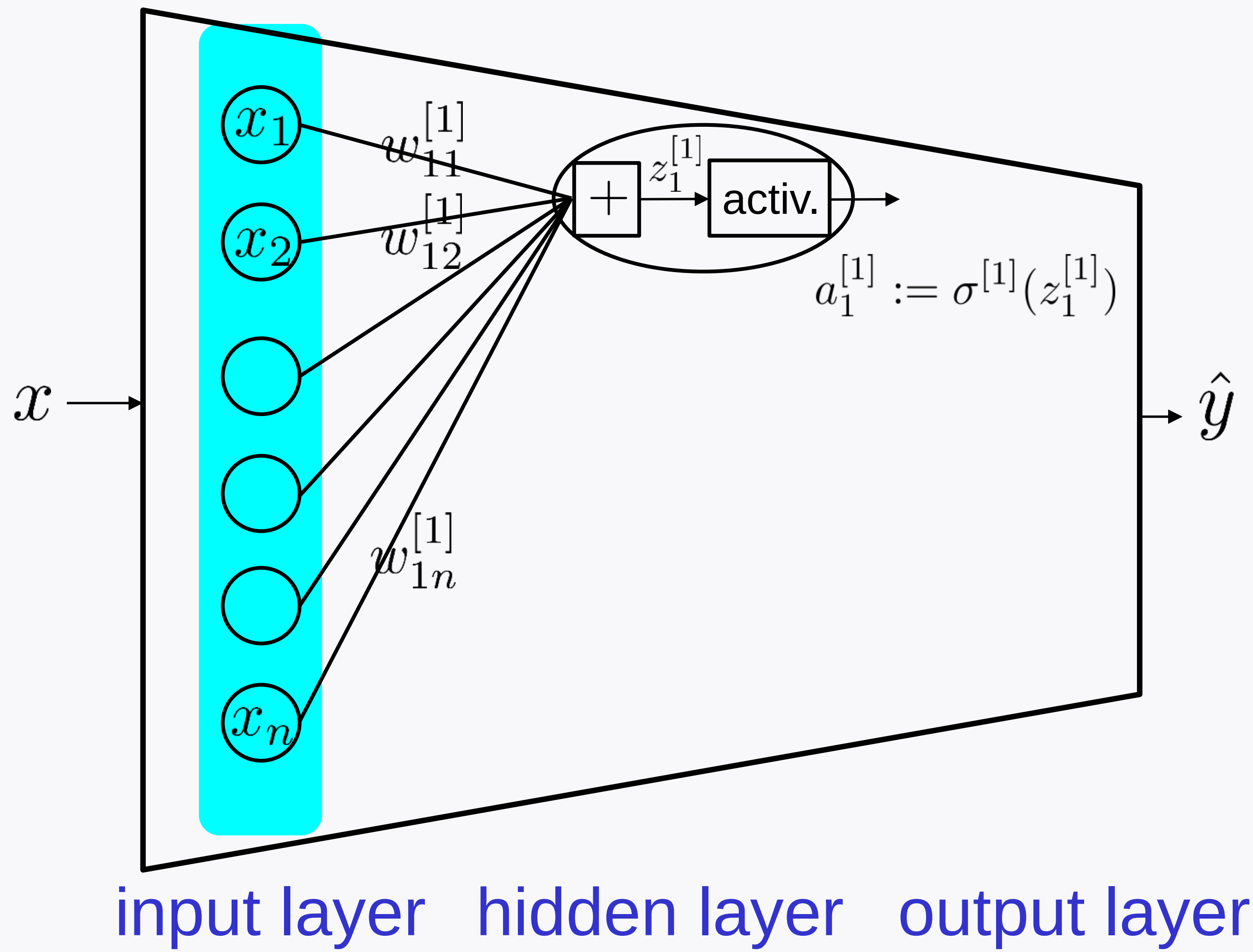
$\underbrace{\textbf{one layer}}_{\text{hidden layer}}$

Definition: A deep neural network is a network that contains hidden layer(s).

