Lecture 6: Hebbian models of plasticity

Jochen Braun

Otto-von-Guericke-Universität Magdeburg,
Cognitive Biology Group

Theoretical Neuroscience II, SS 2020

6. Hebbian plasticity

Activity-driven plasticity, coupled dynamics of activity (faster) and plasticity (slower). **Unsupervised**: *coupled dynamics unconstrained.* **Supervised**: *constrained by externally prescribed activity.* **Reinforcement**: *constrained by external teacher.* **Hebbian** *learning compares pre- and postsynaptic activity ('fire together, wire together'). Success strengthens, failure weakens connection.* **Formal analysis:** *weight* $\Delta \boldsymbol{w}$ *grows with average product* $\langle v \, \boldsymbol{u} \rangle$ *of pre- and postsynaptic activity, or activity compared to threshold* $\langle (v - \theta_v) \boldsymbol{u} \rangle$, *or* $\langle v(\boldsymbol{u} - \boldsymbol{\theta}_u) \rangle$. *Dynamics governed by statistical structure of input patterns (but unstable, without steady-state), measured by correlations (absolute values), covariances (relative values), coefficients (normalized values), and eigenvectors of corresponding matrices.* **Linear analysis:** *Solve for* $\mathbf{w}(\mathbf{t})$ *in terms of eigenvectors of covariance* $\boldsymbol{C}$. *Weights grow fastest along dominant eigenvector.*

Organization of lecture

- 1 Activity-driven plasticity
- 2 Hebbian learning
- 3 Formal analysis
- 4 Correlations and covariances
- 5 Linear analysis of covariance rule
- 6 Learning input covariances

Products of vectors and matrices

Quantity	Definition	Aliases
dot product	$\mathbf{c} \cdot \mathbf{d} = \sum_k c_k d_k$	$\mathbf{c}^T \mathbf{d}$
norm	$\lvert\mathbf{c}\rvert^2 = \mathbf{c} \cdot \mathbf{c} = \sum_k c_k^2$	$\lVert\mathbf{c}\rVert^2$
matrix-vector product	$[\mathbf{K} \cdot \mathbf{c}]_k = \sum_l \mathbf{K}_{kl} c_l$	$\mathbf{K}\mathbf{c}$
vector-matrix product	$[\mathbf{c} \cdot \mathbf{K}]_l = \sum_k c_k \mathbf{K}_{kl}$	$\mathbf{c}^T \mathbf{K}$
quadratic form	$\mathbf{c} \cdot \mathbf{K} \cdot \mathbf{d} = \sum_{kl} c_k \mathbf{K}_{kl} d_l$	$\mathbf{c}^T \mathbf{K} \mathbf{d}$
outer product	$[\mathbf{c}\mathbf{d}]_{kl} = c_k d_l$	$\mathbf{c}\mathbf{d}^T$
matrix-matrix product	$[\mathbf{K} \cdot \mathbf{L}]_{kl} = \sum_m \mathbf{K}_{km} \mathbf{L}_{ml}$	$\mathbf{K}\mathbf{L}$

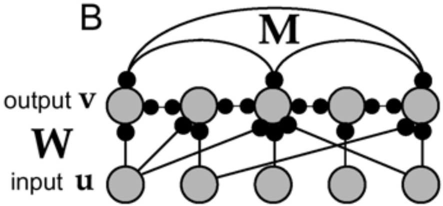
Note: We omit transpose signs unless it would be confusing to do so.

1. Activity-driven plasticity

Consider a two-layer network with inputs $\boldsymbol{u}$ driving recurrently connected outputs $\boldsymbol{v}$. The activity dynamics is governed by

$$\tau_a \frac{d\boldsymbol{v}}{dt} = -\boldsymbol{v} + F\left(\boldsymbol{W} \cdot \boldsymbol{u} + \boldsymbol{M} \cdot \boldsymbol{v}\right)$$

where $\boldsymbol{W}$ and $\boldsymbol{M}$ are the feedforward and recurrent synaptic weight matrices, respectively.



‘Plastic’ connectivity

In general, we speak of “plasticity” when synaptic weights change over time, and of “activity-driven plasticity” when this change depends on network activity.

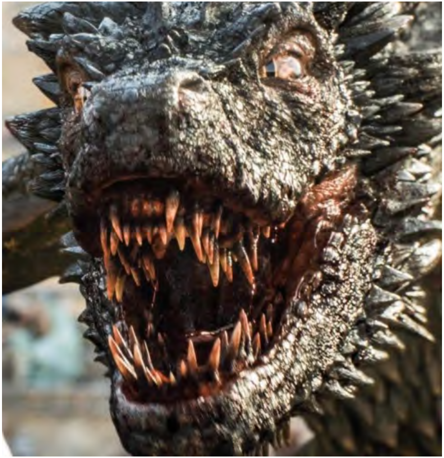
‘Plastic’ feedforward connectivity $\boldsymbol{W}(t)$:

$$\tau_w \frac{dw_{ij}}{dt} = -w_{ij} + f(w_{ij}, \boldsymbol{u}, \boldsymbol{v})$$

The dynamics of activity $\boldsymbol{v}(t)$ depends on feedforward connectivity $\boldsymbol{W}(t)$ while, in return, plasticity of feedforward connectivity $\boldsymbol{W}(t)$ depends on activity $\boldsymbol{v}(t)$.

This co-evolution of two mutually dependent variables is called a **coupled dynamics**.

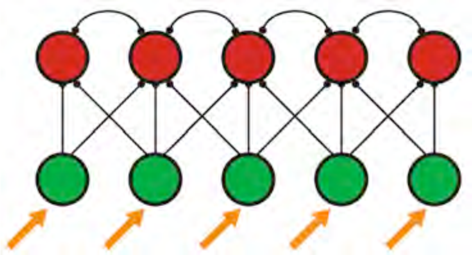
Understanding, controlling, and steering this coupled dynamics of activity and plastic connectivity has been likened to "taming the beast" (Abbott, 2000).



