CHAPTER 9

An Introduction to Linear Algebra in Parallel Distributed Processing

M. I. JORDAN

Many of the properties of the models described in this book are captured by the mathematics of linear algebra. This chapter serves as a introduction to linear algebra and is a good starting place for the reader who wishes to delve further into the models presented in other parts of the book. I will focus on the aspects of linear algebra most essential for the analysis of parallel distributed processing models, particularly the notions of a vector space, the inner product, and linearity. I will also discuss some simple PDP models, and show how their workings correspond to operations on vectors.

VECTORS

A vector is a useful way to describe a pattern of numbers. Consider for example the pattern of numbers that describe the age, height, and weight of an average person. Suppose that Joe is 37 years old, 72 inches tall, and weighs 175 pounds. This information can be summarized in a vector or ordered list of numbers. For each person, there is a corresponding vector, as in Figure 1A. Each vector has three components: age, height, and weight. There is no reason to limit ourselves

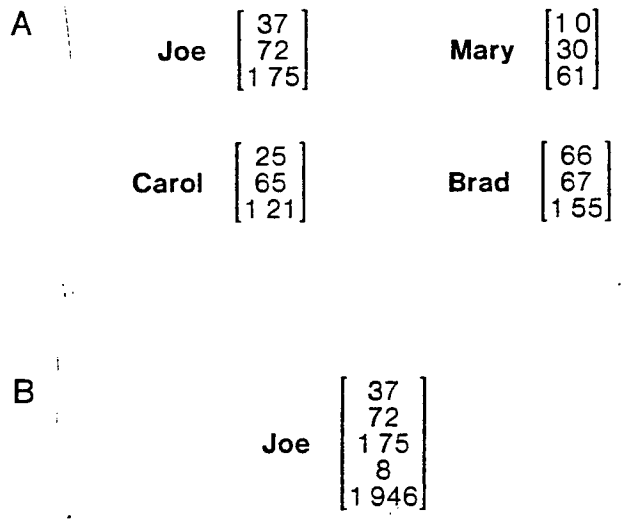

FIGURE 1.

to only three components, however. If, for example, we also wanted to keep track of Joe's shoe size and year of birth, then we would simply make a vector with five components, as in Figure 1B.

One important reason for the great utility of linear algebra lies in the simplicity of its notation. We will use bold, lower-case letters such as **v** to stand for vectors. With this notation, an arbitrarily long list of information can be designated by a single symbol.

When a vector has no more than three components, it can be represented graphically by a point or an arrow in three-dimensional space. An example with three components is given in Figure 2 for the vector corresponding to Mary. Each axis in the figure corresponds to one of the three components of the vector.

It will prove helpful to try and visualize vectors as points or arrows in two- and three-dimensional space in proceeding through this chapter in order to develop geometric intuition for the operations on vectors. Notice, however, that there is no fundamental distinction between such vectors and vectors with more than three components. All of the operations upon vectors described in later sections apply equally well to vectors with any finite number of components.

In a parallel distributed processing model, many quantities are best represented by vectors. The pattern of numbers representing the activations of many processing units is one example. Other examples are the set of weights on the input lines to a particular processing unit, or the set of inputs to a system.

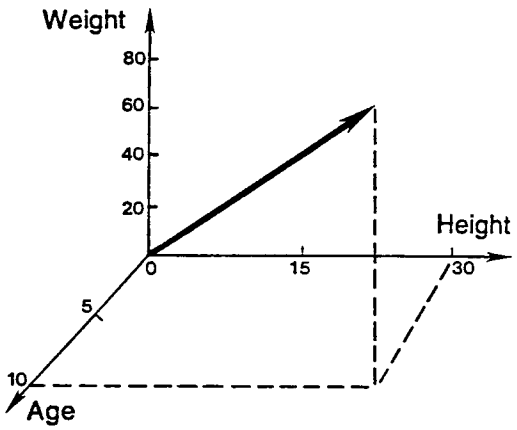

FIGURE 2.

BASIC OPERATIONS

Multiplication by Scalars

In linear algebra, a single real number is referred to as a *scalar*. A vector can be multiplied by a scalar by multiplying every component of the vector by the scalar.

Examples:

$$2\begin{bmatrix}2\\1\end{bmatrix}=\begin{bmatrix}4\\2\end{bmatrix} \qquad 5\begin{bmatrix}-3\\4\\1\end{bmatrix}=\begin{bmatrix}-15\\20\\5\end{bmatrix}$$

Geometrically, scalar multiplication corresponds to lengthening or shortening the vector, while leaving it pointing in the same or opposite direction. As can be seen in Figure 3, multiplying a vector by 2 leaves it pointing in the same direction but twice as long. In general, multiplying a vector by a positive scalar produces a new vector that is longer or shorter by an amount corresponding to the magnitude of the scalar. Multiplication by a negative scalar produces a vector pointing in the opposite direction. It, too, is longer or shorter depending on the magnitude of the scalar. Two vectors that are scalar multiples of one another are said to be *collinear*.

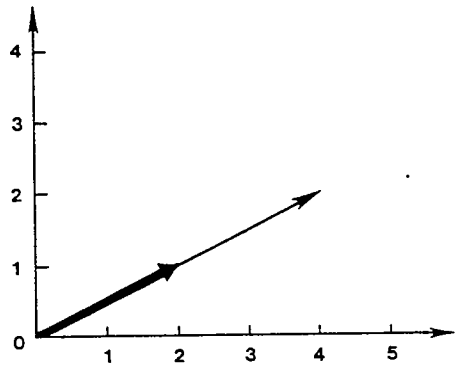

FIGURE 3.

Addition of Vectors

Two or more vectors can be added by adding their components. The vectors must have the same number of components to be added; otherwise the operation is undefined.

Examples:

$$\begin{bmatrix}1\\2\\1\end{bmatrix}+\begin{bmatrix}2\\1\\3\end{bmatrix}=\begin{bmatrix}3\\3\\4\end{bmatrix} \qquad \begin{bmatrix}2\\1\end{bmatrix}+\begin{bmatrix}0\\1\end{bmatrix}+\begin{bmatrix}3\\-3\end{bmatrix}=\begin{bmatrix}5\\-1\end{bmatrix}$$

Vector addition is associative (the vectors can be grouped in any manner) and commutative (the order of addition is unimportant) just like addition in ordinary algebra. This is true because if we consider one component at a time, vector addition *is* just addition in ordinary algebra.

How can vector addition be represented graphically? Consider Figure 4, where the vectors $\mathbf{v}_1 = \begin{bmatrix}1\\2\end{bmatrix}$ and $\mathbf{v}_2 = \begin{bmatrix}3\\1\end{bmatrix}$ are being added. It can be seen that the sum $\mathbf{v}_1 + \mathbf{v}_2$ is a vector $\begin{bmatrix}4\\3\end{bmatrix}$ which lies between $\mathbf{v}_1$ and $\mathbf{v}_2$. Forming the parallelogram with sides $\mathbf{v}_1$ and $\mathbf{v}_2$, we see that the sum of

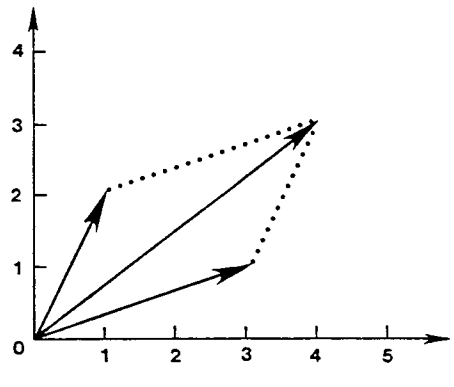

FIGURE 4.

the two vectors is the diagonal of this parallelogram. In two and three dimensions this is easy to visualize, but not when the vectors have more than three components. Nevertheless, it will be useful to imagine vector addition as forming the diagonal of a parallelogram. One implication of this view, which we will find useful, is that the sum of two vectors is a vector that lies in the same plane as the vectors being added.

Example: Calculating averages. We can demonstrate the use of the two operations thus far defined in calculating the average vector. Suppose we want to find the average age, height, and weight of the four individuals in Figure 1A. Clearly this involves summing the components separately and then dividing each sum by 4. Using vectors, this corresponds to adding the four vectors and then multiplying the resulting sum by the scalar 1/4. Using $\mathbf{u}$ to denote the average vector,

$$\mathbf{u} = \frac{1}{4}\left\{\begin{bmatrix} 37 \\ 72 \\ 175 \end{bmatrix} + \begin{bmatrix} 10 \\ 30 \\ 61 \end{bmatrix} + \begin{bmatrix} 25 \\ 65 \\ 121 \end{bmatrix} + \begin{bmatrix} 66 \\ 67 \\ 155 \end{bmatrix}\right\} = \begin{bmatrix} 34.5 \\ 58.5 \\ 128 \end{bmatrix}.$$

Using vector notation, if we denote the four vectors by $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$, and $\mathbf{v}_4$, then we can write the averaging operation as

$$\mathbf{u} = \frac{1}{4}\,(\mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_3 + \mathbf{v}_4).$$

The vector $\mathbf{u}$, then, is a vector whose components are the averages of the components of the four individual vectors. Notice that the same result is obtained if each vector is first multiplied by 1/4, and the resulting vectors are added. This shows that multiplication by scalars and vector addition obey a distributive law, as in ordinary algebra.

LINEAR COMBINATIONS AND LINEAR INDEPENDENCE

Linear Combinations of Vectors

The average vector calculated in the last section is an example of a *linear combination* of vectors. In this section, we pursue this idea further.

Consider the vectors $\mathbf{v}_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$, $\mathbf{v}_2 = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$, and $\mathbf{u} = \begin{bmatrix} 9 \\ 10 \end{bmatrix}$. Can $\mathbf{u}$ be written as the sum of scalar multiples of $\mathbf{v}_1$ and $\mathbf{v}_2$? That is, can scalars c_1 and c_2 be found such that $\mathbf{u}$ can be written in the form

$$\mathbf{u} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 \, ?$$

If so, then $\mathbf{u}$ is said to be a linear combination of the vectors $\mathbf{v}_1$ and $\mathbf{v}_2$. The reader can verify that $c_1 = 3$ and $c_2 = 2$ will work, and thus $\mathbf{u}$ is a linear combination of $\mathbf{v}_1$ and $\mathbf{v}_2$.

This can also be seen directly in Figure 5, where these vectors are plotted. Remembering that multiplication by a scalar shortens or

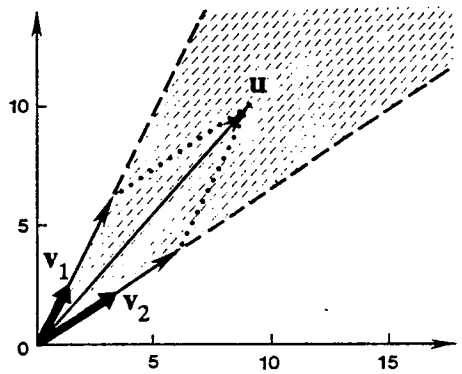

FIGURE 5.

lengthens a vector and that vector addition corresponds to forming the diagonal of a parallelogram, it seems clear that we can find scalars to adjust $\mathbf{v}_1$ and $\mathbf{v}_2$ to form a parallelogram that yields $\mathbf{u}$. This is indicated in the figure. It also seems clear that, using positive scalars, any vector in the shaded area of the figure can be generated this way. By using both negative and positive scalars, any vector in the plane can be written as a linear combination of $\mathbf{v}_1$ and $\mathbf{v}_2$. This is true because multiplication by a negative scalar reverses the direction of a vector as well as shortening or lengthening it. The vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ are said to *span* the plane, because any vector in the plane can be generated from these two vectors.

In general, given a set $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ of vectors, a vector $\mathbf{v}$ is said to be a linear combination of the $\mathbf{v}_i$ if scalars $c_1, c_2, \ldots, c_n$ can be found such that

$$\mathbf{v} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_n\mathbf{v}_n. \tag{1}$$

The set of *all* linear combinations of the $\mathbf{v}_i$ is called the set *spanned* by the $\mathbf{v}_i$.

Example. The three vectors $\begin{bmatrix}1\\0\\0\end{bmatrix}$, $\begin{bmatrix}0\\1\\0\end{bmatrix}$ and $\begin{bmatrix}0\\0\\1\end{bmatrix}$ span all of three-dimensional space since any vector $\mathbf{v} = \begin{bmatrix}a\\b\\c\end{bmatrix}$ can be written as a linear combination $\mathbf{v} = a\begin{bmatrix}1\\0\\0\end{bmatrix} + b\begin{bmatrix}0\\1\\0\end{bmatrix} + c\begin{bmatrix}0\\0\\1\end{bmatrix}$. The vectors are referred to as the standard basis for three-dimensional space (more on the idea of a basis in the next section).

Linear Independence

To say that a set of vectors span a space is to say that all vectors in the space can be generated from the original set by linear combination. We have shown examples in which two vectors span two-dimensional space and three vectors span three-dimensional space. We might be led to expect that, in general, n vectors suffice to span n-dimensional space. In fact, we have been using the term "dimension" without defining what it means; it would seem that a good definition of n-dimensional space is the set of vectors spanned by n vectors.

To make this definition work, we would require that the same size space be generated by any set of n vectors. However, this is not the case, as can be easily shown. Consider any pair of collinear vectors, for example. Such vectors lie along a single line, thus any linear combination of the vectors will lie along the same line. The space spanned by these two vectors is therefore only a one-dimensional set. The collinear vectors $\begin{bmatrix}1\\1\end{bmatrix}$ and $\begin{bmatrix}2\\2\end{bmatrix}$ are a good example. Any linear combination of these vectors will have equal components, thus they do not span the plane.

Another example is a set of three vectors that lie on a plane in three-dimensional space. Any parallelograms that we form will be in the same plane, thus all linear combinations will remain in the plane and we can't span all of three-dimensional space.

The general rule arising from these examples is that of a set of n vectors, if at least one can be written as a linear combination of the others, then the vectors span something less than a full n-dimensional space. We call such a set of vectors *linearly dependent.* If, on the other hand, none of the vectors can be written as a linear combination of the others, then the set is called *linearly independent.* We now revise the definition of dimensionality as follows: n-dimensional space is the set of vectors spanned by a set of n linearly independent vectors. The n vectors are referred to as a *basis* for the space.

Examples:

1. $\begin{bmatrix}1\\1\end{bmatrix}$ and $\begin{bmatrix}2\\2\end{bmatrix}$ are linearly dependent. They span only a one-dimensional space.

2. $\begin{bmatrix}1\\1\end{bmatrix}$ and $\begin{bmatrix}2\\1\end{bmatrix}$ are linearly independent. Thus they span the plane, a two-dimensional space.

3. $\begin{bmatrix}1\\1\end{bmatrix}$, $\begin{bmatrix}2\\1\end{bmatrix}$, and $\begin{bmatrix}-1\\3\end{bmatrix}$ are linearly dependent since 7 times the first vector minus 4 times the second vector is equal to the third vector.

