Submitted to *Handbook of Brain Theory and Neural Networks, 2.*

Helmholtz Machines and Wake-Sleep Learning

Peter Dayan

Gatsby Computational Neuroscience Unit
Alexandra House
17 Queen Square
London WC1N 3AR
ENGLAND

phone: +44 (0) 207 679 1175
fax: +44 (0) 207 679 1173
email: dayan@gatsby.ucl.ac.uk

1 Introduction

Unsupervised learning is largely concerned with finding structure among sets of input patterns such as visual scenes. One important example of structure comes in cases that the input patterns are generated or caused in a systematic way, for instance when objects with different shapes, surface properties and positions are lit by lights of different characters and viewed by an observer with a digital camera at a particular relative location. Here, the inputs can be seen as living on a manifold that has many fewer dimensions than the space of all possible activation patterns over the pixels of the camera, otherwise random visual noise in the camera would appear like a normal visual scene. The manifold should correctly be parameterized by the generators themselves (ie the objects, the lights, *etc*) (see Hinton and Ghahramani, 1997).

The Helmholtz machine (Dayan *et al*, 1995) is an example of an approach to unsupervised learning called analysis by synthesis (*eg* Neisser, 1967). Imagine that we have a perfect computer graphics model, which indicates how objects appear to observers. We can use this model to synthesize inputs patterns that look just like the input patterns the observer would normally receive, with the crucial difference that, since we synthesized them, we know in detail how the images were generated. We can then use these paired images and generators to train a model that analyses new images to find out how they were generated too, *ie* that represents them according to which particular generators underlie them. Conversely, if we have a perfect analysis model, which indicates the generators underlying any image, then it is straightforward to use the paired images and generators to train a graphics model. In the Helmholtz machine, we attempt to have an imperfect graphics or generative model train a better analysis or recognition model; and an imperfect recognition model train a better generative model.

There are three key issues for an analysis by synthesis model. First is the nature of the synthetic or generative model – for the Helmholtz machine, this is a structured belief network (Jordan, 1998) that is a model for hierarchical top-down connections in the cortex. This model has an overall structure (the layers, units within a layer, *etc*), and a set of generative parameters, which determine the probability distribution it expresses. The

units in the lowest layer of the network are observable, in the sense that it is on them that the inputs are presented; units higher up in the network are latent, since they are not directly observable from inputs. The second issue for an analysis by synthesis model is how new inputs are analyzed or recognized in the light of this generative model, *ie* how the states of the latent units are determined so that the input is represented in terms of the way that it would be generated by the generative model. For the Helmholtz machine, this is done in an approximate fashion using a second structured belief network (called the recognition model) over the latent units, whose parameters are also learned. The recognition network is a model for the standard, bottom-up, connections in cortex. The third issue is the way that the generative and recognition models are learned from data. For the wake-sleep learning algorithm for the stochastic Helmholtz machine (Hinton *et al* 1995), this happens in two phases. In the wake phase, the recognition model is used to estimate the underlying generators (*ie* the states of the latent units) for a particular input pattern, and then the generative model is altered so that those generators are more likely to have produced the input that is actually observed. In the sleep phase, the generative model fantasizes inputs by choosing particular generators stochastically, and then the recognition model is altered so that it is more likely to report those particular generators, if the fantasized input were actually to be observed.

2 The Generative Model

Figure 1 shows an example of a three layer Helmholtz machine, involving (for the sake of simplicity) binary, stochastic, units. The generative model uses top-down biases and weights $\mathcal{G} = \{g^x, g^y, g^d, \mathsf{G}^{xy}, \mathsf{G}^{yd}\}$ to parameterize a probability distribution over the input units $\mathbf{d} = (d_1, d_2, \ldots)$. In this model, the units *within* each layer are conditionally independent given the

binary states of the layer above (this is called a *factorial* property). Thus

$$\mathcal{P}[\mathbf{x};\mathcal{G}] = \prod_i \mathcal{P}[x_i;\mathcal{G}] \tag{1}$$

