Local Synaptic Learning Rules Suffice to Maximize Mutual Information in a Linear Network

as presented in the similarly called
paper by Ralph Linsker

Vesa Siivola, October 5, 2000

Motivation

What ?

- A system that maximizes mutual information between the input and output.
- Based on Hebbian learning
- Uses only local connections

Why ?

- Feature extraction
- Model for biological neural networks

More details

- Stochastic gradient ascent on mutual information
- The distribution of input signal and input noise are assumed Gaussian
- Linear mapping
- Hebbian learning phase and anti-Hebbian unlearning phase

Optimization Problem

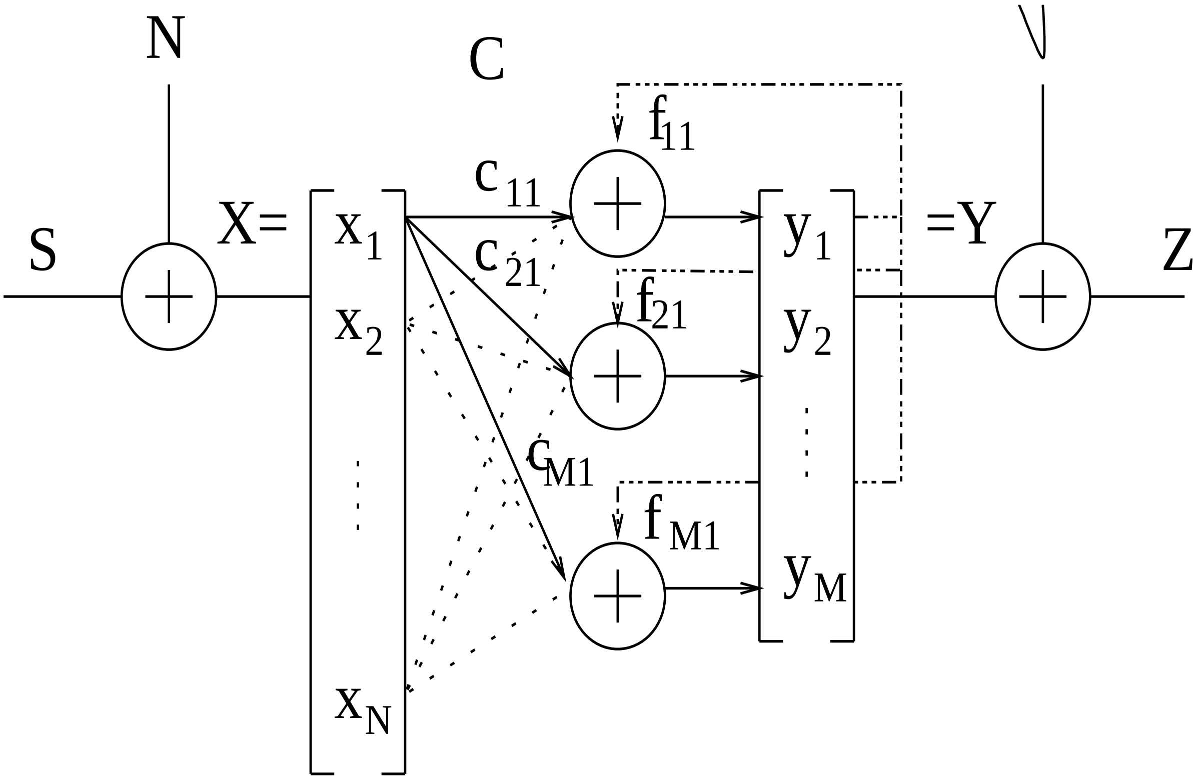


Figure 1: S=signal, N=Input noise, $X = N+S$, C=weights, $Y = CX = C(X+N)$, ν=Output noise, $Z = Y + \nu$, F=feedback

Find a mapping matrix C, which maximizes the ensemble averaged mutual information $R(Z,S)$ between inputs S and outputs Z.

Math: The beginning

Entropy of Gaussian source

$$H = \frac{1}{2} \ln \det \mathrm{cov} X + \text{constant} \tag{1}$$

Ensemble averaged MI

$$R(Z,S) = H(Z) - H(Z|S) \tag{2}$$

$$= \frac{1}{2} \ln \frac{\det \left(C\langle XX^T \rangle C^T + \langle \nu\nu^T \rangle \right)}{\det \left(C\langle NN^T \rangle C^T + \langle \nu\nu^T \rangle \right)}$$

Gradient for numerator in eq. 2, similarly for the denominator

$$\frac{\partial H}{\partial c_{ni}} = \Big(\frac{C\langle XX^T \rangle}{C\langle XX^T \rangle + \langle \nu\nu^T \rangle} \Big)_{ni} = (Q^{-1} C \langle XX^T \rangle)_{ni} \tag{3}$$

If there was no denominator Q in eq. 3

$$\Delta c_{ni} = \gamma (C\langle XX^T \rangle)_{ni} = \gamma \langle y_n x_i \rangle \tag{4}$$

... which is a local update rule.

Math: The inversion

Add connections f_{nm} between outputs and inputs of all Ys. Choose feedback matrix

$$F = I - \alpha Q = I - \alpha(C\langle XX^T\rangle C^T + \langle \nu\nu^T\rangle) \quad (5)$$

$$= 1 - \alpha(\langle YY^T\rangle + \langle \nu\nu^T\rangle)$$

Iterate

$$\mu(0) = Y; \mu(t+1) = Y + F\mu(t) \qquad (6)$$

If $\mu(t)$ converges, $\mu(\infty) = Q^{-1}Y$. The update rule

$$\Delta c_{ni} = \gamma\alpha\langle \mu_n(\infty)x_i\rangle \qquad (7)$$

... is again a local update rule.

