Computational models of learning and memory: synaptic plasticity and Hopfield networks

Bill Podlaski

isiCNI2019
11 Jan 2019

Outline

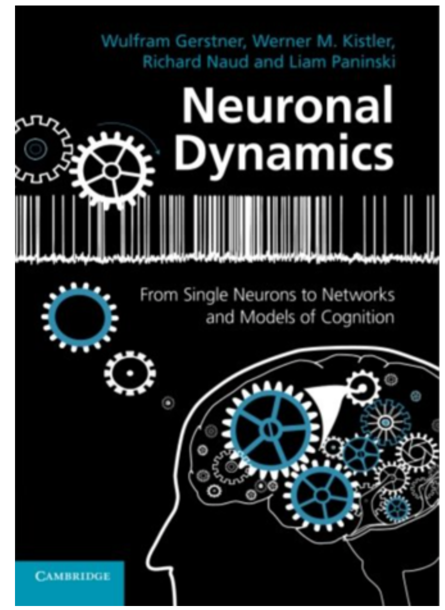
1. **Introduction to learning and memory**
    - The biology of memory and synaptic plasticity
2. **Models of synaptic plasticity**
    - Hebbian learning and stability
    - Rate-based plasticity: BCM and Oja’s rule
    - Spike-based plasticity: STDP
3. **Models of associative memory**
    - The Hopfield network
    - Estimating memory capacity
    - Incremental learning
4. **Summary and open questions**

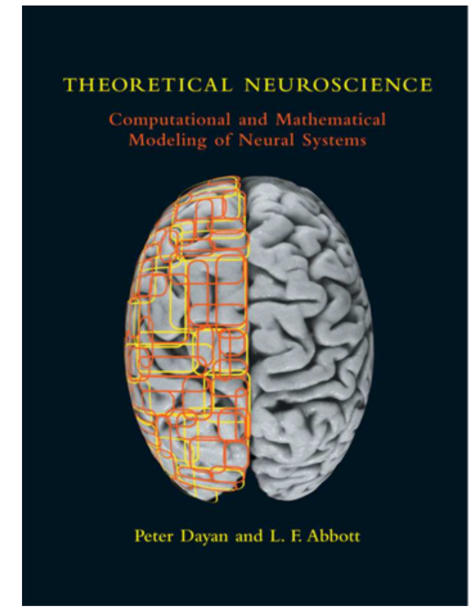
Outline

1. **Introduction to learning and memory**
    - The biology of memory and synaptic plasticity
2. **Models of synaptic plasticity**
    - Hebbian learning and stability
    - Rate-based plasticity: BCM and Oja's rule
    - Spike-based plasticity: STDP
3. **Models of associative memory**
    - The Hopfield network
    - Estimating memory capacity
    - Incremental learning
4. **Summary and open questions**

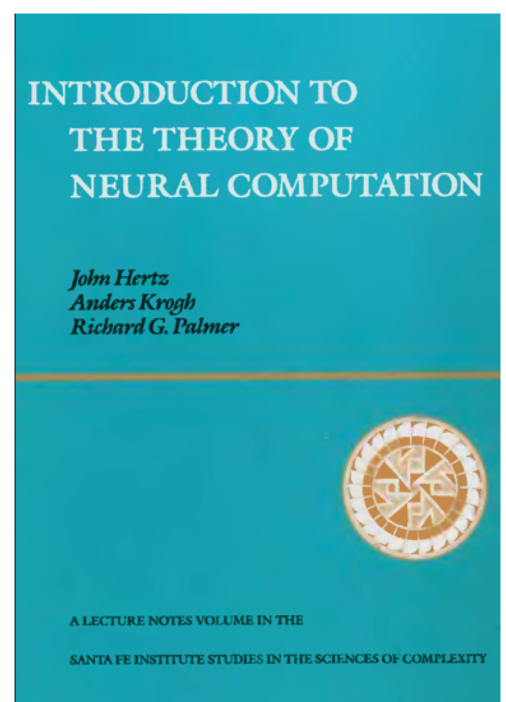
Acknowledgements and resources

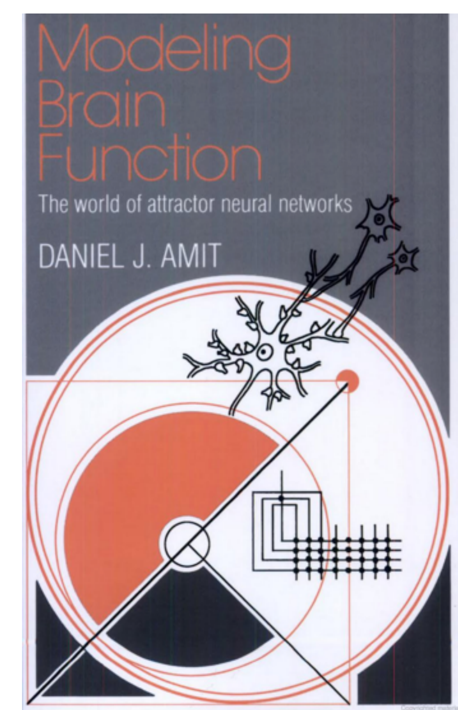
Synaptic plasticity slides
adapted/stolen
from Katharina Wilmes





https://neuronaldynamics.epfl.ch/





Learning and memory

- Learning and memory are the basis for all adaptive behavior
    - **Learning** is the biological process of acquiring new knowledge about the world
    - **Memory** is the process of retaining and reconstructing that knowledge over time

Kandel, Dudai & Mayford 2014

- **Many flavors:** synaptic plasticity (Srikanth), reinforcement learning (Alex), motor control (Byron), working memory (Callie), sensory processing (Grace), & more

Types of learning problems

- Supervised learning
  - Requires labelled data
  - Typical machine learning setting — e.g., regression, classification
- Reinforcement learning
  - Feedback from environment
- Unsupervised learning
  - Learning strictly from the statistics of the input — no labels!
  - E.g., clustering, feature extraction

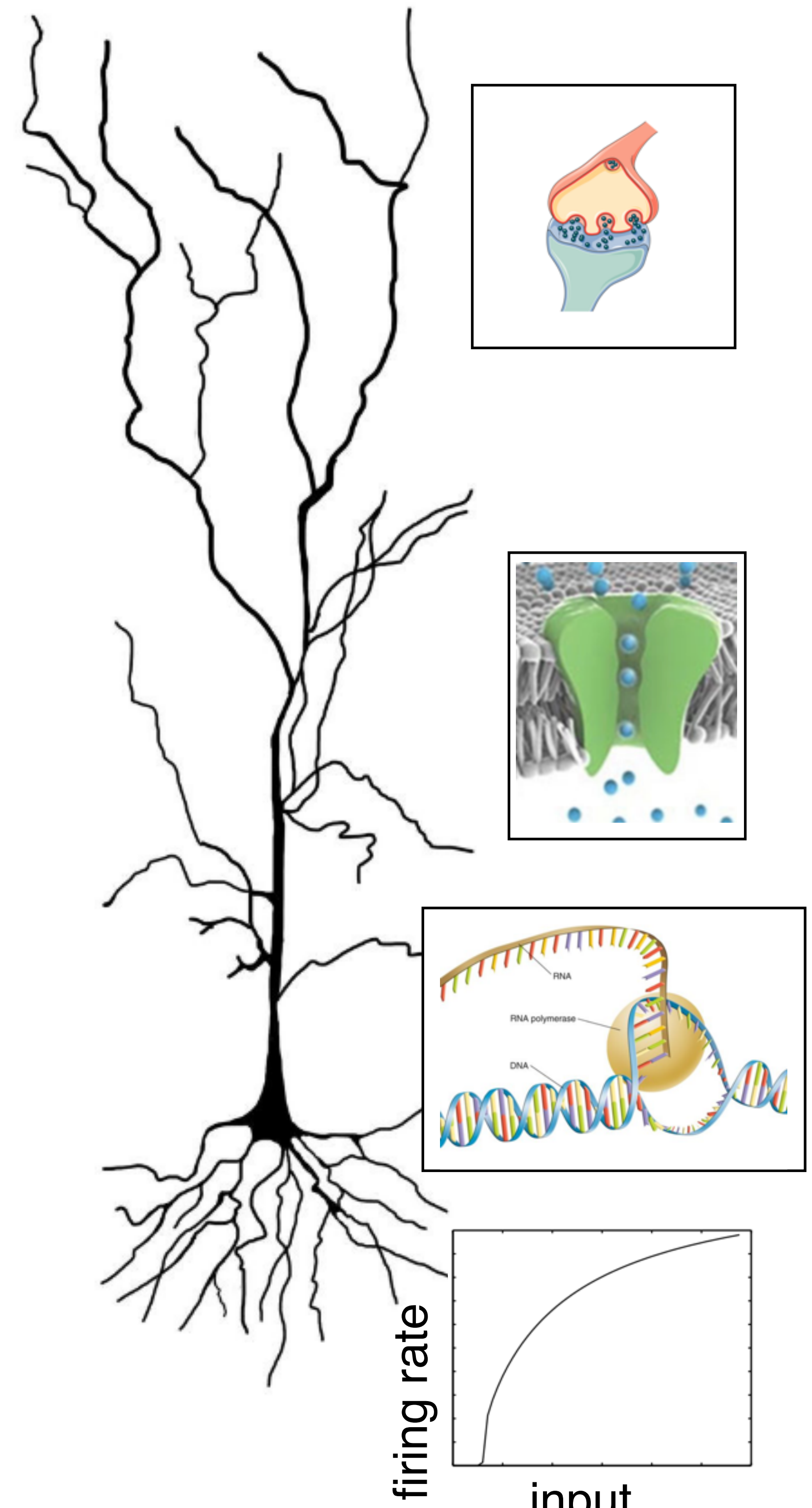
The biology of learning and memory

- Memory requires the retention of information over long timescales

- What is the “state variable” of memory in the brain?
    - Short-term: firing rate dynamics, calcium concentration, synaptic vesicle state, adaptation, …
    - Long-term: mRNA levels, phosphorylation levels, **synaptic strength**, …

- **Synaptic plasticity and memory (SPM) hypothesis:** activity-dependent synaptic plasticity is both necessary and sufficient for the information storage underlying memory formation in the brain *(Martin, Grimwood & Morris 2000)*

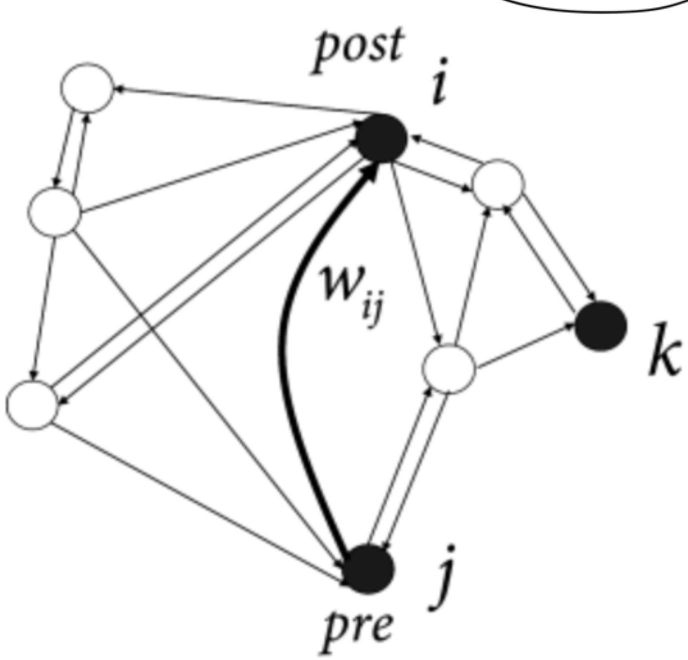
- **Synaptic weight as a proxy for the synaptic machinery**



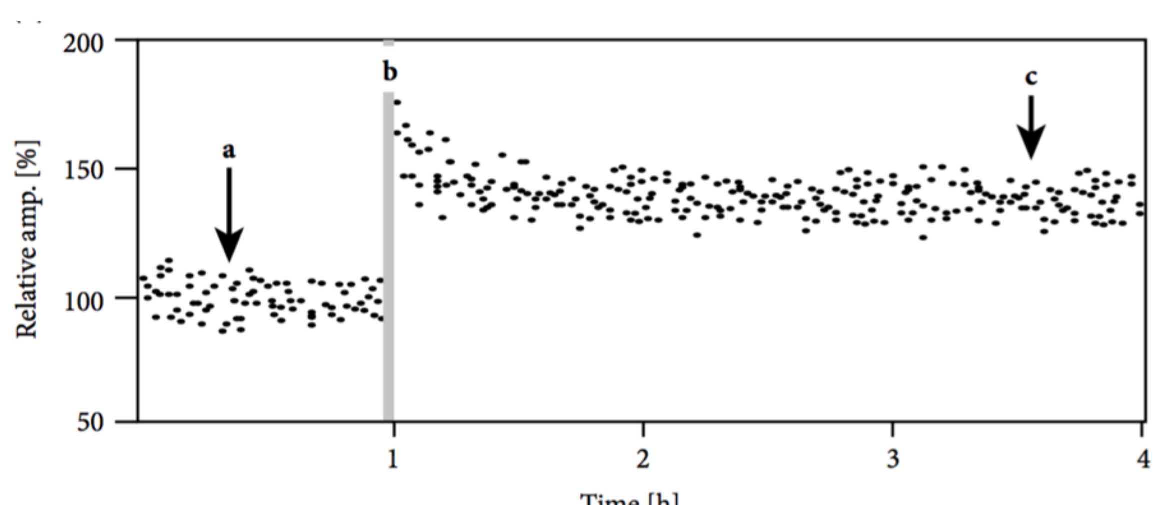
Synaptic plasticity and Hebb's postulate

- **Hebb's postulate:** if one neuron repeatedly and consistently contributes to the firing of another neuron, the connection between them should strengthen *(Hebb 1949)*

