Deep Belief Network - Derivations & Proofs

Paul F. Roysdon, Ph.D.

Contents

1 Mathematical Derivations & Proofs

1.1 Introduction

A Deep Belief Network (DBN) is a generative, multi-layer model that combines (i) a stack of *Restricted Boltzmann Machines* (RBMs) trained *greedily* layer-by-layer, with (ii) a hybrid directed/undirected topography: the *top two* layers form an undirected RBM, and all lower connections are *directed downward* (from higher to lower layers). Greedy pretraining maximizes a variational lower bound on the data log-likelihood and initializes parameters for subsequent discriminative (supervised) or generative fine-tuning. We derive the RBM from first principles, obtain learning rules (including Contrastive Divergence), show how stacking RBMs yields a DBN with a provably improving likelihood bound, and provide complete algorithmic steps with explicit dimensions.

1.2 Data and Notation

Let the observed data be $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^n$ with $\mathbf{x}_i \in \mathbb{R}^d$. For a binary DBN exposition, we use $\mathbf{v} \in \{0,1\}^{n_0}$ as the visible layer (often $\mathbf{v} = \mathbf{x}$ after binarization), and L hidden layers $\{\mathbf{h}^{(1)}, \ldots, \mathbf{h}^{(L)}\}$ with widths n_ℓ. DBN parameters are layerwise:

$$\mathbf{W}^{(\ell)} \in \mathbb{R}^{n_{\ell-1} \times n_\ell}, \qquad \mathbf{b}^{(\ell-1)} \in \mathbb{R}^{n_{\ell-1}}, \qquad \mathbf{c}^{(\ell)} \in \mathbb{R}^{n_\ell}, \qquad \ell = 1, \ldots, L.$$

For the *top RBM*, the pair $(\mathbf{h}^{(L-1)}, \mathbf{h}^{(L)})$ is undirected with energy defined below. Sigmoid $\sigma(a) = 1/(1+e^{-a})$ acts elementwise. Extensions to continuous visibles use Gaussian conditionals (noted later).

1.3 Model Formulation

1.3.1 Restricted Boltzmann Machine

An RBM is a bipartite undirected graphical model with visible units $\mathbf{v} \in \{0,1\}^{n_v}$, hidden units $\mathbf{h} \in \{0,1\}^{n_h}$, weights $\mathbf{W} \in \mathbb{R}^{n_v \times n_h}$, visible bias $\mathbf{b} \in \mathbb{R}^{n_v}$, and hidden bias $\mathbf{c} \in \mathbb{R}^{n_h}$. Define the *energy* and joint distribution

$$E(\mathbf{v},\mathbf{h}) \;=\; -\mathbf{v}^\top \mathbf{W}\mathbf{h} \;-\; \mathbf{b}^\top \mathbf{v} \;-\; \mathbf{c}^\top \mathbf{h}, \qquad p(\mathbf{v},\mathbf{h}) \;=\; \frac{e^{-E(\mathbf{v},\mathbf{h})}}{Z}, \quad Z = \sum_{\mathbf{v},\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h})}. \tag{1}$$

Factorized conditionals. Because the graph is bipartite,

$$p(\mathbf{h} \mid \mathbf{v}) = \prod_{j=1}^{n_h} p(h_j{=}1 \mid \mathbf{v})^{h_j} \big(1 - p(h_j{=}1 \mid \mathbf{v})\big)^{1-h_j}, \qquad p(h_j{=}1 \mid \mathbf{v}) = \sigma\big(c_j + \mathbf{W}_{:j}^\top \mathbf{v}\big), \tag{2}$$

$$p(\mathbf{v} \mid \mathbf{h}) = \prod_{i=1}^{n_v} p(v_i{=}1 \mid \mathbf{h})^{v_i} \big(1 - p(v_i{=}1 \mid \mathbf{h})\big)^{1-v_i}, \qquad p(v_i{=}1 \mid \mathbf{h}) = \sigma\big(b_i + \mathbf{W}_{i:}\mathbf{h}\big). \tag{3}$$

Proof. Holding $\mathbf{v}$ fixed, $E(\mathbf{v},\mathbf{h})$ is linear in each h_j: $E = -\sum_j h_j(c_j + \mathbf{W}_{:j}^\top \mathbf{v}) + \text{const}$, hence the conditional is a product of independent Bernoulli's with logit $c_j + \mathbf{W}_{:j}^\top \mathbf{v}$; similarly for $p(\mathbf{v} \mid \mathbf{h})$. ■

