our search criteria. The next-best matches come from modules 6 ($y_6 = 0.577$) and 3 ($y_3 = 0.567$). If a document doesn't contain any of the search words, then the corresponding column vector of the database matrix will be orthogonal to the search vector. Note that modules 1 and 7 do not have any of the three search words and consequently

$$y_1 = \mathbf{q}_1^T\mathbf{x} = 0 \quad \text{and} \quad y_7 = \mathbf{q}_7^T\mathbf{x} = 0$$

This example illustrates some of the basic ideas behind database searches. Using modern matrix techniques, we can improve the search process significantly. We can speed up searches and at the same time correct for errors due to polysemy and synonymy. These advanced techniques are referred to as *latent semantic indexing* (LSI) and depend on a matrix factorization, the *singular value decomposition*, which we will discuss in Section 5 of Chapter 6.

There are many other important applications involving angles between vectors. In particular, statisticians use the cosine of the angle between two vectors as a measure of how closely the two vectors are correlated.

APPLICATION 2 Statistics—Correlation and Covariance Matrices

Suppose that we wanted to compare how closely exam scores for a class correlate with scores on homework assignments. As an example, we consider the total scores on assignments and tests of a mathematics class at the University of Massachusetts Dartmouth. The total scores for homework assignments during the semester for the class are given in the second column of Table 2. The third column represents the total scores for the two exams given during the semester, and the last column contains the scores on the final exam. In each case, a perfect score would be 200 points. The last row of the table summarizes the class averages.

Table 2 Math Scores Fall 1996

	Scores		
Student	**Assignments**	**Exams**	**Final**
S1	198	200	196
S2	160	165	165
S3	158	158	133
S4	150	165	91
S5	175	182	151
S6	134	135	101
S7	152	136	80
Average	161	163	131

We would like to measure how student performance compares between each set of exam or assignment scores. To see how closely the two sets of scores are correlated and allow for any differences in difficulty, we need to adjust the scores so that each test has a mean of 0. If, in each column, we subtract the average score from each of the

test scores, then the translated scores will each have an average of 0. Let us store these translated scores in a matrix:

$$X = \begin{bmatrix} 37 & 37 & 65 \\ -1 & 2 & 34 \\ -3 & -5 & 2 \\ -11 & 2 & -40 \\ 14 & 19 & 20 \\ -27 & -28 & -30 \\ -9 & -27 & -51 \end{bmatrix}$$

The column vectors of X represent the deviations from the mean for each of the three sets of scores. The three sets of translated data specified by the column vectors of X all have mean 0, and all sum to 0. To compare two sets of scores, we compute the cosine of the angle between the corresponding column vectors of X. A cosine value near 1 indicates that the two sets of scores are highly correlated. For example, correlation between the assignment scores and the exam scores is given by

$$\cos\theta = \frac{\mathbf{x}_1^T \mathbf{x}_2}{\|\mathbf{x}_1\| \, \|\mathbf{x}_2\|} \approx 0.92$$

A perfect correlation of 1 would correspond to the case where the two sets of translated scores are proportional. Thus, for a perfect correlation, the translated vectors would satisfy

$$\mathbf{x}_2 = \alpha \mathbf{x}_1 \qquad (\alpha > 0)$$

and if the corresponding coordinates of $\mathbf{x}_1$ and $\mathbf{x}_2$ were paired off, then each ordered pair would lie on the line $y = \alpha x$. Although the vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ in our example are not perfectly correlated, the coefficient of 0.92 does indicate that the two sets of scores are highly correlated. Figure 5.1.5 shows how close the actual pairs are to lying on a line $y = \alpha x$. The slope of the line in the figure was determined by setting

$$\alpha = \frac{\mathbf{x}_1^T \mathbf{x}_2}{\mathbf{x}_1^T \mathbf{x}_1} = \frac{2625}{2506} \approx 1.05$$

This choice of slope yields an optimal *least squares* fit to the data points. (See Exercise 7 of Section 5.3.)

