Opening Thought Question 3/20

1. What is CCA and how does it contribute to coral reef ecosystems?

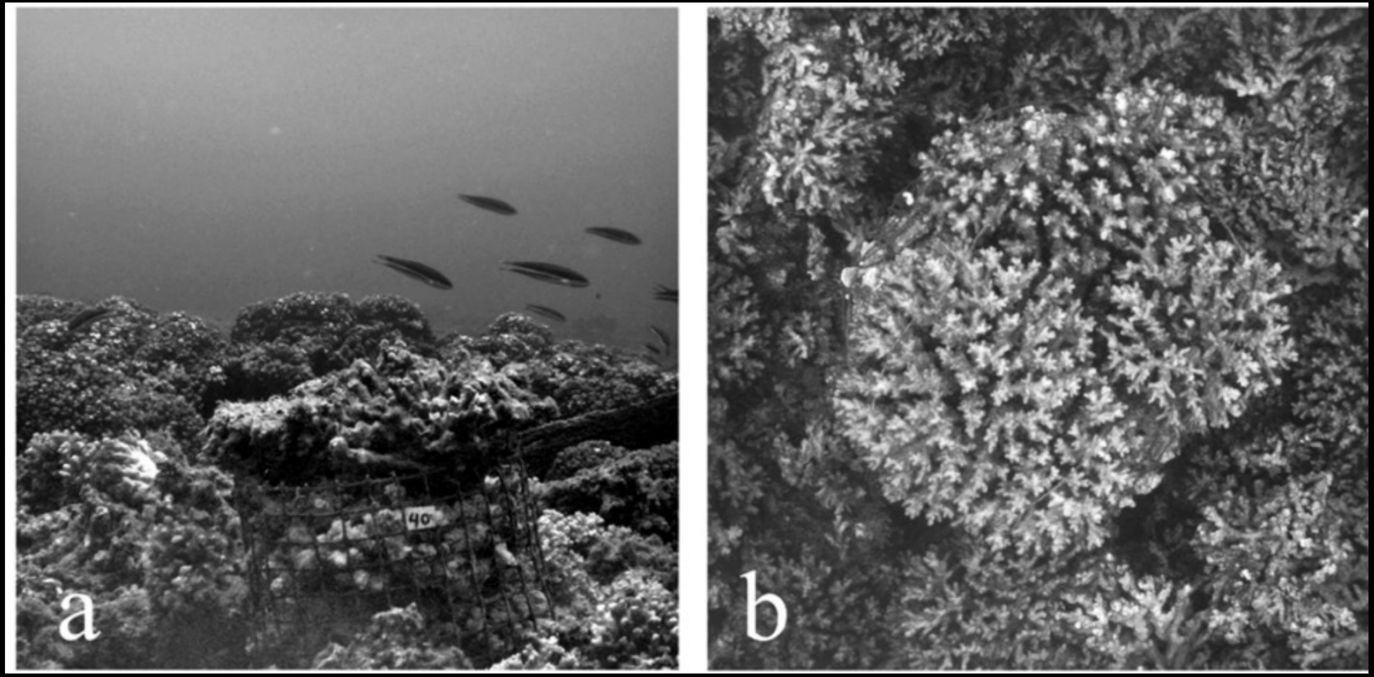
2. Anna told us that the distribution of CCA changes with light and depth. What were the patterns that she described and how do these patterns relate to the broad ecological theory of zonation?

3. How can bioinformatics help Anna to analyze her data?

Kramer et al., 2014 Mar. Ecol. Prog. Ser.

Rubble bags

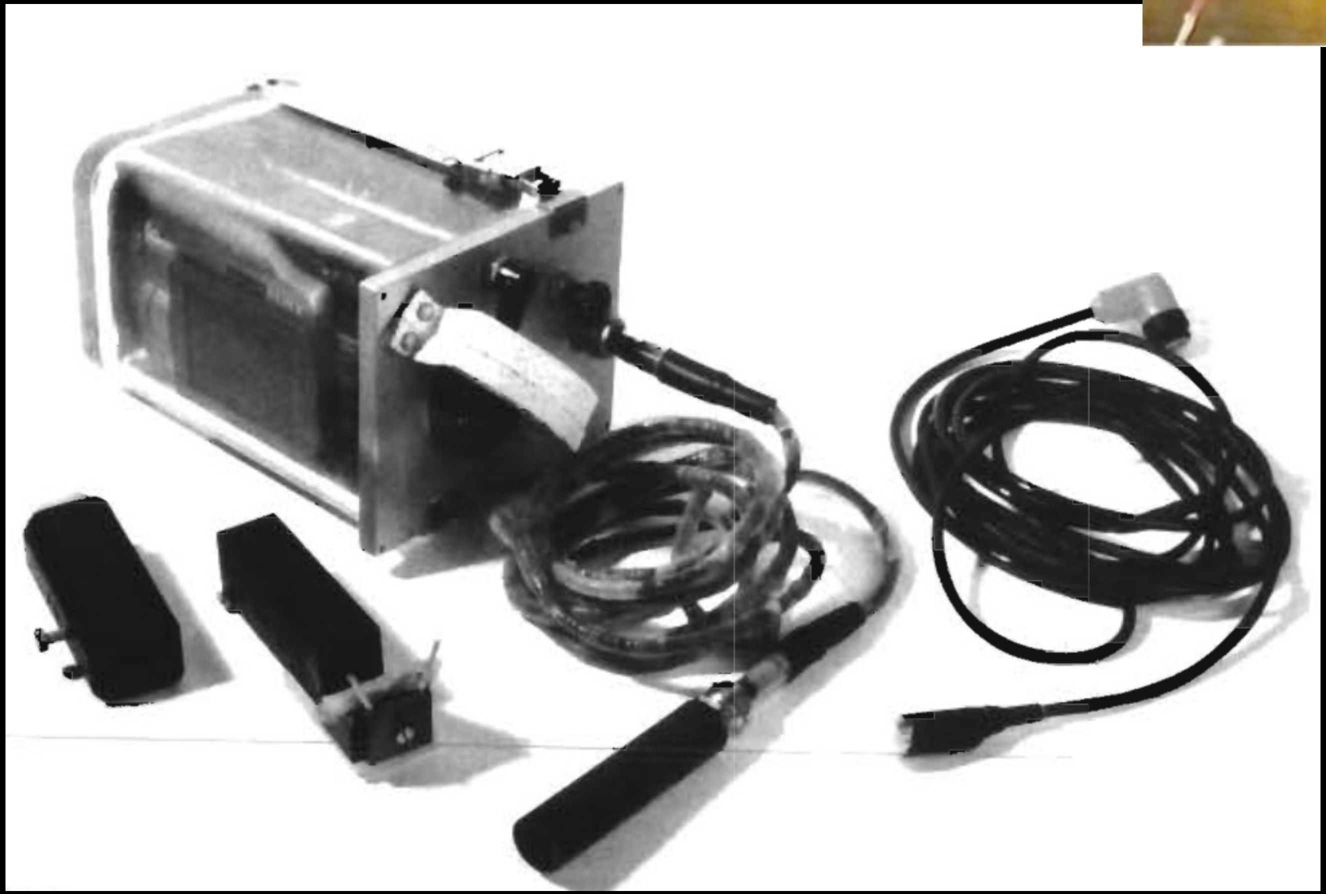
plastic fencing material made to look like live and dead coral



Enochs et al., 2011 Mar. Ecol. Prog. Ser.

CaveCam

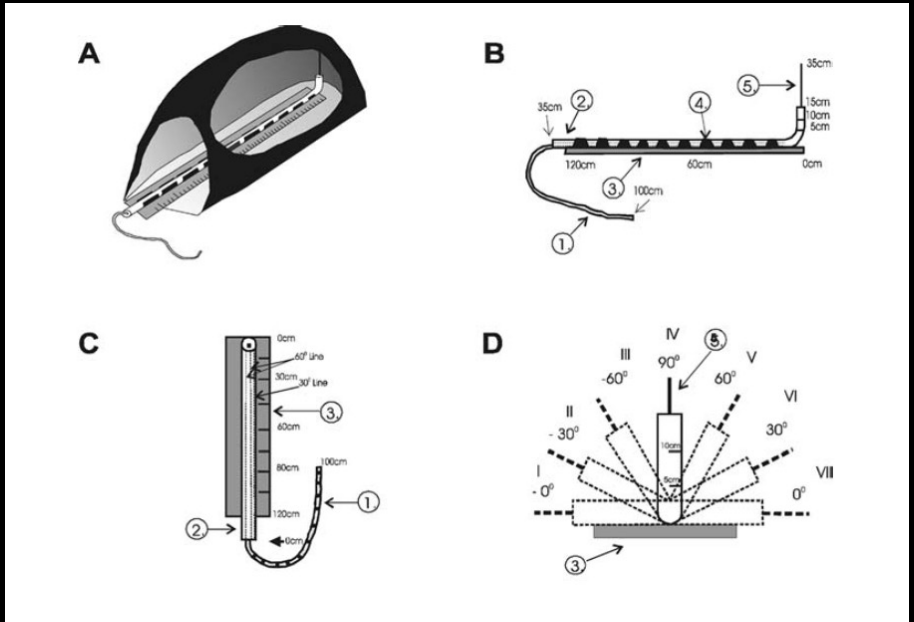
- endoscope for coral crevices
- penetrated 4m into coral matrix (carbonate rock)
- penetrated crevices of 20cm median diameter (range: 16cm-45cm)

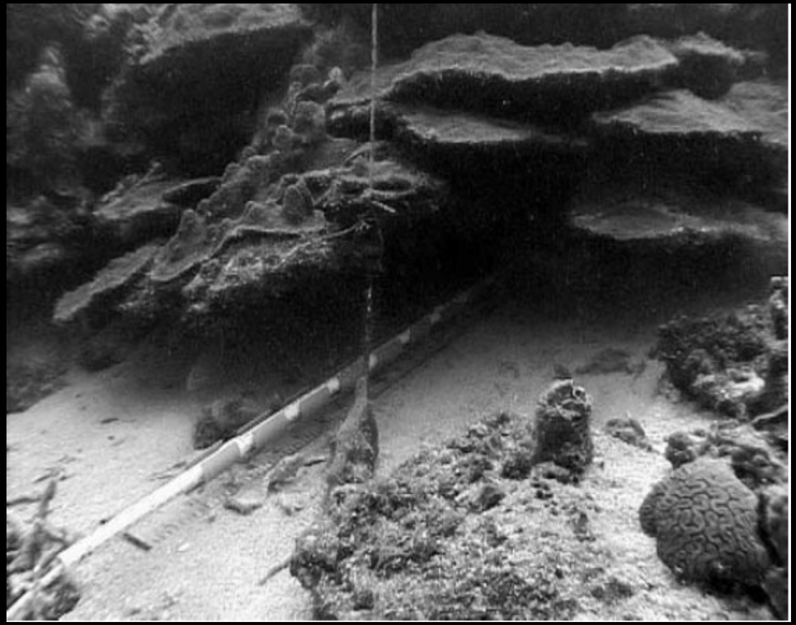


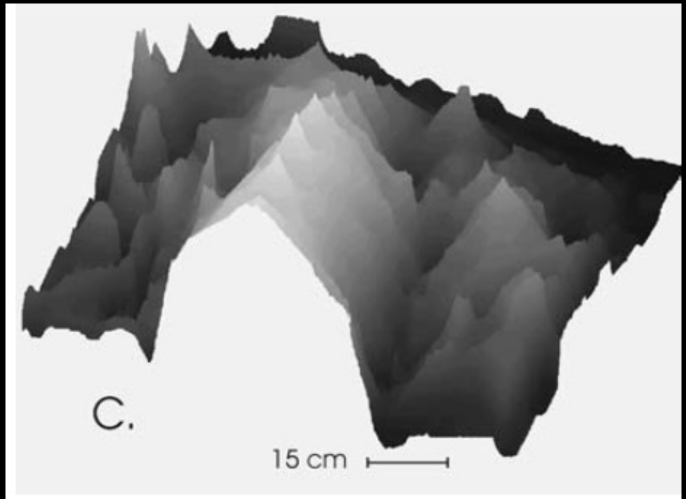
Richter et al. 2001. *Nature* 413:726-7-30;
Wunsch et al. 1998. *Mar. Ecol. Prog. Ser.* 169: 277-282

Cave Profiler

- measures cave size and morphology
- a flexible ruler on a stick that can measure height and angles

