Attractor Neural Networks

1 The Formal McCulloch Pitts Neuron

The McCulloch-Pitts Neuron was one of the first attempts to capture the essential properties of a real "cortical neuron". This simple ersatz incorporated the idea of a threshold and the notion of multiple inputs (postsynaptic potentials) and a single output (action potential). These features in principle allowed networks of interconnected neurons of any size. Formally the McCulloch-Pitts neuron can be described mathematically by

$$h_i(t) = \sum_{j=1}^{N} J_{ij}\sigma_j(t)$$
$$\sigma(t+1) = \psi[h_i(t) > T_i]$$

where

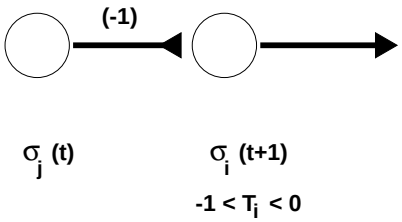
- $h_i(t)$ is the total postsynaptic potential of the ith neuron at time t.
- $\sigma_j(t)$ is the state (0 (off) or 1 (on))of the jth neuron at time t.
- $\psi = 1$ iff $h_i > T_i$ and 0 otherwise.
- T_i can be viewed as a threshold.

2 Instantiation of the Fundamental Boolean Operations using McCulloch-Pitts Neurons

Below we give explicit formulations for the fundamental Boolean operations, NOT, AND and OR. These operations coupled with the identity operation allows the formulation of **any** logical proposition.

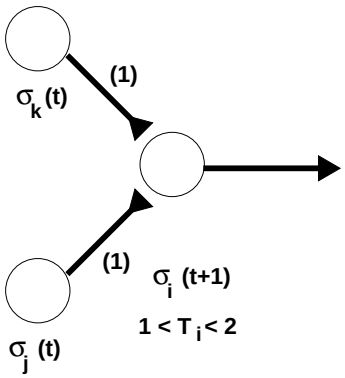
2.1 NEGATION (NOT)

$$\sigma_i(t+1) = \psi[-\sigma_j(t) > \{-1 < T_i < 0\}] \tag{1}$$



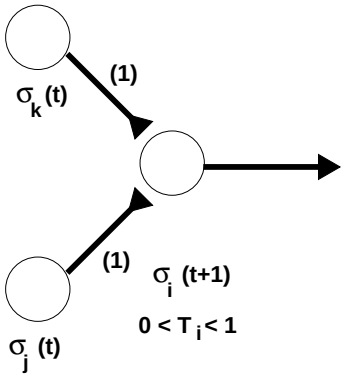
2.2 AND

$$\sigma_i(t+1) = \psi[\sigma_j(t) + \sigma_k(t) > \{1 < T_i < 2\}] \tag{2}$$



2.3 OR

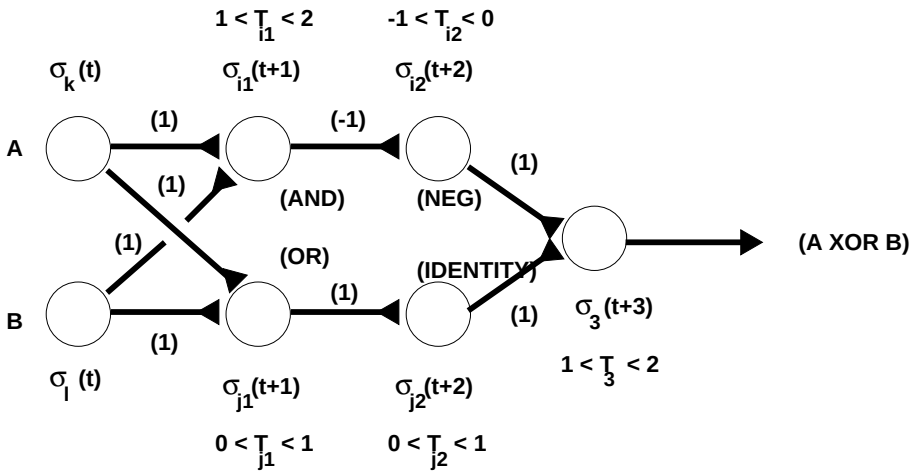
$$\sigma_i(t+1) = \psi[\sigma_j(t) + \sigma_k(t) > \{0 < T_i < 1\}] \tag{3}$$



