1

Foundations of Neural Networks

1.1 OBJECTIVES OF NEURAL NETWORKS

Neural network research can be divided into two areas of investigation. The first area, the *direct problem,* employs computer and engineering techniques to model the human brain. This type of modeling is used extensively by cognitive scientists (Harley, 1998) and can be useful in a number of domains, including neuropsychiatry (Rialle and Stip, 1994, Ruppin, Reggia, and Horn, 1996), and neurophysiology (Saugstad, 1994). For more detailed coverage of the direct problem, the reader should consult MacGregor (1987) and Aakerlund and Hemmingsen (1998).

The second area, the *inverse problem,* simulates biological structures with the objective of creating computer or engineering systems. The inverse problem is applied extensively in building computer-assisted decision aids used in differential diagnosis, modeling of disease processes, and construction of more complex biomedical models. Part I of this book concentrates mainly on the inverse problem, although the two areas cannot be completely separated since one problem often sheds light on the other.

Neural networks are used to solve problems in which the complete formulation is unknown—that is, no causal model or mathematical representation exists, usually because the problem itself is not completely understood. The neural network uses data to derive patterns that are relevant in differentiating the groups. Neural network models fall into the category of soft computing, as do fuzzy logic approaches, in that solutions are found to approximate problems rather than approximating solutions of exact formulations.

1.1.1 Modeling Biomedical Systems

Historically, numerous modeling techniques have been used, including mathematical approaches and simulation. Some of the early systems were quite successful, especially in the area of drug therapy. Realistic models for most biological systems are still difficult to achieve both because of our limited knowledge and the complexity of these systems. Recent approaches have used chaos theory to address nonlinear dynamics in biological systems. Neural network modeling of biomedical systems comprises the direct problem and has resulted in a number of interesting applications in which neural network models successfully mimic characteristics of human learning as well as providing models of learning disorders. In general, modeling and simulation systems are outside the scope of this book with two exceptions: features of neural networks relevant to modeling and the use of chaos theory in a hybrid system (illustrated in Chapter 18). Modeling using symbolic techniques is considered in Part II of this book.

1.1.2 Establishment of Decision-Making Systems

The use of neural network models as decision aids comprises the inverse problem. These systems have their historical foundations in earlier pattern recognition techniques and limited neural network models.

1.2 BIOLOGICAL FOUNDATIONS OF NEURAL NETWORKS

The motivating factor behind neural network modeling was the structure of biological nervous systems, or biological neural networks. To draw attention to this parallel, neural network models are sometimes referred to as artificial neural networks (ANNs). Although some basics are known about biological nervous systems, a great deal remains unknown.

1.2.1 Structure of the Neuron

Figure 1.1 shows a simple biological cell. A semipermeable membrane that is between 70 and 100 Angstroms in thickness surrounds the cell. In the interior of the cell, components include the nucleus, the mitochondria, and the Golgi bodies. The nucleus

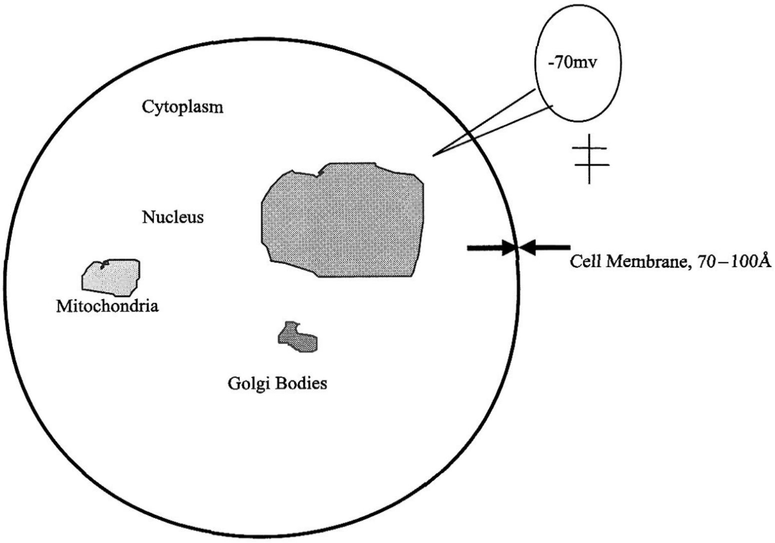

Figure 1.1 Structure of a Biological Cell.

consists of nuclear sap and a nucleoprotein-rich network from which chromosomes and nucleoli arise. A nucleolus contains DNA templates for RNA. The mitochondria produce energy for the cell through cellular respiration. Golgi bodies are involved in the packaging of secretory proteins (Rogers and Kabrisky, 1991).

Figure 1.2 shows a neuron, which is an extension of the simple cell in that two types of appendages have been formed: multiple dendrites and an axon. The dendrites receive input from other neurons, whereas the axon is an output channel to other neurons. Note that a neuron still possesses all the internal features of a regular cell as shown in Figure 1.1. The neuron has important basis characteristics, and it has a number of inputs called *dendrites* and one output called the *axon*. The cell membrane has an electrical resting potential of −70 mV. The resting potential is maintained by pumping positive ions out of the cell. The principal pump is the sodium (Na^+) pump. The main difference between a neuron and an ordinary cell is that the neuron is ex-

