University of BRISTOL

Towards a less ‘artificial’ intelligence

Dabal Pedamonti

Teaching Associate in Engineering Mathematics and Technology

It’s time to give us your unit feedback

It's quick

It's confidential

We're listening

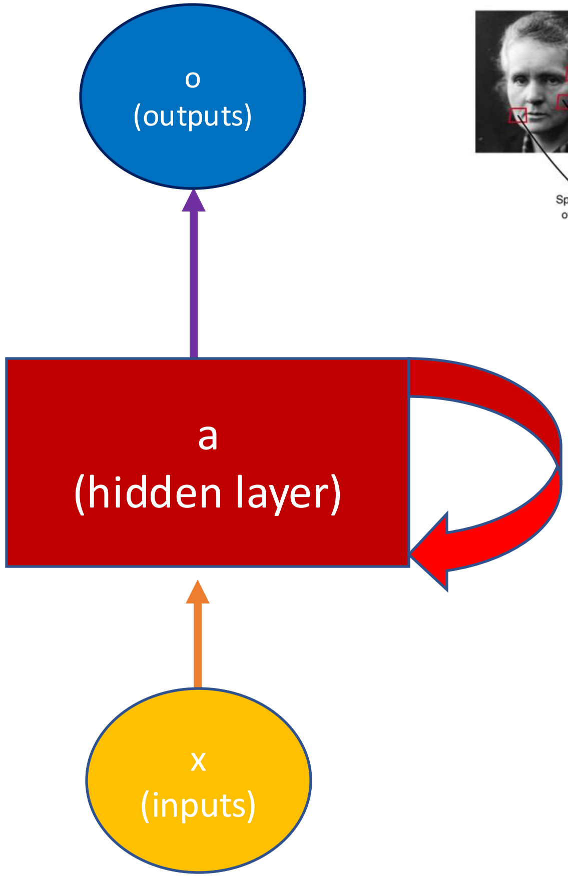
bristol.ac.uk/unit-evaluation

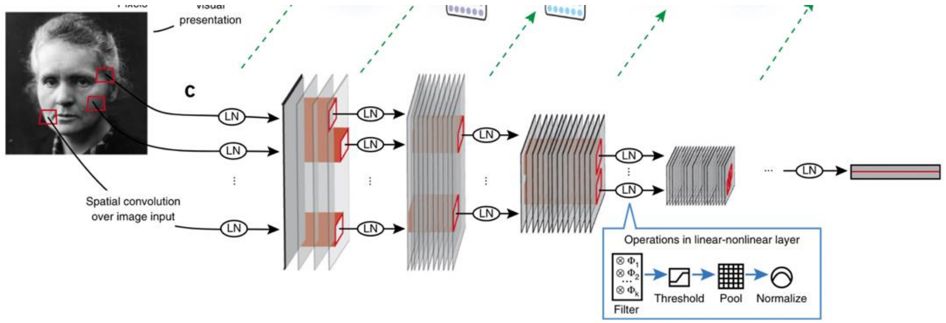
Intended Learning Outcomes

- By the end of this lecture you will be able to:
  - explain the motivation for building brain-realism into AI models
  - identify which aspects of a model are under our control when training a network to perform a task
  - implement features of brain connectivity in a trained recurrent neural network
  - penalise networks for non-brain-like solutions to performing tasks
  - identify unrealistic features of learning in neural networks and some proposed alternatives

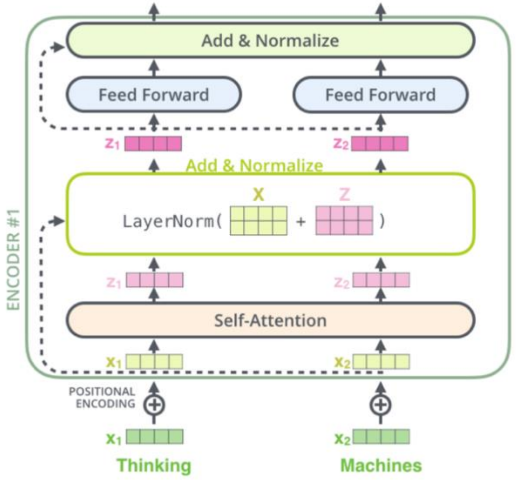
AI models – designed for performance

- Often inspired by the brain
- Many biological details are removed or simplified
- Excel at multiple real-world applications
- Often do not make clear testable predictions for how the brain works



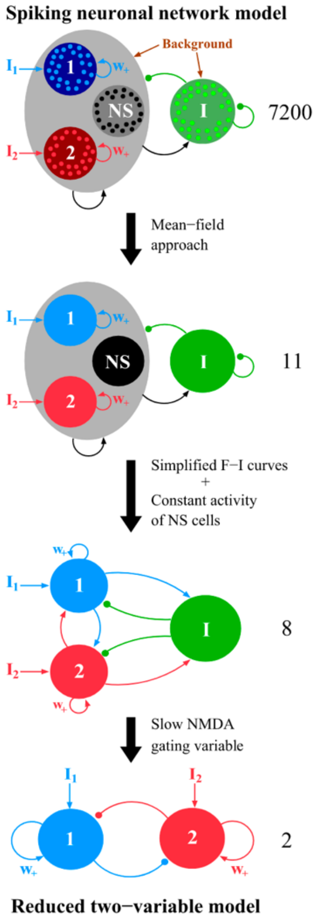


presentation
C
LN
Spatial convolution over image input
Operations in linear-nonlinear layer
Filter
Threshold
Pool
Normalize

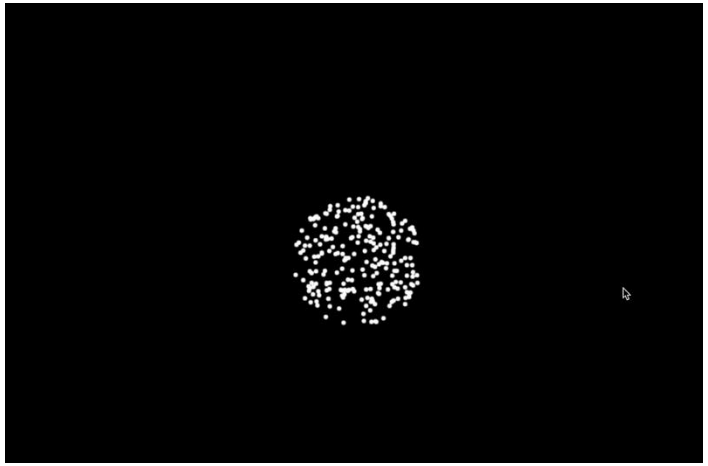


Biologically-"realistic" neural network models

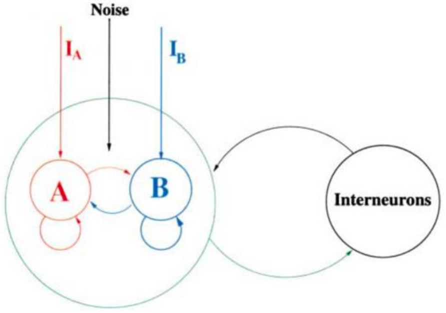
- Parameters are set based on real neural data
- Vary in level of detail, but more biological details are kept vs AI models
- Designed to make directly testable predictions for experiments
- Incapable of complex functions & real-world applications

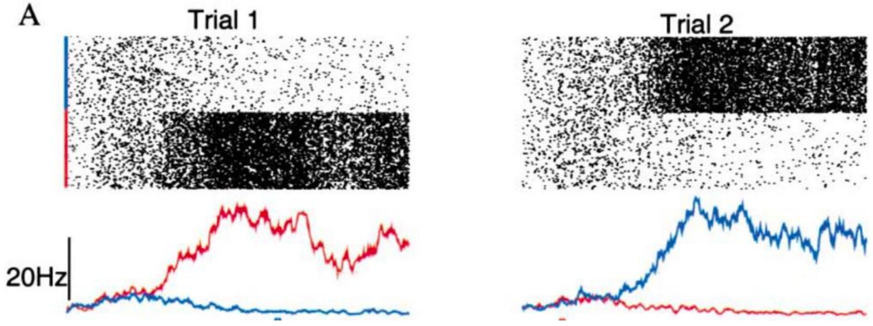


Spiking neuronal network model
Background
I_1
I_2
1
2
NS
I
w_+
7200
Mean−field approach
11
Simplified F−I curves + Constant activity of NS cells
8
Slow NMDA gating variable
2
Reduced two−variable model



Bobby Ines

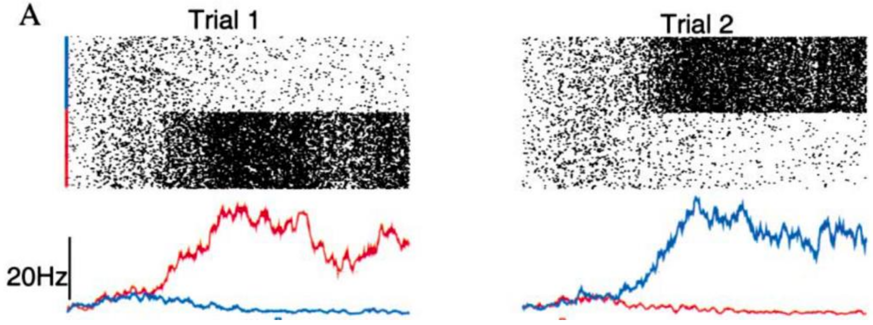


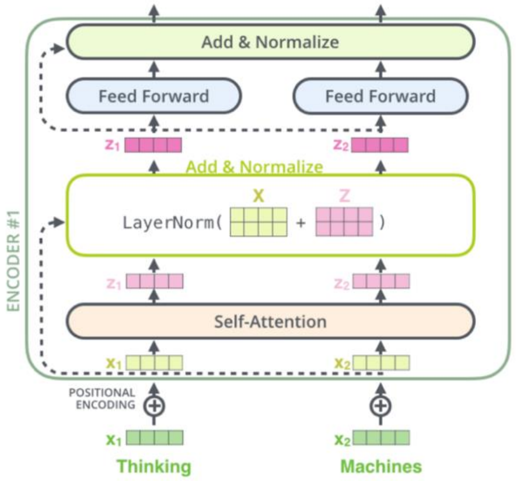


Amit & Brunel, *Cerebral Cortex*, 1997; Wang, *Neuron,* 2002; Wong & Wang, *J. Neurosci*., 2006

Can we bridge the gap?