  "fire together, wire together" *(Shatz 1992)*



- Experimental evidence: long-term potentiation (LTP) — Bliss & Lomo 1973



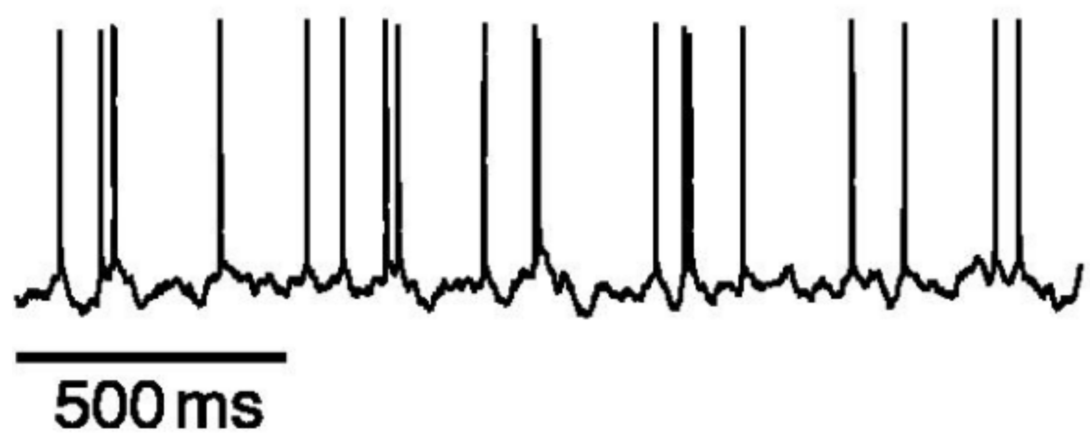
Gerstner et al. 2014

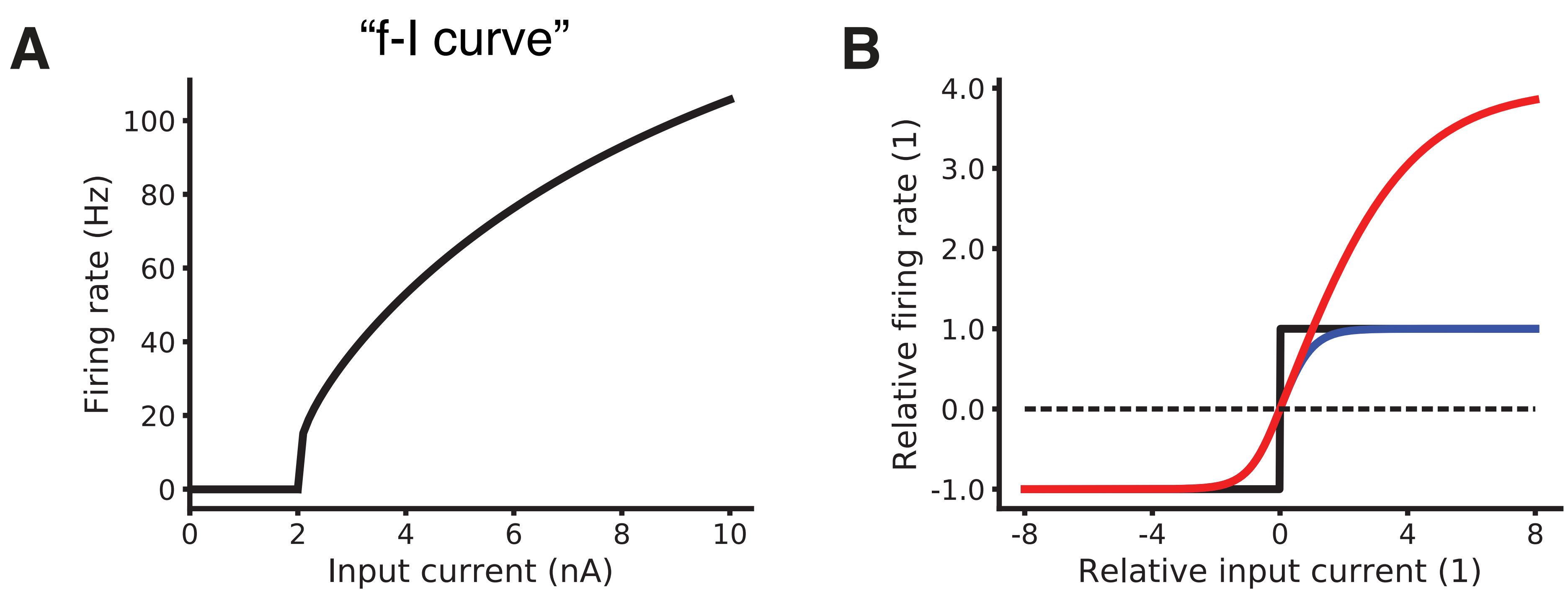
Outline

1. **Introduction to learning and memory**
    - The biology of memory and synaptic plasticity
2. **Models of synaptic plasticity**
    - Hebbian learning and stability
    - Rate-based plasticity: BCM and Oja's rule
    - Spike-based plasticity: STDP
3. **Models of associative memory**
    - The Hopfield network
    - Estimating memory capacity
    - Incremental learning
4. **Summary and open questions**

Review: phenomenological neuron models

- Spiking neuron models (IAF)
- Firing-rate models
- Binary models



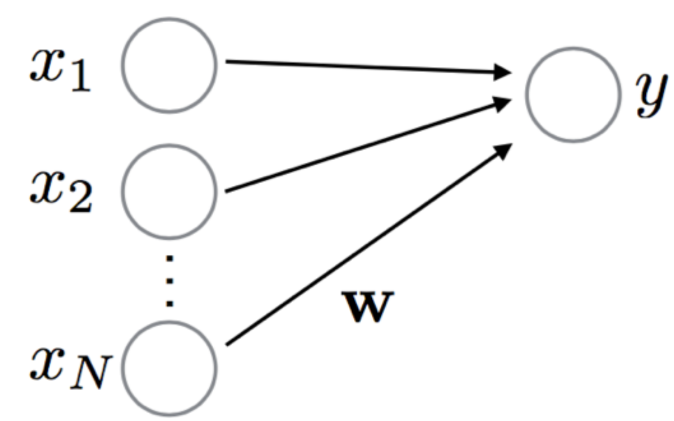


Rate-based models of synaptic plasticity

- What do synaptic plasticity learning rules look like in the brain?

- Assumptions: smooth function, only uses local information

$$\frac{dw_i}{dt} = F(x_i, y)$$



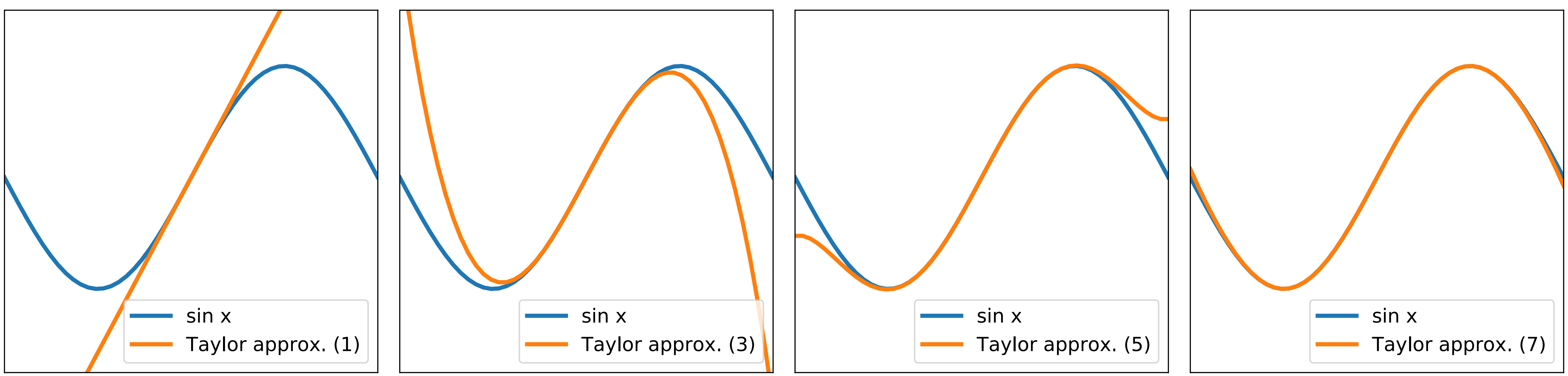
$$y = \sum_i w_i x_i$$

Mathematical digression: Taylor series expansion

- Every smooth function can be rewritten as an infinite sum of polynomial terms. The Taylor formula gives us the polynomial expansion of any smooth function.

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \frac{f^{(3)}(x_0)}{3!}(x - x_0)^3 + \ldots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n + \ldots$$

e.g.
sin(x)



- 2d formulation:

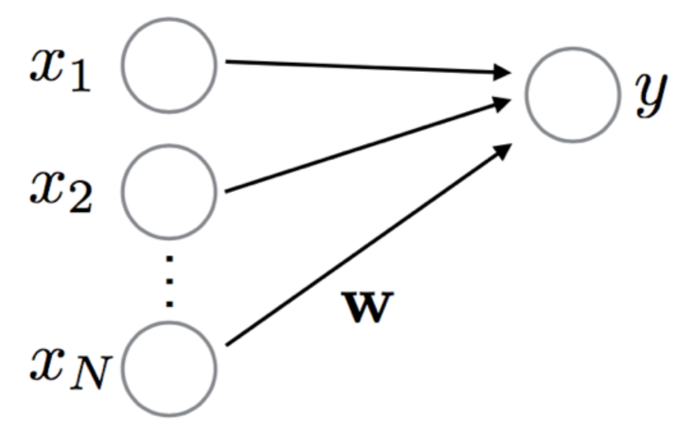
$$\begin{aligned} f(x,y) &= f(x_0,y_0) + f_x(x_0,y_0)(x-x_0) + f_y(x_0,y_0)(y-y_0) + \tfrac{1}{2} f_{xx}(x_0,y_0)(x-x_0)^2 \\ &+ f_{xy}(x_0,y_0)(x-x_0)(y-y_0) + \tfrac{1}{2} f_{xx}(x_0,y_0)(y-y_0)^2 + \ldots \end{aligned}$$

Rate-based models of synaptic plasticity

- What do synaptic plasticity learning rules look like in the brain?

- Assumptions: smooth function, only uses local information

$$\frac{dw_i}{dt} = F(x_i, y)$$

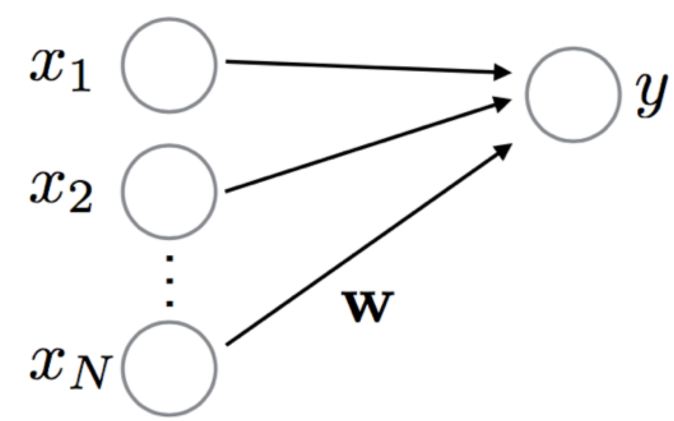


$$y = \sum_i w_i x_i$$

Rate-based models of synaptic plasticity

- What do synaptic plasticity learning rules look like in the brain?

- Assumptions: smooth function, only uses local information



$$y = \sum_i w_i x_i$$

$$\frac{dw_i}{dt} = F(x_i, y)$$

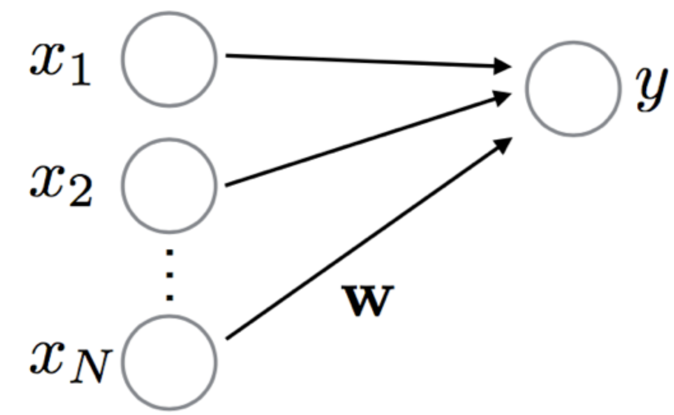
$$\frac{dw_i}{dt} = a_0 + a_1 x_i + a_2 y + a_3 x_i y + a_4 x_i^2 + a_5 y^2 + \ldots$$

Gerstner et al. 2014

Rate-based models of synaptic plasticity

- What do synaptic plasticity learning rules look like in the brain?