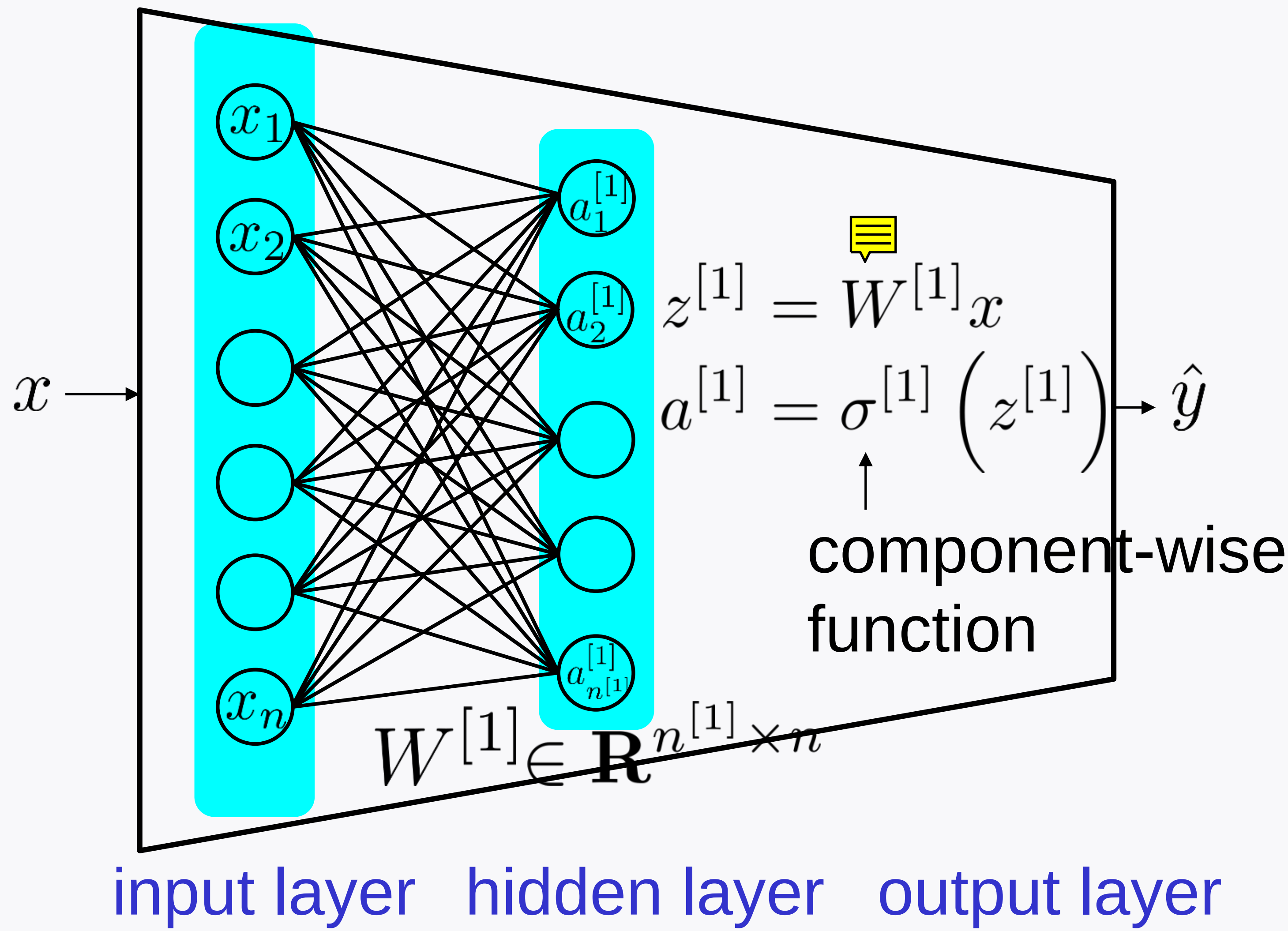
Convention: L-hidden-layer network

= (L+1)-layer network

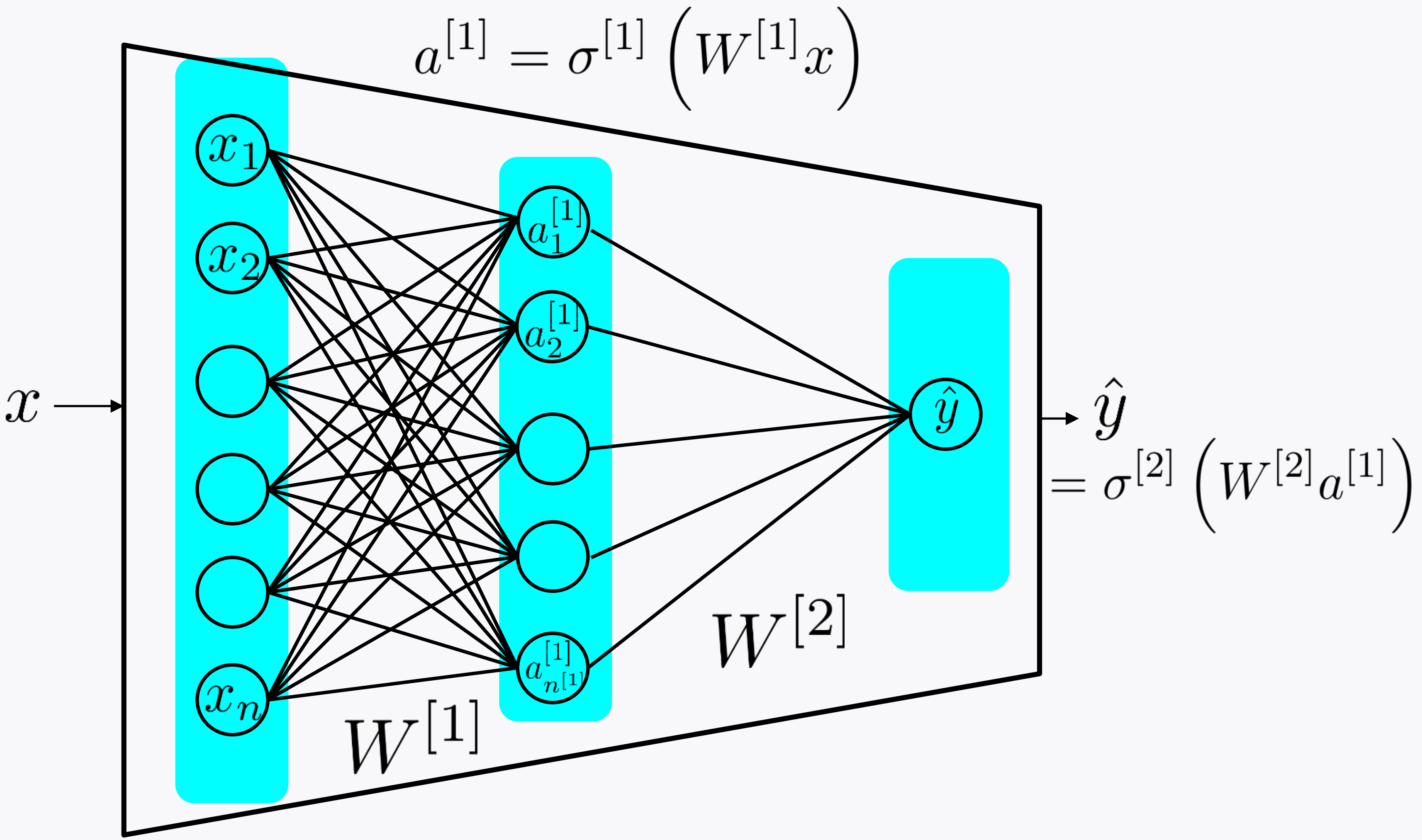
DNN architecture



DNN architecture



DNN architecture



input layer hidden layer output layer

DNN architecture: *L* hidden layers

$$\in \mathbf{R}^{n^{[1]}\times n}$$

$$a^{[1]} = \sigma^{[1]}\left(W^{[1]}x\right) \quad \in \mathbf{R}^{n^{[1]}}$$

$$\in \mathbf{R}^{n^{[2]}\times n^{[1]}}$$

$$a^{[2]} = \sigma^{[2]}\left(W^{[2]}a^{[1]}\right) \quad \in \mathbf{R}^{n^{[2]}}$$

