Chapter 6

Data modeling by least squares

The reconciliation of theory and data is the essence of science. An ubiquitous tool in this task is the method of least-squares fitting. Elementary calculus books generally consider the fitting of a straight line to scattered data points. Such an elementary application gives scant hint of the variety of practical problems which can be solved by the method of least squares. Some geophysical examples which we will consider include locating earthquakes, analyzing tides, expanding the earth's gravity and magnetic fields in spherical harmonics, and doing interesting things with time series. When the past of a time series is available, one may find that least squares can be used to determine a filter which predicts some future values of the time series. When a time series which has been highly predictable for a long stretch of time suddenly becomes much less predictable an "event" is said to have occurred. A filter which emphasizes such events is called a *prediction-error filter.* If one is searching for a particular dispersed wavelet in a time series, it may help to design a filter which compresses the wavelet into some more recognizable shape, an impulse for example. Such a wave-shaping filter may be designed by least squares. With multiple time series which arise from several sensors detecting waves in space, least squares may be used to find filters which respond only to certain directions and wave speeds.

Before we begin with the general theory, let us take up a simple example in the subject of time series analysis. Given the input, say $\mathbf{x} = (2, 1)$ to some filter, say $\mathbf{f} = (f_0, f_1)$ then the output is necessarily $\mathbf{c} = (2f_0,\ f_0,\ +\ 2f_1,\ f_1)$. To design an inverse filter we would wish to have $\mathbf{c}$ come out as close as possible to (1, 0, 0). In order to minimize the difference between the actual and the desired outputs we minimize

$$E\,(f_0, f_1) \quad = \quad (2f_0 - 1)^2 + (f_0 + 2f_1)^2 + (f_1)^2$$

The sum E of the squared errors will attain a minimum if f_0 and f_1 are chosen so that

$$\begin{aligned} 0 \ &= \ \frac{\partial E}{\partial f_0} \ &= \ 2\,(2f_0 - 1)\,2 + 2\,(f_0 + 2f_1) \\ 0 \ &= \ \frac{\partial E}{\partial f_1} \ &= \ 2\,(f_0 + 2f_1)\,2 + 2f_1 \end{aligned}$$

Canceling a 2 and arranging this into the standard form for simultaneous equations, we get

$$\begin{bmatrix} 5 & 2 \\ 2 & 5 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

and the solution is

$$\begin{bmatrix} f_0 \\ f_1 \end{bmatrix} = \frac{1}{21} \begin{bmatrix} 5 & -2 \\ -2 & 5 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{10}{21} \\ -\frac{4}{21} \end{bmatrix}$$

The actual $\mathbf{c}$ which comes out of this filter is $(\frac{20}{21}, +\frac{2}{21}, -\frac{4}{21})$ which is not a bad approximation to $(1, 0, 0)$.

6.1 MORE EQUATIONS THAN UNKNOWNS

When there are more linear equations than unknowns, it is usually impossible to find a solution which satisfies all the equations. Then one often looks for a solution which approximately satisfies all the equations. Let $\mathbf{a}$ and $\mathbf{c}$ be known and $\mathbf{x}$ be unknown in the following set of equations where there are more equations than unknowns.

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & & \vdots \\ a_{31} & & & \\ \vdots & & & \\ a_{n1} & & \cdots & a_{nm} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \approx \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \tag{6.1}$$

Usually there will be no set of x_i which exactly satisfies (6.1). Let us define an error vector e_j by

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & & \\ a_{31} & & & \\ \vdots & & & \\ a_{n1} & & \cdots & a_{nm} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} - \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix} \tag{6.2}$$

It simplifies the development to rewrite this equation as follows (a trick I learned from John P. Berg).

$$\begin{bmatrix} -c_1 & a_{11} & \cdots & a_{1m} \\ -c_2 & a_{21} & & \\ -c_3 & & & \\ \vdots & & & \\ -c_n & a_{n1} & \cdots & a_{nm} \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_m \end{bmatrix} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix} \tag{6.3}$$

We may abbreviate this equation as

$$\mathbf{Bx} \quad = \quad \mathbf{e} \tag{6.4}$$

where $\mathbf{B}$ is the matrix containing $\mathbf{c}$ and $\mathbf{a}$. The ith error may be written as a dot product and either vector may be written as the column

$$e_i \quad = \quad [b_{i1} \quad b_{i2} \quad \cdots] \begin{bmatrix} 1 \\ x_1 \\ \vdots \end{bmatrix} \quad = \quad [1 \quad x_1 \quad \cdots] \begin{bmatrix} b_{i1} \\ b_{i2} \\ \vdots \end{bmatrix}$$

Now we will minimize the sum squared error E defined as $\sum {e_i}^2$

$$E \quad = \quad \sum_i [1 \quad x_1 \quad \cdots] \begin{bmatrix} b_{i1} \\ b_{i2} \\ \vdots \end{bmatrix} [b_{i1} \quad b_{i2} \quad \cdots] \begin{bmatrix} 1 \\ x_1 \\ \vdots \end{bmatrix} \tag{6.5}$$

The summation may be brought inside the constants

$$E \quad = \quad [1 \quad x_1 \quad x_2 \quad \cdots] \left\{ \sum_{i=1}^{n} \begin{bmatrix} -c_i \\ a_{i1} \\ a_{i2} \\ \vdots \end{bmatrix} [-c_i \quad a_{i1} \quad a_{i2} \quad \cdots] \right\} \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} \tag{6.6}$$

The matrix in the center, call it r_{ij}, is symmetrical. It is a positive (more strictly, nonnegative) definite matrix because you will never be able to find an $\mathbf{x}$ for which E is negative, since E is a sum of squared e_i. We find the $\mathbf{x}$ with minimum E by requiring $\partial E/\partial x_1 = 0$, $\partial E/\partial x_2 = 0, \ldots, \partial E/\partial x_m = 0$. Notice that this will give us exactly one equation for each unknown. In order to clarify the presentation we will specialize (6.6) to two unknowns.

$$E \quad = \quad [1 \quad x_1 \quad x_2] \begin{bmatrix} r_{00} & r_{01} & r_{02} \\ r_{10} & r_{11} & r_{12} \\ r_{20} & r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix} \tag{6.7}$$

Setting to zero the derivative with respect to x_1, we get

$$0 \quad = \quad \frac{\partial E}{\partial x_1} \quad = \quad [0 \quad 1 \quad 0]\, R \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix} + [1 \quad x_1 \quad x_2] R \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \tag{6.8}$$

Since $r_{ij} = r_{ji}$, both terms on the right are equal. Thus (6.8) may be written

$$0 \quad = \quad \frac{\partial E}{\partial x_1} \quad = \quad 2[r_{10} \quad r_{11} \quad r_{12}] \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix} \tag{6.9}$$

Likewise, differentiating with respect to x_2 gives

$$0 \quad = \quad \frac{\partial E}{\partial x_2} \quad = \quad 2[r_{20} \quad r_{21} \quad r_{22}] \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix} \tag{6.10}$$

Equations (6.9) and (6.10) may be combined

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad = \quad \begin{bmatrix} r_{10} & r_{11} & r_{12} \\ r_{20} & r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix} \tag{6.11}$$

This form is two equations in two unknowns. One might write it in the more conventional form

$$\begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad = \quad - \begin{bmatrix} r_{10} \\ r_{20} \end{bmatrix} \tag{6.12}$$

The matrix of (6.11) lacks only a top row to be equal to the matrix of (6.7). To give it that row, we may augment (6.11) by

$$v \quad = \quad r_{00} + r_{01}x_1 + r_{02}x_2 \tag{6.13}$$

where (6.13) may be regarded as a definition of a new variable v. Putting (6.13) on top of (6.11) we get

$$\begin{bmatrix} v \\ 0 \\ 0 \end{bmatrix} \quad = \quad \begin{bmatrix} r_{00} & r_{01} & r_{02} \\ r_{10} & r_{11} & r_{12} \\ r_{20} & r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix} \tag{6.14}$$

The solution $\mathbf{x}$ of (6.12) or (6.14) is that set of x_k for which E is a minimum. To get an interpretation of v, we may multiply both sides by $[1 \quad x_1 \quad x_2]$, getting

$$v \quad = \quad [1 \quad x_1 \quad x_2] \begin{bmatrix} v \\ 0 \\ 0 \end{bmatrix} \quad = \quad [1 \quad x_1 \quad x_2] R \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix} \tag{6.15}$$

Comparing (6.15) with (6.7), we see that v is the minimum value of E.