$$\mathcal{P}[\mathbf{y}|\mathbf{x};\mathcal{G}] = \prod_j \mathcal{P}[y_j|\mathbf{x};\mathcal{G}] \tag{2}$$

$$\mathcal{P}[\mathbf{d}|\mathbf{y};\mathcal{G}] = \prod_k \mathcal{P}[d_k|\mathbf{x};\mathcal{G}] \tag{3}$$

and so

$$\mathcal{P}[\mathbf{d};\mathcal{G}] = \sum_{\mathbf{x},\mathbf{y}} \mathcal{P}[\mathbf{x};\mathcal{G}]\mathcal{P}[\mathbf{y}|\mathbf{x};\mathcal{G}]\mathcal{P}[\mathbf{d}|\mathbf{y};\mathcal{G}]. \tag{4}$$

For binary stochastic units,

$$\mathcal{P}[x_i; g_i^x] = \sigma(g_i^x) \qquad \mathcal{P}[y_j|\mathbf{x}; g_j^y, \mathsf{G}^{xy}] = \sigma\left(g_j^y + \sum_i \mathsf{G}_{ji}^{xy} x_i\right) \equiv \hat{y}_j(\mathbf{x}) \tag{5}$$

where $\sigma(u) = 1/(1+\exp(-u))$ is the standard sigmoid function. Although the units $\mathbf{y}$ are conditionally independent given the states of the units $\mathbf{x}$ in the layer above, they are not marginally independent, *ie* $\mathbf{x}$ can capture correlations in the states of $\mathbf{y}$, and likewise for $\mathbf{d}$. This top-down generative model is a simple example of a sigmoid belief net (Neal, 1992; Jordan, 1998). The conditional independence within a layer makes it very straightforward to generate a sample $\mathbf{d}^\bullet$ from $\mathcal{P}[\mathbf{d}|\mathcal{G}]$ by fantasizing a sample $\mathbf{x}^\bullet$, then $\mathbf{y}^\bullet$ given $\mathbf{x}^\bullet$ and then $\mathbf{d}^\bullet$ given $\mathbf{y}^\bullet$.

3 The Recognition Model

When the generative model is used to create such a complete fantasy, we consider $\mathbf{x}^\bullet$ and $\mathbf{y}^\bullet$ as the *generators* of $\mathbf{d}^\bullet$. The task for the recognition model is to take a new example $\mathbf{d}$ and report the state(s) of $\mathbf{x}$ and $\mathbf{y}$ that might have generated it. Using Bayes theorem, we know that

$$\mathcal{P}[\mathbf{x},\mathbf{y}|\mathbf{d};\mathcal{G}] = \mathcal{P}[\mathbf{d}|\mathbf{x},\mathbf{y};\mathcal{G}]\frac{\mathcal{P}[\mathbf{x};\mathcal{G}]\mathcal{P}[\mathbf{y}|\mathbf{x};\mathcal{G}]}{\mathcal{P}[\mathbf{d};\mathcal{G}]} \tag{6}$$

It is straightforward to calculate all the terms on the right hand side *except* for the denominator $\mathcal{P}[\mathbf{d};\mathcal{G}]$, which involves a sum over all the possible states of $\mathbf{x}$ and $\mathbf{y}$ (a set which grows exponentially large as the number of elements in $\mathbf{x}$ and $\mathbf{y}$ grows). Thus, an approximation to $\mathcal{P}[\mathbf{x},\mathbf{y}|\mathbf{d};\mathcal{G}]$ is usually required. The stochastic version of the Helmholtz machine (the only version we discuss here) uses for approximate recognition a bottom-up, belief network (see figure 1) over exactly the same units giving a probability distribution $\mathcal{Q}[\mathbf{x},\mathbf{y}|\mathbf{d};\mathcal{R}] = \mathcal{Q}[\mathbf{y}|\mathbf{d};\mathcal{R}]\mathcal{Q}[\mathbf{x}|\mathbf{y};\mathcal{R}]$ using a separate set of parameters, the bottom-up biases and weights $\mathcal{R} = \{\mathbf{r}^x, \mathbf{r}^y, \mathcal{R}^{dy}, \mathcal{R}^{yx}\}$. A critical approximation is that the recognition model is assumed to be factorial in the bottom-up direction, *eg* y_1 is independent of y_2 given $\mathbf{d}$. Over the course of learning, it is intended that $\mathcal{Q}[\mathbf{x},\mathbf{y}|\mathbf{d};\mathcal{R}]$ should come to be as close to $\mathcal{P}[\mathbf{x},\mathbf{y}|\mathbf{d};\mathcal{G}]$ as possible. Just as it is simple to generate a sample, *ie* a fantasy, top-down from the generative model; it is easy to generate a sample, *ie* to recognize the input in terms of its generators, bottom-up from the recognition model.

