Synaptic Plasticity

The impact of a synapse onto its postsynaptic neuron (amplitude of EPSPs/IPSPs) is termed the *weight* (efficacy) of a synapse.

This weight undergoes dynamics on a variety of time scales

These dynamics can be important in the context of

- learning
- memory
- development
- optimization

The Hebb Rule

Donald Hebb (1949): He proposed that functional relationship between a presynatic neuron (A) and a postsynaptic neuron (B) could change if A frequently took part in exciting B.

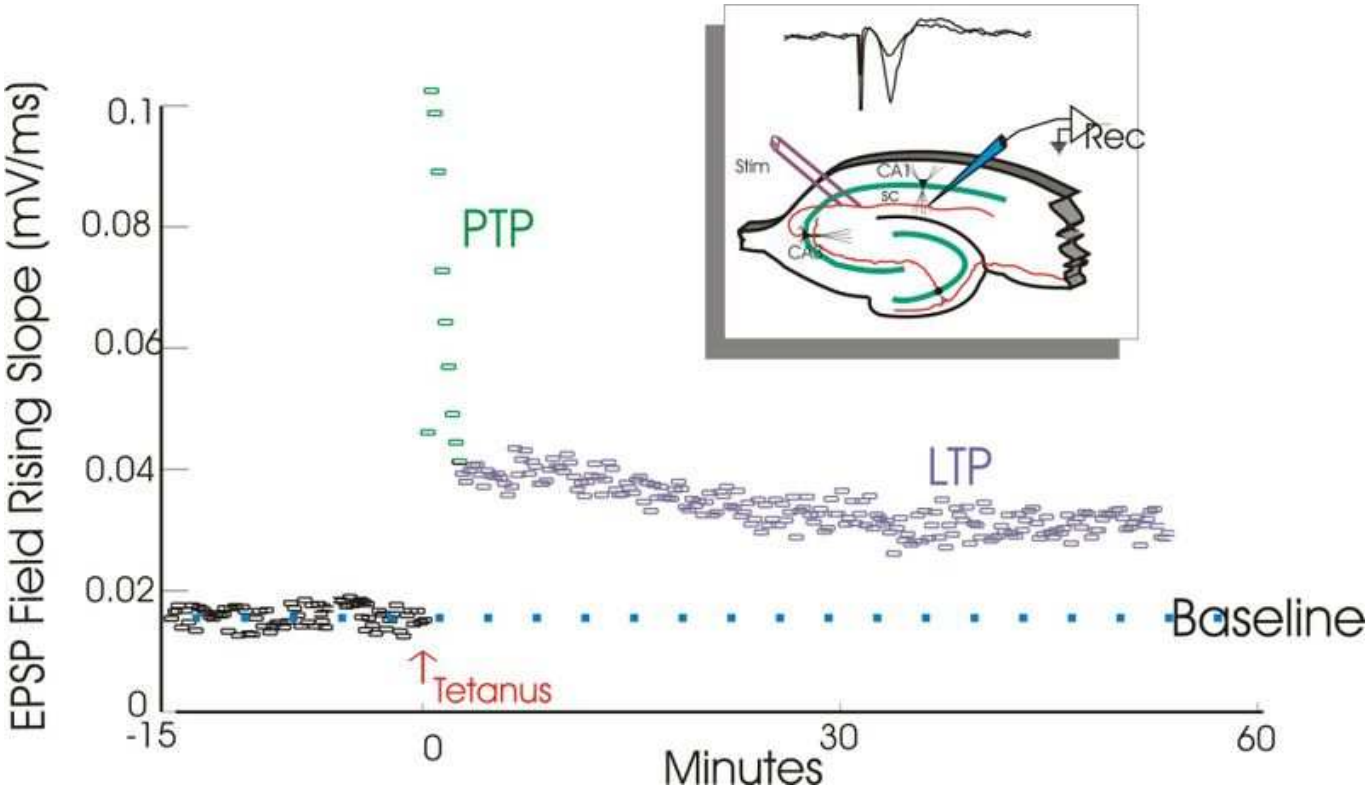
"When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficacy, as one of the cells firing B, is increased."

Experiment: Current is injected into neurons A and B such that they are both firing strongly for some time (ca. 100 msec).

A comparison between the EPSP-amplitudes from A to B before and after stimulation shows a significant increase in synaptic efficacy.

Long-Term Potentiation (LTP)

LTP is the long-lasting enhancement in communication between two neurons that results from stimulating them simultaneously.