Occasionally, it is more convenient to have the essential equations in partitioned matrix form. In partitioned matrix form, we have for the error (6.6)

$$E \quad = \quad [1 \vdots \mathbf{x}]^T \begin{bmatrix} -\mathbf{c}^T \\ \cdots\cdots \\ \mathbf{A}^T \end{bmatrix} \left[\begin{array}{c|c} -\mathbf{c} & \mathbf{A} \end{array} \right] \begin{bmatrix} 1 \\ \cdots \\ \mathbf{x} \end{bmatrix} \tag{6.16}$$

The final equation (6.14) splits into

$$V \quad = \quad \mathbf{c}^T\mathbf{c} - \mathbf{c}^T\mathbf{A}\mathbf{x} \tag{6.17}$$

$$\mathbf{0} \quad = \quad -\mathbf{A}^T\mathbf{c} + \mathbf{A}^T\mathbf{A}\mathbf{x} \tag{6.18}$$

where (6.18) represents simultaneous equations to be solved for $\mathbf{x}$. Equation (6.18) is what you have to set up in a computer. It is easily remembered by a quick and dirty (very dirty) derivation. That is, we began with the overdetermined equations $\mathbf{Ax} \approx \mathbf{c}$; premultiplying by $\mathbf{A}^T$ gives $(\mathbf{A}^T\mathbf{A})\mathbf{x} = \mathbf{A}^T\mathbf{c}$ which is (6.18).

In physical science applications, the variable z_j is frequently a complex variable, say $z_j = x_j + iy_j$. It is always possible to go through the foregoing analyses, treating the problem as though x_i and y_i were real independent variables. There is a considerable gain in simplicity and a saving in computational effort by treating z_j as a single complex variable. The error E may be regarded as a function of either x_j and y_j or z_j and $\overline{z}_j$. In general $j = 1, 2, \ldots, N$, but we will treat the case $N = 1$ here and leave the general case for the Exercises. The minimum is found where

$$0 = \frac{dE}{dx} = \frac{\partial E}{\partial z}\frac{dz}{dx} + \frac{\partial E}{\partial \overline{z}}\frac{d\overline{z}}{dx} = \frac{\partial E}{\partial z} + \frac{\partial E}{\partial \overline{z}} \tag{6.19}$$

$$0 = \frac{dE}{dy} = \frac{\partial E}{\partial z}\frac{dz}{dy} + \frac{\partial E}{\partial \overline{z}}\frac{d\overline{z}}{dy} = i\left(\frac{\partial E}{\partial z} - \frac{\partial E}{\partial \overline{z}}\right) \tag{6.20}$$

Multiplying (6.20) by i and adding and subtracting these equations, we may express the minimum condition more simply as

$$0 = \frac{\partial E}{\partial z} \tag{6.21}$$

$$0 = \frac{\partial E}{\partial \overline{z}} \tag{6.22}$$

However, the usual case is that E is a positive real quadratic function of z and $\overline{z}$ and that $\partial E/\partial z$ is merely the complex conjugate of $\partial E/\partial \overline{z}$. Then the two conditions (6.21) and (6.22) may be replaced by either one of them. Usually, when working with complex variables we are minimizing a positive quadratic form like

$$E(z^*, z) = |\mathbf{Az} - \mathbf{c}|^2 = (\mathbf{z}^*\mathbf{A}^* - \mathbf{c}^*)(\mathbf{Az} - \mathbf{c}) \tag{6.23}$$

where * denotes complex-conjugate transpose. Now (6.22) gives

$$0 = \frac{\partial E}{\partial z^*} = \mathbf{A}^*(\mathbf{Az} - \mathbf{c}) \tag{6.24}$$

which is just the complex form of (6.18).

Let us consider an example. Suppose a set of wave arrival times t_i is measured at sensors located on the x axis at points x_i. Suppose the wavefront is to be fitted to a parabola $t_i \approx a + bx_i + c{x_i}^2$. Here, the x_i are knowns and a, b, and c are unknowns. For each sensor i we have an equation

$$\begin{bmatrix} -t_i & 1 & x_i & {x_i}^2 \end{bmatrix} \begin{bmatrix} 1 \\ a \\ b \\ c \end{bmatrix} \approx 0 \tag{6.25}$$

When i has greater range than 3 we have more equations than unknowns. In this example, (6.14) takes the form

$$\left\{ \sum_{i=1}^{n} \begin{bmatrix} -t_i \\ 1 \\ x_i \\ {x_i}^2 \end{bmatrix} \begin{bmatrix} -t_i & 1 & x_i & {x_i}^2 \end{bmatrix} \right\} \begin{bmatrix} 1 \\ a \\ b \\ c \end{bmatrix} \quad = \quad \begin{bmatrix} v \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{6.26}$$

This may be solved by standard methods for a, b, and c.

The last three rows of (6.26) may be written

$$\sum_{i=1}^{n} \begin{bmatrix} 1 \\ x_i \\ {x_i}^2 \end{bmatrix} e_i \quad = \quad \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{6.27}$$

This says the error vector e_i is perpendicular (or normal) to the functions 1, x, and x^2, which we are fitting to the data. For that reason these equations are often called normal equations.

EXERCISES:

1 Extend (6.24) by fitting waves observed in the x, y plane to a two-dimensional quadratic.

2 Let $y(t)$ constitute a complex-valued function at successive integer values of t. Fit $y(t)$ to a least-squares straight line $y(t) \approx \alpha + \beta t$ where $\alpha = \alpha_r + i\alpha_t$ and $\beta = \beta_r + i\beta_t$. Do it two ways: (a) Assume α_r, α_t, β_i, and β_r are four independent variables, and (b) Assume α, $\bar{\alpha}$, β, and $\bar{\beta}$ are independent variables. (Leave answer in terms of $s_n = \sum_t t^n$.)

3 Equation (6.14) has assumed all quantities are real. Generalize equation (6.14) to all complex quantities. Verify that the matrix is Hermitian.

4 At the jth seismic observatory (latitude x_j, longitude y_j) earthquake waves are observed to arrive at time t_j. It has been conjectured that the earthquake has an origin time t, latitude x, and longitude y. The theoretical travel time may be looked up in a travel time table $T(\Delta)$ where T is the travel time and Δ is the great circle angle. One has

$$\cos\Delta \quad = \quad \sin y \sin y_i + \cos y \cos y_i \cos(x - x_i)$$

The time residual at the jth station, supposing that the earthquake occurred at (x, y, t), is

$$e_j \quad = \quad t + T(\Delta_j) - t_j$$

The time residual, supposing that the earthquake occurred at $(x+dx, y+dy, t+dt)$, is

$$e_j \quad = \quad t + dt + T(\Delta_j) + \left(\frac{\partial T}{\partial \Delta} \frac{\partial \Delta}{\partial x} \right)_j dx + \left(\frac{\partial T}{\partial \Delta} \frac{\partial \Delta}{\partial y} \right)_j dy - t_j$$

Find equations to solve for dx, dy, and dt which minimize the sum-squared time residuals.

5 Gravity g_j has been measured at N irregularly spaced points on the surface of the earth (colatitude x_j, longitude y_j, $j = 1, N$). Show that the matrix of the normal equation which fits the data to spherical harmonics may be written as a sum of a column times its transpose, as in the preceding problem. How would the matrix simplify if there were infinitely many uniformly spaced data points? (NOTE: Spherical harmonics S are the class of functions

$$S_n^m(x, y) + P_n^m(\cos x)\exp(imy)$$

for $(m = -n, \ldots, -1, 0, 1, \ldots, n)$ and $(n = 0, 1, \ldots, \infty)$ where P_n^m is an associated Legendre polynomial of degree n and order m.)

6 Ocean tides fit sinusoidal functions of known frequencies quite accurately. Associated with the tide is an earth tilt. A complex time series may be made from the north-south tilt plus $\sqrt{-1}$ times the east-west tilt. The observed complex time series may be fitted to an analytical form $\sum_{j=1}^{N} A_j e^{i\omega} j^t$. Find a set of equations which may be solved for the A_j which gives the best fit of the formula to the data. Show that some elements of the normal equation matrix are sums which may be summed analytically.