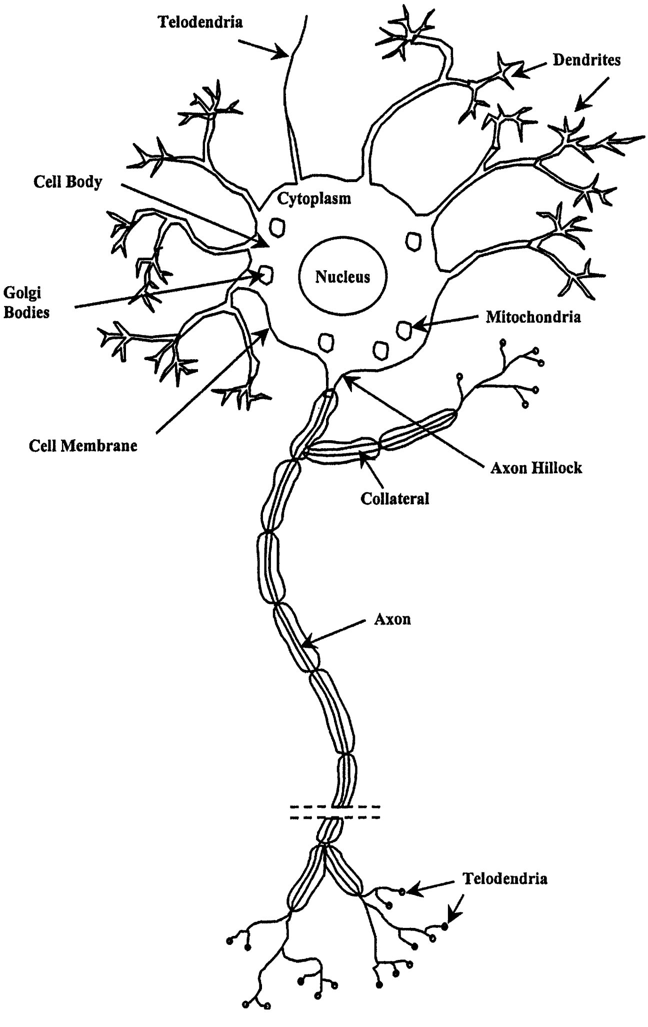

Figure 1.2 Structure of a Neuron.

citable. Because of inputs from the dendrites, the cell may become unable to maintain the −70 mV resting potential, resulting in an action potential that is a pulse transmitted down the axon. Note that the action potential results only after a certain threshold has been exceeded, for example, if the potential is raised above −50 mV. After releasing the pulse, the neuron returns to its resting potential. The action potential causes a release of certain biochemical agents known as *neurotransmitters* that are the means by which messages are transmitted to the dendrites of nearby neurons. These neural transmitters may have either an excitatory or inhibitory effect on neighboring neurons. A number of biochemical transmitters are known, including acetylcholine (usually excitatory), catecholamines, such as dopamine, norepinephrine, and epinephrine, and other amino acid derivatives such as histamine, serotonin, glycine, and γ-aminobutyric acid (GABA). GABA and glycine are two important inhibitory transmitters (Butter, 1968).

1.2.2 Structure of the Central Nervous System

The puzzle of how individual neurons are organized into complex neuronal structures has been the subject of a great deal of research over the years. Santiago Ramón de Cajal was the first to discover the complex interconnection structure in the cerebral cortex summarized in an English translation by DeFelipe and Jones (1988). Along with his associate Camillo Golgi (Golgi, 1886) he produced photographs of the structures by applying dyes that were absorbed differently. For this work, Cajal and Golgi were awarded the 1906 Nobel Prize in medicine.

Later, in the 1930s, Lorente de Nó, one of Cajal's students, examined the types of neurons in the cerebral cortex showing 32 to 34 different types based on shape classification, not on function (Asanuma and Wilson, 1979).

In the 1940s, Hodgkin and Huxley (Hodgkin, 1964; Huxley, 1971) began their well-known work on the giant squid, chosen because of its two very large neurons. Hodgkin and Huxley were awarded the 1963 Nobel Prize for their investigations into threshold, inhibition, and excitation in the giant squid axon.

Next, Hubel and Wiesel (1962) did extensive investigation into the cerebral cortex of the cat. They mapped many complex structures and tracked the path from the optic nerve to the lateral geniculate body to the visual cortex. They found columns of cells in the visual cortex that appeared to be responsible for processing various shapes. In the process, they distinguished between simple, complex, and hypercomplex cells. Their work also emphasized the parallel nature of the visual processing system. Figure 1.3 shows the optical pathways Hubel and Wiesel mapped out.

1.3 EARLY NEURAL MODELS

1.3.1 The McCulloch and Pitts Neuron

In a 1943 paper, McCulloch and Pitts (1943) presented a two-state logical decision element model based on a simplified neuron which they used to compute Boolean functions. They declared that "neural events and the relationship among them can be treated by means of propositional logic" (p. 115). Their artificial neuron performed logical operations on two or more inputs and produced an output if a threshold value was exceeded. This work can be considered the ancestor of artificial neural networks.