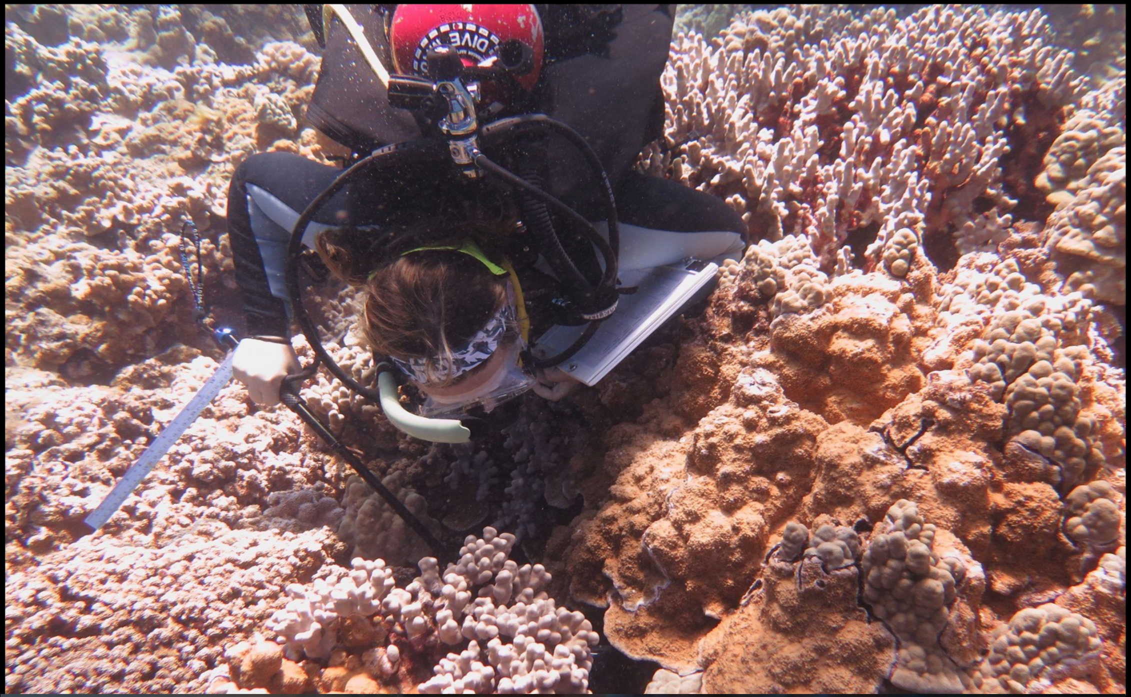






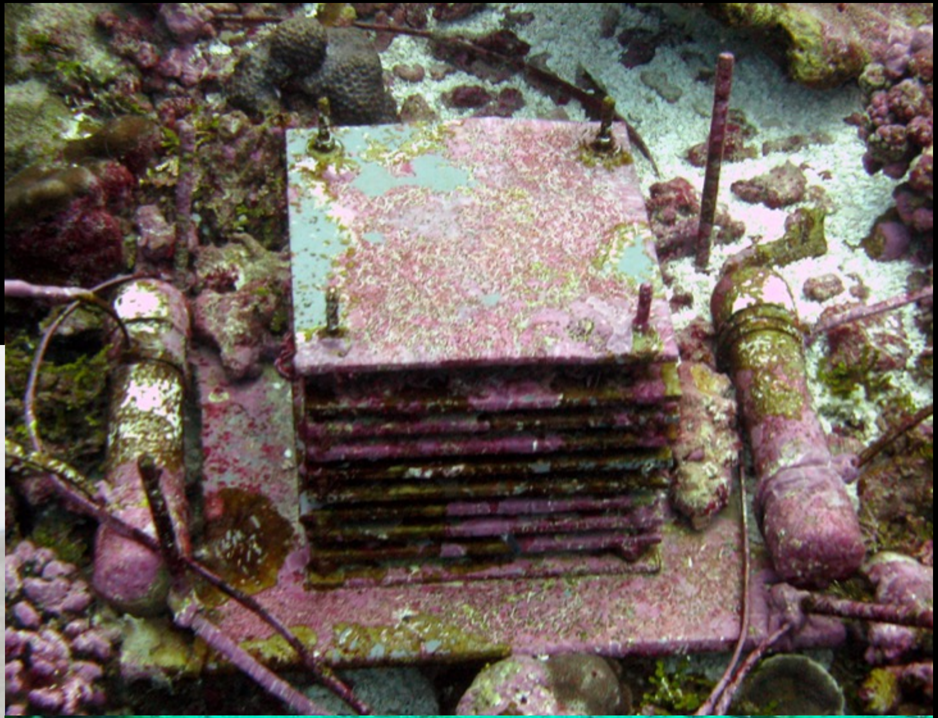
Scheffers et al., 2003 Coral Reefs

GoPro on a selfie stick



ARMS

Autonomous Reef Monitoring Structures



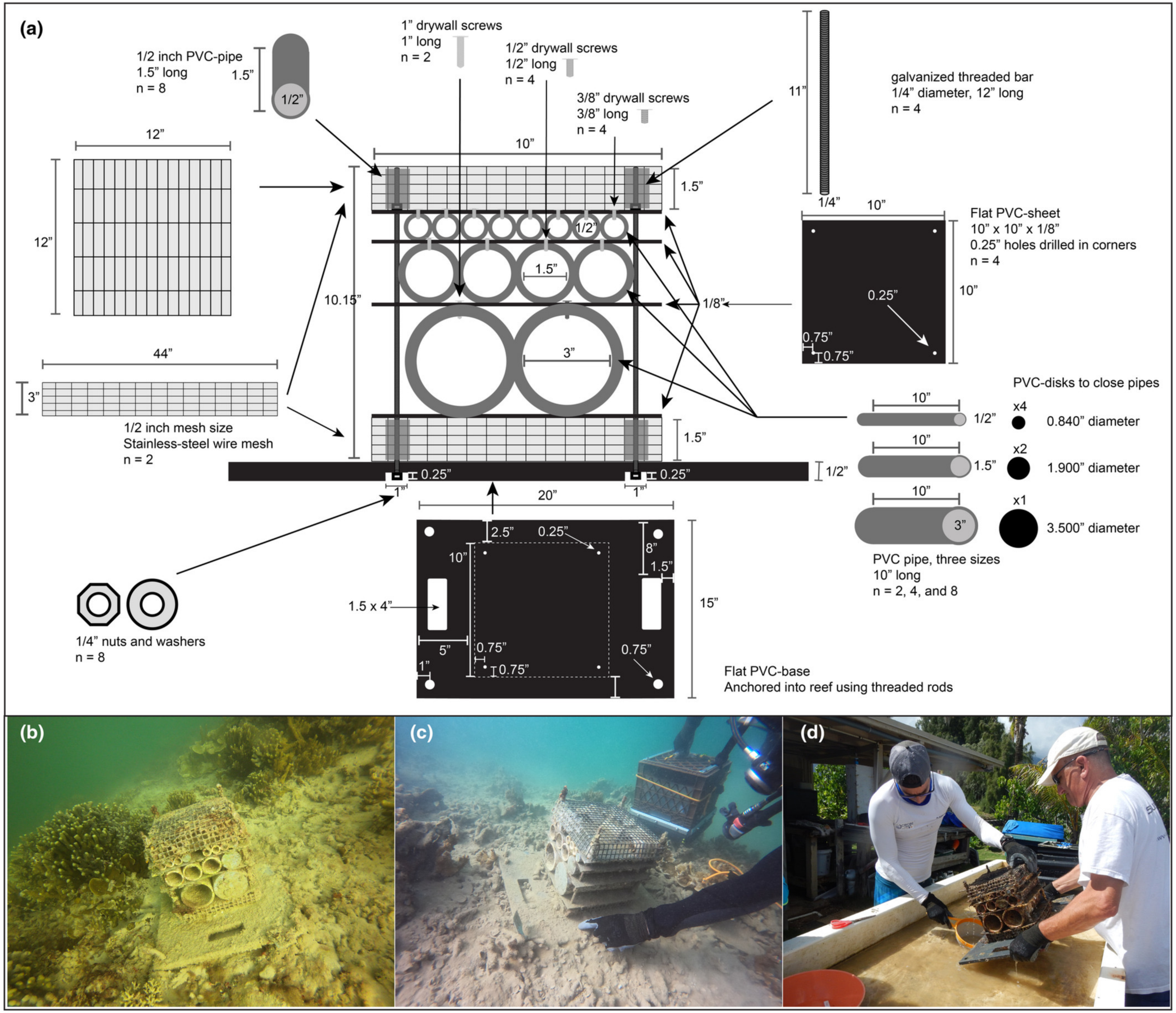
Images: https://www.pifsc.noaa.gov/cred/survey_methods/arms/overview.php
Leray and Knowlton. 2014. *PNAS*
Pearman et al. 2016. *Mar Environ. Res.*

Background: ARMS Research

https://naturalhistory.si.edu/research/global-arms-program

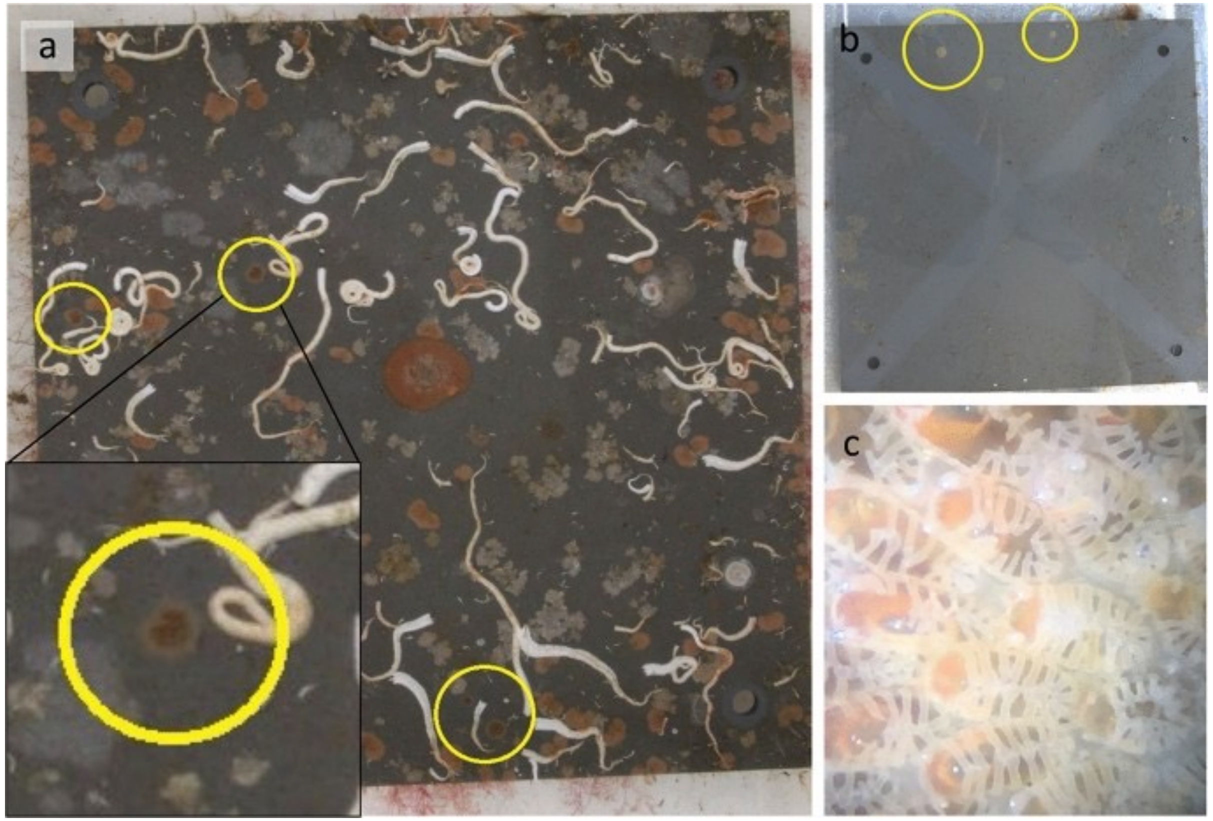
FARMS

Fish-Specific Autonomous Reef Monitoring Structures

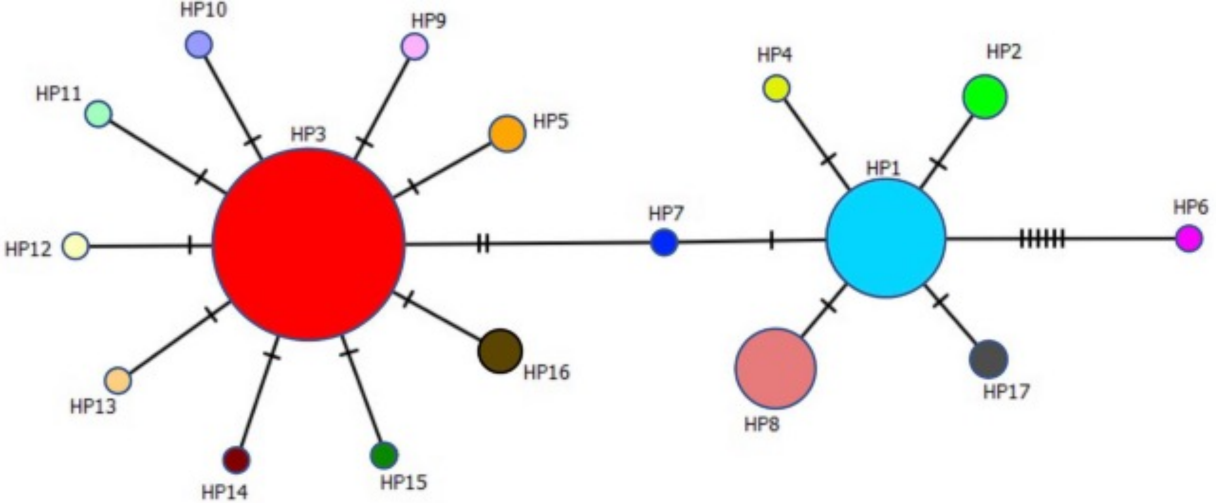


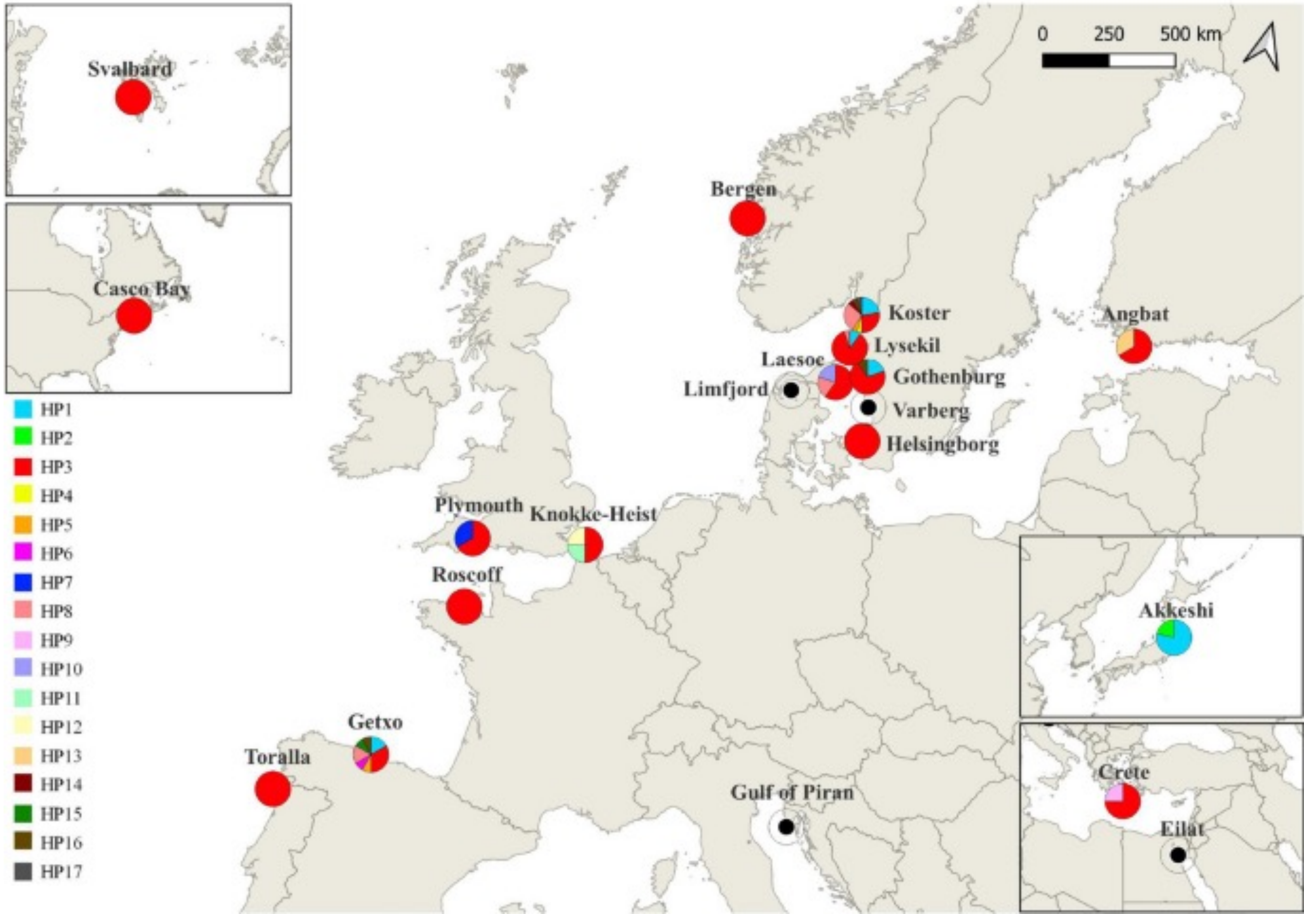
Brandl et al. (2023) *Methods in Evolution and Ecology*

ARMS can tell you about population structure and biogeography (i.e. large-scale distribution) of species



Bryozoans (colonial invertebrates) show fast expansion across Northern Europe





Martaeng et al. (2023) *Hydrobiologia*

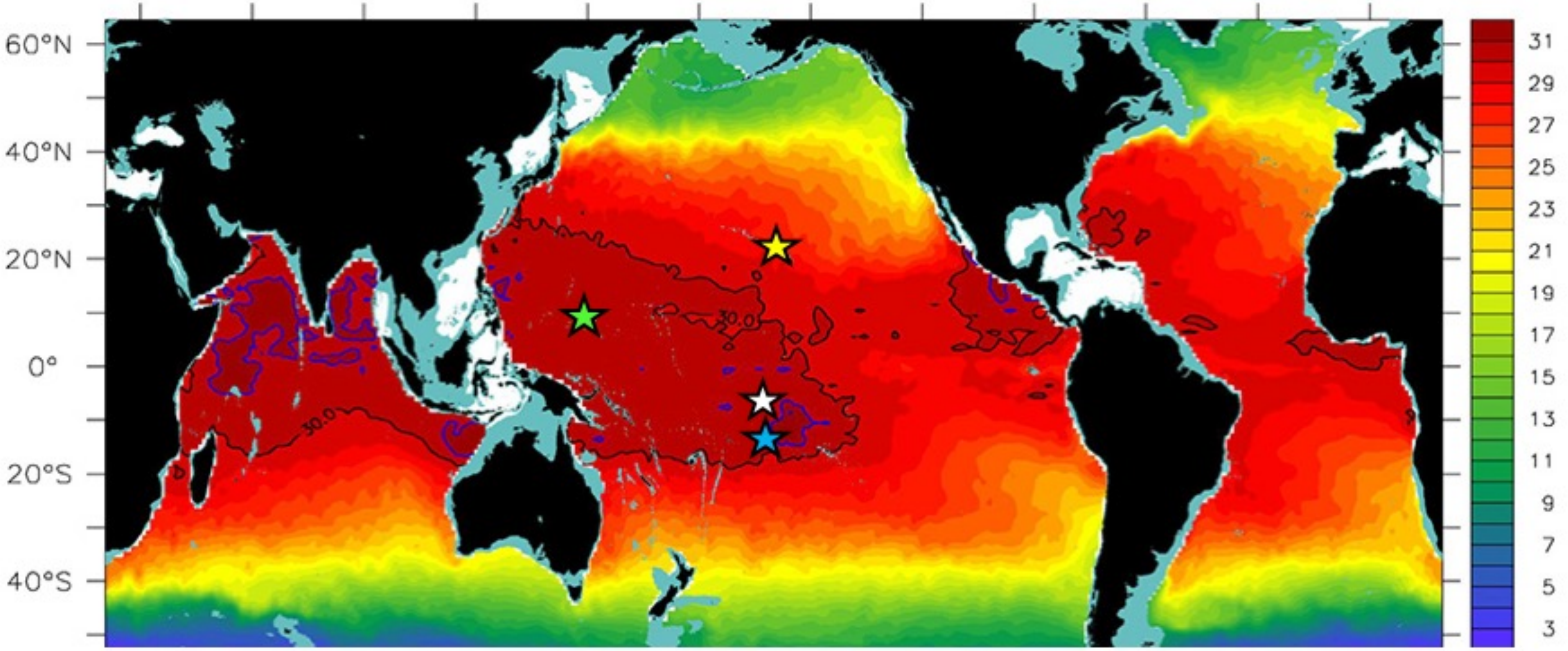
What are the limitations of ARMS?

