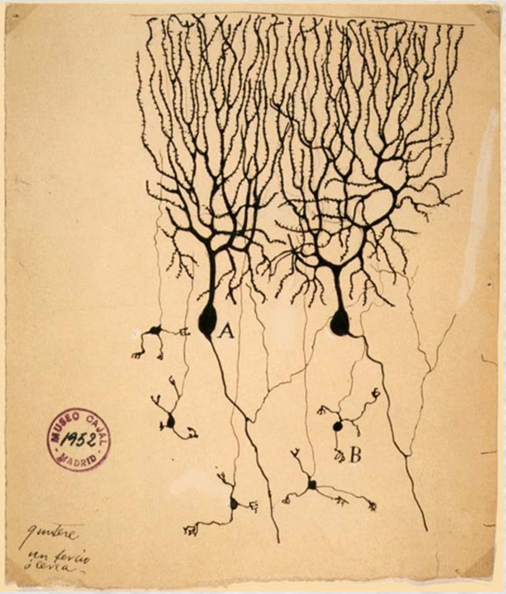
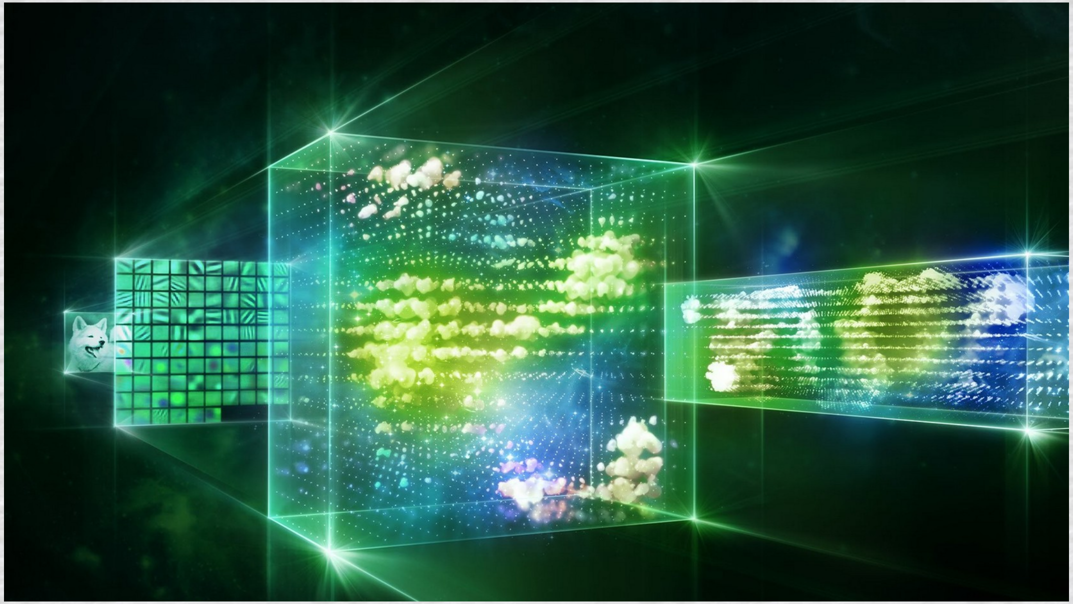


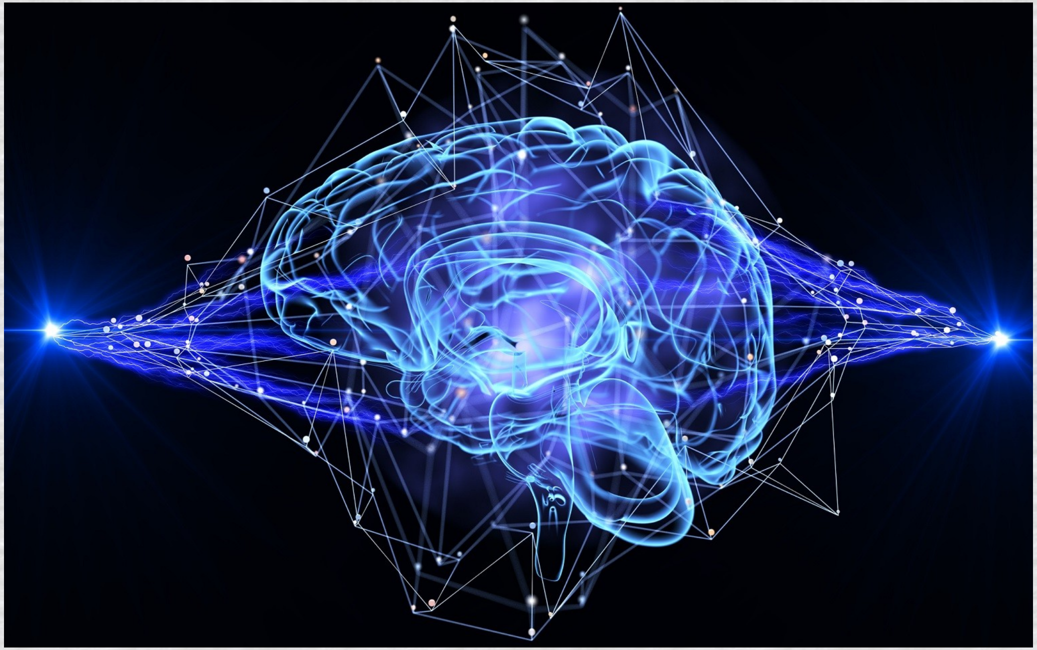


How To Build A Brain

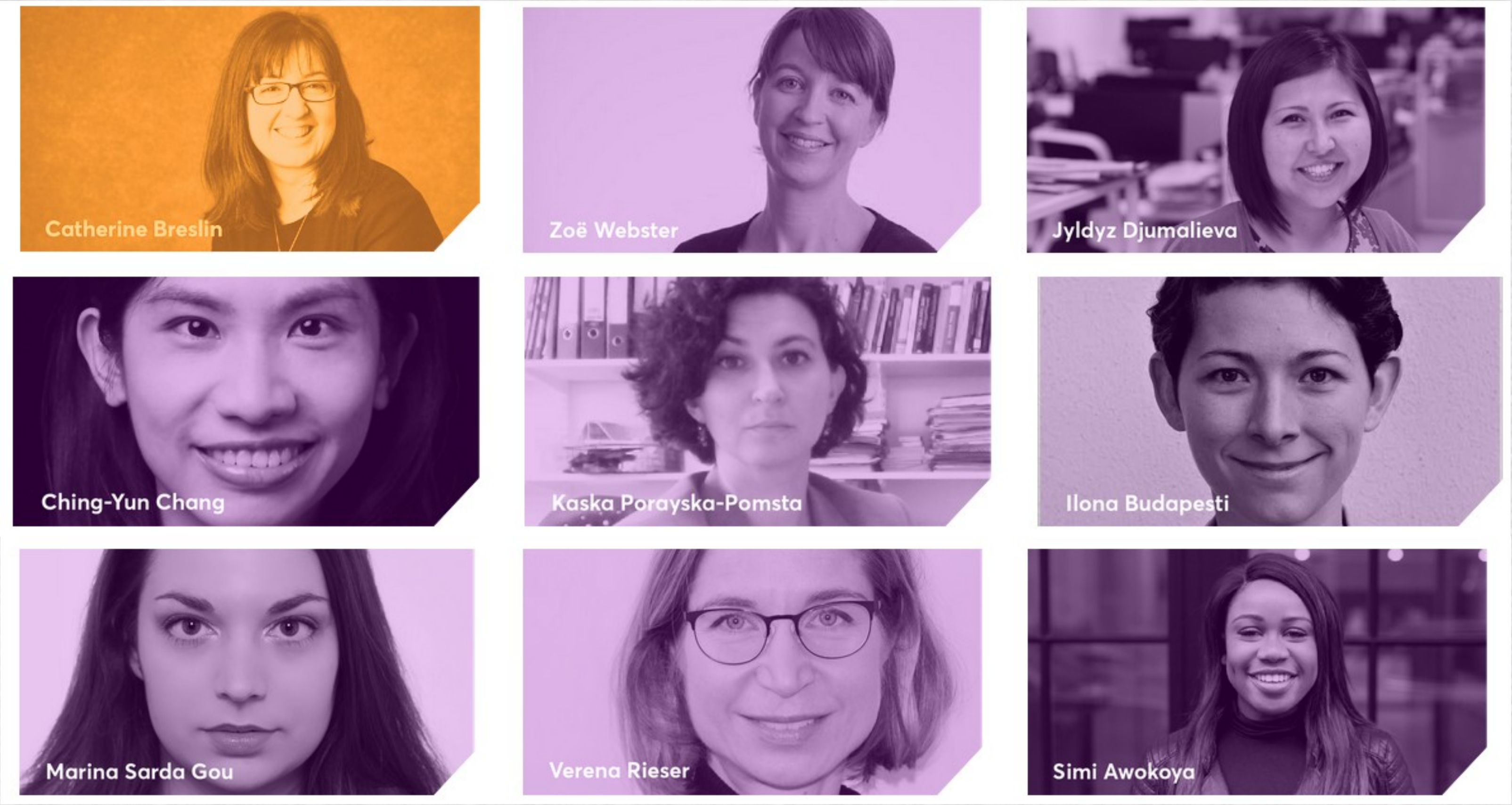
History And Core Concepts

Goals

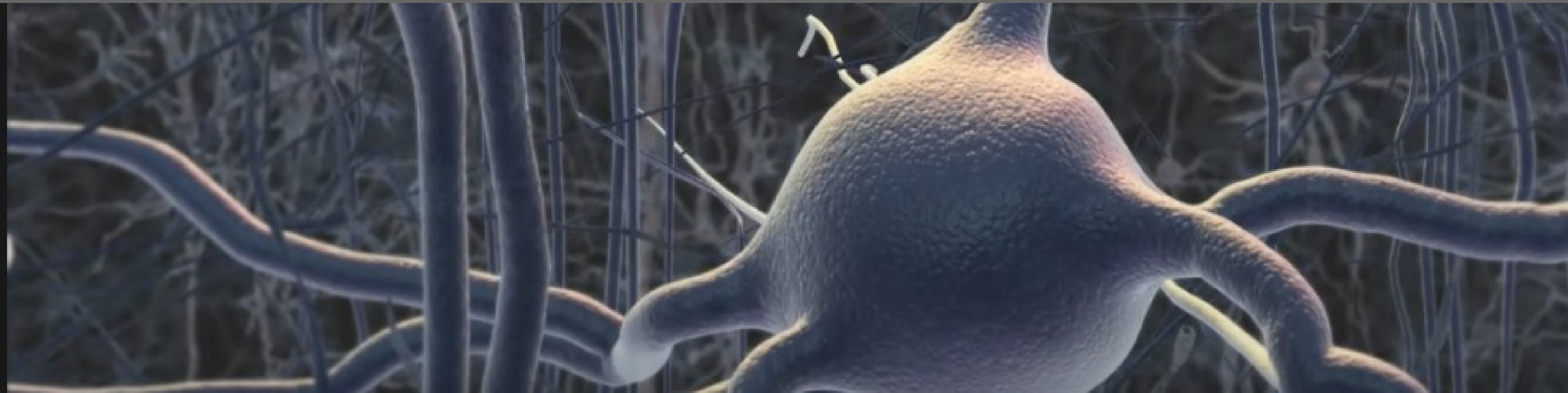
To help you understand the history and core concepts of brain-inspired artificial intelligence.



AI: Not just for old white men

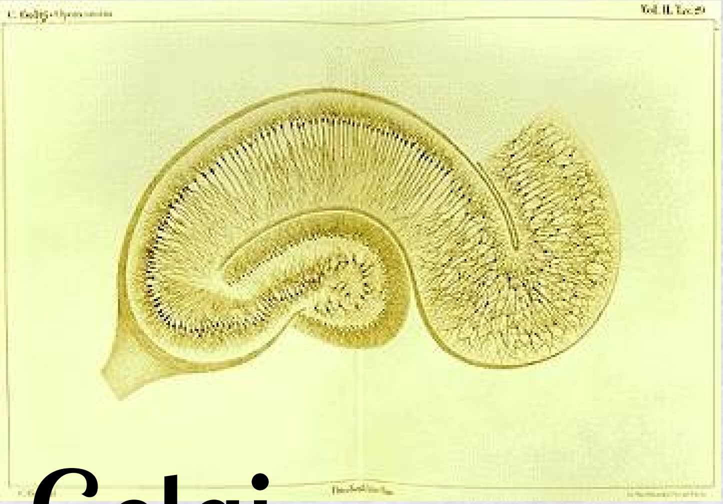


Early history

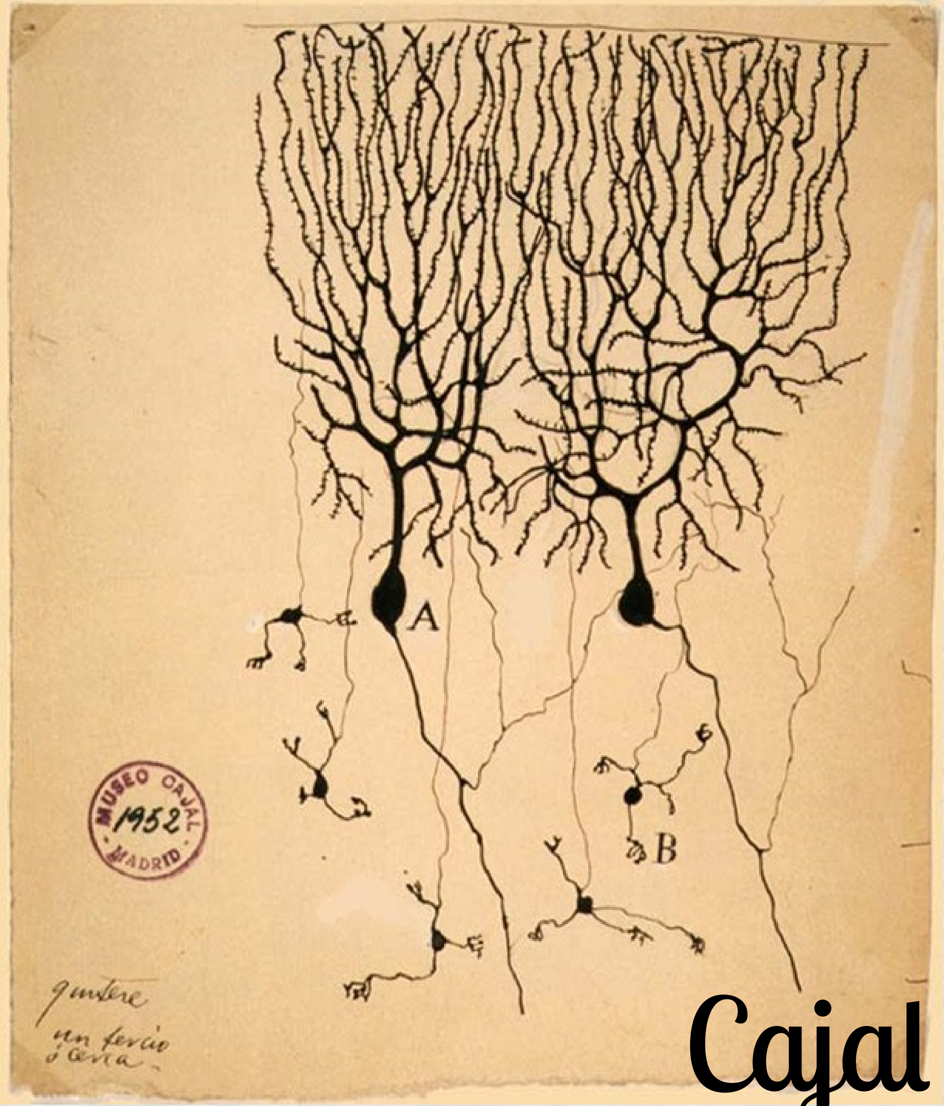


400BC - 1900AD

Neurons

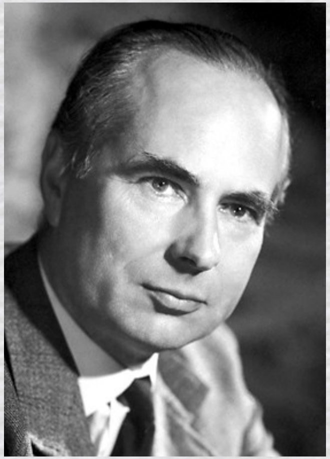


Golgi

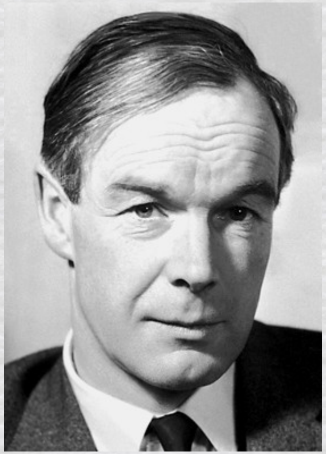


Cajal

Plymouth
Marine Laboratory



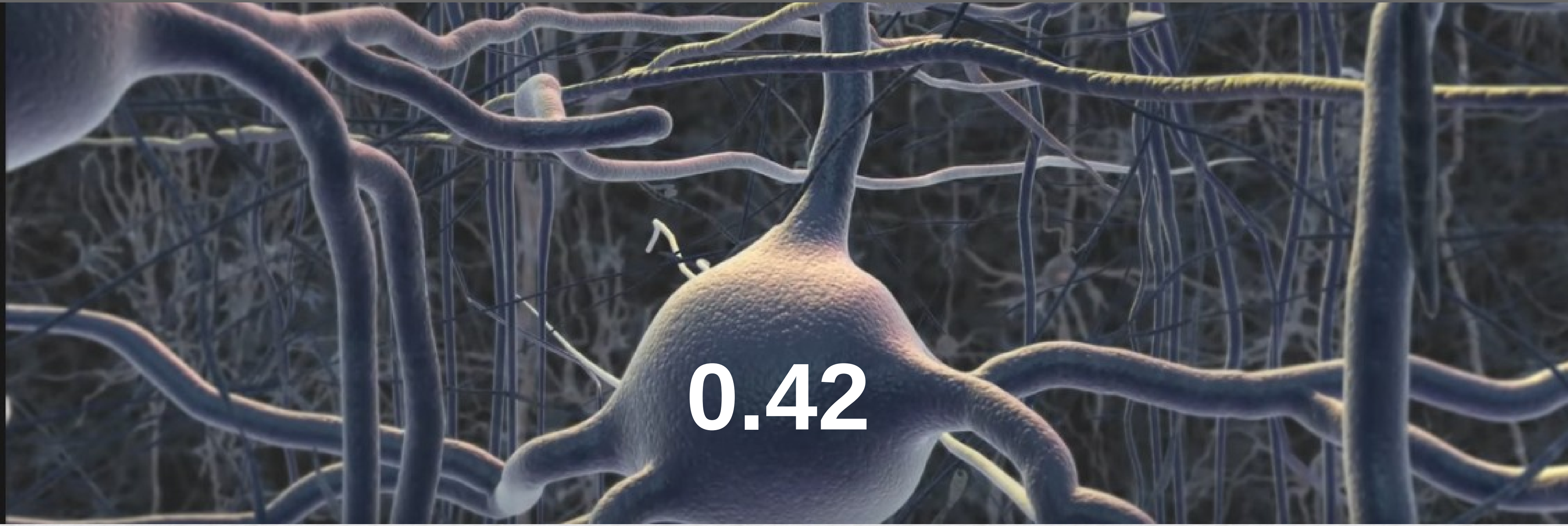
Andrew
Huxley



Alan
Hodgkin

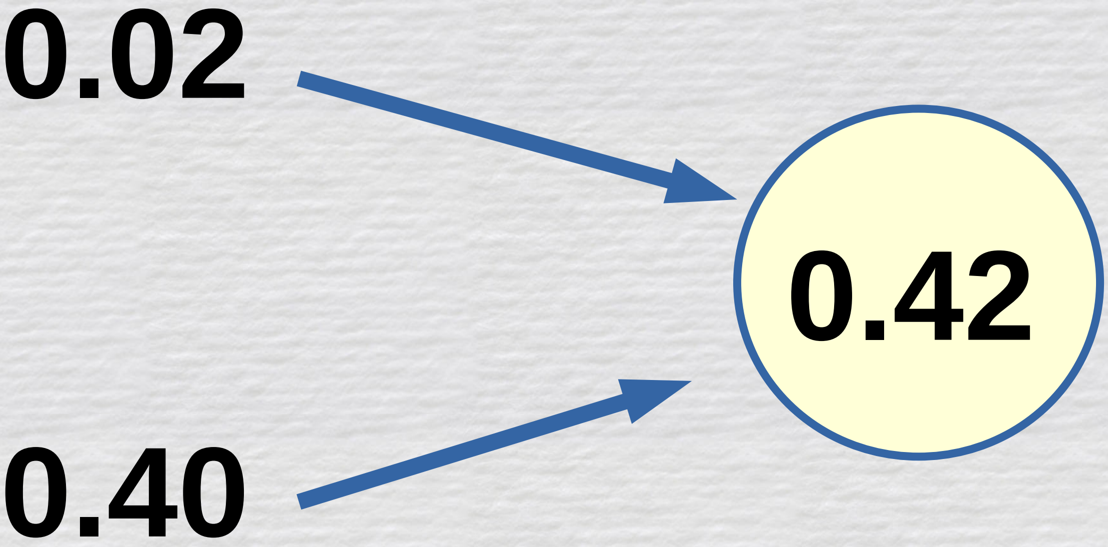
Three abstractions

Output Activation



The activity of a neuron, as a single number

Output Activation



Out = 0.02 + 0.40 = 0.42

Out = Σ in

Output activation is the sum of the inputs.