- Assumptions: smooth function, only uses local information



$$y = \sum_i w_i x_i$$

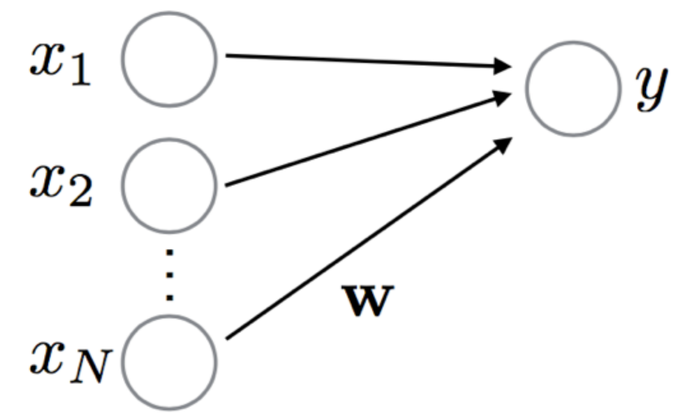
$$\frac{dw_i}{dt} = F(x_i, y, w)$$

$$\frac{dw_i}{dt} = a_0(w) + a_1(w)x_i + a_2(w)y + a_3(w)x_i y + a_4(w)x_i^2 + a_5(w)y^2 + ...$$

Gerstner et al. 2014

Rate-based models of synaptic plasticity

- What do synaptic plasticity learning rules look like in the brain?
- Assumptions: smooth function, only uses local information



$$y = \sum_i w_i x_i$$

$$\frac{dw_i}{dt} = F(x_i, y, w)$$

$$\frac{dw_i}{dt} = a_0(w) + a_1(w)x_i + a_2(w)y + a_3(w)x_i y + a_4(w)x_i^2 + a_5(w)y^2 + ...$$

Post ν_i	Pre ν_j	$\mathrm{d}w_{ij}/\mathrm{d}t \propto \nu_i \nu_j$
ON	ON	+
ON	OFF	0
OFF	ON	0
OFF	OFF	0

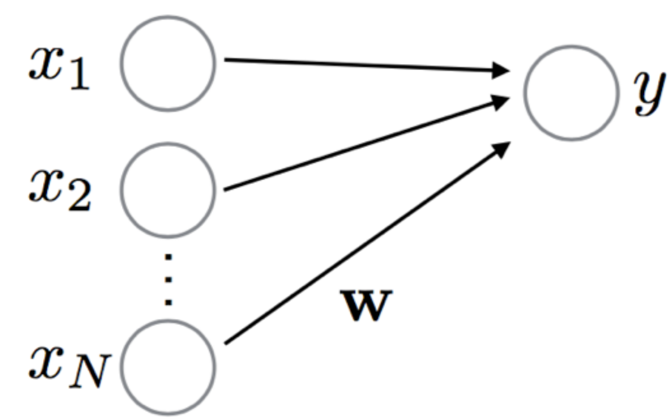
Gerstner et al. 2014

Hebbian is correlation-based learning

- Simplest Hebbian learning rule:

$$\frac{dw_i}{dt} = \eta x_i y$$

$$\frac{d\mathbf{w}}{dt} = \eta y \mathbf{x} \qquad \textit{(vector notation)}$$



$$y = \sum_i w_i x_i = \mathbf{x}^T \mathbf{w}$$

- What does the Hebbian rule learn?

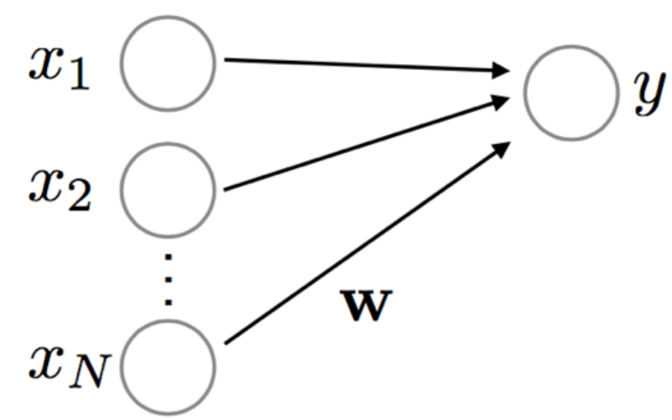
$$\tau_w \frac{d\mathbf{w}}{dt} = \langle y \mathbf{x} \rangle_x$$

Hebbian is correlation-based learning

- Simplest Hebbian learning rule:

$$\frac{dw_i}{dt} = \eta x_i y$$

$$\frac{d\mathbf{w}}{dt} = \eta y \mathbf{x} \qquad \textit{(vector notation)}$$



$$y = \sum_i w_i x_i = \mathbf{x}^T \mathbf{w}$$

- What does the Hebbian rule learn?

$$\tau_w \frac{d\mathbf{w}}{dt} = \langle y\mathbf{x} \rangle_x = \langle \mathbf{x}\mathbf{x}^T \rangle_x \mathbf{w} = \mathbf{Q}\mathbf{w}$$

input correlation matrix

Stability of Hebbian learning

- We can assess the stability of the rule by looking at the derivative as a function of the output (assume constant input):

$$\frac{dw_i}{dt} = \eta x_i y$$

- Or we can look at the dynamics of the vector norm:

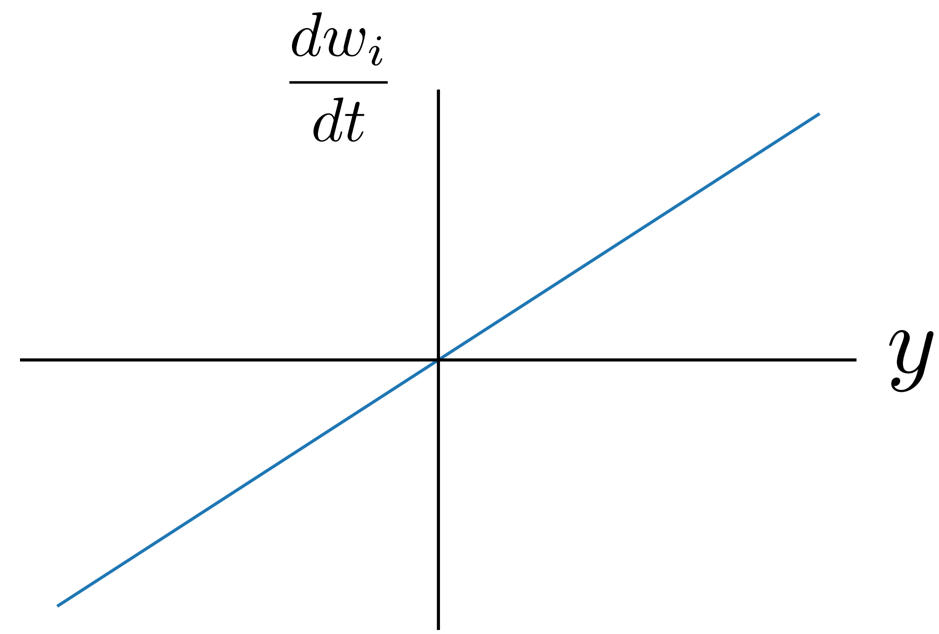
$$||\mathbf{w}||^2 = \mathbf{w}^T\mathbf{w} = \sum_i w_i^2$$

$$\frac{d||\mathbf{w}||^2}{dt} = 2\mathbf{w}^T\frac{d\mathbf{w}}{dt}$$

Stability of Hebbian learning

- We can assess the stability of the rule by looking at the derivative as a function of the output (assume constant input):

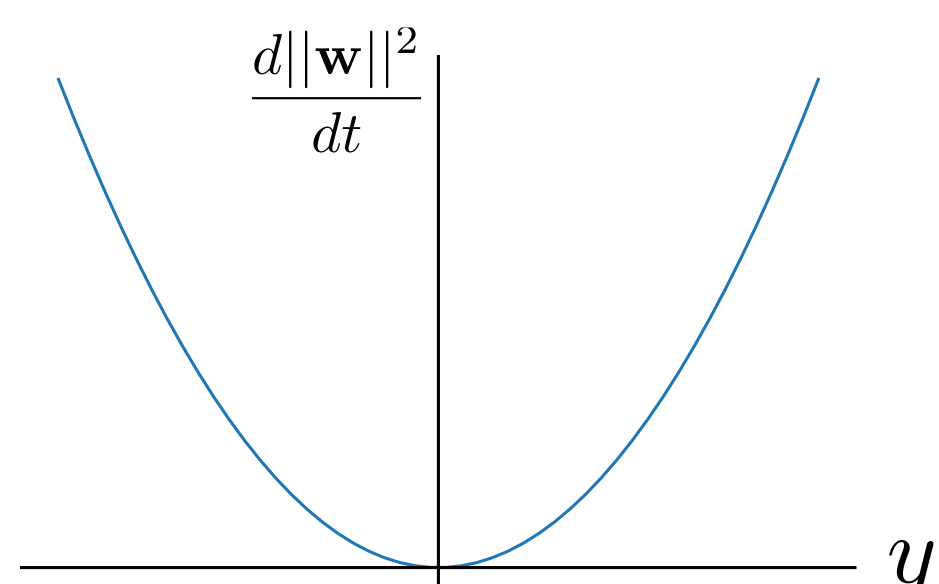
$$\frac{dw_i}{dt} = \eta x_i y$$



- Or we can look at the dynamics of the vector norm:

$$||\mathbf{w}||^2 = \mathbf{w}^T\mathbf{w} = \sum_i w_i^2$$

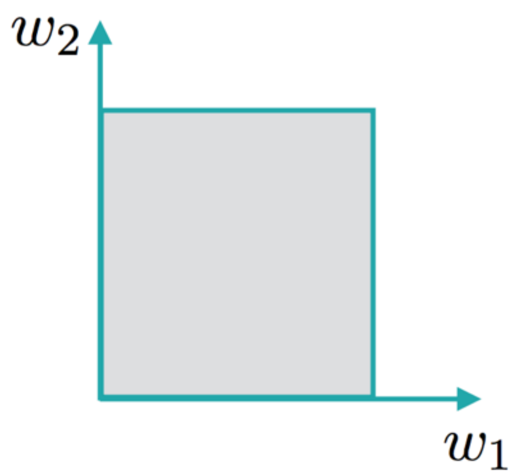
$$\frac{d||\mathbf{w}||^2}{dt} = 2\mathbf{w}^T\frac{d\mathbf{w}}{dt} = 2\eta y \mathbf{w}^T\mathbf{x} = 2\eta y^2$$



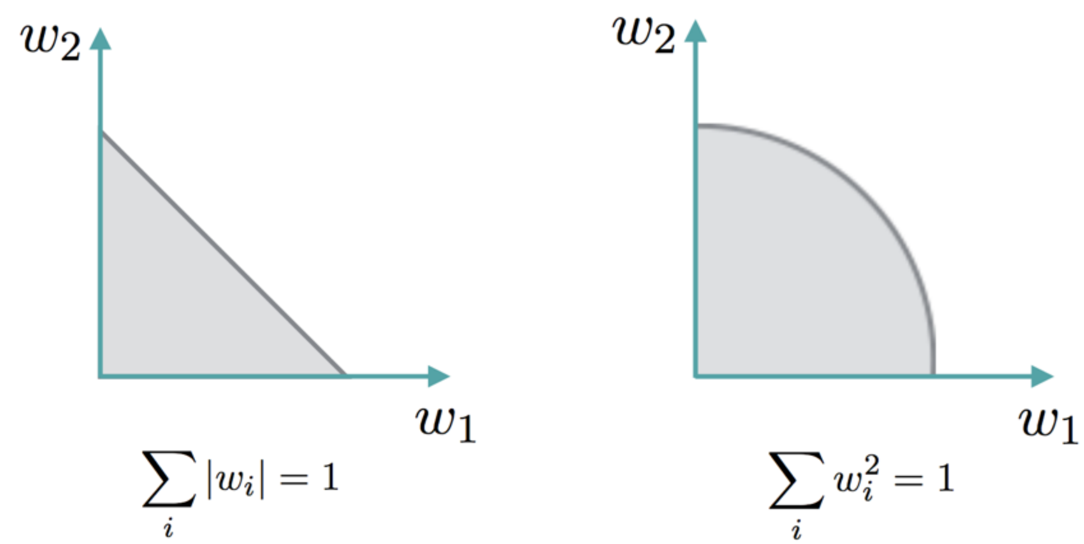
- Simple Hebbian learning is unstable!

How can we make Hebb stable?

(1) Hard bounds on the weights



(2) Soft bounds (normalization)



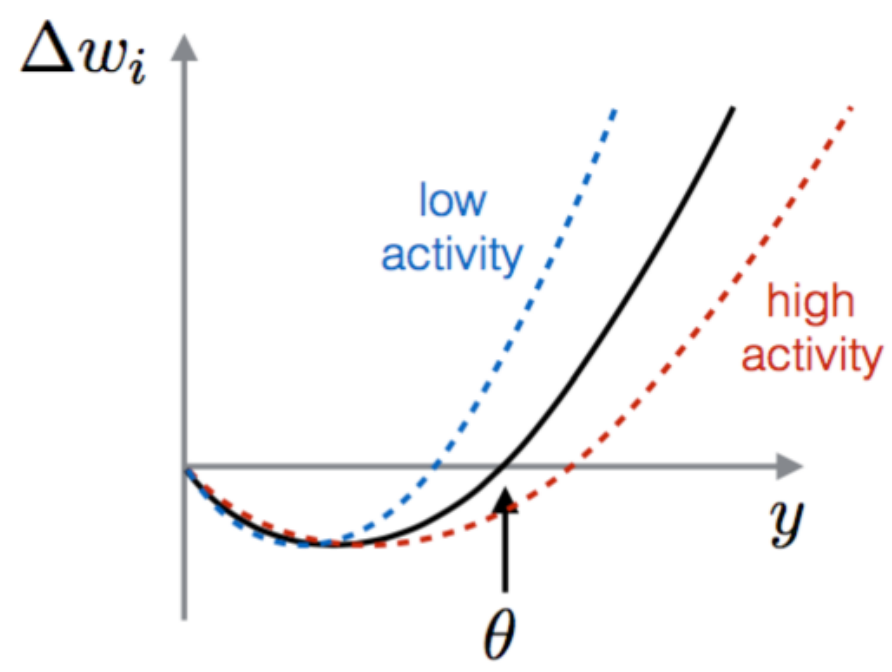
(3) Implicit normalization by introducing other terms

Post v_i	Pre v_j	$dw_{ij}/dt \propto v_i v_j$	$dw_{ij}/dt \propto v_i v_j - c_0$	$dw_{ij}/dt \propto (v_i - v_\theta) v_j$	$dw_{ij}/dt \propto v_i (v_j - v_\theta)$	$dw_{ij}/dt \propto (v_i - \langle v_i \rangle)(v_j - \langle v_j \rangle)$
ON	ON	+	+	+	+	+
ON	OFF	0	–	0	–	–
OFF	ON	0	–	–	0	–
OFF	OFF	0	–	0	0	+

Bienenstock-Cooper-Munro (BCM) rule

- Uses a sliding threshold which controls relative levels of potentiation versus depression

$$\frac{d\mathbf{w}}{dt} = \eta y(y - \theta)\mathbf{x} \qquad \theta = \frac{\langle y \rangle^p}{y_0^{p-1}}$$



★ Introduces competition between different inputs

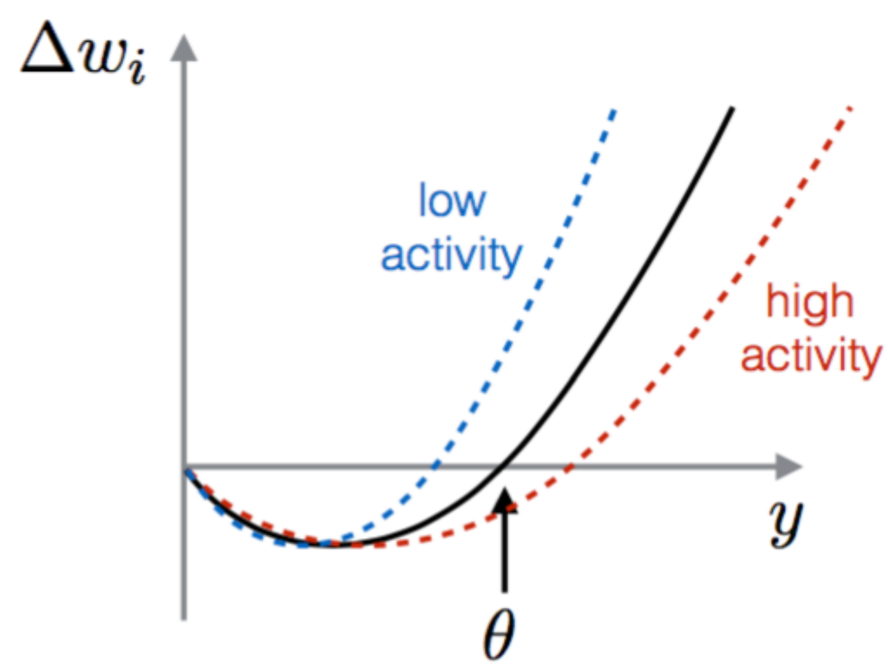
- When will this learning rule be stable?

