Attractor Neural Networks

Presented by Janet Snape

CS 790R Seminar

University of Nevada, Reno

February 10, 2005

References

- Bar-Yam, Y. (1997) *Dynamics of Complex Systems*
  - Chapter 2: *Neural Networks I*
- Flake, G. (1998) *The Computational Beauty of Nature*
  - Chapter 18: *Natural and Analog Computation*

Introduction

- Bar-Yam -- Chapter Concepts
  - Mathematical models correlate to the human information process.
  - Associative content-addressable memory.
  - Imprinting and retrieval of memories.
  - Storage capacity
- Flake -- Chapter Concepts
  - Examination of artificial NNs
  - Associative memory
  - Combinatorial optimization problems and solutions.

Overview

- Introduction
- Neural Network
- A typical neuron
- Neural network models
  – Artificial neural networks
- Extensions
  - Associative memory / Hebbian learning
  - Recalling letters
  - Hopfield NNs
- Summary

Neural Network: Brain and Mind

- Elements responsible for brain function:
  - Neurons and the interactions between them

A “Typical” Neuron

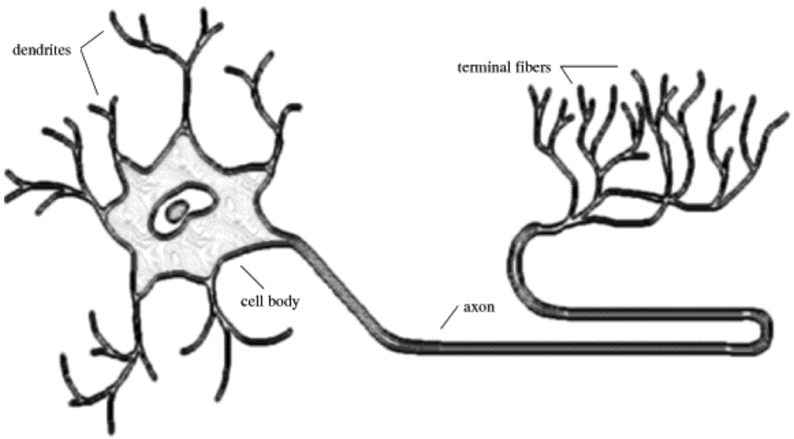
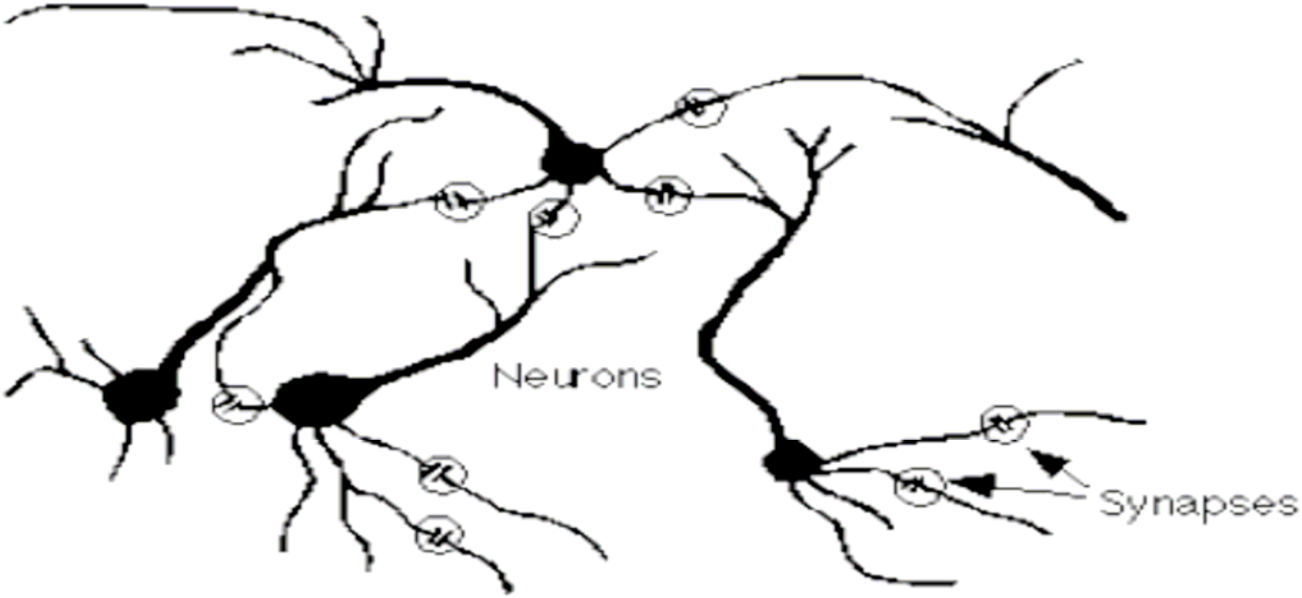