If we scale $\mathbf{x}_1$ and $\mathbf{x}_2$ to make them unit vectors

$$\mathbf{u}_1 = \frac{1}{\|\mathbf{x}_1\|}\mathbf{x}_1 \quad \text{and} \quad \mathbf{u}_2 = \frac{1}{\|\mathbf{x}_2\|}\mathbf{x}_2$$

then the cosine of the angle between the vectors will remain unchanged, and it can be computed simply by taking the scalar product $\mathbf{u}_1^T \mathbf{u}_2$. Let us scale all three sets of translated scores in this way and store the results in a matrix:

$$U = \begin{bmatrix} 0.74 & 0.65 & 0.62 \\ -0.02 & 0.03 & 0.33 \\ -0.06 & -0.09 & 0.02 \\ -0.22 & 0.03 & -0.38 \\ 0.28 & 0.33 & 0.19 \\ -0.54 & -0.49 & -0.29 \\ -0.18 & -0.47 & -0.49 \end{bmatrix}$$

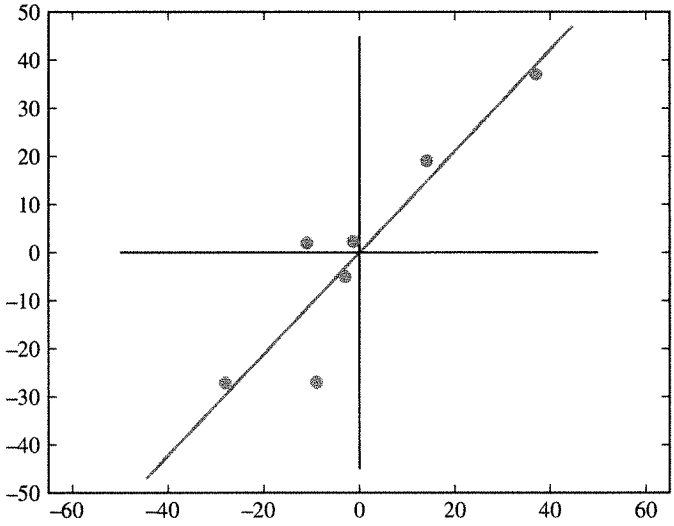


Figure 5.1.5.

If we set $C = U^T U$, then

$$C = \begin{bmatrix} 1 & 0.92 & 0.83 \\ 0.92 & 1 & 0.83 \\ 0.83 & 0.83 & 1 \end{bmatrix}$$

and the (i, j) entry of C represents the correlation between the ith and jth sets of scores. The matrix C is referred to as a *correlation matrix.*

The three sets of scores in our example are all *positively correlated,* since the correlation coefficients are all positive. A negative coefficient would indicate that two data sets were *negatively correlated,* and a coefficient of 0 would indicate that they were *uncorrelated.* Thus, two sets of test scores would be uncorrelated if the corresponding vectors of deviations from the mean were orthogonal.

Another statistically important quantity that is closely related to the correlation matrix is the *covariance matrix.* Given a collection of n data points representing values of some variable x, we compute the mean $\bar{x}$ of the data points and form a vector $\mathbf{x}$ of the deviations from the mean. The *variance,* s^2, is defined by

$$s^2 = \frac{1}{n-1} \sum_1^n x_i^2 = \frac{\mathbf{x}^T \mathbf{x}}{n-1}$$

and the standard deviation s is the square root of the variance. If we have two data sets X_1 and X_2 each containing n values of a variable, we can form vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ of deviations from the mean for both sets. The *covariance* is defined by

$$\text{cov}(X_1, X_2) = \frac{\mathbf{x}_1^T \mathbf{x}_2}{n-1}$$

is denoted by $\mathbb{C}^{m\times n}$. If A and B are elements of $\mathbb{C}^{m\times n}$ and $C \in \mathbb{C}^{n\times r}$, then the following rules are easily verified (see Exercise 9):

I. $(A^H)^H = A$
II. $(\alpha A + \beta B)^H = \overline{\alpha}A^H + \overline{\beta}B^H$
III. $(AC)^H = C^H A^H$

Definition

A matrix M is said to be **Hermitian** if $M = M^H$.

