

Neural Networks I

Lecture 18

What's the hype around neural networks?

Character/handwriting recognition

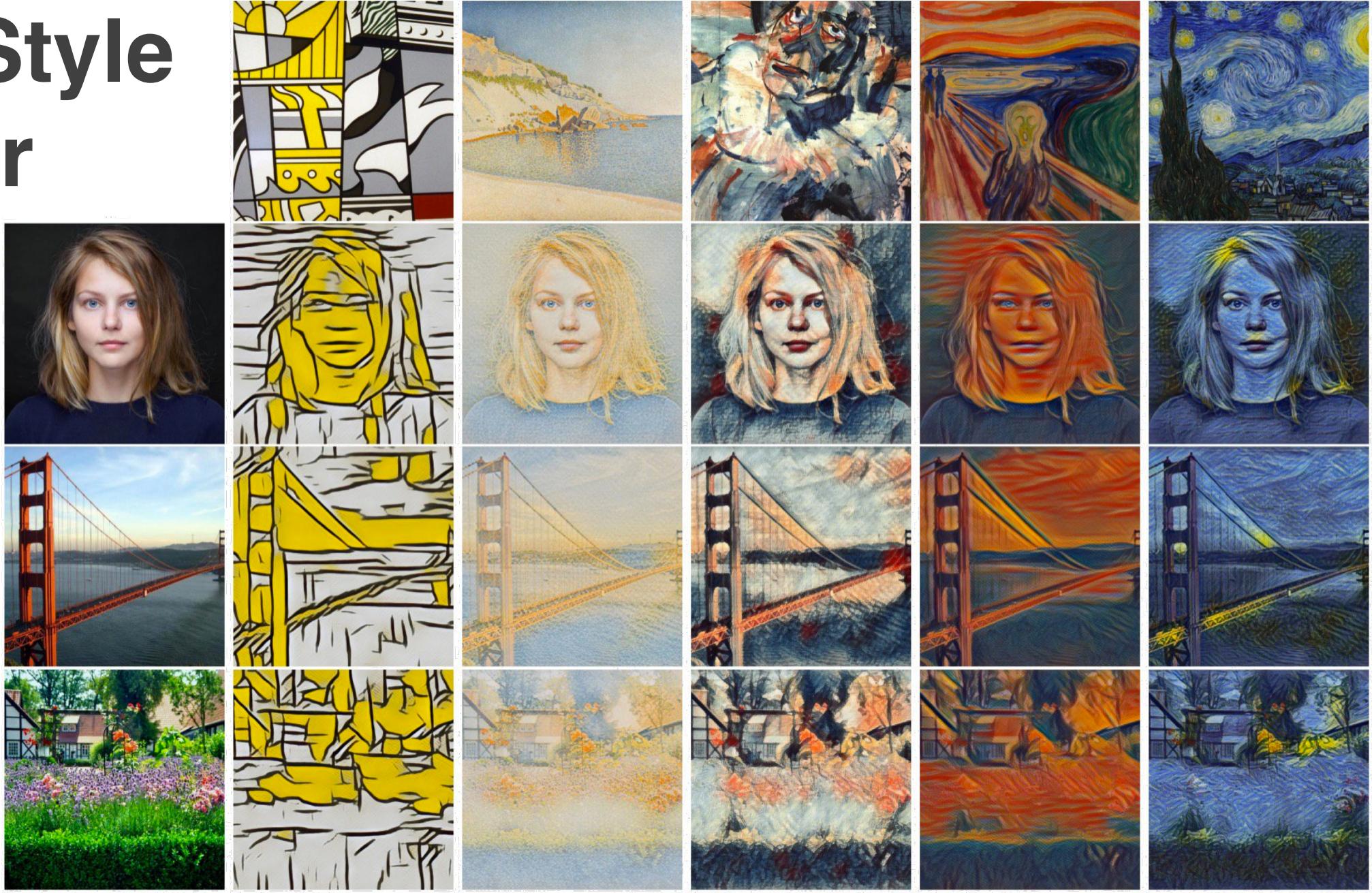
Image compression (autoencoders)

Stock market prediction

Credit approval

And some very recent interesting deep learning applications...

Image Style Transfer



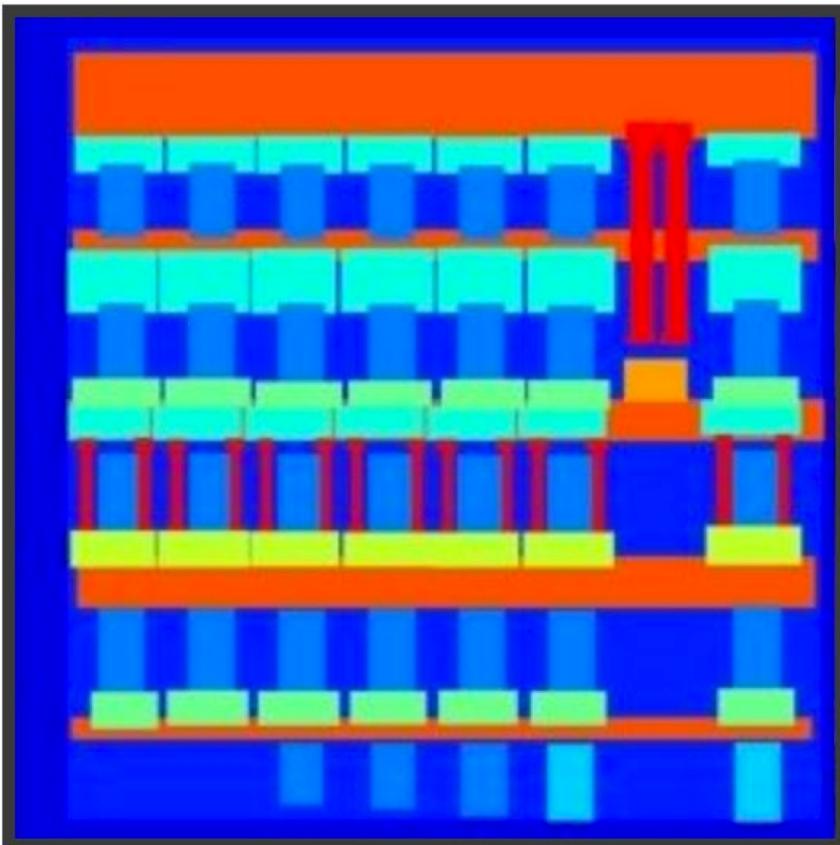
Dumoulin, Vincent, Jonathon Shlens, and Manjunath Kudlur. "A learned representation for artistic style." CoRR, abs/1610.07629 2.4 (2016): 5.

Image-to-image translation

TOOL

- background
- wall
- door
- window**
- window sill
- window head
- shutter
- balcony
- trim
- cornice
- column
- entrance

