Gatsby Computational Neuroscience Unit
University College London
http://www.gatsby.ucl.ac.uk

17 Queen Square, London
WC1N 3AR, United Kingdom
+44 20 7679 1176

Funded in part by the Gatsby Charitable Foundation.

September 27, 1999

GCNU TR 1999–002

Linear Heteroencoders

Sam Roweis
Gatsby Unit

Carlos Brody
Computation and Neural Systems
California Institute of Technology

Abstract

This note gives a closed form expression for the linear transform computed by an optimally trained linear *hetero*encoder network of arbitrary topology trained to minimize squared error. The transform can be thought of as a restricted rank version of the basic linear least-squares regression (discrete Wiener filter) between input and output. The rank restriction is set by the "bottleneck" size of the network – the minimum number of hidden units in any layer. A special case of this expression is the well known result that linear *auto*encoders with a bottleneck of size r perform a transform equivalent to projecting into the subspace spanned by the first r principal components of the data. This result eliminates the need to explicitly train linear heteroencoder networks.

Linear Heteroencoders

Sam Roweis
Gatsby Unit

Carlos Brody
Computation and Neural Systems
California Institute of Technology

1 Linear autoencoders

Because they are fast to train and require few parameters, linear networks[1] provide an important performance comparison with more complex data analysis methods. The equivalent linear transform computed by a linear *auto*encoder network with a bottleneck of size r has long been known to be the projection into the subspace spanned by the first r principal components of the training data. This is true if the training algorithm minimizes squared error at the output and achieves the global minimum of that error. Furthermore, Bourlard and Kamp (1988) have shown that if all layers of the network *after* the bottleneck are linear the optimal transformation remains unchanged even if nonlinear transfer functions are added to units before the bottleneck. In other words, the result still holds for networks that are merely output-linear.

These results are important because they eliminate the need to explicitly *train* linear or output-linear autoencoders. Algorithms such as the singular-value decomposition can be used to quickly compute the optimal transform. These algorithms run in a known fixed time, results are easily reproducible, they are guaranteed to achieve the global minimum of error, and they produce an ordered set of orthogonal eigenvectors.

In this note, we present the analogous results for linear (and output-linear) *hetero*encoder networks with a bottleneck. Enforcing such a bottleneck may be important in cases where high dimensionality of the input space leads to overtraining and poor generalization. The results below allow the equivalent transform (and thus performance) of an optimally trained network to be easily computed without explicit training.

Many previous authors [Baldi and Hornik, 1989, Kung and Diamantaras, 1991] [Diamantaras and Kung, 1994, Scharf, 1991, Stoica and Viberg, 1996, Ghahramani, 1996] have proved portions of the results we review below. However, the proofs are often part of a more detailed or lengthy discussion and as a result are sometimes mathematically more complex. The goal of this note is to provide a short and simple exposition of these results along with practical expressions for their implementation.

† Based on a May 1997 version. During writing, the authors were supported in part by the Center for Neuromorphic Systems Engineering as a part of the National Science Foundation Engineering Research Center Program under grant EEC-9402726 and by the Natural Sciences and Engineering Research Council of Canada under an NSERC 1967 Award.

[1] We refer to any network in which the transfer function of every unit is linear as a *linear network*. Similarly we use the term *linear layer* for layers in which every unit's transfer function is a linear function. A network has a *bottleneck* of size r if no layer has fewer than r units. A network has *no bottleneck* if all layers have at least as many units as both the input dimension and output dimension. If all layers after the bottleneck layer (or after the last bottleneck if there are several) are linear, we call the network *output-linear*.

