Introduction to Neural Networks

Johns Hopkins University
Engineering for Professionals Program
605-447/625-438
Dr. Mark Fleischer

Module 11.1: Restricted Boltzmann Machines

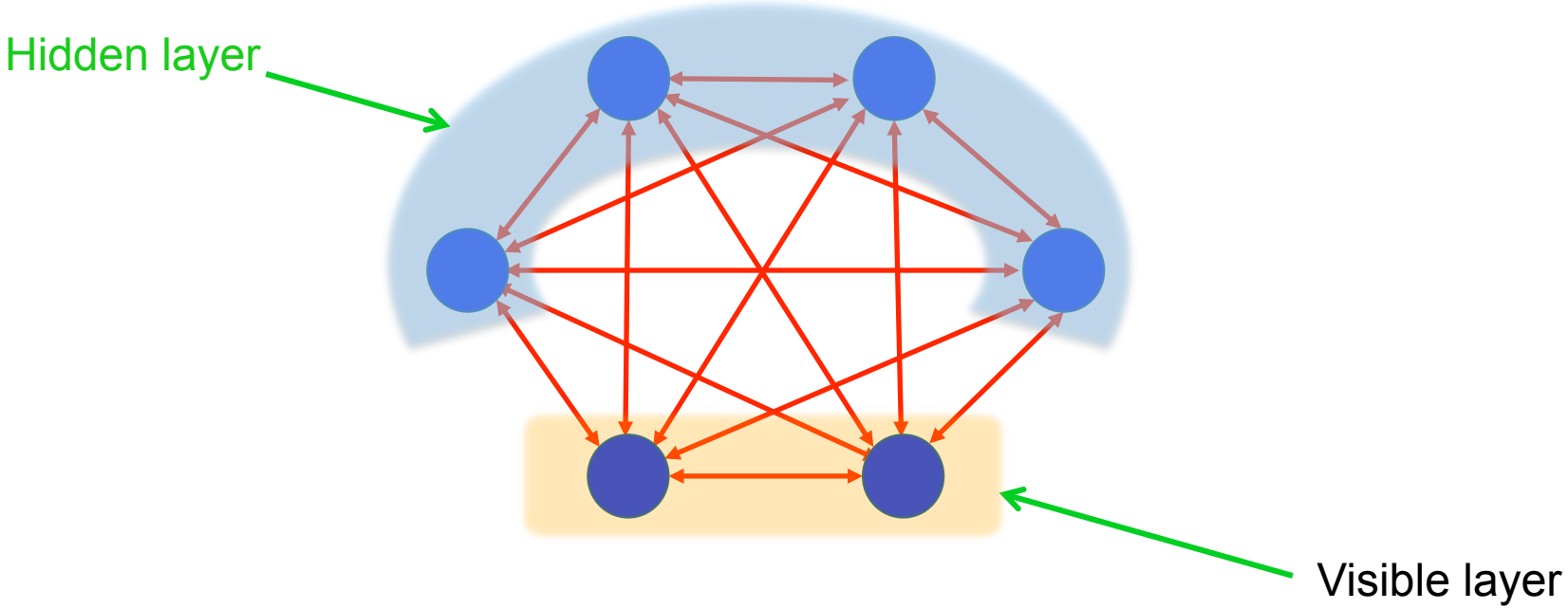
What We've Covered So Far

- Boltzmann machines are essentially stochastic versions of Hopfield networks.
- Stochastic assignment of node states using a binary value.
- Associate a performance metric via the energy or consensus function with network configurations.
- Capability to 'anneal' a BM to solve various optimization problems.

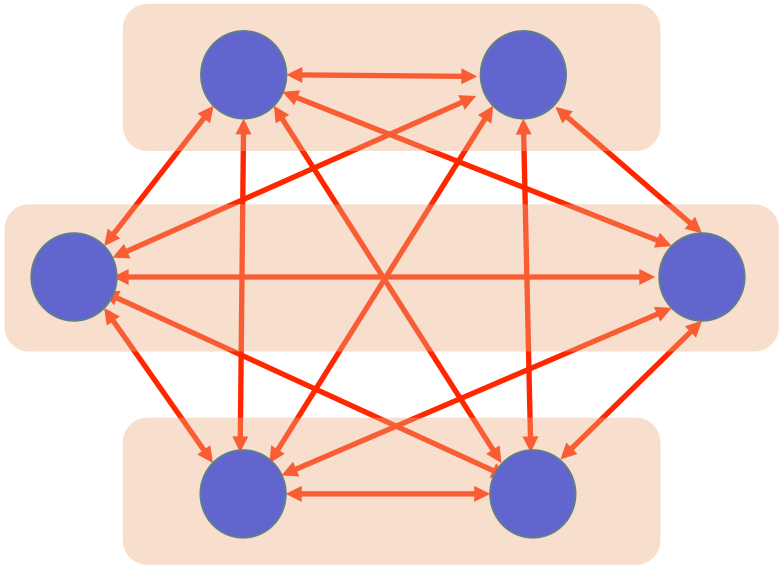
Model Relationships
for Recurrent Networks

Deterministic		Stochastic
Hopfield Networks	→	Boltzmann Machines
Binary Associative Memories	→	Restricted Boltzmann Machines

Restricted Boltzmann Machines



Restricted Boltzmann Machines

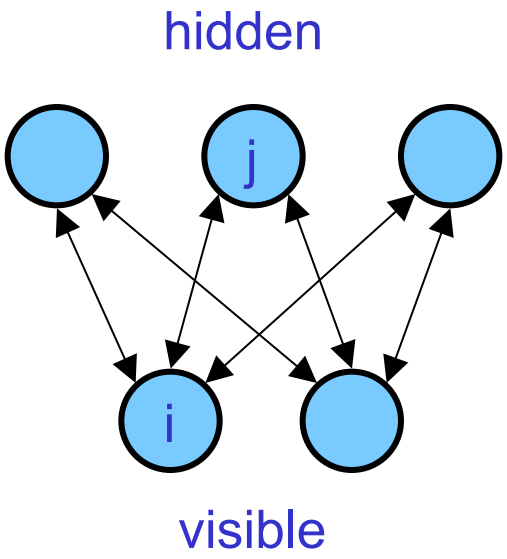


Sort of a multi-layer BAM!

Configured as a *Belief Network*

Restricted Boltzmann Machines

- We restrict the connectivity to make learning easier.
  - Only one layer of hidden units.
    - We will deal with more layers later
  - No connections between hidden units.
- In an RBM, the hidden units are conditionally independent given the visible states.
  - So we can quickly get an unbiased sample from the posterior distribution when given a data-vector.
  - This is a big advantage over directed belief nets



From Hinton 2007

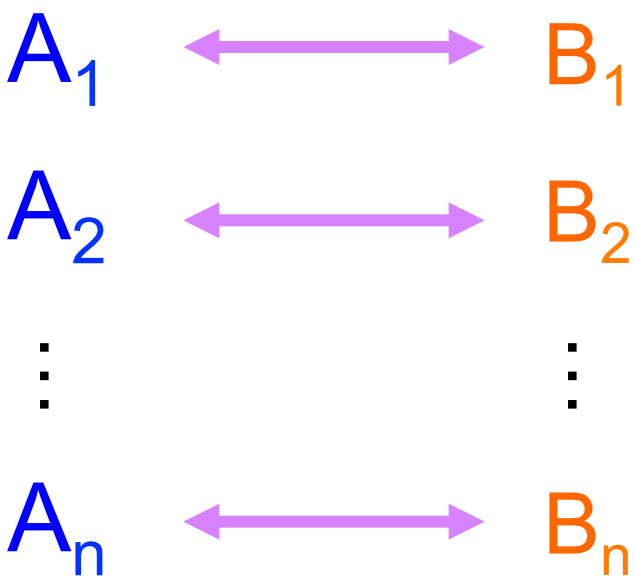
Purpose of RBMs

- Many possible applications.
- Essentially attempts to establish a useful association between visible vectors and hidden vectors similar in nature to how BAMs function.
  - used in factor analysis
  - character recognition
  - many others---active area of research
- Stochastic states allow for greater flexibility and noise

What Do RBMs Do?

- Just like BAMs, they attempt to 'reconstruct' a training vector.
  - In BAMs these were the 'A' vectors or the 'data'.
  - Noisy or variable training vectors probabilistically produce feature detectors in the hidden layer which then probabilistically produce 'reconstructions' of the training vectors.

Binary Associative Memories



Goal: Noisy A_1 produces a correct B_1 which then produces a correct A_1.

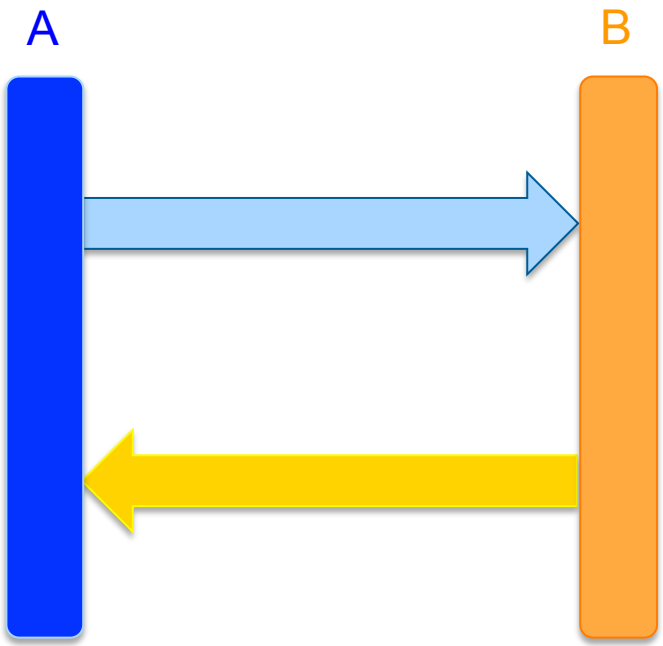
$$\tilde{A}_1 \rightarrow B_1 \rightarrow A_1$$

Binary Associative Memories

- Present a noisy A as input to the A nodes.
- The A nodes produce outputs and are presented to the B nodes.
- The B nodes produce outputs and are presented back to the A nodes.

