# A Comparison of Neural Network Architectures

**Guido Montúar** MONTUFAR@MIS.MPG.DE
Max Planck Institute for Mathematics in the Sciences, 04103 Leipzig, Germany

## Abstract

We collect and extend theoretical results on the representational power of various artificial neural networks. We focus on universal approximation bounds for shallow and deep stochastic feedforward networks and layered Boltzmann machines in the probabilistic and discriminative settings.

## 1. Introduction

The importance of deep architectures in modern AI research is hard to overlook. At the same time, it has been observed more or less consistently that with sufficient data, any sufficiently powerful model will perform well. Hence the choice of a model has been more and more critically influenced by the simplicity with which it can be implemented and trained. An example is the choice of feedforward networks over undirected networks, or the choice of simpler activation functions, such as rectified linear, over more complicated ones. We know from statistical learning theory that with sufficient data any model will generalize well, but usually a less complex model will have a lower variance. However, model selection is a very challenging theoretical problem for the kinds of models that are used in deep learning practice. Computing the learning coefficient and estimating the generalization error of singular models (models with hidden variables) is notoriously difficult, with a few exceptions of course.

The goal of this note is much more humble. We are interested in how the representational power of deep artificial neural networks compares with that of shallow neural networks in the cases of undirected and directed connections between the layers. To this end we collect and extend a number of theoretical results addressing the representational power of some of these models and discuss some of the proof ingredients and the insights that they provide. Some of the results are known, some are very recent, and some are new. Surprisingly, it does not take long to arrive at open questions and cases that have not been covered before, even for very natural models. We fill some of the missing details in the periodic table of artificial neural networks and point at some cases that could be addressed next. The idea is to provide a concise overview of representation bounds, which may give indications about the complexity of the different network architectures. Having said this, we certainly do not attempt to give a full account on the subject. We hope to soon fill in more details and include important references that have been omitted at the current stage.

At an intuitive level, undirected networks are expected to be more powerful than directed networks, as the latter seem to be encompassed by the former. This intuition is not straightforward to verify concretely, but some recent work has provided theoretical clues in the context of deep Boltzmann machines and feedforward networks with sigmoid activation probabilities. We will try to take this a bit further and sketch a few details. The theoretical analysis of feedforward networks in the literature is most frequently focused on the deterministic case. Hence, in order to obtain a good picture we will elaborate a bit on the approximation of stochastic functions (that is, Markov kernels or simply conditional probability distributions) by stochastic feedforward networks and the minimal number of hidden units or layers that is sufficient for this purpose. While universal approximation is not typically aimed at in any practical setting, since it requires an astronomic number of hidden units, investigating the minimal requirements for achieving it will serve as our platform to illuminate interesting and distinctive features of the different models.

In Section 2 we discuss the representation of probability distributions by restricted Boltzmann machines, deep belief networks, and deep Boltzmann machines. In Section 3 we discuss the representation of stochastic maps by conditional restricted Boltzmann machines, two types of conditional deep Boltzmann machines, and two types of stochastic feedforward networks with sigmoid activation probabilities. In Section 4 we discuss the representation of deterministic maps by deterministic feedforward networks with linear threshold units and by conditional restricted Boltzmann machines. In Section 5 we give a synopsis of the results. In Section 6 we offer a discussion and consider a few

*Deep Learning Workshop at the $32^{nd}$ International Conference on Machine Learning*, Lille, France, 2015. 

ideas. We collect the technical definitions of all considered models in Appendix A.

## 2. Probability Distributions

We focus on observations from the set $\{0,1\}^n$.

### 2.1. Restricted Boltzmann Machine

The restricted Boltzmann machine is one of the simplest types of Boltzmann machines which have the universal approximation property. It can be regarded as a product of experts, with each expert being a mixture of two factorizing distributions. See Figure 1.

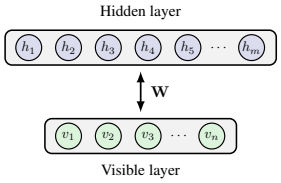


*Figure 1.* Architecture of a restricted Boltzmann machine.

