Appendix E

Neural Networks

By itself a single neuron is not intelligent. But a vast network of neurons can think, feel, remember, perceive, and generate the many remarkable phenomena that are collectively known as "the mind." How does intelligence emerge from the interactions between neurons? This is the central question motivating the study of neural networks. In this appendix we provide a brief historical review of the field, introduce some key concepts, and discuss two influential models of neural networks, the perceptron and the cell assembly. Additional models of neural networks are discussed in Appendix F.

Starting from the 1940s researchers have proposed and studied many brain models in which sophisticated computations are performed by networks of simple neuron-like elements. Most models are based on two shared principles. First, our immediate experience is rooted in ongoing patterns of action potentials in brain cells. Second, our ability to learn from and remember past experiences is based at least partially on long-lasting modifications of synaptic connections. Although these principles are widely accepted by neuroscientists, they immediately suggest many difficult questions.

For example, to our conscious minds, perceiving an object or moving a limb is experienced as a single, unitary event. But in the brain, either act is the result of a collection of a stupendous number of neural events—the discharge of action potentials or the release of neurotransmitter vesicles—indiscernible by the conscious mind. How are these events united into a coherent perception or movement?

Storage of our immediate experience in long-term memory is presumed to occur with changes in synaptic connections. But how exactly is a memory divided up and distributed across many synapses? If some synapses are used to store more than one memory, how then is interference between memories avoided?

When past experiences are recalled from memory, how might synaptic connections evoke a pattern of firing that is similar to a pattern that occurred in the past? Finally, when we reason, daydream, or otherwise float in the stream of consciousness, our mental state is not directly tied to any immediate sensory stimulus or motor output. How do networks of neurons dynamically generate the patterns of activity related to such mental states?

These are profound questions. Many hypothetical answers have been proposed in the form of neural network models, a body of work that spans many decades and which we survey here. Although they are far from being tested conclusively, these hypotheses have influenced the research of a number of experimental neuroscientists and are being developed further today by theoretical neuroscientists.

Early Neural Network Modeling

Perhaps the first attempt to explain behavior in terms of synaptic connectivity was Sherrington's reflex arc. A reflex behavior is defined as a rapid, involuntary, and stereotyped response to a specific stimulus (Chapter 35). For any reflex behavior one can generally identify a reflex arc, a chain of synapses starting from a sensory neuron and ending with a motor neuron. The sequential activation of neurons in this chain is a series of causes and effects that connect the stimulus to the response. The reflex arc can be regarded as an ancestor of neural network models.

In 1938 Rafael Lorente de Nó, a student of Santiago Ramón y Cajal, argued that synaptic loops ("internuncial chains") were the basic circuits of the central nervous system. A synaptic loop is a chain of synapses that starts and ends at the same neuron. It is a closed chain, in contrast to the open chain of a reflex arc. Lorente de Nó suggested that the purpose of these loops was to sustain "reverberating" activity patterns. Sherrington's student, Graham Brown, in his studies of spinal cord rhythmicity, proposed a related view of the brain, involving intrinsic generation of neural activity rather than stimulus-response relationships. These scientists emphasized that the brain has an intrinsic dynamic richer than that of reflex arcs, which are inactive until stimulated by the outside world.

In an influential book published in 1949, Donald Hebb proposed the idea of a "cell assembly" as a functional unit of the nervous system and discussed the form of synaptic plasticity that would become known as Hebb's rule. (The rule had previously been formulated by several other thinkers, of whom the earliest was perhaps the philosopher Alexander Bain in 1873.) Hebb proposed that repeated synaptic communication between neurons could strengthen connections between the neurons, creating synaptic loops that were capable of supporting the reverberating activity patterns of Lorente de Nó.

These ideas of Sherrington, Graham Brown, Lorente de Nó, and Hebb were later formalized in mathematical models of neural networks. Two famous classes of models are perceptrons and associative memory networks. Perceptrons have been popular as models of the visual system because they illustrate how recognition of an object can be decomposed into many feature detection events. A perceptron can be organized hierarchically, so that the decomposition process begins with simple features at the bottom of the hierarchy and proceeds to complex features at the top, as is thought to occur in the visual system (Chapter 29).

Associative memory networks have been used to model how the brain stores and recalls long-term memories. Central to these models is Hebb's concept of the cell assembly, a group of excitatory neurons mutually coupled by strong synapses. Memory storage occurs with the creation of a cell assembly by Hebbian synaptic plasticity (Chapter 66), and memory recall occurs with the neurons in a cell assembly are activated by a stimulus.

The perceptron and the cell assembly have very different synaptic connectivities. As in Sherrington's reflex arc, the polysynaptic pathways in a perceptron all travel in the same overall direction, from the input layer to the output layer. The perceptron generalizes the reflex arc, because it allows many synapses to diverge from a neuron and converge onto a neuron.

The perceptron is a special case of a *feedforward network*, defined as one with no synaptic loops. As noted above, a synaptic loop is defined as a polysynaptic pathway that starts and ends at the same neuron (always crossing synapses in the direction from presynaptic to postsynaptic neuron). Networks with loops are called *recurrent* or *feedback networks*, to distinguish them from feedforward networks. A cell assembly typically contains loops, and is therefore recurrent. In this respect it resembles biological neural networks, which have extensive synaptic loops, as Lorente de Nó and many other neuroanatomists have documented for more than a century.

Lorente de Nó and Hebb postulated that neural activity can persist longer in the brain by circulating through synaptic loops. Thus a cell assembly can maintain a persistent activity pattern resembling patterns observed by neurophysiologists in studies of short-term and working memory. In other words, loops could be important for the generation of persistent mental

states in the brain, which are required for behaviors in which stimulus and response are separated by a long time delay. In contrast, the direct pathways of the perceptron are suited for modeling behavioral responses that immediately follow a stimulus.

Only very simple neural networks are described in this appendix. The "neurons" in these models are much simpler than biological neurons, and the "synapses" do not do justice to the intricacies of biological synapses. When modeling a complex system, simplifying its elements helps one to focus on the properties that emerge from the interactions between them. This strategy has historically been used by neural networks researchers focusing on emergent properties of brain function. More realistic models of how neurons integrate their synaptic inputs can be found in Appendix F.

Neurons Are Computational Devices

Action potentials and synaptic potentials are dynamic events that involve a complex interplay between the membrane voltage of a neuron and the opening and closing of its ion channels. Computational neuroscientists often ignore these complexities in their thinking and instead rely on the following simplification: *A neuron fires an action potential when a sufficiently large number of excitatory synapses onto it are activated simultaneously.*

This statement is based on the fact that a single excitatory postsynaptic potential is typically much smaller in amplitude (less than 0.5 mV) than the gap of many millivolts that separate the resting potential from the threshold for an action potential. Therefore, many simultaneous excitatory postsynaptic potentials need to sum in the postsynaptic neuron to drive its voltage over the threshold for firing.

The biophysical description of neurons pioneered by Alan Hodgkin and Andrew Huxley (Chapter 6) has been the basis for mathematical models of neurons. Surprisingly, the above simplification of the conditions for neuronal firing has inspired a great deal of mathematical formalism. In 1943 Warren McCulloch and Walter Pitts proposed a model of the computation performed by a neuron and the excitatory synapses converging onto it. The McCulloch-Pitts neuron takes multiple inputs and produces a single output. All inputs and the output are binary variables, 0 or 1. The neuron is characterized by a single parameter θ, its threshold. If a subset of θ or more inputs is equal to 1, then the neuron's output is 1; otherwise the output is 0.

In the biological interpretation of the McCulloch-Pitts model each input variable represents the activation of an excitatory synapse at the neuron. The input is equal to 1 when the excitatory synapse is activated. The parameter θ is used to model the threshold of a biological neuron and is equal to the minimum number of excitatory synapses that must be simultaneously activated to produce an action potential. In this interpretation the McCulloch-Pitts model formalizes the above caricature of a biological neuron.

Two McCulloch-Pitts neurons can be connected so that the output of one neuron is the input of another. This corresponds to the biological fact that excitatory synapses converging onto a neuron are activated by the discharging of the presynaptic neurons. By making many such connections, it is possible to construct a model of a neural network.

In the McCulloch-Pitts model neurons are either active ("1") or inactive ("0"). This is admittedly a crude way of describing neural activity, because it does not distinguish between active neurons with different firing rates. But this coarse description is used not only by theorists but also by experimental neurophysiologists, who often speak of active and inactive neurons in the exploratory phases of their experiments before they make precise measurements of firing rates. Although the graded nature of firing rates can be captured by more realistic model neurons (see the equations in Box E–1), here we will limit ourselves to the McCulloch-Pitts model to minimize the use of mathematical equations.

This simplification also allows the application of ideas from Boolean logic, in which the binary values 0 and 1 correspond to "false" and "true." Boolean logic, named after the British mathematician George Boole, is a formalization of deductive reasoning that is based on manipulations of binary variables that represent truth values. Boolean logic is the mathematical foundation of digital electronic circuits. Using their model, McCulloch and Pitts argued that the activity of each neuron signifies the truth of some logical proposition. They concluded that neurons (and by extension networks of neurons) perform logical computations.

A Neuron Can Compute Conjunctions and Disjunctions

If we accept the idea that biological neurons can perform logical computations, then it is natural to ask what types of computations are possible. We will answer this question by studying the behavior of the McCulloch-Pitts model neuron. Of course, biological neurons are more complex and therefore likely to be more powerful computational devices. But by analyzing the McCulloch-Pitts neuron we can expect to establish lower bounds on the computational power of biological neurons. In other words, if a computation

Box E–1 Mathematics of Neural Networks

The McCulloch-Pitts neuron is simple enough that its behavior can be described in words. More sophisticated models require the precision of mathematics for a clear formulation.

The linear-threshold (LT) model neuron corrects a shortcoming of the McCulloch-Pitts neuron that all excitatory inputs are equally effective in bringing the neuron to its firing threshold; the number of active inputs is important, but their identities are not. For a biological neuron in which some synapses are stronger than others, such a simplification is not realistic.

