Hebbian Model Learning

1. Types of Learning.
2. Biology of Learning.
3. Model Learning: Bias and Parsimony.
4. Principal Components Analysis revisited.
5. Conditional Principal Components Analysis (CPCA).
6. Biological Implementation of CPCA.
7. Renormalization and Contrast Enhancement.
8. Self-organizing learning: Competition & CPCA.

Overview of Learning

- Learning is tuning detectors locally to achieve global results.
- Two main types: learning internal **model** of environment, & learning to solve a **task** (produce target output from input).
- Biology suggests Hebbian learning: associative LTP and LTD.

Model Learning

aka Unsupervised learning

- Models learned in self-organizing, unsupervised manner.
- Based directly on Hebbian LTP/LTD mechanisms.
- Hebbian learns about *correlations:*
  - principal components analysis (PCA)
  - *conditional* PCA — PCA on a subset of input patterns.
  - Inhibitory competition provides conditionalizing.

Task Learning

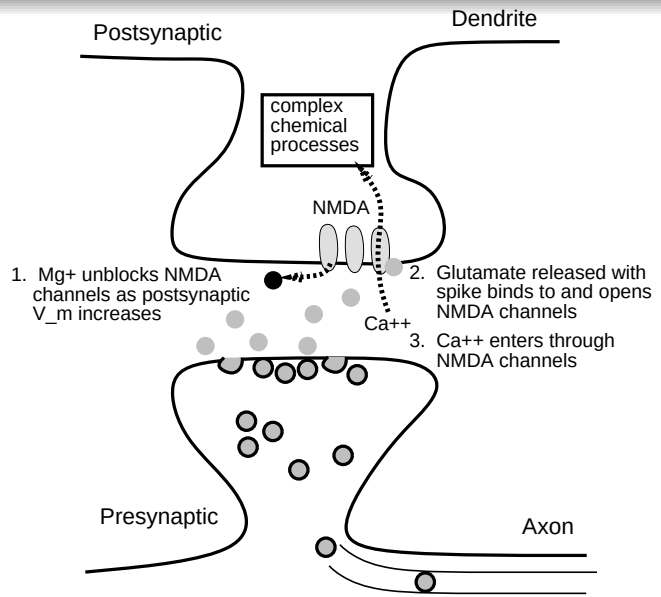
aka Supervised learning

- Task learning = producing output given input.
  - Hebbian can't learn many tasks.
  - Error-driven learning can, using target - output *error*.
- Two related algorithms:
  - Delta rule: no hidden units.
  - Backpropagation: hidden units!
- What about the biology:
  - Biology does not support backpropagation of errors..
  - GeneRec: Errors computed via *activations*, uses LTP/D.