**Theorem 1** (Montúfar & Ay 2011)**.** *A restricted Boltzmann machine with $n$ visible binary units and $m$ hidden binary units is a universal approximator of probability distributions if $m \geq 2^{n-1} - 1$ and only if $m \geq \frac{2^n}{(n+1)} - 1$.*

This result is based on showing that each hidden unit can be used independently to model the probability mass assigned to a pair of adjacent binary vectors. This approach is a refinement of previous work were each hidden unit was assigned to a single vector (Le Roux & Bengio, 2008).

Another approach (Younes, 1996) is based on showing that each hidden unit can be used to model the pure interaction between a group of visible units (the coefficient of a term $\prod_{i\in\lambda} v_i$ in the logarithm of the probability distribution). This leads to weaker universal approximation bounds, but it allows to describe some interesting complementary classes of distributions that can be represented by restricted Boltzmann machines with relatively few hidden units. The intuition of this approach is illustrated in Figure 2.

**Theorem 2.** *A restricted Boltzmann machine with $n$ visible binary units and $m$ hidden binary units can approximate any Markov random filed involving $m$ or less pure higher order interactions arbitrarily well.*

### 2.2. Deep Belief Network

The deep belief network (Hinton et al., 2006) is the architecture to have pioneered the deep revolution taking place in recent years. As a generative model, the deep belief network stacks a restricted Boltzmann machine on top of a stochastic feedforward network, as shown in Figure 3.

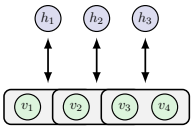

*Figure 2.* Restricted Boltzmann machine for modeling interactions. Each hidden unit corresponds to a term in the free energy of the restricted Boltzmann machine, which can be used to model the interaction of a group of visible units.

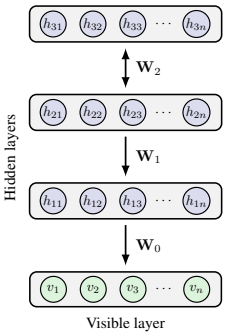


*Figure 3.* Architecture of a deep belief network.

**Theorem 3** (Montúfar & Ay 2011)**.** *A deep belief network with $n$ visible binary units and $L$ hidden layers of $n$ binary units each is a universal approximator if $L \geq \frac{2^{n'}}{2(n'-\log_2(n')-1)}$ for some $n \leq n' = 2^k + k + 1$, $k \in \mathbb{N}$, and only if $L \geq \frac{2^n}{n(n+1)} - \frac{1}{n}$.*

This result is based on showing that each feedforward layer can transform its input distribution in various ways, which amounts to modeling the probability mass that is assigned to roughly $2n$ observations. This approach is a refinement of previous work where each layer was assigned to fewer vectors (Sutskever & Hinton, 2008; Le Roux & Bengio, 2010). The transformations that are computable by a single feedforward is a manifold of Markov kernels. Some of the properties of this set are still not sufficiently well understood. In fact, the same is true for layers of linear threshold units.

### 2.3. Deep Boltzmann Machine

The deep Boltzmann machine is an attractive choice of a deep architecture based on an exponential family. It is the natural generalization of a restricted Boltzmann machine to a deep architecture. This architecture is illustrated in Figure 4.

**Theorem 4** (Montúfar 2015)**.** *A Deep Boltzmann Machine with a visible layer of $n$ binary units and $L$ hidden layers of $n$ binary units each is a universal approximator if $L \geq \frac{2^{n'}}{2(n'-\log_2(n')-1)}$, for some $n \leq n' := 2^k + k + 1$, $k \in \mathbb{N}$,*

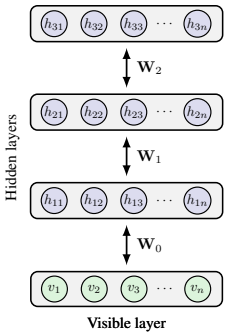


*Figure 4.* Architecture of a deep Boltzmann machine.

*and only if* $L \geq \frac{2^n}{n(n+1)} - \frac{1}{n}$.

This result is fairly recent and settles some intuitions about deep Boltzmann machines being as powerful as deep belief networks. It is based on showing that deep Boltzmann machines can indeed represent many of the probability distributions that can be represented by deep belief networks of the same size. The main argument is that it is possible to effectively fix the marginal distribution represented at some intermediate layer of the network, regardless of how the parameters are chosen in the lower part of the network. Once this is achieved, the lower part of the network can be used to model a feedforward transformation of that intermediate marginal distribution. In this particular situation; that is, for these particular choices of fixed intermediate marginals, the deep Boltzmann machine and the deep belief network coincide.

