Chapter 11

Dimensionality Reduction

There are many sources of data that can be viewed as a large matrix. We saw in Chapter 5 how the Web can be represented as a transition matrix. In Chapter 9, the utility matrix was a point of focus. And in Chapter 10 we examined matrices that represent social networks. In many of these matrix applications, the matrix can be summarized by finding "narrower" matrices that in some sense are close to the original. These narrow matrices have only a small number of rows or a small number of columns, and therefore can be used much more efficiently than can the original large matrix. The process of finding these narrow matrices is called *dimensionality reduction*.

We saw a preliminary example of dimensionality reduction in Section 9.4. There, we discussed UV-decomposition of a matrix and gave a simple algorithm for finding this decomposition. Recall that a large matrix M was decomposed into two matrices U and V whose product UV was approximately M. The matrix U had a small number of columns whereas V had a small number of rows, so each was significantly smaller than M, and yet together they represented most of the information in M that was useful in predicting ratings of items by individuals.

In this chapter we shall explore the idea of dimensionality reduction in more detail. We begin with a discussion of eigenvalues and their use in "principal component analysis" (PCA). We cover singular-value decomposition, a more powerful version of UV-decomposition. Finally, because we are always interested in the largest data sizes we can handle, we look at another form of decomposition, called CUR-decomposition, which is a variant of singular-value decomposition that keeps the matrices of the decomposition sparse if the original matrix is sparse.

11.1 Eigenvalues and Eigenvectors of Symmetric Matrices

We shall assume that you are familiar with the basics of matrix algebra: multiplication, transpose, determinants, and solving linear equations for example. In this section, we shall define eigenvalues and eigenvectors of a symmetric matrix and show how to find them. Recall a matrix is symmetric if the element in row i and column j equals the element in row j and column i.

11.1.1 Definitions

Let M be a square matrix. Let λ be a constant and $\mathbf{e}$ a nonzero column vector with the same number of rows as M. Then λ is an *eigenvalue* of M and $\mathbf{e}$ is the corresponding *eigenvector* of M if $M\mathbf{e} = \lambda\mathbf{e}$.

If $\mathbf{e}$ is an eigenvector of M and c is any constant, then it is also true that $c\,\mathbf{e}$ is an eigenvector of M with the same eigenvalue. Multiplying a vector by a constant changes the length of a vector, but not its direction. Thus, to avoid ambiguity regarding the length, we shall require that every eigenvector be a *unit vector*, meaning that the sum of the squares of the components of the vector is 1. Even that is not quite enough to make the eigenvector unique, since we may still multiply by -1 without changing the sum of squares of the components. Thus, we shall normally require that the first nonzero component of an eigenvector be positive.

Example 11.1 : Let M be the matrix

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix}$$

One of the eigenvectors of M is

$$\begin{bmatrix} 1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix}$$

and its corresponding eigenvalue is 7. The equation

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} 1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix} = 7 \begin{bmatrix} 1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix}$$

demonstrates the truth of this claim. Note that both sides are equal to

$$\begin{bmatrix} 7/\sqrt{5} \\ 14/\sqrt{5} \end{bmatrix}$$

Also observe that the eigenvector is a unit vector, because $(1/\sqrt{5})^2 + (2/\sqrt{5})^2 = 1/5 + 4/5 = 1$. □

11.1.2 Computing Eigenvalues and Eigenvectors

We have already seen one approach to finding an *eigenpair* (an eigenvalue and its corresponding eigenvector) for a suitable matrix M in Section 5.1: start with any unit vector $\mathbf{v}$ of the appropriate length and compute $M^i\mathbf{v}$ iteratively until it converges.[1] When M is a stochastic matrix, the limiting vector is the *principal* eigenvector (the eigenvector with the largest eigenvalue), and its corresponding eigenvalue is 1.[2] This method for finding the principal eigenvector, called *power iteration*, works quite generally, although if the principal eigenvalue (eigenvalue associated with the principal eigenvector) is not 1, then as i grows, the ratio of $M^{i+1}\mathbf{v}$ to $M^i\mathbf{v}$ approaches the principal eigenvalue while $M^i\mathbf{v}$ approaches a vector (probably not a unit vector) with the same direction as the principal eigenvector.

We shall take up the generalization of the power-iteration method to find all eigenpairs in Section 11.1.3. However, there is an $O(n^3)$-running-time method for computing all the eigenpairs of a symmetric $n \times n$ matrix exactly, and this method will be presented first. There will always be n eigenpairs, although in some cases, some of the eigenvalues will be identical. The method starts by restating the equation that defines eigenpairs, $M\mathbf{e} = \lambda\mathbf{e}$ as $(M - \lambda I)\mathbf{e} = \mathbf{0}$, where

1. I is the $n \times n$ *identity matrix* with 1's along the main diagonal and 0's elsewhere.
2. $\mathbf{0}$ is a vector of all 0's.

A fact of linear algebra is that in order for $(M - \lambda I)\mathbf{e} = \mathbf{0}$ to hold for a vector $\mathbf{e} \neq \mathbf{0}$, the determinant of $M - \lambda I$ must be 0. Notice that $(M - \lambda I)$ looks almost like the matrix M, but if M has c in one of its diagonal elements, then $(M - \lambda I)$ has $c - \lambda$ there. While the determinant of an $n \times n$ matrix has $n!$ terms, it can be computed in various ways in $O(n^3)$ time; an example is the method of "pivotal condensation."

The determinant of $(M - \lambda I)$ is an nth-degree polynomial in λ, from which we can get the n values of λ that are the eigenvalues of M. For any such value, say c, we can then solve the equation $M\mathbf{e} = c\,\mathbf{e}$. There are n equations in n unknowns (the n components of $\mathbf{e}$), but since there is no constant term in any equation, we can only solve for $\mathbf{e}$ to within a constant factor. However, using any solution, we can normalize it so the sum of the squares of the components is 1, thus obtaining the eigenvector that corresponds to eigenvalue c.

Example 11.2: Let us find the eigenpairs for the 2×2 matrix M from Example 11.1. Recall $M =$

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix}$$

[1] Recall M^i denotes multiplying by the matrix M i times, as discussed in Section 5.1.2.

[2] Note that a stochastic matrix is not generally symmetric. Symmetric matrices and stochastic matrices are two classes of matrices for which eigenpairs exist and can be exploited. In this chapter, we focus on techniques for symmetric matrices.

Then $M - \lambda I$ is

$$\begin{bmatrix} 3-\lambda & 2 \\ 2 & 6-\lambda \end{bmatrix}$$

The determinant of this matrix is $(3-\lambda)(6-\lambda) - 4$, which we must set to 0. The equation in λ to solve is thus $\lambda^2 - 9\lambda + 14 = 0$. The roots of this equation are $\lambda = 7$ and $\lambda = 2$; the first is the principal eigenvalue, since it is the larger. Let $\mathbf{e}$ be the vector of unknowns

$$\begin{bmatrix} x \\ y \end{bmatrix}$$

We must solve

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 7 \begin{bmatrix} x \\ y \end{bmatrix}$$

When we multiply the matrix and vector we get two equations

$$\begin{aligned} 3\text{x}+2\text{y} &= 7\text{x} \\ 2\text{x}+6\text{y} &= 7\text{y} \end{aligned}$$

Notice that both of these equations really say the same thing: $y = 2x$. Thus, a possible eigenvector is

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

But that vector is not a unit vector, since the sum of the squares of its components is 5, not 1. Thus to get the unit vector in the same direction, we divide each component by $\sqrt{5}$. That is, the principal eigenvector is

$$\begin{bmatrix} 1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix}$$

and its eigenvalue is 7. Note that this was the eigenpair we explored in Example 11.1.

For the second eigenpair, we repeat the above with eigenvalue 2 in place of 7. The equation involving the components of $\mathbf{e}$ is $x = -2y$, and the second eigenvector is

$$\begin{bmatrix} 2/\sqrt{5} \\ -1/\sqrt{5} \end{bmatrix}$$

Its corresponding eigenvalue is 2, of course. □

11.1.3 Finding Eigenpairs by Power Iteration

We now examine the generalization of the process we used in Section 5.1 to find the principal eigenvector, which in that section was the PageRank vector – all we needed from among the various eigenvectors of the stochastic matrix of the Web. We start by computing the principal eigenvector by a slight generalization of the approach used in Section 5.1. We then modify the matrix to, in

effect, remove the principal eigenvector. The result is a new matrix whose principal eigenvector is the second eigenvector (eigenvector with the second-largest eigenvalue) of the original matrix. The process proceeds in that manner, removing each eigenvector as we find it, and then using power iteration to find the principal eigenvector of the matrix that remains.

Let M be the matrix whose eigenpairs we would like to find. Start with any nonzero vector $\mathbf{x}_0$ and then iterate:

$$\mathbf{x}_{k+1} := \frac{M\mathbf{x}_k}{\|M\mathbf{x}_k\|}$$

where $\|N\|$ for a matrix or vector N denotes the *Frobenius norm*; that is, the square root of the sum of the squares of the elements of N. We multiply the current vector $\mathbf{x}_k$ by the matrix M until convergence (i.e., $\|x_k - x_{k+1}\|$ is less than some small, chosen constant). Let $\mathbf{x}$ be $\mathbf{x}_k$ for that value of k at which convergence is obtained. Then $\mathbf{x}$ is (approximately) the principal eigenvector of M. To obtain the corresponding eigenvalue we simply compute $\lambda_1 = \mathbf{x}^{\mathrm{T}} M\mathbf{x}$, which is the equation $M\mathbf{x} = \lambda\mathbf{x}$ solved for λ, since $\mathbf{x}$ is a unit vector.