Activity ⟷ Plasticity

1.1 Unsupervised learning

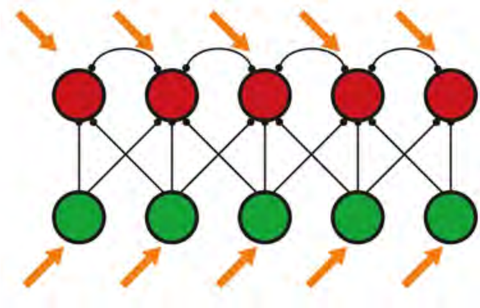
The activity driving the plasticity – $\mathbf{u}(t)$, $\mathbf{v}(t)$ – is constrained only by the input activity $\mathbf{v}(t)$. This situation is termed "unsupervised learning":

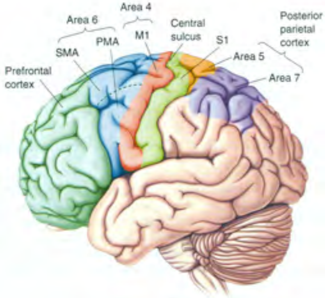


This is the most difficult situation, in terms of controlling plasticity.

1.2 Supervised learning

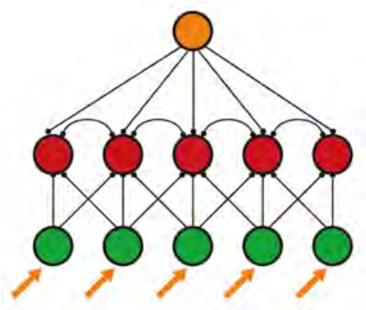
The activity driving the plasticity – $\boldsymbol{u}(t)$, $\boldsymbol{v}(t)$ – is constrained at both input and output levels. This is called “supervised learning”. Feed-forward connections between parietal cortex (sensory representations) and premotor cortex (motor representations) are an example.

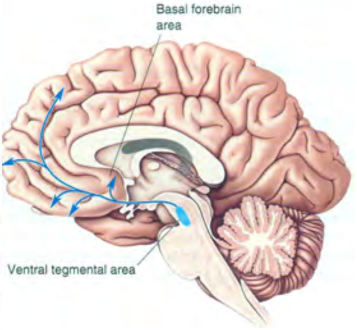




1.3 Reinforcement learning

The activity driving the plasticity – $\boldsymbol{u}(t)$, $\boldsymbol{v}(t)$ – is constrained at the input level an, in addition, by a general "reinforcement" or "teacher" signal to the output level. This may indicate, for example, the success or failure of a classification. Dopamine neurons in the ventral tegmental area are thought to provide such a "reinforcement" signal to neocortex.





Points to note

- Connectivity changes activity, activity changes (plastic) connectivity.
- The coupled dynamics is a theoretical challenge!
- The most difficult case is *unsupervised learning*, where activity and plastic connectivity determine each other.
- An easier case is *supervised learning*, where activity is prescribed by input (not by plastic connectivity).
- An intermediate case is *reinforcement learning*, which uses a central teacher signal.

2 Hebbian Learning

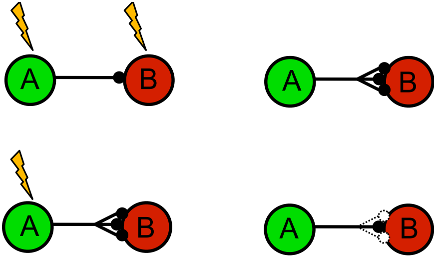
Working at Yerkes Primate Lab, Donald Hebb cortex proposed in "The organization of behavior: a neuropsychological theory" (1949) a fundamental principle for the interaction between neuronal activity and synaptic plasticity:

"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 fining B, is increased."

or paraphrased

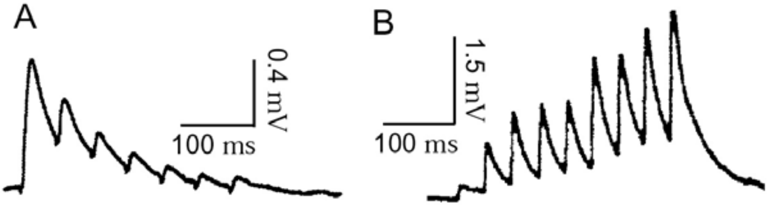
"Cells that fire together wire together"

- The Hebbian rule is *local* in that it compares pre- and post-synaptic firing.
- The success of A in triggering B strengthens the connection.
- The failure of A in triggering B weakens the connection.
- The Hebbian rule is intrinsically dichotomous, leading to either maximally strong or completely broken connections.



2.1 Non-Hebbian plasticity

Not all plasticity depends on both pre- and post-synaptic activity. For example, in short-term-depression or -potentiation (STD, STP), the pre-synaptic activity alone modifies the synaptic weight. In such cases one speaks of “non-Hebbian plasticity”



STD and STP, from Dayan & Abbott (2001)

Points to note

- There are two kinds of activity-dependent plasticity.
- "Hebbian" plasticity depends exclusively on directly pre- and post-synaptic activity.
- "Non-Hebbian" plasticity includes all other possibilities, e.g., plasticity which depends only on directly pre-synaptic activity.

3. Formal analysis

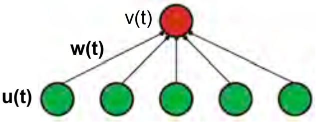
We analyze Hebbian plasticity formally in a simplified feedforward network:

- Output at steady-state and linear activation function:

$$\tau_a \frac{dv}{dt} \equiv 0, \quad F(\boldsymbol{w} \cdot \boldsymbol{u}) \equiv \boldsymbol{w} \cdot \boldsymbol{u} \qquad \Rightarrow \qquad v = \boldsymbol{w} \cdot \boldsymbol{u}$$

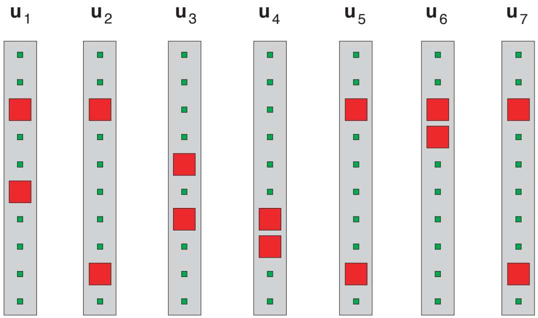
- Plasticity driven by time-average $\langle\rangle$ of pre- and post-synaptic activities

$$\tau_w \frac{d\boldsymbol{w}}{dt} = f\left(\langle v \rangle, \langle \boldsymbol{u} \rangle\right)$$



Input activities

We consider an ensemble of input patterns, from which we randomly draw samples ...