## 3. Stochastic Maps

A stochastic map from $\{0,1\}^k$ to $\{0,1\}^n$ assigns a probability distribution $p(y|x)$ over $y \in \{0,1\}^n$ to each input vector $x \in \{0,1\}^k$.

Boltzmann machines can be used to define stochastic maps by means of clamping the states of some units to the input values $x$, and taking the resulting conditional probability distribution over the states $y$ of some other units as the output distribution.

While it is straightforward to translate results on the representation of probability distributions to the conditional case, this naive approach will not account for the important fact that a conditional model does not need to model the input distributions.

### 3.1. Conditional Restricted Boltzmann Machine

The conditional restricted Boltzmann machine is a direct generalization of the restricted Boltzmann machine to model stochastic maps. It is defined by dividing the visible layer of a restricted Boltzmann machine into input and output units, as shown in Figure 5.

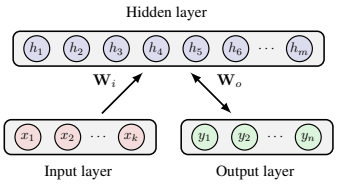


*Figure 5.* Architecture of a conditional restricted Boltzmann machine.

**Theorem 5** (Montúfar et al. 2014)**.** *A conditional restricted Boltzmann machine with an input layer of $k$ binary units, a hidden layer of $m$ binary units, and an output layer of $n$ binary units is a universal approximator of stochastic maps if $m \geq \frac{1}{2}2^k(2^n-1)$ and $k \geq 1$, or $m \geq \frac{3}{8}2^k(2^n-1)+1$ and $k \geq 3$, or $m \geq \frac{1}{4}2^k(2^n-1+1/30)$ and $k \geq 21$, and only if $m \geq \frac{2^k(2^n-1)-n}{(k+n+1)}$.*

There exist tighter bounds for $k \geq 21$, but we omit the details here. The result is based on showing that each hidden unit can be used to model the probability mass assigned to any particular output vector for up to $k$ different input vectors simultaneously. Ultimately the reason why this is possible is that for all input vectors the corresponding output distributions are normalized independently. Hence this approach is able to disregard the input distribution, in the way that is desirable when analyzing conditional models.

### 3.2. Shallow Stochastic Feedforward Network

Shallow feedforward networks have been discussed extensively in the literature. The vast majority of papers address the representation of deterministic functions. We shall briefly discuss the stochastic setting. We consider sigmoid activation probabilities. A shallow feedforward network is illustrated in Figure 6.

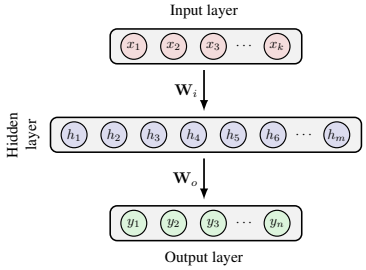


*Figure 6.* Architecture of a shallow stochastic feedforward network.

**Theorem 6.** *A shallow stochastic feedforward network with an input layer of $k$ binary units, a hidden layer of $m$ binary units, and an output layer of $n$ binary units*

*is a universal approximator of stochastic maps if* $m \geq 2^{k-1}(2^n - 1)$ *and only if* $m \geq \frac{2^k(2^n-1)-n}{k+n+1}$.

One way of proving this result is as follows. Associate a block of $2^n-1$ hidden units to each of $2^{k-1}$ disjoint pairs of adjacent input vectors. It is possible to finds weights such that, for each input vector, the associated block of hidden units has an arbitrary factorizing probability distribution, while all other hidden units are zero with probability close to one. The second layer can be defined in such a way that it integrates the probability of all vectors at the active block, whose largest entry with value one coincides, into the probability of an output vector. What this does is to map the set of factorizing distributions over length $N = 2^n - 1$ binary vectors, to the set of all distributions over length $n$ binary vectors. The construction is illustrated in Figure 7. It is interesting to note that the weights of the second layer can be kept fixed and only the weights of the first layer need to be adjusted in order to approximate any particular stochastic map.