Combined Task & Model Learning

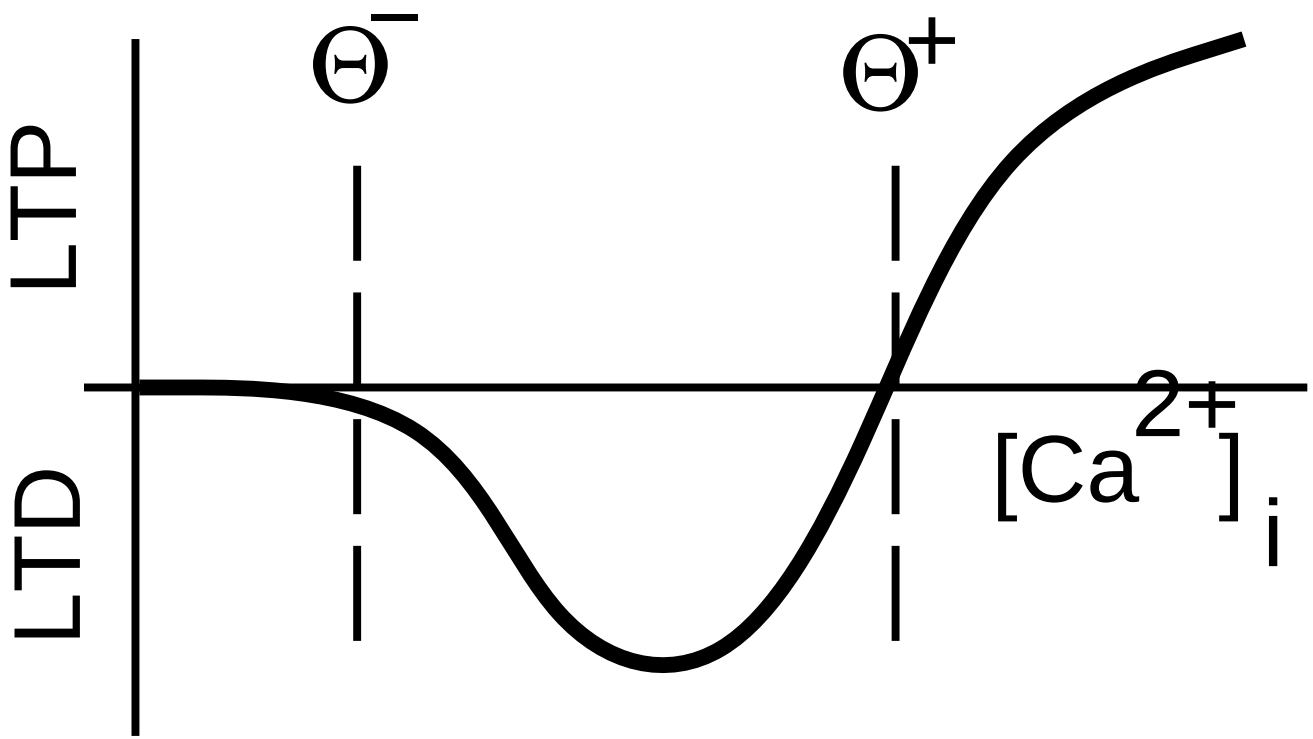
- Combination of model and task better than either alone.
- Problems with sequences and temporal delays:
  - Sequences require dynamically updated *context* reps.
  - Reinforcement learning spans gaps between cause & effect.
  - RL also models conditioning phenomena.

Biology: NMDA-mediated LTP/D



1. As postsynaptic V_m rises, Mg^+ moves out of NMDA receptor channels.
2. Glutamate released by presynaptic spike diffuses across the synaptic cleft and binds to NMDA channel, opening it.
3. Open NMDA channels allow postsynaptic Ca^{++} influx, triggering cascade which modifies synaptic efficacy of primary excitatory (AMPA) receptors.
4. If Ca^{++} high, LTP. If low, LTD.