Input activation

$$a_{out} = 0.2 \times 0.1 = 0.02$$

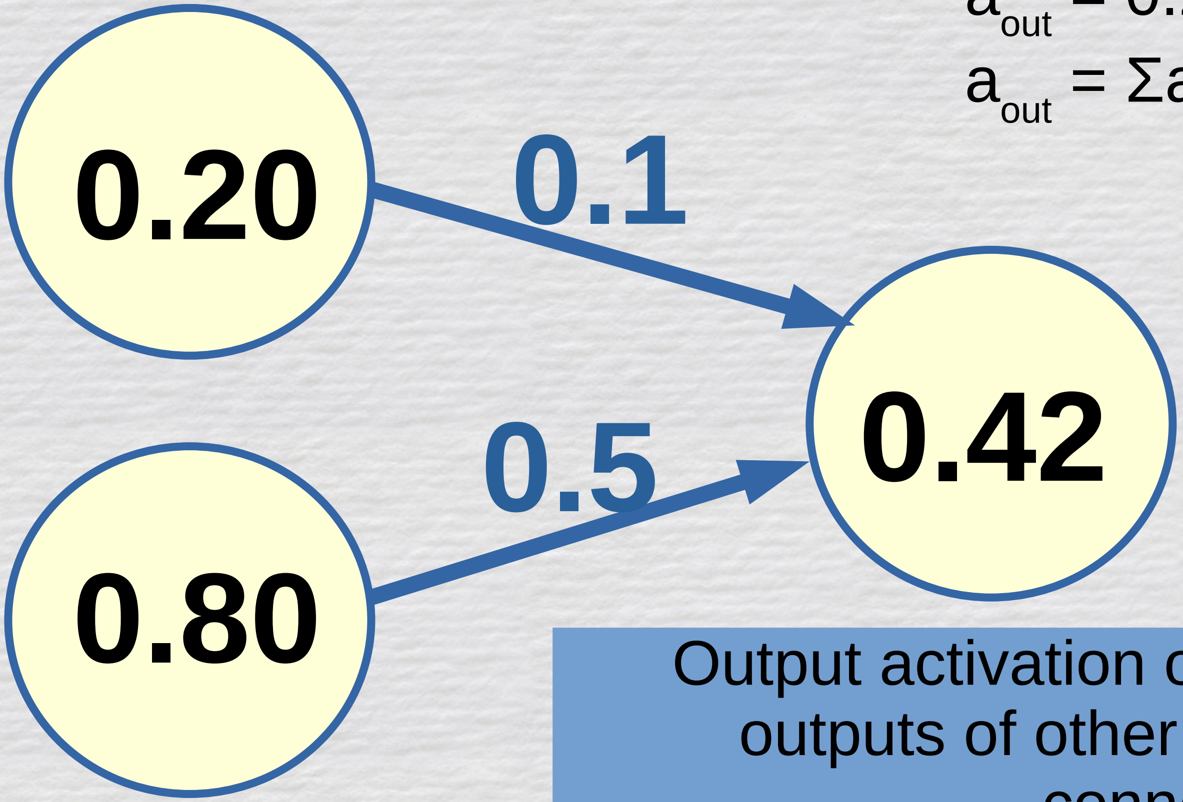
$$a_{out} = a_{in} w$$

Input activation = output activation multiplied by the connection strength

Putting it together

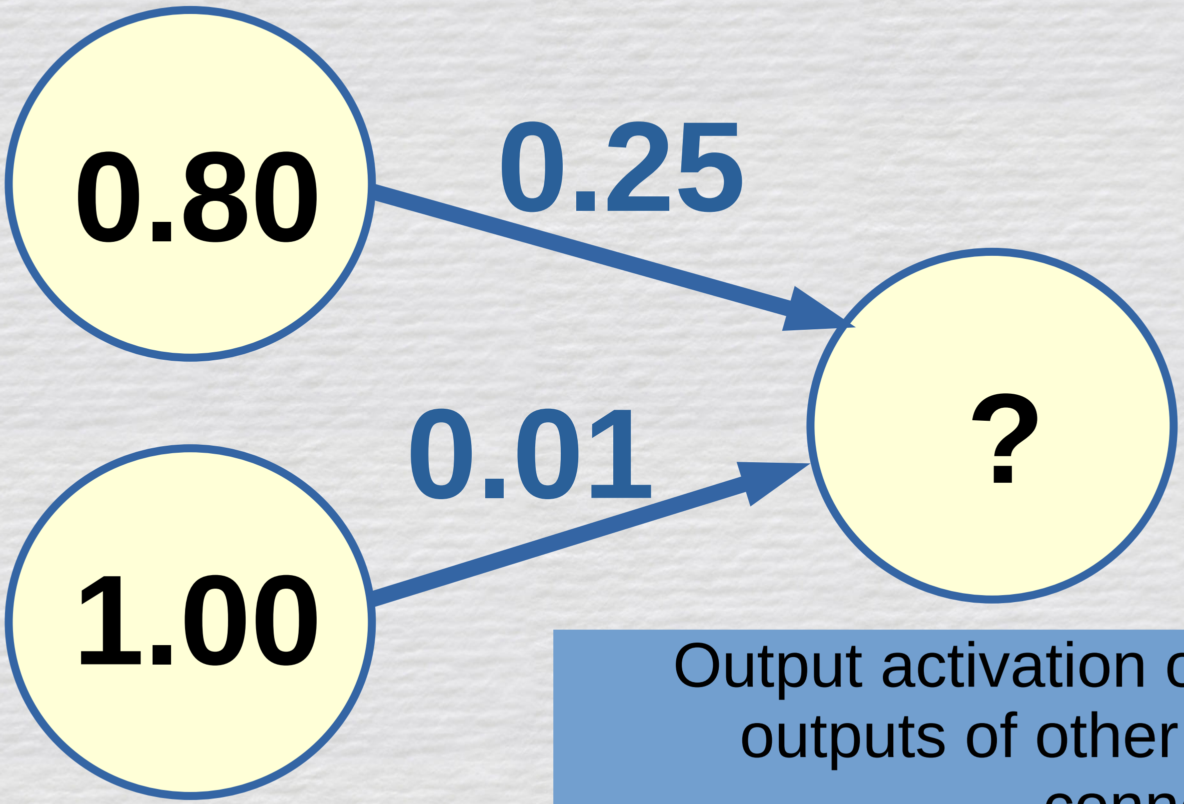
$a_{out} = 0.2 \times 0.1 + 0.8 \times 0.5 = 0.42$

$a_{out} = \Sigma aw$



Output activation of a neuron is the sum of the outputs of other neurons multiplied by the connection weights.

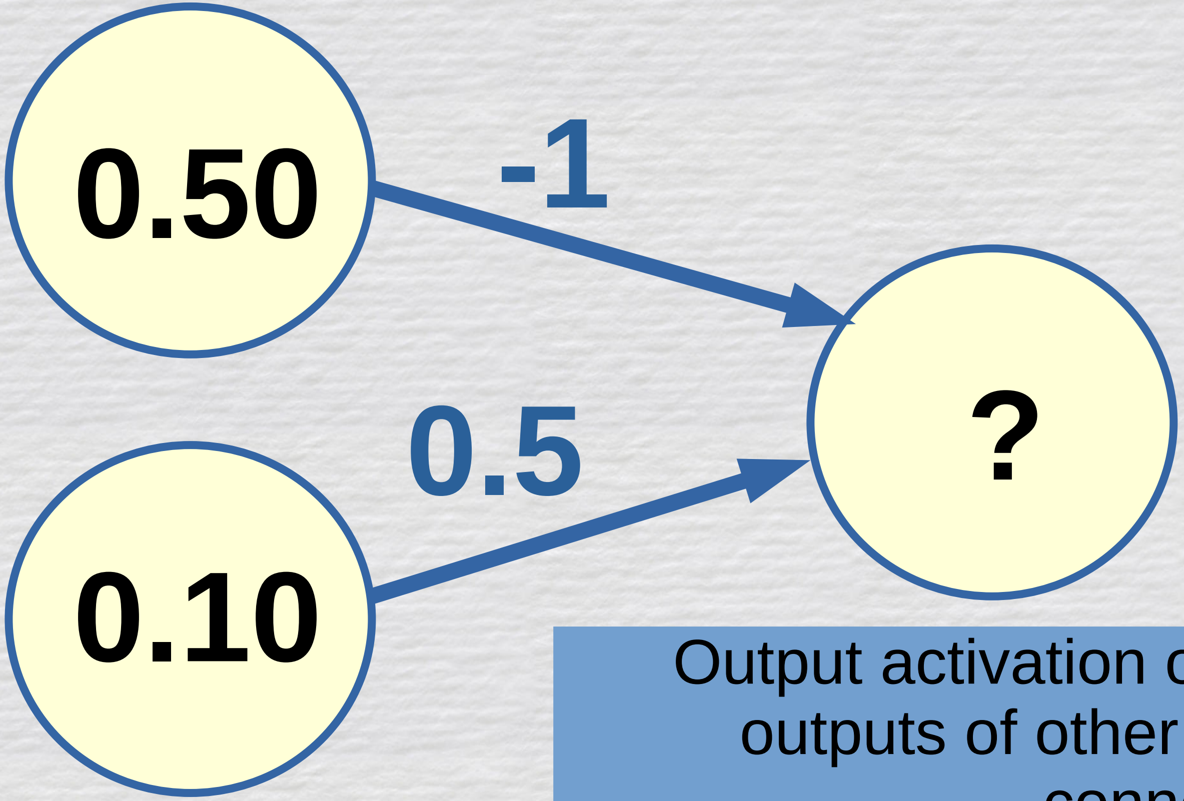
Test your understanding...



$a_{out} = \Sigma aw$

Output activation of a neuron is the sum of the outputs of other neurons multiplied by the connection weights.

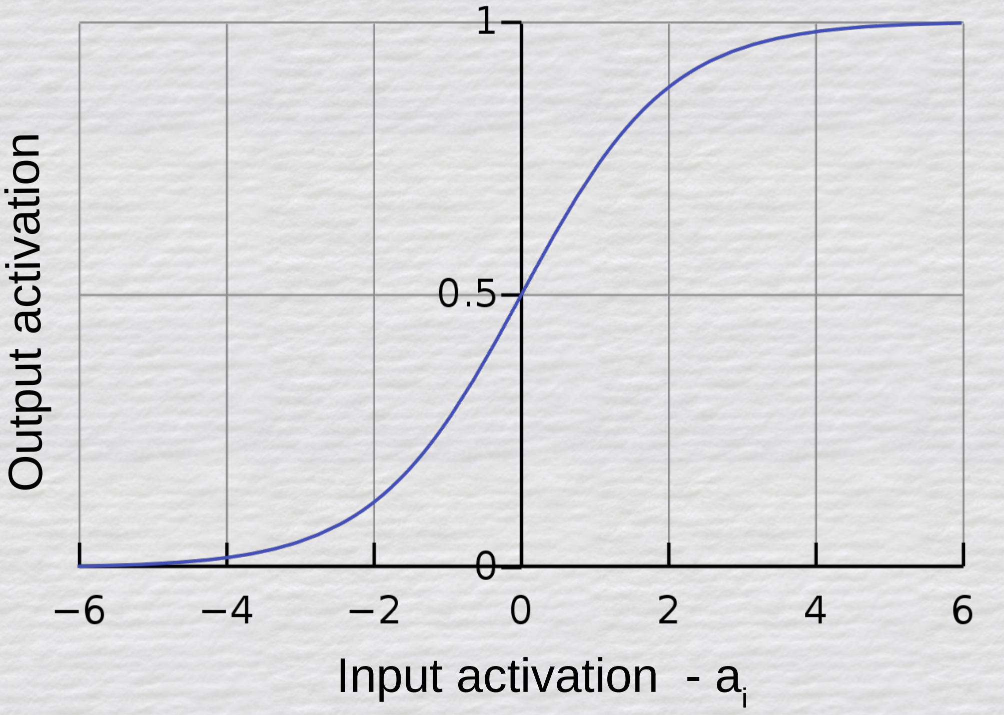
Test your understanding (2)



$a_{out} = \Sigma aw$

Output activation of a neuron is the sum of the outputs of other neurons multiplied by the connection weights.

Activation function

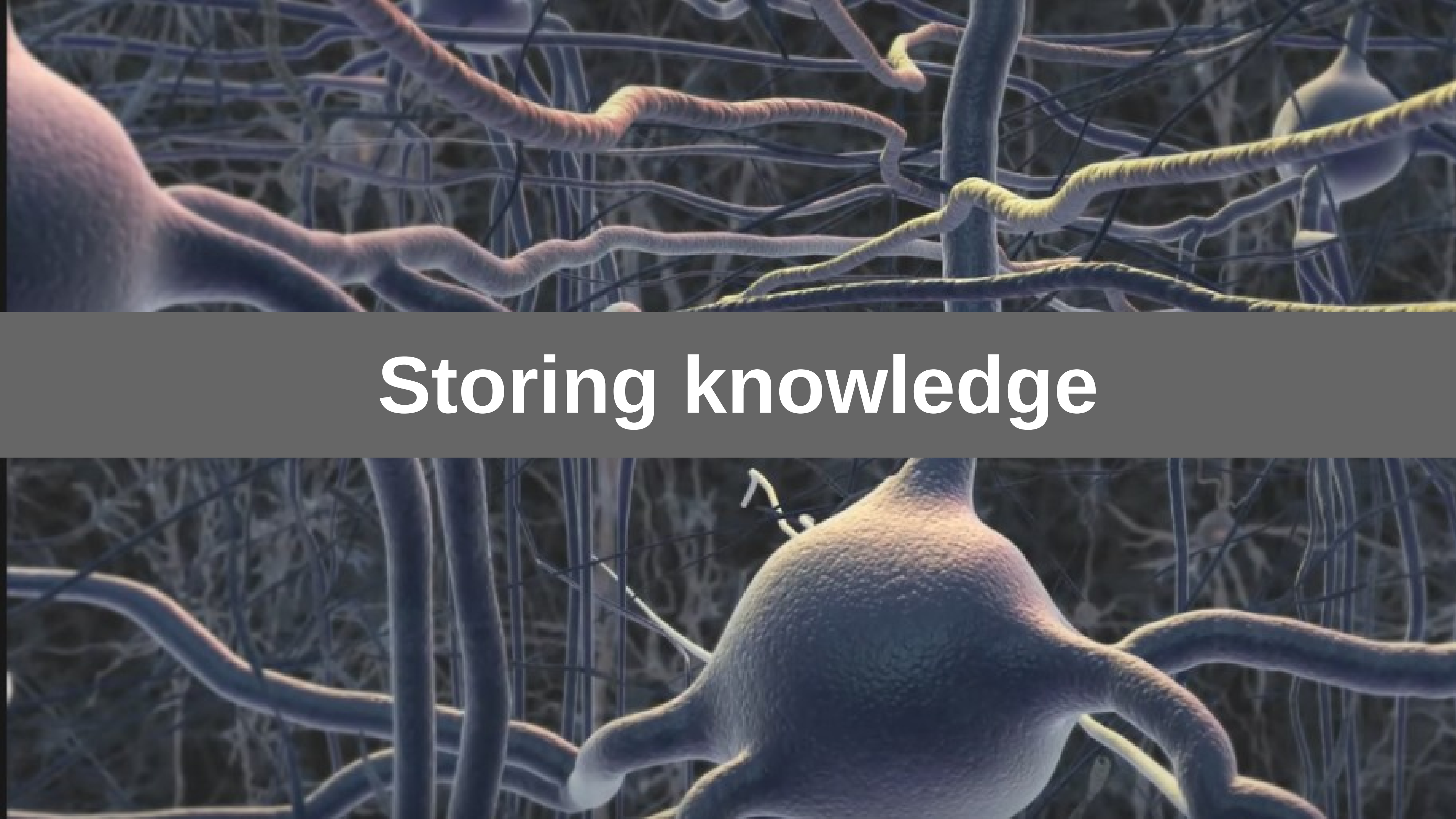


$$a_{out} = \frac{1}{1 + e^{-ka_i}}$$

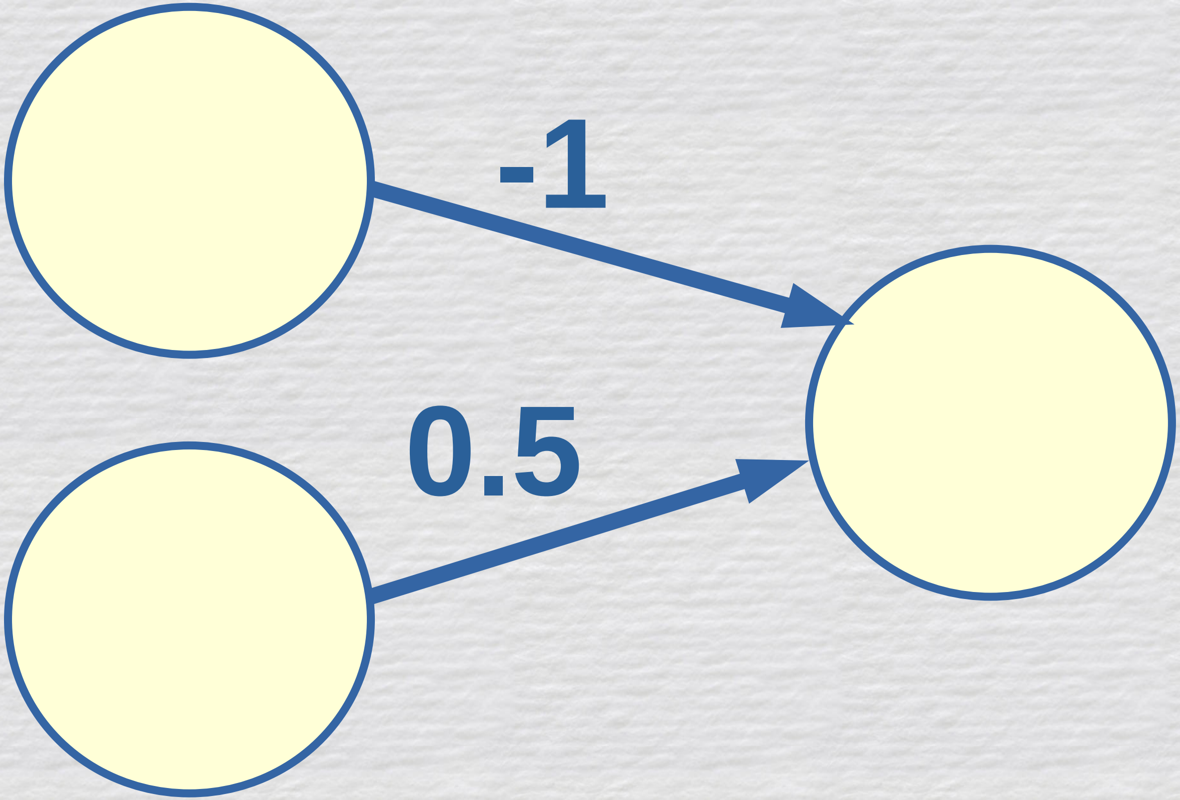
e = ~ 2.72

The logistic function in useful in various ways.

