Chapter 2

FORMAL PROBLEMS IN LINEAR ALGEBRA

2.1. INTRODUCTION

A great many practical problems in the scientific and engineering world give rise to models or descriptions of reality which involve matrices. In consequence, a very large proportion of the literature of numerical mathematics is devoted to the solution of various matrix equations. In the following sections, the major formal problems in numerical linear algebra will be introduced. Some examples are included to show how these problems may arise directly in practice. However, the formal problems will in most cases occur as steps in larger, more difficult computations. In fact, the algorithms of numerical linear algebra are the keystones of numerical methods for solving real problems.

Matrix computations have become a large area for mathematical and computational research. Textbooks on this subject, such as Stewart (1973) and Strang (1976), offer a foundation useful for understanding the uses and manipulations of matrices and vectors. More advanced works detail the theorems and algorithms for particular situations. An important collection of well-referenced material is Golub and Van Loan (1983). Kahaner, Moler and Nash (1989) contains a very readable treatment of numerical linear algebra.

2.2. SIMULTANEOUS LINEAR EQUATIONS

If there are n known relationships

\[ A_{ij} x_i + A_{i2} x_2 + \ldots + A_{in} x_n = b_i \quad i = 1, 2, \ldots, n \]  

between the n quantities \( x_j \) with the coefficients \( A_{ij} \) and right-hand side elements \( b_i, i = 1, 2, \ldots, n \), then (2.1) is a set of n simultaneous linear equations in n unknowns \( x_j, j = 1, 2, \ldots, n \). It is simpler to write this problem in matrix form

\[ A x = b \]  

where the coefficients have been collected into the matrix \( A \), the right-hand side is now the vector \( b \) and the unknowns have been collected as the vector \( x \). A further way to write the problem is to collect each column of \( A \) (say the jth) into a column vector (i.e. \( a_j \)). Then we obtain

\[ \sum_{i=1}^{n} a_j x_i = b. \]  

Numerous textbooks on linear algebra, for instance Mostow and Sampson (1969) or Finkbeiner (1966), will provide suitable reading for anyone wishing to
learn theorems and proofs concerning the existence of solutions to this problem. For the purposes of this monograph, it will suffice to outline a few basic properties of matrices as and when required.

Consider a set of $n$ vectors of length $n$, that is

$$a_1, a_2, \ldots, a_n.$$  \hspace{1cm} (2.4)

These vectors are linearly independent if there exists no set of parameters $x_j$, $j = 1, 2, \ldots, n$ (not all zero), such that

$$\sum_{i=1}^{n} a_i x_i = 0$$  \hspace{1cm} (2.5)

where $\mathbf{0}$ is the null vector having all components zero. If the vectors $a_j$ are now assembled to make the matrix $\mathbf{A}$ and are linearly independent, then it is always possible to find an $\mathbf{x}$ such that (2.2) is satisfied. Other ways of stating the condition that the columns of $\mathbf{A}$ are linearly independent are that $\mathbf{A}$ has full rank or

$$\text{rank}(\mathbf{A}) = n$$  \hspace{1cm} (2.6)

or that $\mathbf{A}$ is non-singular.

If only $k < n$ of the vectors are linearly independent, then

$$\text{rank}(\mathbf{A}) = k$$  \hspace{1cm} (2.7)

and $\mathbf{A}$ is singular. In general (2.2) cannot be solved if $\mathbf{A}$ is singular, though consistent systems of equations exist where $\mathbf{b}$ belongs to the space spanned by $\{a_j: j = 1, 2, \ldots, n\}$.

In practice, it is useful to separate linear-equation problems into two categories. (The same classification will, in fact, apply to all problems involving matrices.)

(i) The matrix $\mathbf{A}$ is of modest order with probably few zero elements (dense).

(ii) The matrix $\mathbf{A}$ is of high order and has most of its elements zero (sparse).

The methods presented in this monograph for large matrices do not specifically require sparsity. The question which must be answered when computing on a small machine is, ‘Does the matrix fit in the memory available?’