Log-likelihood gradients. For a single datum $\mathbf{v}$, the gradient of $\log p(\mathbf{v})$ is

$$\nabla_{\mathbf{W}} \log p(\mathbf{v}) = \underbrace{\mathbb{E}_{p(\mathbf{h}|\mathbf{v})}[\mathbf{v}\mathbf{h}^\top]}_{\text{“data” term}} \;-\; \underbrace{\mathbb{E}_{p(\mathbf{v},\mathbf{h})}[\mathbf{v}\mathbf{h}^\top]}_{\text{“model” term}}, \tag{4}$$

$$\nabla_{\mathbf{b}} \log p(\mathbf{v}) = \mathbb{E}_{p(\mathbf{h}|\mathbf{v})}[\mathbf{v}] \;-\; \mathbb{E}_{p(\mathbf{v},\mathbf{h})}[\mathbf{v}], \quad \nabla_{\mathbf{c}} \log p(\mathbf{v}) = \mathbb{E}_{p(\mathbf{h}|\mathbf{v})}[\mathbf{h}] \;-\; \mathbb{E}_{p(\mathbf{v},\mathbf{h})}[\mathbf{h}]. \tag{5}$$

Derivation. $\nabla_\theta \log p(\mathbf{v}) = -\mathbb{E}_{p(\mathbf{h}|\mathbf{v})}[\nabla_\theta E] + \mathbb{E}_{p(\mathbf{v},\mathbf{h})}[\nabla_\theta E]$; use $\nabla_{\mathbf{W}} E = -\mathbf{v}\mathbf{h}^\top$, $\nabla_{\mathbf{b}} E = -\mathbf{v}$, $\nabla_{\mathbf{c}} E = -\mathbf{h}$.

Contrastive Divergence (CD-k). Exact model expectations in Eqn. (4) are intractable (sum over all states); CD-k approximates them by k alternating Gibbs steps:

$$\mathbf{h}^{(0)} \sim p(\mathbf{h} \mid \mathbf{v}^{(0)} = \mathbf{v}), \quad \mathbf{v}^{(1)} \sim p(\mathbf{v} \mid \mathbf{h}^{(0)}), \quad \cdots, \quad \mathbf{h}^{(k)} \sim p(\mathbf{h} \mid \mathbf{v}^{(k)}).$$

Then update

$$\Delta\mathbf{W} \propto \mathbf{v}^{(0)}(\mathbb{E}[\mathbf{h} \mid \mathbf{v}^{(0)}])^\top \;-\; \mathbf{v}^{(k)}(\mathbb{E}[\mathbf{h} \mid \mathbf{v}^{(k)}])^\top, \tag{6}$$

$$\Delta\mathbf{b} \propto \mathbf{v}^{(0)} - \mathbf{v}^{(k)}, \qquad \Delta\mathbf{c} \propto \mathbb{E}[\mathbf{h} \mid \mathbf{v}^{(0)}] - \mathbb{E}[\mathbf{h} \mid \mathbf{v}^{(k)}], \tag{7}$$

with conditional expectations computed by Eqns. (2)–(3). Persistent CD (PCD) runs a long-chain *fantasy* particle instead of restarting at data.

Gaussian–Bernoulli RBM (continuous visibles). For $\mathbf{v} \in \mathbb{R}^{n_v}$ with diagonal variance σ^2,

$$E(\mathbf{v},\mathbf{h}) = \sum_i \frac{(v_i - b_i)^2}{2\sigma_i^2} - \mathbf{v}^\top(\mathbf{W}\mathbf{h})/\boldsymbol{\sigma}^2 - \mathbf{c}^\top\mathbf{h},$$

$$p(\mathbf{v} \mid \mathbf{h}) = \mathcal{N}\big(\mathbf{b} + \mathbf{W}\mathbf{h},\ \mathrm{diag}(\boldsymbol{\sigma}^2)\big),$$

$$p(h_j{=}1 \mid \mathbf{v}) = \sigma\Big(c_j + \frac{\mathbf{W}_{:j}^\top \mathbf{v}}{\boldsymbol{\sigma}^2}\Big).$$

