The Kohonen neuron with the largest NET value is the "winner." Its output is set to one; all others are set to zero.

Grossberg Layer

The Grossberg layer functions in a familiar manner. Its NET output is the weighted sum of the Kohonen layer outputs $k_1, k_2, \ldots, k_n$, forming the vector **K**. The connecting weight vector designated **V** consists of the weights $v_{11}, v_{21}, \ldots, v_{np}$. The NET output of each Grossberg neuron is then

$$NET_j = \sum_i k_i w_{ij} \tag{4-4}$$

where NET_j is the output of Grossberg neuron j, or in vector form

$$\mathbf{Y} = \mathbf{KV} \tag{4-5}$$

where

Y = the Grossberg-layer output vector
K = the Kohonen-layer output vector
V = the Grossberg layer weight matrix

If the Kohonen layer is operated such that only one neuron's NET is at one and all others are at zero, only one element of the **K** vector is nonzero, and the calculation is simple. In fact, the only action of each neuron in the Grossberg layer is to output the value of the weight that connects it to the single nonzero Kohonen neuron.

TRAINING THE KOHONEN LAYER

The Kohonen layer classifies the input vectors into groups that are similar. This is accomplished by adjusting the Kohonen layer weights so that similar input vectors activate the same Kohonen neuron. It is then the responsibility of the Grossberg layer to produce the desired outputs.

Kohonen training is a self-organizing algorithm that operates in

the unsupervised mode. For this reason, it is difficult (and unnecessary) to predict which specific Kohonen neuron will be activated for a given input vector. It is only necessary to ensure that training separates dissimilar input vectors.

Preprocessing the Input Vectors

It is highly desirable (but not mandatory) to normalize all input vectors before applying them to the network. This is done by dividing each component of an input vector by that vector's length. This length is found by taking the square root of the sum of the squares of all of the vector's components. In symbols

$$x_i' = x_i/(x_1^2 + x_2^2 + \ldots + x_n^2)^{1/2} \tag{4-6}$$

This converts an input vector into a unit vector pointing in the same direction; that is, a vector of unit length in n-dimensional space.

Equation 4-6 generalizes the familiar two-dimensional case in which the length of a vector equals the hypotenuse of the right triangle formed by its x and y components, an application of the familiar Pythagorean theorem. In Figure 4-2a, such a two-dimen-

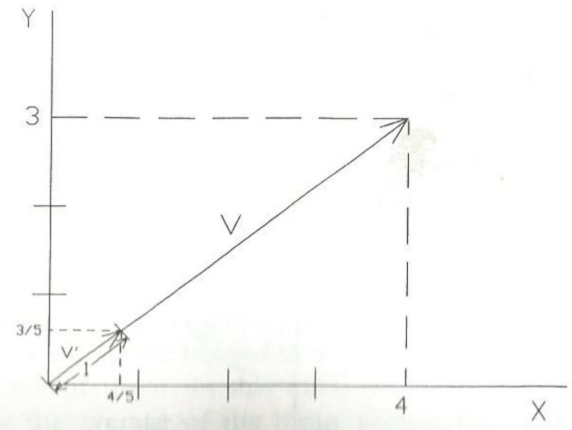

Figure 4-2a. Unit Input Vector