Maximum and Minimum Likelihood Hebbian Learning for Exploratory Projection Pursuit

Donald MacDonald[1], Emilio Corchado[1], Colin Fyfe[1], and Erzsebet Merenyi[2]

[1] Applied Computation Intelligence Research Unit, University of Paisley, High St. Paisley, Scotland
{mcdo-ci0, corc-ci0, fyfe-ci0}@Paisley.ac.uk
http://ces.paisley.ac.uk/
[2] Department of Electrical and Computer Engineering
Rice University, 6100 Main St.
Houston, Texas, USA
{Erzsebet}@Rice.edu

Abstract. This paper presents an extension to the learning rules of the Principal Component Analysis Network which has been derived to be optimal for a specific probability density function. We note this pdf is one of a family of pdfs and investigate the learning rules formed in order to be optimal for several members of this family. We show that the whole family of these learning rules can be viewed as methods for performing Exploratory Projection Pursuit. We show that these methods provide a simple robust method for the identification of structure in remote sensing images.

1 Introduction

One problem with the analysis of high dimensional data is identifying structure or patterns which exist across dimensional boundaries. By projecting the data onto a different basis of the space, these patterns may become visible. This presents a problem - how does one decide which basis is optimal for the visualisation of the patterns, without foreknowledge of the patterns in the data.

One solution is Principal Component Analysis (PCA), which is a statistical technique aimed at finding the orthogonal basis that maximises the variance of the projection for a given dimensionality of basis. This involves finding the direction which accounts for most of the data's variance, the first principal component; this variance is then filtered out. The next component is the direction of maximum variance from the remaining data and orthogonal to the 1^{st} PCA basis vector.

We [7, 6]have over the last few years investigated a negative feedback implementation of PCA defined by (1) - (3). Let us have an N-dimensional input vector, $\mathbf{x}$, and an M-dimensional output vector,$\mathbf{y}$, with W_{ij} being the weight linking the j^{th} input to the i^{th} output. The learning rate, η is a small value which will be annealed to zero over the course of training the network. The activation passing from input to output through the weights is described by (1).

The activation is then fed back though the weights from the outputs and the error,e calculated for each input dimension. Finally the weights are updating using simple Hebbian learning.

$$y_i = \sum_{j=1}^{N} W_{ij} x_j, \forall i \tag{1}$$

$$e_i = x_i - \sum_{i=1}^{M} W_{ij} y_i \tag{2}$$

$$\Delta W_{ij} = \eta e_j y_i \tag{3}$$

We have subsequently modified this network to perform clustering with topology preservation [8], to perform Factor Analysis [12, 4] and to perform Exploratory Projection Pursuit [11, 9] (EPP).

PCA is a very powerful and simple technique for investigating high dimensional data. One of the reasons for this is that structure in data is often associated with high variance. But this is not necessarily so: interesting structure can be found in low variance projections. Thus EPP tries to search for interesting directions onto which the data will be projected so that a human can investigate the projections by eye. We thus have to define what interesting means. Most projections through high dimensional data will be approximately Gaussian and so the degree of interestingness of a projection is measurable by its distance from the Gaussian distribution. We have previously introduced nonlinear functions to (3) and shown that the resulting system performs EPP [11, 9] and applied it to visual [10] and to audio data [14].

This paper will deal with a new variation of the basic network which also performs Exploratory Projection Pursuit and apply it to high dimensional astronomical data.

2 Maximum Likelihood Learning

Various researchers e.g. [19, 16] have shown that the above learning rules can be derived as an approximation to the best linear compression of the data. Thus we may start with the cost function

$$J = 1^T E\{(\mathbf{x} - W\mathbf{y})^2\} \tag{4}$$

which we minimise to get (3).

We may show that the minimisation of J is equivalent to minimising the negative log probabilities of the residual, $\mathbf{e}$, if $\mathbf{e}$ is Gaussian [2] and thus is equal to maximising the probabilities of the residual. Let $p(\mathbf{e}) = \frac{1}{Z}\exp(-\mathbf{e}^2)$ Then we can denote a general cost function associated with the network as

$$J = -\log p(\mathbf{e}) = (\mathbf{e})^2 + K \tag{5}$$

where K is a constant. Therefore performing gradient descent on J we have

$$\Delta W \propto -\frac{\delta J}{\delta W} = -\frac{\delta J}{\delta \mathbf{e}}\frac{\delta \mathbf{e}}{\delta W} \approx \mathbf{y}(2\mathbf{e})^T \tag{6}$$

where we have discarded a relatively unimportant term [16].

