Representational Power of Restricted Boltzmann Machines and Deep Belief Networks

Nicolas Le Roux and Yoshua Bengio
Dept. IRO, Université de Montréal
C.P. 6128, Montreal, Qc, H3C 3J7, Canada
{*lerouxni,bengioy*}*@iro.umontreal.ca*
http://www.iro.umontreal.ca/∼lisa

Abstract

Deep Belief Networks (DBN) are generative neural network models with many layers of hidden explanatory factors, recently introduced by Hinton et al., along with a greedy layer-wise unsupervised learning algorithm. The building block of a DBN is a probabilistic model called a Restricted Boltzmann Machine (RBM), used to represent one layer of the model. Restricted Boltzmann Machines are interesting because inference is easy in them, and because they have been successfully used as building blocks for training deeper models. We first prove that adding hidden units yields strictly improved modelling power, while a second theorem shows that RBMs are universal approximators of discrete distributions. We then study the question of whether DBNs with more layers are strictly more powerful in terms of representational power. This suggests a new and less greedy criterion for training RBMs within DBNs.

1 Introduction

Learning algorithms that learn to represent functions with many levels of composition are said to have a *deep architecture*. Bengio and Le Cun (2007) discuss results in computational theory of circuits that strongly suggest that deep architectures are much more efficient in terms of representation (number of computational elements, number of parameters) than their shallow counterparts. In spite of the fact that 2-level architectures (e.g., a one-hidden layer neural network, a kernel machine, or a 2-level digital circuit) are able to represent any function (see for example (Hornik, Stinchcombe, & White, 1989)), they may need a huge number of elements and, consequently, of training examples. For example, the parity function on d bits (which associates the value 1 with a vector $\mathbf{v}$ if $\mathbf{v}$ has an odd number of bits equal to 1 and 0 otherwise) can be implemented by a digital circuit of depth $\log(d)$ with $O(d)$ elements but requires $O(2^d)$ elements to be represented by a 2-level digital circuit (Ajtai, 1983) (e.g., in conjunctive or disjunctive normal form). We proved a similar result for Gaussian kernel machines: they require $O(2^d)$ non-zero coefficients (i.e., support vectors in a Support Vector Machine) to represent such highly

varying functions (Bengio, Delalleau, & Le Roux, 2006a). On the other hand, training learning algorithms with a deep architecture (such as neural networks with many hidden layers) appears to be a challenging optimization problem (Tesauro, 1992; Bengio, Lamblin, Popovici, & Larochelle, 2007).

Hinton, Osindero, and Teh (2006) introduced a greedy layer-wise *unsupervised* learning algorithm for Deep Belief Networks (DBN). The training strategy for such networks may hold great promise as a principle to help address the problem of training deep networks. Upper layers of a DBN are supposed to represent more "abstract" concepts that explain the input data whereas lower layers extract "low-level features" from the data. In (Bengio et al., 2007; Ranzato, Poultney, Chopra, & LeCun, 2007), this greedy layer-wise principle is found to be applicable to models other than DBNs. DBNs and RBMs have already been applied successfully to a number of classification, dimensionality reduction, information retrieval, and modelling tasks (Welling, Rosen-Zvi, & Hinton, 2005; Hinton et al., 2006; Hinton & Salakhutdinov, 2006; Bengio et al., 2007; Salakhutdinov & Hinton, 2007).

In this paper we show that adding hidden units yields strictly improved modelling power, unless the RBM already perfectly models the data. Then, we prove that an RBM can model any discrete distribution, a property similar to those of neural networks with one hidden layer. Finally, we discuss the representational power of DBNs and find a puzzling result about the best that could be achieved when going from 1-layer to 2-layer DBNs. Note that the proofs of universal approximation by RBMs are constructive but these constructions are not practical as they would lead RBMs with potentially as many hidden units as examples, and this would defy the purpose of using RBMs as building blocks of a deep network that efficiently represents the input distribution. Important theoretical questions therefore remain unanswered concerning the potential for DBNs that stack multiple RBMs to represent a distribution efficiently.

1.1 Background on RBMs

1.1.1 Definition and properties

A Restricted Boltzmann Machine (RBM) is a particular form of the Product of Experts model (Hinton, 1999, 2002) which is also a Boltzmann Machine (Ackley, Hinton, & Sejnowski, 1985) with a bipartite connectivity graph. An RBM with n hidden units is a parametric model of the joint distribution between hidden variables h_i (explanatory factors, collected in vector $\mathbf{h}$ and observed variables v_j (the example, collected in vector $\mathbf{v}$), of the form

$$p(\mathbf{v}, \mathbf{h}) \propto \exp(-E(\mathbf{v}, \mathbf{h})) = e^{\mathbf{h}^T W \mathbf{v} + b^T v + c^T \mathbf{h}}$$

with parameters $\theta = (W, b, c)$ and $v_j, h_i \in \{0, 1\}$. $E(\mathbf{v}, \mathbf{h})$ is called the **energy** of the state $(\mathbf{v}, \mathbf{h})$. We consider here the simpler case of binary units. It is straightforward to show that $P(\mathbf{v}|\mathbf{h}) = \prod_j P(v_j|\mathbf{h})$ and $P(v_j = 1|\mathbf{h}) = \text{sigm}(b_j + \sum_i W_{ij} h_i)$ (where sigm is the sigmoid function defined as $\text{sigm}(x) = \frac{1}{1+\exp(-x)}$), and $P(\mathbf{h}|\mathbf{v})$ has a similar form: $P(\mathbf{h}|\mathbf{v}) = \prod_i P(h_i|\mathbf{v})$ and $P(h_i = 1|\mathbf{v}) = \text{sigm}(c_i + \sum_j W_{ij} v_j)$. Although the marginal distribution $p(\mathbf{v})$ is not tractable, it can be easily computed up to a normalizing constant.

Furthermore, one can also sample from the model distribution using Gibbs sampling. Consider a Monte-Carlo Markov chain (MCMC) initialized with $\mathbf{v}$ sampled from the empirical data distribution (distribution denoted p_0). After sampling $\mathbf{h}$ from $P(\mathbf{h}|\mathbf{v})$, sample $\mathbf{v}'$ from $P(\mathbf{v}'|\mathbf{h})$, which follows a distribution denoted p_1. After k such steps we have samples from p_k, and the model's generative distribution is p_∞ (due to convergence of the Gibbs MCMC).

1.1.2 Training and Contrastive Divergence

Carreira-Perpiñan and Hinton (2005) showed that the derivative of the log-likelihood of the data under the RBM with respect to the parameters is

$$\frac{\partial \log p(\mathbf{v},\mathbf{h})}{\partial \theta} = -\left\langle \frac{\partial \log E(\mathbf{v},\mathbf{h})}{\partial \theta} \right\rangle_0 + \left\langle \frac{\partial \log E(\mathbf{v},\mathbf{h})}{\partial \theta} \right\rangle_\infty \tag{1}$$

where averaging is over both $\mathbf{v}$ and $\mathbf{h}$, $\langle\cdot\rangle_0$ denotes an average with respect to p_0 (the data distribution) multiplied by $P(\mathbf{h}|\mathbf{v})$, and $\langle\cdot\rangle_\infty$ denotes an average with respect to p_∞ (the model distribution): $p_\infty(\mathbf{v},\mathbf{h}) = p(\mathbf{v},\mathbf{h})$.
Since computing the average over the true model distribution is intractable, Hinton et al. (2006) use an approximation of that derivative called **contrastive divergence** (Hinton, 1999, 2002): one replaces the average $\langle\cdot\rangle_\infty$ with $\langle\cdot\rangle_k$ for relatively small values of k. For example, in Hinton et al. (2006), Hinton and Salakhutdinov (2006), Bengio et al. (2007), Salakhutdinov and Hinton (2007), one uses $k = 1$ with great success. The average over $\mathbf{v}$'s from p_0 is replaced by a sample from the empirical distribution (this is the usual stochastic gradient sampling trick) and the average over $\mathbf{v}$'s from p_1 is replaced by a single sample from the Markov chain. The resulting gradient estimator involves only very simple computations, and for the case of binary units, the gradient estimator on weight W_{ij} is simply $P(h_i = 1|\mathbf{v})v_j - P(h_i = 1|\mathbf{v}')v'_j$, where $\mathbf{v}'$ is a sample from p_1 and $\mathbf{v}$ is the input example that starts the chain. The procedure can easily be generalized to input or hidden units that are not binary (e.g., Gaussian or exponential, for continuous-valued units (Welling et al., 2005; Bengio et al., 2007)).