Bienenstock, Cooper & Munro 1982

Bienenstock-Cooper-Munro (BCM) rule

- Uses a sliding threshold which controls relative levels of potentiation versus depression

$$\frac{d\mathbf{w}}{dt} = \eta y(y - \theta)\mathbf{x} \qquad \theta = \frac{\langle y \rangle^p}{y_0^{p-1}}$$



★ Introduces competition between different inputs

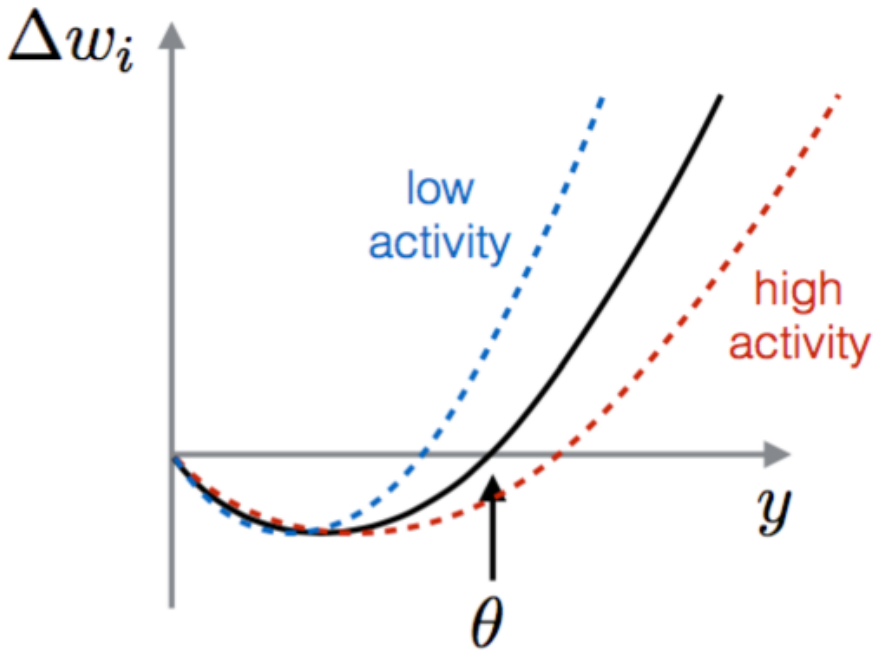
- When will this learning rule be stable?
  - Stable if the threshold grows more rapidly than the output firing rate, i.e., $p > 1$

Bienenstock, Cooper & Munro 1982

Bienenstock-Cooper-Munro (BCM) rule

- Uses a sliding threshold which controls relative levels of potentiation versus depression

$$\frac{d\mathbf{w}}{dt} = \eta y(y - \theta)\mathbf{x} \qquad\qquad \theta = \frac{\langle y \rangle^p}{y_0^{p-1}}$$



★ Introduces competition between different inputs

- When will this learning rule be stable?
    - Stable if the threshold grows more rapidly than the output firing rate, i.e., $p > 1$

Problem set questions 1.1, 1.3

Bienenstock, Cooper & Munro 1982

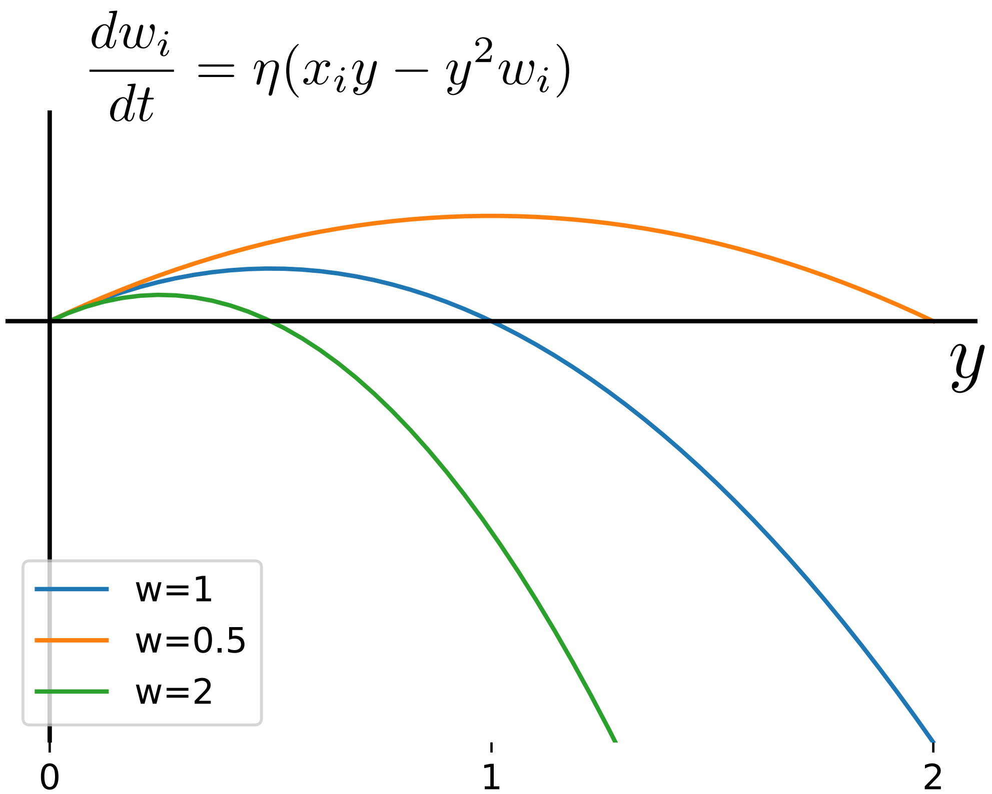
Oja's rule

- Implicit multiplicative weight normalization

$$\frac{d\mathbf{w}}{dt} = \eta(y\mathbf{x} - y^2\mathbf{w})$$

★ Like BCM, Oja's rule also introduces competition between different inputs

- Stability analysis



Oja 1982

Oja's rule

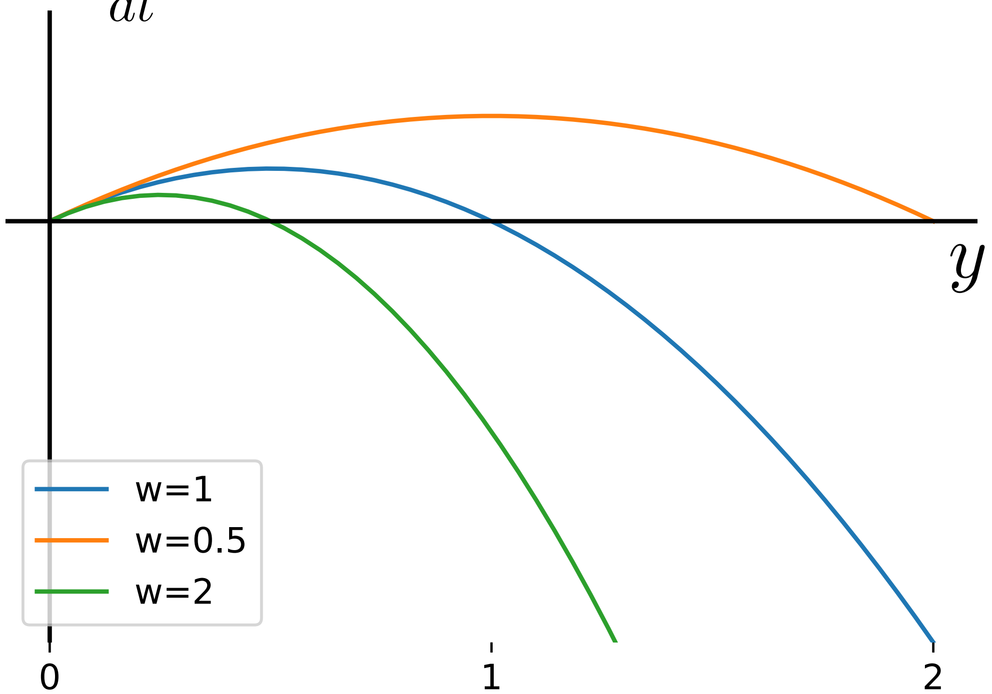
- Implicit multiplicative weight normalization

$$\frac{d\mathbf{w}}{dt} = \eta(y\mathbf{x} - y^2\mathbf{w})$$

★ Like BCM, Oja's rule also introduces competition between different inputs

- Stability analysis

$$\frac{dw_i}{dt} = \eta(x_i y - y^2 w_i)$$



Problem set question 1.2a

Oja 1982

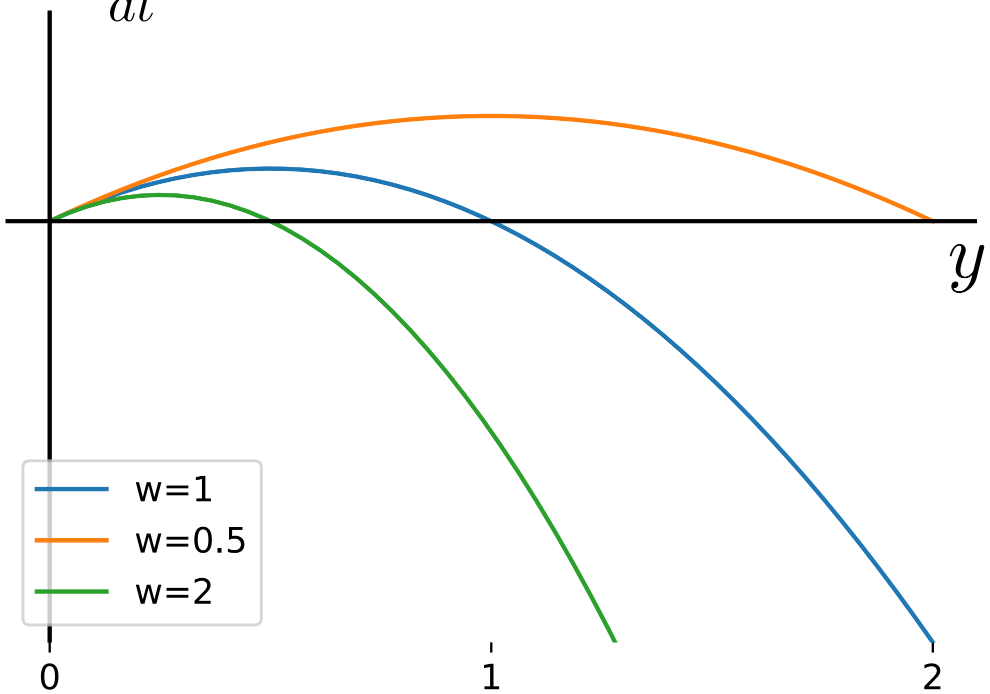
Oja's rule

- Implicit multiplicative weight normalization

$$\frac{d\mathbf{w}}{dt} = \eta(y\mathbf{x} - y^2\mathbf{w})$$

★ Like BCM, Oja's rule also introduces competition between different inputs

$$\frac{dw_i}{dt} = \eta(x_i y - y^2 w_i)$$



- Stability analysis

$$\frac{d||\mathbf{w}||^2}{dt} = 2\mathbf{w}^T\frac{d\mathbf{w}}{dt} = 2\mathbf{w}^T\eta(\mathbf{x}y - y^2\mathbf{w}) = 2\eta(y^2 - y^2\mathbf{w}^T\mathbf{w}) = 2\eta y^2(1 - ||\mathbf{w}||^2)$$

$$||\mathbf{w}||^2 = \sum_i w_i^2 = 1$$

Problem set question 1.2a

Oja 1982

Linear algebra review & principal components analysis (PCA)

Eigendecomposition: $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$

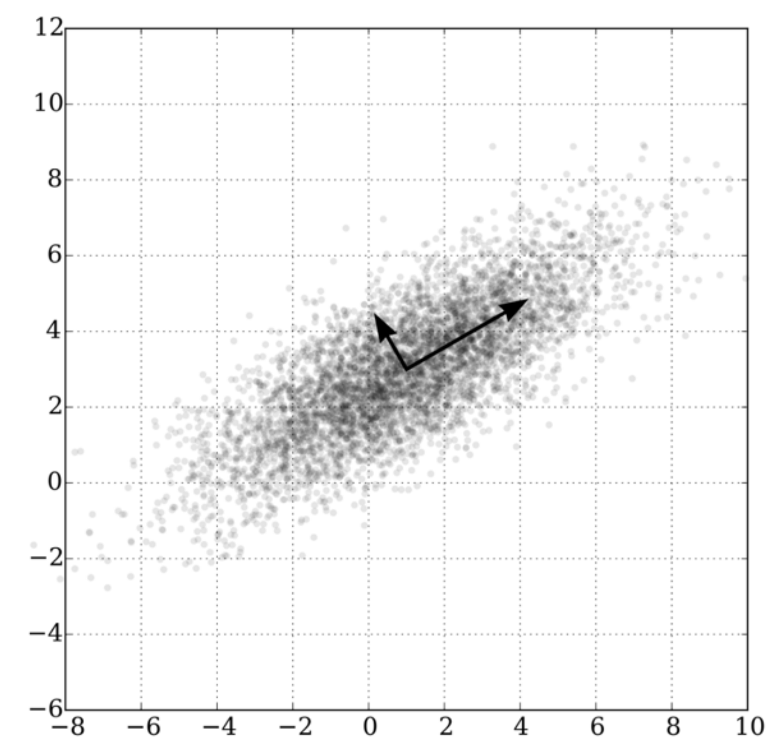
$$\boxed{\mathbf{A}\mathbf{u} = \lambda\mathbf{u} \qquad \mathbf{u}^T\mathbf{A}\mathbf{u} = \lambda}$$

Covariance matrices: $\mathbf{C} = \left\langle (\mathbf{X} - \langle\mathbf{X}\rangle)(\mathbf{X} - \langle\mathbf{X}\rangle)^T \right\rangle$

What's special about covariance matrices?