1.3.2 Deep Belief Network

A DBN with L hidden layers defines the joint distribution

$$p(\mathbf{v}, \mathbf{h}^{(1)}, \dots, \mathbf{h}^{(L)}) \;=\; \underbrace{p(\mathbf{h}^{(L-1)}, \mathbf{h}^{(L)})}_{\text{top RBM}} \prod_{\ell=1}^{L-2} \underbrace{p(\mathbf{h}^{(\ell)} \mid \mathbf{h}^{(\ell+1)})}_{\text{downward sigmoids}} \cdot \underbrace{p(\mathbf{v} \mid \mathbf{h}^{(1)})}_{\text{visible sigmoid}}, \tag{8}$$

where all downward conditionals are factorial Bernoulli with parameters

$$p(h_i^{(\ell)}{=}1 \mid \mathbf{h}^{(\ell+1)}) \;=\; \sigma\big(b_i^{(\ell)} + \mathbf{W}_{i:}^{(\ell)}\, \mathbf{h}^{(\ell+1)}\big), \qquad p(v_i{=}1 \mid \mathbf{h}^{(1)}) \;=\; \sigma\big(b_i^{(0)} + \mathbf{W}_{i:}^{(1)}\, \mathbf{h}^{(1)}\big). \tag{9}$$

The *top RBM* uses parameters $(\mathbf{W}^{(L)}, \mathbf{c}^{(L-1)}, \mathbf{c}^{(L)})$ and energy $E(\mathbf{h}^{(L-1)}, \mathbf{h}^{(L)}) = -(\mathbf{h}^{(L-1)})^\top \mathbf{W}^{(L)} \mathbf{h}^{(L)} - (\mathbf{c}^{(L-1)})^\top \mathbf{h}^{(L-1)} - (\mathbf{c}^{(L)})^\top \mathbf{h}^{(L)}$.

Sampling from a DBN. (i) Run Gibbs sampling in the top RBM to draw $(\mathbf{h}^{(L-1)}, \mathbf{h}^{(L)})$; (ii) sample downward using Eqn. (9) layer by layer to obtain $\mathbf{v}$.

1.4 Inference and Recognition Pass

Exact posteriors $p(\mathbf{h}^{(1)}, \dots, \mathbf{h}^{(L)} \mid \mathbf{v})$ are intractable. A standard recognition (encoder) approximation uses upward factorial sigmoids

$$q\big(h_j^{(\ell)}{=}1 \,\big|\, \mathbf{h}^{(\ell-1)}\big) \;=\; \sigma\big(\tilde{c}_j^{(\ell)} + (\mathbf{W}^{(\ell)})_{:j}^\top \mathbf{h}^{(\ell-1)}\big), \qquad \mathbf{h}^{(0)} \equiv \mathbf{v}, \tag{10}$$

optionally tying $\tilde{c}^{(\ell)} = \mathbf{c}^{(\ell)}$ and using the *transpose* weights for a symmetric encoder/decoder in pretraining.

1.5 Greedy Layer-wise Pretraining and Likelihood Bound

Write the data log-likelihood and a variational lower bound via Jensen:

$$\log p(\mathbf{v}) = \log \sum_{\mathbf{h}^{(1)}} p(\mathbf{v}, \mathbf{h}^{(1)}) \;\geq\; \mathbb{E}_{q(\mathbf{h}^{(1)} \mid \mathbf{v})}\big[\log p(\mathbf{v}, \mathbf{h}^{(1)}) - \log q(\mathbf{h}^{(1)} \mid \mathbf{v})\big] \;\triangleq\; \mathcal{F}_1(\mathbf{v}; q), \tag{11}$$

with q any tractable posterior approximation (e.g., Eqn. (10)). Recursively introducing deeper latents yields nested bounds $\mathcal{F}_\ell(\mathbf{v}) = \mathbb{E}_{q(\mathbf{h}^{(1:\ell)} \mid \mathbf{v})}\big[\log p(\mathbf{v}, \mathbf{h}^{(1:\ell)}) - \log q(\mathbf{h}^{(1:\ell)} \mid \mathbf{v})\big]$.