2 RBMs are Universal Approximators

We will now prove that RBMs with a data-selected number of hidden units become non-parametric and possess universal approximation properties relating them closely to classical multilayer neural networks, but in the context of probabilistic unsupervised learning of an input distribution.

2.1 Better Model with Increasing Number of Units

We show below that when the number of hidden units of an RBM is increased, there are weight values for the new units that guarantee improvement in the training log-likelihood or equivalently in the KL divergence between the data distribution p_0 and the

model distribution $p_\infty = p$. These are equivalent since

$$KL(p_0||p) = \sum_{\mathbf{v}} p_0(\mathbf{v}) \log \frac{p_0(\mathbf{v})}{p(\mathbf{v})} = -H(p_0) - \frac{1}{N}\sum_{i=1}^{N} \log p(\mathbf{v}^{(i)})$$

when p_0 is the empirical distribution, with $\mathbf{v}^{(i)}$ the i^{th} training vector and N the number of training vectors.
Consider the objective of approximating an arbitrary distribution p_0 with an RBM. Let p denote the distribution over visible units $\mathbf{v}$ obtained with an RBM that has n hidden units and $p_{w,c}$ denote the input distribution obtained when adding a hidden unit with weights w and bias c to that RBM. The RBM with this extra unit has the same weights and biases for all other hidden units, and the same input biases.

Lemma 2.1. *Let R_p be the equivalence class containing the RBMs whose associated marginal distribution over the visible units is p. The operation of adding a hidden unit to an RBM of R_p preserves the equivalence class. Thus, the set of RBMs composed of an RBM of R_p and an additional hidden unit is also an equivalence class (meaning that all the RBMs of this set have the same marginal distribution over visible units).*

Proof in appendix.

R_p will be used here to denote any RBM in this class. We also define $R_{p_{w,c}}$ as the set of RBMs obtained by adding a hidden unit with weight w and bias c to an RBM from R_p and $p_{w,c}$ the associated marginal distribution over the visible units. As demonstrated in the above lemma, this does not depend on which particular RBM from R_p we choose.
We then wish to prove, that, regardless p and p_0, if $p \neq p_0$, there exists a pair (w, c) such that $KL(p_0||p_{w,c}) < KL(p_0||p)$, i.e., that one can improve the approximation of p_0 by inserting an extra hidden unit with weight vector w and bias c.
We will first state a trivial lemma needed for the rest of the proof. It says that inserting a unit with bias $c = -\infty$ does not change the input distribution associated with the RBM.

Lemma 2.2. *Let p be the distribution over binary vectors $\mathbf{v}$ in $\{0,1\}^d$, obtained with an RBM R_p and let $p_{w,c}$ be the distribution obtained when adding a hidden unit with weights w and bias c to R_p. Then*

$$\forall p, \forall w \in \mathbb{R}^d, p = p_{w,-\infty}$$

Proof. Denoting $\widetilde{\mathbf{h}} = \begin{bmatrix} \mathbf{h} \\ h_{n+1} \end{bmatrix}$, $\widetilde{W} = \begin{bmatrix} W \\ w^T \end{bmatrix}$ and $\widetilde{C} = \begin{bmatrix} C \\ c \end{bmatrix}$ where w^T denotes the transpose of w and introducing $z(\mathbf{v}, \mathbf{h}) = \exp(\mathbf{h}^T W \mathbf{v} + B^T \mathbf{v} + C^T \mathbf{h})$, we can express $p(\mathbf{v}, \mathbf{h})$ and $p_{w,c}(\mathbf{v}, \widetilde{\mathbf{h}})$ as follows:

$$\begin{aligned} p(\mathbf{v}, \mathbf{h}) &\propto z(\mathbf{v}, \mathbf{h}) \\ p_{w,c}(\mathbf{v}, \widetilde{\mathbf{h}}) &\propto \exp\left(\widetilde{\mathbf{h}}^T \widetilde{W} \mathbf{v} + B^T \mathbf{v} + \widetilde{C}^T \widetilde{\mathbf{h}}\right) \\ &\propto z(\mathbf{v}, \mathbf{h}) \exp\left(h_{n+1} w^T \mathbf{v} + c h_{n+1}\right) \end{aligned}$$

If $c = -\infty$, $p_{w,c}(\mathbf{v}, \widetilde{\mathbf{h}}) = 0$ if $h_{n+1} = 1$. Thus, we can discard all terms where $h_{n+1} = 1$, keeping only those where $h_{n+1} = 0$. Marginalizing over the hidden units, we have:

$$\begin{aligned} p(\mathbf{v}) &= \frac{\sum_{\mathbf{h}} z(\mathbf{v}, \mathbf{h})}{\sum_{\mathbf{h}^{(0)}, \mathbf{v}^0} z(\mathbf{v}^0, \mathbf{h}^{(0)})} \\ p_{w,-\infty}(\mathbf{v}) &= \frac{\sum_{\widetilde{\mathbf{h}}} z(\mathbf{v}, \mathbf{h}) \exp\left(h_{n+1} w^T \mathbf{v} + c h_{n+1}\right)}{\sum_{\widetilde{\mathbf{h}^{(0)}}, \mathbf{v}^0} z(\mathbf{v}^0, \mathbf{h}^{(0)}) \exp\left(h^{(0)}_{n+1} w^T \mathbf{v} + c h^{(0)}_{n+1}\right)} \\ &= \frac{\sum_{\mathbf{h}} z(\mathbf{v}, \mathbf{h}) \exp(0)}{\sum_{\mathbf{h}^{(0)}, \mathbf{v}^0} z(\mathbf{v}^0, \mathbf{h}^{(0)}) \exp(0)} \\ &= p(\mathbf{v}) \end{aligned}$$

□

We now state the main theorem.

Theorem 2.3. *Let p_0 be an arbitrary distribution over $\{0,1\}^n$ and let R_p be an RBM with marginal distribution p over the visible units such that $KL(p_0||p) > 0$. Then there exists an RBM $R_{p_{w,c}}$ composed of R_p and an additional hidden unit with parameters (w, c) whose marginal distribution $p_{w,c}$ over the visible units achieves $KL(p_0||p_{w,c}) < KL(p_0||p)$.*

Proof in appendix.

2.2 A Huge Model can Represent Any Distribution

The second set of results are for the limit case when the number of hidden units is very large, so that we can represent any discrete distribution exactly.

Theorem 2.4. *Any distribution over $\{0,1\}^n$ can be approximated arbitrarily well (in the sense of the KL divergence) with an RBM with $k+1$ hidden units where k is the number of input vectors whose probability is not 0.*

Proof sketch (Universal approximator property). We constructively build an RBM with as many hidden units as the number of input vectors whose probability is strictly positive. Each hidden unit will be assigned to one input vector. Namely, when $\mathbf{v}_i$ is the visible units vector, all hidden units have a probability 0 of being on except the one corresponding to $\mathbf{v}_i$ which has a probability $\text{sigm}(\lambda_i)$ of being on. The value of λ_i is directly tied with $p(\mathbf{v}_i)$. On the other hand, when all hidden units are off but the i^{th} one, $p(\mathbf{v}_i|\mathbf{h}) = 1$. With probability $1 - \text{sigm}(\lambda_i)$, all the hidden units are turned off, which yields independent draws of the visible units. The proof consists in finding the appropriate weights (and values λ_i) to yield that behaviour. □

Proof in appendix.