To model this aspect of synaptic function, the LT neuron takes the weighted sum of its inputs, where the weights of the sum represent synaptic strengths. If the sum exceeds a threshold, the LT neuron becomes active.

To model a network of LT neurons, assume that their activities at time t are given by the N variables, $x_1(t)$, $x_2(t)...x_N(t)$ which take on the values 0 or 1, that is, a neuron is either active ("1") or silent ("0"). Then the activities at time $t + 1$ are given by

$$x_i(t+1) = H\left(\sum_{j=1}^{N} W_{ij}x_j(t) - \theta_i\right) \qquad \textbf{(E–1)}$$

where H is the Heaviside step function defined by $H(u) = 1$ for $u \geq 0$ and $H(u) = 0$ otherwise, W_{ij} is the strength or weight of the synapse between neuron i and the presynaptic neuron j, and θ_i is the threshold of neuron i. For a network of N neurons, the synaptic weights W_{ij} form an $N \times N$ matrix, and the thresholds θ_i an N-dimensional vector.

The LT model and the McCulloch-Pitts model are equivalent if the synaptic strengths of the LT model satisfy two conditions. First, the strengths of all excitatory synapses must equal one to yield the uniformity of strengths discussed above. Second, each inhibitory synapse must be so strong that activating it is enough to keep the LT neuron below threshold, no matter how many excitatory inputs are active.

This second condition is in accord with the behavior of inhibition in the original definition of a McCulloch-Pitts neuron and could be regarded as a crude model of shunting inhibition (Chapter 8). This second condition can be realized by making the strength of each inhibitory synapse onto a neuron greater than the number of excitatory synapses onto the neuron.

The LT neuron of Equation E–1 can perform many different types of computation, depending on the choice of synaptic weights and thresholds. By arguments similar to those given in the main text, any Boolean function can be realized by combining LT neurons into a network. A perceptron network can be implemented by a synaptic weight matrix in which certain elements are constrained to be zero. (Such elements are not included in the perceptron model illustrated in Figure E–1.) An associative memory network can be constructed by choosing W_{ij} to be a correlation matrix (see Box E–3).

The LT neuron is either active or inactive, but the firing rates of biological neurons are continuously graded quantities. This can be modeled by replacing the Heaviside step function H in Equation E–1 by some other function F with graded output. Neural activity is described by continuously graded variables $r_1...r_N$ rather than binary variables, and are interpreted as rates of action-potential firing. Furthermore, time can be treated continuously in the differential equation

$$\tau\frac{dr_i}{dt} + r_i = F\left(\sum_{j=1}^{N} W_{ij}r_j - \theta_i\right) \qquad \textbf{(E–2)}$$

rather than discretely as in Equation E–1. This type of model is discussed in more detail in Appendix F.

In Equation E–2 the soma of the neuron is regarded as a device that converts input current into the cell's rate of firing. This point of view is often taken by electrophysiologists, who characterize a neuron by its *f-I* curve, plotted by injecting current into a neuron and recording the resulting firing rate (Chapter 10). The dendrite of the neuron is assumed to linearly combine the currents produced by its synapses, a good approximation in some biological neurons (Chapter 10). Each synapse generates a current that is proportional to the firing rate of its presynaptic neuron.

Equation E–2 is still quite crude in its description of neural activity as an overall firing rate. More sophisticated models have differential equations governing voltages and conductances and generate individual action potentials. For example, the voltages in the numerical simulations of Figure E–5 were generated by leaky integrate-and-fire model neurons. More about this and other spiking model neurons can be found in works listed in the bibliography at the end of the appendix, as well as in Appendix F.

is possible for a McCulloch-Pitts neuron it should be possible for a biological neuron, although the converse is not necessarily true.

Suppose that the threshold parameter θ of a McCulloch-Pitts neuron is set at a high value, equal to the total number of inputs. Then the neuron is active if and only if all of its synaptic inputs are active. In other words, the output of the neuron is the *conjunction* of its input variables, which is also known as the logical AND operation. Alternatively, the threshold can be set at a low value, equal to one, such that activation of one or more synaptic inputs is enough to activate the neuron. In this case the output of the neuron is the *disjunction* of its input variables, which is also known as the logical OR operation.

Although a McCulloch-Pitts neuron can compute some logical functions, it cannot compute others. A famous example is the exclusive-or (XOR) operation. By definition the XOR operation on two inputs results in "1" if and only if exactly one of its inputs is "1." Thus if both inputs are "1," the XOR function outputs "0," while the OR function outputs "1." Proving that a single McCulloch-Pitts neuron cannot compute the XOR operation is left as an exercise to the reader. However, XOR can be computed by a network of McCulloch-Pitts neurons, as is explained below.

A Network of Neurons Can Compute Any Boolean Logical Function

What functions can be computed by a network of McCulloch-Pitts neurons? Conjunctions and disjunctions are basic building blocks of Boolean logic. The original definition of a McCulloch-Pitts neuron included both inhibitory and excitatory synapses. It turns out that synaptic inhibition can be used for the operation of negation (logical NOT).

Consider a neuron that is spontaneously active and receives a single strong inhibitory synapse. When the inhibitory synapse is inactive, the neuron is spontaneously active. But when the inhibitory synapse is active, the neuron is inactive, silenced by inhibition. In other words, the neuron responds with 1 when its input is 0 but with 0 when its input is 1. This is exactly the NOT operation.

It is well known that any function of Boolean logic can be synthesized by combining the AND, OR, and NOT operations. Because McCulloch-Pitts neurons can compute all of these operations, it follows that networks of McCulloch-Pitts neurons can compute any function of Boolean logic, including XOR.

Why is it important that these models compute Boolean functions? Boolean logic lies at the heart of modern digital computers. The computers on our desktops, and in fact all digital electronic circuits, are designed to implement Boolean logic. When a digital computer runs a software program, it simply executes sequences of logical operations. Thus networks of McCulloch-Pitts neurons and digital computers compute the same.[1]

These facts about networks of McCulloch-Pitts neurons were discovered in the 1940s and 1950s when neural network models played a role in the formal theory of automata and computation. This line of research showed that neural network models have great computational power in principle. Nevertheless, a difficult question remains: How are computations actually performed by brains? This question cannot be answered by formal arguments alone. It is now being addressed both by theoretical and experimental neuroscientists who try to understand how the brain works, and by computer scientists and engineers who create artificial systems that emulate capabilities of the brain.

The notion that a neuron is a device for computing conjunctions and disjunctions is prominent in the ensuing discussion of neural network models of the visual system.

Perceptrons Model Sequential and Parallel Computation in the Visual System

The term *perceptron* was coined in the 1950s by Frank Rosenblatt to describe his neural network models of visual perception. In a perceptron neurons are organized in layers (Figure E–1).[2] The first layer is the input to the network and the last layer the output. Each layer sends synapses only to the next layer, so that information flows in the "forward" direction from the input to the output. Because of this, the term *feedforward* is sometimes used to describe this type of network architecture, as opposed to *feedback* or *recurrent* architectures with synaptic loops. Although perceptrons can be

[1] A formal model of a digital computer called a *Turing machine* is more powerful than a network of McCulloch-Pitts neurons because it has a memory with an infinite capacity. But any real digital computer has finite memory and is therefore less powerful than the idealized Turing machine.

[2] There is some variation in the use of the term *perceptron*. Some people call the network in Figure E–1 a "multilayer perceptron," and use "perceptron" to refer only to a network with a single layer of synapses. Here we use *perceptron* as a generic term covering both multilayer and single-layer perceptrons.

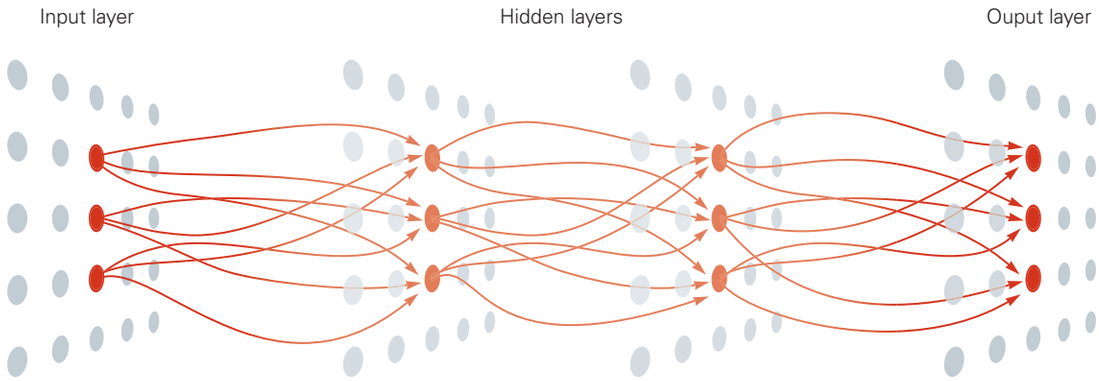

Figure E–1 The perceptron model. A perceptron is a network of idealized neurons arranged in layers with synaptic connections from each layer to the succeeding one. In general, any number of "hidden layers" may intervene between the input and output. Each disk represents a neuron. An arrow pointing from the presynaptic neuron to the postsynaptic neuron represents a synapse. There are no loops in the network.

constructed from various kinds of model neurons, we will use the simple McCulloch-Pitts neurons.

The computations in a perceptron, as in the visual system, occur through both sequential and parallel processing of information. The layers of a perceptron can be regarded as a sequence of steps in a computation. The neurons within each layer perform similar operations that are executed in parallel during a single step of the computation. Because vision is often quite fast compared to other cognitive tasks, it may require only a few sequential steps, but each step involves a large number of operations performed by many neurons working in parallel. It is natural to represent this kind of computation by a perceptron with a small number of layers, each with many neurons.