The precise mechanisms for this enhancement have not been fully established

The opposite effect is termed ***Long-Term-Depression (LTD)***.

Analysis of Hebb's rule

We consider an analog neuron with firing rate y and N presynaptic neurons coupled by weights $\mathbf{w} = (w_1, \ldots, w_N)^T$.

The presynaptic neurons have firing rates $\mathbf{x} = (x_1, \ldots, x_N)^T$.

$$\begin{aligned} y &= \sum_{j=1}^{N} w_j \cdot x_j \\ \Delta w_i &= \gamma \cdot x_i \cdot y \end{aligned}$$

γ is the *lerning rate*: $0 < \gamma \ll 1$

Consider a set of p static input-patterns $\{x^\mu, 1 \leq \mu \leq p\}$

$x^\mu = (x_1^\mu, \ldots, x_N^\mu)^T$

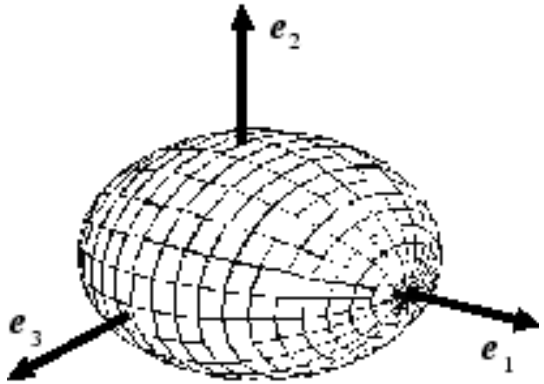
Analysis of Hebb's rule

$$y = \sum_{j=1}^{N} w_j \cdot x_j \qquad \Delta w_i = \gamma \cdot x_i \cdot y$$

Analysis of Hebb's rule

The *correlation matrix* C is the matrix with entries $c_{i,j} = \frac{1}{p}\sum_{\mu=1}^{p} x_i^\mu \cdot x_j^\mu$.

If the input patterms x^μ are 0 mean, then the eigenvector of C with the largest eigenvalue λ_{max} gives us the direction that maximizes the variance of the data set.



The operation of finding this direction is called *Principal Component Analysis (PCA)*.

When applying Hebb's rule, the expectaion value of w converges to this eigenvector.

PCA in general

PCA is an *unsupervised* learning technique to find the linearly relevant dimensions in data.

Consider d-dimensional data $\mathbf{x}^1 \dots \mathbf{x}^p \in \mathbb{R}^d$.

We want to find $M < d$ many basis vectors $\mathbf{u}_i$ such that the error $E = \frac{1}{2} \sum_{i=1}^{p} ||\mathbf{x}^i - \tilde{\mathbf{x}}^i||^2$ of the approximation

$$\tilde{\mathbf{x}}^i = \sum_{j=1}^{M} z_j(i)\mathbf{u}_j + \mathbf{b}$$

is minimal, where $\mathbf{b}$ is some constant vector and $\tilde{\mathbf{x}}^i$ is an approximation on $\mathbf{x}^i$.

One can show that the error is minimal for $\mathbf{b}$ being the mean of the data and **choosing the M eigvectors with largest eigenvalues of the covariance matrix as basis**. These eigenvectors are called the *principal components* of the data.

Covariance matrix of the data

$$\Sigma = \sum_{i=1}^{p} (\mathbf{x}^i - \overline{\mathbf{x}})(\mathbf{x}^i - \overline{\mathbf{x}})^T.$$

Problems of Hebb's rule

- The weights grow unbounded.

  Extensions: Normalise the weights e.g. by ***Oja's rule***

$$\Delta w_i = \gamma \cdot x_i \cdot y - \gamma \cdot w_i \cdot y^2$$

  Normalizes the weight vector to $\|w\| = 1$.

- Principal components are only meaningful if the input distribution has mean 0. However, rates are always positive.

  Extensions: The ***covariance rule*** considers deviations from the mean firing rate $\langle \cdot \rangle$.

$$\Delta w_i = \gamma \cdot (x_i - \langle x_i \rangle) \cdot (y - \langle y \rangle)$$

Independent Component Analysis (ICA)

- The projections of data on its principal components are uncorrelated.
- ICA tries to find components such that projections of the data are statistically independent.