Difficult to learn about the ecology of cryptic coral habitats *in situ* (i.e. in the wild)

Paisance et al. (2021) *PLOS ONE*

The Biggest ARMS data set

- NOAA's Pacific Reef Assessment and Monitoring Program (Pacific RAMP), 2004-2017.
- Hundreds of multi-plate deployments at pristine coral reefs in the Pacific basin.
  - Marianas ★
  - Hawai'I ☆
  - Line Islands ☆
  - Samoa ★
- Thousands of plate photos.
- COI barcode sequences.



Some sample ARMS images

Guam

Pearl and Hermes Atoll

Some sample ARMS images

Palmyra

Tutuila

ARMS Analysis Challenges

DNA analysis

- Not quantitative
- Dependent on high-quality local databases of barcode gene

Photo analysis

- Human ➔ doesn’t scale up
- Semi-automated ➔ CPCe
- Fully automated ➔ significant progress on the difficult first half

ARMS Analysis Challenges

DNA analysis

- Not quantitative
- Dependent on high-quality local databases of barcode gene

Photo analysis

- Fully visual ➔ doesn't scale up
- Semi-automated ➔ CPCe
- Fully automated ➔ significant progress on the difficult first half

ARMS Analysis Challenges

DNA analysis

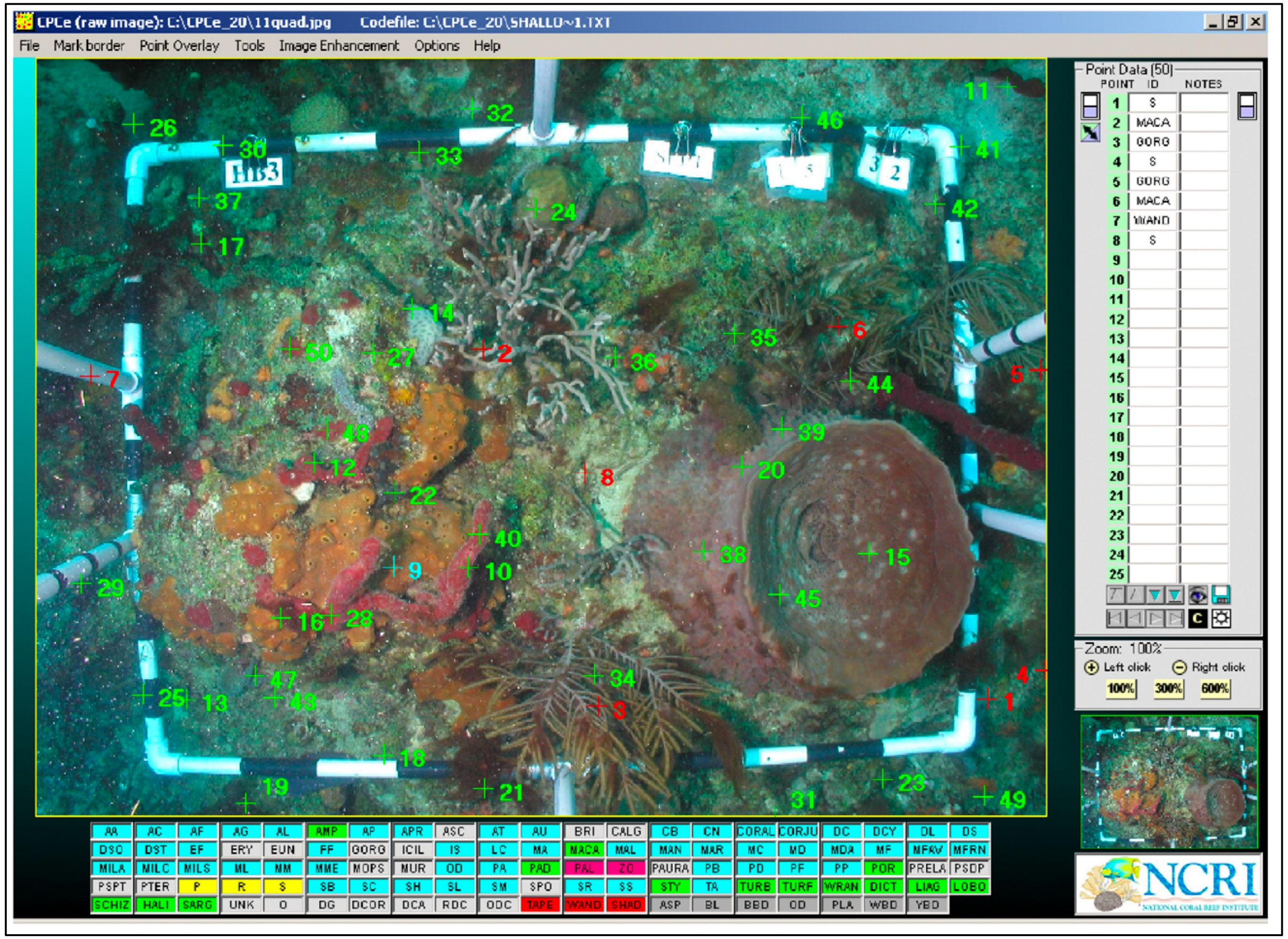
- Not quantitative
- Dependent on high-quality local databases of barcode gene

Photo analysis

- Fully visual → doesn’t scale up
- Semi-automated → CPCe
- Fully automated → significant progress on the difficult first half

CPCe

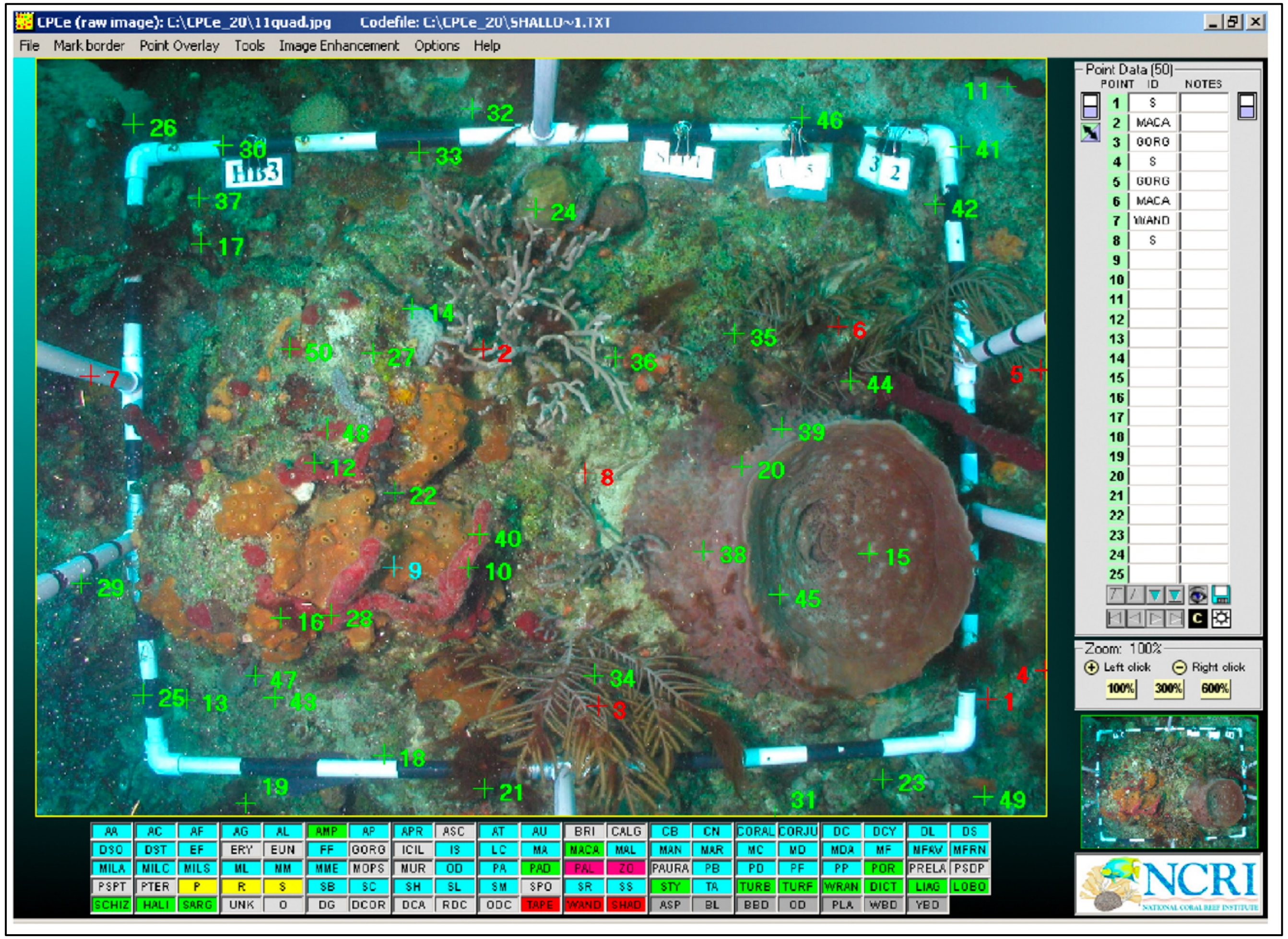
- SOTA (State of the Art)
- Operator chooses a photo (this example isn't an ARMS)
- Software randomly chooses n points (often 50 or 100)
- Operator assigns a taxonomy (or n/a) to the organism at each point



Kohler & Gill, Computers and Geosciences (2006)

CPCe: So many possible failure points

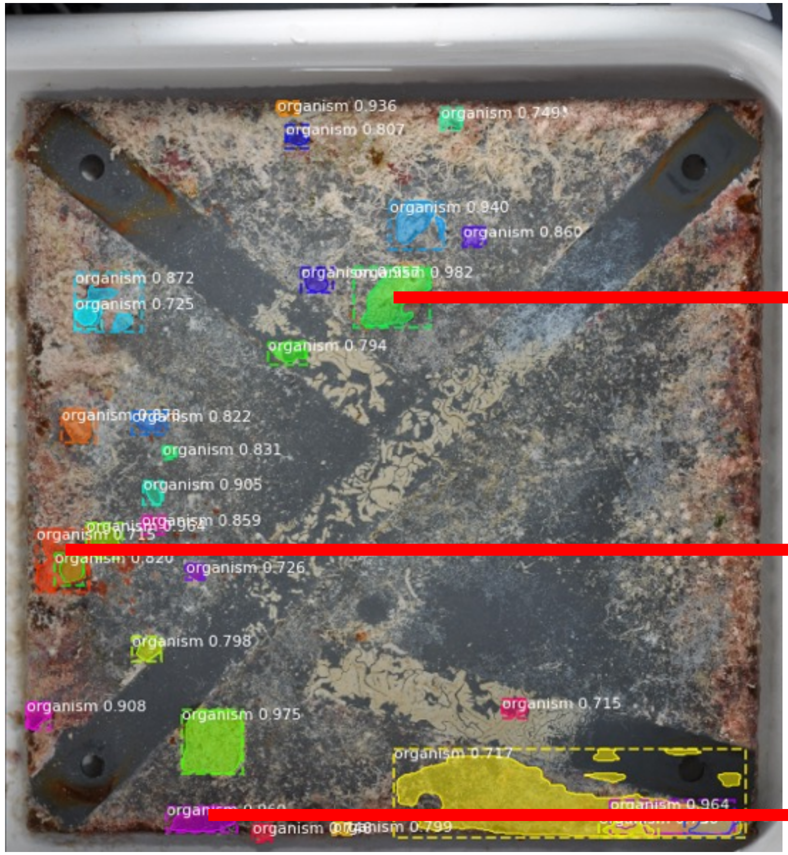
- Statistical strength: how many points is enough?
- Operator error
- Error propagation due to chain of training



Kohler & Gill, Computers and Geosciences (2006)

Can ARMs photos be analyzed computationally?

Original jpeg



Segmented

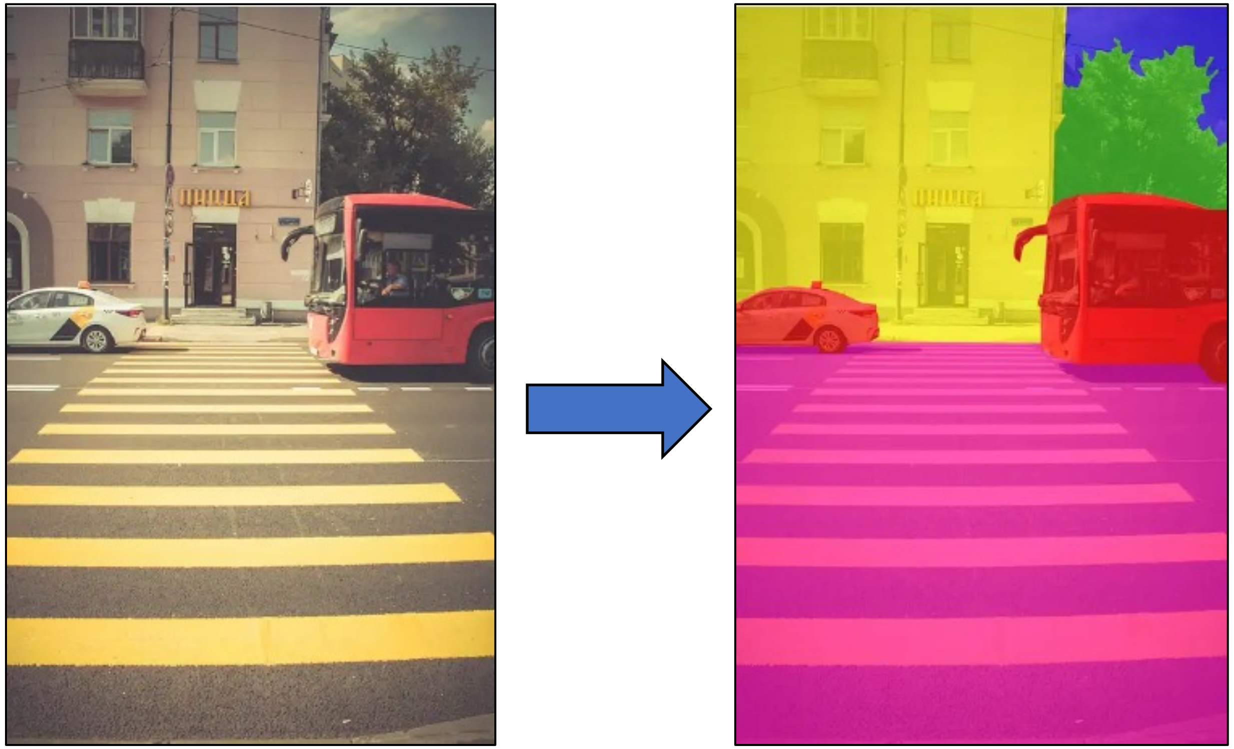
Suberitidae (just guessin')

Turf

CCA

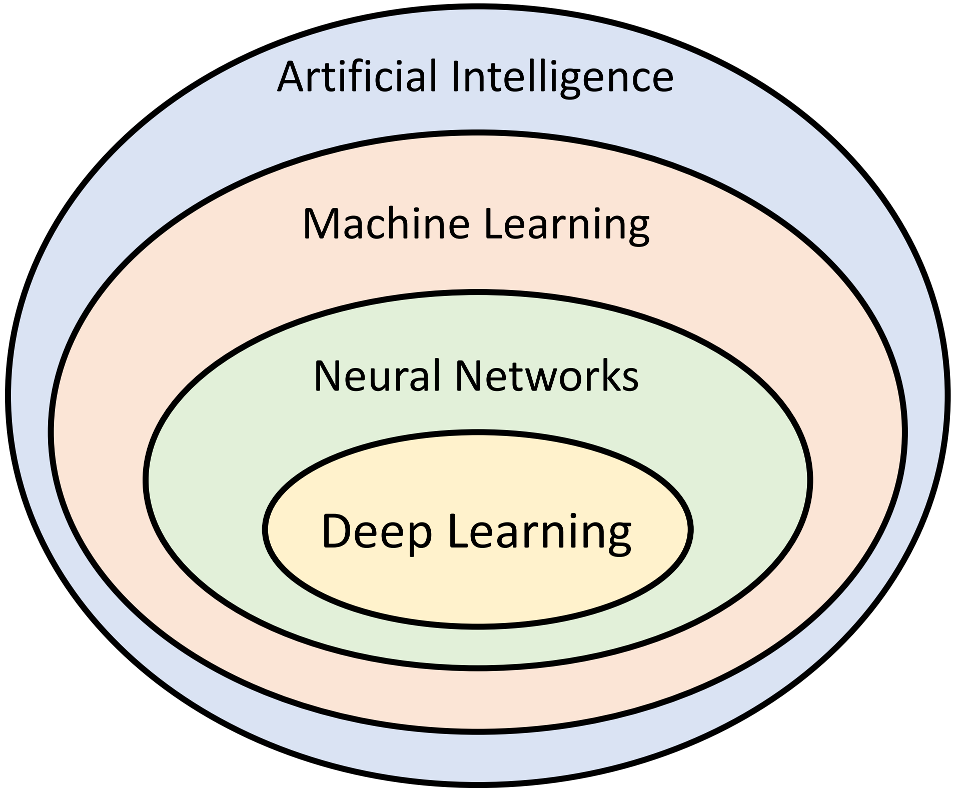
Identified

Segmentation: the hardest step (by a factor of a gazillion)