INPUT



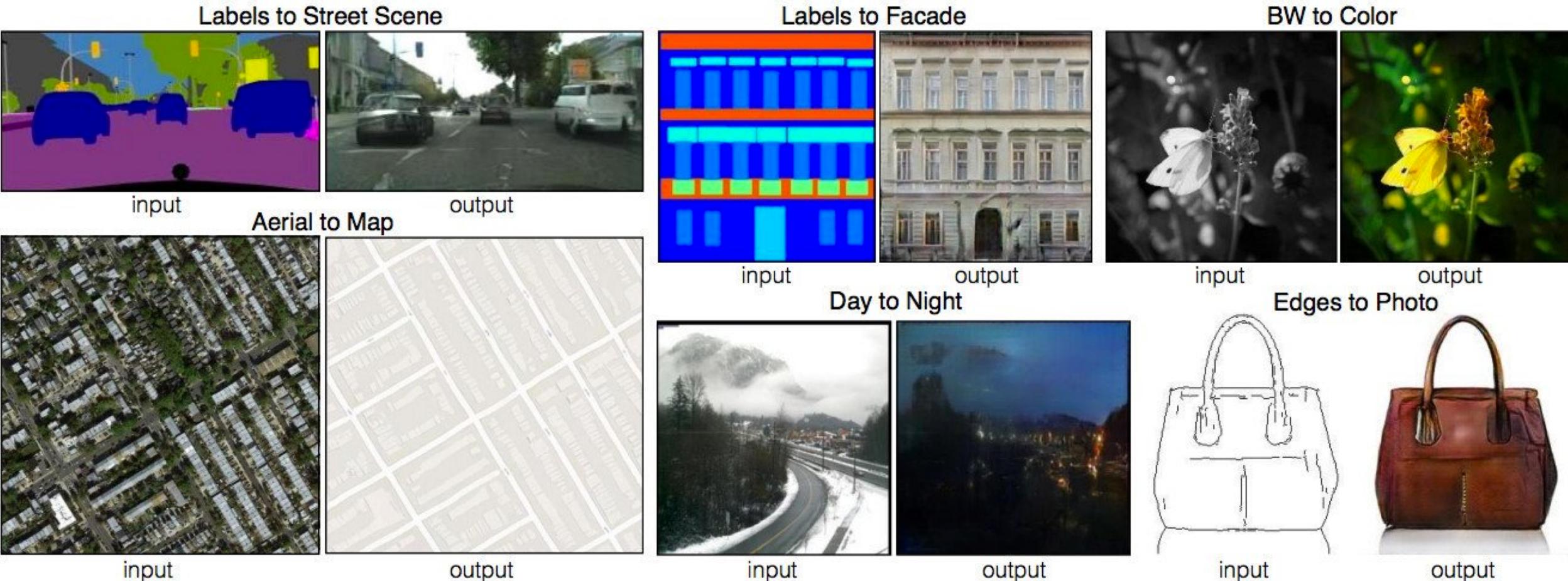
pix2pix
process

OUTPUT



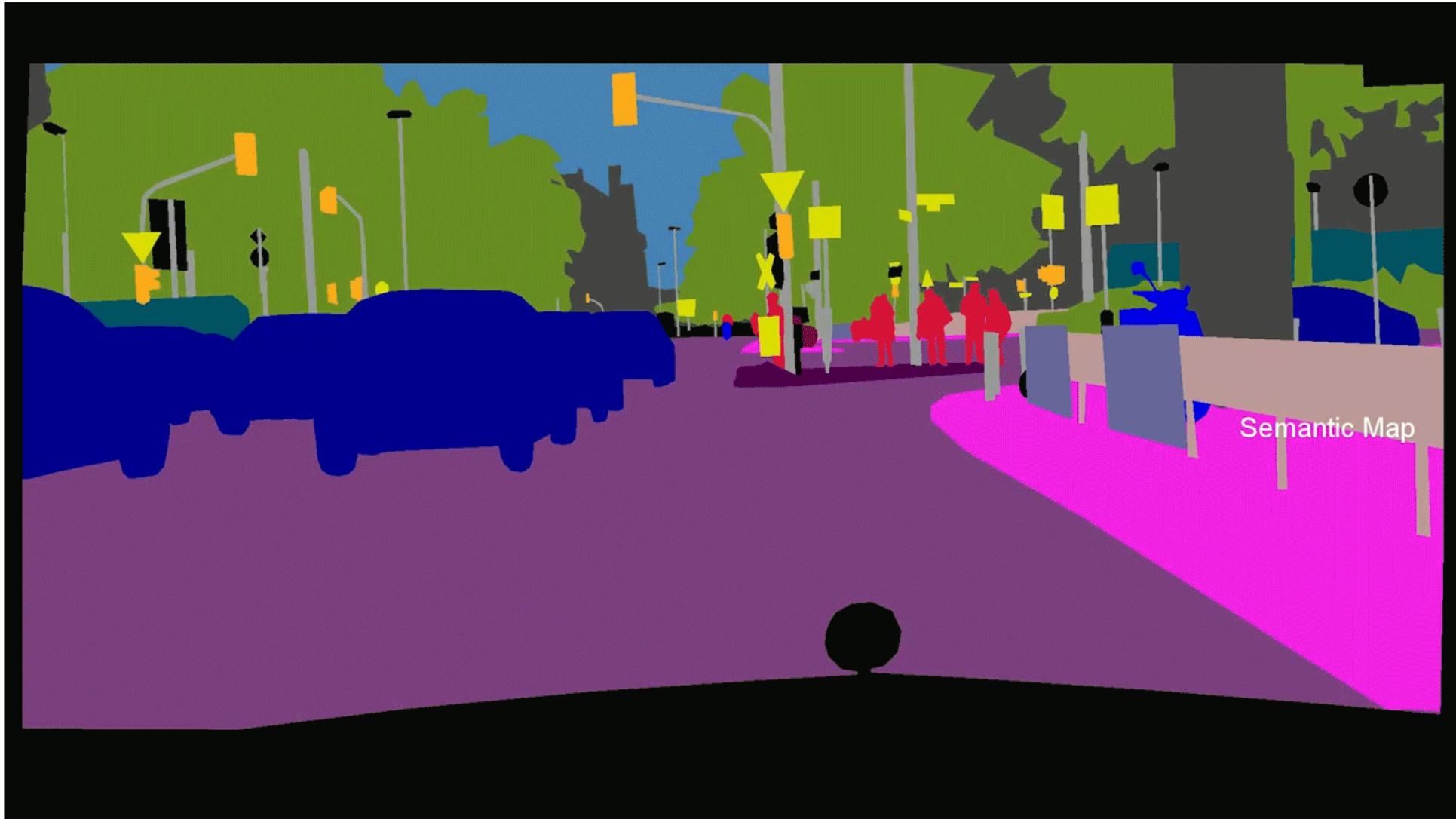
Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." arXiv preprint (2017).

Image-to-image translation



Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." arXiv preprint (2017).

Image-to-image translation



Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." arXiv preprint (2017).

Search for a location in the San Francisco metro region



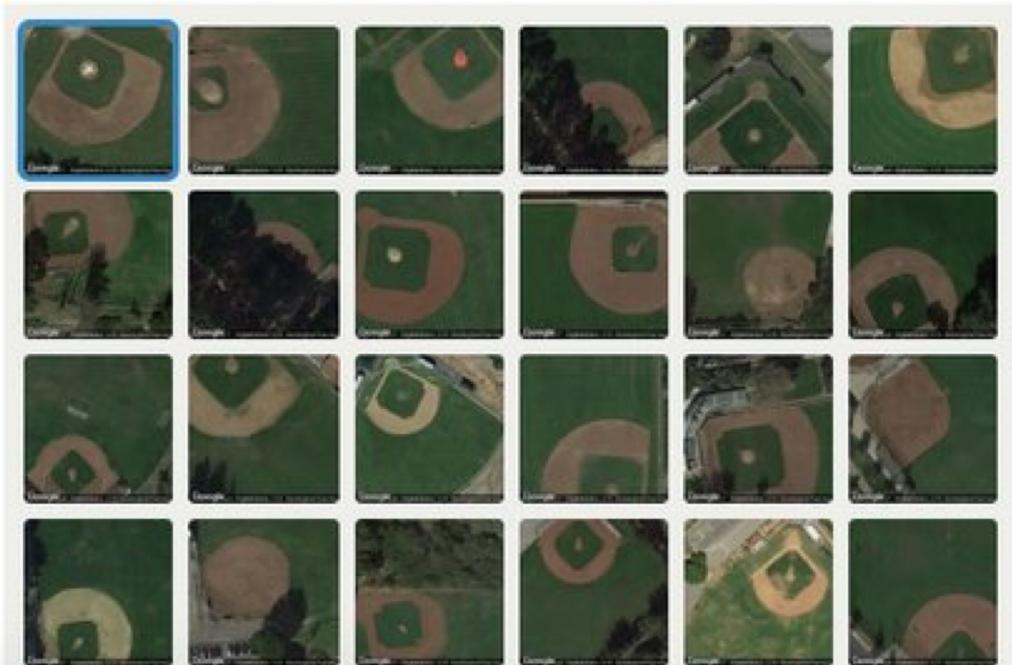
TerraPattern

<http://www.terrapattern.com/>

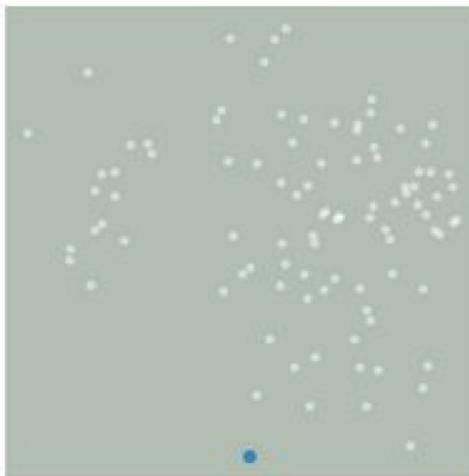
Geographical Plot



Search Results



Similarity Plot



What is a neural network and **how does it work?**

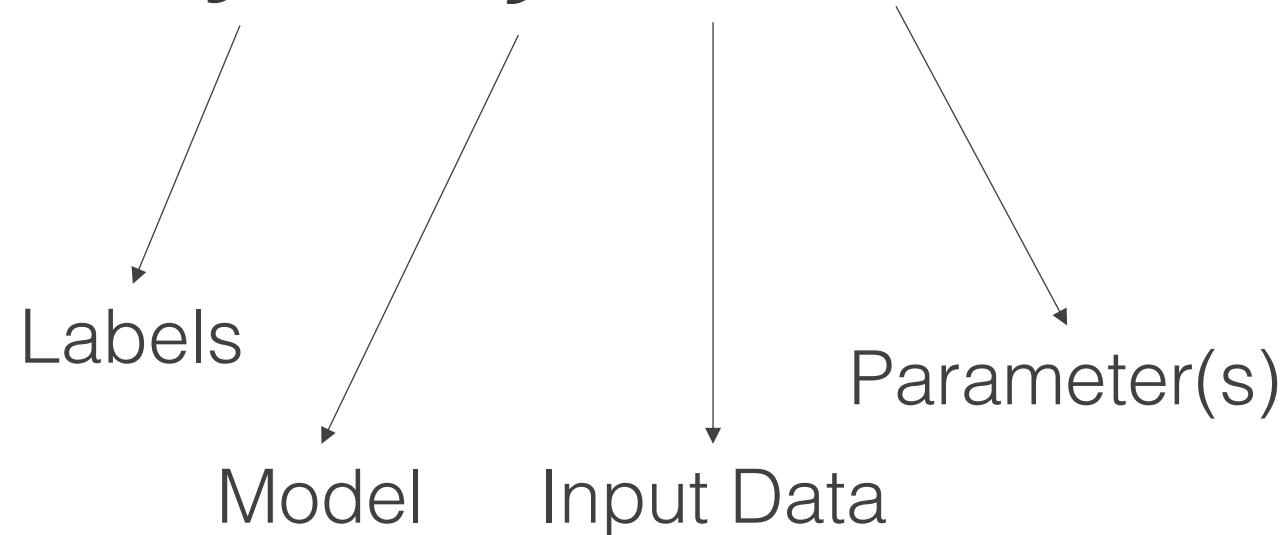
How do we **choose model weights**?
(i.e. how do we fit our model to data)

What are the challenges of using neural networks?

