CONNECTIONISM (I)

COGS 200

GOALS FOR TODAY

- Understand the basic architecture of a connectionist network (and the rationale for it).
- Understand how networks can be trained to model cognitive processes.
- Understand the properties that make connectionist representations special (as forms of knowledge representation).
- See one objection to the biological plausibility of connectionism.

LESSONS FROM AMBIGUITY

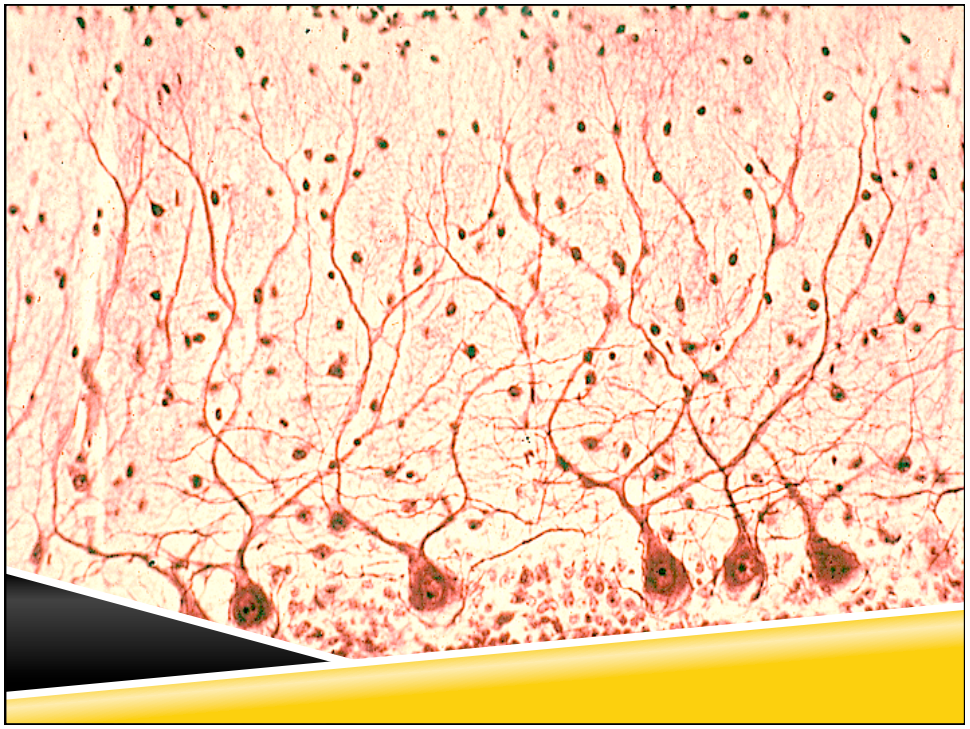
- Some ambiguities are *lexical*, some are *structural*.
- Therefore, understanding a sentence is not just a matter of knowing what its constituent words mean.
- If understanding enables us to disambiguate, it must also require a representation of structure.

Chomsky's guiding insight:

- Any system that is able to cope with human verbal behaviour must employ structured transformable representations.

COMPUTATIONALISM

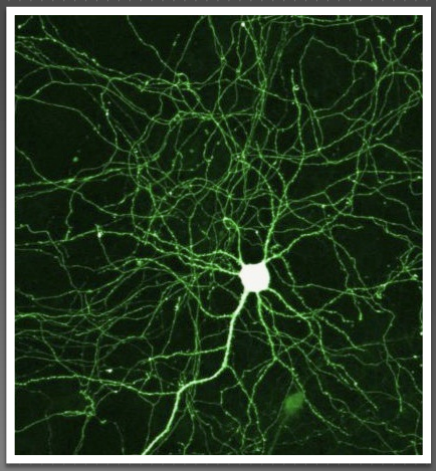
- Any system that is able to cope with human verbal behaviour must have structured transformable representations.
- We are able to cope with human verbal behaviour.
- Therefore: We must have structured transformable representations (presumably, they are somewhere in our heads).