Biology: NMDA-mediated LTP/D

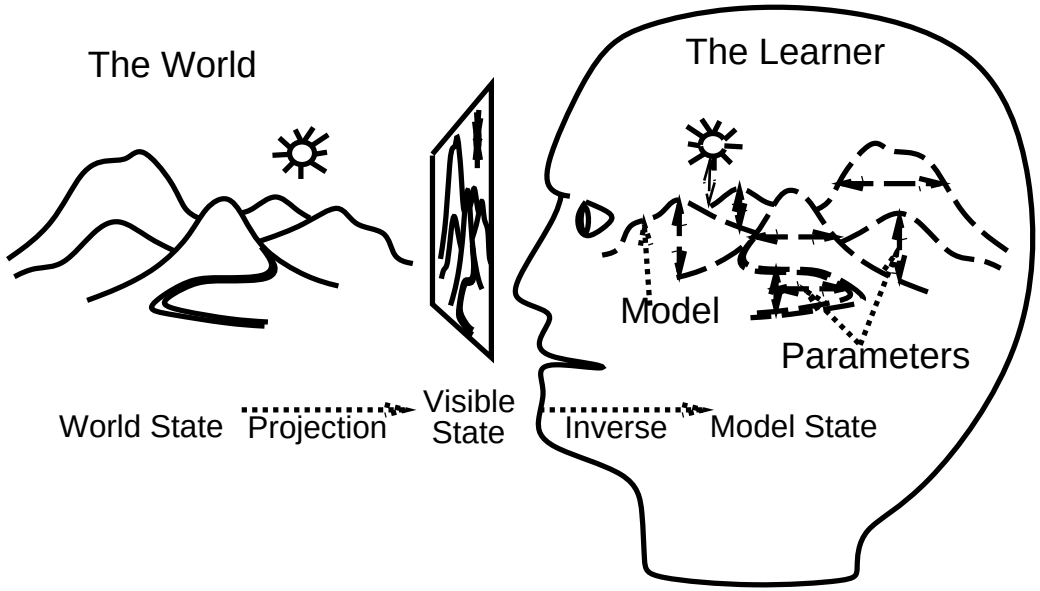


Strong activity (Ca^{++}) = LTP, weak = LTD

Complications

- Metabatropic glutamate (mGlu).
- Multiple sources of calcium (voltage-gated channels).
- Critical inducing properties (timing, intensity, frequency, duration, etc) largely unknown (100Hz for 1 sec implausible).
- Regulation by modulatory NTs (dopamine, seratonin).

Model Learning: Bias & Parsimony



Get a lot of poor quality information.

Learning needs to generate internal models of the world.

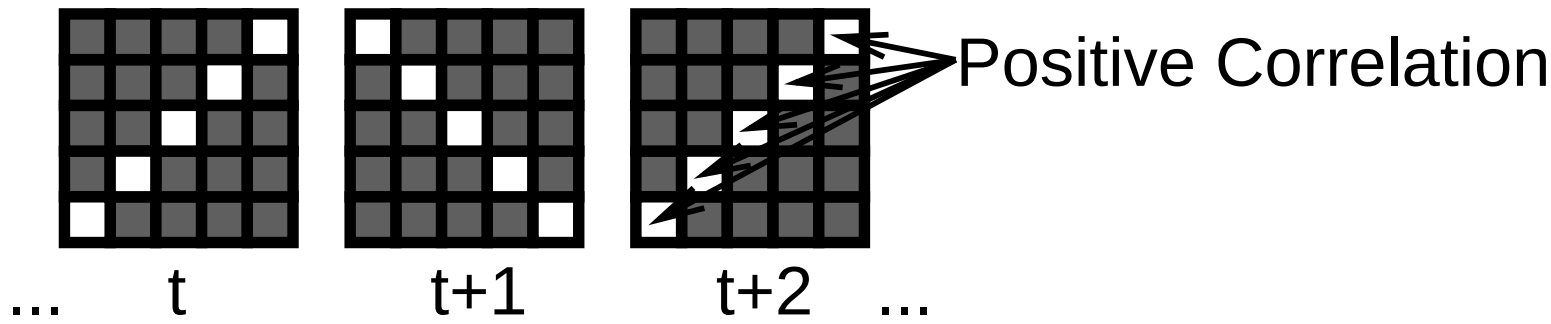
Inversion process is ill-posed: integrate across experiences.

Also need biases to augment and structure sensory info.

Bias-variance dilemma.

Bias towards parsimony critical (Occam's razor).

