The Hopfield model

Sebastian Seung

9.641 Lecture 15: November 7, 2002

1   The Hebbian paradigm

In his 1949 book *The Organization of Behavior*, Donald Hebb predicted a form of synaptic plasticity with the following property

> When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased.

In 1973, the phenomenon of long-term potentiation (LTP) was discovered. The study of LTP is now a small industry within the field of neuroscience. Among the various forms of LTP, those that depend on the NMDA subtype of glutamate receptor are regarded as "Hebbian." These forms depend on temporal contiguity of presynaptic and postsynaptic activity. The requirement of temporal contiguity is expressed in the following ditty

> Neurons that fire together, wire together.
> Neurons that fire out of sync, fail to link.

Hebbian synaptic plasticity is remarkable for having been predicted on theoretical grounds well before it was discovered experimentally. This is not a common occurrence in biology.

How did Hebb make his remarkable prediction? He was motivated by an old tradition in Western thought known as associationism. This is the idea that the brain is nothing more than an engine for storing and retrieving associations. In the late 19th century, it was found that the brain consisted of neurons connected by an intricate web of synapses. This transformed the older associationist tradition into connectionism, the doctrine that associations are stored as synaptic connections. This is an example of the idea that structure determines function in biology. Hume and other empiricist philosophers had already expressed the idea that associations were learned from temporal contiguity. It only stood to reason that connections should also be learned from temporal contiguity.

While these ideas are very suggestive, they are admittedly rather vague and metaphorical. The challenge of connectionism is to make these ideas precise.

A number of theorists have formulated neural network models with the goal of explaining how Hebbian synaptic plasticity could be used to store memories.

2 Binary model neurons

For the most part, we have studied neural network models in which the activity of each neuron is described by a single analog variable. A more drastic simplification is to use binary neurons, which are either active ($s_i = 1$) or inactive ($s_i = -1$). In such a dynamics, each neuron updates itself according to the rule

$$s_i' = \text{sgn} \left(\sum_j W_{ij} s_j \right) , \tag{1}$$

where s_i' is the state of neuron i after the update. Note that sometimes $s_i' = s_i$, in which case the update results in no change. If the argument of the sgn function happens to be zero, there is an ambiguity to the definition, which could be resolved by a coin flip or arbitrarily defining $\text{sgn}(0) = 1$.

Equation (1) can be used to define several types of network dynamics, depending on the exact manner in which it is applied. In a sequential dynamics, the neurons are updated one at a time, typically in a random order. In a parallel dynamics, Eq. (1) is applied to all neurons simultaneously. It is often easier to prove theorems about the sequential case, but the parallel case is sometimes easier to implement (for example in MATLAB it is more natural).

In a simulation on a digital computer, it is natural to make the updates at discrete times. However, the update times could in principle be continuous. For example, they could occur at random for each neuron according to a Poisson process with some mean rate.

As we shall see, all these version of network dynamics behave qualitatively the same in general, but there can be subtle differences.

3 Hopfield model

Suppose that synapses change according to the learning rule

$$\Delta W_{ij} = \eta s_i s_j$$

where $\eta > 0$ is a learning rate. This could be called Hebbian, although it has some weird features. The synapse is potentiated if both neurons are active, or both are inactive. The synapse is depressed if one neuron is active, and the other is inactive.

Suppose further that the network is exposed to P binary patterns $\xi_i^1, \ldots, \xi_i^P$ during a training phase. More specifically, the state s_i is set to each pattern in turn, and the synaptic weights are changed by the Hebbian update. If $W_{ij} = 0$ at the beginning of the training phase, Hebbian learning will result in

$$W_{ij} = \frac{1}{N} \sum_\mu \xi_i^\mu \xi_j^\mu \tag{2}$$

The prefactor $1/N$ corresponds to a particular choice of learning rate $\eta = 1/N$. It is a convenient choice for the calculations we are about to perform, but is otherwise arbi-

trary, as W_{ij} can be multiplied by any positive prefactor without affecting the dynamics (1).