Example 11.3: Take the matrix from Example 11.2:

$$M = \begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix}$$

and let us start with $\mathbf{x}_0$ a vector with 1 for both components. To compute $\mathbf{x}_1$, we multiply $M\mathbf{x}_0$ to get

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 8 \end{bmatrix}$$

The Frobenius norm of the result is $\sqrt{5^2 + 8^2} = \sqrt{89} = 9.434$. We obtain $\mathbf{x}_1$ by dividing 5 and 8 by 9.434; that is:

$$\mathbf{x}_1 = \begin{bmatrix} 0.530 \\ 0.848 \end{bmatrix}$$

For the next iteration, we compute

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} 0.530 \\ 0.848 \end{bmatrix} = \begin{bmatrix} 3.286 \\ 6.148 \end{bmatrix}$$

The Frobenius norm of the result is 6.971, so we divide to obtain

$$\mathbf{x}_2 = \begin{bmatrix} 0.471 \\ 0.882 \end{bmatrix}$$

We are converging toward a normal vector whose second component is twice the first. That is, the limiting value of the vector that we obtain by power iteration is the principal eigenvector:

$$\mathbf{x} = \begin{bmatrix} 0.447 \\ 0.894 \end{bmatrix}$$

Finally, we compute the principal eigenvalue by

$$\lambda = \mathbf{x}^{\mathrm{T}} M \mathbf{x} = \begin{bmatrix} 0.447 & 0.894 \end{bmatrix} \begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} 0.447 \\ 0.894 \end{bmatrix} = 6.993$$

Recall from Example 11.2 that the true principal eigenvalue is 7. Power iteration will introduce small errors due either to limited precision, as was the case here, or due to the fact that we stop the iteration before reaching the exact value of the eigenvector. When we computed PageRank, the small inaccuracies did not matter, but when we try to compute all eigenpairs, inaccuracies accumulate if we are not careful. □

To find the second eigenpair we create a new matrix $M^* = M - \lambda_1 \mathbf{x}\mathbf{x}^{\mathrm{T}}$. Then, use power iteration on M^* to compute its largest eigenvalue. The obtained $\mathbf{x}*$ and $\lambda*$ correspond to the second largest eigenvalue and the corresponding eigenvector of matrix M.

Intuitively, what we have done is eliminate the influence of a given eigenvector by setting its associated eigenvalue to zero. The formal justification is the following two observations. If $M^* = M - \lambda \mathbf{x}\mathbf{x}^{\mathrm{T}}$, where $\mathbf{x}$ and λ are the eigenpair with the largest eigenvalue, then:

1. $\mathbf{x}$ is also an eigenvector of M^*, and its corresponding eigenvalue is 0. In proof, observe that

   $$M^*\mathbf{x} = (M - \lambda \mathbf{x}\mathbf{x}^{\mathrm{T}})\mathbf{x} = M\mathbf{x} - \lambda \mathbf{x}\mathbf{x}^{\mathrm{T}}\mathbf{x} = M\mathbf{x} - \lambda \mathbf{x} = 0$$

   At the next-to-last step we use the fact that $\mathbf{x}^{\mathrm{T}}\mathbf{x} = 1$ because $\mathbf{x}$ is a unit vector.

2. Conversely, if $\mathbf{v}$ and λ_v are an eigenpair of a symmetric matrix M other than the first eigenpair $(\mathbf{x}, \lambda)$, then they are also an eigenpair of M^*. *Proof*:

   $$M^*\mathbf{v} = (M^*)^{\mathrm{T}}\mathbf{v} = (M - \lambda \mathbf{x}\mathbf{x}^{\mathrm{T}})^{\mathrm{T}}\mathbf{v} = M^{\mathrm{T}}\mathbf{v} - \lambda \mathbf{x}(\mathbf{x}^{\mathrm{T}}\mathbf{v}) = M^{\mathrm{T}}\mathbf{v} = \lambda_v \mathbf{v}$$

   This sequence of equalities needs the following justifications:

   (a) If M is symmetric, then $M = M^{\mathrm{T}}$.

   (b) The eigenvectors of a symmetric matrix are *orthogonal*. That is, the dot product of any two distinct eigenvectors of a matrix is 0. We do not prove this statement here.

Example 11.4: Continuing Example 11.3, we compute

$$M^* = \begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} - 6.993 \begin{bmatrix} 0.447 & 0.894 \end{bmatrix} \begin{bmatrix} 0.447 \\ 0.894 \end{bmatrix} =$$

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} - \begin{bmatrix} 1.397 & 2.795 \\ 2.795 & 5.589 \end{bmatrix} = \begin{bmatrix} 1.603 & -0.795 \\ -0.795 & 0.411 \end{bmatrix}$$

We may find the second eigenpair by processing the matrix above as we did the original matrix M. □

11.1.4 The Matrix of Eigenvectors

Suppose we have an $n \times n$ symmetric matrix M whose eigenvectors, viewed as column vectors, are $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n$. Let E be the matrix whose ith column is $\mathbf{e}_i$. Then $EE^{\mathrm{T}} = E^{\mathrm{T}}E = I$. The explanation is that the eigenvectors of a symmetric matrix are *orthonormal*. That is, they are orthogonal unit vectors.

Example 11.5: For the matrix M of Example 11.2, the matrix E is

$$\begin{bmatrix} 2/\sqrt{5} & 1/\sqrt{5} \\ -1/\sqrt{5} & 2/\sqrt{5} \end{bmatrix}$$

E^{T} is therefore

$$\begin{bmatrix} 2/\sqrt{5} & -1/\sqrt{5} \\ 1/\sqrt{5} & 2/\sqrt{5} \end{bmatrix}$$

When we compute EE^{T} we get

$$\begin{bmatrix} 4/5 + 1/5 & -2/5 + 2/5 \\ -2/5 + 2/5 & 1/5 + 4/5 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The calculation is similar when we compute $E^{\mathrm{T}}E$. Notice that the 1's along the main diagonal are the sums of the squares of the components of each of the eigenvectors, which makes sense because they are unit vectors. The 0's off the diagonal reflect the fact that the entry in the ith row and jth column is the dot product of the ith and jth eigenvectors. Since eigenvectors are orthogonal, these dot products are 0. □

11.1.5 Exercises for Section 11.1

Exercise 11.1.1: Find the unit vector in the same direction as the vector $[1, 2, 3]$.

Exercise 11.1.2: Complete Example 11.4 by computing the principal eigenvector of the matrix that was constructed in this example. How close to the correct solution (from Example 11.2) are you?

Exercise 11.1.3: For any symmetric 3×3 matrix

$$\begin{bmatrix} a - \lambda & b & c \\ b & d - \lambda & e \\ c & e & f - \lambda \end{bmatrix}$$

there is a cubic equation in λ that says the determinant of this matrix is 0. In terms of a through f, find this equation.

Exercise 11.1.4: Find the eigenpairs for the following matrix:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 3 & 5 \end{bmatrix}$$

using the method of Section 11.1.2.

! **Exercise 11.1.5:** Find the eigenpairs for the following matrix:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 3 & 6 \end{bmatrix}$$

using the method of Section 11.1.2.

Exercise 11.1.6: For the matrix of Exercise 11.1.4:

(a) Starting with a vector of three 1's, use power iteration to find an approximate value of the principal eigenvector.

(b) Compute an estimate the principal eigenvalue for the matrix.

(c) Construct a new matrix by subtracting out the effect of the principal eigenpair, as in Section 11.1.3.

(d) From your matrix of (c), find the second eigenpair for the original matrix of Exercise 11.1.4.

(e) Repeat (c) and (d) to find the third eigenpair for the original matrix.

Exercise 11.1.7: Repeat Exercise 11.1.6 for the matrix of Exercise 11.1.5.

11.2 Principal-Component Analysis

Principal-component analysis, or PCA, is a technique for taking a dataset consisting of a set of tuples representing points in a high-dimensional space and finding the directions along which the tuples line up best. The idea is to treat the set of tuples as a matrix M and find the eigenvectors for MM^{T} or $M^{\mathrm{T}}M$. The matrix of these eigenvectors can be thought of as a rigid rotation in a high-dimensional space. When you apply this transformation to the original data, the axis corresponding to the principal eigenvector is the one along which the points are most "spread out," More precisely, this axis is the one along which the variance of the data is maximized. Put another way, the points can best be viewed as lying along this axis, with small deviations from this axis. Likewise, the axis corresponding to the second eigenvector (the eigenvector corresponding to the second-largest eigenvalue) is the axis along which the variance of distances from the first axis is greatest, and so on.

We can view PCA as a data-mining technique. The high-dimensional data can be replaced by its projection onto the most important axes. These axes are the ones corresponding to the largest eigenvalues. Thus, the original data is approximated by data that has many fewer dimensions and that summarizes well the original data.