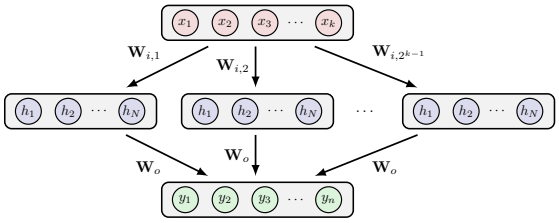

*Figure 7.* Illustration of the construction used for proving Theorem 6. The set of input vectors is divided into $2^{k-1}$ disjoint pairs. Each pair has an associated block of $N = 2^n - 1$ hidden units.

### 3.3. Deep Stochastic Feedforward Network

A natural question that arises is whether one can exchange the width of the hidden layer of a feedforward network for depth, to obtain a model as the one shown in Figure 8. This is indeed possible. One way of doing this is by assigning an active column of hidden layers of width $n$ to each possible input vector. This would reduce the width $2^{k-1}(2^n - 1)$ of the shallow network to $2^k n$, which is at least not exponential in $n$, although it still is exponential in $k$.

**Theorem 7.** *A deep stochastic feedforward network with an input layer of* $k$ *binary units,* $L$ *hidden layers of* $n2^k$ *binary units, and an output layer of* $n$ *binary units is a universal approximator of stochastic maps if* $L \geq 2^n$ *and only if* $L \geq \frac{2^n-1+n^2+n(k+1)}{n(2^k n+1)}$.

A proof can be given as follows. As for the shallow network, one finds weights such that, for each input vector, the associated block in the first hidden layer has an arbitrary factorizing distribution, while all other units in the first hidden layer are zero with probability arbitrarily close to one. Choosing an appropriate factorizing distribution on that block, one can define $L'$ feeforward layers of width $n$, which map that distribution into any arbitrary distribution over length $n$ binary vectors. A loose sufficiency bound is $L' \geq 1 + 2 + 2^2 + \cdots + 2^{n-1} = 2^n - 1$. This corresponds to using each layer to model one observation. Finally, the output of the active column can be passed unmodified to the output layer of the entire network.

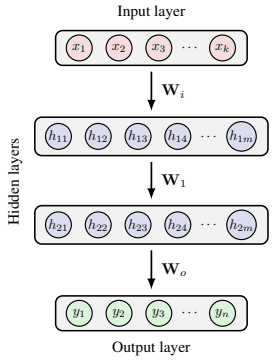


*Figure 8.* Architecture of a deep stochastic feedforward network.

The proof given above constructs a deep network with $2^k$ independent columns of hidden layers. This is a rather simplistic construction and hence we expect that the result can be improved. Nevertheless, the result shows that it is possible to trade the exponential width, with respect to the number of output units, for exponential depth.

An interesting question that is left open at this point is to what extent one can also trade the exponential width with respect to the number of input units for depth.

### 3.4. Conditional Deep Boltzmann Machine

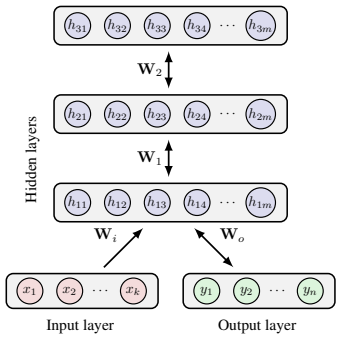


*Figure 9.* Architecture of a conditional deep Boltzmann machine with input and output units at the bottom layer.

One way of using deep Boltzmann machines to define stochastic maps is by dividing the visible units in the bottom layer into input and output units, as shown in Figure 9.

**Theorem 8** (Montúfar 2015)**.** *A deep Boltzmann machine*

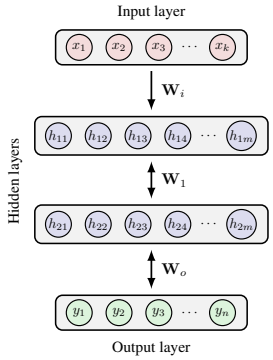


*Figure 10.* Architecture of a conditional deep Boltzmann machine with inputs at the top and outputs at the bottom.