3 Representational power of Deep Belief Networks

3.1 Background on Deep Belief Networks

A DBN with ℓ layers models the joint distribution between observed variables v_j and ℓ hidden layers $\mathbf{h}^{(k)}, k = 1, \ldots, \ell$ made of binary units $h_i^{(k)}$ (here all binary variables), as follows:

$$p(\mathbf{v}, \mathbf{h}^{(1)}, \mathbf{h}^{(2)}, \ldots, \mathbf{h}^{(\ell)}) = P(\mathbf{v}|\mathbf{h}^{(1)})P(\mathbf{h}^{(1)}|\mathbf{h}^{(2)}) \ldots P(\mathbf{h}^{(\ell-2)}|\mathbf{h}^{(\ell-1)})p(\mathbf{h}^{(\ell-1)}, \mathbf{h}^{(\ell)})$$

Denoting $\mathbf{v} = \mathbf{h}^{(0)}$, $b^{(k)}$ the bias vector of layer k and $W^{(k)}$ the weight matrix between layer k and layer $k+1$, we have:

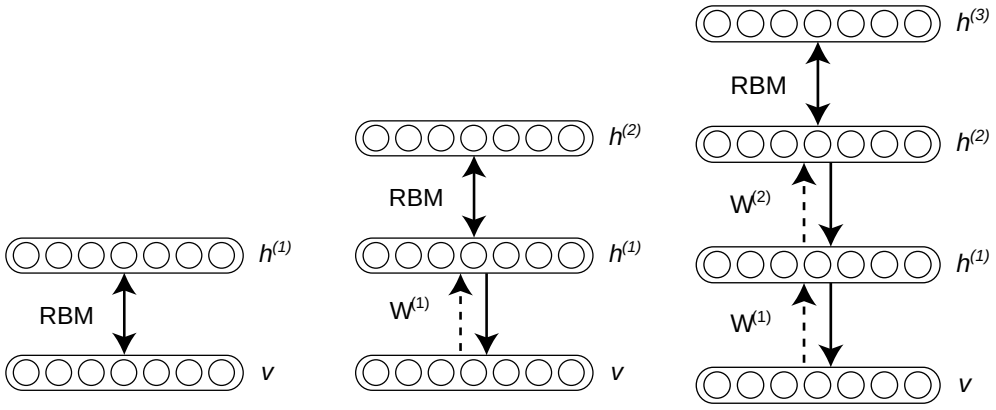
$$\begin{aligned} P(\mathbf{h}^{(k)}|\mathbf{h}^{(k+1)}) &= \prod_i P(\mathbf{h}_i^{(k)}|\mathbf{h}^{(k+1)}) \text{ (factorial conditional distribution)} \\ P(\mathbf{h}_i^{(k)} = 1|\mathbf{h}^{(k+1)}) &= \text{sigm}\left(b_i^{(k)} + \sum_j W_{ij}^{(k)} \mathbf{h}_j^{(k+1)}\right) \end{aligned} \quad (2)$$

and $p(\mathbf{h}^{(\ell-1)}, \mathbf{h}^{(\ell)})$ is an RBM.

The original motivation found in Hinton et al. (2006) for having a deep network versus a single hidden layer (i.e., a DBN versus an RBM) was that the representational power of an RBM would be too limited and that more capacity could be achieved by having more hidden layers. However, we have found here that an RBM with enough hidden units can model any discrete distribution. Another motivation for deep architectures is discussed in Bengio and Le Cun (2007) and Bengio et al. (2007): deep architectures can represent functions much more efficiently (in terms of number of required parameters) than shallow ones. In particular, theoretical results on circuit complexity theory prove that shallow digital circuits can be exponentially less efficient than deeper ones (Ajtai, 1983; Hastad, 1987; Allender, 1996). Hence the original motivation (Hinton et al., 2006) was probably right when one considers the restriction to reasonably sized models.

3.2 Trying to Anticipate a High-Capacity Top Layer

In the greedy training procedure of Deep Belief Networks proposed in (Hinton et al., 2006), one layer is added on top of the network at each stage, and only that top layer is trained (as an RBM, see figure 1). In that greedy phase, one does not take into account the fact that other layers will be added next. Indeed, while trying to optimize the weights, we restrict the marginal distribution over its hidden units to be the one induced by the RBM. On the contrary, when we add a new layer, that distribution (which is the marginal distribution over the visible units of the new RBM) does not have that restriction (but another one which is to be representable by an RBM of a given size). Thus, we might be able to better optimize the weights of the RBM, knowing that the marginal distribution over the hidden units will have more freedom when extra layers are added. This would lead to an alternative training criterion for DBNs.



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure 1: Greedy learning of an RBM. After each RBM has been trained, the weights are frozen and a new layer is added. The new layer is trained as an RBM.

Consider a 2-layer DBN ($\ell = 2$, that is with three layers in total). To train the weights between $\mathbf{h}^{(1)}$ and $\mathbf{h}^{(2)}$ (see figure 1), the greedy strategy maximizes a lower bound on the likelihood of the data (instead of the likelihood itself), called the **variational bound** (Hinton et al., 2006):

$$\begin{aligned} \log p(\mathbf{v}) \quad \geq \quad & \sum_{\mathbf{h}^{(1)}} Q(\mathbf{h}^{(1)}|\mathbf{v}) \left[\log p(\mathbf{h}^{(1)}) + \log P(\mathbf{v}|\mathbf{h}^{(1)}) \right] \\ & - \sum_{\mathbf{h}^{(1)}} Q(\mathbf{h}^{(1)}|\mathbf{v}) \log Q(\mathbf{h}^{(1)}|\mathbf{v}) \end{aligned} \tag{3}$$

where

- $Q(\mathbf{h}^{(1)}|\mathbf{v})$ is the posterior on hidden units $\mathbf{h}^{(1)}$ given visible vector $\mathbf{v}$, according to the first RBM model, and is determined by $W^{(1)}$. It is the assumed distribution used in the variational bound on the DBN likelihood.
- $p(\mathbf{h}^{(1)})$ is the marginal distribution over $\mathbf{h}^1$ in the DBN (thus induced by the second RBM, between $\mathbf{h}^{(1)}$ and $\mathbf{h}^{(2)}$)
- $P(\mathbf{v}|\mathbf{h}^{(1)})$ is the posterior over $\mathbf{v}$ given $\mathbf{h}^1$ in the DBN and in the first RBM, and is determined by $W^{(1)}$.

Once the weights of the first layer ($W^{(1)}$) are frozen, the only element that can be optimized is $p(\mathbf{h}^{(1)})$. We can show that there is an analytic formulation for the distribution $p^*(\mathbf{h}^{(1)})$ that maximizes this variational bound:

$$p^*(\mathbf{h}^{(1)}) = \sum_{\mathbf{v}} p_0(\mathbf{v}) Q(\mathbf{h}^{(1)}|\mathbf{v}) \tag{4}$$

where p_0 is the empirical distribution of input examples. One can sample from $p^*(\mathbf{h}^{(1)})$ by first randomly sampling a $\mathbf{v}$ from the empirical distribution and then propagating it stochastically through $Q(\mathbf{h}^{(1)}|\mathbf{v})$. Using theorem 2.4, there exists an RBM that can approximate this optimal distribution $p^*(\mathbf{h}^{(1)})$ arbitrarily well.
Using an RBM that achieves this "optimal" $p^*(\mathbf{h}^{(1)})$ (optimal in terms of the variational bound, but not necessarily with respect to the likelihood), we can determine the distribution represented by DBN. Let p_1 be the distribution one obtains when starting from p_0 clamped in the visible units of the lower layer ($\mathbf{v}$), sampling the hidden units $\mathbf{h}^{(1)}$ given $\mathbf{v}$ and then sampling a $\mathbf{v}$ given $\mathbf{h}^{(1)}$.