11.2.1 An Illustrative Example

We shall start the exposition with a contrived and simple example. In this example, the data is two-dimensional, a number of dimensions that is too small to make PCA really useful. Moreover, the data, shown in Fig. 11.1 has only four points, and they are arranged in a simple pattern along the 45-degree line to make our calculations easy to follow. That is, to anticipate the result, the points can best be viewed as lying along the axis that is at a 45-degree angle, with small deviations in the perpendicular direction.

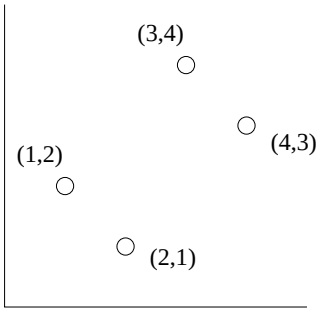

Figure 11.1: Four points in a two-dimensional space

To begin, let us represent the points by a matrix M with four rows – one for each point – and two columns, corresponding to the x-axis and y-axis. This matrix is

$$M = \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 4 \\ 4 & 3 \end{bmatrix}$$

Compute $M^{\mathrm{T}}M$, which is

$$M^{\mathrm{T}}M = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 4 \\ 4 & 3 \end{bmatrix} = \begin{bmatrix} 30 & 28 \\ 28 & 30 \end{bmatrix}$$

We may find the eigenvalues of the matrix above by solving the equation

$$(30 - \lambda)(30 - \lambda) - 28 \times 28 = 0$$

as we did in Example 11.2. The solution is $\lambda = 58$ and $\lambda = 2$.

Following the same procedure as in Example 11.2, we must solve

$$\begin{bmatrix} 30 & 28 \\ 28 & 30 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 58 \begin{bmatrix} x \\ y \end{bmatrix}$$

When we multiply out the matrix and vector we get two equations

$$\begin{array}{rcl} 30\text{x}+28\text{y} & = & 58\text{x} \\ 28\text{x}+30\text{y} & = & 58\text{y} \end{array}$$

Both equations tell us the same thing: $x = y$. Thus, the unit eigenvector corresponding to the principal eigenvalue 58 is

$$\begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

For the second eigenvalue, 2, we perform the same process. Multiply out

$$\begin{bmatrix} 30 & 28 \\ 28 & 30 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 2 \begin{bmatrix} x \\ y \end{bmatrix}$$

to get the two equations

$$\begin{array}{rcl} 30\text{x}+28\text{y} & = & 2\text{x} \\ 28\text{x}+30\text{y} & = & 2\text{y} \end{array}$$

Both equations tell us the same thing: $x = -y$. Thus, the unit eigenvector corresponding to the principal eigenvalue 2 is

$$\begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

While we promised to write eigenvectors with their first component positive, we choose the opposite here because it makes the transformation of coordinates easier to follow in this case.

Now, let us construct E, the matrix of eigenvectors for the matrix $M^{\mathrm{T}}M$. Placing the principal eigenvector first, the matrix of eigenvectors is

$$E = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$$

Any matrix of *orthonormal vectors* (unit vectors that are orthogonal to one another) represents a rotation of the axes of a Euclidean space. The matrix above can be viewed as a rotation 45 degrees counterclockwise. For example, let us multiply the matrix M that represents each of the points of Fig. 11.1 by E. The product is

$$ME = \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 4 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 3/\sqrt{2} & 1/\sqrt{2} \\ 3/\sqrt{2} & -1/\sqrt{2} \\ 7/\sqrt{2} & 1/\sqrt{2} \\ 7/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$$

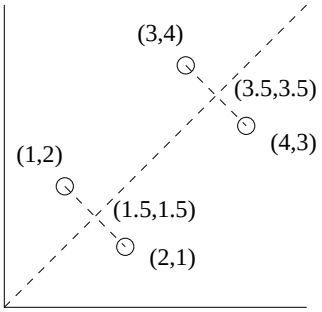

Figure 11.2: Figure 11.1 with the axes rotated 45 degrees counterclockwise

We see the first point, $[1, 2]$, has been transformed into the point

$$[3/\sqrt{2}, 1/\sqrt{2}]$$

If we examine Fig. 11.2, with the dashed line representing the new x-axis, we see that the projection of the first point onto that axis places it at distance $3/\sqrt{2}$ from the origin. To check this fact, notice that the point of projection for both the first and second points is $[1.5, 1.5]$ in the original coordinate system, and the distance from the origin to this point is

$$\sqrt{(1.5)^2 + (1.5)^2} = \sqrt{9/2} = 3/\sqrt{2}$$

Moreover, the new y-axis is, of course, perpendicular to the dashed line. The first point is at distance $1/\sqrt{2}$ above the new x-axis in the direction of the y-axis. That is, the distance between the points $[1, 2]$ and $[1.5, 1.5]$ is

$$\sqrt{(1 - 1.5)^2 + (2 - 1.5)^2} = \sqrt{(-1/2)^2 + (1/2)^2} = \sqrt{1/2} = 1/\sqrt{2}$$

Figure 11.3 shows the four points in the rotated coordinate system.

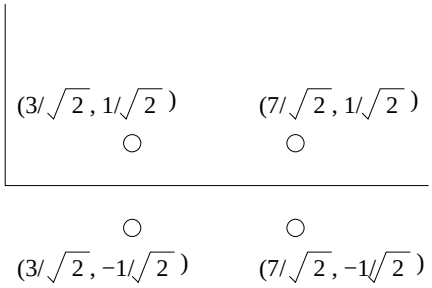

Figure 11.3: The points of Fig. 11.1 in the new coordinate system

The second point, $[2, 1]$ happens by coincidence to project onto the same point of the new x-axis. It is $1/\sqrt{2}$ below that axis along the new y-axis, as is

confirmed by the fact that the second row in the matrix of transformed points is $[3/\sqrt{2}, -1/\sqrt{2}]$. The third point, $[3, 4]$ is transformed into $[7/\sqrt{2}, 1/\sqrt{2}]$ and the fourth point, $[4, 3]$, is transformed to $[7/\sqrt{2}, -1/\sqrt{2}]$. That is, they both project onto the same point of the new x-axis, and that point is at distance $7/\sqrt{2}$ from the origin, while they are $1/\sqrt{2}$ above and below the new x-axis in the direction of the new y-axis.

11.2.2 Using Eigenvectors for Dimensionality Reduction

From the example we have just worked out, we can see a general principle. If M is a matrix whose rows each represent a point in a Euclidean space with any number of dimensions, we can compute $M^{\mathrm{T}}M$ and compute its eigenpairs. Let E be the matrix whose columns are the eigenvectors, ordered as largest eigenvalue first. Define the matrix L to have the eigenvalues of $M^{\mathrm{T}}M$ along the diagonal, largest first, and 0's in all other entries. Then, since $M^{\mathrm{T}}M\mathbf{e} = \lambda\mathbf{e} = \mathbf{e}\lambda$ for each eigenvector $\mathbf{e}$ and its corresponding eigenvalue λ, it follows that $M^{\mathrm{T}}ME = EL$.

We observed that ME is the points of M transformed into a new coordinate space. In this space, the first axis (the one corresponding to the largest eigenvalue) is the most significant; formally, the variance of points along that axis is the greatest. The second axis, corresponding to the second eigenpair, is next most significant in the same sense, and the pattern continues for each of the eigenpairs. If we want to transform M to a space with fewer dimensions, then the choice that preserves the most significance is the one that uses the eigenvectors associated with the largest eigenvalues and ignores the other eigenvalues.

That is, let E_k be the first k columns of E. Then ME_k is a k-dimensional representation of M.

Example 11.6: Let M be the matrix from Section 11.2.1. This data has only two dimensions, so the only dimensionality reduction we can do is to use $k = 1$; i.e., project the data onto a one dimensional space. That is, we compute ME_1 by

$$\begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 4 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 3/\sqrt{2} \\ 3/\sqrt{2} \\ 7/\sqrt{2} \\ 7/\sqrt{2} \end{bmatrix}$$

The effect of this transformation is to replace the points of M by their projections onto the x-axis of Fig. 11.3. While the first two points project to the same point, as do the third and fourth points, this representation makes the best possible one-dimensional distinctions among the points. □

11.2.3 The Matrix of Distances

Let us return to the example of Section 11.2.1, but instead of starting with $M^{\mathrm{T}}M$, let us examine the eigenvalues of MM^{T}. Since our example M has more rows than columns, the latter is a bigger matrix than the former, but if M had more columns than rows, we would actually get a smaller matrix. In the running example, we have

$$MM^{\mathrm{T}} = \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 4 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{bmatrix} = \begin{bmatrix} 5 & 4 & 11 & 10 \\ 4 & 5 & 10 & 11 \\ 11 & 10 & 25 & 24 \\ 10 & 11 & 24 & 25 \end{bmatrix}$$

Like $M^{\mathrm{T}}M$, we see that MM^{T} is symmetric. The entry in the ith row and jth column has a simple interpretation; it is the dot product of the vectors represented by the ith and jth points (rows of M).

There is a strong relationship between the eigenvalues of $M^{\mathrm{T}}M$ and MM^{T}. Suppose $\mathbf{e}$ is an eigenvector of $M^{\mathrm{T}}M$; that is,

$$M^{\mathrm{T}}M\mathbf{e} = \lambda\mathbf{e}$$

Multiply both sides of this equation by M on the left. Then

$$MM^{\mathrm{T}}(M\mathbf{e}) = M\lambda\mathbf{e} = \lambda(M\mathbf{e})$$

Thus, as long as $M\mathbf{e}$ is not the zero vector $\mathbf{0}$, it will be an eigenvector of MM^{T} and λ will be an eigenvalue of MM^{T} as well as of $M^{\mathrm{T}}M$.