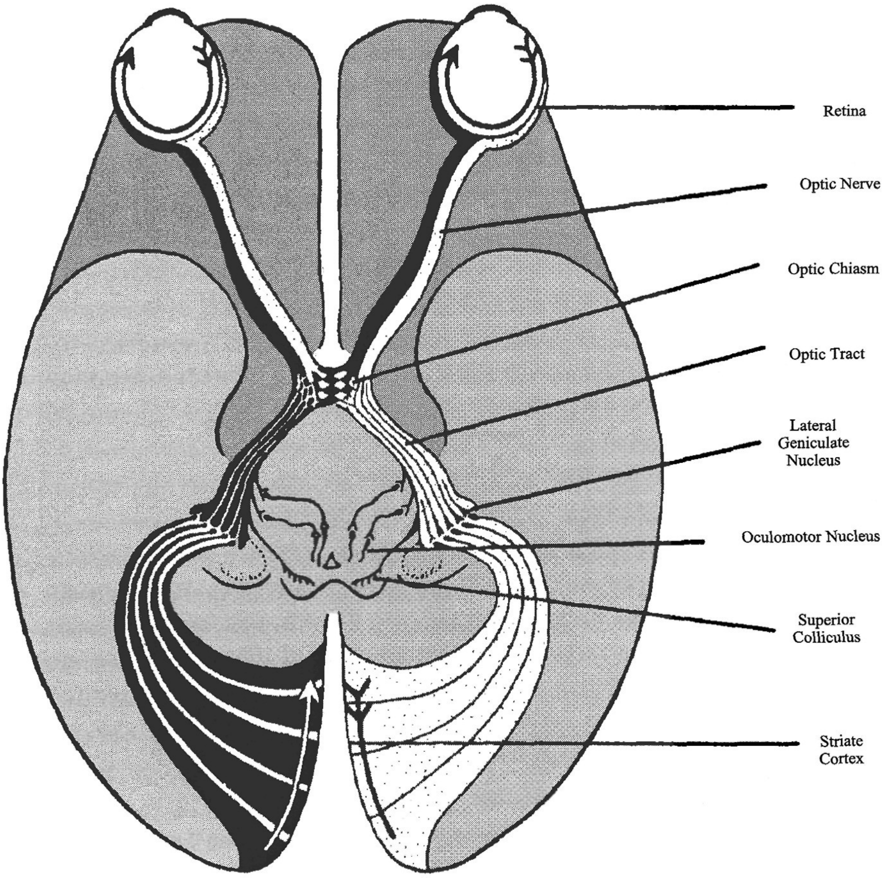

Figure 1.3 Optical Pathways.

1.3.2 Hebbian Learning

In 1949, Donald Hebb (1949) published his approach to learning laws. In his original approach, excitatory neuron coupling weights were increased by a subsequent firing, based on the idea of learning driven by activity. However, weights could only increase. (Many later models were based on this initial work and are discussed in detail in Chapter 5.)

1.3.3 ADALINE

ADALINE, an acronym for ADAptive LINear Element, was developed by Bernard Widrow (Widrow and Stearns, 1985). He used the mathematics of adaptive signal processing to produce the first commercial neural network.

1.3.4 Rosenblatt Perceptron

In the 1950s, Rosenblatt (1962) introduced models of the brain which he called perceptrons. Although his representation of artificial neurons was based on the neuron models of McCulloch and Pitts, he departed from their approach by basing his model on probability theory rather than symbolic logic. The photoperceptron as defined by Rosenblatt responded to optical patterns, and contained a sensory, an association, and a response area (Figure 1.4). The sensory area corresponds to the retinal structure. Each point responds to light in an on/off manner; input is then transmitted to the association area. The connections have three possible weights: 1 (excitatory), −1 (inhibitory), or 0. When a pattern is presented to the sensory area, a unit in the association area becomes active providing its value exceeds a predetermined threshold θ. At time t, the output from the association area is defined as

$$y(t) = \operatorname{sgn} \Sigma \left[x_i(t)\, w_i(t) - \theta\right] \tag{1.1}$$

where sgn is either +1 (for positive argument) or −1 (for negative argument), $x_i(t)$ is the ith input signal, and $w_i(t)$ is the weight of the ith input to the node.

The basic perceptron model was an example of a learning algorithm. Nilsson (1965) summarizes these early learning systems.

1.3.5 Problems with Early Systems

Neural network research experienced a general setback following the publication of a paper by Minsky and Pappert (1969) proving that a single-layer perceptron could not solve the exclusive or (XOR) problem. In fact, single-layer perceptrons can only separate categories that are linearly separable, that is, separable by a hyperplane (in two dimensions, a line). Figure 1.5 shows the XOR problem; c_0 is the category in which the polarity of the features is the same, which should have an output of 0 for the XOR, and c_1 is the category in which the polarity differs, which should have an output of 1 for the XOR. There is no line that can separate these categories. Unfortunately, even though Rosenblatt had proposed the use of multilayer networks to overcome this problem, these criticisms stymied neural network research for well over a decade. The limitation of the current computers in terms of both memory and speed was one reason for the loss of interest in the early neural network research. The problems addressed as examples in the neural network models were fairly simple, with few nodes. The training often took hours to accomplish. Many justifiably felt that these time and

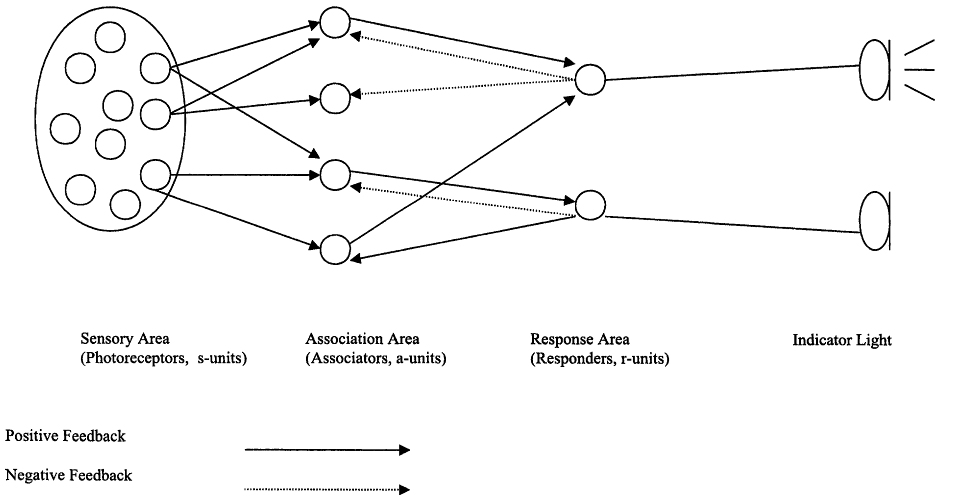

Figure 1.4 Diagram of Simple Photoperceptron.