Principal components analysis (PCA)

- finds a low-dimensional representation of data which minimizes the sum of square distances with the original data points
- finds directions that explain the most variance in the data
- does this by using the eigenvectors of the covariance matrix of the data



Linear algebra review & principal components analysis (PCA)

Eigendecomposition: $\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{-1}$

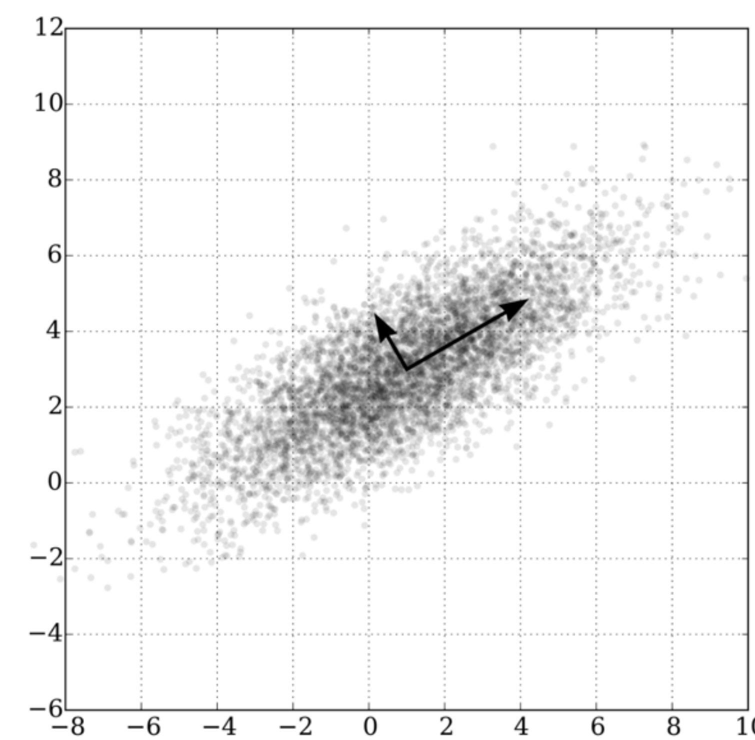
$\boxed{\mathbf{A}\mathbf{u} = \lambda\mathbf{u} \qquad \mathbf{u}^T\mathbf{A}\mathbf{u} = \lambda}$

Covariance matrices: $\mathbf{C} = \left\langle (\mathbf{X} - \langle\mathbf{X}\rangle)(\mathbf{X} - \langle\mathbf{X}\rangle)^T \right\rangle$

What's special about covariance matrices?

Principal components analysis (PCA)

- finds a low-dimensional representation of data which minimizes the sum of square distances with the original data points
- finds directions that explain the most variance in the data
- does this by using the eigenvectors of the covariance matrix of the data



Resource: "A tutorial on principal components analysis" by Lindsay I Smith

Non-linear synaptic plasticity

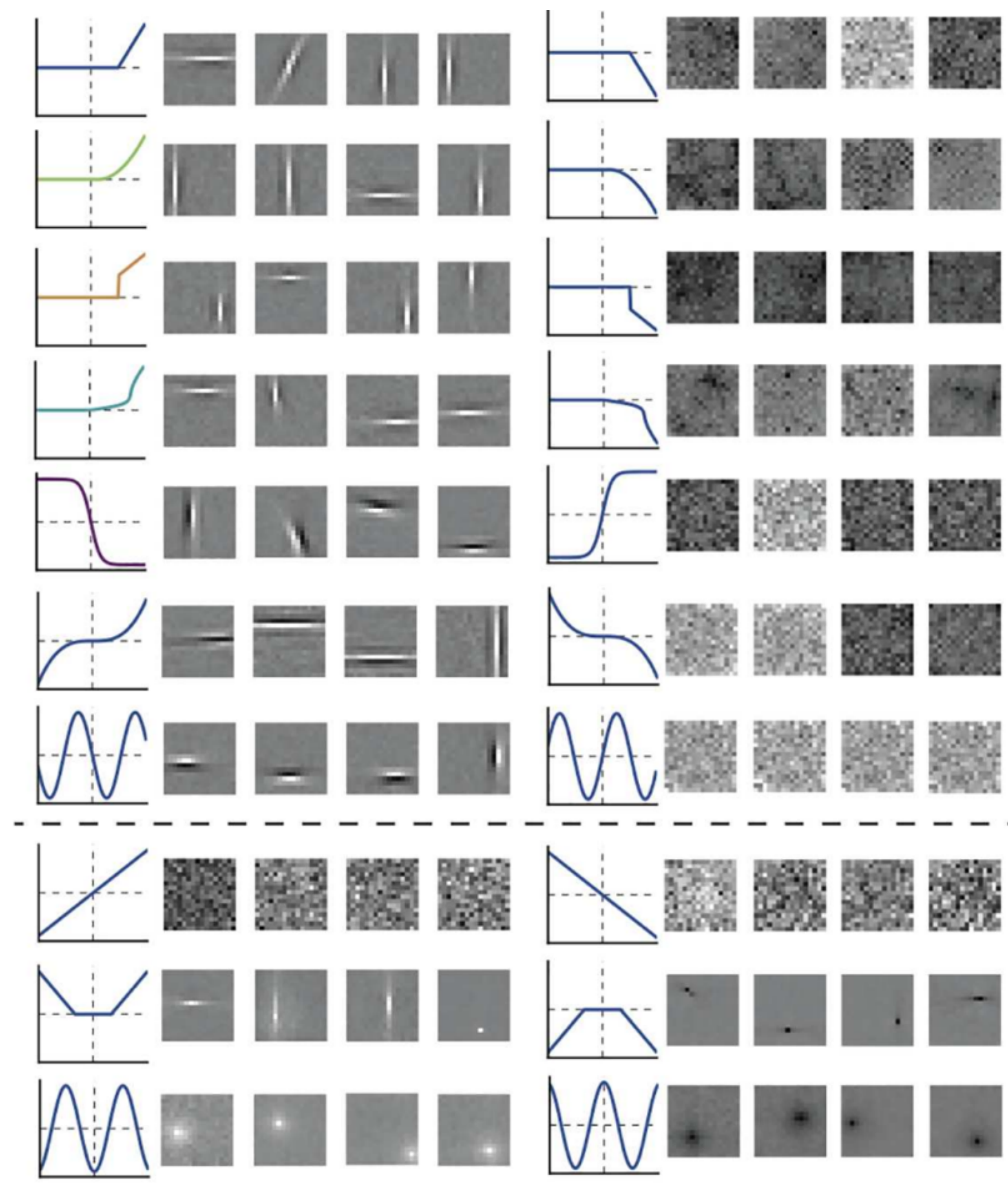
- Up to now we've only considered linear neurons, but real neurons have nonlinear responses to inputs — what happens when we add a nonlinearity?

$$y = g(\mathbf{w}^T \mathbf{x})$$

- We get independent components analysis (ICA)

Linear neuron + Hebb = PCA
Nonlinear neuron + Hebb = ICA
Linear neuron + nonlinear Hebb = ICA

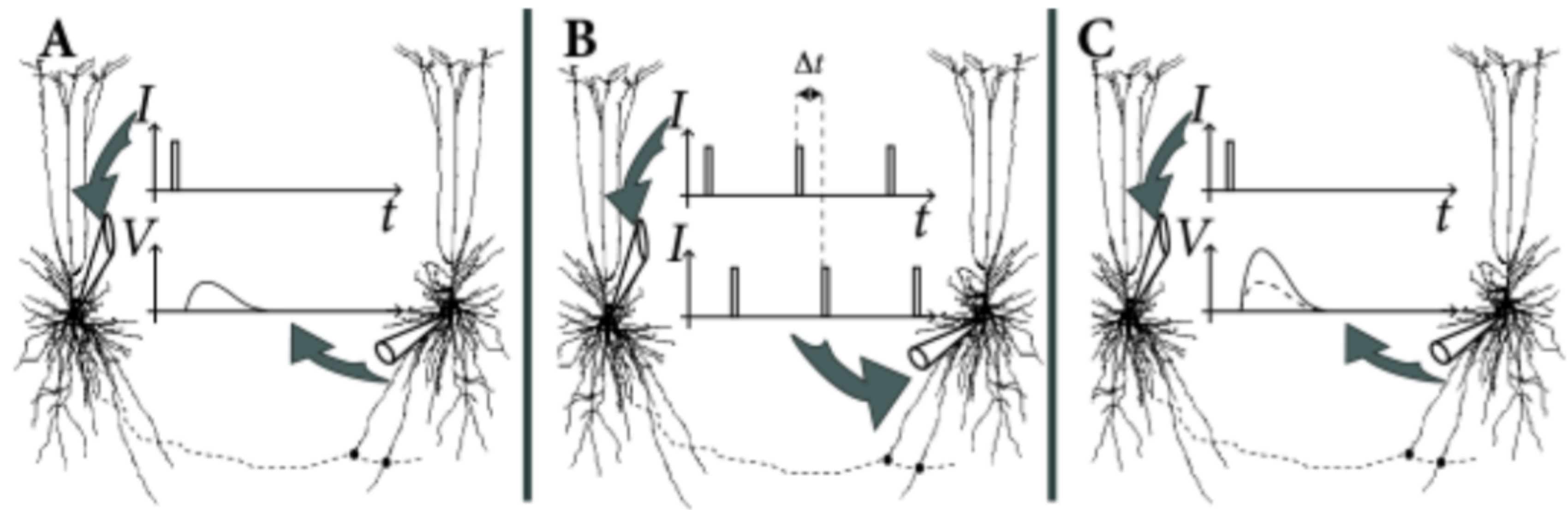
$$\Delta \mathbf{w} = \mathbf{x}\ h(\mathbf{w}^T \mathbf{x})$$



Hyvarinen & Oja 1998

Brito & Gerstner 2016

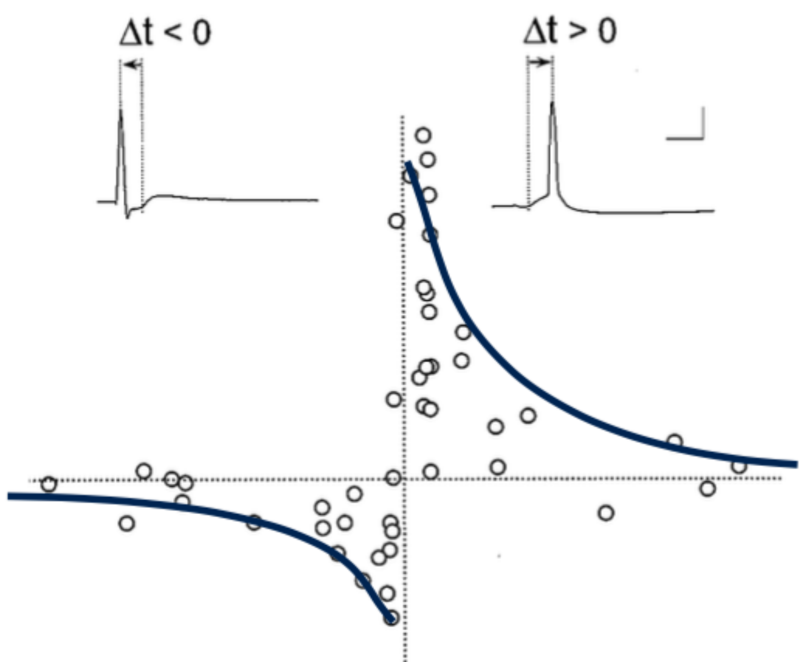
Spike-timing-dependent plasticity (STDP)