4 Wake-Sleep Learning

As for many unsupervised learning methods, the underlying goal of wake-sleep learning is to perform maximum likelihood density estimation by maximising the log probability of the observed data $\mathcal{D} = \{\mathbf{d}(1), \mathbf{d}(2), \ldots\}$ under the generative model, that is $E(\mathcal{G}) = \sum_t \log \mathcal{P}[\mathbf{d}(t)|\mathcal{G}]$. One key idea, due to Neal and Hinton (1998); Zemel (1994) is that

$$\log \mathcal{P}[\mathbf{d};\mathcal{G}] = \sum_{\mathbf{x},\mathbf{y}} \mathcal{P}[\mathbf{x},\mathbf{y}|\mathbf{d};\mathcal{G}] \log \mathcal{P}[\mathbf{x},\mathbf{y},\mathbf{d};\mathcal{G}] + \mathcal{H}\left[\mathcal{P}[\mathbf{x},\mathbf{y}|\mathbf{d};\mathcal{G}]\right] \tag{7}$$

$$\geq \sum_{\mathbf{x},\mathbf{y}} \mathcal{Q}[\mathbf{x},\mathbf{y};\mathbf{d},\mathcal{R}] \log \mathcal{P}[\mathbf{x},\mathbf{y},\mathbf{d};\mathcal{G}] + \mathcal{H}\left[\mathcal{Q}[\mathbf{x},\mathbf{y};\mathbf{d},\mathcal{R}]\right] \tag{8}$$

$$= \log \mathcal{P}[\mathbf{d};\mathcal{G}] - \mathrm{KL}\left[\mathcal{Q}[\mathbf{x},\mathbf{y};\mathbf{d},\mathcal{R}], \mathcal{P}[\mathbf{x},\mathbf{y}|\mathbf{d};\mathcal{G}]\right] \tag{9}$$

$$\equiv -\mathcal{F}[\mathbf{d};\mathcal{R},\mathcal{G}] \tag{10}$$

where, $\mathcal{H}[\mathcal{A}] = -\sum_{\mathbf{a}} \mathcal{A}[\mathbf{a}] \log \mathcal{A}[\mathbf{a}]$ is the entropy of probability distribution $\mathcal{A}$, $\mathrm{KL}[\mathcal{A},\mathcal{B}] = \sum_{\mathbf{a}} \mathcal{A}[\mathbf{a}] \log \mathcal{A}[\mathbf{a}]/\mathcal{B}[\mathbf{a}]$ is the Kullback-Liebler (KL) divergence between two distributions $\mathcal{A}$ and $\mathcal{B}$, and, in inequality 8,

$\mathcal{Q}[\mathbf{x}, \mathbf{y}; \mathbf{d}, \mathcal{R}]$ can be *any* probability distribution over $\mathbf{x}, \mathbf{y}$. From equation 9, equality holds when $\mathcal{Q}[\mathbf{x}, \mathbf{y}; \mathbf{d}, \mathcal{R}]$ is the true analytical distribution $\mathcal{P}[\mathbf{x}, \mathbf{y}|\mathbf{d}; \mathcal{G}]$. Expression $\mathcal{F}[\mathbf{d}; \mathcal{R}, \mathcal{G}]$ can be seen as a Helmholtz free energy, hence the name of the machine.