Proposition 3.1. *In a 2-layer DBN, using a second layer RBM achieving $p^*(\mathbf{h}^{(1)})$, the model distribution p is equal to p_1.*

This is equivalent to making one "up-down" in the first RBM trained.

Proof. We can write the marginal $p^*(\mathbf{h}^{(1)})$ by summing over hidden values $\tilde{\mathbf{h}}^0$:

$$p^*(\mathbf{h}^{(1)}) = \sum_{\tilde{\mathbf{h}}^0} p_0(\tilde{\mathbf{h}}^0) Q(\mathbf{h}^{(1)}|\tilde{\mathbf{h}}^0).$$

Thus, the probability of the data under the 2-layer DBN when the top-layer RBM achieves $p^*(\mathbf{h}^{(1)})$ is

$$\begin{aligned} p(\mathbf{h}^{(0)}) &= \sum_{\mathbf{h}^{(1)}} P(\mathbf{h}^{(0)}|\mathbf{h}^{(1)}) p^*(\mathbf{h}^{(1)}) \qquad (5) \\ &= \sum_{\tilde{\mathbf{h}}^0} p_0(\tilde{\mathbf{h}}^0) \sum_{\mathbf{h}^{(1)}} Q(\mathbf{h}^{(1)}|\tilde{\mathbf{h}}^0) P(\mathbf{h}^{(0)}|\mathbf{h}^{(1)}) \\ p(\mathbf{h}^{(0)}) &= p_1(\mathbf{h}^{(0)}) \qquad (6) \end{aligned}$$

The last line can be seen to be true by considering the stochastic process of first picking an $\tilde{\mathbf{h}}^0$ from the empirical distribution p_0, then sampling an $\mathbf{h}^{(1)}$ from $Q(\mathbf{h}^{(1)}|\tilde{\mathbf{h}}^0)$, and finally computing the probability of $\mathbf{h}^{(0)}$ under $P(\mathbf{h}^{(0)}|\mathbf{h}^{(1)})$ for that $\mathbf{h}^{(1)}$. □

Proposition 3.1 tells us that, even with the best possible model for $p(\mathbf{h}^{(1)}, \mathbf{h}^{(2)})$ according to the variational bound (i.e., the model that can achieve $p^*(\mathbf{h}^{(1)})$), we obtain a KL divergence between the DBN and the data equal to $KL(p_0||p_1)$. Hence if we train the 2nd level RBM to model the stochastic output of the 1st level RBM (as suggested in Hinton et al. (2006)), the best $KL(p_0||p)$ we can achieve with model p of the 2-level DBN cannot be better than $KL(p_0||p_1)$. Note that this result does not preclude that a better likelihood could be achieved with p if a better criterion is used to train the 2nd level RBM.
For $KL(p_0||p_1)$ to be 0, one should have $p_0 = p_1$. Note that a weight vector with this property would not only be a fixed point of $KL(p_0||p_1)$ but also of the likelihood and of contrastive divergence for the first-level RBM. $p_0 = p_1$ could have been obtained with a one-level DBN (i.e., a single RBM) that perfectly fits the data. This can happen when

the first RBM has infinite weights i.e., is deterministic, and just encodes $\mathbf{h}^{(0)} = \mathbf{v}$ in $\mathbf{h}^{(1)}$ perfectly. In that case the second layer $\mathbf{h}^{(2)}$ seems useless.

Does that mean that adding layers is useless? We believe the answer is no; first, even though having the distribution that maximizes the variational bound yields $p = p_1$, this does not mean that we cannot achieve $KL(p_0||p) < KL(p_0||p_1)$ with a 2-layer DBN (though we have no proof that it can be achieved either). Indeed, since the variational bound is not the quantity we truly want to optimize, another criterion might lead to a better model (in terms of the likelihood of the data). Besides that, even if adding layers does not allow us to perfectly fit the data (which might actually only be the case when we optimize the variational bound rather than the likelihood), the distribution of the 2-layer DBN is closer to the empirical distribution than is the first layer RBM (we do only one "up-down" Gibbs step instead of doing an infinite number of such steps). Furthermore, the extra layers allow us to regularize and hopefully obtain a representation in which even a very high capacity top layer (e.g., a memory-based non-parametric density estimator) could generalize well. This approach suggests using alternative criteria to train DBNs, that approximate $KL(p_0||p_1)$ and can be computed before $\mathbf{h}^{(2)}$ is added, but, unlike contrastive divergence, take into account the fact that more layers will be added later.

Note that computing $KL(p_0||p_1)$ exactly is intractable in an RBM because it involves summing over all possible values of the hidden vector $\mathbf{h}$. One could use a sampling or mean-field approximation (replacing the summation over values of the hidden unit vector by a sample or a mean-field value), but even then there would remain a double sum over examples:

$$\sum_{i=1}^{N} \frac{1}{N} \log \sum_{j=1}^{N} \frac{1}{N} \hat{P}(V^1 = v_i | V^0 = v_j)$$

where v_i denotes the i-th example and $\hat{P}(V^1 = v_i|V^0 = v_j)$ denotes an estimator of the probability of observing $V^1 = v_i$ at iteration 1 of the Gibbs chain (that is after a "up-down" pass) given that the chain is started from $V^0 = v_j$. We write $\hat{P}$ rather than P because computing P exactly might involve an intractable summation over all possible values of $\mathbf{h}$. In Bengio et al. (2007), the reconstruction error for training an auto-encoder corresponding to one layer of a deep network is $\log \hat{P}(V^1 = v_i|V^0 = v_i)$. Hence $\log \hat{P}(V^1 = v_i|V^0 = v_j)$ is like a reconstruction error when one tries to reconstruct or predict v_i according to $P(v_i|\mathbf{h})$ when starting from v_j, sampling $\mathbf{h}$ from $P(\mathbf{h}|v_j)$. This criterion is essentially the same as one already introduced in a different context in (Bengio, Larochelle, & Vincent, 2006b), where $\hat{P}(V^1 = v_i|V^0 = v_j)$ is computed deterministically (no hidden random variable is involved), and the inner sum (over v_j's) is approximated by using only the 5 nearest neighbours of v_i in the training set. However, the overall computation time in (Bengio et al., 2006b) is $O(N^2)$ because like most non-parametric learning algorithms it involves comparing all training examples with each other. In contrast, the contrastive divergence gradient estimator can be computed in $O(N)$ for a training set of size N.

To evaluate whether tractable approximations of $KL(p_0||p_1)$ would be worth investigating, we performed an experiment on a toy dataset and toy model where the computations are feasible. The data are 10-element bit vectors with patterns of 1, 2 or 3 consecutive

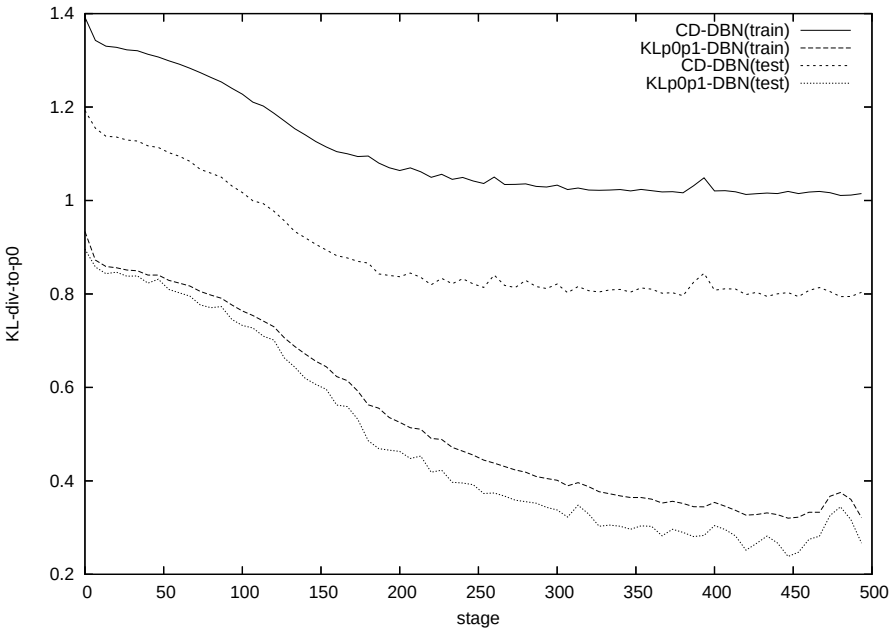


Figure 2: KL divergence w.r.t. number epochs after adding the 2nd level RBM, between empirical distribution p_0 (either training or test set) and (top curves) DBN trained greedily with contrastive divergence at each layer, or (bottom curves) DBN trained greedily with $KL(p_0||p_1)$ on the 1st layer, and contrastive divergence on the 2nd.