4. $\begin{bmatrix}1\\2\\0\end{bmatrix}$, $\begin{bmatrix}3\\2\\0\end{bmatrix}$, and $\begin{bmatrix}9\\10\\0\end{bmatrix}$ are linearly dependent. Clearly they cannot span all of three-dimensional space, because no vector with a nonzero third component can be generated from this set.

Notice the relationship between examples (2) and (3). The vectors in example (2) are linearly independent, therefore they span the plane. Thus any other vector with two components is a linear combination of these two vectors. In example (3), then, we know that the set will be linearly dependent before being told what the third vector is. This suggests the following rule: There can be no more than n linearly independent vectors in n-dimensional space.

A linearly independent set of vectors has the important property that a vector can be written as a linear combination of the set in only one way. In other words, the coefficients c_i in Equation 1 are unique if the vectors $\mathbf{v}_i$ are linearly independent. This fact can be easily seen, for example, in the case of the standard basis, for there is only one vector in the basis which has a nonzero entry for any given component.

For linearly dependent vectors, however, the situation is different. If a vector can be written as a linear combination of a linearly dependent set of vectors, then there are an infinite number of sets of coefficients that will work. Let us attempt to demonstrate this fact with the aid of geometric intuition. Suppose that we wish to write vector $\mathbf{v}$ as a linear combination of three vectors $\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$ in the plane. Let us choose any arbitrary coefficient c_1 for the vector $\mathbf{v}_1$. As shown in Figure 6, there must be a vector $\mathbf{w}$ such that $\mathbf{v} = c_1\mathbf{v}_1 + \mathbf{w}$. Thus, if we can write $\mathbf{w}$ as a linear combination of $\mathbf{v}_2$ and $\mathbf{v}_3$, i.e., $\mathbf{w} = c_2\mathbf{v}_2 + c_3\mathbf{v}_3$, then we have succeeded in writing $\mathbf{v}$ as a linear combination of $\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$. But clearly we can do this, because $\mathbf{w}$ is a vector in the plane, and $\mathbf{v}_2$ and $\mathbf{v}_3$ together span the plane.

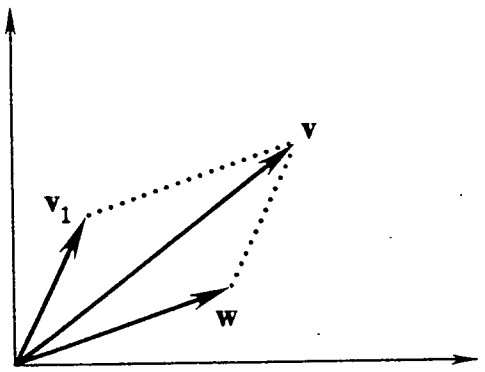

FIGURE 6.

VECTOR SPACES

Let us pause to reflect for a moment upon what a vector is. I have implied that a vector is a list of numbers, and I have also used the term to refer to a point or an arrow in space. Are both of these objects vectors, or is one just a heuristic representation for the other? Are there other objects that should be called vectors? Just what is a vector?

As is often the case in mathematics, these kinds of questions are solved by being avoided. Consider the following definition of an abstract vector space, and try to decide what a vector is.

A *vector space* is a set V of elements, called vectors, with the following properties:

- To every pair, $\mathbf{u}$ and $\mathbf{v}$, of vectors in V, there corresponds a vector $\mathbf{u} + \mathbf{v}$ also in V, called the sum of $\mathbf{u}$ and $\mathbf{v}$, in such a way that addition is commutative and associative.
- For any scalar c and any vector $\mathbf{v}$ in V, there is a vector $c\mathbf{v}$ in V, called the product of c and $\mathbf{v}$, in such a way that multiplication by scalars is associative and distributive with respect to vector addition.[1]

The answer to the question is that a vector is an undefined object in linear algebra, much like a line in geometry. The definition of a vector space simply lists the properties that vectors must have, without specifying what a vector must be. Thus, any set of objects that obey these properties can be called a vector space. Lists of numbers are vectors when addition is defined as adding components separately and scalar multiplication is defined as multiplying all the components by the scalar, because these operations fill all the requirements of a vector space. Arrows or points in space are also vectors when addition is defined geometrically as taking the diagonal of a parallelogram and scalar multiplication is defined as lengthening or shortening the arrow, because again, these operations fill the requirements of a vector space. A seemingly unrelated example of a vector space is the set of polynomials of order n, with addition and scalar multiplication defined in the obvious way.

This sort of abstraction is common in mathematics. It is useful because any theorem that is true about a general vector space must be

[1] I have left out certain technicalities usually included as axioms for a vector space. These include the axiom that there must be a zero vector, and for every vector, there is an additive inverse.

true about any instantiation of a vector space. We can therefore discuss general properties of vector spaces without being committed to choosing a particular representation such as a list of numbers. Much of the discussion about linear combinations and linear independence was of this nature.

When we do choose numbers to represent vectors, we use the following scheme. First we choose a basis for the space. Since every vector in the space can be written as a linear combination of the basis vectors, each vector has a set of coefficients $c_1, c_2, \ldots, c_n$ which are the coefficients in the linear combination. These coefficients are the numbers used as the components of the vector. As was shown in the previous section, the coefficients of a given vector are unique because basis vectors are linearly independent.

There is a certain arbitrariness in assigning the numbers, since there are infinitely many sets of basis vectors, and each vector in the space has a different description depending on which basis is used. That is, the coefficients, which are referred to as *coordinates*, are different for different choices of basis. The implications of this fact are discussed further in a later section where I also discuss how to relate the coordinates of a vector in one basis to the coordinates of the vector in another basis. Chapter 22 contains a lengthy discussion of several issues relating to the choice of basis.

INNER PRODUCTS

As of yet, we have no way to speak of the length of a vector or of the similarity between two vectors. This will be rectified with the notion of an inner product.

The inner product of two vectors is the sum of the products of the vector components. The notation for the inner product of vectors $\mathbf{v}$ and $\mathbf{w}$ is $\mathbf{v} \cdot \mathbf{w}$. As with vector addition, the inner product is defined only if the vectors have the same number of components.

Example:

$$\mathbf{v}=\begin{bmatrix} 3 \\ -1 \\ 2 \end{bmatrix} \qquad \mathbf{w}=\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$$

$$\mathbf{v} \cdot \mathbf{w} = (3 \cdot 1) + (-1 \cdot 2) + (2 \cdot 1) = 3.$$

The inner product is a kind of multiplication between vectors, although somewhat of a strange sort of multiplication, since it produces a single number from a pair of vectors. What does this single number "measure"?

Length

As a special case, consider taking the inner product of a vector with itself. An example is the vector $\mathbf{v} = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$ in Figure 7. The inner product of $\mathbf{v}$ with itself is

$$\mathbf{v} \cdot \mathbf{v} = 3^2 + 4^2 = 25.$$

Consider the right triangle in Figure 7 with sides corresponding to the components of $\mathbf{v}$, and hypotenuse $\mathbf{v}$ itself. The Pythagorean theorem tells us that the square of the length of $\mathbf{v}$ is equal to the sum of the squares of the sides. Since this is exactly what is calculated by the inner product $\mathbf{v} \cdot \mathbf{v}$, it appears that a reasonable definition of the *length* of a vector is the square root of the inner product of the vector with itself. Thus we define the length of a vector $\mathbf{v}$, denoted by $\|\mathbf{v}\|$, as

$$\|\mathbf{v}\| = (\mathbf{v} \cdot \mathbf{v})^{1/2}.$$

Although the definition was motivated by an example in two dimensions, it can be applied to any vector. Notice that many of the

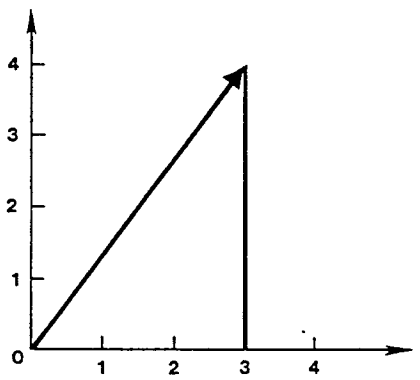

FIGURE 7.

properties we intuitively associate with length are included in this definition. For example, if a vector has larger components than another vector, it will be longer, because the squared components will contribute to a larger inner product. Multiplying a vector by a scalar produces a new vector whose length is the absolute value of the scalar times the length of the old vector:

$$\|c\mathbf{v}\| = |c|\,\|\mathbf{v}\|.$$

This is a property that can be easily proved. Somewhat harder to prove is the so-called triangle inequality, which states that the length of the sum of two vectors is less than or equal to the sum of the lengths of the two vectors:

$$\|\mathbf{v}_1 + \mathbf{v}_2\| \leqslant \|\mathbf{v}_1\| + \|\mathbf{v}_2\|.$$

Geometrically, the triangle inequality corresponds to the statement that one side of a triangle is no longer than the sum of the lengths of the other two sides.

Thus, in the special case where the operands are the same vector, the inner product is closely related to the idea of length. What if the operands are different vectors?

Angle

The angle between two vectors $\mathbf{v}$ and $\mathbf{w}$ is defined in terms of the inner product by the following definition:

$$\cos\theta = \frac{\mathbf{v}\cdot\mathbf{w}}{\|\mathbf{v}\|\,\|\mathbf{w}\|} \tag{2}$$

where θ is the angle between $\mathbf{v}$ and $\mathbf{w}$. Note that all of the quantities on the right hand side of the equation are easily calculated for n-dimensional vectors. At the end of this section, I will show geometrically why this formula is correct in two-dimensional space, using the ordinary geometrical definition of angle.

Example. Find the angle θ between the vectors $\mathbf{v}_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ and $\mathbf{v}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$. First, we calculate the necessary inner product and lengths:

$$\mathbf{v}_1 \cdot \mathbf{v}_2 = 1 \qquad \|\mathbf{v}_1\| = 1 \qquad \|\mathbf{v}_2\| = \sqrt{2},$$

and then substitute these values in Equation 2:

$$\cos\theta = \frac{1}{1 \cdot \sqrt{2}} = 0.707.$$

Thus,

$$\theta = \cos^{-1}(0.707) = 45°.$$

This result could also have been found using basic trigonometry, but clearly the inner product method is superior in general (consider finding the angle between vectors with forty components!).

The inner product is often said to measure the "match" or "similarity" between two vectors. In a vague sense, this seems to be the case from the definition of the inner product as the sum of products. Equation 2, however, shows this in a clearer way: Writing out the equation in terms of the components of the vectors gives

$$\cos\theta = \frac{\sum_{i=1}^{n} v_i w_i}{(\sum_{i=1}^{n} v_i^2)^{1/2} (\sum_{i=1}^{n} w_i^2)^{1/2}}.$$

This is the formula for the correlation between two sets of numbers with zero means.

We can use our geometrical intuitions about angles and our understanding of correlation to turn Equation 2 around and gain a better understanding of the inner product. This understanding is important for the analysis of PDP models, because as will be seen, PDP models often compute inner products. Let us imagine moving two vectors around in space like the hands on a clock. If we hold the lengths of the vectors constant, then Equation 2 says that the inner product is proportional to the cosine of the angle: $\mathbf{v} \cdot \mathbf{w} = \|\mathbf{v}\| \|\mathbf{w}\| \cos\theta$. For example, if the angle between the vectors is zero, where the cosine is at a maximum, the inner product must therefore be at a maximum. As the two vectors move farther apart, the cosine decreases, thus the inner product decreases. It reaches zero when the angle is 90° , and its most negative value when the angle between the vectors is 180°, that is, when the vectors point in opposite directions. Thus, the closer the two vectors are, the larger the inner product. The more the vectors point in opposite directions, the more negative the inner product.

We must be careful, however, in claiming that two vectors are closer together than two others because they have a larger inner product. We

must remember to divide the inner product by the lengths of the vectors involved to make such comparative statements.

An important special case occurs when the inner product is zero. In this case, the two vectors are said to be *orthogonal.* Plugging zero into the right side of Equation 2 gives

$$\cos\theta = 0.$$

which implies that the angle between the vectors is 90°. Thus, orthogonal vectors are vectors which lie at right angles to one another.

We will often speak of a set of orthogonal vectors. This means that every vector in the set is orthogonal to every other vector in the set. That is, every vector lies at a right angle to every other vector. A good example in three-dimensional space is the standard basis referred to earlier. Although we will skip the proof, it is probably clear that any orthogonal set is linearly independent. Indeed, orthogonality is stronger than linear independence: whereas every orthogonal set is linearly independent, there are very many linearly independent sets of vectors that are not orthogonal. An example in two-dimensional space is the pair $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ and $\begin{bmatrix} 2 \\ 1 \end{bmatrix}$. When we choose a basis for a space, we typically choose an orthogonal basis. In fact, in much of classical physics and mathematics, there is not the slightest hint that a basis should be anything but orthogonal.

Projections

A further application of the inner product, closely related to the ideas of length and angle, is the notion of a projection of one vector onto another. An example is given in Figure 8. The distance x is the projection of $\mathbf{v}$ on $\mathbf{w}$. In two dimensions, we readily know how to calculate the projection. It is

$$x = \|\mathbf{v}\| \cos\theta \tag{3}$$

where θ is the angle between $\mathbf{v}$ and $\mathbf{w}$. This formula generalizes, and for any vectors $\mathbf{v}$ and $\mathbf{w}$, the projection of $\mathbf{v}$ on $\mathbf{w}$ is given by Equation 3. It is a scalar which can be thought of as indicating how much $\mathbf{v}$ is pointing in the direction of $\mathbf{w}$.

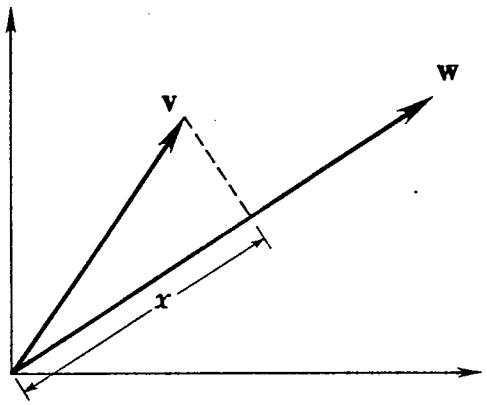

FIGURE 8.

There is a close relationship between the inner product and the projection. Using Equation 2, we can rewrite the formula for the projection:

$$\begin{aligned} x &= \|\mathbf{v}\| \cos\theta \\ &= \|\mathbf{v}\| \frac{\mathbf{v} \cdot \mathbf{w}}{\|\mathbf{v}\| \, \|\mathbf{w}\|} \\ &= \frac{\mathbf{v} \cdot \mathbf{w}}{\|\mathbf{w}\|}. \end{aligned}$$

Thus, the projection is the inner product divided by the length of $\mathbf{w}$. In particular, if $\mathbf{w}$ has length one, then $\|\mathbf{w}\| = 1$, and the projection of $\mathbf{v}$ on $\mathbf{w}$ and the inner product of $\mathbf{v}$ and $\mathbf{w}$ are the same thing. This way of thinking about the inner product is consistent with our earlier comments. That is, if we hold the lengths of $\mathbf{v}$ and $\mathbf{w}$ constant, then we know that the inner product gets larger as $\mathbf{v}$ moves toward $\mathbf{w}$. From the picture, we see that the projection gets larger as well. When the two vectors are orthogonal, the projection as well as the inner product are zero.