$$\vdots$$

$$\in \mathbf{R}^{n^{[L]}\times n^{[L-1]}}$$

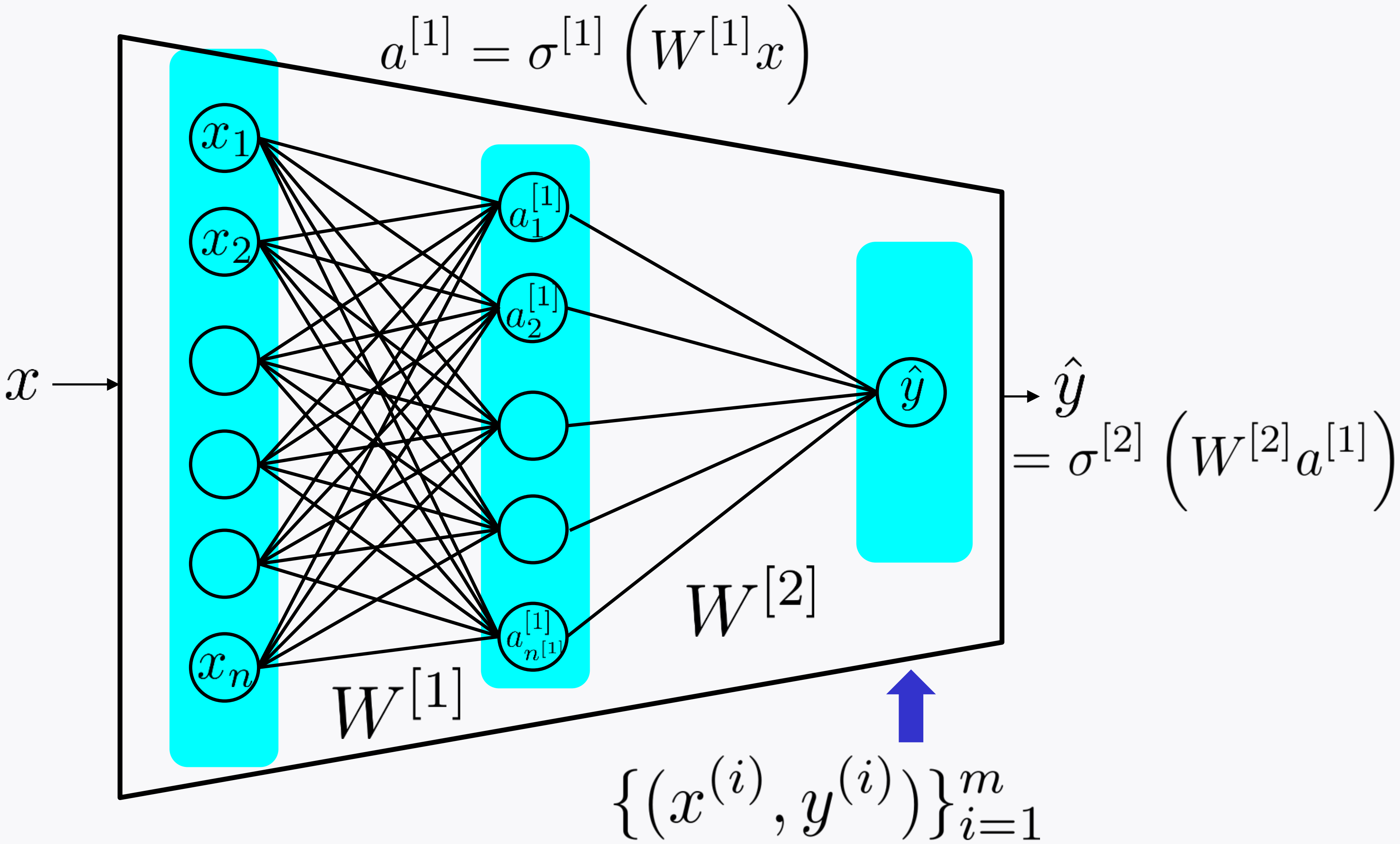
$$a^{[L]} = \sigma^{[L]}\left(W^{[L]}a^{[L-1]}\right) \in \mathbf{R}^{n^{[L]}}$$