*with a visible layer of $k$ input binary units and $n$ binary output units and $L$ hidden layers of $(k+n)$ units each is a universal approximator of stochastic maps, provided $L$ is as in Theorem 4.*

This result is a direct implication of Theorem 4, since $p(v) = p(x, y)$ stands in one-to-one relation with the pair $(p(x), p(y|x))$ (for strictly positive distributions). A refinement that ignores the input distribution is pending. Interestingly, this architecture is extremely narrow.

Another way of defining stochastic maps is to use the top layer as input and the bottom layer as output, as shown in Figure 10. We consider hidden layers of the same width as we did for the deep feedforward network.

**Theorem 9.** *A deep Boltzmann machine with an input layer of $k$ binary units, $L$ hidden layers of $n2^k$ binary units, and an output layer of $n$ binary units is a universal approximator of stochastic maps, provided $L$ is as in Theorem 7*

This is based on the ability of deep Boltzmann machines to represent certain types of transformations that can be represented by feedforward networks (Montúfar, 2015) and on the proof of Theorem 7. Any refinement of the latter will directly translate to a refinement of this result.

## 4. Deterministic Maps

A universal approximator of stochastic maps is also a universal approximator of deterministic maps. Indeed, every deterministic map $x \mapsto y = f(x)$ can be regarded as the special type of stochastic map $x \mapsto \delta_{f(x)}(y)$, where $\delta_{f(x)}$ is the Dirac delta assigning probability one to $y = f(x)$.

However, the set of deterministic maps is finite, and one can expect that representing it requires fewer hidden units than representing the set of all stochastic maps, which is infinite. In particular, the number of model parameters does not directly relate to the ability of a model to represent a finite set of functions. Instead, this is conceptually closer related to some sort of VC dimension.

### 4.1. Shallow Feedforward Linear Threshold Network

The stochastic feedforward networks considered in the previous sections define deterministic networks when all weights are multiplied by a large constant (assuming a generic choice of weights). In this limit the network becomes a linear threshold network, where each unit is either on or off with probability one.

**Theorem 10** (Wenzel et al. 2000)**.** *A shallow feedforward linear threshold network with an input layer of $k$ units, a hidden layer of $m$ units, and an output layer of $n$ units can represent all functions $\{0,1\}^k \to \{0,1\}^n$ if $m \geq 3 \cdot 2^{k-1-\lfloor \log_2(k+1) \rfloor}$ and only if $m \geq 2^{k/2} - \frac{k^2}{2}$.*

### 4.2. Conditional Restricted Boltzmann Machine

**Theorem 11** (Montúfar et al. 2014)**.** *A conditional restricted Boltzmann machine with an input layer of $k$ binary units, a hidden layer of $m$ binary units, and an output layer of $n$ binary units can approximate all functions $\{0,1\}^k \to \{0,1\}^n$ arbitrarily well if $m \geq \min\left\{2^k - 1, n \cdot 3 \cdot 2^{k-1-\lfloor \log_2(k+1) \rfloor}\right\}$ and only if $m \geq 2^{k/2} - \frac{(n+k)^2}{2n}$.*

The sufficiency bound is based on two observations. The first is that each hidden unit can be used to model the deterministic output of an input vector. The second is the interesting fact that, in the case of modeling deterministic maps, conditional restricted Boltzmann machines are at least as powerful as feedforward networks of the same size. Let us formulate this in some more detail.

**Theorem 12.** *A conditional restricted Boltzmann machine with $k$ input binary units, $m$ hidden binary units, and $n$ output binary units can approximate a given stochastic map arbitrarily well, whenever it can be represented by a feedforward network with $k$ input binary units, $m$ hidden linear threshold units, and $n$ output stochastic sigmoid units.*

In particular, the conditional restricted Boltzmann machine can represent any given deterministic function arbitrarily well, whenever it can be represented by a feedforward linear threshold network of the same size. On the other hand, it can be show that for a conditional restricted Boltzmann machine to approximate a given deterministic function arbitrarily well, this function has to satisfy a certain combinatorial constraints which are quite similar to those that apply for linear threshold networks.