Simple and Complex Cells Could Compute Conjunctions and Disjunctions

We shall develop the analogy between perceptrons and the visual system by exploring its implications for primary visual cortex (V1). As discussed in Chapter 28, the "simple cells" of V1 respond selectively to stimuli in the visual field that have a certain spatial orientation. A simple cell responds to a bar of light close to a particular orientation but not to bars with other orientations.

In a classic 1962 paper David Hubel and Torsten Wiesel described this property of orientation selectivity in V1 and also proposed the first model of how it is achieved. They assumed that what they called a "simple" cortical cell receives synaptic inputs from cells in the lateral geniculate nucleus (LGN) and suggested that orientation selectivity depends on the spatial arrangement of the receptive fields of the LGN cells. Thus, if the center-surround receptive fields of the LGN cells were arranged along a straight line (see Figure 28–12), a bar of light with the same orientation as this line would activate all the LGN inputs of the simple cell simultaneously, driving the simple cell that receives these inputs above the threshold for firing action potentials. Conversely, a bar of light at nonpreferred orientations would stimulate only some of the LGN inputs, leaving that simple cell below threshold for firing.

The preceding model of a simple cell can be interpreted as a McCulloch-Pitts neuron computing an AND operation (Figure E–2A), because a simple cell fires when *all* of its LGN inputs are activated. Recall that a McCulloch-Pitts neuron computes a conjunction if its threshold is set sufficiently high, and intuitively it makes sense that a high threshold goes along with high selectivity.

In addition to simple cells, V1 also contains "complex" cells, also first described by Hubel and Wiesel. Like simple cells, complex cells are orientation selective, but their responses are not sensitive to the location of the stimulus within the receptive field, whereas simple cells are quite sensitive to the precise alignment of the stimulus within the excitatory subregions of their receptive field.

Hubel and Wiesel proposed that a complex cell receives synaptic input from simple cells with similar orientation selectivity (Figure E–2C). The receptive fields of the simple cells add together to form the receptive field of the complex cell. If a visual stimulus

with the preferred orientation activates any one of the simple cells, the complex cell is driven over the threshold for firing. This model is intended to explain why spatial location of the stimulus in the receptive field is not a factor in activating the complex cell.

This model of a complex cell can be interpreted as a McCulloch-Pitts neuron computing an OR operation (Figure E–2B) since a complex cell fires when *any* of its simple cell inputs is activated. A McCulloch-Pitts neuron computes a disjunction if its threshold is set sufficiently low, and intuitively it makes sense that a low threshold is appropriate for nonselective responses.

In effect, Hubel and Wiesel imagined simple and complex cells as McCulloch-Pitts neurons, although they did not use such language. For a McCulloch-Pitts neuron, the threshold determines whether responses are selective or invariant. The simple cell's high threshold is responsible for the cell's orientation selectivity, while the complex cell's low threshold accounts for the invariance of its response to the location of the stimulus within its receptive field.

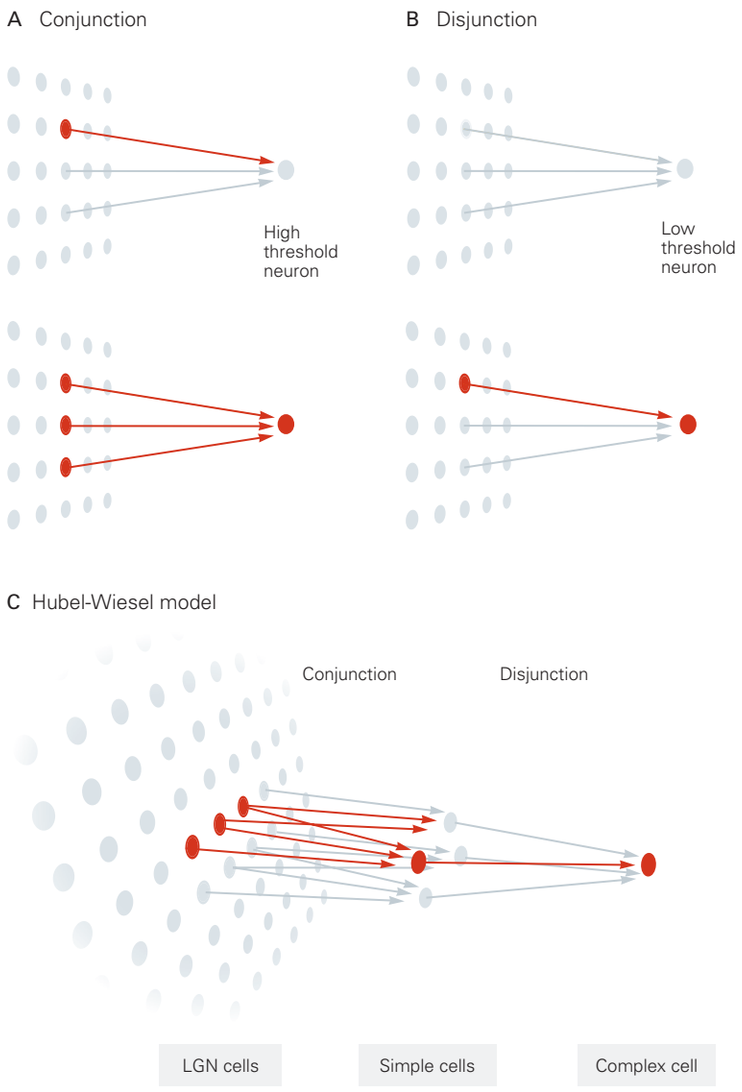

Figure E–2 A perceptron implementing conjunction (AND), disjunction (OR), and the Hubel-Wiesel neurobiological model of simple and complex cells in visual cortex. Neurons are represented by disks and synapses by arrows. Active neurons and synapses are colored red.

A. A neuron with a high threshold can compute the conjunction of three inputs. The neuron does not respond to only one input (**top**) or two inputs (not shown). It becomes active only when all three inputs are active (**bottom**).

B. A neuron with a low threshold can compute a disjunction of three inputs. The neuron remains inactive if all of its inputs are inactive (**top**). It becomes active if a single input neuron is active (**bottom**) or more than one input neuron is active (not shown).

C. In this realization of the Hubel-Wiesel model a disjunction neuron (**right**) receives inputs from a set of conjunction neurons (**middle**), which in turn receive inputs from a grid of neurons (**left**). The neurons in the grid represent lateral geniculate nucleus (LGN) cells, which are assumed to be either all ON-center or OFF-center cells and retinotopically organized so that the location of each cell in the grid corresponds to the location of its receptive field on the retina. A horizontally oriented visual stimulus activates three LGN cells in a row, which activate a "simple cell" (conjunction) that in turn activates a "complex cell" (disjunction). Like actual simple cells of primary visual cortex, each conjunction neuron responds selectively to stimuli with a particular orientation (horizontal in this case) and at a particular location. Likewise, like actual complex cells, the disjunction neuron responds selectively to stimuli with a particular orientation but is invariant to the exact location of the stimulus.

The Primary Visual Cortex Has Been Modeled as a Multilayer Perceptron

If the Hubel-Wiesel model is extended to many neurons, each with a receptive field that covers a different location in the visual field and tuned to a preferred orientation, then it amounts to a perceptron with three layers of neurons (Figure E–3).

Indeed, like this perceptron, visual areas of the brain generally have a retinotopic organization: Neighboring cells have receptive fields that cover adjacent areas in the visual field. This means that a sheet of cortical tissue functions like a map of the visual field, and its activity patterns can actually resemble images. Similarly, each layer of the model in Figure E–3 is retinotopically organized so that at any moment a map of the overall activity pattern of its neurons depicts the stimulus image. Connections between the layers respect the spatial arrangements of receptive fields described above and shown in Figure E–2. The thresholds are set to yield conjunctions and disjunctions in simple cell and complex cell layers, respectively.

The structure of the model is idealized in a number of ways to facilitate understanding. All cells are arranged in uniformly spaced grids. Furthermore, both the simple cell and complex cell layers are divided into a number of "feature maps." All cells in a feature map detect exactly the same feature but in different locations of the visual field. In the cortex, the cells detecting different features would be intermingled, but in the model they are segregated for convenience.

A map of active neurons in the LGN layer of the model resembles the visual stimulus, whereas the simple and complex cell layers contain more abstract representations of the stimulus because of the orientation selectivity of neurons. In particular, the representation of the stimulus in the complex cell layer is robust and does not reflect small variations in the stimulus (see Figure E–3).

Selectivity and Invariance Must be Explained by Any Model of Vision

The dichotomy between selectivity and invariance has been important in our discussion of the primary visual cortex and simple stimuli like bars. More generally, this dichotomy is relevant throughout the visual system

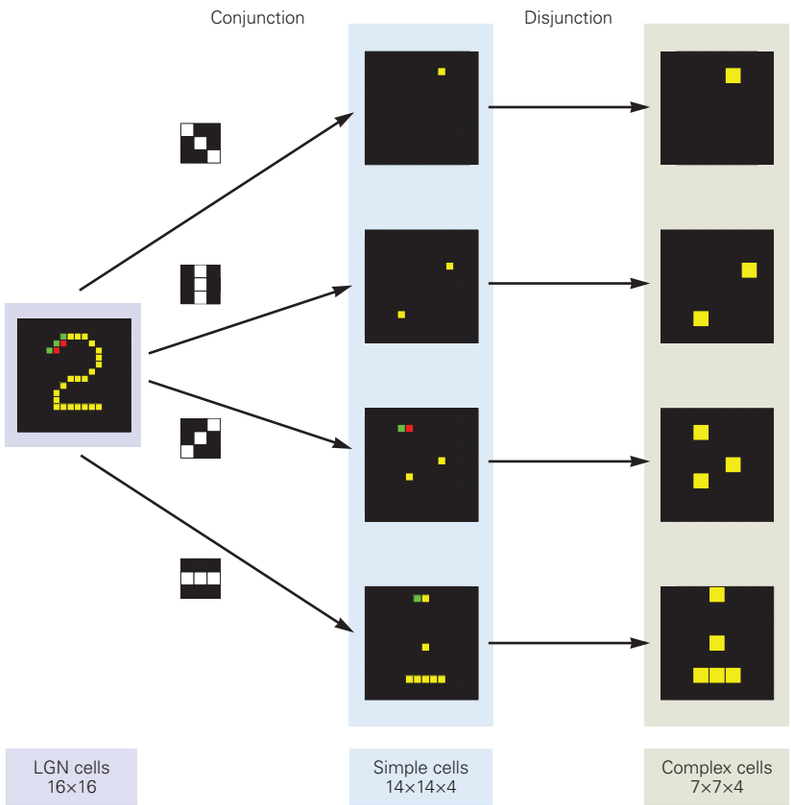

Figure E–3 A perceptron implementing the Hubel-Wiesel model of selectivity and invariance in. The network in Figure E–2C can be extended to grids of many cells by specifying synaptic connectivity at all locations in the visual field. The resulting network can be repeated four times, one for each preferred orientation (horizontal, vertical, and two diagonals). This yields four retinotopically organized grids of simple cells, one for each preferred orientation, as well as four grids of complex cells. Each grid is called a *feature map*. Throughout the network the responses to two slightly different images of the numeral 2 are superimposed for comparison. A **yellow pixel** indicates a neuron that responds to both stimuli. A **red pixel** indicates a neuron that responds to one of the stimuli, and a **green pixel** indicates a neuron that responds to the other.