XOR Truth Table

Input 1	Input 2	Output
0	0	0
0	1	1
1	0	1
1	1	0

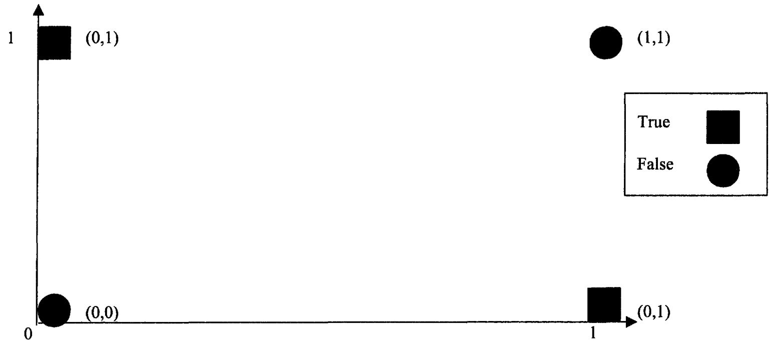

Figure 1.5 The Exclusive OR Problem (XOR).

memory considerations made it difficult to tackle practical problems. With the advent of faster and faster hardware with large, inexpensive memory, these worries ceased to be considerations in the new generation of neural network models.

1.4 PRECURSOR TO CURRENT MODELS: PATTERN CLASSIFICATION

Pattern classification (sometimes called pattern recognition) was one of the first methods applied to medical applications and has found applications in diverse areas from electrocardiograms to genetic sorting. (For an historical perspective of pattern recognition, see Chapter 9.)

What is a pattern recognition problem? As an example, consider a group of patients who have come to the emergency room with chest pain. Subsequently, some of these patients are found to have had a myocardial infarction (MI), and others are found to have had angina. The first objective of a pattern classification system is to determine which parameters enabled the medical staff to distinguish between these two diagnoses. This is a two-category problem. The initial phase consists of feature extraction. *Features* are properties of items to be classified that will aid in discriminating between classes.

1.4.1 Feature Extraction

Determining features is the most crucial step in designing a pattern recognition decision aid. In the emergency room example given earlier, we must identify parameters useful in distinguishing between the two classes. Identification of possible features requires domain knowledge or access to domain knowledge relevant to the application. As a simple illustration, suppose we know that patients with MIs in general have low blood pressure, whereas those with angina in general have elevated blood pressure. If we plot the histograms for blood pressure for all patients with either disease, we may get a plot similar to that shown in Figure 1.6. Note the area of overlap between the two groups, so that the groups cannot be completely separated by this one variable. In addition, we know that patients with MIs may have elevated white blood counts, whereas patients with angina have normal white blood counts. If we consider only these two parameters, or features, we have a two-variable problem. We combine these features into a two-dimensional feature vector $\mathbf{x} = (x_1, x_2)$, where x_1 = systolic blood pressure (BP) and x_2 = white blood count (WBC). For the sake of this example, we will consider only systolic blood pressure. In this simple case we can plot x_1 versus x_2. Figure 1.7 shows a sample plot of five cases in each category. The squares represent cases with MI, and the circles represent cases with angina.

The second objective of a pattern classification system is to find a separator that will divide these two classes by placing as many samples into the correct category as possible. The dashed line in Figure 1.7 shows a possible separator with one misclassification. Additional features may result in better classification or a more robust model. The following considerations should be kept in mind:

1. Look for a classification that minimizes error.
   Ideal: all cases classified correctly; if not possible, minimize either the number of errors or the cost of errors.
2. More features may be needed.
   For three features, Figure 1.6 becomes 3-D, for four or more, no picture!

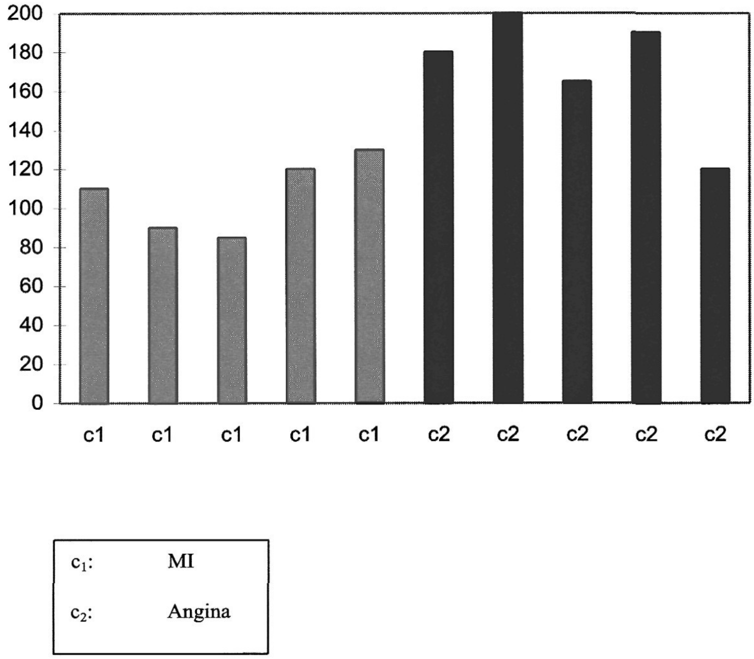

Figure 1.6 Histograms of Systolic Blood Pressures for Myocardial Infarction (MI) and Angina.