ones (or zeros) against a background of zeros (or ones), demonstrating simple shift invariance. There are 60 possible examples (p_0), 40 of which are randomly chosen to train first an RBM with 5 binomial hidden units, and then a 2-layer DBN. The remaining 20 are a test set. The second RBM has 10 hidden units (so that we could guarantee improvement of the likelihood by the addition of the second layer). The first RBM is either trained by contrastive divergence or to minimize $KL(p_0||p_1)$, using gradient descent and a learning rate of 0.1 for 500 epochs (parameters are updated after each epoch). Other learning rates and random initialization seeds gave similar results, diverged, or converged slower. The second RBM is then trained for the same number of epochs, by contrastive divergence with the same learning rate. Figure 2 shows the exact $KL(p_0||p)$ of the DBN p while training the 2nd RBM. The advantage of the $KL(p_0||p_1)$ training is clear. This suggests that future research should investigate tractable approximations of $KL(p_0||p_1)$.

3.3 Open Questions on DBN Representational Power

The results described in the previous section were motivated by the following question: since an RBM can represent any distribution, what can be gained by adding layers to a DBN, in terms of representational power? More formally, let R^n_ℓ be a Deep Belief Network with $\ell + 1$ hidden layers, each of them composed of n units. Can we say

something about the representational power of R_ℓ^n as ℓ increases? Denoting D_ℓ^n the set of distributions one can obtain with R_ℓ^n, it follows from the unfolding argument in Hinton et al. (2006) that $D_\ell^n \subseteq D_{\ell+1}^n$. The unfolding argument shows that the last layer of an ℓ-layer DBN corresponds to an infinite directed graphical model with tied weights. By untying the weights in the $(\ell+1)$-th RBM of this construction from those above, we obtain an $(\ell+1)$-layer DBN. Hence every element of D_ℓ^n can be represented in $D_{\ell+1}^n$. Two questions remain:

- do we have $D_\ell^n \subset D_{\ell+1}^n$, at least for $\ell = 1$?
- what is D_∞^n?

4 Conclusions

We have shown that when the number of hidden units is allowed to vary, Restricted Boltzmann Machines are very powerful and can approximate any distribution, eventually representing them exactly when the number of hidden units is allowed to become very large (possibly 2 to the number of inputs). This only says that parameter values exist for doing so, but it does not prescribe how to obtain them efficiently. In addition, the above result is only concerned with the case of discrete inputs. It remains to be shown how to extend that type of result to the case of continuous inputs.
Restricted Boltzmann Machines are interesting chiefly because they are the building blocks of Deep Belief Networks, which can have many layers and can theoretically be much more efficient at representing complicated distributions (Bengio & Le Cun, 2007). We have introduced open questions about the expressive power of Deep Belief Networks. We have not answered these questions, but in trying to do so, we obtained an apparently puzzling result concerning Deep Belief Networks: the best that can be achieved by adding a second layer (with respect to some bound) is limited by the first layer's ability to map the data distribution to something close to itself ($KL(p_0||p_1)$), and this ability is good when the first layer is large and models well the data. So why do we need the extra layers? We believe that the answer lies in the ability of a Deep Belief Network to generalize better by having a more compact representation. This analysis also suggests to investigate $KL(p_0||p_1)$ (or an efficient approximation of it) as a less greedy alternative to contrastive divergence for training each layer, because it would take into account that more layers will be added.

Acknowledgements

The authors would like to thank the following funding organizations for support: NSERC, MITACS, and the Canada Research Chairs. They are also grateful for the help and comments from Olivier Delalleau and Aaron Courville.

References

Ackley, D., Hinton, G., & Sejnowski, T. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, *9*.

Ajtai, M. (1983). $\sum_1^1$-formulae on finite structures. *Annals of Pure and Applied Logic*, *24*(1), 48.

Allender, E. (1996). Circuit complexity before the dawn of the new millennium. In *16th Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, pp. 1–18. Lecture Notes in Computer Science 1180.

Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H. (2007). Greedy layer-wise training of deep networks. In Schölkopf, B., Platt, J., & Hoffman, T. (Eds.), *Advances in Neural Information Processing Systems 19*. MIT Press.

Bengio, Y., Delalleau, O., & Le Roux, N. (2006a). The curse of highly variable functions for local kernel machines. In Weiss, Y., Schölkopf, B., & Platt, J. (Eds.), *Advances in Neural Information Processing Systems 18*, pp. 107–114. MIT Press, Cambridge, MA.

Bengio, Y., Larochelle, H., & Vincent, P. (2006b). Non-local manifold parzen windows. In Weiss, Y., Schölkopf, B., & Platt, J. (Eds.), *Advances in Neural Information Processing Systems 18*, pp. 115–122. MIT Press.

Bengio, Y., & Le Cun, Y. (2007). Scaling learning algorithms towards AI. In Bottou, L., Chapelle, O., DeCoste, D., & Weston, J. (Eds.), *Large Scale Kernel Machines*. MIT Press.

Carreira-Perpiñan, M., & Hinton, G. (2005). On contrastive divergence learning. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics, Jan 6-8, 2005, Savannah Hotel, Barbados*.

Hastad, J. T. (1987). *Computational Limitations for Small Depth Circuits*. MIT Press.

Hinton, G. E., Osindero, S., & Teh, Y. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*, 1527–1554.

Hinton, G. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation*, *14*, 1771–1800.

Hinton, G., & Salakhutdinov, R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*(5786), 504–507.

Hinton, G. (1999). Products of experts. In *Proceedings of the Ninth International Conference on Artificial Neural Networks (ICANN)*, Vol. 1, pp. 1–6.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*, 359–366.

Ranzato, M., Poultney, C., Chopra, S., & LeCun, Y. (2007). Efficient learning of sparse representations with an energy-based model. In Schölkopf, B., Platt, J., & Hoffman, T. (Eds.), *Advances in Neural Information Processing Systems 19*. MIT Press.

Salakhutdinov, R., & Hinton, G. (2007). Learning a nonlinear embedding by preserving class neighbourhood structure. In *To Appear in Proceedings of AISTATS'2007*.

Tesauro, G. (1992). Practical issues in temporal difference learning. *Machine Learning*, *8*, 257–277.

Welling, M., Rosen-Zvi, M., & Hinton, G. (2005). Exponential family harmoniums with an application to information retrieval. In Saul, L., Weiss, Y., & Bottou, L. (Eds.), *Advances in Neural Information Processing Systems 17*. MIT Press.