We have considered an extension of the above in [13, 5] with a more general cost function

$$J = f_1(e) = f_1(\mathbf{x} - W\mathbf{y}) \tag{7}$$

Let us now consider the residual after the feedback to have probability density function

$$p(\mathbf{e}) = \frac{1}{Z}\exp(-|\mathbf{e}|^p) \tag{8}$$

The we can denote a general cost function associated with this network as

$$J = -\log p(\mathbf{e}) = (\mathbf{e})^p + K \tag{9}$$

where K is a constant, Therefore performing gradient descent on J we have

$$\Delta W \propto -\frac{\delta J}{\delta W} = -\frac{\delta J}{\delta \mathbf{e}}\frac{\delta \mathbf{e}}{\delta W} \approx \mathbf{y}(p(|\mathbf{e}|^{p-1}sign(\mathbf{e}))^T \tag{10}$$

We would expect that for leptokurtotic residuals (more kurtotic than a Gaussian distribution), values of $p < 2$ would be appropriate, while platykurtotic residuals (less kurtotic than a Gaussian), values of $p > 2$ would be appropriate. It is a common belief in the ICA community [15] that it is less important to get the exactly the correct distribution when searching for a specific source than it is to get an approximately correct distribution i.e. all supergaussian signals can be retrieved using a generic leptokurtotic distribution and all subgaussian signals can be retrieved using a generic platykurtotic distribution.

Therefore the network operation is:
Feedforward: $y_i = \sum_{j=1}^{N} W_{ij}x_j, \forall i$
Feedback: $e_j = x_j - \sum_{i=1}^{M} W_{ij}y_i$
Weight change: $\Delta W_{ij} = \eta y_i sign(e_j)|e_j|^p$
Fyfe and MacDonald [13] used a single value of p (p=1)in this rule and described this method as performing a robust PCA, but this is not strictly true since only the original ordinary Hebbian learning rule actually performs PCA. It might seem to be more appropriate to link this family of learning rules to Principal Factor Analysis since this method makes an assumption about the noise in a data set and then removes the assumed noise from the covariance structure of the data before performing a PCA. We are doing something similar here in that we are biasing our PCA-type rule on the assumed distribution of the residual. By maximising the likelihood of the residual with respect to the actual distribution, we are matching the learning rule to the pdf of the residual.

More importantly, we may also link the method to EPP. Now the nature and quantification of the interestingness is in terms of how likely the residuals are under a particular model of the pdf of the residual. As with standard EPP, we also sphere the data before applying the learning method to the sphered data.

2.1 Minimum Likelihood Hebbian Learning

Now it is equally possible to perform gradient ascent on J. In this case we find a rule which is the opposite of the above rules in that it is attempting to minimise the likelihood of the residual under the current assumptions about the residual's pdf. The operation of the network is as before but this time we have
Weight change: $\Delta W_{ij} = -\eta y_i sign(e_j)|e_j|^p$
This corresponds to the well known anti-Hebbian learning rule. Now one advantage of this formulation compared with the Maximum Likelihood Hebbian rule is that, in making the residuals as unlikely as possible, we are having the weights learn the structure corresponding to the pdf determined by the parameter p. With the Maximum rule, the weights learn to remove the projections of the data which are furthest from that determined by p. Thus if we wish to search for clusters in our data (typified by a pdf with $p > 2$), we can use Maxmimum Likelihood learning with $p < 2$ [5], which would result in weights which are removing any projections which make these residuals unlikely. Therefore the clusters would be found by projecting onto these weights. Alternatively we may use Minimum Likelihood learning with $p > 2$, which would perform the same job: the residuals have to be unlikely under this value of p and so the weights converge to remove those projections of the data which exhibit clustering.

3 Experimental Results

In this paper we use two remote sensing data sets: the first is the 65 colour spectra of 115 asteroids used by [17]. The data set is composed of a mixture of the 52-colour survey by Bell et al. [1] together with the 8-colour survey conducted by Zellner et al. [20] providing a set of asteroid spectra spanning 0.3-2.5μm. A more detailed description of the dataset is given in [17]. When this extended data set was compared by [17] to the results of Tholen [18] it was found that the additional refinement to the spectra leads to more classes than the taxonomy produced by Tholen. Figure 1 compares Maximum Likelihood Hebbian learning ($p = 0$, an improper distribution since it does not integrate to 1 but we use it to illustrate how robust the method is) with Principal Component Analysis (PCA, $p = 2$) on the Asteroid Data set. Since we wish to maximise the likelihood of the residuals coming from a leptokurtotic distribution, this method is ideal for identifying the outliers in the data set. Clearly the ML method is better at this than standard PCA (Figure 1).

The second data set is an data set constructed by [3] which consists of 16 sets of spectra in which each set contains 32x32 spectra descriptions of which are given in [17] and [3]. Figure 2 gives a comparison of the projections found by the Minimum Likelihood Hebbian learning with values of $p = 2.2$ and $p = 2.3$. We see that both show the 16 clusters in the data set very precisely (standard PCA with $p = 2$ does not perform nearly so well) but that even with this small change in the p-parameter, the projections are quite different.