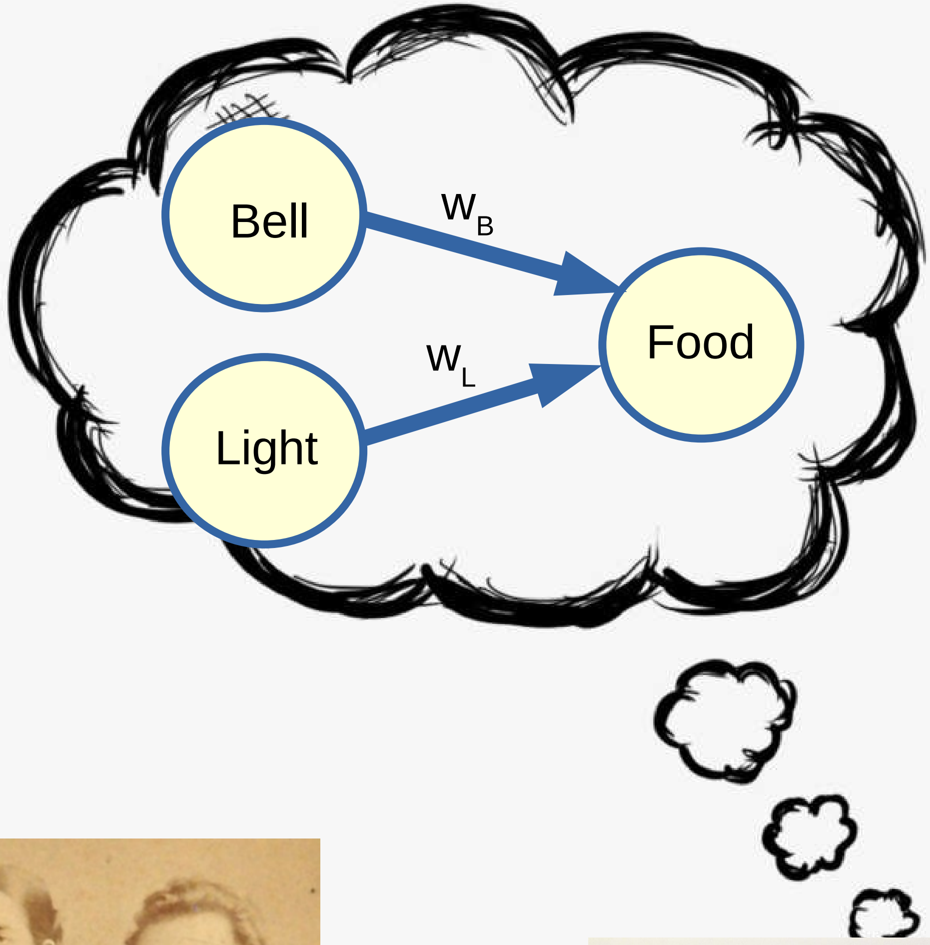
Storing knowledge



Neural Network



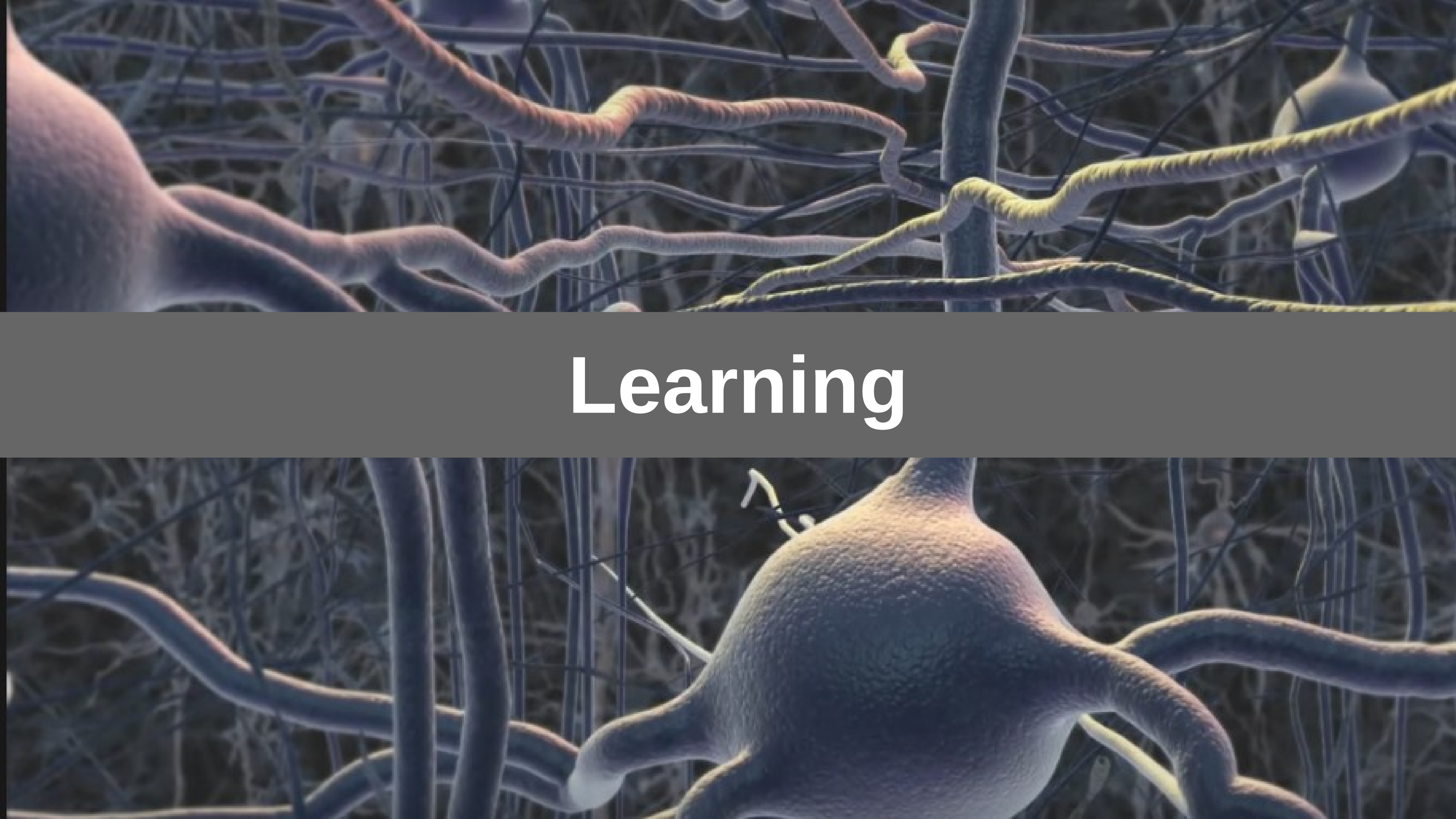
Neural Network



Bell	Light	Food
1	0	1
1	1	0

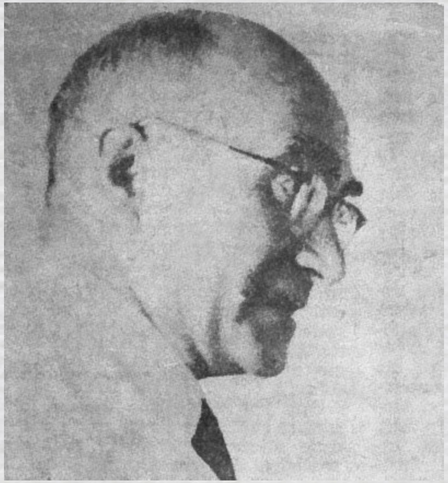
What should w_B and w_L be?

Learning



Learning

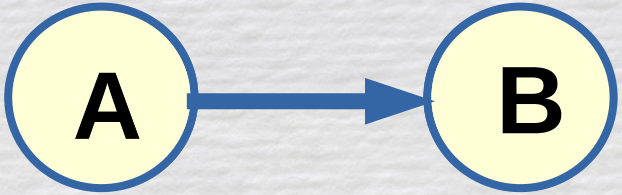
Cajal
Spain, 1894



Konorski
Russia, 1948

Hebb
USA, 1949

Hebbian Learning



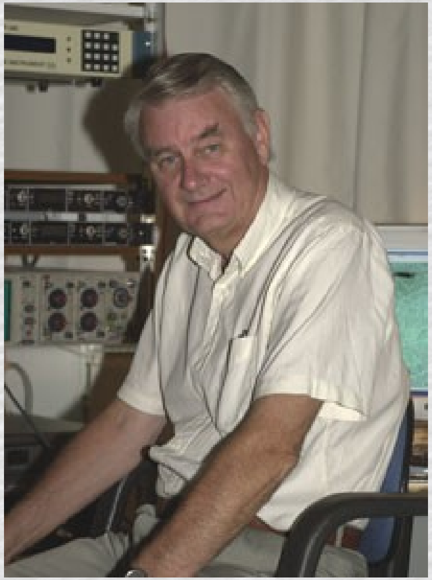
"When an axon of cell A is near enough to excite cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased"

Hebb

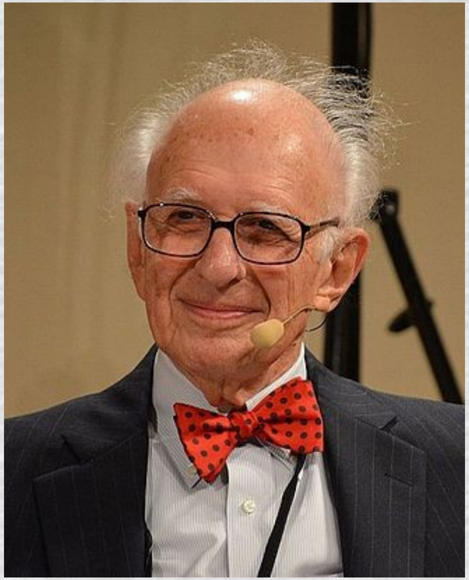
Neurons that fire together wire together

Long-term potentiation

Neurons that fire together wire together



Terje Lømo

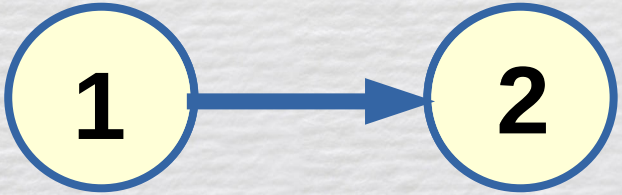


Eric Kandel

Aplysia Californica

Hebbian Learning

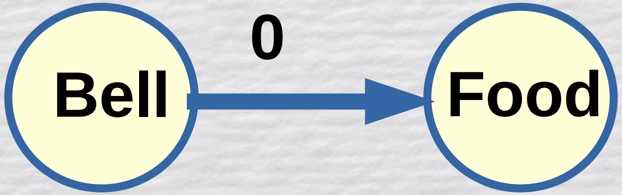
Neurons that fire together wire together



$$\Delta w_{12} = G \, . \, a_1 \, . \, a_2$$

Hebbian Learning

Neurons that fire together wire together

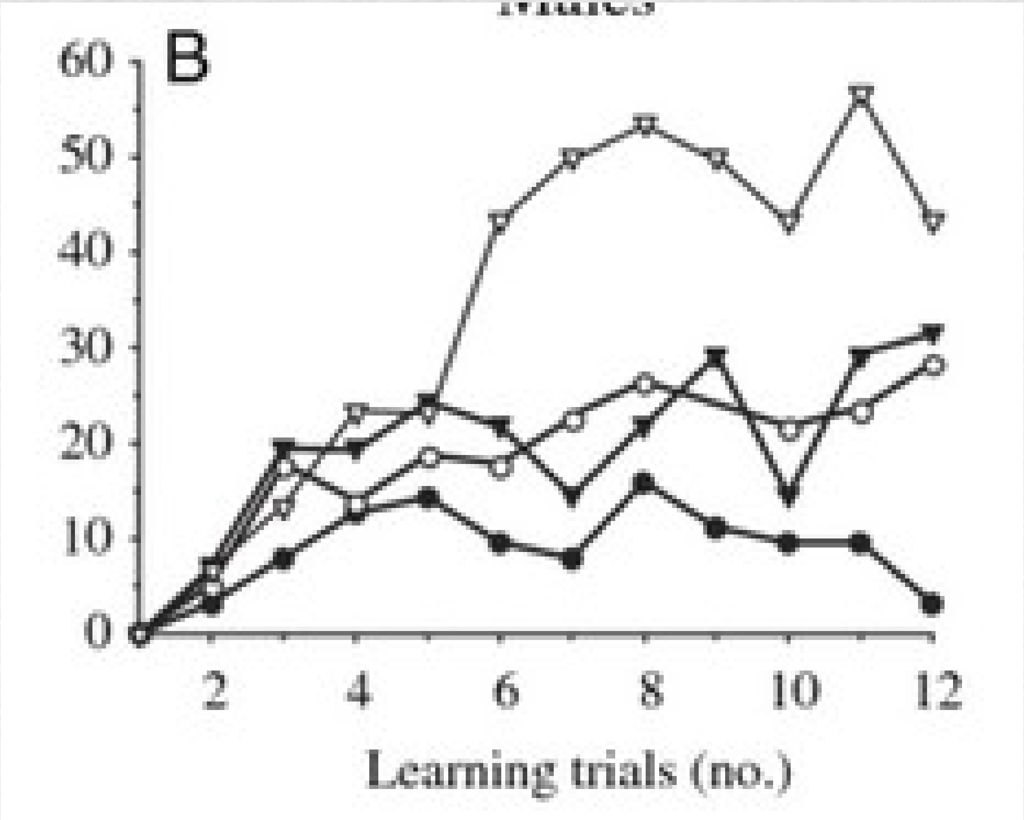


$$\Delta w_{BF} = G \,.\, a_B \,.\, a_F$$

G= 0.1

Trial	Bell	Food	Δw	w
1	1	1	.1	.1
2	1	1	.1	.2
3...	1	1	.1	.3
...1000	1	1	.1	10 (!)

Acquisition

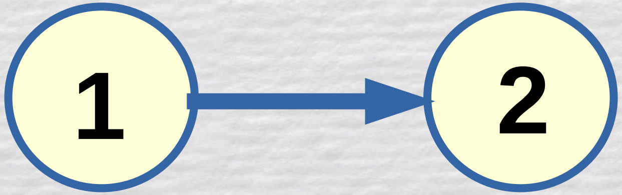


Learning tends to slow as it proceeds.

Bush-Mosteller

The more you know, the less you learn.

Bush



$$\Delta w_{12} = G(t - a_1 w_{12})a_1$$

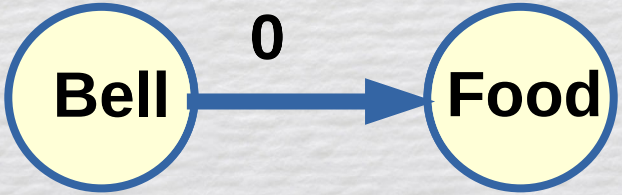
Teacher

Student prediction

Mosteller

Bush-Mosteller

The more you know, the less you learn.

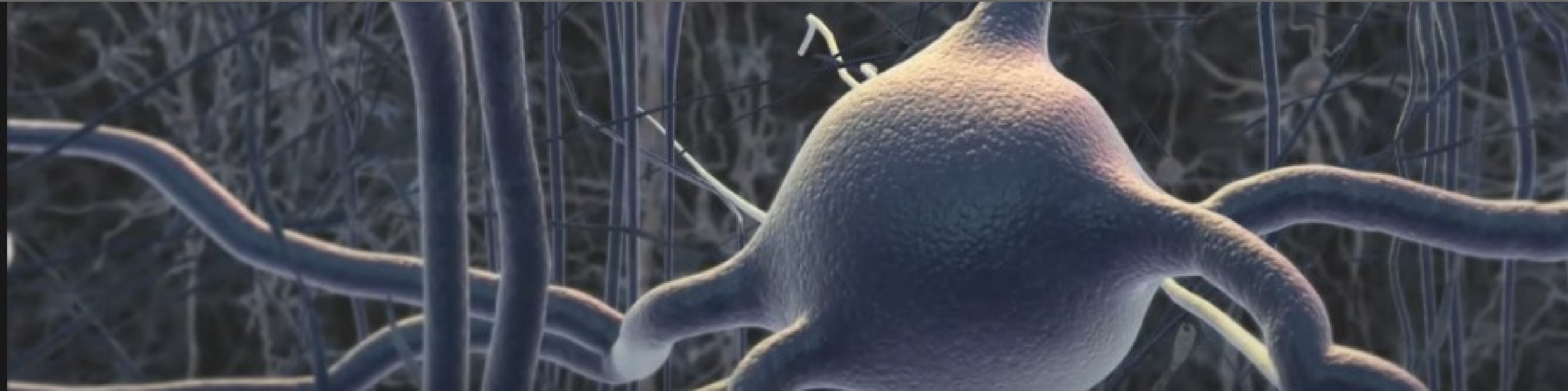


$$\Delta w_{12} = G(t - a_1 w_{12})a_1$$

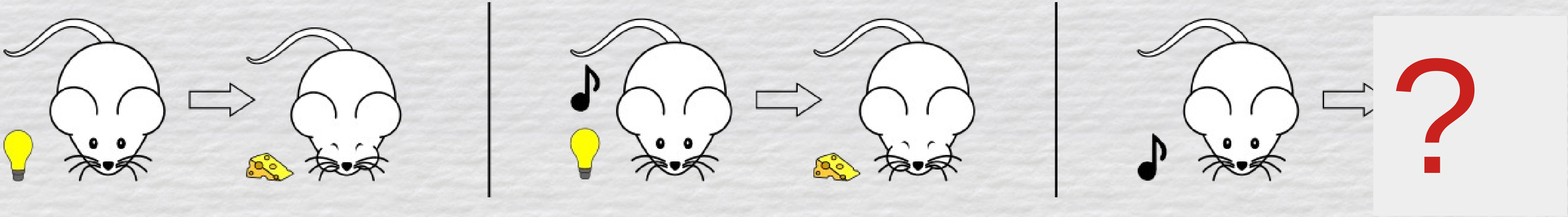
G= 0.1

Trial	Bell (a_1)	Food (t)	Δw	w
1	1	1	.1 x (1-0x1) = .1	.1
2	1	1	.1 x (1-.1x1) = .09	.19
3...	1	1	.1 x (1-.19) = .081	.271
...1000	1	1	.1 x (1-1) = 0	1

Delta rule



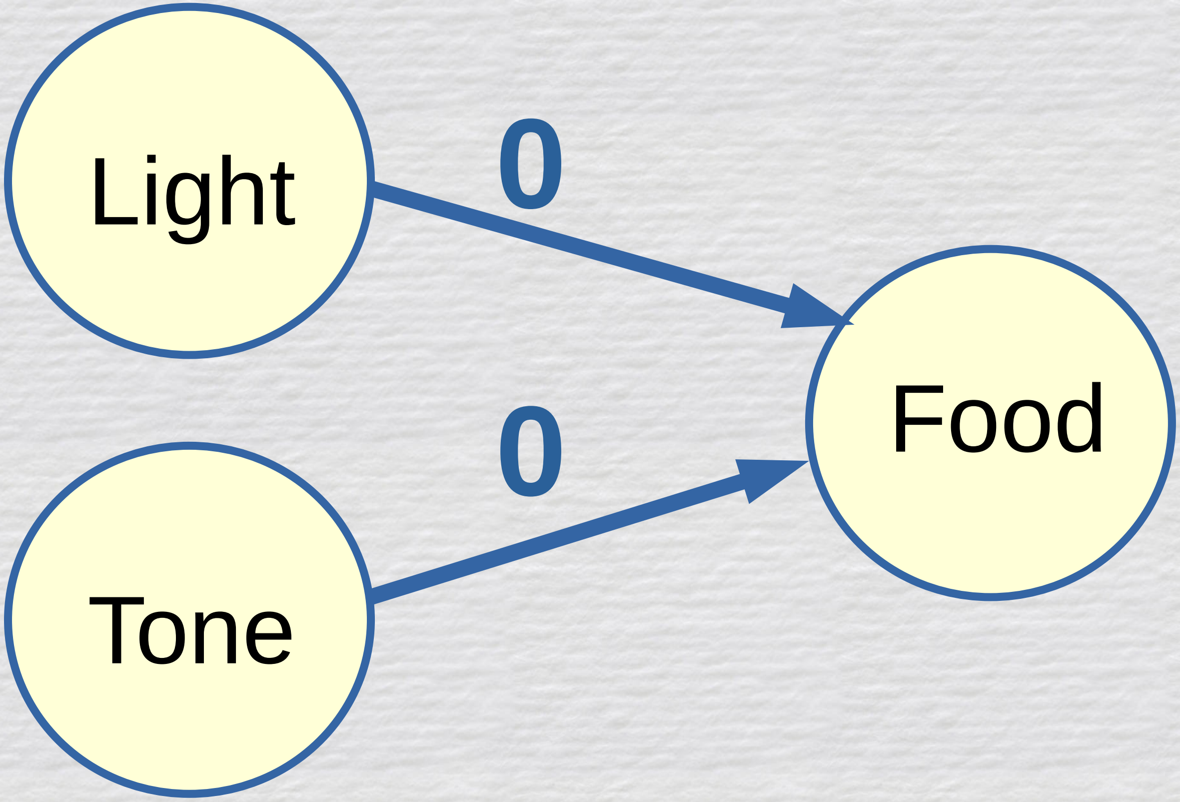
Blocking (Kamin, 1969)