**Example 2.1. Mass - spectrograph calibration**

To illustrate a use for the solution of a system of linear equations, consider the determination of the composition of a mixture of four hydrocarbons using a mass spectrograph. Four lines will be needed in the spectrum. At these lines the intensity for the sample will be $b_i$, $i = 1, 2, 3, 4$. To calibrate the instrument, intensities $A_{ij}$ for the $i$th line using a pure sample of the $j$th hydrocarbon are measured. Assuming additive line intensities, the composition of the mixture is then given by the solution $\mathbf{x}$ of

$$\mathbf{A}\mathbf{x} = \mathbf{b}.$$

**Example 2.2. Ordinary differential equations: a two-point boundary-value problem**

Large sparse sets of linear equations arise in the numerical solution of differential
Formal problems in linear algebra

equations. Fröberg (1965, p 256) considers the differential equation
\[ y'' + \frac{y}{1+x^2} = 7x \]  
with the boundary conditions
\[
\begin{align*}
  y &= 0 & \text{for } x = 0 \\
  y &= 2 & \text{for } x = 1.
\end{align*}
\]

To solve this problem numerically, Fröberg replaces the continuum in \( x \) on the interval \([0, 1]\) with a set of \((n+1)\) points, that is, the step size on the grid is \( h = 1/n \). The second derivative is therefore replaced by the second difference at point \( j \)
\[
(y_{j+1} - 2y_j + y_{j-1})/h^2.
\]
The differential equation \(2.8\) is therefore approximated by a set of linear equations of which the \( j \)th is
\[
\begin{align*}
  \frac{y_{j+1} - 2y_j + y_{j-1}}{h^2} + \frac{y_j}{1+j^2h^2} &= 7jh \\
  y_{j+1} & \frac{2 - h^2 + 2j^2h^2}{1+j^2h^2} y_j + y_{j-1} = 7jh^3.
\end{align*}
\]

Because \( y_0 = 0 \) and \( y_n = 2 \), this set of simultaneous linear equations is of order \((n-1)\). However, each row involves at most three of the values \( y_j \). Thus, if the order of the set of equations is large, the matrix of coefficients is sparse.

2.3. THE LINEAR LEAST-SQUARES PROBLEM

As described above, \( n \) linear equations give relationships which permit \( n \) parameters to be determined if the equations give rise to linearly independent coefficient vectors. If there are more than \( n \) conditions, say \( m \), then all of them may not necessarily be satisfied at once by any set of parameters \( x \). By asking a somewhat different question, however, it is possible to determine solutions \( x \) which in some way approximately satisfy the conditions. That is, we wish to write
\[
A x \approx b
\]
where the sense of the sign \( \approx \) is yet to be defined.

By defining the residual vector
\[
r = b - Ax
\]
we can express the lack of approximation for a given \( x \) by the norm of \( r \)
\[
\| r \|. 
\]
This must fulfill the following conditions:
\[
\| r \| > 0 
\]
for \( r \neq 0 \), and \( \| 0 \| = 0 \),
\[
\| cr \| = \| c \| \cdot \| r \|
\]
for an arbitrary complex number \( c \), and
\[
\| r + s \| \leq \| r \| + \| s \| \tag{2.19}
\]
where \( s \) is a vector of the same order as \( r \) (that is, \( m \)).

Condition (2.19) is called the triangle inequality since the lengths of the sides of a triangle satisfy this relationship. While there exist many norms, only a few are of widespread utility, and by and large in this work only the Euclidean norm
\[
\| r \|_E = (r^T r)^{1/2} \tag{2.20}
\]
will be used. The superscript \( T \) denotes transposition, so the norm is a scalar. The square of the Euclidean norm of \( r \)
\[
r^T r = \sum_{i=1}^{m} r_i^2 \tag{2.21}
\]
is appropriately called the sum of squares. The least-squares solution \( x \) of (2.14) is that set of parameters which minimises this sum of squares. In cases where \( \text{rank}(A) < n \) this solution is not unique. However, further conditions may be imposed upon the solution to ensure uniqueness. For instance, it may be required that in the non-unique case, \( x \) shall be that member of the set of vectors which minimises \( r^T r \) which has \( x^T x \) a minimum also. In this case \( x \) is the unique minimum-length least-squares solution.

If the minimisation of \( r^T r \) with respect to \( x \) is attempted directly, then using (2.15) and elementary calculus gives
\[
A^T A x = A^T b \tag{2.22}
\]
as the set of conditions which \( x \) must satisfy. These are simply \( n \) simultaneous linear equations in \( n \) unknowns \( x \) and are called the normal equations. Solution of the least-squares problem via the normal equations is the most common method by which such problems are solved. Unfortunately, there are several objections to such an approach if it is not carefully executed, since the special structure of \( A^T A \) and the numerical instabilities which attend its formation are ignored at the peril of meaningless computed values for the parameters \( x \).