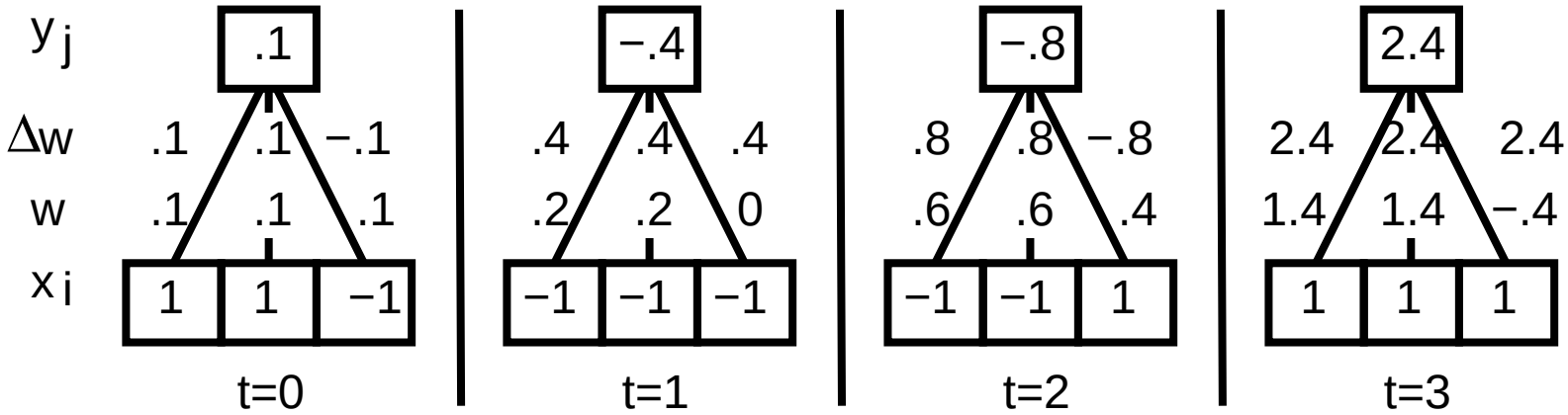
Correlational Learning: PCA revisited



Represent the strongest (principal) correlations in the environment.

PCA = Principal Components Analysis.

PCA Demo



Linear Activation:

$$y_j = \sum_i x_i w_{ij} \tag{1}$$

Simple Hebb rule:

$$\Delta_t w_{ij} = \epsilon x_i y_j \tag{2}$$

Weights get stronger for the two units that are correlated, don't get stronger for uncorrelated unit!

Mathematical Analysis

Sum over events:

$$\Delta w_{ij} = \epsilon \sum_t x_i y_j \tag{3}$$

Expected value:

$$\langle \Delta_t w_{ij} \rangle_t = \langle x_i y_j \rangle_t \tag{4}$$

Weights dominated by strongest component of the correlation matrix **C**:

$$\begin{aligned} \langle \Delta w_{ij} \rangle_t &= \langle x_i \sum_k x_k w_{kj} \rangle_t \\ &= \sum_k \langle x_i x_k \rangle_t \langle w_{kj} \rangle_t \\ &= \sum_k \mathbf{C}_{ik} \langle w_{kj} \rangle_t \end{aligned} \tag{5}$$

Mathematical Analysis

True correlations:

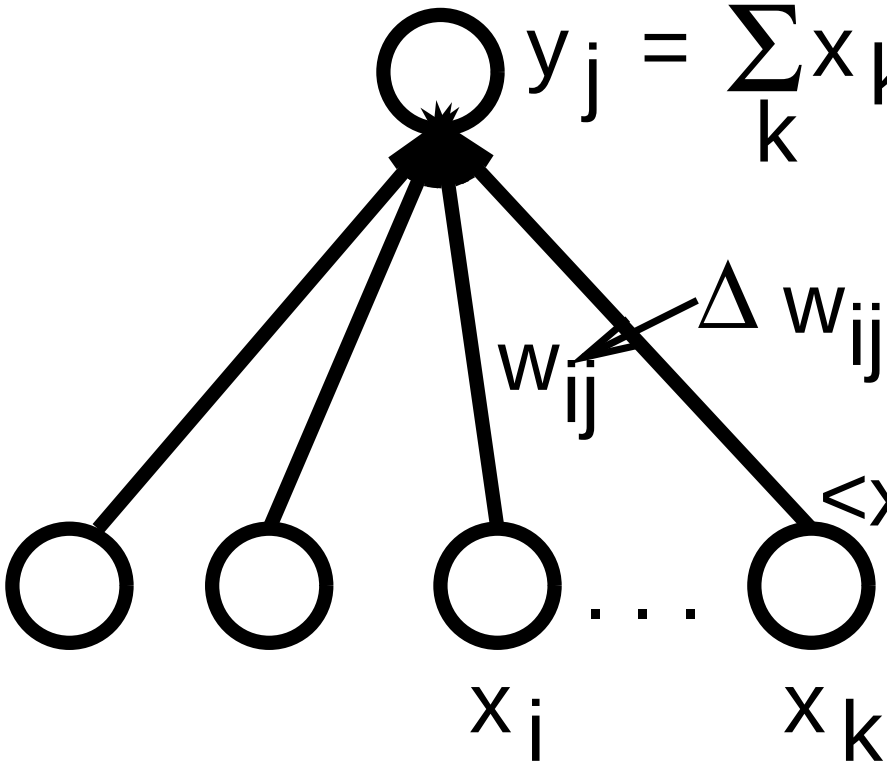
$$\mathbf{C}_{ik} = \frac{\langle (x_i - \mu_i)(x_k - \mu_k) \rangle_t}{\sqrt{\sigma_i^2 \sigma_k^2}} \tag{6}$$