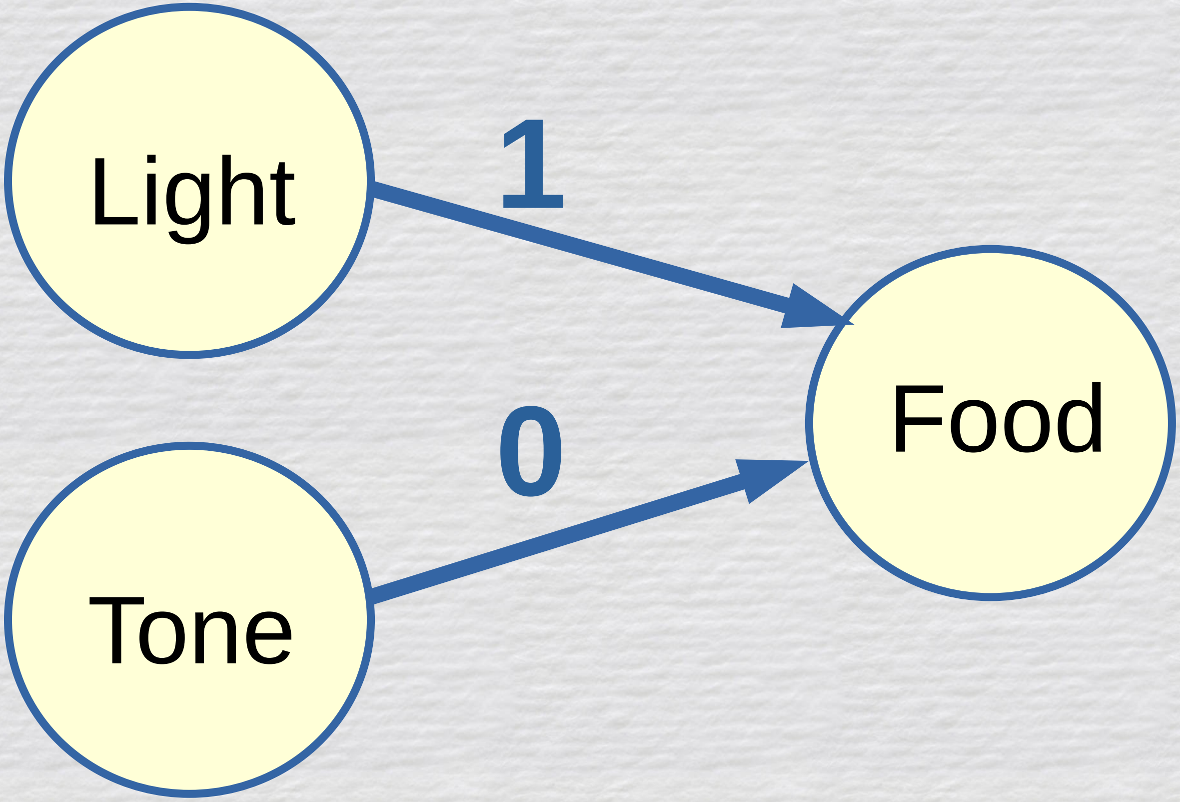
Kamin

1. Tone → Food
2. Tone + Light → Food
3. Light → ?

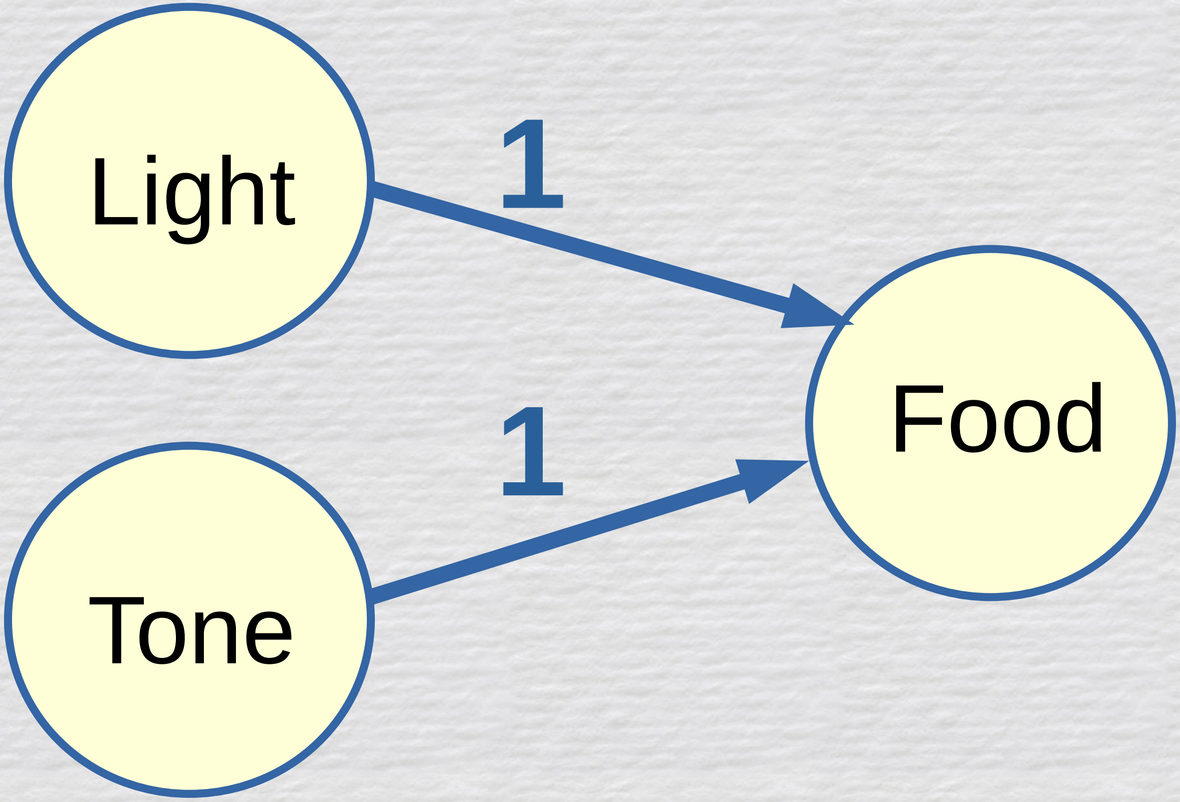
Bush-Mosteller (Blocking)



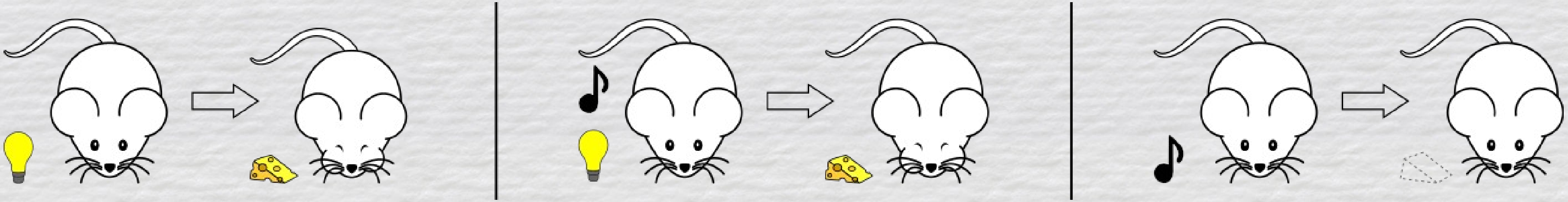
Bush-Mosteller (Blocking)



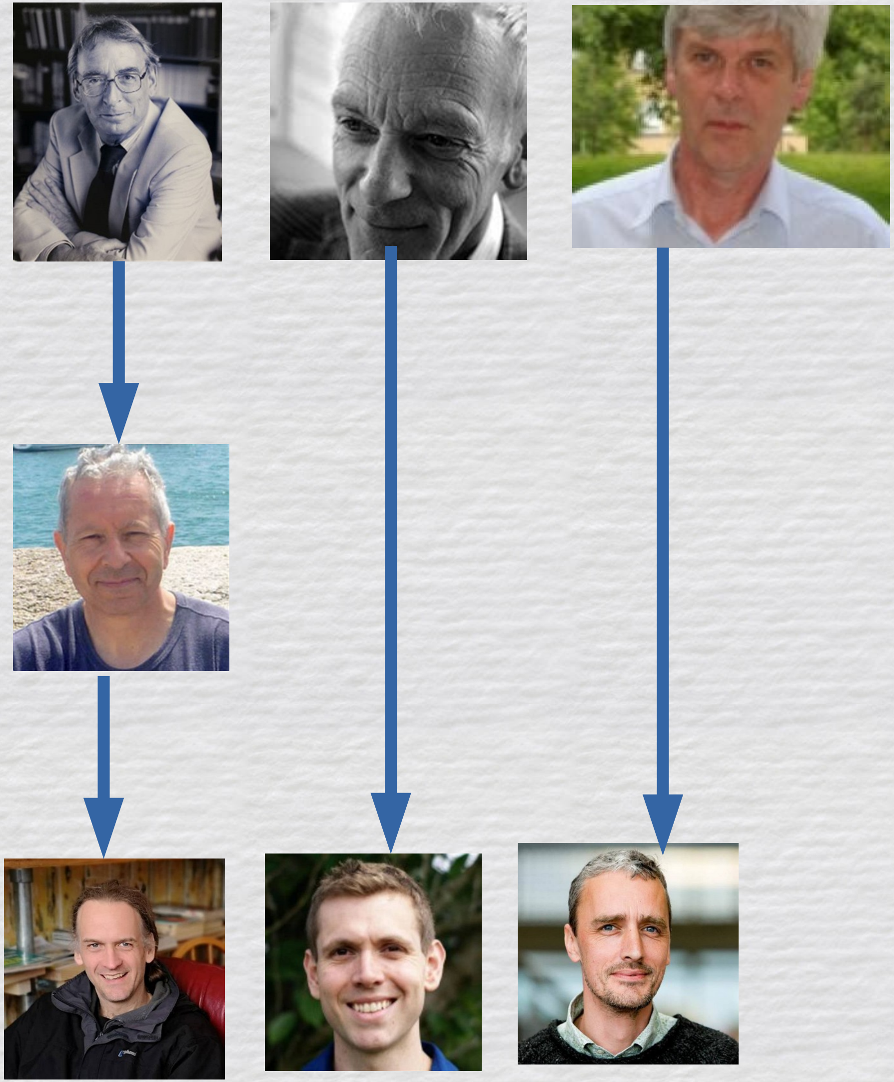
Bush-Mosteller (Blocking)



Blocking (Kamin, 1969)



1. Light → Food
2. Tone + Light → Food
3. Tone → Little salivation



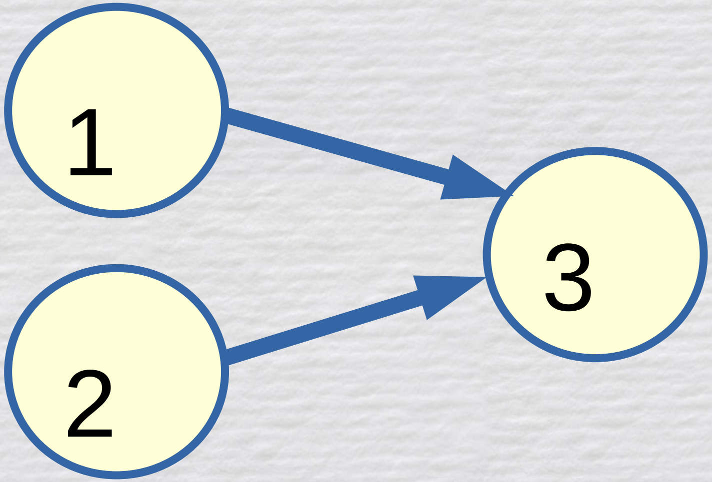
Why blocking?

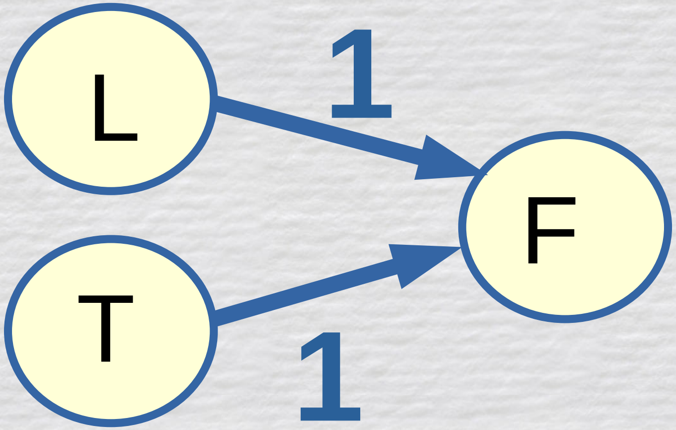
Rescorla-Wagner (1972)

Rescorla

Wagner

Bush-Mosteller



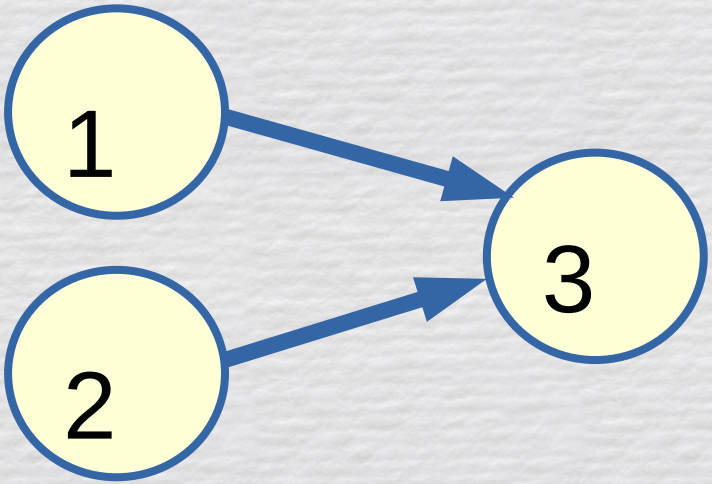


$$\Delta w_{13} = G(t - a_1 w_{1,2})a_1$$

Teacher

Student prediction

Bush-Mosteller → Rescorla-Wagner



$$\Delta w_{13} = G(t - a_1 w_{1,2})a_1$$

Teacher

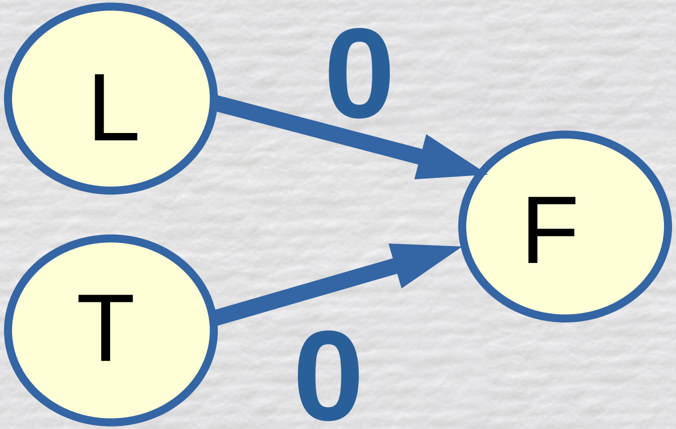
Student prediction

$$\Delta w_{13} = G(t - \Sigma aw)a_1$$

Teacher

Student prediction

Stage 1: Light → Food

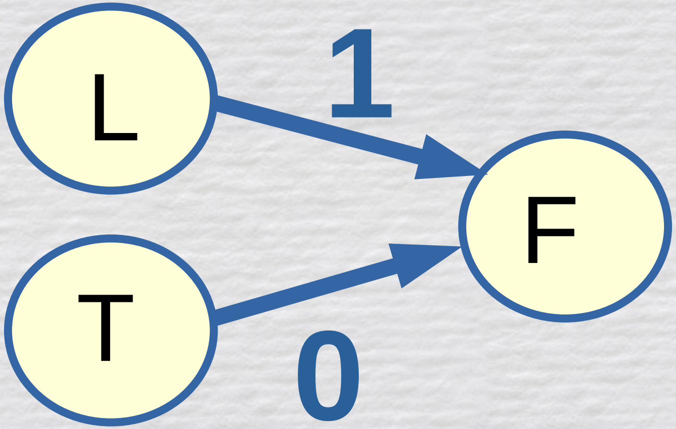


$$\Delta w_{LF} = G(t - \Sigma aw)a_L$$

G= 0.1

Trial	Light (a_L)	Food (t)	Δw_{LF}	w_{LF}
1	1	1	.1 x (1-0) = .1	.1
2	1	1	.1 x (1-.1) = .09	.19
3...	1	1	.1 x (1-.19) = .081	.271
...1000	1	1	.1 x (1-1) = 0	1

Stage 2: Light + Tone → Food

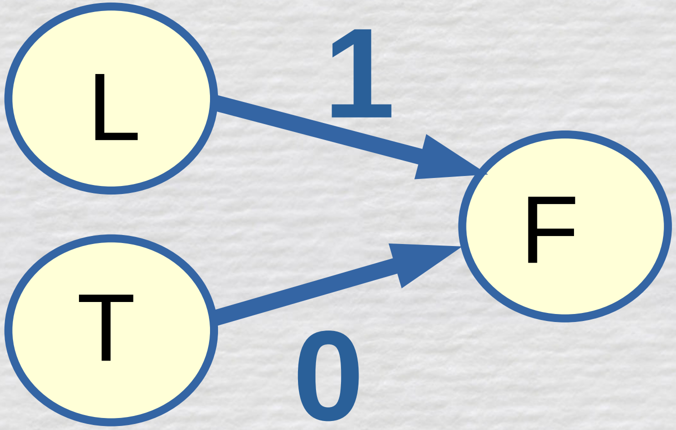


Trial	Tone (a_T)	Food (t)	Δw_{TF}	w_{TF}
1	1	1	.1 x (1-1) = 0	0
2	1	1	.1 x (1-1) = 0	0
3...	1	1	.1 x (1-1) = 0	0
...1000	1	1	.1 x (1-1) = 0	0

$$\Delta w_{TF} = G(t - \Sigma aw)a_T$$

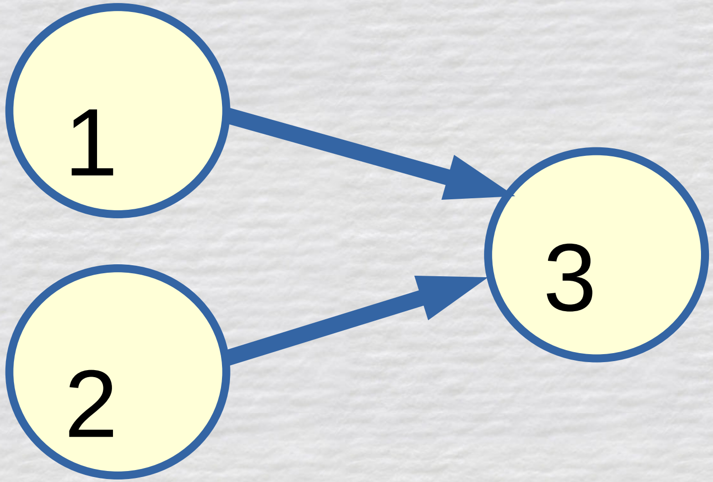
G= 0.1

Stage 3: Tone → ?

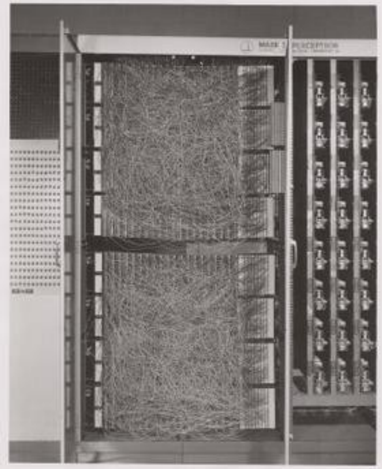


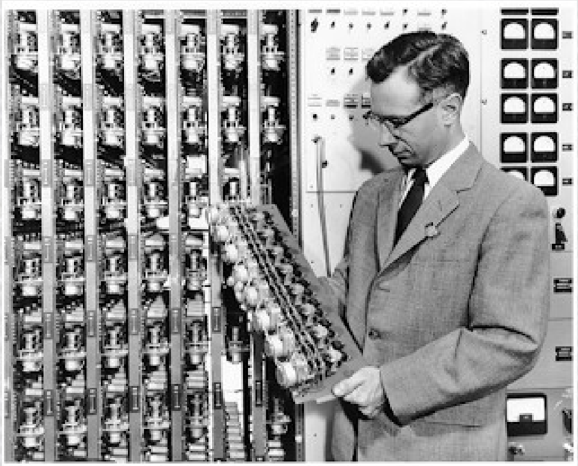
AI: Convergence rule

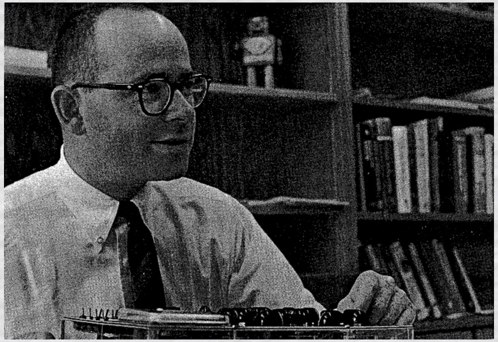
Mathematical proof: anything that can be learned by a single-layer network will be learned by the delta rule.



1958-1969: AI optimism







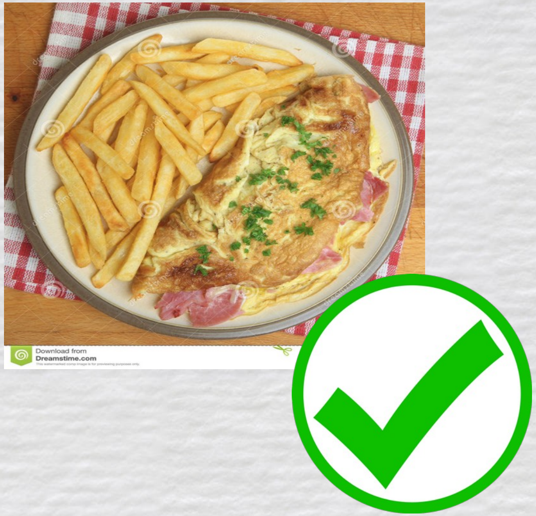
"the embryo of an electronic computer that [the Navy] expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence."

