Submitteed to the *International Conference on Independent Component Analysis and Blind Signal Separation (ICA2001)*

ADAPTIVE LATERAL INHIBITION FOR NON-NEGATIVE ICA

Mark Plumbley

Audio & Music Lab
Department of Electronic Engineering
King's College London, Strand, London WC2R 2LS
mark.plumbley@kcl.ac.uk

ABSTRACT

We consider the problem of decomposing an observed input matrix (or vector sequence) $\mathbf{X}$ into the product of a mixing matrix $\mathbf{W}$ with a component matrix $\mathbf{Y}$, i.e. $\mathbf{X} = \mathbf{WY}$, where (a) the elements of the mixing matrix and the component matrix are non-negative, and (b) the underlying components are considered to be observations from an independent source. This is therefore a problem of non-negative independent component analysis. Under certain reasonable conditions, it appears to be sufficient simply to ensure that the output matrix has diagonal covariance (in addition to the non-negativity constraints) to find the independent basis. Neither higher-order statistics nor temporal correlations are required. The solution is implemented as a neural network with error-correcting forward/backward weights and linear anti-Hebbian lateral inhibition, and is demonstrated on small artificial data sets including a linear version of the Bars problem.

1. INTRODUCTION

In many real-world applications, we would like to find a set of underlying causes (components, factors) which combine to produce data that we observe.

A particular problem of interest to us is that of musical signal analysis, and in particular automatic music transcription. Here, the sound that we hear is composed of a number of notes and/or instruments being played at the same time, and we would like to decompose this into the note characteristics, and when each note is being played. In previous work we have approached this problem in the frequency spectrum domain, using Saund's [1] Multiple-Cause Model [2] and sparse coding [3].

In its simplest form, this type of problem, in common with many others, can be considered to be a non-negative factor analysis problem. Each note has a positive (or zero) volume, and the playing of a given note contributes (approximately) a positive amount of power to a given frequency band. [1]

In this paper we investigate an approach to the construction of a multi-cause model with both non-negativity constraints, and a requirement for the underlying causes to be independent: a type of non-negative independent component analysis.

2. NON-NEGATIVITY CONSTRAINTS

Several authors have investigated the construction of linear generative models with non-negative (or rectification) constraints.

Harpur [4] considered non-negative constraints for his recurrent error correction (REC) network, and gave an interesting illustration of this as an underdetermined problem. Charles and Fyfe [5] used a negative feedback network (similar to Harpur network) with rectified outputs and/or weights. They experimented with several types of rectification constraint, finding that on the well-known bars problem, an exponential nonlinearity on the output performed better than simple semilinear non-negative constraint or a sigmoid, suggesting that this encourages a *sparse* representation [6].

Hoyer and Hyvärinen [7] explicitly combined a non-negative constraint with a sparseness property, requiring the sources to have probability densities highly peaked at zero and with heavy tails. When provided with 'complex-cell' responses to natural images patches, their network learned basis functions that coded for contours composed of several aligned complex cells.

Lee and Seung [8, 9] discussed this as a problem of *conic coding* or *non-negative matrix factorization*. They applied this to handwritten digit recognition, showing that an image is naturally decomposed into recognizable constituent parts. Positivity constraints can also play a crucial role in the operation of other networks, such as Spratling's

[1] A well-known exception to this is anti-noise systems, which deliberately cancel unwanted sound by addition of a matching sound 180 degrees out of phase.

pre-synaptic lateral inhibition network, which has also been applied to the Bars problem [10].

3. PROBLEM STATEMENT

Suppose we are presented with an $n \times p$ data matrix $\mathbf{X}$. We often consider this to be a sequence of p input vectors $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p)$ where each vector $\mathbf{x}_k$ contains n simultaneous observations.

We wish to decompose $\mathbf{X}$ as

$$\mathbf{X} = \mathbf{W}\mathbf{Y} \tag{1}$$

where the $m \times p$ matrix $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_p)$ can be considered to be a sequence of p output vectors, and $\mathbf{W}$ is an $n \times m$ generative weight matrix, that *generates* each input vector $\mathbf{x}_k$ as

$$\mathbf{x}_k = \mathbf{W}\mathbf{y}_k \tag{2}$$

from its corresponding output vector $\mathbf{y}_k$, hence this is a *generative model*. If we are given less than n output components, specifically if $m < \text{rank}(\mathbf{X})$, an exact solution may not be possible, and we typically look for a solution that minimises the mean squared error between $\mathbf{X}$ and its reconstruction $\mathbf{W}\mathbf{Y}$. Without any further constraints, this is clearly an underdetermined problem, since any transformation $\mathbf{W} \leftarrow \mathbf{W}\mathbf{B}$, $\mathbf{Y} \leftarrow \mathbf{B}^{-1}\mathbf{Y}$ for some invertible matrix $\mathbf{B}$ will also solve this problem.