Aggregated posterior. Given a dataset, the *aggregated posterior* at layer ℓ is

$$\tilde{p}_\ell(\mathbf{h}^{(\ell)}) \;\triangleq\; \frac{1}{n} \sum_{i=1}^{n} q\big(\mathbf{h}^{(\ell)} \,\big|\, \mathbf{h}^{(\ell-1)} = \mathbf{h}_i^{(\ell-1)}\big), \quad \text{where } \mathbf{h}_i^{(\ell-1)} \text{ are samples or means from } q(\cdot \mid \mathbf{v}_i). \tag{12}$$

Greedy pretraining theorem (Hinton 2006). *Proof.* Suppose at layer ℓ we train an RBM between $\mathbf{h}^{(\ell-1)}$ and $\mathbf{h}^{(\ell)}$ to (approximately) maximize $\sum_i \log p(\mathbf{h}_i^{(\ell-1)})$, where $\{\mathbf{h}_i^{(\ell-1)}\}$ are samples from the previous recognition pass. Then replacing the prior on $\mathbf{h}^{(\ell)}$ by the model learned at layer ℓ *increases* a variational lower bound on $\sum_i \log p(\mathbf{v}_i)$. *Sketch.* The bound improvement equals the decrease in $\mathrm{KL}\big(\tilde{p}_\ell(\mathbf{h}^{(\ell)}) \,\|\, p(\mathbf{h}^{(\ell)})\big)$ attained by fitting the RBM prior. Since training the RBM reduces this KL (by increasing likelihood of $\tilde{p}_\ell$ under the RBM), the bound tightens layer by layer. ■

Greedy pretraining procedure.

1. **Layer 1 (visible $\leftrightarrow$ hidden-1):** Train an RBM on $\mathbf{v}$ to model $p(\mathbf{v})$ via CD/PCD; store parameters $(\mathbf{W}^{(1)}, \mathbf{b}^{(0)}, \mathbf{c}^{(1)})$.

2. **Form data for layer 2:** For each $\mathbf{v}_i$, compute hidden means $\hat{\mathbf{h}}_i^{(1)} = \sigma(\mathbf{c}^{(1)} + (\mathbf{W}^{(1)})^\top \mathbf{v}_i)$ (or sample) and treat them as *data* for the next RBM.

3. **Layer ℓ:** Train an RBM between $\mathbf{h}^{(\ell-1)}$ and $\mathbf{h}^{(\ell)}$. Repeat up to the *top RBM*.

1.6 Fine-Tuning

After pretraining, one can:

(a) Discriminative fine-tuning. Attach a classifier to $\mathbf{h}^{(L)}$ or $\mathbf{h}^{(L-1)}$ and minimize cross-entropy $\mathcal{L}_{\text{sup}} = -\sum_i \sum_k y_{ik} \log p_\theta(y{=}k \mid \mathbf{v}_i)$ by backprop, initializing weights from RBMs (transpose if necessary).

(b) Generative fine-tuning (Wake–Sleep). Introduce separate recognition parameters $\{\tilde{\mathbf{W}}^{(\ell)}, \tilde{\mathbf{c}}^{(\ell)}\}$ for q. *Wake phase:* clamp $\mathbf{v}$, sample upward $q(\mathbf{h}^{(\ell)} \mid \mathbf{h}^{(\ell-1)})$, update *generative* weights to increase $\log p(\mathbf{v}, \mathbf{h}^{(1:L)})$ (i.e., RBM CD at top, logistic regression losses for downward sigmoids). *Sleep phase:* sample from the model (top RBM Gibbs + downward), then update *recognition* weights to better predict the sampled hidden states (minimize $\mathrm{KL}(p \parallel q)$).

1.7 Algorithm (DBN Training)

1. **Input:** data $\{\mathbf{v}_i\}_{i=1}^n$, layer widths $n_0, \ldots, n_L$, learning rates η_ℓ, CD steps k.

2. **For $\ell = 1$ to L (greedy pretraining):**

   (a) Construct dataset $\{\tilde{\mathbf{v}}_i\}$ for this RBM: $\tilde{\mathbf{v}}_i = \mathbf{v}_i$ if $\ell = 1$; otherwise $\tilde{\mathbf{v}}_i = \hat{\mathbf{h}}_i^{(\ell-1)}$ from the previous layer.

   (b) Train RBM($n_{\ell-1} \leftrightarrow n_\ell$) by CD-$k$ updates on mini-batches using Eqns. (2)–(3).