- Models capable of complex perceptual, cognitive and motor functions
- and capable of making directly testable predictions for experiments





Ideas?

When designing an artificial neural network to be perform a task, what is not under our control?

- the details of the task
- the exact weights

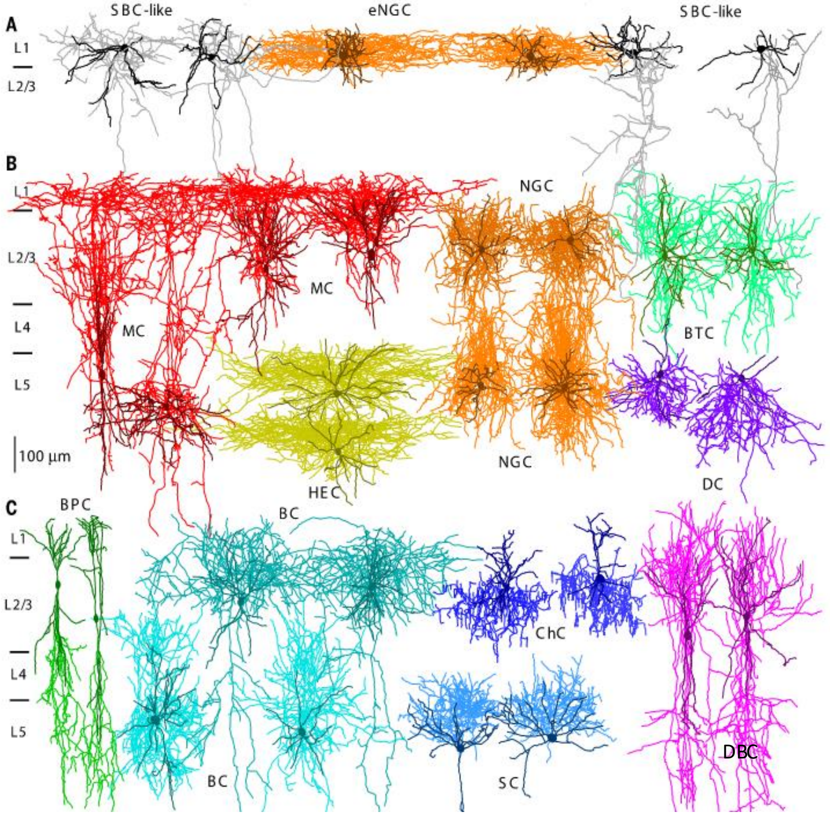
When designing an artificial neural network to be perform a task, what *is* under our control?

- the architecture
- the cost function
- the learning rule

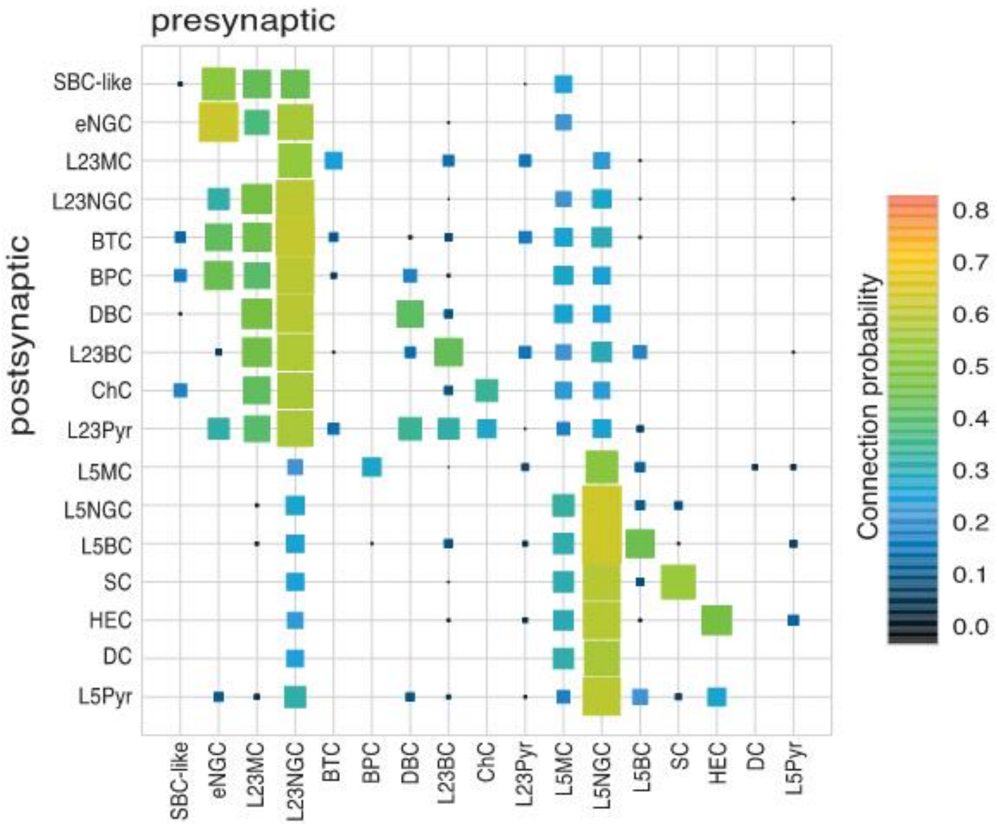
The architecture

- AI – choose an architecture that can best solve the problem
- Neuro AI – choose an architecture that can best solve the problem with pieces connected in a brain-like manner