2 Linear heteroencoders with no bottleneck

The classic linear least-squares regression result (discrete Wiener filter) gives the optimal linear mapping from a set of n input points $\mathbf{X}$ to corresponding output points $\mathbf{Y}$ in terms of the input auto-correlation and the input-output cross-correlation (we assume the data are zero mean):

$$\mathbf{A}^* = (\mathbf{Y}\mathbf{X}^T)(\mathbf{X}\mathbf{X}^T)^{-1} \tag{1}$$

where $\mathbf{X}$ and $\mathbf{Y}$ are $p \times n$ and $q \times n$ matrices containing the p-dimensional input data and q-dimensional output data respectively. The $q \times p$ matrix $\mathbf{A}^*$ minimizes the total reconstruction error:

$$\text{error} = \|\mathbf{A}\mathbf{X} - \mathbf{Y}\|^2 = \text{Tr}[(\mathbf{A}\mathbf{X} - \mathbf{Y})(\mathbf{A}\mathbf{X} - \mathbf{Y})^T] \tag{2}$$

over all possible matrices $\mathbf{A}$ which estimate $\mathbf{Y}$ as $\mathbf{A}\mathbf{X}$. (Here $\|\mathbf{M}\|^2 = \text{trace}[\mathbf{M}\mathbf{M}^T]$ is the Frobenius norm of $\mathbf{M}$.) It represents the linear transform computed by an optimally trained linear heteroencoder network with no bottleneck.

3 Linear heteroencoders with a bottleneck

What is the result for a linear heteroencoder network with a bottleneck of size r? In other words, what is the optimal linear mapping *of rank no more than* r between a set of n input points $\mathbf{X}$ and corresponding output points $\mathbf{Y}$? As we show in section 5 below, the correct transform involves taking the singular-value decomposition of the matrix $\mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1/2}$ and setting all but the r largest singular-values to zero. (Here $\mathbf{0}^{-1/2}$ denotes the matrix square root of the non-negative definite matrix $\mathbf{0}$.) First we do the singular-value decomposition:

$$\mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1/2} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \tag{3}$$

Here $\mathbf{U}$ and $\mathbf{V}$ are unitary matrices and $\mathbf{\Sigma}$ is a $q \times p$ "diagonal" matrix (in other words, only the $\mathbf{\Sigma}_{ii}$ entries are nonzero) with positive elements. The optimal restricted rank transform $\mathbf{A}_r^*$ is now:

$$\mathbf{A}_r^* = \mathbf{U}\mathbf{\Sigma}_r\mathbf{V}^T(\mathbf{X}\mathbf{X}^T)^{-1/2} \tag{4}$$

where $\mathbf{\Sigma}_r$ is the diagonal matrix obtained by setting to zero all but the r largest elements of $\mathbf{\Sigma}$. The $q \times p$ matrix $\mathbf{A}_r^*$ minimizes the total reconstruction error:

$$\text{error} = \|\mathbf{A}_r\mathbf{X} - \mathbf{Y}\|^2 \tag{5}$$

over all possible matrices $\mathbf{A}_r$ *of rank no more than* r which estimate $\mathbf{Y}$ as $\mathbf{A}_r\mathbf{X}$. It represents the linear transform computed by an optimally trained linear *hetero*encoder network with a bottleneck of size r. Furthermore, if all of the units *after* the bottleneck are linear, the result of Bourlard and Kamp applies and the optimal transformation remains unchanged even if nonlinearities are added to lower layers. This result eliminates the need to explicitly train linear heteroencoder or output-linear heteroencoder networks. Below we provide the MATLAB code to compute the optimal transform `Ar` of rank `r` assuming the p by n matrix `x` holds the input data and the q by n matrix `y` holds the targets.

```
croot = sqrtm(inv(x*x’));
[u,s,v] = svd(y*x’*croot);
for zz=(r+1):min(size(s)) s(zz,zz)=0; end
Ar = u*s*v’*croot;
```

4 Relationship to Canonical Correlations

Canonical Correlation Analysis[2] (see for example [Mardia et al., 1979]) tries to find linear combinations $\mathbf{a}_i^T\mathbf{x}$ and $\mathbf{b}_i^T\mathbf{y}$ of multidimensional variables $\mathbf{x}$ and $\mathbf{y}$ such that the linear combinations are highly correlated. In particular $\mathbf{a}_i$ and $\mathbf{b}_i$ are the i^{th} canonical correlation vectors for $\mathbf{x}$ and $\mathbf{y}$ if they maximize the expected value of $(\mathbf{a}_i^T\mathbf{x})(\mathbf{b}_i^T\mathbf{y}) = \mathbf{a}_i^T\mathbf{x}\mathbf{y}^T\mathbf{b}_i$ subject to the conditions that the expected values of $\mathbf{a}_i^T\mathbf{x}$ and $\mathbf{b}_i^T\mathbf{y}$ both equal unity and that canonical correlation vectors for $i \neq j$ are uncorrelated.