2.4 XOR $(A \vee B) \wedge \sim (A \wedge B)$

Let $\sigma_k(t)$ and $\sigma_l(t)$ correspond to the local propositions A and B and σ_3 correspond to the result of the logical operation A XOR B. Thus the following equations describe this composite boolean operation.

$$
\begin{aligned}
\sigma_{i1}(t+1) &= \psi[\sigma_k(t) + \sigma_l(t) > \{1 < T_i < 2\}] \ \text{(AND)} \\
\sigma_{j1}(t+1) &= \psi[\sigma_k(t) + \sigma_l(t) > \{0 < T_i < 1\}] \ \text{(OR)} \\
\sigma_{i2}(t+2) &= \psi[-\sigma_{i1}(t+1) > \{-1 < T_i < 0\}] \ \text{(NEG)} \\
\sigma_{j2}(t+2) &= \psi[\sigma_{j1}(t+1) > \{0 < T_i < 1\}] \ \text{(IDENTITY)} \\
\sigma_3(t+3) &= \psi[\sigma_{i2}(t+2) + \sigma_{j2}(t+2) > \{1 < T_i < 2\}] \ \text{(AND)}
\end{aligned}
$$



Thus using very simple two-state processing elements we can instantiate and calculate any logical proposition. Modifications of this schema lead to the Rosenblatt's Perceptron and Multi-Preceptron.

3 The Perceptron and the Multi-Perceptron

The Perceptron is a formal neuron in which there are greater than two inputs. Thus if there are n inputs there will be 2^n possible input states if the inputs are binary (i.e 0 or 1). As there are only two output states all input states will be classified into two classes 0 or 1.

Possible variations in the Perceptron give rise to the Multi-Perceptron. Possible variations are

- more complex predicates computed by "sensory" elements. This gives rise to a multilayer perceptron.
- more complex predicates computed by "sensory" elements of perceptron i.e a multi-layer perceptron.
- computation of non-linear predicates.

- combined variation in which say N units at the input are multiply connected to many second order perceptrons i.e the 2^N possible combinations of of input predicates is projected into the space of 2^Q combinations of Q output spikes. The $N \times Q$ weights (J_{ij}) will determine the properties of this projection.

4 Network States

- each neuron is characterized by a discrete two-valued variable e.g (0,1) or (-1,1).
- the state of a collection of synaptically connected neurons can be represented by a list of simultaneous neural states.
- thus the network state can be represented by an N-bit word.
- the possible number of network states is 2^N.

Not all network states can have equal significance. Only special **dynamical** sequences will be selected as candidates for the description of cognitive events, such as retrieval or recognition. i.e specific cognitive events are identified as particular points in state space.

5 The Multi-Perceptron as a building block for the Attractor Neural Network

Let the output axon of each element in the multi-perceptron be closed upon itself i.e the output of each elementary neuron is the input to another neuron within the network (including the possibility of itself). i.e the output axon σ_i is now identified with the input predicate. Thus $Q = N$ (compare with combined variation of the multi-perceptron) and we now have the **dynamical equations**

$$\begin{aligned} h_i(t+1) &= \sum_j J_{ij}\sigma_j(t) \\ \sigma_i(t+1) &= \psi[h_i(t+1) - T_i] \end{aligned} \tag{4}$$