5 Appendix

5.1 Proof of Lemma 2.1

Proof. Denoting $\widetilde{\mathbf{h}} = \begin{bmatrix} \mathbf{h} \\ h_{n+1} \end{bmatrix}$, $\widetilde{W} = \begin{bmatrix} W \\ w^T \end{bmatrix}$ and $\widetilde{C} = \begin{bmatrix} C \\ c \end{bmatrix}$ where w^T denotes the transpose of w and introducing $z(\mathbf{v}, \mathbf{h}) = \exp(\mathbf{h}^T W \mathbf{v} + B^T \mathbf{v} + C^T \mathbf{h})$, we can express $p(\mathbf{v}, \mathbf{h})$ and $p_{w,c}(\mathbf{v}, \widetilde{\mathbf{h}})$ as follows:

$$\begin{aligned} p(\mathbf{v}, \mathbf{h}) &\propto z(\mathbf{v}, \mathbf{h}) \\ p_{w,c}(\mathbf{v}, \widetilde{\mathbf{h}}) &\propto \exp\left(\widetilde{\mathbf{h}}^T \widetilde{W} \mathbf{v} + B^T \mathbf{v} + \widetilde{C}^T \widetilde{\mathbf{h}}\right) \\ &\propto z(\mathbf{v}, \mathbf{h}) \exp\left(h_{n+1} w^T \mathbf{v} + c h_{n+1}\right) \end{aligned}$$

Expanding the expression of $p_{w,c}(\mathbf{v})$ and regrouping the terms similar to the expression of $p(\mathbf{v})$, we get:

$$\begin{aligned} p_{w,c}(\mathbf{v}) &= \frac{\sum_{\widetilde{\mathbf{h}}} \exp\left(\mathbf{h}^T W \mathbf{v} + h_{n+1} w^T \mathbf{v} + B^T \mathbf{v} + C^T \mathbf{h} + c h_{n+1}\right)}{\sum_{\widetilde{\mathbf{h}^{(0)}}, \mathbf{v}^0} \exp\left(\mathbf{h}^{(0)^T} W \mathbf{v}^0 + h_{n+1}^{(0)} w^T \mathbf{v}^0 + B^T \mathbf{v}^0 + C^T \mathbf{h}^{(0)} + c h_{n+1}^{(0)}\right)} \\ &= \frac{\sum_{\mathbf{h}} z(\mathbf{v}, \mathbf{h}) \left(1 + \exp\left(w^T \mathbf{v} + c\right)\right)}{\sum_{\mathbf{h}^{(0)}, \mathbf{v}^0} z(\mathbf{v}^0, \mathbf{h}^{(0)}) \left(1 + \exp\left(w^T \mathbf{v}^0 + c\right)\right)} \\ &= \frac{\left(1 + \exp\left(w^T \mathbf{v} + c\right)\right) \sum_{\mathbf{h}} z(\mathbf{v}, \mathbf{h})}{\sum_{\mathbf{v}^0} \left(1 + \exp\left(w^T \mathbf{v}^0 + c\right)\right) \sum_{\mathbf{h}^0} z(\mathbf{v}^0, \mathbf{h}^{(0)})} \end{aligned}$$

But $\sum_h z(\mathbf{v}, \mathbf{h}) = p(\mathbf{v}) K$ with $K = \sum_{\mathbf{v}, \mathbf{h}} z(\mathbf{v}, \mathbf{h})$. Thus,

$$p_{w,c}(\mathbf{v}) = \frac{\left(1 + \exp\left(w^T \mathbf{v} + c\right)\right) p(\mathbf{v})}{\sum_{\mathbf{v}^0} \left(1 + \exp\left(w^T \mathbf{v}^0 + c\right)\right) p(\mathbf{v}^0)}$$

which does not depend on our particular choice of R_p (since it does only depend on p). □

5.2 Proof of theorem 2.3

Proof. Expanding the expression of $p_{w,c}(\mathbf{v})$ and regrouping the terms similar to the expression of $p(\mathbf{v})$, we get:

$$\begin{aligned} p_{w,c}(\mathbf{v}) &= \frac{\sum_{\widetilde{\mathbf{h}}} \exp\left(\mathbf{h}^T W \mathbf{v} + h_{n+1} w^T \mathbf{v} + B^T \mathbf{v} + C^T \mathbf{h} + c h_{n+1}\right)}{\sum_{\widetilde{\mathbf{h}^{(0)}}, \mathbf{v}^0} \exp\left(\mathbf{h}^{(0)^T} W \mathbf{v}^0 + h_{n+1}^{(0)} w^T \mathbf{v}^0 + B^T \mathbf{v}^0 + C^T \mathbf{h}^{(0)} + c h_{n+1}^{(0)}\right)} \\ &= \frac{\sum_{\mathbf{h}} z(\mathbf{v}, \mathbf{h}) \left(1 + \exp\left(w^T \mathbf{v} + c\right)\right)}{\sum_{\mathbf{h}^{(0)}, \mathbf{v}^0} z(\mathbf{v}^0, \mathbf{h}^{(0)}) \left(1 + \exp\left(w^T \mathbf{v}^0 + c\right)\right)} \\ &= \frac{\left(1 + \exp\left(w^T \mathbf{v} + c\right)\right) \sum_{\mathbf{h}} z(\mathbf{v}, \mathbf{h})}{\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} \left(1 + \exp\left(w^T \mathbf{v}^0 + c\right)\right) z(\mathbf{v}^0, \mathbf{h}^{(0)})} \end{aligned}$$

Therefore, we have:

$$\begin{aligned}
KL(p_0||p_{w,c}) &= \sum_{\mathbf{v}} p_0(\mathbf{v}) \log p_0(\mathbf{v}) - \sum_{\mathbf{v}} p_0(\mathbf{v}) \log p_{w,c}(\mathbf{v}) \\
&= -H(p_0) - \sum_{\mathbf{v}} p_0(\mathbf{v}) \log \left(\frac{\left(1 + \exp\left(w^T\mathbf{v} + c\right)\right) \sum_{\mathbf{h}} z(\mathbf{v}, \mathbf{h})}{\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} \left(1 + \exp\left(w^T\mathbf{v}^0 + c\right)\right) z(\mathbf{v}^0, \mathbf{h}^{(0)})} \right) \\
&= -H(p_0) - \sum_{\mathbf{v}} p_0(\mathbf{v}) \log\left(1 + \exp\left(w^T\mathbf{v} + c\right)\right) - \sum_{\mathbf{v}} p_0(\mathbf{v}) \log \left(\sum_{\mathbf{h}} z(\mathbf{v}, \mathbf{h}) \right) \\
&\quad + \sum_{\mathbf{v}} p_0(\mathbf{v}) \log \left(\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} \left(1 + \exp\left(w^T\mathbf{v}^0 + c\right)\right) z(\mathbf{v}^0, \mathbf{h}^{(0)}) \right)
\end{aligned}$$

Assuming $w^T\mathbf{v} + c$ is a very large negative value for all $\mathbf{v}$, we can use the logarithmic series identity around 0 ($\log(1 + x) = x + o_{x \to 0}(x)$) for the second and the last term. The second term becomes[1]

$$\sum_{\mathbf{v}} p_0(\mathbf{v}) \log\left(1 + \exp\left(w^T\mathbf{v} + c\right)\right) = \sum_{\mathbf{v}} p_0(\mathbf{v}) \exp\left(w^T\mathbf{v} + c\right) + o_{c \to -\infty}\left(\exp(c)\right)$$

and the last term becomes

$$\begin{aligned}
\left(\sum_{\mathbf{v}} p_0(\mathbf{v}) \right) \log & \left(\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} \left(1 + \exp\left(w^T\mathbf{v}^0 + c\right)\right) z(\mathbf{v}^0, \mathbf{h}^{(0)}) \right) \\
&= \log \left(\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} z(\mathbf{v}^0, \mathbf{h}^{(0)}) \right) + \log \left(1 + \frac{\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} \exp\left(w^T\mathbf{v}^0 + c\right) z(\mathbf{v}^0, \mathbf{h}^{(0)})}{\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} z(\mathbf{v}^0, \mathbf{h}^{(0)})} \right) \\
&= \log \left(\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} z(\mathbf{v}^0, \mathbf{h}^{(0)}) \right) + \frac{\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} \exp\left(w^T\mathbf{v}^0 + c\right) z(\mathbf{v}^0, \mathbf{h}^{(0)})}{\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} z(\mathbf{v}^0, \mathbf{h}^{(0)})} + o_{c \to -\infty}\left(\exp(c)\right)
\end{aligned}$$

But

$$\begin{aligned}
\frac{\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} \exp\left(w^T\mathbf{v}^0 + c\right) z(\mathbf{v}^0, \mathbf{h}^{(0)})}{\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} z(\mathbf{v}^0, \mathbf{h}^{(0)})} &= \sum_{\mathbf{v}} \exp\left(w^T\mathbf{v} + c\right) \frac{\sum_{\mathbf{h}^{(0)}} z(\mathbf{v}, \mathbf{h}^{(0)})}{\sum_{\mathbf{v}^0, \mathbf{h}^{(0)}} z(\mathbf{v}^0, \mathbf{h}^{(0)})} \\
&= \sum_{\mathbf{v}} \exp\left(w^T\mathbf{v} + c\right) p(\mathbf{v})
\end{aligned}$$

[1] $o_{x \to \infty}()$ notation: $f(x) = o_{x \to +\infty}\left(g(x)\right)$ if $\lim_{x \to +\infty} \frac{f(x)}{g(x)}$ exists and equals 0.