Estimate Q as running average over recent input X

$$\Delta\tilde{q}_{nm} = \frac{1}{M}(y_n y_m + \nu_n\nu_m - \tilde{q}_{nm}) \qquad (8)$$

Math: Putting it together

Algorithm "A",

- Select X, compute Y and Z
- Update feedback connections F
- Iterate $\mu(\infty)$, stop when converged
- update feedforward connections C

Batch algorithm:

$$\Delta c_{ni} = \gamma \frac{\partial R(Z,S)}{\partial c_{ni}} \tag{9}$$

$$= \gamma[\alpha_L \langle \mu_n(\infty; L) x_i \rangle_L - \gamma[\alpha_U \langle \mu_n(\infty; L) x_i \rangle_U$$

Iterate algorithm "A" alternatively over learning set and unlearning set.

Math: Additional considerations

Limit for the absolute value of c_{ni}

Maximizing the R(Z,S) will lead to elements $|c_{ni}|$ growing without bounds. Need constraints. Instead maximize

$$R(Z,S) - \sum_n (1 - (C\langle XX^T\rangle C^T)_{nn})^2 \tag{10}$$

where $(C\langle XX^T\rangle C^T)_{nn}$ is the output variance at node n during learning.

Constrain the number of discriminable values at output

Realistic nonlinear processing elements have limited dynamic range, which limits the number of discriminable output values in the presence of output noise.

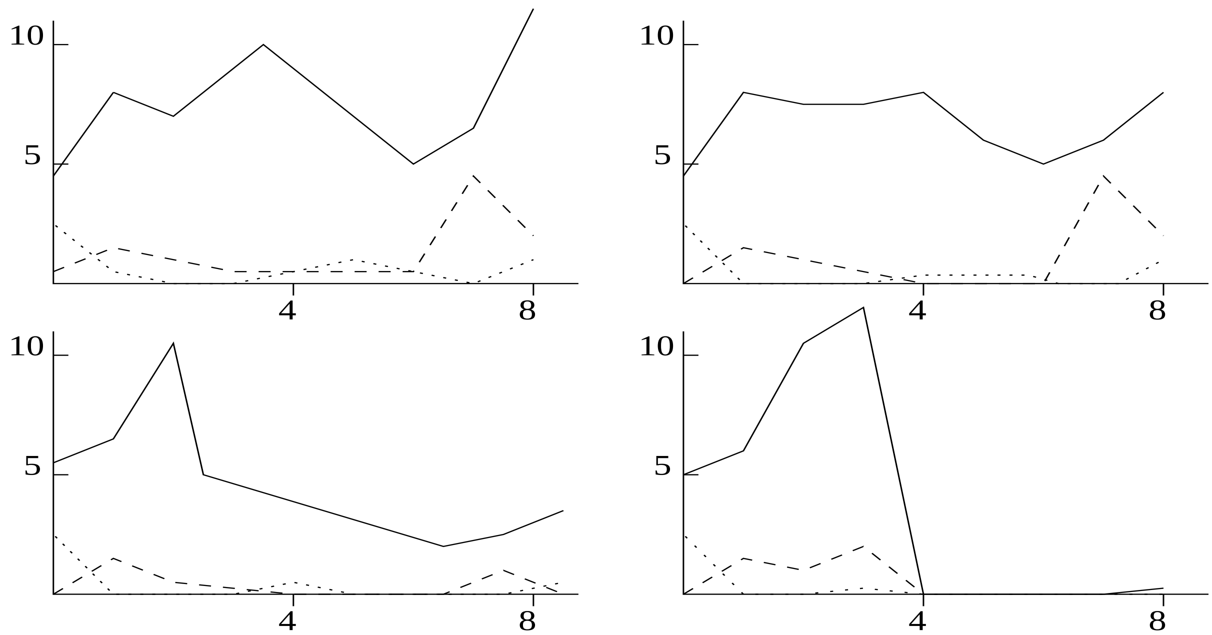
Multiply output noise by $\sqrt{(C\langle XX^T\rangle C^T)_{nn}}$, so that increasing output variance does not lead to increase of discriminable values. Perform gradient ascent on this augmented model.

Analysis

- When input noise covariance $\langle NN^T \rangle = \eta I$ and output noise variance 0, the output nodes develop to span a PCA subspace of input space
- MI maximization leads to
  - maximizing output variance that reflects variance in the input signal as opposed to the input noise.
  - Removing correlations among output signals y_x when output noise variance is small
  - introduce redundancy, when output noise variance large
- If input and/or noise not Gaussian, also strikes balance between the properties of previous item, but does not maximize MI

Example

Signal vector S is a set of values at D uniformly spaced sites on a one-dimensional "retina", which has it's ends connected (is periodical). Signal S is white noise convolved with a Gaussian filter. Input noise N is white. There are 16 input nodes and 8 output nodes.



Shown are the Fourier transformed weights $\hat{C}_{nk}$ for output nodes n, input sites i and spatial frequencies k. $|\hat{C}_{nk}|^2$ for two nodes (dotted and dashed nodes) and the sum of all components $\sum_n |\hat{C}_{nk}|^2$. First picture is initialization, last convergence, two others in between. Corresponding MIs 2.00, 2.33, 2.87, 3.04.