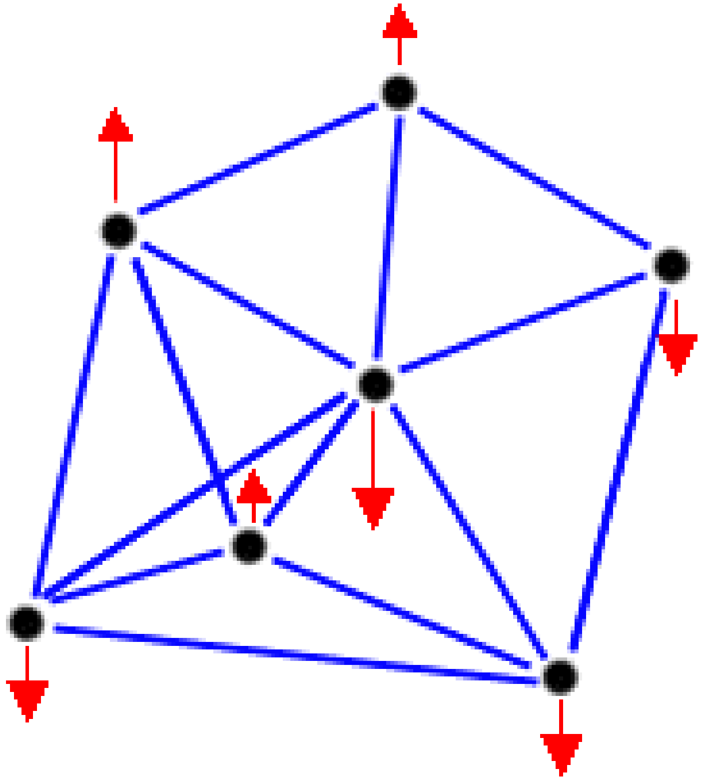
Figure 18.1 A “typical” neuron with major components identified

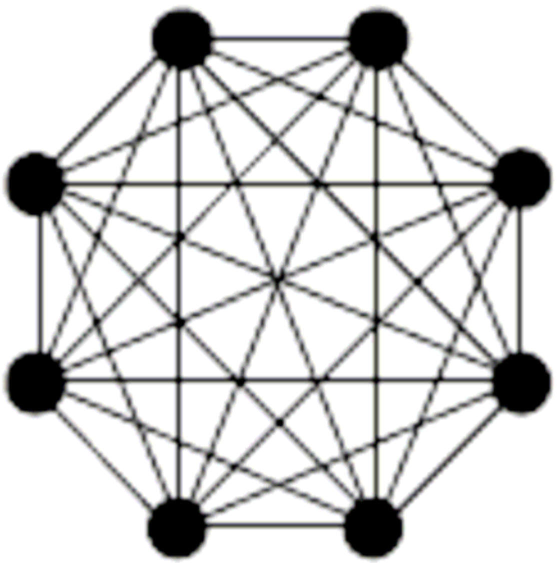
- Interface
  - Cell body
  - Dendrites
  - Axon
  - Synapses
- Behavior
  - Send / receive signals
  - “Fired”
  - Recursive stimulation

Neural Network Example



Artificial Neural Network Contrast





Development of Neural Networks

- McCulloch-Pitts discrete model (1940s)
- Associative Hebbian Learning (1949)
- Hopfield continuous model (1980s)

McCulloch-Pitts

- McCulloch-Pitts NN (1940s)
  - Realistic model
  - Neuron has a discrete state and and changes in discrete time steps

A single McCulloch-Pitts neuron

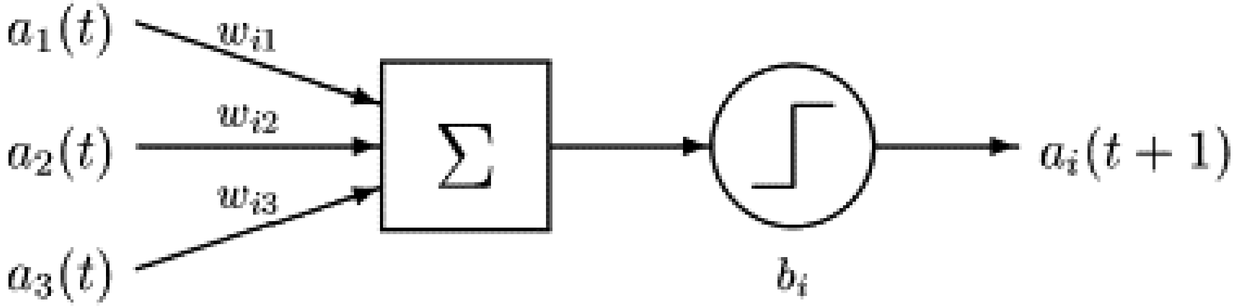


Figure 18.2 A single McCulloch-Pitts neuron

McCulloch-Pitts cont'd...