Inner Products in Two Dimensions

Equation 2 can be shown to be correct in two-dimensional space with the help of some simple geometry. Let $\mathbf{v}$ and $\mathbf{w}$ be two vectors in the plane, and θ be the angle between them, as shown in Figure 9. Denote the x and y coordinates of $\mathbf{v}$ and $\mathbf{w}$ by v_x, v_y and w_x, w_y, respectively.

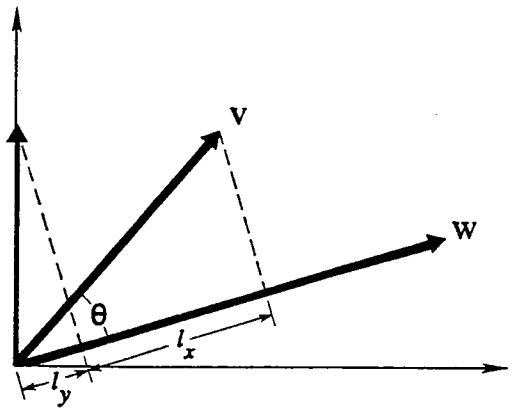

FIGURE 9.

Let l denote the projection of $\mathbf{v}$ on $\mathbf{w}$. We have $l = \|\mathbf{v}\| \cos\theta$ from geometry. We can break l into two pieces l_x and l_y as shown in the figure. l_y can be computed from the diagram by noticing that triangles OAD and COB, in Figure 10, are similar triangles. Thus, the ratio of corresponding sides is constant:

$$\frac{l_y}{v_y} = \frac{w_y}{\|\mathbf{w}\|},$$

giving

$$l_y = \frac{v_y w_y}{\|\mathbf{w}\|}.$$

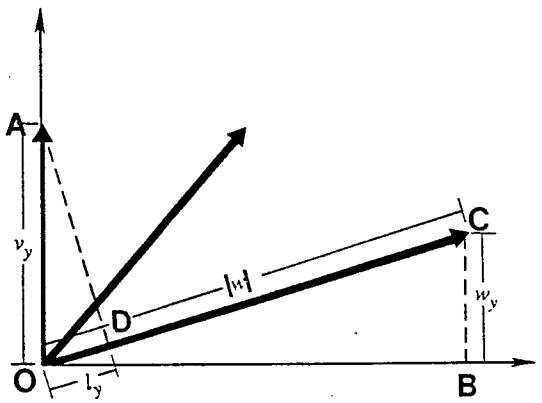

FIGURE 10.

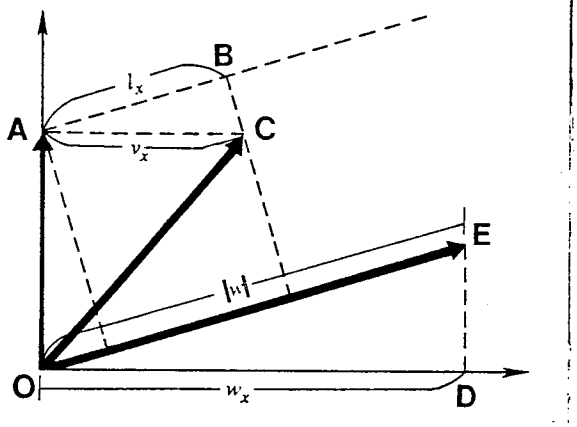

FIGURE 11.

In Figure 11, we see how to compute l_x, by observing that triangles EOD and CAB are similar. Thus,

$$\frac{l_x}{v_x} = \frac{w_x}{\|\mathbf{w}\|},$$

giving

$$l_x = \frac{v_x\, w_x}{\|\mathbf{w}\|}.$$

We can now write $l = l_x + l_y$, which yields

$$l = \|\mathbf{v}\| \cos\theta = l_x + l_y = \frac{v_x\, w_x}{\|\mathbf{w}\|} + \frac{v_y\, w_y}{\|\mathbf{w}\|} = \frac{\mathbf{v}\cdot\mathbf{w}}{\|\mathbf{w}\|}.$$

Thus,

$$\cos\theta = \frac{\mathbf{v}\cdot\mathbf{w}}{\|\mathbf{v}\|\,\|\mathbf{w}\|}.$$

Algebraic Properties of the Inner Product

In this section, we collect together some useful algebraic theorems concerning inner products. Most of these theorems can be easily proved using the definition of the inner product and properties of real

numbers. In what follows, c and c_i will be any scalars, and the $\mathbf{v}$ and $\mathbf{w}$ will be n-dimensional vectors.

$$\mathbf{v} \cdot \mathbf{w} = \mathbf{w} \cdot \mathbf{v} \tag{4}$$

$$c\,(\mathbf{v} \cdot \mathbf{w}) = (c\,\mathbf{v}) \cdot \mathbf{w} = \mathbf{v} \cdot (c\,\mathbf{w}) \tag{5}$$

$$\mathbf{w} \cdot (\mathbf{v}_1 + \mathbf{v}_2) = \mathbf{w} \cdot \mathbf{v}_1 + \mathbf{w} \cdot \mathbf{v}_2 \tag{6}$$

The first theorem says simply that order is unimportant; the inner product is commutative. The second and third theorems show that the inner product is a *linear* function, as we will discuss at length in a later section. We can combine these two equations to get $\mathbf{w} \cdot (c_1\mathbf{v}_1 + c_2\mathbf{v}_2) = c_1\,(\mathbf{w} \cdot \mathbf{v}_1) + c_2\,(\mathbf{w} \cdot \mathbf{v}_2)$. It is also well worth our while to use mathematical induction to generalize this formula, giving us

$$\mathbf{w} \cdot (c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_n\mathbf{v}_n) =$$

$$c_1\,(\mathbf{w} \cdot \mathbf{v}_1) + c_2\,(\mathbf{w} \cdot \mathbf{v}_2) + \cdots + c_n\,(\mathbf{w} \cdot \mathbf{v}_n). \tag{7}$$

This important result tells us how to calculate the inner product of $\mathbf{w}$ and a linear combination of vectors.

Another useful theorem is

$$|\mathbf{v} \cdot \mathbf{w}| \leqslant \|\mathbf{v}\|\,\|\mathbf{w}\| \tag{8}$$

This is known as the Cauchy-Schwartz inequality. It gives an upper bound on the inner product.

ONE UNIT IN A PARALLEL DISTRIBUTED PROCESSING SYSTEM

In this section, we show how some of the concepts we have introduced can be used in analyzing a very simple PDP model. Consider the processing unit in Figure 12 which receives inputs from the n units below. Associated with each of the $n+1$ units there is a scalar *activation value*. We shall use the scalar u to denote the activation of the output unit and the vector $\mathbf{v}$ to denote the activations of the n input units. That is, the ith component of $\mathbf{v}$ is the activation of the ith input unit. Since there are n input units, $\mathbf{v}$ is an n-dimensional vector.

Associated with each link between the input units and the output unit, there is a scalar weight value, and we can think of the set of n

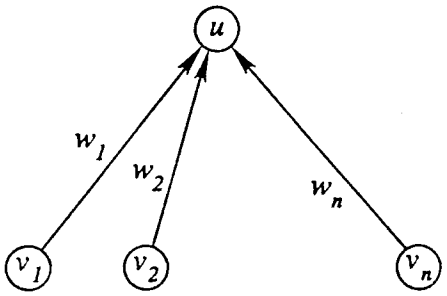

FIGURE 12.

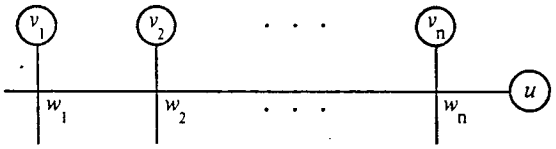

FIGURE 13.

weights as an n-dimensional vector $\mathbf{w}$. This is the *weight vector* corresponding to the output unit. Later we will discuss a model with many output units, each of which will have its own weight vector.

Another way to draw the same model is shown in Figure 13. Here we have drawn the n input units at the top with the output unit on the right. The components of the weight vector are stored at the junctions where the vertical input lines meet the horizontal output line. Which diagram is to be preferred (Figure 12 or Figure 13) is mostly a matter of taste, although we will see that the diagram in Figure 13 generalizes better to the case of many output units.

Now to the operation of the model: Let us assume that the activation of each input unit is multiplied by the weight on its link, and that these products are added up to give the activation of the output unit. Using the definition of the inner product, we translate that statement into mathematics as follows:

$$u = \mathbf{w} \cdot \mathbf{v}.$$

The activation of the output unit is the inner product of its weight vector with the vector of input activations.

The geometric properties of the inner product give us the following picture to help in understanding what the model is computing. We imagine that the set of possible inputs to the model is a vector space. It is an n-dimensional space, where n is the number of input lines. The weight vector also has n components, thus we can plot the weight vector in the input space. The advantage of doing this is that we can now state how the system will respond to the various inputs. As we have seen, the inner product gives an indication of how close two vectors are. Thus, in this simple PDP model, the output activation gives an indication or measurement of how close the input vector is to the stored weight vector. The inputs lying close to the weight vector will yield a large positive response, those lying near 90° will yield a zero response, and those pointing in the opposite direction will yield a large negative response. If we present a succession of input vectors of constant length, the output unit will respond most strongly to that input vector which is closest to its weight vector, and will drop off in response as the input vectors move away from the weight vector.

One way to describe the functioning of the processing unit is to say that it splits the input space into two parts, the part where the response is negative and the part where the response is positive. We can easily imagine augmenting the unit in the following way: if the inner product is positive, output a 1; if the inner product is negative, output a 0. This unit, referred to as a *linear threshold unit*, explicitly computes which part of the space the input lies in.

In some models, the weight vector is assumed to be normalized, that is, $\|\mathbf{w}\| = 1$. As we have seen, in this case, the activation of the output unit is simply the projection of the input vector on the weight vector.

MATRICES AND LINEAR SYSTEMS

The first section introduced the concepts of a vector space and the inner product. We have seen that vectors may be added together and multiplied by scalars. Vectors also have a length, and there is an angle between any pair of vectors. Thus, we have good ways of describing the structure of a set of vectors.

The usefulness of vectors can be broadened considerably by introducing the concept of a matrix. From an abstract point of view, matrices are a kind of "operator" that provide a mapping from one vector space

to another vector space. They are at the base of most of the models in this book which take vectors as inputs and yield vectors as outputs.

First, we will define matrices and show that they have an algebra of their own which is analogous to that of vectors. In particular, matrices can be added together and multiplied by scalars.

MATRICES

A matrix is simply an array of real numbers. If the array has m rows and n columns, then we will refer to the matrix as an $m \times n$ matrix. Capital letters will be used to denote matrices.

Examples:

$$\mathbf{M} = \begin{bmatrix} 3 & 4 & 5 \\ 1 & 0 & 1 \end{bmatrix} \qquad \mathbf{N} = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{P} = \begin{bmatrix} 10 & -1 \\ -1 & 27 \end{bmatrix}$$

M is a 2 × 3 matrix, **N** is a 3 × 3 matrix, and **P** is a 2 × 2 matrix.

Some special matrices. There are several classes of matrices that are useful to identify. A *square* matrix is a matrix with the same number of rows and columns. The matrices **N** and **P** are examples of square matrices. A *diagonal* matrix is a square matrix that is zero everywhere except on its main diagonal. An example is matrix **N**. A *symmetric* matrix is a square matrix whose i,jth element is equal to its j,ith element. Any diagonal matrix is symmetric. Matrix **P** is an example of a symmetric matrix that is not diagonal. Finally, the diagonal matrix that has all ones on its main diagonal is referred to as the identity matrix, and is denoted **I**.

Multiplication by Scalars

A matrix can be multiplied by a scalar by multiplying every element in the matrix by that scalar.

Example:

$$3\,\mathbf{M} = 3 \begin{bmatrix} 3 & 4 & 5 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 9 & 12 & 15 \\ 3 & 0 & 3 \end{bmatrix}$$

Addition of Matrices

Matrices are added together by adding corresponding elements. Only matrices that have the same number of rows and columns can be added together.

Example:

$$\mathbf{M}+\mathbf{N}=\begin{bmatrix}3 & 4 & 5\\ 1 & 0 & 1\end{bmatrix}+\begin{bmatrix}-1 & 0 & 2\\ 4 & 1 & -1\end{bmatrix}=\begin{bmatrix}2 & 4 & 7\\ 5 & 1 & 0\end{bmatrix}$$

Notice that there is a close relationship between these definitions and the corresponding definitions for vectors. In fact, for fixed integers m and n, the set of all $m \times n$ matrices is another example of a vector space. However, we will not exploit this fact, rather, we will think about matrices in another way, in terms of functions from one vector space to another. This is the subject of the next section.

Multiplication of a Vector by a Matrix

We now link up vectors and matrices by showing how a vector can be multiplied by a matrix to produce a new vector. Consider the matrix $\mathbf{W}=\begin{bmatrix}3 & 4 & 5\\ 1 & 0 & 1\end{bmatrix}$ and the vector $\mathbf{v}=\begin{bmatrix}1\\ 0\\ 2\end{bmatrix}$. We wish to define a vector $\mathbf{u}$ which is the product of $\mathbf{W}$ and $\mathbf{v}$, and denoted

$$\mathbf{u}=\mathbf{W}\mathbf{v}=\begin{bmatrix}3 & 4 & 5\\ 1 & 0 & 1\end{bmatrix}\begin{bmatrix}1\\ 0\\ 2\end{bmatrix}.$$

To define this operation, first imagine breaking the matrix into its rows. Each row of the matrix is a list of three numbers. We can think of the row as a three-dimensional vector and speak of the *row vectors* of the matrix. There are two such row vectors. Now consider forming the inner products of each of these row vectors with the vector $\mathbf{v}$. This will yield two numbers. These two numbers can be thought of as a two-dimensional vector $\mathbf{u}$, which is defined to be the product $\mathbf{W}\mathbf{v}$.

Example:

$$\mathbf{u} = \mathbf{W}\mathbf{v} = \begin{bmatrix} 3 & 4 & 5 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} 3 \cdot 1 + 4 \cdot 0 + 5 \cdot 2 \\ 1 \cdot 1 + 0 \cdot 0 + 1 \cdot 2 \end{bmatrix} = \begin{bmatrix} 13 \\ 3 \end{bmatrix}$$

The components of **u** are the inner products of **v** with the row vectors of **W**.

For a general $m \times n$ matrix **W** and an n-dimensional vector **v**,[2] the product **Wv** is an m-dimensional vector **u**, whose elements are the inner products of **v** with the row vectors of **W**. As suggested by Figure 14, the ith component of **u** is the inner product of **v** with the ith row vector of **W**. Thus, the multiplication of a vector by a matrix can be thought of as simply a shorthand way to write down a series of inner products of a vector with a set of other vectors. The vector **u** tabulates the results. This way of thinking about the multiplication operation is a good way to conceptualize what is happening in a PDP model with many output units, as we will see in the next section.