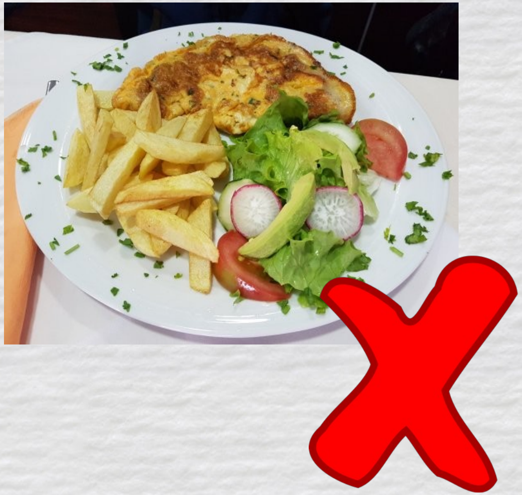
First AI Winter

XOR

“Omelette, with chips or salad”

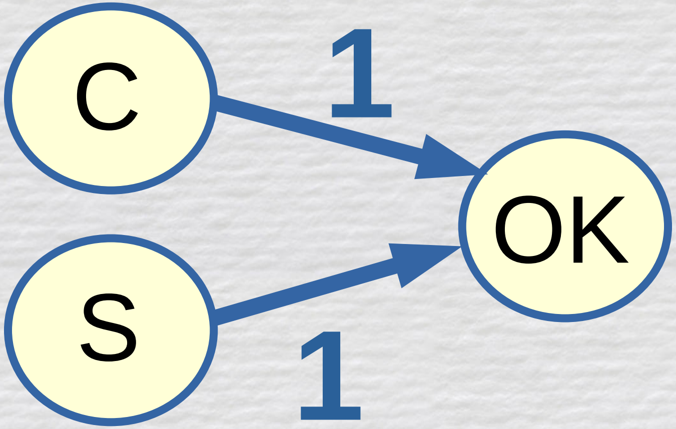
XOR





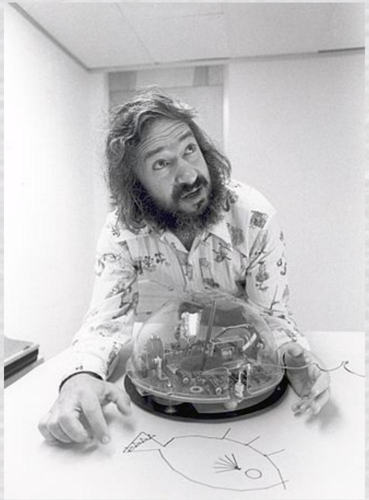
“Omelette, with chips or salad”

Limitations of single-layer nets

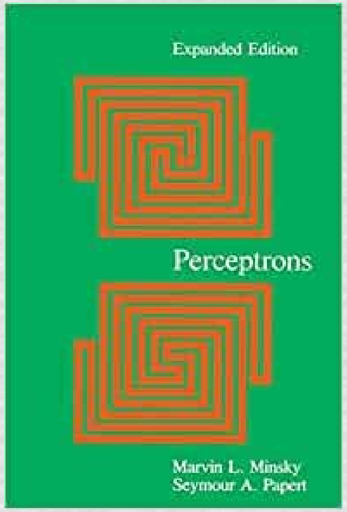


Perceptrons (1969)

Minksy



Papert

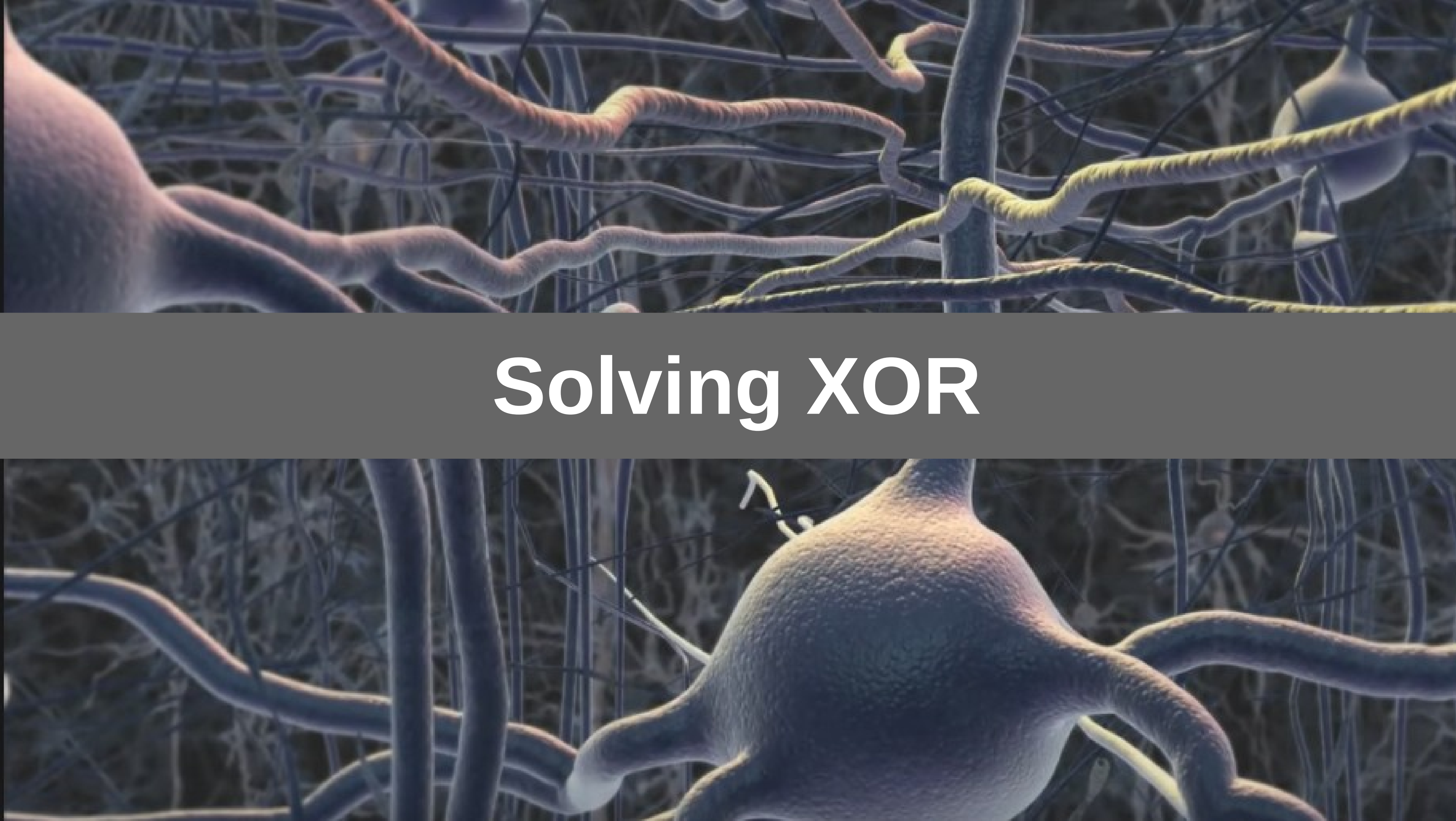


1st AI winter (1974-1980)

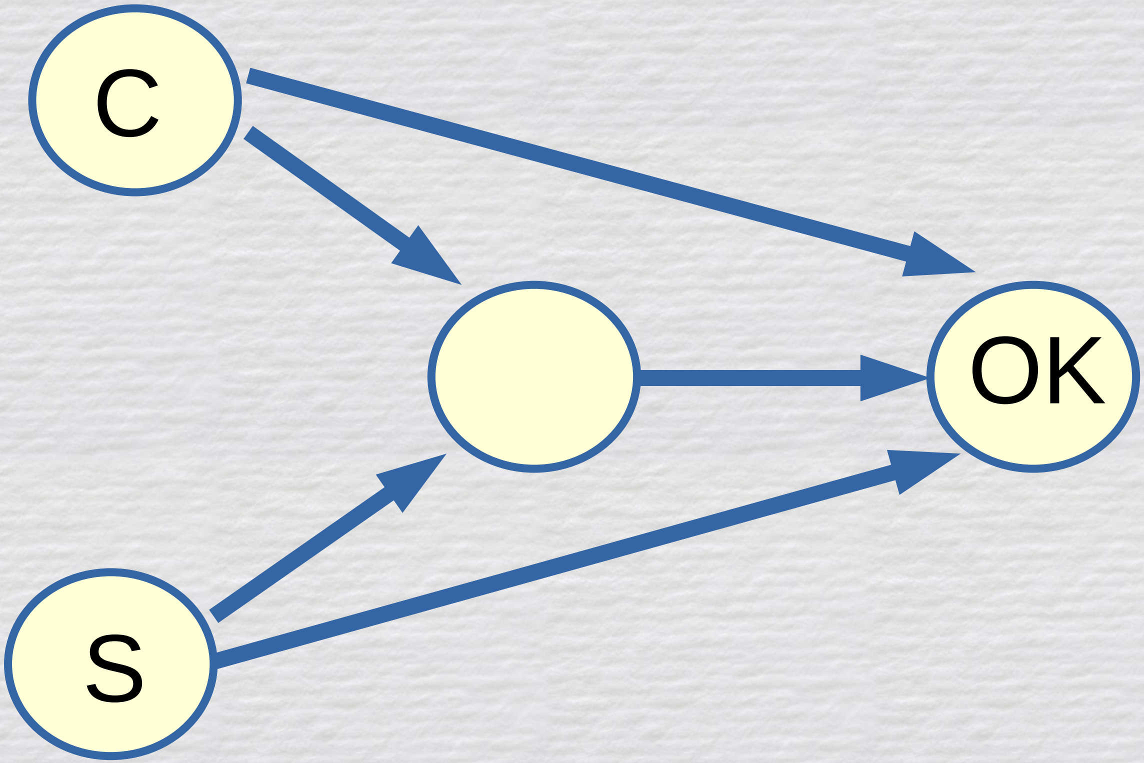
Lighthill

"In no part of the field have the discoveries made so far produced the major impact that was then promised"
- Lighthill report (1974)

Solving XOR



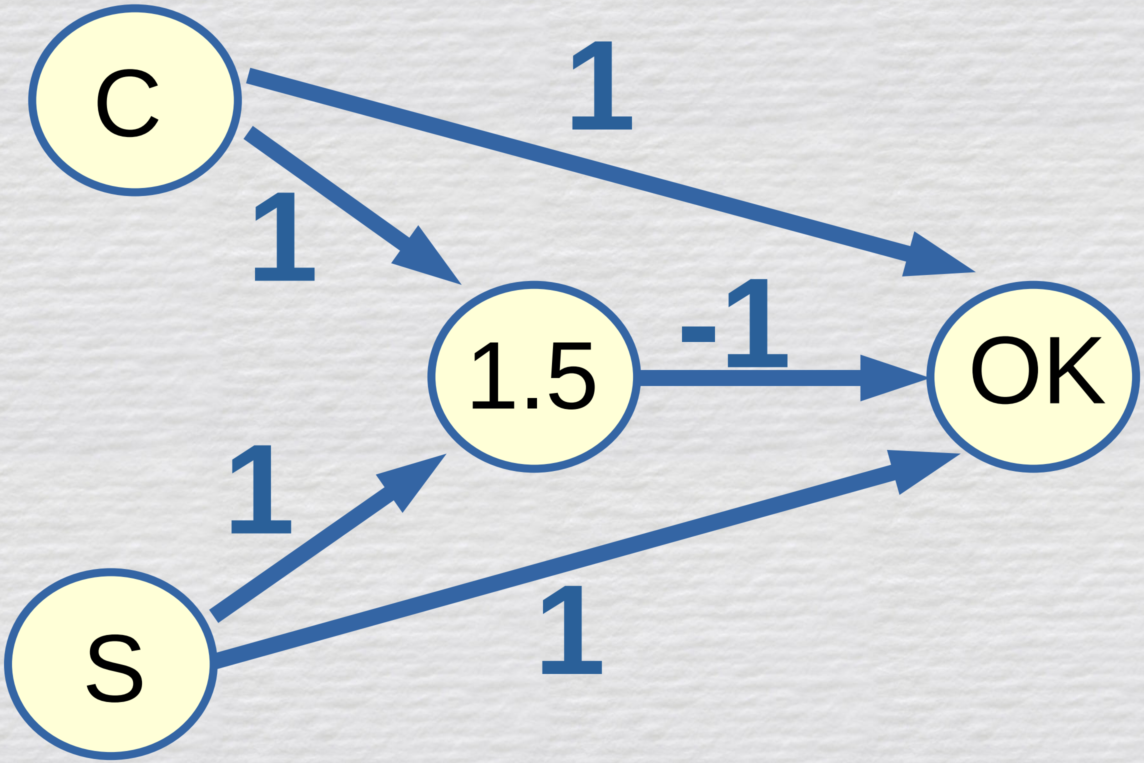
Solving XOR



Threshold activation

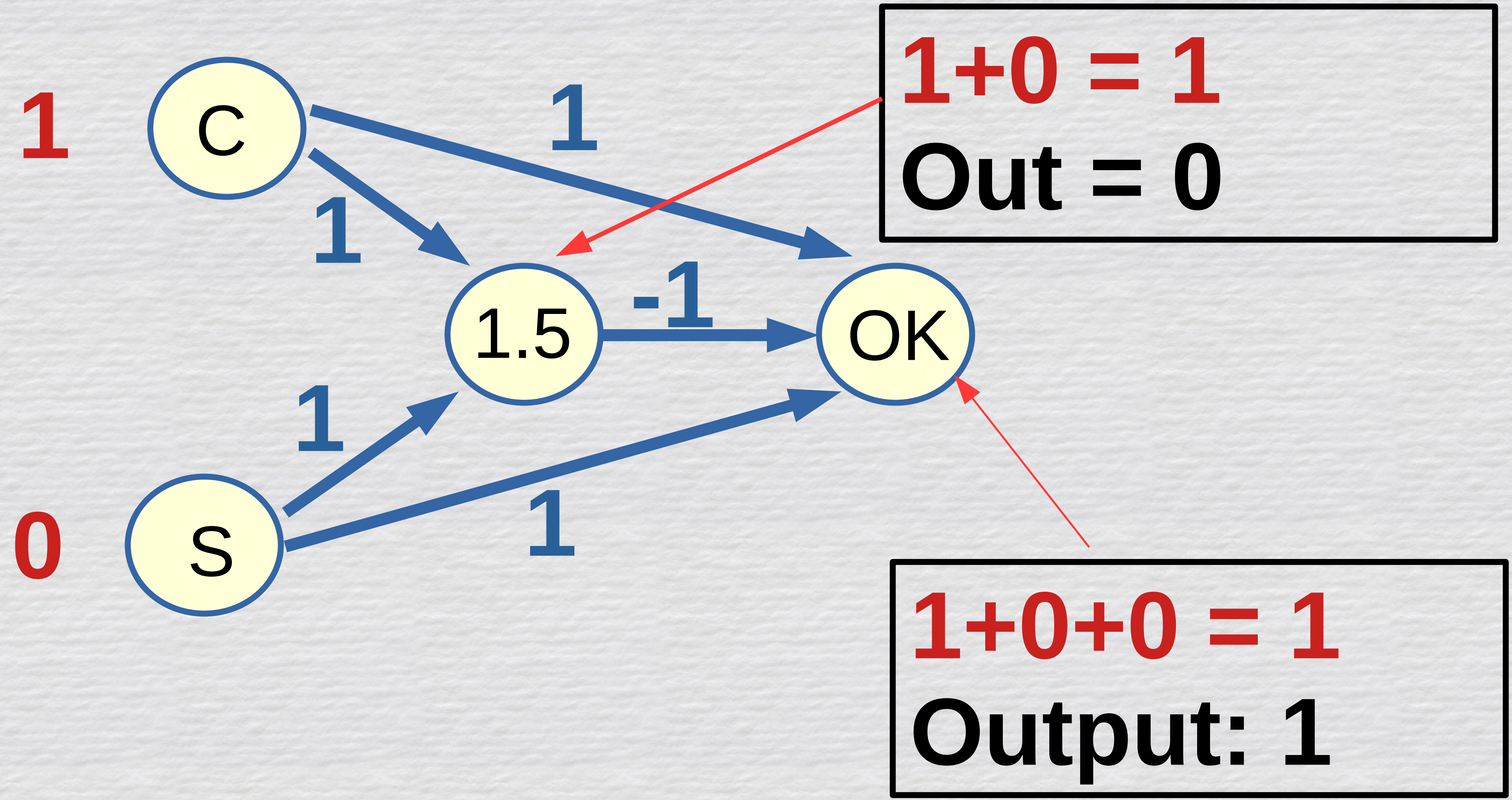
If *in* < *T*, *out* = 0
Otherwise, *out* = *in*

Solving XOR

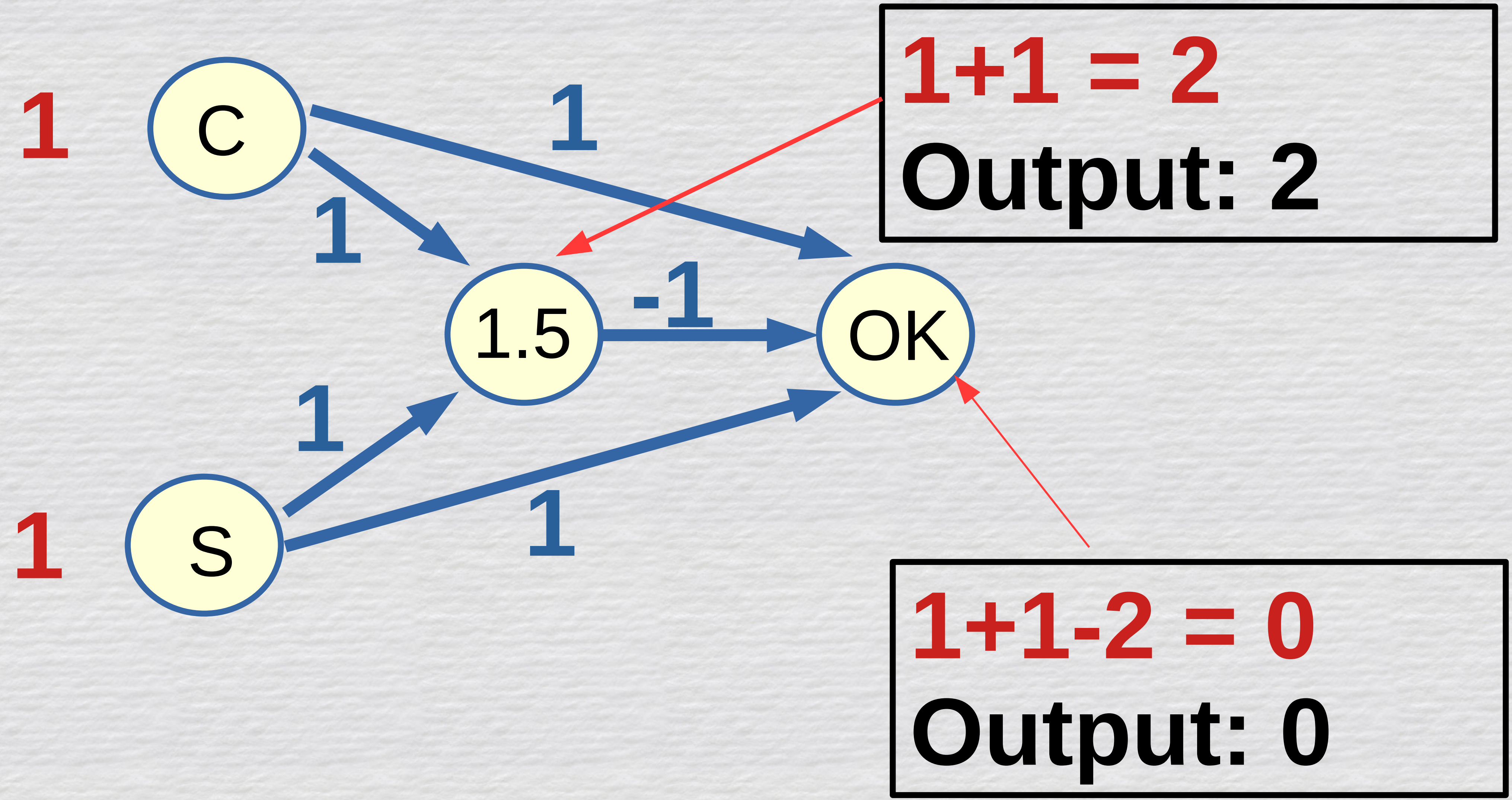


Threshold activation

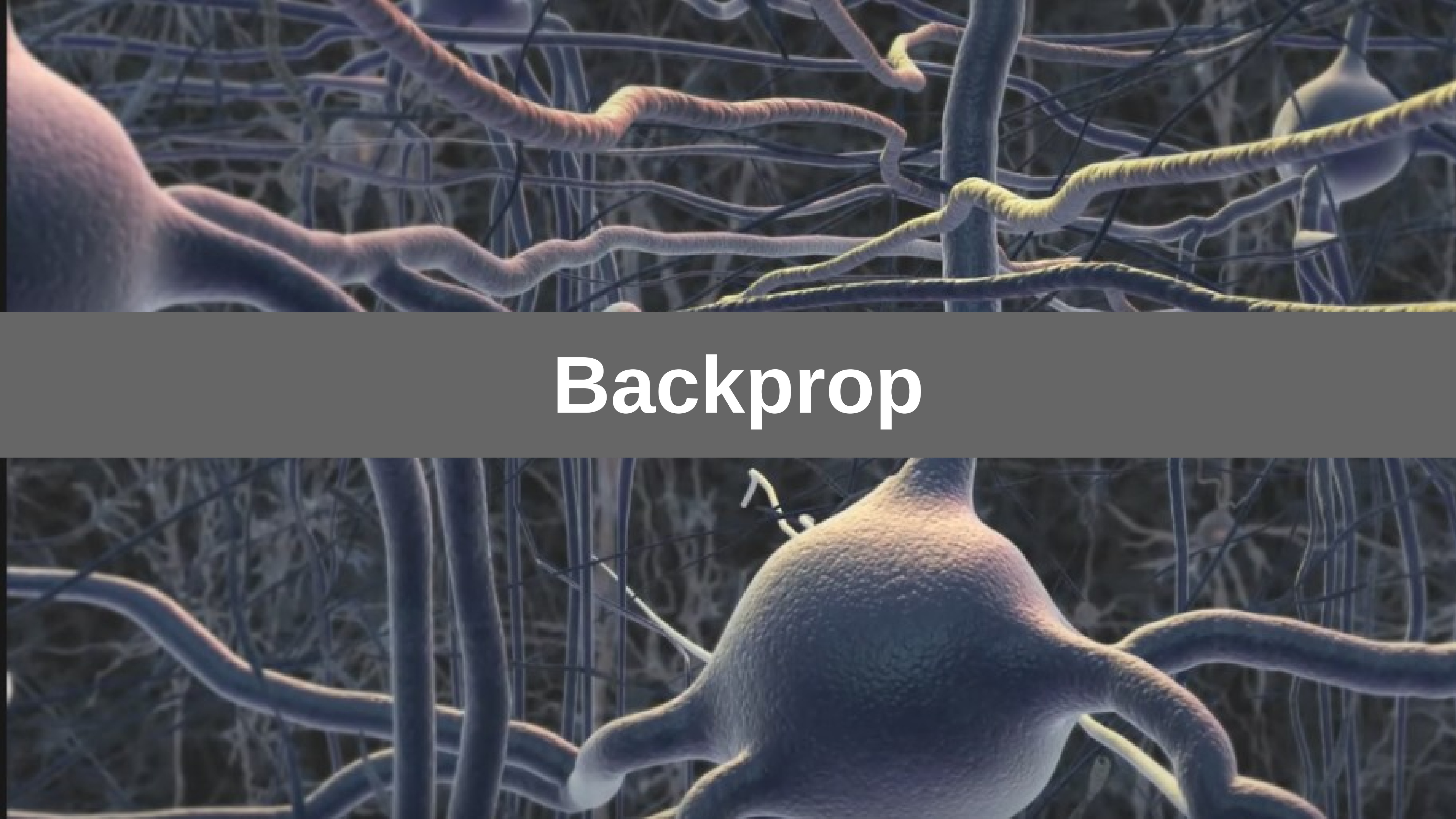
If *in* < *T*, *out* = 0
Otherwise, *out* = *in*