7 The general solution to Laplace's equation in cylindrical coordinates (r, θ) for a potential field P which vanishes at $r = \infty$ is given by

$$P(r, \theta) \quad = \quad \mathrm{Re} \sum_{m=0}^{\infty} A_m \frac{e^{im\theta}}{r^{m+1}}$$

Find the potential field surrounding a square object at the origin which is at unit potential. Do this by finding N of the coefficients A_m by minimizing the squared difference between $P(r, \theta)$ and unity integrated around the square. Give the answer in terms of an inverse matrix of integrals. Which coefficients A_m vanish exactly by symmetry?

6.2 WEIGHTS AND CONSTRAINTS

It often happens that some observations are considered more reliable than others. One may desire to weight the more reliable data more heavily in the calculation. In other words, we may multiply the ith equation by a weight $\sqrt{w_i}$

$$\sqrt{w_i}\,[-c_i \quad a_{i1} \quad a_{i2} \quad \cdots] \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} \quad = \quad \sqrt{w_i} e_i \tag{6.28}$$

Now the weighted sum-squared error will be

$$E = \sum_i w_i {e_i}^2 \tag{6.29}$$

Following the method of the last section, it is easy to show that the $\mathbf{x}$ which minimizes the weighted error E of (6.29) is the $\mathbf{x}$ which satisfies the simultaneous equations

$$\left\{ \sum_i w_i \begin{bmatrix} -c_i \\ a_{i1} \\ a_{i2} \\ \vdots \end{bmatrix} \begin{bmatrix} -c_i & a_{i1} & a_{i2} & \cdots \end{bmatrix} \right\} \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} v \\ 0 \\ 0 \\ \vdots \end{bmatrix} \tag{6.30}$$

Choice of a set of weights is often a rather subjective matter. However, if data are of uneven quality, it cannot be avoided. Omitting w is equivalent to choosing it equal to unity.

A case of common interest is where some equations should be solved exactly. Such equations are called constraint equations. Constraint equations often arise out of theoretical considerations so they may, in principle, not have any error. The rest of the equations often involve some measurement. Since the measurement can often be made many times, it is easy to get a lot more equations than unknowns. Since measurement always involves error, we then use the method of least squares to minimize the average error. In order to be certain that the constraint equations are solved exactly, one could use the trick of applying very large weight factors to the constraint equations. A problem is that "very large" is not well defined. A weight equal 10^{10} might not be large enough to guarantee the constraint equation is satisfied with sufficient accuracy. On the other hand, 10^{10} might lead to disastrous round-off when solving the simultaneous equations in a computer with eight-digit accuracy. The best approach is to analyze the situation theoretically for $w \to \infty$.

An example of a constraint equation is that the sum of the x_i equals M. Another constraint would be $x_1 = x_2$. Arranged in a matrix, these two constraint equations are

$$\begin{bmatrix} -M & 1 & 1 & 1 & 1 & \cdots \\ 0 & 1 & -1 & 0 & 0 & \cdots \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{6.31}$$

We write a general set of k constraint equations as

$$\mathbf{G} \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \end{bmatrix} \tag{6.32}$$

Minimizing the error as $w \to \infty$ of the equations

$$\begin{aligned} \sqrt{w}\mathbf{G}\mathbf{x} &\approx \mathbf{0} \\ \mathbf{B}\mathbf{x} &\approx \mathbf{0} \end{aligned}$$

is algebraically similar to minimizing the error of $\mathbf{Bx} \approx \mathbf{0}$. The rows of $\sqrt{w}\mathbf{G}$ are just like some extra rows for $\mathbf{B}$. The resulting equation for $\mathbf{x}$ is

$$\left\{ \sum_{i=1}^{n} \begin{bmatrix} -c_i \\ a_{i1} \\ \vdots \end{bmatrix} \begin{bmatrix} -c_i & a_{i1} & \cdots \end{bmatrix} + \sum_{i=1}^{k} w_i \begin{bmatrix} g_{i0} \\ g_{i1} \\ \vdots \end{bmatrix} \begin{bmatrix} g_{i0} & g_{i1} & \cdots \end{bmatrix} \right\} \begin{bmatrix} 1 \\ x_i \\ \vdots \end{bmatrix} = \begin{bmatrix} v \\ 0 \\ \vdots \end{bmatrix} \tag{6.33}$$

Now we will take all the w_i to equal $1/\varepsilon$ and we will let ε tend to zero. Also let

$$\mathbf{x} = \mathbf{x}^{(0)} + \varepsilon \mathbf{x}^{(1)} + \varepsilon^2 \mathbf{x}^{(2)} + \cdots \tag{6.34}$$

$$\mathbf{v} = \mathbf{v}^{(0)} + \varepsilon \mathbf{v}^{(1)} + \varepsilon^2 \mathbf{v}^{(2)} + \cdots \tag{6.35}$$

With this, (6.33) may be written

$$\left(\mathbf{B}^T\mathbf{B} + \frac{1}{\varepsilon}\mathbf{G}^T\mathbf{G} \right) (\mathbf{x}^{(0)} + \varepsilon \mathbf{x}^{(1)} + \cdots) = \mathbf{v}^{(0)} + \mathbf{v}^{(1)}\varepsilon + \cdots \tag{6.36}$$

Identify coefficients of powers of ε

$$\varepsilon^{-1} : \mathbf{G}^T\mathbf{G}\mathbf{x}^{(0)} = \mathbf{0} \tag{6.37}$$

$$\varepsilon^{0} : \mathbf{B}^T\mathbf{B}\mathbf{x}^{(0)} + \mathbf{G}^T\mathbf{G}\mathbf{x}^{(1)} = \mathbf{v}^{(0)} \tag{6.38}$$

$$\varepsilon^1, \varepsilon^2 : \text{not required}$$

Equation (6.37) is m equations in m unknowns. It will automatically be satisfied if the k equations in (6.32) are satisfied. Equation (6.38) appears to involve the m unknowns in $\mathbf{x}^{(0)}$ plus m more unknowns in $\mathbf{x}^{(1)}$. In fact, we do not need $\mathbf{x}^{(1)}$; the k unknowns

$$\lambda = \mathbf{G}\mathbf{x}^{(1)} \tag{6.39}$$

will suffice.

Arranging (6.38) and (6.32) together and dropping superscripts, we get a square matrix in $m + k$ unknowns.

$$\left[\begin{array}{c|c} \mathbf{B}^T\mathbf{B} & \mathbf{G}^T \\ \hline \mathbf{G} & \mathbf{0} \end{array} \right] \left[\begin{array}{c} \mathbf{x} \\ \hline \lambda \end{array} \right] = \begin{bmatrix} v \\ 0 \\ \vdots \end{bmatrix} \tag{6.40}$$

Equation (6.40) is now a simultaneous set for the unknowns $\mathbf{x}$ and λ. It might also be thought of as the solution to the problem of minimizing the quadratic form

$$\begin{aligned} E &= \begin{bmatrix} \mathbf{x}^T & \lambda^T \end{bmatrix} \begin{bmatrix} \mathbf{B}^T\mathbf{B} & \mathbf{G}^T \\ \mathbf{G} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \lambda \end{bmatrix} \\ &= \mathbf{x}^T\mathbf{B}^T\mathbf{B}\mathbf{x} + \lambda^T\mathbf{G}\mathbf{x} + \mathbf{x}^T\mathbf{G}^T\lambda \end{aligned}$$

and since we can always transpose a scalar,

$$E = \mathbf{x}^T\mathbf{B}^T\mathbf{B}\mathbf{x} + 2\lambda^T\mathbf{G}\mathbf{x} \tag{6.41}$$

According to the method of Lagrange multipliers, one may minimize a quadratic form subject to constraints by minimizing instead a sum of the quadratic form plus constraint terms where each constraint term is the product of a constraint equation multiplied by a Lagrange multiplier λ_i. This is precisely what we have in (6.41), and the solution is given by (6.40). Lagrange multipliers frequently arise in connection with integral equations. The concept is readily transformed to matrices merely by approximating integration by summation.

EXERCISES:

1 In determining a density vs. depth profile of the earth one might minimize the squared difference between some theoretical quantities (say, the frequencies of free oscillation) and the observed quantities. By astronomical means, total mass and moment of inertia of the earth are very well known. If the earth is divided into arbitrarily thin shells of equal thickness, what are the two astronomical constraint equations on the layer densities p_i? If the least-squares problem is nonlinear (as it often is) it may be linearized by assuming that a given set of densities p_i is a good guess which satisfies the constraints and doing least squares for the perturbation dp_i. What are the constraint equations on dp_i?