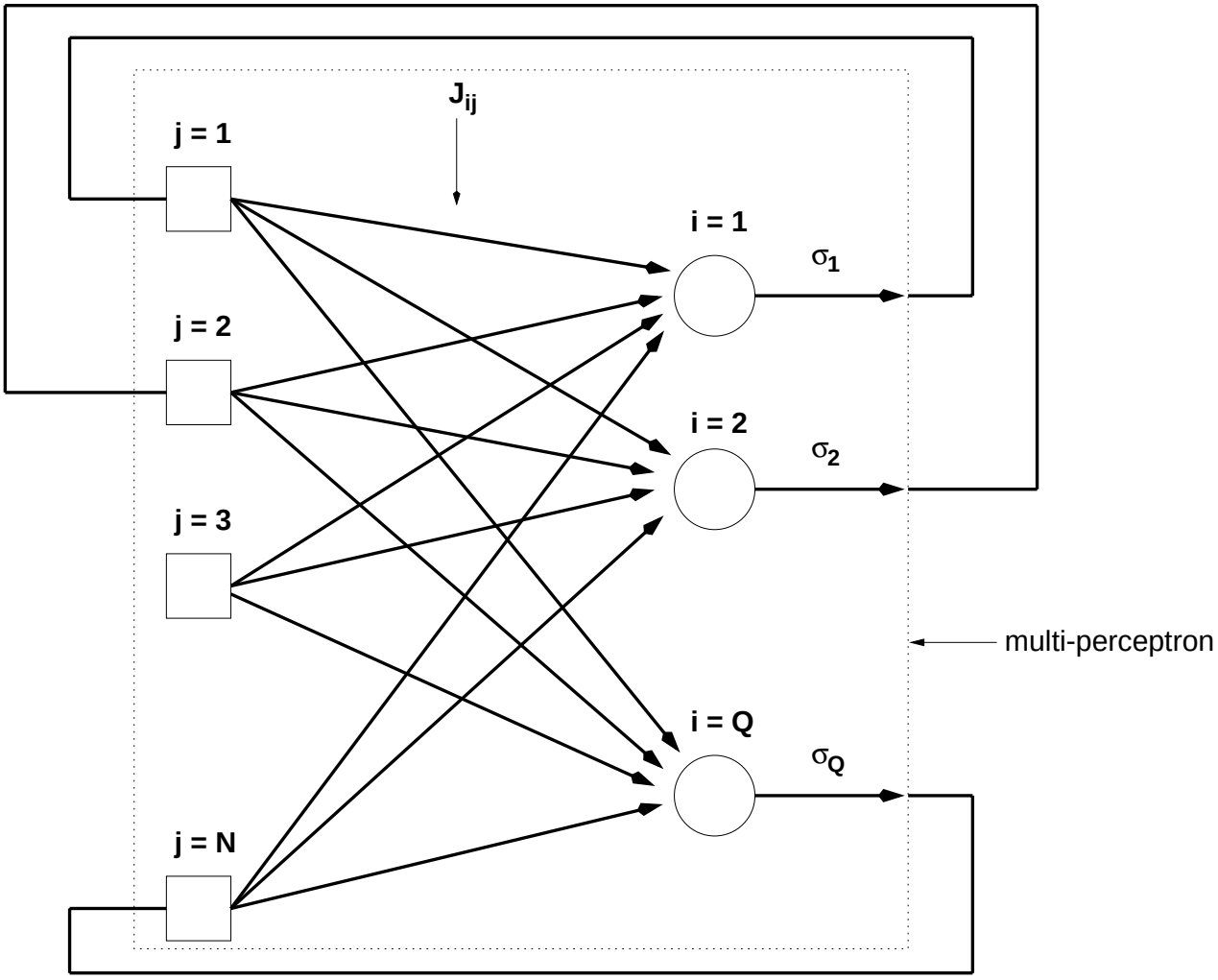


Figure 1: A multi-perceptron closed on to itself to form an Attractor Neural Network (ANN) (adapted from Amit(1989)

where

$$\psi[x] = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

These equations imply the following

- individual neurons have **no** memory as T_i are not dependent on time and the h_i depend only on activity **one** cycle time earlier.
- neurons are homogeneous with respect to type i.e each neuron can have excitatory and inhibitory outputs.