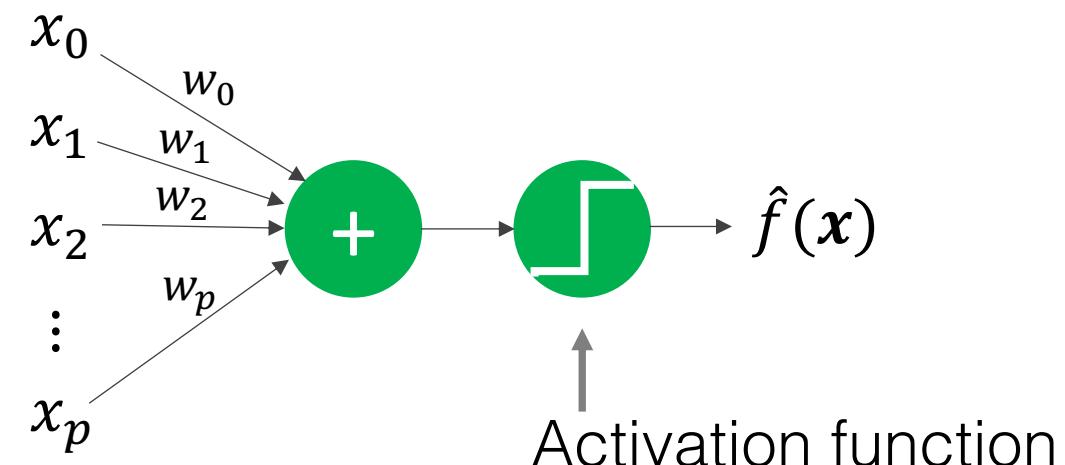
Recall our goal in supervised learning

Perceptron

$$y = f(\mathbf{x}, \mathbf{w})$$



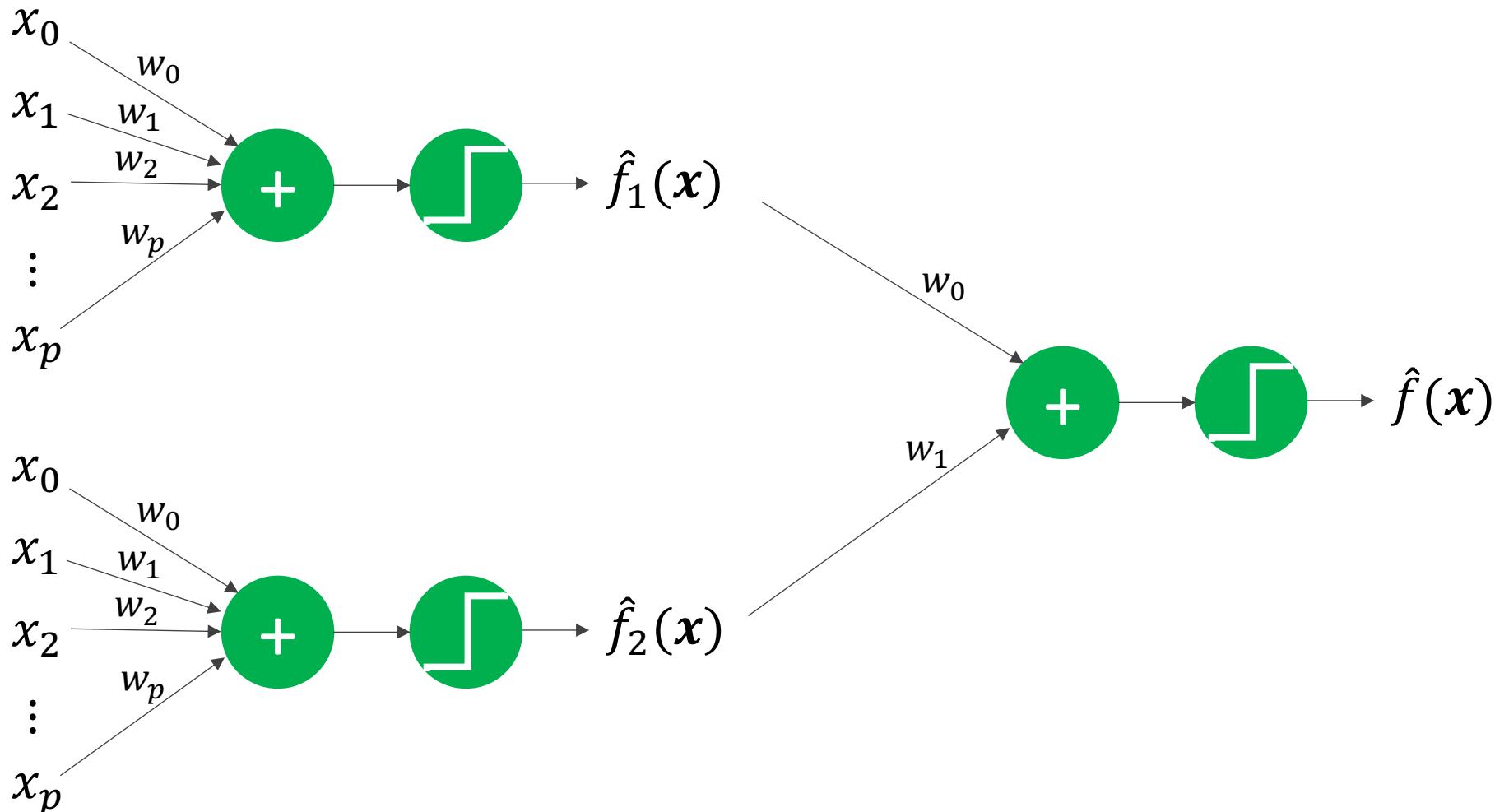
$$\hat{f}(\mathbf{x}) = \text{sign} \left(\sum_{i=0}^N w_i x_i \right)$$



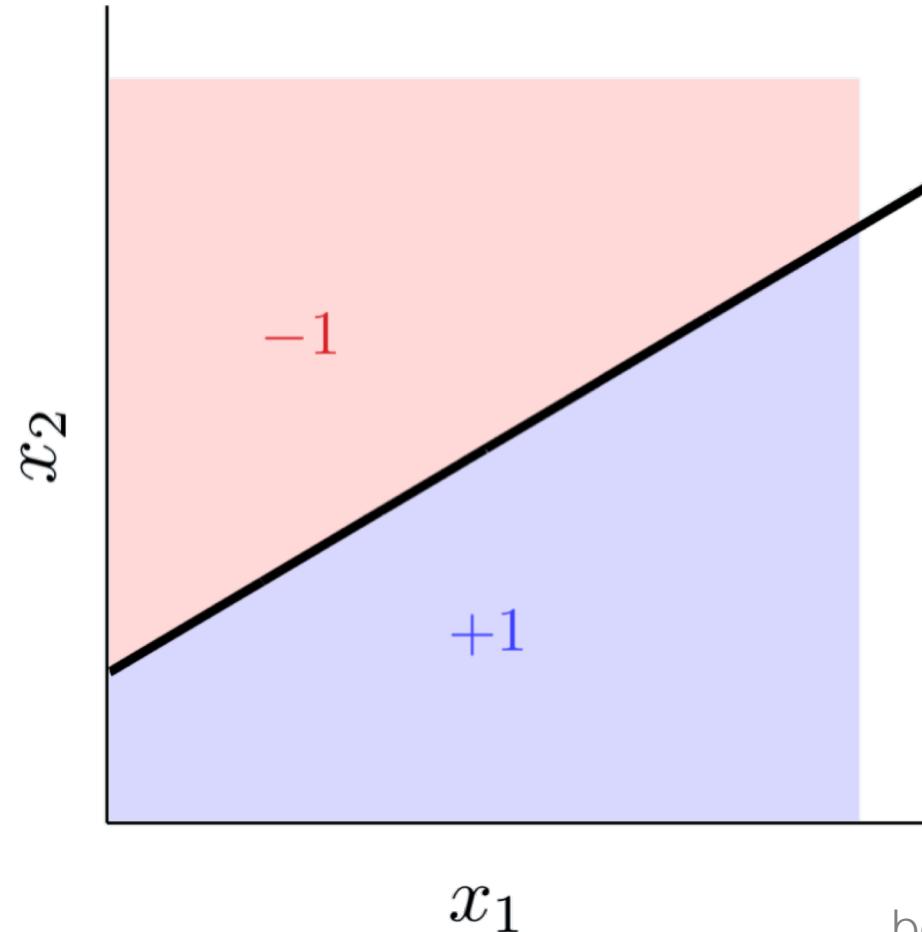
Source: Abu-Mostafa, Learning from Data, Caltech

Multilayer Perceptron

What if we stuck multiple perceptrons together?