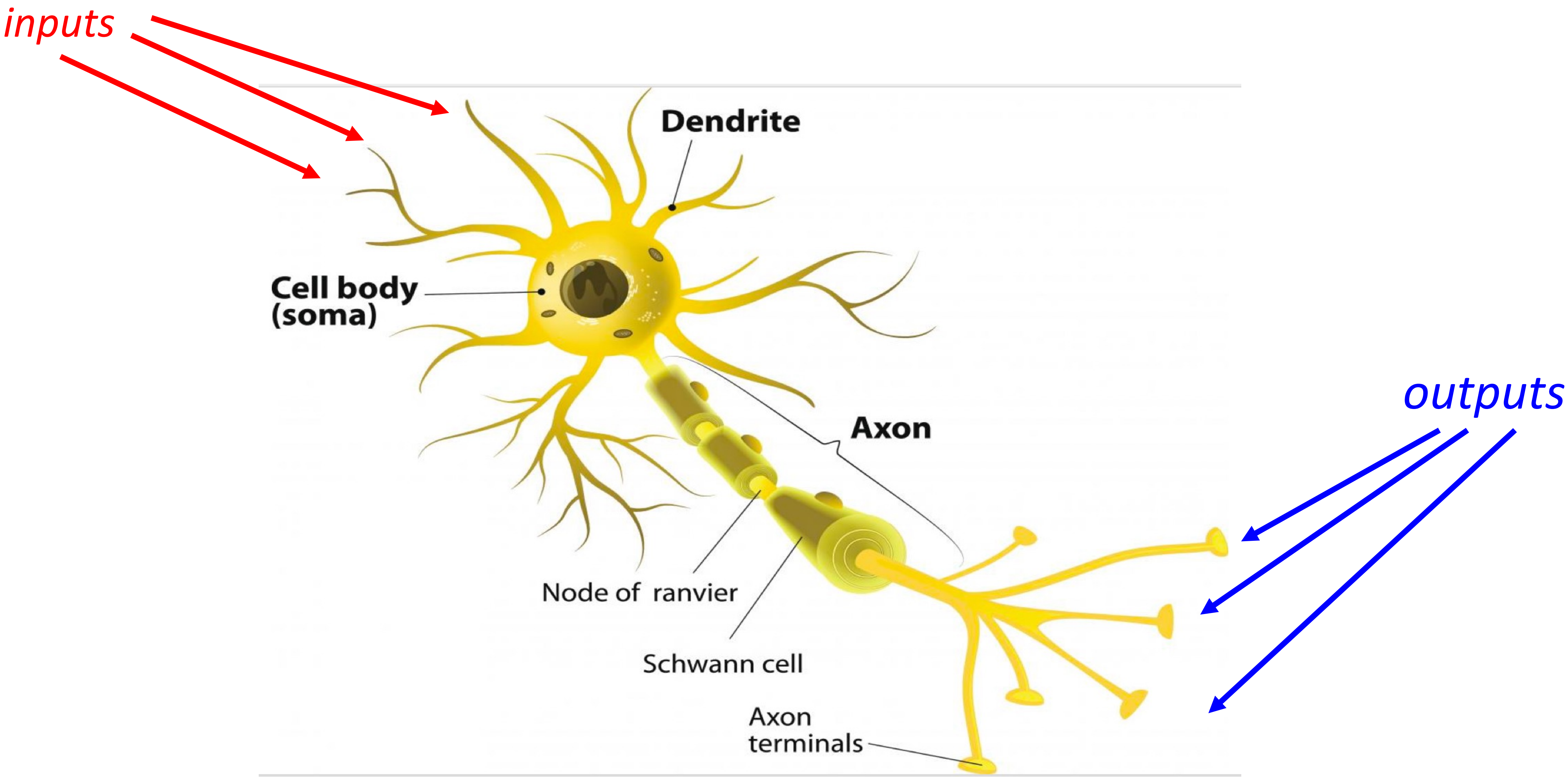
- Given a digitized photo, identify the pixels that belong to individual objects
- Classify the individuals in a later step
- Impossible in the general case without deep-learning neural networks

Segmentation using Deep Learning

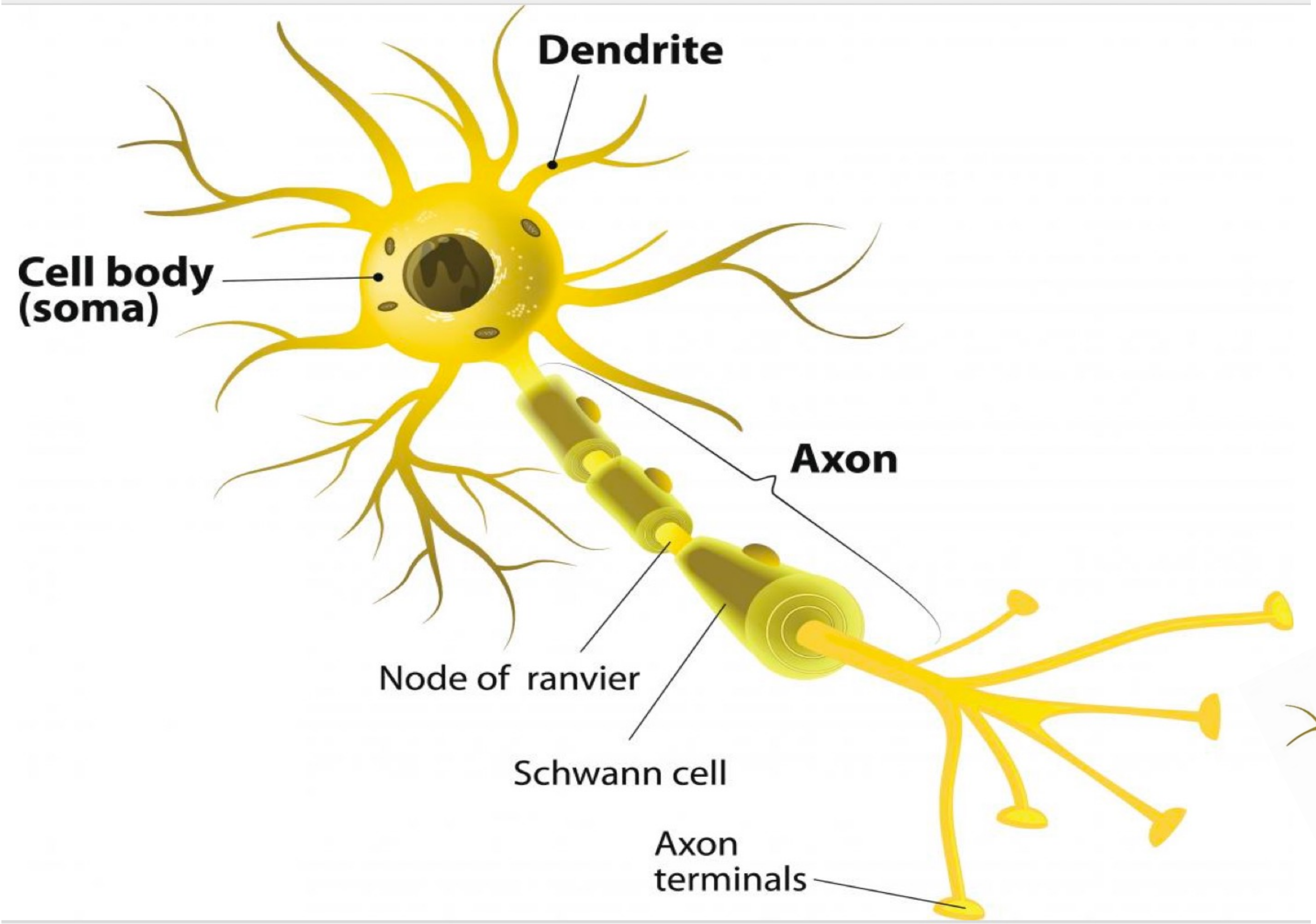


- AI: Software that does something otherwise only done by human intelligence.
- ML: AI where software “learns” optimal parameters by analyzing a training set.
- NN: ML where software simulates 1 or more nerve cells (neurons).
- DL: NN with >= 1 hidden layer.

A nerve cell (neuron)

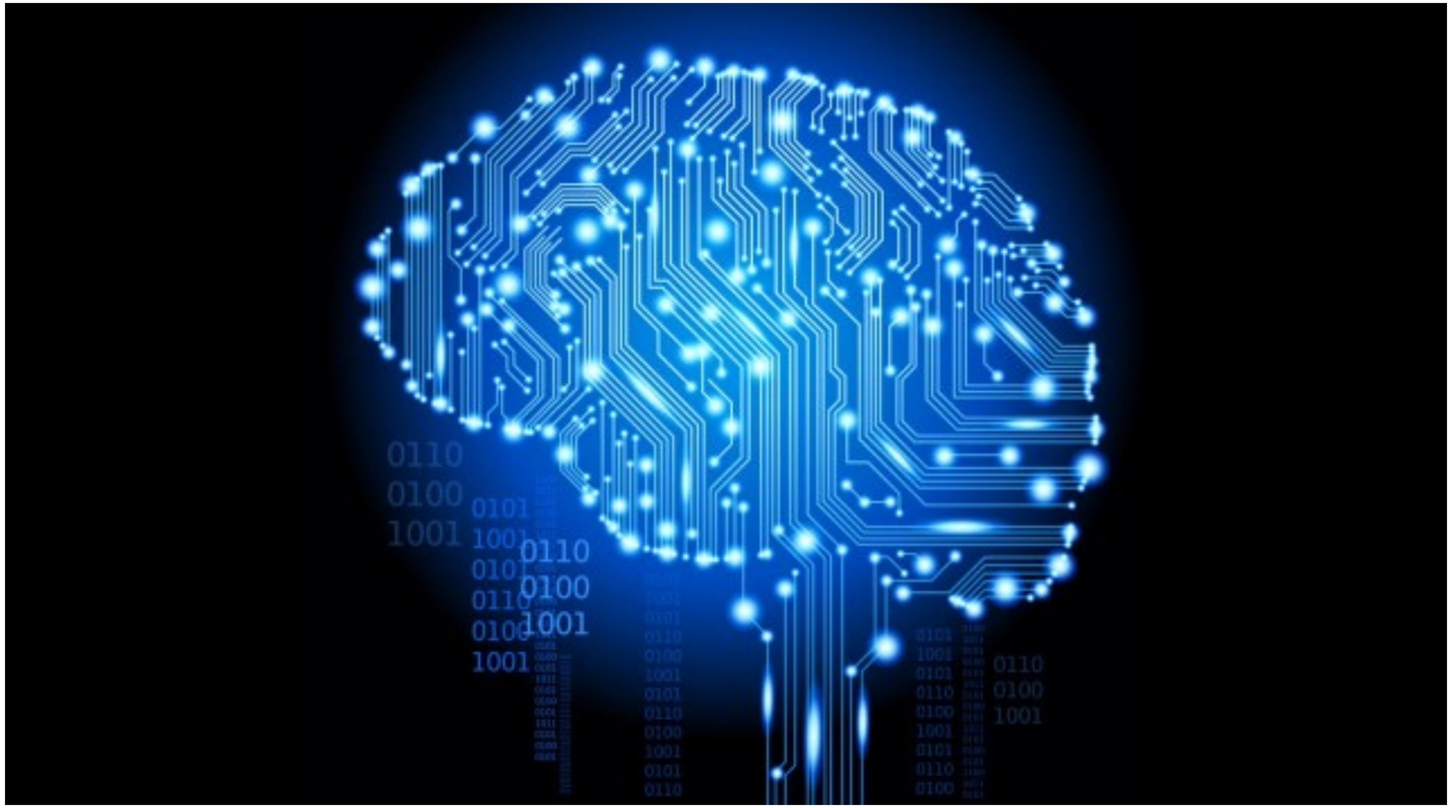


A nerve cell (neuron)

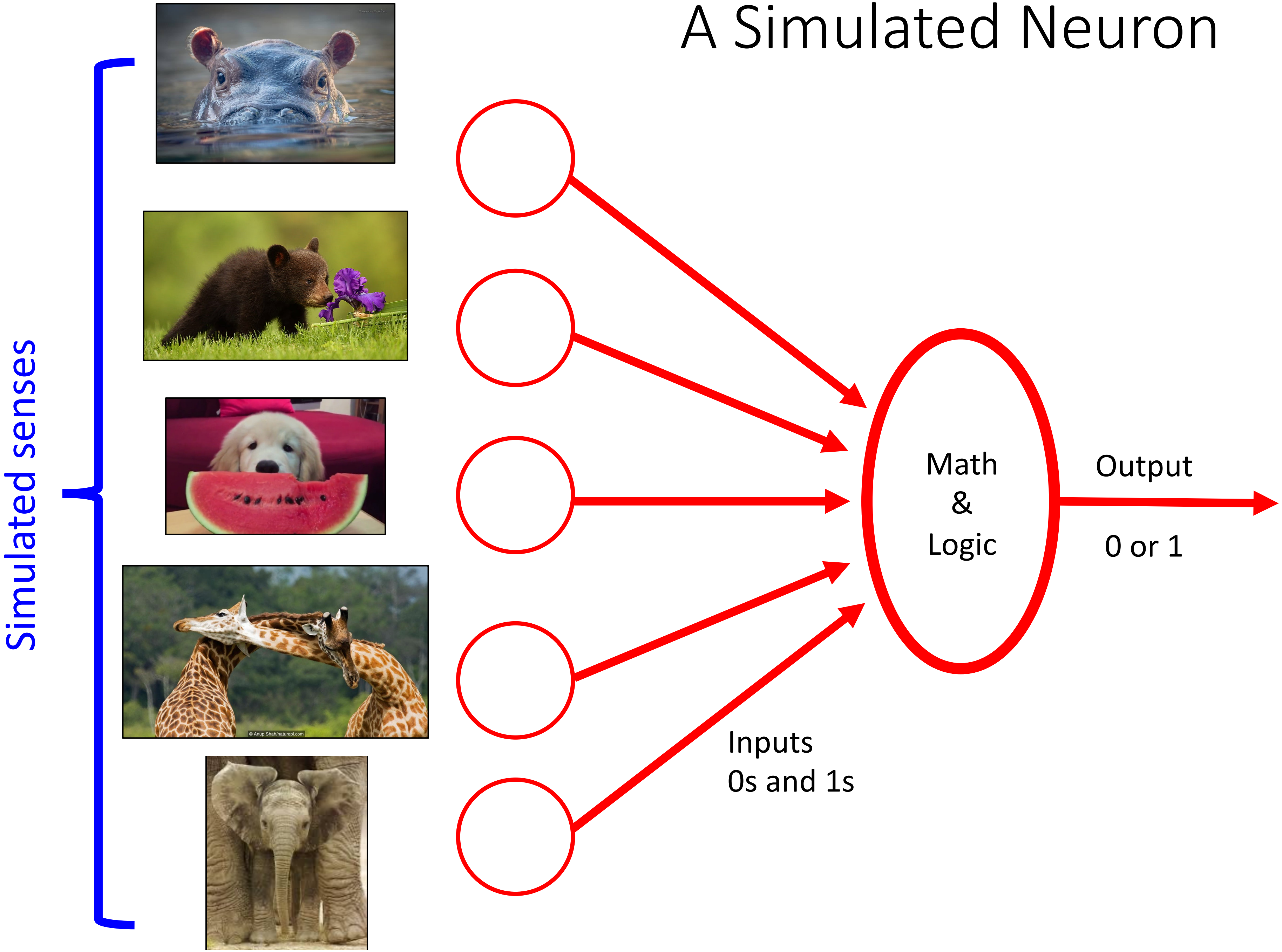


Neural networks: The big idea

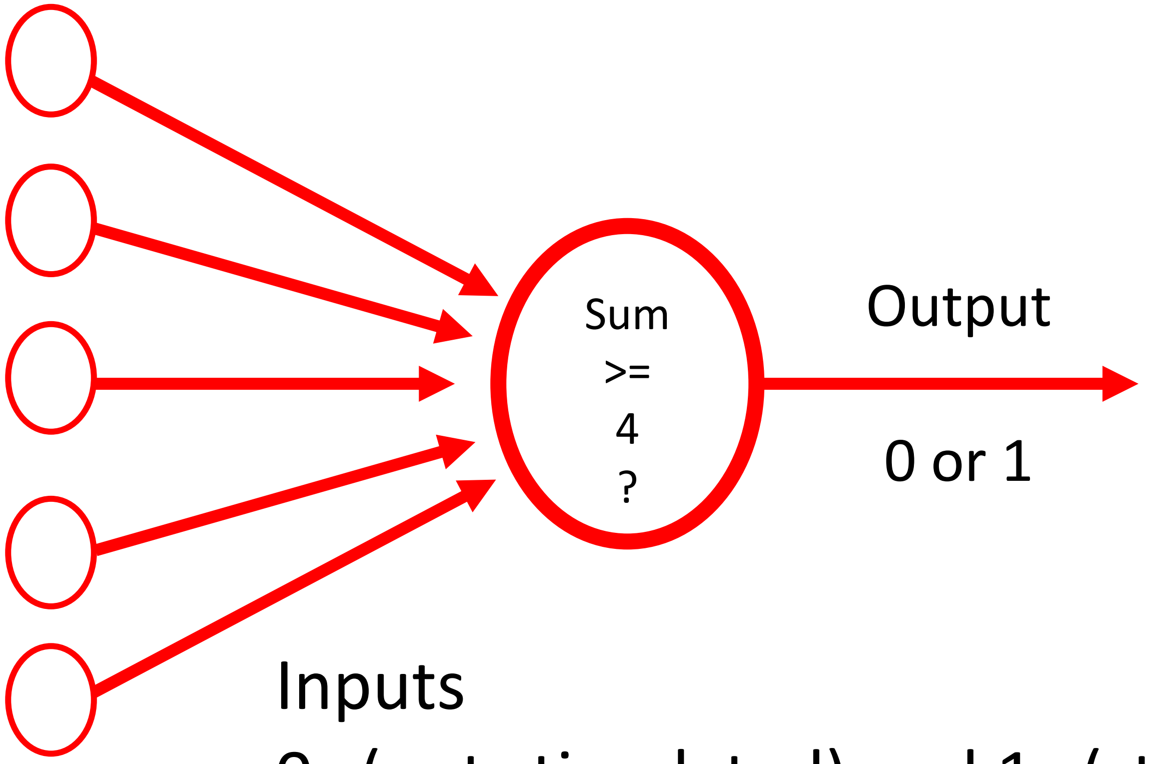
- To model a brain in software:
  - Model a neuron (trivial algorithm)
  - Scale up and interconnect