The converse holds as well. That is, if $\mathbf{e}$ is an eigenvector of MM^{T} with corresponding eigenvalue λ, then start with $MM^{\mathrm{T}}\mathbf{e} = \lambda\mathbf{e}$ and multiply on the left by M^{T} to conclude that $M^{\mathrm{T}}M(M^{\mathrm{T}}\mathbf{e}) = \lambda(M^{\mathrm{T}}\mathbf{e})$. Thus, if $M^{\mathrm{T}}\mathbf{e}$ is not $\mathbf{0}$, then λ is also an eigenvalue of $M^{\mathrm{T}}M$.

We might wonder what happens when $M^{\mathrm{T}}\mathbf{e} = \mathbf{0}$. In that case, $MM^{\mathrm{T}}\mathbf{e}$ is also $\mathbf{0}$, but $\mathbf{e}$ is not $\mathbf{0}$ because $\mathbf{0}$ cannot be an eigenvector. However, since $\mathbf{0} = \lambda\mathbf{e}$, we conclude that $\lambda = 0$.

We conclude that the eigenvalues of MM^{T} are the eigenvalues of $M^{\mathrm{T}}M$ plus additional 0's. If the dimension of MM^{T} were less than the dimension of $M^{\mathrm{T}}M$, then the opposite would be true; the eigenvalues of $M^{\mathrm{T}}M$ would be those of MM^{T} plus additional 0's.

$$\begin{bmatrix} 3/\sqrt{116} & 1/2 & 7/\sqrt{116} & 1/2 \\ 3/\sqrt{116} & -1/2 & 7/\sqrt{116} & -1/2 \\ 7/\sqrt{116} & 1/2 & -3/\sqrt{116} & -1/2 \\ 7/\sqrt{116} & -1/2 & -3/\sqrt{116} & 1/2 \end{bmatrix}$$

Figure 11.4: Eigenvector matrix for MM^{T}

Example 11.7: The eigenvalues of MM^{T} for our running example must include 58 and 2, because those are the eigenvalues of $M^{\mathrm{T}}M$ as we observed in Section 11.2.1. Since MM^{T} is a 4×4 matrix, it has two other eigenvalues, which must both be 0. The matrix of eigenvectors corresponding to 58, 2, 0, and 0 is shown in Fig. 11.4. □

11.2.4 Exercises for Section 11.2

Exercise 11.2.1: Let M be the matrix of data points

$$\begin{bmatrix} 1 & 1 \\ 2 & 4 \\ 3 & 9 \\ 4 & 16 \end{bmatrix}$$

(a) What are $M^{\mathrm{T}}M$ and MM^{T}?

(b) Compute the eigenpairs for $M^{\mathrm{T}}M$.

! (c) What do you expect to be the eigenvalues of MM^{T}?

! (d) Find the eigenvectors of MM^{T}, using your eigenvalues from part (c).

! **Exercise 11.2.2:** Prove that if M is any matrix, then $M^{\mathrm{T}}M$ and MM^{T} are symmetric.

11.3 Singular-Value Decomposition

We now take up a second form of matrix analysis that leads to a low-dimensional representation of a high-dimensional matrix. This approach, called *singular-value decomposition* (SVD), allows an exact representation of any matrix, and also makes it easy to eliminate the less important parts of that representation to produce an approximate representation with any desired number of dimensions. Of course the fewer the dimensions we choose, the less accurate will be the approximation.

We begin with the necessary definitions. Then, we explore the idea that the SVD defines a small number of "concepts" that connect the rows and columns of the matrix. We show how eliminating the least important concepts gives us a smaller representation that closely approximates the original matrix. Next, we see how these concepts can be used to query the original matrix more efficiently, and finally we offer an algorithm for performing the SVD itself.

11.3.1 Definition of SVD

Let M be an $m \times n$ matrix, and let the rank of M be r. Recall that the *rank* of a matrix is the largest number of rows (or equivalently columns) we can choose

for which no nonzero linear combination of the rows is the all-zero vector **0** (we say a set of such rows or columns is *independent*). Then we can find matrices U, Σ, and V as shown in Fig. 11.5 with the following properties:

1. U is an $m \times r$ *column-orthonormal matrix*; that is, each of its columns is a unit vector and the dot product of any two columns is 0.
2. V is an $n \times r$ column-orthonormal matrix. Note that we always use V in its transposed form, so it is the rows of V^{T} that are orthonormal.
3. Σ is a diagonal matrix; that is, all elements not on the main diagonal are 0. The elements of Σ are called the *singular values* of M.

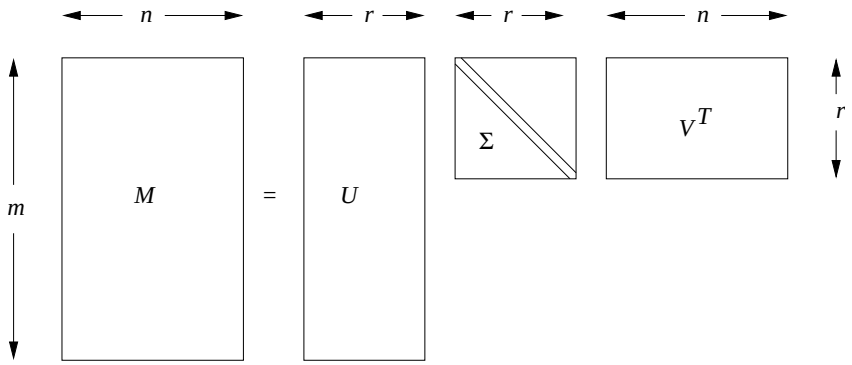

Figure 11.5: The form of a singular-value decomposition

Example 11.8: Figure 11.6 gives a rank-2 matrix representing ratings of movies by users. In this contrived example there are two "concepts" underlying the movies: science-fiction and romance. All the boys rate only science-fiction, and all the girls rate only romance. It is this existence of two strictly adhered to concepts that gives the matrix a rank of 2. That is, we may pick one of the first four rows and one of the last three rows and observe that there is no nonzero linear sum of these rows that is 0. But we cannot pick three independent rows. For example, if we pick rows 1, 2, and 7, then three times the first minus the second, plus zero times the seventh is **0**.

We can make a similar observation about the columns. We may pick one of the first three columns and one of the last two coluns, and they will be independent, but no set of three columns is independent.

The decomposition of the matrix M from Fig. 11.6 into U, Σ, and V, with all elements correct to two significant digits, is shown in Fig. 11.7. Since the rank of M is 2, we can use $r = 2$ in the decomposition. We shall see how to compute this decomposition in Section 11.3.6. □

	Matrix	Alien	Star Wars	Casablanca	Titanic
Joe	1	1	1	0	0
Jim	3	3	3	0	0
John	4	4	4	0	0
Jack	5	5	5	0	0
Jill	0	0	0	4	4
Jenny	0	0	0	5	5
Jane	0	0	0	2	2

Figure 11.6: Ratings of movies by users

$$\underbrace{\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 0 & 0 & 2 & 2 \end{bmatrix}}_{M} = \underbrace{\begin{bmatrix} .14 & 0 \\ .42 & 0 \\ .56 & 0 \\ .70 & 0 \\ 0 & .60 \\ 0 & .75 \\ 0 & .30 \end{bmatrix}}_{U} \underbrace{\begin{bmatrix} 12.4 & 0 \\ 0 & 9.5 \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} .58 & .58 & .58 & 0 & 0 \\ 0 & 0 & 0 & .71 & .71 \end{bmatrix}}_{V^{\mathrm{T}}}$$

Figure 11.7: SVD for the matrix M of Fig. 11.6

11.3.2 Interpretation of SVD

The key to understanding what SVD offers is in viewing the r columns of U, Σ, and V as representing *concepts* that are hidden in the original matrix M. In Example 11.8, these concepts are clear; one is "science fiction" and the other is "romance." Let us think of the rows of M as people and the columns of M as movies. Then matrix U connects people to concepts. For example, the person Joe, who corresponds to row 1 of M in Fig. 11.6, likes only the concept science fiction. The value 0.14 in the first row and first column of U is smaller than some of the other entries in that column, because while Joe watches only science fiction, he doesn't rate those movies highly. The second column of the first row of U is 0, because Joe doesn't rate romance movies at all.

The matrix V relates movies to concepts. The 0.58 in each of the first three columns of the first row of V^{T} indicates that the first three movies – *The Matrix*, *Alien*, and *Star Wars* – each are of the science-fiction genre, while the 0's in the last two columns of the first row say that these movies do not partake of the concept romance at all. Likewise, the second row of V^{T} tells us that the

movies *Casablanca* and *Titanic* are exclusively romances.

Finally, the matrix Σ gives the strength of each of the concepts. In our example, the strength of the science-fiction concept is 12.4, while the strength of the romance concept is 9.5. Intuitively, the science-fiction concept is stronger because the data provides more information about the movies of that genre and the people who like them.

