Chapter 10

Factor Analysis

10.1 THE FACTOR ANALYSIS PROBLEM

Consider an ocean whose sediment is derived from the simple mixing of continental source rocks A and B (Figure 10.1). Suppose that the concentrations of three elements are determined for many samples of sediment and then plotted on a graph whose axes are percentages of those elements. Since all the sediments are derived from only two source rocks, the sample compositions lie on the triangular portion of a plane bounded by the compositions of A and B (Figure 10.2).

The factor analysis problem is to deduce the number of the source rocks (called *factors*) and their composition from observations of the composition of the sediments (called *samples*). It is therefore a problem in inverse theory. We shall discuss it separately, since it provides an interesting example of the use of some of the vector space analysis techniques developed in Chapter 7.

The model proposes that the samples are simple mixtures (linear combinations) of the factors. If there are N samples containing M elements and if there are p factors, we can state this model algebraically with the equation

$$\mathbf{S} = \mathbf{CF} \tag{10.1}$$

where S_{ij} is the fraction of element j in sample i:

$$\mathbf{S} = \begin{bmatrix} \text{element 1 in sample 1} & \cdots & \text{element } M \text{ in sample 1} \\ \text{element 1 in sample 2} & \cdots & \text{element } M \text{ in sample 2} \\ \vdots & \ddots & \vdots \\ \text{element 1 in sample } N & \cdots & \text{element } M \text{ in sample } N \end{bmatrix} \tag{10.2}$$

(The word "element" is used in the generic sense, since most factor analysis problems will not involve *chemical* elements.) We will refer to individual samples as $\mathbf{s}^{(i)}$, with $\mathbf{s}^{(i)\mathrm{T}}$ a row of $\mathbf{S}$. Similarly, F_{ij} is the fraction of element j in factor i:

$$\mathbf{F} = \begin{bmatrix} \text{element 1 in factor 1} & \cdots & \text{element } M \text{ in factor 1} \\ \text{element 1 in factor 2} & \cdots & \text{element } M \text{ in factor 2} \\ \vdots & \ddots & \vdots \\ \text{element 1 in factor } p & \cdots & \text{element } M \text{ in factor } p \end{bmatrix} \tag{10.3}$$

Geophysical Data Analysis: Discrete Inverse Theory. DOI: 10.1016/B978-0-12-397160-9.00010-2

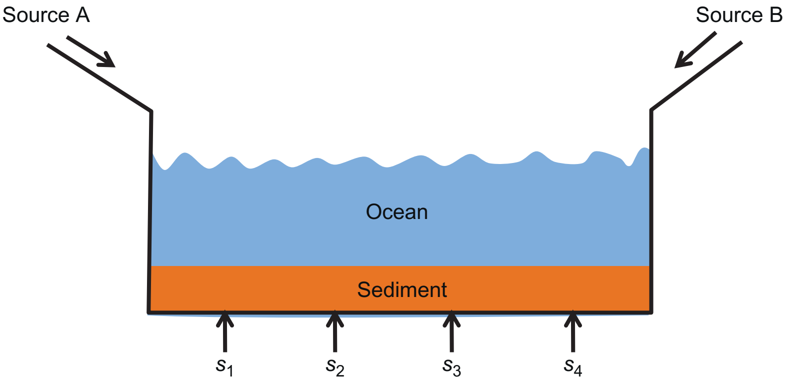

FIGURE 10.1 Material from sources A and B is eroded into the ocean and deposited to form sediment. Samples s_i of the sediment are collected and their chemical composition is determined. The data are used to infer the composition of the sources.

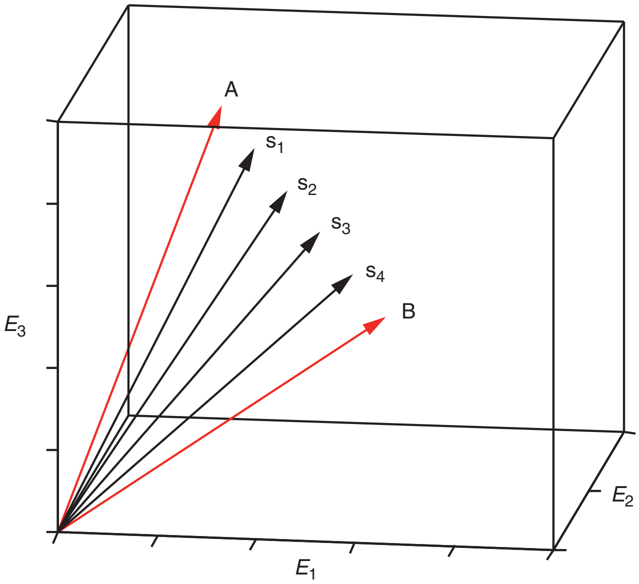

FIGURE 10.2 The composition of the samples $\mathbf{s}_i$ (black arrows) lies on a triangular sector of a plane bounded by the composition of the sources A and B (red arrows). *MatLab* script gda10_01.

We will refer to individual factors as $\mathbf{f}^{(i)}$, with $\mathbf{f}^{(i)T}$ a row of $\boldsymbol{F}$. C_{ij} is the fraction of factor i in sample j:

$$\mathbf{C} = \begin{bmatrix} \text{factor 1 in sample 1} & \cdots & \text{factor } p \text{ in sample 1} \\ \text{factor 1 in sample 2} & \cdots & \text{factor } p \text{ in sample 2} \\ \vdots & \ddots & \vdots \\ \text{factor 1 in sample } N & \cdots & \text{factor } p \text{ in sample } N \end{bmatrix} \tag{10.4}$$

The elements of the matrix **C** are referred to as the *factor loadings* (or sometimes just the *loadings*).

The inverse problem is to factor the matrix **S** into **C** and **F**. Each sample (each row of **S**) is represented as a linear combination of factors (rows of **F**), with the elements of **C** giving the coefficients of the combination. As long as we pick an **F** whose rows span the space spanned by the rows of **S**, we can perform the factorization. For $p \geq M$, any linearly independent set of factors will do, so in this sense the factor analysis problem is completely nonunique. It is much more interesting to ask what the minimum number of factors is that can be used to represent the samples. Then the factor analysis problem is equivalent to examining the space spanned by **S** and determining its dimension. This problem can easily be solved by representing the sample matrix with its singular-value decomposition as

$$\mathbf{S} = \mathbf{U}_p \boldsymbol{\Lambda}_p \mathbf{V}_p^{\mathrm{T}} = (\mathbf{U}_p \boldsymbol{\Lambda}_p)(\mathbf{V}_p^{\mathrm{T}}) = \mathbf{CF} \tag{10.5}$$

Only the eigenvectors with nonzero singular values appear in the decomposition. The number of factors is given by the number of nonzero singular values. One possible set of factors is the p eigenvectors. This set of factors is not unique (Figure 10.3). Any set of factors that spans the p space will do. Mathematically, we transform the factors with any matrix **T** that possesses an inverse, in which case $\mathbf{S} = \mathbf{CF} = \mathbf{CIF} = (\mathbf{CT}^{-1})(\mathbf{TF}) = \mathbf{C}'\mathbf{F}'$ with the transformed factors being $\mathbf{F}' = (\mathbf{TF})$ and the new loadings being $\mathbf{C}' = \mathbf{CT}^{-1}$.

If we write out Equation (10.5), we find that the composition of the ith sample $\mathbf{s}^{(i)}$ is related to the eigenvectors $\mathbf{v}^{(i)}$ and singular values λ_i by

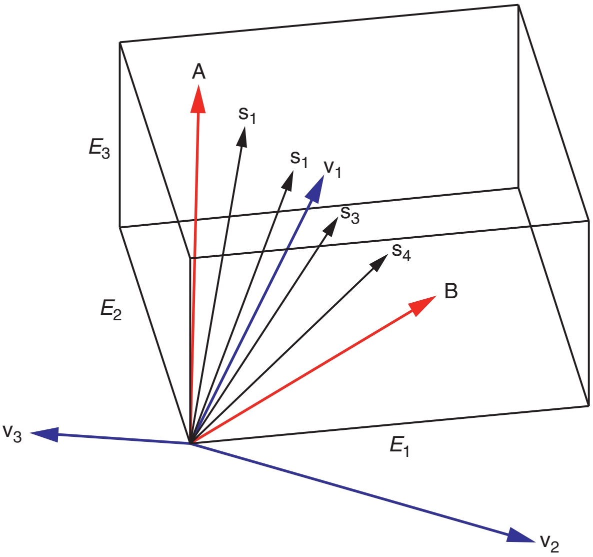

FIGURE 10.3 Eigenvectors $\mathbf{v}_1$ and $\mathbf{v}_2$ lie in the plane of the samples ($\mathbf{v}_1$ is closest to the mean sample). Eigenvector $\mathbf{v}_3$ is normal to the plane. *MatLab* script gda10_02.

$$
\begin{aligned}
\mathbf{s}^{(1)} &= [\mathbf{U}_p]_{11}\lambda_1\mathbf{v}^{(1)} + [\mathbf{U}_p]_{12}\lambda_2\mathbf{v}^{(2)} + \cdots + [\mathbf{U}_p]_{1p}\lambda_p\mathbf{v}^{(p)} \\
&\cdots \\
\mathbf{s}^{(N)} &= [\mathbf{U}_p]_{N1}\lambda_1\mathbf{v}^{(1)} + [\mathbf{U}_p]_{N2}\lambda_2\mathbf{v}^{(2)} + \cdots + [\mathbf{U}_p]_{Np}\lambda_p\mathbf{v}^{(p)}
\end{aligned}
\tag{10.6}
$$

If the singular values are arranged in descending order, then most of each sample is composed of factor 1, with a smaller contribution from factor 2, etc. Because $\mathbf{U}_p$ and $\mathbf{v}^{(p)}$ are composed of unit vectors, on average their elements are of equal size. We have identified the most "important" factors (Figure 10.4). Even if $p = M$, it might be possible to neglect some of the smaller singular values and still achieve a reasonably good prediction of the sample compositions; that is, $\mathbf{S} \approx \mathbf{CF} = (\mathbf{U}_q\boldsymbol{\Lambda}_q)(\mathbf{V}_q^T)$ with $q < p$.

The eigenvector with the largest singular value is near the mean of the sample vectors. It is easy to show that the sample mean $\langle\mathbf{s}\rangle$ maximizes the sum of dot products with the data $\sum_i [\mathbf{s}^{(i)} \cdot \langle\mathbf{s}\rangle]$, while the eigenvector $\mathbf{v}$ with largest singular value maximizes the sum of squared dot products $\sum_i [\mathbf{s}^{(i)}{}_i \cdot \mathbf{v}]^2$. (To show this, maximize the given functions using Lagrange multipliers, with the constraint that $\langle\mathbf{s}\rangle$ and $\mathbf{v}$ are unit vectors.) As long as most of the samples are in the same quadrant, these two functions have roughly the same maximum.

The *MatLab* code for computing the singular-value decomposition is

```
[U, LAMBDA, V] = svd(S,0);
lambda = diag(LAMBDA);
F = V';
C = U*LAMBDA;
```

(*MatLab* gda10_04)

Here we use the "economy" version of `svd()`, which has a second argument of zero. In the $N > M$ case, it returns $\mathbf{U}$ as $N \times M$ and $\boldsymbol{\Lambda}$ as $M \times M$ (since the bottom $N - M$ rows of $\boldsymbol{\Lambda}$ are zero and hence the right $N - M$ columns of $\mathbf{U}$ do not

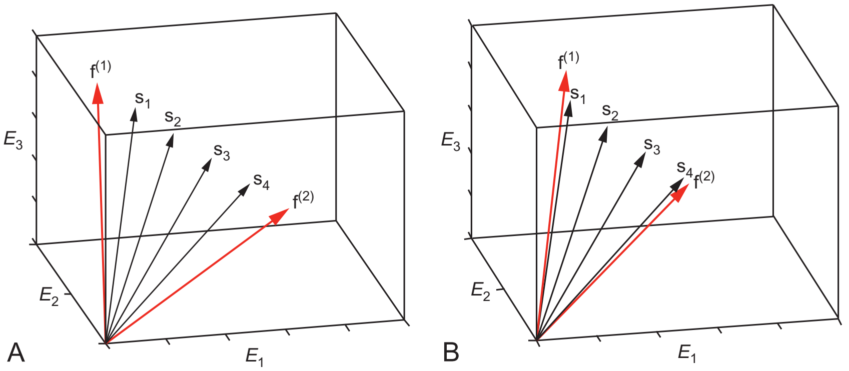

FIGURE 10.4 Any two factors $\mathbf{f}^{(1)}$ and $\mathbf{f}^{(2)}$ (red arrows) that lie in the plane of the samples and that bound the range of sample compositions (black arrows) are acceptable, such as those shown in (A) and (B). *MatLab* script gda10_03.

contribute to **G**). The `svd()` function does not throw out any of the zero (or near-zero) eigenvalues; this is left to the user. The diagonal of `LAMBDA` has been copied into the column-vector, `lambda`, for convenience.

We apply factor analysis to rock chemistry data taken from a petrologic database (PetDB at www.petdb.org). This database contains chemical information on igneous and metamorphic rocks collected from the floor of all the world's oceans, but we analyze here $N=6356$ samples from the Atlantic Ocean that have the following chemical species: SiO_2, TiO_2, Al_2O_3, FeO_{total}, MgO, CaO, Na_2O, and K_2O (units of weight percent).

A plot of the singular values of the Atlantic Rock data set (Figure 10.5) reveals that the first value is by far the largest, values 2 through 5 are intermediate in size, and values 6 through 8 are near-zero. The fact that the first singular value λ_1 is much larger than all the others reflects the composition of the rock samples having only a small range of variability. Thus, all rock samples contain a large amount of the first factor, $\mathbf{f}^{(1)}$—the typical sample. Only four additional factors, $\mathbf{f}^{(2)}$, $\mathbf{f}^{(3)}$, $\mathbf{f}^{(4)}$, and $\mathbf{f}^{(5)}$, out of a total of eight are needed to describe the variability about the typical sample:

Element	$f^{(1)}$	$f^{(2)}$	$f^{(3)}$	$f^{(4)}$	$f^{(5)}$
SiO_2	+0.908	+0.007	−0.161	+0.209	+0.309
TiO_2	+0.024	−0.037	−0.126	+0.151	−0.100
Al_2O_3	+0.275	−0.301	+0.567	+0.176	−0.670
FeO-total	+0.177	−0.018	−0.659	−0.427	−0.585
MgO	+0.141	+0.923	+0.255	−0.118	−0.195
CaO	+0.209	−0.226	+0.365	−0.780	+0.207
Na_2O	+0.044	−0.058	−0.0417	+0.302	−0.145
K_2O	+0.003	−0.007	−0.006	+0.073	+0.015

Each role of each of the factors can be understood by examining its elements. Factor 2, for instance, increases the amount of MgO while decreasing mostly Al_2O_3 and CaO, with respect to the typical sample.

The *factor analysis* has reduced the dimensions of variability of the rock data set from eight elements to four factors, improving the effectiveness of scatter plots. *MatLab*'s three-dimensional plotting capabilities are useful in this case, since any three of the four factors can be used as axes and the resulting

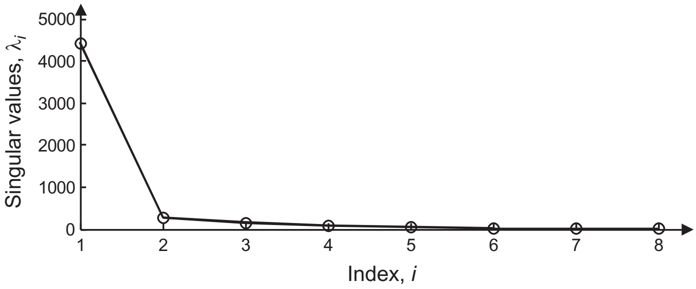

FIGURE 10.5 Singular values λ_i of the Atlantic Ocean Rock dataset. *MatLab* script gda10_04.

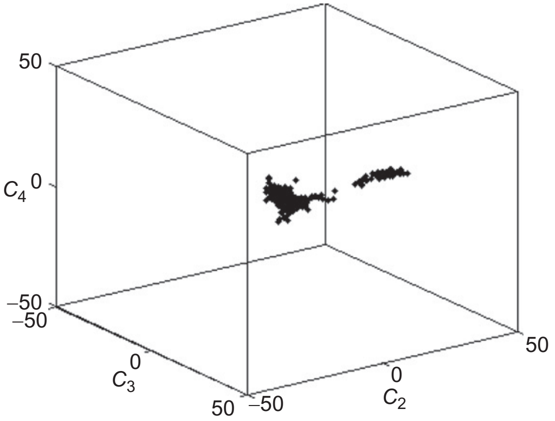

FIGURE 10.6 Three-dimensional perspective view of the coefficients C_i of factors 2, 3, and 4 in each of the rock samples (dots) of the Atlantic Ocean Rock data set. *MatLab* script gda10_04.

three-dimensional scatter plot viewed from a variety of perspectives. The following *MatLab* command plots the coefficients of factors 2 through 4 for each sample:

```
plot3( C(:,2), C(:,3), C(:,4), 'k.' );
```

(*MatLab* script gda10_04)

The plot can then be viewed from different perspectives by using the rotation controls of the Figure Window (Figure 10.6). Note that the samples appear to form two populations, one in which the variability is due to $\mathbf{f}^{(2)}$ and another due to $\mathbf{f}^{(3)}$.

10.2 NORMALIZATION AND PHYSICALITY CONSTRAINTS

In many instances, an element can be important even though it occurs only in trace quantities. In such cases, one cannot neglect factors simply because they have small singular values. They may contain an important amount of the trace elements. It is therefore appropriate to normalize the matrix **S** so that there is a direct correspondence between singular-value size and importance. This is usually done by defining a diagonal matrix of weights **W** (usually proportional to the reciprocal of the standard deviations of measurement of each of the elements) and then forming a new weighted sample matrix $\mathbf{S}' = \mathbf{SW}$.

The singular-value decomposition enables one to determine a set of factors that span, or approximately span, the space of samples. These factors, however, are not unique in the sense that one can form linear combinations of factors that also span the space. This transformation is typically a useful thing to do since, ordinarily, the singular-value decomposition eigenvectors violate *a priori* constraints on what "good" factors should be like. One such constraint is that the factors should have a unit L_1 norm, that is, their elements should sum to one.

If the components of a factor represent fractions of chemical elements, for example, it is reasonable that the elements should sum to 100%. Another constraint is that the elements of both the factors and the factor loadings should be nonnegative. Ordinarily a material is composed of a positive combination of components. Given an initial representation of the samples $\mathbf{S}=\mathbf{CFW}^{-1}$, we could imagine finding a new representation consisting of linear combinations of the old factors, defined by $\mathbf{F}'=\mathbf{TF}$, where $\mathbf{T}$ is an arbitrary $p\times p$ transformation matrix. The problem can then be stated.

Find $\mathbf{T}$ subject to the following constraints:

$$\sum_{j=1}^{M}[\mathbf{F}'\mathbf{W}^{-1}]_{ij}=1 \quad \text{for all } i$$

$$[\mathbf{CT}^{-1}]_{ij}\geq 0 \quad \text{and} \quad [\mathbf{F}'\mathbf{W}^{-1}]_{ij}\geq 0 \quad \text{for all } i \text{ and } j \tag{10.7}$$

These conditions do not uniquely determine $\mathbf{T}$, as can be seen from Figure 10.4. Note that the second constraint is nonlinear in the elements of $\mathbf{T}$. This is a very difficult constraint to implement and in practice is often ignored.

To find a unique solution, one must add some *a priori* information. One possibility is to find a set of factors that maximize some measure of simplicity. One such measure is *spikiness*: the notion that a factor should have only a few large elements, with the other elements being near-zero. Minerals, for example, obey this principle. While a rock can contain upward of 20 chemical elements, typically it will be composed of minerals such as fosterite (Mg_2SiO_4), anorthite ($CaAl_2Si_2O_8$), rutile (TiO_2), etc., each of which contains just a few elements. Spikiness is more or less equivalent to the idea that the elements of the factors should have *high variance*. The usual formula for estimated variance, σ_d^2, of a data set, $\mathbf{d}$, is

$$\sigma_\mathrm{d}^2=\frac{1}{N}\left(\sum_{i=1}^{N}(d_i-\bar{d})^2\right)=\frac{1}{N^2}\left(N\sum_{i=1}^{N}d_i^2-\left(\sum_{i=1}^{N}d_i\right)^2\right) \tag{10.8}$$

Its generalization to a factor, f_i, is

$$\sigma_\mathrm{f}^2=\frac{1}{M^2}\left(M\sum_{i=1}^{M}f_i^4-\left(\sum_{i=1}^{M}f_i^2\right)^2\right) \tag{10.9}$$

Note that this is the variance of the *squares* of the elements of the factors. Thus, a factor, $\mathbf{f}$, has a large variance, ${\sigma_\mathrm{f}}^2$, if the absolute values of its elements have high variation. The signs of the elements are irrelevant.

The *varimax* procedure is a way of constructing a matrix, $\mathbf{T}$, that increases the variance of the factors while preserving their orthogonality (Kaiser, 1958). It is an iterative procedure, with each iteration operating on only one pair of factors, with other pairs being operated upon in subsequent iterations. The idea is to view the

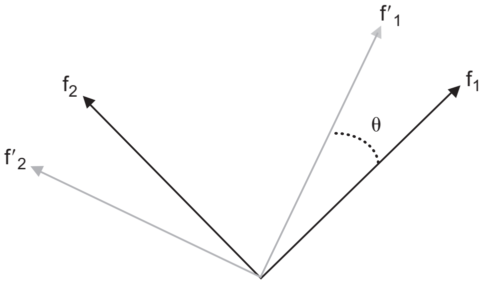

FIGURE 10.7 Two mutually perpendicular factors $\mathbf{f}_1$ and $\mathbf{f}_2$ are rotated in their plane by an angle, θ, creating two new mutually orthogonal vectors, $\mathbf{f}'_1$ and $\mathbf{f}'_2$.

factors as vectors and to rotate them in their plane (Figure 10.7) by an angle, θ, chosen to maximize the sum of their variances. The rotation changes only the two factors, leaving the other $p-2$ factors unchanged, as in the following example:

$$\begin{bmatrix} \mathbf{f}^{(1)\mathrm{T}} \\ \mathbf{f}^{(2)\mathrm{T}} \\ \cos(\theta)\mathbf{f}^{(3)\mathrm{T}} + \sin(\theta)\mathbf{f}^{(5)\mathrm{T}} \\ \mathbf{f}^{(4)\mathrm{T}} \\ -\sin(\theta)\mathbf{f}^{(3)\mathrm{T}} + \cos(\theta)\mathbf{f}^{(3)\mathrm{T}} \\ \mathbf{f}^{(6)\mathrm{T}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \cos(\theta) & 0 & \sin(\theta) & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\sin(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{f}^{(1)\mathrm{T}} \\ \mathbf{f}^{(2)\mathrm{T}} \\ \mathbf{f}^{(3)\mathrm{T}} \\ \mathbf{f}^{(4)\mathrm{T}} \\ \mathbf{f}^{(5)\mathrm{T}} \\ \mathbf{f}^{(6)\mathrm{T}} \end{bmatrix} \tag{10.10}$$

or $\mathbf{F}' = \mathbf{TF}$. Here, only the pair, $\mathbf{f}^{(3)}$ and $\mathbf{f}^{(5)}$, is changed.

In Equation (10.10), the matrix, $\mathbf{T}$, represents a rotation of *one pair* of vectors. The rotation matrix for many such rotations is just the product of a series of pair-wise rotations. Note that the matrix, $\mathbf{T}$, obeys the rule, $\mathbf{T}^{-1} = \mathbf{T}^{\mathrm{T}}$ (that is, $\mathbf{T}$ is a *unary* matrix). For a given pair of factors, $\mathbf{f}^A$ and $\mathbf{f}^B$, the rotation angle θ is determined by minimizing $\Phi(\theta) = M^2({\sigma_{fA}}^2 + {\sigma_{fB}}^2)$ with respect to θ (that is, by solving $\mathrm{d}\Phi/\mathrm{d}\theta = 0$).

The minimization requires a substantial amount of algebraic and trigonometric manipulation, so we omit it here. The result is (Kaiser, 1958)

$$\theta = \frac{1}{4}\tan^{-1}\frac{2M\sum_i u_i v_i - \sum_i u_i \sum_i v_i}{M\sum_i (u_i^2 - v_i^2) - \left(\left(\sum_i u_i\right)^2 - \left(\sum_i v_i\right)^2\right)} \quad \text{with} \tag{10.11}$$

$$u_i = \left(f_i^A\right)^2 - \left(f_i^B\right)^2 \quad \text{and} \quad v_i = 2f_i^A f_i^B$$

By way of example, we note that the two vectors

$$\mathbf{f}^A = \frac{1}{2}[1 \;\; 1 \;\; 1 \;\; 1]^{\mathrm{T}} \quad \text{and} \quad \mathbf{f}^B = \frac{1}{2}[1 \;\; -1 \;\; 1 \;\; -1]^{\mathrm{T}} \tag{10.12}$$

are extreme examples of two *nonspiky* orthogonal vectors because all their elements have the same absolute value. When applied to them, the varimax procedure returns

$$\mathbf{f}^{A'} = \frac{1}{\sqrt{2}}[1\ \ 0\ \ 1\ \ 0]^{\mathrm{T}} \quad \text{and} \quad \mathbf{f}^{B'} = \frac{1}{\sqrt{2}}[0\ \ -1\ \ 0\ \ -1]^{\mathrm{T}} \tag{10.13}$$

which are significantly spikier than the originals. The *MatLab* code is

```
u = fA.^2 - fB.^2;
v = 2* fA .* fB;

A = 2*M*u'*v;
B = sum(u)*sum(v);
top = A - B;

C = M*(u'*u-v'*v);
D = (sum(u)^2)-(sum(v)^2);
bot = C - D;
q = 0.25 * atan2(top,bot);

cq = cos(q);
sq = sin(q);

fAp = cq*fA + sq*fB;
fBp = -sq*fA + cq*fB;
```

(*MatLab* gda10_05)

Here, the original pair of factors are `fA` and `fB`, and the rotated pair are `fAp` and `fBp`.

We now apply this procedure to factors $\mathbf{f}_2$ through $\mathbf{f}_5$ of the Atlantic Rock data set (that is, the factors related to deviations about the typical rock). The varimax procedure is applied to all pairs of these factors and achieves convergence after several such iterations. The *MatLab* code for the loops are

```
FP=F;

% spike these factors using the varimax procedure
k = [2, 3, 4, 5]';
Nk = length(k);

for iter = [1:3]
for ii = [1:Nk]
for jj = [ii+1:Nk]

% spike factors i and j
i=k(ii);
j=k(jj);
```

```
% copy factors from matrix to vectors
fA = FP(i,:)';
fB = FP(j,:)';

% standard varimax procedure to determine rotation angle q
- - -

% copy rotated factors back to matrix
FP(i,:) = fAp';
FP(j,:) = fBp';

end
end
end
```

(*MatLab* gda10_05)

Here the rotated matrix of factors `FP` is initialized to the original matrix of factors `F` and then modified by the varimax procedure (omitted and replaced with a "- - -"), with each pass through the inner loop rotating one pair of factors. The procedure converges very rapidly, with three iterations of the outside loop being sufficient. The resulting factors (Figure 10.8) are much spikier than the original ones. Each now involves mainly variations in one chemical element. For example, $\mathbf{f}'_2$ mostly represents variations in MgO, and $\mathbf{f}'_5$ mostly represents variations in Al_2O_3.

Another possible way of adding *a priori* information is to find factors that are in some sense close to a set of *a priori* factors. If closeness is measured by the L_1 or L_2 norm and if the constraint on the positivity of the factor loadings is

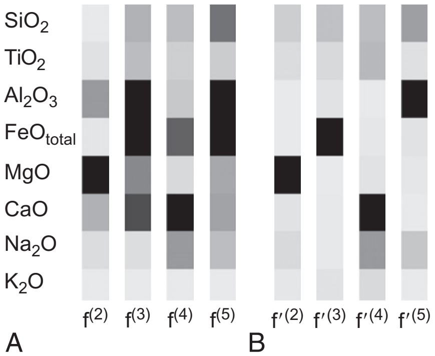

FIGURE 10.8 (A) Factors $\mathbf{f}^{(2)}$ through $\mathbf{f}^{(5)}$ of the Atlantic Rock data set, as calculated by singular-value decomposition. (B) Factors $\mathbf{f}'^{(2)}$ through $\mathbf{f}'^{(5)}$, after application of the varimax procedure. *MatLab* script gda10_05.

omitted, then this problem can be solved using the techniques of Chapters 7 and 12. One advantage of this latter approach is that it permits one to test whether a particular set of *a priori* factors can be factors of the problem (that is, whether or not the distance between *a priori* factors and actual factors can be reduced to an insignificant amount).

10.3 Q-MODE AND R-MODE FACTOR ANALYSIS

The eigenvectors $\mathbf{U}$ and $\mathbf{V}$ play completely symmetric roles in the singular-value decomposition of the sample matrix $\mathbf{S}=\mathbf{U}\boldsymbol{\Lambda}\mathbf{V}^{\mathrm{T}}$. We introduced an asymmetry when we grouped them as $\mathbf{S}=(\mathbf{U}\boldsymbol{\Lambda})(\mathbf{V}^{\mathrm{T}})=\mathbf{CF}$ to define the loadings $\mathbf{C}$ and factors $\mathbf{F}$.

This grouping is associated with the term *R-mode factor analysis*. It is appropriate when the focus is on patterns among the elements, which is to say, reducing a large number of elements to a smaller number of factors. Thus, for example, we might note that the elements in the Atlantic Rock data set contain a pattern, associated with factor $\mathbf{f}^{(2)}$, in which Al_2O_3 and MgO are strongly and negatively correlated and another pattern, associated with factor $\mathbf{f}^{(3)}$, in which Al_2O_3 and FeO_{total} are strongly and negatively correlated. The effect of these correlations is to reduce the effective number of elements, that is, to allow us to substitute a small number of factors for a large number of elements.

Alternately, we could have grouped the singular-value decomposition as $\mathbf{S}=(\mathbf{U})(\boldsymbol{\Lambda}\mathbf{V}^{\mathrm{T}})$, an approach associated with the term *Q-mode factor analysis*. The equivalent transposed form $\mathbf{S}^{\mathrm{T}}=(\mathbf{V}\boldsymbol{\Lambda})(\mathbf{U}^{\mathrm{T}})$ is more frequently encountered in the literature and is also more easily understood, since it can be interpreted as "normal factor analysis" applied to the matrix $\mathbf{S}^{\mathrm{T}}$. The transposition has reversed the sense of samples and elements, so the factor matrix $\mathbf{U}^{\mathrm{T}}$ quantifies patterns of variability among samples, in the same way that $\mathbf{V}^{\mathrm{T}}$ quantifies patterns of variability among elements. To pursue this comparison further, consider an R-mode problem in which there is only one factor, $\mathbf{v}^{(1)}=[1, 1, \ldots]^{\mathrm{T}}$. This factor implies that all of the samples in the data table contain a 1:1 ratio of elements 1 and 2. Similarly, a Q-mode problem in which there is only one factor, $\mathbf{u}^{(1)}=[1, 1, \ldots]^{\mathrm{T}}$ implies that all of the elements in the transposed data table have a 1:1 ratio of samples 1 and 2. This approach is especially useful in detecting clustering among the samples; indeed, Q-mode factor analysis is often referred to as a form of *cluster analysis*.

10.4 EMPIRICAL ORTHOGONAL FUNCTION ANALYSIS

Factor analysis need not be limited to data that contain actual mixtures of components. Given any set of vectors $\mathbf{s}^{(i)}$, one can perform the singular-value decomposition and represent $\mathbf{s}^{(i)}$ as a linear combination of a set of orthogonal factors. Even when the factors have no obvious physical interpretation, the decomposition can be useful as a tool for quantifying the similarities between the

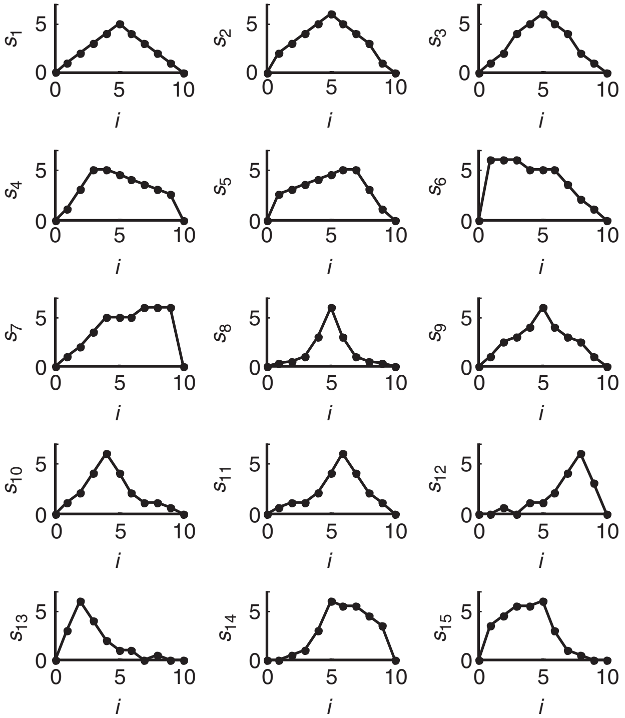

FIGURE 10.9 A set of hypothetical mountain profiles. The variability of shape will be determined using factor analysis. *MatLab* script gda10_06

$\mathbf{s}^{(i)}$ vectors. This kind of factor analysis is often called *empirical orthogonal function* (*EOF*) analysis.

As an example of this application of factor analysis, consider the set of $N = 14$ shapes shown in Figure 10.9. These shapes might represent profiles of mountains or other subjects of interest. The problem we shall consider is how these profiles might be ordered to bring out the similarities and differences between the shapes. A geomorphologist might desire such an ordering because, when combined with other kinds of geological information, it might reveal the kinds of erosional processes that cause the shape of mountains to evolve with time.

We begin by discretizing each profile and representing it as a unit vector (in this case of length $M = 11$). These unit vectors make up the matrix $\mathbf{S}$, on which we perform factor analysis. Since the factors do not represent any particular physical object, there is no need to impose any positivity constraints on them, and we use the untransformed singular-value decomposition factors. The three most important EOFs (factors) (that is, the ones with the three largest singular values) are shown in Figure 10.10. The first EOF, as expected, appears to be simply an "average" mountain; the second seems to control the skewness, or degree of asymmetry, of the mountain; and the third, the sharpness of the mountain's summit.

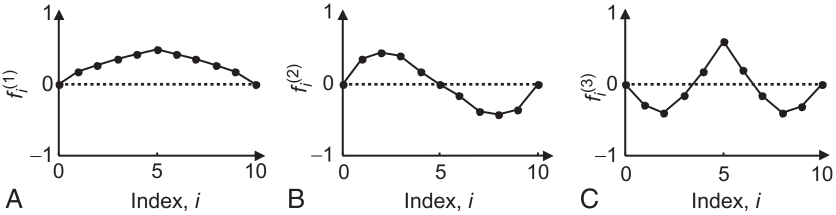

FIGURE 10.10 The three largest factors $\mathbf{f}^{(1)}$, $\mathbf{f}^{(2)}$, and $\mathbf{f}^{(3)}$ in the representation of the mountain profiles in Figure 10.9. (A) The factor with the largest singular value $\lambda_1 = 38.4$ has the shape of the average profile. (B) The factor with the second largest singular value $\lambda_2 = 12.7$ quantifies the asymmetry of the profiles. (C) The factor with the third largest singular value $\lambda_3 = 7.4$ quantifies the sharpness of the mountain summits. *MatLab* script gda10_06.

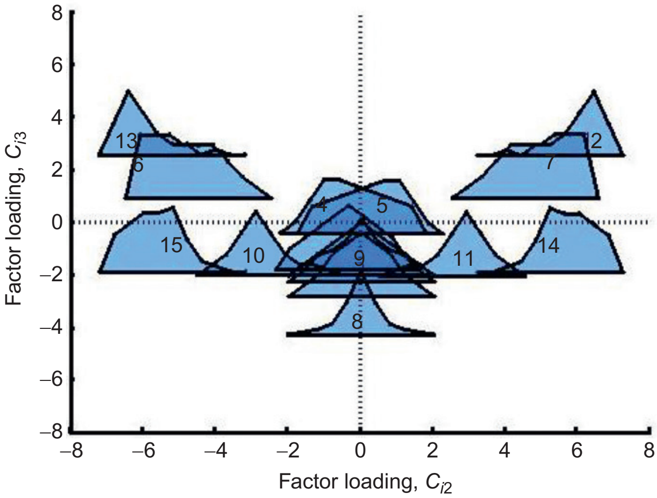

FIGURE 10.11 The mountain profiles of Figure 10.9, arranged according to the relative amounts of factors 2 and 3 contained in each profile's orthogonal decomposition; that is, by the size of the factor loadings C_{i2} and C_{i3}. *MatLab* script gda10_06.

We emphasize, however, that this interpretation was made after the EOF analysis and was not based on any *a priori* notions of how mountains might differ. We can then use the loadings as a measure of the similarities between the mountains. Since the amount of the first factor does not vary much between mountains, we use a two-dimensional ordering based on the relative amounts of the second and third factors in each of the mountain profiles (Figure 10.11).

EOF analysis is especially useful when the data have an ordering in space, time, or some other sequential variable. For instance, suppose that profiles in Figure 10.9 are measured at sequential times. Then we can understand the model equation $\mathbf{S} = \mathbf{CF}$ to mean

$$S(t_i, x_j) = \sum_{k=1}^{p} C_k(t_i) F_k(x_j) \tag{10.14}$$

Here the quantity $S(t_i, x_j)$, which varies with both time and space, has been broken up into the sum of two sets of function, the EOFs $F_k(x_j)$, which vary only with space, and the corresponding loadings $C_k(t_i)$, which vary only with time. A plot of the kth loading $C_k(t_i)$ as a function of time t_i reveals how the importance of the kth EOF varies with time.

This analysis can be extended to functions of two or more spatial dimensions by writing Equation (10.14) as

$$S(t_i, \mathbf{x}^{(j)}) = \sum_{k=1}^{P} C_k(t_i) F_k(\mathbf{x}^{(j)}) \tag{10.15}$$

Here the spatial observation points $\mathbf{x}$ are multidimensional, but they have been given a linear ordering through the index k. As an example, suppose that the data are a sequence of two-dimensional images, where each image represents a physical parameter observed on the (x, y) plane. This image can be *unfolded* (reordered) into a vector (Figure 10.12), which then becomes a row of the sample matrix **S**. The resulting EOFs have this same ordering and must be folded back into two-dimensional images before being interpreted.

As an example, we consider a sequence of $N = 25$ images, each of which contains a grid of $20 \times 20 = 400$ pixels (Figure 10.13). Each image represents the spatial variation of a physical parameter such as pressure or temperature at a fixed time, with the overall sequence being time-sequential. These data are synthetic and are constructed by summing three spatial patterns (EOFs) with coefficients (loadings) that vary systematically with time, and then adding random noise. As expected, only $p = 3$ singular values are found to be significant (Figure 10.14).

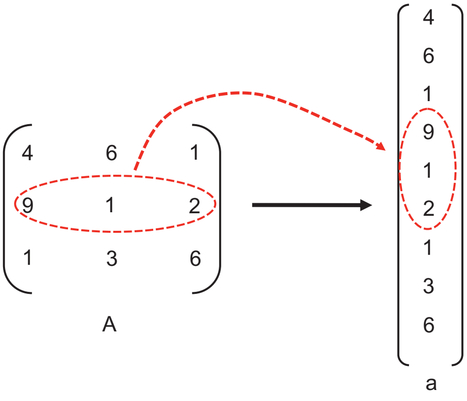

FIGURE 10.12 A matrix **A** representing a discrete version of a two-dimensional function $A_{ij} = a(x_i, y_j)$ is unfolded row-wise into a vector **a** using the rule $a_k = A_{ij}$ with $k = (i-1)M + j$.

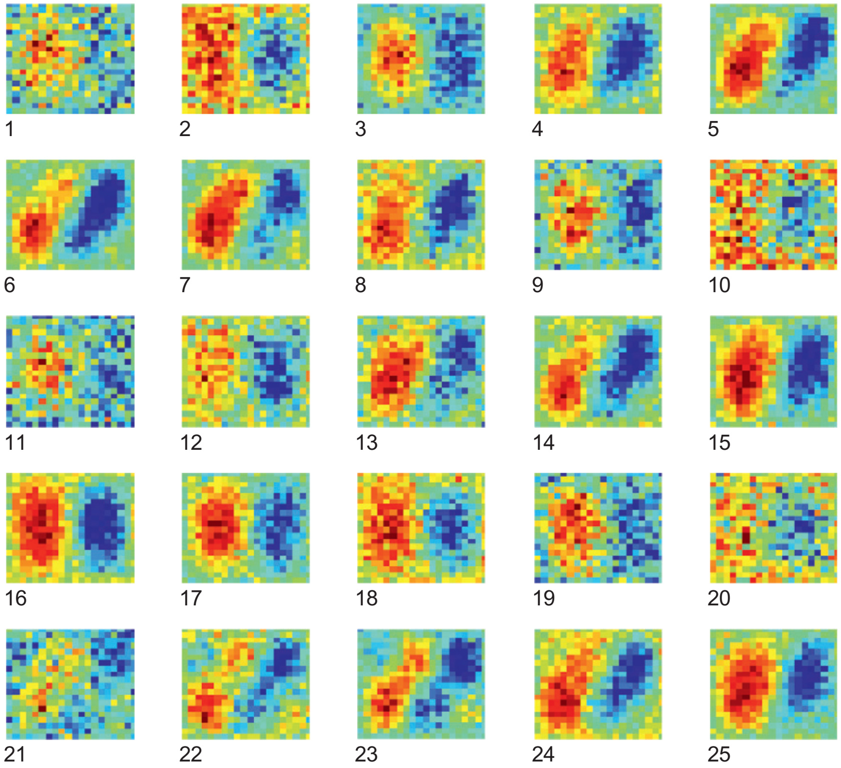

FIGURE 10.13 Time sequence of $N = 25$ images. Each image represents a parameter, such as pressure or temperature, that varies spatially in the (x, y) plane. *MatLab* script gda10_07.

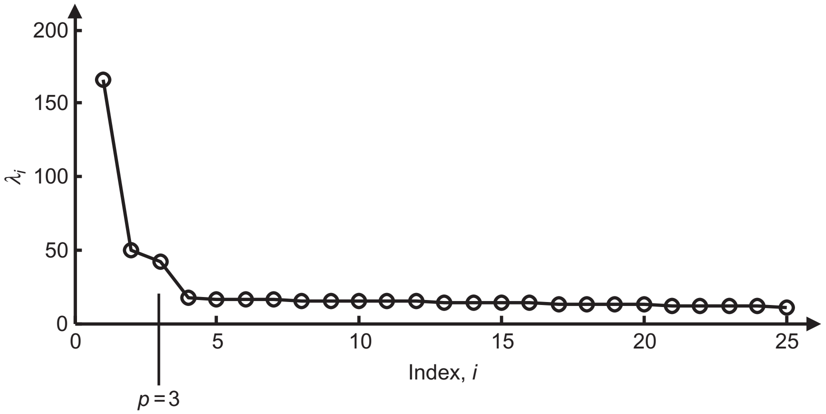

FIGURE 10.14 Singular values λ_i of the image sequence shown in Figure 10.12. Only $p = 3$ singular values have significant amplitudes. *MatLab* script gda10_07.

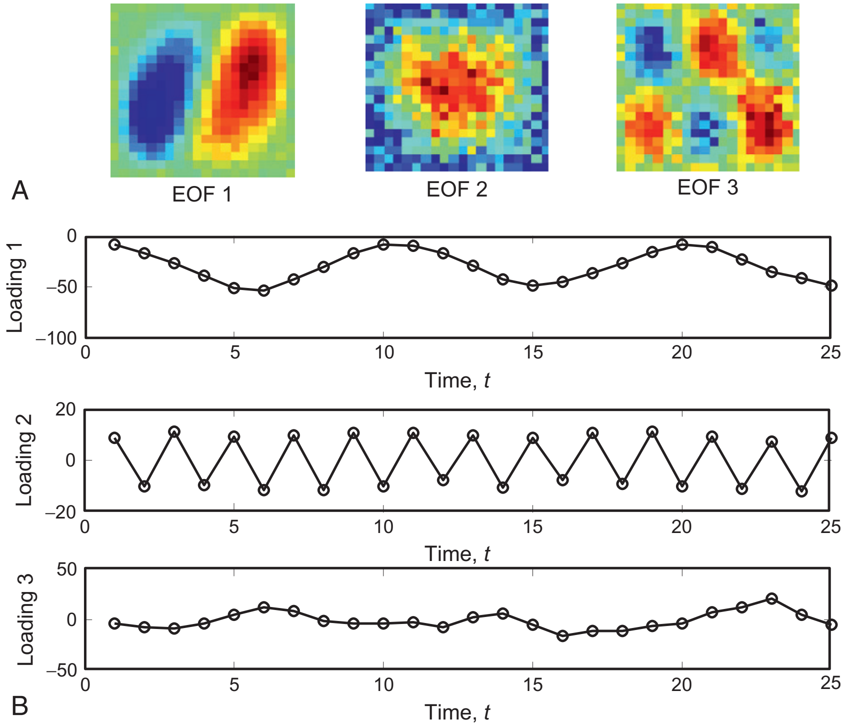

FIGURE 10.15 (A) First three empirical orthogonal functions (EOFs) of the image sequence shown in Figure 10.12. (B) Corresponding loadings as a function of time, *t*. *MatLab* script gda10_07.

The corresponding EOFs and loadings are shown in Figure 10.15. Had this analysis been based upon actual data, the time variation of each of the loadings, which have different periodicities, would be of special interest and might possibly provide insight into the physical processes associated with each of the EOFs. A similar example that uses actual ocean temperature data to examine the El Nino-Southern Oscillation climate instability is given by Menke and Menke (2011, their Section 8.5).

10.5 PROBLEMS

10.1 Suppose that a set of $N > M$ samples are represented as $\mathbf{S} \approx \mathbf{C}_p\mathbf{F}_p$ where the matrix $\mathbf{F}_p$ contains $p < M$ factors whose values are prescribed (that is, known *a priori*). (A) How can the loadings $\mathbf{C}_p$ be determined? (B) Write a *MatLab* script that implements your procedure for the case $M = 3, N = 10, p = 2$. (C) Make a three-dimensional plot of your results.

10.2 Write a *MatLab* script that verifies the varimax result given in Equation (10.12). Use the following steps. (A) Compute the factors $\mathbf{f}'^{A}$ and $\mathbf{f}'^{B}$ for a complete suite of angles θ using the rotation

$$\begin{aligned}\mathbf{f}'^{A} &= \cos(\theta)\mathbf{f}^{A} + \sin(\theta)\mathbf{f}^{B} \\ \mathbf{f}'^{B} &= -\sin(\theta)\mathbf{f}^{A} + \cos(\theta)\mathbf{f}^{B}\end{aligned}$$

(B) Compute and plot the variance $\sigma^2_{fA'} + \sigma^2_{fB'}$ as a function of angle θ. (C) Note the angle of the minimum variance and verify that the angle is the one predicted by the varimax formula. (D) Verify that the factors corresponding to the angle are as stated in Equation (10.12).

10.3 Suppose that a data set represents a function of three spatial dimensions; that is, with samples $S(t, x, y, z)$ on an evenly spaced three-dimensional grid. How can these samples be unfolded into a matrix, $\mathbf{S}$?

REFERENCES

Kaiser, H.F., 1958. The varimax criterion for analytic rotation in factor analysis. Psychometrika 23, 187–200.

Menke, W., Menke, J., 2011. Environmental Data Analysis with MatLab. Academic Press, Elsevier Inc, Oxford, UK 263pp.