In the LGN layer the difference between the two stimuli is evident (see red and green pixels at the top of the numeral). In the simple cell layer the bottom two feature maps show different responses to the images (red and green pixels), but the top two are the same (all yellow pixels). Finally, the responses of the complex cells are the same for both images (all yellow pixels). Thus invariance and selectivity occur together in one network, although the invariance is limited (it does not hold for all distortions) and the selectivity is fairly simple.

and even for complex stimuli like entire objects. Let's step back and think about the computations that the entire visual system must accomplish.

Even though the act of seeing is effortless for humans and animals, vision evidently is a difficult computational problem. In spite of enormous progress in algorithms, speed, and memory capacity, modern digital computers are still far from equaling the performance of biological vision systems. In particular, one of the main functions of vision is the recognition of objects. One reason this task is difficult for computers is that the images of a single object are highly variable. Factors such as lighting, location, and distance all cause changes in retinal images that the visual system must *ignore* when it recognizes objects—recognition requires some invariance in responding—to image changes. The visual system cannot, however, ignore all changes, because it has to distinguish between different objects—it must therefore also be selective for certain aspects of images. Although the properties of invariance and selectivity may seem conflicting, they are somehow reconciled by the visual system.

How does the visual system accomplish object recognition? Neurophysiologists have investigated this question by recording from high-level visual areas, such as inferotemporal cortex. To give one example of their findings, certain inferotemporal neurons respond selectively to images of faces. These face-selective neurons have large receptive fields and the exact location of the face within the receptive field is not a factor in the cells' responses. Instead, the responses appear to be closely related to complex features or entire objects rather than simple features like bars or edges.

How are selectivity and invariance achieved by the face-selective neurons? According to one theory, all visual cortical areas are arranged in a hierarchy (see Figure 25–11) and the Hubel-Wiesel model of simple and complex cells in the primary visual cortex (V1) can be generalized to the higher levels of the visual system. In this hierarchical model V1 is at the bottom and areas in the inferotemporal cortex are near the top. Neurons near the bottom of the hierarchy are selective for simple features, have small receptive fields, and are sensitive to small changes in stimulus location. Neurons near the top of the hierarchy are selective for complex features, have large receptive fields, and are invariant to large changes in stimulus location. Neuronal connections from each level to the next are organized so as to carry out computations analogous to the ones performed by simple and complex cells in V1. As we shall see, this hierarchical conception of visual recognition of objects has been formulated precisely in a number of neural network models.

Visual Object Recognition Could Be Accomplished by Iteration of Conjunctions and Disjunctions

Could perceptrons be used to model not just V1 but also the rest of the visual system? We introduced the idea that conjunctions create selectivity in V1, and disjunctions create invariance. Repeated alternation between conjunctions and disjunctions can be used to build up progressively greater selectivity and invariance, culminating in invariant recognition of entire objects.

Indeed, this idea was implemented in 1980 by Kunihiko Fukushima in the neocognitron, a network model designed to recognize handwritten digits. Handwritten numbers may be less complex than images of natural stimuli such as faces or animals, but they are still quite challenging to recognize, as postal workers or anyone who has ever graded handwritten exams can attest. Indeed, digits produced by different writers often look very different, and even repetitions by a single writer can vary considerably.

The neocognitron has a multilayer, feedforward architecture like that of a perceptron (although inhibition is treated somewhat differently).[3] The first layer functions like a retina in which neurons represent an image of a handwritten digit, and subsequent layers contain multiple feature maps (Figure E–4). Although the first layers are analogous to the layers of simple cells and complex cells of the network in Figure E–3, the subsequent layers are meant to model visual areas of cortex beyond V1. Using Boolean logic as an approximation to the operations performed by the elements in the neocognitron, one can say roughly that layers alternate between computing conjunctions and disjunctions.[4] In other words, the conjunction-disjunction scheme of the Hubel-Wiesel model is cascaded to form a hierarchical system. In the output layer, retinotopic organization disappears completely. There are only 10 output neurons, each of which is selective for one of the digits "0" through "9." In a number of simulations the output neurons show an impressive degree of invariance to the location of the digit in the retina as well as to distortions of the digit.

A similar model, called LeNet, was later developed by Yann LeCun and his colleagues. This model adheres

[3]The strengths of the synapses in the neocognitron were not specified by its designer. Instead, the neocognitron learned from a sequence of visual stimuli through synaptic modifications based on a model of Hebbian plasticity (see Box E–2).

[4]Boolean logic is just an approximation, as the model neurons in the neocognitron are actually analog rather than binary.

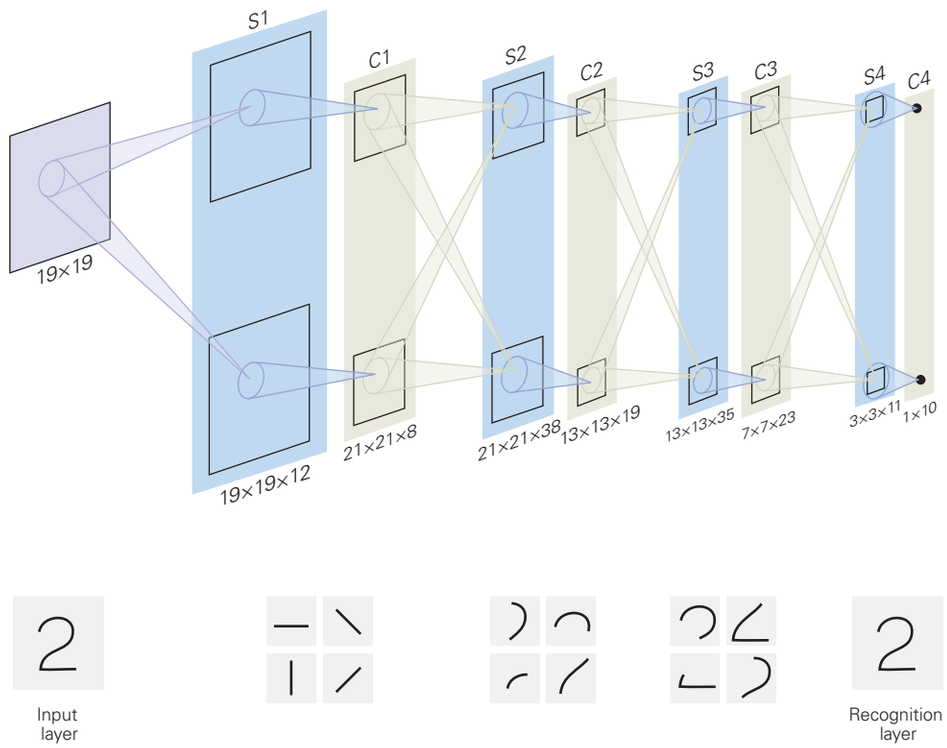

Figure E–4 The neocognitron model of object recognition. Each layer in the network is composed of a set of feature maps, and alternating layers contain "S-cells" or "C-cells." All feature maps are retinotopically organized because each cell receives input from neighboring cells of the previous layer. Each cell in a feature map detects the same feature but at different locations in the image.

An S-cell is analogous to a simple cell in the Hubel-Wiesel neurobiological model. It detects conjunction of features detected by C-cells in the previous layer. A C-cell is analogous to a complex cell in the Hubel-Wiesel model. It can be activated by any of the S-cells in the previous layer, which detect the same feature but at slightly different locations in the image. Receptive fields of cells become larger until the retinotopic organization vanishes completely in the final (recognition) layer.

The neocognitron was constructed for the purpose of recognizing images of handwritten digits. Accordingly, the output neurons are detectors for the digits "0" through "9" and are highly invariant to small variations. Each S-cell layer generates more complex feature selectivity, and each C-cell layer yields more spatial invariance.

The images at the bottom are examples of preferred stimuli of cells in each layer. S1 and C1 cells respond selectively to oriented bars; S2 and C2 cells are selective for more complex features, such as the conjunction of bars; S3 and C3 cells are selective for still more complex features.

closely to the standard definition of a perceptron. The backpropagation algorithm was used to change the synaptic strengths of LeNet so as to reduce the error rate in recognizing images (Box E–2). LeNet achieved sufficient accuracy in recognizing handwritten characters to be used in some commercial applications. Its descendants are still being used today in the field of computer vision and are competitive with other state-of-the-art approaches.