Perceptron #1

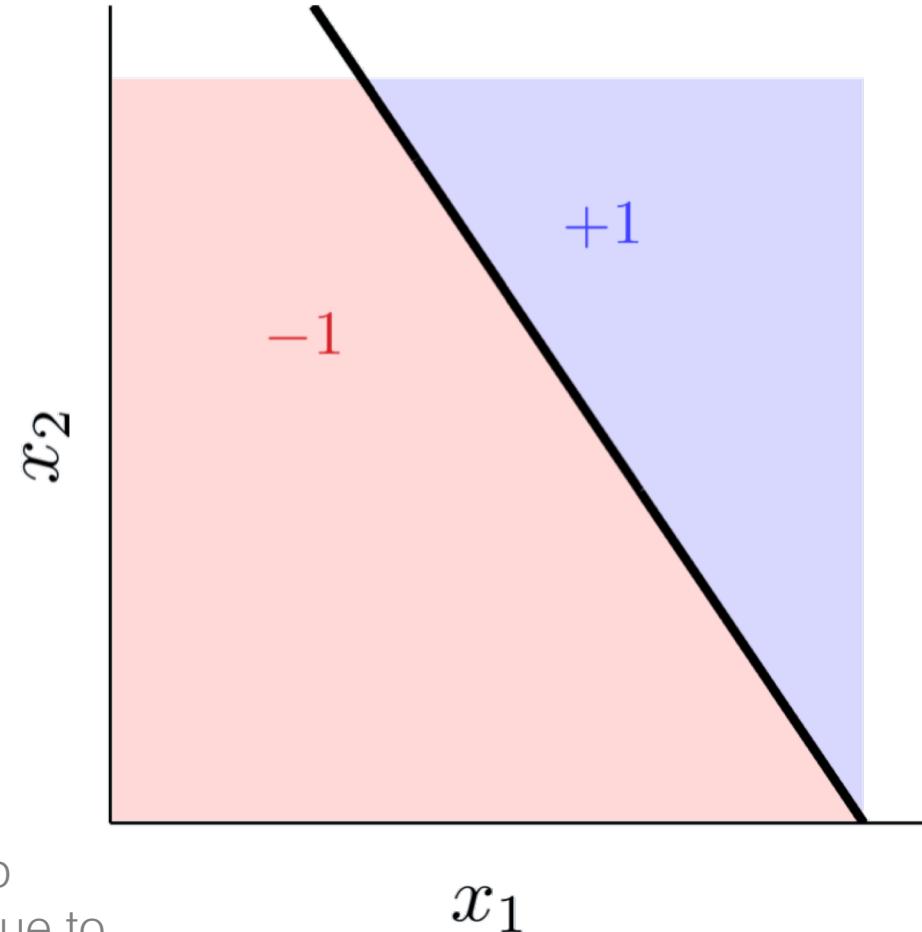


$$\hat{f}_1(\mathbf{x}) = \text{sign}(\mathbf{w}_1^T \mathbf{x})$$

The sharp
boundary is due to
our sign function



Perceptron #2

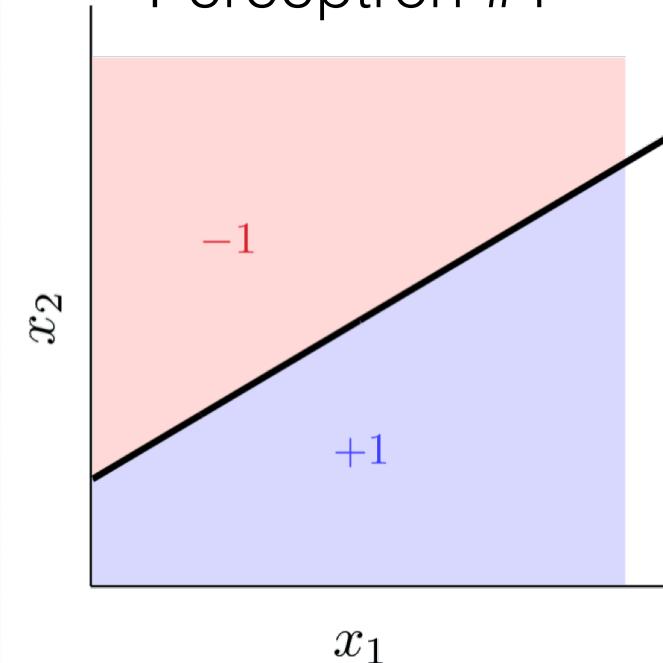


$$\hat{f}_2(\mathbf{x}) = \text{sign}(\mathbf{w}_2^T \mathbf{x})$$

Source: Abu-Mostafa, Learning from Data, Caltech

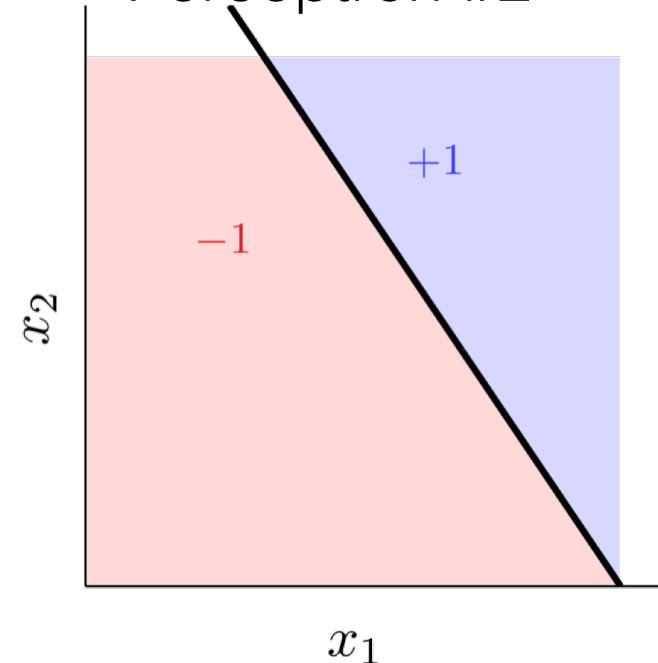
Multilayer perceptron: $\hat{f}(x) = \begin{cases} +1 & \hat{f}_1(-\hat{f}_2) + (-\hat{f}_1)\hat{f}_2 > 0 \\ -1 & \text{else} \end{cases}$