We are interested in the case of *non-negative* outputs and weights, i.e. $w_{ij} \geq 0$ and $y_{jk} \geq 0$. (Note that this implies that all inputs x_{ik} must also be non-negative.) However, even this non-negativity constraint may not be sufficient to completely determine the solution [4].

Let us suppose that our vectors $\mathbf{x}_k$ were, in fact, a finite number of samples observed from some random vector $\mathcal{X} = [\mathcal{X}_1, \ldots, \mathcal{X}_n]^T$, such that

$$\mathcal{X} = \mathbf{A}\mathcal{S} \tag{3}$$

where $\mathcal{S} = [\mathcal{S}_1, \ldots, \mathcal{S}_n]^T$ is a random vector of sources and the random variables $\mathcal{S}_j$ are *independent* from each other. This is then the standard independent component analysis (ICA) problem [11, 12] but with additional non-negativity constraints on the weights and components.

4. INDEPENDENCE THROUGH ADAPTIVE LATERAL INHIBITION

In this paper, we would like to investigate the use of a neural network with lateral inhibition to perform this non-negative ICA.

In a linear network, it is well known that adaptive lateral inhibition can be used to *decorrelate* the outputs from a network. For example, Barlow and Földiák [13] proposed a network (fig. 1(a)) with output defined by

$$d\mathbf{y}/d\tau = (\mathbf{x} - \mathbf{V}\mathbf{y}(\tau)) - \mathbf{y}(\tau) = \mathbf{x} - (\mathbf{I} + \mathbf{V})\mathbf{y}(\tau) \tag{4}$$

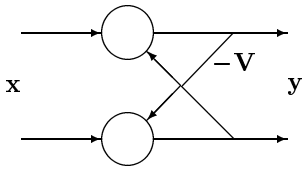
with a symmetric lateral inhibition matrix $\mathbf{V}$, leading to

$$\mathbf{y} = (\mathbf{I} + \mathbf{V})^{-1}\mathbf{x} \tag{5}$$

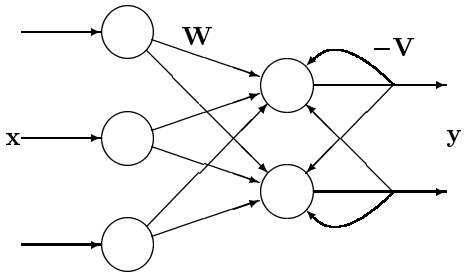
when the outputs have settled (considered to be over a short timescale), and an anti-Hebbian learning rule

$$\Delta\mathbf{V} = \eta_v \cdot \text{offdiag}(\mathbf{y}\mathbf{y}^T) \tag{6}$$

which converges to a $\mathbf{V}$ such that the outputs are decorrelated, i.e. that $E(\mathbf{y}\mathbf{y}^T)$ is diagonal.



(a) Barlow and Földiák's [13] decorrelating network.



(b) Decorrelating subspace network

Fig. 1. Anti-Hebbian decorrelating networks.

These and other anti-Hebbian networks have been combined with Hebbian feed-forward networks (fig. 1(b)) to discover uncorrelated, sparse, and/or information-efficient representations of a subspace of the input [14, 15].

Now it is well known that (for a linear network with zero-mean inputs at least) this linear anti-Hebbian network will only find *decorrelated* rather than *independent* outputs (see e.g. [12]). The early Jutten and Herault ICA (or 'INCA') network [16] used this type of network with a *nonlinear* anti-Hebbian learning algorithm to overcome this limitation and perform true independent component analysis

If we find a combination of non-negative components which have zero *covariance*, i.e. $E(\tilde{\mathbf{y}}\tilde{\mathbf{y}}^T)$ is diagonal, where $\tilde{\mathbf{y}} = \mathbf{y} - \bar{\mathbf{y}}$ and $\bar{\mathbf{y}}$ is the mean of $\mathbf{y}$, then this is a necessary (but still not sufficient) condition for independence. In the unconstrained linear case, we could rotate the basis vectors within the same subspace, and we could still produce a decorrelated output.