In the neocognitron and LeNet the Hubel-Wiesel neurobiological model of V1 is elaborated to the entire process of object recognition. In spite of several decades of intense scrutiny, there remain significant hurdles to testing neural network models of visual processing. To test a model, two questions must be addressed. Are there synaptic connections in the brain like those of the model? Is the model a good approximation without other types of connections that are not included? Much experimental evidence concerning these questions is rather indirect and circumstantial. In particular, anatomical techniques for determining the connectivity of cortical circuits are still in their infancy. For example, there is no direct evidence for the hypothesis that simple cells in V1 are driven by

Box E–2 Learning in Neural Networks

The brain can perform many computational tasks that are beyond the capabilities of today's electronic computers, but it is also remarkable for another reason: It is a self-assembled system, wiring up its own synaptic connections, unlike an electronic computer that is actually built by external agents (humans or machines).

To emulate this process of self-assembly or self-organization, many neural models are equipped with dynamic processes that continually reorganize their synaptic connections. Some processes create or eliminate neurons or their connections, whereas others adjust the strengths of existing synaptic connections or change other properties of neurons.

To describe the process of self-organization, it is helpful to introduce some terminology for describing the synaptic organization of neural networks. The term *synaptic weight* is often used to refer to the strength of a particular synaptic connection, whereas the term *synaptic weight matrix* applies to the set of all synaptic weights in a network. The strength of the synapse onto neuron i from neuron j is written as W_{ij}. This is the element of the weight matrix located at the intersection of row i and column j.

In many neural network models the weight matrix evolves in time according to a *synaptic plasticity rule*, a mathematical model governing the modifications of synaptic strengths. This is often called a *learning rule*, although strictly speaking, learning is a behavior of a network rather than a synapse.

The network typically starts out in a naïve state, that is, the weight matrix is initialized with random values. Then the network is exposed to a series of stimuli, each of which causes the weight matrix to be modified by the learning rule. Learning rules can take many forms. Much effort has been devoted to devising them and exploring their properties. The Hebbian rule is popular in neurobiological models; with this rule synapses are modified based on temporally contiguous activity of presynaptic and postsynaptic neurons.

It is common to apply the same learning rule to all synapses (or sometimes all excitatory synapses). In spite of this uniformity, the weight matrix becomes heterogeneous because the learning rule depends on activity, and activity patterns are typically nonuniform across a network. Therefore, very complex networks can be produced by a simple learning rule.

In some cases the life of the network is separated into training and operating phases. In the training phase synapses change, whereas in the operating phase the learning rules are turned off. This is analogous to the way in which plasticity seems stronger in juvenile animals than in adults. In other cases the learning rules may be turned off gradually. In fully online learning the learning rules are never turned off, so that the network is always able to adapt to new situations.

It is commonly assumed that reorganization of neural networks in the brain is a decentralized process in which synapses are modified as a result of the interaction of the pre- and postsynaptic neurons rather than in response to signals from some central authority. The Hebbian rule is an example. A consequence of such localized self-organization is that one synapse on a neuron can be modified while another remains unchanged. Such specificity is generally observed in many biological experiments on Hebbian plasticity, although some exceptions have been reported.

In addition to the signals of pre- and postsynaptic signals, retrograde messengers such as nitric oxide may also play a role in synaptic plasticity (Chapter 13), although their role has not been extensively explored in models. The diffuse neuromodulatory systems also have effects on synaptic plasticity (Chapter 13), and some neural network models have attempted to include interaction between global signals broadcast from a central source and local signals as a factor in synaptic modification.

Learning rules are sometimes classified as unsupervised or supervised. *Supervised learning* involves an external "teacher" that evaluates the performance of the entire network and sends a reward or error signal that somehow reaches the synapses. The learning rule is devised so that it produces synaptic modifications that improve the performance of the network as evaluated by the teacher.

One of the most popular supervised learning methods is known as *backpropagation*. When implemented in a perceptron, an error signal is propagated back through the network, starting with the output neurons and moving toward the input neurons. The synapses are then modified based on neural activity and the backpropagated error signal.

Backpropagation has been used by engineers for practical applications, such as a computer system for recognizing handwritten numbers based on LeNet. However, it is unclear whether backpropagation is a biologically plausible learning mechanism, even if it may be useful for engineers.

Unsupervised learning rules, such as the Hebbian rule, learn from sensory inputs without an explicit error signal. These learning rules can have a number of computational functions, such as associative learning, discovering useful stimulus features, or reducing the dimensionality of complex stimuli. They have been used to model the self-organization of feature maps in the primary visual cortex during the course of neural development (see Box E–3), as well as to train networks like the neocognitron.

LGN neurons with receptive fields lined up in a row, as originally proposed by Hubel and Wiesel, although there is some indirect evidence.

As mentioned earlier, attempts have been made to arrange visual areas of cortex in a hierarchy that is consistent with the known anatomical connections between areas. When the visual system is modeled as a perceptron, only "bottom-up" connections are included. In reality, however, there are also "top-down" connections. In some cases, such as the pathways between LGN and V1, the top-down connections far outnumber the bottom-up ones. It is thought that top-down connections are important for allowing cognitive factors such as expectation to influence perception.

Given these uncertainties and limitations, how useful are perceptrons as models of vision? Although, perceptrons are simplistic—they encompass only a subset of the connections in the visual system—they may capture some essence of the way that neural circuits perform visual computations. Indeed, perceptrons perform impressively on visual tasks such as recognizing handwritten digits, although they still fall short of human performance. Such engineering applications show how far one can push the simple ideas embodied in the perceptron.

Neural networks like the neocognitron and LeNet model the visual system as a perceptron organized into a hierarchy of feature detectors. These models propose an answer to one of the questions posed at the beginning of this appendix: How is the psychological event of recognizing an object related to the huge number of neural events that underlie it? In a hierarchical perceptron the recognition of an object involves a relatively small number of sequential steps, consists of a large number of operations executed in parallel. Each of these operations is very simple, carried out by a neuron that is activated when its synaptic inputs drive it above threshold. The sequential steps alternate between selectivity for more complex features and invariance to small distortions of these features. The neurons at the end of this sequence are selective for entire objects, ignoring variations in their appearance. Thus, object recognition can be considered as an emergent property of the network, one that requires the coordinated activation of many neurons, located at many different steps.

Fifty years after Rosenblatt's pioneering work it is clear that perceptrons have been important in developing models of computations in the visual system. In the study of visual perception, as in other fields of science, formal models have proved to be valuable aids to experimentalists.

Associative Memory Networks Use Hebbian Plasticity to Store and Recall Neural Activity Patterns

The sight of a familiar face evokes a name. A simple odor triggers the vivid recollection of a past meal and the persons who were there. These everyday experiences illustrate that the facts and ideas stored in our memories are associated with each other. Philosophers and psychologists have argued that association is the basic principle of all mental activity. Neuroanatomists have studied the way that neurons are bound together in a web of synaptic connections. The two traditions converge in an intuitively appealing idea: perhaps synaptic connections are the material substrate of mental associations.

This idea has been formalized in a number of neural network models of associative memory. A fundamental assumption in these models is that information is transferred back and forth between neural activity and synaptic connections. When novel information first enters the brain it is encoded in a pattern of neural activity. If this information is stored as memory, the neural activity leaves a trace in the brain in the form of modified synaptic connections. The stored information can be recalled when the modified connections again become active. This scheme assumes that synaptic connections remain stable for long periods of time, whereas neural activity is ephemeral and represents immediate experience only.

The transfer of information from neural activity to synapses is hypothesized to occur through Hebbian synaptic plasticity: A long-lasting increase in synaptic efficacy is induced if the presynaptic neuron repeatedly participates in the firing of its postsynaptic neuron (Box E–3). Some prominent forms of long-term potentiation involving the NMDA-type glutamate receptor are regarded as Hebbian (Chapters 66 and 67). Conversely, the transfer of information from synapses to neural activity is thought to occur through a process of pattern completion in which activity spreads through an assembly of neurons coupled by synaptic loops. This idea will be explained in more detail below.

Hebbian Plasticity May Store Activity Patterns by Creating Cell Assemblies

How might Hebbian plasticity transfer information from neural activity into the synapses of a neural network? One approach is illustrated in Figure E–5, which depicts a population of excitatory neurons that could represent pyramidal neurons in the hippocampus

Box E–3 Mathematical Models of Hebbian Plasticity

Associative memory networks were developed by a number of researchers.[1] In their modern form they have two essential features. First, the synaptic strengths are specified by a special type of matrix, called a correlation matrix. Second, the neurons are nonlinear, which enhances the ability of the models to perform the operation of pattern completion that was described in the main text.[2]

To store an activity pattern in long-term memory in a nonlinear network of the form written in Equation E–1 in Box E–1, synaptic strengths are changed by

$$\Delta W_{ij} \propto x_i x_j \qquad \textbf{Hebbian rule (E–3)}$$

This synaptic learning rule is Hebbian because it depends on the simultaneous activation of the postsynaptic neuron i and the presynaptic neuron j. (For binary neurons the change in Equation E–3 is only nonzero if x_i and x_j are both equal to 1.) If Equation E–3 is repeatedly applied with activity patterns drawn from an ensemble, then W_{ij} becomes proportional to the statistical *correlation* between the activities of neurons i and j (hence the term *correlation matrix*).

A popular modification of the basic Hebbian rule is to replace Equation E–3 by

$$\Delta W_{ij} \propto (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle)$$

Covariance rule (E–4)

where $\langle x_i \rangle$ is the average activity of neuron i. When this is applied to an ensemble of activity patterns, W_{ij} becomes proportional to the statistical *covariance* between the activities of neurons i and j.

The number of patterns that can be stored in synaptic connections is limited because the patterns eventually interfere with each other (see Figure E–6). The maximal number that can be stored is called the *capacity* of the network. In 1985 Daniel Amit, Hanoch Gutfreund, and Haim Sompolinsky introduced techniques from the statistical physics of disordered systems to calculate memory capacity. Later researchers used these techniques to find that the covariance rule of Equation E–4 is generally superior to the basic Hebb rule of Equation E–3 because it reduces interference between patterns and therefore enhances storage capacity.