During the *wake* phase, a single pattern $\mathbf{d}^\circ$ is sampled from $\mathcal{D}$, and is presented to the recognition model. This is executed bottom-up to produce a single sample $\mathbf{y}^\circ$ given $\mathbf{d}$ and $\mathbf{x}^\circ$ given $\mathbf{y}^\circ$. Then, the parameters $\mathcal{G}$ of the generative model are changed using stochastic gradient ascent of the lower bound to the log probability, *ie* proportionally to

$$\nabla_{\mathcal{G}} \log \mathcal{P}[\mathbf{x}^\circ, \mathbf{y}^\circ, \mathbf{d}; \mathcal{G}] = \nabla_{\mathcal{G}} \left\{ \log \mathcal{P}[\mathbf{x}^\circ; \mathcal{G}] + \log \mathcal{P}[\mathbf{y}^\circ|\mathbf{x}^\circ; \mathcal{G}] + \log \mathcal{P}[\mathbf{d}^\circ|\mathbf{y}^\circ; \mathcal{G}] \right\}$$

For activation functions such as those in equation 5, this leads to particularly simple 'delta' learning rules such as

$$\Delta g_j^y \propto \left(y_j^\circ - \hat{y}_j(\mathbf{x}^\circ) \right) \quad \Delta \mathsf{G}_{ij}^{xy} \propto \left(y_j^\circ - \hat{y}_j(\mathbf{x}^\circ) \right) x_i^\circ$$

in which the output of the recognition model is used as the *target* for the generative model.

The ideal for the *sleep* phase would be to change the recognition weights $\mathcal{R}$ using stochastic gradient descent also of the KL divergence in equation 9. Unfortunately, this is not generally tractable. The second key idea in wake-sleep learning is to attempt during sleep to minimize $\mathrm{KL}\left[\mathcal{P}[\mathbf{x}, \mathbf{y}|\mathbf{d}; \mathcal{G}]; \mathcal{Q}[\mathbf{x}, \mathbf{y}; \mathbf{d}, \mathcal{R}]\right]$ instead. This is not the same, since the KL divergence is not symmetric, although they are equal at their joint minimum where $\mathcal{P}[\mathbf{x}, \mathbf{y}|\mathbf{d}; \mathcal{G}] = \mathcal{Q}[\mathbf{x}, \mathbf{y}; \mathbf{d}, \mathcal{R}]$. The KL divergence the wrong way round can be minimized by fantasizing sample $\mathbf{x}^\bullet, \mathbf{y}^\bullet$ and $\mathbf{d}^\bullet$ from the generative model, and then changing the recognition weights according to

$$\nabla_{\mathcal{R}} \log \mathcal{P}[\mathbf{x}^\bullet, \mathbf{y}^\bullet, \mathbf{d}^\bullet; \mathcal{R}] = \nabla_{\mathcal{R}} \left\{ \log \mathcal{P}[\mathbf{d}^\bullet; \mathcal{R}] + \log \mathcal{P}[\mathbf{y}^\bullet|\mathbf{d}^\bullet; \mathcal{R}] + \log \mathcal{P}[\mathbf{x}^\bullet|\mathbf{y}^\bullet; \mathcal{R}] \right\}$$

For activation functions such as those in equation 5, this leads to the same simple 'delta' learning rules as for the generative model, except that the output of the generative model is used as the target for the recognition model.

Since sleep learning involves an approximation, it is only in very special cases (see Neal and Dayan, 1997) that it is possible to prove even that it

is appropriately stable. Nevertheless, the model has been shown to work quite well in practice. Figure 1B shows the result of applying wake-sleep learning to a set of input patterns (left column) that are binary images of the handwritten digit '9'. The right column shows fantasized samples following learning, and these can be seen to be generated by a distribution close to that in the training distribution.