The architecture of a cortical area

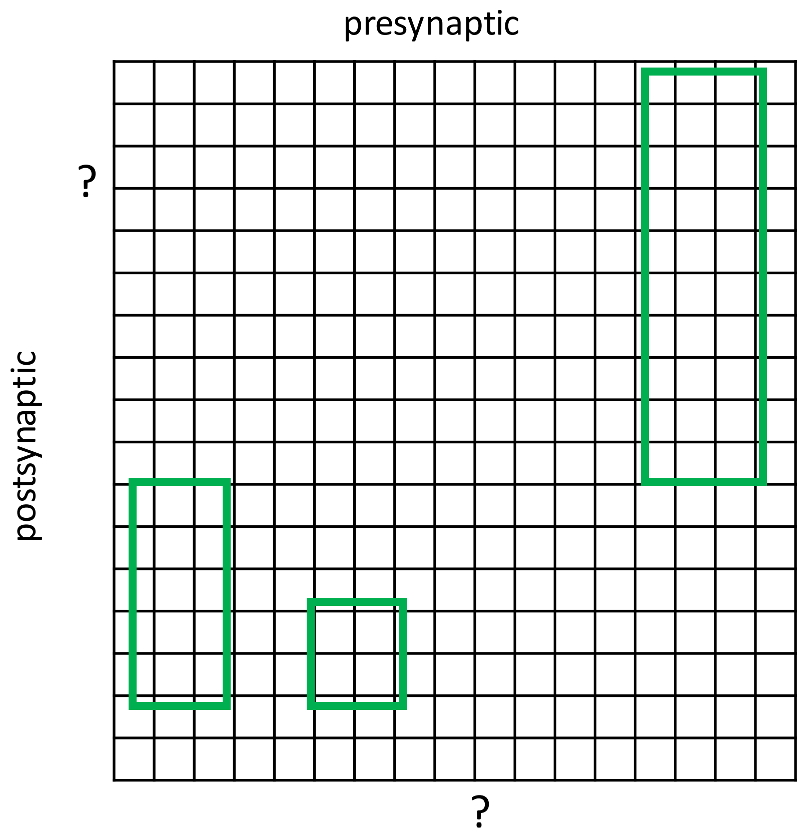


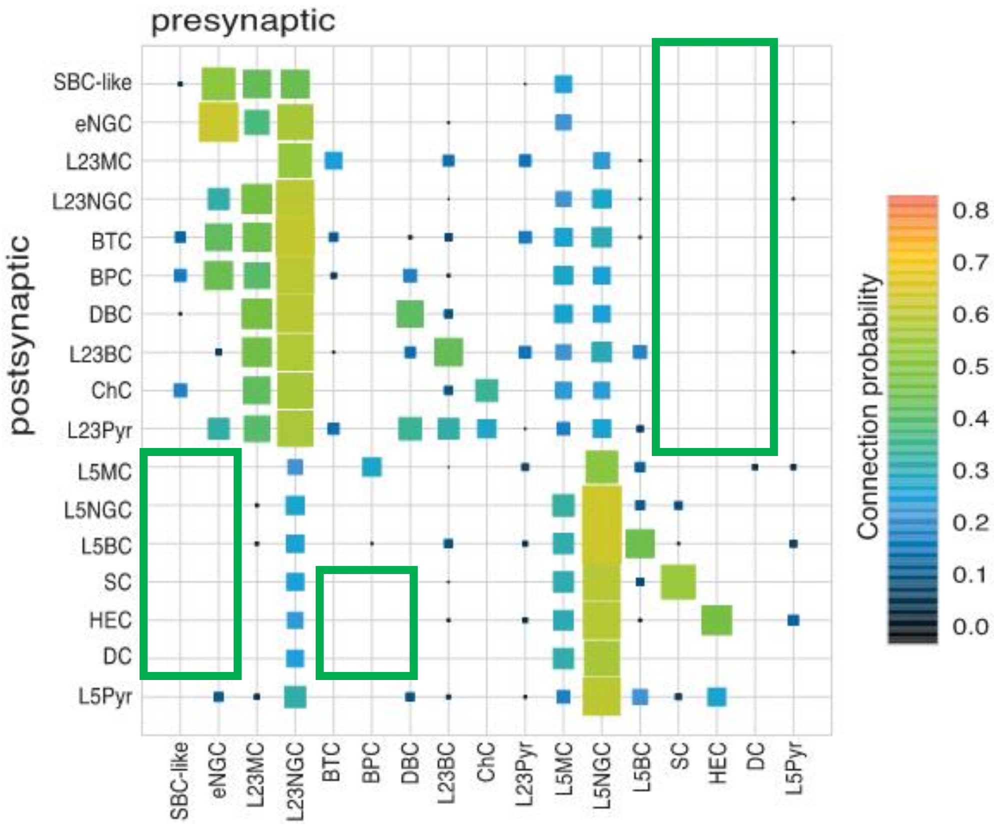
What do you notice about the connectivity?



Jiang et al., *Science*, 2015

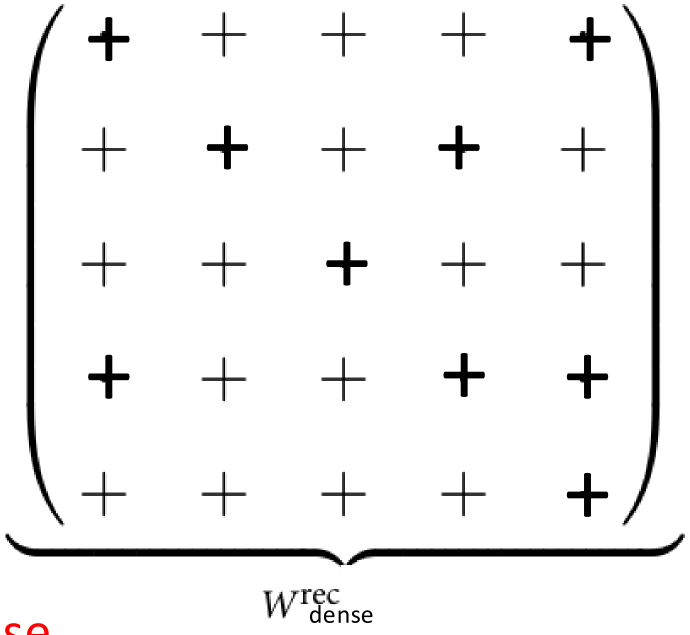
The architecture of an RNN
- sparsity





Jiang et al., *Science*, 2015

Implementing a brain-inspired architecture - sparsity



element-wise
(a.k.a. Hadamard)
product

Mask

Song et al., *eLife*, 2016

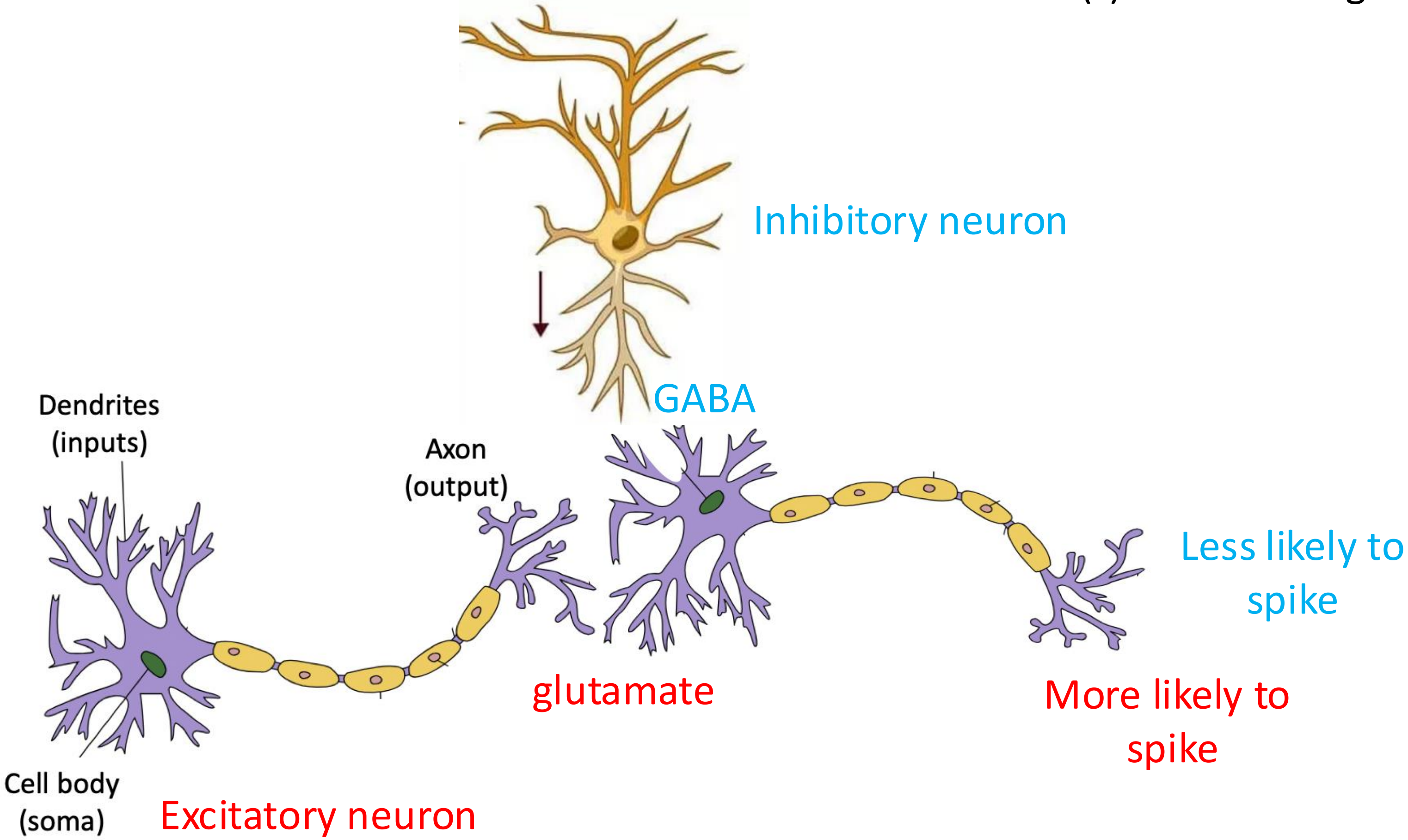
Implementing a brain-inspired architecture – Dale’s law

Retrieval practice:

What is Dale’s law?

Recap: Neurons are not all the same

The main division of neurons is between excitatory and inhibitory neurons
”Dale’s law”: a neuron releases the same neurotransmitter(s) at all its outgoing synapses.*