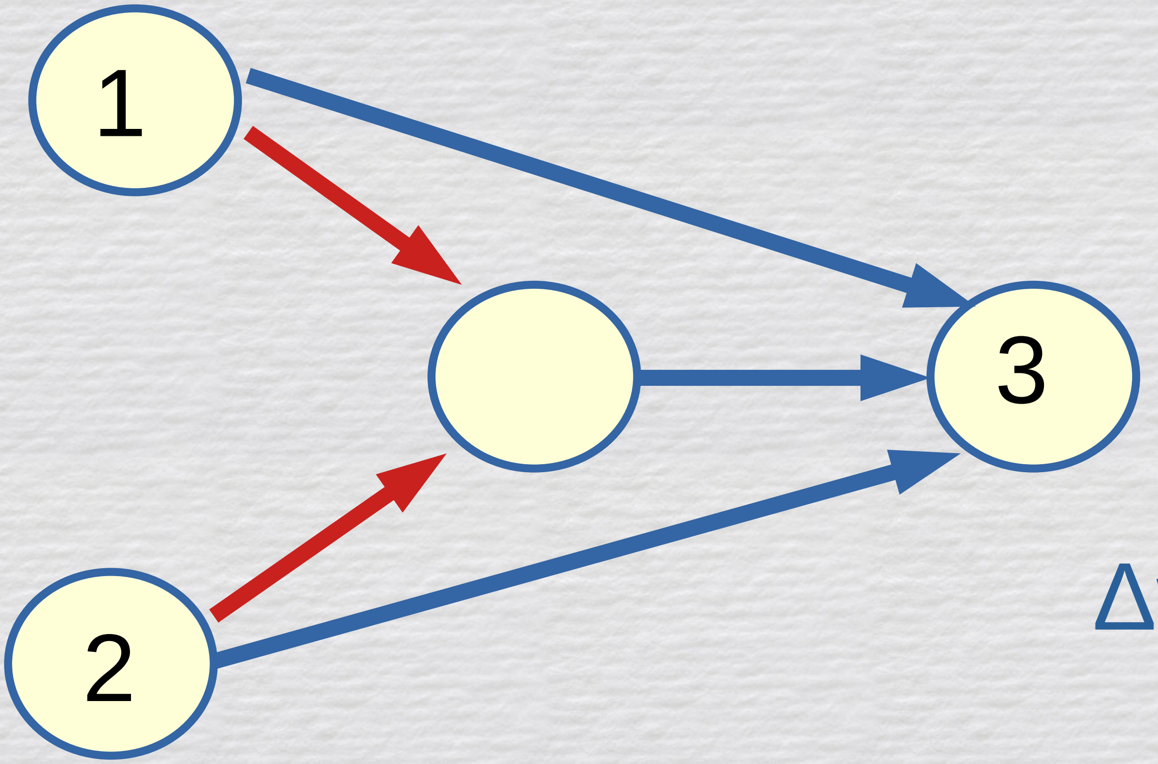
1
C
1
1
1.5
-1
OK
1
S
1
0
1+0 = 1
Out = 0
1+0+0 = 1
Output: 1


1
C
1
1
1.5
-1
OK
1
S
1
1
1+1 = 2
Output: 2
1+1-2 = 0
Output: 0

Backprop



Backpropagation of error



$$\Delta w_{13} = G(t - \Sigma a w)a_1$$

Teacher

Student prediction

Defining error

$$\Delta w_{13} = G(t - \Sigma aw\,)a_1$$

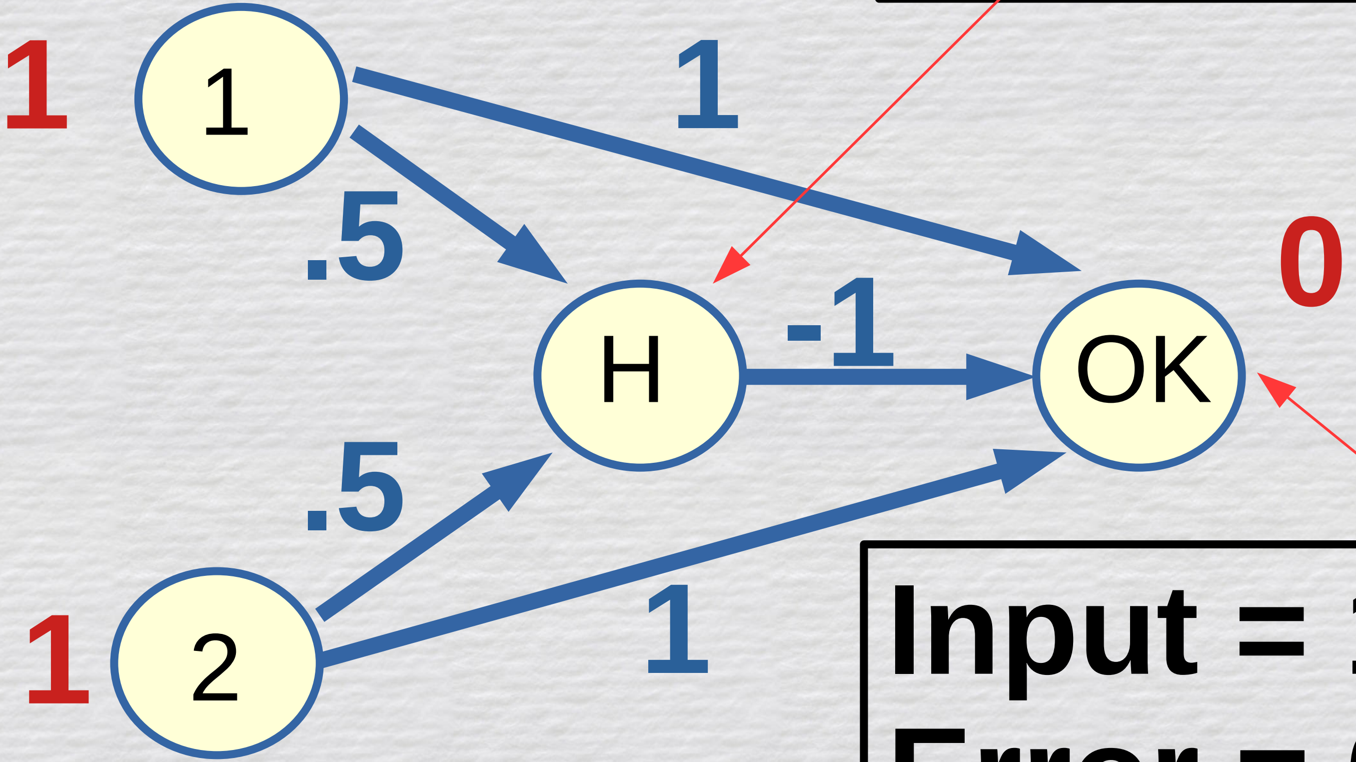
Teacher

Student prediction

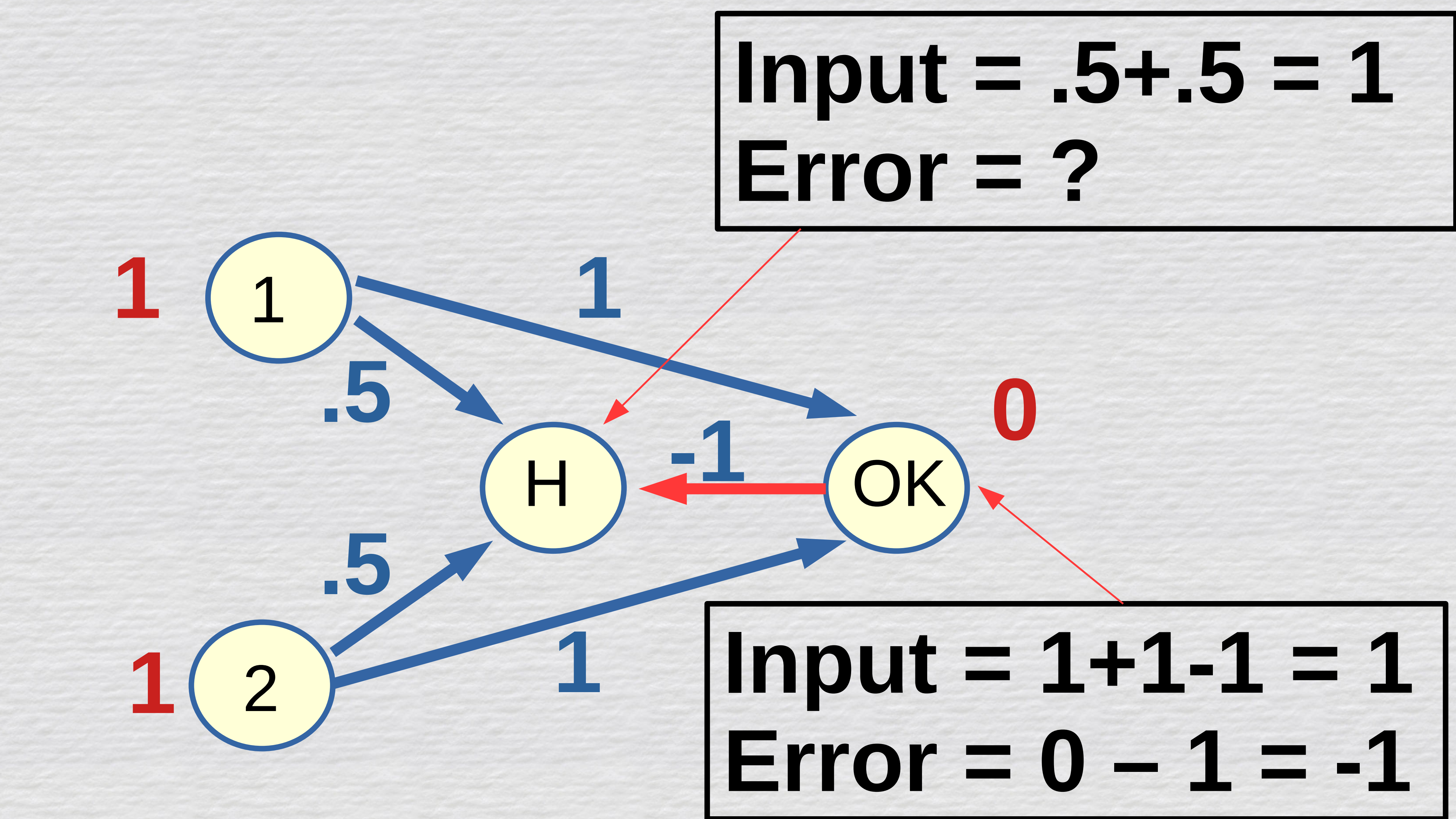
$$E_3 = (t - \Sigma aw\,)$$

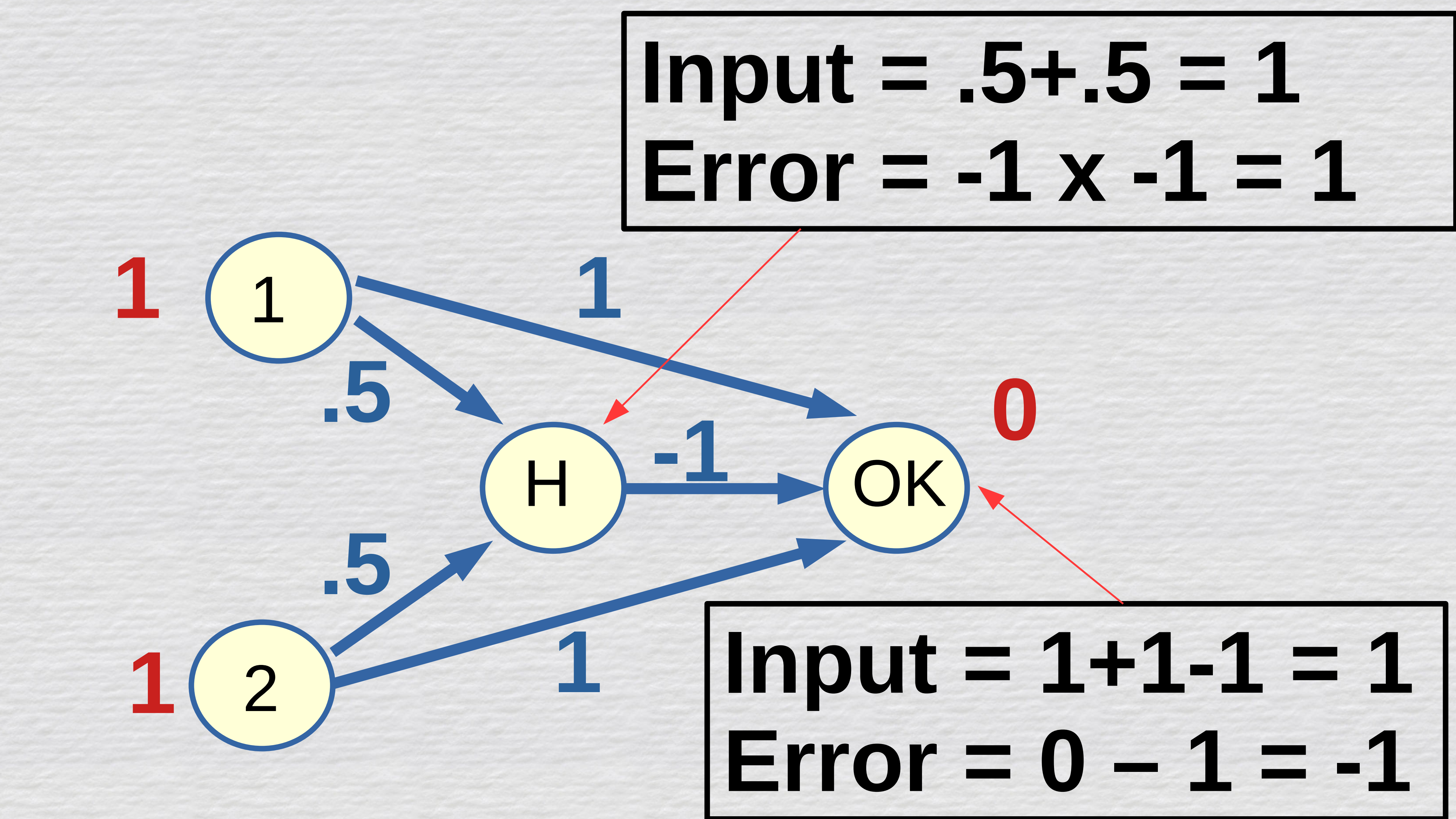
$$\Delta w_{13} = G.E_{3.}a_1$$

Input = .5+.5 = 1
Error = ?

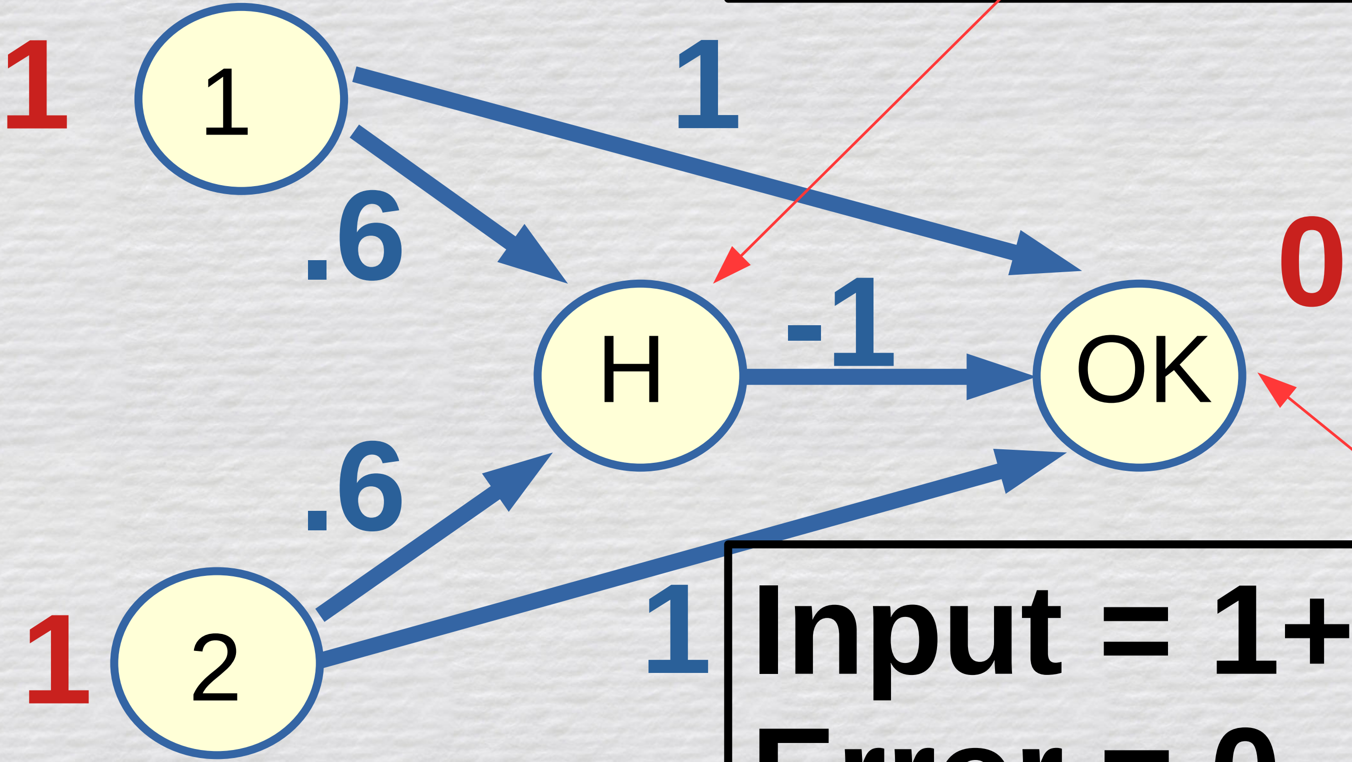


Input = 1+1-1 = 1
Error = 0 – 1 = -1


Input = .5+.5 = 1
Error = ?
1
1
1
.5
H
-1
OK
0
.5
1
2
1
Input = 1+1-1 = 1
Error = 0 – 1 = -1