In general, the concepts will not be so clearly delineated. There will be fewer 0's in U and V, although Σ is always a diagonal matrix and will always have 0's off the diagonal. The entities represented by the rows and columns of M (analogous to people and movies in our example) will partake of several different concepts to varying degrees. In fact, the decomposition of Example 11.8 was especially simple, since the rank of the matrix M was equal to the desired number of columns of U, Σ, and V. We were therefore able to get an exact decomposition of M with only two columns for each of the three matrices U, Σ, and V; the product $U\Sigma V^{\mathrm{T}}$, if carried out to infinite precision, would be exactly M. In practice, life is not so simple. When the rank of M is greater than the number of columns we want for the matrices U, Σ, and V, the decomposition is not exact. We need to eliminate from the exact decomposition those columns of U and V that correspond to the smallest singular values, in order to get the best approximation. The following example is a slight modification of Example 11.8 that will illustrate the point.

	Matrix	Alien	Star Wars	Casablanca	Titanic
Joe	1	1	1	0	0
Jim	3	3	3	0	0
John	4	4	4	0	0
Jack	5	5	5	0	0
Jill	0	2	0	4	4
Jenny	0	0	0	5	5
Jane	0	1	0	2	2

Figure 11.8: The new matrix M', with ratings for *Alien* by two additional raters

Example 11.9 : Figure 11.8 is almost the same as Fig. 11.6, but Jill and Jane rated *Alien*, although neither liked it very much. The rank of the matrix in Fig. 11.8 is 3; for example the first, sixth, and seventh rows are independent, but you can check that no four rows are independent. Figure 11.9 shows the decomposition of the matrix from Fig. 11.8.

We have used three columns for U, Σ, and V because they decompose a matrix of rank three. The columns of U and V still correspond to concepts. The first is still "science fiction" and the second is "romance." It is harder to

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

M'

$$\begin{bmatrix} .13 & .02 & -.01 \\ .41 & .07 & -.03 \\ .55 & .09 & -.04 \\ .68 & .11 & -.05 \\ .15 & -.59 & .65 \\ .07 & -.73 & -.67 \\ .07 & -.29 & .32 \end{bmatrix} \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \begin{bmatrix} .56 & .59 & .56 & .09 & .09 \\ .12 & -.02 & .12 & -.69 & -.69 \\ .40 & -.80 & .40 & .09 & .09 \end{bmatrix}$$

U Σ V^{T}

Figure 11.9: SVD for the matrix M' of Fig. 11.8

explain the third column's concept, but it doesn't matter all that much, because its weight, as given by the third nonzero entry in Σ, is very low compared with the weights of the first two concepts. □

In the next section, we consider eliminating some of the least important concepts. For instance, we might want to eliminate the third concept in Example 11.9, since it really doesn't tell us much, and the fact that its associated singular value is so small confirms its unimportance.

11.3.3 Dimensionality Reduction Using SVD

Suppose we want to represent a very large matrix M by its SVD components U, Σ, and V, but these matrices are also too large to store conveniently. The best way to reduce the dimensionality of the three matrices is to set the smallest of the singular values to zero. If we set the s smallest singular values to 0, then we can also eliminate the corresponding s columns of U and V.

Example 11.10: The decomposition of Example 11.9 has three singular values. Suppose we want to reduce the number of dimensions to two. Then we set the smallest of the singular values, which is 1.3, to zero. The effect on the expression in Fig. 11.9 is that the third column of U and the third row of V^{T} are

multiplied only by 0's when we perform the multiplication, so this row and this column may as well not be there. That is, the approximation to M' obtained by using only the two largest singular values is that shown in Fig. 11.10.

$$\begin{bmatrix} .13 & .02 \\ .41 & .07 \\ .55 & .09 \\ .68 & .11 \\ .15 & -.59 \\ .07 & -.73 \\ .07 & -.29 \end{bmatrix} \begin{bmatrix} 12.4 & 0 \\ 0 & 9.5 \end{bmatrix} \begin{bmatrix} .56 & .59 & .56 & .09 & .09 \\ .12 & -.02 & .12 & -.69 & -.69 \end{bmatrix}$$

$$= \begin{bmatrix} 0.93 & 0.95 & 0.93 & .014 & .014 \\ 2.93 & 2.99 & 2.93 & .000 & .000 \\ 3.92 & 4.01 & 3.92 & .026 & .026 \\ 4.84 & 4.96 & 4.84 & .040 & .040 \\ 0.37 & 1.21 & 0.37 & 4.04 & 4.04 \\ 0.35 & 0.65 & 0.35 & 4.87 & 4.87 \\ 0.16 & 0.57 & 0.16 & 1.98 & 1.98 \end{bmatrix}$$

Figure 11.10: Dropping the lowest singular value from the decomposition of Fig. 11.7

The resulting matrix is quite close to the matrix M' of Fig. 11.8. Ideally, the entire difference is the result of making the last singular value be 0. However, in this simple example, much of the difference is due to rounding error caused by the fact that the decomposition of M' was only correct to two significant digits. □

11.3.4 Why Zeroing Low Singular Values Works

The choice of the lowest singular values to drop when we reduce the number of dimensions can be shown to minimize the root-mean-square error between the original matrix M and its approximation. Since the number of entries is fixed, and the square root is a monotone operation, we can simplify and compare the Frobenius norms of the matrices involved. Recall that the *Frobenius norm* of a matrix M, denoted $\|M\|$, is the square root of the sum of the squares of the elements of M. Note that if M is the difference between one matrix and its approximation, then $\|M\|$ is proportional to the RMSE (root-mean-square error) between the matrices.

To explain why choosing the smallest singular values to set to 0 minimizes the RMSE or Frobenius norm of the difference between M and its approximation, let us begin with a little matrix algebra. Suppose M is the product of three matrices $M = PQR$. Let m_{ij}, p_{ij}, q_{ij}, and r_{ij} be the elements in row i and column j of M, P, Q, and R, respectively. Then the definition of matrix

How Many Singular Values Should We Retain?

A useful rule of thumb is to retain enough singular values to make up 90% of the *energy* in Σ. That is, the sum of the squares of the retained singular values should be at least 90% of the sum of the squares of all the singular values. In Example 11.10, the total energy is $(12.4)^2 + (9.5)^2 + (1.3)^2 = 245.70$, while the retained energy is $(12.4)^2 + (9.5)^2 = 244.01$. Thus, we have retained over 99% of the energy. However, were we to eliminate the second singular value, 9.5, the retained energy would be only $(12.4)^2/245.70$ or about 63%.

multiplication tells us

$$m_{ij} = \sum_k \sum_\ell p_{ik} q_{k\ell} r_{\ell j}$$

Then

$$\|M\|^2 = \sum_i \sum_j (m_{ij})^2 = \sum_i \sum_j \Big(\sum_k \sum_\ell p_{ik} q_{k\ell} r_{\ell j} \Big)^2 \tag{11.1}$$

When we square a sum of terms, as we do on the right side of Equation 11.1, we effectively create two copies of the sum (with different indices of summation) and multiply each term of the first sum by each term of the second sum. That is,

$$\Big(\sum_k \sum_\ell p_{ik} q_{k\ell} r_{\ell j} \Big)^2 = \sum_k \sum_\ell \sum_m \sum_n p_{ik} q_{k\ell} r_{\ell j} p_{in} q_{nm} r_{mj}$$

we can thus rewrite Equation 11.1 as

$$\|M\|^2 = \sum_i \sum_j \sum_k \sum_\ell \sum_n \sum_m p_{ik} q_{k\ell} r_{\ell j} p_{in} q_{nm} r_{mj} \tag{11.2}$$

Now, let us examine the case where P, Q, and R are really the SVD of M. That is, P is a column-orthonormal matrix, Q is a diagonal matrix, and R is the transpose of a column-orthonormal matrix. That is, R is *row-orthonormal*; its rows are unit vectors and the dot product of any two different rows is 0. To begin, since Q is a diagonal matrix, $q_{k\ell}$ and q_{nm} will be zero unless $k = \ell$ and $n = m$. We can thus drop the summations for ℓ and m in Equation 11.2 and set $k = \ell$ and $n = m$. That is, Equation 11.2 becomes

$$\|M\|^2 = \sum_i \sum_j \sum_k \sum_n p_{ik} q_{kk} r_{kj} p_{in} q_{nn} r_{nj} \tag{11.3}$$

Next, reorder the summation, so i is the innermost sum. Equation 11.3 has only two factors p_{ik} and p_{in} that involve i; all other factors are constants as far as summation over i is concerned. Since P is column-orthonormal, We know

that $\sum_i p_{ik}p_{in}$ is 1 if $k = n$ and 0 otherwise. That is, in Equation 11.3 we can set $k = n$, drop the factors p_{ik} and p_{in}, and eliminate the sums over i and n, yielding

$$\|M\|^2 = \sum_j \sum_k q_{kk} r_{kj} q_{kk} r_{kj} \tag{11.4}$$

Since R is row-orthonormal, $\sum_j r_{kj} r_{kj}$ is 1. Thus, we can eliminate the terms r_{kj} and the sum over j, leaving a very simple formula for the Frobenius norm:

$$\|M\|^2 = \sum_k (q_{kk})^2 \tag{11.5}$$

Next, let us apply this formula to a matrix M whose SVD is $M = U\Sigma V^{\mathrm{T}}$. Let the ith diagonal element of Σ be σ_i, and suppose we preserve the first n of the r diagonal elements of Σ, setting the rest to 0. Let Σ' be the resulting diagonal matrix. Let $M' = U\Sigma' V^{\mathrm{T}}$ be the resulting approximation to M. Then $M - M' = U(\Sigma - \Sigma')V^{\mathrm{T}}$ is the matrix giving the errors that result from our approximation.