- Nothing in the brain looks like a computer.
- Nothing in the mind behaves quite like a computer:
  - Computers have characteristic patterns of breakdown.
  - They have characteristic patterns of learning.
  - They operate in a step by step sequence

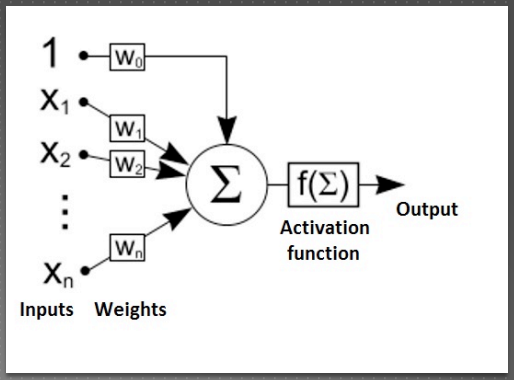
The Neuron

Cell Body
Dendrites
Axon
Synapses



The McColloch Pitts Model of the Neuron

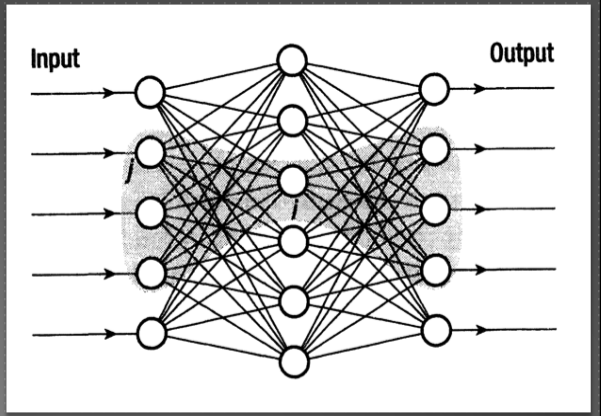
A set of weighted inputs.
A simple summation of those inputs.
A simple function from that sum of inputs to a single output.

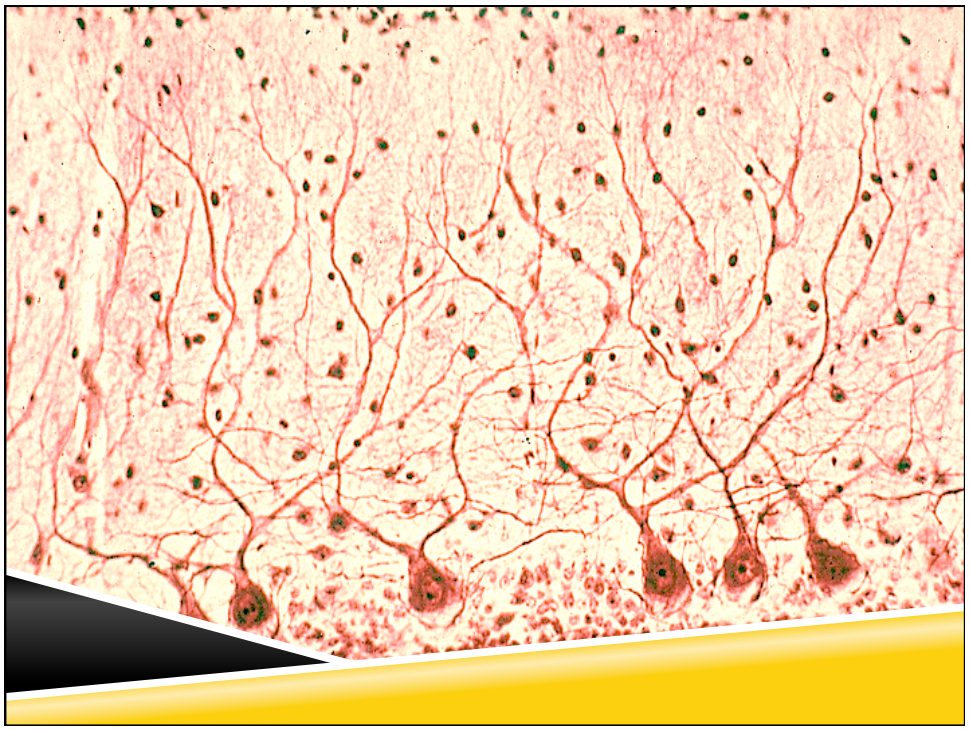


A Neural Network

Typical features:

- Three layer, feed forward architecture.
- Each neuron at each level passes its output to every neuron at the next level.