Input = .5+.5 = 1
Error = -1 x -1 = 1
1
1
1
.5
H
-1
OK
0
.5
1
2
1
Input = 1+1-1 = 1
Error = 0 – 1 = -1

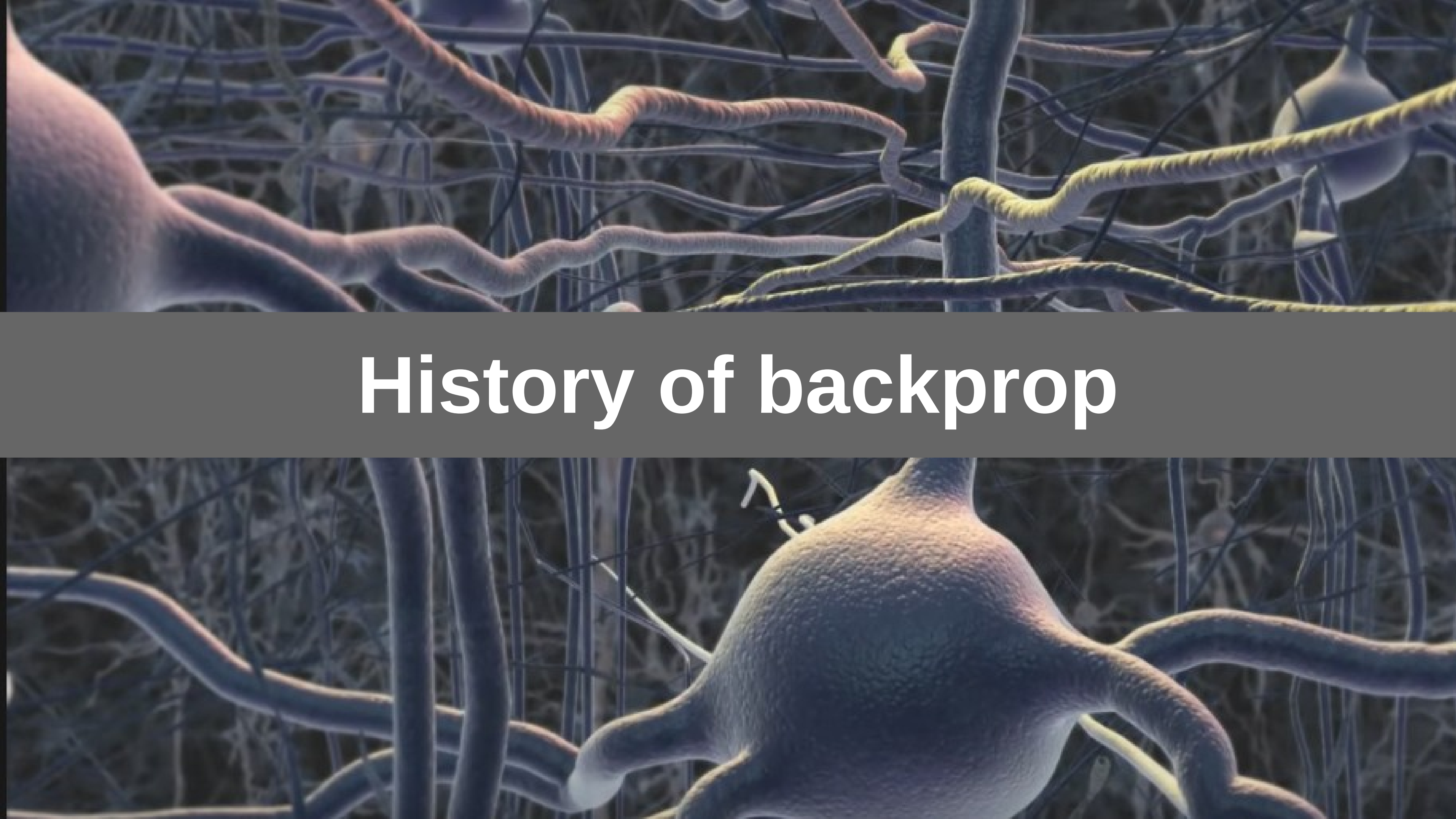
Input = .6+.6 = 1.2



Input = 1+1-1.2 = .8

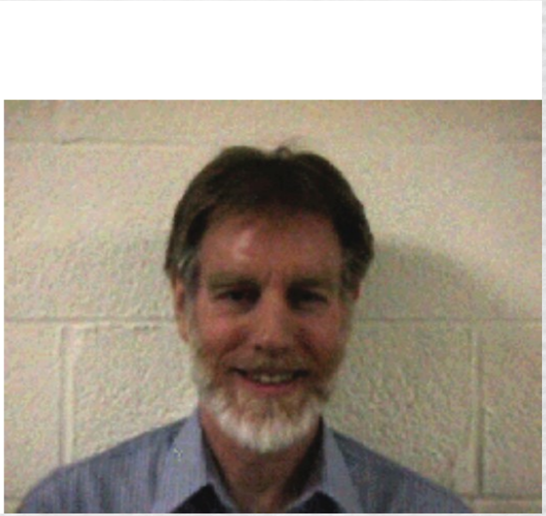
Error = 0 – .8 = -.8

History of backprop



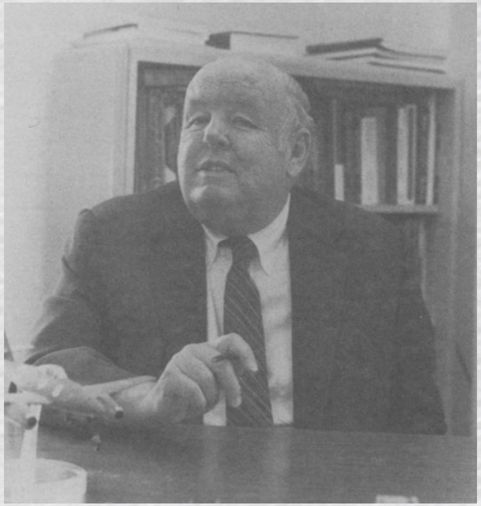
Standard story





Rumelhart, Hinton & Williams (1986)

Invention of backprop



Kelley (1960)

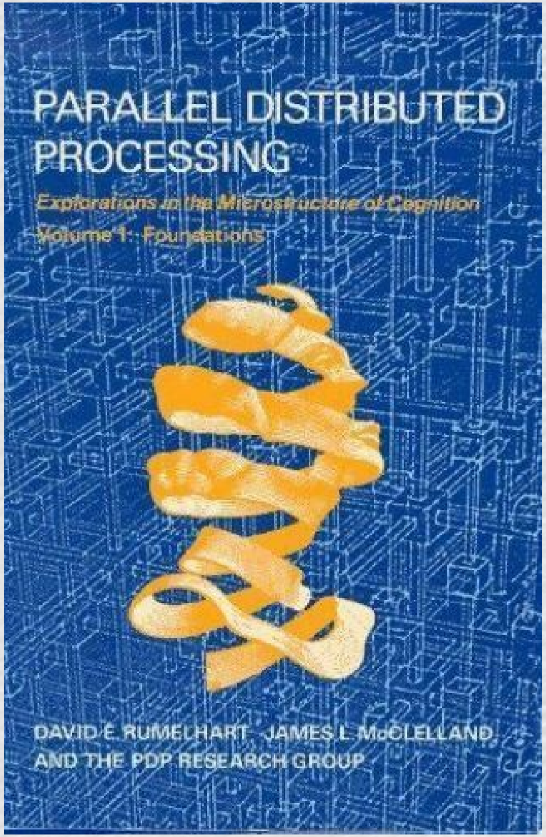
Werbos (1974)

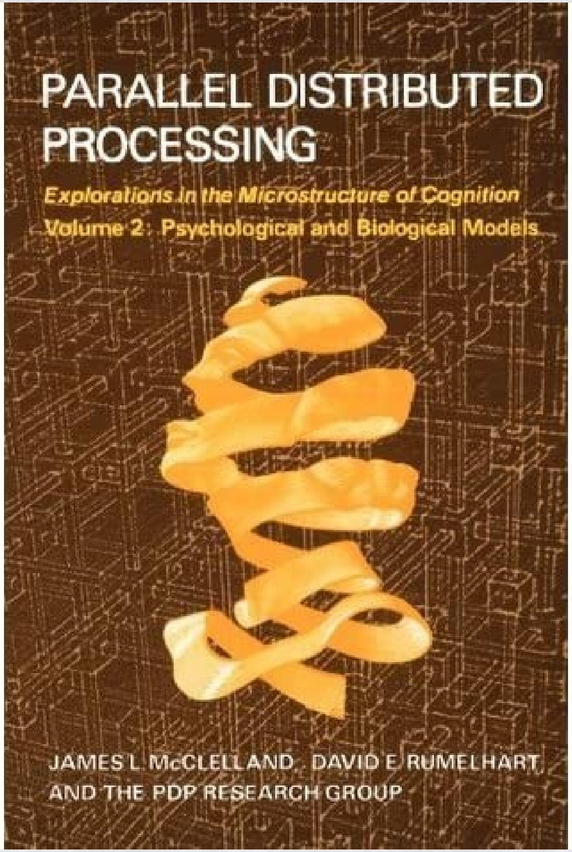
Invention of backprop

Yann LeCun

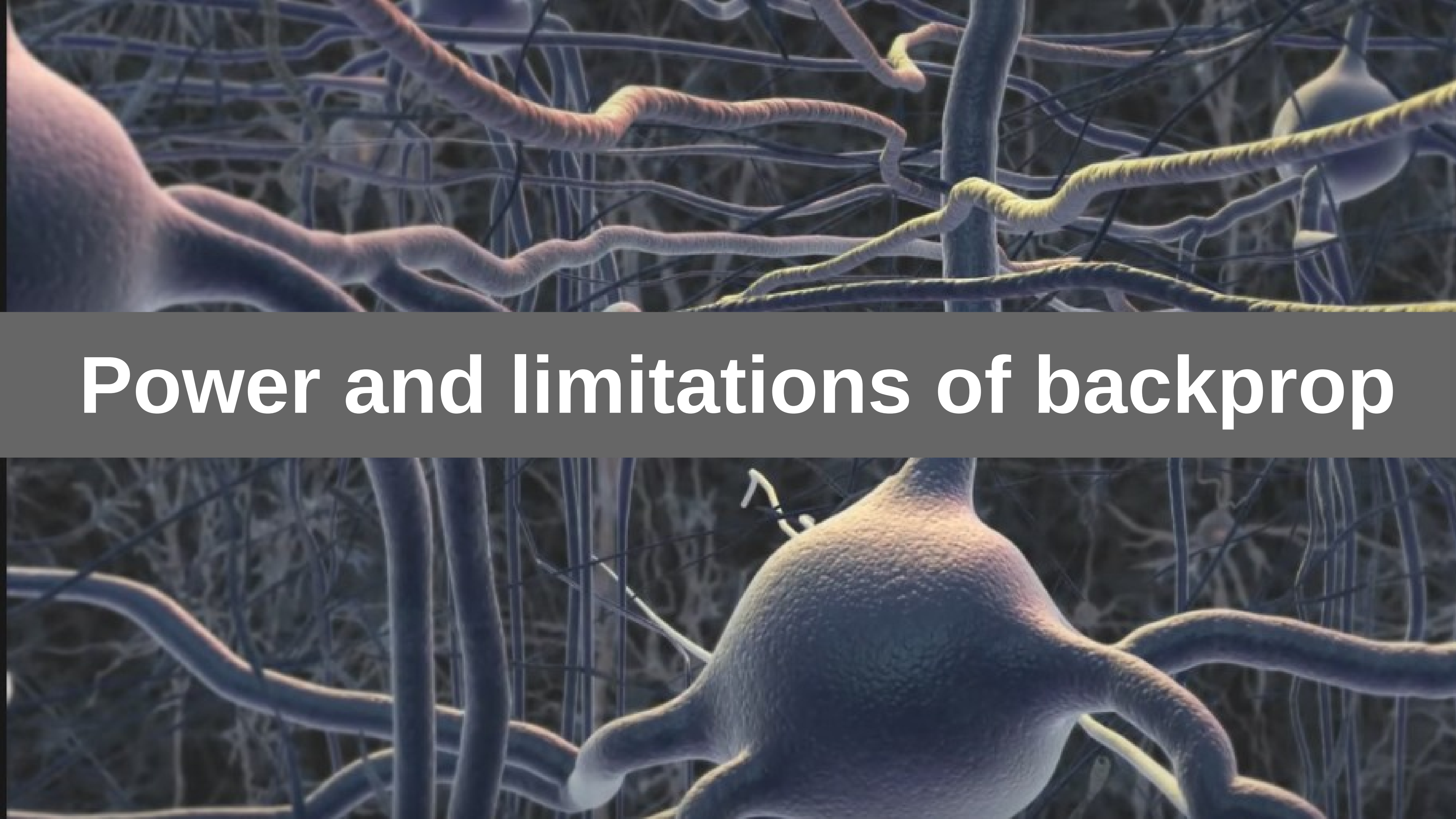
Sun-1 (1983)
£24,000 in today's money

Connectionism

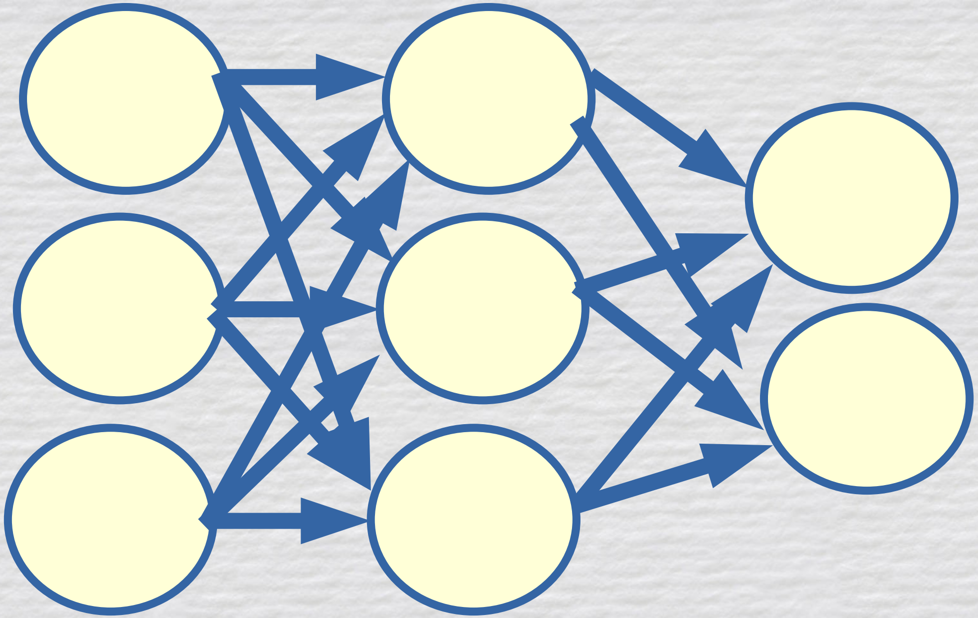




Power and limitations of backprop



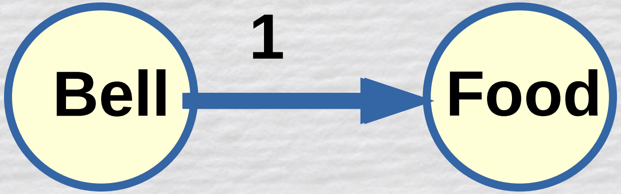
Universal approximators

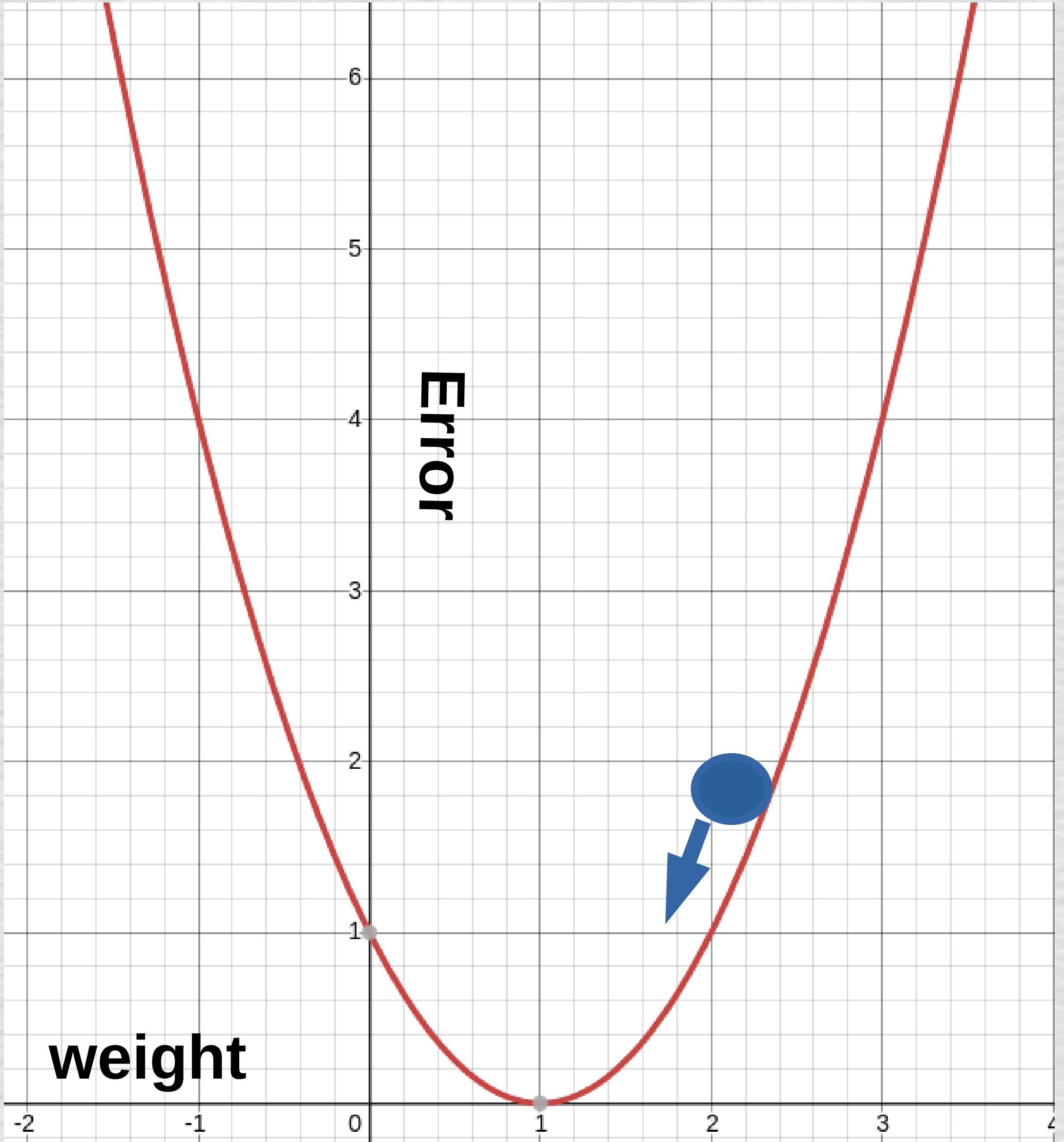


A multilayer feedforward network, with sufficient hidden units can represent any deterministic mapping between its inputs and its outputs

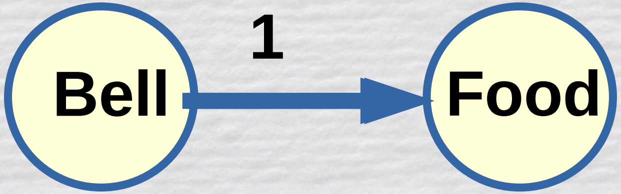
- Hornik, Stinchcombe & White (1989)