This synaptic weight matrix is the famous Hopfield model, along with the dynamics (1), and the assumption that the patterns are chosen at random. This last assumption makes sense if we assume that there is a data compression stage that encodes sensory data efficiently before it reaches the Hopfield network.

In his original paper, Hopfield set the diagonal terms of the weight matrix to be zero ($W_{ii} = 0$). If this is not done, the dynamics takes the form

$$s_i' = \text{sgn} \left(\frac{P}{N} s_i + \sum_{j, j \neq i} W_{ij} s_j \right) ,$$

The network still basically works, but there are some subtle differences, which we will discuss later.

As we shall see in the following, the Hopfield model functions as an associative memory, because the patterns are stored as dynamical attractors. If the network is initialized with a corrupted or incomplete version of a pattern, convergence to an attractor can recall the correct pattern.

4 Single pattern

Let's start with a simple case, a single pattern

$$W_{ij} = \frac{1}{N} \xi_i \xi_j \tag{3}$$

The dynamics takes the form

$$s_i' = \text{sgn} \left(\xi_i \frac{1}{N} \sum_j \xi_j s_j \right)$$

In terms of the overlap

$$m = \frac{1}{N} \sum_j \xi_j s_j$$

between ξ and s, we can write

$$s_i' = \text{sgn} \left(\xi_i m \right)$$

Since $m = 1$ when $s = \xi$, it is a steady state of the dynamics. Similarly, $m = -1$ when $s = -\xi$, so it is also a steady state of the dynamics. If $m > 0$, updates can only cause m to increase. Likewise if $m < 0$, updates can only cause m to decrease. Therefore these steady states are attractors of the dynamics.

5 Many patterns

The case of many patterns is more interesting. The weight matrix (2) is just the superposition of single pattern outer products (3). The danger here is that the patterns might interfere with each other. If the patterns were exactly orthogonal to each other, there would be no interference. If they are chosen randomly, there is some possibility of crosstalk, which is quantified below.

To check whether the patterns are steady states, we calculate

$$\sum_{j,j\neq i} W_{ij}\xi_j^\nu = \frac{1}{N}\sum_{j,j\neq i}\sum_\mu \xi_i^\mu\xi_j^\mu\xi_j^\nu \tag{4}$$

$$= \xi_i^\nu + \frac{1}{N}\sum_{j,j\neq i}\sum_{\mu,\mu\neq\nu}\xi_i^\mu\xi_j^\mu\xi_j^\nu \tag{5}$$

$$= \xi_i^\nu\left(1 + \frac{1}{N}\sum_{j,j\neq i}\sum_{\mu,\mu\neq\nu}\xi_i^\mu\xi_i^\nu\xi_j^\mu\xi_j^\nu\right) \tag{6}$$

The last term is called the crosstalk or interference term. If it is greater than -1 for all i, then the νth pattern is a steady state

$$\xi_i^\nu = \text{sgn}\left(\sum_j W_{ij}\xi_j^\nu\right)$$

If we try to store too many patterns, then the interference between them becomes large, and storage is not possible.

To study the stability of a pattern, imagine that the network is initialized at the pattern, and try to estimate how many bit flips take place. We can approximate the crosstalk term as the sum of Np independent coin flips. There is an error when the sum is more negative than $-N$. The derivation in the book uses the Gaussian approximation to the binomial distribution. We will do something coarser, which is the Hoeffding bound on the deviation between frequency and probability. According to the one-sided Hoeffding bound for m coin tosses

$$Prob[\hat{p} < p - \Delta] < \exp(-2\Delta^2 m)$$

Here we are interested in deviations of the frequency of heads from $\frac{1}{2}$ by more than $\frac{1}{2p}$. We find that

$$P_{error} < \exp(-2Np(1/2p)^2) = \exp\left(-\frac{N}{2p}\right)$$

There are several definitions of capacity.

- (fraction of bits are corrupted, $P_{error} < 0.01$) Suppose that we would like less than one percent of the bits corrupted. Then we need $\exp(-N/(2p)) < 0.01$, or $p < -N/(2\log(0.01)) = 0.11N$. The problem with this calculation is that the flipping of the first bits could cause an avalanche. In reality, the capacity is about $p = 0.14N$. Calculating this number requires some heavy mathematics, like the replica trick.