However, together with non-negativity constraints, this zero covariance condition appears to be is sufficient to fix the remaining underdetermined parameters of the solution (apart from the usual scaling and permutation) provided that the sources have a non-zero likelihood of activation in the region close to zero. Sparse sources, which have high likelihood of activation near zero, would be particularly helpful here, although sources y_i with a pdf where $P(0 \leq y_i < c) = 0$ for some c is likely to leave some scope for 'slack' in the rotation of the solution basis. Here we demonstrate this with some simulations, although a rigorous proof remains to be developed.

5. ALGORITHMS AND SIMULATION RESULTS

5.1. Initial iterative algorithm

Most of the algorithms we describe are based around a network similar to Harpur's Recurrent Error Correction network [4] with non-negative weights and outputs, combined with a covariance-removing (rather than decorrelating) anti-Hebbian recurrent inhibition layer (fig. 1).

Consider first the non-negative output activations. We have

$$d\mathbf{y}(\tau)/d\tau = \mathbf{W}^T\mathbf{x} - (\mathbf{I} + \mathbf{V})\mathbf{y}(\tau) \tag{7}$$

with constraints

$$y_j \geq 0 \qquad 1 \leq j \leq m \tag{8}$$

which could be solved directly by iterating eqn (7), but (as pointed out by Lee and Seung [8]) there are polynomial time solutions for this type of problem. Our particular method was modified from the Matlab non-negative least squares (NNSL) routine. (Note that our neural approach is not the same as the NNLS solution for $\mathbf{y}$ to $\mathbf{W}^T\mathbf{x} = (\mathbf{I} + \mathbf{V})\mathbf{y}$).

We then update $\mathbf{W}$ and $\mathbf{V}$ according to

$$\Delta\mathbf{W} = \eta_w(\mathbf{x} - \mathbf{r})\mathbf{y}^T \tag{9}$$

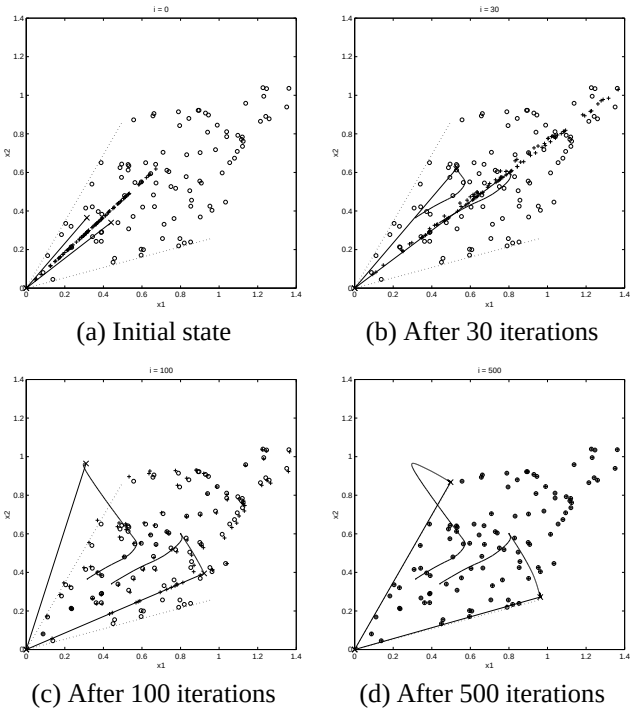
$$\Delta\mathbf{V} = \eta_v(\tilde{\mathbf{y}}\tilde{\mathbf{y}}^T) \tag{10}$$

where $\mathbf{r} = \mathbf{W}\mathbf{y}$ is a reconstruction for $\mathbf{x}$, and $\tilde{\mathbf{y}} = \mathbf{y} - \bar{\mathbf{y}}$. We used batch updating in our simulations, with random initial values for $\mathbf{W}$ and $\mathbf{V}$ initilialized to the identity, and some adjustments to update rate schedules to speed up learning. The results on a small 2-D dataset show the operation of the algorithm (figs. 2, 3).

5.2. Faster algorithms

While the iterative algorithm (9)–(10) is simple and direct, there are modifications that can be made that directly attempt to force the outputs to be decorrelated.