There is another way of writing the multiplication operation that gives a different perspective on what is occurring. If we imagine breaking the matrix up into its columns, then we can equally well speak of the *column vectors* of the matrix. It can then be easily shown that the multiplication operation **Wv** produces a vector **u** that is a linear combination of the column vectors of **W**. Furthermore, the coefficients of the linear combination are the components of **v**. For example, letting $\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3$ be the column vectors of **W**, we have

$$\mathbf{u} = v_1\mathbf{w}_1 + v_2\mathbf{w}_2 + v_3\mathbf{w}_3 = \left[1 \begin{bmatrix} 3 \\ 1 \end{bmatrix} + 0 \begin{bmatrix} 4 \\ 0 \end{bmatrix} + 2 \begin{bmatrix} 5 \\ 1 \end{bmatrix} \right] = \begin{bmatrix} 13 \\ 3 \end{bmatrix}$$

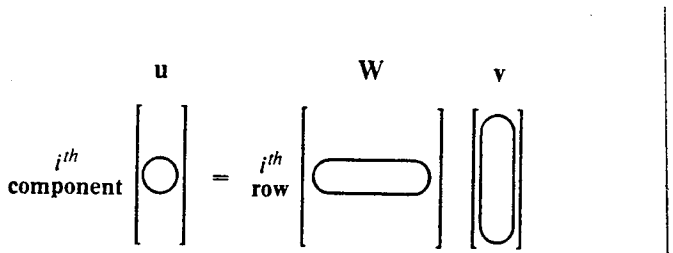

FIGURE 14.

[2] The dimensionality of **v** must be equal to the number of columns of **W** so that the inner products can be defined.

where the v_i are the components of $\mathbf{v}$. This way of viewing the multiplication operation is suggested in Figure 15 for a matrix with n columns.

If we let the term *column space* refer to the space spanned by the column vectors of a matrix, then we have the following interesting result: The vector $\mathbf{u}$ is in the column space of $\mathbf{W}$.

Finally, it is important to understand what is happening on an abstract level. Notice that for each vector $\mathbf{v}$, the operation $\mathbf{Wv}$ produces another vector $\mathbf{u}$. The operation can thus be thought of as a mapping or function from one set of vectors to another set of vectors. That is, if we consider an n-dimensional vector space $\mathbf{V}$ (the domain) and an m-dimensional vector space U (the range), then the operation of multiplication by a fixed matrix $\mathbf{W}$ is a function from $\mathbf{V}$ to U, as shown in Figure 16. It is a function whose domain and range are both vector spaces.

Algebraic Properties of Matrix Mapping

Several properties of matrix-vector multiplication follow directly from the properties of the inner product. In all cases, the number of

$$\underset{\mathbf{W}}{\begin{bmatrix} | & & | \\ \mathbf{w}_1 & \cdots & \mathbf{w}_n \\ | & & | \end{bmatrix}} \underset{\mathbf{v}}{\begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix}} = \underset{\mathbf{u}}{\begin{bmatrix} | \\ v_1\mathbf{w}_1 + \cdots + v_n\mathbf{w}_n \\ | \end{bmatrix}}$$

FIGURE 15.

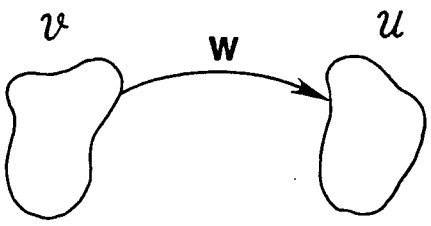

FIGURE 16.

components of the vector must be the same as the number of columns of the matrix.

$$\mathbf{W}(a\mathbf{v}) = a\mathbf{W}\mathbf{v} \tag{9}$$

$$\mathbf{W}(\mathbf{u} + \mathbf{v}) = \mathbf{W}\mathbf{u} + \mathbf{W}\mathbf{v} \tag{10}$$

These equations are the counterparts to Equations 5 and 6. As in that section, they can be combined and generalized to general linear combinations:

$$\mathbf{W}(c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_n\mathbf{v}_n) = c_1(\mathbf{W}\mathbf{v}_1) + c_2(\mathbf{W}\mathbf{v}_2) + \cdots + c_n(\mathbf{W}\mathbf{v}_n) \tag{11}$$

In the next theorem, the matrices **M** and **N** must have the same number of rows and columns.

$$\mathbf{M}\mathbf{v} + \mathbf{N}\mathbf{v} = (\mathbf{M} + \mathbf{N})\mathbf{v} \tag{12}$$

ONE LAYER OF A PARALLEL DISTRIBUTED PROCESSING SYSTEM

I now generalize the simple model presented earlier to show how matrices can be used in analyzing PDP models. Consider Figure 17, which is the generalization of Figure 12 to the case of many output units. Suppose that there are m output units, each one connected to all of the n input units. Denote the activation of the output units by $u_1, u_2, \ldots, u_m$. Each output unit has its own weight vector $\mathbf{w}_i$, separate from the other output units. As before, the activation rule

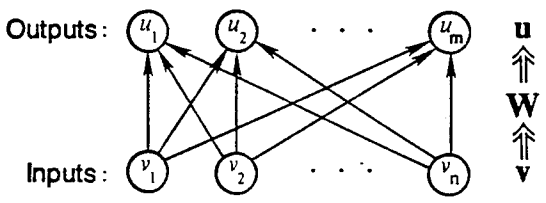

FIGURE 17.

says that the activation of an output unit is given by the inner product of its weight vector with the input vector, thus,

$$u_i = \mathbf{w}_i \cdot \mathbf{v}.$$

If we form a matrix $\mathbf{W}$ whose row vectors are the $\mathbf{w}_i$, then we can use the rule for matrix-vector multiplication to write all of the computations at once. Let $\mathbf{u}$ be the vector whose components are the u_i. Then

$$\mathbf{u} = \mathbf{W}\mathbf{v}.$$

This is a very succinct expression of the computation performed by the network. It says that for each input vector $\mathbf{v}$, the network produces an output vector $\mathbf{u}$ whose components are the activations of the output units.

Another way to draw the network is shown in Figure 18, which is the generalization of Figure 13 to the case of many output units. At each junction in the diagram there is a weight connecting an input unit with an output unit.[3] The weight vectors associated with each output unit appear on the horizontal lines. When drawn this way, it is clear why a matrix appears in the equation linking the output vector to the input vector: The array of junctions in the diagram is exactly the weight matrix $\mathbf{W}$.

Now let us attempt to understand geometrically what is being computed by the model. Each output unit is computing the inner product

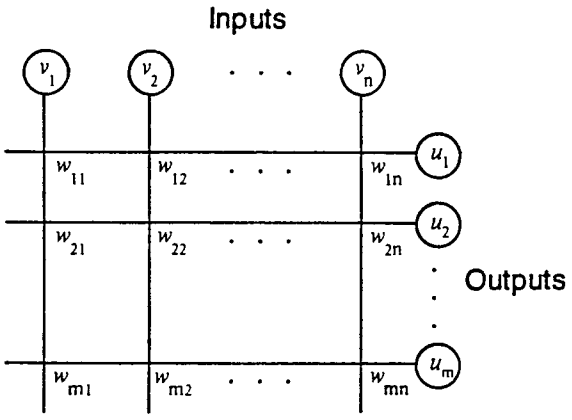

FIGURE 18.

[3] Note that the weight in the ith row and jth column connects the jth input unit to the ith output unit.

of its weight vector and the input vector (which is common to all output units). Thus, each unit can be thought of as computing how close its weight vector is to the input vector. A larger activation is attained the closer the two vectors are. If all of the weight vectors have the same length, then that output unit with the largest activation will be the unit whose weight vector is closest to the input vector.

In the model with only one output unit, we imagined plotting the weight vector in the input vector space. This enabled us to see directly which input vectors led to a large response and which input vectors led to a small response. In the model with several output units, we can generalize by plotting each weight vector in the input space. Now we can see for each unit which inputs it responds to. If the weight vectors are spread around in the space, then every input will lead to some response. Also, the different units will respond to different inputs. If the weight vectors are assumed to have unit length, then the activation of the ith output unit is just the projection of $\mathbf{v}$ on the ith weight vector. For a given input, we can draw the projections of the input on the weight vectors. This gives us a graphic representation of the output of the network. It should be emphasized, however, that this representation is useful mostly as a conceptual tool. The graphic approach cannot be used in most systems, which can have hundreds or thousands of input lines.

Another perspective on the operation of the model can be obtained by focusing on the columns of the weight matrix rather than on its rows. Whereas the rows of the matrix are the weights on the lines coming *in* to the processing units, the columns correspond to the weights on the lines going *out* from the processing units. Each unit on the lower row in Figure 17 is associated with such a vector: The components of the vector are the weights linking that unit with the output units above. These vectors are referred to as the *outgoing weight vectors*, as contrasted with the *incoming weight vectors* which are the rows of the weight matrix.[4] In the previous section, it was seen that when a matrix multiplies a vector, the resulting vector is a linear combination of the columns of the matrix. This view applies to the PDP model as follows: The output vector $\mathbf{u}$ is a linear combination of the outgoing weight vectors from the input units. The coefficients in the linear combination are the activations of the input units. Thus, in this perspective, each input unit multiplies its outgoing weight vector by its activation, and the resulting vectors are added to yield the output vector of the system.

In general, as will be discussed further in a later section, a unit can

[4] This is not standard terminology, and I will continue to use the term *weight vector* to refer to the incoming weight vectors.

appear in a multilayer system and thus have both an incoming weight vector and an outgoing weight vector, as shown in Figure 19. In this case, both views of matrix-vector multiplication can be useful: The unit can be thought of as matching its incoming weight vector to the current input using the inner product, and sending the result of this match multiplied by the outgoing weight vector to the next level.

LINEARITY

A distinction is often made between a *linear* system and a *nonlinear* system. In general, linear systems are relatively easy to analyze and understand, whereas nonlinear systems can be difficult. In this section, I will characterize linear systems. Nonlinear systems are defined simply as everything else. In a later section, I will give some specific examples of nonlinear systems.

Suppose that there is a function f which represents a system in that for each input x to the system, the output y is given by

$$y = f(x).$$

The x and y might be scalars or they might be vectors, depending on the particular system. The function f is said to be *linear* if for any inputs x_1 and x_2, and any real number c, the following two equations hold:

$$f(cx) = c f(x). \tag{13}$$

$$f(x_1 + x_2) = f(x_1) + f(x_2). \tag{14}$$

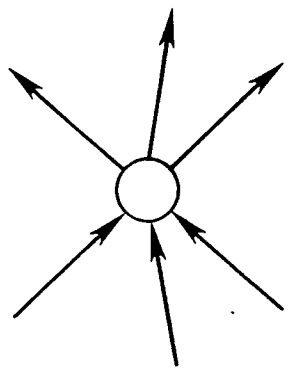

FIGURE 19.

The first of these two equations implies that if we multiply the input by some constant, then the output is multiplied by the same constant. The second equation is more important. Consider presenting the inputs x_1 and x_2 separately to the system and measuring the outputs. In a linear system, knowing how the system responds separately to the inputs is all we need to predict the output of the system when the sum $x_1 + x_2$ is presented. We simply add the outputs found separately to obtain the response to the sum. In a nonlinear system, on the other hand, we might find that the response to the sum is much larger or smaller than would be expected based on the inputs taken separately. The response to the sum might be zero even when strong responses are obtained separately.

If we restrict ourselves to scalar functions of a scalar variable, then the only linear functions are those in which the output is proportional to the input, i.e., for some real number c:

$$y = cx.$$

However, many systems are scalar or vector functions of a vector input. For example, for a fixed vector $\mathbf{w}$, the function

$$u = \mathbf{w} \cdot \mathbf{v}$$

is a scalar function of a vector input $\mathbf{v}$. This function is a linear function because

$$\mathbf{w} \cdot (c\,\mathbf{v}) = c\,(\mathbf{w} \cdot \mathbf{v})$$

and

$$\mathbf{w} \cdot (\mathbf{v}_1 + \mathbf{v}_2) = \mathbf{w} \cdot \mathbf{v}_1 + \mathbf{w} \cdot \mathbf{v}_2.$$

The PDP model with one output unit is an example of such a linear system.

A system in which the output is obtained from the input by matrix multiplication is also a linear system, according to Equations 9 and 10. It turns out that these are the only linear vector functions. That is, if a function f which maps from one vector space to another vector space is linear, then it can be represented by matrix multiplication.[5]

The PDP model discussed in the previous section is an example of a linear system because it is represented by matrix multiplication. In such a system, because of linearity, we know what the output will be when the sum of two vectors is presented if we know the outputs when

[5] Let $\mathbf{v}_i$ be the ith standard basis vector and let $\mathbf{w}_i = f(\mathbf{v}_i)$. Then if $\mathbf{W}$ is a matrix whose columns are the $\mathbf{w}_i$, $f(\mathbf{v}_i) = \mathbf{W}\mathbf{v}$ for all $\mathbf{v}$.

the vectors are presented separately. We also know what the output will be to scalar multiples of a vector. These properties imply that if we know the output to all of the vectors in some set $\{\mathbf{v}_i\}$, then we can calculate the output to any linear combination of the $\mathbf{v}_i$. That is, if $\mathbf{v} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_n\mathbf{v}_n$, then the output when $\mathbf{v}$ is presented to the system is

$$\mathbf{W}\mathbf{v} = \mathbf{W}(c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_n\mathbf{v}_n) = c_1(\mathbf{W}\mathbf{v}_1) + c_2(\mathbf{W}\mathbf{v}_2) + \cdots + c_n(\mathbf{W}\mathbf{v}_n) \tag{15}$$

The terms in the parentheses on the right are known vectors: They are the outputs to the vectors $\mathbf{v}_i$. Thus, we simply multiply these vectors by the c_i to calculate the output when $\mathbf{v}$ is presented. If the $\mathbf{v}_i$ are a basis for some vector space, then every vector in the space is a linear combination of the $\mathbf{v}_i$. Therefore, knowing the outputs of the system to the basis vectors allows us to calculate immediately the output to any other vector in the vector space without reference to the system matrix $\mathbf{W}$. The preceding statement should be studied carefully, because it expresses an extremely important defining property of linear systems. Another way to say the same thing is as follows: Imagine that we are studying some physical system by measuring its responses to various inputs. The system might be electronic or physiological, for example. If it is a linear system, then we should first measure the responses to a set of inputs that constitute a basis for the input space. We then have no need to make any further measurements. The responses of the system to any other input vector can be immediately calculated based on the measurements that we have already made.