measure connection strength between two connected neurons

evoke spikes in both neurons at precise times

measure connection strength between the two neurons

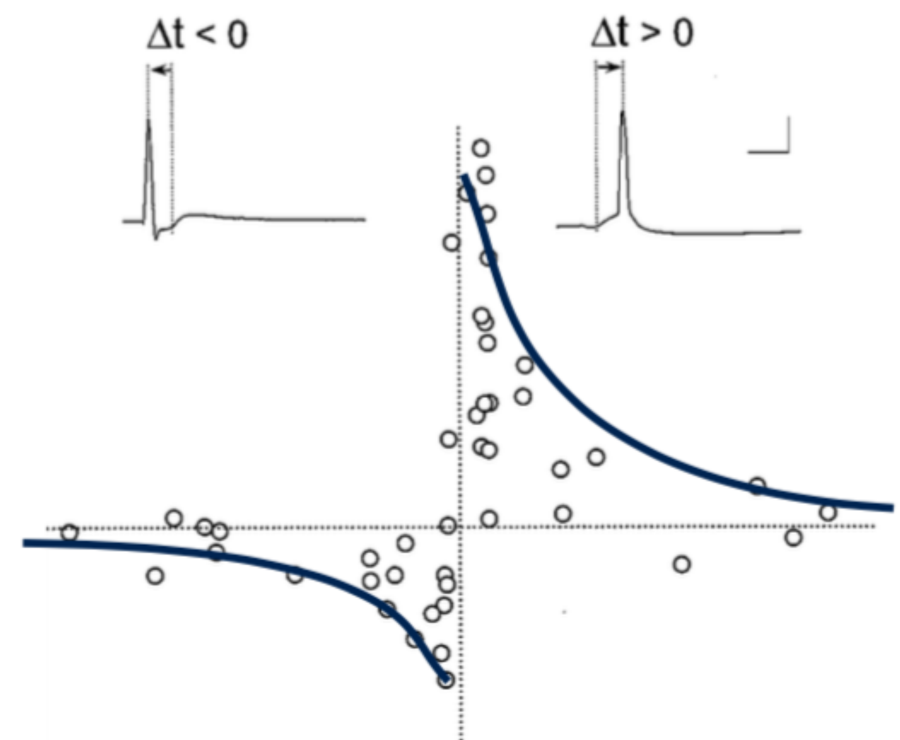


- if neuron i fired before neuron j -> synapse strengthened (LTP)
- if neuron i fired after neuron j -> synapse weakened (long-term depression: LTD)

images: Gerstner et al. 2014, Bi and Poo 1998

STDP models

$$\Delta w = \begin{cases} A_p \exp(\frac{|\Delta t|}{\tau_p}) & \text{if } \Delta t \geq 0 \\ -A_d \exp(\frac{|\Delta t|}{\tau_d}) & \text{if } \Delta t < 0 \end{cases}$$

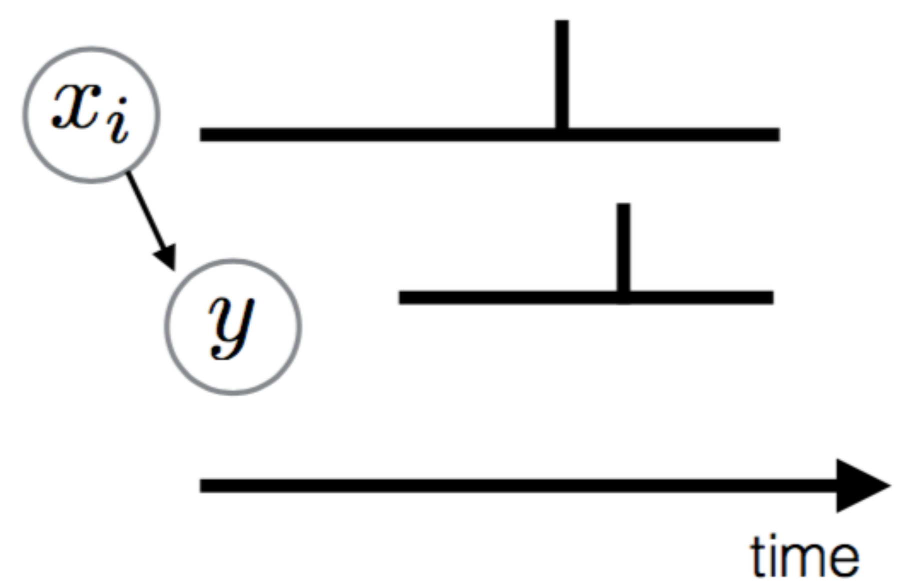


presynaptic activity $S_i(t) = \sum_f \delta(t - t_i^f)$

spike times

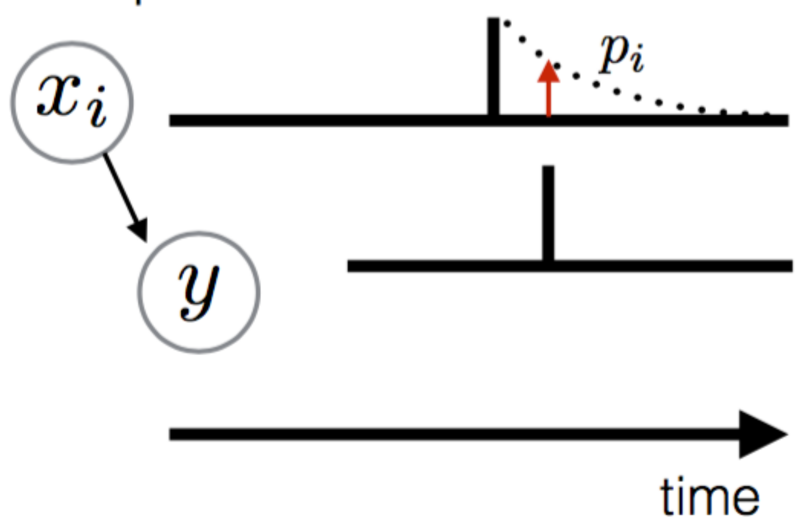
postsynaptic activity $S(t) = \sum_f \delta(t - t^f)$

learning rule $\frac{d}{dt}\vec{w} = \eta\phi[S_i(t'), S(t'), \vec{w}]$



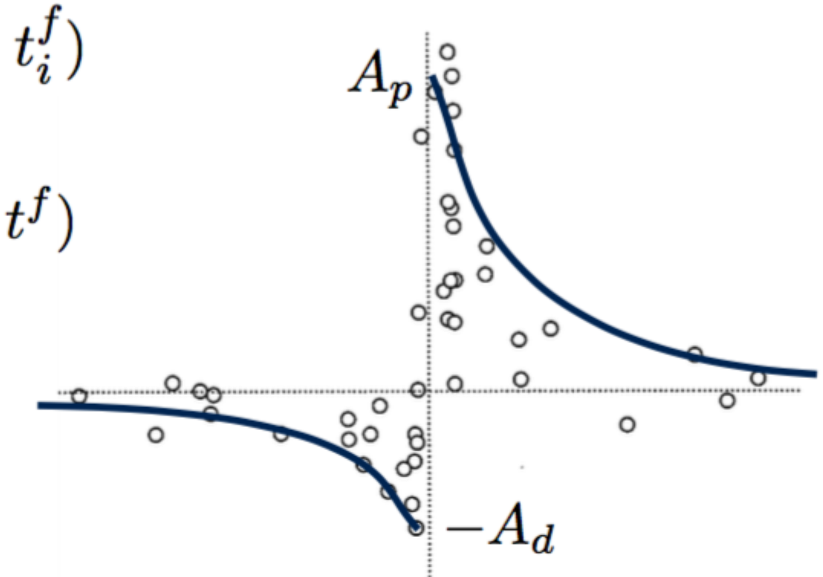
STDP implementation: traces

pre before post

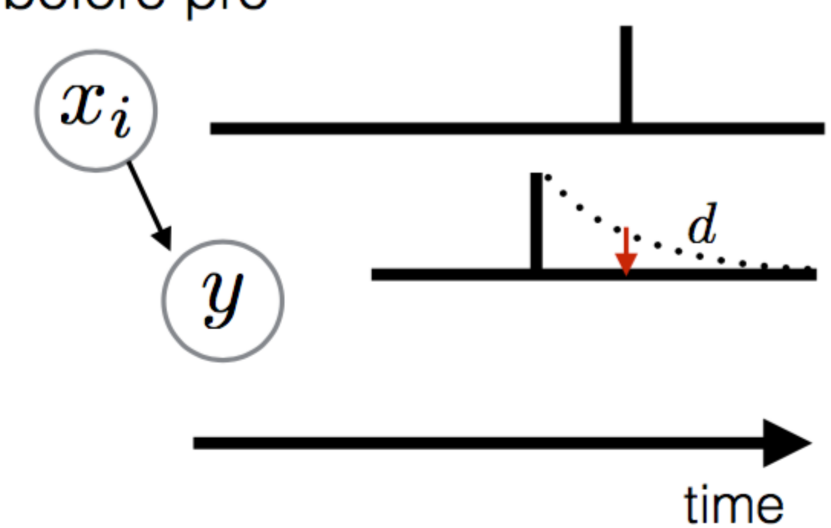


$$S_i(t) = \sum_f \delta(t - t_i^f)$$

$$S(t) = \sum_f \delta(t - t^f)$$



post before pre



$$\frac{d}{dt} w_i = \eta \left(A_p \, p_i(t) S(t) - A_d \, d(t) S_i(t) \right)$$

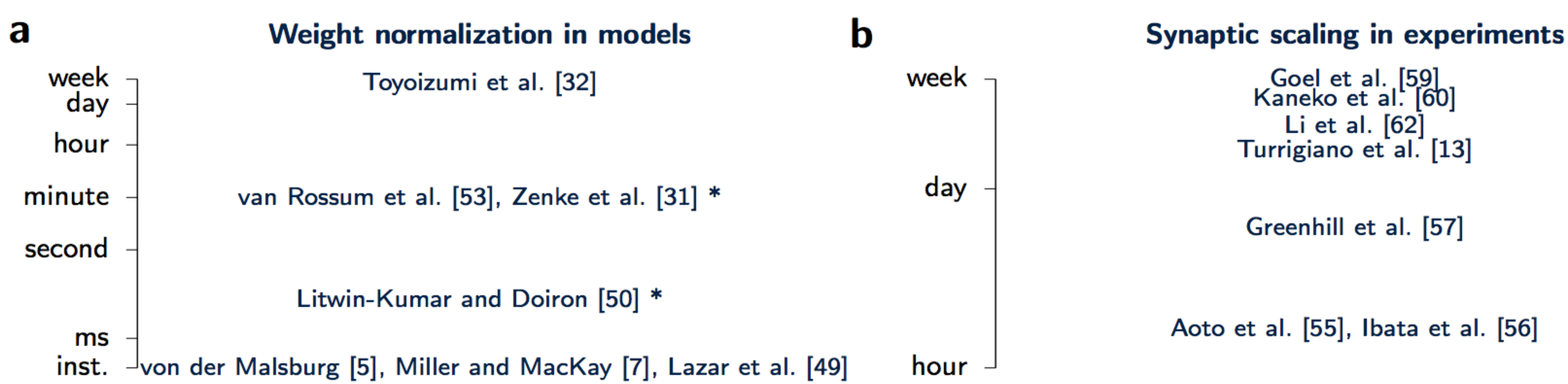
with

$$\tau_p \frac{d}{dt} p_i = -p_i + S_i(t) \quad \text{presynaptic trace}$$

$$\tau_d \frac{d}{dt} d = -d + S(t) \quad \text{postsynaptic trace}$$

Stability and homeostasis

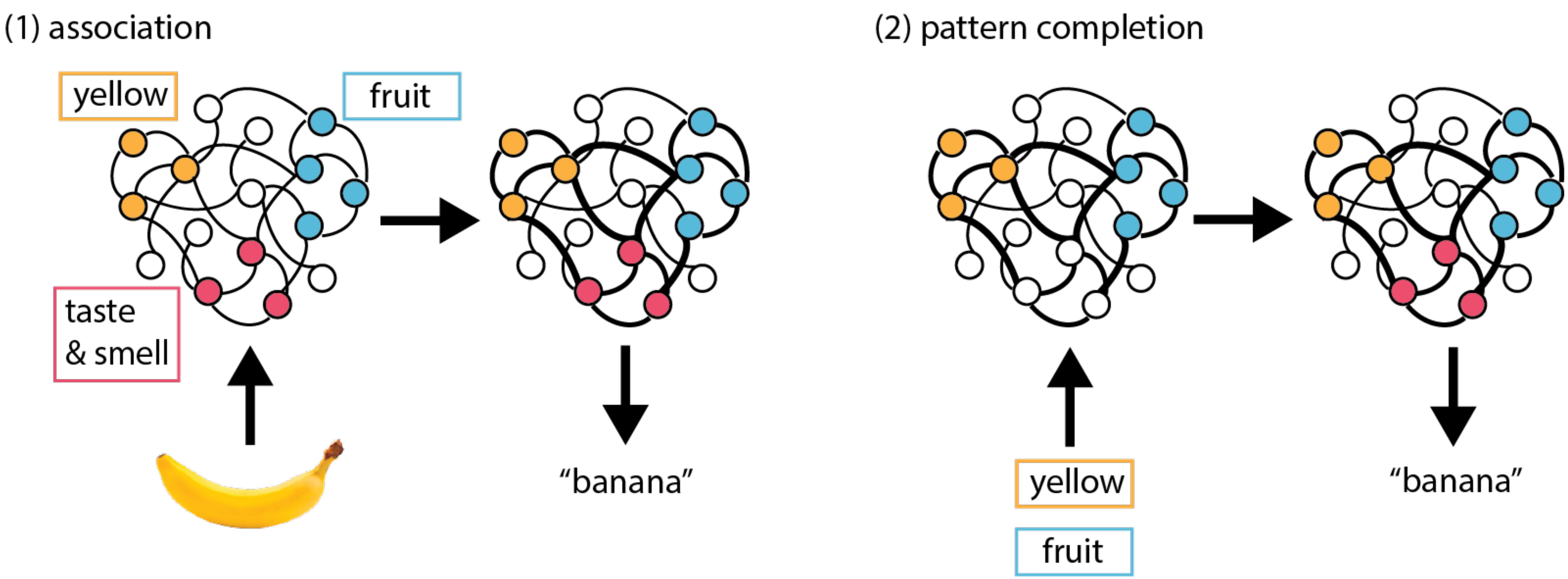
- Stability is an integral part of all synaptic plasticity, but may come in two flavors:
  - Learning rules may have intrinsic stability mechanisms, such as the BCM or Oja learning rules
  - Additional mechanisms may be in place to ensure stability
- **Homeostatic plasticity:** neuronal change that tends to return a neuron back to an initial set point
  - E.g., synaptic scaling *(Turrigiano et al. 1998)* & *i*nhibitory plasticity *(Vogels, Sprekeler et al. 2011)*
- Important point: if homeostasis is responsible for stabilizing neuronal dynamics, then it must be fast enough to counteract Hebbian plasticity *(Zenke et al. 2013,2017)*