Here, red squares represent positive, green squares negative values. Average value is aways zero.

3.1 Correlation rule

A simple implementation of Hebb's rule is to make weights $\boldsymbol{w}$ grow in proportion to the average product $v\,\boldsymbol{u}$:

$$\tau_w \frac{d\boldsymbol{w}}{dt} = \langle v\,\boldsymbol{u} \rangle, \qquad v = \boldsymbol{w} \cdot \boldsymbol{u}$$

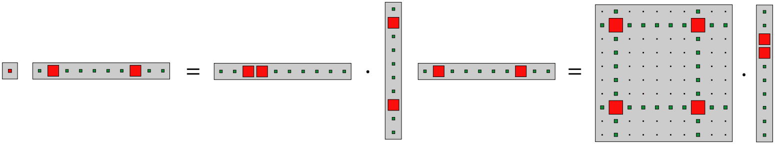
$$\tau_w \frac{dw_k}{dt} = \langle v\, w_k \rangle, \qquad v = \sum_k w_k \cdot u_k$$

In other words, the weight w_k of a synapse $u_k \rightarrow v$ grows comparatively rapidly, if both u_k *and* v are large, but comparatively slowly when *either* u_k *or* v are small.

Correlation formulation

This can be reformulated in terms of the correlation matrix $\boldsymbol{Q}$ of inputs $\boldsymbol{u}$:

$$\tau_w \frac{d\boldsymbol{w}}{dt} = \langle v\,\boldsymbol{u} \rangle = \langle \boldsymbol{w} \cdot \boldsymbol{u}\,\boldsymbol{u} \rangle = \langle \boldsymbol{u}\,\boldsymbol{u} \cdot \boldsymbol{w} \rangle = \langle \boldsymbol{Q} \rangle \cdot \boldsymbol{w}$$



where $\boldsymbol{Q} \equiv \langle \boldsymbol{u}\,\boldsymbol{u} \rangle$ is correlation matrix and $\langle\rangle$ is taken only over input activities. The dimensions of vectors, matrices, and their products are illustrated graphically.

2D example

$$\langle v \rangle = \langle \boldsymbol{w} \cdot \boldsymbol{u} \rangle = \boldsymbol{w} \cdot \langle \boldsymbol{u} \rangle = \begin{pmatrix} w_1 & w_2 \end{pmatrix} \cdot \begin{pmatrix} \langle u_1 \rangle \\ \langle u_2 \rangle \end{pmatrix} = w_1 \langle u_1 \rangle + w_2 \langle u_2 \rangle$$

$$\begin{aligned} \langle v \boldsymbol{u} \rangle &= \langle \boldsymbol{w} \cdot \boldsymbol{u}\, \boldsymbol{u} \rangle = \boldsymbol{w} \cdot \langle u\, u \rangle = \begin{pmatrix} w_1 & w_2 \end{pmatrix} \cdot \langle \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \begin{pmatrix} u_1 & u_2 \end{pmatrix} \rangle = \\ &= \begin{pmatrix} w_1 & w_2 \end{pmatrix} \cdot \begin{pmatrix} \langle u_1^2 \rangle & \langle u_1 u_2 \rangle \\ \langle u_2 u_1 \rangle & \langle u_2^2 \rangle \end{pmatrix} = \begin{pmatrix} w_1 \langle u_1^2 \rangle + w_2 \langle u_1 u_2 \rangle \\ w_1 \langle u_2 u_1 \rangle + w_2 \langle u_2^2 \rangle \end{pmatrix} \end{aligned}$$

$$\begin{aligned} \langle v \boldsymbol{u} \rangle &= \langle \boldsymbol{u}\, \boldsymbol{u} \rangle \cdot \boldsymbol{w} = \langle \boldsymbol{Q} \rangle \cdot \boldsymbol{w} = \\ &= \begin{pmatrix} \langle u_1^2 \rangle & \langle u_1 u_2 \rangle \\ \langle u_2 u_1 \rangle & \langle u_2^2 \rangle \end{pmatrix} \cdot \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \cdot \begin{pmatrix} w_1 \langle u_1^2 \rangle + w_2 \langle u_1 u_2 \rangle \\ w_1 \langle u_2 u_1 \rangle + w_2 \langle u_2^2 \rangle \end{pmatrix} \end{aligned}$$

This example illustrates the derivation in the previous slide.

No steady-state

How will weights develop over time? Will they reach a steady-state? Consider the norm of weights

$$|\mathbf{w}|^2 \equiv \mathbf{w} \cdot \mathbf{w}$$

It's time derivative is

$$\tau_w \frac{d|\mathbf{w}|^2}{dt} = \tau_w \frac{d(\mathbf{w} \cdot \mathbf{w})}{dt} = 2\,\mathbf{w} \cdot \tau_w \frac{d\mathbf{w}}{dt} =$$

$$= 2\,\mathbf{w} \cdot \langle v\mathbf{u} \rangle = 2\langle \mathbf{w} \cdot v\mathbf{u} \rangle = 2\langle v^2 \rangle > 0$$

and therefore positive (as long as the activity $v \neq 0$). Thus, there is no steady-state and weights grow without bound.