In fact, it turns out to be easier to specify that the output should have a specific covariance matrix, e.g. $\mathbf{C_y} = E(\tilde{\mathbf{y}}\tilde{\mathbf{y}}^T) = \alpha\mathbf{I}$ rather than just forcing the off-diagonal entries to be zero.



(a) Initial state (b) After 30 iterations

(c) After 100 iterations (d) After 500 iterations

Fig. 2. Simulation of algorithm (9)–(10) on a 2-D artificial data set generated as uniformly distributed amounts of positive basis vectors (dotted). The plots show the original generated data ('o'), weight vectors $\mathbf{w}_1, \mathbf{w}_2$ ('x' with solid line from zero), and the reconstructed data points ('+'). After initial overall scaling (b), the outputs covariance is removed (c), then the basis set rotated to include those data points on the 'outside' of the hull formed by the weight vectors (d).

Starting from $\mathbf{C_y}$ we linearly remove the effect of the current symmetrical inhibition matrix $\mathbf{V}$ to calculate an effective pre-inhibition covariance of

$$\mathbf{C}^{\text{eff}}_{\mathbf{wx}} = (\mathbf{I} + \mathbf{V})\mathbf{C_y}(\mathbf{I} + \mathbf{V}) \tag{11}$$

which we wish to be diagonalized (set to $\alpha\mathbf{I}$) by our new inhibition matrix, i.e.

$$\alpha\mathbf{I} = (\mathbf{I} + \mathbf{V}^{(1)})^{-1}\mathbf{C}^{\text{eff}}_{\mathbf{wx}}(\mathbf{I} + \mathbf{V}^{(1)})^{-1} \tag{12}$$

giving

$$\mathbf{V}^{(1)} = (1/\alpha)(\mathbf{C}^{\text{eff}}_{\mathbf{wx}})^{1/2} - \mathbf{I} \tag{13}$$

where $(\mathbf{C}^{\text{eff}}_{\mathbf{wx}})^{1/2}$ is chosen to be real and symmetric. We actually only attempt to *reduce* eigenvalues of $\mathbf{C_y}$ (compared with those of $\mathbf{C}^{\text{eff}}_{\mathbf{wx}}$) to α rather than increasing them, to avoid $\mathbf{V}$ becoming singular in the process (see also [15] for an iterative neural algorithm for this type of problem). Note

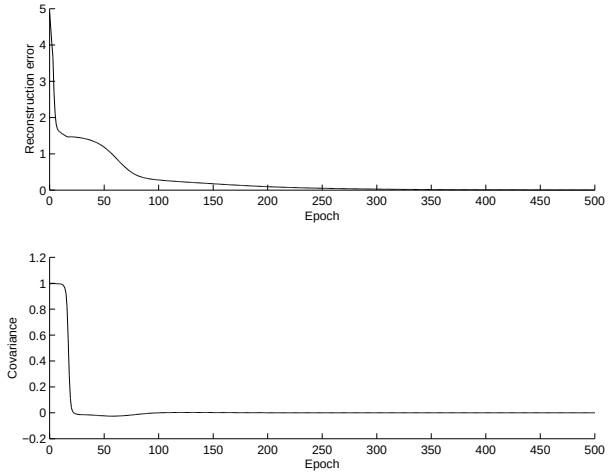


Fig. 3. Learning curves corresponding to fig. 2 showing reduction in sum-squared reconstruction error (upper curve) and normalised covariance between the two outputs (lower curve).

that there is some danger of overshoot with this method, since the calculation relies on the output $\mathbf{y}$ only, rather than the original data, so the new $\mathbf{V}$ may attempt to push data points on the 'edges' beyond their original positions once the new boundary points are calculated on the following iteration. In comparison with the simple iterative algorithm, some speeding up can be observed (fig. 4).