Outline

1. **Introduction to learning and memory**
    - The biology of memory and synaptic plasticity
2. **Models of synaptic plasticity**
    - Hebbian learning and stability
    - Rate-based plasticity: BCM and Oja’s rule
    - Spike-based plasticity: STDP
3. **Models of associative memory**
    - The Hopfield network
    - Estimating memory capacity
    - Incremental learning
4. **Summary and open questions**

Associative memory



Desired qualities of a memory system:

(1) States that can persist over time

(2) Sufficient capacity or number of states

(3) Different inputs should lead to different memory states

(4) Robustness to noise

Tradeoffs:

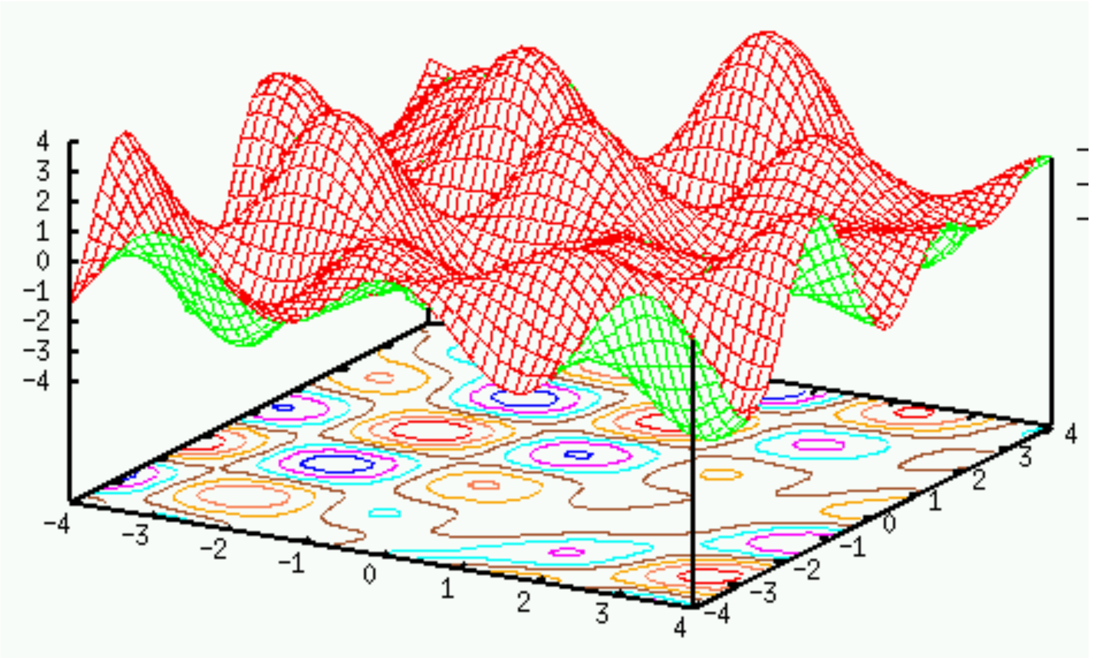
- Robustness versus capacity
- Stability-plasticity dilemma

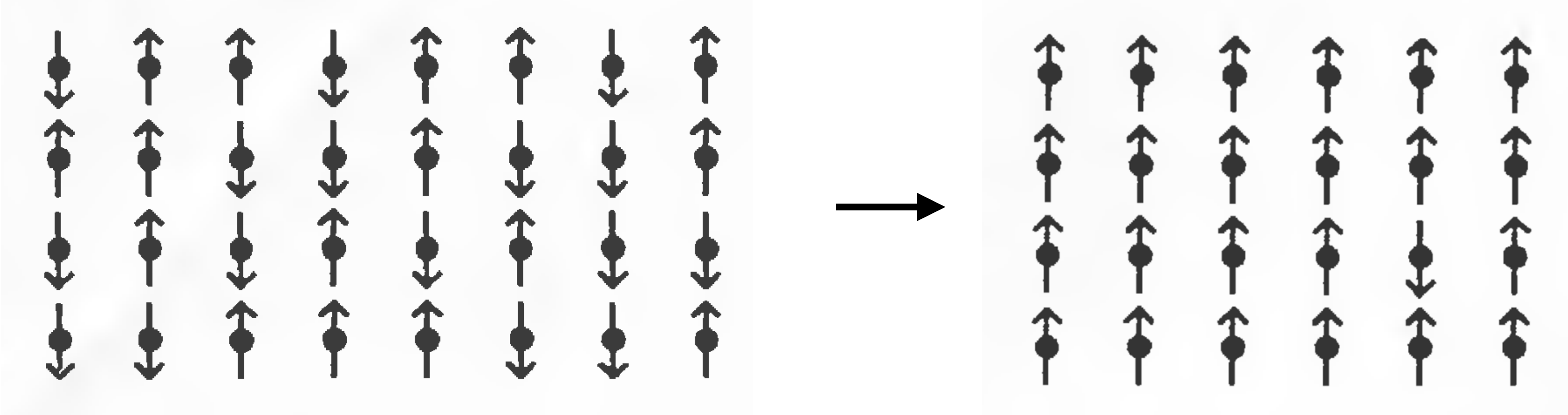
Chaudhuri & Fiete 2016

Hopfield networks

Hopfield 1982

- Analogy between recurrent neural networks and physical systems with collective behavior
- Each neuron is influenced by a local field, which affects its state
- We can think of memories as being low-energy valleys of the network

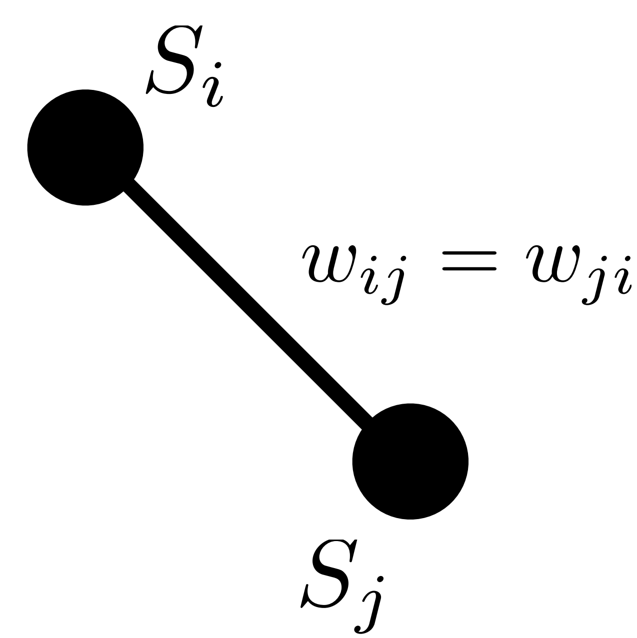




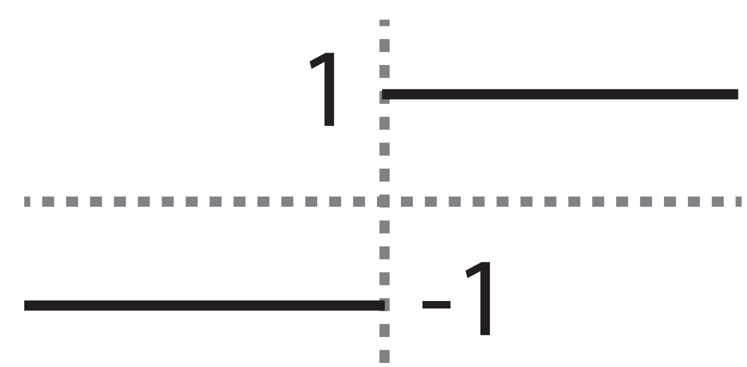
Hertz et al. 1991

Description of network

- Recurrent network model with N units, denoted $S_i \in \{-1, 1\}$
- Stores p patterns, denoted $\xi_i^\mu \in \{-1, 1\}$
- Dynamics: $S_i = \mathrm{sgn}(h_i) = \mathrm{sgn}\Big(\sum_j w_{ij} S_j\Big)$
- Energy function: $H = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij} S_i S_j$



Problem set question 2.1



Hebbian learning and memory capacity

- Hopfield networks can store memory patterns as attractor states with a local Hebbian rule that only depends on pre and post-synaptic activity.

$$w_{ij} = \frac{1}{N} \sum_{i=1}^{\mu} \xi_i^{\mu} \xi_j^{\mu}$$

- Problem set exercise 2.2 — simulate pattern completion in a simple Hopfield network with four memory patterns

- Memory capacity: how many memory patterns can be stored in the network such that each one is stable?

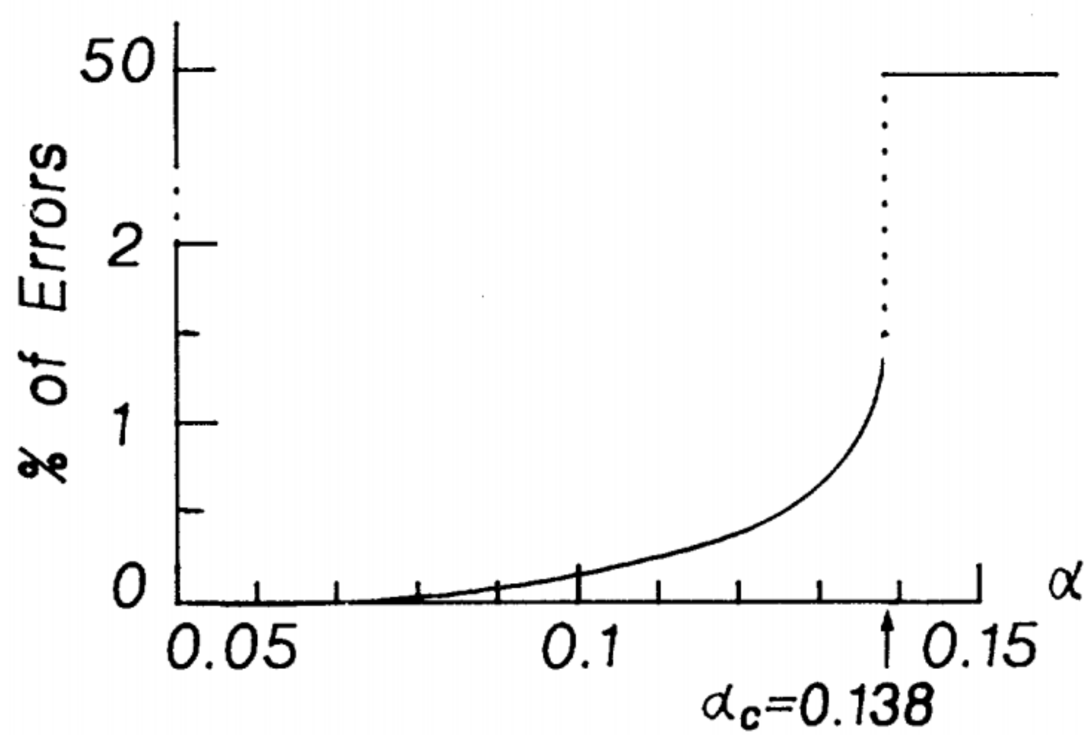
    - Consider the stability of a single memory pattern

    - Problem set exercises 2.3 and 2.4 — estimate the memory capacity of the Hopfield network, and confirm with numerical simulations

Memory capacity

- Capacity can be estimated using much more accurate mean-field methods from statistical physics

$$\longrightarrow \quad \alpha_c = p/N \approx 0.138$$



(Amit, Gutfreund & Sompolinsky 1985)

Extensions and adding biological realism

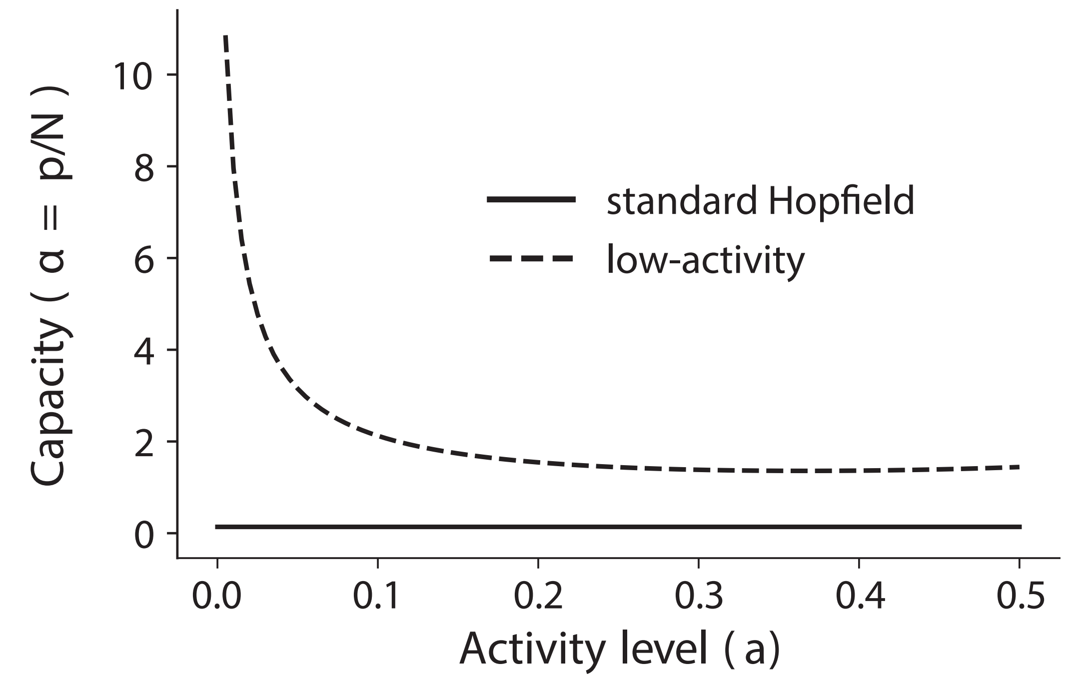
- Correlated patterns and the pseudoinverse rule *(Kanter & Sompolinsky 1987)*