Putting all terms back together, we have

$$\begin{aligned} KL(p_0||p_{w,c}) &= -H(p_0) - \sum_{\mathbf{v}} p_0(\mathbf{v}) \exp\left(w^T\mathbf{v} + c\right) + \sum_{\mathbf{v}} p(\mathbf{v}) \exp\left(w^T\mathbf{v} + c\right) + o_{c\to-\infty}\left(\exp(c)\right) \\ &- \sum_{\mathbf{v}} p_0(\mathbf{v}) \log\left(\sum_{\mathbf{h}} z(\mathbf{v},\mathbf{h})\right) + \log\left(\sum_{\mathbf{v}^0,\mathbf{h}^{(0)}} z(\mathbf{v}^0,\mathbf{h}^{(0)})\right) \\ &= KL(p_0||p) + \sum_{\mathbf{v}} \exp\left(w^T\mathbf{v} + c\right)\left(p(\mathbf{v}) - p_0(\mathbf{v})\right) + o_{c\to-\infty}\left(\exp(c)\right) \end{aligned}$$

Finally, we have

$$KL(p_0||p_{w,c}) - KL(p_0||p) = \exp(c) \sum_{\mathbf{v}} \exp\left(w^T\mathbf{v}\right)\left(p(\mathbf{v}) - p_0(\mathbf{v})\right) + o_{c\to-\infty}\left(\exp(c)\right) \quad (7)$$

The question now becomes: can we find a w such that $\sum_{\mathbf{v}} \exp\left(w^T\mathbf{v}\right)\left(p(\mathbf{v}) - p_0(\mathbf{v})\right)$ is negative?
As $p_0 \neq p$, there is a $\widehat{\mathbf{v}}$ such that $p(\widehat{\mathbf{v}}) < p_0(\widehat{\mathbf{v}})$. Then there exists a positive scalar a such that $\widehat{w} = a\left(\widehat{\mathbf{v}} - \frac{1}{2}e\right)$ (with $e = [1\ldots1]^T$) yields $\sum_{\mathbf{v}} \exp\left(\widehat{w}^T\mathbf{v}\right)\left(p(\mathbf{v}) - p_0(\mathbf{v})\right) < 0$.
Indeed, for $\mathbf{v} \neq \widehat{\mathbf{v}}$, we have

$$\begin{aligned} \frac{\exp(\widehat{w}^T\mathbf{v})}{\exp(\widehat{w}^T\widehat{\mathbf{v}})} &= \exp\left(\widehat{w}^T(\mathbf{v} - \widehat{\mathbf{v}})\right) \\ &= \exp\left(a\left(\widehat{\mathbf{v}} - \frac{1}{2}e\right)^T(\mathbf{v} - \widehat{\mathbf{v}})\right) \\ &= \exp\left(a\sum_i \left(\widehat{\mathbf{v}}_i - \frac{1}{2}\right)(\mathbf{v}_i - \widehat{\mathbf{v}}_i)\right) \end{aligned}$$

For i such that $\mathbf{v}_i - \widehat{\mathbf{v}}_i > 0$, we have $\mathbf{v}_i = 1$ and $\widehat{\mathbf{v}}_i = 0$. Thus, $\widehat{\mathbf{v}}_i - \frac{1}{2} = -\frac{1}{2}$ and the term inside the exponential is negative (since a is positive). For i such that $\mathbf{v}_i - \widehat{\mathbf{v}}_i < 0$, we have $\mathbf{v}_i = 0$ and $\widehat{\mathbf{v}}_i = 1$. Thus, $\widehat{\mathbf{v}}_i - \frac{1}{2} = \frac{1}{2}$ and the term inside the exponential is also negative. Furthermore, the terms come close to 0 as a goes to infinity. Since the sum can be decomposed as

$$\begin{aligned} \sum_{\mathbf{v}} \exp\left(\widehat{w}^T\mathbf{v}\right)\left(p(\mathbf{v}) - p_0(\mathbf{v})\right) &= \exp(\widehat{w}^T\widehat{\mathbf{v}})\left(\sum_{\mathbf{v}} \frac{\exp\left(\widehat{w}^T\mathbf{v}\right)}{\exp(\widehat{w}^T\widehat{\mathbf{v}})}\left(p(\mathbf{v}) - p_0(\mathbf{v})\right)\right) \\ &= \exp(\widehat{w}^T\widehat{\mathbf{v}})\left(p(\widehat{\mathbf{v}}) - p_0(\widehat{\mathbf{v}}) + \sum_{\mathbf{v}\neq\widehat{\mathbf{v}}} \frac{\exp\left(\widehat{w}^T\mathbf{v}\right)}{\exp(\widehat{w}^T\widehat{\mathbf{v}})}\left(p(\mathbf{v}) - p_0(\mathbf{v})\right)\right) \end{aligned}$$

we have[2]

$$\sum_{\mathbf{v}} \exp\left(\widehat{w}^T\mathbf{v}\right)\left(p(\mathbf{v}) - p_0(\mathbf{v})\right) \sim_{a\to+\infty} \exp(\widehat{w}^T\widehat{\mathbf{v}})(p(\widehat{\mathbf{v}}) - p_0(\widehat{\mathbf{v}})) < 0.$$

[2] $\sim_{x\to\infty}$ notation: $f(x) \sim_{x\to+\infty} g(x)$ if $\lim_{x\to+\infty} \frac{f(x)}{g(x)}$ exists and equals 1.

Therefore, there is a value $\widehat{a}$ such that, if $a > \widehat{a}$, $\sum_{\mathbf{v}} \exp\left(w^T\mathbf{v}\right)\left(p(\mathbf{v}) - p_0(\mathbf{v})\right) < 0$. This concludes the proof. □

5.3 Proof of theorem 2.4

Proof. In the former proof, we had

$$p_{w,c}(\mathbf{v}) = \frac{\left(1 + \exp\left(w^T\mathbf{v} + c\right)\right)\sum_{\mathbf{h}} z(\mathbf{v},\mathbf{h})}{\sum_{\mathbf{v}^0,\mathbf{h}^{(0)}}\left(1 + \exp\left(w^T\mathbf{v}^0 + c\right)\right) z(\mathbf{v}^0,\mathbf{h}^{(0)})}$$

Let $\tilde{\mathbf{v}}$ be an arbitrary input vector and $\widehat{w}$ be defined in the same way as before, i.e. $\widehat{w} = a\left(\tilde{\mathbf{v}} - \frac{1}{2}\right)$.
Now define $\widehat{c} = -\widehat{w}^T\tilde{\mathbf{v}} + \lambda$ with $\lambda \in \mathbb{R}$. We have:

$$\begin{aligned}\lim_{a\to\infty} 1 + \exp(\widehat{w}^T\mathbf{v} + \widehat{c}) = &\quad 1 \quad &\text{for } \mathbf{v} \neq \tilde{\mathbf{v}}\\ 1 + \exp(\widehat{w}^T\tilde{\mathbf{v}} + \widehat{c}) = &\quad 1 + \exp(\lambda)\end{aligned}$$