3.2 Covariance rule

To ensure that some weights decrease while others increase, we can introduce a threshold, say, on the presynaptic side. A convenient choice is to use the average presynaptic activity

$$\tau_w \frac{d\boldsymbol{w}}{dt} = \langle v \, (\boldsymbol{u} - \boldsymbol{\theta_u}) \rangle, \qquad \boldsymbol{\theta_u} = \langle \boldsymbol{u} \rangle, \qquad v = \boldsymbol{w} \cdot \boldsymbol{u}$$

This can be reformulated

$$\tau_w \frac{d\boldsymbol{w}}{dt} = \langle \boldsymbol{w} \cdot \boldsymbol{u} \, (\boldsymbol{u} - \boldsymbol{\theta_u}) \rangle = \boldsymbol{w} \langle \boldsymbol{u} \, \boldsymbol{u} \rangle - \boldsymbol{w} \langle \boldsymbol{u} \rangle \langle \boldsymbol{u} \rangle = \boldsymbol{C} \cdot \boldsymbol{w}$$

where $\boldsymbol{C} \equiv \langle \boldsymbol{u} \, \boldsymbol{u} \rangle - \langle \boldsymbol{u} \rangle \langle \boldsymbol{u} \rangle$ is the covariance matrix.

Alternative motivation

Alternatively, we can introduce a threshold on the post-synaptic side. Surprisingly, the resulting average rule is the same

$$\tau_w \frac{d\boldsymbol{w}}{dt} = \langle (v - \theta_v)\, \boldsymbol{u} \rangle, \qquad \theta_v = \langle v \rangle, \qquad v = \boldsymbol{w} \cdot \boldsymbol{u}$$

Reformulate

$$\tau_w \frac{d\boldsymbol{w}}{dt} = \langle (\boldsymbol{w} \cdot \boldsymbol{u} - \theta_v)\, \boldsymbol{u} \rangle = \boldsymbol{w} \langle \boldsymbol{u}\, \boldsymbol{u} \rangle - \boldsymbol{w} \langle \boldsymbol{u} \rangle \langle \boldsymbol{u} \rangle = \boldsymbol{C} \cdot \boldsymbol{w}$$

where $\boldsymbol{C} \equiv \langle \boldsymbol{u}\, \boldsymbol{u} \rangle - \langle \boldsymbol{u} \rangle \langle \boldsymbol{u} \rangle$ is once again the covariance matrix.

No steady-state

Although the covariance rule ensures that weights increase and decrease, it still does not provide for stability. In fact, now the weights grow without bound in both directions. Consider the norm (sum of squares) of the weights

$$|\boldsymbol{w}|^2 \equiv \boldsymbol{w} \cdot \boldsymbol{w}$$

It's time derivative is

$$\tau_w \frac{d|\boldsymbol{w}|^2}{dt} = \tau_w \frac{d(\boldsymbol{w} \cdot \boldsymbol{w})}{dt} = 2\, \boldsymbol{w} \cdot \tau_w \frac{d\boldsymbol{w}}{dt} =$$

$$= 2\, \boldsymbol{w} \cdot \langle (v - \theta_v)\, \boldsymbol{u} \rangle = 2 \langle v\, (v - \theta_v) \rangle = 2 \left(\langle v^2 \rangle - \langle v \rangle^2 \right) > 0$$

and therefore positive as long as the activity v is not identically zero. We will return to the problem of instability in the next lecture.

Points to note

- In Hebbian plasticity, weight change depends on both pre- and post-synaptic activity.
- As post-synaptic activity also depends on pre-synaptic activity, weight change effectively depends doubly on pre-synaptic activity.
- Assuming that weight change is slow and driven by the ensemble average, we obtain the *correlation rule* ...

$$\tau_w \frac{d\boldsymbol{w}}{dt} = \langle \boldsymbol{Q} \rangle \cdot \boldsymbol{w} \qquad\qquad \boldsymbol{Q} \equiv \langle \boldsymbol{u}\,\boldsymbol{u} \rangle$$

- ... or, if we assume a threshold, the *covariance rule*

$$\tau_w \frac{d\boldsymbol{w}}{dt} = \langle \boldsymbol{C} \rangle \cdot \boldsymbol{w} \qquad\qquad \boldsymbol{C} \equiv \langle \boldsymbol{u}\,\boldsymbol{u} \rangle - \langle \boldsymbol{u} \rangle^2$$

- Neither rule provides for stable weights, at least not without further assumptions.

4. Correlations and covariances (reminder)

What are these "correlations" and "covariances" ? Correlation is the *average product* of absolute values

$$\boldsymbol{Q} = \langle \boldsymbol{u}\ \boldsymbol{u} \rangle = \langle u_i\, u_j \rangle$$

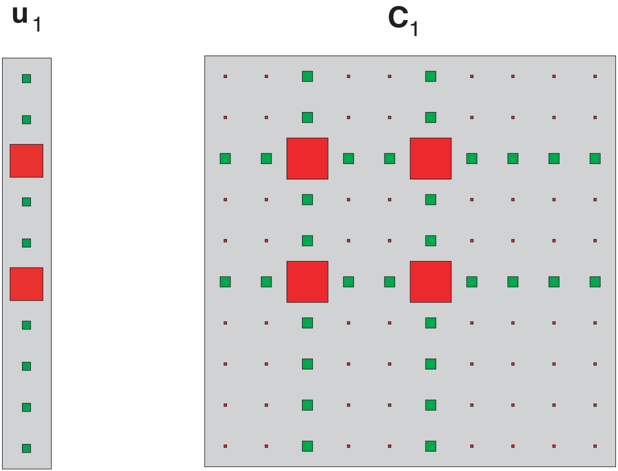
and covariance the *average product* of relative values (to the mean)

$$\boldsymbol{C} = \left\langle (\boldsymbol{u} - \langle \boldsymbol{u} \rangle)^2 \right\rangle = \langle \boldsymbol{u}\ \boldsymbol{u} \rangle - \langle \boldsymbol{u} \rangle \langle \boldsymbol{u} \rangle = \langle u_i\, u_j \rangle - \langle u_i \rangle \langle u_j \rangle$$

Correlation and covariance "coefficients" are additionally normalized by individual variance:

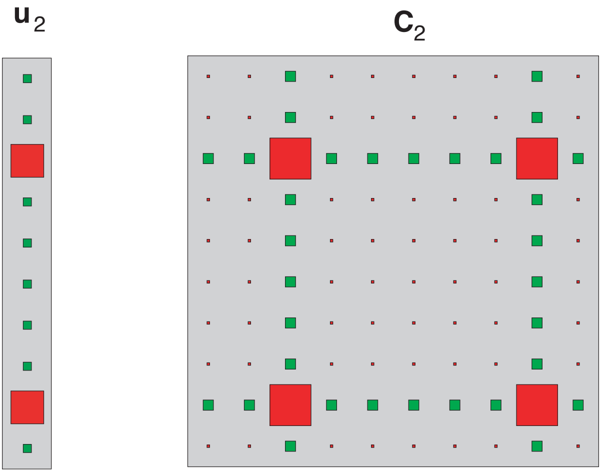
$$\boldsymbol{Q}_N = \frac{\langle u_i\, u_j \rangle}{\sqrt{\langle u_i^2 \rangle \langle u_j^2 \rangle}}, \qquad \boldsymbol{C}_N = \frac{\langle u_i\, u_j \rangle - \langle u_i \rangle \langle u_j \rangle}{\sqrt{\langle u_i^2 \rangle \langle u_j^2 \rangle}}$$

A zero-mean activity pattern, $\mathbf{u_1}$, and its covariance matrix $\mathbf{C_1}$.



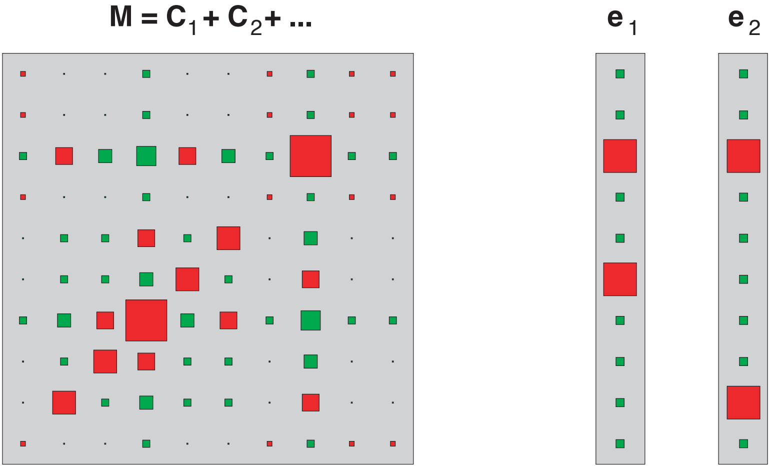
Red squares represent positive values, green squares negative values. Average value is aways zero.

A zero-mean activity pattern, $\mathbf{u_2}$, and its covariance matrix $\mathbf{C_2}$.



Red squares represent positive values, green squares negative values. Average value is aways zero.

Combined covariance matrix $\mathbf{M} = \mathbf{C_1} + \mathbf{C_2}$ and its two principal eigenvectors, $\mathbf{e_1}$ and $\mathbf{e_2}$. (Recall definition $\mathbf{M} \cdot \mathbf{e} = \lambda \mathbf{e}$.)



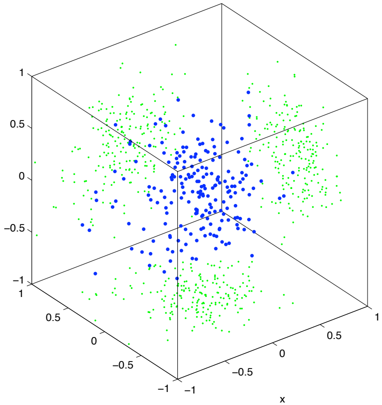
Red squares represent positive values, green squares negative values. Average value is aways zero.

Statistical structure of activity patterns

- Statistical structure of activity patterns is described by *correlations* (absolute values) and *covariances* (values relative to the mean).
- How informative one activity is about another ('are they a team?').
- Eigenvectors of *correlation* / *covariance* matrix are typical activity patterns.
- Dominant eigenvectors are dominant patterns.
- Statistical structure is often illustrated by scatter plots (see below).

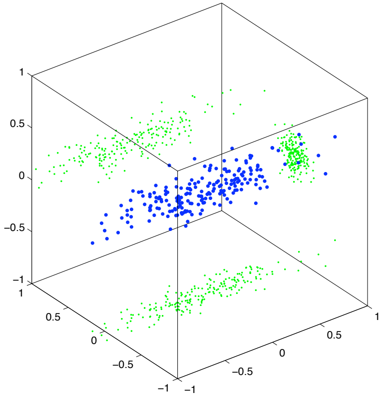
- Analyze statistical structure in eigenvector space (see also Lecture 14 Principal component analysis).

Consider an ensemble of 3-dimensional input vectors $\boldsymbol{u}$ with **zero mean** ($\langle u_i \rangle = 0$) and no evident statistical structure:



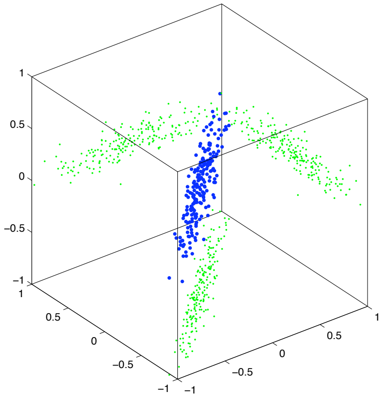
$$\boldsymbol{Q}_N \approx \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad \boldsymbol{C}_N \approx \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

A zero-mean ensemble with 4-times greater variance in one element:



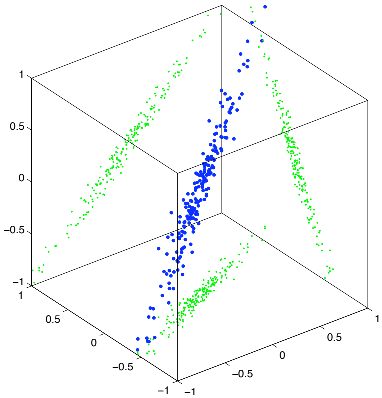
$$\boldsymbol{Q} \approx \begin{pmatrix} 4 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad \boldsymbol{Q}_N \approx \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

A zero-mean ensemble with correlated variance between u_1 and u_2 elements (they 'form a team'):



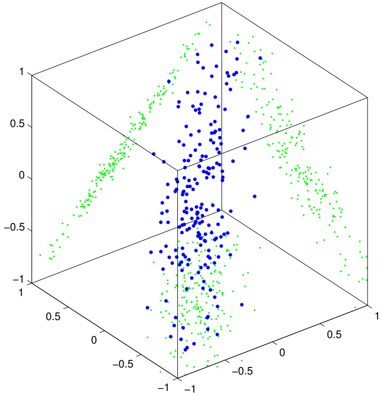
$$\boldsymbol{Q}_N \approx \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad \boldsymbol{C}_N \approx \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

A zero-mean ensemble with *differentially* correlated variance between three elements (note different *slopes*):



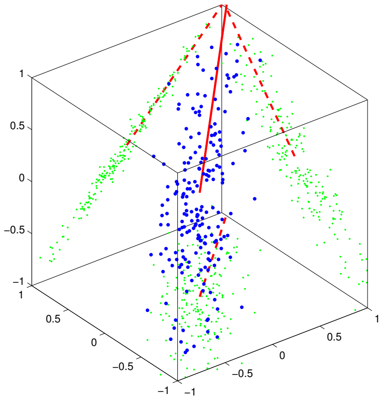
$$\boldsymbol{Q}_N \approx \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \qquad \boldsymbol{C}_N \approx \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

A zero-mean ensemble with *differentially* correlated variance between three elements (note different *tightness*):



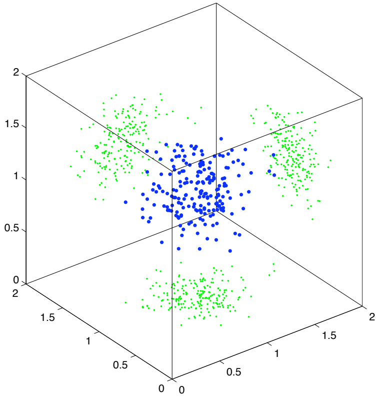
$$\boldsymbol{Q}_N \approx \begin{pmatrix} 1 & 0.8 & 1 \\ 0.8 & 1 & 0.8 \\ 1 & 0.8 & 1 \end{pmatrix}, \qquad \boldsymbol{C}_N \approx \begin{pmatrix} 1 & 0.8 & 1 \\ 0.8 & 1 & 0.8 \\ 1 & 0.8 & 1 \end{pmatrix}$$

Principal eigenvector of correlation matrix



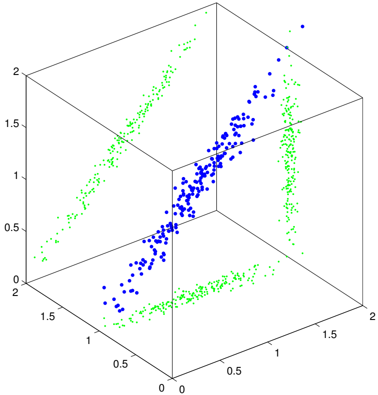
$$\boldsymbol{Q}_N \approx \begin{pmatrix} 1 & 0.8 & 1 \\ 0.8 & 1 & 0.8 \\ 1 & 0.8 & 1 \end{pmatrix}, \qquad \boldsymbol{e}_1 = \begin{pmatrix} 0.6 \\ 0.55 \\ 0.6 \end{pmatrix}, \qquad \lambda_1 = 2.75$$

An ensemble with **non-zero mean** ($\langle u_i \rangle = 1$) and no other statistical structure:



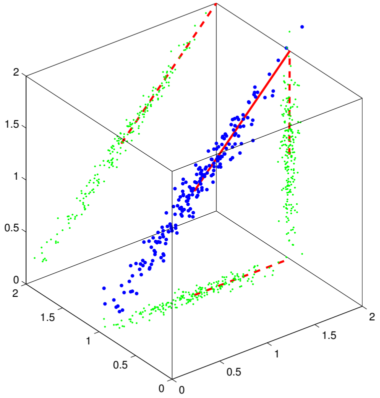
$$\boldsymbol{Q}_N \approx \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \qquad \boldsymbol{C}_N \approx \begin{pmatrix} 0.05 & 0 & 0 \\ 0 & 0.05 & 0 \\ 0 & 0 & 0.05 \end{pmatrix}$$

Non-zero mean ($\langle u_i \rangle = 1$) with correlated variance between two elements:



$$\boldsymbol{Q}_N \approx \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \qquad \boldsymbol{C}_N \approx \begin{pmatrix} 0.12 & 0 & 0.12 \\ 0 & 0.12 & 0 \\ 0.12 & 0 & 0.12 \end{pmatrix}$$

Principal eigenvector of covariance matrix



$$\boldsymbol{C}_N \approx \begin{pmatrix} 0.12 & 0 & 0.12 \\ 0 & 0.12 & 0 \\ 0.12 & 0 & 0.12 \end{pmatrix}, \qquad \boldsymbol{e}_1 = \begin{pmatrix} 0.7 \\ 0 \\ 0.7 \end{pmatrix}, \qquad \lambda_1 = 2.7$$

4. Points to note

- Statistical structure of value pairs is described by average products: *correlations* (absolute values) and *covariances* (values relative to the mean).
- They measure how informative one value is about the other ('do they form a team?').
- Informativeness depends on the 'tightness', not the 'slope', of a scatter plot.
- Exceptions confirm the rule: zero or infinitive slopes are uninformative.
- Correlation and covariance *coefficients* are normalized to the variability of individual values

Know the definitions of correlations, covariances, and coefficients! Understand their link to eigenvectors!