Perceptron #1

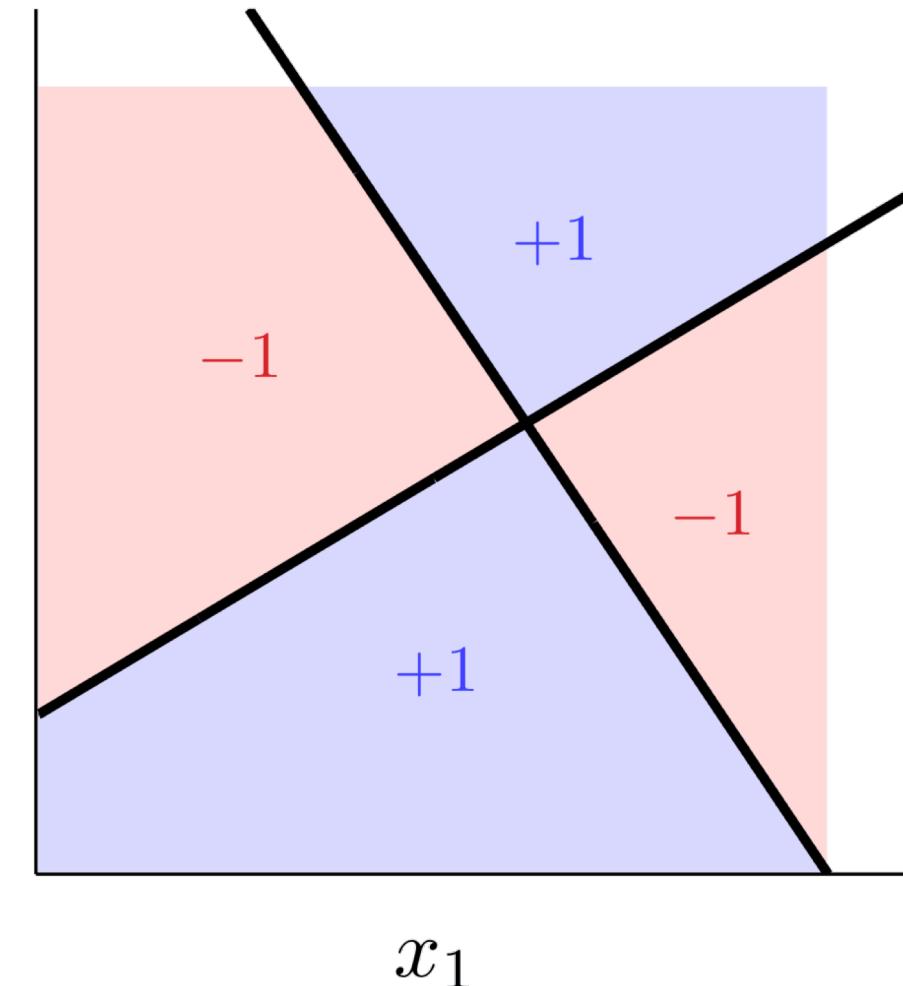


$$\hat{f}_1(x) = \text{sign}(\mathbf{w}_1^T \mathbf{x})$$

Perceptron #2

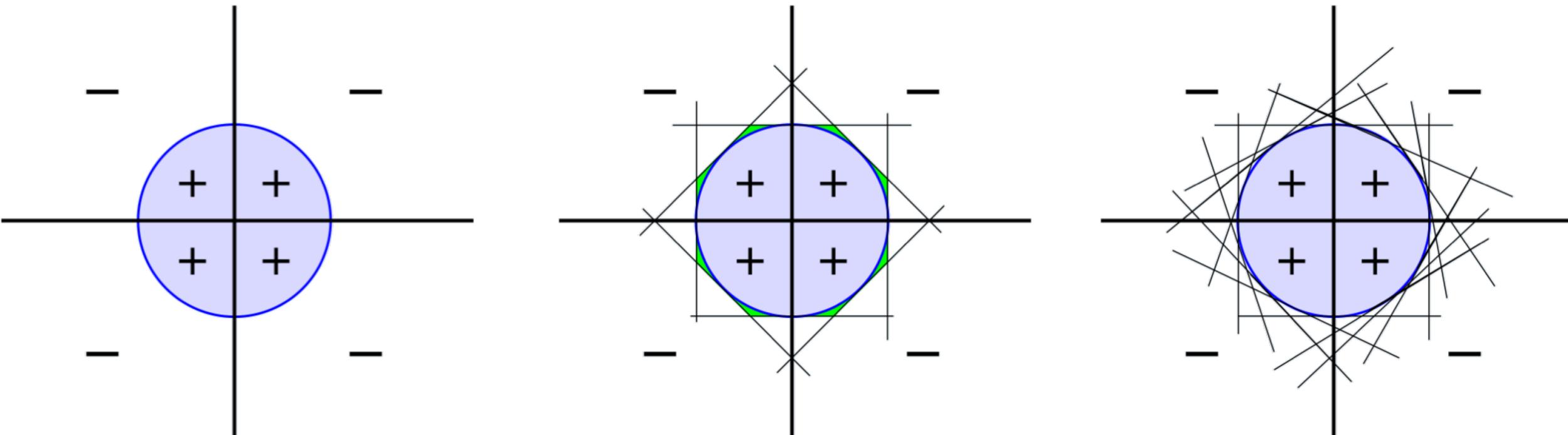


$$\hat{f}_2(x) = \text{sign}(\mathbf{w}_2^T \mathbf{x})$$



Source: Abu-Mostafa, Learning from Data, Caltech

Multilayer Perceptron



Target

8 perceptrons

16 perceptrons

The more nodes/neurons, the more flexible is the model

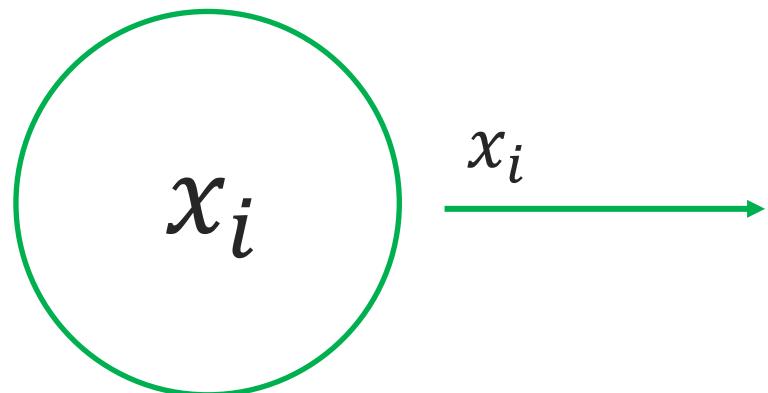
Source: Abu-Mostafa, Learning from Data, Caltech

Universal function approximation

“A **feedforward network** with a single layer is sufficient to represent **any function**, but the layer may be infeasibly large and may fail to learn and generalize correctly.”

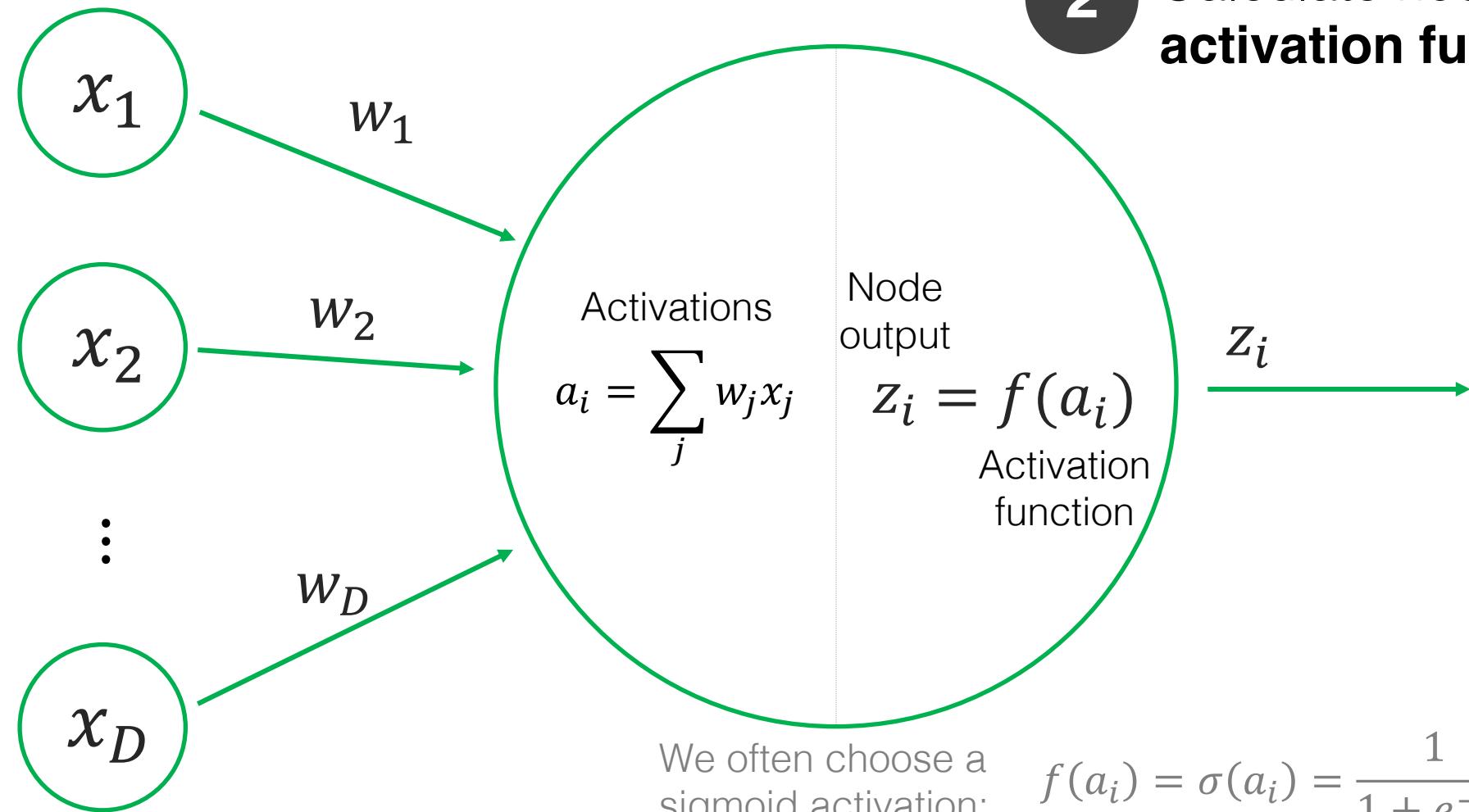
Ian Goodfellow, Deep Learning
Creator of generative adversarial networks

Input nodes / neurons



Simply passes the input value to the next layer

Hidden & output nodes



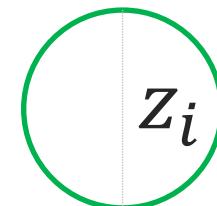
1

Calculate the **activations**: linear combinations of weights and the last layer's output

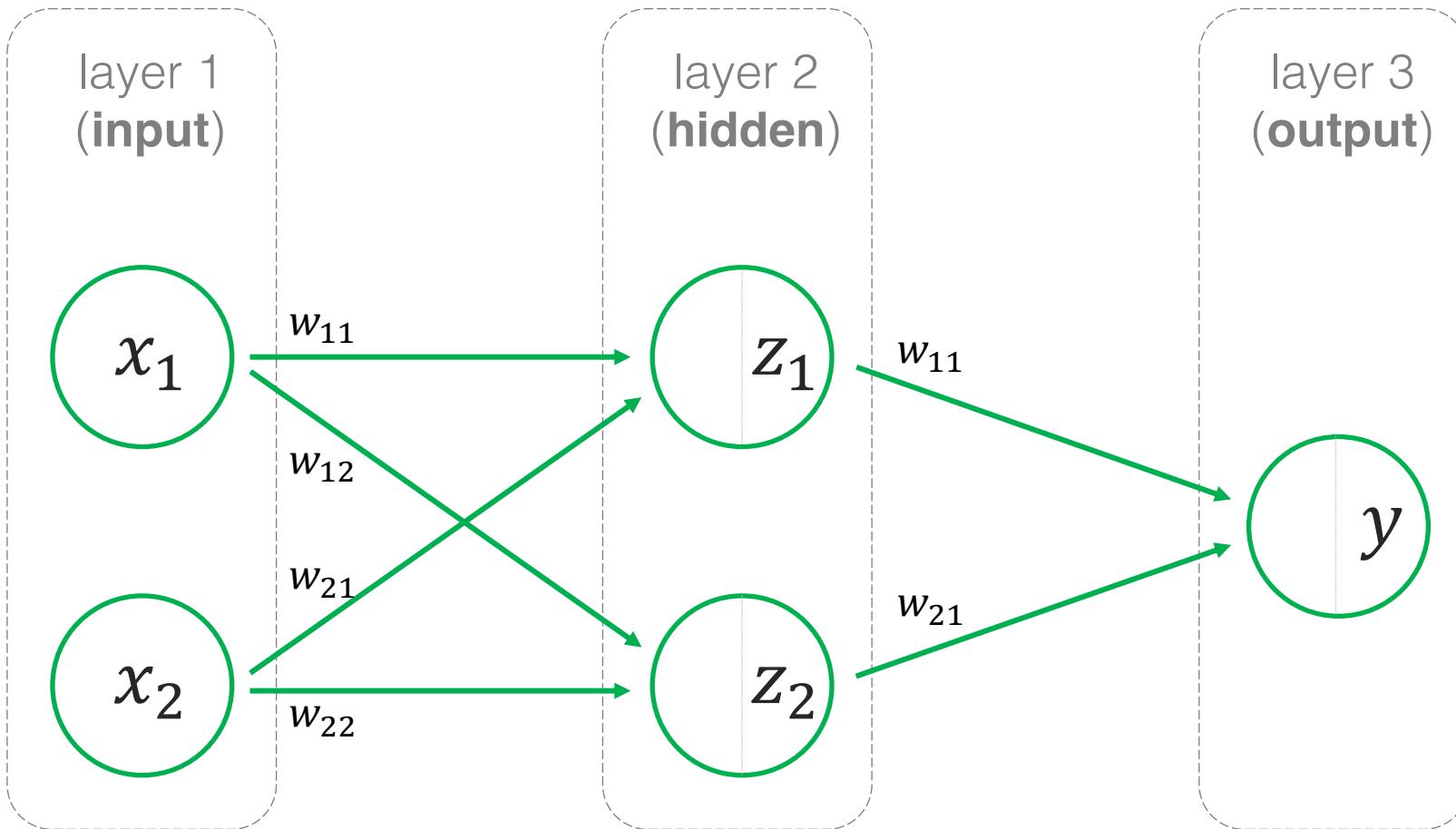
2

Calculate node output: apply the **activation function** to the activations

Represented as:



Simple Neural Network



Notational shorthand:
(a more precise
alternative notation)

$$w_{ij} = w_{ij}^{(1)}$$

$$z_i = z_i^{(1)}$$

$$w_{ij} = w_{ij}^{(2)}$$

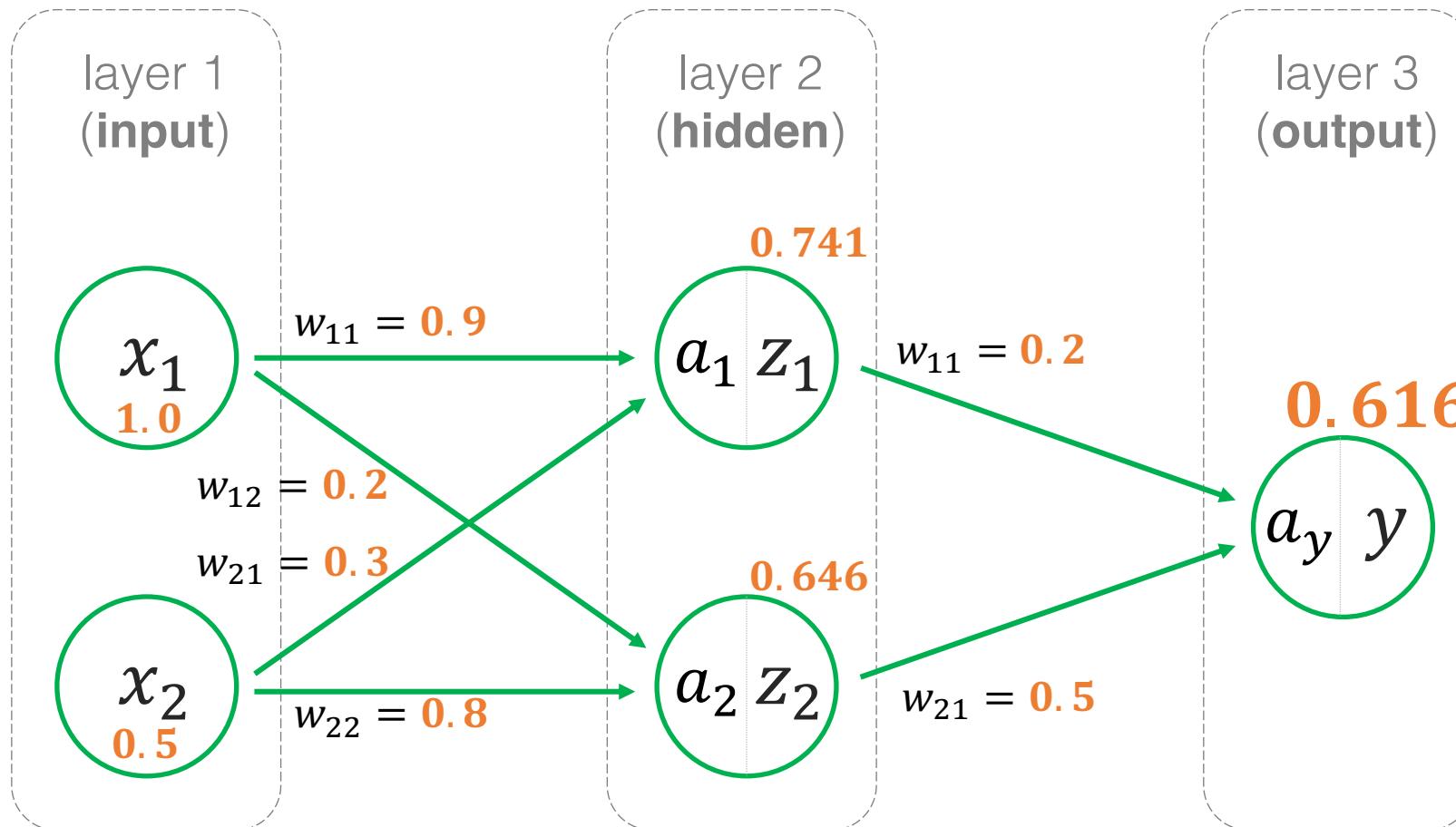
$$y = z_1^{(2)}$$

w_{ij}
From node i
(in the last layer)
to node j
(in the next layer)

Forward Propagation

Calculating the output from input

$$a_1 = (0.9)(1.0) + (0.3)(0.5) = 1.05 \quad \text{Hidden layer calculations}$$
$$a_2 = (0.2)(1.0) + (0.8)(0.5) = 0.6$$
$$z_1 = \sigma(a_1) = \sigma(1.05) = 0.741$$
$$z_2 = \sigma(a_2) = \sigma(0.6) = 0.646$$



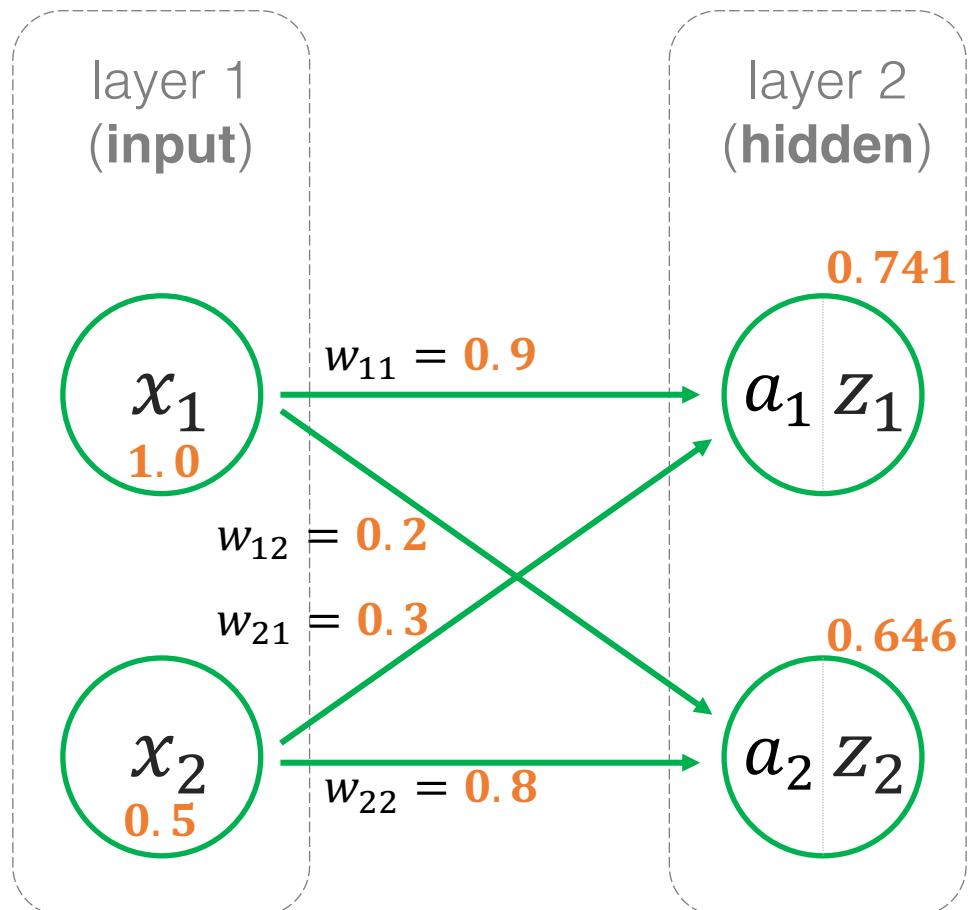
Output layer calculations

$$a_y = (0.2)(0.741) + (0.5)(0.646) = 0.471$$
$$y = \sigma(a_y) = \sigma(0.471) = 0.616$$

$$\sigma(a_i) = \frac{1}{1 + e^{-a_i}}$$

Forward Propagation

Calculating the output from input



Hidden layer matrix calculations

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad \mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \quad \mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