Inhibitory neuron

GABA

Less likely to spike

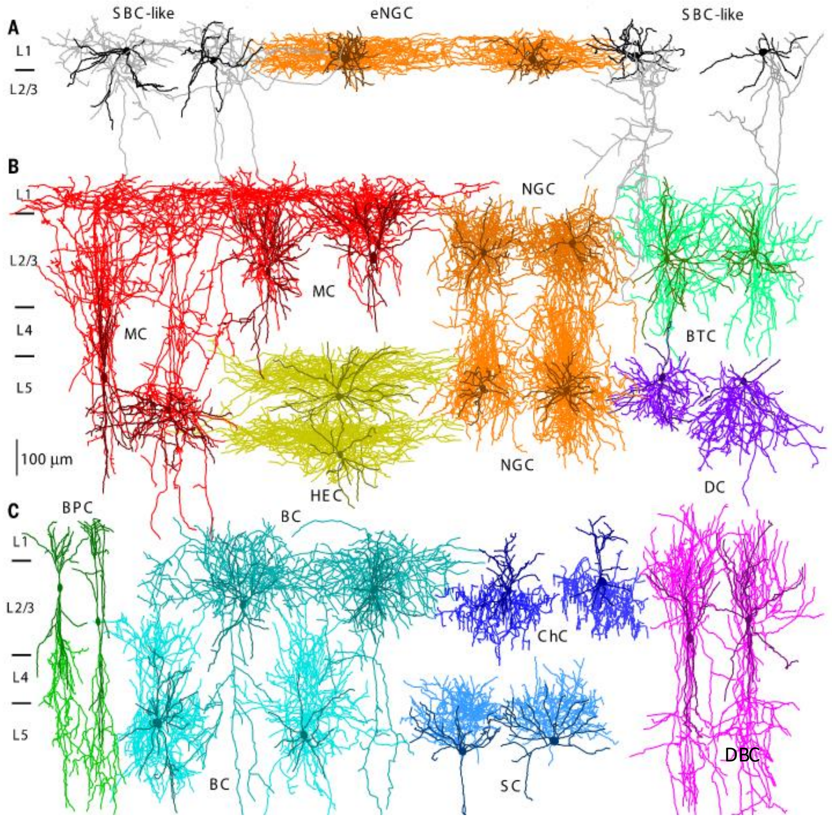
glutamate

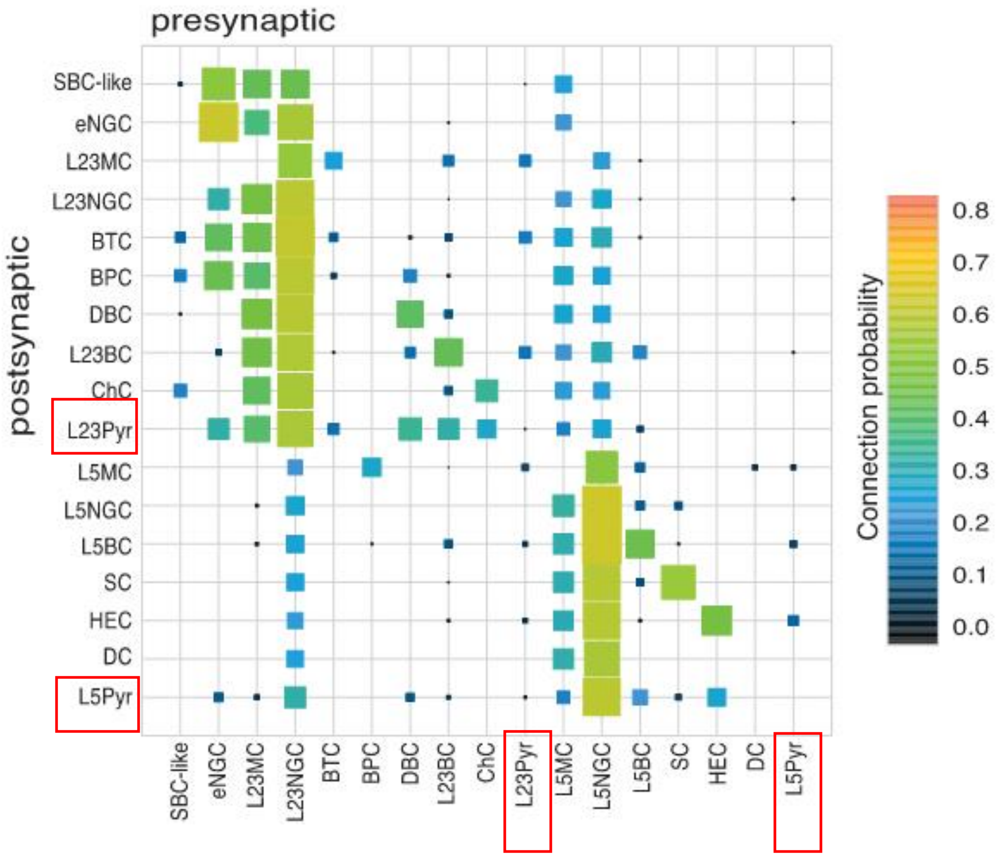
More likely to spike

Excitatory neuron

* Laws in biology never strictly work, there are always exceptions! However, Dale’s law is still a good first approximation to reality.