Random variables x_1, x_2 are independent if knowing the value of x_1 does not give any information about the value of x_2, or (for $p(x)$ being the prob. density function of x)

$$p(x_1, x_2) = p(x_1)p(x_2).$$

It follows:

$$E[h_1(x_1)h_2(x_2)] = E[h_1(x_1)]E[h_2(x_2)]$$

for any functions h_1, h_2

Independent Component Analysis (ICA)

Observe a stochastic vector $\mathbf{x}(t) = (x_1(t), \ldots, x_m(t))$.

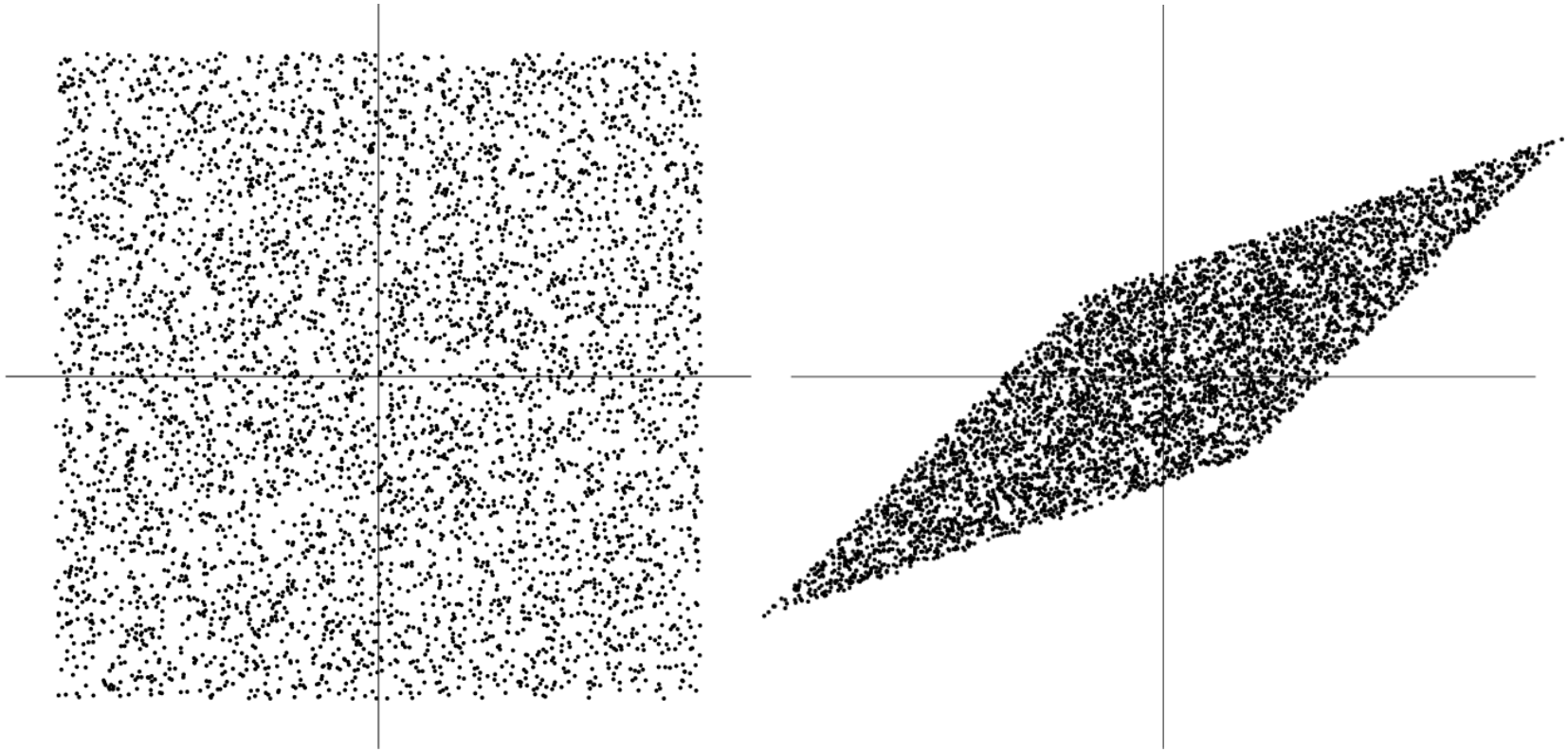
Each $x_i(t)$ is assumed to be a linear combination of n unknown *independent components*

$\boldsymbol{s}(t) = (s_1(t), \ldots, s_n(t))$

$\mathbf{x}(t) = A\boldsymbol{s}(t)$.