EXAMPLE 2 The matrix

$$M = \begin{bmatrix} 3 & 2-i \\ 2+i & 4 \end{bmatrix}$$

is Hermitian, since

$$M^H = \begin{bmatrix} \overline{3} & \overline{2-i} \\ \overline{2+i} & \overline{4} \end{bmatrix}^T = \begin{bmatrix} 3 & 2-i \\ 2+i & 4 \end{bmatrix} = M$$ ■

If M is a matrix with real entries, then $M^H = M^T$. In particular, if M is a real symmetric matrix, then M is Hermitian. Thus we may view Hermitian matrices as the complex analogue of real symmetric matrices. Hermitian matrices have many nice properties, as we shall see in the next theorem.

Theorem 6.4.1 *The eigenvalues of a Hermitian matrix are all real. Furthermore, eigenvectors belonging to distinct eigenvalues are orthogonal.*

Proof Let A be a Hermitian matrix. Let λ be an eigenvalue of A and let $\mathbf{x}$ be an eigenvector belonging to λ. If $\alpha = \mathbf{x}^H A\mathbf{x}$, then

$$\overline{\alpha} = \alpha^H = (\mathbf{x}^H A\mathbf{x})^H = \mathbf{x}^H A\mathbf{x} = \alpha$$

Thus, α is real. It follows that

$$\alpha = \mathbf{x}^H A\mathbf{x} = \mathbf{x}^H \lambda\mathbf{x} = \lambda\|\mathbf{x}\|^2$$

and hence

$$\lambda = \frac{\alpha}{\|\mathbf{x}\|^2}$$

is real. If $\mathbf{x}_1$ and $\mathbf{x}_2$ are eigenvectors belonging to distinct eigenvalues λ_1 and λ_2, respectively, then

$$(A\mathbf{x}_1)^H\mathbf{x}_2 = \mathbf{x}_1^H A^H\mathbf{x}_2 = \mathbf{x}_1^H A\mathbf{x}_2 = \lambda_2\mathbf{x}_1^H\mathbf{x}_2$$

and

$$(A\mathbf{x}_1)^H\mathbf{x}_2 = (\mathbf{x}_2^H A\mathbf{x}_1)^H = (\lambda_1\mathbf{x}_2^H\mathbf{x}_1)^H = \lambda_1\mathbf{x}_1^H\mathbf{x}_2$$

Consequently,

$$\lambda_1\mathbf{x}_1^H\mathbf{x}_2 = \lambda_2\mathbf{x}_1^H\mathbf{x}_2$$

and since $\lambda_1 \neq \lambda_2$, it follows that

$$\langle \mathbf{x}_2, \mathbf{x}_1 \rangle = \mathbf{x}_1^H \mathbf{x}_2 = 0$$

■

Definition An $n \times n$ matrix U is said to be **unitary** if its column vectors form an orthonormal set in $\mathbb{C}^n$.

Thus U is unitary if and only if $U^H U = I$. If U is unitary, then, since the column vectors are orthonormal, U must have rank n. It follows that

$$U^{-1} = IU^{-1} = U^H U U^{-1} = U^H$$

A real unitary matrix is an orthogonal matrix.

Corollary 6.4.2 *If the eigenvalues of a Hermitian matrix A are distinct, then there exists a unitary matrix U that diagonalizes A.*

Proof Let $\mathbf{x}_i$ be an eigenvector belonging to λ_i for each eigenvalue λ_i of A. Let $\mathbf{u}_i = (1/\|\mathbf{x}_i\|)\mathbf{x}_i$. Thus $\mathbf{u}_i$ is a unit eigenvector belonging to λ_i for each i. It follows from Theorem 6.4.1 that $\{\mathbf{u}_1, \ldots, \mathbf{u}_n\}$ is an orthonormal set in $\mathbb{C}^n$. Let U be the matrix whose ith column vector is $\mathbf{u}_i$ for each i; then U is unitary and U diagonalizes A. ■

EXAMPLE 3 Let

$$A = \begin{bmatrix} 2 & 1-i \\ 1+i & 1 \end{bmatrix}$$

Find a unitary matrix U that diagonalizes A.

Solution

The eigenvalues of A are $\lambda_1 = 3$ and $\lambda_2 = 0$, with corresponding eigenvectors $\mathbf{x}_1 = (1-i, 1)^T$ and $\mathbf{x}_2 = (-1, 1+i)^T$. Let

$$\mathbf{u}_1 = \frac{1}{\|\mathbf{x}_1\|}\mathbf{x}_1 = \frac{1}{\sqrt{3}}(1-i, 1)^T$$

and

$$\mathbf{u}_2 = \frac{1}{\|\mathbf{x}_2\|}\mathbf{x}_2 = \frac{1}{\sqrt{3}}(-1, 1+i)^T$$

Thus

$$U = \frac{1}{\sqrt{3}} \begin{bmatrix} 1-i & -1 \\ 1 & 1+i \end{bmatrix}$$

and

$$U^H A U = \frac{1}{3} \begin{bmatrix} 1+i & 1 \\ -1 & 1-i \end{bmatrix} \begin{bmatrix} 2 & 1-i \\ 1+i & 1 \end{bmatrix} \begin{bmatrix} 1-i & -1 \\ 1 & 1+i \end{bmatrix}$$

$$= \begin{bmatrix} 3 & 0 \\ 0 & 0 \end{bmatrix}$$

■

It follows from Theorem 6.4.4 that each Hermitian matrix A can be factored into a product UDU^H, where U is unitary and D is diagonal. Since U diagonalizes A, it follows that the diagonal elements of D are the eigenvalues of A and the column vectors of U are eigenvectors of A. Thus, A cannot be defective. It has a complete set of eigenvectors that form an orthonormal basis for $\mathbb{C}^n$. This is, in a sense, the ideal situation. We have seen how to express a vector as a linear combination of orthonormal basis elements (Theorem 5.5.2), and the action of A on any linear combination of eigenvectors can easily be determined. Thus, if A has an orthonormal set of eigenvectors $\{\mathbf{u}_1, \ldots, \mathbf{u}_n\}$ and $\mathbf{x} = c_1\mathbf{u}_1 + \cdots + c_n\mathbf{u}_n$, then

$$A\mathbf{x} = c_1\lambda_1\mathbf{u}_1 + \cdots + c_n\lambda_n\mathbf{u}_n$$

Furthermore,

$$c_i = \langle \mathbf{x}, \mathbf{u}_i \rangle = \mathbf{u}_i^H\mathbf{x}$$

or, equivalently, $\mathbf{c} = U^H\mathbf{x}$. Hence,

$$A\mathbf{x} = \lambda_1(\mathbf{u}_1^H\mathbf{x})\mathbf{u}_1 + \cdots + \lambda_n(\mathbf{u}_n^H\mathbf{x})\mathbf{u}_n$$

The Real Schur Decomposition

If A is a real $n \times n$ matrix, then it is possible to obtain a factorization that resembles the Schur decomposition of A, but involves only real matrices. In this case, $A = QTQ^T$ where Q is an orthogonal matrix and T is a real matrix of the form

$$T = \begin{bmatrix} B_1 & \times & \cdots & \times \\ & B_2 & & \times \\ & O & \ddots & \\ & & & B_j \end{bmatrix} \tag{2}$$

where the B_i's are either 1×1 or 2×2 matrices. Each 2×2 block will correspond to a pair of complex conjugate eigenvalues of A. The matrix T is referred to as the *real Schur form* of A. The proof that every real $n \times n$ matrix A has such a factorization depends on the property that, for each pair of complex conjugate eigenvalues of A, there is a two-dimensional subspace of $\mathbb{R}^n$ that is invariant under A.

Definition

A subspace S of $\mathbb{R}^n$ is said to be **invariant** under a matrix A if, for each $\mathbf{x} \in S$, $A\mathbf{x} \in S$.

Lemma 6.4.5 *Let A be a real $n \times n$ matrix with eigenvalue $\lambda_1 = a + bi$ (where a and b are real and $b \neq 0$), and let $\mathbf{z}_1 = \mathbf{x} + i\mathbf{y}$ (where $\mathbf{x}$ and $\mathbf{y}$ are vectors in $\mathbb{R}^n$) be an eigenvector belonging to λ_1. If $S = \text{Span}(\mathbf{x}, \mathbf{y})$, then $\dim S = 2$ and S is invariant under A.*

Proof Since λ is complex, $\mathbf{y}$ must be nonzero; otherwise we would have $A\mathbf{z} = A\mathbf{x}$ (a real vector) equal to $\lambda\mathbf{z} = \lambda\mathbf{x}$ (a complex vector). Since A is real, $\lambda_2 = a - bi$ is also an eigenvalue of A and $\mathbf{z}_2 = \mathbf{x} - i\mathbf{y}$ is an eigenvector belonging λ_2. If there were a scalar c such that $\mathbf{x} = c\mathbf{y}$, then $\mathbf{z}_1$ and $\mathbf{z}_2$ would both be multiples of $\mathbf{y}$ and could not

We may assume that the columns of V have been ordered so that the corresponding eigenvalues satisfy

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$$

The singular values of A are given by

$$\sigma_j = \sqrt{\lambda_j} \qquad j = 1, \ldots, n$$

Let r denote the rank of A. The matrix A^TA will also have rank r. Since A^TA is symmetric, its rank equals the number of nonzero eigenvalues. Thus,

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_r > 0 \qquad \text{and} \qquad \lambda_{r+1} = \lambda_{r+2} = \cdots = \lambda_n = 0$$

The same relation holds for the singular values

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0 \qquad \text{and} \qquad \sigma_{r+1} = \sigma_{r+2} = \cdots = \sigma_n = 0$$

Now let

$$V_1 = (\mathbf{v}_1, \ldots, \mathbf{v}_r), \qquad V_2 = (\mathbf{v}_{r+1}, \ldots, \mathbf{v}_n)$$

and

$$\Sigma_1 = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_r \end{bmatrix} \tag{1}$$

Hence, Σ_1 is an $r \times r$ diagonal matrix whose diagonal entries are the nonzero singular values $\sigma_1, \ldots, \sigma_r$. The $m \times n$ matrix Σ is then given by

$$\Sigma = \begin{bmatrix} \Sigma_1 & O \\ O & O \end{bmatrix}$$

The column vectors of V_2 are eigenvectors of A^TA belonging to $\lambda = 0$. Thus

$$A^TA\mathbf{v}_j = \mathbf{0} \qquad j = r+1, \ldots, n$$

and, consequently, the column vectors of V_2 form an orthonormal basis for $N(A^TA) = N(A)$. Therefore,

$$AV_2 = O$$

and since V is an orthogonal matrix, it follows that

$$\begin{aligned} I &= VV^T = V_1V_1^T + V_2V_2^T \\ A &= AI = AV_1V_1^T + AV_2V_2^T = AV_1V_1^T \end{aligned} \tag{2}$$

So far we have shown how to construct the matrices V and Σ of the singular value decomposition. To complete the proof, we must show how to construct an $m \times m$ orthogonal matrix U such that

$$A = U\Sigma V^T$$

or, equivalently,

$$AV = U\Sigma \tag{3}$$

and suppose that the machine epsilon is 5×10^{-15}. To determine the numerical rank, we compare the singular values to

$$\sigma_1 \max(m, n)\epsilon = 4 \cdot 5 \cdot 5 \times 10^{-15} = 10^{-13}$$

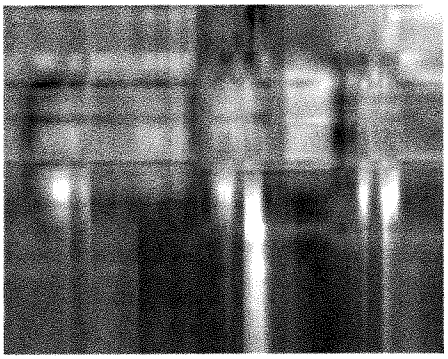
Since three of the singular values are greater than 10^{-13}, the matrix has numerical rank 3. ■

APPLICATION 2 Digital Image Processing

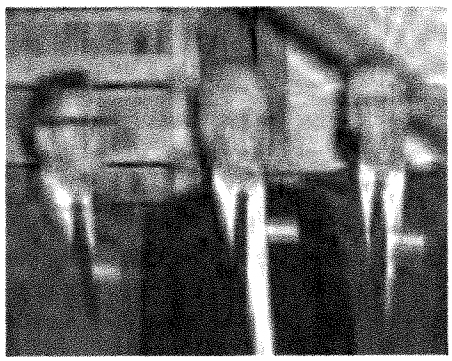
A video image or photograph can be digitized by breaking it up into a rectangular array of cells (or pixels) and measuring the gray level of each cell. This information can be stored and transmitted as an $m \times n$ matrix A. The entries of A are nonnegative numbers corresponding to the measures of the gray levels. Because the gray levels of any one cell generally turn out to be close to the gray levels of its neighboring cells, it is possible to reduce the amount of storage necessary from mn to a relatively small multiple of $m + n + 1$. Generally, the matrix A will have many small singular values. Consequently, A can be approximated by a matrix of much lower rank.

Original 176 by 260 Image

Rank 5 Approximation to Image



Rank 15 Approximation to Image



Rank 30 Approximation to Image

Figure 6.5.1. Courtesy Oakridge National Laboratory

If A has singular value decomposition $U\Sigma V^T$, then A can be represented by the outer product expansion

$$A = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \cdots + \sigma_n \mathbf{u}_n \mathbf{v}_n^T$$

The closest matrix of rank k is obtained by truncating this sum after the first k terms:

$$A_k = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \cdots + \sigma_k \mathbf{u}_k \mathbf{v}_k^T$$

The total storage for A_k is $k(m + n + 1)$. We can choose k to be considerably less than n and still have the digital image corresponding to A_k very close to the original. For typical choices of k, the storage required for A_k will be less than 20 percent of the amount of storage necessary for the entire matrix A.

Figure 6.5.1 shows an image corresponding to a 176×260 matrix A and three images corresponding to lower rank approximations of A. The gentlemen in the picture are (left to right) James H. Wilkinson, Wallace Givens, and George Forsythe (three pioneers in the field of numerical linear algebra).

APPLICATION 3 Information Retrieval—Latent Semantic Indexing

We return again to the information retrieval application discussed in Sections 1.3 and 5.1. In this application a database of documents is represented by a database matrix Q. To search the database, we form a unit search vector $\mathbf{x}$ and set $\mathbf{y} = Q^T\mathbf{x}$. The documents that best match the search criteria are those corresponding to the entries of $\mathbf{y}$ that are closest to 1.

Because of the problems of polysemy and synonymy, we can think of our database as an approximation. Some of the entries of the database matrix may contain extraneous components due to multiple meanings of words, and some may miss including components because of synonymy. Suppose that it were possible to correct for these problems and come up with a perfect database matrix P. If we set $E = Q - P$, then, since $Q = P + E$, we can think of E as a matrix representing the errors in our database matrix Q. Unfortunately, E is unknown, so we cannot determine P exactly. However, if we can find a simpler approximation Q_1 for Q, then Q_1 will also be an approximation for P. Thus $Q_1 = P + E_1$ for some error matrix E_1. In the method of *latent semantic indexing* (LSI), the database matrix Q is approximated by a matrix Q_1 with lower rank. The idea behind the method is that the lower rank matrix may still provide a good approximation to P and, because of its simpler structure, may actually involve less error; that is, $\|E_1\| < \|E\|$.

The lower rank approximation can be obtained by truncating the outer product expansion of the singular value decomposition of Q. This is equivalent to setting

$$\sigma_{r+1} = \sigma_{r+2} = \cdots = \sigma_n = 0$$

and then setting $Q_1 = U_1 \Sigma_1 V_1^T$, the compact form of the singular value decomposition of the rank r matrix. Furthermore, if $r < \min(m, n)/2$, then this factorization is computationally more efficient to use and the searches will be speeded up. The speed of computation is proportional to the amount of arithmetic involved. The matrix vector multiplication $Q^T\mathbf{x}$ requires a total of mn scalar multiplications (m multiplications for