6.3 FEWER EQUATIONS THAN UNKNOWNS

What is one to do when one has fewer equations than unknowns: give up? Certainly not, just apply the principle of simplicity. Let us find the simplest solution which satisfies all the equations. This situation often arises. Suppose, after having made a finite number of measurements one is trying to determine a continuous function, for example, the mass density $\rho(r)$ as a function of depth in the earth. Then, in a computer $\rho(r)$ would be represented by $\rho(r)$ sampled at N depths $r_i, i = 1, 2, \ldots, N$. Then merely taking N large, one has more unknowns than equations.

One measure of simplicity is that the unknown function x_i has minimum wiggliness. In other words minimize

$$E \quad = \quad \sum (x_i - x_{i-1})^2 \tag{6.42}$$

subject to satisfying exactly the observation or constraint equations

$$\mathbf{G}\mathbf{x} \quad = \quad \mathbf{0} \tag{6.43}$$

Another more popular measure of simplicity (which does not imply an ordering of the variables x_i) is the minimization of

$$E \quad = \quad \sum_{i=1}^{n} {x_i}^2 \tag{6.44}$$

If we set out to minimize (6.44) without any constraints, x would satisfy the simultaneous equations

$$\begin{bmatrix} 1 & & & \text{zeros} \\ & 1 & & \\ & & 1 & \\ \text{zeros} & & & \ddots \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} v \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{6.45}$$

By inspection one sees the obvious result that $x_i = 0$. Now let us include two constraint equations and, for definiteness, take three unknowns. The method of the previous section gives

$$\left[\begin{array}{cccc|cc} 1 & & & & -d_1 & -d_2 \\ & 1 & & & g_{11} & g_{21} \\ & & 1 & & g_{12} & g_{22} \\ & & & 1 & g_{13} & g_{23} \\ \hline -d_1 & g_{11} & g_{12} & g_{13} & 0 & 0 \\ -d_2 & g_{21} & g_{22} & g_{23} & 0 & 0 \end{array}\right] \left[\begin{array}{c} 1 \\ x_1 \\ x_2 \\ x_3 \\ \hline \lambda_1 \\ \lambda_2 \end{array}\right] = \begin{bmatrix} v \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{6.46}$$

Equation (6.46) has a size equal to the number of variables plus the number of constraints. It may be solved numerically or it may be reduced to a matrix whose size is given by the number of constraints. Let us split up (6.46) into two equations:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} g_{11} & g_{21} \\ g_{12} & g_{22} \\ g_{13} & g_{23} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{6.47}$$

and

$$\begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \tag{6.48}$$

We abbreviate these equations by $\mathbf{x} + \mathbf{G}^T\lambda = \mathbf{0}$ and $\mathbf{G}\mathbf{x} = \mathbf{d}$, Premultiply (6.47) by $\mathbf{G}$,

$$\mathbf{G}\mathbf{x} + \mathbf{G}\mathbf{G}^T\lambda = \mathbf{0}$$

insert (6.48)

$$\mathbf{d} + \mathbf{G}\mathbf{G}^T\lambda = \mathbf{0}$$

solve for λ

$$\lambda = -(\mathbf{G}\mathbf{G}^T)^{-1}\mathbf{d}$$

put back into (6.47)

$$\mathbf{x} = \mathbf{G}^T(\mathbf{G}\mathbf{G}^T)^{-1}\mathbf{d}$$

Written out in full this is

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \mathbf{G}^T \end{bmatrix} \begin{bmatrix} \mathbf{G}\mathbf{G}^T \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{d} \end{bmatrix} \tag{6.49}$$

This is the final result, a minimum wiggliness solution $\mathbf{x}$ which exactly satisfies an underdetermined set called the constraint equations.

EXERCISES:

1 If wiggliness is defined by (6.42) instead of (6.44), what form does (6.49) take?

2 Given the mass and moment of inertia of the earth, calculate mass density as a function of depth utilizing the principle of minimum wiggliness (6.49). What criticism do you have of this procedure? (HINT: An elegant solution uses integrals instead of infinite sums.)

3 Use the techniques of this section on (6.40) to reduce the size of the matrices to be inverted.

6.4 HOUSEHOLDER TRANSFORMATIONS AND GOLUB'S METHOD

Our previous discussions of least squares always led us to matrices of the form $\mathbf{A}^T\mathbf{A}$ which then needed to be inverted. Golub's method of using Householder transformations works directly with the matrix $\mathbf{A}$ and has the advantage that it is considerably more accurate than methods with invert $\mathbf{A}^T\mathbf{A}$. It seems that about twice as much precision is required to invert $\mathbf{A}^T\mathbf{A}$ than is needed to deal directly with $\mathbf{A}$. Another reason for learning about Golub's method is that the calculation is organized in a completely different way; therefore, it will often turn out to have other advantages or disadvantages which differ from one application to the next.

A reflection transformation is a matrix of the form $\mathbf{R} = (\mathbf{I} - 2\mathbf{v}\mathbf{v}^T/\mathbf{v}^T\mathbf{v})$ where $\mathbf{v}$ is an arbitrary vector. Obviously $\mathbf{R}$ is symmetric, that is, $\mathbf{R} = \mathbf{R}^T$. It also turns out that the reflection transformation is its own inverse, that is, $\mathbf{R} = \mathbf{R}^{-1}$. To see this, we verify by substitution that $\mathbf{R}^2 - \mathbf{I}$.

$$\begin{aligned} \left(\mathbf{I} - \frac{2\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}}\right)^2 &= \mathbf{I} - \frac{4\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} + \frac{4\mathbf{v}(\mathbf{v}^T\mathbf{v})\mathbf{v}^T}{(\mathbf{v}^T\mathbf{v})^2} \\ &= \mathbf{I} \end{aligned} \tag{6.50}$$

A matrix transformation $\mathbf{M}$ is said to be *unitary* if $\mathbf{M}^T\mathbf{M} = \mathbf{I}$. When a matrix $\mathbf{M}$ is unitary it means that the vector $\mathbf{x}$ has the same length as the vector $\mathbf{M}\mathbf{x}$. These lengths are $\mathbf{x}^T\mathbf{x}$ and $(\mathbf{M}\mathbf{x})^T(\mathbf{M}\mathbf{x}) = \mathbf{x}^T\mathbf{M}^T\mathbf{M}\mathbf{x} = \mathbf{x}^T\mathbf{I}\mathbf{x} = \mathbf{x}^T\mathbf{x}$ which are the same. Reflection transformations are unitary because $\mathbf{R}^{-1} = \mathbf{R}^T$. They have a simple physical interpretation. Consider an orthogonal coordinate system in which one of the coordinate axes is aligned along the $\mathbf{v}$ vector. Reflection transformation reverses the sign of this coordinate axis vector (since $\mathbf{R}\mathbf{v} = -\mathbf{v}$) but it leaves unchanged all the other coordinate axis vectors. Thus it is obvious geometrically that reflection transformations preserve lengths and that applying the transformation twice returns any original vector to itself. Now, we seek a special reflection transformation called the Householder transformation which converts a matrix of the form on the left side to the form on the right where a is an arbitrary element

$$\mathbf{H}\begin{bmatrix} a & a & a & a \\ 0 & a & a & a \\ 0 & 0 & a & a \\ 0 & 0 & a & a \\ 0 & 0 & a & a \end{bmatrix} = \begin{bmatrix} a & a & a & a \\ 0 & a & a & a \\ 0 & 0 & a & a \\ 0 & 0 & 0 & a \\ 0 & 0 & 0 & a \end{bmatrix} \tag{6.51}$$