5 Linear analysis

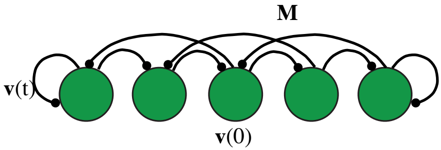
Recall linear *recurrent* network with connectivity $\boldsymbol{M}$

$$\tau \frac{d\boldsymbol{v}}{dt} = -\boldsymbol{v} + \boldsymbol{M} \cdot \boldsymbol{v} = (\boldsymbol{M} - \boldsymbol{I}) \cdot \boldsymbol{v} = \boldsymbol{A} \cdot \boldsymbol{v}$$

solved by eigenvector expansion

$$\boldsymbol{v}(t) = \sum_{\mu=1}^{N} C_\mu \, e^{\lambda_\mu \, t/\tau} \, \boldsymbol{e}_\mu, \qquad \lambda_\mu \, \boldsymbol{e}_\mu = \boldsymbol{A} \cdot \boldsymbol{e}_\mu$$

Recurrent propagation of 'eigenpatterns' of connection matrix.



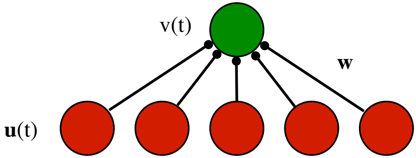
Compare analysis of Hebbian plasticity of *feedforward* connectivity

$$\tau \frac{d\boldsymbol{w}}{dt} = \boldsymbol{C} \cdot \boldsymbol{w}, \qquad \boldsymbol{C} = \langle \boldsymbol{u}\,\boldsymbol{u} \rangle - \langle \boldsymbol{u} \rangle \langle \boldsymbol{u} \rangle$$

solved by eigenvector expansion

$$\boldsymbol{w}(t) = \sum_{\mu=1}^{N} C_\mu \, e^{\lambda_\mu \, t/\tau} \, \boldsymbol{e}_\mu, \qquad \lambda_\mu \, \boldsymbol{e}_\mu = \boldsymbol{C} \cdot \boldsymbol{e}_\mu$$

Growth of feedforward weights depends on 'eigenpatterns' of input covariance!



Eigenvectors of covariance matrix

If we are willing to ignore the instability, we can analyze the covariance rule with standard techniques, obtaining an explicit solution for $\boldsymbol{w}(t)$ in terms of eigenvalues and eigenvectors of the covariance $\boldsymbol{C}$.

$$\tau_w \frac{d\boldsymbol{w}}{dt} = \langle (v - \langle v \rangle)\, \boldsymbol{u} \rangle = \boldsymbol{C} \cdot \boldsymbol{w}, \qquad \boldsymbol{C} \equiv \langle \boldsymbol{u}\, \boldsymbol{u} \rangle - \langle \boldsymbol{u} \rangle \langle \boldsymbol{u} \rangle$$

$$\boldsymbol{C} \cdot \boldsymbol{e}_\mu = \lambda_\mu\, \boldsymbol{e}_\mu, \qquad \mu = 1, \ldots, N_{\boldsymbol{u}}, \qquad \lambda_i \geq \lambda_{i+1} \geq 0$$

where $\boldsymbol{e}_\mu$ and λ_μ are the μ-th eigenvector and eigenvalue of $\boldsymbol{C}$. As covariances are symmetric, all eigenvalues λ_μ are real and non-negative. Thus, any solutions will be *instable*!

ctd

The solution takes the form

$$\boldsymbol{w}(t) = \sum_{\mu=1}^{N_{\boldsymbol{u}}} c_\mu(t)\, \boldsymbol{e}_\mu$$

where the unknown coefficients $c_\mu(t)$ are obtained by substitution:

$$\tau_w \frac{d\boldsymbol{w}}{dt} = \tau_w \sum_{\mu=1}^{N_{\boldsymbol{u}}} \frac{dc_\mu(t)}{dt}\, \boldsymbol{e}_\mu = \boldsymbol{C} \cdot \boldsymbol{w} = \boldsymbol{C} \cdot \sum_{\mu=1}^{N_{\boldsymbol{u}}} c_\mu(t)\, \boldsymbol{e}_\mu$$

$$\tau_w \sum_{\mu=1}^{N_{\boldsymbol{u}}} \frac{dc_\mu(t)}{dt}\, \boldsymbol{e}_\mu = \sum_{\mu=1}^{N_{\boldsymbol{u}}} c_\mu(t)\, \lambda_\mu\, \boldsymbol{e}_\mu \quad \Bigg| \quad \cdot\, \boldsymbol{e}_\nu$$

$$\tau_w \frac{dc_\nu(t)}{dt} \quad = \quad \lambda_\nu\, c_\nu(t)$$

ctd

Solving

$$\tau_w \frac{dc_\mu(t)}{dt} = \lambda_\mu \, c_\mu(t)$$

gives

$$c_\mu(t) = c_\mu(0) \, e^{\lambda_\mu t/\tau_w}$$

and

$$\boldsymbol{w}(t) = \sum_{\mu=1}^{N_{\boldsymbol{u}}} c_\mu(0) \, e^{\lambda_\mu t/\tau_w} \, \boldsymbol{e}_\mu$$

Connection weights grow exponentially ($\lambda_\mu \geq 0$) in the direction of the eigenvectors $\boldsymbol{e}_\mu$.

Speed of growth

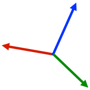
Speed of growth in each 'eigendirection' depends on associated eigenvalue.

If the largest (=principal) eigenvalue λ_1 is unique, weight growth along the associated (=principal) eigenvector $\boldsymbol{e}_1$ dominates development:

$$\boldsymbol{w}(t \to \infty) \propto \boldsymbol{e}_1$$

Caution: this is true only if the initial weights do not prevent such development, if $\boldsymbol{w}(0) \cdot \boldsymbol{e}_1 \neq 0$.