If we apply Equation 11.5 to the matrix $M - M'$, we see that $\|M - M'\|^2$ equals the sum of the squares of the diagonal elements of $\Sigma - \Sigma'$. But $\Sigma - \Sigma'$ has 0 for the first n diagonal elements and σ_i for the ith diagonal element, where $n < i \leq r$. That is, $\|M - M'\|^2$ is the sum of the squares of the elements of Σ that were set to 0. To minimize $\|M - M'\|^2$, pick those elements to be the smallest in Σ. Doing so gives the least possible value of $\|M - M'\|^2$ under the constraint that we preserve n of the diagonal elements, and it therefore minimizes the RMSE under the same constraint.

11.3.5 Querying Using Concepts

In this section we shall look at how SVD can help us answer certain queries efficiently, with good accuracy. Let us assume for example that we have decomposed our original movie-rating data (the rank-2 data of Fig. 11.6) into the SVD form of Fig. 11.7. Quincy is not one of the people represented by the original matrix, but he wants to use the system to know what movies he would like. He has only seen one movie, *The Matrix*, and rated it 4. Thus, we can represent Quincy by the vector $\mathbf{q} = [4, 0, 0, 0, 0]$, as if this were one of the rows of the original matrix.

If we used a collaborative-filtering approach, we would try to compare Quincy with the other users represented in the original matrix M. Instead, we can map Quincy into "concept space" by multiplying him by the matrix V of the decomposition. We find $\mathbf{q}V = [2.32, 0]$.[3] That is to say, Quincy is high in science-fiction interest, and not at all interested in romance.

We now have a representation of Quincy in concept space, derived from, but different from his representation in the original "movie space." One useful thing we can do is to map his representation back into movie space by multiplying

[3]Note that Fig. 11.7 shows V^{T}, while this multiplication requires V.

$[2.32, 0]$ by V^{T}. This product is $[1.35, 1.35, 1.35, 0, 0]$. It suggests that Quincy would like *Alien* and *Star Wars*, but not *Casablanca* or *Titanic*.

Another sort of query we can perform in concept space is to find users similar to Quincy. We can use V to map all users into concept space. For example, Joe maps to $[1.74, 0]$, and Jill maps to $[0, 5.68]$. Notice that in this simple example, all users are either 100% science-fiction fans or 100% romance fans, so each vector has a zero in one component. In reality, people are more complex, and they will have different, but nonzero, levels of interest in various concepts. In general, we can measure the similarity of users by their cosine distance in concept space.

Example 11.11 : For the case introduced above, note that the concept vectors for Quincy and Joe, which are $[2.32, 0]$ and $[1.74, 0]$, respectively, are not the same, but they have exactly the same direction. That is, their cosine distance is 0. On the other hand, the vectors for Quincy and Jill, which are $[2.32, 0]$ and $[0, 5.68]$, respectively, have a dot product of 0, and therefore their angle is 90 degrees. That is, their cosine distance is 1, the maximum possible. □

11.3.6 Computing the SVD of a Matrix

The SVD of a matrix M is strongly connected to the eigenvalues of the symmetric matrices $M^{\mathrm{T}}M$ and MM^{T}. This relationship allows us to obtain the SVD of M from the eigenpairs of the latter two matrices. To begin the explanation, start with $M = U\Sigma V^{\mathrm{T}}$, the expression for the SVD of M. Then

$$M^{\mathrm{T}} = (U\Sigma V^{\mathrm{T}})^{\mathrm{T}} = (V^{\mathrm{T}})^{\mathrm{T}}\Sigma^{\mathrm{T}}U^{\mathrm{T}} = V\Sigma^{\mathrm{T}}U^{\mathrm{T}}$$

Since Σ is a diagonal matrix, transposing it has no effect. Thus, $M^{\mathrm{T}} = V\Sigma U^{\mathrm{T}}$.

Now, $M^{\mathrm{T}}M = V\Sigma U^{\mathrm{T}}U\Sigma V^{\mathrm{T}}$. Remember that U is an orthonormal matrix, so $U^{\mathrm{T}}U$ is the identity matrix of the appropriate size. That is,

$$M^{\mathrm{T}}M = V\Sigma^2V^{\mathrm{T}}$$

Multiply both sides of this equation on the right by V to get

$$M^{\mathrm{T}}MV = V\Sigma^2V^{\mathrm{T}}V$$

Since V is also an orthonormal matrix, we know that $V^{\mathrm{T}}V$ is the identity. Thus

$$M^{\mathrm{T}}MV = V\Sigma^2 \tag{11.6}$$

Since Σ is a diagonal matrix, Σ^2 is also a diagonal matrix whose entry in the ith row and column is the square of the entry in the same position of Σ. Now, Equation (11.6) should be familiar. It says that V is the matrix of eigenvectors of $M^{\mathrm{T}}M$ and Σ^2 is the diagonal matrix whose entries are the corresponding eigenvalues.

Thus, the same algorithm that computes the eigenpairs for $M^{\mathrm{T}}M$ gives us the matrix V for the SVD of M itself. It also gives us the singular values for this SVD; just take the square roots of the eigenvalues for $M^{\mathrm{T}}M$.

Only U remains to be computed, but it can be found in the same way we found V. Start with

$$MM^{\mathrm{T}} = U\Sigma V^{\mathrm{T}}(U\Sigma V^{\mathrm{T}})^{\mathrm{T}} = U\Sigma V^{\mathrm{T}}V\Sigma U^{\mathrm{T}} = U\Sigma^2 U^{\mathrm{T}}$$

Then by a series of manipulations analogous to the above, we learn that

$$MM^{\mathrm{T}}U = U\Sigma^2$$

That is, U is the matrix of eigenvectors of MM^{T}.

A small detail needs to be explained concerning U and V. Each of these matrices have r columns, while $M^{\mathrm{T}}M$ is an $n \times n$ matrix and MM^{T} is an $m \times m$ matrix. Both n and m are at least as large as r. Thus, $M^{\mathrm{T}}M$ and MM^{T} should have an additional $n - r$ and $m - r$ eigenpairs, respectively, and these pairs do not show up in U, V, and Σ. Since the rank of M is r, all other eigenvalues will be 0, and these are not useful.

11.3.7 Exercises for Section 11.3

Exercise 11.3.1: In Fig. 11.11 is a matrix M. It has rank 2, as you can see by observing that the first column plus the third column minus twice the second column equals **0**.

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

Figure 11.11: Matrix M for Exercise 11.3.1

(a) Compute the matrices $M^{\mathrm{T}}M$ and MM^{T}.

! (b) Find the eigenvalues for your matrices of part (a).

(c) Find the eigenvectors for the matrices of part (a).

(d) Find the SVD for the original matrix M from parts (b) and (c). Note that there are only two nonzero eigenvalues, so your matrix Σ should have only two singular values, while U and V have only two columns.

(e) Set your smaller singular value to 0 and compute the one-dimensional approximation to the matrix M from Fig. 11.11.

(f) How much of the energy of the original singular values is retained by the one-dimensional approximation?

Exercise 11.3.2: Use the SVD from Fig. 11.7. Suppose Leslie assigns rating 3 to Alien and rating 4 to Titanic, giving us a representation of Leslie in "movie space" of $[0, 3, 0, 0, 4]$. Find the representation of Leslie in concept space. What does that representation predict about how well Leslie would like the other movies appearing in our example data?

! **Exercise 11.3.3:** Demonstrate that the rank of the matrix in Fig. 11.8 is 3.

! **Exercise 11.3.4:** Section 11.3.5 showed how to guess the movies a person would most like. How would you use a similar technique to guess the people that would most like a given movie, if all you had were the ratings of that movie by a few people?

11.4 CUR Decomposition

There is a problem with SVD that does not show up in the running example of Section 11.3. In large-data applications, it is normal for the matrix M being decomposed to be very sparse; that is, most entries are 0. For example, a matrix representing many documents (as rows) and the words they contain (as columns) will be sparse, because most words are not present in most documents. Similarly, a matrix of customers and products will be sparse because most people do not buy most products.

We cannot deal with dense matrices that have millions or billions of rows and/or columns. However, with SVD, even if M is sparse, U and V will be dense.[4] Since Σ is diagonal, it will be sparse, but Σ is usually much smaller than U and V, so its sparseness does not help.

In this section, we shall consider another approach to decomposition, called CUR-decomposition. The merit of this approach lies in the fact that if M is sparse, then the two large matrices (called C and R for "columns" and "rows") analogous to U and V in SVD are also sparse. Only the matrix in the middle (analogous to Σ in SVD) is dense, but this matrix is small so the density does not hurt too much.

Unlike SVD, which gives an exact decomposition as long as the parameter r is taken to be at least as great as the rank of the matrix M, CUR-decomposition is an approximation no matter how large we make r. There is a theory that guarantees convergence to M as r gets larger, but typically you have to make r so large to get, say within 1% that the method becomes impractical. Nevertheless, a decomposition with a relatively small value of r has a good probability of being a useful and accurate decomposition.

[4] In Fig. 11.7, it happens that U and V have a significant number of 0's. However, that is an artifact of the very regular nature of our example matrix M and is not the case in general.