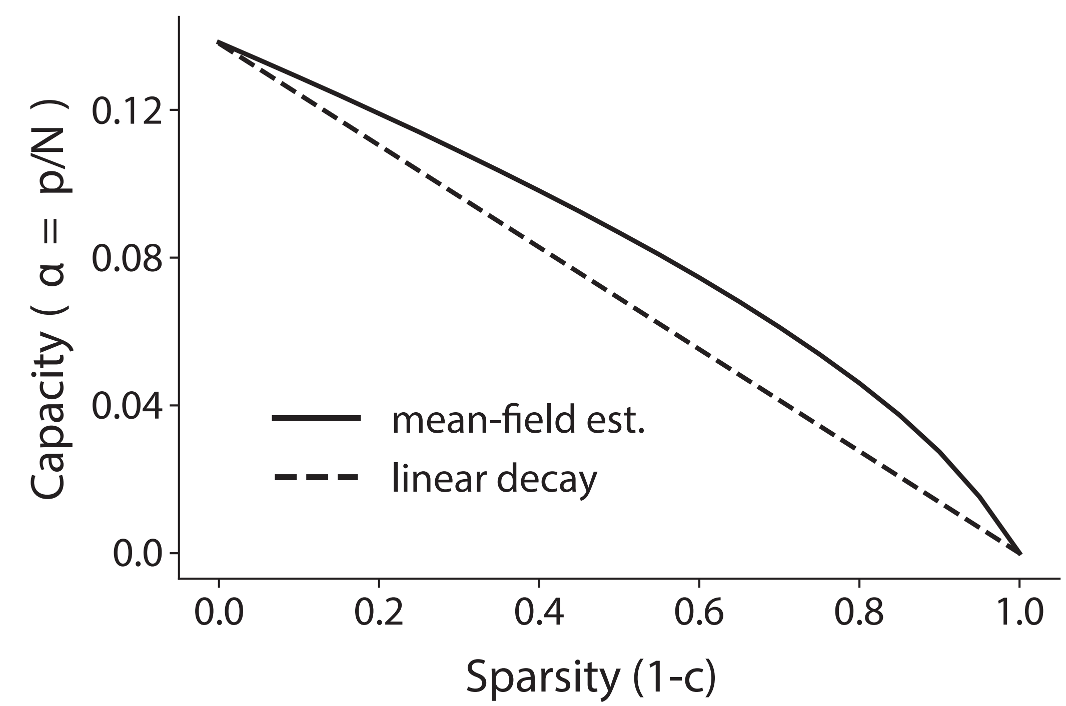
- Low-activity patterns: improves memory capacity *(Tsodyks & Feigelman 1988)*

- Sparse connectivity: reduces capacity linearly with sparsity *(Sompolinsky 1987)*

- Overall idea: increasing sparsity to reduce interference

- Alternate learning rules can work better, e.g., perceptron learning rule gives a capacity of 2N *(Gardner 1988)*





Shortcomings and incremental learning

- Issues with the Hopfield network
  - Unrealistic neurons and dynamics
  - Batch learning
  - Unbounded synaptic weights
- Incremental learning
  - The brain learns new memories over time, but how can a network incorporate new patterns without disrupting previous ones? (catastrophic forgetting)
  - We can modify the Hebbian learning rule such that it incrementally incorporates memories over time and keeps the weights bounded

$$w_{ij}^{\text{new}} = w_{ij}^{\text{old}} + \eta \xi_i^\mu \xi_j^\mu \qquad\qquad w_{ij} \in [-A, A]$$

  - Problem set exercise 2.5 — simulate a network which adds new patterns over time. What happens to the capacity?

Outline

1. **Introduction to learning and memory**
    - The biology of memory and synaptic plasticity
2. **Models of synaptic plasticity**
    - Hebbian learning and stability
    - Rate-based plasticity: BCM and Oja's rule
    - Spike-based plasticity: STDP
3. **Models of associative memory**
    - The Hopfield network
    - Estimating memory capacity
    - Incremental learning
4. **Summary and open questions**

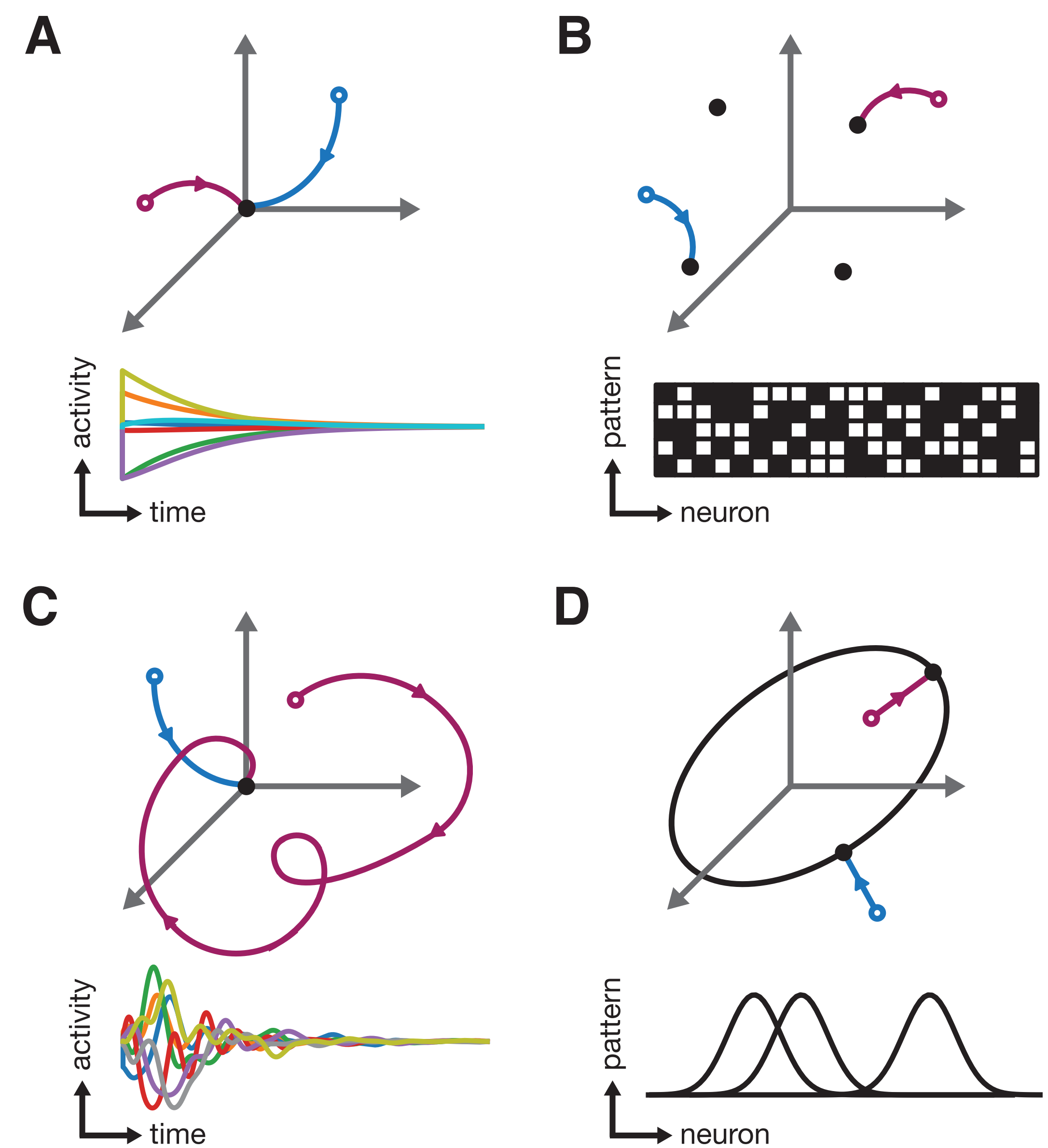
Summary

- Methods and techniques
  - Taylor expansion
  - Fixed points and stability analysis
  - Linear algebra and PCA
  - Energy functions
  - Signal-to-noise analysis
- Synaptic plasticity, Hebbian learning and stability
- Associative memory and Hopfield networks

Open questions

Memory representation:

- What is the state-variable for memory in the brain? (synaptic plasticity or something else?)
- Biologically, what does synaptic weight represent? (e.g., binomial release model)
- What does memory recall look like? e.g., discrete attractors, transients, etc.

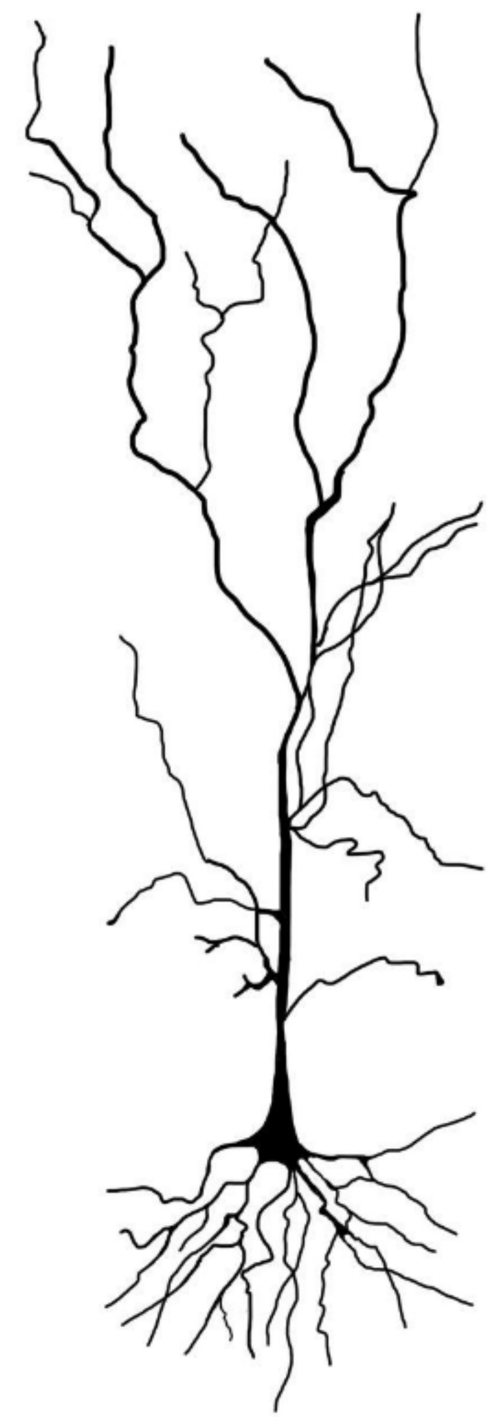


adapted from Chaudhuri & Fiete 2016

Open questions

Synaptic plasticity:

- What learning rules exist in the brain?
- Is STDP a good model for plasticity?
- Do different brain areas and cell types have distinct learning rules?
- How do we design good experiments to find plasticity rules?



ARTICLES

nature neuroscience

Inferring learning rules from distributions of firing rates in cortical neurons

Sukbin Lim[1], Jillian L McKee[1], Luke Woloszyn[2], Yali Amit[3,4], David J Freedman[1], David L Sheinberg[5] & Nicolas Brunel[1,3]

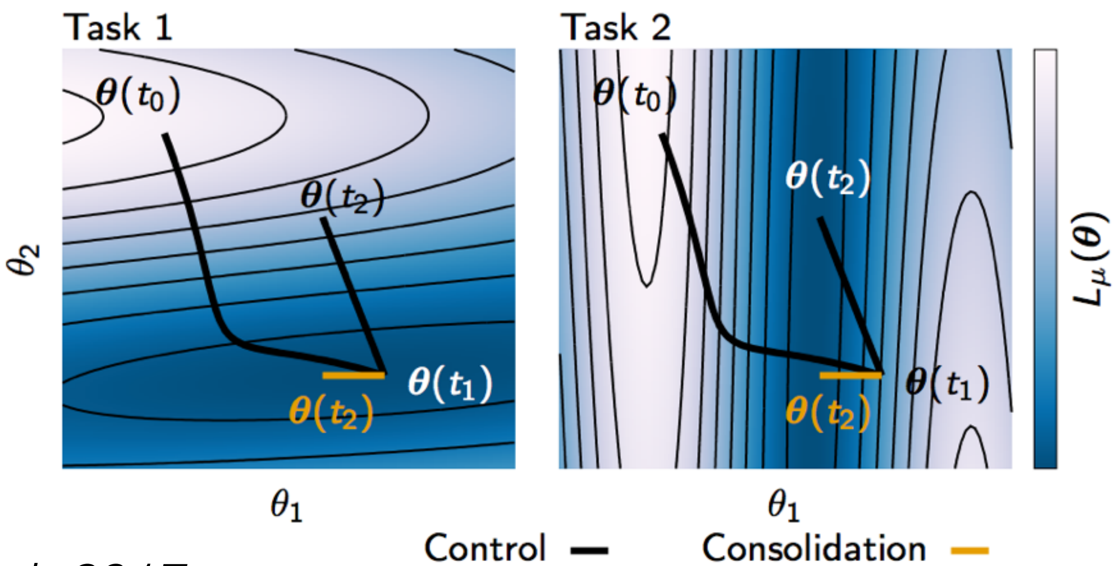
Open questions

The stability-plasticity dilemma:

- How is plasticity able to learn quickly while also remaining stable?
- Is plasticity always active, or is it gated and controlled by neuromodulation and brain state?

Continual and lifetime learning:

- How is the brain able to continuously update and store new memories throughout life without interfering with previous memories? (experience replay and complex synapses)



Zenke et al. 2017

Project ideas