We see therefore that both Maximum and Minimum Likelihood Hebbian learning can be used to identify structure in data. First results suggest that

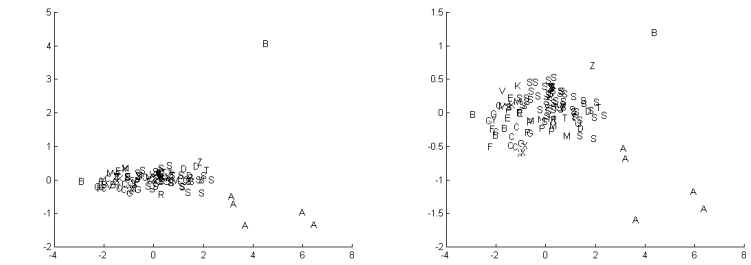

Fig. 1. Principal Component Analysis (left figure) and Maximum Maximum Likelihood Hebbian Learning (right figure) on the Asteroid Data. We used a value of p = 0. It can be clearly seen that the Maximum Likelihood method identifies more structure than PCA.

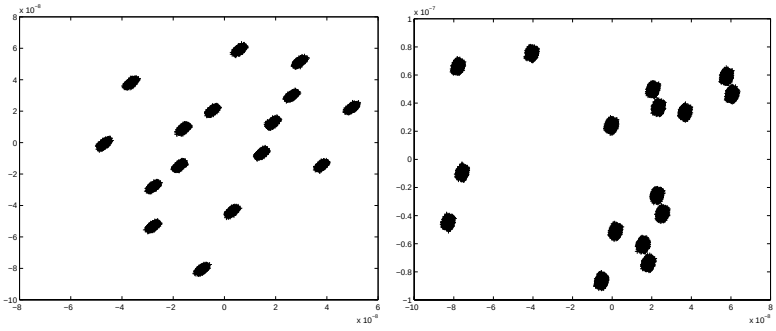

Fig. 2. The figure shows the results of the Minimum Likelihood Hebbian Learning network using $p = 2.3$ (left figure) and $p = 2.2$ (right figure).

Minimum Likelihood is more stable and accurate and that both methods outperform the previous neural implementation of EPP [11]. Future work will be based on a rigorous comparison of these methods.

References

1. J. F. Bell, P. D. Owensby, B.R. Hawke, and M.J. Gaffey. The 52 colour asteroid survey: Final results and interpretation. *Lunalr Planet Sci. Conf, XiX*, pages 57–58, 1988.
2. C. Bishop. *Neural Networks for Pattern Recognition*. Oxford:Clarendon Press, 1995.
3. H. Cetin and D. W. Levandowski. Interactive classification and mapping of multi-dimensional remotely sensed data using n-dimensional probability density functions (npdf). *Photogrammetric Engineering and Remote Sensing*, 57(12):1579–1587, 1991.
4. D. Charles and C. Fyfe. Modelling multiple cause structure using rectification constraints. *Network: Computation in Neural Systems*, 9:167–182, May 1998.
5. E. Corchado and C. Fyfe. Maximum likelihood hebbian learning. In *Tenth European Symposium on Artificial Neural Networks, ESANN2002*, 2002.
6. C. Fyfe. Pca properties of interneurons. In *From Neurobiology to Real World Computing, ICANN 93*, pages 183–188, 1993.
7. C. Fyfe. Introducing asymmetry into interneuron learning. *Neural Computation*, 7(6):1167–1181, 1995.
8. C. Fyfe. Radial feature mapping. In *International Conference on Artificial Neural Networks, ICANN95*, Oct. 1995.
9. C. Fyfe. A comparative study of two neural methods of exploratory projection pursuit. *Neural Networks*, 10(2):257–262, 1997.
10. C. Fyfe and R. Baddeley. Finding compact and sparse distributed representations of visual images. *Network: Computation in Neural Systems*, 6(3):1–12, 1995.
11. C. Fyfe and R. Baddeley. Non-linear data structure extraction using simple hebbian networks. *Biological Cybernetics*, 72(6):533–541, 1995.
12. C. Fyfe and D. Charles. Using noise to form a minimal overcomplete basis. In *Seventh International Conference on Artificial Neural Networks, ICANN99*, 1999.
13. C. Fyfe and D. MacDonald. Epsilon-insensitive hebbian learning. *Neurocomputing*, 2001.
14. M. Girolami and C. Fyfe. An extended exploratory projection pursuit network with linear and non-linear lateral connections applied to the cocktail party problem. *Neural Networks*, 1997.
15. A. Hyvarinen and E. Oja. Independent component analysis : A tutorial. Technical report, Helsinki University of Technology, April 1999.
16. Juha Karhunen and Jyrki Joutsensalo. Representation and separation of signals using nonlinear pca type learning. *Neural Networks*, 7(1):113–127, 1994.
17. E. Merenyi. Self -organising anns for planetary surface composition research. *Journal of Geophysical Research*, 99(E5):10847–10865, 1994.
18. D.J. Tholen. *Asteroid Taxonomy from Cluster Analysis of Photomerty*. PhD thesis, University of Arizona, Tucson, 1984.
19. Lei Xu. Least mean square error reconstruction principle for self-organizing neural-nets. *Neural Networks*, 6(5):627 – 648, 1993.
20. B. Zellner, D.J. Tholen, and E.F. Tedesco. The eight colour asteroid survey: Results from 589 minor planets. *Icarus*, pages 355–416, 1985.