(we assume zero mean, unit variance).

Eigenvectors:

$$\Delta \mathbf{w_j} = \mathbf{C} \mathbf{w_j} \tag{7}$$

Summary of Hebbian PCA



Problems

1. Weights will grow infinitely large with a simple Hebb rule.
2. What happens with additional receiving (output) units? They will all represent the same correlations!

Out of Bounds: Normalization

Weights will grow infinitely large.

Oja's normalization rule:

$$\Delta w_{ij} = \epsilon(x_i y_j - y_j^2 w_{ij}) \tag{8}$$

Based on $\mathbf{C_{ij}}$ so will still perform PCA. See HKP.

Equilibrium weight analysis:

$$\begin{aligned} 0 &= \epsilon x_i y_j - y_j^2 w_{ij} \\ w_{ij} &= \frac{x_i}{y_j} \\ w_{ij} &= \frac{x_i}{\sum_k x_k w_{kj}} \end{aligned} \tag{9}$$

Weight from a given input unit will end up representing the ratio of that input's activation to the total weighted activation over *all* the inputs. Denominator keeps weights from growing without bound.

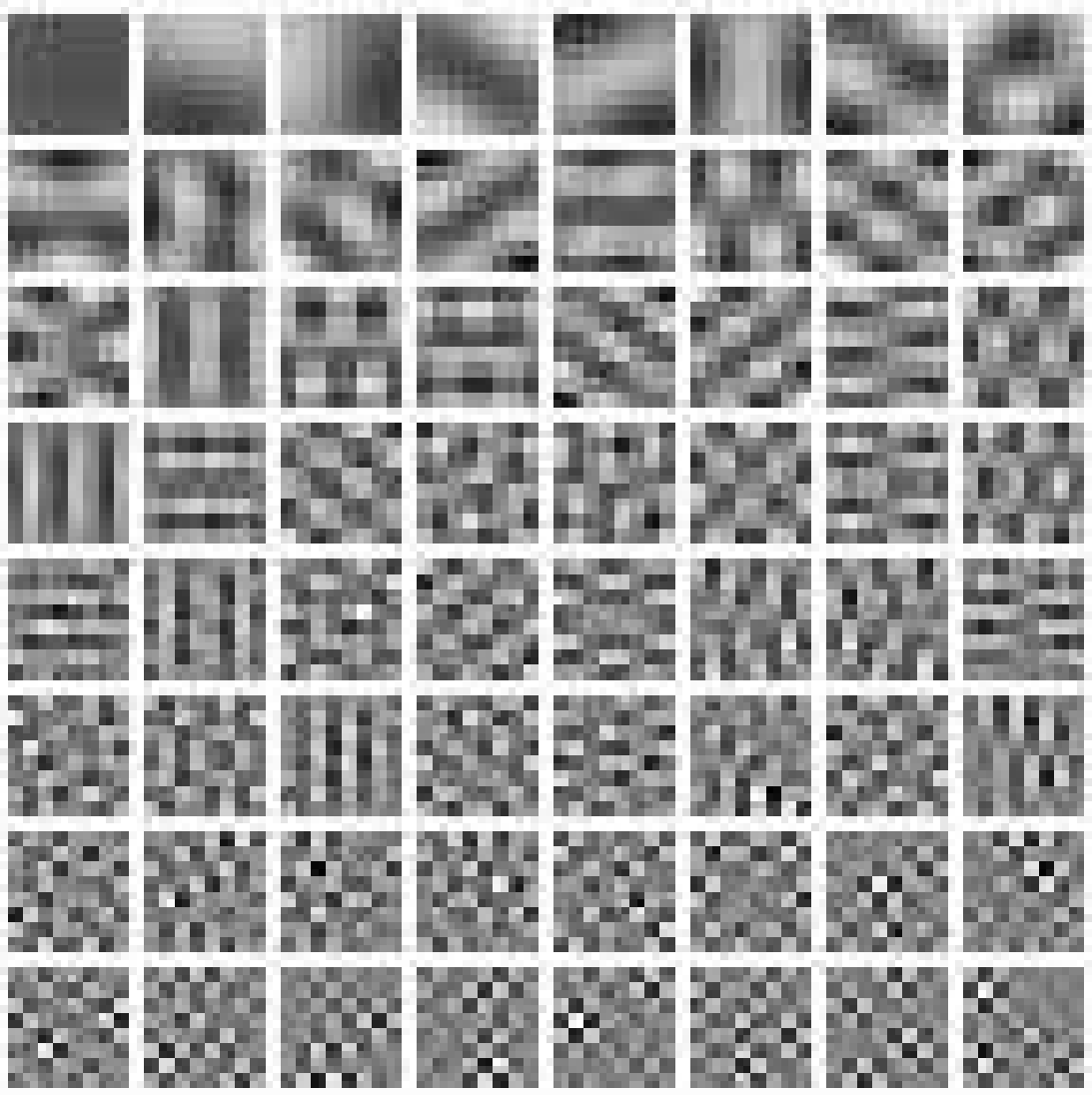
Multiple Units: Sequential PCA (SPCA)

One way of using multiple receiving units is to sequentially order them from strongest correlations to weakest.

This is what the standard PCA algorithm does — we call it “Sequential PCA” (SPCA)

E.g., Oja “Neural networks, principal components, and subspaces”, International Journal of Neural Networks, 1989.

Sequential PCA (SPCA) of Natural Visual Scenes



First principal component upper left, each square shows 8x8 weight matrix from input units.

Problem with Sequential PCA (SPCA)

If you average over everything, all the interesting correlations disappear!

The world is a big *heterarchy*, not a *hierarchy* — there isn't one overall principal component, just a bunch of smaller ones..