A Simulated Neuron



Realistic? Imagine a primitive metazoan with an early eye



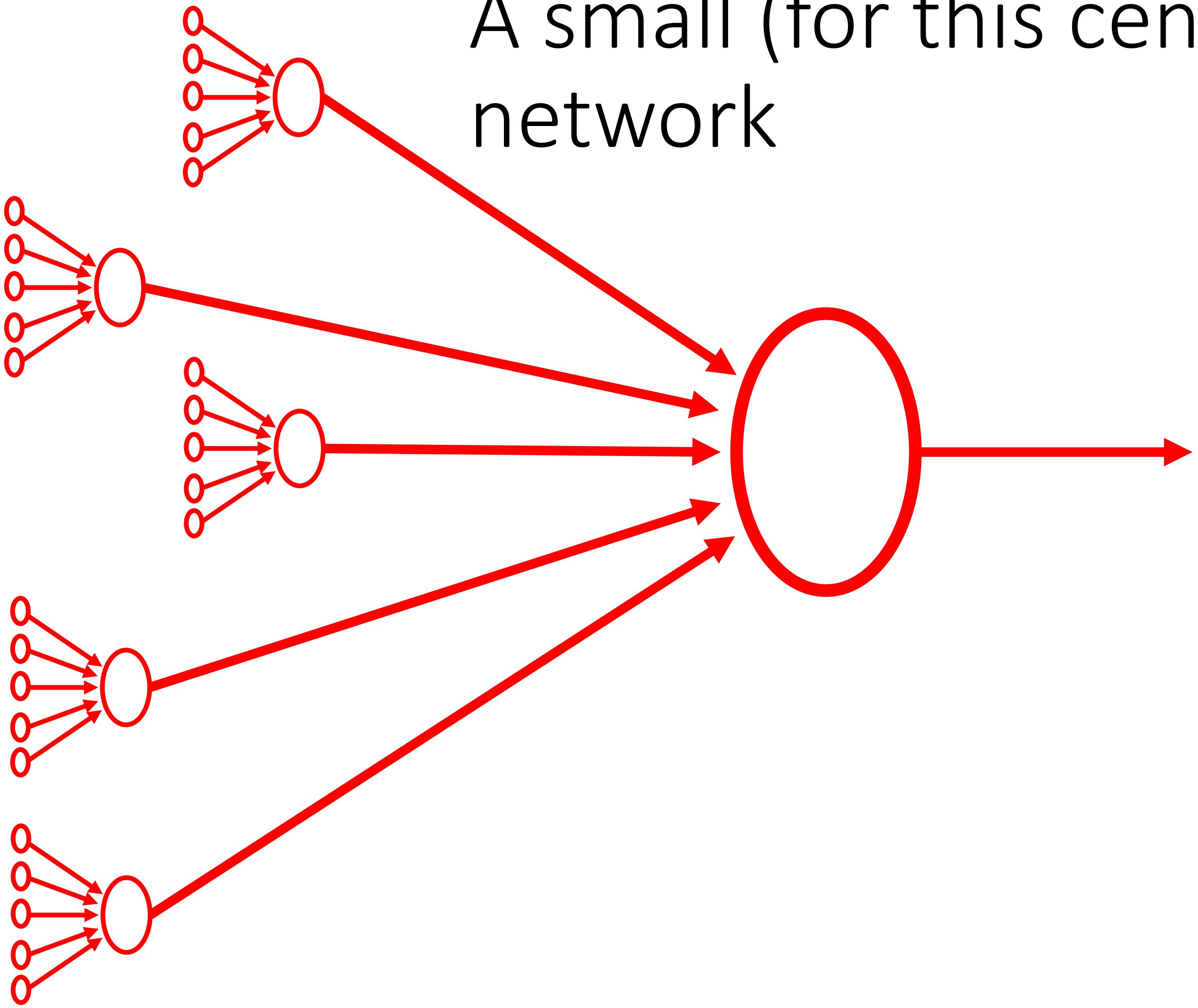
0s (not stimulated) and 1s (stimulated)

Output=1 means a predator just got close

➔

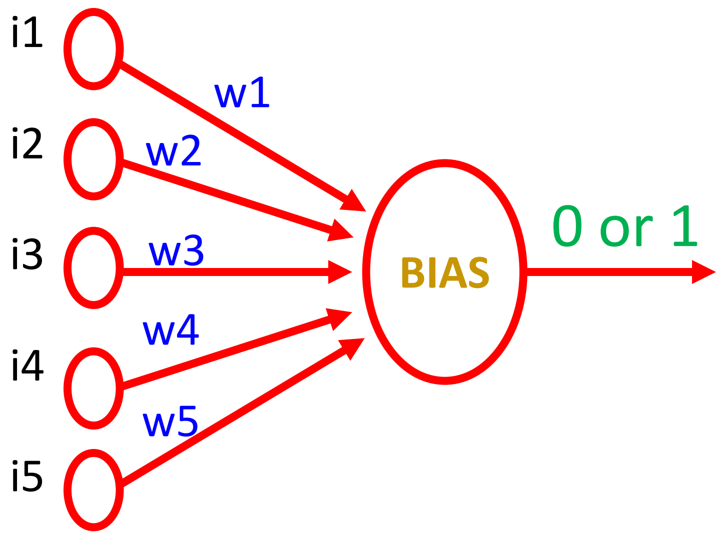
Fire a muscle, maybe escape

A small (for this century) network



Looking closer at an artificial neuron

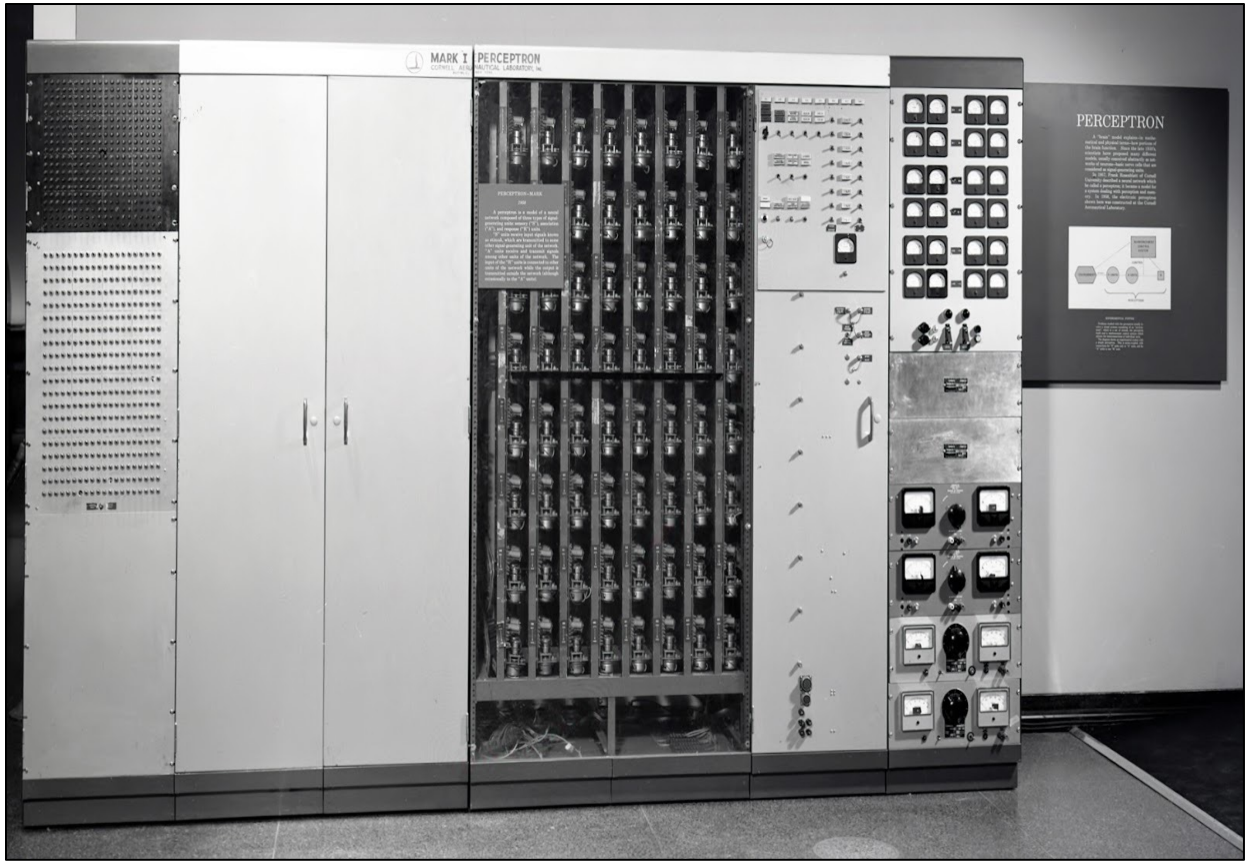
- Input = 0 (“off”) or 1 (“on”) (simplest implementation).
- Output (= 0 or 1) is a function of “learned” parameters called *bias* and *weights*.
  - 1 weight per input.
  - Sum all weights where input is on.
  - Output = 1 if sum >= bias, otherwise output = 0.
  - “Learning” these parameters is the secret sauce.



Neural Networks: History

- 1940s, 1950s:
  - Brains are made of neurons
  - Brains think
  - Therefore if we model neurons we can model thought
- 1960s:
  - Computers have been invented
  - Interesting results with even a single software neuron
  - Theory about multiple neurons, but computers aren't powerful enough
- 1980s:
  - Moderate commercial success, e.g. handwriting recognition
  - AI boom and bust
  - "The AI Winter" – through end of century
- 21st century:
  - Computers catch up
  - Success in some application domains, including image recognition
  - Google releases TensorFlow

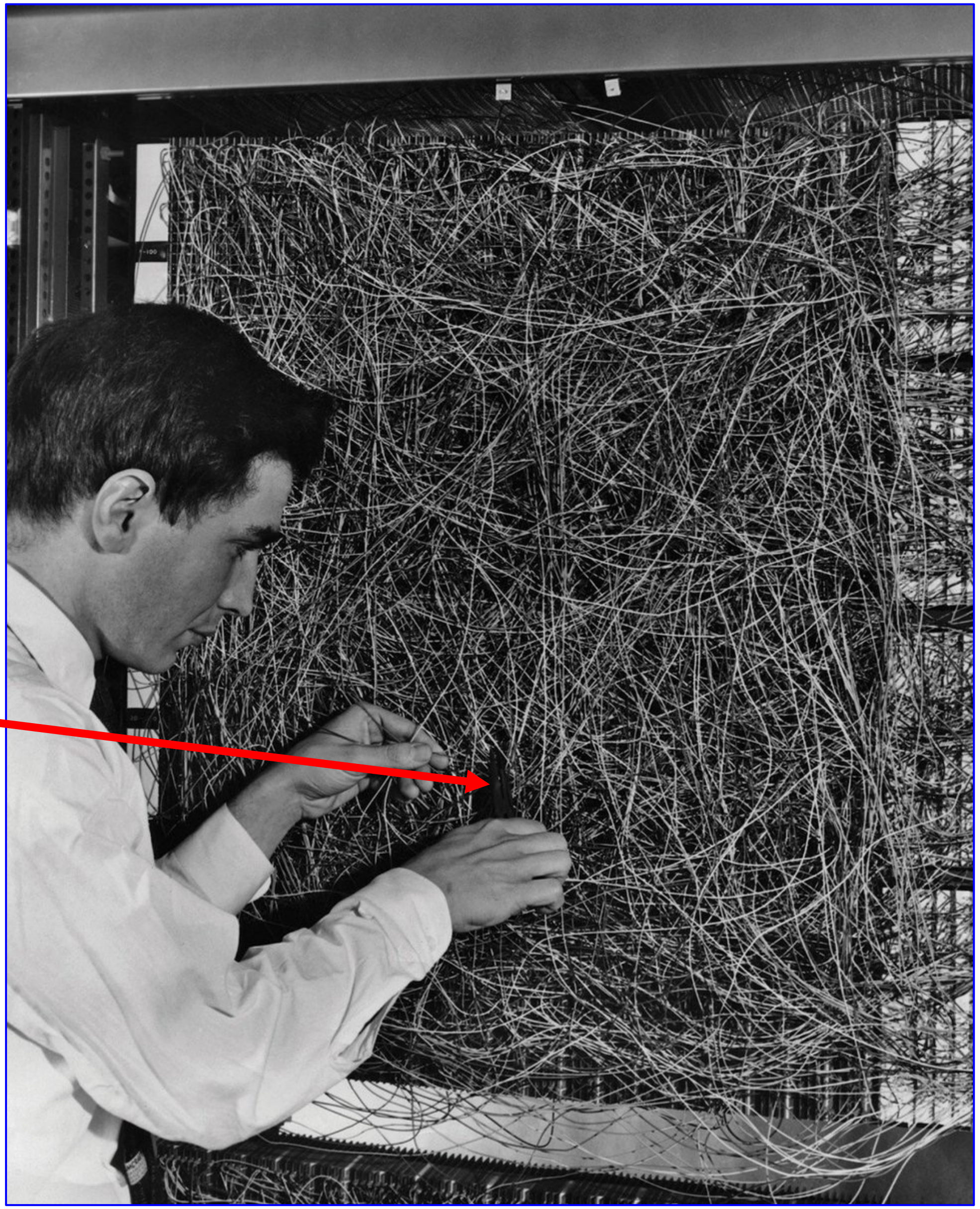
The MARK 1 Perceptron



- 1958
- Filled a room
- $2,600,000 (1958)
- $27.3M today

The MARK 1 Perceptron

Frank Rosenblatt doing brain surgery with a wire cutter



Neural Networks: History

- 1940s, 1950s:
  - Brains are made of neurons
  - Brains think
  - Therefore if we model neurons we can model thought
- 1960s:
  - Computers have been invented
  - Interesting results with even a single software neuron
  - Theory about multiple neurons, but computers aren't powerful enough
- 1980s:
  - Moderate commercial success, e.g. handwriting recognition
  - AI boom and bust
  - "The AI Winter" – through end of century
- 21st century:
  - Computers catch up
  - Success in some application domains, including image recognition
  - Google releases TensorFlow