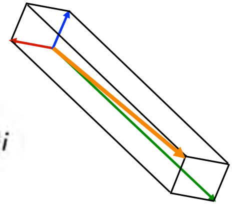
The weight vector grows in the direction of all eigenvectors, but fastest in the direction of the principal eigenvector!



$\mathbf{e}_{1,2,3}$

$\lambda_{1,2,3}\,\mathbf{e}_{1,2,3}$

$$\mathbf{w}(t) = \sum_{i=1}^{3} \exp\left(\lambda_i\, t\right)\, \mathbf{e}_i$$



Points to note

- Linear analysis of the covariance rule provides useful *qualitative* information.
- The weight vector grows as a weighted sum of the eigenvectors of the covariance matrix:

$$\mathbf{w}(t) = \sum_{\mu=1}^{N_{\mathbf{u}}} c_\mu(0)\, e^{\lambda_\mu t/\tau_w}\, \mathbf{e}_\mu$$

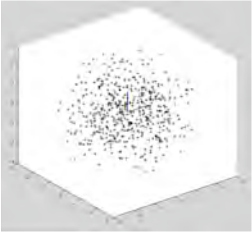
- While growth in all direction is exponential, it is dominated by growth along the principal eigenvector.
- Thus, the weight vector aligns itself with the principal eigenvector.
- The *principal eigenvector* identifies the ‘tightest’ direction of the scatter plot of correlations.

6 Learning input covariances

Lets consider a Hebbian covariance rule with weight normalization. At every time-step, all weights are reduced proportionately to maintain an overall square weight of unity:

$$\tau_w \frac{d\boldsymbol{w}}{dt} = v\,(\boldsymbol{u} - \langle \boldsymbol{u} \rangle)\,, \qquad \sqrt{\boldsymbol{w}\cdot\boldsymbol{w}} = 1$$

For uncorrelated inputs $\boldsymbol{u}$ ...

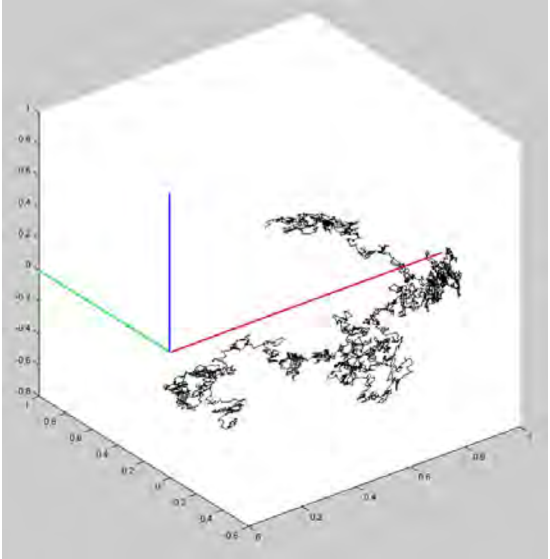


$\lambda_{1,2,3} =$ 1.0706 1.0631 1.0332

$\lambda_1 = 1.07$
$\lambda_2 = 1.06$
$\lambda_3 = 1.03$

ctd

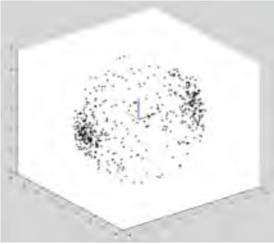
... the weights $\boldsymbol{w}$ fail to converge



ctd

$$\tau_w \frac{d\boldsymbol{w}}{dt} = v \left(\boldsymbol{u} - \langle \boldsymbol{u} \rangle\right), \qquad \sqrt{\boldsymbol{w} \cdot \boldsymbol{w}} = 1$$

Whereas for correlated inputs $\boldsymbol{u}$...

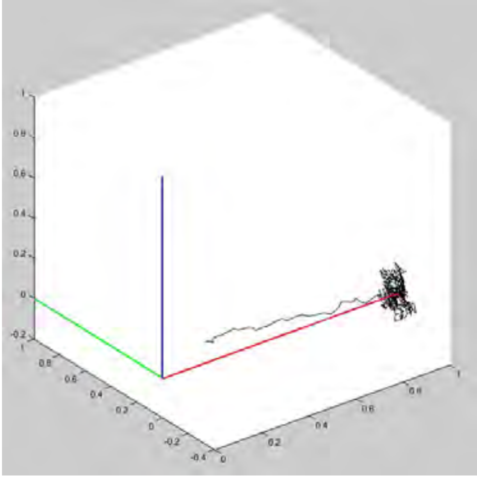


$\lambda_{1,2,3} =$ 6.7563 1.0347 0.7292

$$\lambda_1 = 6.76$$
$$\lambda_2 = 1.035$$
$$\lambda_3 = 0.73$$

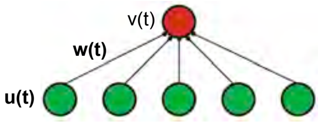
ctd

... the weights $\boldsymbol{w}$ converge to the principal eigenvector:



Intuitive summary

- For simplicity, assume that all inputs are (on average) equally large and equally variable.
- The weights of co-varying inputs ("team-forming") are strengthened, because they succeed more reliably in triggering post-synaptic activity.
- The weights of other inputs ("loners") are weakened, because they succeed less reliably.
- The Hebbian covariance rule "internalizes" the statistical input structure: statistical "teams" becomes groups with large synaptic weights.



General summary

- Formal learning models distinguish "unsupervised", "supervised", and "reinforcement " learning.

- When the Hebbian rule ("fire together, wire together") is formalized, it leads to the "covariance rule"

$$\tau_w \, \dot{\boldsymbol{w}} = \boldsymbol{C} \cdot \boldsymbol{w}$$

- Ignoring instability permits linear analysis and shows that weights $\boldsymbol{w}$ converge to principal eigenvector(s) $\boldsymbol{e}_{1,\ldots}$ of the input covariance $\mathbf{C} = \mathbf{u} \cdot \mathbf{u}$.

- Hebbian synapses learn input covariances, internalizing the statistical structure of the outside world.

Next:
Stable Hebbian Plasticity