A PARALLEL DISTRIBUTED PROCESSOR

- Take several model neurons and arrange them into layers, such that the output of every neuron in the *n*th layer is received as an input by every neuron in layer *n+1*.
- Assign weights to the connections (in some arbitrary way).
- Determine some code so that the input to your problem can be represented as a sequence of numbers between 0 and 1.
- Give these numbers to the neurons in the first layer of the network.
- Let the network do its thing.
- It will give you another sequence of numbers between zero and one.

GETTING THE PROCESSOR TO DO SOMETHING USEFUL

- Settle upon some way of interpreting the sequences that the machine gives you.
- Settle upon some criterion of what is going to count as a correct answer.
- Compare the answers you get with the correct answer.
- Use the difference to determine how much you want to adjust the weights between the neurons.
- "Backpropagation of Error"

USEFUL THINGS

- Let the inputs by acoustic profiles and the output be classifications of those profiles.
- Let the inputs be pictures and the outputs be those same pictures.
- Let the inputs be words and the outputs be pronunciations of those words.
- Let the inputs be verb stems and the outputs be past tense inflections of those verb stems.

USEFUL THINGS

- Let the inputs by acoustic profiles and the output be classifications of those profiles.

MINE VS. ROCK SONAR REFLECTIONS

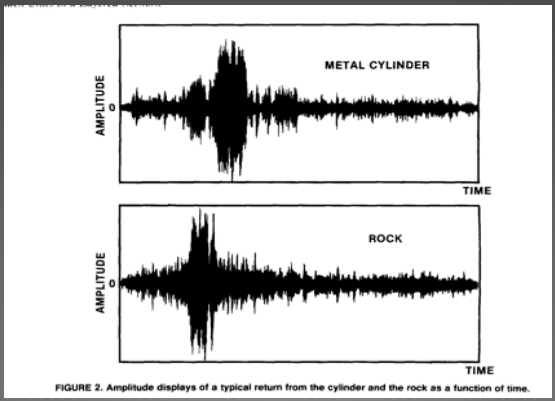
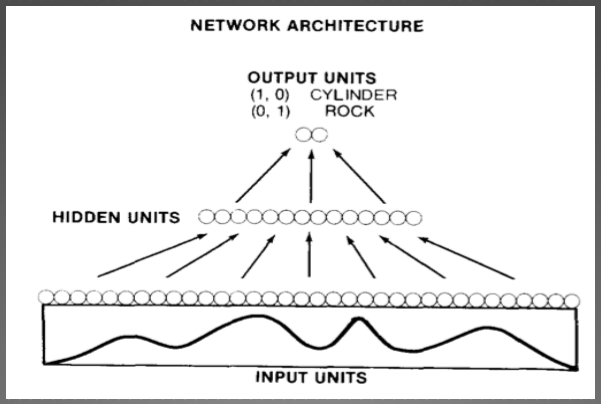


FIGURE 2. Amplitude displays of a typical return from the cylinder and the rock as a function of time.

GORMAN AND SEJNOWSKI'S NETWORK



SOME FEATURES OF NETWORKS

- Content addressability.
- Collapse of hardware/software distinction.
- Graceful degradation.

FEATURES OF CONNECTIONIST REPRESENTATION

- Distributed
  - Each of the representations generated is composed from several computationally independent parts of the system.
- Overlapping
  - The different representations generated are composed from overlapping parts
- Holographic
  - Any part of any given representation carries information about every part of the content represented by it