Having determined the required transformation, we will know how to convert any matrix to an upper triangular form like

$$\begin{bmatrix} a & a & a & a \\ 0 & a & a & a \\ 0 & 0 & a & a \\ 0 & 0 & 0 & a \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{6.52}$$

by a succession of Householder transforms. Golub recognized the value of this technique in solving overdetermined sets of simultaneous equations. He noted that when the error vector $\mathbf{e} = \mathbf{Ax} - \mathbf{b}$ is transformed by a unitary matrix $\mathbf{Ue}$ the problem of minimizing the length $(\mathbf{e}^T\mathbf{U}^T\mathbf{Ue})^{1/2}$ of $\mathbf{Ue}$ by variation of $\mathbf{x}$ reduces to exactly the same problem as minimizing the length $(\mathbf{e}^T\mathbf{e})^{1/2}$ of $\mathbf{e}$ with respect to variation of $\mathbf{x}$. Thus a succession of Householder transforms could be found to reduce $\mathbf{e} = \mathbf{Ax} - \mathbf{b}$ to the form

$$\begin{bmatrix} \\ \mathbf{e}_1 \\ \\ - \\ \\ \mathbf{e}_2 \\ \\ \end{bmatrix} = \begin{bmatrix} a & a & a \\ 0 & a & a \\ 0 & 0 & a \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} - \begin{bmatrix} a \\ a \\ a \\ a \\ a \\ a \\ a \end{bmatrix} \tag{6.53}$$

Now for the clever observation that because of the zeros in the bottom part of the transformed $\mathbf{A}$ matrix there is no possibility of choosing any x_i values which alter $\mathbf{e}_2$ in any way. The top part of the transformed $\mathbf{A}$ matrix is an upper triangular matrix which for any value of $\mathbf{e}_1$ can be solved exactly for the x_i. The least-squares solution x_i is the one for which $\mathbf{e}_1$ has been set equal to zero.

Now we turn to the task of finding the special reflection transformation, called the Householder transformation, which accomplishes (6.51). Observe that the left-hand operator below is a reflection transformation for any numerical choice of s.

$$\left\{\begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix} - \frac{2}{(a_3-s)^2+{a_4}^2+{a_5}^2}\begin{bmatrix} 0 \\ 0 \\ a_3-s \\ a_4 \\ a_5 \end{bmatrix}\begin{bmatrix} 0 & 0 & (a_3-s) & a_4 & a_5 \end{bmatrix}\right\}\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix}$$

$$= \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ (a_3-s) \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ s \\ 0 \\ 0 \end{bmatrix} \tag{6.54}$$

Alternatively, if (6.54) is to be valid, then s must take a particular value such that

$$1 = \frac{2}{(a_3-s)^2+{a_4}^2+{a_5}^2}\begin{bmatrix} 0 & 0 & (a_3-s) & a_4 & a_5 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} \tag{6.55}$$

or

$$1 = \frac{-2sa_3+2({a_3}^2+{a_4}^2+{a_5}^2)}{s^2-2sa_3+({a_3}^2+{a_4}^2+{a_5}^2)} \tag{6.56}$$

This will be true only for s given by

$$s = \pm({a_3}^2+{a_4}^2+{a_5}^2)^{1/2} \tag{6.57}$$

Now let us see why the left-hand operator in (6.54) can achieve (6.51). Choice of the a vector as the third column in the matrix of (6.51) introduces the desired zeros on the right-hand side. Finally, it is necessary also to observe that this choice of **H** does not destroy any of the zeros which already existed on the left-hand side in (6.51).

A subroutine for least squares fitting programmed by Don C. Riley. (This program does not do the square matrix case. It is necessary that $M > N$.)

```
      SUBROUTINE GOLUB  (A,X,B,M,N)
C
C     A(M,N) ; B(M)   GIVEN WITH M>N    SOLVES FOR X(N) SUCH THAT
C                 ||  B - AX || = MINIMUM
C     METHOD OF G.GOLUB,  NUMERISCHE MATHEMATIK  7,206-216 (1965)
C
      IMPLICIT DOUBLE PRECISION (D)
      REAL A(M,N),X(N),B(M),U(50)
C.........DIMENSION U(M)
C.........PERFORM N ORTHOGONAL TRANSFORMATIONS TO A(.,.) TO
```

```
C.........UPPER TRIANGULARIZE THE MATRIX
        DO 3010 K=1,N
        DSUM=0.0D0
        DO 1010 I=K,M
        DAJ=A(I,K)
  1010  DSUM=DSUM+DAJ**2
        DAI=A(K,K)
        DSIGMA=DSIGN(DSQRT(DSUM),DAI)
        DBI=DSQRT(1,0D0+DAI/DSIGMA)
        DFACT=1.0D0/(DSIGMA*DBI)
        U(K)=DBI
        FACT-=DFACT
        KPLUS=K+1
        DO 1020 I=KPLUS,M
  1020 U(I)=FACT*A(I,K)
C.........I - UU' IS A SYMMETRIC, ORTHOGONAL MATRIX WHICH WHEN APPLIED
C......... TO A(.,.) WILL ANNIHILATE THE ELEMENTS BELOW THE DIAGONAL K
        DO 2030 J=K,N
c.........APPLY THE ORTHOGONAL TRANSFORMATION
        FACT=0.0
        DO 2010 I=K,M
  2010  FACT=FACT+U(I)*A(I,J)
        DO 2020 I=K,M
  2020 A(I,J)=A(I,J)-FACT*U(I)
  2030 CONTINUE
        FACT=0.0
        DO 2040 I=K,M
  2040 FACT=FACT+U(I)*B(I)
        DO 2050 I=K,M
  2050 B(I)=B(I)-FACT*U(I)
  3010 CONTINUE
C.........BACK SUBSTITUTE TO RECURSIVELY YIELD X(.)
        X(N)=B(N)/A(N,N)
        LIM=N-1
        DO 4020 I=1,LIM
        IROW=N-1
        SUM=0.0
        DO 4010 J=1,I
  4010 SUM=SUM+X(N-J+1)*A(IROW,N-J+1)
  4020  X(IROW)=(B(IROW)-SUM)/A(IROW,IROW)
        RETURN
        END
```

Householder transformations can also be used in problems with constraints. In the set

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{A} \end{bmatrix} [\mathbf{x}] = \begin{bmatrix} \mathbf{d} \\ \mathbf{b} \end{bmatrix} \tag{6.58}$$

one may desire to satisfy the top block exactly and the bottom block only in the least-squares sense. Define $\mathbf{y}$ as a succession of Householder transforms on $\mathbf{x}$; for example, $\mathbf{y} = \mathbf{H}_2\mathbf{H}_1\mathbf{x}$. Then substitute $\mathbf{x} = \mathbf{H}_1\mathbf{H}_2\mathbf{H}_2\mathbf{H}_1\mathbf{x} = \mathbf{H}_1\mathbf{H}_2\mathbf{y}$ into (6.58). Householder transforms used as postmultipliers on the matrix of (6.58) can be chosen to introduce zeros in the top two rows of (6.58), for example

$$\begin{bmatrix} a & 0 & 0 & 0 \\ a & a & 0 & 0 \\ a & a & a & a \\ a & a & a & a \\ a & a & a & a \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \approx \begin{bmatrix} d_1 \\ d_2 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{6.59}$$

Now we could use premultiplying Householder transforms on (6.59) to bring it to the form

$$\begin{bmatrix} a & 0 & 0 & 0 \\ a & a & 0 & 0 \\ a & a & a & a \\ a & a & 0 & a \\ a & a & 0 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \approx \begin{bmatrix} d_1 \\ d_2 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{6.60}$$

Since the top two equations of (6.59) or of (6.60) are to be satisfied exactly, then y_1 and y_2 are uniquely determined. They cannot be adjusted to help attain minimum error in the bottom three equations. Likewise the top two equations place no restraint on y_3 and y_4, so they may be adjusted to produce minimum error in the bottom three equations. No amount of adjustment in y_3 and y_4 can change the amount of error in the last equation, so we can ignore the last equation in the determination of y_3 and y_4. The third and fourth equations can be satisfied with zero error by suitable choice of y_3 and y_4. This must be the minimum-squared-error answer. Given $\mathbf{y}$ we can go back and get $\mathbf{x}$ with $\mathbf{x} = \mathbf{H}_1\mathbf{H}_2\mathbf{y}$.