Firstly, any matrix \( B \) such that
\[
x^T B x > 0 \tag{2.23}
\]
for all \( x \neq 0 \) is called positive definite. If
\[
x^T B x \geq 0 \tag{2.24}
\]
for all \( x \), \( B \) is non-negative definite or positive semidefinite. The last two terms are synonymous. The matrix \( A^T A \) gives the quadratic form
\[
Q = x^T A^T A x \tag{2.25}
\]
for any vector \( x \) of order \( n \). By setting
\[
y = A x \tag{2.26}
\]
\[
Q = y^T y \geq 0 \tag{2.27}
\]
so that $A^T A$ is non-negative definite. In fact, if the columns of $A$ are linearly independent, it is not possible for $y$ to equal the order-$m$ null vector $0$, so that in this case $A^T A$ is positive definite. This is also called the full-rank case.

Secondly, many computer programs for solving the linear least-squares problem ignore the existence of special algorithms for the solution of linear equations having a symmetric, positive definite coefficient matrix. Above it has already been established that $A^T A$ is positive definite and symmetry is proved trivially. The special algorithms have advantages in efficiency and reliability over the methods for the general linear-equation problem.

Thirdly, in chapter 5 it will be shown that the formation of $A^T A$ can lead to loss of information. Techniques exist for the solution of the least-squares problem without recourse to the normal equations. When there is any question as to the true linear independence of the columns of $A$, these have the advantage that they permit the minimum-length least-squares solution to be computed.

It is worth noting that the linear-equation problem of equation (2.2) can be solved by treating it as a least-squares problem. Then for singular matrices $A$ there is still a least-squares solution $x$ which, if the system of equations is consistent, has a zero sum of squares $r^T r$. For small-computer users who do not regularly need solutions to linear equations or whose equations have coefficient matrices which are near-singular (ill conditioned is another way to say this), it is my opinion that a least-squares solution method which avoids the formation of $A^T A$ is useful as a general approach to the problems in both equations (2.2) and (2.14).

As for linear equations, linear least-squares problems are categorised by whether or not they can be stored in the main memory of the computing device at hand. Once again, the traditional terms dense and sparse will be used, though some problems having $m$ large and $n$ reasonably small will have very few zero entries in the matrix $A$.

**Example 2.3. Least squares**

It is believed that in the United States there exists a linear relationship between farm money income and the agricultural use of nitrogen, phosphate, potash and petroleum. A model is therefore formulated using, for simplicity, a linear form

\[
\text{(money income)} = x_1 + x_2 \text{(nitrogen)} + x_3 \text{(phosphate)} + x_4 \text{(potash)} + x_5 \text{(petroleum)}.
\]  

(2.28)

For this problem the data are supplied as index numbers ($1940 = 100$) to avoid difficulties associated with the units in which the variables are measured. By collecting the values for the dependent variable (money income) as a vector $b$ and the values for the other variables as the columns of a matrix $A$ including the constant unity which multiplies $x_1$, a problem

\[
A x = b
\]

(2.14)

is obtained. The data and solutions for this problem are given as table 3.1 and example 3.2.
Example 2.4. Surveying-data fitting

Consider the measurement of height differences by levelling (reading heights off a vertical pole using a levelled telescope). This enables the difference between the heights of two points $i$ and $j$ to be measured as

$$b_{ij} = h_i - h_j + r_{ij}$$  \hspace{1cm} (2.29)

where $r_{ij}$ is the error made in taking the measurement. If $m$ height differences are measured, the best set of heights $h$ is obtained as the solution to the least-squares problem

$$\text{minimise } r^T r$$  \hspace{1cm} (2.30)

where

$$r = b - Ah$$

and each row of $A$ has only two non-zero elements, 1 and -1, corresponding to the indices of the two points involved in the height-difference measurement. Sometimes the error is defined as the weighted residual

$$r_{ij} = [b_{ij} - (h_i - h_j)]d_{ij}$$

where $d_{ij}$ is the horizontal distance between the two points (that is, the measurement error increases with distance).

A special feature of this particular problem is that the solution is only determined to within a constant, $h_0$, because no origin for the height scale has been specified. In many instances, only relative heights are important, as in a study of subsidence of land. Nevertheless, the matrix $A$ is rank-deficient, so any method chosen to solve the problem as it has been presented should be capable of finding a least-squares solution despite the singularity (see example 19.2).