$$\in \mathbf{R}^{1\times n^{[L]}}$$

$$\hat{y} = \sigma^{[L+1]}\left(W^{[L+1]}a^{[L]}\right) \in \mathbf{R}$$

DNN-based optimization

Optimization



Optimization

$$\min_{w} \sum_{i=1}^{m} \ell(y^{(i)}, \hat{y}^{(i)})$$

$$\hat{y}^{(i)} = \sigma^{[2]}\left(W^{[2]}a^{[1],(i)}\right)$$

$$a^{[1],(i)} = \sigma^{[1]}\left(W^{[1]}x^{(i)}\right)$$

$$w = (W^{[1]}, W^{[2]})$$

How to choose a loss function?

$$\min_w \sum_{i=1}^{m} \ell(y^{(i)}, \hat{y}^{(i)})$$

$$\underline{\hat{y}^{(i)} = \sigma^{[2]}} \left(W^{[2]} a^{[1],(i)} \right)$$

$$a^{[1],(i)} = \sigma^{[1]} \left(W^{[1]} x^{(i)} \right)$$

$$w = (W^{[1]}, W^{[2]})$$

Use a logistic function for

$$\sigma^{[2]}(z) = \sigma(z) := \frac{1}{1 + e^{-z}}$$

Use cross entropy loss.

Optimization

$$\min_{w} \sum_{i=1}^{m} -y^{(i)} \log \hat{y}^{(i)} - (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})$$

$$\hat{y}^{(i)} = \sigma \left(W^{[2]} a^{[1],(i)} \right)$$

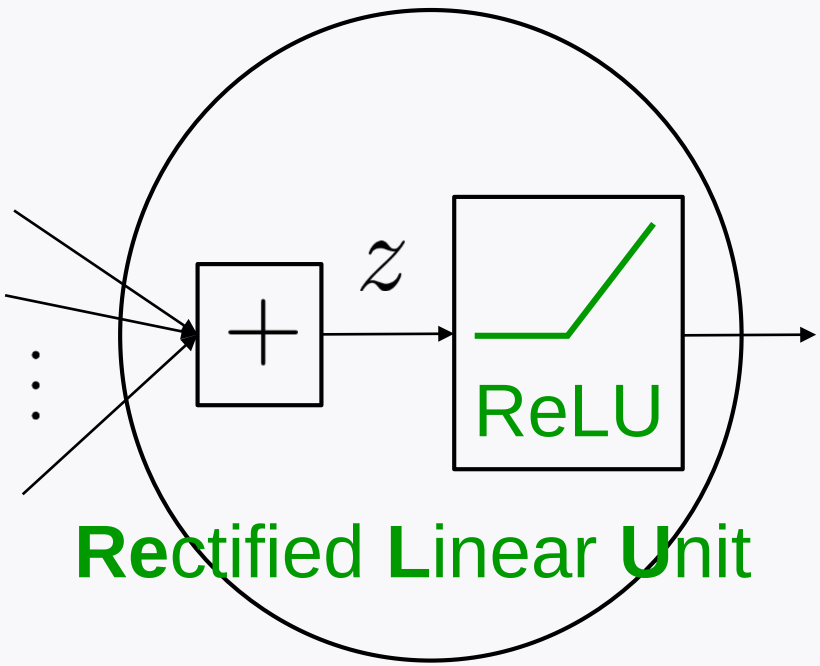
$$a^{[1],(i)} = \underline{\sigma^{[1]}} \left(W^{[1]} x^{(i)} \right)$$

How to choose $\sigma^{[1]}(\cdot)$?