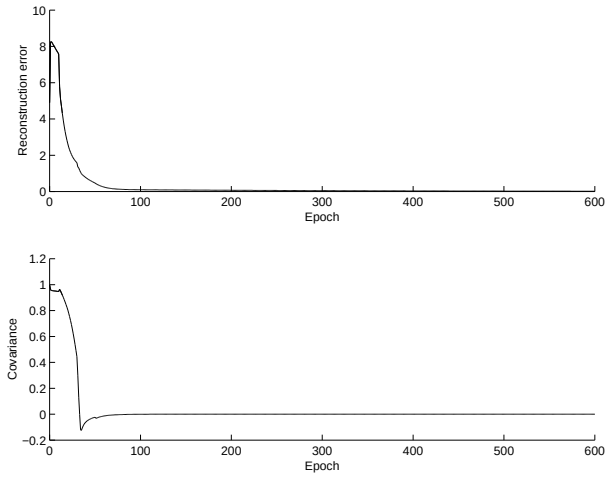


Fig. 4. Speed up in learning due to direct calculation of $\mathbf{V}$ in eqn (13), in comparison to fig. 3(e).

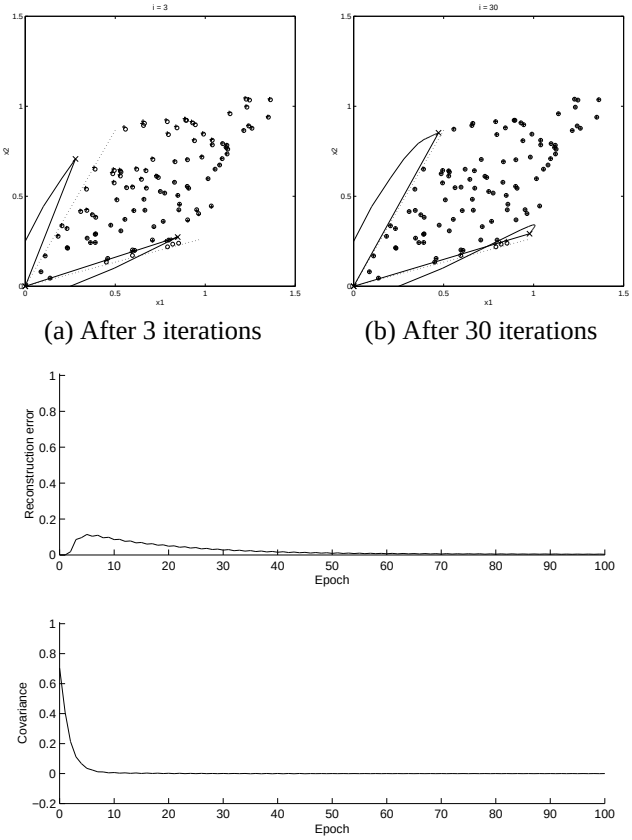
For this simple problem, additional speed-up is possible by directly calculating a new weight matrix $\mathbf{W}$. Following the calculation of $\mathbf{V}$ in (13) and calculation of the new output matrix $\mathbf{Y}$, we find a new weight matrix $\mathbf{W}$ that minimises the least mean square reconstruction error of $\mathbf{X}$ from this $\mathbf{Y}$. Specifically we use the Matlab matrix left division

$$\mathbf{W} = (\mathbf{Y}^T \backslash \mathbf{X}^T)^T \tag{14}$$

and then apply the positivity constraint to $\mathbf{W}$.

For problems such as this with equal inputs and outputs, it can help to set the initial weight matrix to the identity matrix $\mathbf{I}$, since then initially all data points are in the linear region. (This approach would not be a good idea for problems that require symmetry breaking).

We found that this combination gives considerable speed-up, with good values for reconstruction and covariance in just a few iterations (fig. 5). However, the combination of approximations appear to have a tendency to make this particular approach unstable on more difficult problems.



(a) After 3 iterations (b) After 30 iterations

(c) Learning curves

Fig. 5. Fast convergence of algorithm (13)–(14). Note that the diagonal initial weight vectors permit perfect initial reconstruction, and only small reconstruction error is introduced as a few data points fall outside of the representation region between the weight vectors.

5.3. Linear bars problem

We also illustrate the operation of this algorithm on a linear version of the well-known Bars dataset introduced by [14].

For this problem, we show results with $n = 8 \times 8$ dimensional input image formed from linear addition of bars (1 if present, 0 otherwise), with each of the 16 possible bars (8 horizontal and 8 vertical) present with probability P. In this illustration we used $P = 0.2$ with $p = 100$ patterns, with $\mathbf{W}$ initialized to uniform random numbers in the range $[0, 1]$.