MATRIX MULTIPLICATION AND MULTILAYER SYSTEMS

The systems considered until now have been *one-layer* systems. That is, the input arrives at a set of input units, is passed through a set of weighted connections described by a matrix, and appears on a set of output units. Let us now arrange two such systems in *cascade*, so that the output of the first system becomes the input to the next system, as shown in Figure 20. The composite system is a *two-layer* system and is described by two matrix-vector multiplications. An input vector $\mathbf{v}$ is first multiplied by the matrix $\mathbf{N}$ to produce a vector $\mathbf{z}$ on the set of intermediate units:

$$\mathbf{z} = \mathbf{N}\mathbf{v},$$

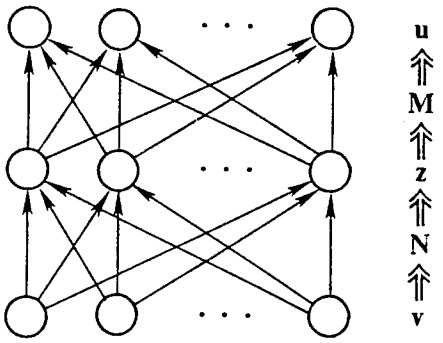

FIGURE 20.

and then $\mathbf{z}$ is multiplied by $\mathbf{M}$ to produce a vector $\mathbf{u}$ on the uppermost set of units:

$$\mathbf{u} = \mathbf{M}\mathbf{z}.$$

Substituting $\mathbf{N}\mathbf{v}$ for $\mathbf{z}$ yields the response for the composite system:

$$\mathbf{u} = \mathbf{M}(\mathbf{N}\mathbf{v}). \tag{16}$$

This equation relates the input vectors $\mathbf{v}$ to the output vectors $\mathbf{u}$.

We will now define an operation on matrices, called *matrix multiplication*, which will simplify the analysis of cascaded systems, allowing us to replace the two matrices $\mathbf{M}$ and $\mathbf{N}$ in Equation 16 by a single matrix $\mathbf{P}$. Matrices $\mathbf{M}$ and $\mathbf{N}$ can be multiplied to produce a matrix $\mathbf{P} = \mathbf{M}\mathbf{N}$ as follows: The i,jth element of $\mathbf{P}$ is the inner product of the ith row of $\mathbf{M}$ with the jth column of $\mathbf{N}$. Note that the order of multiplication is important—the product $\mathbf{M}\mathbf{N}$ is generally not equal to the product $\mathbf{N}\mathbf{M}$. This is to be expected from the asymmetric treatment of $\mathbf{M}$ and $\mathbf{N}$ in the definition.

Example:

$$\begin{bmatrix} 3 & 4 & 5 \\ 1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & 0 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} (3+8-5) & (6+0+5) \\ (1+0-1) & (2+0+1) \\ (0+2-2) & (0+0+2) \end{bmatrix} = \begin{bmatrix} 6 & 11 \\ 0 & 3 \\ 0 & 2 \end{bmatrix}$$

Another way to think about matrix multiplication follows from the definition of matrix-vector multiplication. Each column vector of $\mathbf{P}$ is the product of the matrix $\mathbf{M}$ with the corresponding column vector of $\mathbf{N}$. For example, the first column of $\mathbf{P}$ is computed by multiplying the

first column of **N** by the matrix **M**. This is shown in Figure 21, where we have explicitly shown the column vectors of **N** and **P**.

The product of two matrices is defined only if the number of columns of the first matrix is equal to the number of rows of the second matrix. Otherwise, the inner products cannot be formed. A handy rule is the following: Multiplying an $r \times s$ matrix and an $s \times t$ matrix yields an $r \times t$ matrix.

Let us return to Figure 20 and Equation 16, which describes the system. I make the claim that the matrices **M** and **N** in the equation can be replaced by the matrix **P**, if **P** is the matrix product of **M** and **N**. In other words,

$$\mathbf{u} = \mathbf{M}(\mathbf{N}\mathbf{v}) = (\mathbf{M}\mathbf{N})\mathbf{v} = \mathbf{P}\mathbf{v}.$$

What this equation says is that the two-layer system in Figure 20 is equivalent to a one-layer system with weight matrix **P**. For every input vector **v**, the two systems will produce the same output vector **u**. Thus, for linear systems at least, the distinction between two-layer systems and one-layer systems is more apparent than real.[6]

We can attempt to justify our claim and, in so doing, get a better understanding of matrix multiplication if we examine the system in Figure 20 more closely. Let us assume that a matrix **P** exists which can replace the cascaded pair **M**, **N**, and consider what the element in the first row and the first column of **P** should be. This element gives the strength of the connection between the first component of the input vector **v** and the first component of the output vector **u**. In the cascaded system, there are s paths through which this connection can occur, as shown in Figure 22. We must multiply the weights along each path and add the values for the paths to get the strength of the connection in the equivalent one-layer system. This is calculated as

$$p_{11} = m_{11}n_{11} + m_{12}n_{21} + \cdots + m_{1s}n_{s1}.$$

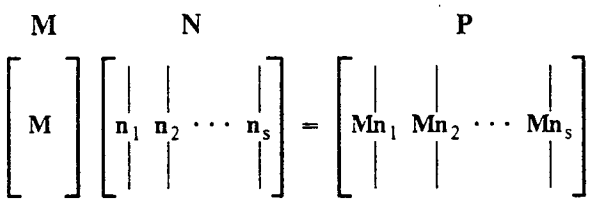

FIGURE 21.

[6] The two systems are identical in the sense that they compute the same function. Of course, they may have different internal dynamics and therefore take different amounts of time to compute their outputs.

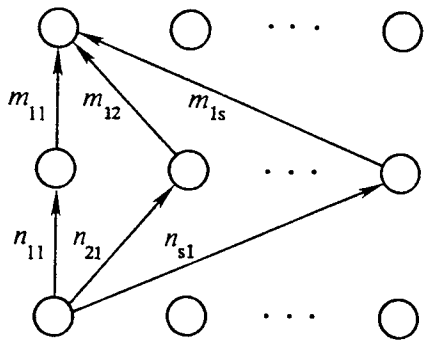

FIGURE 22.

This equation can be easily generalized to give the strength of the connection between the jth element of $\mathbf{v}$ and the ith element of $\mathbf{u}$:

$$p_{ij} = m_{i1} n_{1j} + m_{i2} n_{2j} + \cdots + m_{is} n_{sj}.$$

This formula calculates the inner product between the ith row of $\mathbf{M}$ and the jth column of $\mathbf{N}$, which shows that $\mathbf{P}$ is equal to the product $\mathbf{MN}$.

This result can be extended to systems with more than two layers by induction. For example, in a three-layer system, the first two layers can be replaced with a matrix (as we have just seen), and then that matrix can be multiplied by the matrix of the remaining layer to get a single matrix for the whole system. In general, the cascaded matrices of any n-layer linear system can be replaced by a single matrix which is the product of the n matrices.

As a final comment, the definition of matrix multiplication may seem somewhat odd, especially since it would seem more straightforward to define it by analogy with matrix addition as the element-wise product. In fact, it would be perfectly acceptable to define multiplication as the element-wise product, and then to use another name for the operation we have discussed in this section. However, element-wise multiplication has never found much of an application in linear algebra. Therefore, the term multiplication has been reserved for the operation described in this section, which proves to be a useful definition, as the application to multilayer systems demonstrates.

Algebraic Properties of Matrix Multiplication

The following properties are identical to the corresponding properties of matrix-vector multiplication. This is to be expected given the relationship between matrix multiplication and matrix-vector multiplication (cf. Figure 21).

$$\mathbf{M}(c\mathbf{N}) = c\mathbf{MN} \tag{17}$$

$$\mathbf{M}(\mathbf{N} + \mathbf{P}) = \mathbf{MN} + \mathbf{MP} \tag{18}$$

$$(\mathbf{N} + \mathbf{P})\mathbf{M} = \mathbf{NM} + \mathbf{PM} \tag{19}$$

EIGENVECTORS AND EIGENVALUES

The next two sections develop some of the mathematics important for the study of *learning* in PDP networks. First, I will discuss *eigenvectors* and *eigenvalues* and show how they relate to matrices. Second, I will discuss *outer products*. Outer products provide one way of constructing matrices from vectors. In a later section, I will bring these concepts together in a discussion of learning.

Recall the abstract point of view of matrices and vectors that was discussed earlier: The equation $\mathbf{u} = \mathbf{Wv}$ describes a *function* or *mapping* from one space, called the *domain*, to another space, called the *range*. In such vector equations, both the domain and the range are vector spaces, and the equation associates a vector $\mathbf{u}$ in the range with each vector $\mathbf{v}$ in the domain.

In general, a function from one vector space to another can associate an arbitrary vector in the range with each vector in the domain. However, knowing that $\mathbf{u} = \mathbf{Wv}$ is a linear function highly constrains the form the mapping between the domain and range can have. For example, if $\mathbf{v}_1$ and $\mathbf{v}_2$ are close together in the domain, then the vectors $\mathbf{u}_1 = \mathbf{Wv}_1$ and $\mathbf{u}_2 = \mathbf{Wv}_2$ must be close together in the range. This is known as a *continuity* property of linear functions. Another important constraint on the form of the mapping is the following, which has already been discussed. If $\mathbf{v}_3$ is a linear combination of $\mathbf{v}_1$ and $\mathbf{v}_2$, and the vectors $\mathbf{u}_1 = \mathbf{Wv}_1$ and $\mathbf{u}_2 = \mathbf{Wv}_2$ are known, then $\mathbf{u}_3 = \mathbf{Wv}_3$ is completely determined—it is the same linear combination of $\mathbf{u}_1$ and $\mathbf{u}_2$. Furthermore, if we have a set of basis vectors for the domain, and it is known which vector in the range each basis vector maps to, then the

mappings of all other vectors in the domain are determined (cf. Equation 15).

In this section, let us specialize to the case of square matrices, that is, matrices with the same number of rows as columns. In this case, the domain and the range will have the same number of dimensions (because the vectors $\mathbf{v}$ and $\mathbf{u}$ must have the same number of components), and the vectors in the domain and the range can be plotted in the same space. This is done in Figure 23, where we have shown two vectors before and after multiplication by a matrix.

In general, vectors in this space will change direction as well as length when multiplied by a matrix. However, as demonstrated by one of the vectors in Figure 23, there will be some vectors that will change only in length, not direction. In other words, for these vectors, multiplication by the matrix is no different than multiplication by a simple scalar. Such vectors are known as *eigenvectors*. Each eigenvector $\mathbf{v}$ of a matrix obeys the equation

$$\mathbf{W}\mathbf{v} = \lambda\mathbf{v} \tag{20}$$

where λ is a scalar. λ is called an *eigenvalue*, and indicates how much $\mathbf{v}$ is shortened or lengthened after multiplication by $\mathbf{W}$.

Example:

$$\begin{bmatrix} 4 & -1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} = 2 \begin{bmatrix} 1 \\ 2 \end{bmatrix} \qquad \begin{bmatrix} 3 & 0 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 3 \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

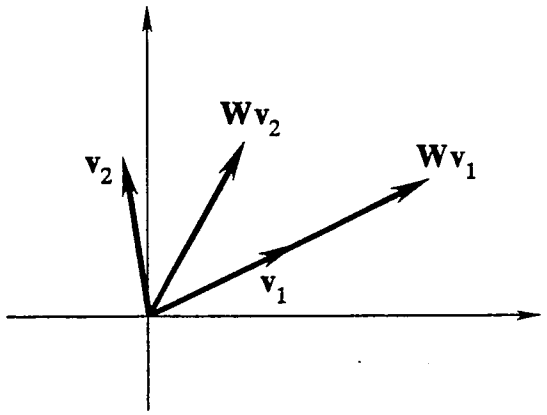

FIGURE 23.

A matrix can have more than one eigenvector, which, geometrically, means that it is possible to have eigenvectors in more than one direction. For example, the leftmost matrix above also has the eigenvector $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ with eigenvalue 3, and the diagonal matrix on the right also has the eigenvector $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$ with eigenvalue 4.

There is another, more trivial, sense in which a matrix can have multiple eigenvectors: Each vector that is collinear with an eigenvector is itself an eigenvector. If $\mathbf{v}$ is an eigenvector with eigenvalue λ, and if $\mathbf{y} = c\,\mathbf{v}$, then it is easy to show that $\mathbf{y}$ is also an eigenvector with eigenvalue λ. For the ensuing discussion, the collinear eigenvectors will just confuse things, so I will adopt the convention of reserving the term eigenvector only for vectors of length 1. This is equivalent to choosing a representative eigenvector for each direction in which there are eigenvectors.

Let us now return to the diagonal matrix $\begin{bmatrix} 3 & 0 \\ 0 & 4 \end{bmatrix}$. We have seen that this matrix has two eigenvectors, $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$, with eigenvalues 3 and 4. The fact that the eigenvalues are the same as the diagonal elements of the matrix is no coincidence: This is true for all diagonal matrices, as can be seen by multiplying any diagonal matrix by one of its eigenvectors—a vector in the standard basis. It is also true that this matrix has only two eigenvectors. This can be seen by considering any vector of the form $\begin{bmatrix} a \\ b \end{bmatrix}$, where a and b are both nonzero. Then we have

$$\begin{bmatrix} 3 & 0 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 3a \\ 4b \end{bmatrix}.$$

Such a vector is not an eigenvector, because the components are multiplied by different scalars. The fact that the matrix has distinct eigenvalues is the determining factor here. If the diagonal elements had been identical, then any two-dimensional vector would indeed have been an eigenvector. This can also be seen in the case of the $n \times n$ identity matrix $\mathbf{I}$, for which every n-dimensional vector is an eigenvector with eigenvalue 1.

In general, an $n \times n$ matrix can have up to, but no more than, n distinct eigenvalues. Furthermore, distinct eigenvalues correspond to distinct directions. To be more precise, if a matrix has n distinct

eigenvalues, then the n associated eigenvectors are *linearly independent*. Although the conditions under which a matrix has a full set of distinct eigenvalues are beyond the scope of this chapter, it is quite possible to have matrices with fewer than n eigenvalues, as in the case of the identity matrix.

I will not discuss how to find eigenvectors and eigenvalues for a particular matrix, but refer the reader to the books on linear algebra listed at the end of the chapter. There are several methods, all of which can be computationally expensive for large matrices. In a later section I will discuss how to construct a certain class of matrices given a set of desired eigenvectors.

The goal now is to show how eigenvectors can be used. To do so, let us begin by assuming that we are dealing with the most favorable case: an $n \times n$ matrix $\mathbf{W}$ with n distinct eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$. Denote the associated linearly independent eigenvectors by $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$. Recall that if we have a set of basis vectors for the domain of a matrix, and if we know the vectors in the range associated with each basis vector, then the mapping of all other vectors in the domain are determined. The eigenvectors of $\mathbf{W}$ form such a basis. This is because there are n eigenvectors, and they are linearly independent. Furthermore, we know the vectors in the range associated with each eigenvector $\mathbf{v}_i$; they are simply the scalar multiples given by $\mathbf{W}\mathbf{v} = \lambda\mathbf{v}$.