Thus, we can see that, for $\mathbf{v} \neq \tilde{\mathbf{v}}$:

$$\begin{aligned}\lim_{a\to\infty} p_{\widehat{w},\widehat{c}}(\mathbf{v}) &= \frac{\sum_{\mathbf{h}} z(\mathbf{v},\mathbf{h})}{\sum_{\mathbf{v}^0\neq\tilde{\mathbf{v}},\mathbf{h}^{(0)}} z(\mathbf{v}^0,\mathbf{h}^{(0)}) + \sum_{\mathbf{h}^{(0)}}\left(1 + \exp\left(\widehat{w}^T\tilde{\mathbf{v}} + \widehat{c}\right)\right) z(\tilde{\mathbf{v}},\mathbf{h}^{(0)})}\\ &= \frac{\sum_{\mathbf{h}} z(\mathbf{v},\mathbf{h})}{\sum_{\mathbf{v}^0,\mathbf{h}^{(0)}} z(\mathbf{v}^0,\mathbf{h}^{(0)}) + \sum_{\mathbf{h}^{(0)}} \exp\left(\lambda\right) z(\tilde{\mathbf{v}},\mathbf{h}^{(0)})}\\ &= \frac{\sum_{\mathbf{h}} z(\mathbf{v},\mathbf{h})}{\sum_{\mathbf{v}^0,\mathbf{h}^{(0)}} z(\mathbf{v}^0,\mathbf{h}^{(0)})}\frac{1}{1 + \exp(\lambda)\frac{\sum_{\mathbf{h}^{(0)}} z(\tilde{\mathbf{v}},\mathbf{h}^{(0)})}{\sum_{\mathbf{v}^0,\mathbf{h}^{(0)}} z(\mathbf{v}^0,\mathbf{h}^{(0)})}}\end{aligned}$$

Remembering $p(\mathbf{v}) = \frac{\sum_{\mathbf{h}} z(\mathbf{v},\mathbf{h})}{\sum_{\mathbf{v}^0,\mathbf{h}^{(0)}} z(\mathbf{v}^0,\mathbf{h}^{(0)})}$, we have for $\mathbf{v} \neq \tilde{\mathbf{v}}$:

$$\lim_{a\to\infty} p_{\widehat{w},\widehat{c}}(\mathbf{v}) = \frac{p(\mathbf{v})}{1 + \exp(\lambda)p(\tilde{\mathbf{v}})} \tag{8}$$

Similarly, we can see that

$$\lim_{a\to\infty} p_{\widehat{w},\widehat{c}}(\tilde{\mathbf{v}}) = \frac{[1 + \exp(\lambda)]p(\tilde{\mathbf{v}})}{1 + \exp(\lambda)p(\tilde{\mathbf{v}})} \tag{9}$$

Depending on the value of λ, one can see that adding a hidden unit allows one to increase the probability of an arbitrary $\tilde{\mathbf{v}}$ and to uniformly decrease the probability of every other $\mathbf{v}$ by a multiplicative factor. However, one can also see that, if $p(\tilde{\mathbf{v}}) = 0$, then $p_{\widehat{w},\widehat{c}}(\tilde{\mathbf{v}}) = 0$ for all λ.
We can therefore build the desired RBM as follows. Let us index the $\mathbf{v}$'s over the integers from 1 to 2^n and sort them such that

$$p_0(\mathbf{v}_{k+1}) = \ldots = p_0(\mathbf{v}_{2^n}) = 0 < p_0(\mathbf{v}_1) \leq p_0(\mathbf{v}_2) \leq \ldots \leq p_0(\mathbf{v}_k)$$

Let us denote p^i the distribution of an RBM with i hidden units. We start with an RBM whose weights and biases are all equal to 0. The marginal distribution over the visible units induced by that RBM is the uniform distribution. Thus,

$$p^0(\mathbf{v}_1) = \ldots = p^0(\mathbf{v}_{2^n}) = 2^{-n}$$

We define $w_1 = a_1(\mathbf{v}_1 - \frac{1}{2})$ and $c_1 = -w_1^T\mathbf{v}_1 + \lambda_1$.
As shown before, we now have:

$$\begin{aligned}\lim_{a_1\to+\infty} p^1(\mathbf{v}_1) &= \frac{[1+\exp(\lambda_1)]2^{-n}}{1+\exp(\lambda_1)2^{-n}} \\ \lim_{a_1\to+\infty} p^1(\mathbf{v}_i) &= \frac{2^{-n}}{1+\exp(\lambda_1)2^{-n}} \quad \forall i \geq 2\end{aligned}$$

As we can see, we can set $p^1(\mathbf{v}_1)$ to a value arbitrarily close to 1, with a uniform distribution over $\mathbf{v}_2, \ldots, \mathbf{v}_{2^n}$. Then, we can choose λ_2 such that $\frac{p^2(\mathbf{v}_2)}{p^2(\mathbf{v}_1)} = \frac{p(\mathbf{v}_2)}{p(\mathbf{v}_1)}$. This is possible since we can arbitrarily increase $p^2(\mathbf{v}_2)$ while multiplying the other probabilities by a constant factor and since $\frac{p(\mathbf{v}_2)}{p(\mathbf{v}_1)} \geq \frac{p^1(\mathbf{v}_2)}{p^1(\mathbf{v}_1)}$. We can continue the procedure until obtaining $p^k(\mathbf{v}_k)$. The ratio $\frac{p^i(\mathbf{v}_j)}{p^i(\mathbf{v}_{j-1})}$ does not depend on the value of i as long as $i > j$ (because at each such step i, the two probabilities are multiplied by the same factor). We will then have

$$\begin{aligned}\frac{p^k(\mathbf{v}_k)}{p^k(\mathbf{v}_{k-1})} = \frac{p(\mathbf{v}_k)}{p(\mathbf{v}_{k-1})}, \quad \ldots \quad , \frac{p^k(\mathbf{v}_2)}{p^k(\mathbf{v}_1)} = \frac{p(\mathbf{v}_2)}{p(\mathbf{v}_1)} \\ p^k(\mathbf{v}_{k+1}) = \quad \ldots \quad = p^k(\mathbf{v}_{2^n})\end{aligned}$$

From that, we can deduce that $p^k(\mathbf{v}_1) = \nu_k p(\mathbf{v}_1), \ldots, p^k(\mathbf{v}_k) = \nu_k p(\mathbf{v}_k)$ with $\nu_k = 1 - (2^n - k)p^k(\mathbf{v}_{2^n})$.
We also have $\frac{p^k(\mathbf{v}_1)}{p^k(\mathbf{v}_{2^n})} = \frac{p^1(\mathbf{v}_1)}{p^1(\mathbf{v}_{2^n})} = 1 + \exp(\lambda_1)$.
Thus, $p^k(\mathbf{v}_1) = p(\mathbf{v}_1)[1 - (2^n - k)p^k(\mathbf{v}_{2^n})] = (1 + \exp(\lambda_1))p^k(\mathbf{v}_{2^n})$.
Solving the above equations, we have

$$p^k(\mathbf{v}_i) = \frac{p(\mathbf{v}_1)}{1+\exp(\lambda_1)+p(\mathbf{v}_1)(2^n-k)} \quad \text{for } i > k \tag{10}$$

$$p^k(\mathbf{v}_i) = p(\mathbf{v}_i)\frac{1+\exp(\lambda_1)}{1+\exp(\lambda_1)+p(\mathbf{v}_1)(2^n-k)} \quad \text{for } i \leq k \tag{11}$$

Using the logarithmic series identity around 0 ($\log(1+x) = x + o_{x\to 0}(x)$) for $KL(p||p^k)$ when λ_1 goes to infinity, we have

$$KL(p||p^k) = \sum_i p(\mathbf{v}_i)\frac{(2^n-k)p(\mathbf{v}_i)}{1+\exp(\lambda_1)} + o(\exp(-\lambda_1)) \underset{\lambda_1\to\infty}{\longrightarrow} 0 \tag{12}$$

This concludes the proof. □