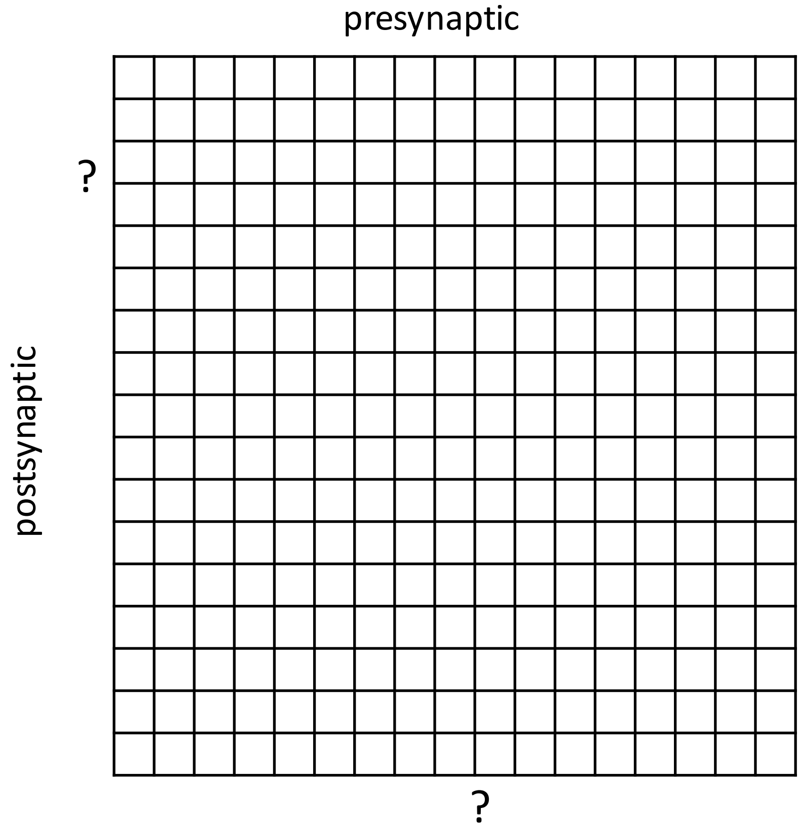
The architecture of a cortical area

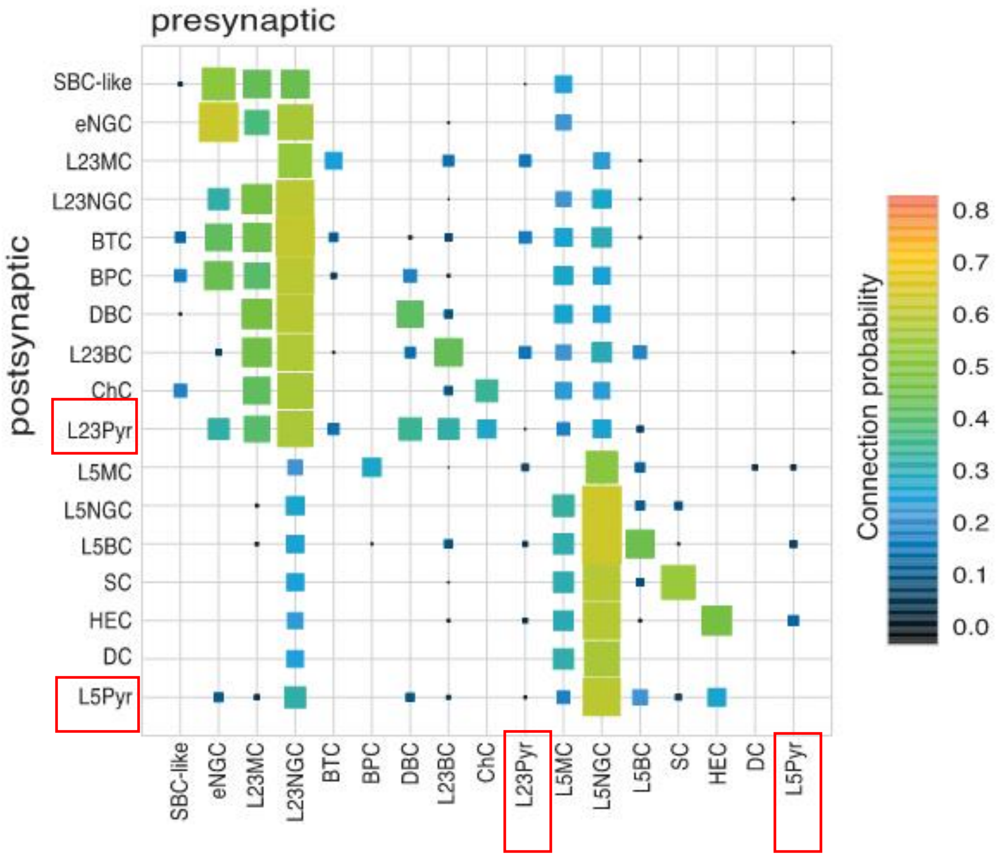




Jiang et al., *Science*, 2015

The architecture of an RNN
- cell-types





Jiang et al., *Science*, 2015

Implementing a brain-inspired architecture – Dale’s law

- modelling separate Excitatory and Inhibitory cells

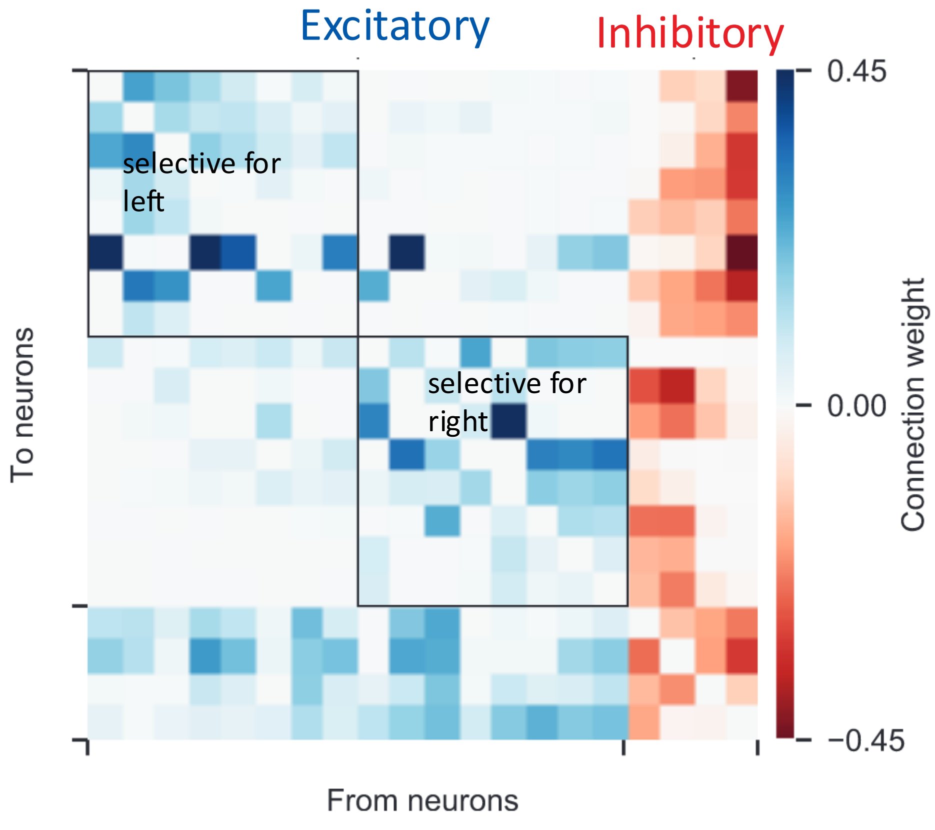
$$\underbrace{\begin{pmatrix} & + & + & + & - \\ + & & + & + & - \\ + & + & & + & - \\ + & + & + & & - \\ + & + & + & + & \end{pmatrix}}_{W^{\text{rec}}} = \underbrace{\begin{pmatrix} & + & + & + & + \\ + & & + & + & + \\ + & + & & + & + \\ + & + & + & & + \\ + & + & + & + & \end{pmatrix}}_{W^{\text{rec},+}} \underbrace{\begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & -1 \end{pmatrix}}_{D},$$

Step 1 – rectify the weights

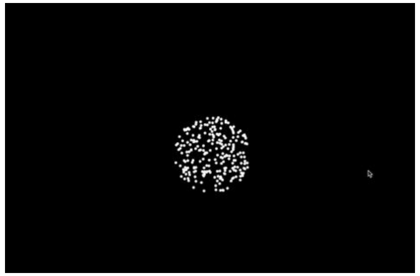
(after each learning step, set any weights below 0, to zero – a ReLU for the weights)

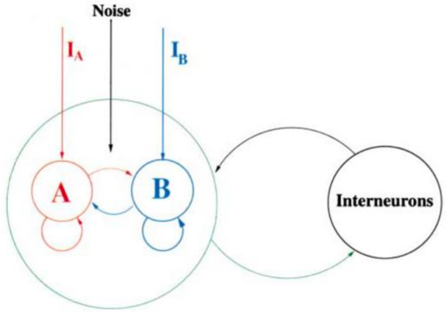
Step 2 – matrix multiply by a diagonal matrix, with 1 in the column of all Excitatory units, and -1 in the column of all Inhibitory units

Imposing biologically-motivated restrictions on RNN connectivity can lead to brain-like solutions