6.5 CHOICE OF A MODEL NORM

In recent years, a popular view of geophysical data modeling has been that the earth is a continuum and that we should regard the number of unknowns as infinite but the number of our observations as finite. In a computer one therefore approximates the situation by a highly underdetermined system of simultaneous equations. In order to get a unique answer, the solution should extremalize some integral. In practice, a sum of squares is often minimized in such a way as to produce a smooth solution. A typical mathematical formulation is to do a least squares fitting of an equation set like

$$\begin{bmatrix} \mathbf{A} \\ \hline ? \end{bmatrix} \mathbf{x} \approx \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix} \tag{6.61}$$

where the top of block $\mathbf{A}$ denotes the underdetermined constraint equations with the data vector $\mathbf{d}$; the unknowns are in the $\mathbf{x}$ vector; and the bottom block is a band matrix (all the nonzero elements cluster about the diagonal) which says that some filtered version of $\mathbf{x}$ should vanish. The filter is often a roughing filter like the first-difference operator. In the absence of data, the first-difference operator leads to a constant solution which is sensible. What is not sensible is that it forces the result to be smoothed even though realistic earth models often contain step discontinuities (where two homogeneous media lie in contact).

The choice of a filter is a rather subjective matter, the choice often being made on the basis of the solution it will produce. Unfortunately, the solution is often a rather sensitive function of the subjectively chosen weights and filters; this fact makes the whole business an art, a matter of experience and judgment. General purpose theories of inversion exist, but they do not prepare the geophysicist to exploit the peculiarities of any particular situation or data set. Inversion theories, like mathematical statistics, should be used like a lamp post – to light the way, not to lean upon.

One useful concept in inversion theory is the idea of coordinate-system invariance. The idea is that one should get the same answer in an electrical conductivity problem whether one parameterizes the earth by an unknown conductivity at every point on a sufficiently dense mesh, or one parameterizes with resistivity on the mesh, or one parameterizes by coefficients of some expansion in a complete set of basis functions. Clearly, the idea of fitting low-order coefficients in some expansion setting the high-order coefficients equal to zero is not a coordinate-invariant approach. A different origin for polynomial expansions can change everything. A different set of basis functions would change everything. Of course, it is not essential to use a coordinate-system invariant technique in data inversion. But if one does not, one should beware of the sensitivity of one's solution to changes in the coordinate system.

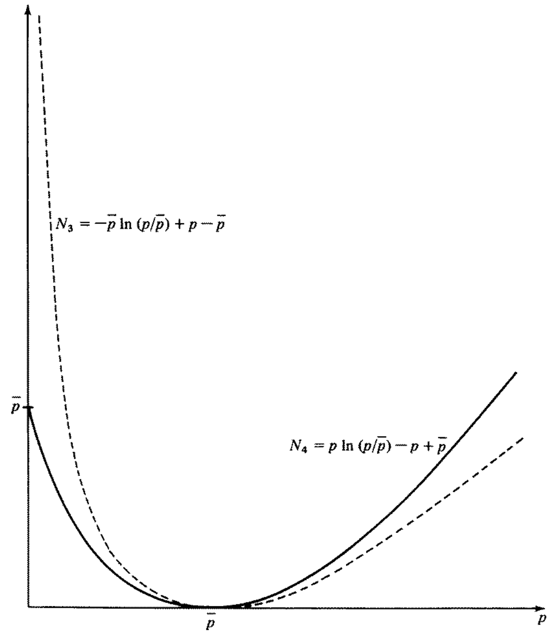

Figure 6.1: Minimizing either of these two functions will drive p toward $\bar{p}$. $\boxed{\text{c6-6-2}}$ [NR]

Let us consider some inversion procedures which are coordinate-system invariant. We will restrict ourselves to physical problems in which we can identify a positive density function p as energy or dissipation per unit volume. Let us denote by $\bar{p}$ the value of this power as a function of space in the default model of the earth. The default model is the one we want to find when we have no measurements. It will often be one in which the material properties are constant functions of space. Now we will need some functions which we will call *model norms*. They have the properties of being positive for all (positive) p and $\bar{p}$ and being minimized at $p = \bar{p}$. Some examples plotted in Figure 6.1 are

$$\begin{aligned} N_1(p, \bar{p}) &= |p - \bar{p}| \\ N_2(p, \bar{p}) &= \frac{(p - \bar{p})^2}{\bar{p}} \\ N_3(p, \bar{p}) &= -\bar{p} \ln \left(\frac{p}{\bar{p}} \right) + p - \bar{p} \\ N_4(p, \bar{p}) &= p \ln \left(\frac{p}{\bar{p}} \right) - p + \bar{p} \end{aligned}$$

Now let the adjustable earth properties be denoted by x, a function of space. We can choose x to minimize some volume integral of one of the model norms subject to the constraint that the model produce the required observations. Sometimes we have observations from $j = 1, 2, 3, \ldots, n_s$ source locations. We then need to compute the default power distribution $\bar{p}_j$ for each. Then we can minimize a sum of volume integrals

$$\min_x \sum_{j=1}^{n_s} \int N(p_j, \bar{p}_j)\, dV$$

subject to fitting all the data values.

It will be noted that the model-norm functions are all homogeneous of order 1. This means that $N(ap,\, a\bar{p}) = aN(p,\, \bar{p})$ for $a > 0$. This is our assurance that N is a volume density. Without this property we would have the difficulty that a sum of $N_k(p,\, \bar{p})$ over a set of subvolumes ΔV_k would change as the mesh were refined. Coordinate-system invariance is provided by the usual rules for conversion of volume integrals from one coordinate system to another.

Now let us take up an example from filter theory which turns out to be related to maximum-entropy spectral estimation. We are given a known input spectrum $R(Z)$ and are to find the finite length filter $X(Z) = x_0 + x_1 Z + x_2 Z^2$ whose output is as white as possible in the sense of minimizing the integral of N_3 across the spectrum. Let the spectrum of the filter be $S(Z) = \bar{X}(1/Z)\, X(Z)$. We have $\bar{p} = 1$ and $p = R(Z)\, S(Z)$. Thus, the minimization is

$$\min \quad = \quad \int (-\ln RS + RS) d\omega$$

Setting the derivative with respect to $\bar{x}_k$ equal to zero we have

$$\begin{aligned} 0 &= \int \left(-\frac{1}{S} + R \right) \frac{\partial S}{\partial \bar{x}_k} d\omega \\ &= \int \left(-\frac{1}{S} + R \right) Z^{-k} X(Z) d\omega \end{aligned}$$

$$= \int Z^{-k} \left[-\frac{1}{\bar{X}(1/Z)} + R(Z)X(Z) \right] d\omega$$

Since we know that minimum-phase functions can represent any spectrum, we take $\bar{X}(1/Z)^{-1}$ to be expandable as $(\bar{b}_0 + \bar{b}_1/Z + \bar{b}_2/Z^2 + \cdots)$

$$0 = \int Z^{-k} \left[-\left(b_0 + \frac{b_1}{Z} + \frac{b_2}{Z^2} + \cdots \right) + RX \right] d\omega$$

We recall that this integral selects the coefficient of Z^0 of the argument. If we suppose that the filter is constrained to have $x_k = 0$ for $k \geq 3$, we get the familiar Toeplitz system

$$\begin{bmatrix} r_0 & r_{-1} & r_{-2} \\ r_1 & r_0 & r_{-1} \\ r_2 & r_1 & r_0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_0 \\ 0 \\ 0 \end{bmatrix}$$

6.6 ROBUST MODELING

The median and the mean are two kinds of statistical average. In a normal situation they behave in about the same way. At the present time physical scientists almost always use the mean and hence tend to be unaware of the dramatic ability of the median to cast off the effect of blunders in the data. As an example, consider an expensive, all-day-long experiment which yields only one number for a result. On the first day the result is 2.17, on the second day it is 2.14, and on the third and final day it is 1638.03. The mean of these results is 547.78 but the median (middle value) is 2.17. If one suspects a blunder on the third day, one will obviously prefer the median. Statisticians call this the "robust" property of the median. The objective of this section is to show how many kinds of geophysical data fitting can be made to be robust. In particular, all the calculations we now do which amount to solving overdetermined linear simultaneous equations by means of summed squared-error minimization can be made robust by minimizing summed absolute values of errors, instead. Computer costs are often comparable to those of least-squares methods. The algorithms turn out to solve a slightly broader class of problem than minimizing the summed absolute errors. Positive errors may be penalized with a differend weight factor than negative errors. Such an arrangement is called an asymmetric linear norm. A special case of an asymmetric norm is an inequality. Not surprisingly, it turns out that all linear programming problems are special cases of asymmetric linear-norm problems and the solution techniques for asymmetric linear norms are similar to linear programming.