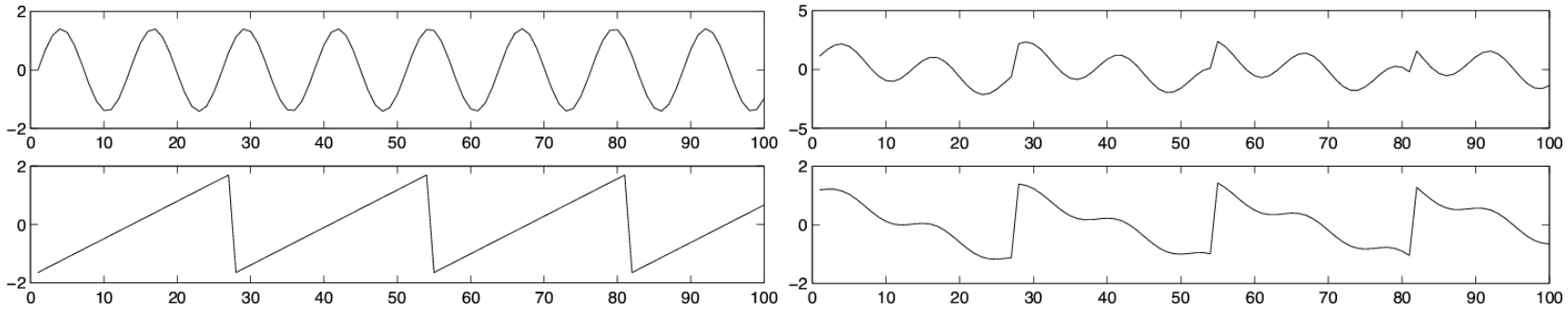
In ICA, one tries to estimate the *mixing matrix* A and the independent components $\boldsymbol{s}(t)$, knowing the observed variables only.

Examples of mixed signals (distributions)

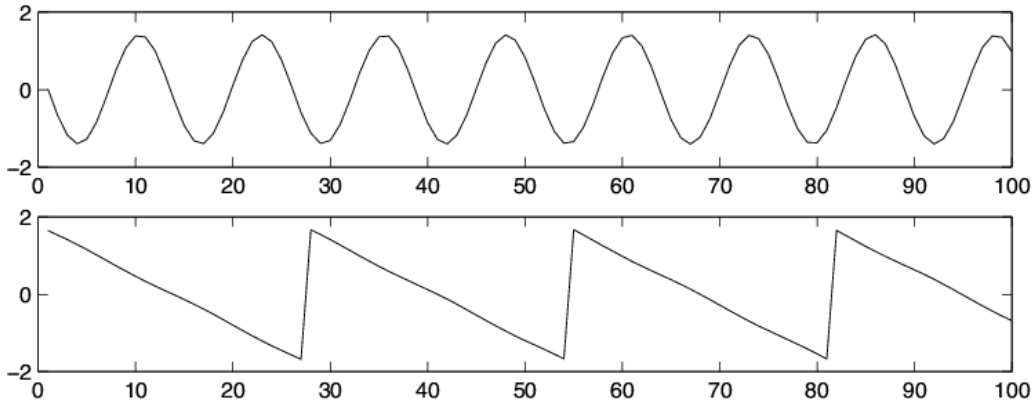


Examples of mixed signals (distributions)

The origina signals, the mixed signals, and



the reconstructed signals (with ICA).



Why do we talk about ICA?

- How is coding organized in primary sensory areas (e.g. V1)?
- The coding is believed to be optimized for something.
- It is believed that the coding (receptive field properties) is learned during development.
- What are the principles behind that learning?
- ICA is one candidate for that.
  - It produces an efficient coding scheme.
  - It was shown that ICA (and related approaches) can reproduce many properties of cells in visual cortex.
  - It tries to find the causes of the observed signal (generative model).

Linear Generative Models of Natural Images

$p(x, y)$ Pixel gray-scale value of an image patch at position x, y.

Each image patch is considered to be a linear superposition of features or basis vectors a_i:

$$p(x, y) = \sum_{i=1}^{n} a_i(x, y) s_i \qquad \text{, or} \qquad \boldsymbol{p} = A\boldsymbol{s}.$$