5 Discussion

As a directed belief network for analysis by synthesis that is trained according to maximum likelihood density estimation, the Helmholtz machine lives in what has become a rather crowded space. Congeners include probabilistic autoencoders (see Zemel, 1994); forward-inverse models (Kawato, Hayakama and Inui, 1993), maximum likelihood Independent Component Analysis (Bell and Sejnowski, 1995), sparse coding models (Olshausen and Field, 1996), hierarchical Gaussian models (Luettgen and Willsky, 1995; Rao and Ballard, 1997), mean field models (see Jordan, 1998), and rectified Gaussian belief networks (Hinton and Ghahramani, 1997). In this context, the key property of the Helmholtz machine is that it uses an explicit recognition model which has its own parameters rather than performing recognition by an iterative process involving only the parameters of the generative model. In some ways this is an advantage – in particular, recognition can occur swiftly in a single pass. Learning during the sleep phase can be considered as a way of caching knowledge about how to do recognition effectively. In other ways it is a disadvantage, since the recognition model introduces a extra set of parameters that need to be learned and since, unlike the iterative mean field recognition methods that underlie most of the architectures mentioned above, the approximation involved in the recognition model in the Helmholtz machine cannot be tailored on-line to the particular input pattern that is presented. Another key feature is that, unlike many of these methods, the Helmholtz machine is explicitly designed to be hierarchical – units in one layer capture (*ie* both represent and generate) correlations in the layer below. Unlike Independent Components Analysis, for instance, units within a layer are not forced to be marginally independent in the generative model, only conditionally independent *given* the activities in the layer above. This po-

tentially allows it a much richer representation of the inputs. Also, the recognition model in the Helmholtz machine allows at least some correlations among the states of the hierarchical generators, a feature denied to mean field methods.

The Helmholtz machine also bears an interesting relationship to the Boltzmann machine (Hinton and Sejnowski, 1986), which can be seen as an undirected belief net. In the Boltzmann machine, which also lacks an explicit recognition model, a potentially drawn-out process of Gibbs sampling is used to recognize and generate inputs, since there is nothing like the simple, one-pass, directed recognition and generative belief networks of the Helmholtz machine. Also, the Boltzmann machine learning rule performs true stochastic gradient ascent of the log likelihood using a contrastive procedure, which, confusingly, involves wake and sleep phases that are quite different from the wake and the sleep phases of the wake-sleep algorithm. The two phases of the Boltzmann machine contrast the statistics of the activations of the network when input patterns are presented with the statistics of the activations of the network when it is running 'free'. This contrastive procedure involves substantial noise and is therefore slow. In the wake-sleep learning procedure for the Helmholtz machine, the wake and sleep phases are not contrastive. Rather, the recognition and generative models are forced to chase the other. Hinton's (1999) recent product of experts (PoE) architecture, which can be seen as a restricted Boltzmann machine, is more closely related to the Helmholtz machine, since it uses a model for which recognition truly has the factorial structure that is only approximate here. The PoE model also uses a different, and more efficient, learning rule than the Boltzmann machine.

The most important open issue for the Helmholtz machine as a model of top-down and bottom-up connections in the cortex is how to weaken the approximation that the recognition and generative models are factorial within layers, without destroying the simplicity of sampling from and learning the models.

6 References

Bell, A.J. and T.J. Sejnowski, 1995. An information maximisation approach to blind separation and blind deconvolution, Neural Computation, 7:1129-1159.

Dayan, P., G.E. Hinton, R.M. Neal and , R.S. Zemel, 1995. The Helmholtz machine, Neural Computation, 7:889-904.

Hinton, G.E. and Z. Ghahramani, 1997. Generative models for discovering sparse distributed representations, Philosophical Transactions Royal Society, B, 352:1177-1190.

Hinton, G.E., 1999. Products of experts, in Proceedings of the Ninth International Conference on Artificial Neural Networks, 1:1-6.

Hinton, G.E., P. Dayan, B.J. Frey and R.M. Neal, 1995. The wake-sleep algorithm for unsupervised neural networks, Science, 268:1158-1160.

Hinton, G.E. and T.J. Sejnowski, 1986. Learning and relearning in Boltzmann machines, in Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations, (D.E. Rumelhart, J.L. McClelland and the PDP research group, eds.) Cambridge, MA: MIT Press, pp. 282-317.

Jordan, M.I. editor, 1998. Learning in Graphical Models, Dordrecht: Kluwer.