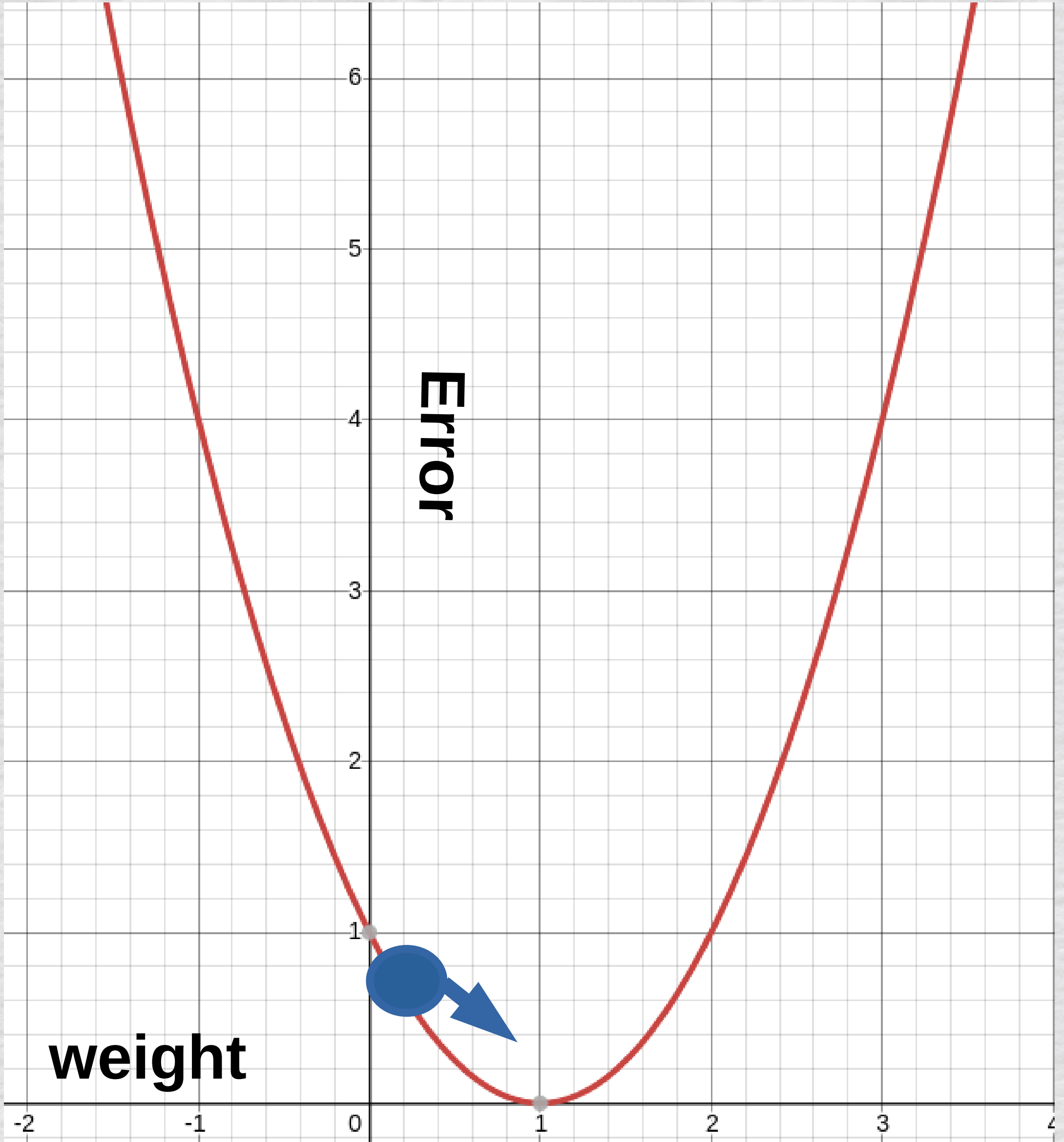
Error surface



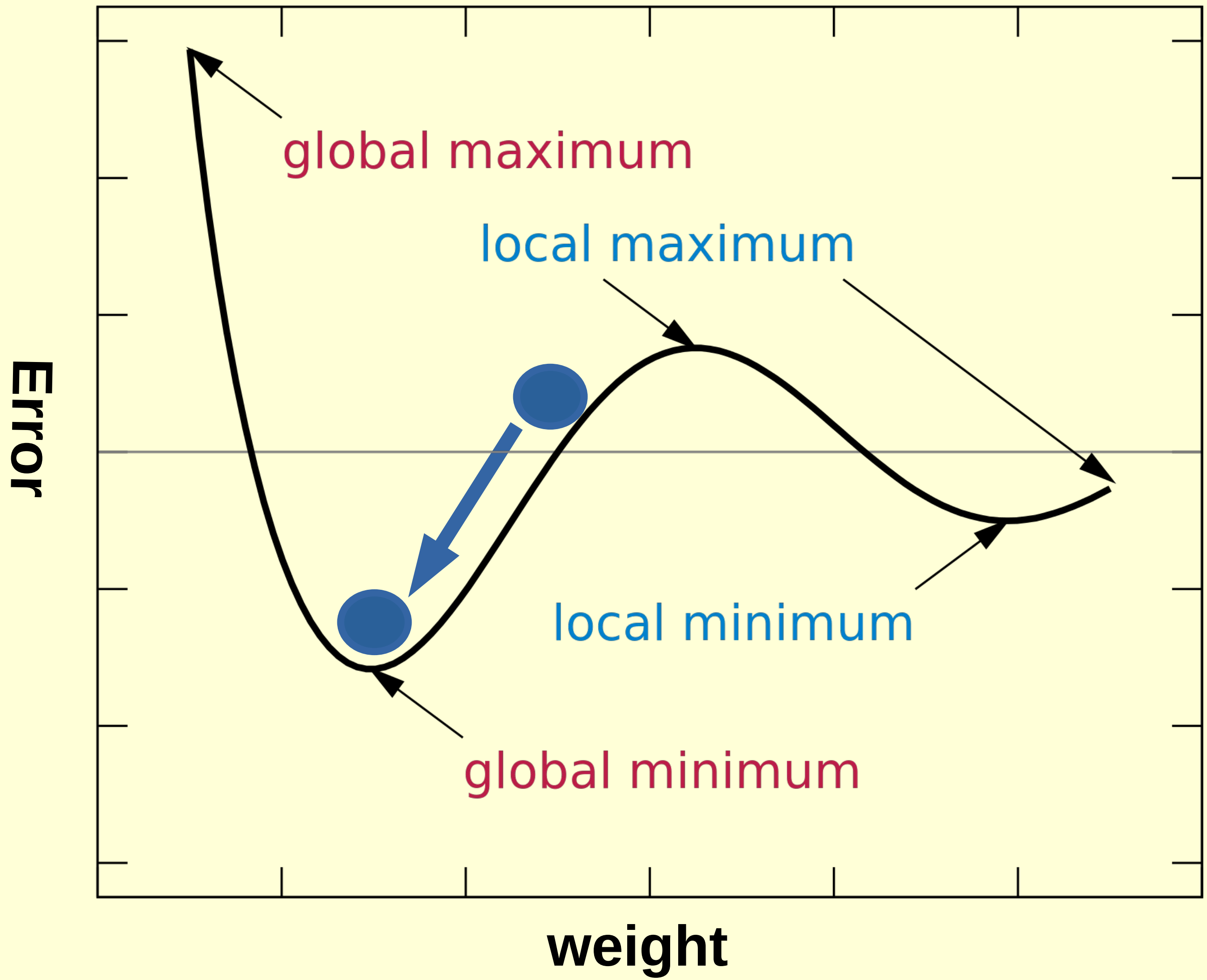


Error surface

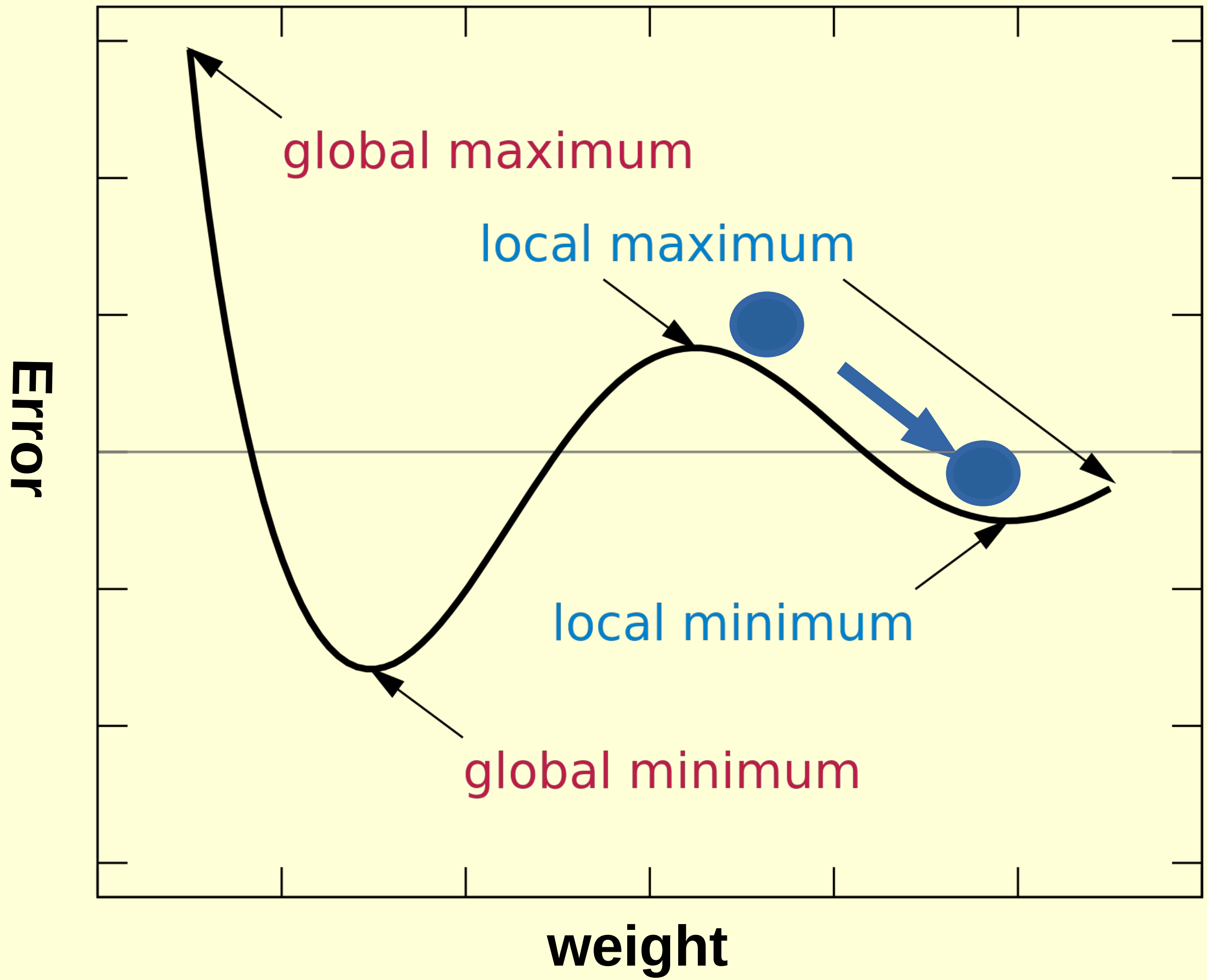




Error surface



Error surface



Interference

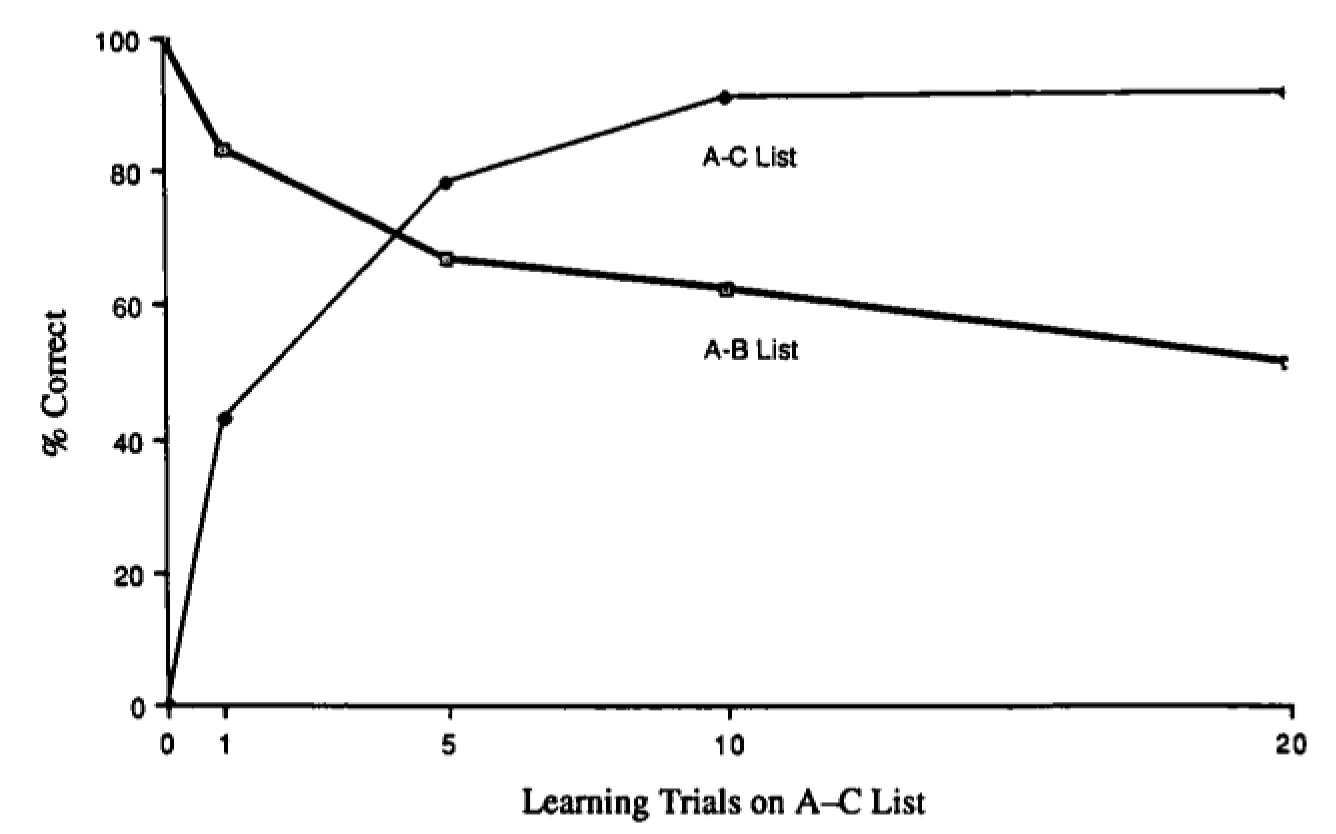
"Every time I learn something new, it pushes some old stuff out of my brain"
- Homer Simpson

People: Retroactive Interference

Retroactive interference
Barnes & Underwood (1959)

<u>List 1 (A-B)</u>
dax – regal
…

<u>List 2 (A-C)</u>
dax – cabbage
…

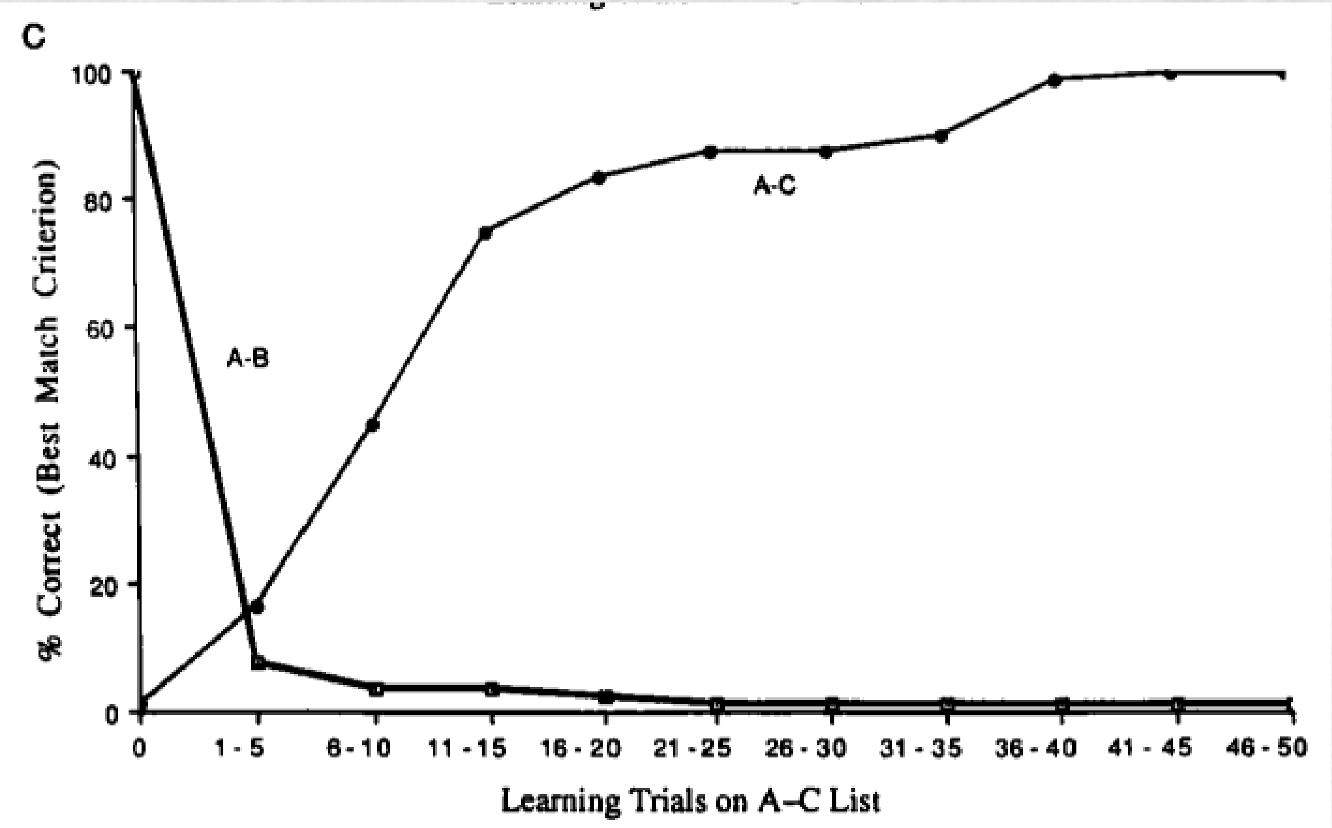


Backprop: Catastrophic Interference

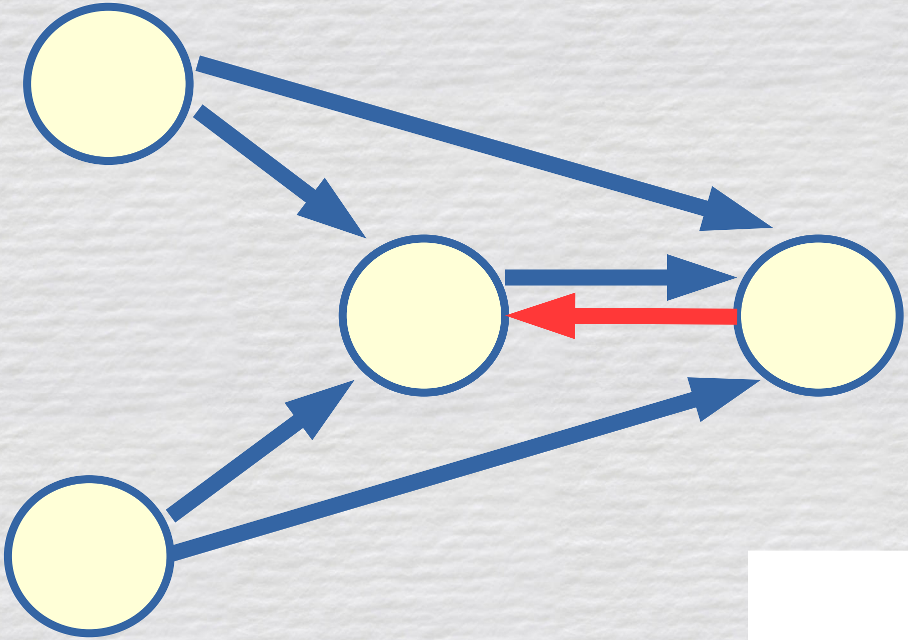
Catastrophic interference
McCloskey & Cohen (1989)

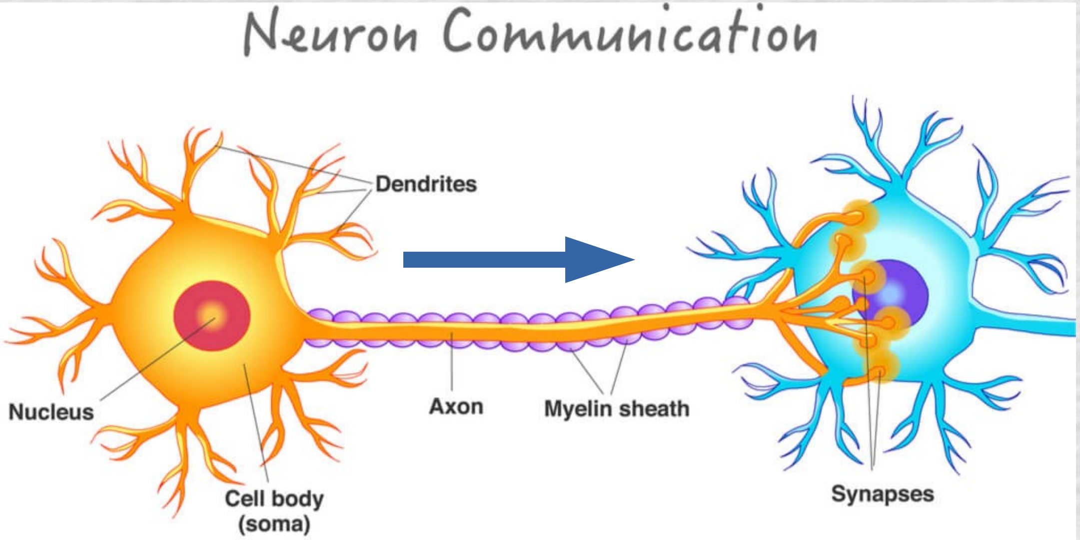
<u>List 1 (A-B)</u>
dax – regal
…

<u>List 2 (A-C)</u>
dax – cabbage
…



Neural plausibility





Summary