6 The Attractor

If from any initial configuration of network states the network evolves in time until

1. a given configuration of network states repeats itself (point attractor)
2. a given sequence of network states repeats itself (limit cycle attractor)
3. the network states evolve non-periodically

then an *attractor* (and generally there will be more than one) for the network is said to exist. Initial network states that are "sufficiently close" together will **evolve in time** to the same attractor.

7 Recognizing Attractors

How does what is external to the network recognize that an ANN (attractor neural network) has reached a special temporal sequence of network states ? Amit (1989) conceives of the two following extremes.

- *Type I read-out output neurons*: Neurons which transmit the output of one ANN to another via a one-to-one mapping from each neuron in one ANN to each neuron in the other ANN.

- *Type II read-out output neurons*: Neuron(s) identifying a *specific* network state by connecting efferently to a large sample of neurons in the ANN i.e the output neuron can be viewed as a singular perceptron.

8 Temporal averaging by output neurons

Temporal averaging is required by output neurons to ensure that what is read out will be those temporal sequences of network states in which the sampled ANN neurons are repeating their activity in most cycles within the averaging period.

9 Freedom from the Homunculus

Despite large differences in sophistication and biological plausibility most "neural network models" are either

- feed-forward networks
- ANNs

Feed-forward networks are characterised by the fact that for each input an output of the **same status** is produced. The only way in these architectures to avoid the notion of an homunculus is to have some *a priori* knowledge of the statistical distribution of of various inputs.

The speculation that resident cells (type II output neurons) can recognize special dynamical sequences and lead to biological activity (such as a motor response) is **freedom from the homunculus**. i.e the fact that the network enters a repeating pattern of neural activity implies that a cognitively significant event has occurred. This leads onto a view of brain function known as Connectionism whose fundamental proposition is

The Fundamental Proposition of Connectionism: *All significant cognitive events take place at the level of the network*

10 Getting Closer to Biology: Analog Neurons, two state neurons and spikes

Attractor neural networks are composed of processing elements (model neurons) that are either discrete time two state devices or elements whose output is continuous in time and firing rate. While it has been shown that the behaviour of an ANN is largely independent of it's representation it is important to consider both the analog neuron and the stochastic neuron as they are indicative of attempts at achieving biological veracity.

11 Networks of Analog Neurons

The dynamical variable that is of interest are the soma membrane potentials U_i ($i = 1, \ldots, N$) which are the "accumulated" post-synaptic potentials at the soma of each of the N neurons comprising the network. These membrane potentials will vary due to

- currents induced by pre-synaptic neuron activity.
- leakage via the finite resistivity of the neuron membrane.
- input currents from outside the network

This "analog" neuron, often called the **leaky-integrator**, has the following **continuous** dynamical equation for the evolution of the membrane potentials U_i,

$$C_i \frac{dU_i}{dt} = \sum_{j, j \neq i} J_{ij} g(U_j) \; - \; \frac{U_i}{R_i} \; + \; I_i \tag{5}$$

where the J_{ij} are the synaptic weights and g is the function describing the transformation of pre-synaptic activity to neuronal firing rate. The function g varies on the interval $[0, 1]$ and is assumed sigmoidal for the following reasons

- firing rates will be bounded above and below.
- mean firing rate is a **continuous** function of the soma membrane potential U_i.

A convenient choice for the gain function g is

$$g(U) = \frac{1}{2}[1 + \tanh(GU)] \tag{6}$$

where the parameter describes the slope of the sigmoid at the point of inflection[1] The stochastic nature of networks of real neurons can be recovered if we view the probability that a neuron fires in some small time interval dt as $g(U_i)\, dt$.