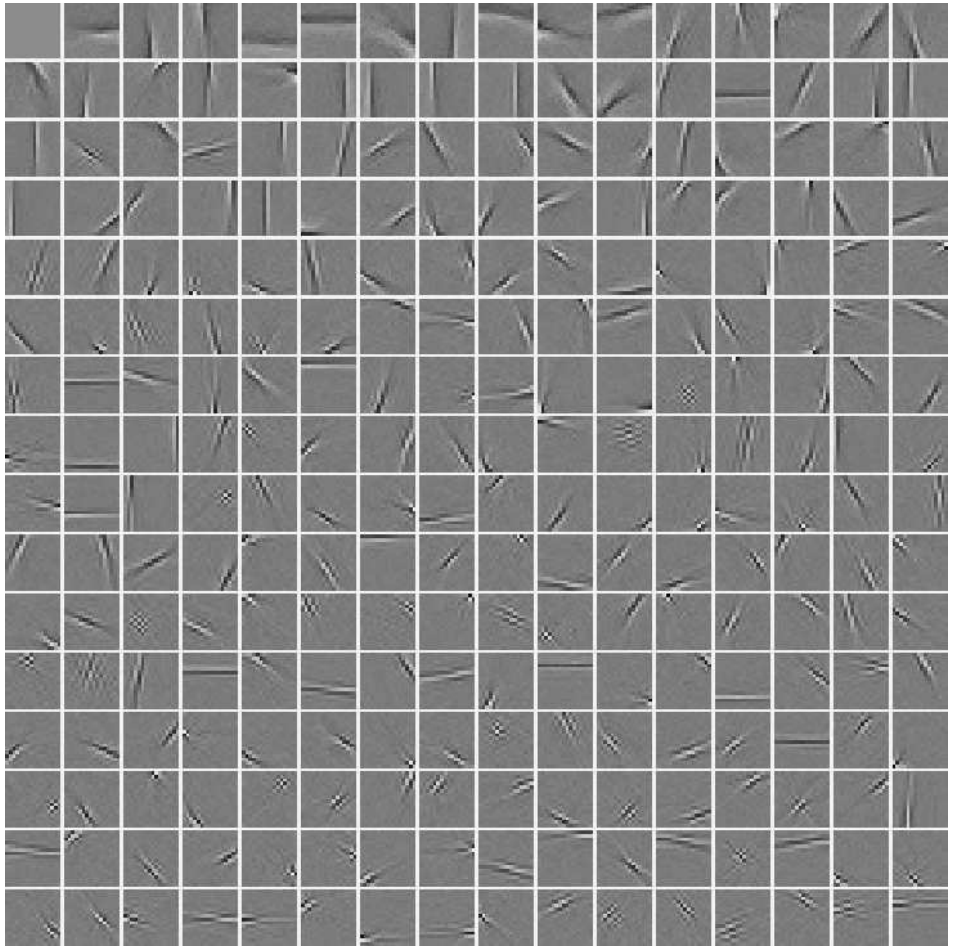
The s_i are stochastic coefficient, different from patch to patch (also called *latent variables*).

“Simple Cell” output in an biological interpretation:

$$y_i = \sum_{x,y} w_i(x, y) p(x, y) \qquad \text{, or} \qquad \boldsymbol{y} = W\boldsymbol{p} \quad \text{with} \quad W = A^{-1}.$$

Reproduction of Receptive Fields by ICA

An ICA was performed on natural image data (videos).



ICA by maximisation of NonGaussianity

When two variables are mixed, the distribution of the resulting variable is closer to a Gaussian.

Non-Gaussianity can be measured by Kurtosis (we assume zero mean):

$$kurt(y) = E[y^4] - 3(E[y^2])^2$$

For a Gaussian, the Kurtosis is zero.

Subgaussian variables have negative kurtosis.

Supergaussian variables have positive kurtosis.

ICA by gradient ascent on kurtosis

We compute $y = \mathbf{w}^T\mathbf{x}$ and want y to be an independent component of $\mathbf{x}$.

We do this by maximising the (absolute value) of the kurtosis (here we assume that the data is white, i.e., it has zero mean, is uncorrelated, and has unit variance).

$$\frac{|\partial kurt(y)|}{\partial \mathbf{w}} = 4\text{sign}(kurt(\mathbf{w}^T\mathbf{x}))(E[\mathbf{x}(\mathbf{w}^T\mathbf{x})^3] - 3\mathbf{w} \parallel \mathbf{w} \parallel^2)$$

We can do this with the online-learning rule

$$\Delta\mathbf{w} \propto \text{sign}(kurt(\mathbf{w}^T\mathbf{x}))\mathbf{x}(\mathbf{w}^T\mathbf{x})^3$$

and normalizing $\mathbf{w}$ after each update.

Note that the latter part of the rule $\mathbf{x}(\mathbf{w}^T\mathbf{x})^3$ is a non-linear Hebbian term $\mathbf{x}y^3$.