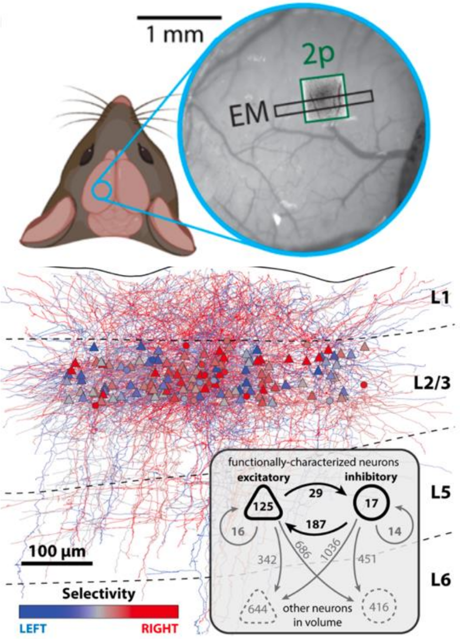


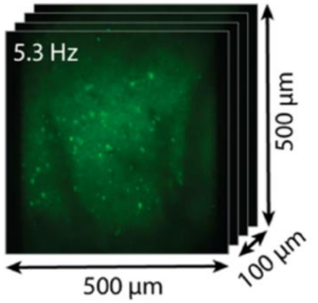
Yang & Wang, *Neuron*, 2020

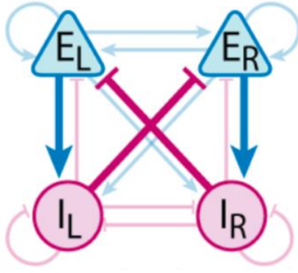


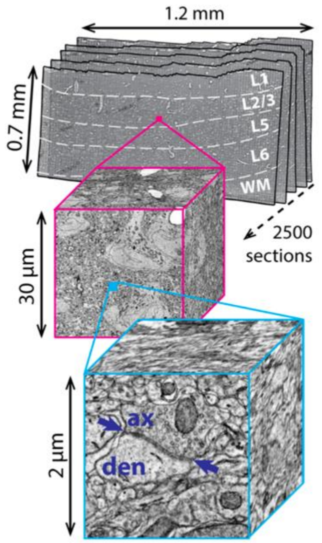


Wang, *Neuron*, 2002



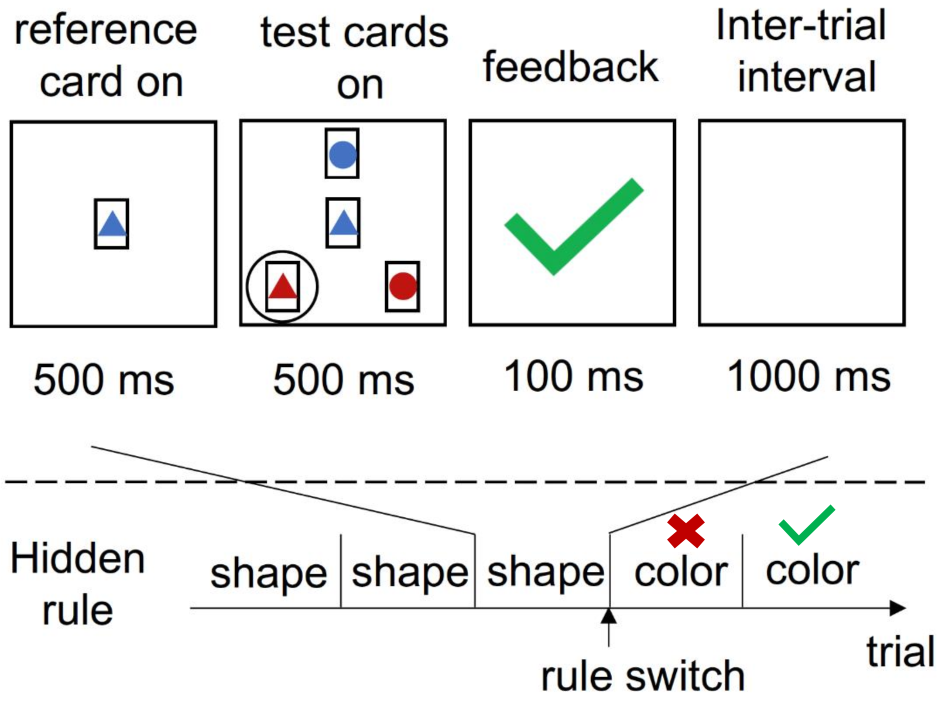






Kuan et al., *Nature*, 2024

Modelling higher cognition - the Wisconsin Card Sorting Test



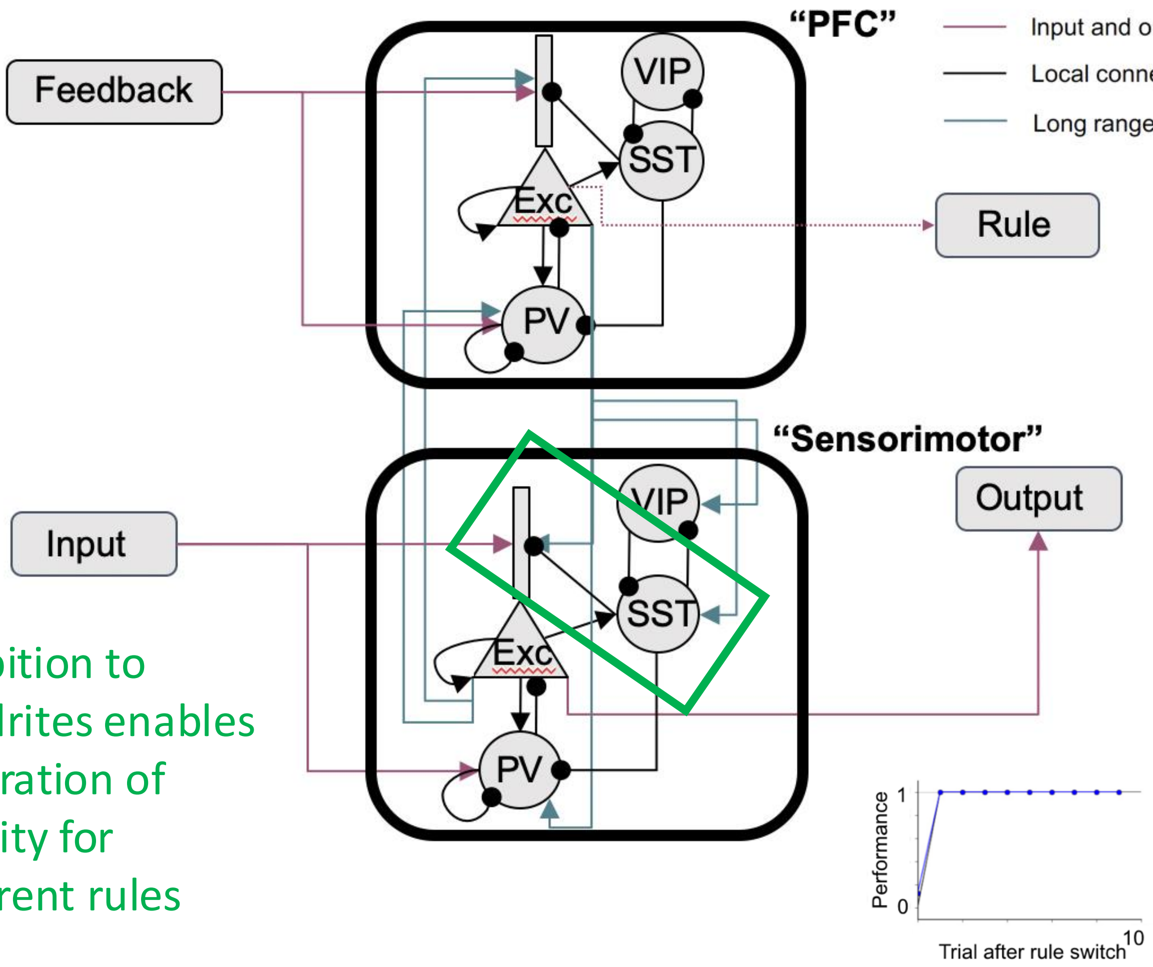
- Correct response depends on the task rule, which must be inferred from the feedback of previous trials.

- **Performance depends on the prefrontal cortex.**

Milner, Arch. Neurol. (1963)
Passingham, Neuropsychologia (1972)
Mansouri et al., J. Neurosci. (2006)
Kamigaki et al., Neuron (2009)
Purcell and Kiani., PNAS (2016)
Liu et al., Neuron (2021)
Xue et al., Neuron (2022)

Yue Liu

Implementing the connectivity of multiple cell-types to understand mechanisms of higher cognition



Inhibition to dendrites enables separation of activity for different rules

Song et al., *eLife*, 2016

$$W^{\mathrm{rec},+} = \underbrace{\begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix}}_{M^{\mathrm{rec}}} \odot \underbrace{\begin{pmatrix} \cdot & + & + & + & \cdot \\ + & \cdot & + & \cdot & + \\ + & + & \cdot & + & + \\ \cdot & + & + & \cdot & \cdot \\ + & + & + & + & \cdot \end{pmatrix}}_{W^{\mathrm{rec,plastic},+}} + \underbrace{\begin{pmatrix} 0 & 0 & 0 & 0 & w_1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_2 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}}_{W^{\mathrm{rec,fixed},+}}$$

Mask - Specific patterns of possible connections for each cell-type

Trainable weights

Fixed weights (dendrites to cell-body)

Liu & Wang, *Nature Communications*, 2024

Successfully performs complex cognitive task (Wisconsin Card Sorting Task)

Blackboard - quiz

- What is Dale’s law? How can Dale’s law be enforced on a recurrent neural network in which each unit uses the ReLU activation function?

When designing an artificial neural network to be perform a task, what *is* under our control?

- the architecture
- the cost function
- the learning rule

The cost function

- AI – choose a cost function to minimise the error in the task across the training set
- Neuro AI – choose a cost function to minimise the error in the task across the training set, and penalise the network for solutions that are not “brain-like”

The cost function - sparsity

Sparsity constraint (add penalty for weights)

Using the absolute value encourages weights to be sent to zero

$$J = J_{task} + J_{L1,weight} = J_{task} + \beta_{L1,weight} \sum_{j=1}^{N} \sum_{i=1}^{N} \left| w_{i \rightarrow j} \right|$$

total cost
(penalise poor performance
and dense weights)

regular (task) cost
(penalise poor performance)

how to balance task performance vs
sparsity of weights

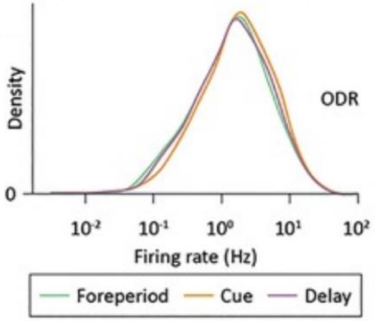
known as L^1 *regularisation of weights*

e.g. Yang et al., *Nature Neuroscience*, 2019

The cost function – low firing rates

firing rates during working memory

monkeys
(task 1)



monkeys
(task 2)

mice
(task 3)

How can we discourage RNNs from using unrealistically high firing rates?

Activity constraint
(add penalty for
high activity)

Using the square discourages
large activities

$$J = J_{task} + J_{L2,rates} = J_{task} + \beta_{L2,rates} \sum_{t=1}^{T} \sum_{i=1}^{N} a_{i,t}^{2}$$

total cost
(penalise poor
performance
and high activity)

regular (task) cost
(penalise poor performance)

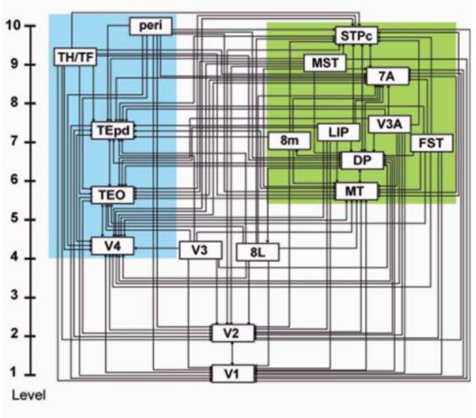
how to balance task
performance vs low activity
rates

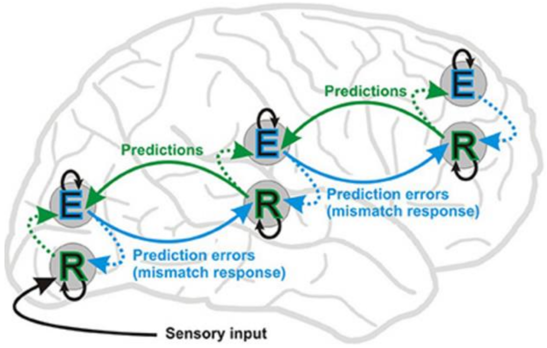
known as L^2 *regularisation of firing rates*

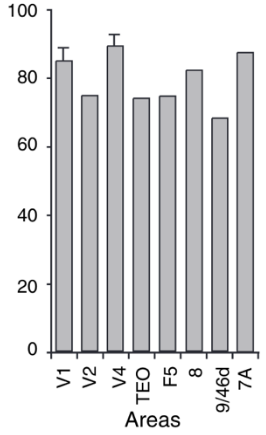
e.g. Goudar et al., *Nature Neuroscience*, 2023

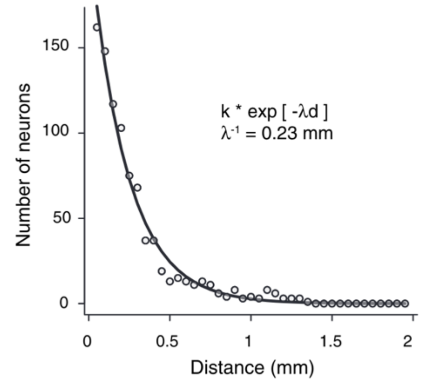
Reminder- where are the recurrent connections in the brain?