$$\mathbf{W} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix}$$

→ The weights out of node 1
→ The weights out of node 2

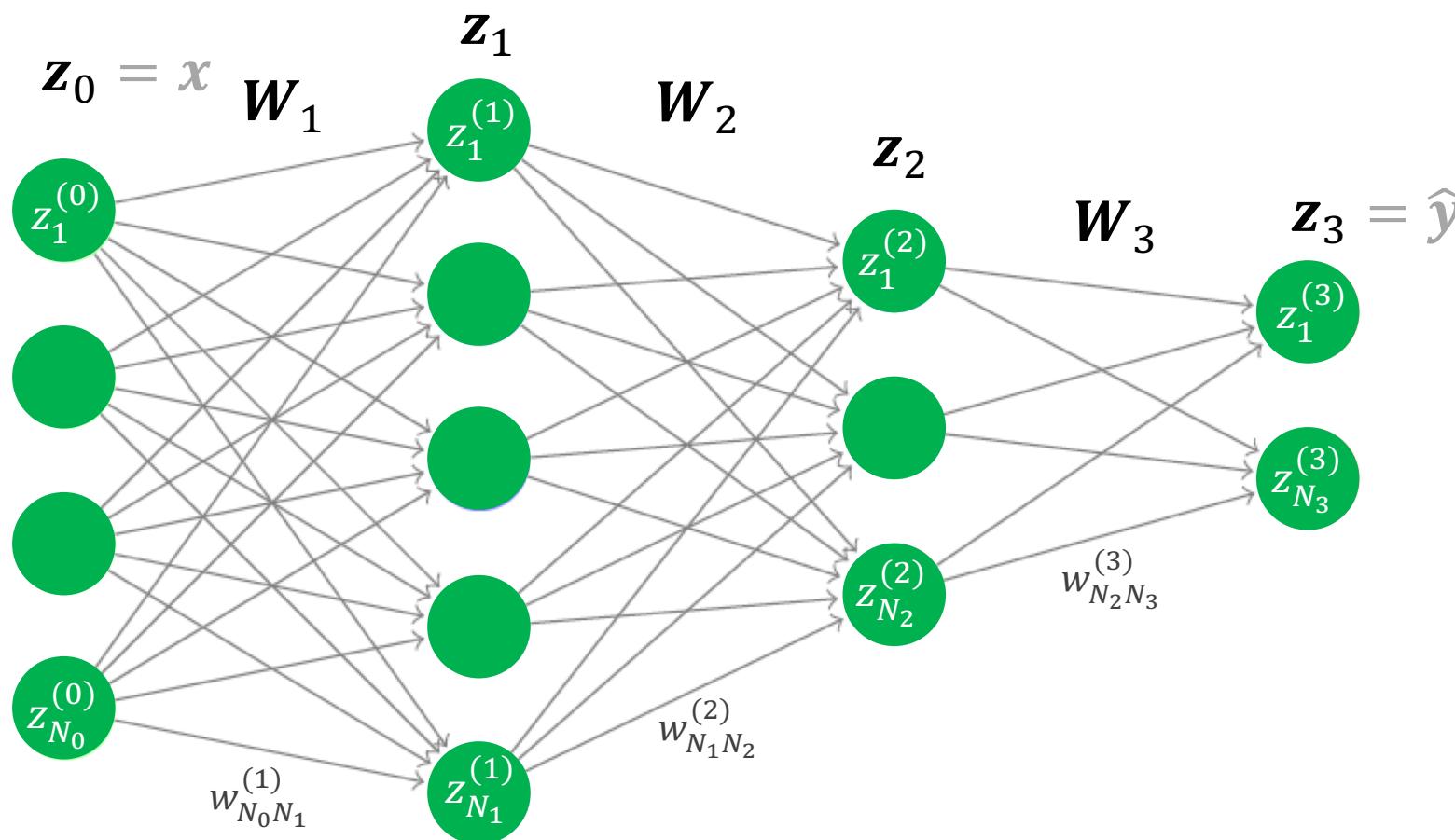
$$\mathbf{a} = \mathbf{W}^T \mathbf{x} = \begin{bmatrix} w_{11} & w_{21} \\ w_{12} & w_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$= \begin{bmatrix} w_{11}x_1 + w_{21}x_2 \\ w_{12}x_1 + w_{22}x_2 \end{bmatrix}$$

$$\mathbf{z} = \sigma(\mathbf{a}) = \begin{bmatrix} \sigma(w_{11}x_1 + w_{21}x_2) \\ \sigma(w_{12}x_1 + w_{22}x_2) \end{bmatrix}$$

Forward Propagation

Example neural network with $L = 3$ layers and the i th layer has N_i nodes



Simple steps for forward propagation:

For $i = 1$ to L :

$$\mathbf{z}_i = \sigma(\mathbf{W}_i^T \mathbf{z}_{i-1})$$

Where:

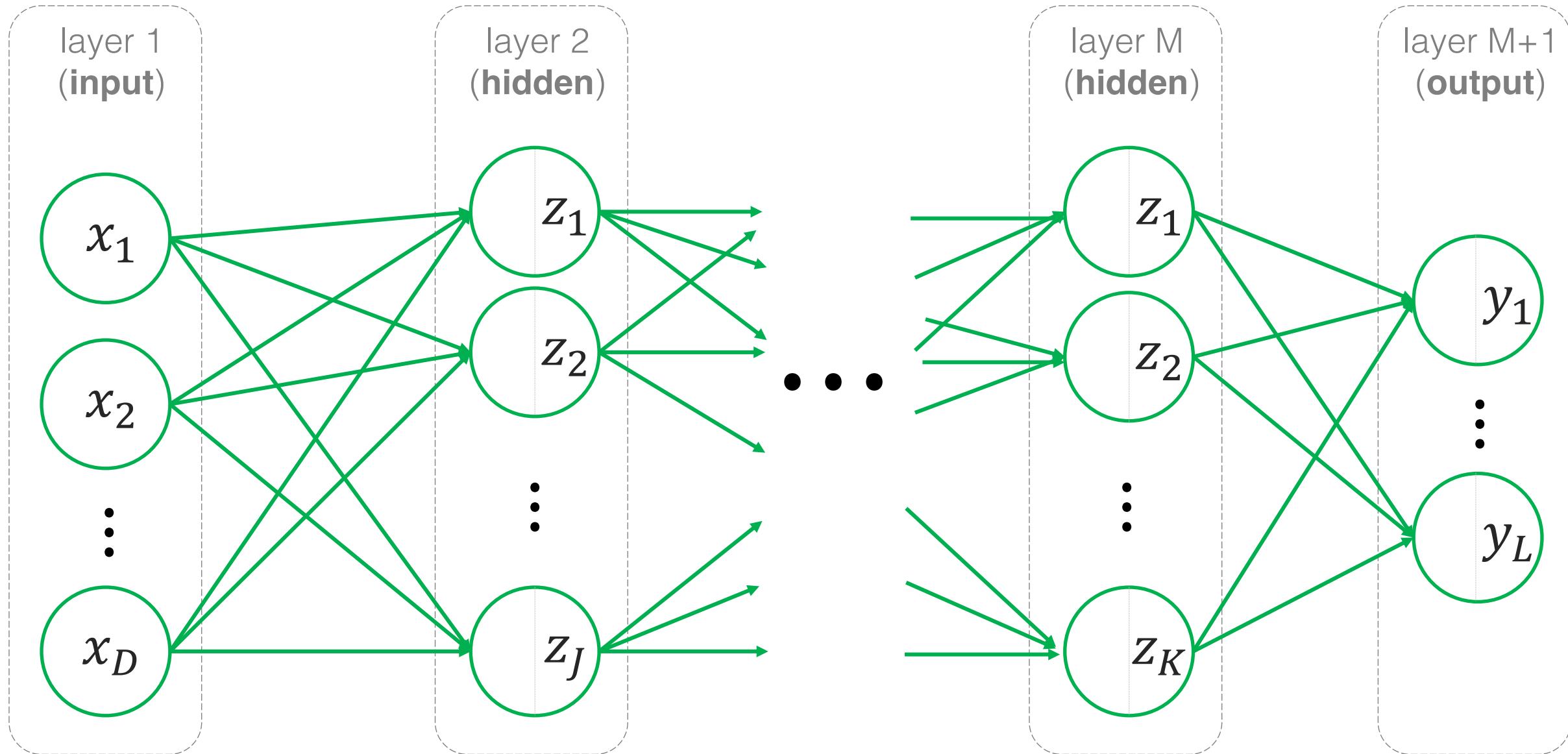
$$\mathbf{z}_0 = \mathbf{x}$$

$$\hat{\mathbf{y}} = \mathbf{z}_L$$

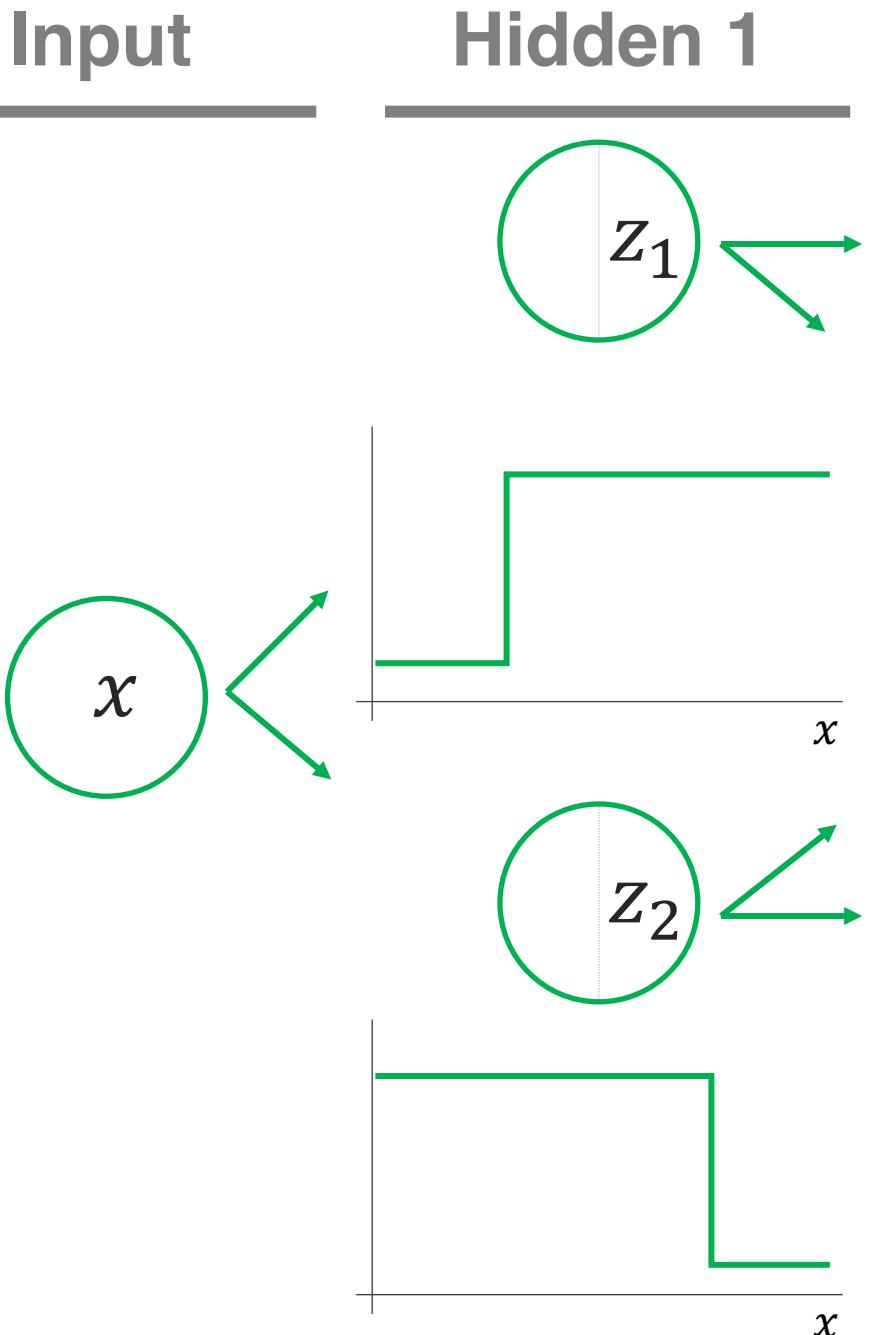
Prediction error is measured:

$$E = \frac{1}{2} \sum_k (\hat{y}_k - y_k)^2$$

Neural networks can be customized



Input



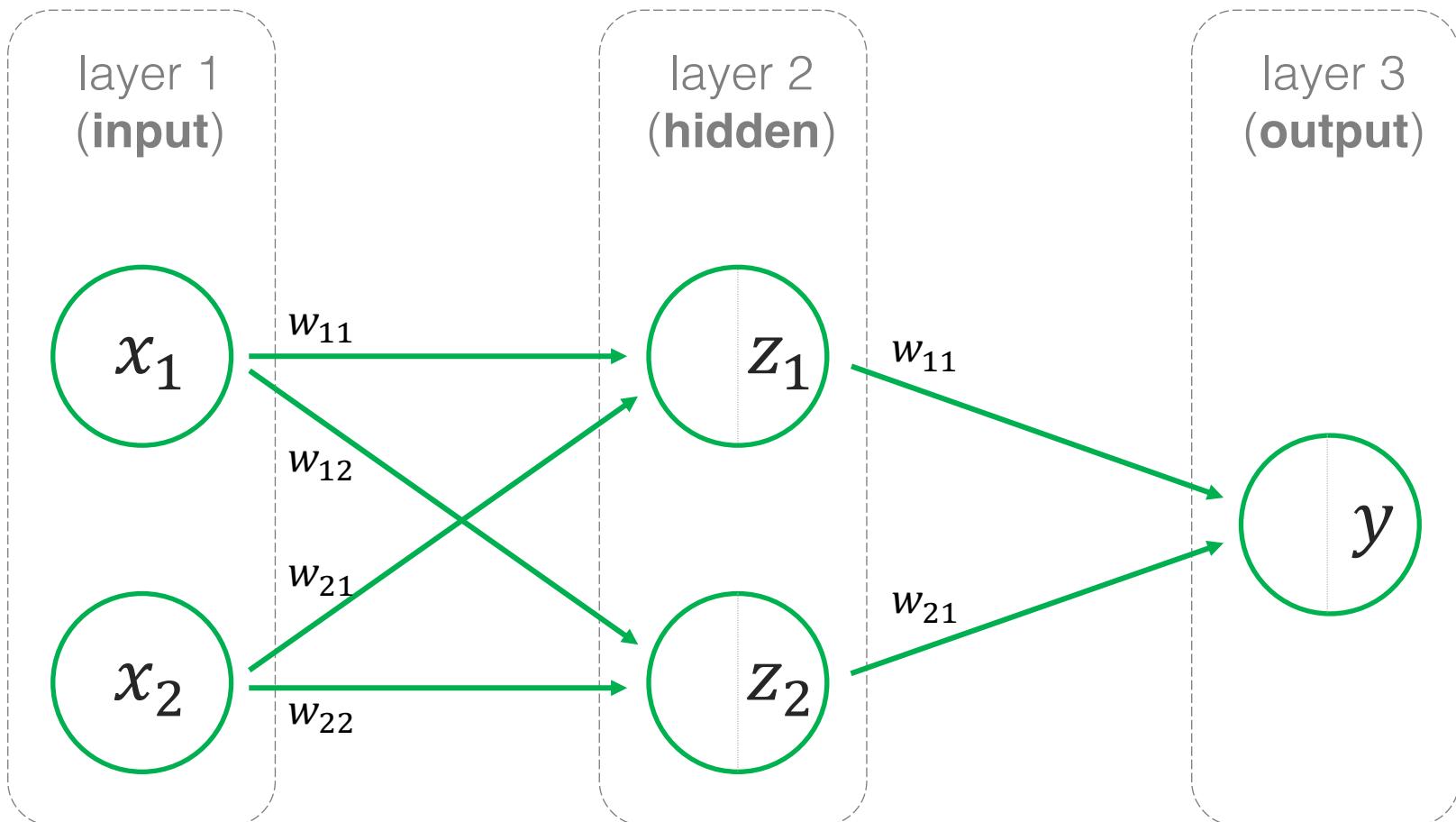
Hidden 1

Hidden 2

Output

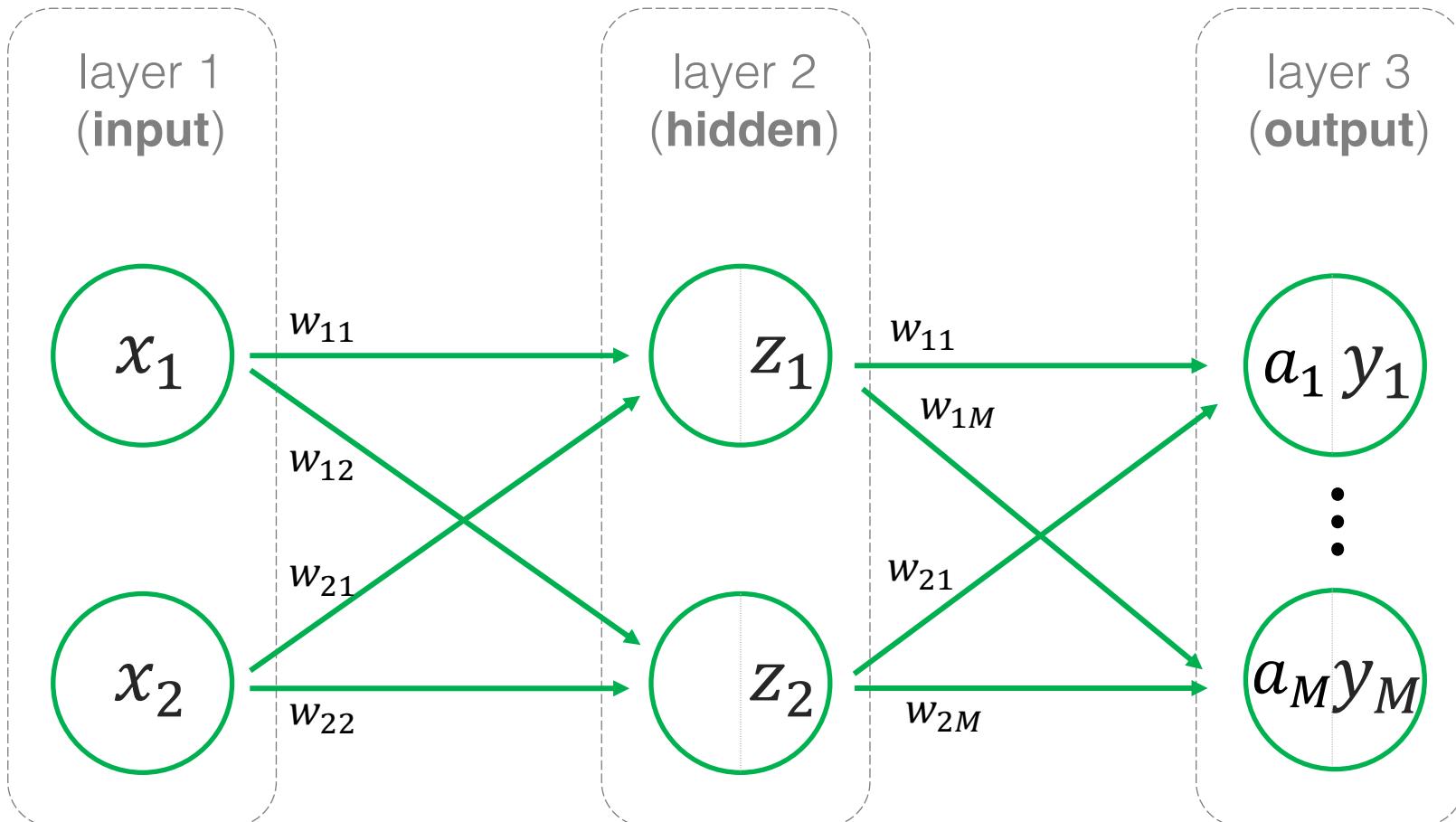
Multilayer neural nets can build up from basic building blocks to more complex structures

From regression to classification



For **binary classification** with a sigmoid activation function, the output is between zero and one, so threshold this value to assign the class

From regression to classification



For **multiclass problems**, we can have multiple outputs and use a softmax function:

$$y_i = g(a_i) = \frac{e^{a_i}}{\sum_{n=1}^M e^{a_n}}$$

Choose the largest y value as the predicted class

As with many aspects of neural networks this is one of a number of approaches

Next time....

What is a neural network and **how does it work?**

How do we **choose model weights?**
(i.e. how do we fit our model to data)

What are the challenges of using neural networks?