We found that the direct approach (13)–(14) was unstable on this problem, so used the iterative algorithm (9) for $\mathbf{W}$ with the direct calculation (13) for $\mathbf{V}$. The results are shown in figs. 6, 7. This particular example is interesting since it shows two potential bars being 'forced apart' by the covariance reduction requirement. Although the final reconstruction error and distance from target covariance are both non zero, the bars have clearly been extracted.

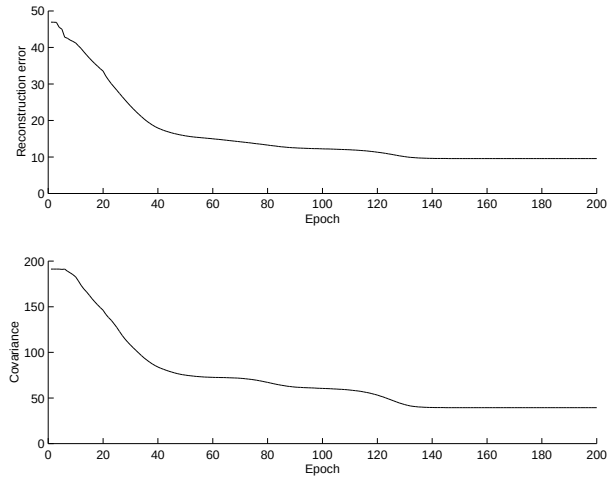


Fig. 6. Learning curves for linear bars. The 'Covariance' measure is the Frobenius norm $\|\mathbf{C_y} - \alpha\mathbf{I}\|_{\mathrm{F}}$ measuring how much $\mathbf{C_y}$ is away from the target $\alpha\mathbf{I}$.

6. CONCLUSIONS

In this paper we considered the task of finding non-negative independent components $\mathbf{S}$ together with non-negative linear weights $\mathbf{A}$ to give an observation matrix $\mathbf{X} = \mathbf{AS}$.

Provided that the non-negative sources have non-zero pdf around zero, to identify this mixture it appears that it is sufficient to identify non-negative weights $\mathbf{W}$ and non-negative 'outputs' $\mathbf{Y}$ such that the output covariance matrix $\mathbf{C_y}$ is diagonal, and that no higher-order statistics or temporal correlations are required. This is likely to be the case for sparse components, since they have a large probability density around zero.

We developed a neural network approach based on this concept, using a network with symmetrical forward and backward weights with an error correction algorithm, but with covariance reduced by a linear anti-Hebbian stage.

We demonstrated the operation of this network on small artificial learning tasks, specifically a 2-D visualization problem, and a linear version of the well-known Bars problem. These can be learned successfully using one of a number of algorithm variants.

There are a number of remaining problems to be tackled. A rigorous proof that non-negativity and covariance removal is sufficient, and under what conditions, is needed. Some algorithm variants make assumptions that may not hold for all problems, and so may not be stable in all cases.

In addition, the polynomial algorithm we are currently using to find the final non-negative outputs does not scale well as the output dimension m increases, since it operates by adding potential non-negative outputs to a 'positive' set one at a time. We are currently investigating alternative solutions which will permit the application of this method to larger, real-world problems, such as our music transcription problem, with reasonable learning times.

7. ACKNOWLEDGEMENTS

The author would like to thank Samer Abdallah for many interesting discussions on this and related topics.

8. REFERENCES

[1] E. Saund, "A multiple cause mixture model for unsupervised learning," *Neural Computation*, vol. 7, pp. 51–71, 1995.

[2] J. Klingseisen and M. D. Plumbley, "Towards musical instrument separation using multiple-cause neural networks," in *Proceedings of the International Workshop on Independent Component Analysis And Blind Signal Separation, 19-22 June 2000, Helsinki, Finland*, 2000, pp. 447–452.

[3] S. A. Abdallah and M. D. Plumbley, "Sparse coding of music signals," Submitted to *Neural Computation*, 2001.

[4] G. F. Harpur, *Low Entropy Coding with Unsupervised Neural Networks*, Ph.D. thesis, Department of Engineering, University of Cambridge, February 1997.

[5] D. Charles and C. Fyfe, "Modelling multiple-cause structure using rectification constraints," *Network: Computation in Neural Systems*, vol. 9, pp. 167–182, 1998.

[6] D. J. Field, "What is the goal of sensory coding?," *Neural Computation*, vol. 6, pp. 559–601, 1994.

[7] P. O. Hoyer and A. Hyvärinen, "A non-negative sparse coding network learns contour coding and integration from natural images," Preprint, 2001.