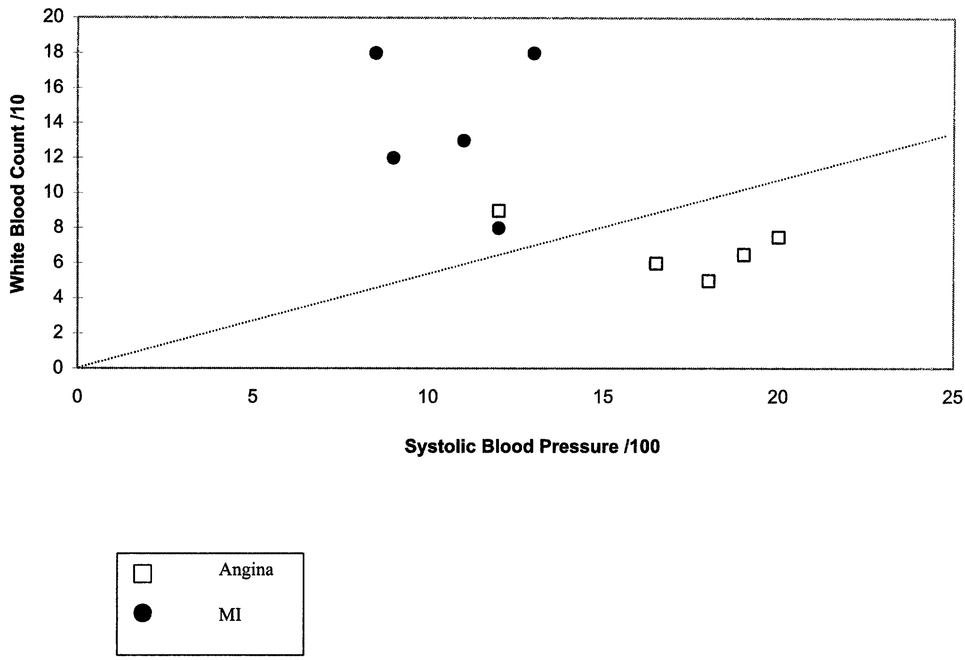

Figure 1.7 Plot of White Blood Count versus Systolic Blood Pressure.

3. More classes may be relevant.
   For example, MI, angina, and congestive heart failure.

The final objective of pattern classification is to use the separator to classify new cases. In this way, the pattern recognition system is used as a decision aid.

1.4.2 Supervised Learning

The preceding classification is an example of supervised learning: data of known classification are used to determine important parameters (components of the feature vector) that contribute to the correct decision. To use supervised learning, a training set must be available for development of the separating vector. A test set is then used to determine the accuracy of the separator. Ideally, the training set and test set should be disjoint.

The question that remains is, How can the separating vector be obtained? In our simple example, we did it geometrically; for data of higher dimensionality, this will not be possible. The separator is determined through a learning algorithm that is the heart of the method. (Learning algorithms will be discussed shortly and in detail in Chapter 6.)

1.4.3 Unsupervised Learning

Unsupervised learning is a much more difficult problem. In this case, data of unknown classification are used. The objective is to try to find patterns in the data that will allow the data to be grouped or clustered according to similar characteristics with the characteristics defined in the feature vector. The main method for accomplishing unsupervised learning is clustering, with a number of variations. (Clustering will be discussed in detail in Chapter 5. Recent approaches also include data mining and genetic algorithms, discussed in Chapter 14.)

1.4.4 Learning Algorithms

The purpose of a learning algorithm is to determine which features are important for a particular decision as well as their relative importance. In most pattern classification systems, a feature vector is defined as

$$\mathbf{x} = (x_1, x_2, \ldots, x_n,) \tag{1.2}$$

where each x_i is a feature and n is the dimensionality of the vector. In classification programs, the objective in the most straightforward two-class problem is to obtain a decision surface that can separate the data. The two-variable equivalent to this is shown in Figure 1.7. For the n-dimensional case, we want the following to hold:

$$D(\mathbf{x}) > 0 \Rightarrow \mathbf{x} \text{ belongs in class 1}$$
$$D(\mathbf{x}) < 0 \Rightarrow \mathbf{x} \text{ belongs in class 2}$$
$$(D(\mathbf{x}) = 0 \text{ is indeterminate})$$

where

$$D(\mathbf{x}) = \sum_{i=1}^{n} = w_i x_i \tag{1.3}$$

or in vector format

$$D(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} \tag{1.4}$$

In order to find the value for $D(\mathbf{x})$, the values for the two vectors **w** and **x** must be known. The values for **x** are obtained from the data. It is the job of the learning algorithm to determine the values for **w**. In supervised learning, an additional important piece of information is available: for each **x**, the class to which it belongs is known.

A general algorithm for supervised learning follows:

```
Make an initial guess for each component of $\mathbf{w}$.
Select a training set of data.
For each vector in the training set:
   Compute $D(\mathbf{x})$
   If $D(\mathbf{x}) > 0$ and $\mathbf{x}$ ε class 1 or $D(\mathbf{x}) < 0$ and $\mathbf{x}$ ε class 2, do not adjust $\mathbf{w}$
   If $D(\mathbf{x}) > 0$ and $\mathbf{x}$ ε class 2 adjust $\mathbf{w}$ according to rule 1
   If $D(\mathbf{x}) < 0$ and $\mathbf{x}$ ε class 1 adjust $\mathbf{w}$ according to rule 2
Until $\mathbf{w}$ does not change (or until criterion function is minimized).
```

Basically, learning algorithms differ in the definition of rules 1 and 2 in the preceding algorithm and in the determination of the criterion function that determines when the iterative weight adjustment should stop. A number of approaches have been used, including Bayes learning (Chapter 15), perceptrons (Chapter 4), potential functions (Chapter 4), and backpropagation (Chapter 4).

The simple algorithm given above is complicated in practice by a number of factors. The most obvious problem is what to do if **w** does not cease to change, which will happen when it is not possible to correctly classify all samples in the training set. If all samples can be correctly classified, the set is said to be *linearly separable*. If not, the algorithm must terminate on some other condition, which will hopefully ensure that as many samples as possible are classified correctly. This is handled by defining what is known as a *criterion function*. These functions are defined differently depending on the approach taken and will be discussed in detail later in this book.