Physiologists have found that Hebbian plasticity can depend on the precise timing of presynaptic and postsynaptic spiking. One example of such a mechanism is spike-timing dependent plasticity (Chapter 10). To incorporate this dependence, models more sophisticated than Equations E–3 and E–4 have been proposed (see the bibliography at the end of the appendix).

The main text of this appendix focuses on the use of the Hebbian rule in models of associative memory. However, the Hebbian rule has also been used to model the development of retinotopic maps in visual areas of cortex. Also, it is believed that Hebbian plasticity allows neuronal activity to influence the patterning and refinement of connections during neural development (Chapter 56). In 1973 Christoph von der Malsburg advanced a neural network model of primary visual cortex in which Hebbian plasticity underlies the self-organization of orientation maps when the model network is exposed to visual stimuli.

In 1982 Teuvo Kohonen proposed a simplification of von der Malsburg's model, known as the self-organizing map (SOM). Kohonen showed how the SOM served as a general method of mapping the abstract high-dimensional space of stimuli onto a low-dimensional neural representation, as in a sheet of cortical tissue. Kohonen's learning rule causes neighboring neurons in the network to develop preferences for similar stimuli. This yields a low-dimensional map of the stimulus space based on similarities between sensory inputs.

[1] See *Neurocomputing: Foundations of Research* in the bibliography at the end of the appendix.

[2] These two properties were first combined in associative memory networks by Shun-ichi Amari and Kaoru Nakano, working independently in 1972.

or neocortex. It is common to assume that Hebbian plasticity modifies the synapses between pyramidal neurons and does not modify synapses involving inhibitory neurons. According to this theory, inhibitory neurons play only a supporting role in memory storage and recall by helping to prevent overexcitation of the network, or "confused" recall of multiple memories at the same time. For simplicity, inhibitory neurons are not included in the model in Figure E–5.

The initial state of this network has no connections between neurons (Figure E–5A). This should not be taken literally. It depicts an initial situation in which synapses exist, but they are all very weak. Now suppose that three neurons are stimulated by synapses

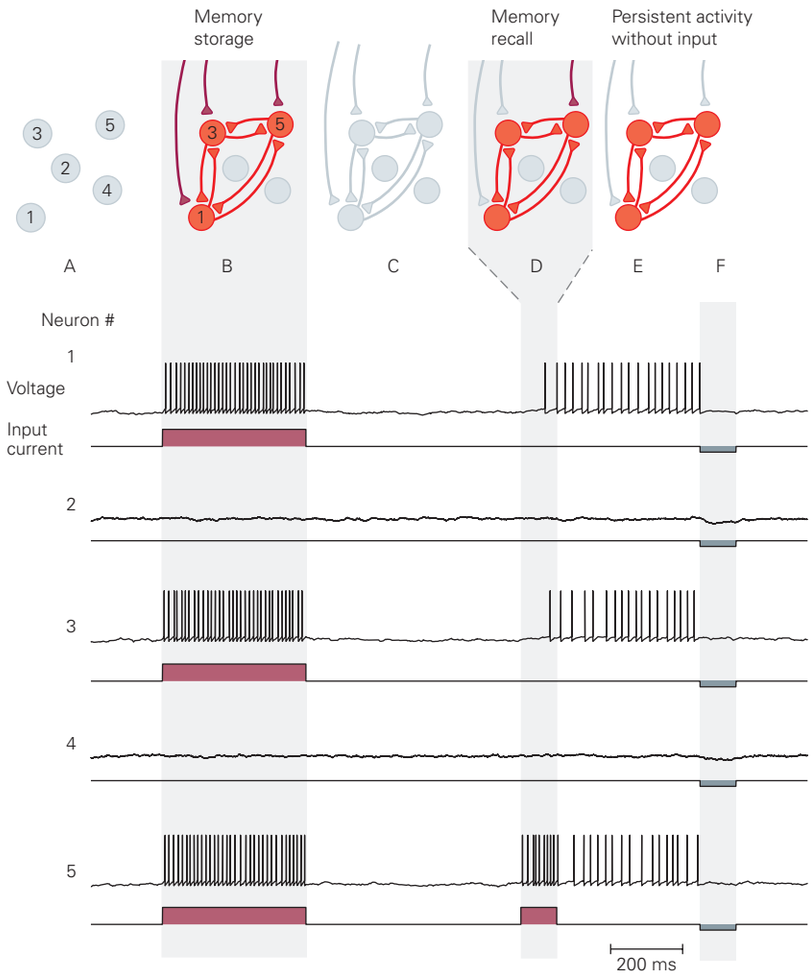

Figure E–5 Associative memory and persistent activity in a network of model neurons. Numerical simulations were done using the leaky integrate-and-fire model neuron described in Appendix F. This model neuron generates spike times but not the detailed shape of the action potential.

A. The synaptic connections between five neurons are initially very weak or nonexistent, and here are not drawn at all. Neurons 1, 3, and 5 are about to be activated by external input.

B. Input current activates the three neurons and Hebbian plasticity strengthens the synaptic connections between the neurons, a form of associative memory storage.

C. When the input current ceases neuronal activity also ceases. However, the pattern of activity in the synaptic connections between the three neurons is not abolished when activity ceases.

D. Input current stimulates just one of the original three neurons, but the excitatory connections complete the entire pattern. All three neurons of the pattern become activated. **E.** Even after the input current has ended, the neurons remain persistently active.

F. A nonselective inhibitory input to all the neurons (circuit not depicted) quenches the persistent activity pattern, and the circuit returns to a quiescent state.

from sources outside the circuit. This situation corresponds roughly to activation of a distributed pattern of neural activity in the brain by a sensory stimulus, as is often observed in neurophysiological studies. Every synapse between a pair of active neurons is therefore exposed to coincident presynaptic and postsynaptic activity, thus strengthening the synapses.

After this strengthening has occurred, a group of three neurons that are strongly coupled by excitatory synapses form a *cell assembly* (Figure E–5C). Neuroscientists generally use this term rather imprecisely. One must look to mathematical models of networks for more precise definitions, which generally have something to do with the presence of strong mutual excitatory interactions within a group of neurons. The word "assembly" emphasizes that the group did not initially exist but was constructed through the strengthening of its synapses, which in turn was caused by the simultaneous activation of the neurons in the group.

In effect, the information in the original activity pattern is transferred to the configuration of strong synapses in the cell assembly. Assuming that the synaptic changes persist, the information is maintained even after the original activity pattern has ceased. It could be said that the network has learned an activity pattern by storing it into its synaptic strengths. Moreover, because of this, the resulting cell assembly can recall the original activity pattern, as will be explained below.

Cell Assemblies Can Complete Activity Patterns

If inputs are limited to one neuron in the three-cell assembly, the neuron starts to generate action potentials (Figure E–5D). Although the external inputs to the other two neurons do not change, they also become activated after a short latency because they are driven by synaptic input from the first neuron. This spreading

of activation from one neuron to the other two is an example of *pattern completion*.

Researchers have scaled up the same idea to very large networks, and when only part of the cell assembly is activated the rest of it becomes active. Such a neural process is thought to be responsible for the psychological phenomenon of memory retrieval. Consider the example of seeing a friend and remembering his name and occupation. Partial information triggers recall of more information based on the completion of a neural activity pattern.

In the simulation in Figure E–5, stimulating any one out of the three neurons would result in completion of the entire pattern. This is a kind of symmetry and is analogous to the way in which memory retrieval can be symmetric; it is equally possible for a face to evoke recall of a name and vice versa. Symmetric pattern completion is possible for a cell assembly because of the lack of directionality in its connectivity. Activity can spread in any direction within a cell assembly except that activity in the last layer cannot spread backward to the rest of the network.

In the CA3 region of the hippocampus, a brain area that has been implicated in episodic memory (Chapter 65), pyramidal neurons make synapses onto each other in a recurrent fashion, and Hebbian plasticity has been observed at these synapses. Therefore CA3 seems a prime candidate for a network containing cell assemblies, as many theorists have speculated in the past. The hypothesis that Hebbian synaptic plasticity stores memories as cell assemblies in CA3 has been investigated in studies of hippocampal place cells in rodents, the connections between which are thought to store spatial memories (Chapter 67). Susumu Tonegawa and his colleagues created mutant mice in which a subunit of the NMDA-type glutamate receptor was deleted from CA3 pyramidal neurons. Long-term potentiation was impaired at these synapses, supporting the idea that the NMDA receptor is critical for Hebbian synaptic plasticity. Interestingly, mutant mice are still able to form spatial memories but have difficulty recalling them if some of the original visual landmarks are missing. Tonegawa and his colleagues interpreted this deficit in recall as impaired pattern completion, and ascribed it to impaired formation of cell assemblies in CA3.

Cell Assemblies Can Maintain Persistent Activity Patterns

Up to now our discussion of memory storage in synaptic connections has focused on long-term memory. Recall of a long-term memory occurs through the reactivation of a previous activity pattern, triggered by activation of a subset of the pattern. Once the activity pattern has been reactivated it can persist even after the extrinsic drive has ended because the neurons excite each other through their mutual excitatory connections.

Such persistent activity could also function as a short-term memory trace of the input that activated it. This is our first mention of short-term memory, which is generally regarded as distinct from long-term memory. For example, the famous patient H.M. lost the ability to store new long-term memories but had intact short-term memory, evidence that these are two distinct functions (Chapter 65).

A classic example of short-term memory is the temporary memorization of a phone number for a few seconds after reading or hearing it. After dialing the phone number the information is rapidly lost from memory. If the phone number becomes too long, as when dialing internationally, it can be difficult to retain for even a few seconds. As this example illustrates, short-term memories last for only a short time and contain limited information. In contrast, long-term memories can last a lifetime, and our brains seem to have a virtually unlimited capacity for them.