First, we will see why means and medians relate to squares and absolute values. Let x_i be an arbitrary number. Let us define m_2 by the minimization of the sum of squared differences (called the L_2 norm) between m_2 and x_i:

$$m_2 : \qquad \min \sum_{i=1}^{N} (m_2 - x_i)^2 \tag{6.62}$$

It is a straight forward task to find the minimum by setting the partial derivative of the sum with respect to m_2 equal to zero. We get

$$0 = \sum_{i=1}^{N} 2(m_2 - x_i)$$

or

$$m_2 = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{6.63}$$

Obviously, m_2 has turned out to be given by the usual definition of *mean.* Next, let us define m_1 by minimizing the summed absolute values (called the L_1 norm). We have

$$m_1 : \qquad \min \sum_{i=1}^{N} |m_1 - x_i| \tag{6.64}$$

To find the minimum we may again set the partial derivative with respect to m_1 equal to zero

$$0 = \sum_{i-1}^{N} \mathrm{sgn}\,(\mathrm{m_1} - \mathrm{x_i}) \tag{6.65}$$

Here the sgn function is +1 when the argument is positive, -1 when the argument is negative, and somewhere in between when the argument is zero. Equation (6.65) says that m_1 should be chosen so that m_1 exceeds x_i for $N/2$ terms, m_1 is less than x_i for $N/2$ terms, and if there is an x_i left in the middle, m_1 equals that x_i. this defines m_1 as a *median.* [For an even number N the definition (6.64) requires only the m_1 lie anywhere between the middle two values of the x_i.]

The computational cost for a mean is proportional to N, the number of points. The cost for completely ordering a list of numbers is $N \ln \mathrm{N}$, but complete ordering is not required for finding the median. Hoare provided an algorithm for finding the median which requires about $3N$ operations. A computer algorithm based on Hoare's algorithm will be provided for weighted medians. Weighted medians are analogous to weighted sums. Ordinarily, 2.17 is taken to be the median of the numbers (2.14, 2.17, 1638.03) because we implicitly applied weights (1, 1, 1). If we applied weights (3, 1, 1) it would be like having the numbers 2.14, 2.14, 2.14, 2.17, 1638.03 and the median would then be 2.14. Formally, a weight median may be defined by the minimization

$$m_1 : \qquad \min \sum |w_i|\,|m_1 - x_i| \tag{6.66}$$

Obviously if the weight factors are all unity, this reduces to the earlier definition whereas using a weight factor equal to 3, for example, is just like including the same term three times with a weight of 1. Figure 6.2 illustrates the definition (6.66) for a simple case. From Figure 6.2 it is apparent that a median is always equal to one of the x_i4 even if the weights are not integers. If the weights are all unity and there is an even number of numbers, then the error norm will be flat between the two middle

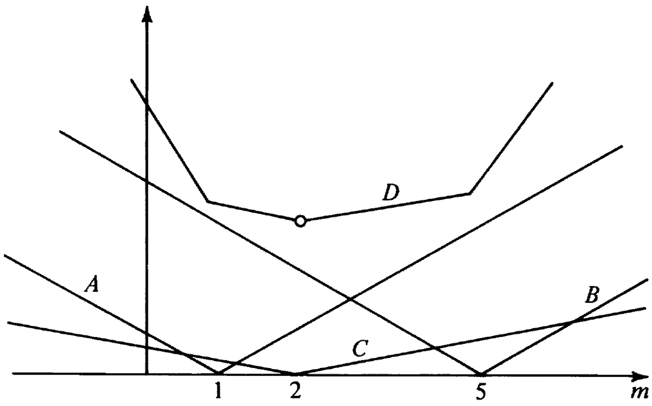

Figure 6.2: A sum of weighted absolute value norms. The function labeled A is $.5|m-1|$, B is $.5|m-5|$, C is $.1|m-2|$, and D is the sum of A, B, and C. The sum D is minimized at $m-2$, a point which exactly solves $C = 0 = .1|m-2|$. $\boxed{\text{c6-6-3}}$ [NR]

numbers. Then any value in between satisfies our definition of median by minimizing the sum.

Let us rearrange (6.66) by bringing $|w_i|$ into the other absolute-value function. We have

$$m_1: \qquad \min \sum_i ||w_i|m - |w_i|x_i| = \min \sum_i |w_i m - w_i x_i| \tag{6.67}$$

We will now relabel things from the conventions of statistics to the usual convention of simultaneous equations and linear programming. Let

$$a_i := w_i \tag{6.68}$$
$$b_i := w_i x_i \tag{6.69}$$
$$x := m$$

With these new definitions (6.67) becomes

$$x: \qquad \min \sum_i |a_i x - b_i| \tag{6.70}$$

The definition (6.70) says, in other words, to solve the rank one overdetermined equations

$$\mathbf{a}x \quad \cong \quad \mathbf{b} \tag{6.71}$$

for x by minimizing the L_1 norm. This is, in effect, a weighted median problem. If (6.71) were solved by minimizing the L_2 norm (least squares) x would turn out to be the weighted average $x = (\mathbf{a} \cdot \mathbf{b})/(\mathbf{a} \cdot \mathbf{a})$.

We now consider a solution technique for the minimization (6.66). Essentially, it is Hoare's algorithm. On a trial basis we select a random equation from the set (6.71) to be exactly satisfied. This equation, called the basis equation, can be denoted $a_k x_{\text{trial}} = b_k$. Inserting x_{trial} into (6.71) we get equations with positive errors, negative errors, and zero errors. If we have been lucky with x_{trial}, then we find that the zero error group has enough weight to swing the balance between positive and negative weights in either direction. Otherwise, we must pick a new trial basis equation from the stronger of the positive or negative group. Fortunately, we need no longer look into the weaker group because these residuals cannot change signs as we descent into minimum. This may be seen geometrically on a figure like Figure 6.2. We always wish to go downhill, so once it has been ascertained that a data point is uphill from the present point then it is never necessary to reinspect the uphill point. Thus, the size of the group being inspected rapidly diminishes.

Below are subroutines to compute weighted and skewed medians. (A “skewed median” is often called a quantile.) This subroutine is somewhat complicated because it takes special care to do the correct thing when weight factors are zero and because it provides pointers to all equations (occasionally there is more than one) which are satisfied at the final minimum.

```
subroutine skewer(n,w,f,gu,gd,small,k,t,ml,mh)
# For a Fortran version all in upper case see FGDP p. 126
#       solve rank 1 overdetermined equations with skew norm
#       inputs- n,w,f,gu,gd,small,k.  outputs- k,t,ml,mh.
#   find t to minimize
#        n
#  ls = sum skewnorm(k,f(k)-w(k)*t)
#       k=1
#  where            (gu(k)*(er-small) if er>+small gu>0
#  skewnorm(k,er) = (gd(k)*(er+small) if er<-small gd<0
#                   (0.               if abs(er) <= small
#       gu,gd,w,and f are referenced indirectly as w(k(i)),i=1,n etc
#       minima will be at equations k(ml),k(ml+1),...k(mh).
real w(n),f(n),gu(n),gd(n),g(100)
integer k(n)
integer i,n,low,large,ml,mh,mlt,mht,itry,l
real gn,gp,t,er,gnt,gpt,gplx,gmix,small,grad
low = 1
large = n
ml = n
mh = 1
gn = 0.
gp = 0.
do itry = 1,n {
        l = k(low+mod((large-low)/3+itry,large-low+1))
        if (abs(w(l))!=0.) {                # "!=" means "not equal"
                t = f(l)/(w(l))
                f(l) = w(l)*t
                do i = low,large {
                        l = k(i)
                        er = f(l)-w(l)*t
                        g(l) = 0.
                        if (er>small)
                                g(l) = -w(l)*gu(l)
                        if (er<(-small))
                                g(l) = -w(l)*gd(l)
```

```
                        }
                call split(low,large,k,g,mlt,mht)
                gnt = gn
                do i = low,mlt
                        gnt = gnt+g(k(i))
                gpt = gp
                do i = mht,large
                        gpt = gpt+g(k(i))
                gplx = 0.
                gmix = 0.
                do i = mlt,mht {
                        l = k(i)
                        if (w(l)<0.) {
                                gplx = gplx-w(l)*gu(l)
                                gmix = gmix-w(l)*gd(l)
                                }
                        if (w(l)>0.) {
                                gplx = gplx-w(l)*gd(l)
                                gmix = gmix-w(l)*gu(l)
                                }
                        }
                grad = gnt+gpt
                if ((grad+gplx)*(grad+gmix)<0.)
                        break 1
                if (grad>=0.)
                        low = mht+1
                if (grad<=0)
                        large = mlt-1
                if (low>large)
                        break 1
                if (grad>=0.)
                        gn = gnt+gmix
                if (grad<=0.)
                        gp = gpt+gplx
                if (grad+gplx==0.)
                        ml = mlt
                }
        if (grad+gmix==0.)
                mh = mht
        }
ml = min0(ml,mlt)
mh = max0(mh,mht)
return
end

subroutine split(low,large,k,g,ml,mh)
# For a Fortran version all in upper case, see FGDP p 127
# given g(k(i)),i=low,large
# then rearrange k(i),i=low,large and find ml and mh so that
# (g(k(i)),i=low,(ml-1)) < 0. and
# (g(k(i)),i=ml,mh)=0. and
# (g(k(i)),i=(mh+1),large) > 0.
real g(100)
integer k(100)
```

```
#integer low,large,k,ml,mh,keep,i,ii
integer low,large,ml,mh,keep,i,ii
ml = low
mh = large
repeat {
ml = ml-1
repeat
ml = ml+1
until(g(k(ml))>=0)
repeat {
mh = mh+1
repeat
mh = mh-1
until(g(k(mh))<=0)
keep = k(mh)
k(mh) = k(ml)
k(ml) = keep
if (g(k(ml))!=g(k(mh)))
break 1 # Break out of enclosing "repeat{"
do i = ml,mh {
ii = i
if (g(k(i))!=0.0)
go to 30
}
break 2 # Break out of two enclosing "repeat{"'s
30  keep = k(mh)
k(mh) = k(ii)
k(ii) = keep
}
}
return
end
```

