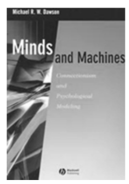
Psychology 452
Week 12: Deep Learning

- **What Is Deep Learning?**
- **Preliminary Ideas (that we already know!)**
- **The Restricted Boltzmann Machine (RBM)**
- **Many Layers of RBMs**
- **Pros and Cons of Deep Learning**

Course Structure

When	What
Weeks 1, 2, 3	Connectionist Building Blocks
Weeks 4, 5, 6	Case Studies of Connectionism
Week 7	Midterm Exam
Weeks 8, 9, 10	Interpreting Connectionist Networks
Weeks 11, 12	**Deep Learning Basics**
Week 13	Final Exam

Chapter 8 Discussion



- **Questions?**
- **Important Terms**
  - **Visuomotor module**
  - **Synthesis and representation**
  - **Metric space**
  - **Minimality principle**
  - **Symmetry principle**
  - **Triangle inequality principle**
  - **Cognitive map**
  - **Place cell**
  - **Allocentric coding**

Part I: Deep Learning: What and Why?

Rufus

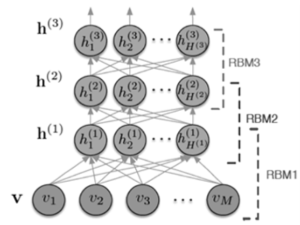
- **Rufus is supposedly a pug crossed with shih tzu, though we think he is part terrier.**

Rufus Processed By Fetch

- **A new deep learning app from Microsoft Garage, called Fetch, is 99% sure that Rufus is something else, based on appearance**

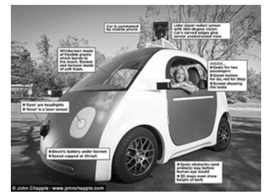
What Is Deep Learning?

- Deep learning trains networks of many layers of hidden units
- Each layer provides a more abstract model of the previous layer
- Can be used to learn regularities in huge datasets
- Saviour of AI research?



Who Uses Deep Learning?

- IBM's Watson
- Google Car
- Google Glasses
- Facebook face recognition
- Image processing
- Speech processing
- Medical applications
- Mining big data
- Etc, etc, etc

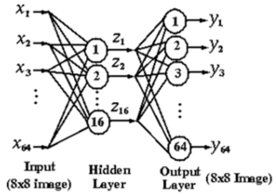


Why We Should Learn About It

Part II: Models and Probability

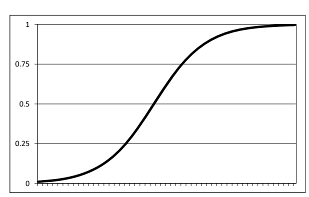
Hidden Units As Models

- In an encoder network, a small number of hidden units is used to compress input data, and then reconstruct it
- The hidden units provide an abstract model of the input patterns



The Logistic Equation

- The logistic equation is the core of much neural network theory
- It also has nice properties that relate it to mathematical definitions of probability



$$a_i = \frac{1}{1 + e^{(-(net_i + \theta))}}$$

Probabilistic Choices

- Logistic activity = probability of getting a reward given the current set of cues
- We can use this probability to make a choice – make the choice (1) with the probability given by the logistic
- This can be used to model operant conditioning

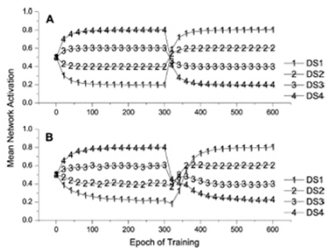
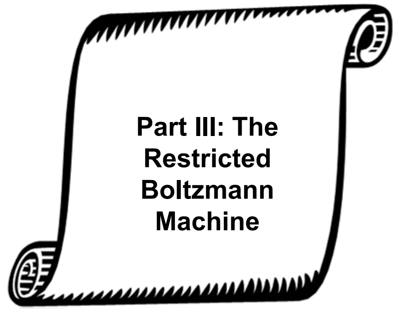


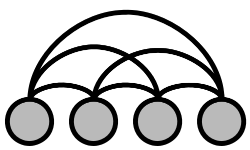
Fig. 1. Average responses of ten different perceptrons to each of the four stimuli as a function of training epoch. (a) Responses for networks from the first simulation which used standard training procedures. (b) Responses for networks from the second simulation which used an operant training procedure.