As an example, consider our two-dimensional problem given earlier. This is a two-category problem. We will consider the presence of MI to be class 1 and the presence of angina to be class 2. Our problem is then defined by the following components:

$$D(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} = w_1 x_2 + w_2 x_2 \tag{1.5}$$

where

x_1: systolic blood pressure
x_2: white blood count

If $D(\mathbf{x}) > 0$, then we will assume that $\mathbf{x}$ belongs to class 1 (MI); if $D(\mathbf{x}) < 0$, we will assume that $\mathbf{x}$ belongs to class 2 (angina); if $D(\mathbf{x}) = 0$, then we can make no determination.

For the purpose of illustration, we will use the perceptron learning rule, defined as

$$w_i(t + 1) = w_i(t) + \eta[d(t) - y(t)]x_i(t) \tag{1.6}$$

that computes each weight adjustment. The iteration is represented by t, and η is the learning rate, which we will set to 0.01. We define $y(t)$ and $d(t)$ as follows:

$$\begin{aligned} y(t) &= 1 \text{ if } D(\mathbf{x}) > 0 \\ y(t) &= -1 \text{ if } D(\mathbf{x}) < 0 \\ d(t) &= 1 \text{ if vector belongs to class 1} \\ d(t) &= -1 \text{ if vector belongs to class 2} \end{aligned}$$

Table 1.1 contains values for our ten feature vectors. To make our calculations simpler, we can scale the data so that both values are of similar magnitudes. We will divide all WBC values by 1000 and all blood pressure values by 10. We will select the first two vectors of each class, alternating classes, for inclusion in the training set:

$\mathbf{t}_1 = (11.0, 13.0)$ (vector $\mathbf{x}_1$, class 1)
$\mathbf{t}_2 = (18.0, 5.0)$ (vector $\mathbf{x}_6$, class 2)

TABLE 1.1 Feature Vector Values for Differentiation between Myocardial Infarction (MI) and Angina

Feature Vector	Diagnosis	Systolic Blood Pressure	White Blood Count
$\mathbf{x}_1$	MI	110	13,000
$\mathbf{x}_2$	MI	90	12,000
$\mathbf{x}_3$	MI	85	18,000
$\mathbf{x}_4$	MI	120	8,000
$\mathbf{x}_5$	MI	130	18,000
$\mathbf{x}_6$	Angina	180	5,000
$\mathbf{x}_7$	Angina	200	7,500
$\mathbf{x}_8$	Angina	165	6,000
$\mathbf{x}_9$	Angina	190	6,500
$\mathbf{x}_{10}$	Angina	120	9,000

$$\mathbf{t}_3 = (9.0, 12.0) \qquad \text{(vector } \mathbf{x}_2\text{, class 1)}$$
$$\mathbf{t}_4 = (20.0, 7.5) \qquad \text{(vector } \mathbf{x}_7\text{, class 2)}$$

We will make an initial guess for each weight as $w_1 = -0.3$, $w_2 = 1.0$. Initially, we substitute vector $\mathbf{t}_1$ into Eq. (1.5):

$$D(\mathbf{t}_1) = -0.3\ (11.0)\ +1.0(13) > 0; \text{therefore } y(t) = 1$$
$$\mathbf{t}_1 \text{ belongs to class 1; therefore } d(t) = 1$$

Substituting into Eq. (1.6), we see that as the classification is correct, no weight adjustment is made. We then proceed with the second vector substitution, which also results in no weight adjustment as does the third. For the fourth vector

$$D(\mathbf{t}_4) = -0.3(20.0) + 1.0(7.5) > 0, y(t) = 1$$
$$\mathbf{t}_4 \text{ belongs to class 2}$$

Therefore, substituting into Eq. (1.6)

$$w_1(1) = -0.3 + 0.01[(-1 - (1)]\ 20.0 = -0.7$$
$$w_2(1) = 1.0 + 0.01[-1 - (1)]7.5 = 0.85$$

The process must then begin again with $\mathbf{t}_1$ and continue until all vectors are classified correctly. After completion of this process, the resulting weights are:

$$w_1 = -0.7$$
$$w_2 = 0.85$$

Our decision surface is

$$D(\mathbf{x}) = -0.7x_1 + 0.85x_2 \tag{1.7}$$

The remainder of the vectors in Table 1.1 will be our test set, which will be used to determine how well our decision surface works. For example, substituting vector $\mathbf{x}_3$ from Table 1.1 in Eq. (1.5):

$D(\mathbf{x}_3) = -0.7(8.5) + 0.85{*}(18) > 0$, which is correct since vector $\mathbf{x}_3$ belongs to class 1.

1.5 RESURGENCE OF THE NEURAL NETWORK APPROACH

Neural networks have found a wide range of applications in the last decade (Carpenter and Grossberg, 1988; Sabbatini, 1992; *Computer Magazine,* 1988) and in many cases have replaced knowledge-based approaches that became popular in the 1970s (Davis and Lenat, 1982; Barr and Feigenbaum, 1982). Neural networks permit rapid development of a model through the learning algorithm if sufficient data are available.