The next step up the ladder of complexity is to consider two unknowns. The obvious generalization of (6.71) is

$$\begin{bmatrix} a_1 & c_1 \\ a_2 & c_2 \\ \vdots & \vdots \\ a_k & c_k \\ \vdots & \vdots \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad \cong \quad \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \\ \vdots \end{bmatrix} \tag{6.72}$$

We will assume that the reader is familiar with the solution to (6.72) by the least-squares method. Solution by minimizing the sum of the absolute values of the errors begins in a similar way. We begin by defining the error

$$E \quad = \quad \sum_{k=1}^{N} |b_k - a_k x - c_k y| \tag{6.73}$$

Then we will set the x derivative of the error equal to zero and the y derivative of the error equal to zero.

$$0 = \frac{\partial E}{\partial x} = \sum_{k=1}^{N} -a_k \operatorname{sgn}\left(\mathrm{b_k} - \mathrm{a_k x} - \mathrm{c_k y}\right) \tag{6.74}$$

$$0 = \frac{\partial E}{\partial y} = \sum_{k=1}^{N} -c_k \operatorname{sgn}\left(\mathrm{b_k} - \mathrm{a_k x} - \mathrm{c_k y}\right) \tag{6.75}$$

Now we run into a snag. If the sgn function always takes the value +1 or -1, then (6.74) implies that the a_k may be divided into two piles of equal weight. Clearly many, indeed most, collections of numbers cannot be so balanced (for example, if all the a_i except one are integers). The difficulty will be avoided if at least one of the equations of (6.72) is solved exactly so that sgn takes an indeterminate value for the term. Any algebraic confusion may be quickly dispelled by recollection of Figure 6.2 and the result that even with one unknown the minimum generally occurs at a corner where the first derivative is discontinuous. The same situation must again apply to (6.75). The usual situation is that for N equations and M unknowns precisely M of the N equations will be exactly satisfied in order to enable the error gradient to vanish at the minimum. Common usage in the field of linear programming is to refer to any nonsingular subset of M out of the N equations as a set of *basis equations.* The particular set of M equations which is solved when the error is minimized is called the *optimum basis.*

Although linear programming is a twentieth-century development, the basic ideas seem to have been well known before Laplace in the eighteenth century. Indeed, in the words of Gauss' *Theoria Motus Corporum Coelestium* which appeared in 1980: Laplace made use of another principle for the solution of linear equations, the number of which is greater than the number of unknown quantities, which had been previously proposed by Boscovich, namely that the differences themselves, both all of them taken positively, should make up as small a sum as possible. It can be easily shown, that a system of values of unknown quantities, derived from this principle alone, must necessarily (except the special cases in which the problem remains, to some extent, indeterminate) exactly satisfy as many equations out of the number proposed, as there are unknown quantities, so that the remaining equations come into consideration only so far as they help to *determine the choice.*

Next a simple but effective technique for descent down a multidimensional error surface will be described. The position of $\mathbf{x}$ on a line through $\mathbf{x}_0$ can be indicated by a scalar parameter t. The direction of the line can be specified by an M component vector $\mathbf{g}$. Then any point $\mathbf{x}$ on the line may be represented as

$$\mathbf{x} = \mathbf{x}_0 + \mathbf{g}t \tag{6.76}$$

Inserting (6.76) into the overdetermined set

$$\mathbf{Ax} \cong \mathbf{b} \tag{6.77}$$

we obtain

$$\mathbf{A}(\mathbf{x}_0 + \mathbf{g}t) \cong \mathbf{b} \tag{6.78}$$

$$(\mathbf{Ag})t \cong \mathbf{b} - \mathbf{Ax}_0 \tag{6.79}$$

Defining $\mathbf{w}$ and $\mathbf{e}$ by

$$\mathbf{w} = \mathbf{Ag} \tag{6.80}$$

$$\mathbf{e} = \mathbf{b} - \mathbf{Ax}_0 \tag{6.81}$$

(6.79) becomes

$$\mathbf{w}t \cong \mathbf{e} \tag{6.82}$$

Solving (6.82) by minimizing the summed absolute errors also given the minimum error along the line in (6.78). But (6.82) is the weighted median problem disussed earlier. Recall that the solution t to (6.82) which gives minimum absolute error will exactly satisfy one of the equations in (6.82). Let us say $t = e_k/w_k$. For this value of t, the kth equation in (6.77) will also be satisfied exactly. The kth equation in now considered to be a good candidate for the basis, and we will next show how to pick the vector $\mathbf{g}$ so as to continue to satisfy the kth equation (stay on the kth hyperplane) as we adjust t in the next iteration.

Now we need a set of basis equations. This is a set of M equations which is temporarily taken to be satisfied. Then, as new equations are introduced into the basis by the weighted median solution, old equations are dropped out. The strategy of the present algorithm is merely to drop out the one which has been in longest. Let us denote our basis equations by

$$\mathbf{A}'\mathbf{x} = \mathbf{d}' \tag{6.83}$$

$\mathbf{A}'$ is a square matrix. The inverse of the matrix $\mathbf{A}'$ will be required and will be denoted by $\mathbf{B}$. Now suppose we decide to throw out the pth equation from the basis matrix $\mathbf{A}'$. Then for $\mathbf{g}$ we select the pth column of $\mathbf{B}$. To see why this works not that since $\mathbf{A}'\mathbf{B} = \mathbf{I}$ the M vector $\mathbf{A}'\mathbf{g}$ will now be the pth column from the identity matrix. Therefore, in the N vector $\mathbf{w} = \mathbf{Ag}$ there is a component equal to $+1$, there are M -1 components equal to 0, and there are $N - M$ other unspecified elements. If the kth equation in (6.77) or (6.82) has been kept in the basis (6.83), then the kth equation in $\mathbf{Ag}t = ingt - \mathbf{Ax}$ now reads

$$\text{zero t} = \text{zero} \tag{6.84}$$

The left-hand zero is an element from the identity matrix and the right-hand zero is from the statement that the kth equation is exactly satisfied. Clearly, we can now adjust t as much as we like to attain a new local minimum and the kth equation will still be exactly satisfied. There is also one equation of the form

$$\text{one t} = \text{zero} \tag{6.85}$$

It will be satisfied only if t is zero. Geometrically, this means that if we must move to get to a minimum, then this equation is not satisfied and so we are jumping from this hyperplane. This equation is the one leaving the basis. Of course, if t turns out to be zero, then it reenters the basis. The foregoing steps are iterated until such time that for M successive iterations the quations thrown out of the basis by virtue of its age has immediately reappeared because $t = 0$. This means that the basis can no longer be improved and we have arrived at the optimum basis and the final solution.