Why the Pseudoinverse Works

In general suppose a matrix M is equal to a product of matrices XZY. If all the inverses exist, then the rule for inverse of a product tell us $M^{-1} = Y^{-1}Z^{-1}X^{-1}$. Since in the case we are interested in, XZY is an SVD, we know X is column-orthonormal and Y is row-orthonormal. In either of these cases, the inverse and the transpose are the same. That is, XX^{T} is an identity matrix of the appropriate size, and so is YY^{T}. Thus, $M^{-1} = Y^{\mathrm{T}}Z^{-1}X^{\mathrm{T}}$.

We also know Z is a diagonal matrix. If there are no 0's along the diagonal, then Z^{-1} is formed from Z by taking the numerical inverse of each diagonal element. It is only when there are 0's along the diagonal of Z that we are unable to find an element for the same position in the inverse such that we can get an identity matrix when we multiply Z by its inverse. That is why we resort to a "pseudoinverse," accepting the fact that the product ZZ^{+} will not be an identity matrix, but rather a diagonal matrix where the ith diagonal entry is 1 if the ith element of Z is nonzero and 0 if the ith element of Z is 0.

11.4.1 Definition of CUR

Let M be a matrix of m rows and n columns. Pick a target number of "concepts" r to be used in the decomposition. A *CUR-decomposition* of M is a randomly chosen set of r columns of M, which form the $m \times r$ matrix C, and a randomly chosen set of r rows of M, which form the $r \times n$ matrix R. There is also an $r \times r$ matrix U that is constructed from C and R as follows:

1. Let W be the $r \times r$ matrix that is the intersection of the chosen columns of C and the chosen rows of R. That is, the element in row i and column j of W is the element of M whose column is the jth column of C and whose row is the ith row of R.

2. Compute the SVD of W; say $W = X\Sigma Y^{\mathrm{T}}$.

3. Compute Σ^{+}, the *Moore-Penrose pseudoinverse* of the diagonal matrix Σ. That is, if the ith diagonal element of Σ is $\sigma \neq 0$, then replace it by $1/\sigma$. But if the ith element is 0, leave it as 0.

4. Let $U = Y(\Sigma^{+})^2X^{\mathrm{T}}$.

We shall defer to Section 11.4.3 an example where we illustrate the entire CUR process, including the important matter of how the matrices C and R should be chosen to make the approximation to M have a small expected value.

11.4.2 Choosing Rows and Columns Properly

Recall that the choice of rows and columns is random. However, this choice must be biased so that the more important rows and columns have a better chance of being picked. The measure of importance we must use is the square of the Frobenius norm, that is, the sum of the squares of the elements of the row or column. Let $f = \sum_{i,j} m_{ij}^2$, the square of the Frobenius norm of M. Then each time we select a row, the probability p_i with which we select row i is $\sum_j m_{ij}^2/f$. Each time we select a column, the probability q_j with which we select column j is $\sum_i m_{ij}^2/f$.

	Matrix	Alien	Star Wars	Casablanca	Titanic
Joe	1	1	1	0	0
Jim	3	3	3	0	0
John	4	4	4	0	0
Jack	5	5	5	0	0
Jill	0	0	0	4	4
Jenny	0	0	0	5	5
Jane	0	0	0	2	2

Figure 11.12: Matrix M, repeated from Fig. 11.6

Example 11.12: Let us reconsider the matrix M from Fig. 11.6, which we repeat here as Fig. 11.12. The sum of the squares of the elements of M is 243. The three columns for the science-fiction movies *The Matrix*, *Alien*, and *Star Wars* each have a squared Frobenius norm of $1^2 + 3^2 + 4^2 + 5^2 = 51$, so their probabilities are each $51/243 = .210$. The remaining two columns each have a squared Frobenius norm of $4^2 + 5^2 + 2^2 = 45$, and therefore their probabilities are each $45/243 = .185$.

The seven rows of M have squared Frobenius norms of 3, 27, 48, 75, 32, 50, and 8, respectively. Thus, their respective probabilities are .012, .111, .198, .309, .132, .206, and .033. □

Now, let us select r columns for the matrix C. For each column, we choose randomly from the columns of M. However, the selection is not with uniform probability; rather, the jth column is selected with probability q_j. Recall that probability is the sum of the squares of the elements in that column divided by the sum of the squares of all the elements of the matrix. Each column of C is chosen independently from the columns of M, so there is some chance that a column will be selected more than once. We shall discuss how to deal with this situation after explaining the basics of CUR-decomposition.

Having selected each of the columns of M, we scale each column by dividing its elements by the square root of the expected number of times this column would be picked. That is, we divide the elements of the jth column of M, if it is selected, by $\sqrt{rq_j}$. The scaled column of M becomes a column of C.

Rows of M are selected for R in the analogous way. For each row of R we select from the rows of M, choosing row i with probability p_i. Recall p_i is the sum of the squares of the elements of the ith row divided by the sum of the squares of all the elements of M. We then scale each chosen row by dividing by $\sqrt{rp_i}$ if it is the ith row of M that was chosen.

Example 11.13: Let $r = 2$ for our CUR-decomposition. Suppose that our random selection of columns from matrix M of Fig. 11.12 is first *Alien* (the second column) and then *Casablanca* (the fourth column). The column for *Alien* is $[1, 3, 4, 5, 0, 0, 0]^{\mathrm{T}}$, and we must scale this column by dividing by $\sqrt{rq_2}$. Recall from Example 11.12 that the probability associated with the *Alien* column is .210, so the division is by $\sqrt{2 \times 0.210} = 0.648$. To two decimal places, the scaled column for *Alien* is $[1.54, 4.63, 6.17, 7.72, 0, 0, 0]^{\mathrm{T}}$. This column becomes the first column of C.

The second column of C is constructed by taking the column of M for *Casablanca*, which is $[0, 0, 0, 0, 4, 5, 2]^{\mathrm{T}}$, and dividing it by $\sqrt{rp_4} = \sqrt{2 \times 0.185} = 0.608$. Thus, the second column of C is $[0, 0, 0, 0, 6.58, 8.22, 3.29]^{\mathrm{T}}$ to two decimal places.

Now, let us choose the rows for R. The most likely rows to be chosen are those for Jenny and Jack, so let's suppose these rows are indeed chosen, Jenny first. The unscaled rows for R are thus

$$\begin{bmatrix} 0 & 0 & 0 & 5 & 5 \\ 5 & 5 & 5 & 0 & 0 \end{bmatrix}$$

To scale the row for Jenny, we note that its associated probability is 0.206, so we divide by $\sqrt{2 \times 0.206} = 0.642$. To scale the row for Jack, whose associated probability is 0.309, we divide by $\sqrt{2 \times 0.309} = 0.786$. Thus, the matrix R is

$$\begin{bmatrix} 0 & 0 & 0 & 7.79 & 7.79 \\ 6.36 & 6.36 & 6.36 & 0 & 0 \end{bmatrix}$$

□

11.4.3 Constructing the Middle Matrix

Finally, we must construct the matrix U that connects C and R in the decomposition. Recall that U is an $r \times r$ matrix. We start the construction of U with another matrix of the same size, which we call W. The entry in row i and column j of W is the entry of M whose row is the one from which we selected the ith row of R and whose column is the one from which we selected the jth column of C.

Example 11.14: Let us follow the selections of rows and columns made in Example 11.13. We claim

$$W = \begin{bmatrix} 0 & 5 \\ 5 & 0 \end{bmatrix}$$

The first row of W corresponds to the first row of R, which is the row for Jenny in the matrix M of Fig. 11.12. The 0 in the first column is there because that is the entry in the row of M for Jenny and the column for *Alien*; recall that the first column of C was constructed from the column of M for *Alien*. The 5 in the second column reflects the 5 in M's row for Jenny and column for *Casablanca*; the latter is the column of M from which the second column of C was derived. Similarly, the second row of W is the entries in the row for Jack and columns for *Alien* and *Casablanca*, respectively. □

The matrix U is constructed from W by the Moore-Penrose pseudoinverse described in Section 11.4.1. It consists of taking the SVD of W, say $W = X\Sigma Y^{\mathrm{T}}$, and replacing all nonzero elements in the matrix Σ of singular values by their numerical inverses, to obtain the pseudoinverse Σ^+. Then $U = Y(\Sigma^+)^2 X^{\mathrm{T}}$.