12 An alternative binary representation of single neuron activity

The McCulloch-Pitts neuron was a two state binary unit its output being either 0 (not firing) or 1 (firing). An alternative representation which provides an analogy with a well known class of physical systems called *Ising* or spin glass systems, allows the concept of energy and thus dynamical evolution to be incorporated. Thus we let the binary representation $(0, 1)$ and $(-1, 1)$ be equivalent. These equivalent representations are related by

$$S_i = 2\sigma_i - 1$$

where $\sigma_i \in \{0, 1\},\ S_i \in \{-1, 1\}$. Thus our dynamical equation 4 now reads

$$U_i = \frac{1}{2} \sum_{j, j \neq i}^{N} J_{ij}(S_j + 1) \tag{7}$$

The response of the neuron in the **absence of noise** is now, from equation 4

[1] actually $\frac{dg}{dU}\Big|_{U=0} = G/2$.

$$S_i = \text{sign}(U_i - T_i) \tag{8}$$

which can be rewritten as

$$\begin{aligned} S_i &= \text{sign}(h_i + h_i^e) \\ \text{where} & \\ h_i &= \frac{1}{2}\sum_{j,j\neq i}^{N} J_{ij}S_j \\ h_i^e &= \frac{1}{2}\sum_{j,j\neq i}^{N} J_{ij} - T_i \end{aligned} \tag{9}$$

It is often convenient to assume that $h_i^e = 0$ as the response of the model neuron will be most sensitive to changes in its inputs.

13 Noisy Neurons

Real neurons, as alluded to previously, are not deterministic, their behaviour has a large element of stochasticity. Sources of such noise are

- the number of vesicles discharged per action potential is a Poisson distribution.
- variations in the size of quanta (vesicles) obey approximately a Gaussian density.
- there exists a small random rate of spontaneous discharge of neurotransmitter even when the neuron is at "rest".

If these effects are taken into account it can be shown that the probability that event S_i occurs is given by

$$Pr\{S_i\} = \frac{1}{2}\left[1 + \text{erf}\left(\frac{h_i S_i}{\delta\sqrt{2}}\right)\right] \tag{10}$$

where δ is the standard deviation of the variation in the total postsynaptic response. This equation can be approximated to within 1% by

$$Pr\{S_i\} = \frac{1}{2}[1 + \tanh(\beta h_i S_i)] \tag{11}$$

where

$$\beta^{-1} = 2\sqrt{2}\sigma$$

The similarity of this expression to the response function for the analog neuron should be noted.

14 Energy and the Lyapunov Function

The use of the representation $(-1, 1)$ for network states allows us to define an analog of energy for network states, where a give network state is associated with a definite energy. We will show that the dynamical evolution of the network is such that at each time step the **energy associated with the network states either stays the same or is reduced.**

For an attractor neural network , the energy function (Lyapunov energy function), H is defined as

$$H = -\frac{1}{2}\sum_{ij} J_{ij} S_i S_j \tag{12}$$

The central property of a Lyapunov energy function is *that it always decreases or remains constant as the system evolves according to its dynamical rule e.g equation 4.* For **symmetric connections** (i.e $J_{ij} = J_{ji}$) H can be rewritten as

$$H = K - \sum_{(ij)} J_{ij} S_i S_j \tag{13}$$

where (ij) means all **distinct** pairs and where (ii) terms are excluded from the sum and give rise to the constant K. Now let S_i' be the new value, at time Δt later, for some particular unit i, i.e

$$S_i' = \text{sign}(\sum_j J_{ij} S_j)$$

Obviously if $S_i' = S_i$ then the energy H is unchanged. Now the only other possibility is that $S_i' = -S_i$. Picking out the terms that involve the particular unit S_i we have

$$\begin{aligned} H_i &= K - S_i \sum_{j \neq i} J_{ij} S_j \\ H_i' &= K - S_i' \sum_{j \neq i} J_{ij} S_j \end{aligned} \tag{14}$$

therefore

$$\Delta H_i \;=\; H_i' - H_i \;=\; -S_i' \sum_{j \neq i} J_{ij} S_j \;+\; S_i \sum_{j \neq i} J_{ij} S_j \tag{15}$$

remembering that $S_i' = -S_i$ we get

$$\begin{aligned} \Delta H_i &= 2S_i \sum_{j \neq i} J_{ij} S_j \\ &= 2S_i \sum_j J_{ij} S_j - J_{ii} \end{aligned} \tag{16}$$