[8] D. D. Lee and H. S. Seung, "Unsupervised learning by convex and conic coding," in *Advances in Neural Information Processing Systems*, Cambridge, MA, 1997, vol. 9, pp. 515–521, MIT Press.

[9] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems 13*, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds., 2001.

[10] M. W. Spratling, "Pre-synaptic lateral inhibition provides a better architecture for self-organizing neural networks," *Network: Computation in Neural Systems*, vol. 10, pp. 285–301, 1999.

[11] P. Comon, "Independent component analysis - a new concept?," *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994.

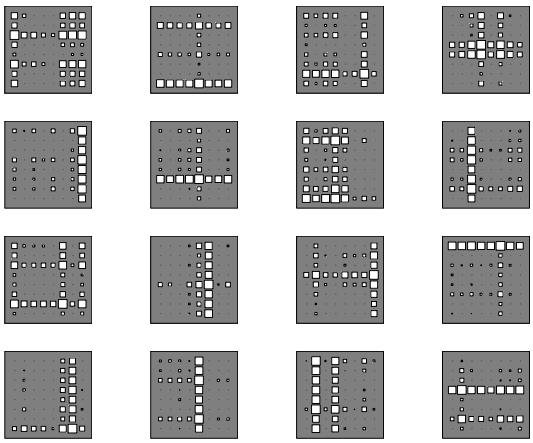
[12] A. J. Bell and T. J. Sejnowski, "An information maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, pp. 1129–1159, 1995.

[13] H. B. Barlow and P. Földiák, "Adaptation and decorrelation in the cortex," in *The Computing Neuron*, Richard Durbin, Christopher Miall, and Graeme Mitchison, Eds., pp. 54–72. Addison-Wesley, Wokingham, England, 1989.

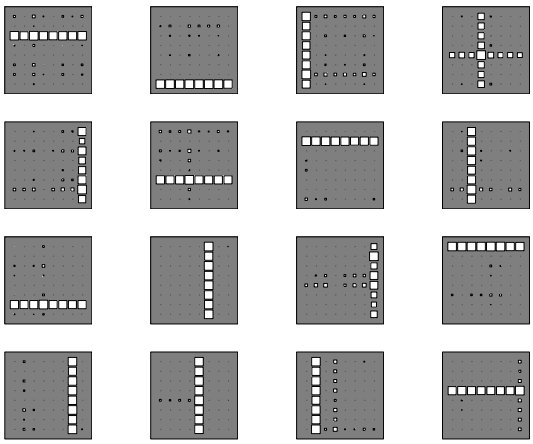
[14] P. Földiák, "Forming sparse representations by local anti-Hebbian learning," *Biological Cybernetics*, vol. 64, pp. 165–170, 1990.

[15] M. D. Plumbley, "A Hebbian/anti-Hebbian network which optimizes information capacity by orthonormalizing the principal subspace," in *Proceedings of the IEE Artificial Neural Networks Conference, ANN-93, Brighton, UK*, 1993, pp. 86–90.

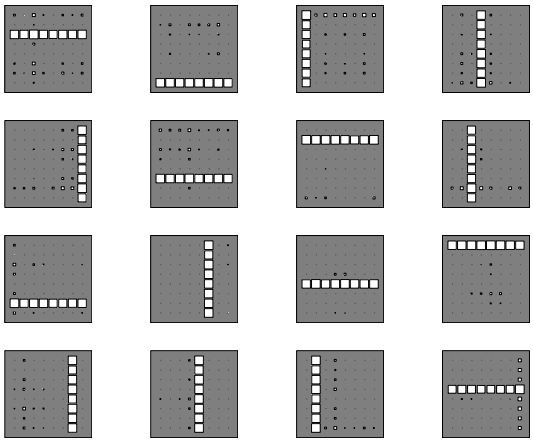
[16] C. Jutten and J. Herault, "Blind separation of sources, Part I: An adaptive algorithm based on neuromimetic architecture," *Signal Processing*, vol. 24, pp. 1–10, 1991.



(a) After 30 iterations



(a) After 100 iterations



(a) After 150 iterations

Fig. 7. Learning the linear bars problem. Note that the decorrelation condition has forced two of the bars apart (positions (1,3) and (3,2)) between 150 and 100 iterations: a corresponding reduction in the covariance measure can be seen in fig. 6.