Part III: The Restricted Boltzmann Machine

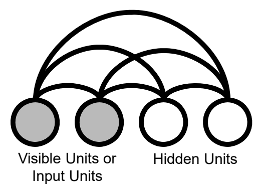
The Hopfield Network

- Simplest version:
- Fully connected
- Every unit is an input unit
- Unit activities are binary
- Variations of this architecture lead us to deep belief networks



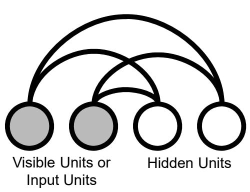
The Boltzmann Machine

- Like “a Hopfield network with hidden units”
- Fully connected
- Input units can be activated by the environment
- Hidden units can only be affected by other units
- Generates probabilistic models of inputs



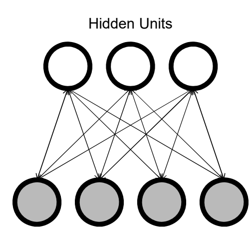
The Restricted Boltzmann Machine

- A Boltzmann machine with fewer connections
- <u>Not</u> fully connected
- Input units not connected to one another
- Hidden Units not connected to one another
- Input units can be activated by the environment
- Hidden units can only be affected by other units
- Generates probabilistic models of inputs



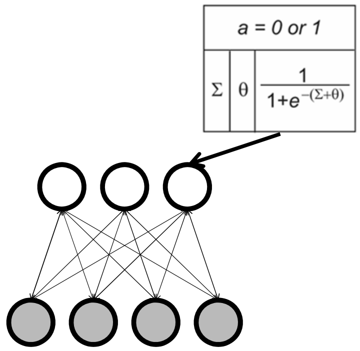
RBM In Layers

- Hidden units serve as a model of the inputs
- Signals from hidden units ideally will reconstruct input patterns with the same probability as observed in the real world



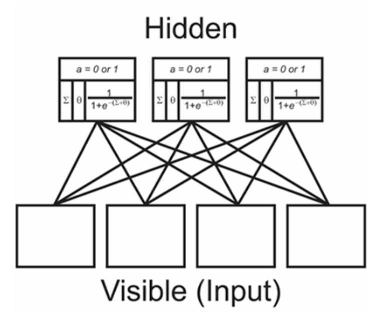
Hidden Unit As Operant Perceptron

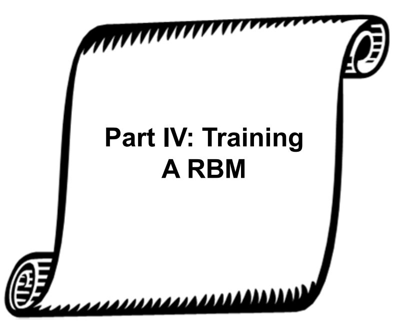
- One hidden unit in a RBM is like an operant perceptron:
  - Compute net input
  - Add bias
  - Compute logistic of net input
  - Turn on with probability of the logistic



RBM As Operant Perceptron

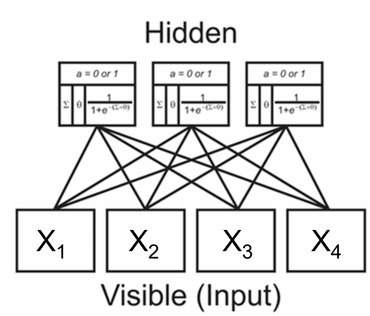
- Hidden units need to learn how to model the probability distribution of input binary patterns
- This is done with an algorithm called contrastive divergence





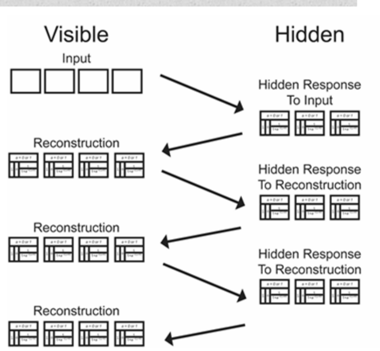
Step 1: Get Numbers From A Pattern

- Present a pattern to the input units
- Record the probability generated for each hidden unit (the logistic of the net input)
- We will use the pattern and these probabilities soon



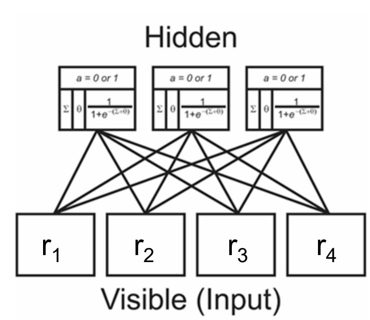
Step 2: Reconstruct Pattern

- Use hidden units to send signal to reconstruct input
- Possibly iterate k times
- For many RBMs k = 1
- This is called Gibbs sampling