Resurgence of the neural network approach began in the late 1970s and early 1980s with the work of Kohonen, Hopfield, Grossberg, and Rummelhart. In the 1970s, Grossberg (1988) developed the adaptive resonance theory (ART) and theories about the functioning of biological nervous systems that Carpenter and Grossberg (1988) later developed into self-organizing neural network architectures. Kohonen (1984) also did pioneering work on self-organizing networks. In the early 1980s, Hopfield and others introduced new approaches based on the early work of Hebb (1949). Rummelhart and his group (Rummelhart and McClelland, 1986) developed the backpropagation method, which became one of the most widely used approaches in neural network design. Hypernet, developed by Cohen and Hudson in the early 1980s (Cohen, Hudson, and Anderson, 1989), extended the potential function approach and in the process introduced the single and multidimensional Cohen orthogonal functions that encompassed the possibility of fractional contribution of nodes. The new approaches developed by these researchers, as well as others, overcame the limitations of the early neural network approaches. These methods, together with the advances made in computer architecture providing faster processing and cheaper memory, made the neural network concept practical. (In Chapters 2 through 5 we will examine in detail the new neural network structures that began in the 1980s, along with biomedical applications for each method.)

1.6 BASIC CONCEPTS

1.6.1 Artificial Neurons

One of the basic ideas behind neural networks is to construct artificial neurons that have the characteristics of actual neurons. Artificial neurons, or nodes as they are often called, receive input from multiple other nodes. These multiple inputs can be considered as dendrites in the biological neuron. Like neurons, the nodes produce one output that can be associated with the axon. In computing the output, the input information is weighted, either positively or negatively. These weights are analogous to the excitatory and inhibitory action of the chemical transmitters in the actual neuron. In neurons, an output results only if a certain threshold voltage is exceeded. This action is sometimes simulated by use of threshold values in the node, although not all models use the threshold approach.

1.6.2 Selection of Input Nodes

In the initial design of a neural network, the number and type of input nodes must be determined. These decisions are based on the nature of the problem. As we will see in the next chapter, nodes may be binary, representing only an on or an off state, or they may accept continuous values. The input nodes must be able to represent all relevant information that is pertinent to the problem. The process of defining input nodes is connected with feature selection in which salient features of the problem under consideration are analyzed. This process is discussed in Chapter 3.

1.6.3 Network Structure

The early neural networks were only two-layer structures. As discussed earlier, this construction greatly limited their usefulness in that only linear problems could be represented. In the second generation of neural networks, new structures were developed which consisted of three or more layers. The most common structure is the three-layer network as illustrated in Figure 1.8. These three layers consist of the input layer, the hidden or interactive layer, and the output layer. Many other network configurations have been used, but in general the three-layer network is capable of addressing all problems which the more complex structures address. The manner in which nodes are connected is different depending on the approach and will be described in detail in later chapters when each method is discussed.

1.6.3.1 Feed-Forward Networks. The methods described in Section 1.4 apply to feed-forward networks. These networks compute weights that are used to determine output from a node that is subsequently fed to the next layer. In the detailed example

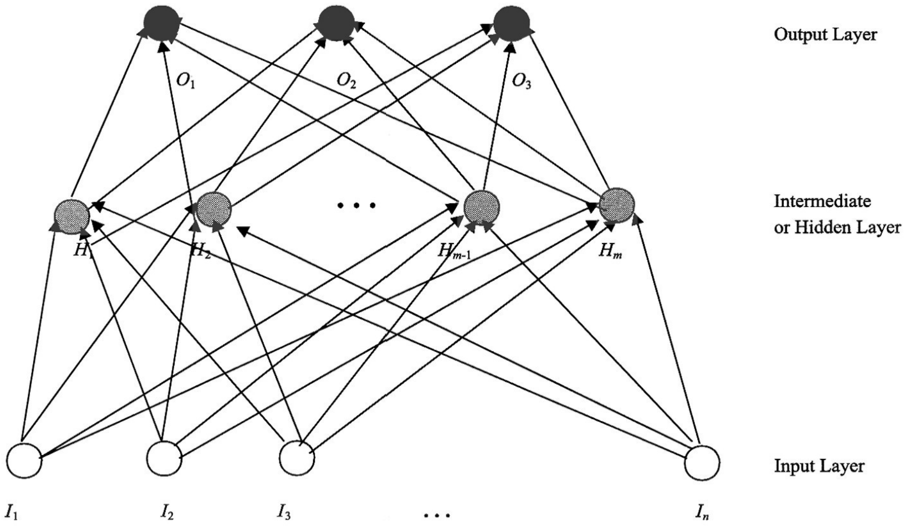

Figure 1.8 Three-Layer Neural Network Structure.

given earlier, the weights determined the impact that the input nodes have on the output, but no information is fed back to the input nodes.

1.6.3.2 Feed-Backward Networks. The revival of neural networks began in the early 1980s with the work of Hopfield (1982). The Hopfield model was completely different from earlier approaches in that the neurons, or nodes, had two-way connections. Instead of adjusting weights to tune the output of nodes, the network stored patterns that were later used to process unknown input vectors. (The Hopfield net and other feed-backward approaches will be described in detail in Chapter 2.)

1.6.4 Learning Mechanism

We saw an example of a learning algorithm in Section 1.4, with a specific learning rule given by the perceptron learning rule. As we will learn in subsequent chapters, many different learning mechanisms have been tried in neural networks. All have advantages and disadvantages. Some offer strong mathematical foundations, whereas others are more ad hoc. The learning mechanism affects the speed of convergence of the network, and indeed determines whether or not it converges at all. It can also affect the accuracy of the model in classification of unknown cases.

1.6.5 Output

Many neural networks have only one output node. This is not the only possible structure. As we will see in subsequent chapters, it is possible to have multiple output nodes and even output nodes that feed into other types of decision-making strategies, such as symbolic reasoning.

1.7 SUMMARY

In this chapter we have reviewed some of the components of biological nervous systems that are important contributors to the foundations of artificial neural networks. In addition to these biological precursors, the most important technical precursor to neural networks, pattern classification, which was used successfully for many years in design of medical decision-making aids, was summarized. In the subsequent chapters of Part I, we review pattern classification in more depth, along with different types of neural networks and corresponding learning algorithms as well as their uses in biomedical problem solving.