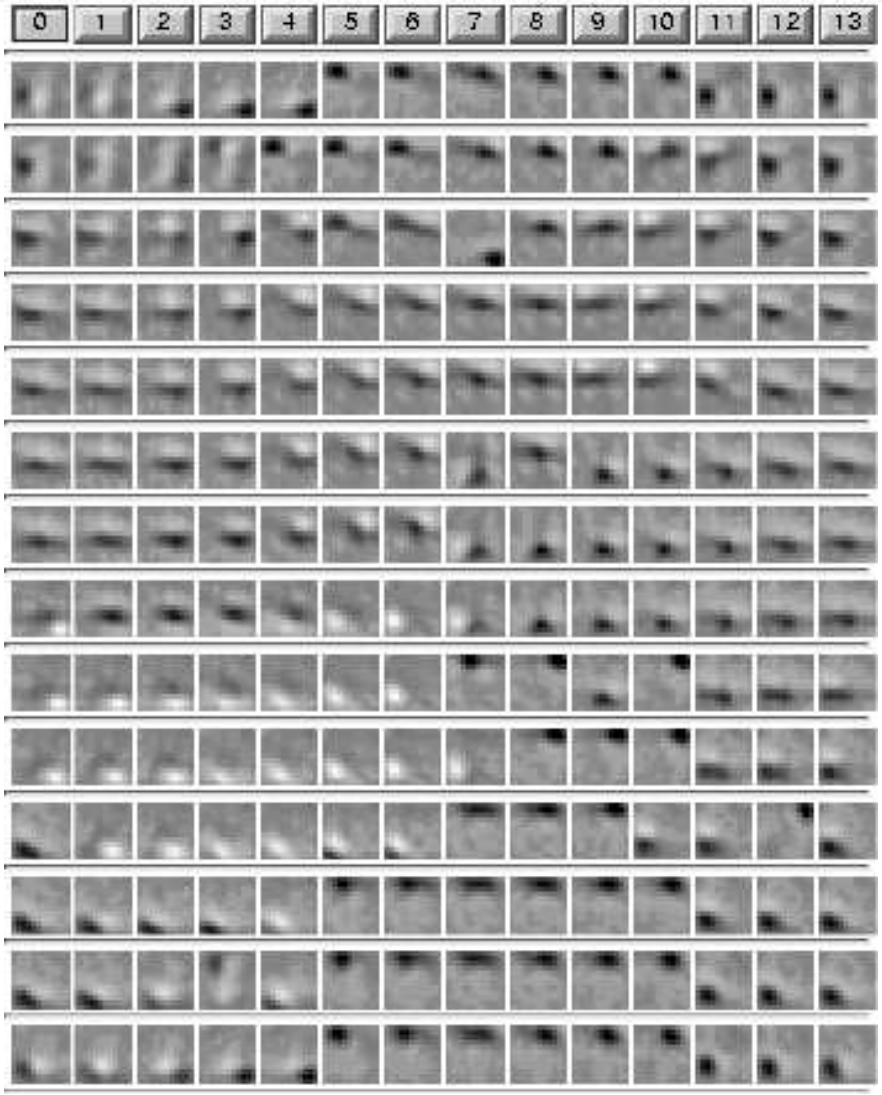
Example: Say you want to extract general principles of social interactions. These vary according to contexts! Need to separately consider a party vs. class vs. job vs. home, etc. Otherwise everything will wash out.

Conditional PCA (CPCA)

Our approach: *Conditionalize* the PCA computation to only apply to a subset of inputs.

Social interactions example: compute different PCA's only on separate contexts (one for party, one for class, etc.) Then, you can extract something meaninful (be "rowdy" and "cool" at party, but "respectful" and "intelligent" in class, etc.)

Conditional PCA (CPCA) of Natural Visual Scenes



From simulation of a model in O'Reilly & Munakata, chapter 8.
Represents a large, heterogenous collection of visual features.

CPCA Equations

Weight = probability that sender x_i is active given receiver y_j is:

$$\begin{aligned} w_{ij} &= P(x_i = 1 | y_j = 1) \\ &= P(x_i | y_j) \end{aligned} \tag{10}$$

Achieved by learning rule (wt moves toward x_i, conditional on y_j):

$$\begin{aligned} \Delta w_{ij} &= \epsilon(y_j x_i - y_j w_{ij}) \\ &= \epsilon y_j (x_i - w_{ij}) \end{aligned} \tag{11}$$

(See O'Reilly & Munakata, 4.5.2, for derivation)

Compare to Oja's rule:

$$\Delta w_{ij} = \epsilon(x_i y_j - y_j^2 w_{ij}) \tag{12}$$

Compare with *generative models*: what is the probability of the input data given an internal model of the world?

Biological Implementation

LTP when both y_j and x_i large: lots of Ca^{++}

LTD when y_j large but x_i small: some Ca^{++}.

Nothing when y_j small.

Weight value *bounded* in 0-1 range, approaches extremes "softly".

Corrections

- CPCA weights don't have much *dynamic range* or *selectivity*. See exploration 4.6
- Solution: *renormalizing* weights and *contrast enhancement*.
- Quantitative adjustments – retain qualitative features of CPCA motivated by biology.