3. **(Optional) Fine-tune:**

   - *Supervised:* stack pretrained weights into a feedforward net; run backprop on labeled data.
   - *Generative:* apply wake–sleep or joint variational learning of p and q.

1.8 Proofs and Identities

Variational bound and monotonic improvement. *Proof.* For any $q(\mathbf{h}^{(1:\ell)} \mid \mathbf{v})$, $\log p(\mathbf{v}) = \mathcal{F}_\ell(\mathbf{v}) + \mathrm{KL}\big(q(\mathbf{h}^{(1:\ell)} \mid \mathbf{v}) \parallel p(\mathbf{h}^{(1:\ell)} \mid \mathbf{v})\big)$. Maximizing $\mathcal{F}_\ell$ w.r.t. the *prior* on $\mathbf{h}^{(\ell)}$ while holding the recognition q fixed is equivalent to minimizing the KL between the aggregated posterior $\tilde{p}_\ell$ and the prior, which training the RBM achieves. Hence $\mathcal{F}_\ell$ increases (or stays the same) after each layer. ■

RBM gradient form (data minus model). *Proof.* From Eqn. (1), $\nabla_\theta \log p(\mathbf{v}) = -\mathbb{E}_{p(\mathbf{h}|\mathbf{v})}[\nabla_\theta E] + \mathbb{E}_{p(\mathbf{v},\mathbf{h})}[\nabla_\theta E]$; substituting $\nabla_\theta E$ yields Eqn. (4). ■

Factorized conditionals. *Proof.* Given bipartite structure, $\log p(\mathbf{h} \mid \mathbf{v})$ separates across h_j, giving independent Bernoulli conditionals with logits $c_j + \mathbf{W}_{:j}^\top \mathbf{v}$; similarly for visibles. ■

1.9 Computational Aspects

One CD-k step requires two sparse-dense passes per Gibbs step: $\mathcal{O}(n_v n_h)$ for computing logits (matrix-vector products) per data case. Mini-batch training amortizes costs. PCD maintains a persistent Markov chain for better model expectation estimates.

1.10 Summary of Variables and Their Dimensions

- $\mathbf{v} \in \{0,1\}^{n_0}$: visible layer (or real-valued for Gaussian RBM).
- $\mathbf{h}^{(\ell)} \in \{0,1\}^{n_\ell}$: hidden layer ℓ, $\ell = 1, \ldots, L$ (Bernoulli units; other choices possible).
- $\mathbf{W}^{(\ell)} \in \mathbb{R}^{n_{\ell-1} \times n_\ell}$: weights between layers $\ell - 1$ and ℓ.
- $\mathbf{b}^{(0)} \in \mathbb{R}^{n_0}$: visible biases; $\mathbf{b}^{(\ell)} \in \mathbb{R}^{n_\ell}$: downward biases for $p(\mathbf{h}^{(\ell)} \mid \mathbf{h}^{(\ell+1)})$.
- $\mathbf{c}^{(\ell)} \in \mathbb{R}^{n_\ell}$: upward/hidden biases in RBM conditionals (pretraining); for the top RBM, $(\mathbf{c}^{(L-1)}, \mathbf{c}^{(L)})$.
- $E(\cdot)$, Z: RBM energy and partition function; conditionals given by Eqns. (2)–(3).
- $q(\cdot)$: recognition distribution used in the upward pass Eqn. (10).
- $\mathcal{F}_\ell$: variational lower bound at depth ℓ Eqn. (11); $\tilde{p}_\ell$: aggregated posterior Eqn. (12).
- k: number of Gibbs steps in CD-k; PCD uses persistent chains.

1.11 Summary

From first principles, a DBN is built by stacking RBMs: each RBM defines an energy-based model with factorized conditionals and *data-minus-model* gradients; Contrastive Divergence (or PCD) supplies tractable updates. Stacking proceeds greedily by fitting each RBM to the aggregated posterior of the layer below, which provably tightens a variational lower bound on the data log-likelihood. The resulting hybrid model factors as a top undirected RBM with downward directed sigmoids; sampling uses top-level Gibbs followed by ancestral generation. Pretraining yields a strong initialization for discriminative backprop or for wake–sleep generative fine-tuning. All variables, dimensions, objectives, gradients, and algorithms have been stated explicitly for implementation.