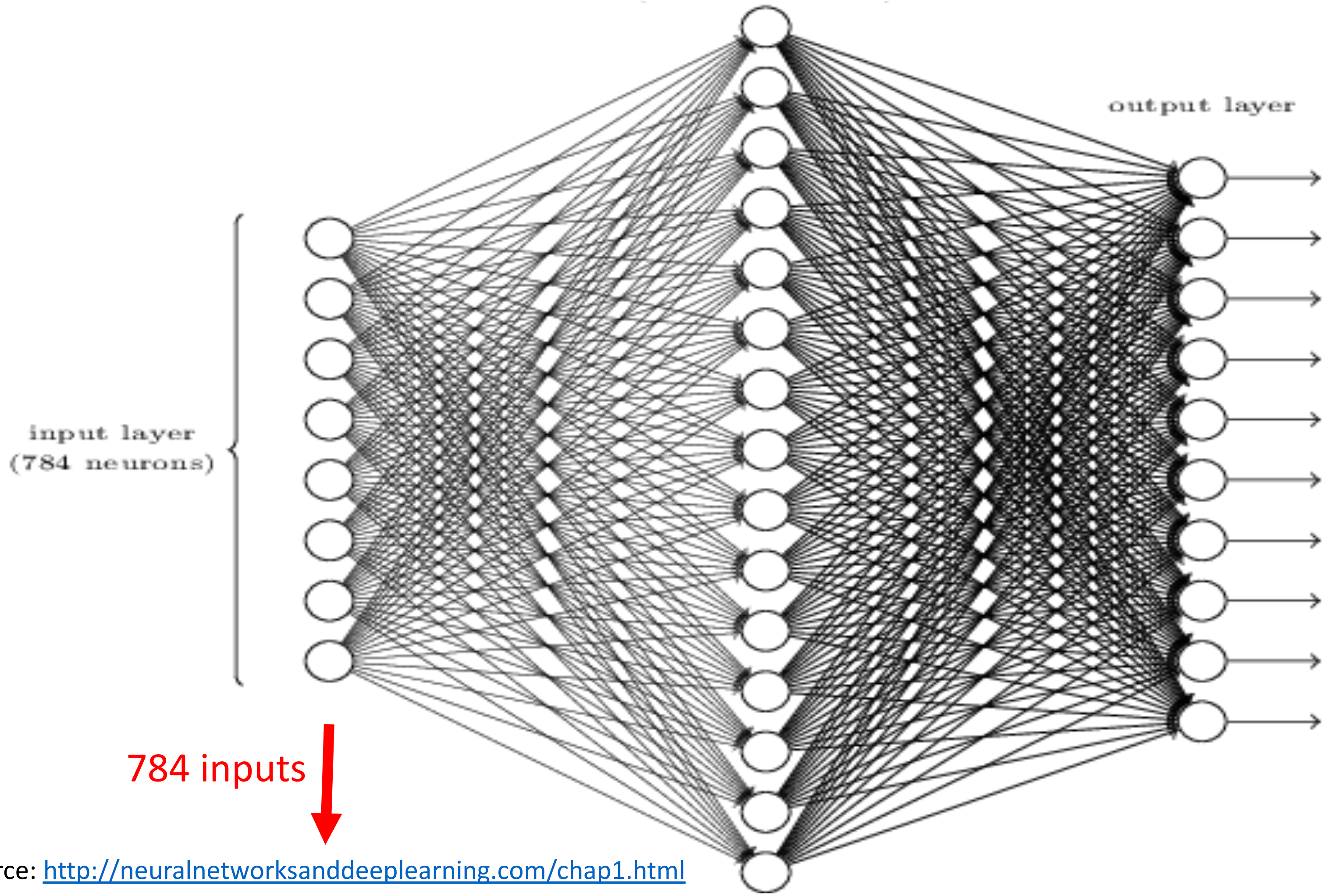
Neural Networks: History

- 1940s, 1950s:
  - Brains are made of neurons
  - Brains think
  - Therefore if we model neurons we can model thought
- 1960s:
  - Computers have been invented
  - Interesting results with even a single software neuron
  - Theory about multiple neurons, but computers aren't powerful enough
- 1980s:
  - Moderate commercial success, e.g. handwriting recognition
  - AI boom and bust
  - "The AI Winter" – through end of century
- 21^{st} century:
  - Computers catch up
  - Success in some application domains, including image recognition
  - Google releases TensorFlow

Neural Networks: History

- 1940s, 1950s:
  - Brains are made of neurons
  - Brains think
  - Therefore if we model neurons we can model thought
- 1960s:
  - Computers have been invented
  - Interesting results with even a single software neuron
  - Theory about multiple neurons, but computers aren't powerful enough
- 1980s:
  - Moderate commercial success, e.g. handwriting recognition
  - AI boom and bust
  - "The AI Winter" – through end of century
- 21st century:
  - Computers catch up
  - Success in some application domains, including image recognition
  - Google releases TensorFlow

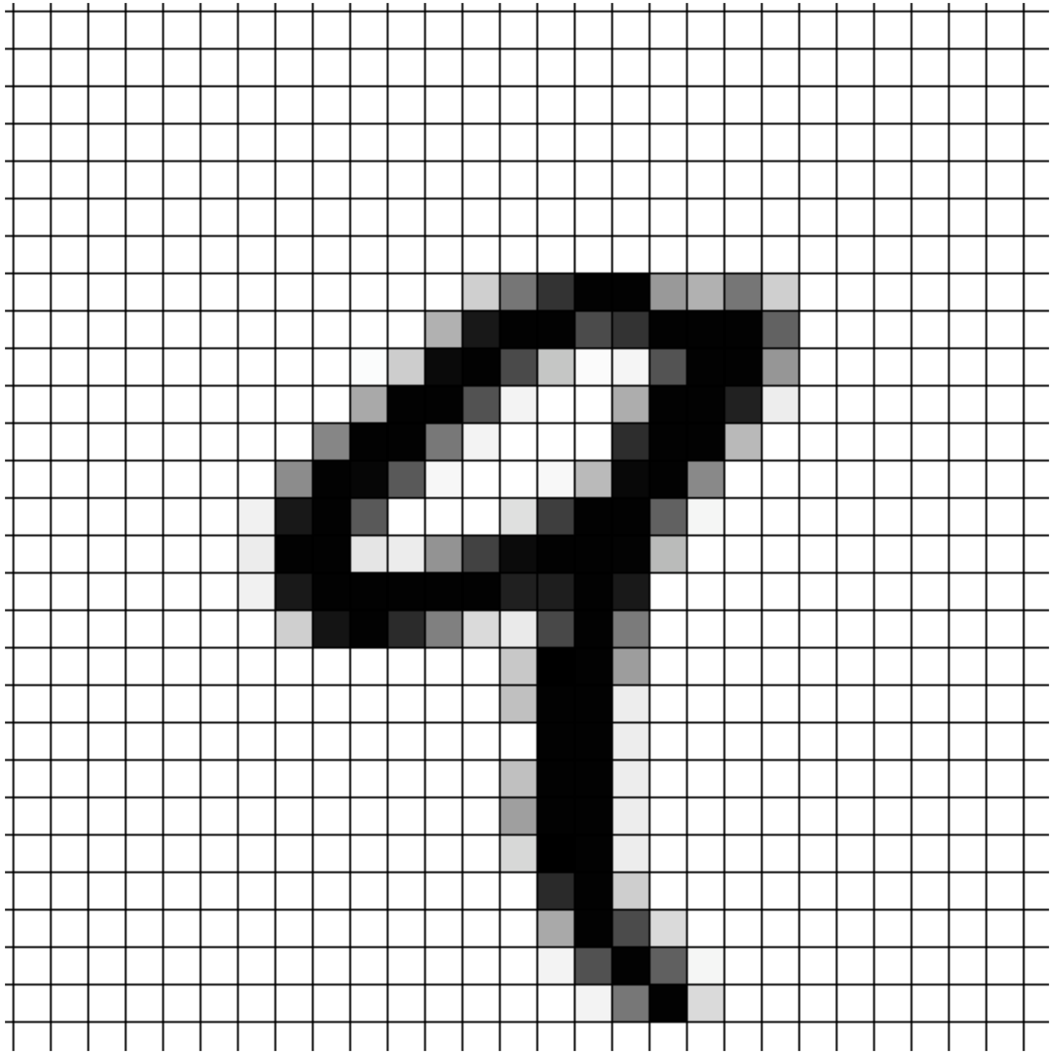
A Deep Learning neural network for recognizing handwritten digits



784 inputs

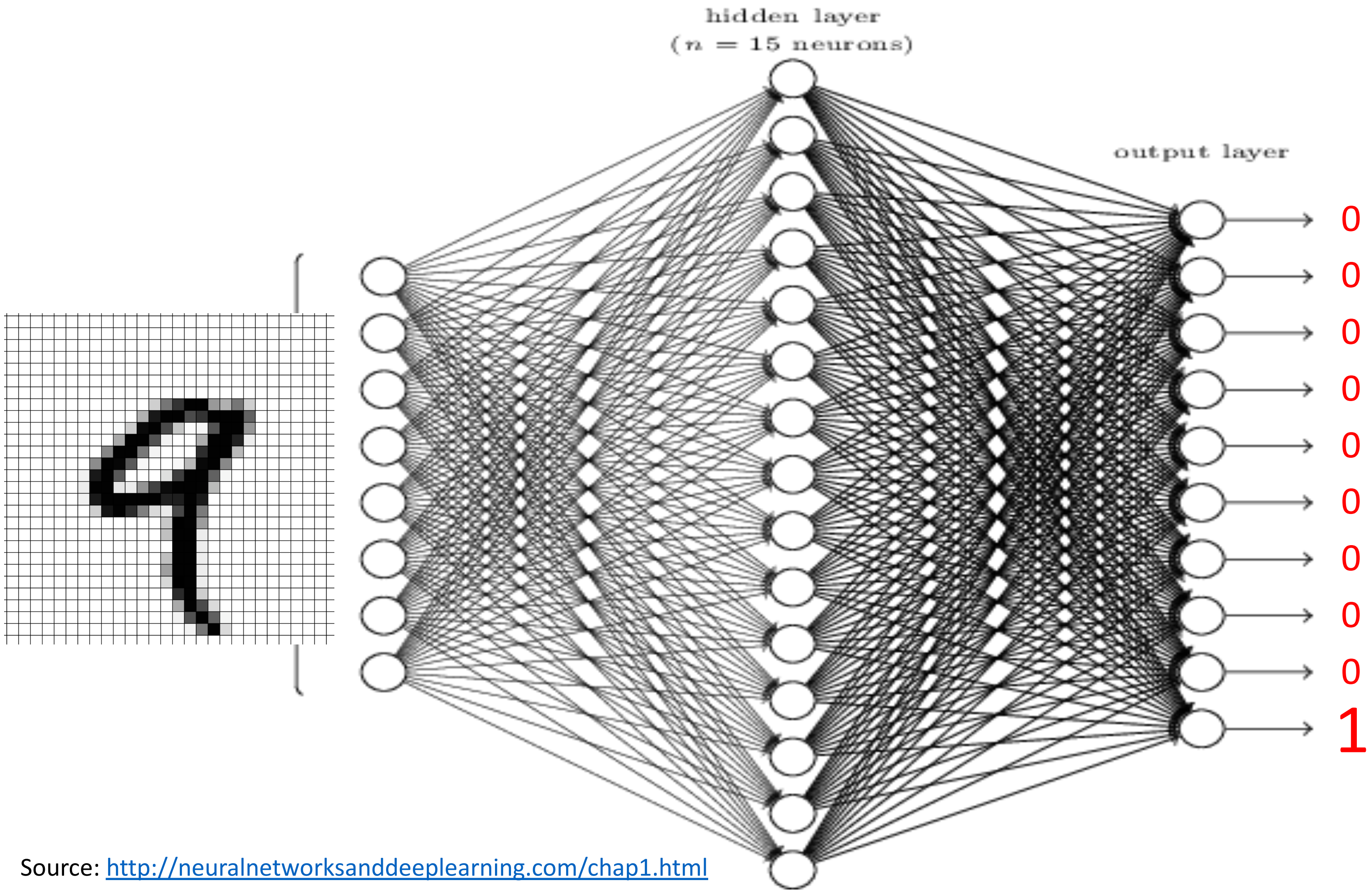
Source: http://neuralnetworksanddeeplearning.com/chap1.html

784 inputs = 28 x 28 pixels



- Convert each pixel to a black-or-white representation
- 0 or 1
- These are the inputs to the 1st layer of neurons

What we want:

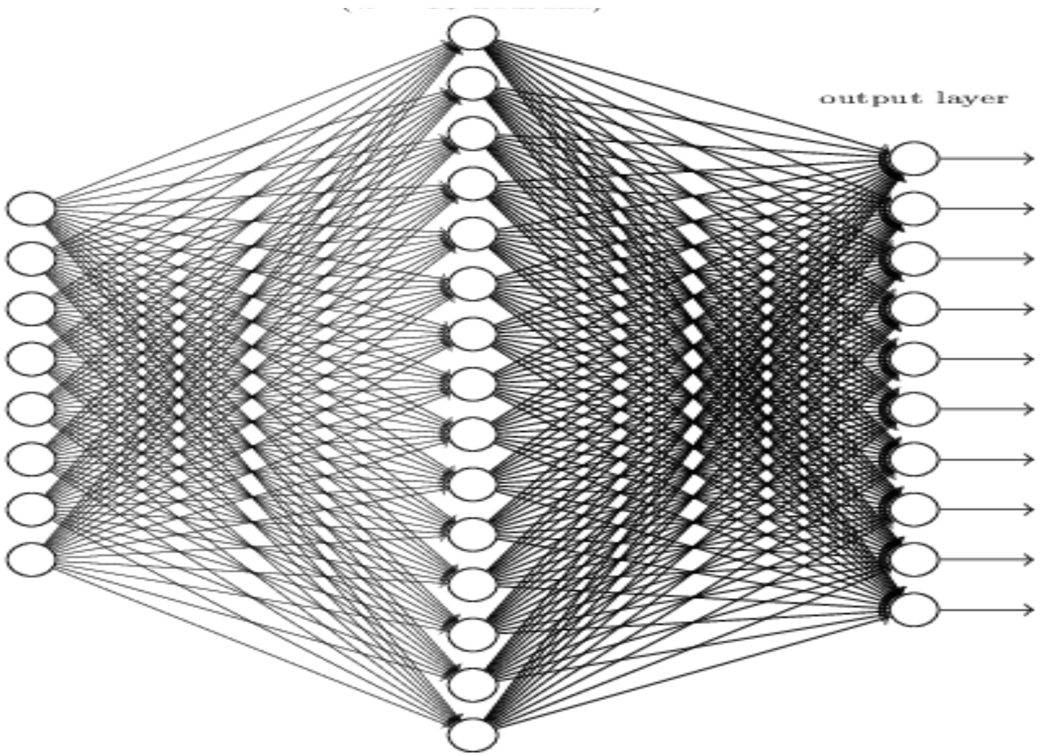


Source: http://neuralnetworksanddeeplearning.com/chap1.html

Now all we need is …

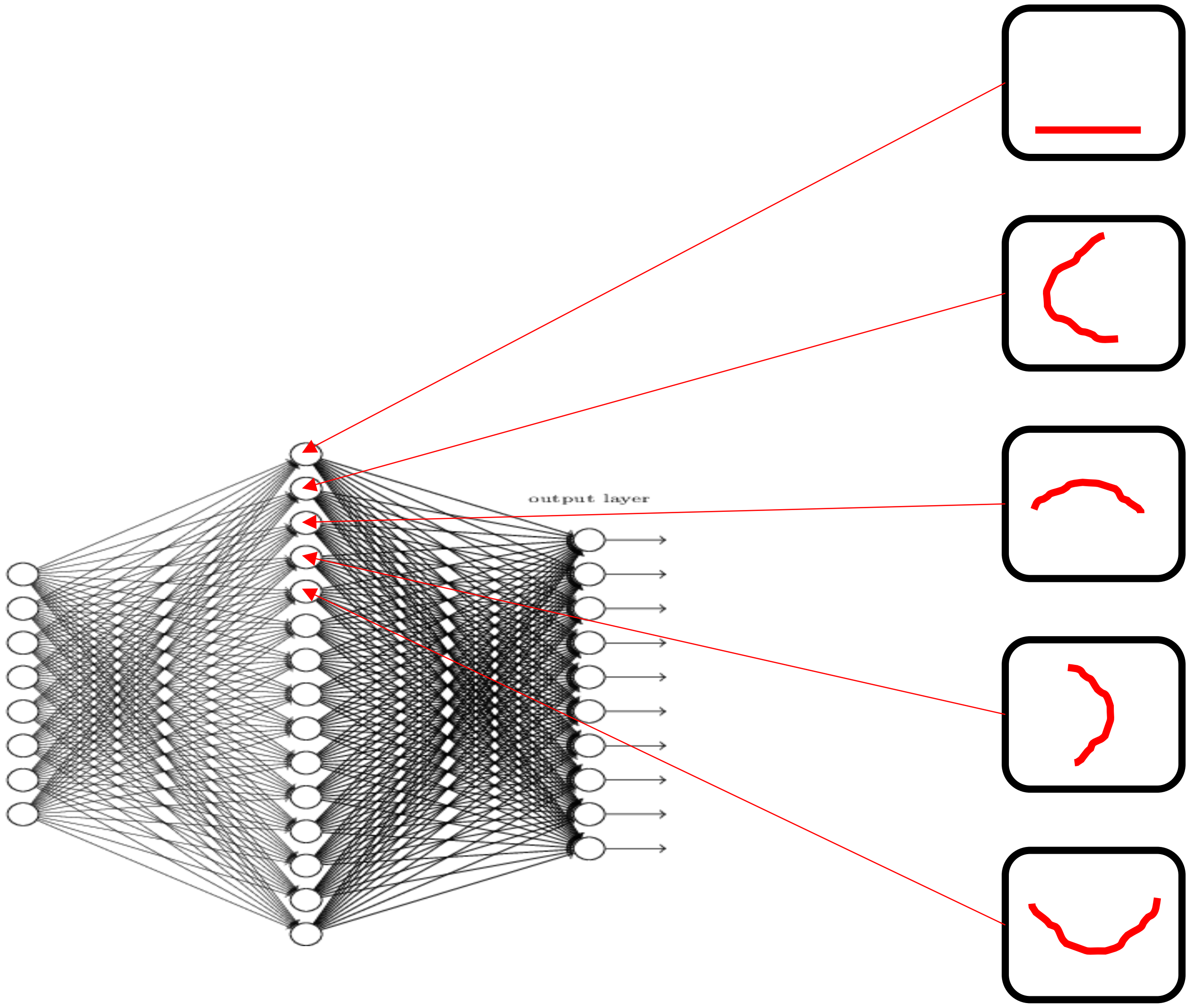
- A bias for each of 809 neurons
- A weight for each input in the hidden and output layers:
  - 784 inputs to each of 15 “hidden” neurons = 11,760 numbers
  - A weight for each of 15 inputs to each of 10 output neurons = 150 numbers
- ➔ Compute the optimal value for each of 809 + 11,760 + 150 = very many numbers