EXERCISES

1. What is the main reason that the neural network approach introduced in the late 1950s was abandoned for over twenty years?
2. In what ways do neural network models correspond to biological nervous systems? Can you list aspects of biological nervous systems that have not been incorporated into neural networks?
3. Explain why the two-layer neural networks of the 1950s and 1960s could not solve the exclusive OR problem.
4. In the example based on Table 1.1, we computed the weighting factors for the first four passes. Complete this calculation, stopping when all four vectors in the training set have been classified correctly. Check to make sure that your weights agree with those given in the text.
5. Substitute the remainder of the vectors in Table 1.1 into Eq. (1.5). How many of them are correctly classified? Does this correspond to the geometrical results in Figure 1.7?
6. Repeat exercise 4, but change the order of the vectors in your training set to $\mathbf{x}_6, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_7$. Do you get the same values for w_1 and w_2?
7. If you add a third variable, the linear separator is no longer a line. What is it? What happens for four or more variables? Can the same approach be utilized?
8. Devise an alternative strategy for determining a decision surface if the two groups are not linearly separable.
9. What happens if the classification problem has more than two classes? For example, assume the three possibilities are angina, MI, and congestive heart failure. Is it possible to use a perceptron-type model to solve this problem?
10. Formulate mathematically the perceptron approach for four variables: white blood count, systolic blood pressure, diastolic blood pressure, and pH of the blood.

REFERENCES

Aakerlund, L., and Hemmingsen, R. 1998. Neural networks as models of psychopathology. *Biological Psychiatry* **43(7):** 471–482.

Asanuma, H., and Wilson, V.J. (eds.). 1979. Integration in the nervous system: a symposium in honor of David P.C. Lloyd and Rafael Lorente de No, The Rockefeller University, Tokyo.

Barr, A., and Feigenbaum, E. 1982. *The Handbook of Artificial Intelligence,* vol. 2. Reading, MA: Addison-Wesley.

Butter, C.M. 1968. *Neuropsychology: The Study of Brain and Behavior.* Belmont, CA: Brooks/Cole Publishing Company.

Carpenter, G., and Grossberg, S. 1988. The art of adaptive pattern recognition by a self-organizing network. *Computer* **21:** 152–169.

Cohen, M.E., Hudson, D.L., and Anderson, M.F. 1989. A neural network learning algorithm with medical applications. *SCAMC* **13:** 307–311.

Computer Magazine. 1988. IEEE **21(3).**

Davis, R., and Lenat, D.B. 1982. *Knowledge-Based Systems in Artificial Intelligence.* New York: McGraw-Hill.

DeFelipe, J., and Jones, E.G. (eds.). 1988. *Cajal on the Cerebral Cortex: An Annotated Translation of the Complete Writings.* New York: Oxford University Press.

Golgi, C. 1886. *Sulla fina anatomia degli organi centrali del sistema nervoso.* Milan, Italy: Hoepli.

Grossberg, S. 1988. *Neural Networks and Natural Intelligence.* Cambridge, MA: MIT Press.

Harley, T.A. 1998. Connectionist modeling of the recovery of language functions following brain damage. *Brain and Language* **52(1):** 7–24.

Hebb, D.O. 1949. *The Organization of Behavior.* New York: John Wiley & Sons.

Hodgkin, A.L. 1964. *The Conduction of the Nervous Impulse.* Liverpool, England: Liverpool University Press.

Hopfield, J.J. 1982. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA* **79:** 2554–2558.

Hubel, D.H., and Wiesel, T.N. 1962. Receptive fields, binocular interaction, and functional architecture of the cat visual cortex. *J. Physiology* **160(1):** 106–154.

Hudson, D.L., Cohen, M.E., and Anderson, M.F. 1991. Use of neural network techniques in a medical expert system. *International Journal of Intelligent Systems* **6(2):** 213–223.

Huxley, A.F., and Simmons, R.M. 1971. Proposed mechanism of force generation in striated muscle. *Nature* **233(5321):** 533–538.

Kohonen, T. 1984. *Self-Organization and Associative Memory.* Berlin: Springer-Verlag.

MacGregor, R.J. 1987. *Neural and Brain Modeling.* San Diego: Academic Press.

McCulloch, W.S., and Pitts, W. 1943. A logical calculus of the ideas immanent in neural nets. *Bull. Math. Biophys.* **5:** 115–137.

Minsky, M., and Papert, S. 1969. *Perceptrons: An Introduction to Computational Geometry.* Cambridge, MA: MIT Press.

Nilsson, N.J. 1965. *Learning Machines.* New York: McGraw-Hill.

Rialle, V., and Stip, E. 1994. Cognitive models in psychiatry: from symbolic models to parallel and distributed models (French). *J. of Psychiatry and Neuroscience* **19(3):** 178–192.

Rogers, S.K., and Kabrisky, M. 1991. *An Introduction to Biological and Artificial Neural Networks for Pattern Recognition.* Bellingham, WA: SPIE Optical Engineering Press.

Rosenblatt, F. 1962. *Principles of Neurodynamics.* New York: Spartan Books.

Rummelhart, D.E., and McClelland, J.L. 1986. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition.* Cambridge, MA: MIT Press.

Ruppin, E., Reggia, J.A., and Horn, D. 1996. Pathogenesis of schizophrenic delusions and hallucinations: a neural network model. *Schizophrenia Bulletin* **22(1):** 105–123.

Sabbatini, R.M.E. 1992. Applications of connectionist systems in biomedicine. *MEDINFO*, pp. 418–425. New York: Elsevier.

Saugstad, L.F. 1994. Deviation in cerebral excitability: possible clinical implications. *Int. J. of Psychophysiology* **18(3):** 205–212.

Widrow, B., and Stearns, S.D. 1985. *Adaptive Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall.