CHAPTER 8
Principal-Components Analysis

Problem 8.5

From Example 8.2 in the text:

$$\lambda_0 = 1 + \sigma^2 \tag{1}$$

$$\mathbf{q}_0 = \mathbf{s} \tag{2}$$

The correlation matrix of the input is

$$\mathbf{R} = \mathbf{s}\mathbf{s}^T + \sigma^2\mathbf{I} \tag{3}$$

where $\mathbf{s}$ is the signal vector and σ^2 is the variance of an element of the additive noise vector. Hence, using (2) and (3):

$$\begin{aligned}
\lambda_0 &= \frac{\mathbf{q}_0^T\mathbf{R}\mathbf{q}_0}{\mathbf{q}_0^T\mathbf{q}_0} \\
&= \frac{\mathbf{s}^T(\mathbf{s}\mathbf{s}^T + \sigma^2\mathbf{I})\mathbf{s}}{\mathbf{s}^T\mathbf{s}} \\
&= \frac{(\mathbf{s}^T\mathbf{s})(\mathbf{s}^T\mathbf{s}) + \sigma^2(\mathbf{s}^T\mathbf{s})}{\mathbf{s}^T\mathbf{s}} \\
&= \mathbf{s}^T\mathbf{s} + \sigma^2 \\
&= \|\mathbf{s}\|^2 + \sigma^2 \qquad (4)
\end{aligned}$$

The vector $\mathbf{s}$ is a signal vector of unit length:

$$\|\mathbf{s}\| = 1$$

Hence, (4) simplifies to

$$\lambda_0 = 1 + \sigma^2$$

which is the desired result given in (1).

Problem 8.6

From (8.46) in the text we have

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta y(n)[\mathbf{x}(n) - y(n)\mathbf{w}(n)] \tag{1}$$

As $n \to \infty$, $\mathbf{w}(n) \to \mathbf{q}_1$, and so we deduce from (1) that

$$\mathbf{x}(n) = y(n)\mathbf{q}_1 \qquad \text{for } n \to \infty \tag{2}$$

where $\mathbf{q}_1$ is the eigenvector associated with the largest eigenvalue λ_1 of the correlation matrix $\mathbf{R} = \mathbf{E}[\mathbf{x}(n)\mathbf{x}^T(n)]$, where $\mathbf{E}$ is the expectation operator. Multiplying (2) by its own transpose and then taking expectations, we get

$$\mathbf{E}[\mathbf{x}(n)\mathbf{x}^T(n)] = \mathbf{E}[y^2(n)]\mathbf{q}_1\mathbf{q}_1^T$$

Equivalently, we may write

$$\mathbf{R} = \sigma_Y^2\mathbf{q}_1\mathbf{q}_1^T \tag{3}$$

where σ_Y^2 is the variance of the output $y(n)$. Post-multiplying (3) by $\mathbf{q}_1$:

$$\mathbf{R}\mathbf{q}_1 = \sigma_Y^2\mathbf{q}_1\mathbf{q}_1^T\mathbf{q}_1 = \sigma_Y^2\mathbf{q}_1 \tag{4}$$

where it is noted that $\|\mathbf{q}_1\| = 1$ by definition. From (4) we readily see that $\sigma_Y^2 = \lambda_1$, which is the desired result.

Problem 8.7

Writing the learning algorithm for minor components analysis in matrix form:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \eta y(n)[\mathbf{x}(n) - y(n)\mathbf{w}(n)]$$

Proceeding in a manner similar to that described in Section (8.5) of the textbook, we have the nonlinear differential equation:

$$\frac{d}{dt}\mathbf{w}(t) = [\mathbf{w}^T(t)\mathbf{R}\mathbf{w}(t)]\mathbf{w}(t) - \mathbf{R}\mathbf{w}(t)$$

Define

$$\mathbf{w}(t) = \sum_{k=1}^{M} \theta_k(t)\mathbf{q}_k \tag{1}$$

where $\mathbf{q}_k$ is the kth eigenvector of correlation matrix $\mathbf{R} = \mathbf{E}[\mathbf{x}(n)\mathbf{x}^T(n)]$ and the coefficient $\theta_k(t)$ is the projection of $\mathbf{w}(t)$ onto $\mathbf{q}_k$. We may then identify two cases as summarized here:

***Case* I:** $1 \le k < m$

For this first case, we define

$$\alpha_k(t) = \frac{\theta_k(t)}{\theta_m(t)} \qquad \text{for some fixed } m \tag{2}$$

Accordingly, we find that

$$\frac{d\alpha_k(t)}{dt} = -(\lambda_m - \lambda_k)\alpha_k(t) \tag{3}$$

With the eigenvalues of $\mathbf{R}$ arranged in decreasing order:

$\lambda_1 > \lambda_2 > \ldots > \lambda_k > \ldots > \lambda_m > 0$

it follows that $\alpha_k(t) \to 0$ as $t \to \infty$.

***Case* II:** $k = m$

For this second case, we find that

$$\frac{d\theta_m(t)}{dt} = \lambda_m\theta_m(t)(\theta_m^2(t) - 1) \text{ for } t \to \infty \tag{4}$$

Hence, $\theta_m(t) = \pm 1$ as $t \to \infty$.

Thus, in light of the results derived for cases I and II, we deduce from (1) that:

$\mathbf{w}(t) \to \mathbf{q}_m$ = eigenvector associated with the smallest eigenvalue λ_m as $t \to \infty$, and $\sigma_Y^2 = \mathbf{E}[y^2(n)] \to \lambda_m$.

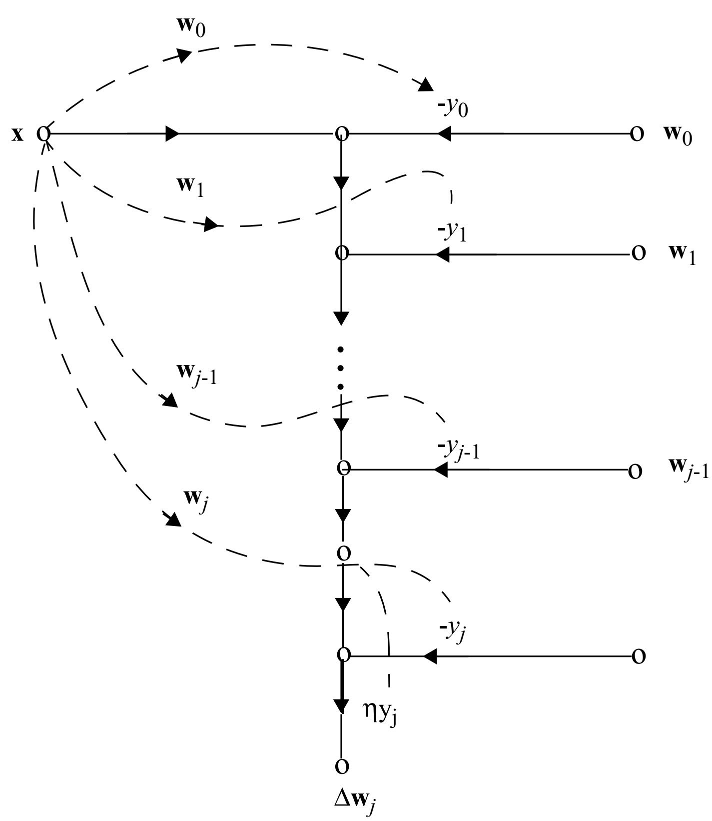
Problem 8.8

From (8.87) and (8.88) of the text:

$$\Delta \mathbf{w}_j = \eta y_j \mathbf{x}' - \eta y_j^2 \mathbf{w}_j \tag{1}$$

$$\mathbf{x}' = \mathbf{x} - \sum_{k=0}^{j-1} \mathbf{w}_k y_k \tag{2}$$

where, for convenience of presentation, we have omitted the dependence on time n. Equations (1) and (2) may be represented by the following vector-valued signal flow graph:



Note: The dashed lines indicate inner (dot) products formed by the input vector $\mathbf{x}$ and the pertinent synaptic weight vectors $\mathbf{w}_0, \mathbf{w}_1, ..., \mathbf{w}_j$ to produce $y_0, y_1, ..., y_j$, respectively.

Problem 8.9

Consider a network consisting of a single layer of neurons with feedforward connections. The algorithm for adjusting the matrix of synaptic weights $\mathbf{W}(n)$ of the network is described by the recursive equation (see Eq. (8.91) of the text):

$$\mathbf{W}(n) = \mathbf{W}(n) + \eta(n)\{\mathbf{y}(n)\mathbf{x}^T(n) - LT[\mathbf{y}(n)\mathbf{y}^T(n)]\mathbf{W}(n)\} \tag{1}$$

where $\mathbf{x}(n)$ is the input vector, $\mathbf{y}(n)$ is the output vector; and $LT[.]$ is a matrix operator that sets all the elements above the diagonal of the matrix argument to zero, thereby making it lower triangular.

First, we note that the asymptotic stability theorem discussed in the text does not apply directly to the convergence analysis of stochastic approximation algorithms involving matrices; it is formulated to apply to vectors. However, we may write the elements of the parameter (synaptic weight) matrix $\mathbf{W}(n)$ in (1) as a vector, that is, one column vector stacked up on top of another. We may then interpret the resulting nonlinear update equation in a corresponding way and so proceed to apply the asymptotic stability theorem directly.

To prove the convergence of the learning algorithm described in (1), we may use the *method of induction* to show that if the first j columns of matrix $\mathbf{W}(n)$ converge to the first j eigenvectors of the correlation matrix $\mathbf{R} = \boldsymbol{E}[\mathbf{x}(n)\mathbf{x}^T(n)]$, then the $(j + 1)$th column will converge to the $(j + 1)$th eigenvector of $\mathbf{R}$. Here we use the fact that in light of the convergence of the maximum eigenfilter involving a single neuron, the first column of the matrix $\mathbf{W}(n)$ converges with probability 1 to the first eigenvector of $\mathbf{R}$, and so on.

Problem 8.10

The results of a computer experiment on the training of a single-layer feedforward network using the generalized Hebbian algorithm are described by Sanger (1990). The network has 16 output neurons, and 4096 inputs arranged as a 64 x 64 grid of pixels. The training involved presentation of 2000 samples, which are produced by low-pass filtering a white Gaussian noise image and then multiplying wi6th a Gaussian window function. The low-pass filter was a Gaussian function with standard deviation of 2 pixels, and the window had a standard deviation of 8 pixels.

Figure 1, presented on the next page, shows the first 16 receptive field masks learned by the network (Sanger, 1990). In this figure, positive weights are indicated by “white” and negative weights are indicated by “black”; the ordering is left-to-right and top-to-bottom.

The results displayed in Fig. 1 are rationalized as follows (Sanger, 1990):

- The first mask is a low-pass filter since the input has most of its energy near dc (zero frequency).
- The second mask cannot be a low-pass filter, so it must be a band-pass filter with a mid-band frequency as small as possible since the input power decreases with increasing frequency.
- Continuing the analysis in the manner described above, the frequency response of successive masks approaches dc as closely as possible, subject (of course) to being orthogonal to previous masks.

The end result is a sequence of orthogonal masks that respond to progressively higher frequencies.

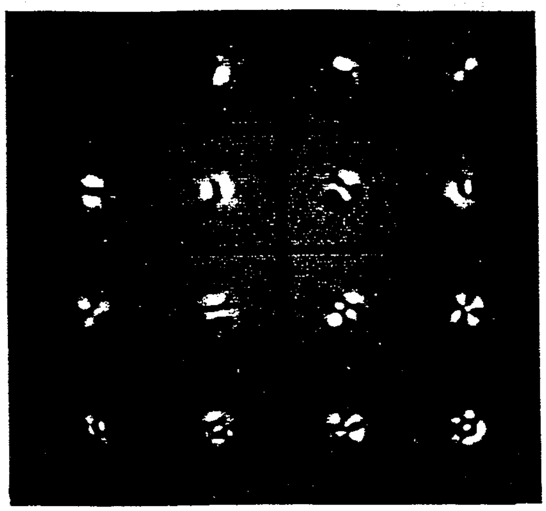


Figure 1: Problem 8.10 (Reproduced with permission of Biological Cybernetics)