- (fraction of patterns are corrupted, $P_{error} < 0.01/N$) We could impose a more stringent requirement, which is than less than one percent of the patterns have a bit corrupted.

$$p = \frac{N}{2 \log N}$$

- (fraction of samples are corrupted, $P_{error} < 0.01/(pN)$. The most stringent requirement is that no pattern in the sample is corrupted, with confidence greater than 99%. Taking $\log p \approx \log N$, we obtain

$$p = \frac{N}{4 \log N}$$

Is this good? According to the Cover argument, the maximum should be $p = 2N$.

We can store patterns as attractors of the network dynamics (with some corruption). However, there can be other attractors, or spurious states. These are of three types.

- There are reversed states $-\xi$ due to the $\pm$ symmetry of the network dynamics.

- There are mixture states, which are a superposition of an odd number of patterns. For example,
$$\xi_i^{mix} = \operatorname{sgn}(\pm\xi_i^{\mu_1} \pm \xi_i^{\mu_2} \pm \xi_i^{\mu_3})$$
An even number won't work, because the sum works out to zero on some sites.

- There are spin glass states for large p, which are not correlated with any finite number of the patterns ξ.

6 Energy function

If the interactions W_{ij} are symmetric, then

$$E = -\frac{1}{2} \sum_{ij} W_{ij} s_i s_j$$

is nonincreasing under the dynamics (1), assuming asynchronous update.

To prove this, note that the $i = j$ terms in the sum are unchanging, since $s_i^2 = 1$. Therefore the change in E due to the update (1) is given by

$$\Delta E \quad = \quad -\frac{1}{2}(s_i' - s_i) \sum_{j, j \neq i} W_{ij} s_j \tag{7}$$

$$= \quad -\frac{1}{2}(s_i' - s_i) \sum_j W_{ij} s_j + \frac{1}{2}(s_i' - s_i) W_{ii} s_i \tag{8}$$

If $s_i' = s_i$, then $\Delta E = 0$ and we are done. In the other case, $s_i' = -s_i$, and we can write

$$\Delta E = -s_i' \sum_j W_{ij} s_j - W_{ii} s_i^2 \leq 0$$

Therefore the dynamics can be understood as descent on an energy landscape.

Statistical physicists like binary neurons because they are analogous to magnetic spins. The synaptic interaction W_{ij} can be compared to a magnetic interaction. If $W_{ij} > 0$, the spins want to line up in the same direction, as in a ferromagnet. If $W_{ij} < 0$, the interaction is called antiferromagnetic.

7   Energy function

Let's consider the case $W_{ij} = 1$ for all i and j. This corresponds to a pure ferromagnet, in which all interactions try to make the spins line up in the same direction. In this case,

$$E = -\frac{1}{2}\left(\sum_i s_i\right)^2$$

Obviously there are two minima, the fully magnetized states $s_i = 1$ for all i and $s_i = -1$ for all i. Any initial condition with more up spins than down spins will converge to the all up state, and a similar statement can be made about the down state.

Suppose that we would like to store a single pattern ξ as an attractor of the dynamics. This is basically the same as the previous case. If we choose

$$W_{ij} = \xi_i \xi_j$$

then the energy function is

$$
\begin{aligned}
E &= -\frac{1}{2}\sum_{ij}\xi_i\xi_j s_i s_j & (9)\\
&= -\frac{1}{2}\left(\sum_i \xi_i s_i\right) & (10)
\end{aligned}
$$

The attractors of this dynamics are $s_i = \xi_i$ and $s_i = -\xi_i$, as required. In statistical mechanics, this is known as the Mattis model.

Note that the energy function for this network is

$$E = -\frac{1}{2N}\sum_\mu\left(\sum_i s_i\xi_i^\mu\right)^2$$

It's plausible that the patterns should be local minima, but they might interfere with each other.

8   Content-addressable memory

The input is encoded in the initial condition of the network dynamics. Convergence to an attractor corresponds to recall of the closest stored memory.