2.4. THE INVERSE AND GENERALISED INVERSE OF A MATRIX

An important concept is that of the inverse of a square matrix $A$. It is defined as that square matrix, labelled $A^{-1}$, such that

$$A^{-1}A = AA^{-1} = I_n$$  \hspace{1cm} (2.31)

where $I_n$ is the unit matrix of order $n$. The inverse exists only if $A$ has full rank. Algorithms exist which compute the inverse of a matrix explicitly, but these are of value only if the matrix inverse itself is useful. These algorithms should not be used, for instance, to solve equation (2.2) by means of the formal expression

$$x = A^{-1}b$$  \hspace{1cm} (2.32)

since this is inefficient. Furthermore, the inverse of a matrix $A$ can be computed by setting the right-hand side $b$ in equation (2.2) to the $n$ successive columns of the unit matrix $I_n$. Nonetheless, for positive definite symmetric matrices, this monograph presents a very compact algorithm for the inverse in §8.2.

When $A$ is rectangular, the concept of an inverse must be generalised. Corresponding to (2.32) consider solving equation (2.14) by means of a matrix $A^+$, yet to be defined, such that

$$x = A^+b.$$  \hspace{1cm} (2.33)
In other words, we have

\[ A^+ A = 1_n. \]

(2.34)

When \( A \) has only \( k \) linearly independent columns, it will be satisfactory if

\[
A^+ A = \begin{bmatrix}
1_k \\
0 \\
0 \\
\vdots \\
0 \\
(n-k)
\end{bmatrix}
\]

(2.35)

but in this case \( x \) is not defined uniquely since it can contain arbitrary components from the orthogonal complement of the space spanned by the columns of \( A \). That is, we have

\[ x = A^+ b + (1_n - A^+ A) g \]

(2.36)

where \( g \) is any vector of order \( n \).

The normal equations (2.22) must still be satisfied. Thus in the full-rank case, it is straightforward to identify

\[ A^+ = (A^T A)^{-1} A^T. \]

(2.37)

In the rank-deficient case, the normal equations (2.22) imply by substitution of (2.36) that

\[
A^T A x = A^T A A^+ b + (A^T A - A^T A A^+) g = A^T b.
\]

(2.38)

If

\[ A^T A A^+ = A^T \]

(2.39)

then equation (2.38) is obviously true. By requiring \( A^+ \) to satisfy

\[ A A^+ A = A \]

(2.40)

and

\[ (A A^+)^T = AA^+ \]

(2.41)

d this can indeed be made to happen. The proposed solution (2.36) is therefore a least-squares solution under the conditions (2.40) and (2.41) on \( A^+ \). In order that (2.36) gives the minimum-length least-squares solution, it is necessary that \( x^T x \) be minimal also. But from equation (2.36) we find

\[
x^T x = b^T (A^+)^T A^+ b + g^T (1 - A^+ A)(1 - A^+ A) g + 2 g^T (1 - A^+ A)^T A^+ b \]

(2.42)

which can be seen to have a minimum at

\[ g = 0 \]

(2.43)

if

\[ (1 - A^+ A)^T \]
is the annihilator of $A^+ b$, thus ensuring that the two contributions (that is, from $b$ and $g$) to $x^T x$ are orthogonal. This requirement imposes on $A^+$ the further conditions

$$A^+ A A^+ = A^+$$  \hspace{2cm} (2.44)

$$ (A^+ A)^T = A^+ A. $$  \hspace{2cm} (2.45)

The four conditions (2.40), (2.41), (2.44) and (2.45) were proposed by Penrose (1955). The conditions are not, however, the route by which $A^+$ is computed.

Here attention has been focused on one generalised inverse, called the Moore-Penrose inverse. It is possible to relax some of the four conditions and arrive at other types of generalised inverse. However, these will require other conditions to be applied if they are to be specified uniquely. For instance, it is possible to consider any matrix which satisfies (2.40) and (2.41) as a generalised inverse of $A$ since it provides, via (2.33), a least-squares solution to equation (2.14). However, in the rank-deficient case, (2.36) allows arbitrary components from the null space of $A$ to be added to this least-squares solution, so that the two-condition generalised inverse is specified incompletely.

Over the years a number of methods have been suggested to calculate ‘generalised inverses’. Having encountered some examples of dubious design, coding or applications of such methods, I strongly recommend testing computed generalised inverse matrices to ascertain the extent to which conditions (2.40), (2.41), (2.44) and (2.45) are satisfied (Nash and Wang 1986).