As described in Chapter 67, similar short-term persistent activity has been observed in the primate brain during the performance of delayed-match-to-sample tasks that are designed to test short-term memory. For example, in each trial of an experiment a monkey views a sample image on a screen, then a blank screen during a delay period, and then another image. The monkey is trained to indicate whether the second image matches the first. In the primary visual cortex neural activity is observed only when the images are presented. However, in higher-level areas, such as inferotemporal and prefrontal cortex, persistent activity is also observed during the interval between images (see Figure 28–11). By sampling many neurons during this delay period, neurophysiologists have recorded distinct activity patterns corresponding to different sample images, suggesting that these activity patterns encode information about previously viewed images.

To summarize, the cell assembly concept has been used to explain both long-term and short-term memory. According to this concept a long-term memory is stored as strengthened connections between neurons in a cell assembly, while a short-term memory is maintained by persistent activity of the neurons in a cell assembly. Whether these ideas are correct remains uncertain, and some of their problematic aspects will be noted later. It should also be noted that not all associative memory networks depend on persistent activity. For example, in the network in Figure E–5 some numerical parameters could be changed so that pattern completion occurs during the stimulus presentation and does not persist after the stimulus is gone.

Interference Between Memories Limits Capacity

Figure E–5 illustrates the storage and retrieval of a single activity pattern. In fact, however, a single network can store multiple patterns. If Hebbian synaptic modifications store multiple patterns, many cell assemblies are created. A stored pattern can be retrieved by stimulating some of the neurons in the corresponding cell assembly, leading to completion of the entire activity pattern.

However, the storage capacity of a network is not infinite. If the cell assemblies are completely nonoverlapping (share no neurons in common) they will not interfere with each other (Figure E–6A), and in these cases the number of patterns that can be stored is equal to the total number of neurons in the network divided by the size of a cell assembly.

Higher storage capacity can be achieved if the cell assemblies overlap (share neurons). However, overlap means there is the possibility of interference (Figure E–6B). Interference can lead to corruption of memories, so that the activity patterns expressed by the network deviate from the original patterns that were stored by Hebbian plasticity. If we attempt to store too many patterns in the network, interference eventually becomes catastrophic—the stored patterns disappear altogether. Therefore interference effects limit the storage capacity of the network, and mathematical theorists have studied these effects in detail.

Synaptic Loops Can Lead to Multiple Stable States

A cell assembly can maintain an activity pattern triggered by an external input even after the input has ceased. To describe this phenomenon we use the concept of *multistability*, a term from dynamical systems theory.

In Figure E–5 the circuit is active during the interval between D and E but quiescent both before and after the interval. During the quiescent and active states there is no external input, yet the circuit has two very different firing patterns. Thus the network possesses two possible stable states (active and inactive) for a single input, a phenomenon known as *bistability*. The transient currents at D and E in Figure E–5 switch the circuit from one stable state to the other.

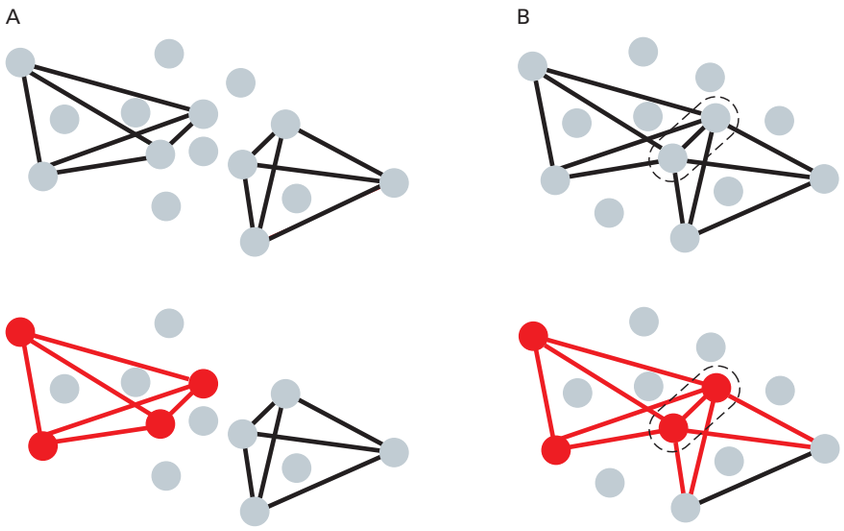

Figure E–6 The potential for interference between overlapping associative memory networks. Each link in the diagram represents a bidirectional pair of excitatory synapses.

A. Two non-overlapping cell assemblies. Each assembly is a group of neurons that is fully coupled by strong excitatory synapses. Because the cell assemblies share neither neurons nor synapses in common, they are completely independent. The lower drawing shows the activation of one cell assembly (**red**). Alternatively, the other cell assembly can be activated or both assemblies can be activated simultaneously (not shown).

B. Two overlapping cell assemblies. Because some neurons are involved in both cell assemblies (**dashed line**), there is potential for interference. Activation of one cell assembly could potentially spread to the other cell assembly (**lower drawing**). This can be prevented. If the threshold for neural activation is sufficiently high, the neurons belonging uniquely to the second cell assembly remain below threshold. Conversely, if the threshold is low, then it will be impossible to activate a single cell assembly without the other (not shown).

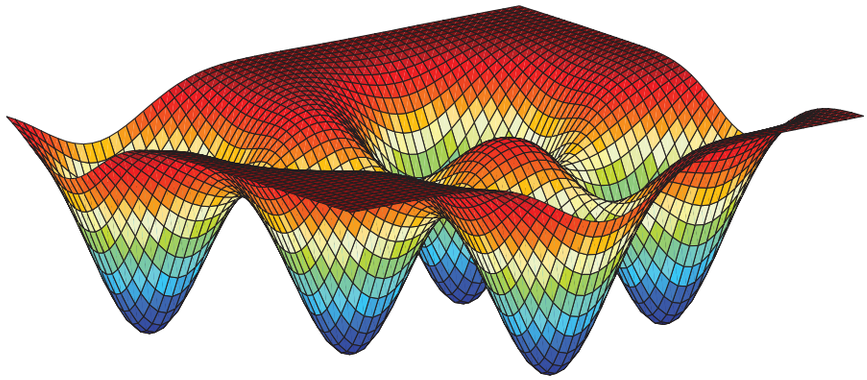

Figure E–7 Multiple stable states can be depicted as minima of an energy-like function. The dynamics of a multistable dynamical system can be visualized as descent on an energy landscape with multiple valleys.

A network with multiple cell assemblies is said to be *multistable* because activation of any one cell assembly produces a distinct stable state of the network. When a multistable system is at a steady state, this state depends on past as well as present input. This dependence on the past explains why the transient inputs of Figure E–5 can have a lasting effect on activity.

Multistability is caused by the connectivity of the cell assemblies. More generally, networks that contain synaptic loops (a cell assembly is a special case of this) can have multiple stable states. In contrast, a perceptron is a type of network that has no loops and does not exhibit multistability. A network with multiple stable states is often called an *attractor neural network*, a term that also comes from dynamical systems theory.[5] A stable steady state is called an attractor of the dynamics because dynamical trajectories (the temporal evolution of the activity of the network) that start from similar initial conditions will converge (are attracted) to the stable state.

Symmetric Networks Minimize Energy-Like Functions

Further insight into multistability can be gained from a physical analogy. If the curved surface shown in Figure E–7 is slippery, a small object placed on the surface will slide downhill, ultimately coming to rest near the bottom of a valley, assuming that there is a little friction to damp the motion. The object could end up in any one of the valleys, depending on its starting point. Therefore the dynamics of the object is multistable.

The object's motion can be understood using the physical concept of energy. Because the gravitational potential energy of the object is a linear function of its height, the surface can be regarded as a graph of energy versus location in the horizontal plane. The object behaves as if its goal were to minimize its potential energy, in the sense that its downhill motion causes the potential energy to decrease until a minimum is reached. The multiple stable states correspond to the multiple minima of the energy.

In an influential paper published in 1982, John Hopfield constructed a mathematical function that assigns a number to any activity pattern of a neural network model. He proved mathematically that this number is guaranteed to decrease as the activity of the network evolves in time until a stable state is reached. Because of this property, Hopfield's function represents the "energy of the network," and we will call it an *energy function*.[6] The energy of the network is analogous to the height of the sliding object in Figure E–7, and the activity of the network is analogous to the horizontal location of the sliding object. Of course, Figure E–7 is an impoverished depiction of a network energy function because the activity pattern of a network of n neurons is an n-dimensional vector, not a two-dimensional location in the horizontal plane.

As a special case, Hopfield applied the energy function to associative memory networks, showing that the process of memory recall by pattern completion (Figure E–5) is analogous to an object sliding down an energy landscape (Figure E–7).

[5]Some apply the term *attractor network* rather loosely to any recurrent network, whereas others restrict application to recurrent networks with multistability.

[6]It should be stressed that this is an analogy, that the energy function is distinct from energy in the sense of physics. A minimum of the network energy function might actually correspond to an activity pattern in which neurons are firing at high rates and using large amounts of energy.

Hopfield's construction of the energy function required that the interactions between neurons be symmetric: Any connection from one neuron to another is mirrored by another connection of equal strength in the opposite direction. This is the case, for example, in the cell assemblies of Figure E–5. Although perfect symmetry of interactions is not biologically plausible, approximate symmetry might be a property of some biological neural networks, so that Hopfield's networks might be regarded as an idealization of them.

In 1986 Hopfield and David Tank pointed out that a neural network with an energy function can be used to perform a type of computation known as *optimization*. Many interesting problems in computer science can be formulated as the optimization of some kind of function. For example, in the traveling salesman problem, a salesman would like to find the shortest route by which he can visit multiple cities and return to his starting point. In this problem the function to be optimized is the length of the route. Hopfield and Tank showed how to construct a network that finds solutions to the traveling salesman problem. The energy of their network is equal to the distance of the route, which is encoded by the activity of the network. Because the network converges to a minimum of the energy, it effectively searches for an optimal solution to the traveling salesman problem.

This general approach was applied by many others to construct neural networks that solve a variety of optimization problems. The Hopfield-Tank approach could be viewed as an extension of the cell assembly to a general method of encoding a computational problem in the connections of a recurrent network, which solves the problem by converging to a steady state.