To show how to take advantage of these observations, pick an arbitrary vector $\mathbf{v}$ in the domain of $\mathbf{W}$. It can be written as a linear combination of the eigenvectors, because the eigenvectors form a basis:

$$\mathbf{v} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_n\mathbf{v}_n.$$

We can now write:

$$\mathbf{u} = \mathbf{W}\mathbf{v}$$

$$\mathbf{u} = \mathbf{W}\,(c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_n\mathbf{v}_n).$$

Using linearity,

$$\mathbf{u} = c_1(\mathbf{W}\mathbf{v}_1) + c_2(\mathbf{W}\mathbf{v}_2) + \cdots + c_n(\mathbf{W}\mathbf{v}_n).$$

If we next substitute for each of the quantities $\mathbf{W}\mathbf{v}_i$, using Equation 20:

$$\mathbf{u} = c_1\lambda_1\mathbf{v}_1 + c_2\lambda_2\mathbf{v}_2 + \cdots + c_n\lambda_n\mathbf{v}_n. \tag{21}$$

Notice that there are no matrices in this last equation. Each term $c_i\lambda_i$ is a scalar; thus we are left with a simple linear combination of vectors after having started with a matrix multiplication.

This equation should give some idea of the power and utility of the eigenvectors and eigenvalues of a matrix. If we know the eigenvectors and eigenvalues, then, in essence, we can throw away the matrix. We simply write a vector as a linear combination of eigenvectors, then multiply each term by the appropriate eigenvalue to produce Equation 21, which can be recombined to produce the result. Eigenvectors turn matrix multiplication into simple multiplication by scalars.

It is also revealing to consider the magnitudes of the eigenvalues for a particular matrix. In Equation 21, all of the vectors $\mathbf{v}_i$ are of unit length, thus the length of the vector $\mathbf{u}$ depends directly on the product of the magnitudes of the c_i and the eigenvalues λ_i. Consider the vectors that tend to point in the directions of the eigenvectors with large eigenvalues. These are the vectors with large c_i for those eigenvectors. Equation 21 says that after multiplication by the matrix they will be longer than vectors of the same initial length that point in other directions. In particular, of all unit length vectors, the vector that will be the longest after multiplication by the matrix is the eigenvector with the largest eigenvalue. In other words, knowledge of the eigenvectors and eigenvalues of a system tells which input vectors the system will give a large response to. This fact can be useful in the analysis of linear models.

TRANSPOSES AND THE OUTER PRODUCT

The transpose of an $n \times m$ matrix $\mathbf{W}$ is an $m \times n$ matrix denoted $\mathbf{W}^T$. The i,jth element of $\mathbf{W}^T$ is the j,ith element of $\mathbf{W}$.

Example:

$$\begin{bmatrix} 3 & 4 & 5 \\ 1 & 0 & 2 \end{bmatrix}^T = \begin{bmatrix} 3 & 1 \\ 4 & 0 \\ 5 & 2 \end{bmatrix}$$

Another way to describe the transpose is as follows: The row vectors of $\mathbf{W}^T$ are the column vectors of $\mathbf{W}$, and the column vectors of $\mathbf{W}^T$ are the row vectors of $\mathbf{W}$.

Algebraic Properties of the Transpose

$$(\mathbf{W}^T)^T = \mathbf{W}$$

$$(c\mathbf{W})^T = c\mathbf{W}^T$$

$$(\mathbf{M} + \mathbf{N})^T = \mathbf{M}^T + \mathbf{N}^T$$

$$(\mathbf{MN})^T = \mathbf{N}^T\mathbf{M}^T$$

If a matrix is its own transpose, that is if $\mathbf{W}^T = \mathbf{W}$, then the matrix is symmetric.

Outer Products

Before discussing outer products, let me attempt to ward off what could be a confusing aspect of the notation we are using. Consider, for example, the entity below. Is it a matrix with one column or is it a vector?

$$\begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix}$$

The answer is that it could be either—there is no way of distinguishing one from the other based on the notation. There is nothing wrong with this failure to distinguish between vectors and $n \times 1$ matrices for the following reason. In equations involving vectors and matrices, the same results will be obtained whether entities such as the one above are treated as vectors or as matrices. This is true because the algebra for vectors and matrices is exactly the same, as a review of the relevant earlier sections will show. Thus, as long as we are simply interested in calculating values and manipulating equations, there is no need to distinguish between vectors and $n \times 1$ matrices. Rather, by treating them as the same thing, we have a uniform set of procedures for dealing with all equations involving vectors and matrices.

Nevertheless, on the conceptual level, it is important to distinguish between vectors and matrices. The way we are using the terms, a vector is an element in a vector space, whereas a matrix can be used to define a linear mapping from one vector space to another. These are very different concepts.

With this caveat in mind, we will continue to take advantage of the uniformity of notation, blurring the distinction between a vector and an

$n \times 1$ matrix. For example, for every n-dimensional vector $\mathbf{v}$, we can form the transpose $\mathbf{v}^T$, which is simply a matrix with one row. We can then form the product $\mathbf{v}^T\mathbf{u}$, where $\mathbf{u}$ is any n-dimensional vector, as in the following example.

Example:

$$\mathbf{v} = \begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix} \qquad \mathbf{u} = \begin{bmatrix} 0 \\ 4 \\ 1 \end{bmatrix}$$

$$\mathbf{v}^T\mathbf{u} = \begin{bmatrix} 3 & 1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 4 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \end{bmatrix}$$

Notice that the result has only a single component, and that this component is calculated by taking the inner product of the vectors $\mathbf{v}$ and $\mathbf{u}$. In many applications, there is no need to distinguish between vectors with one component and scalars, thus the notation $\mathbf{v}^T\mathbf{u}$ is often used for the inner product.

Let us next consider the product $\mathbf{u}\mathbf{v}^T$. This is a legal product because the number of columns in $\mathbf{u}$ and the number of rows in $\mathbf{v}^T$ are the same, namely one. Following the rule for matrix multiplication, we find that there are n^2 inner products to calculate and that each inner product involves vectors of length one.

Example:

$$\mathbf{u}\mathbf{v}^T = \begin{bmatrix} 1 \\ 4 \\ 0 \end{bmatrix} \begin{bmatrix} 3 & 1 & 2 \end{bmatrix} = \begin{bmatrix} 3 & 1 & 2 \\ 12 & 4 & 8 \\ 0 & 0 & 0 \end{bmatrix}$$

The i,jth element of the resulting matrix is equal to the product $u_i\, v_j$.

For those who may have forgotten the noncommutativity of matrix multiplication, this serves as a good reminder: Whereas the product $\mathbf{v}^T\mathbf{u}$ has a single component, a simple change in the order of multiplication yields an $n \times n$ matrix.

Products of the form $\mathbf{u}\mathbf{v}^T$ are referred to as *outer products*, and will be discussed further in the next section. Note that the rows of the resulting matrix are simply scalar multiples of the vector $\mathbf{v}$. In other words, if we let $\mathbf{W}$ be the matrix $\mathbf{u}\mathbf{v}^T$, and let $\mathbf{w}_i$ be the ith row of $\mathbf{W}$, then we have

$$\mathbf{w}_i = u_i\mathbf{v}$$

where u_i is the ith component of the vector $\mathbf{u}$.

OUTER PRODUCTS, EIGENVECTORS, AND LEARNING

In this section, I discuss two example PDP systems that bring together several of the concepts discussed previously, including eigenvectors and outer products. These systems are described in J. A. Anderson, Silverstein, Ritz, and Jones (1977) and Kohonen (1977).

We have seen that simple linear PDP systems can be modeled with the equation $\mathbf{u} = \mathbf{W}\mathbf{v}$, where $\mathbf{W}$ is a weight matrix. The rows of $\mathbf{W}$ are the weight vectors associated with each of the units in the upper level of the system. Until now, we have taken the matrix $\mathbf{W}$ to be a given, and have discussed how it maps input vectors to output vectors. Let us now consider a simple scheme, referred to as a Hebbian learning rule, whereby we can choose a matrix that associates a particular output vector $\mathbf{u}$ with a particular input vector $\mathbf{v}$. A system that can autonomously implement such a scheme is capable of a rudimentary form of associative learning.

The scheme will only work with input vectors of unit length, so let us begin by making that assumption. Thus, we have $\mathbf{v} \cdot \mathbf{v} = 1$. Let us consider the simplest case, in which the output vector $\mathbf{u}$ has only one component, which we will denote by u. This is the system discussed in Figure 13. We wish a weight vector $\mathbf{w}$ such that when $\mathbf{v}$ is present as the input, the output is u: $u = \mathbf{w} \cdot \mathbf{v}$. Note that u and $\mathbf{v}$ are the given here, and $\mathbf{w}$ is the unknown. To make a choice for $\mathbf{w}$, we can use the following logic. We wish to convert the vector $\mathbf{v}$ into a scalar u. If we were to choose $\mathbf{v}$ itself as the weight vector, then we would have $\mathbf{v} \cdot \mathbf{v} = 1$. Since we wish the scalar u, not 1, we choose $\mathbf{v}$ multiplied by u, which gives the desired result. This can be seen using simple algebra as follows:

$$\begin{aligned} \mathbf{w} \cdot \mathbf{v} &= (u\mathbf{v}) \cdot \mathbf{v} \\ &= u(\mathbf{v} \cdot \mathbf{v}) \\ &= u. \end{aligned}$$

Geometrically, the problem of finding $\mathbf{w}$ corresponds to finding a vector whose projection on $\mathbf{v}$ is u. As shown in Figure 24, any vector along the dotted line will work, because each such vector projects to the same place on $\mathbf{v}$. Our solution involved making the simple choice of the vector that points in the same direction as $\mathbf{v}$.

It is not difficult to generalize to the case of an output vector $\mathbf{u}$ with more than one component. To do so, let us consider the PDP system of Figure 18. Each output unit has a weight vector, and these weight vectors form the rows of the weight matrix $\mathbf{W}$. As discussed earlier, each unit calculates the inner product between its weight vector and the

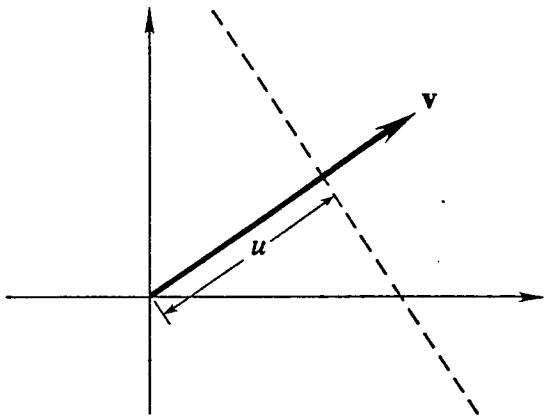

FIGURE 24.

input vector $\mathbf{v}$, and these inner products are the components of the output vector $\mathbf{u}$. To implement a learning scheme, we need to be able to choose weight vectors that produce the desired components of $\mathbf{u}$. Clearly, for each component, we can use the scheme already described for the single unit model above. In other words, the ith weight vector should be given by

$$\mathbf{w}_i = u_i\mathbf{v}. \tag{22}$$

The ith unit will then produce the ith component of $\mathbf{u}$ when presented with $\mathbf{v}$. Thus, the system as a whole will produce the vector $\mathbf{u}$ when presented with $\mathbf{v}$. We now would like a way to write a matrix $\mathbf{W}$ whose rows are given by Equation 22. This is done by noting that Equation 22 is a set of equations calculating the outer product of $\mathbf{u}$ and $\mathbf{v}$. Thus, $\mathbf{W}$ can be written as follows:

$$\mathbf{W} = \mathbf{u}\mathbf{v}^T.$$

We can check the correctness of this choice for $\mathbf{W}$ as follows:

$$\begin{aligned}\mathbf{W}\mathbf{v} &= (\mathbf{u}\mathbf{v}^T)\mathbf{v} \\ &= \mathbf{u}\,(\mathbf{v}^T\mathbf{v}) \\ &= \mathbf{u}\end{aligned}$$

using the fact that $\mathbf{v}$ is of length one in making the last step.

The fact that $\mathbf{W}$ is an outer product has important implications for the implementation of Hebbian learning in PDP networks. As discussed

in the previous section, the i,jth element of $\mathbf{W}$ is equal to the product $u_i v_j$, which is the product of the activation of the jth input unit and the ith output unit. Both of these quantities are available in a physically circumscribed area on the link joining these two units. Thus, the weight on that link can be changed by autonomous local processes. The Hebb rule is often referred to as a *local* learning rule for this reason.

To summarize, we have established a procedure for finding a matrix $\mathbf{W}$ which will associate any particular pair of input and output vectors. Clearly for every pair of vectors, we can find a different weight matrix to perform the association. What is less obvious is that the same matrix can be used for several pairs of associations. Let us assume that we are given n n-dimensional output vectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$ which we want to associate with n n-dimensional input vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$. In other words, for each i, we wish to have

$$\mathbf{u}_i = \mathbf{W}\mathbf{v}_i.$$

Let us further assume that the vectors $\mathbf{v}_i$ form a mutually orthogonal set and that each vector $\mathbf{v}_i$ is of unit length. That is, we assume

$$\mathbf{v}_i{}^T\mathbf{v}_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases}$$

We now form a set of matrices $\mathbf{W}_i$ using the learning scheme developed above:

$$\mathbf{W}_i = \mathbf{u}_i\mathbf{v}_i^T$$

Finally, we form a composite weight matrix $\mathbf{W}$ which is the sum of the $\mathbf{W}_i$:

$$\mathbf{W} = \mathbf{W}_1 + \cdots + \mathbf{W}_i + \cdots + \mathbf{W}_n.$$

We already know that, for example, $\mathbf{W}_1$ above will associate $\mathbf{v}_1$ and $\mathbf{u}_1$. It is also true that $\mathbf{W}$ will perform all such associations. Thus, for arbitrary i:

$$\begin{aligned}
\mathbf{W}\mathbf{v}_i &= (\mathbf{W}_1 + \cdots + \mathbf{W}_i + \cdots + \mathbf{W}_n)\mathbf{v}_i \\
&= (\mathbf{u}_1\mathbf{v}_1^T + \cdots + \mathbf{u}_i\mathbf{v}_i^T + \cdots + \mathbf{u}_n\mathbf{v}_n^T)\mathbf{v}_i \\
&= (\mathbf{u}_1\mathbf{v}_1^T)\mathbf{v}_i + \cdots + (\mathbf{u}_i\mathbf{v}_i^T)\mathbf{v}_i + \cdots + (\mathbf{u}_n\mathbf{v}_n^T)\mathbf{v}_i \\
&= \mathbf{u}_1(\mathbf{v}_1^T\mathbf{v}_i) + \cdots + \mathbf{u}_i(\mathbf{v}_i^T\mathbf{v}_i) + \cdots + \mathbf{u}_n(\mathbf{v}_n^T\mathbf{v}_i) \\
&= 0 + \cdots + \mathbf{u}_i(\mathbf{v}_i^T\mathbf{v}_i) + \cdots + 0 \\
&= \mathbf{u}_i.
\end{aligned}$$

The property of orthogonality was crucial here, because it forced the disappearance of all terms involving vectors other than $\mathbf{u}_i$ in the next to last step. The reader may find it useful to justify the steps in this derivation.

When the set of input vectors is not orthogonal, the Hebb rule will not correctly associate output vectors with input vectors. However, a modification of the Hebb rule, known as the *delta rule*, or the *Widrow-Hoff rule*, can make such associations. The requirement for the delta rule to work is that the input vectors be linearly independent. The delta rule is discussed further in Chapter 11, and at length in Kohonen (1977).