$$a_i(t+1) = \Theta\left(\sum_{j=1}^{n} w_{ij} \times a_j(t) - b_i\right)$$

<u>A Neuron's State Activation Rule</u>

IF the weighted sum of incoming signals > than the threshold b_i --- neuron fires with an activation of 1

(i.e., send signal)

ELSE

neuron's activation is 0

(i.e., cannot send signal)

Artificial NNs

- Types of artificial NNs
  - Feedback networks
  - Attractor networks

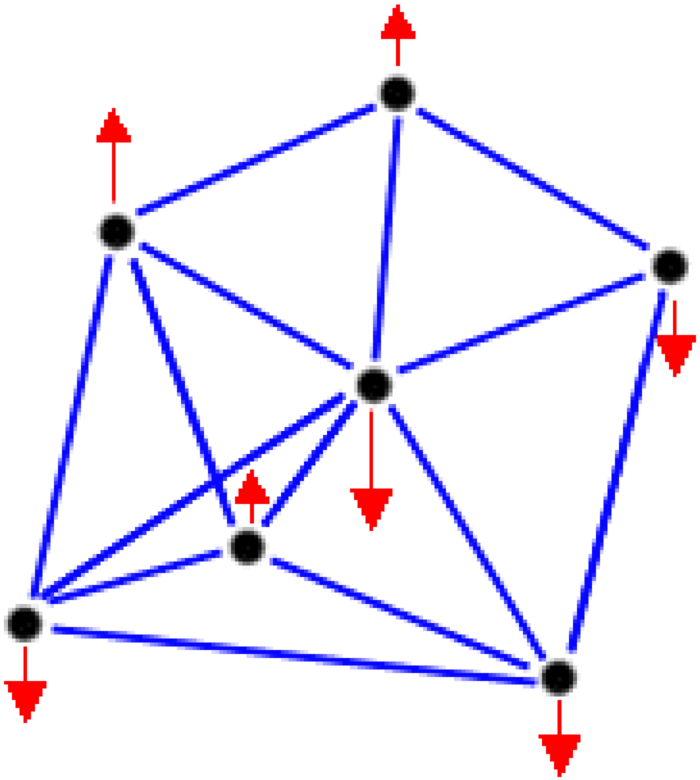
Feedback Neural Networks

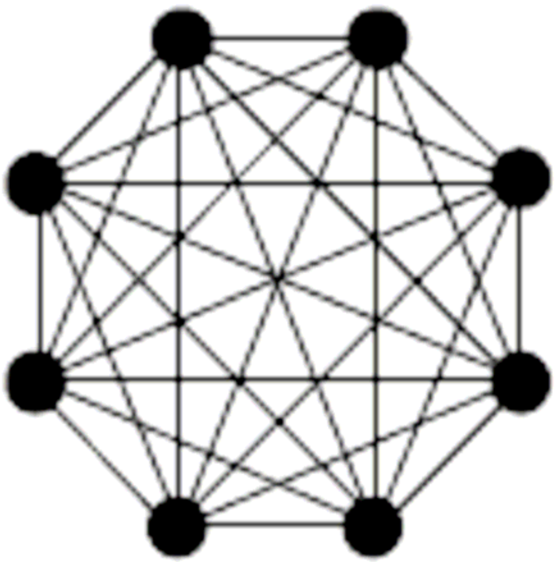
- NNs have a collection of neurons
  - Fixed set of neurons and thresholds (w_{ij} and b_i)
- Neuron state activation
  - ***Synchronous*** (all neurons at once / deterministic)
  - ***Asynchronous*** (one neuron at a time / realistic)

Extensions of Artificial NNs

- Attractor networks (Hopfield NNs)
- Associative memory / Hebbian learning (1949)
- Recalling letters

Schematic of an Attractor Network





Defining an Attractor Network

- Definition
- Operating and training attractor networks
- Energy

Features of Attractor Networks

- Binary variables for the neuron activity values +1 (ON) -1 (OFF)
- No self-action by a neuron.
- Symmetric synapses.
- Synchronous / asynchronous neuron activity update Eq (2.2.4):

$$s_i(t) = \mathrm{sign}\left(\sum_j J_{ij} s_j(t-1)\right)$$

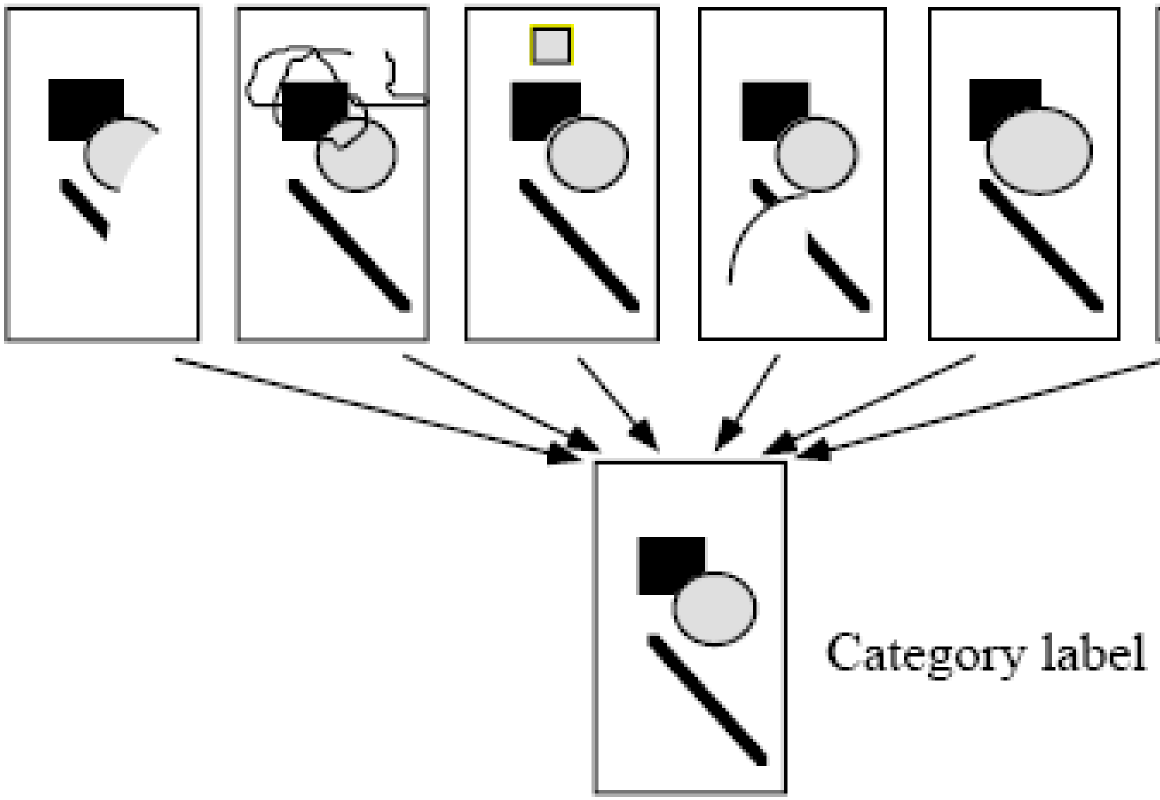
Attractor Network Process (Operation)

- Input pattern
- Continuous evolution to a steady state
- Output network state

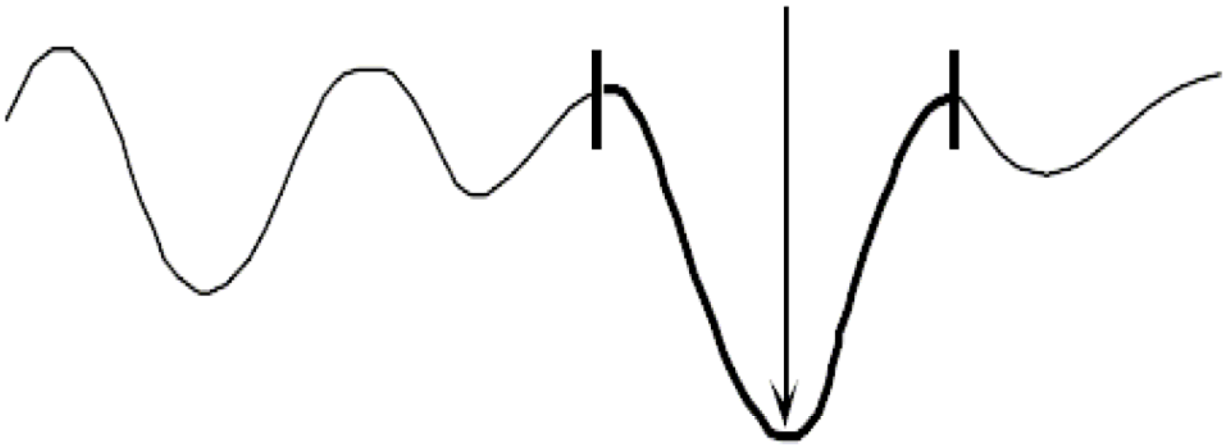
Training Attractor Networks

- Consists of imprinting a set of selected neuron firing patterns.
- Described as an associative memory.

ANNs Categorize



Energy analog



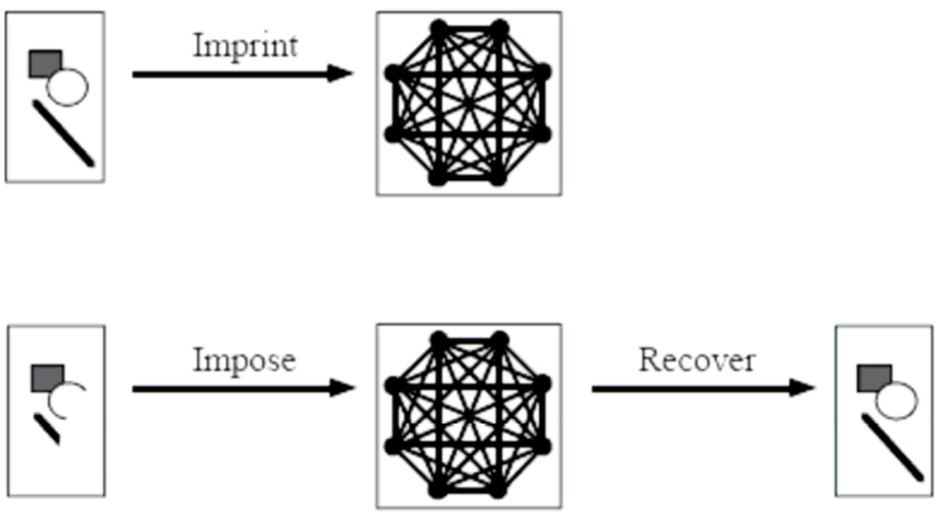
Imprinting on an attractor network.

Basin of Attraction

- The basin of attraction is the region of patterns, near the imprinted pattern that will evolve under the neural updating back to the imprinted pattern (feedback loop).

Attractor NNs cont'd...

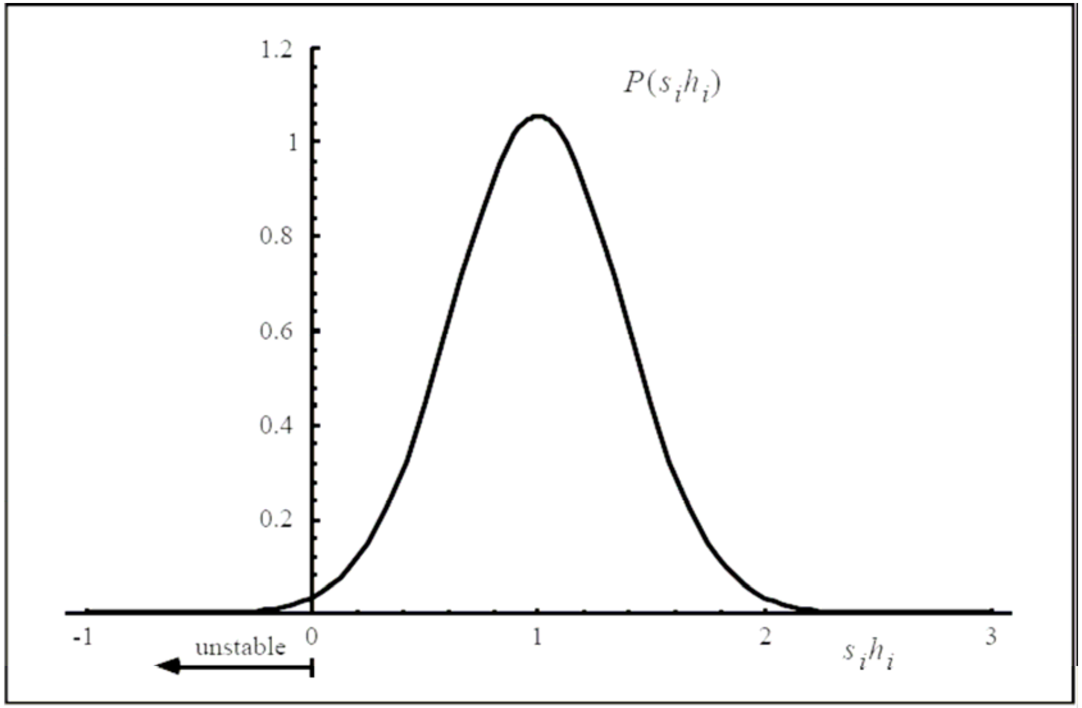
- Associative content-addressable memory.
- Memory capacity important.



Memory Capacity

- Overloaded => Full => Basin of attraction is small
- Memory not usable.
- Can only recover pattern if we already know it.

Stability of Imprinted Pattern



Probability distribution of neuron activity times the local field $s_i h_i$

Associative Memory

- Content-addressable memory
- How does associative memory work?
- Rule: ***Hebbian learning***

Hebbian Learning

- Donald Hebb (1949)
- Reinforces neurons either *on* or *off* at the same time.
- Stores memory that can be recalled.

Hebbian learning cont'd...

- Computation
- How does recall work?
  - Associative memory
  - Recall letters

Recall Letters

- Train
- Find similarities between *seed* pattern and the stored pattern
- Converges to a single stored pattern
- Computational biases

Recall letters cont'd…

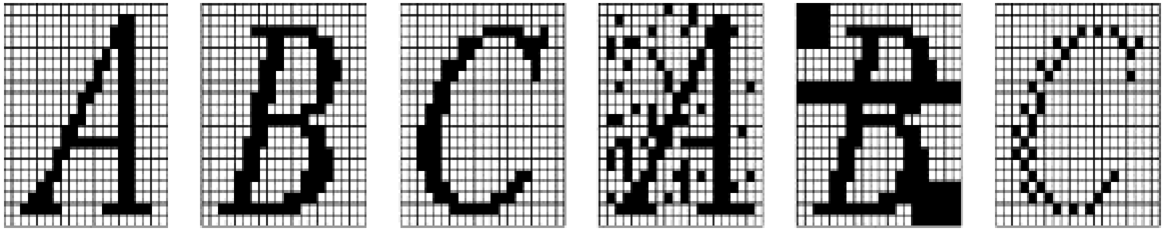


Figure 18.3 Bitmapped images of letters from the alphabet: The first three are clean version that are used as patterns to be stored. The last three are used as seed images that the associative memory must use to recall one of the first three.

Recall letters cont'd…

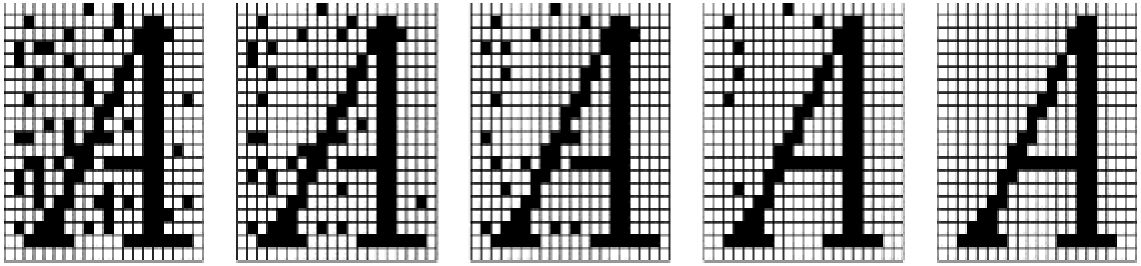


Figure 18.4 Recalling the letter A from a damaged seed image

Hebbian Unstable Imprints

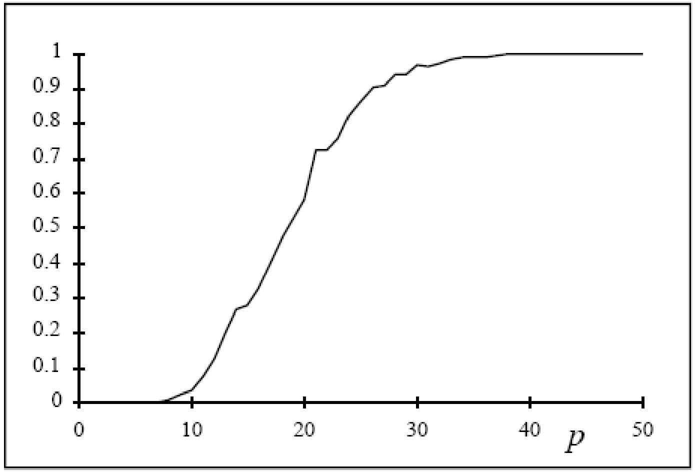


Figure 2.2.5 Fraction of unstable imprints as a function of the number of imprints p on a neural network of 100 neurons using Hebbian imprinting. For p less than 10 the stability of all of the stored patterns is perfect. Above this value the percentage of unstable patterns increases until all patterns are unstable. ∎

- < 10, !00% stability of all stored patterns
- > 10, unstable patterns increase until ALL patterns are unstable

Hebbian Stable Imprints

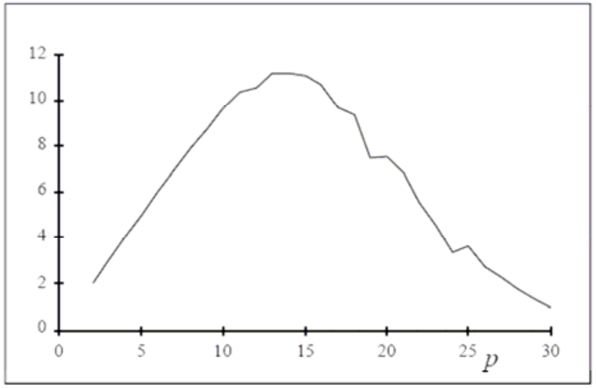


Figure 2.2.6 Number of stable imprints as a function of the number of imprints p on a neural network of 100 neurons using Hebbian imprinting. For p less than 10 all patterns are stable. The maximum number of stable imprinted patterns is less than 12. Above 15 imprints the number of stable patterns decreases gradually to zero. However, throughout this regime the basins of attraction of the patterns are very small and the system is not usable as a memory. ∎

- Max number of stable patterns = 12
- < 10 patterns, stable patterns
- > 15 patterns, number of stable patterns decrease to 0

Pros and Cons

- Advantages
  - Efficient for some applications
  - Fault tolerant.
- Disadvantages
  - Prone to recall a composite of many of the stored patterns.

Hopfield NN Model (1980s)

- Internal continuous state continuously varies over time
- External state is a function of the internal state.

Hopfield cont'd...

- Network starts out in a random state with each neuron close to 0
- Update activations
- Set weights and inputs to solve a problem.
- Application example: Cost optimization
- Task assignment problem
- Task assignment solution

Task Assignment Problem

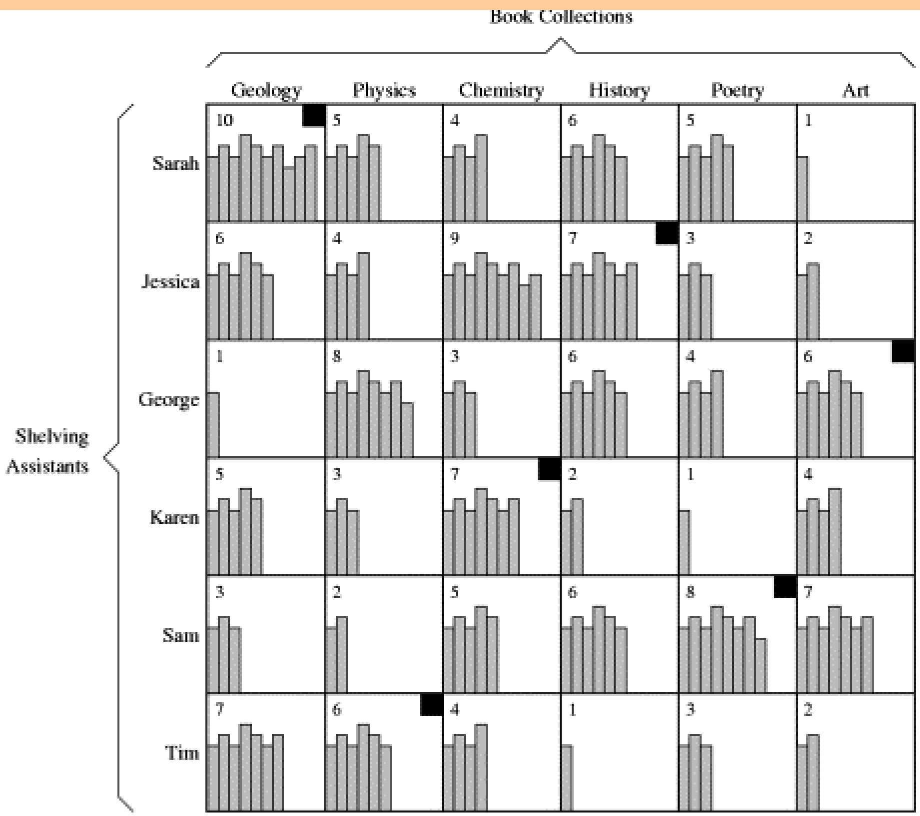


Figure 18.6 The task assignment problem: Black squares in the entries denotes the optimal assignment with a total shelving rate of 44.

Task Assignment Solution

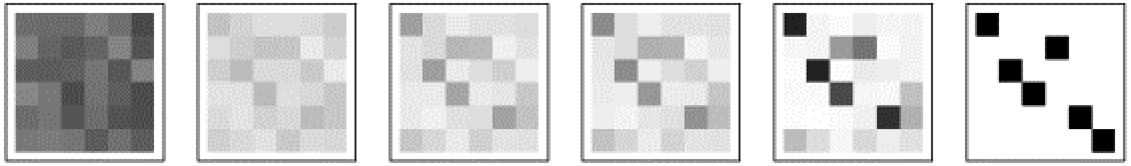


Figure 18.7 Computing a neural solution to the task assignment problem: This particular solution yields a total rate of 42, which is just less than the optimal solution.

Size of Basin of Attraction

- More imprints => smaller basin of attraction
- More imprints => unstable patterns
- More imprints => affects storage capacity
- More imprints => unable to retrieve pattern

BofA cont'd...

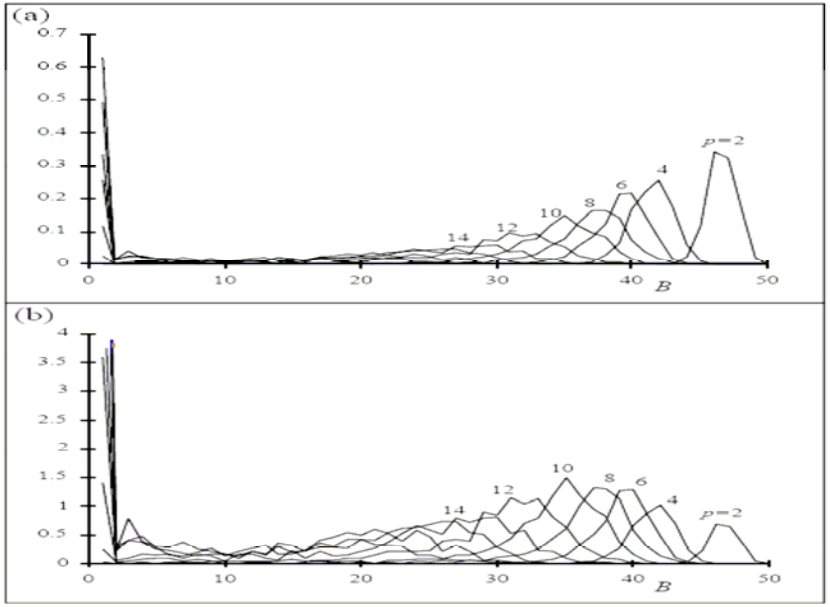


Figure 2.2.7 (a) Probability distribution of the size of the basin of attraction of imprinted patterns for a neural network of 100 neurons, and (b) histograms of the number of imprinted patterns with a particular basin of attraction. The horizontal axis is the Hamming distance, which measures the size of the basin of attraction. The probability distributions are normalized to 1, while the histograms are normalized to p. Each curve is for a different number of imprinted patterns as shown. The size of the basin of attraction decreases as the number of imprints increases. The probability distribution also broadens. When the number of imprints becomes greater than 10, the number of imprints with basins of attraction of zero begins to increase. This is the probability that a pattern is unstable as shown in Fig. 2.2.4. ∎

Hamming Distance

- The distance between two patterns =

  the number of neurons that differ between the two patterns.
- Used in the Hopfield java applet demonstration
- Eq. 2.2.16

$$d(\mathbf{s},\mathbf{s}') = \frac{N}{2} - \frac{1}{2}\sum_i s_i s_i'$$

Demonstration

- Hopfield Java Applet…

Summary

- Attractor neural networks can be used to model the human brain.
- These networks developed from the simple McCulloch-Pitts 1940s NN discrete model into other extensions:
  - Associative memory led to Hebbian learning.
  - Hopfield NN continuous model led to a more general use
- Thus, ANNs can be used to solve combinatorial problems.
- New patterns => learning.
- No new patterns => no learning.
- **Storage capacity depends on the number of neurons!**

Questions and Answers

- Open discussion...