- There are many long-distance recurrent loops across brain areas.
- In primates, approximately 2/3 of all possible connections between brain areas do exist (~97% in mice).
- However most connections (~80%) are from within the same brain area (in the cortex)





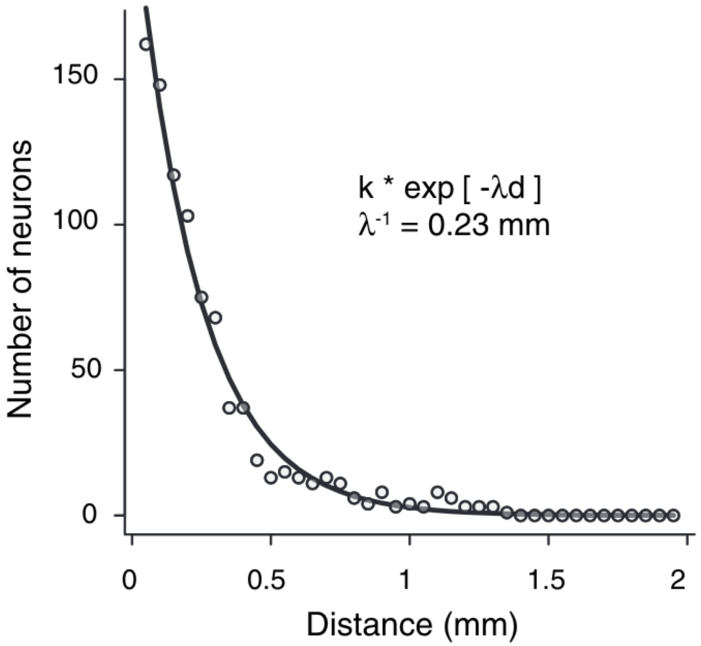




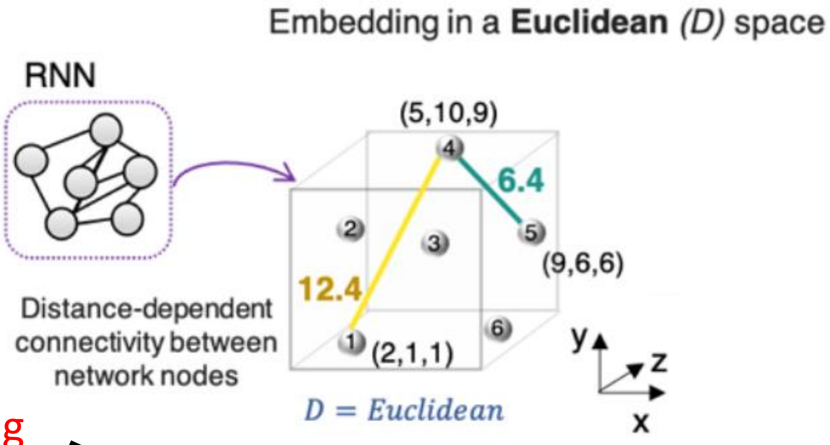
The number of connections between two brain areas decreases (exponentially) with distance.

Markov et al., *Cerebral Cortex*., 2011 Markov et al., *J. Comp. Neurol*., 2014 Stefanics et al., *Front. Hum. Neurosci, 2014*

The cost function – distance



How can we discourage RNNs from relying too much on long-distance connections?



Embedding in a Euclidean (D) space
RNN
Distance-dependent connectivity between network nodes
(5,10,9)
4
6.4
5
(9,6,6)
2
3
12.4
1
(2,1,1)
6
y
z
x
D = Euclidean

penalise strong, long connections

$$J = J_{task} + J_{WD} = J_{task} + \beta_{WD} \sum_{j=1}^{N} \sum_{i=1}^{N} |w_{i \to j}||d_{i \to j}|$$

how to balance task performance vs weighted distance

each connection incurs a penalty according to the product of weight and distance

1. Give each unit a location in space
2. Calculate the distance between each pair of units
3. Penalise strong weights between distant units

Achterberg et al., *Nature Machine Intelligence*, 2023

When designing an artificial neural network to be perform a task, what *is* under our control?

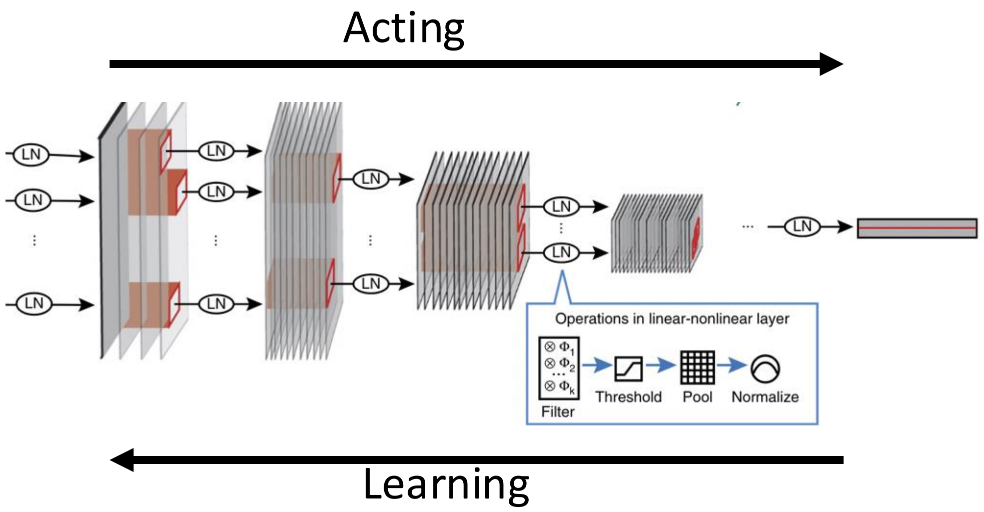
- the architecture
- the cost function
- the learning rule

Recap - the backpropagation of error algorithm (backprop) is not biologically realistic because:

1. The weights going forwards equal the weights going back
2. The weight update depends on information from distant neurons
3. The network acts (forward-propagates activity) and learns (back-propagates errors) in two separate phases

$$\frac{\delta J_0}{\delta a^{(L-1)}} = w^{(L)} f'(z^{(L)})\, 2(a^{(L)} - y)$$

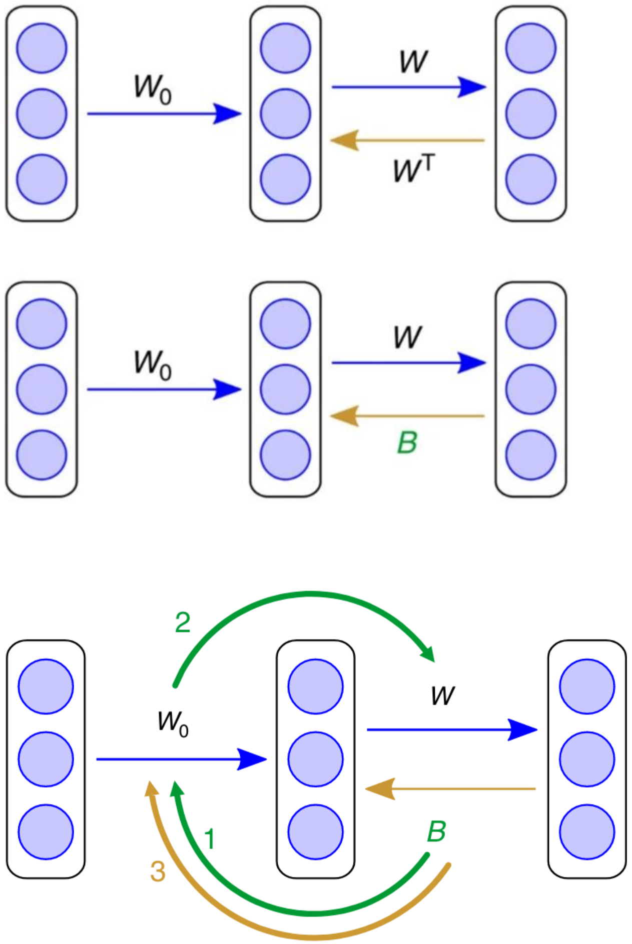
$$\frac{\delta J_0}{\delta w^{(L-1)}} = \frac{\delta z^{(L-1)}}{\delta w^{(L-1)}} \frac{\delta a^{(L-1)}}{\delta z^{(L-1)}} \frac{\delta J_0}{\delta a^{(L-1)}}$$