Binary Associative Memories

Restricted Boltzmann Machines

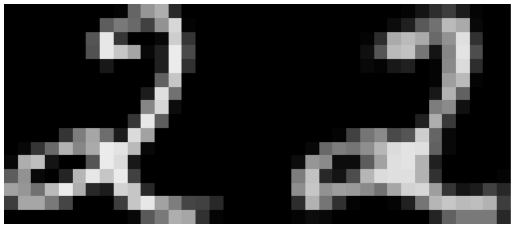


Only Stochastically!

How well can we reconstruct the digit images from the binary feature activations?

Data

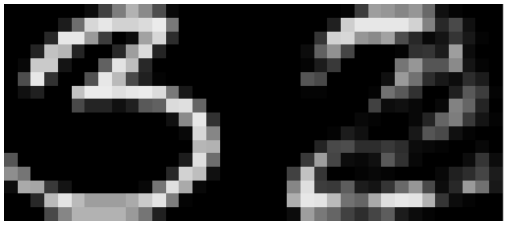
Reconstruction from activated binary features



New test images from the digit class that the model was trained on

Data

Reconstruction from activated binary features



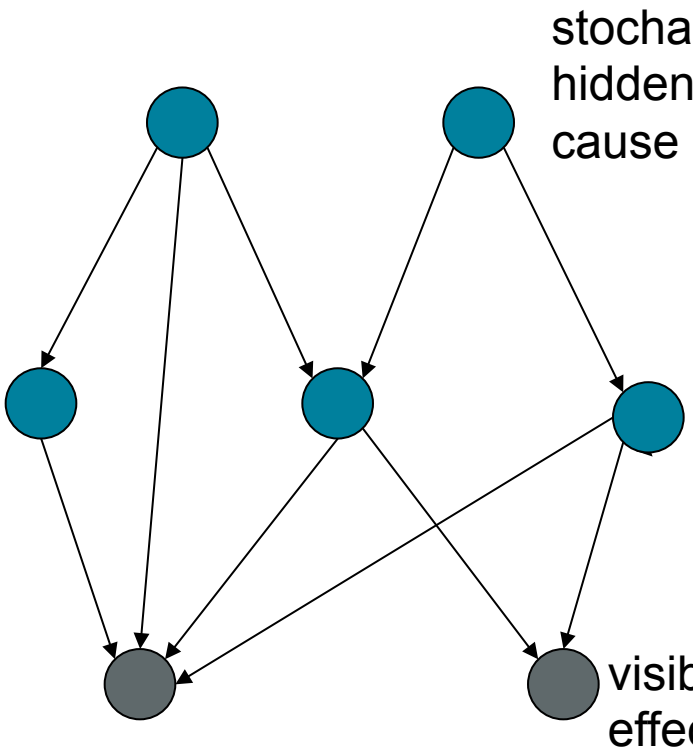
Images from an unfamiliar digit class
(the network tries to see every image as a 2)

From Hinton 2007

Belief Nets

- A belief net is a directed acyclic graph composed of stochastic variables.
- We get to observe some of the variables and we would like to solve two problems:
- The inference problem: Infer the states of the unobserved variables.
- The learning problem: Adjust the interactions between variables to make the network more likely to generate the observed data.

From Hinton 2007

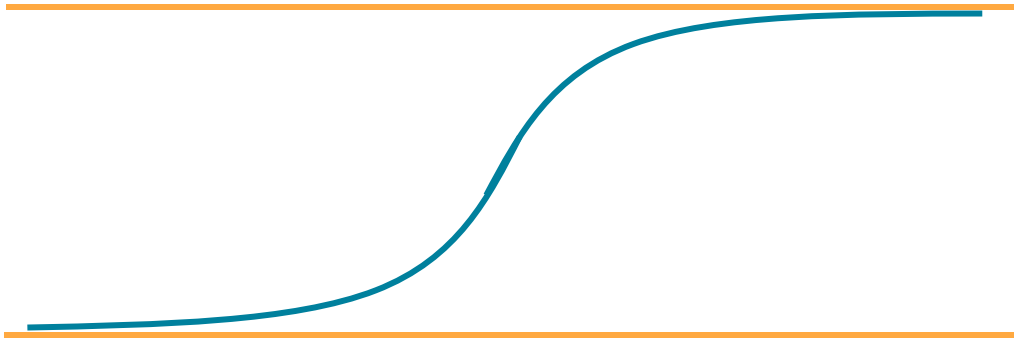


We will use nets composed of layers of stochastic binary variables with weighted connections

Dynamics

Recall the Sigmoid activation function

$$\frac{1}{1+e^{-S_i/T}}$$



Again, we're dealing with stochastic neurons.
What does this curve remind you of?

Dynamics

Set cell activation (state) according to:

$$x_i = \begin{cases} 1 & \text{with probability } p_i = \dfrac{1}{1+e^{-S_i/T}} \\ 0 & \text{with probability } 1-p_i \end{cases}$$

The Energy/Consensus Function

Define the energy function *E*. With the BM we used

$$E = -\sum_{i<j} w_{ij} x_i x_j - \sum_i \theta_i x_i - \sum_j \xi_j x_j$$

In RBMs, only the connections between layers is important.

$$E = -\sum_{i,j} w_{ij} v_i h_j - \sum_i \theta_i v_i - \sum_j \xi_i h_i$$

The Energy of a joint configuration

(ignoring terms with biases)

binary state of visible unit i

binary state of hidden unit j

$$E(\mathbf{v},\mathbf{h}) = -\sum_{i,j} v_i h_j w_{ij}$$

Energy with configuration v on the visible units and h on the hidden units

weight between units i and j

$$\frac{\partial E(\mathbf{v},\mathbf{h})}{\partial w_{ij}} = -v_i h_j$$

From Hinton 2007 (modified)

Think in terms of consensus

$$C(\mathbf{v},\mathbf{h}) = \sum_{i=1}^{m}\sum_{j=1}^{n} w_{ij} v_i h_j$$

If we want to strengthen an association where both v_i and h_j are 1s,

make w_{ij} a large positive number.

If we want to strengthen an association where v_i and h_j have opposite states,

make w_{ij} a large negative number.

Summary

- RBMs are stochastic versions of BAMs.
  - Two layers: *visible* and *hidden*
- Use probabilistic machinery of Boltzmann machines.
  - Probability of a node's state is based on sigmoid function with activity value based on weighted inputs from 'the other set of nodes'
  - Energy of a (v, h) pair is defined as sum of the weighted products of visible node states and hidden node states.

Introduction to Neural Networks

Johns Hopkins University
Engineering for Professionals Program
605-447/625-438
Dr. Mark Fleischer

Module 11.2: RBM Mathematics

What We've Covered So Far

- Probabilistic foundations of RBMs.
  - *Energy/Consensus* associated with a visible and hidden pair of vectors.
  - Probability of a node's states

Goal

- Raise the probability that a visible vector with be faithfully reconstructed when a 'hidden' vector is presented to the visible layer.

Question:

How do we train an RBM so that 'reconstructions' are likely to create a reasonable facsimile of the original data?

Weights → Energies → Probabilities

- Each possible joint configuration of the visible and hidden units has an energy
  - The energy is determined by the weights and biases (as in a Hopfield net).
- The energy of a joint configuration of the visible and hidden units determines its probability:

$$p(\mathbf{v},\mathbf{h}) \propto e^{-E(\mathbf{v},\mathbf{h})}$$

- The probability of a configuration over the visible units is found by summing the probabilities of all the joint configurations that contain it.

From Hinton 2007 (modified)

Using energies to define probabilities

- The probability of a joint configuration over both visible and hidden units depends on the energy of that joint configuration compared with the energy of all other joint configurations.
- The probability of a configuration of the visible units is the sum of the probabilities of all the joint configurations that contain it.

$$p(\mathbf{v},\mathbf{h}) = \frac{e^{-E(\mathbf{v},\mathbf{h})}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}$$

partition function

$$p(\mathbf{v}) = \frac{\sum_{g} e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}$$

From Hinton 2007 (modified)

How Do We Train an RBM?

$$p(\mathbf{v}) = \frac{\sum_{g} e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}} \rightarrow \ln p(\mathbf{v}) = \ln \left(\frac{\sum_{g} e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}} \right)$$

$$= \ln \sum_{g} e^{-E(\mathbf{v},\mathbf{h}^g)} - \ln \sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}$$

$$\frac{\partial \ln p(\mathbf{v})}{\partial w_{ij}} = \underbrace{\frac{\partial}{\partial w_{ij}} \ln \sum_{g} e^{-E(\mathbf{v},\mathbf{h}^g)}}_{\text{A}} - \underbrace{\frac{\partial}{\partial w_{ij}} \ln \sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}_{\text{B}}$$

Looking at Term A:

$$\frac{\partial}{\partial w_{ij}} \ln \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} = \frac{1}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} \cdot \frac{\partial}{\partial w_{ij}} \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}$$

$$= \frac{1}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} \cdot \sum_g \frac{\partial e^{-E(\mathbf{v},\mathbf{h}^g)}}{\partial w_{ij}}$$

$$= \frac{1}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} \cdot \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} \cdot \frac{\partial\left(-E(\mathbf{v},\mathbf{h}^g)\right)}{\partial w_{ij}}$$

Looking at Term A:

Recall that $E(\mathbf{v},\mathbf{h}) = -\sum_{i,j} v_i h_j w_{ij}$

$$\frac{\partial}{\partial w_{ij}} \ln \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} = \frac{1}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} \cdot \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} \cdot \frac{\partial\left(-E(\mathbf{v},\mathbf{h}^g)\right)}{\partial w_{ij}}$$

$$= \frac{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} v_i h_j^g}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} = \sum_g p(\mathbf{h}^g \mid \mathbf{v}) v_i h_j^g = \langle v_i \cdot h_j \rangle_{\mathbf{v}}$$

Where does that conditional probability come from?

$$\frac{e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} = p(\mathbf{h}^g \,|\, \mathbf{v})$$

$$p(\mathbf{h}^g \,|\, \mathbf{v}) = \frac{p(\mathbf{v},\mathbf{h}^g)}{p(\mathbf{v})} = \frac{\frac{1}{Z} e^{-E(\mathbf{v},\mathbf{h}^g)}}{\frac{1}{Z}\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} = \frac{e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}}$$

Looking at Term B:

$$\frac{\partial}{\partial w_{ij}} \ln \sum_{u,g} e^{-E(\mathbf{v}^u, \mathbf{h}^g)} = \frac{1}{\sum_{u,g} e^{-E(\mathbf{v}^u, \mathbf{h}^g)}} \cdot \frac{\partial}{\partial w_{ij}} \sum_{u,g} e^{-E(\mathbf{v}^u, \mathbf{h}^g)}$$

$$= \frac{1}{\sum_{u,g} e^{-E(\mathbf{v}^u, \mathbf{h}^g)}} \cdot \sum_{u,g} \frac{\partial e^{-E(\mathbf{v}^u, \mathbf{h}^g)}}{\partial w_{ij}}$$

Looking at Term B:

$$\frac{\partial}{\partial w_{ij}} \ln \sum_{u,g} e^{-E(\mathbf{v}^u, \mathbf{h}^g)} = \frac{\sum_{u,g} \frac{\partial e^{-E(\mathbf{v}^u, \mathbf{h}^g)}}{\partial w_{ij}}}{\sum_{u,g} e^{-E(\mathbf{v}^u, \mathbf{h}^g)}}$$

$$= \frac{\sum_{u,g} e^{-E(\mathbf{v}^u, \mathbf{h}^g)} \frac{\partial\left(-E(\mathbf{v}^u, \mathbf{h}^g)\right)}{\partial w_{ij}}}{\sum_{u,g} e^{-E(\mathbf{v}^u, \mathbf{h}^g)}}$$

$$= \frac{\sum_{u,g} e^{-E(\mathbf{v}^u, \mathbf{h}^g)} v_i^u h_j^g}{\sum_{u,g} e^{-E(\mathbf{v}^u, \mathbf{h}^g)}} = \sum_{u,g} p(\mathbf{v}^u, \mathbf{h}^g) v_i^u h_j^g = \langle v_i \cdot h_j \rangle_{\mathbf{vh}}$$

Basis of *Contrastive Divergence*

$$\frac{\partial \ln p(\mathbf{v})}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} \ln \sum_{g} e^{-E(\mathbf{v},\mathbf{h}^g)} - \frac{\partial}{\partial w_{ij}} \ln \sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}$$

$$= \langle v_i \cdot h_j \rangle_{\mathbf{v}} - \langle v_i \cdot h_j \rangle_{\mathbf{vh}}$$

Summary

- Showed the derivative of the log probability with respect to weights
- This can serve as the basis of a gradient ascent method for increasing the probability of the reconstructed vector v.

Introduction to Neural Networks

Johns Hopkins University
Engineering for Professionals Program
605-447/625-438
Dr. Mark Fleischer

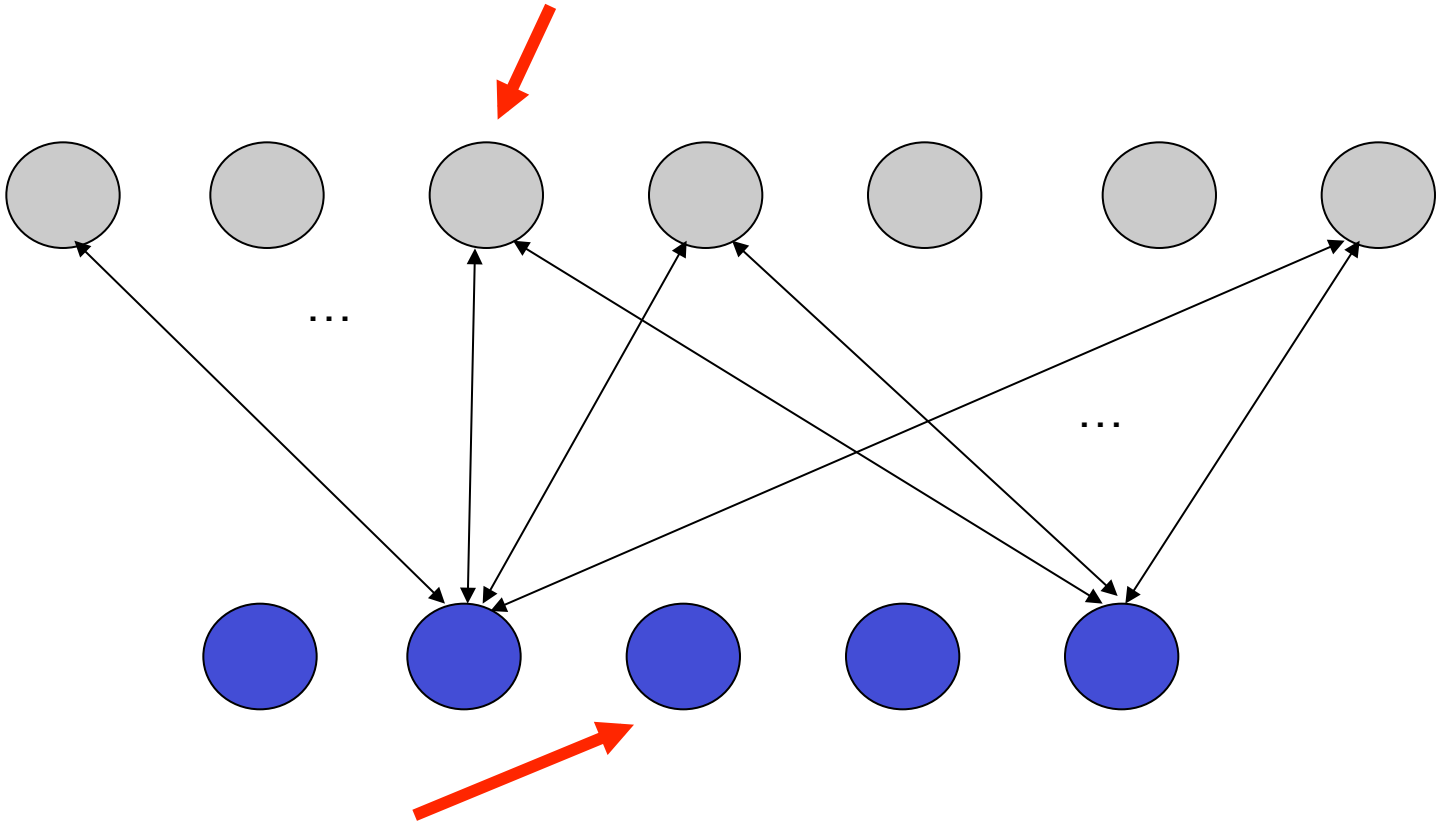
Module 11.3: An RBM Training Example

What We've Covered So Far

- Examined the mathematical foundations for an efficient approach to training RBMs.
- Derivative of the log probability of a vector **v.**
- Two expectations involved.
  - First term based on frequency of $v_i h_j$ over the training set of vectors **v.**
  - Second term based on frequency of $v_i h_j$ over all vectors **v** and **h.**

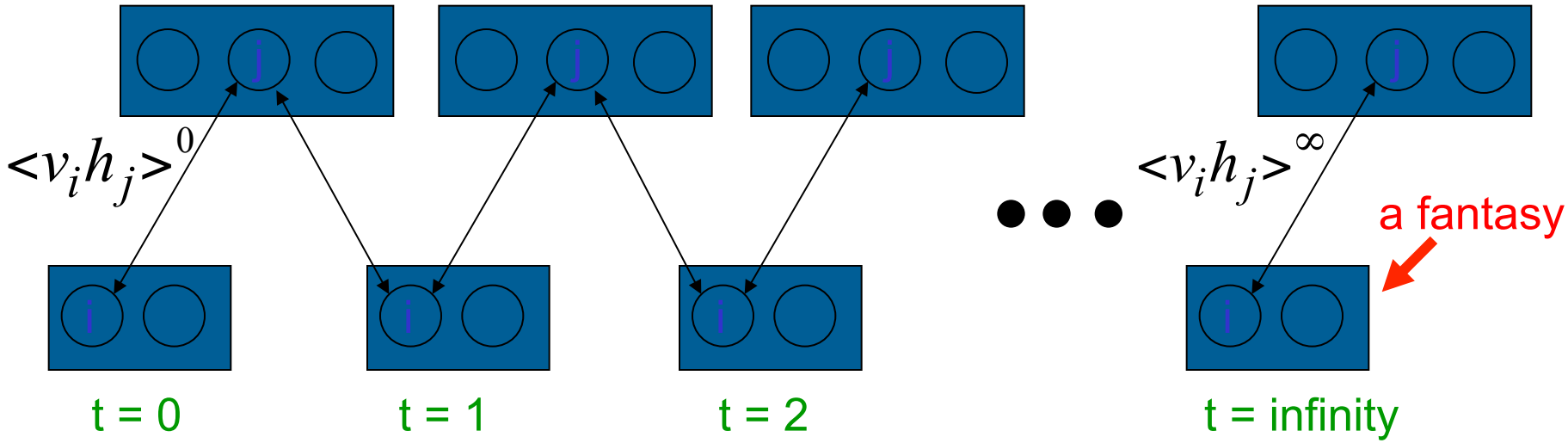
RBM

Hidden Layer of Nodes/Feature Detectors



Visible Layer of Nodes

A picture of the maximum likelihood learning algorithm for an RBM



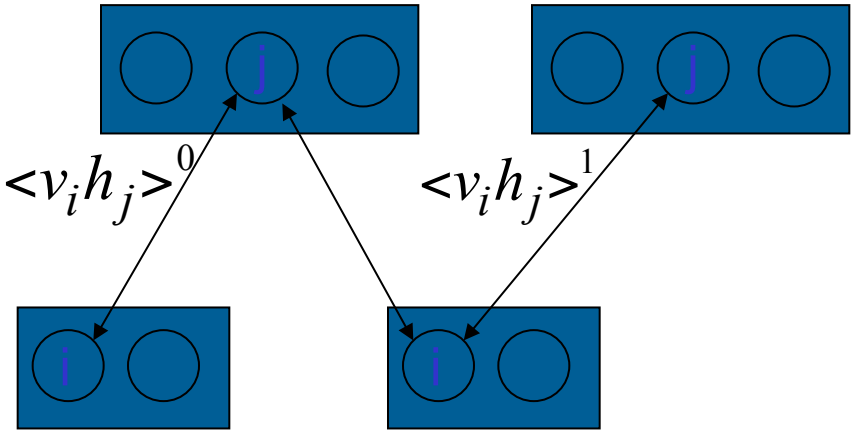
Start with a training vector on the visible units.