Kawato, M., H. Hayakama and T. Inui, 1993. A forward-inverse optics model of reciprocal connections between visual cortical areas, Network: Computation in Neural Systems, 4:415-422.

Luettgen, M.R. and A.S. Willsky, 1995. Likelihood calculation for a class of multiscale stochastic models, with application to texture discrimination, IEEE Transactions on Image Processing, 4:194-207.

Neal, R.M., 1992. Connectionist learning of belief networks, Artificial Intelligence, 56:71-113.

Neal, R.M. and P. Dayan, 1997. Factor analysis using delta-rule wake-sleep learning. Neural Computation 9:1781-1803.

Neal, R.M. and G.E. Hinton, 1994. A view of the EM algorithm that justifies incremental, sparse, and other variants, In Jordan (1998), pp. 355-368.

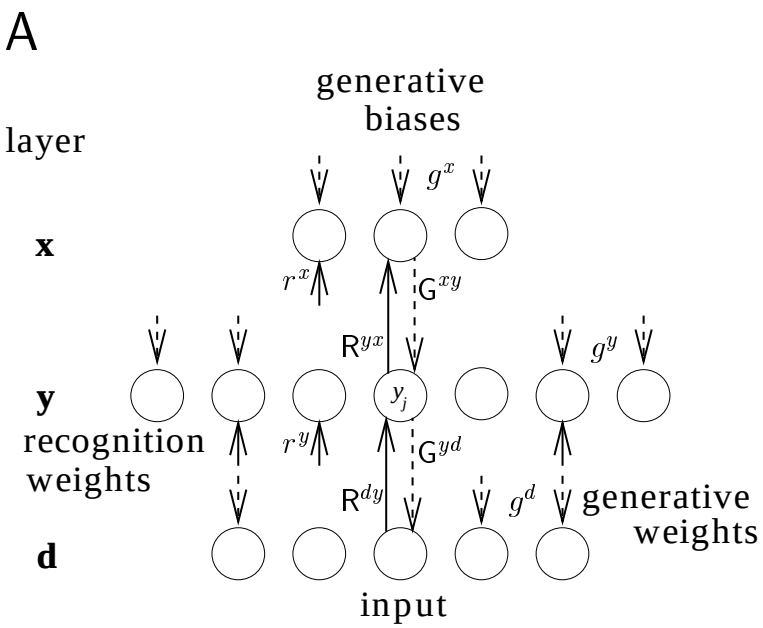
Neisser, U., 1967. Cognitive psychology. New York, Appleton-Century-Crofts.

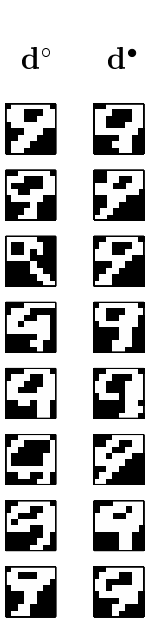
Olshausen, B.A. and D.J. Field, 1996. Emergence of simple-cell receptive field properties by learning a sparse code for natural images, Nature, 381:607-609.

Rao, R.P.N. and D.M. Ballard, 1997. Dynamic model of visual recognition predicts neural response properties in the visual cortex, Neural Computation, 9:721-763.

Zemel, R.S., 1994. A minimum description length framework for unsupervised learning, PhD Dissertation, Computer Science, University of Toronto.

Figure 1: Helmholtz machine. A) Structure of a three-layer Helmholtz machine, with generative weights and biases $\mathcal{G}$ (dashed) and recognition weights and biases $\mathcal{R}$ (solid). B) Handwritten digit example. The left column shows eight samples from a training set of binarised, 8×8, handwritten '9's; the right column shows eight samples produced by the generative model after training. The training set is as described in Hinton *et al* (1995).


A
generative
biases
layer
x
g^x
r^x
G^{xy}
R^{yx}
g^y
y
y_j
recognition
weights
r^y
G^{yd}
R^{dy}
g^d
generative
weights
d
input


B
$\mathbf{d}^{\circ}$
$\mathbf{d}^{\bullet}$