## 5. Summary

Tables 1, 2, and 3 summarize the bounds on the number of hidden units and layers of universal approximators of

probability distributions, stochastic maps, and deterministic maps presented in the previous sections. In these tables, SFF stands for shallow feedforward, DFF stands for deep feedforward, and the other abbreviations have the obvious meanings.

| network | width | depth | Thm. |
|---|---|---|---|
| RBM | $2^{n-1}-1$ | 1 | 1 |
| DBN | $n$ | $\frac{2^n}{2(n-\log_2(n)-1)}$ | 3 |
| DBM | $n$ | $\frac{2^n}{2(n-\log_2(n)-1)}$ | 4 |

*Table 1.* Upper bounds on the minimal size of universal approximators of probability distributions on $\{0,1\}^n$.

| network | width | depth | Thm. |
|---|---|---|---|
| CRBM | $2^{k-1}(2^n-1)$ | 1 | 5 |
| SFF | $2^{k-1}(2^n-1)$ | 1 | 6 |
| DFF | $n2^k$ | $2^n$ | 7 |
| CDBM | $k+n$ | $\frac{2^{k+n}}{2(k+n-\log_2(k+n)-1)}$ | 8 |
| CDBM2 | $n2^k$ | $2^n$ | 9 |

*Table 2.* Upper bounds on the minimal size of universal approximators of stochastic maps from $\{0,1\}^k$ to $\{0,1\}^n$.

| network | width | depth | Thm. |
|---|---|---|---|
| CRBM | $\min\{2^k-1, \frac{3n}{k+2}2^k\}$ | 1 | 11 |
| SFF | $\frac{3n}{k+2}2^k$ | 1 | 7 |

*Table 3.* Upper bounds on the minimal size of universal approximators of deterministic maps from $\{0,1\}^k$ to $\{0,1\}^n$.

## 6. Discussion

The results of the previous sections lead us to the following observations. Consider the setting of representing stochastic maps. For the conditional restricted Boltzmann machine we have the sufficiency bound $2^{k-1}(2^n-1)$, and if the number $k$ of input units is large enough, $\frac{1}{4}2^k(2^n-1+1/30)$ suffice. These bounds sandwich the corresponding bound for shallow stochastic feedforward networks ($2^{k-1}(2^n-1)$). Considering that the derivation of both results is quite different, the similarity of the bounds is remarkable and suggests a deep similarity of both models. A definite verification of the tightness of these bounds would be desirable, although at the moment this seems to be very challenging.

As mentioned after Theorem 4, an undirected network can represent many of the Markov kernels that can be represented by a stochastic feedforward network of the same size. We suspect that actually the two architectures represent quite similar sets of stochastic maps, when restricting attention to a certain level of stochasticity. This intuition derives from Theorem 4 and Theorem 12. Especially, when it comes to representing deterministic functions, we have seen that shallow feedforward networks and conditional restricted Boltzmann machines are quite similar if not identical. We suspect that using undirected networks has the most benefit over using directed networks only when one is aimed at representing functions with a certain degree of stochasticity. Here, by stochasticity we refer loosely to the number of nonzero probability output vectors per input vector.

Of course universal approximation bounds are a very coarse characteristic of neural network architectures, and taking more information into account will allow more precise statements. However, we think that these bounds can be regarded as a sort of complexity measure, similar to the dimension of a manifold, or a measure of effective degrees of freedom, which are fundamental quantities in learning theory.

Various interesting problems remain open at this point. For instance, generalizing Theorem 2 to describe interactions in the case of deep Boltzmann machines. Also, estimating how narrow can be a conditional deep Boltzmann machine with input at the top and output at the bottom, while retaining the universal approximation capability.

## References


Hinton, Geoffrey E., Osindero, Simon, and Teh, Yee-Whye. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, July 2006.

Le Roux, Nicolas and Bengio, Yoshua. Representational power of restricted Boltzmann machines and deep belief networks. *Neural Computation*, 20(6):1631–1649, 2008.

Le Roux, Nicolas and Bengio, Yoshua. Deep belief networks are compact universal approximators. *Neural Computation*, 22:2192–2207, 2010.

Montúfar, Guido. Deep narrow Boltzmann machines are universal approximators. In *International Conference on Learning Representations*, ICLR '15, 2015.

Montúfar, Guido and Ay, Nihat. Refinements of universal approximation results for deep belief networks and restricted Boltzmann machines. *Neural Computation*, 23 (5):1306–1319, 2011.