Then alternate between updating all the hidden units in parallel and updating all the visible units in parallel.

From Hinton 2007

$$\frac{\partial \log p(v)}{\partial w_{ij}} = <v_i h_j>^0 - <v_i h_j>^\infty$$

A quick way to learn an RBM



Start with a training vector on the visible units.

Update all the hidden units in parallel

Update the all the visible units in parallel to get a "reconstruction".

Update the hidden units again.

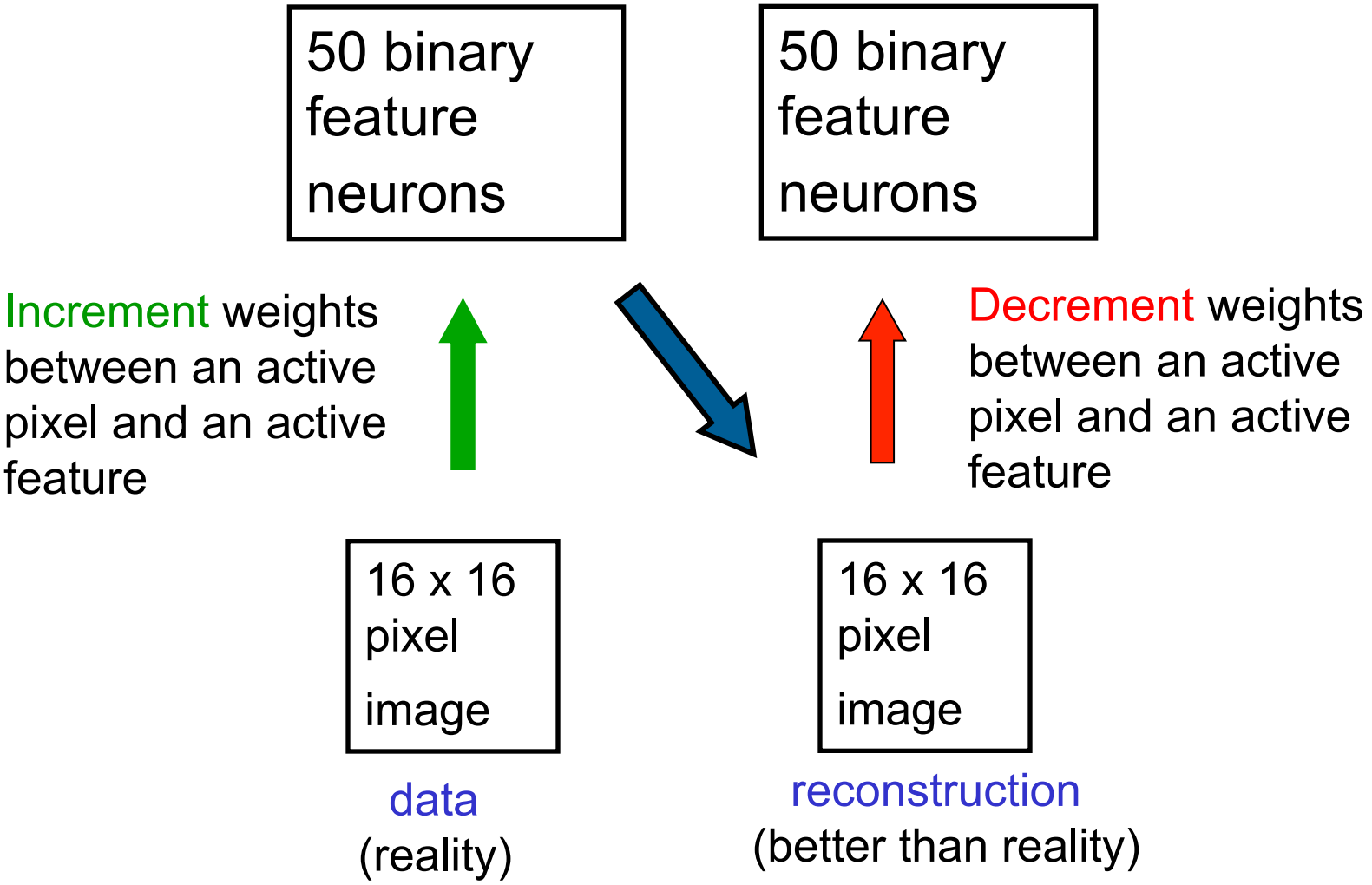
$$\Delta w_{ij} = \varepsilon \left(<v_i h_j>^0 - <v_i h_j>^1 \right)$$

This is not following the gradient of the log likelihood. But it works well.

It is approximately following the gradient of another objective function.

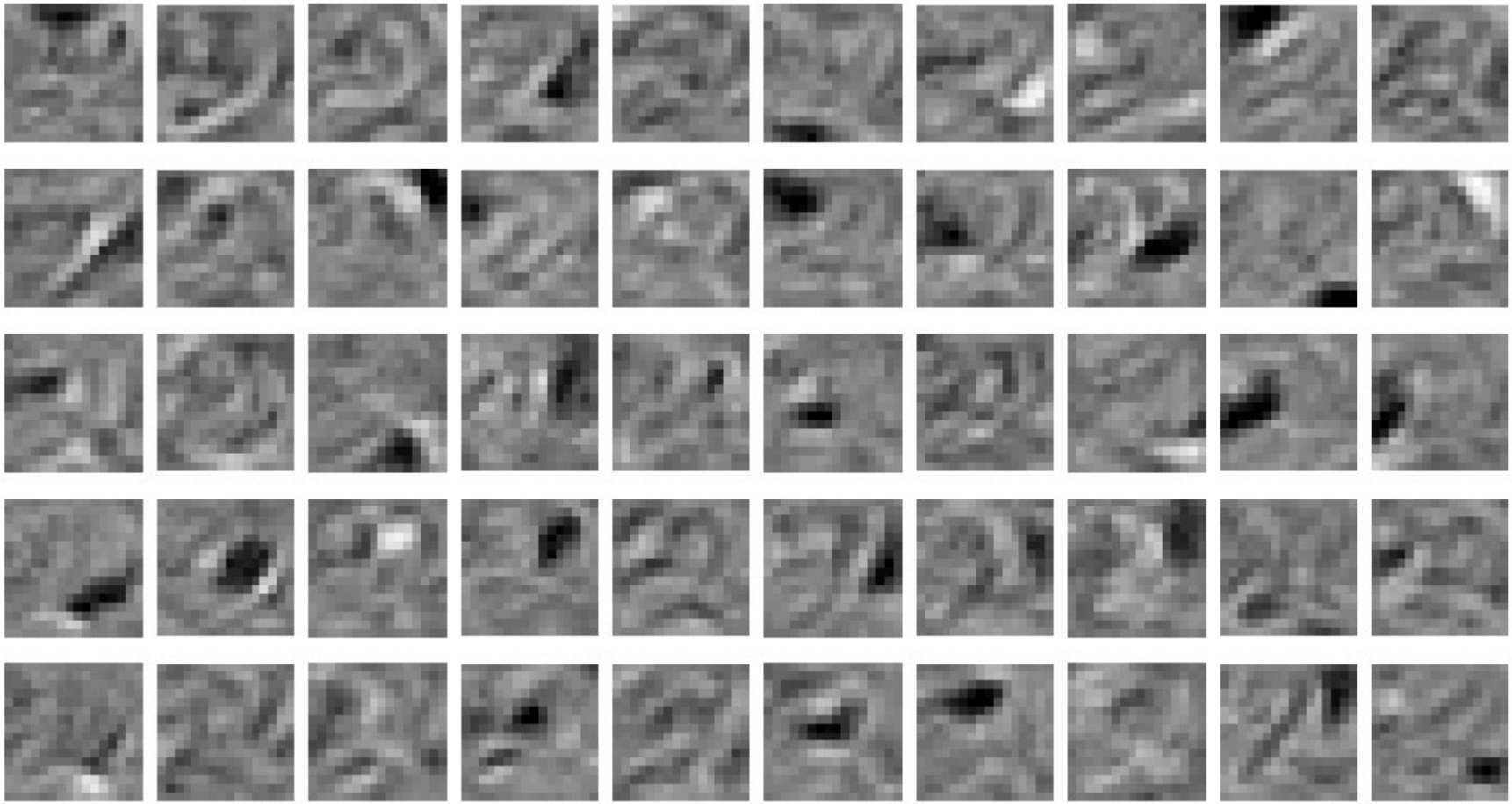
From Hinton 2007

How to learn a set of features that are good for reconstructing images of the digit 2



From Hinton 2007

The final 50 x 256 weights



Each neuron grabs a different feature.