Earlier it was discussed how, at least for square matrices, knowledge of the eigenvectors of a matrix permits an important simplification to be made. The matrix multiplication of a vector can be replaced by scalar multiplication (cf. Equation 21). I will now show that the Hebbian learning scheme fits nicely with the notion of eigenvectors. Suppose that we wish to associate vectors with scalar copies of themselves. This is what is done, for example, in an auto-associator like those discussed in J. A. Anderson et al. (1977); see Chapters 2 and 17. In other words, we want the vectors $\mathbf{u}_i$ to be of the form $\lambda_i \mathbf{v}_i$ where $\mathbf{v}_i$ are the input vectors. Let us further assume that the n scalars λ_i are distinct. Using the outer product learning rule, we have

$$\mathbf{W} = \mathbf{W}_1 + \cdots + \mathbf{W}_i + \cdots + \mathbf{W}_n$$

where

$$\mathbf{W}_i = \mathbf{u}_i \mathbf{v}_i^T = \lambda_i \mathbf{v}_i \mathbf{v}_i^T.$$

If we now present the vector $\mathbf{v}_i$ to the matrix $\mathbf{W}$ thus formed, we have

$$\begin{aligned}
\mathbf{W}\mathbf{v}_i &= (\mathbf{W}_1 + \cdots + \mathbf{W}_i + \cdots + \mathbf{W}_n)\mathbf{v}_i \\
&= (\lambda_1 \mathbf{v}_1 \mathbf{v}_1^T + \cdots + \lambda_i \mathbf{v}_i \mathbf{v}_i^T + \cdots + \lambda_n \mathbf{v}_n \mathbf{v}_n^T)\mathbf{v}_i \\
&= 0 + \cdots + \lambda_i \mathbf{v}_i (\mathbf{v}_i^T \mathbf{v}_i) + \cdots + 0 \\
&= \lambda_i \mathbf{v}_i.
\end{aligned}$$

This equation shows that $\mathbf{v}_i$ is an eigenvector of $\mathbf{W}$ with eigenvalue λ_i.

Let me summarize. When we calculate a weight matrix $\mathbf{W}$ using the Hebbian learning rule and associate input vectors to scalar multiples of themselves, then those input vectors are the eigenvectors of $\mathbf{W}$. It is important to note that the matrix $\mathbf{W}$ need not even be calculated—as was stated in the section on eigenvectors, once we have the eigenvectors and eigenvalues of a matrix, we can throw away the matrix. All input-output computations can be done by using Equation 21. This

approach is in contrast to a scheme in which we first calculate a matrix **W** from the input vectors, and then calculate the eigenvectors from the matrix **W**. Here, the eigenvectors are available in the statement of the problem.

Why should one want to associate vectors with scalar copies of themselves? Essentially, the answer is that a system which learns in this way will exhibit the desirable property of *completion*. That is, when partial versions of previously learned vectors are presented to the system, it will be able to produce the whole vector. Readers desiring more details on how this is done should consult Anderson et al. (1977).

MATRIX INVERSES

Throughout this chapter, I have discussed the linear vector equation $\mathbf{u} = \mathbf{W}\mathbf{v}$. First, I discussed the situation in which $\mathbf{v}$ was a known vector and $\mathbf{W}$ a known matrix. This corresponds to knowing the input to a system and its matrix, and wanting to know the output of the system. Next, I discussed the situation in which $\mathbf{v}$ and $\mathbf{u}$ were known vectors, and a matrix $\mathbf{W}$ was desired to associate the two vectors. This is the learning problem discussed in the previous section. Finally, in this section, I discuss the case in which both $\mathbf{u}$ and $\mathbf{W}$ are known, but $\mathbf{v}$ is unknown. There are many situations in which this problem arises, including the change of basis discussed in the next section.

As we will see, the solution to this problem involves the concept of a *matrix inverse*. Let us first assume that we are dealing with square matrices. The inverse of a matrix $\mathbf{W}$, if it exists, is another matrix denoted $\mathbf{W}^{-1}$ that obeys the following equations:

$$\mathbf{W}^{-1}\mathbf{W} = \mathbf{I}$$

$$\mathbf{W}\mathbf{W}^{-1} = \mathbf{I}$$

where $\mathbf{I}$ is the identity matrix.

Example:

$$\mathbf{W} = \begin{bmatrix} 1 & \frac{1}{2} \\ -1 & 1 \end{bmatrix} \qquad \mathbf{W}^{-1} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} \\ \frac{2}{3} & \frac{2}{3} \end{bmatrix}$$

$$\mathbf{W}\mathbf{W}^{-1} = \begin{bmatrix} 1 & \frac{1}{2} \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} \\ \frac{2}{3} & \frac{2}{3} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{W}^{-1}\mathbf{W} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} \\ \frac{2}{3} & \frac{2}{3} \end{bmatrix} \begin{bmatrix} 1 & \frac{1}{2} \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

A good discussion of how to calculate a matrix inverse can be found in Strang (1976).

Let us now show that the matrix inverse is the tool we need to solve the equation $\mathbf{u} = \mathbf{W}\mathbf{v}$, where $\mathbf{v}$ is the unknown. We multiply both sides of the equation by $\mathbf{W}^{-1}$, which yields

$$\begin{aligned} \mathbf{W}^{-1}\mathbf{u} &= \mathbf{W}^{-1}\mathbf{W}\mathbf{v} \\ &= \mathbf{I}\mathbf{v} \\ &= \mathbf{v}. \end{aligned}$$

Thus the solution of the equation simply involves multiplying $\mathbf{u}$ by the matrix $\mathbf{W}^{-1}$.

Example. We wish to find the vector $\mathbf{v}$ that satisfies the equation

$$\begin{bmatrix} 1 & \frac{1}{2} \\ -1 & 1 \end{bmatrix} \mathbf{v} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}.$$

To do so, we use the matrix $\mathbf{W}^{-1}$ given above:

$$\mathbf{v} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} \\ \frac{2}{3} & \frac{2}{3} \end{bmatrix} \begin{bmatrix} 3 \\ 3 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \end{bmatrix}.$$

We can now check the result as follows:

$$\begin{bmatrix} 1 & \frac{1}{2} \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 4 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}.$$

It is important to realize that $\mathbf{W}^{-1}$, despite the new notation, is simply a matrix like any other. Furthermore, the equation $\mathbf{v} = \mathbf{W}^{-1}\mathbf{u}$ is nothing more than a linear mapping of the kind we have studied throughout this chapter. The domain of this mapping is the range of

$\mathbf{W}$, and the range of the mapping is the domain of $\mathbf{W}$. This inverse relationship is shown in Figure 25. The fact that $\mathbf{W}^{-1}$ represents a function from one vector space to another has an important consequence. For every $\mathbf{u}$ in the domain of $\mathbf{W}^{-1}$, there can be only one $\mathbf{v}$ in the range such that $\mathbf{v} = \mathbf{W}^{-1}\mathbf{u}$. This is true because of the definition of a function. Now let us look at the consequence of this fact from the point of view of the mapping represented by $\mathbf{W}$. If $\mathbf{W}$ maps any two distinct points $\mathbf{v}_1$ and $\mathbf{v}_2$ in its domain to the same point $\mathbf{u}$ in its range, that is, if $\mathbf{W}$ is not one-to-one, then there can be no $\mathbf{W}^{-1}$ to represent the inverse mapping.

We now wish to characterize matrices that can map distinct points in the domain to a single point in the range, for these are the matrices that do not have inverses. To do so, first recall that one way to view the equation $\mathbf{u} = \mathbf{W}\mathbf{v}$ is that $\mathbf{u}$ is a linear combination of the column vectors of $\mathbf{W}$. The coefficients of the linear combination are the components of $\mathbf{v}$. Thus, there is more than one $\mathbf{v}$ which maps to the same point $\mathbf{u}$ exactly in the case in which there is more than one way to write $\mathbf{u}$ as a linear combination of the column vectors of $\mathbf{W}$. These are completely equivalent statements. As discussed earlier, we know that a vector $\mathbf{u}$ can be written as a unique linear combination of a set of vectors only in the case where the vectors are linearly independent. Otherwise, if the vectors are linearly dependent, then there are an infinite number of ways to write $\mathbf{u}$ as a linear combination. Therefore, we have the result that a matrix has an inverse only if its column vectors are linearly independent.

For square matrices with linearly dependent column vectors and for non-square matrices, it is possible to define an inverse called the *generalized inverse*, which performs part of the inverse mapping. In the case in which an infinite number of points map to the same point, there will be an infinite number of generalized inverses for a particular matrix, each of which will map from the point in the range to one of the points in the domain.

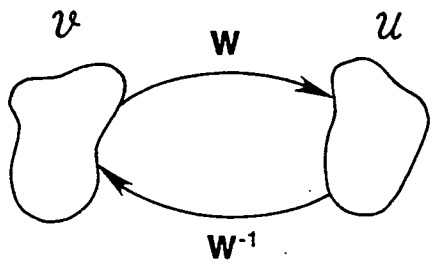

FIGURE 25.

In summary, the matrix inverse $\mathbf{W}^{-1}$ can be used to solve the equation $\mathbf{u} = \mathbf{W}\mathbf{v}$, where $\mathbf{v}$ is the unknown, by multiplying $\mathbf{u}$ by $\mathbf{W}^{-1}$. The inverse exists only when the column vectors of $\mathbf{W}$ are linearly independent. Let me mention in passing that the maximum number of linearly independent column vectors of a matrix is called the *rank* of the matrix.[7] An $n \times n$ matrix is defined to have *full rank* if the rank is equal to n. Thus, the condition that a matrix have an inverse is equivalent to the condition that it have full rank.

CHANGE OF BASIS

As was discussed earlier, a basis for a vector space is a set of linearly independent vectors that span the space. Although we most naturally tend to think in terms of the standard basis, for a variety of reasons it is often convenient to change the basis. For example, some relationships between vectors or operations on vectors are easier to describe when a good choice of basis has been made. To make a change of basis, we need to be able to describe the vectors and matrices we are using in terms of the new basis. In this section, I use the results of the previous section to discuss the problems that arise under a change of basis. I also discuss some of the implications of a change of basis for linear PDP models.

The numbers that are used to represent a vector, it should be remembered, are relative to a particular choice of basis. When we change the basis, these numbers, which we refer to as *coordinates*, change. Our first task, then, is to find a way to relate the coordinates in a new basis to the coordinates in the old basis. Let me begin with an example. In Figure 26, there is a vector $\mathbf{v}$, which in the standard basis has the coordinates $\begin{bmatrix} 2 \\ 1 \end{bmatrix}$. We now change basis by choosing two new basis vectors, $\mathbf{y}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$ and $\mathbf{y}_2 = \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix}$. As shown in Figure 27, $\mathbf{v}$ can be written as a linear combination of $\mathbf{y}_1$ and $\mathbf{y}_2$. It turns out, as we shall see below, that the coefficients 1 and 2 are the correct coefficients of $\mathbf{y}_1$ and $\mathbf{y}_2$ in the linear combination. Let the symbol $\mathbf{v}^*$ represent $\mathbf{v}$ in the new basis. Thus, $\mathbf{v}^* = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$.

[7] An important theorem in linear algebra establishes that, for any matrix, the maximum number of linearly independent column vectors is equal to the maximum number of linearly independent row vectors. Thus, the rank can be taken as either.

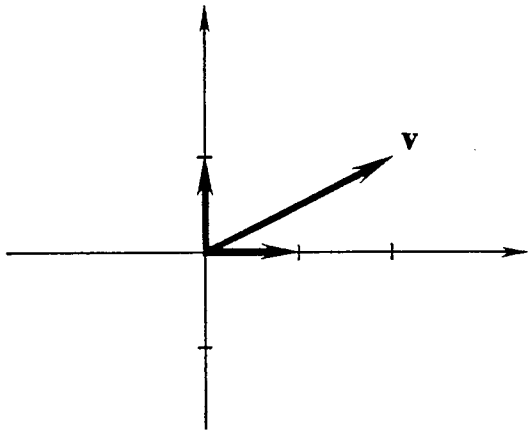

FIGURE 26.

We now want to show how to find the coordinates of a vector $\mathbf{v}$ in a new basis $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n$. These coordinates are simply the coefficients c_i in the equation

$$\mathbf{v} = c_1\mathbf{y}_1 + c_2\mathbf{y}_2 + \cdots + c_n\mathbf{y}_n. \tag{23}$$

Let us form a matrix $\mathbf{Y}$ whose columns are the new basis vectors $\mathbf{y}_i$, and let $\mathbf{v}^*$ be the vector whose components are the c_i. Then Equation 23 is equivalent to the following equation:

$$\mathbf{v} = \mathbf{Y}\mathbf{v}^* \tag{24}$$

where $\mathbf{v}^*$ is the unknown. The solution to the problem is now clear: we use the inverse matrix $\mathbf{Y}^{-1}$ to calculate the unknown vector as in the previous section:

$$\mathbf{v}^* = \mathbf{Y}^{-1}\mathbf{v}.$$

Example. Letting $\mathbf{y}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$ and $\mathbf{y}_2 = \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix}$, we have $\mathbf{Y} = \begin{bmatrix} 1 & \frac{1}{2} \\ -1 & 1 \end{bmatrix}$ and $\mathbf{Y}^{-1} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} \\ \frac{2}{3} & \frac{2}{3} \end{bmatrix}$.

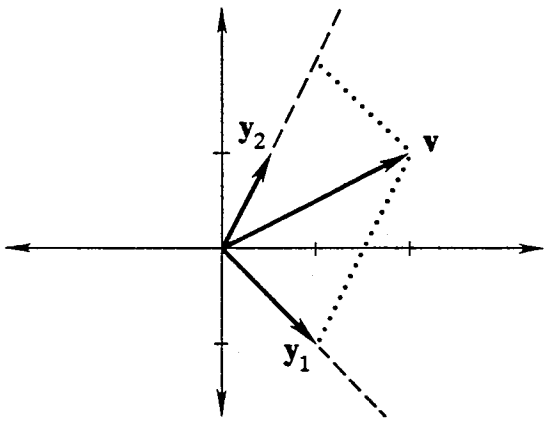

FIGURE 27.

Thus,

$$\mathbf{v}^* = \mathbf{Y}^{-1}\mathbf{v} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} \\ \frac{2}{3} & \frac{2}{3} \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

Notice that we have also solved the inverse problem along the way. That is, suppose that we know the coordinates $\mathbf{v}^*$ in the new basis, and we wish to find the coordinates $\mathbf{v}$ in the old basis. This transformation is that shown in Equation 24: We simply multiply the vector of new coordinates by $\mathbf{Y}$.

We have shown how to represent vectors when the basis is changed. Now, let us accomplish the same thing for matrices. Let there be a square matrix $\mathbf{W}$ that transforms vectors in accordance with the equation $\mathbf{u} = \mathbf{W}\mathbf{v}$. Suppose we now change basis and write $\mathbf{v}$ and $\mathbf{u}$ in the new basis as $\mathbf{v}^*$ and $\mathbf{u}^*$. We want to know if there is a matrix that does the same thing in the new basis as $\mathbf{W}$ did in the original basis. In other words, we want to know if there is a matrix $\mathbf{W}^*$ such that $\mathbf{u}^* = \mathbf{W}^*\mathbf{v}^*$. This is shown in the diagram in Figure 28, where it should be remembered that $\mathbf{v}$ and $\mathbf{v}^*$ (and $\mathbf{u}$ and $\mathbf{u}^*$) are really the same vector, just described in terms of different basis vectors.

