CHAPTER NINETEEN

Matrix Computation

To set up a philosophy against physics is rash; philosophers who have done so have always ended in disaster.

Bertrand Russell

Matrix computation pervades many discussions on numerical and statistical techniques. Two major concerns here are **multivariate statistical analysis** and **curve fitting** with **splines**. Both have extensive applications in statistical inference. **Factor analysis** is presented as an interesting application.

19.1 Fundamental Definitions and Results

Let $A \equiv [\, a_{ij} \,]_{1 \le i \le m, 1 \le j \le n}$, or simply $A \in \boldsymbol{R}^{m \times n}$, denote an $m \times n$ matrix. It can also be represented as $[\, a_1, a_2, \ldots, a_n \,]$, where $a_i \in \boldsymbol{R}^m$ are vectors. Vectors are column vectors unless stated otherwise. It is a **square matrix** when $m = n$. The **rank** of a matrix is the largest number of linearly independent columns. An $m \times n$ matrix is **rank deficient** if its rank is less than $\min(m, n)$; otherwise, it has **full rank**. It has full **column rank** if its rank equals n: All of its columns are linearly independent. A square matrix A is said to be **symmetric** if $A^{\mathrm{T}} = A$. A real $n \times n$ matrix $A \equiv [\, a_{ij} \,]_{i,j}$ is **diagonally dominant** if $|a_{ii}| > \sum_{j \ne i} |a_{ij}|$ for $1 \le i \le n$. Such matrices are nonsingular [224]. A **leading principal submatrix** of an $n \times n$ matrix is a submatrix consisting of the first k rows and the first k columns for some $1 \le k \le n$. The expression $\| x \| \equiv \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$ denotes the length of vector $x \equiv [\, x_1, x_2, \ldots, x_n \,]^{\mathrm{T}}$. It is also called the **Euclidean norm**. A **diagonal** $m \times n$ **matrix** $D \equiv [\, d_{ij} \,]_{i,j}$ may be denoted by $\mathrm{diag}[\, D_1, D_2, \ldots, D_q \,]$, where $q \equiv \min(m, n)$ and $D_i = d_{ii}$ for $1 \le i \le q$ (see Fig. 19.1). The **identity matrix** is the square matrix $I \equiv \mathrm{diag}[\, 1, 1, \ldots, 1 \,]$.

A vector set $\{ x_1, x_2, \ldots, x_p \}$ is **orthogonal** if all its vectors are nonzero and the inner products $x_i^{\mathrm{T}} x_j$ equal zero for $i \ne j$. It is **orthonormal** if, furthermore,

$$x_i^{\mathrm{T}} x_j = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases}.$$

A real square matrix Q is said to be **orthogonal**[1] if $Q^{\mathrm{T}} Q = I$. For such matrices, $Q^{-1} = Q^{\mathrm{T}}$ and $QQ^{\mathrm{T}} = I$. A real symmetric matrix A is **positive definite** (**positive semidefinite**) if $x^{\mathrm{T}} A x = \sum_{i,j} a_{ij} x_i x_j > 0$ ($x^{\mathrm{T}} A x \ge 0$, respectively) for any nonzero vector x. It is known that a matrix A is positive semidefinite if and only if there

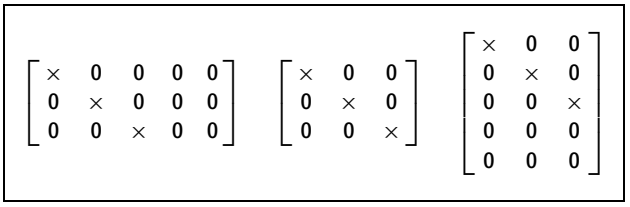

Figure 19.1: Diagonal matrices. Three basic forms of $m \times n$ diagonal matrices corresponding to (left to right) $m < n$, $m = n$, and $m > n$.

exists a matrix W such that $A = W^{\mathrm{T}}W$; A is positive definite if and only if this W has full column rank.

19.1.1 Gaussian Elimination

Gaussian elimination is a standard method for solving a linear system $Ax = b$, where $A \in \boldsymbol{R}^{n \times n}$. It is due to Gauss in 1809 [825]. After $O(n^3)$ operations, the system is transformed into an equivalent system $A'x = b'$, where A' is an upper triangular $n \times n$ matrix. This is the **forward-reduction** phase. **Backward substitution** is then applied to compute x after $O(n^2)$ more operations. The total running time is therefore cubic. See Fig. 19.2 for the algorithm.

Efficiency can often be improved if A has special structures. A case in point is when A is **banded**, that is, if all the nonzero elements are placed near the diagonal of the matrix. We say that $A = [\, a_{ij} \,]_{i,j}$ has **upper bandwidth** u if $a_{ij} = 0$ for $j - i > u$ and **lower bandwidth** l if $a_{ij} = 0$ for $i - j > l$. A tridiagonal matrix, for instance, has upper bandwidth one and lower bandwidth one (see Fig. 19.3). For banded matrices, Gaussian elimination can be easily modified to run in $O(nul)$ time by skipping

Gaussian elimination algorithm:

```
input:   A[1..n][1..n], b[1..n];
real     x[1..n];
integer  i, j, k;
// Forward reduction.
for (k = 1 to n-1)
        for (i = k+1 to n) {
                c := A[i][k]/A[k][k];
                for (j = k+1 to n)
                        A[i][j] := A[i][j] - c × A[k][j];
                b[i] := b[i] - c × b[k];
        }
// Backward substitution.
for (k = n down to 1)
        x[k] := (b[k] - Σ_{j=k+1}^{n} A[k][j] × x[j])/A[k][k];
return x[];
```

Figure 19.2: Gaussian elimination algorithm. This algorithm solves $Ax = b$ for x. It assumes that the diagonal elements $A[k][k]$ are nonzero throughout. This can be guaranteed if all of A's leading principal submatrices are nonsingular.

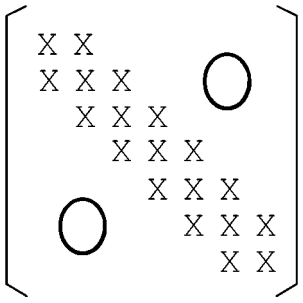

Figure 19.3: Banded matrix. Shown here is a tridiagonal matrix.

elements that are zero. This bound is substantially smaller than n^3 when u or l is small.

Gaussian elimination can be used to factor any square matrix whose leading principal submatrices are nonsingular into a product of a lower triangular matrix L and an upper triangular matrix U: $A = LU$. This is called the **LU decomposition**. The conditions are satisfied by positive definite matrices and diagonally dominant matrices. Positive definite matrices can in fact be factored as $A = LL^{\mathrm{T}}$, called the **Cholesky decomposition**. See Fig. 19.4 for a cubic-time algorithm.

19.1.2 Eigenvalues and Eigenvectors

An **eigenvalue** of a square matrix A is a complex number λ such that $Ax = \lambda x$ for some nonzero vector x, called an **eigenvector**. For example, because

$$\begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 0.707107 & -0.707107 \\ 0.707107 & 0.707107 \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 0.707107 & 0.707107 \\ -0.707107 & 0.707107 \end{bmatrix},$$

the two eigenvalues are 3 and -1, and $[\,0.707107, -0.707107\,]^{\mathrm{T}}$ and $[\,0.707107, 0.707107\,]^{\mathrm{T}}$ are the corresponding eigenvectors. The eigenvalues for a real symmetric matrix are real numbers; in particular, the **Schur decomposition theorem** (also called the **principal-axes theorem** or the **spectral theorem**) says that there exists a real orthogonal matrix Q such that $Q^{\mathrm{T}}AQ = \mathrm{diag}[\,\lambda_1, \lambda_2, \ldots, \lambda_n\,]$. Note that Q's

Cholesky decomposition algorithm:

```
input:   A[1..n][1..n];
real     L[1..n][1..n];
integer  i, j, k;
for (j = 1 to n) {
        L[j][j] := (A[j][j] − Σ_{k=1}^{j−1} L[j][k]^2)^{1/2};
        for (i = j + 1 to n) {
                L[i][j] := L[j][j]^{−1} × (A[i][j] − Σ_{k=1}^{j−1} L[i][k] × L[j][k]);
                L[j][i] := 0;
        }
}
return L[ ][ ];
```

Figure 19.4: Cholesky decomposition. The algorithm solves $A = L\,L^{\mathrm{T}}$ for positive definite A.

ith column is the eigenvector corresponding to λ_i, and the eigenvectors form an orthonormal set. The eigenvalues of positive definite matrices are furthermore positive.

Principal Components

Let $\boldsymbol{x} \equiv [\, x_1, x_2, \ldots, x_n \,]^{\mathrm{T}}$ be a vector random variable with the covariance matrix

$$C \equiv [\, \mathrm{Cov}[\, x_i, x_j \,] \,]_{1 \le i,j \le n} = E[\, \boldsymbol{x}\boldsymbol{x}^{\mathrm{T}} \,] \not\equiv \mathbf{0},$$

where $E[\, \boldsymbol{x} \,] = \mathbf{0}$ and $\mathbf{0}$ denotes the zero vector. Covariance matrices are positive definite provided that individual variances $\mathrm{Var}[\, x_i \,]$ are all positive, which will be assumed throughout [343]. C's eigenvalues $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_n > 0$ are therefore real.

We are interested in knowing which normalized linear combinations of the x_is give rise to the maximum variance, thus explaining most of the total variability of the system. Mathematically, we seek a vector $b \equiv [\, b_1, b_2, \ldots, b_n \,]^{\mathrm{T}}$ with $\|b\| = 1$ that maximizes

$$\mathrm{Var}[\, b^{\mathrm{T}}\boldsymbol{x} \,] \equiv E[\, (b^{\mathrm{T}}\boldsymbol{x})^2 \,] = E[\, (b_1x_1 + b_2x_2 + \cdots + b_nx_n)^2 \,].$$

The answer turns out to be simple: The maximum variance equals C's largest eigenvalue λ_1, and its corresponding eigenvector u_1 is the b we are after. Among the normalized linear combinations of the x_i's uncorrelated with $u_1^{\mathrm{T}}\boldsymbol{x}$, which gives rise to the maximum variance? The answer is similar: The maximum variance is λ_2, and its corresponding eigenvector u_2 produces the desired linear combination $u_2^{\mathrm{T}}\boldsymbol{x}$. This process can be repeated for a total of n steps, with the jth time leading to a normalized linear combination uncorrelated with all the previous $j-1$ combinations that has the maximum variance λ_j [23].

The proof goes like this. By the Schur decomposition theorem, the real orthogonal matrix B whose columns are the orthonormal eigenvectors satisfies

$$\Sigma \equiv \mathrm{diag}[\, \lambda_1, \lambda_2, \ldots, \lambda_n \,] = B^{\mathrm{T}}CB, \tag{19.1}$$

with $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_n$. The vector of **principal components** of $\boldsymbol{x}$,

$$P \equiv [\, p_1, p_2, \ldots, p_n \,]^{\mathrm{T}} = B^{\mathrm{T}}\boldsymbol{x}, \tag{19.2}$$

has the covariance matrix

$$E[\, PP^{\mathrm{T}} \,] = E[\, B^{\mathrm{T}}\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}B \,] = B^{\mathrm{T}}CB = \Sigma.$$

Each principal component p_j is a normalized linear combination of the x_is and corresponds to the jth combination mentioned previously (see Exercise 19.1.2). This identity also confirms that any two distinct components of P are uncorrelated and p_j has variance λ_j.

Principal components can be used to find linear combinations with large variances. Specifically, the p_js in Eq. (19.2) can be considered uncorrelated new variables, and those variables p_j whose corresponding eigenvalue λ_j is small are thrown away. The resulting variables can be much smaller in number than the original variables $x_1, x_2, \ldots, x_n$.

➤ **Exercise 19.1.1** Let $C \equiv [\, c_{ij} \,]$ denote the covariance matrix of $\boldsymbol{x} \equiv [\, x_1, x_2, \ldots, x_n \,]^{\mathrm{T}}$. Show that C and $\boldsymbol{x}$'s correlation matrix P are related by $C = \Sigma P \Sigma$, where $\Sigma \equiv \mathrm{diag}[\, \sqrt{c_{11}}, \sqrt{c_{22}}, \ldots, \sqrt{c_{nn}} \,]$. (Hence, C is positive definite if and only if P is.)

➤ **Exercise 19.1.2** Prove that the jth principal component p_j has the maximum variance among all the normalized linear combinations of the x_is that are uncorrelated with $p_1, p_2, \ldots, p_{j-1}$.

Generation of the Multivariate Normal Distribution

Let $\boldsymbol{x} \equiv [\, x_1, x_2, \ldots, x_n \,]^{\mathrm{T}}$ be a vector random variable with a positive definite covariance matrix C. As usual, assume that $E[\, \boldsymbol{x} \,] = \boldsymbol{0}$. This distribution can be generated by $P\boldsymbol{y}$, where $C = PP^{\mathrm{T}}$ is the Cholesky decomposition of C and $\boldsymbol{y} \equiv [\, y_1, y_2, \ldots, y_n \,]^{\mathrm{T}}$ is a vector random variable with a covariance matrix equal to the identity matrix. This holds because $\mathrm{Cov}[\, P\boldsymbol{y} \,] = P\,\mathrm{Cov}[\, \boldsymbol{y} \,]\, P^{\mathrm{T}} = PP^{\mathrm{T}} = C$ (see Exercise 6.1.3).

Suppose we want to generate the multivariate normal distribution with a covariance matrix $C = PP^{\mathrm{T}}$. We start with independent standard normal distributions $y_1, y_2, \ldots, y_n$. Then $P[\, y_1, y_2, \ldots, y_n \,]^{\mathrm{T}}$ has the desired distribution. Generating the multivariate normal distribution is essential for the Monte Carlo pricing of multivariate derivatives.

➤ **Exercise 19.1.3** Verify the correctness of the procedure for generating the bivariate normal distribution in Subsection 6.1.2.

➤ **Exercise 19.1.4** Prove that $W\boldsymbol{X} \sim N(W\boldsymbol{\mu}, WCW^{\mathrm{T}})$ if $\boldsymbol{X} \sim N(\boldsymbol{\mu}, C)$.

19.1.3 The Singular Value Decomposition

The following theorem is the basis for the **singular value decomposition** (**SVD**).

THEOREM 19.1.1 (SVD). *For a real $m \times n$ matrix A,*

$$\Sigma \equiv U^T A V = \mathrm{diag}[\, \sigma_1, \sigma_2, \ldots, \sigma_p \,] \in \boldsymbol{R}^{m \times n}$$

for some orthogonal matrices $U \equiv [\, u_1, u_2, \ldots, u_m \,] \in \boldsymbol{R}^{m \times m}$ and $V \equiv [\, v_1, v_2, \ldots, v_n \,] \in \boldsymbol{R}^{n \times n}$, where $p \equiv \min(m, n)$ and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_p \geq 0$. In fact, $\sigma_1 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_p = 0$ with $\mathrm{rank}(A) = r$. The σ_is are called the **singular values**.

For example,

$$\begin{aligned} \begin{bmatrix} 1 & 2 \\ 1 & 4 \\ 1 & 6 \end{bmatrix} &= \begin{bmatrix} -0.282970 & 0.867906 \\ -0.538373 & 0.208536 \\ -0.793777 & -0.450834 \end{bmatrix} \begin{bmatrix} 7.654435 & 0. \\ 0. & 0.640018 \end{bmatrix} \\ &\quad \times \begin{bmatrix} -0.211005 & -0.977485 \\ 0.977485 & -0.211005 \end{bmatrix} \\ &= U \Sigma V^{\mathrm{T}}. \end{aligned} \tag{19.3}$$

Because U and V are orthogonal, the two singular values are 7.654435 and 0.640018. (The third column of U and the third row of Σ, which is a zero vector, were not shown.)

The SVD can be computed in cubic time and quadratic space in terms of m and n [392, 586]. It is implemented in every decent mathematical software library. Theorem 19.1.1 implies that

$$A = U \Sigma V^{\mathrm{T}} = \sum_{i=1}^{p} \sigma_i u_i v_i^{\mathrm{T}}. \tag{19.4}$$

➤ **Exercise 19.1.5** (1) Prove Eq. (19.4). (2) Show that A and A^{T} share the same singular values.

19.2 Least-Squares Problems

The **least-squares (LS) problem** is concerned with $\min_{x \in R^n} \|Ax - b\|$, where $A \in \boldsymbol{R}^{m \times n}$, $b \in \boldsymbol{R}^m$, $m \geq n$. The LS problem is called **regression analysis** in statistics and is equivalent to minimizing the mean-square error. Often stated as $Ax = b$, the LS problem is **overdetermined** when there are more equations than unknowns ($m > n$).

EXAMPLE 19.2.1 In polynomial regression, $\beta_0 + \beta_1 x + \cdots + \beta_n x^n$ is used to fit the data $\{(x_1, b_1), (x_2, b_2), \ldots, (x_m, b_m)\}$. This leads to the LS problem $Ax = b$, where

$$A \equiv \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \cdots & x_m^n \end{bmatrix}, \quad x \equiv \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_n \end{bmatrix}, \quad b \equiv \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}. \tag{19.5}$$

Linear regression corresponds to $n = 1$.

The LS problem can be solved by a geometric argument. Because Ax is a linear combination of A's columns with coefficients $x_1, x_2, \ldots, x_n$, the LS problem finds the minimum distance between b and A's column space. A solution x_{LS} must identify a point Ax_{LS} that is at least as close to b as any other point in the column space. Therefore the error vector $Ax_{\mathrm{LS}} - b$ must be perpendicular to that space, that is,

$$(Ay)^{\mathrm{T}}(Ax_{\mathrm{LS}} - b) = y^{\mathrm{T}}(A^{\mathrm{T}}Ax_{\mathrm{LS}} - A^{\mathrm{T}}b) = \mathbf{0}$$

for all y. We conclude that any solution x must satisfy the **normal equations**

$$A^{\mathrm{T}}Ax = A^{\mathrm{T}}b. \tag{19.6}$$

➤ **Exercise 19.2.1** What are the normal equations for linear regression (6.13)?

➤ **Exercise 19.2.2** (1) Phrase multiple regression as an LS problem. (2) Write the normal equations.

➤ **Exercise 19.2.3** Let $\Phi(x) \equiv (1/2)\|Ax - b\|^2$. Prove that its **gradient vector**,

$$\nabla\Phi(x) \equiv \left[\frac{\partial\Phi(x)}{\partial x_1}, \frac{\partial\Phi(x)}{\partial x_2}, \ldots, \frac{\partial\Phi(x)}{\partial x_n} \right]^{\mathrm{T}},$$

equals $A^{\mathrm{T}}(Ax - b)$, where $x \equiv [x_1, x_2, \ldots, x_n]^{\mathrm{T}}$. (Normal equations are $\nabla\Phi(x) = \mathbf{0}$).

➤ **Exercise 19.2.4** Define the **bandwidth** of a banded matrix A as $l + u + 1$, where l is the lower bandwidth and u is the upper bandwidth. Prove that $A^{\mathrm{T}}A$'s bandwidth is at most $\omega - 1$ if A has bandwidth ω.

Comment 19.2.2 The result in Exercise 19.2.4 holds under a more generous definition of banded matrices. In that definition, a matrix is banded with bandwidth ω if the nonzero elements of every row lie within a band with width ω [75]. This result can save some computational efforts.

19.2.1 The Full-Rank Case

The LS problem is called the **full-rank LS problem** when A has full column rank. Because $A^{\mathrm{T}}A$ is then nonsingular, the unique solution for normal equations is $x_{\mathrm{LS}} = (A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}b$, which is called the **ordinary least-squares (OLS) estimator**. As $A^{\mathrm{T}}A$ is positive definite, the normal equations can be solved by the Cholesky decomposition. This approach is usually not recommended because its numerical stability is lower than the alternative SVD approach (see Theorem 19.2.3 below) [35, 586, 870].

Suppose the following linear regression model is postulated between x and b:

$$b = Ax + \epsilon, \tag{19.7}$$

where ϵ is a vector random variable with zero mean and finite variance. The Gauss–Markov theorem says that the OLS estimator x_{LS}, now a random variable, is unbiased and has the smallest variance among all unbiased *linear* estimators if ϵ's components have identical, known variance σ^2 and are uncorrelated (the covariance matrix of ϵ is hence $\sigma^2 I$) [802]. Under these assumptions,

$$\mathrm{Cov}[\,x_{\mathrm{LS}}\,] = \sigma^2(A^{\mathrm{T}}A)^{-1}. \tag{19.8}$$

Hence $(A^{\mathrm{T}}A)^{-1}$, properly scaled, is an unbiased estimator of the covariance matrix of x_{LS}. If ϵ is moreover normally distributed, then x_{LS} has the smallest variance among all unbiased estimators, linear or otherwise [422].

➤ **Exercise 19.2.5** Show that the sample residuals of the OLS estimate, $Ax_{\mathrm{LS}} - b$, are orthogonal to the columns of A.

➤ **Exercise 19.2.6** Suppose that $\sigma^2 C$ is the covariance matrix of ϵ for a positive definite C in linear regression model (19.7). How do we solve the LS problem $Ax = b$ by using the Gauss–Markov theorem?

➤ **Exercise 19.2.7** Verify Eq. (19.8).

19.2.2 The (Possibly) Rank-Deficient Case

When A is rank deficient, there are an infinite number of solutions to the LS problem. In this case, we are interested in the x with the minimum length such that $\|Ax - b\|$ is minimized. This x is unique because $\{\, x \in \boldsymbol{R}^m : \|Ax - b\| \text{ is minimized} \,\}$ is convex. The linkage between the SVD and the general LS problem when A may be rank deficient is established in the following theorem.

THEOREM 19.2.3 *Let $U \equiv [\,u_1, u_2, \ldots, u_m\,] \in \boldsymbol{R}^{m\times m}$ and $V \equiv [\,v_1, v_2, \ldots, v_n\,] \in \boldsymbol{R}^{n\times n}$ such that $U^{\mathrm{T}}AV$ is the SVD of $A \in \boldsymbol{R}^{m\times n}$ and r is A's rank. Then*

$$x_{\mathrm{LS}} = \sum_{i=1}^{r} \left(\frac{u_i^T b}{\sigma_i} \right) v_i$$

is the solution to the LS problem and $\|Ax_{\mathrm{LS}} - b\|^2 = \sum_{i=r+1}^{m} (u_i^T b)^2$.

Let $A = U\Sigma V^{\mathrm{T}}$, where $\Sigma \equiv \mathrm{diag}[\,\sigma_1, \sigma_2, \ldots, \sigma_n\,]$, be the SVD of $A \in \boldsymbol{R}^{m\times n}$. Define

$$\Sigma^{+} \equiv \mathrm{diag}\big[\,\sigma_1^{-1}, \sigma_2^{-1}, \ldots, \sigma_r^{-1}, 0, 0, \ldots, 0\,\big] \in \boldsymbol{R}^{n\times m}.$$

Then $x_{\text{LS}} = V\Sigma^{+}U^{\text{T}}b$. The matrix

$$A^{+} \equiv V\Sigma^{+}U^{\text{T}} \tag{19.9}$$

is called the **pseudoinverse** of A. Note that $\|Ax_{\text{LS}} - b\| = \|(I - AA^{+})\,b\|$ and Σ^{+} is the pseudoinverse of Σ. For example, it is not hard to verify numerically that the pseudoinverse of matrix A in Eq. (19.3) is

$$\begin{bmatrix} 4/3 & 1/3 & -2/3 \\ -1/4 & 0 & 1/4 \end{bmatrix}.$$

➤ **Exercise 19.2.8** Prove that if the A in Theorem 19.2.3 has full column rank, then $A^{+} = (A^{\text{T}}A)^{-1}A^{\text{T}}$. (This implies, in particular, that $A^{+} = A^{-1}$ when A is square.)

➤ **Exercise 19.2.9** Prove that the pseudoinverse of the pseudoinverse is itself, i.e., $(A^{+})^{+} = A$.

➤ **Exercise 19.2.10 (Underdetermined Linear Equations)** Suppose as before that $A \in \boldsymbol{R}^{m\times n}$ and $b \in \boldsymbol{R}^{m}$, but $m \le n$. Assume further that $m = \text{rank}(A)$. Let $U\Sigma V^{\text{T}}$ be the SVD of A. Argue that all solutions to $Ax = b$ are of the form

$$\widehat{x} = V\Sigma^{+}U^{\text{T}}b + V\begin{bmatrix} \mathbf{0} \\ y \end{bmatrix}\begin{matrix} \}\,m \\ \}\,n-m \end{matrix} = A^{+}b + V_2 y \tag{19.10}$$

for arbitrary $y \in \boldsymbol{R}^{n-m}$, where $V \equiv [\ \underbrace{V_1}_{m}, \underbrace{V_2}_{n-m}\ ]\,\}\,n$.

19.2.3 The Weighted Least-Squares Problem

The **weighted LS problem** is concerned with

$$\min_{x\in R^{n}} \|WAx - Wb\|, \tag{19.11}$$

where $W \in \boldsymbol{R}^{m\times m}$ is nonsingular. Clearly all the above-mentioned results regarding the LS problem apply after we replace their A with WA and their b with Wb. In particular, the solution x satisfies the **weighted normal equations**:

$$(WA)^{\text{T}}WAx = (WA)^{\text{T}}Wb.$$

With $H \equiv W^{\text{T}}W$ (hence positive definite), the above equations can be restated as

$$A^{\text{T}}HAx = A^{\text{T}}Hb. \tag{19.12}$$

In particular, if A has full column rank, then the unique solution for x is

$$(A^{\text{T}}HA)^{-1}\,A^{\text{T}}Hb. \tag{19.13}$$

Classic regression analysis assumes that the rows of $Ax - b$ have zero mean and a covariance matrix C. The regression model is hence $b = Ax + \epsilon$, where ϵ is a vector random variable with zero mean and $\text{Cov}[\,\epsilon\,] = C$. The optimal solution for x then satisfies

$$A^{\text{T}}C^{-1}Ax = A^{\text{T}}C^{-1}b. \tag{19.14}$$

(The H^{-1} in Eq. (19.12) thus plays the role of the covariance matrix.) When A has full column rank, the unique optimal solution $(A^{\mathrm{T}}C^{-1}A)^{-1}A^{\mathrm{T}}C^{-1}b$ is called the **generalized least-squares (GLS) estimator**, which is unbiased with covariance matrix $(A^{\mathrm{T}}C^{-1}A)^{-1}$.

➤ **Exercise 19.2.11** Prove Eq. (19.14). (Hint: Exercise 19.2.6.)

19.2.4 The Least-Squares Problem with Side Constraints

The **constrained LS problem** is an LS problem over a proper subset of $\boldsymbol{R}^n$. We are mainly interested in problems with linear equality constraints:

$$\min_{x\in\boldsymbol{R}^n,\ Bx=d} \|Ax-b\|, \quad B\in\boldsymbol{R}^{p\times n},\ d\in\boldsymbol{R}^p.$$

Assume that p is the rank of B. When the solution is not unique, we seek the unique x with minimum length. The solution is unique if and only if the $(m+p)\times n$ augmented matrix

$$\begin{bmatrix} B \\ A \end{bmatrix}$$

has rank n [586]. Algorithms for the problem generally adopt the paradigm of transforming it into an unconstrained LS problem from whose solution the desired x is constructed.

➤ **Exercise 19.2.12** Design an algorithm for the constrained LS problem by using the SVD.

➤ **Exercise 19.2.13 (Lawson–Hanson Algorithm)** It is known that any $A\in\boldsymbol{R}^{m\times n}$ can be decomposed as $A=Q^{\mathrm{T}}R$ such that $Q\in\boldsymbol{R}^{m\times m}$ is orthogonal and $R\in\boldsymbol{R}^{m\times n}$ is zero below the main diagonal. This is called the **QR decomposition**. Solve the LS problem with linear constraints by using the QR decomposition instead of the SVD.

19.2.5 Factor Analysis

Factor analysis postulates that the multitude of influences that affect our concern, which will be interest rate changes here, can be summarized by a few variables called **factors** [91, 390, 804]. Factors are said to explain changes in interest rates, and the quantitative relations between interest rate changes and factors are called **factor loadings**.

Fundamentals

At each time t, where $t=1,2,\ldots,T$, a p-dimensional vector y_t is observed, such as the interest rates of various maturities. The orthogonal factor model assumes that these observations are linearly related to m underlying, unobserved factors f_t by means of

$$\underset{p\times 1}{y_t-\mu} = \underset{p\times m}{L} \times \underset{m\times 1}{f_t} + \underset{p\times 1}{\epsilon_t}, \tag{19.15}$$

where $m < p$, $E[\, y_t \,] = \mu$, $E[\, f_t \,] = E[\, \epsilon_t \,] = \mathbf{0}$, $\mathrm{Cov}[\, f_t \,] = E[\, f_t f_t^{\mathrm{T}} \,] = I$, and

$$\Psi \equiv \mathrm{Cov}[\, \epsilon_t \,] = \mathrm{diag}[\, \Psi_1, \Psi_2, \ldots, \Psi_p \,] \in \boldsymbol{R}^{p \times p}.$$

Moreover, f_t and ϵ_t are independent. In Eq. (19.15) L contains the factor loadings, ϵ_t is the vector of **specific factors** or individual residual errors, and Ψ is the vector of **specific variances**. Each factor has zero mean and unit standard deviation and is uncorrelated with others. Note that L, f_t, and ϵ_t are unknown and the model is linear in the factors.

Equation (19.15) implies that

$$C \equiv E\,[\,(y_t - \mu)(y_t - \mu)^{\mathrm{T}}\,] = LL^{\mathrm{T}} + \Psi,$$

in which f_t and ϵ_t drop out. This makes it possible to compute L and Ψ as follows. Start with the estimated covariance matrix $C \in \boldsymbol{R}^{p \times p}$. By the Schur decomposition theorem, $C = B \times \mathrm{diag}[\, \lambda_1, \lambda_2, \ldots, \lambda_n \,] \times B^{\mathrm{T}}$ with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$. Let L be the first m columns of $B \times \mathrm{diag}[\, \sqrt{\lambda_1}, \sqrt{\lambda_2}, \ldots, \sqrt{\lambda_n} \,]$ and let Ψ be constructed by zeroing the off-diagonal elements of $C - LL^{\mathrm{T}}$. Of course, if some off-diagonal elements of the residual matrix $C - LL^{\mathrm{T}}$ are "large," there may be additional omitted factors, which calls for a larger m.[2]

The solution to L is not unique when $m > 1$: If A is an orthogonal matrix, then $\tilde{L} \equiv LA$ is also a solution as

$$\tilde{L}\tilde{L}^{\mathrm{T}} + \Psi = LAA^{\mathrm{T}}L^{\mathrm{T}} + \Psi = LL^{\mathrm{T}} + \Psi = C.$$

When L is replaced with $\tilde{L}$, the factors become $\tilde{f}_t \equiv A^{\mathrm{T}} f_t$ because

$$\tilde{L}\tilde{f}_t = LAA^{\mathrm{T}} f_t = Lf_t.$$

Orthogonal transformation of the factor loadings L, however, has no effects on Ψ and the diagonal elements of LL^{T}. This suggests we look for an A that gives the factor loadings $\tilde{L}$ intuitive economic interpretations. For instance, we might desire loadings that let rate changes at all maturities have approximately equal loading on the first factor, signifying a parallel shift. For this purpose, under the case of three factors ($m = 3$), each of the following orthogonal matrices called **Givens transformations** rotates through an angle θ, leaving one dimension unchanged:

$$A_1 \equiv \begin{bmatrix} \cos\theta_1 & \sin\theta_1 & 0 \\ -\sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad A_2 \equiv \begin{bmatrix} \cos\theta_2 & 0 & \sin\theta_2 \\ 0 & 1 & 0 \\ -\sin\theta_2 & 0 & \cos\theta_2 \end{bmatrix},$$

$$A_3 \equiv \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_3 & \sin\theta_3 \\ 0 & -\sin\theta_3 & \cos\theta_3 \end{bmatrix}.$$

Note that $A \equiv A_1 A_2 A_3$ is an orthogonal matrix with parameters θ_1, θ_2, and θ_3. Now find $\theta_1, \theta_2, \theta_3 \in [\, -\pi, \pi \,]$ that minimize the variance of the first column of $\tilde{L}$.

Not all positive definite matrices can be factored as $C = [\, C_{ij} \,] = LL^{\mathrm{T}} + \Psi$ for the given $m < p$. Even those that can be so factored may give statistically meaningless numbers such as a negative Ψ_i. Because $\mathrm{Cov}[\, y_t, f_t \,] = [\, L_{ij} \,]$ (see Exercise 19.2.15), $[\, L_{ij} C_{ij}^{-1/2} \,]_{i,j}$ is the correlation matrix of y_t and f_t, but there is no guarantee that $L_{ij} C_{ij}^{-1/2}$ will lie within the $[\, -1, 1 \,]$ range. These are some of the problems associated with factor analysis.

➤ **Exercise 19.2.14** Once the appropriate estimated factor loadings have been obtained, verify that the factors themselves can be estimated by $f_t = (L^{\mathrm{T}}\Psi^{-1}L)^{-1}L^{\mathrm{T}}\Psi^{-1}(y_t - \mu)$.

➤ **Exercise 19.2.15** Prove $\mathrm{Cov}[\, y_t,\, f_t\,] = L$.

Factors Affecting Interest Rate Movements

Because the U.S. Treasury spot rate curve varies more at shorter maturities than at longer maturities, more data are typically used for spot rates at shorter maturities. For instance, one may use the 11 rates at 3 months, 6 months, 1 year, 2 years, 3 years, 5 years, 7 years, 10 years, 15 years, 20 years, and 30 years (so $p = 11$). Research has shown that three factors can explain more than 90% of the variation in interest rate changes (so $m = 3$) [607]. These factors can be interpreted as **level**, **slope**, and **curvature**. The first factor has approximately equal effects on all maturities in that a change in it produces roughly parallel movements in interest rates of all maturities. The second factor affects the slope of the term structure but not the average level of interest rates. It produces movements in the long and the short ends of the term structure in opposite directions, twisting the curve, so to speak, with relatively smaller changes at intermediate maturities. For the third factor, the loadings are zero or negative at the shortest maturity, positive for intermediate maturities, and then they decline to become negative for the longest maturities. Thus a positive change in this factor tends to increase intermediate rates and decrease long rates or even short rates, altering the curvature of the term structure [91, 390, 804]. The factor analysis of interest rates is not without its problems, however [591].

➤ **Exercise 19.2.16** How would you define **slope duration** and **curvature duration**?

19.3 Curve Fitting with Splines

The purpose of curve fitting is to approximate the data with a "smooth" curve. Linear regression or its generalization, polynomial regression, is not ideal because it tries to fit a single polynomial over the entire data. The spline approach is different. It divides the domain interval into several subintervals and uses a different polynomial for each subinterval.

Suppose we want to fit a curve $f(x)$ over $n+1$ data points,

$$(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n),$$

with $x_0 < x_1 < \cdots < x_n$ such that the curve agrees with the data at each **breakpoint** (or **knot**) x_i. Divide $[\, x_0, x_n\,]$ into n subintervals: $[\, x_0, x_1\,], [\, x_1, x_2\,], \ldots, [\, x_{n-1}, x_n\,]$. If polynomials of degree zero are used for each subinterval, the curve is a step function. Its major disadvantage is the curve's discontinuity at the breakpoints. If polynomials of degree one or two are used instead, the discontinuity problem remains – it now applies, respectively, to the first and the second derivatives of the curve. The curves are therefore not "smooth" enough. If we require that the curve be a polynomial of degree three in each subinterval and that its first and second derivatives be continuous on $[\, x_0, x_n\,]$, then a (cubic) spline results. See Fig. 19.5 for illustration.

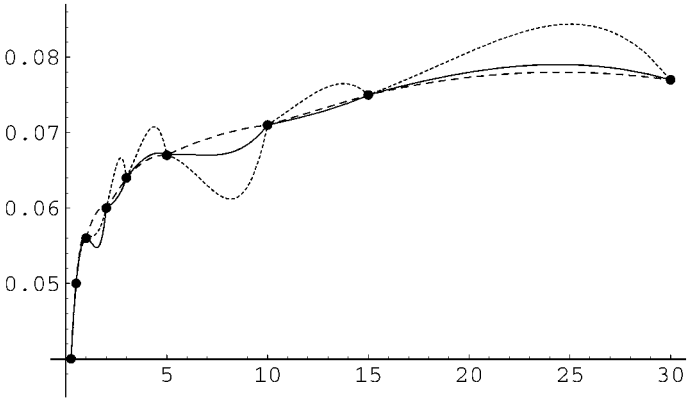

Figure 19.5: Splines of varying degrees. Plotted are splines of degrees two, three, and four. A higher degree leads to greater oscillation.

19.3.1 Cubic Splines

By considering the continuity of $f(x)$ and $f'(x)$, we find that the spline $f(x)$ over $x \in [\, x_{i-1}, x_i \,]$ is

$$f(x) = f'_{i-1} \frac{(x_i - x)^2 (x - x_{i-1})}{h_i^2} - f'_i \frac{(x - x_{i-1})^2 (x_i - x)}{h_i^2} + y_{i-1} \frac{(x_i - x)^2 [\, 2(x - x_{i-1}) + h_i \,]}{h_i^3} + y_i \frac{(x - x_{i-1})^2 [\, 2(x_i - x) + h_i \,]}{h_i^3},$$

where $h_i \equiv x_i - x_{i-1}$ with $n+1$ unknown values $f'_i \equiv f'(x_i)$, $i = 0, 1, \ldots, n$, to be determined. Now,

$$f''(x_i -) = \frac{2}{h_i} (f'_{i-1} + 2 f'_i) - 6 \frac{y_i - y_{i-1}}{h_i^2}, \tag{19.16}$$

$$f''(x_i +) = -\frac{2}{h_{i+1}} (2 f'_i + f'_{i+1}) + 6 \frac{y_{i+1} - y_i}{h_{i+1}^2}. \tag{19.16'}$$

If the second derivative of the spline is also continuous at each interior breakpoint, that is, $f''(x_i -) = f''(x_i +)$ for $i = 1, 2, \ldots, n-1$, then we have the following equations:

$$\frac{1}{h_i} f'_{i-1} + 2 \left(\frac{1}{h_i} + \frac{1}{h_{i+1}} \right) f'_i + \frac{1}{h_{i+1}} f'_{i+1} = 3 \frac{y_i - y_{i-1}}{h_i^2} + 3 \frac{y_{i+1} - y_i}{h_{i+1}^2}, \tag{19.17}$$

$i = 1, 2, \ldots, n-1$. See Fig. 19.3.1 for illustration. The equations form a diagonally dominant, tridiagonal system, which admits linear-time and stable numerical procedures.

We need $n + 1 - (n-1) = 2$ more auxiliary conditions relating $f'_0, f'_1, \ldots, f'_n$ in order to solve them. Typical conditions specify the values of (1) $f''(x_0)$ and $f''(x_n)$ or (2) $f'(x_0)$ and $f'(x_n)$. For example, letting $f''(x_0) = f''(x_n) = 0$ leads to the so-called **natural splines**. With Eq. (19.16), these two conditions eliminate variables f'_0 and f'_n from Eq. (19.17) because f'_0 can be replaced with a linear function of f'_1 and f'_n can be replaced with a linear function of f'_{n-1}. Choice (2) forces the slope at each

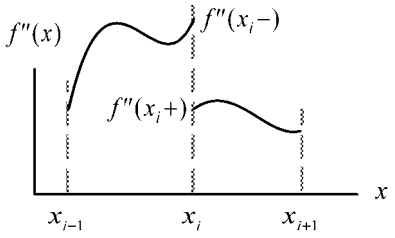

Figure 19.6: Construction of splines.

end to assume specified values, thus immediately eliminating variables f_0' and f_n'. Other alternatives exist [381]. The natural-spline approximation is the "smoothest" among all interpolating functions [391].

Splines with **uniform spacing** are splines with equally spaced breakpoints, $h_i \equiv x_i - x_{i-1} = h$. We can now simplify Eq. (19.17) to

$$f'_{i-1} + 4f'_i + f'_{i+1} = 3\,\frac{y_{i+1} - y_{i-1}}{h}, \quad i = 1, \ldots, n-1.$$

➤ **Exercise 19.3.1** Verify Eq. (19.16).

➤ **Exercise 19.3.2** Write the tridiagonal system for the natural spline.

An Alternative Formulation

The continuity of $f(x)$ and $f''(x)$ results in

$$f(x) = y_{i-1}\,\frac{x_i - x}{h_i} + y_i\,\frac{x - x_{i-1}}{h_i} - \frac{h_i^2}{6}\,f''_{i-1}\left[\frac{x_i - x}{h_i} - \left(\frac{x_i - x}{h_i}\right)^3\right]$$
$$- \frac{h_i^2}{6}\,f''_i\left[\frac{x - x_{i-1}}{h_i} - \left(\frac{x - x_{i-1}}{h_i}\right)^3\right], \quad x \in [\,x_{i-1}, x_i\,],$$

with $n+1$ values $f_i'' \equiv f''(x_i)$ to be determined. Imposing continuity of $f'(x)$ at each interior breakpoint results in

$$h_i\,f''_{i-1} + 2(h_i + h_{i+1})\,f''_i + h_{i+1}\,f''_{i+1} = 6\left(\frac{y_{i+1} - y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i}\right), \tag{19.18}$$

$i = 1, 2, \ldots, n-1$. The preceding linear equations form a diagonally dominant, tridiagonal system. As in Eq. (19.17), two more conditions are needed to solve it. This formulation is convenient for natural splines because, in this case, $f_0'' = f_n'' = 0$ [417].

EXAMPLE 19.3.1 For the data set $\{(0,0), (1,2), (2,1), (3,5), (4,4)\}$, we have $n = 4$, $h_1 = h_2 = h_3 = h_4 = 1$, $x_i = i$ $(1 \le i \le 4)$, $y_0 = 0$, $y_1 = 2$, $y_2 = 1$, $y_3 = 5$, $y_4 = 4$. The spline is

$$p_i(x) \equiv y_{i-1}\,(x_i - x) + y_i(x - x_{i-1})$$
$$-\frac{1}{6}\,f''_{i-1}[\,(x_i - x) - (x_i - x)^3\,] - \frac{1}{6}\,f''_i[\,(x - x_{i-1}) - (x - x_{i-1})^3\,]$$

for $x \in [\,x_{i-1}, x_i\,]$. To fit the data with a natural spline, we solve

$$\begin{bmatrix} 4 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix} \begin{bmatrix} f_1'' \\ f_2'' \\ f_3'' \end{bmatrix} = \begin{bmatrix} -18 \\ 30 \\ -30 \end{bmatrix}.$$

The solutions are $f_1'' = -7.5$, $f_2'' = 12$, and $f_3'' = -10.5$. We finally obtain the spline by adding $f_0'' = f_4'' = 0$. Its four cubic polynomials are

$$p_1(x) = 2x + \frac{7.5}{6}(x - x^3),$$

$$p_2(x) = 2(2-x) + (x-1) + \frac{7.5}{6}\left[(2-x) - (2-x)^3\right] - \frac{12}{6}\left[(x-1) - (x-1)^3\right],$$

$$p_3(x) = (3-x) + 5(x-2) - \frac{12}{6}\left[(3-x) - (3-x)^3\right] + \frac{10.5}{6}\left[(x-2) - (x-2)^3\right],$$

$$p_4(x) = 5(4-x) + 4(x-3) + \frac{10.5}{6}\left[(4-x) - (4-x)^3\right].$$

See the solid curve in Fig. 19.7. If y_2 is perturbed slightly to 1.3, the whole spline changes, to the dotted curve in Fig. 19.7.

➤ **Exercise 19.3.3** Verify Eq. (19.18).

➤ **Exercise 19.3.4** Construct the spline in Example 19.3.1 when $y_2 = 1.3$.

19.3.2 Cubic Splines and the Constrained Least-Squares Problem

In applications that have more data points than breakpoints, we find a spline that minimizes the distance to the data. Let the cubic polynomial for the interval $[x_{i-1}, x_i]$ be written as

$$p_i(x) \equiv a_i + b_i x + c_i x^2 + d_i x^3, \quad i = 1, 2, \ldots, n.$$

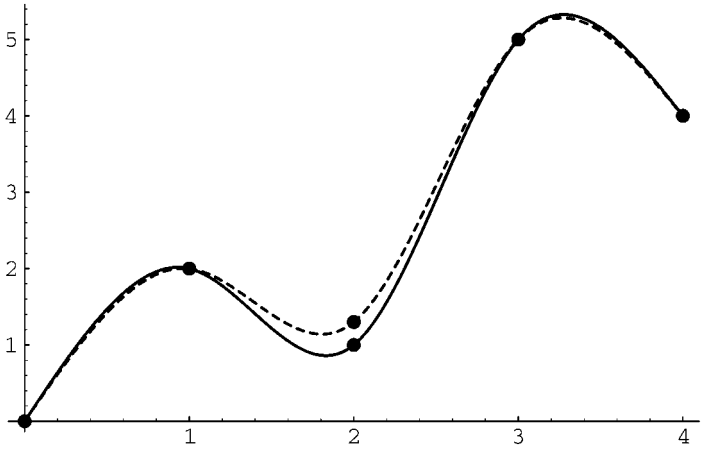

Figure 19.7: Curve fitting with cubic splines. The whole spline is changed, to the dotted curve, if just a data point is perturbed. See Example 19.3.1 for the details.

Because cubic splines are by definition twice continuously differentiable,

$$p_i(x_i) = p_{i+1}(x_i), \quad p_i'(x_i) = p_{i+1}'(x_i), \quad p_i''(x_i) = p_{i+1}''(x_i), \quad i = 1, 2, \ldots, n-1,$$

where $x_0, x_1, \ldots, x_n$ are the breakpoints. The preceding equations imply that

$$\begin{aligned} a_i + b_i x_i + c_i x_i^2 + d_i x_i^3 &= a_{i+1} + b_{i+1} x_i + c_{i+1} x_i^2 + d_{i+1} x_i^3, \\ b_i + 2c_i x_i + 3d_i x_i^2 &= b_{i+1} + 2c_{i+1} x_i + 3d_{i+1} x_i^2, \\ 2c_i + 6d_i x_i &= 2c_{i+1} + 6d_{i+1} x_i. \end{aligned}$$

Let $x \equiv [\, a_1, b_1, c_1, d_1, \ldots, a_n, b_n, c_n, d_n \,]^{\mathrm{T}} \in \boldsymbol{R}^{4n}$. Then the preceding equations can be written as $Bx = d$ for some $B \in \boldsymbol{R}^{3(n-1)\times 4n}$ and $d \in \boldsymbol{R}^{3(n-1)}$. Assume that $3(n-1)$ is the rank of B. Consider the data set $\{(\tilde{x}_1, \tilde{y}_1), (\tilde{x}_2, \tilde{y}_2), \ldots, (\tilde{x}_m, \tilde{y}_m)\}$. Suppose $\tilde{x}_j$ falls within $[\, x_{i-1}, x_i \,]$. Then the LS formulation is

$$p_i(\tilde{x}_j) \equiv a_i + b_i \tilde{x}_j + c_i \tilde{x}_j^2 + d_i \tilde{x}_j^3 = \tilde{y}_j.$$

These m equations can be put into the form $Ax = b$ for some $A \in \boldsymbol{R}^{m\times 4n}$ and $b \in \boldsymbol{R}^m$. Thus we have the constrained LS problem $\min_{x\in R^{4n},\ Bx=d} \| Ax - b \|$. Note that A is banded (see Comment 19.2.2).

➤ **Exercise 19.3.5** Write the matrices for A, B, b, and d.

19.3.3 B-Splines and the Least-Squares Problem

A **B-spline** (**basic spline function**) B_i is a fixed cubic spline determined by five breakpoints $x_i, x_{i+1}, \ldots, x_{i+4}$. It has zero value outside of $[\, x_i, x_{i+4} \,]$. Every cubic spline with breakpoints $x_0, x_1, \ldots, x_n$ can be represented as

$$s(t) \equiv \sum_{i=-3}^{n-1} \alpha_i B_i(x), \quad x_0 \le x \le x_n$$

for unique α_is. (See [35] for the form of $B_i(x)$ and [391] for the simpler case of equally spaced breakpoints.) Let $\{(\tilde{x}_1, \tilde{y}_1), (\tilde{x}_2, \tilde{y}_2), \ldots, (\tilde{x}_m, \tilde{y}_m)\}$ be the data. We seek $\alpha \equiv [\, \alpha_{-3}, \alpha_{-2}, \ldots, \alpha_{n-1} \,]^{\mathrm{T}}$ that minimizes

$$\sum_{j=1}^{m} (s(\tilde{x}_j) - \tilde{y}_j)^2 = \| A\alpha - \tilde{y} \|^2 \tag{19.19}$$

for some matrix $A \in \boldsymbol{R}^{m\times(n+3)}$, where $\tilde{y} \equiv [\, \tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_m \,]^{\mathrm{T}}$. Because the only B-splines with nonzero values for $x \in [\, x_{i-1}, x_i \,]$ are $B_{i-3}, B_{i-2}, B_{i-1}$, and B_i, matrix A is banded with a bandwidth of four. The B-spline approach, in contrast to the cubic-spline approach to the same problem, does not lead to a constrained LS problem.

➤ **Exercise 19.3.6** Write the matrix A in Eq. (19.19).

Additional Reading

The literature on matrix computation is vast [391, 392, 417, 447, 465, 701, 830, 870]. See [825] for the history of the SVD and [392, 465] for applications. Comprehensive treatments of the LS problem can be found in [75, 586], and statistical properties of the LS estimator are covered in [422]. Reference [523] has a good coverage of factor

analysis. Consult [417] for smoothing techniques. Splines in their present form are due to Schoenberg (1903–1990) in the 1940s [35, 447, 863].

NOTES

1. This term has become firmly entrenched even though "orthonormal" might be more consistent.
2. An alternative is to assume that the residuals have a multivariate normal distribution and then estimate the model parameters by using maximum likelihood [523, 632].