From Hinton 2007

Training a deep network

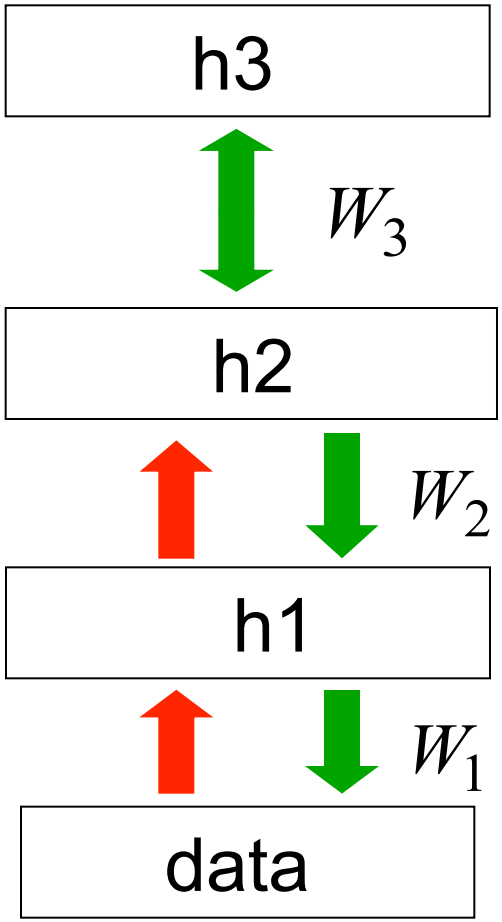
- First train a layer of features that receive input directly from the pixels.
- Then treat the activations of the trained features as if they were pixels and learn features of features in a second hidden layer.
- It can be proved that each time we add another layer of features we improve a variational lower bound on the log probability of the training data.
  - The proof is slightly complicated.
  - But it is based on a neat equivalence between an RBM and a deep directed model (described later)

From Hinton 2007

The generative model after learning 3 layers

- To generate data:
  1. Get an equilibrium sample from the top-level RBM by performing alternating Gibbs sampling.
  2. Perform a top-down pass to get states for all the other layers.

  So the lower level bottom-up connections are not part of the generative model. They are just used for inference.



From Hinton 2007

Why does greedy learning work?

The weights, W, in the bottom level RBM define p(v|h) and they also, indirectly, define p(h).

So we can express the RBM model as

$$p(v) = \sum_h p(h)\ p(v \mid h)$$

If we leave p(v|h) alone and improve p(h), we will improve p(v).

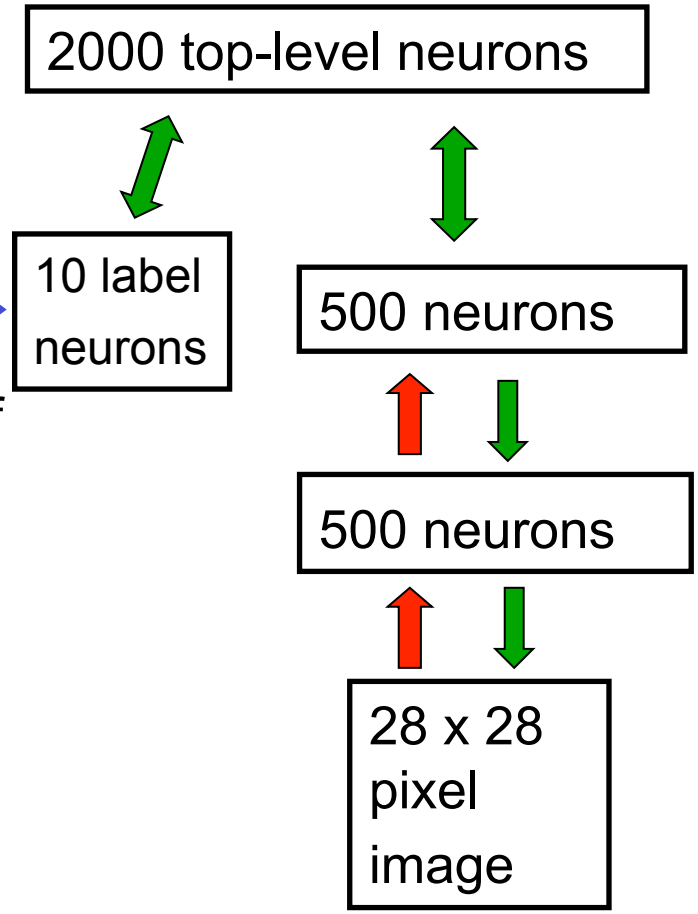
To improve p(h), we need it to be a better model of the aggregated posterior distribution over hidden vectors produced by applying W to the data.

From Hinton 2007

A neural model of digit recognition

The top two layers form an associative memory whose energy landscape models the low dimensional manifolds of the digits.

The energy valleys have names



The model learns to generate combinations of labels and images.

To perform recognition we start with a neutral state of the label units and do an up-pass from the image followed by a few iterations of the top-level associative memory.

From Hinton 2007

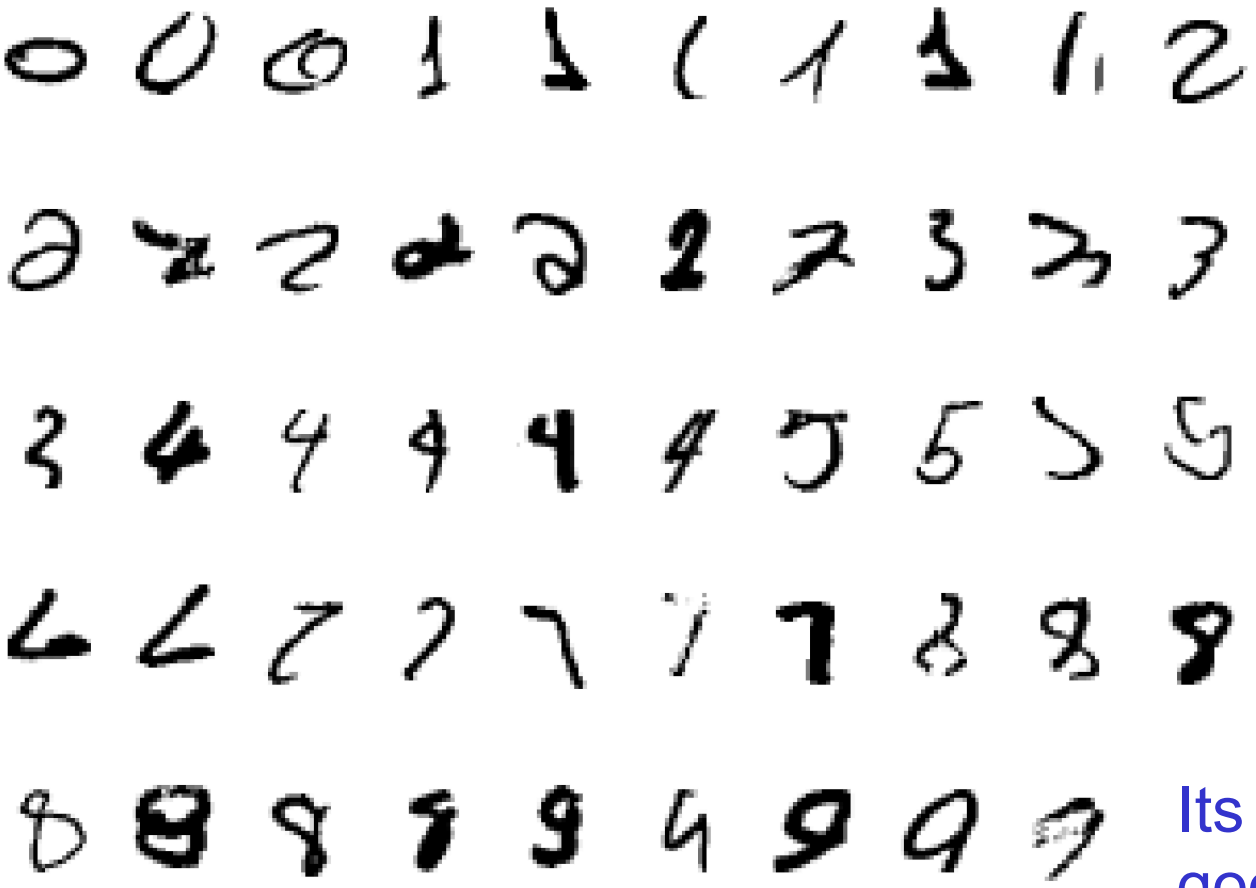
Fine-tuning with a contrastive divergence version of the "wake-sleep" algorithm

- After learning many layers of features, we can fine-tune the features to improve generation.
- 1. Do a stochastic bottom-up pass
  - Adjust the top-down weights to be good at reconstructing the feature activities in the layer below.
- 2. Do a few iterations of sampling in the top level RBM
  - Use CD learning to improve the RBM
- 3. Do a stochastic top-down pass
  - Adjust the bottom-up weights to be good at reconstructing the feature activities in the layer above.

Samples generated by letting the associative memory run with one label clamped. There are 1000 iterations of alternating Gibbs sampling between samples.

From Hinton 2007

Examples of correctly recognized handwritten digits that the neural network had never seen before



From Hinton 2007

Its very good

How well does it discriminate on MNIST test set with no extra information about geometric distortions?

• Generative model based on RBM's	1.25%
• Support Vector Machine (Decoste et. al.)	1.4%
• Backprop with 1000 hiddens (Platt)	~1.6%
• Backprop with 500 -->300 hiddens	~1.6%
• K-Nearest Neighbor	~ 3.3%

- Its better than backprop and much more neurally plausible because the neurons only need to send one kind of signal, and the teacher can be another sensory input.

From Hinton 2007

Summary

- Examined an application using RBMs as deep belief networks to recognize handwritten digits.
- Showed an efficient approximation—*contrastive divergence*—of the derivative of the log of the probability.
  - Involved calculating the average of <v,h> after the initial presentation, and
  - Calculating the average of <v,h> after the first reconstruction.
  - Using their difference multiplied by a learning parameter as the basis of a gradient ascent scheme.

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Introduction to Neural Networks

Johns Hopkins University
Engineering for Professionals Program
605-447/625-438
Dr. Mark Fleischer

Module 11.4.1: RBM Mathematics, Insights and Getting Your Head Around It!

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

What We've Covered So Far

- Probabilistic foundations of RBMs.
  - *Energy/Consensus* associated with a visible and hidden pair of vectors.
  - Probability of a node's states

Goal

- Raise the probability that a visible vector with be faithfully reconstructed when a 'hidden' vector is presented to the visible layer.

Question:

How do we train an RBM so that 'reconstructions' are likely to create a reasonable facsimile of the original data?

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Weights → Energies → Probabilities

- Each possible joint configuration of the visible and hidden units has an energy
  - The energy is determined by the weights and biases (as in a Hopfield net).
- The energy of a joint configuration of the visible and hidden units determines its probability:

$$p(\mathbf{v},\mathbf{h}) \propto e^{-E(\mathbf{v},\mathbf{h})}$$

- The probability of a configuration over the visible units is found by summing the probabilities of all the joint configurations that contain it.

From Hinton 2007 (modified)

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Using energies to define probabilities

- The probability of a joint configuration over both visible and hidden units depends on the energy of that joint configuration compared with the energy of all other joint configurations.
- The probability of a configuration of the visible units is the sum of the probabilities of all the joint configurations that contain it.

$$p(\mathbf{v},\mathbf{h}) = \frac{e^{-E(\mathbf{v},\mathbf{h})}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}$$

partition function

$$p(\mathbf{v}) = \frac{\sum_{g} e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}$$

From Hinton 2007 (modified)

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

How Do We Train an RBM?

$$p(\mathbf{v}) = \frac{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}} \rightarrow \ln p(\mathbf{v}) = \ln\left(\frac{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}\right)$$

$$= \ln \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} - \ln \sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}$$

$$\frac{\partial \ln p(\mathbf{v})}{\partial w_{ij}} = \underbrace{\frac{\partial}{\partial w_{ij}} \ln \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}}_{\text{A}} - \underbrace{\frac{\partial}{\partial w_{ij}} \ln \sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}_{\text{B}}$$

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Looking at Term A:

$$\frac{\partial}{\partial w_{ij}} \ln \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} = \frac{1}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} \cdot \frac{\partial}{\partial w_{ij}} \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}$$

$$= \frac{1}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} \cdot \sum_g \frac{\partial e^{-E(\mathbf{v},\mathbf{h}^g)}}{\partial w_{ij}}$$

$$= \frac{1}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} \cdot \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} \cdot \frac{\partial\left(-E(\mathbf{v},\mathbf{h}^g)\right)}{\partial w_{ij}}$$

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Looking at Term A:

Recall that $E(\mathbf{v},\mathbf{h}) = -\sum_{i,j} v_i h_j w_{ij}$

$$\frac{\partial}{\partial w_{ij}} \ln \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} = \frac{1}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} \cdot \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} \cdot \frac{\partial\left(-E(\mathbf{v},\mathbf{h}^g)\right)}{\partial w_{ij}}$$

$$= \frac{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)} v_i h_j^g}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} = \sum_g p(\mathbf{h}^g \mid \mathbf{v}) v_i h_j^g = \langle v_i \cdot h_j \rangle_{\mathbf{v}}$$

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Where does that conditional probability come from?

$$\Pr\{A \mid B\} = \frac{\Pr\{A \cap B\}}{\Pr\{B\}}$$

$$p(\mathbf{h}^g \mid \mathbf{v}) = \frac{p(\mathbf{v},\mathbf{h}^g)}{p(\mathbf{v})} = \frac{\frac{1}{Z} e^{-E(\mathbf{v},\mathbf{h}^g)}}{\frac{1}{Z} \sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} = \frac{e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}}$$

$$\frac{e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}} = \frac{p\left(\mathbf{v},\mathbf{h}^g\right)}{p\left(\mathbf{v}\right)} = p(\mathbf{h}^g \mid \mathbf{v})$$

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Looking at Term B:

$$\frac{\partial}{\partial w_{ij}} \ln \sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)} = \frac{1}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}} \cdot \frac{\partial}{\partial w_{ij}} \sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}$$

$$= \frac{1}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}} \cdot \sum_{u,g} \frac{\partial e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}{\partial w_{ij}}$$

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Looking at Term B:

$$\frac{\partial}{\partial w_{ij}} \ln \sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)} = \frac{\sum_{u,g} \frac{\partial e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}{\partial w_{ij}}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}$$

$$= \frac{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)} \frac{\partial\left(-E(\mathbf{v}^u,\mathbf{h}^g)\right)}{\partial w_{ij}}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}$$

$$= \frac{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)} v_i^u h_j^g}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}} = \sum_{u,g} p(\mathbf{v}^u,\mathbf{h}^g) v_i^u h_j^g = \langle v_i \cdot h_j \rangle_{\mathbf{vh}}$$

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Basis of *Contrastive Divergence*

$$\frac{\partial \ln p(\mathbf{v})}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} \ln \sum_{g} e^{-E(\mathbf{v},\mathbf{h}^g)} - \frac{\partial}{\partial w_{ij}} \ln \sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}$$
$$= \langle v_i \cdot h_j \rangle_{\mathbf{v}} - \langle v_i \cdot h_j \rangle_{\mathbf{vh}}$$

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Using Contrastive Divergence

- Use the derivative to perform *stochastic* gradient ascent!

$$\frac{\partial \ln p(\mathbf{v})}{\partial w_{ij}} \quad \propto \quad \Delta w_{ij} = \eta\left(\langle v_i \cdot h_j \rangle_{\mathbf{v}} - \langle v_i \cdot h_j \rangle_{\mathbf{vh}}\right)$$

But why?

Engineering for Professionals

JOHNS HOPKINS
WHITING SCHOOL of ENGINEERING

Introduction to Neural Networks

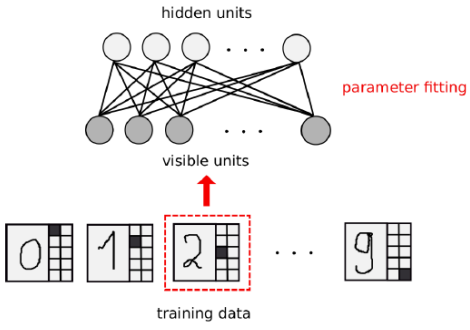
Johns Hopkins University
Engineering for Professionals Program
605-447/625-438
Dr. Mark Fleischer

Module 11.4.2: RBM Mathematics, Insights and Getting Your Head Around It!

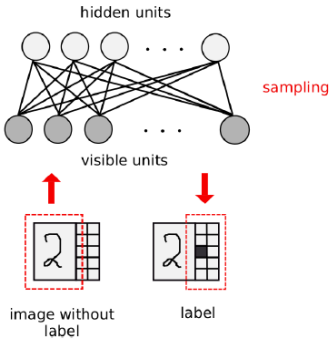
JOHNS HOPKINS
WHITING SCHOOL of ENGINEERING

Some Motivation

learning with labels



classification



From Fischer, Asja, "Training Restricted Boltzmann Machines", Doctoral Thesis, Faculty of Science, University of Copenhagen, 2014 at p.19

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Our Goal

- Strengthen the probability of reconstructed visible vectors so that they correspond to their probability of occurring in a training set of data.

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

The Energy/Probability Relationships

$$E(\mathbf{v},\mathbf{h}) = -\sum_{i,j} w_{ij} v_i h_j - \sum_i a_i v_i - \sum_j b_j h_j$$

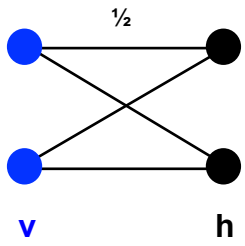
$$\Pr(\mathbf{v},\mathbf{h}) = \frac{e^{-E(\mathbf{v},\mathbf{h})}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}$$

$$\Pr(\mathbf{v}) = \frac{\sum_g e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}$$

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

A Numerical Example



We are going to increase the probability of the vector v = [0, 1].

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

So What is the Generative Model?

v	h	Joint Probability
0 0	0 0	
0 0	0 1	
0 0	1 0	
0 0	1 1	
0 1	0 0	
0 1	0 1	
0 1	1 0	
0 1	1 1	

v	h	Joint Probability
1 0	0 0	
1 0	0 1	
1 0	1 0	
1 0	1 1	
1 1	0 0	
1 1	0 1	
1 1	1 0	
1 1	1 1	

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

So What is the Generative Model?

	v1	v2	h1	h2	E	e^{-E}	Probability
1	0	0	0	0	0	1	0.031390208
2	0	0	0	1	0	1	0.031390208
3	0	0	1	0	0	1	0.031390208
4	0	0	1	1	0	1	0.031390208
5	0	1	0	0	0	1	0.031390208
6	0	1	0	1	-0.5	1.648721271	0.051753703
7	0	1	1	0	-0.5	1.648721271	0.051753703
8	0	1	1	1	-1	2.718281828	0.085327431
9	1	0	0	0	0	1	0.031390208
10	1	0	0	1	-0.5	1.648721271	0.051753703
11	1	0	1	0	-0.5	1.648721271	0.051753703
12	1	0	1	1	-1	2.718281828	0.085327431
13	1	1	0	0	0	1	0.031390208
14	1	1	0	1	-1	2.718281828	0.085327431
15	1	1	1	0	-1	2.718281828	0.085327431
16	1	1	1	1	-2	7.389056099	0.231944006
						31.8570685	1

All energy and probability values are based solely on the weights and biases!

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

What Does This Tell Us?

- The different configurations will occur with the indicated probability.
- How?
  - Present (activate) a initial visible vector onto the visible nodes (set their states).
  - Stochastically assign states to the hidden vector nodes.
  - Let the hidden vector nodes stochastically influence the assignment of states to the visible nodes, and so on.

If we do this back and forth for a very large number of cycles and count the occurances of the different vectors (configurations), they will occur with the frequency from the preceding table!

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

What is the Frequency of Occurrence of Visible Vector [0,1]?

From the table, we can calculate the marginal probability.

	v1	v2	h1	h2	E	e^{-E}	Probability
5	0	1	0	0	0	1	0.031390208
6	0	1	0	1	-0.5	1.648721271	0.051753703
7	0	1	1	0	-0.5	1.648721271	0.051753703
8	0	1	1	1	-1	2.718281828	0.085327431

$$p(\mathbf{v}) = \frac{\sum_{g} e^{-E(\mathbf{v},\mathbf{h}^g)}}{\sum_{u,g} e^{-E(\mathbf{v}^u,\mathbf{h}^g)}}$$

This sums to about 0.22022505.

All based on the energy values and Boltzmann distribution function.

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Let's See How Stochastic Update Functions are consistent with the table values.

- Thus, given a visible vector, the hidden vectors are assigned states with certain probabilities based on the activity function.
- Remember?

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Introduction to Neural Networks

Johns Hopkins University
Engineering for Professionals Program
605-447/625-438
Dr. Mark Fleischer

Module 11.4.3: RBM Mathematics, Insights and Getting Your Head Around It!

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

What are the probabilities of h given v?

$$\Pr\left\{h_1 = 1 \middle| \mathbf{v}\right\} = \frac{1}{1 + e^{-S_{h_1}/T}}$$

If S = 0, then this probability is 1/2.

Can you determine the first row of the table?

$$\Pr\left\{\mathbf{v}, \mathbf{h}\right\} = \Pr\left\{\mathbf{h} \middle| \mathbf{v}\right\} \times \Pr\left\{\mathbf{v}\right\}$$

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

What are the probabilities of h given v?

$$\Pr\left\{h_1 = 1 \middle| \mathbf{v}\right\} = \frac{1}{1+e^{-S_{h_1}/T}}$$

Let's assume for now that **v** = [0, 0]

So, $$S_{h_1} = \sum_i w_i v_i + \theta_i$$

$$= \tfrac{1}{2}\cdot 0 + \tfrac{1}{2}\cdot 0 + 0 = 0$$

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

The Generative Model

Since $S_{h1} = 0$, then

$$\Pr\left\{h_1 = 1 \middle| \mathbf{v}\right\} = \frac{1}{1+e^{-S_{h_1}/T}} = \frac{1}{1+1} = \frac{1}{2}$$

But for the first row of the table, we want to know the probability of $h_1 = 0$ (not 1). Also, we want to determine the probability of the **vector h given the vector $\mathbf{v}_1$**

Probability of **h**, given that $v_1 = [0, 0]$, is ¼.

This is because h_1 is independent of h_2.

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

So What is the Generative Model?

	v	h	Joint Probability
P_{v1}	0 0	0 0	$P_{v1} / 4$
	0 0	0 1	$P_{v1} / 4$
	0 0	1 0	$P_{v1} / 4$
	0 0	1 1	$P_{v1} / 4$
P_{v2}	0 1	0 0	
	0 1	0 1	
	0 1	1 0	
	0 1	1 1	

	v	h	Joint Probability
P_{v3}	1 0	0 0	
	1 0	0 1	
	1 0	1 0	
	1 0	1 1	
P_{v4}	1 1	0 0	
	1 1	0 1	
	1 1	1 0	
	1 1	1 1	

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

What are the probabilities of h given v?

$$\Pr\left\{h_1 = 1 \middle| \mathbf{v}_2\right\} = \frac{1}{1 + e^{-S_{h_1}/T}}$$

Now, **v** = [0, 1]

So, again, $S_{h_1} = \sum_i w_i v_i + \theta_i$

$$= \tfrac{1}{2} \cdot 0 + \tfrac{1}{2} \cdot 1 = \tfrac{1}{2}$$

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

What are the probabilities of h given v?

$$\Pr\left\{h_1 = 1 \middle| \mathbf{v}_2\right\} = \frac{1}{1+e^{-1/2}} = 0.622459331$$

So, given that **v** = [0, 1], Pr { **h** = [0,0] | $\mathbf{v}_2$ } = 0.3775 × 0.3775 = 0.1425.

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

So What is the Generative Model?

	v	h	Joint Probability
P_{v1}	0 0	0 0	P_{v1} / 4
	0 0	0 1	P_{v1} / 4
	0 0	1 0	P_{v1} / 4
	0 0	1 1	P_{v1} / 4
P_{v2}	0 1	0 0	P_{v2} • 0.1425
	0 1	0 1	
	0 1	1 0	
	0 1	1 1	

	v	h	Joint Probability
P_{v3}	1 0	0 0	
	1 0	0 1	
	1 0	1 0	
	1 0	1 1	
P_{v4}	1 1	0 0	
	1 1	0 1	
	1 1	1 0	
	1 1	1 1	

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

What are the probabilities of h given v?

$$\Pr\left\{h_1 = 1 \middle| \mathbf{v}_2\right\} = \frac{1}{1+e^{-1/2}} = 0.622459331$$

So, given that **v** = [0, 1], Pr { **h =** [0,1] | $\mathbf{v}_2$ } = 0.3775 × 0.6224 = 0.2350.

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

So What is the Generative Model?

	v	h	Joint Probability
P_{v1}	0 0	0 0	P_{v1} / 4
	0 0	0 1	P_{v1} / 4
	0 0	1 0	P_{v1} / 4
	0 0	1 1	P_{v1} / 4
P_{v2}	0 1	0 0	P_{v2} • 0.1425
	0 1	0 1	P_{v2} • 0.2350
	0 1	1 0	P_{v2} • 0.2350
	0 1	1 1	

	v	h	Joint Probability
P_{v3}	1 0	0 0	
	1 0	0 1	
	1 0	1 0	
	1 0	1 1	
P_{v4}	1 1	0 0	
	1 1	0 1	
	1 1	1 0	
	1 1	1 1	

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

What are the probabilities of h given v?

$$\Pr\left\{h_1 = 1 \middle| \mathbf{v}_2\right\} = \frac{1}{1+e^{-1/2}} = 0.622459331$$

So, given that **v** = [0, 1], Pr { **h =** [1,1] | $\mathbf{v}_2$ } = 0.6224 × 0.6224 = 0.3874.

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

So What is the Generative Model?

	v	h	Joint Probability
P_{v1}	0 0	0 0	P_{v1} / 4
	0 0	0 1	P_{v1} / 4
	0 0	1 0	P_{v1} / 4
	0 0	1 1	P_{v1} / 4
P_{v2}	0 1	0 0	P_{v2} • 0.1425
	0 1	0 1	P_{v2} • 0.2350
	0 1	1 0	P_{v2} • 0.2350
	0 1	1 1	P_{v2} • 0.3874

	v	h	Joint Probability
P_{v3}	1 0	0 0	
	1 0	0 1	
	1 0	1 0	
	1 0	1 1	
P_{v4}	1 1	0 0	
	1 1	0 1	
	1 1	1 0	
	1 1	1 1	

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

So What is the Generative Model?

	v	h	Joint Probability
P_{v1}	0 0	0 0	P_{v1} / 4
	0 0	0 1	P_{v1} / 4
	0 0	1 0	P_{v1} / 4
	0 0	1 1	P_{v1} / 4
P_{v2}	0 1	0 0	P_{v2} • 0.1425
	0 1	0 1	P_{v2} • 0.2350
	0 1	1 0	P_{v2} • 0.2350
	0 1	1 1	P_{v2} • 0.3874

	v	h	Joint Probability
P_{v3}	1 0	0 0	P_{v2} • 0.1425
	1 0	0 1	P_{v2} • 0.2350
	1 0	1 0	P_{v2} • 0.2350
	1 0	1 1	P_{v2} • 0.3874
P_{v4}	1 1	0 0	
	1 1	0 1	
	1 1	1 0	
	1 1	1 1	

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

What are the probabilities of h given v?

$$S_{v_4} = \tfrac{1}{2}\cdot 1 + \tfrac{1}{2}\cdot 1 = 1$$

$$\Pr\left\{h_1 = 1 \middle| \mathbf{v}_4\right\} = \frac{1}{1+e^{-1}} = 0.7310$$

So, given that **v** = [1, 1], Pr { **h** = [0,0] | $\mathbf{v}_2$ } = 0.2689 × 0.2689 = 0.0723.

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

So What is the Generative Model?

	v	h	Joint Probability
P_{v1}	0 0	0 0	P_{v1} / 4
	0 0	0 1	P_{v1} / 4
	0 0	1 0	P_{v1} / 4
	0 0	1 1	P_{v1} / 4
P_{v2}	0 1	0 0	P_{v2} • 0.1425
	0 1	0 1	P_{v2} • 0.2350
	0 1	1 0	P_{v2} • 0.2350
	0 1	1 1	P_{v2} • 0.3874

	v	h	Joint Probability
P_{v3}	1 0	0 0	P_{v3} • 0.1425
	1 0	0 1	P_{v3} • 0.2350
	1 0	1 0	P_{v3} • 0.2350
	1 0	1 1	P_{v3} • 0.3874
P_{v4}	1 1	0 0	P_{v4} • 0.0723
	1 1	0 1	P_{v4} • 0.1966
	1 1	1 0	P_{v4} • 0.1966
	1 1	1 1	P_{v4} • 0.5344

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

So, based on stochastic updating…

- What is the probability of v_2?
- Let's look at the 5th row of the preceding slide.

P_{v2} • 0.142536957 = 0.031390208

From the spreadsheet For the probability of the configuration [0,1], [0, 0]

Solving for P_{v2} = 0.220225047!

As expected, stochastic updating is consistent with the energy/ probability functions defined earlier ---- that was the basis of stochastic updating afterall!

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Introduction to Neural Networks

Johns Hopkins University
Engineering for Professionals Program
605-447/625-438
Dr. Mark Fleischer

Module 11.4.4: RBM Mathematics, Insights and Getting Your Head Around It!

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Let's Increase the Probability that v = [0, 1] occurs

$$\Delta w_{ij} = \eta\left(\langle v_i \cdot h_j \rangle_{\mathbf{v}} - \langle v_i \cdot h_j \rangle_{\mathbf{vh}}\right)$$

	v1	v2	h1	h2	E	e^{-E}	Probability
5	0	1	0	0	0	1	0.031390208
6	0	1	0	1	-0.5	1.648721271	0.051753703
7	0	1	1	0	-0.5	1.648721271	0.051753703
8	0	1	1	1	-1	2.718281828	0.085327431

Recall that the first term is ... $= \sum_g p(\mathbf{h}^g \mid \mathbf{v}) v_i h_j^g = \langle v_i \cdot h_j \rangle_{\mathbf{v}}$

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Let's Do Some Calculations

$$\langle v_i \cdot h_j \rangle_{\mathbf{v}} = \sum_g p(\mathbf{h}^g \,|\, \mathbf{v}) v_i h_j^g = \sum_g \left(\frac{p(\mathbf{h}^g, \mathbf{v})}{p(\mathbf{v})} \right) v_i h_j^g$$

5	0	1	0	0	0	1	0.031390208
6	0	1	0	1	−0.5	1.648721271	0.051753703
7	0	1	1	0	−0.5	1.648721271	0.051753703
8	0	1	1	1	−1	2.718281828	0.085327431

So for Δw_{11}, this term is:

$$\langle v_1 \cdot h_1 \rangle_{\mathbf{v}} = \left(\frac{0.0313}{0.2202} \right) \bullet 0 \bullet 0 + \left(\frac{0.0517}{0.2202} \right) \bullet 0 \bullet 0 + \left(\frac{0.0517}{0.2202} \right) \bullet 0 \bullet 1 + \left(\frac{0.0853}{0.2202} \right) \bullet 0 \bullet 1 = 0$$

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Now for the Second Term

Recall that $\sum_{u,g} p(\mathbf{v}^u, \mathbf{h}^g) v_i^u h_j^g = \langle v_i \cdot h_j \rangle_{\mathbf{vh}}$

	v1	v2	h1	h2	E	e^−E	Probability
1	0	0	0	0	0	1	0.031390208
2	0	0	0	1	0	1	0.031390208
3	0	0	1	0	0	1	0.031390208
4	0	0	1	1	0	1	0.031390208
5	0	1	0	0	0	1	0.031390208
6	0	1	0	1	−0.5	1.648721271	0.051753703
7	0	1	1	0	−0.5	1.648721271	0.051753703
8	0	1	1	1	−1	2.718281828	0.085327431
9	1	0	0	0	0	1	0.031390208
10	1	0	0	1	−0.5	1.648721271	0.051753703
11	1	0	1	0	−0.5	1.648721271	0.051753703
12	1	0	1	1	−1	2.718281828	0.085327431
13	1	1	0	0	0	1	0.031390208
14	1	1	0	1	−1	2.718281828	0.085327431
15	1	1	1	0	−1	2.718281828	0.085327431
16	1	1	1	1	−2	7.389056099	0.231944006
						31.8570685	1

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

The Second Term for v_1 h_1

So, $\sum_{u,g} p(\mathbf{v}^u, \mathbf{h}^g) v_1^u h_1^g = \langle v_1 \cdot h_1 \rangle_{\mathbf{vh}} = 0.45435257$

$$\begin{aligned}\Delta w_{11} &= \eta\left(\langle v_1 \cdot h_1 \rangle_{\mathbf{v}} - \langle v_1 \cdot h_1 \rangle_{\mathbf{vh}}\right) \\ &= 0.1\left(0 - 0.45435257\right) \\ &= -0.045435257\end{aligned}$$

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Doing the Same Calculations for all the other weights, we get …

$$\langle v_1 \cdot h_1 \rangle_{\mathbf{v}} = \left(\frac{0.0313}{0.2202}\right)\bullet 0 \bullet 0 + \left(\frac{0.0517}{0.2202}\right)\bullet 0 \bullet 0 + \left(\frac{0.0517}{0.2202}\right)\bullet 0 \bullet 0 + \left(\frac{0.0853}{0.2202}\right)\bullet 0 \bullet 0 = 0$$

$$\langle v_1 \cdot h_2 \rangle_{\mathbf{v}} = \left(\frac{0.0313}{0.2202}\right)\bullet 0 \bullet 0 + \left(\frac{0.0517}{0.2202}\right)\bullet 0 \bullet 1 + \left(\frac{0.0517}{0.2202}\right)\bullet 0 \bullet 0 + \left(\frac{0.0853}{0.2202}\right)\bullet 0 \bullet 1 = 0$$

$$\langle v_2 \cdot h_1 \rangle_{\mathbf{v}} = \left(\frac{0.0313}{0.2202}\right)\bullet 1 \bullet 0 + \left(\frac{0.0517}{0.2202}\right)\bullet 1 \bullet 0 + \left(\frac{0.0517}{0.2202}\right)\bullet 1 \bullet 1 + \left(\frac{0.0853}{0.2202}\right)\bullet 1 \bullet 1 = 0.622459331$$

$$\langle v_2 \cdot h_2 \rangle_{\mathbf{v}} = \left(\frac{0.0313}{0.2202}\right)\bullet 1 \bullet 0 + \left(\frac{0.0517}{0.2202}\right)\bullet 1 \bullet 1 + \left(\frac{0.0517}{0.2202}\right)\bullet 1 \bullet 0 + \left(\frac{0.0853}{0.2202}\right)\bullet 1 \bullet 1 = 0.622459331$$

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Now for all the Second Terms

$$\langle v_1 \cdot h_1 \rangle_{\mathbf{vh}} = 0.0517 + 0.0853 + 0.0853 + .2319 = 0.454352573$$
$$\langle v_1 \cdot h_2 \rangle_{\mathbf{vh}} = 0.0517 + 0.0853 + 0.0853 + .2319 = 0.454352573$$
$$\langle v_2 \cdot h_1 \rangle_{\mathbf{vh}} = 0.0517 + 0.0853 + 0.0853 + .2319 = 0.454352573$$
$$\langle v_2 \cdot h_2 \rangle_{\mathbf{vh}} = 0.0517 + 0.0853 + 0.0853 + .2319 = 0.454352573$$

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Updated Weights

$$\Delta w_{11} = \eta\left(\langle v_1 \cdot h_1 \rangle_{\mathbf{v}} - \langle v_1 \cdot h_1 \rangle_{\mathbf{vh}}\right) = 0.1\left(0 - 0.45435257\right) = -0.045435257$$
$$\Delta w_{12} = 0.1\left(0 - 0.45435257\right) = -0.045435257$$
$$\Delta w_{21} = 0.1\left(0.622459331 - 0.45435257\right) = 0.016810676$$
$$\Delta w_{22} = 0.1\left(0.622459331 - 0.45435257\right) = 0.016810676$$

Engineering for Professionals

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

The Updated Configurations

	v1	v2	h1	h2	E	e^-E	Probability
1	0	0	0	0	0	1	0.032197018
2	0	0	0	1	0	1	0.032197018
3	0	0	1	0	0	1	0.032197018
4	0	0	1	1	0	1	0.032197018
5	0	1	0	0	0	1	0.032197018
6	0	1	0	1	-0.516810676	1.676671664	0.053983827
7	0	1	1	0	-0.516810676	1.676671664	0.053983827
8	0	1	1	1	-1.033621352	2.811227869	0.090513154
9	1	0	0	0	0	1	0.032197018
10	1	0	0	1	-0.454564743	1.575487492	0.050725999
11	1	0	1	0	-0.454564743	1.575487492	0.050725999
12	1	0	1	1	-0.909129486	2.482160837	0.079918176
13	1	1	0	0	0	1	0.032197018
14	1	1	0	1	-0.971375419	2.641575235	0.085050845
15	1	1	1	0	-0.971375419	2.641575235	0.085050845
16	1	1	1	1	-1.942750838	6.97791972	0.224668205
						31.05877721	1

So now the total probability Pr{v=[0,1]} = 0.230677826

If eta = 1, the probability goes up to 0.332961042!

Recall, it was 0.22022505

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Hinton's Approximation in vector form

1. Take a training sample v, compute the probabilities of the hidden units and sample a hidden activation vector h from this probability distribution.
2. Compute the outer product of **v** and **h** and call this the positive gradient.
3. From **h**, sample a reconstruction **v'** of the visible units, then resample the hidden activations **h'** from this. (Gibbs sampling step)
4. Compute the outer product of **v'** and **h'** and call this the negative gradient.
5. Let the update to the weight matrix *W* be the positive gradient minus the negative gradient, times some learning rate: $\Delta W = \epsilon (\mathbf{v}\mathbf{h}^{\mathsf{T}} - \mathbf{v'}\mathbf{h'}^{\mathsf{T}})$.
6. Update the biases *a* and *b* analogously: $\Delta a = \epsilon(\mathbf{v} - \mathbf{v'})$, $\Delta b = \epsilon (\mathbf{h} - \mathbf{h'})$.

From Wikipedia.

Summary

- Showed the derivative of the log probability with respect to weights
- This can serve as the basis of a gradient ascent method for increasing the probability of the reconstructed vector v.