Example 11.15: Let us construct U from the matrix W that we constructed in Example 11.14. First, here is the SVD for W:

$$W = \begin{bmatrix} 0 & 5 \\ 5 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

That is, the three matrices on the right are X, Σ, and Y^{T}, respectively. The matrix Σ has no zeros along the diagonal, so each element is replaced by its numerical inverse to get its Moore-Penrose pseudoinverse:

$$\Sigma^+ = \begin{bmatrix} 1/5 & 0 \\ 0 & 1/5 \end{bmatrix}$$

Now X and Y are symmetric, so they are their own transposes. Thus,

$$U = Y(\Sigma^+)^2 X^{\mathrm{T}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1/5 & 0 \\ 0 & 1/5 \end{bmatrix}^2 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1/25 \\ 1/25 & 0 \end{bmatrix}$$

□

11.4.4 The Complete CUR Decomposition

We now have a method to select randomly the three component matrices C, U, and R. Their product will approximate the original matrix M. As we mentioned at the beginning of the discussion, the approximation is only formally guaranteed to be close when very large numbers of rows and columns are selected. However, the intuition is that by selecting rows and columns that tend to have high "importance" (i.e., high Frobenius norm), we are extracting

$$CUR = \begin{bmatrix} 1.54 & 0 \\ 4.63 & 0 \\ 6.17 & 0 \\ 7.72 & 0 \\ 0 & 9.30 \\ 0 & 11.63 \\ 0 & 4.65 \end{bmatrix} \begin{bmatrix} 0 & 1/25 \\ 1/25 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 11.01 & 11.01 \\ 8.99 & 8.99 & 8.99 & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0.55 & 0.55 & 0.55 & 0 & 0 \\ 1.67 & 1.67 & 1.67 & 0 & 0 \\ 2.22 & 2.22 & 2.22 & 0 & 0 \\ 2.78 & 2.78 & 2.78 & 0 & 0 \\ 0 & 0 & 0 & 4.10 & 4.10 \\ 0 & 0 & 0 & 5.12 & 5.12 \\ 0 & 0 & 0 & 2.05 & 2.05 \end{bmatrix}$$

Figure 11.13: CUR-decomposition of the matrix of Fig. 11.12

the most significant parts of the original matrix, even with a small number of rows and columns. As an example, let us see how well we do with the running example of this section.

Example 11.16: For our running example, the decomposition is shown in Fig. 11.13. While there is considerable difference between this result and the original matrix M, especially in the science-fiction numbers, the values are in proportion to their originals. This example is much too small, and the selection of the small numbers of rows and columns was arbitrary rather than random, for us to expect close convergence of the CUR decomposition to the exact values. □

11.4.5 Eliminating Duplicate Rows and Columns

It is quite possible that a single row or column is selected more than once. There is no great harm in using the same row twice, although the rank of the matrices of the decomposition will be less than the number of row and column choices made. However, it is also possible to combine k rows of R that are each the same row of the matrix M into a single row of R, thus leaving R with fewer rows. Likewise, k columns of C that each come from the same column of M can be combined into one column of C. However, for either rows or columns, the remaining vector should have each of its elements multiplied by $\sqrt{k}$.

When we merge some rows and/or columns, it is possible that R has fewer rows than C has columns, or vice versa. As a consequence, W will not be a square matrix. However, we can still take its pseudoinverse by decomposing it into $W = X\Sigma Y^{\mathrm{T}}$, where Σ is now a diagonal matrix with some all-0 rows or

columns, whichever it has more of. To take the pseudoinverse of such a diagonal matrix, we treat each element on the diagonal as usual (invert nonzero elements and leave 0 as it is), but then we must transpose the result.

Example 11.17: Suppose

$$\Sigma = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 \end{bmatrix}$$

Then

$$\Sigma^+ = \begin{bmatrix} 1/2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1/3 \\ 0 & 0 & 0 \end{bmatrix}$$

□

11.4.6 Exercises for Section 11.4

Exercise 11.4.1: The SVD for the matrix

$$M = \begin{bmatrix} 48 & 14 \\ 14 & -48 \end{bmatrix}$$

is

$$\begin{bmatrix} 48 & 14 \\ 14 & -48 \end{bmatrix} = \begin{bmatrix} 3/5 & 4/5 \\ 4/5 & -3/5 \end{bmatrix} \begin{bmatrix} 50 & 0 \\ 0 & 25 \end{bmatrix} \begin{bmatrix} 4/5 & -3/5 \\ 3/5 & 4/5 \end{bmatrix}$$

Find the Moore-Penrose pseudoinverse of M.

! **Exercise 11.4.2:** Find the CUR-decomposition of the matrix of Fig. 11.12 when we pick two "random" rows and columns as follows:

(a) The columns for *The Matrix* and *Alien* and the rows for Jim and John.

(b) The columns for *Alien* and *Star Wars* and the rows for Jack and Jill.

(c) The columns for *The Matrix* and *Titanic* and the rows for Joe and Jane.

! **Exercise 11.4.3:** Find the CUR-decomposition of the matrix of Fig. 11.12 if the two "random" rows are both Jack and the two columns are *Star Wars* and *Casablanca*.

11.5 Summary of Chapter 11

✦ *Dimensionality Reduction*: The goal of dimensionality reduction is to replace a large matrix by two or more other matrices whose sizes are much smaller than the original, but from which the original can be approximately reconstructed, usually by taking their product.

- *Eigenvalues and Eigenvectors*: A matrix may have several eigenvectors such that when the matrix multiplies the eigenvector, the result is a constant multiple of the eigenvector. That constant is the eigenvalue associated with this eigenvector. Together the eigenvector and its eigenvalue are called an eigenpair.
- *Finding Eigenpairs by Power Iteration*: We can find the principal eigenvector (eigenvector with the largest eigenvalue) by starting with any vector and repeatedly multiplying the current vector by the matrix to get a new vector. When the changes to the vector become small, we can treat the result as a close approximation to the principal eigenvector. By modifying the matrix, we can then use the same iteration to get the second eigenpair (that with the second-largest eigenvalue), and similarly get each of the eigenpairs in turn, in order of decreasing value of the eigenvalue.
- *Principal-Component Analysis*: This technique for dimensionality reduction views data consisting of a collection of points in a multidimensional space as a matrix, with rows corresponding to the points and columns to the dimensions. The product of this matrix and its transpose has eigenpairs, and the principal eigenvector can be viewed as the direction in the space along which the points best line up. The second eigenvector represents the direction in which deviations from the principal eigenvector are the greatest, and so on.
- *Dimensionality Reduction by PCA*: By representing the matrix of points by a small number of its eigenvectors, we can approximate the data in a way that minimizes the root-mean-square error for the given number of columns in the representing matrix.
- *Singular-Value Decomposition*: The singular-value decomposition of a matrix consists of three matrices, U, Σ, and V. The matrices U and V are column-orthonormal, meaning that as vectors, the columns are orthogonal, and their lengths are 1. The matrix Σ is a diagonal matrix, and the values along its diagonal are called singular values. The product of U, Σ, and the transpose of V equals the original matrix.
- *Concepts*: SVD is useful when there are a small number of concepts that connect the rows and columns of the original matrix. For example, if the original matrix represents the ratings given by movie viewers (rows) to movies (columns), the concepts might be the genres of the movies. The matrix U connects rows to concepts, Σ represents the strengths of the concepts, and V connects the concepts to columns.
- *Queries Using the Singular-Value Decomposition*: We can use the decomposition to relate new or hypothetical rows of the original matrix to the concepts represented by the decomposition. Multiply a row by the matrix V of the decomposition to get a vector indicating the extent to which that row matches each of the concepts.

- ✦ *Using SVD for Dimensionality Reduction*: In a complete SVD for a matrix, U and V are typically as large as the original. To use fewer columns for U and V, delete the columns corresponding to the smallest singular values from U, V, and Σ. This choice minimizes the error in reconstructing the original matrix from the modified U, Σ, and V.

- ✦ *Decomposing Sparse Matrices*: Even in the common case where the given matrix is sparse, the matrices constructed by SVD are dense. The CUR decomposition seeks to decompose a sparse matrix into sparse, smaller matrices whose product approximates the original matrix.

- ✦ *CUR Decomposition*: This method chooses from a given sparse matrix a set of columns C and a set of rows R, which play the role of U and V^T in SVD; the user can pick any number of rows and columns. The choice of rows and columns is made randomly with a distribution that depends on the Frobenius norm, or the square root of the sum of the squares of the elements. Between C and R is a square matrix called U that is constructed by a pseudo-inverse of the intersection of the chosen rows and columns.

11.6 References for Chapter 11

A well regarded text on matrix algebra is [4].

Principal component analysis was first discussed over a century ago, in [6].

SVD is from [3]. There have been many applications of this idea. Two worth mentioning are [1] dealing with document analysis and [8] dealing with applications in Biology.

The CUR decomposition is from [2] and [5]. Our description follows a later work [7].

1. S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *J. American Society for Information Science* **41**:6 (1990).

2. P. Drineas, R. Kannan, and M.W. Mahoney, "Fast Monte-Carlo algorithms for matrices III: Computing a compressed approximate matrix decomposition," *SIAM J. Computing* **36**:1 (2006), pp. 184–206.

3. G.H. Golub and W. Kahan, "Calculating the singular values and pseudo-inverse of a matrix," *J. SIAM Series B* **2**:2 (1965), pp. 205–224.

4. G.H.Golub and C.F. Van Loan, *Matrix Computations*, JHU Press, 1996.

5. M.W. Mahoney, M. Maggioni, and P. Drineas, Tensor-CUR decompositions For tensor-based data, *SIGKDD*, pp. 327–336, 2006.

6. K. Pearson, "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine* **2**:11 (1901), pp. 559–572.

7. J. Sun, Y. Xie, H. Zhang, and C. Faloutsos, "Less is more: compact matrix decomposition for large sparse graphs," *Proc. SIAM Intl. Conf. on Data Mining*, 2007.

8. M.E. Wall, A. Reichtsteiner and L.M. Rocha, "Singular value decomposition and principal component analysis," in *A Practical Approach to Microarray Data Analysis* (D.P. Berrar, W. Dubitzky, and M. Granzow eds.), pp. 91–109, Kluwer, Norwell, MA, 2003.