Hebbian Plasticity May Create Sequential Synaptic Pathways

In the simulations in Figure E–5 it is assumed that synapses between pairs of neurons are strengthened when the two neurons are active simultaneously. If signaling flows in both directions between the neurons, the synapses in both directions will be strengthened, preserving the symmetry of the interactions between the neurons.

However, Hebb actually argued that a synapse is strengthened when the presynaptic neuron is activated immediately before the postsynaptic neuron (activity in the presynaptic cell leads to an excitatory postsynaptic potential that contributes to firing the postsynaptic action potential). Hebbian plasticity that depends on temporal order has been observed in spike timing-dependent plasticity (Chapter 10). The temporal asymmetry of this learning rule can lead to synaptic connectivities that are asymmetric, as opposed to those shown in Figure E–5.

Such asymmetry in the connectivity could be appropriate for the storage and recall of motor sequences, needed in skills such as playing a musical instrument, and which consist of temporally structured steps. Motor sequences are presumably created by sequential activation of groups of neurons. One can imagine storing a sequence in a network by giving it extrinsic inputs that activate neurons in some order. Hebbian plasticity would lead to a set of strengthened connections that are organized like the perceptron of Figure E–1. Later on the sequence could be recalled by activating the first group of neurons, which would activate the second group, and so on. The network would generate the sequence that had been stored by extrinsic input. This would be another example of pattern completion, one in which the pattern is a temporal sequence rather than a stable state as in Figure E–5.

In this hypothetical example the strong connections all point in the same direction, so that the interactions in the network are asymmetric. The network is unable to generate the same sequence in the opposite order because of the asymmetry. This is consistent with the fact that many well-practiced motor sequences are difficult to carry out in reverse order.

It seems plausible that symmetric and asymmetric connections could be important for storing different types of associations. The memory of a telephone number is sequential and asymmetric; remembering it forward is much easier than trying to remember it backward. But other types of associations are more symmetric: A face may evoke a name as easily as a name evokes a face.

In associative memory networks long-term memories are stored through modifications of synaptic strengths that last for long times. But such Hebbian style long-term potentiation is just one type of modification of biological synapses (Chapter 67). The strengths of biological synapses can change more transiently. Diverse types of transient modification—short-term facilitation, short-term depression, augmentation, and so on—have been classified by their time scales and other properties. It is natural to speculate that these different forms of synaptic plasticity could be used by the brain for a whole spectrum of memory processes with different time scales. In this view a firm distinction between short-term and long-term memory is too simplistic.

Previously we explained that a cell assembly supports both short-term and long-term memories. Does this mean that one can only maintain short-term memories of items that have already been stored as

long-term memories? Everyday experience suggests that one can briefly maintain a short-term memory of a telephone number that has never been encountered before, for which no long-term memory exists. This issue could perhaps be solved if, as suggested above, the sharp distinction between short-term and long-term memory were replaced by a spectrum of memory processes with different time scales.

As noted in the introduction, the idea that persistent activity is maintained by cell assemblies, or more generally by synaptic loops, dates back to Hebb and Lorente de Nó. Many researchers have developed detailed and realistic simulations based on this idea, simulations that are more convincing than the simple one shown in Figure E–5. But demonstrating empirically that a specific example of persistent activity in the brain is caused by synaptic loops has been difficult. Persistent activity could also arise as an intrinsic property of the biophysics of single neurons, rather than an emergent property of networks. Hence the biological mechanisms of persistent activity are still controversial.

Perceptrons and associative memory networks are two historic types of neural network models still in use today. A perceptron is a layered network with no synaptic loops. Its layers represent sequential steps of a computation, where each layer can be regarded as many operations performed in parallel. The visual system has been modeled as a perceptron in which neurons are feature detectors and are hierarchically organized. According to this hierarchical perceptron model, visual recognition of an object is a sequential process in which each step consists of many feature detection events executed in parallel.

Because perceptrons lack synaptic loops, their dynamical behaviors are relatively simple. But the dynamics of the brain can evolve in ways that are disconnected from immediate sensory stimuli or motor responses. These rich intrinsic dynamics are likely to depend on loops in the synaptic connectivity of the brain. Hebbian plasticity is thought to create cell assemblies, which contain synaptic loops. These loops can endow a neural network with the property of multistability, and also lead to persistent activity patterns resembling those observed in neurophysiology experiments on short-term memory. Finally, symmetric neural networks, which contain synaptic loops, have been used to solve optimization problems and could therefore be viewed as a general class of computational devices.

Although decades have passed since perceptrons and associative memory networks were invented, it is still unclear how well these models explain visual perception and the storage and recall of memories. Given that these are some of the deepest and most complex issues in neuroscience, perhaps it is not surprising that testing the models experimentally is difficult. But given today's rapid progress in developing new experimental methods, one could imagine that neural network models will eventually come to play as central a role in systems neuroscience as the Hodgkin-Huxley model of the action potential plays in cellular neurophysiology.

Sebastian Seung
Rafael Yuste

Selected Readings

Anderson JA, Pellionisz A, Rosenfeld E. 1990. *Neurocomputing 2: Directions of Research*. Cambridge, MA: MIT Press.

Anderson JA, Rosenfeld E. 1988. *Neurocomputing: Foundations of Research*. Cambridge, MA: MIT Press.

Churchland PS, Sejnowski TJ. 1992. *The Computational Brain*. Cambridge, MA: MIT Press.

Dayan P, Abbott LF. 2001. *Theoretical Neuroscience*. Cambridge, MA: MIT Press.

Hebb DO. 1949. *The Organization of Behavior*. New York: Wiley.

Hopfield JJ, Tank DW. 1986. Computing with neural circuits: a model. Science 233:625–633.

Minsky ML. 1967. *Computation: Finite and Infinite Machines*. Englewood Cliffs, NJ: Prentice-Hall.

Rumelhart DE, McClelland JL. 1986. *Parallel Distributed Processing, Vol. 1: Explorations in the Microstructure of Cognition*. Cambridge, MA: MIT Press.

McClelland JL, Rumelhart DE. 1986. *Parallel Distributed Processing, Vol. 2: Psychological and Biological Models*. Cambridge, MA: MIT Press.

Rolls ET, Treves A. 1998. *Neural Networks and Brain Function*. New York: Oxford Univ. Press.

Trappenberg TP. 2002. *Fundamentals of Computational Neuroscience*. New York: Oxford Univ. Press.

References

Abbott LF, Nelson SB. 2000. Synaptic plasticity: taming the beast. Nat Neurosci 3:1178–1183.

Amari S-I. 1972. Learning patterns and pattern sequences by self-organizing nets of threshold elements. IEEE Trans Comput C-21:1197–1206.

Amit DJ, Gutfreund H, Sompolinsky H. 1985. Spin-glass models of neural networks. Phys Rev A32:1007–1018.

Bain A. 1873. *Mind and Body: The Theories of Their Relation*. New York: Appleton.

Ben-Yishai R, Bar-Or RL, Sompolinsky H. 1995. Theory of orientation tuning in visual cortex. Proc Natl Acad Sci U S A 92:3844–3848.

Bonhoeffer T, Staiger V, Aertsen A. 1989. Synaptic plasticity in rat hippocampal slice cultures: local "Hebbian" conjunction of pre- and postsynaptic stimulation leads to distributed synaptic enhancement. Proc Natl Acad Sci U S A 86:8113–8117.

Cohen MA, Grossberg S. 1983. Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. IEEE Trans Syst Man Cybern SMC-13:815–826.

Felleman DJ, Van Essen DC. 1991. Distributed hierarchical processing in the primate cerebral cortex. Cerebr Cortex 1:1–47.

Ferster D, Miller KD. 2000. Neural mechanisms of orientation selectivity in the visual cortex. Annu Rev Neurosci 23:441–471.

Fukushima KM. 1980. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol Cybern 36:193–202.

Griniasty M, Tsodyks MV, Amit DJ. 1993. Conversion of temporal correlations between stimuli to spatial correlations between attractors. Neural Comput 5:1–17.

Hopfield JJ. 1982. Neural networks and physical systems with emergent collective computational abilities. Proc Natl Acad Sci U S A 79:2554–2558.

Hubel DH, Wiesel TN. 1962. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. J Physiol 160:106–154.

Kohonen T. 1989. *Self-organization and Associative Memory*. Berlin: Springer-Verlag.

LeCun Y, Boser B, Denker JS, Henderson D, Howard RE, Hubbard W, Jackel LD. 1989. Backpropagation applied to handwritten zip code recognition. Neural Comput 1:541–551.

Lorente de Nó, R. 1938. Analysis of the activity of the chains of internuncial neurons. J Neurophysiol 1:207–244.

Major G, Tank DW. 2004. Persistent neural activity: prevalence and mechanisms. Curr Opin Neurobiol 14:675–684.

McNaughton BL. 1996. Deciphering the hippocampal polyglot: the hippocampus as a path integration system. J Exp Biol 199:173–185.

Leutgeb S, Leutgeb JK, Moser MB, Moser EI. 2005. Place cells, spatial maps and the population code for memory. Curr Opin Neurobiol 15:738–746.

Nakazawa K, Quirk MC, Chitwood RA, Watanabe M, Yeckel MF, Sun LD, Kato A, et al. 2002. Requirement for hippocampal CA3 NMDA receptors in associative memory recall. Science 297:211–218.

Riesenhuber M, Poggio T. 1999. Hierarchical models of object recognition in cortex. Nat Neurosci 2:1019–1025.

Shapley R, Hawken M, Ringach DL. 2003. Dynamics of orientation selectivity in the primary visual cortex and the importance of cortical inhibition. Neuron 38:689–699.

Somers D, Nelson SB, Sur M. 1995. An emergent model of orientation selectivity in cat visual cortical simple cells. J Neurosci 15:5448–5465.

Tsodyks M, Feigelman M. 1988. Enhanced storage capacity in neural networks with low level of activity. Europhys Lett 6:101–105.