It turns out that the first r canonical correlation vector pairs span exactly the same space as the hidden units in a restricted rank linear heteroencoder with a bottleneck of r. (Of course the canonical vectors are an orthonormal basis for this space while the weight vectors of the hidden units in a linear heteroencoder are in general not orthogonal or unit length.) Thus, linear heteroencoders with a bottleneck perform exactly canonical correlation analysis on the input-output data[3]

5 Derivation of $\mathbf{A}_r^*$

First we recast the problem into an equivalent one in different spaces by whitening and rotating the inputs and by rotating the outputs to decorrelate. Let the new inputs be $\mathbf{W}$ and the new outputs be $\mathbf{Z}$:

$$\mathbf{W} = \mathbf{V}^T(\mathbf{X}\mathbf{X}^T)^{-1/2}\mathbf{X} \qquad \mathbf{Z} = \mathbf{U}^T\mathbf{Y} \tag{6}$$

$$(\mathbf{W}\mathbf{W}^T) = \mathbf{I} \qquad (\mathbf{Z}\mathbf{W}^T) = \mathbf{\Sigma} \tag{7}$$

where $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{\Sigma}$ come from the singular-value decomposition (3) above. If we use a linear transformation $\mathbf{B}$ in the new spaces to estimate $\mathbf{Z}$ by $\mathbf{B}\mathbf{W}$ then the equivalent transform in the original spaces is:

$$\mathbf{A} = \mathbf{U}\mathbf{B}\mathbf{V}^T(\mathbf{X}\mathbf{X}^T)^{-1/2} \tag{8}$$

Notice that since we have applied only a rotation to the outputs we have not affected the squared reconstruction error. The error (2) can thus be written as:

$$\begin{aligned} \text{error} &= \|\mathbf{B}\mathbf{W} - \mathbf{Z}\|^2 & (9)\\ &= \text{Tr}[(\mathbf{B}\mathbf{W} - \mathbf{Z})(\mathbf{B}\mathbf{W} - \mathbf{Z})^T] \\ &= \text{Tr}[\mathbf{B}\mathbf{W}\mathbf{W}^T\mathbf{B}^T - 2\mathbf{Z}\mathbf{W}^T\mathbf{B}^T + \mathbf{Z}\mathbf{Z}^T] & (10) \end{aligned}$$

which by the relations (7) and the fact that $\mathbf{Z}\mathbf{Z}^T$ does not depend on $\mathbf{B}$, can be rewritten as:

$$\begin{aligned} \text{error} &= \text{Tr}[\mathbf{B}\mathbf{B}^T - 2\mathbf{\Sigma}\mathbf{B}^T + \mathbf{\Sigma}\mathbf{\Sigma}^T] + \text{constant} \\ &= \|\mathbf{B} - \mathbf{\Sigma}\|^2 + \text{constant} & (11) \end{aligned}$$

where the constant does not depend on $\mathbf{B}$. If $\mathbf{B}$ is of unrestricted rank, the optimal solution is by $\mathbf{B}^* = \mathbf{\Sigma}$, which is exactly equivalent to the Wiener filter $\mathbf{A}^*$ in the original spaces. If $\mathbf{B}$ is of restricted rank r, the optimal solution is $\mathbf{B}_r^* = \mathbf{\Sigma}_r$, giving the result $\mathbf{A}_r^*$ as in section 3.

[2] Thanks to Zoubin Ghahramani for pointing out the link to canonical correlations.