$$w = (W^{[1]}, W^{[2]})$$

Widely-used activation function

Operation at a neuron in the hidden layer:



$$a = \begin{cases} z & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases}$$

$$a = \max(0, z)$$

Convex vs. non-convex?

$$\arg \min_{w=(W^{[1]},W^{[2]})} \sum_{i=1}^{m} -y^{(i)} \log \hat{y}^{(i)} - (1-y^{(i)}) \log(1-\hat{y}^{(i)})$$

$$\hat{y}^{(i)} = \sigma\left(W^{[2]} \max\left(0, W^{[1]} x^{(i)}\right)\right)$$

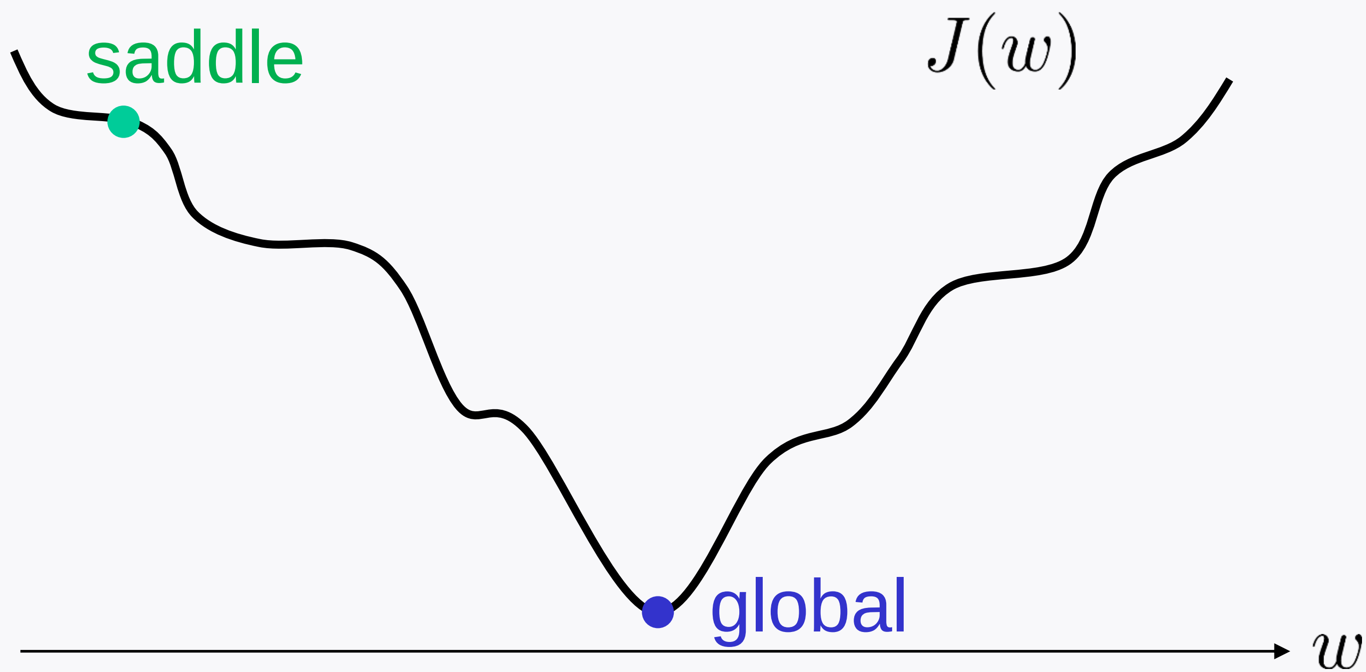
Turns out: Objective function is non-convex.

A way to handle such non-convex problem?

Observation (by many practitioners):

Experimental results revealed that in most cases:

Any local minimum is the global minimum!



Suggests a good way

Any local minimum is the global minimum!

⇩

Find *any stationary point (gradient =0)* and then take it as a solution.

Saddle points are unstable, so difficult to converge to.

How to find a stationary point then?

Via gradient descent!

Look ahead

Will explore details on gradient descent.