Montúfar, Guido, Ay, Nihat, and Zahedi, Keyan. Geometry and expressive power of conditional restricted Boltzmann machines. *arXiv preprint arXiv:1402.3346*, 2014.

Sutskever, Ilya and Hinton, Geoffrey E. Deep narrow sigmoid belief networks are universal approximators. *Neural Computation*, 20:2629–2636, 2008.

Wenzel, Walter, Ay, Nihat, and Pasemann, Frank. Hyperplane arrangements separating arbitrary vertex classes in n-cubes. *Adv. Appl. Math.*, 25(3):284–306, 2000.

Younes, Laurent. Synchronous Boltzmann machines can be universal approximators. *Applied Mathematics Letters*, 9(3):109 – 113, 1996.

## A. Definitions

- An RBM:

$$p(v) = \sum_h \frac{1}{Z(W,b,c)} \exp(h^\top \mathbf{W} x + c^\top h + b^\top v)$$

- A DBN:

$$\begin{aligned} p(v = h_0) = &\sum_{h_1,\ldots,h_L} \frac{1}{Z(\mathbf{W}_{L-1}, \mathbf{b}_L, \mathbf{b}_{L-1})} \\ &\times \exp(h_L^\top \mathbf{W}_l h_{L-1} + \mathbf{b}_L^\top h_L + \mathbf{b}^\top h_L) \\ &\times \prod_{l=0}^{L-1} \frac{1}{Z(h_{l+1}\mathbf{W}_l, \mathbf{b}_l)} \exp(h_{l+1}^\top \mathbf{W} h_l + \mathbf{b}^\top h_l) \end{aligned}$$

- A DBM:

$$\begin{aligned} p(v = h_0) = &\sum_{h_1,\ldots,h_L} \frac{1}{Z(\mathbf{W}, \mathbf{b})} \\ &\times \exp(\sum_l h_{l+1}^\top \mathbf{W}_l h_l + \mathbf{b}_l^\top h_l + \mathbf{b}_L^\top h_L) \end{aligned}$$

- A CDBM:

$$\begin{aligned} &p(y = (h_{0k+1}, \ldots, h_{0k+n}) | x = (h_{01}, \ldots, h_{0k})) \\ &= \sum_{h_1,\ldots,h_L} \frac{1}{Z(\mathbf{W}, (h_{01}, \ldots, h_{0k}), \mathbf{b})} \\ &\times \exp(\sum_l h_{l+1}^\top \mathbf{W}_l h_l + \mathbf{b}_l^\top h_l + \mathbf{b}_L^\top h_L) \end{aligned}$$

- A CDBM2:

$$\begin{aligned} p(y = h_0 | x = h_L) = &\sum_{h_1,\ldots,h_{L-1}} \frac{1}{Z(\mathbf{W}, h_L, \mathbf{b})} \\ &\times \exp(\sum_l h_{l+1}^\top \mathbf{W}_l h_l + \mathbf{b}_l^\top h_l + \mathbf{b}_L^\top h_L) \end{aligned}$$

- A Markov random field with interactions $I \subseteq 2^{[n]}$:

$$p(v) = \frac{1}{Z(\theta)} \exp(\sum_{\lambda \in I} \theta_\lambda \prod_{i \in \lambda} v_i)$$

- An SFF:

$$p(y|x) = \sum_h \frac{\exp(h^\top \mathbf{W}_i x + \mathbf{b}_1^\top h)}{Z(\mathbf{W}_i x + \mathbf{b}_1)} \frac{\exp(y^\top \mathbf{W}_o h + \mathbf{b}_o^\top y)}{Z(\mathbf{W}_o h + \mathbf{b}_o)}$$

- A DFF:

$$\begin{aligned} p(y = h_L | x = h_0) = &\sum_{h_1,\ldots,h_{L-1}} \prod_l p(h_{l+1}|h_l) \\ p(h_{l+1}|h_l) = &\frac{\exp(h_{l+1}^\top \mathbf{W}_l h_l + \mathbf{b}_{l+1}^\top h_{l+1})}{Z(\mathbf{W}_l x + \mathbf{b}_{l+1})} \end{aligned}$$