Learning from your neighbour

Graeme Mitchison and Richard Durbin

WHAT can artificial neural networks tell us about the brain? One view is that they can be used to explore the consequences of different synaptic learning rules in a simplified formal setting. However, the most powerful learning algorithms, such as back propagation[1], need an external 'supervisor' to correct the mistakes made by the network, which is an unrealistic requirement especially for early stages of sensory processing. How, therefore, does one learn effectively without a supervisor? On page 161 of this issue[2], Becker and Hinton propose an answer to this question. Theirs is not the first unsupervised learning algorithm, but they take a new approach which has a paradoxical charm: in effect, different pieces of the inputs train each other.

The goal of an unsupervised learning algorithm is to extract meaningful features or variables from a set of input patterns. For example, we can try to find those features that allow the data to be reconstructed as faithfully as possible. This is the goal of principal component analysis, a standard tool of engineering and statistics. By identifying the combinations of inputs with maximum variance, it finds the variables that can be most effectively used to characterize the inputs. Remarkably enough it turns out that the first neurobiological learning rule to be formulated, Hebb's rule[3], is closely related to principal component analysis. Given a simple neural-network model consisting of a single unit, Hebb's rule results in that unit extracting the largest principal component, assuming some form of normalization of synaptic connection strengths[4,5]. With a small amount of modification, a set of units can be made to learn not just the largest component, but a set of components which together capture the greatest part of the variance[6,7].

Principal components appear in at least some cases to be involved in biological processes. In the local processing of visual images, for example, the principal components include edge segments, which are among the first features extracted in primary visual cortex[8]. However, other important variables, such as stereoscopic disparity, will not be explicitly extracted. Becker and Hinton show how one could set about extracting these more elusive variables. One way to describe their approach is that they assume that the interesting properties are more stable than the noise. For example, the depth of a surface, as measured by stereoscopic disparity, will tend to vary smoothly in scanning across an image, whereas the local pixel intensities may vary rapidly because of texture.

Consider a system looking at two neighbouring, non-overlapping patches, and suppose that, corresponding to each patch, there is a unit whose inputs come from that patch only. One could try to make the units extract a stable property by requiring that they both perform the same computation on their input and by minimizing the difference in their responses. But then they might end up both doing nothing (that is, give a zero response). To avoid this, one could try to mimic hebbian principal component learning, and ask the units to maximize the variance in their responses. Becker and Hinton combine these requirements by making the units maximize the variance of the sum of their outputs divided by the variance of their difference.

On the assumption that both the underlying variable and the noise have a gaussian distribution, this is equivalent to maximizing the mutual information of the two outputs. Here one can see particularly clearly how the algorithm works: the mutual information can be large only if, first, the units convey information (that is, they behave nontrivially) and if, second, they respond similarly, so they share this information. The Hebb rule essentially imposes the first constraint alone. By adding the second constraint the new rule allows information to be thrown away when it is not shared by other patches.

Maximizing mutual information can also be interpreted as prediction, because each unit can be used to predict the behaviour of neighbouring patches. The notion of prediction is more general than that of stability: we can look for properties that predict future inputs, or predict one set of sensory data through another sensory modality. Prediction can help to complete or interpret missing data, and where prediction fails something interesting is likely to be happening. For example, places where disparity changes sharply will usually correspond to the edges of objects.

Neural networks are inspired by real neurons, but is there is any reverse flow of inspiration? Might a rule such as this operate in the brain? It seems unlikely that neurons compute something as mathematically complex as the ratio of variances, let alone the determinants which occur in the more general expression for more than two units. Furthermore, some of the difficulties of back propagation apply to the multilayer version of this algorithm, which must somehow feed back a complex error signal to earlier stages in the neural pathway. But it is important not to be too intimidated by the mathematical formulation. After all, principal component analysis, which in its standard form requires matrix inversion, might seem an unlikely operation for neurons to accomplish. Yet it can be carried out by suitably organized hebbian machinery. It seems likely, in fact, that there are natural ways for neurons to carry out Becker and Hinton's kind of analysis, or something very close to it, and this may provide another clue to help us explore synaptic learning rules in the brain. □

Graeme Mitchison is in the Physiological Laboratory, University of Cambridge, Cambridge CB2 3EG, UK. Richard Durbin is in the MRC Laboratory of Molecular Biology, Hills Road, Cambridge CB2 2QH, UK.

1. Rumelhart, D. E., Hinton, G. E. & Williams, R. J. *Nature* **323**, 533–536 (1986).
2. Becker, S. & Hinton, G. E. *Nature* **355**, 161–163 (1992).
3. Hebb, D. O. *The Organization of Behavior* (Wiley, New York, 1949).
4. Oja, E. *J. math. Biol.* **15**, 267–273 (1982).
5. Linsker, R. *Computer* 105–117 (March 1988).
6. Sanger, T. D. *Neural Networks* **2**, 459–473 (1989).
7. Földiák, P. In *Proc. Int. Joint. Conf. Neural Networks* Vol. 1, 401–405 (IEEE, New York, 1989).
8. Hubel, D. H. & Wiesel, T. N. *J. Physiol.* **160**, 106–154 (1962).