How to train a neural network to recognize handwriting

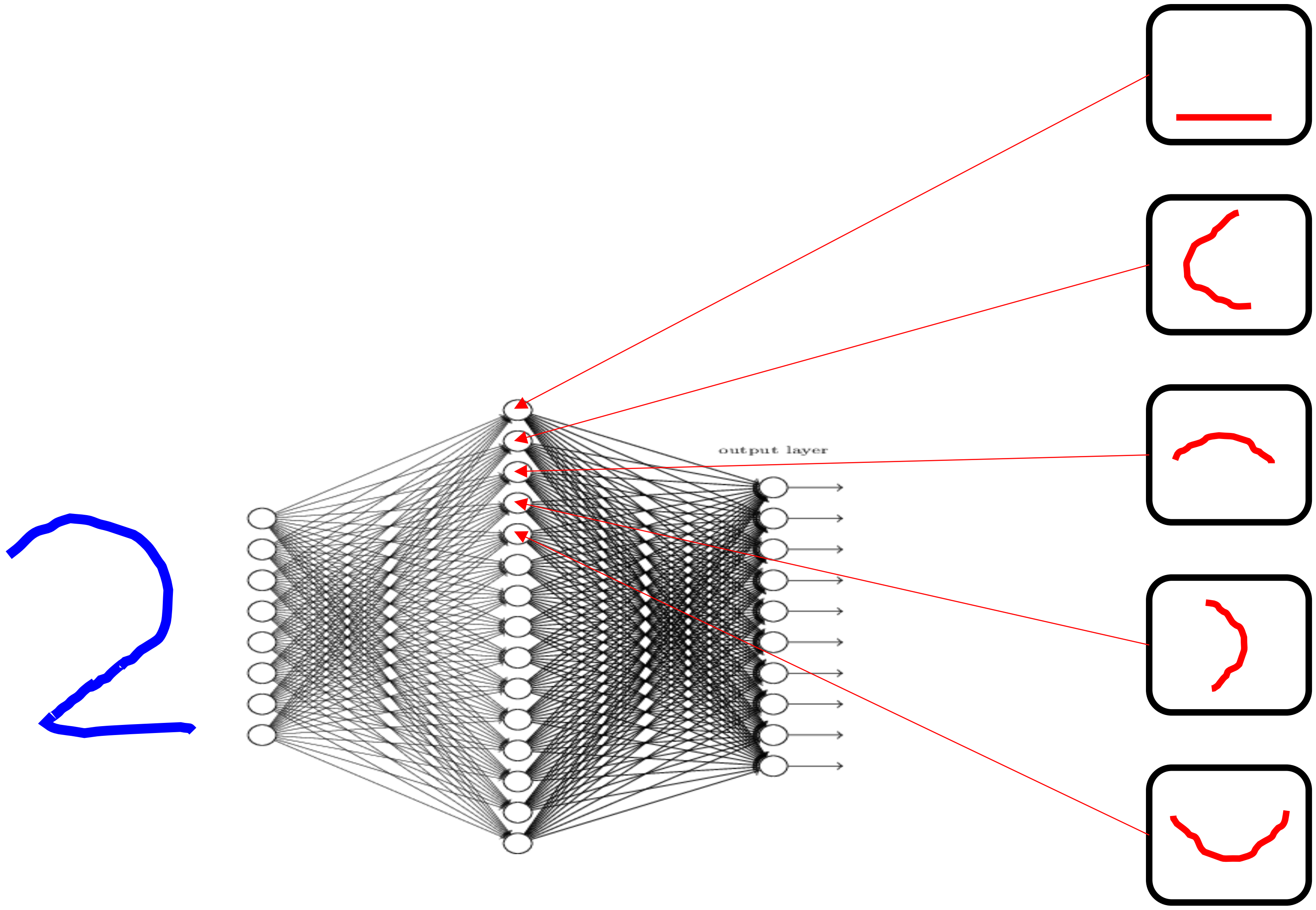


- Initialize params to random
- Expose to lots of 0s (“These are zeroes”), 1s (“These are ones”), etc
- Adjust params based on response to training examples

Just 1 of many possible training outcomes

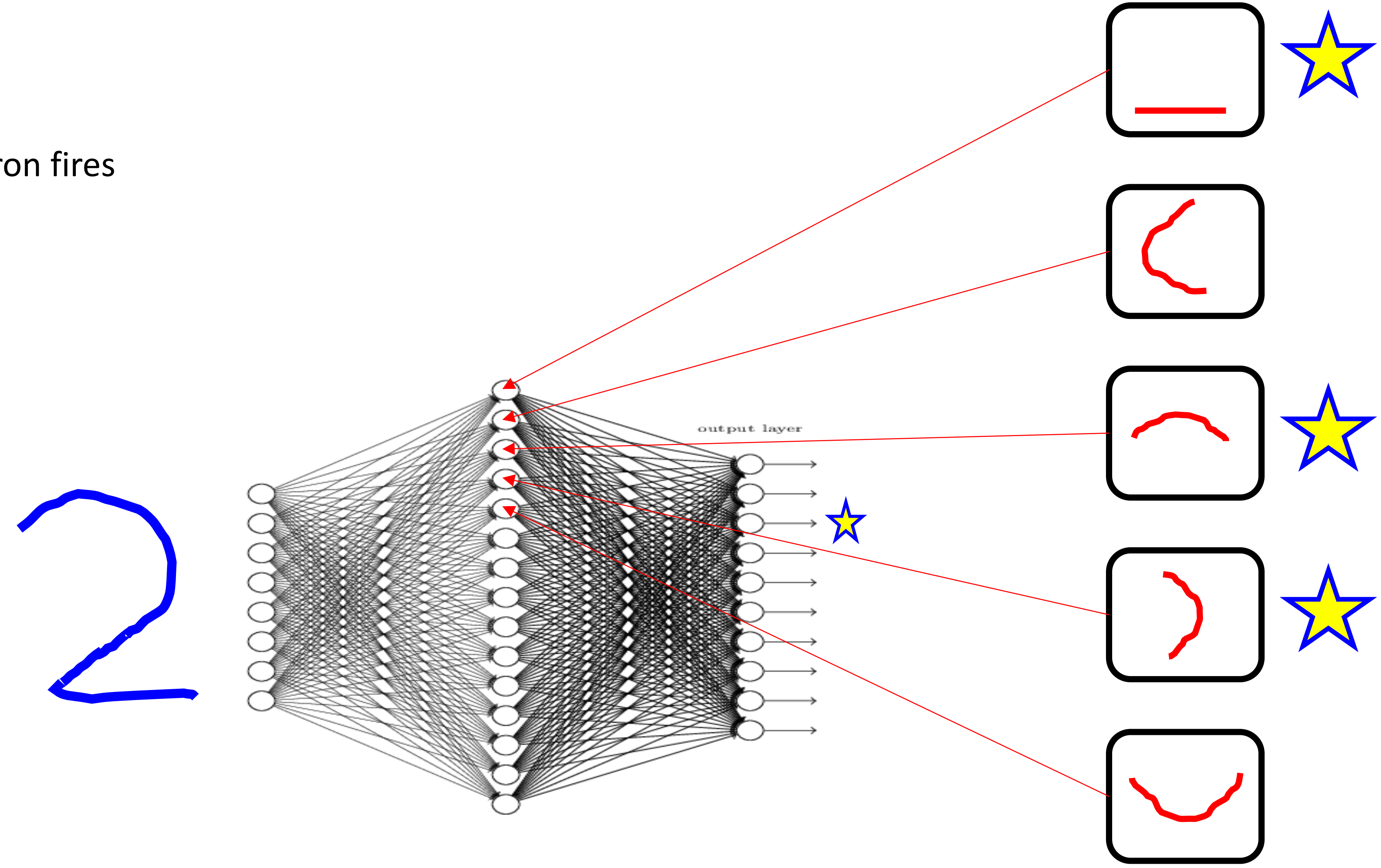


When this network “sees” a 2 …

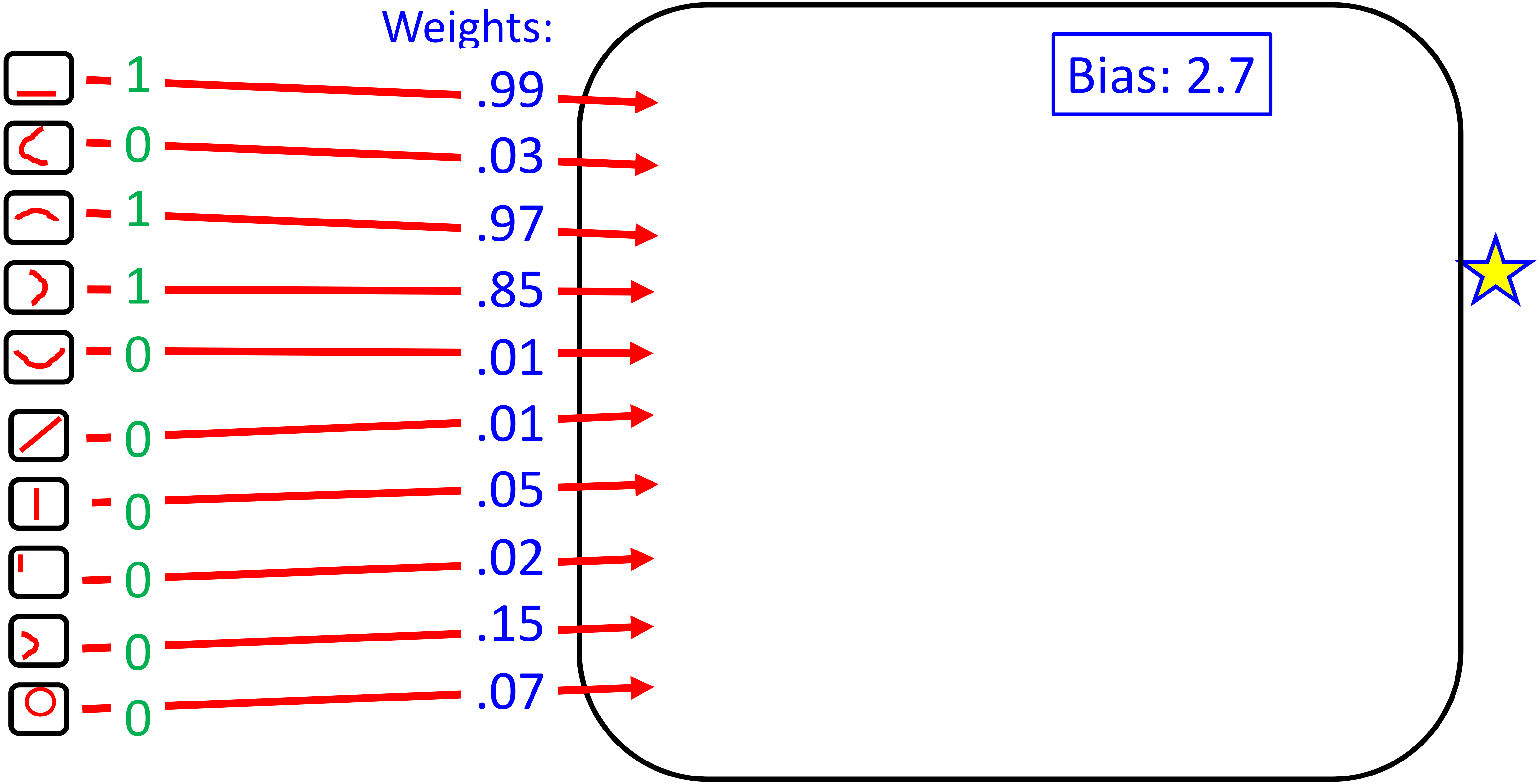


When this network “sees” a 2 …

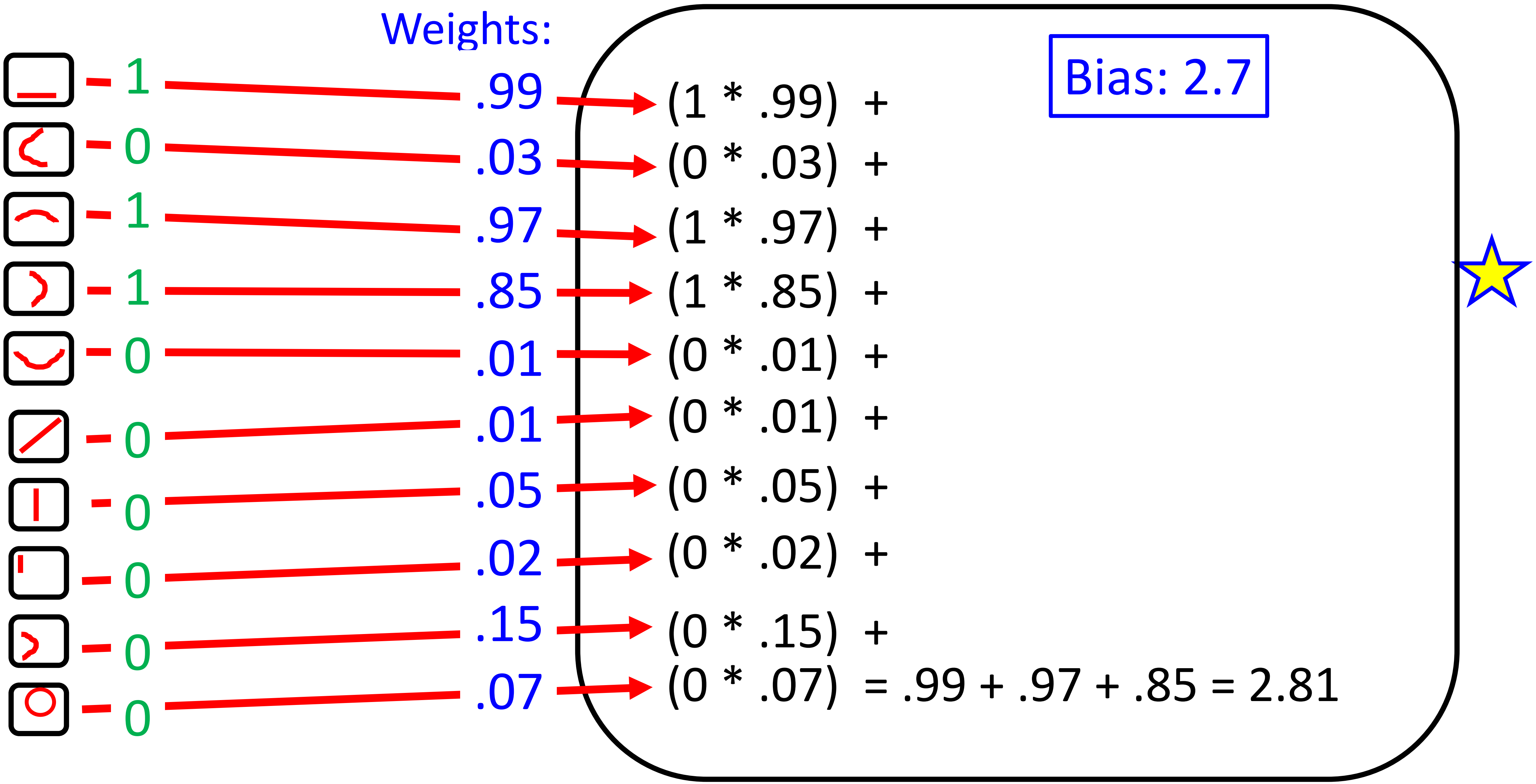
= neuron fires



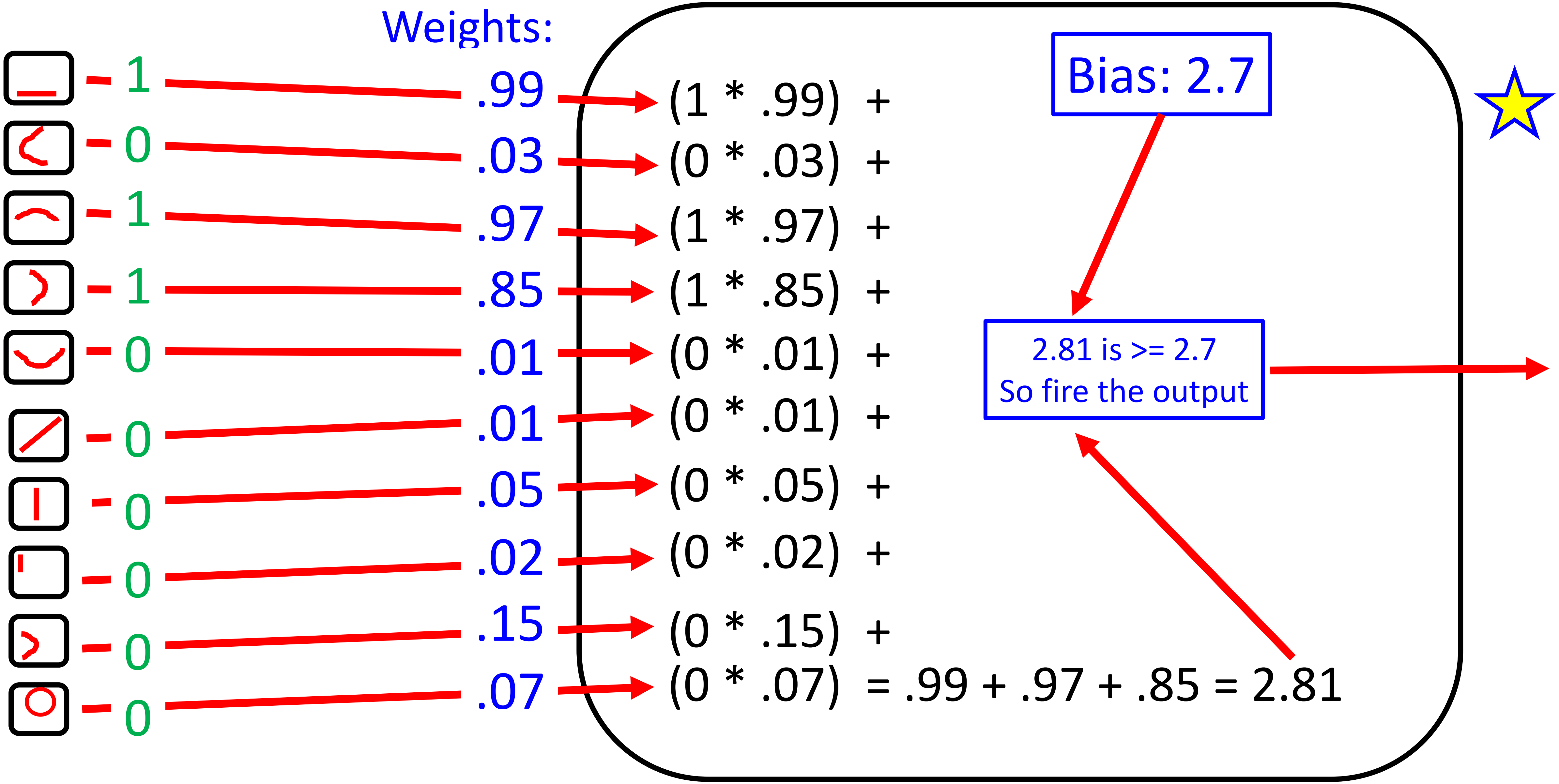
The “I think it’s a 2” output neuron



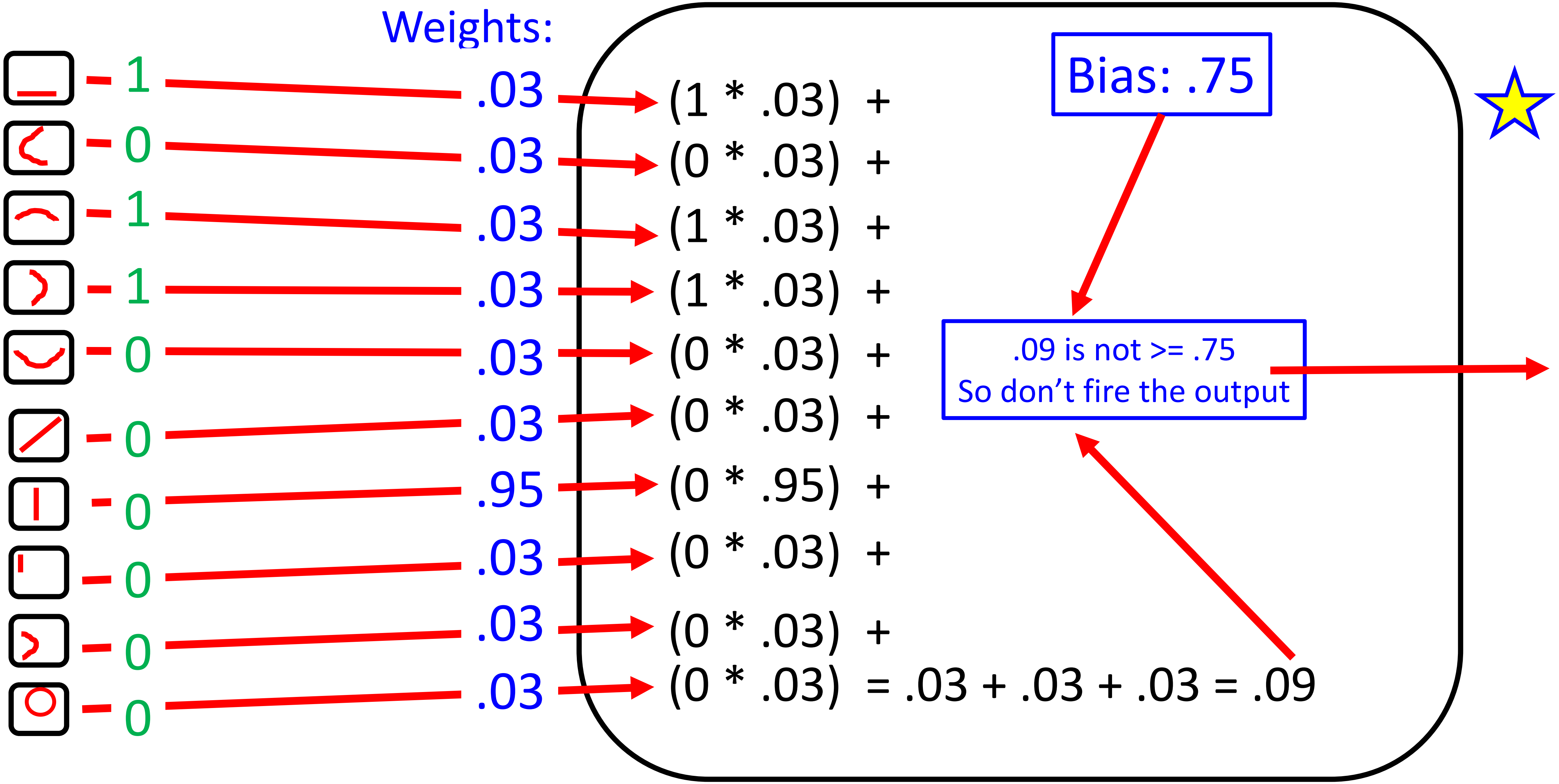
The "I think it's a 2" output neuron



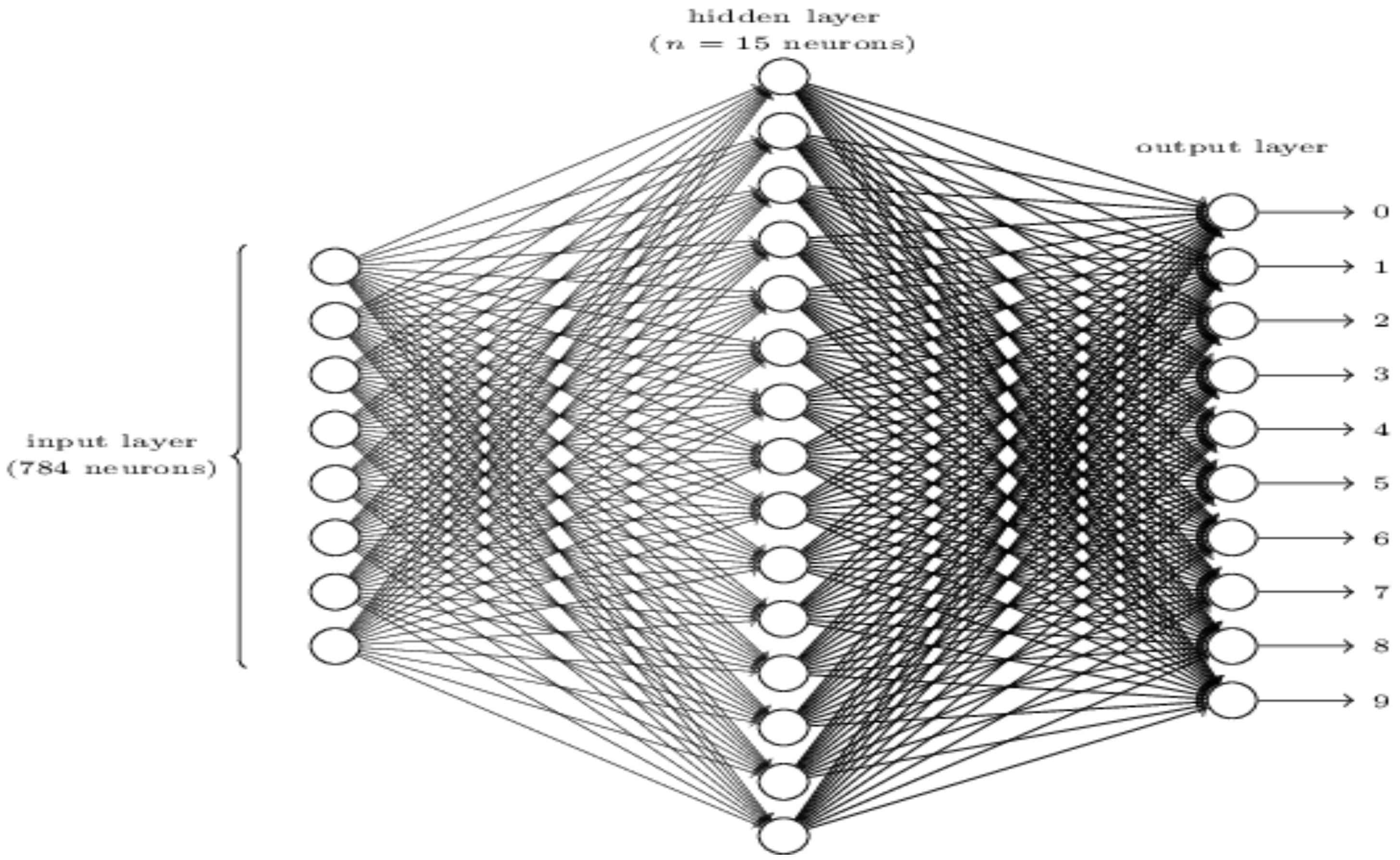
The "I think it's a 2" output neuron



The “I think it’s a 1” output neuron

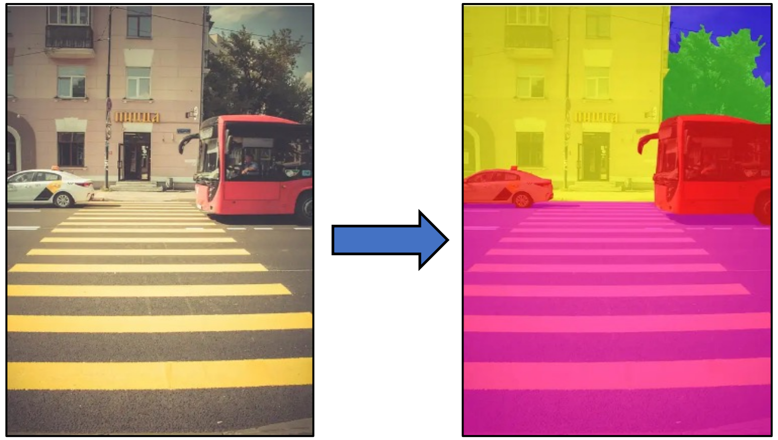


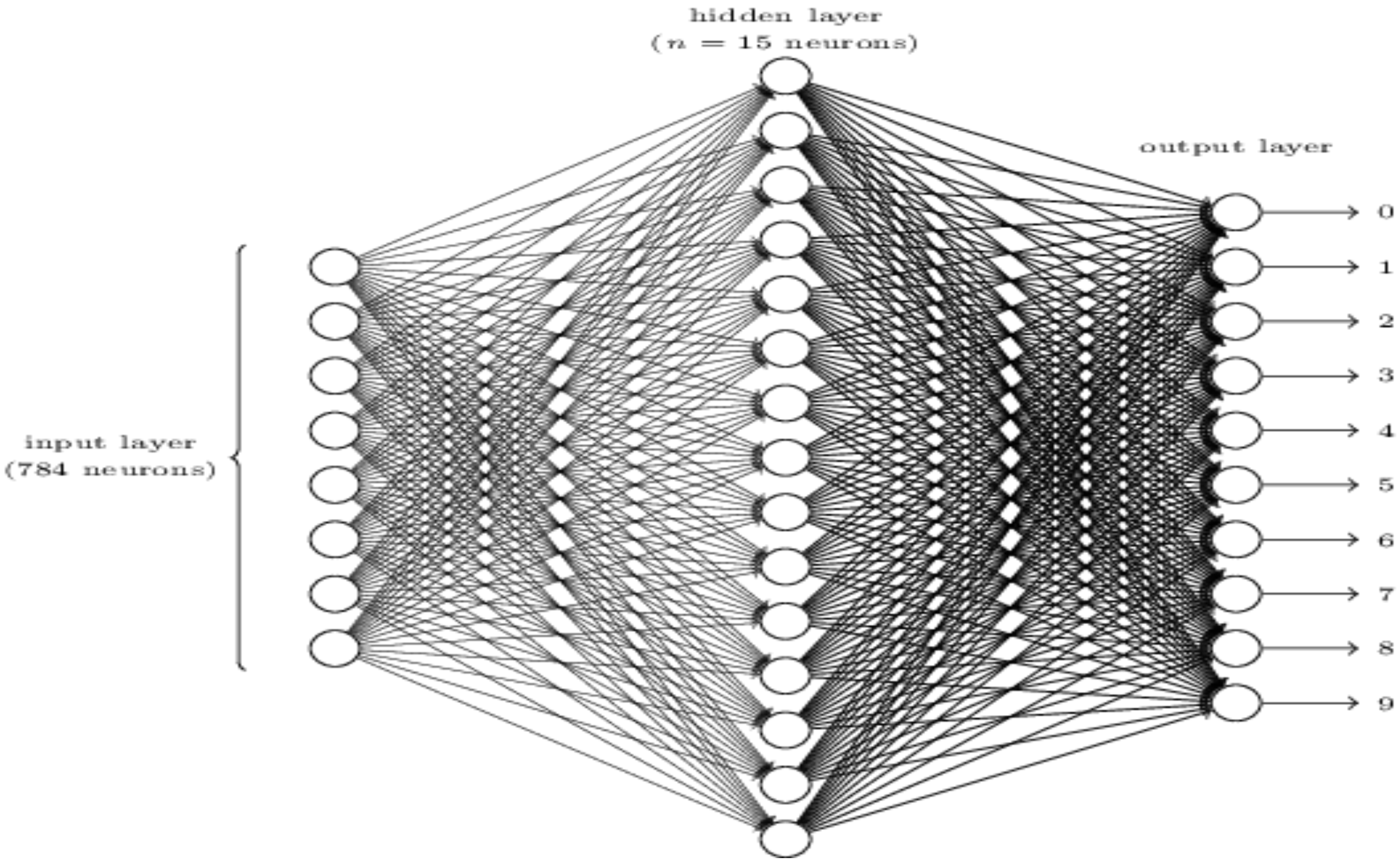
Deep Learning means >=1 hidden layer
Modern DL networks have dozens



Input Layer Hidden Layer Output Layer

Segmentation neural networks: a variation of what you just saw





For each discovered object in an image, segmenters output 2 pieces of data:

- The set of pixels that are the object
- Confidence (0-100%)
- It’s time for a demo