[3] Canonical correlations is often studied when the error metric on the outputs $\mathbf{y}$ is not Euclidean (as we have assumed here) but rather defined by some covariance matrix $\mathbf{C}_{yy}$. If the sample covariance $\mathbf{Y}\mathbf{Y}^T$ is used to estimate this metric, this makes the problem completely symmetric in $\mathbf{x}$ and $\mathbf{y}$. Linear heteroencoders can also be studied in this way by including the output metric in the error analysis; for non-Euclidean output metrics this makes them again equivalent to canonical correlations.

Sketch of proof:
Let $\mathrm{B}^{(c)}$ be the space spanned by the columns of $\mathbf{B}$. For a given $\mathrm{B}^{(c)}$, the minimum of $\|\mathbf{B}-\mathbf{\Sigma}\|^2$ will be achieved when the i-th column of $\mathbf{B}$ is the projection of the i-th column of $\mathbf{\Sigma}$ onto $\mathrm{B}^{(c)}$, for all columns of $\mathbf{B}$. If $\mathbf{B}$ is of restricted rank r, $\mathrm{B}^{(c)}$ is at most r-dimensional, and should then be chosen to span the r columns of $\mathbf{\Sigma}$ with the largest magnitudes. In this case the optimal solution is $\mathbf{B}_r^* = \mathbf{\Sigma}_r$, giving the result $\mathbf{A}_r^*$ of section 3.
To see the why the above choice of $\mathrm{B}^{(c)}$ is optimal, consider the plane spanned by any two columns of $\mathbf{\Sigma}$ (call the two columns $\mathbf{c}_1$ and $\mathbf{c}_2$, with magnitudes $\sigma_1 > \sigma_2$). Let $\mathrm{B}^{(c)}$ intersect this plane at an angle α with respect to $\mathbf{c}_1$. The error these two columns will contribute is then

$$\sigma_1{}^2 \sin^2\alpha + \sigma_2{}^2 \cos^2\alpha = (\sigma_1{}^2 - \sigma_2{}^2)\sin^2\alpha + \sigma_2{}^2, \tag{12}$$

which is minimized when $\alpha = 0$. In other words, if $\mathrm{B}^{(c)}$ does not span both $\mathbf{c}_1$ and $\mathbf{c}_2$, it should be parallel to $\mathbf{c}_1$, the column with the larger magnitude. Since this holds for any pair of columns of $\mathbf{\Sigma}$, the optimal $\mathrm{B}^{(c)}$ must span the r columns of $\mathbf{\Sigma}$ with the largest magnitudes.

References

[Baldi and Hornik, 1989] Baldi, P. and Hornik, K. (1989). Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2(1):53–58.

[Bourlard and Kamp, 1988] Bourlard, H. and Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics*, 59:291–294.

[Diamantaras and Kung, 1994] Diamantaras, K. I. and Kung, S.-Y. (1994). Multilayer neural networks for reduced-rank approximation. *IEEE Transactions on Neural Networks*, 5(5):684–697.

[Ghahramani, 1996] Ghahramani, Z. (1996). One hidden layer linear networks and canonical correlations. Unpublished draft, February 1996.

[Kung and Diamantaras, 1991] Kung, S. and Diamantaras, K. (1991). Neural networks for extracting unsymmetric principal components. In Juang, H., Kung, Y., and Kamm, C., editors, *Proceedings of IEEE Workshop on Neural Networks for Signal Processing*. IEEE.

[Mardia et al., 1979] Mardia, K., Kent, J., and Bibby, J. (1979). *Multivariate Analysis*. Academic Press.

[Scharf, 1991] Scharf, L. (1991). The SVD and reduced rank signal processing. In Vaccaro, R., editor, *SVD and Signal Processing II: Algorithms, Analysis and Applications*, chapter 1, pages 3–31. Elsevier Science.

[Stoica and Viberg, 1996] Stoica, P. and Viberg, M. (1996). Maximum likelihood parameter and rank estimation in reduced-rank multivariate linear regressions. *IEEE Transactions on Signal Processing*, 44(12):3069–3078.