To see how to find $\mathbf{W}^*$, consider a somewhat roundabout way of solving $\mathbf{u}^* = \mathbf{W}^*\mathbf{v}^*$. We can convert $\mathbf{v}^*$ back to the original basis, then map from $\mathbf{v}$ to $\mathbf{u}$ using the matrix $\mathbf{W}$, and finally convert $\mathbf{u}$ to $\mathbf{u}^*$.

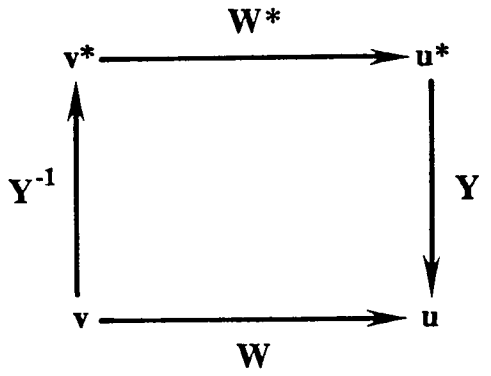

FIGURE 28.

Luckily, we already know how to make each of these transformations—they are given by the equations:

$$\mathbf{v} = \mathbf{Y}\mathbf{v}^*$$

$$\mathbf{u} = \mathbf{W}\mathbf{v}$$

$$\mathbf{u}^* = \mathbf{Y}^{-1}\mathbf{u}.$$

Putting these three equations together, we have

$$\begin{aligned}\mathbf{u}^* &= \mathbf{Y}^{-1}\mathbf{u} \\ &= \mathbf{Y}^{-1}\mathbf{W}\mathbf{v} \\ &= \mathbf{Y}^{-1}\mathbf{W}\mathbf{Y}\mathbf{v}^*.\end{aligned}$$

Thus, $\mathbf{W}^*$ must be equal to $\mathbf{Y}^{-1}\mathbf{W}\mathbf{Y}$. Matrices related by an equation of the form $\mathbf{W}^* = \mathbf{Y}^{-1}\mathbf{W}\mathbf{Y}$ are called *similar*.

One aspect of this discussion needs further elaboration. We have been treating matrices as linear operators on a vector space. However, as the results of this section make clear, a matrix is tied to a particular basis. That is, the numbers in the matrix are just as arbitrary as the numbers used for representing vectors. When the basis changes, the numbers change according to the equation $\mathbf{W}^* = \mathbf{Y}^{-1}\mathbf{W}\mathbf{Y}$. The underlying mapping, which remains the same when the matrix $\mathbf{W}$ is used in the original basis and the matrix $\mathbf{W}^*$ is used in the new basis, is called a *linear transformation*. The same linear transformation is represented by different matrices in different bases.

It is interesting to recast the results on eigenvectors in terms of a change of basis. For some matrix $\mathbf{W}$, let us consider changing basis to

the eigenvectors of $\mathbf{W}$. Let us find the matrix $\mathbf{W}^*$ in the new basis. For each eigenvector $\mathbf{y}_i$, by definition

$$\mathbf{W}\mathbf{y}_i = \lambda_i \mathbf{y}_i. \tag{25}$$

If $\mathbf{Y}$ is a matrix whose columns are the $\mathbf{y}_i$, then we can write Equation 25 for all of the eigenvectors at once as follows (cf. Figure 21):

$$\mathbf{W}\mathbf{Y} = \mathbf{Y}\Lambda$$

where Λ is a diagonal matrix whose entries on the main diagonal are the eigenvalues λ_i. You should try to convince yourself of the correctness of this equation, particularly the placement of Λ. Now premultiply both sides by $\mathbf{Y}^{-1}$ to give

$$\mathbf{Y}^{-1}\mathbf{W}\mathbf{Y} = \Lambda.$$

Thus, the matrix $\mathbf{W}^*$ is equal to Λ. In other words, when we use the eigenvectors of $\mathbf{W}$ as the new basis, the matrix corresponding to $\mathbf{W}$ in the new basis is a diagonal matrix whose entries are the eigenvalues. This is really nothing more than a restatement of the earlier results on eigenvectors, but seen in a different perspective.

It is worthwhile to consider the implications of a change of basis for PDP models. How does the behavior of the model depend on the basis that is chosen? This question is discussed in depth in Chapter 22. For now, let us simply note that the linear structure of a set of vectors remains the same over a change of basis. That is, if a vector can be written as a linear combination of a set of vectors in one basis, then it can be written as the same linear combination of those vectors in all bases. For example, let $\mathbf{w} = a\mathbf{v}_1 + b\mathbf{v}_2$. Let $\mathbf{Y}$ be the matrix of a change of basis. Then we have

$$\begin{aligned}
\mathbf{w}^* &= \mathbf{Y}^{-1}\mathbf{w} \\
&= \mathbf{Y}^{-1}(a\mathbf{v}_1 + b\mathbf{v}_2) \\
&= a\mathbf{Y}^{-1}\mathbf{v}_1 + b\mathbf{Y}^{-1}\mathbf{v}_2 \\
&= a\mathbf{v}_1^* + b\mathbf{v}_2^*.
\end{aligned}$$

The coefficients in the linear combination are the same in the old and in the new basis. The equations show that this result holds because change of basis is a *linear* operation.

The behavior of a linear PDP model depends entirely on the linear structure of the input vectors. That is, if $\mathbf{w} = a\mathbf{v}_1 + b\mathbf{v}_2$, then the response of the system to $\mathbf{w}$ is determined by its response to $\mathbf{v}_1$ and $\mathbf{v}_2$ and the coefficients a and b. The fact that a change of basis preserves

the linear structure of the vectors shows that it is this linear structure that is relevant to the behavior of the model, and not the particular basis chosen to describe the vectors.

NONLINEAR SYSTEMS

The use of nonlinearity occurs throughout this book and throughout the literature on parallel distributed processing systems (Anderson et al., 1977; Grossberg, 1978; Hopfield, 1982; Kohonen, 1977). In this section, I will indicate some of the reasons why nonlinearities are deemed necessary.[8] Although these reasons are based on the desire for behaviors outside the domain of linear models, it should be stated that linear systems have a great deal of power in themselves, and that many of the nonlinearities represent comparatively small changes to underlying models which are linear. Other models are more fundamentally nonlinear. Further discussions of nonlinear mathematics can be found in Chapters 10 and 22.

One simple nonlinearity has already arisen in the discussion of a PDP system with one output unit. Such a system computes the inner product of its weight vector and the input vector. This is a linear system, given the linearity of the inner product. The geometrical properties of the inner product led us to picture the operation of this system as computing the closeness of input vectors to the weight vector in space.

Suppose we draw a line perpendicular to the weight vector at some point, as in Figure 29. Since all vectors on this line project to the same point on the weight vector, their inner products with the weight vector are equal. Furthermore, all vectors to the left of this line have a smaller inner product, and all vectors to the right have a larger inner product. Let us choose a fixed number as a *threshold* for the unit by requiring that if the inner product is greater than the threshold, the unit outputs a 1, otherwise it outputs a 0. Such a unit breaks the space into two parts by producing a different response to vectors in the two parts.

This use of a threshold is natural in using the unit to classify patterns as belonging to one group or another. The essential point is that the threshold permits the unit to make a *decision*. Other units in a larger

[8] Since nonlinear systems in general are systems that are defined as "not linear," it is important to understand clearly what "linear" means. A review of the section on linearity may be necessary before proceeding.

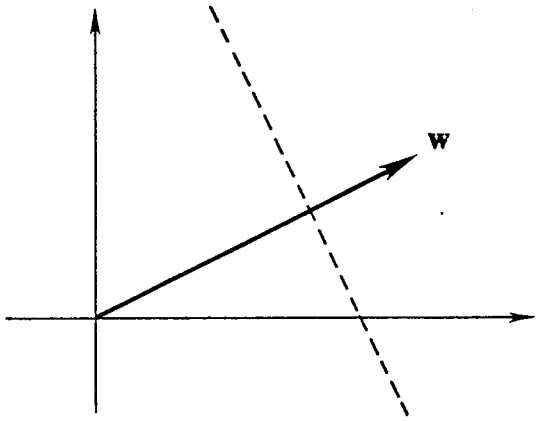

FIGURE 29.

system that take their input from this unit could choose completely different behaviors based on the decision. Notice also that the unit is a categorizer: All input vectors that are on the same side of the space lead to the same response.

To introduce a threshold into the mathematical description of the processing unit, it is necessary to distinguish between the activation of the unit and its output. A function relating the two quantities is shown in Figure 30. It produces a one or a zero based on the magnitude of the activation. It is also possible to have a probabilistic threshold. In this case, the farther the activation is above the threshold, the more

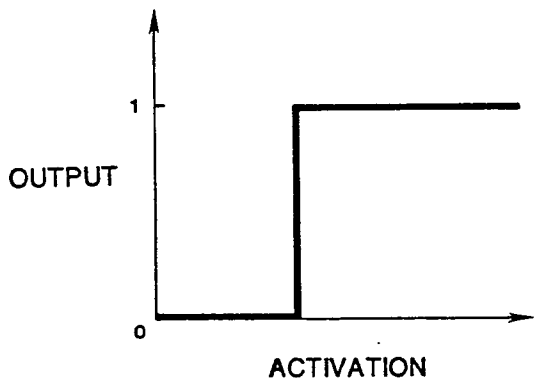

FIGURE 30.

likely the unit is to have an output of one, and the farther the activation is below the threshold, the more likely the unit is to have an output of zero. Units such as these are discussed in Chapters 6 and 7.

The threshold unit is a good example of many of the nonlinearities that are to be found in PDP models. An underlying linear model is modified with a nonlinear function relating the output of a unit to its activation. Another related example of such a nonlinearity is termed *subthreshold summation*. It is often observed in biological systems that two stimuli presented separately to the system provoke no response, although when presented simultaneously, a response is obtained. Furthermore, once the system is responding, further stimuli are responded to in a linear fashion. Such a system can be modeled by endowing a linear PDP unit with the nonlinear output function in Figure 31. Note that only if the sum of the activations produced by vectors exceeds T will a response be produced. Also, there is a *linear range* in which the system responds linearly. It is often the case in nonlinear systems that there is such a linear range, and the system can be treated as linear provided that the inputs are restricted to this linear range.

One reason why subthreshold summation is desirable is that it suppresses noise. The system will not respond to small random inputs that are assumed to be noise.

All physical systems have a limited *dynamic range*. That is, the response of the system cannot exceed a certain maximum response. This fact can be modeled with the output function in Figure 32, which shows a linear range followed by a cutoff. The system will behave linearly until the output reaches M, at which point no further increase can occur. In Figure 33, a nonlinear function is shown which also has a

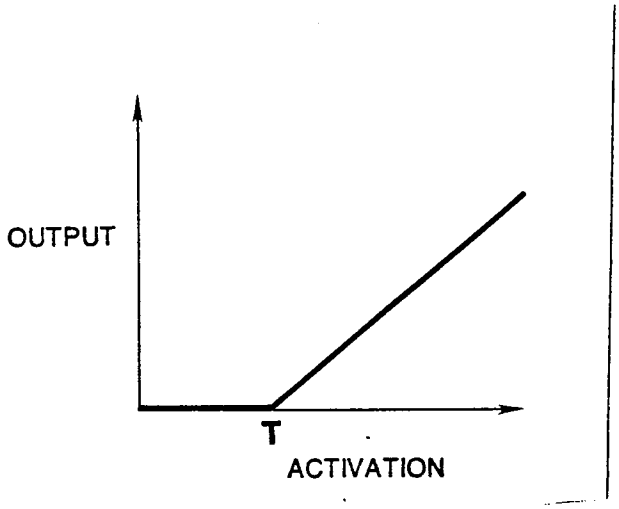

FIGURE 31.

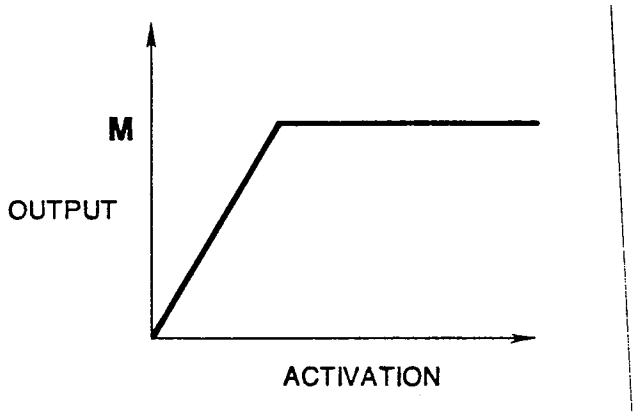

FIGURE 32.

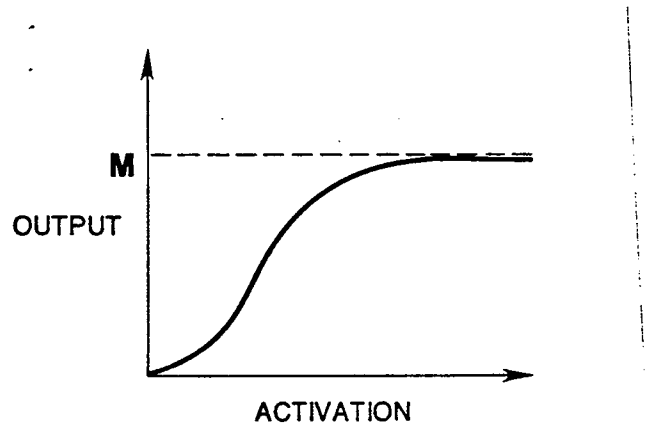

FIGURE 33.

maximum output M. This curve, called a *sigmoid*, is a sort of hybrid between Figure 31 and Figure 32. It combines noise suppression with a limited dynamic range. Chapter 8 shows how such units are necessary for certain kinds of interesting behavior to arise in layered networks.

To summarize, I have described some of the ways in which linear systems are modified to produce nonlinear systems that exhibit certain desired behaviors. All of these systems have an important linear component and are sometimes referred to as *semilinear*. Furthermore, several of the systems have a linear range in which the nonlinearities can be ignored. The next chapter discusses more fundamentally nonlinear systems.

FURTHER READING

Halmos, P. R. (1974). *Finite-dimensional vector spaces.* New York: Springer-Verlag. For the more mathematically minded. An excellent account of linear algebra from an abstract point of view.

Kohonen, T. (1977). *Associative memory: A system theoretic approach.* Berlin: Springer-Verlag. This book has a short tutorial on linear algebra. The discussion of associative memory depends heavily on the mathematics of linear algebra.

Strang, G. (1976). *Linear algebra and its applications.* New York: Academic Press. A general textbook treating most of the essentials of linear algebra. It is especially good in its treatment of computational topics. A good place to find out about calculating matrix inverses and eigenvalues.