Step 3: Get Numbers From The Reconstructed Pattern

- Present the reconstructed pattern to the input units
- Record the probability generated for each hidden unit (the logistic of the net input)
- We will use the pattern and these second probabilities next



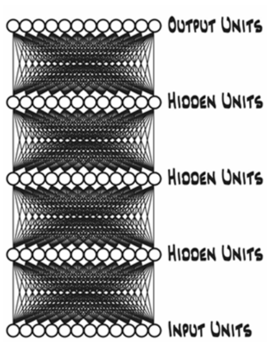
Step 4: Update The Weights

- Let $P(H_j|x_i)$ be the probability for hidden unit j given the activity of input unit i – that is, its logistic value; define a similar term for when the reconstructed pattern is presented
- $W_{ij}^{new} = W_{ij}^{old} + \alpha(x_i \cdot P(H_j|x_i) - r_i \cdot P(H_j|r_i))$
- Related equations can be used to update the biases of the hidden units, as well as the biases of the input units
- In matrix form, the difference between two outer products is computed, weighted, and added to the weight matrix
- Why does this rule seem to make sense?

Part V: Deep Learning: Layers of RBMs

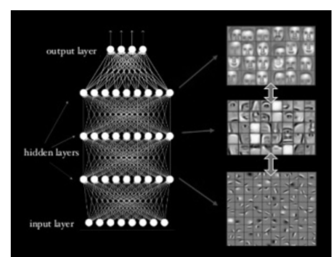
Deep Belief Network

- A deep belief network consists of many layers of hidden units
- We can consider each layer to be a restricted Boltzmann machine
- Each layer of units provides a model of the layer that feeds into it



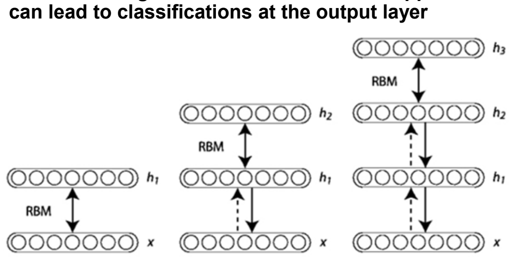
Levels of Abstraction

- Moving from the input layer towards the output layer, each layer of hidden units provides a more abstract model of the input
- This is something like a recursive encoder network
- Each layer exploits a more abstract set of features



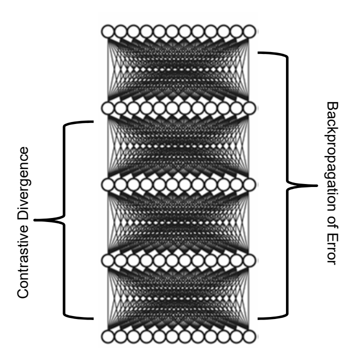
Training Strategy 1: Unsupervised

- Train the network layer by layer using the RBM approach we have already seen
- Some have argued that variations of this approach can lead to classifications at the output layer



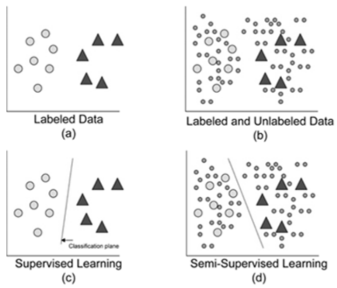
Training Strategy 2: Two Stages

- First, use unsupervised learning to generate layers of abstract features
- Second, use backpropagation to fine tune category judgements represented in the output layer



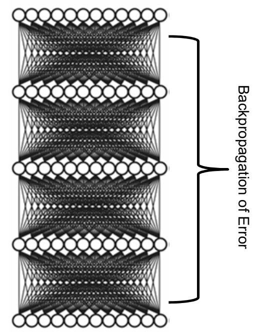
Training Strategy 3: Semi Supervised

- Use unsupervised learning to get layers of features from all data
- Use supervised learning on subset of data that is labeled
- Permits training on large data sets when not all instances are labeled



Why Not Backprop?

- Why don't we simply use backpropagation of error to train deep belief networks?
- In principle, should work
- In practice, #fail
- Problem of the vanishing gradient



Part VI: Deep Learning and Cognitive Science

Deep Learning Pros

- Deep learning is hot because it has led to many successes
- Some argue that it is appropriate for models in cognitive science because deep belief nets are more brain-like than are other nets
- www.youtube.com/watch?v=CEv_0r5huTY

Deep Learning Cons

- Deep learning may provide compelling artifacts
- However, such networks are very difficult to interpret
- They provide a steep and slippery slope into Bonini's paradox

Charles Pius Bonini