Using the Hebb rule it can be shown that the second term is negative. The first term is also negative because $S_i = -\sum_j J_{ij} S_j$. Thus as $\Delta H = \sum_i \Delta H_i$ the energy H decreases at every time step as claimed. Thus the dynamical trajectory is always towards decreasing network energy. This concept can be approximately represented by considering the network states as an ordered one-dimensional array, as in the figure below. Associated with each network state will be an energy, with the lowest energy network states representing stored memories. Thus the network from any given initial state will evolve until the network energy is a minima.

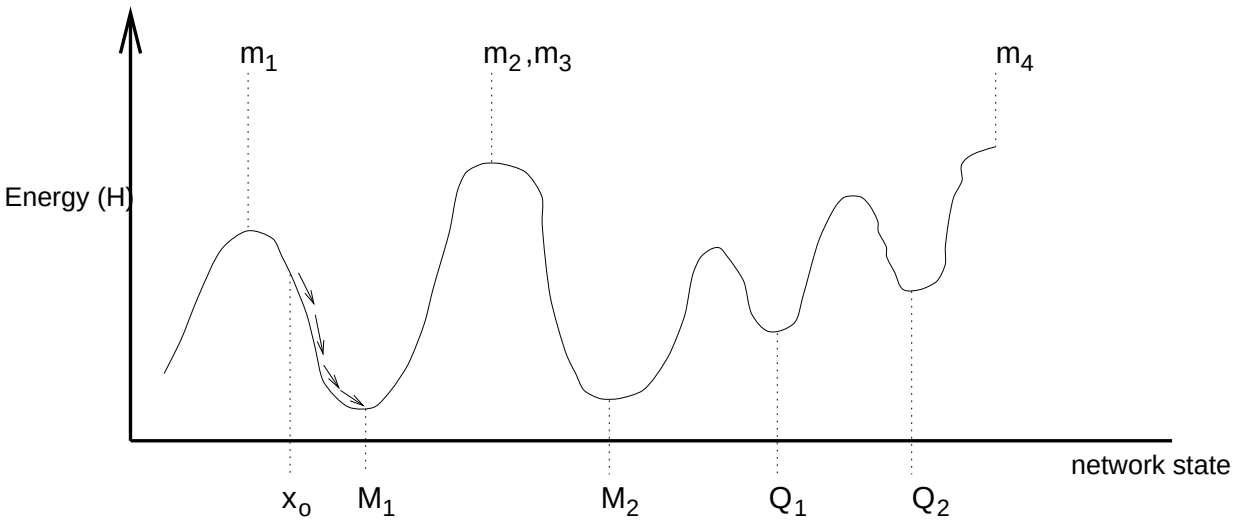


Figure 2: A one-dimensional landscape metaphor for associative, content addressable memory. M_1, M_2 are memories, Q_1, Q_2 are spurious states, $m_1 \ldots m_4$ are maxima delimiting basins of attraction. Shown is the initial network state x_o and its evolution with time (arrows).

15 Summary

The special emphasis one places on fixed points is because they are used to represent [or model] our elementary cognitive events.

Given a certain initial network state [e.g imposed by a certain set of external stimuli] the network follows a **dynamical** trajectory as determined by its synapses [synaptic weights]

The realization of recall is a pattern of activity which corresponds to a fixed network state. Thus in line with much topographic localisationism patterns of activity are identified with "representations".

Concrete examples of psychologically relevant and biologically veracious ANN models may help articulate the abstract debate between representation and computation for the following reasons

- an N bit network state may contain a mixture of contextual data.
- association to learnt stimuli is a key concept in the attractor metaphor.

16 References

Amit D. *Modelling Brain Function: The World of Attractor Neural Nets.* Cambridge University Press, New York, 1989.

Hertz J, Krogh A and Palmer RG. *Introduction to the Theory of Neural Computation.* Addison-Wesley, Redwood City, CA. 1991.