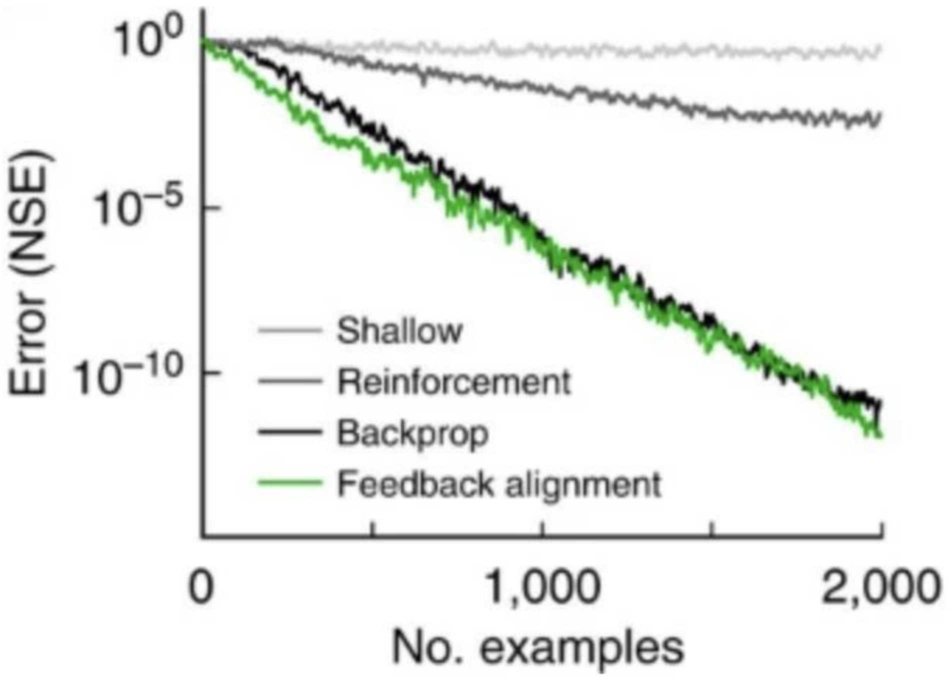


Acting

Learning

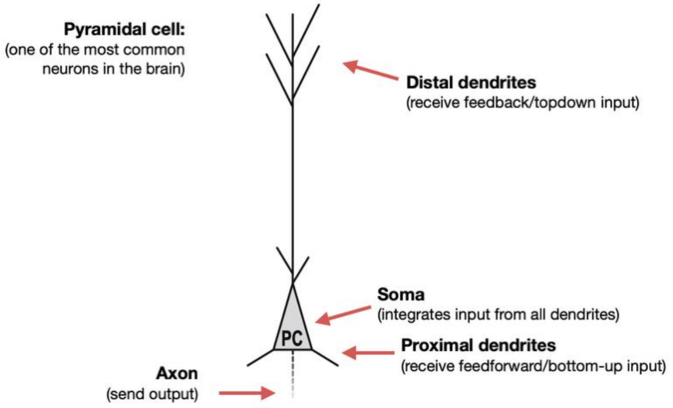
Recap – feedback alignment

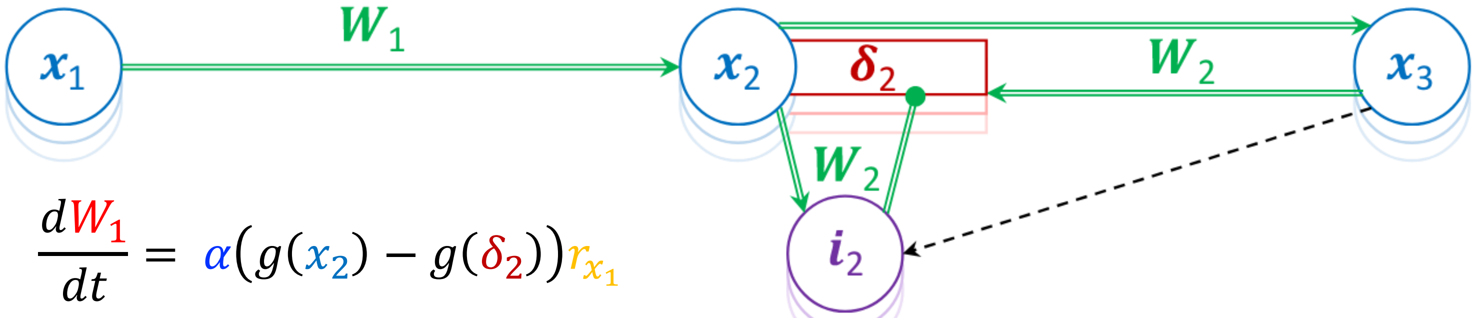




Lillicrap et al., *Nat. Comms.*, 2016

The learning rule – recap on dendritic error model


Pyramidal cell:
(one of the most common neurons in the brain)
Distal dendrites
(receive feedback/topdown input)
Soma
(integrates input from all dendrites)
PC
Proximal dendrites
(receive feedforward/bottom-up input)
Axon
(send output)



$$\frac{dW_1}{dt} = \alpha\big(g(x_2) - g(\delta_2)\big)r_{x_1}$$

The weights are updated according to a learning rate, firing rate input from the lower area, and the difference in voltage between the dendrite and cell body

When this network converges to the equilibrium, the neurons encode their corresponding error terms in their dendrites.

A single neuron is used simultaneously for activity propagation (at the cell body), error encoding (at dendrites) and error propagation to the cell-body without the need for separate phases.

Sacramento et al., *NeurIPS*, 2018
Whittington & Bogacz, *TiCS*, 2019

Biological (im)plausibility of BPTT

1. The weights going forwards equal the weights going back (like in backpropagation for feedforward networks)
2. The same neuron must store and retrieve, with perfect accuracy, the values of its activities from all points in past

The learning rule - Random Feedback Local Online (RFLO)

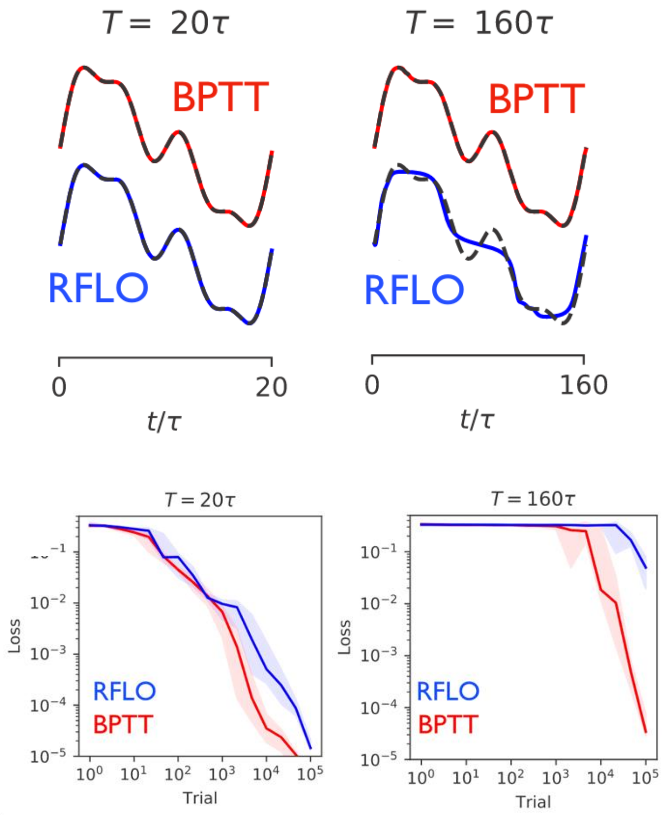
$$\partial W_{ab}(t) = \alpha\,[B\delta(t)]_a\, p_{ab}(t)$$

The change in weight between neurons a and b at time t depends on

the learning rate

the error distributed in proportion to random weights

and a moving average of recent co-activations between neuron a and b.



Strengths - only depends on information that each neuron should have (i.e. local)

no need re-use weights in the forward and backward directions

no need to run the activations backwards at the end of the trial to learn

Limitation - co-activations are forgotten at a rate determined by the neuronal timescale

$$p_{ab}(t) = \frac{1}{\tau}\phi'(u_a(t))h_b(t-1) + \left(1-\frac{1}{\tau}\right)p_{ab}(t-1),$$

Murray, *eLife*, 2019

Blackboard - quiz

- Which three components are specified by the designer in artificial neural networks? What is not specified by the designer?

Recap

- In neuro-AI, we aim to build models capable of performing complex tasks, that are also able to make testable predictions for how the brain may solve the task.
- When designing an artificial neural network to perform a task
  - the architecture
  - the cost function and
  - the learning rule
  - are under our control
- Sparse networks can be learned by using a mask, or L^1 regularisation (of weights or activity)
- Dale's law (excitatory and inhibitory units) can be implemented by rectifying the weights and multiplying by a diagonal matrix of 1s and -1s.
- Solutions with high firing rates can be penalised by imposing an L^2 regularisation on the activity
- Strong long-distance connections can be discouraged by
  - embedding each unit in a spatial position
  - calculating the distance between each pair of units
  - penalizing the product of the weight and distance of each connection
- Biologically-realistic & powerful learning is an area of ongoing research. Suggestions include
  - feedback alignment
  - dendritic error accumulation
  - random feedback local online (RFLO) learning

It’s time to give us your feedback on the unit – please complete the survey

It's quick

It's confidential

We're listening

bristol.ac.uk/unit-evaluation

Still curious? You can dive in deeper to any of today’s topics:

- Cell-type connectivity of a cortical area
  - Jiang, Xiaolong, Shan Shen, Cathryn R. Cadwell, Philipp Berens, Fabian Sinz, Alexander S. Ecker, Saumil Patel, and Andreas S. Tolias. "Principles of connectivity among morphologically defined cell types in adult neocortex." *Science* 350, no. 6264 (2015): aac9462.
- Training excitatory-inhibitory recurrent neural networks
  - Song, H. Francis, Guangyu R. Yang, and Xiao-Jing Wang. "Training excitatory-inhibitory recurrent neural networks for cognitive tasks: a simple and flexible framework." *PLoS computational biology* 12, no. 2 (2016): e1004792.
- Modelling the Wisconsin Card Sorting Task
  - Liu, Yue, and Xiao-Jing Wang. "Flexible gating between subspaces by a disinhibitory motif: a neural network model of internally guided task switching." *bioRxiv* (2023): 2023-08
- Penalising long connections
  - Achterberg, Jascha, Danyal Akarca, D. J. Strouse, John Duncan, and Duncan Astle. "Spatially-embedded recurrent neural networks reveal widespread links between structural and functional neuroscience findings." *bioRxiv* (2022): 2022-11.
- Feedback alignment
  - Lillicrap, Timothy P., Daniel Cownden, Douglas B. Tweed, and Colin J. Akerman. "Random synaptic feedback weights support error backpropagation for deep learning." *Nature communications* 7, no. 1 (2016): 13276.
- Dendritic error model
  - Sacramento, João, Rui Ponte Costa, Yoshua Bengio, and Walter Senn. "Dendritic cortical microcircuits approximate the backpropagation algorithm." *Advances in neural information processing systems* 31 (2018)
- Biologically reasonable alternative to backpropagation through time
  - Murray, James M. "Local online learning in recurrent networks with random feedback." *Elife* 8 (2019): e43299.
- Training neural networks for neuroscience research
  - Yang, Guangyu Robert, and Xiao-Jing Wang. "Artificial neural networks for neuroscientists: a primer." *Neuron* 107, no. 6 (2020): 1048-1070.