Renormalization

Problem: with sparse uncorrelated inputs, tend to get low weights. Goal: for uncorrelated inputs, a weight of .5, even with sparse activity:

Rewritten form of weight update:

$$\begin{aligned} \Delta w_{ij} &= \epsilon[y_j x_i - y_j w_{ij}] \\ &= \epsilon[y_j x_i (1 - w_{ij}) + y_j (1 - x_i)(0 - w_{ij})] \end{aligned} \tag{13}$$

Then set maximum to m instead of 1:

$$\Delta w_{ij} = \epsilon[y_j x_i (m - w_{ij}) + y_j (1 - x_i)(0 - w_{ij})] \tag{14}$$

Set m to compensate for sparse activity α

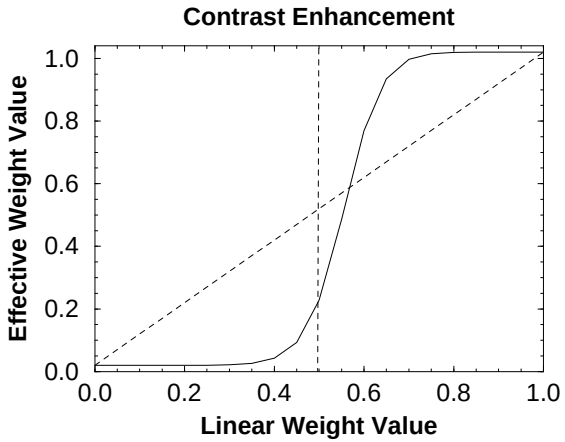
$$m = \frac{.5}{\alpha} \tag{15}$$

Introduce a parameter q_m (`savg_cor`) to adjust:

$$\alpha_m = .5 - q_m(.5 - \alpha) \tag{16}$$

Contrast Enhancement

Enhance contrast between strongest and weaker correlations using a sigmoidal function:



Contrast enhancement via *gain* γ (**wt_gain**):

$$\hat{w}_{ij} = \frac{1}{1 + \left(\frac{w_{ij}}{1 - w_{ij}}\right)^{-\gamma}} \tag{17}$$

With offset (movable threshold) θ (**wt_off**):

$$\hat{w}_{ij} = \frac{1}{1 + \left(\theta \frac{w_{ij}}{1 - w_{ij}}\right)^{-\gamma}} \tag{18}$$

Self-Organized Learning

Hebbian learning with multiple receiving units competing (kWTA)!

Competition means each unit active for only a *subset* of inputs.

Subset will be those inputs that best fit that unit's weights.

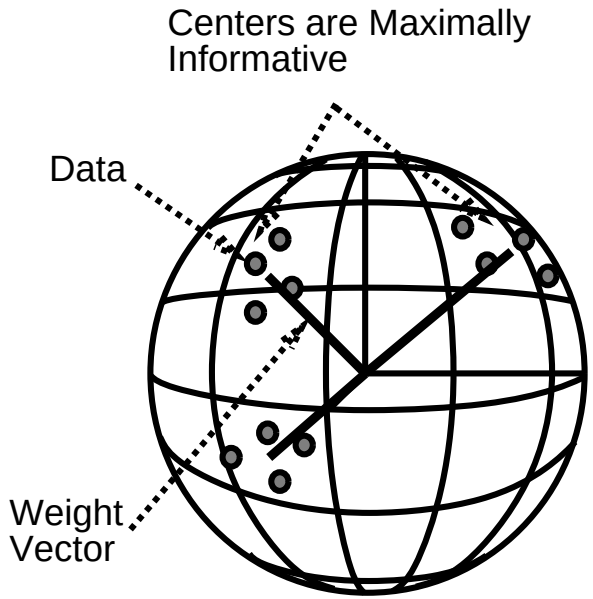
CPCA will then make those weights fit *even better*.

(Positive feedback loop)

Evolution: survival & adaptation of the fittest.

Like classical competitive learning, but using sparse, distributed output instead of single units.

Related Interpretations



- Competitive Learning.
- Kohonen Networks.
- Clustering.
- Information Maximization (Linsker).
- MDL Tradeoff.