### 2.5. DECOMPOSITIONS OF A MATRIX

In order to carry out computations with matrices, it is common to decompose them in some way to simplify and speed up the calculations. For a real $m$ by $n$ matrix $A$, the QR decomposition is particularly useful. This equates the matrix $A$ with the product of an orthogonal matrix $Q$ and a right- or upper-triangular matrix $R$, that is

$$A = QR$$  \hspace{2cm} (2.46)

where $Q$ is $m$ by $m$ and

$$Q^T Q = QQ^T = I_m$$  \hspace{2cm} (2.47)

and $R$ is $m$ by $n$ with all elements

$$R_{ij} = 0 \hspace{1cm} \text{for} \hspace{0.5cm} i > j.$$  \hspace{2cm} (2.48)

The QR decomposition leads to the singular-value decomposition of the matrix $A$ if the matrix $R$ is identified with the product of a diagonal matrix $S$ and an orthogonal matrix $V^T$, that is

$$R = SV^T$$  \hspace{2cm} (2.49)

where the $m$ by $n$ matrix $S$ is such that

$$S_{ij} = 0 \hspace{1cm} \text{for} \hspace{0.5cm} i \neq j$$  \hspace{2cm} (2.50)

and $V$, $n$ by $n$, is such that

$$V^T V = VV^T = I_n.$$  \hspace{2cm} (2.51)
Note that the zeros below the diagonal in both \( R \) and \( S \) imply that, apart from orthogonality conditions imposed by (2.47), the elements of columns \((n + 1), (n + 2), \ldots, m\) of \( Q \) are arbitrary. In fact, they are not needed in most calculations, so will be dropped, leaving the \( m \) by \( n \) matrix \( U \), where

\[
U^T U = 1_n.
\] (2.52)

Note that it is no longer possible to make any statement regarding \( UU^T \). Omitting rows \((n + 1)\) to \( m\) of both \( R \) and \( S \) allows the decompositions to be written as

\[
A = UR = USV^T
\] (2.53)

where \( A \) is \( m \) by \( n \), \( U \) is \( m \) by \( n \) and \( U^T U = 1_m \), \( R \) is \( n \) by \( n \) and upper-triangular, \( S \) is \( n \) by \( n \) and diagonal, and \( V \) is \( n \) by \( n \) and orthogonal. In the singular-value decomposition the diagonal elements of \( S \) are chosen to be non-negative.

Both the \( QR \) and singular-value decompositions can also be applied to square matrices. In addition, an \( n \) by \( n \) matrix \( A \) can be decomposed into a product of a lower- and an upper-triangular matrix, thus

\[
A = LR.
\] (2.54)

In the literature this is also known as the \( LU \) decomposition from the use of ‘\( U \)’ for ‘upper triangular’. Here another mnemonic, ‘\( U \)’ for ‘unitary’ has been employed. If the matrix \( A \) is symmetric and positive definite, the decomposition

\[
A = LL^T
\] (2.55)

is possible and is referred to as the Choleski decomposition.

A scaled form of this decomposition with unit diagonal elements for \( L \) can be written

\[
A = LDL^T
\]

where \( D \) is a diagonal matrix.

To underline the importance of decompositions, it can be shown by direct substitution that if

\[
A = USV^T
\] (2.53)

then the matrix

\[
A^+ = VS^+ U^T
\] (2.56)

where

\[
S^+_{ii} = \begin{cases} 
1/S_{ii} & \text{for } S_{ii} \neq 0 \\
0 & \text{for } S_{ii} = 0 
\end{cases}
\] (2.57)

satisfies the four conditions (2.40), (2.41), (2.44) and (2.45), that is

\[
AA^+ A = USV^T VS^+ U^T USV^T
\]
\[
= USS^+ SV^T
\]
\[
= USV^T = A
\] (2.59)
Compact numerical methods for computers

\[ A^+ A A^+ = V S^+ U^T U S V^T V S^+ U^T \]
\[ = V S^+ S S^+ U^T = V S^+ U^T = A^+ \]  \hfill (2.60)

and

\[ (A^+ A)^T = (V S^+ U^T U S V^T)^T = (V S^+ S V^T)^T \]
\[ = V S^+ S V^T = A^+ A. \]  \hfill (2.61)

Several of the above relationships depend on the diagonal nature of \( S \) and \( S^+ \) and on the fact that diagonal matrices commute under multiplication.

### 2.6. THE MATRIX EIGENVALUE PROBLEM

An eigenvalue \( e \) and eigenvector \( x \) of an \( n \) by \( n \) matrix \( A \), real or complex, are respectively a scalar and vector which together satisfy the equation

\[ A x = e x. \]  \hfill (2.62)

There will be up to \( n \) eigensolutions \( (e, x) \) for any matrix (Wilkinson 1965) and finding them for various types of matrices has given rise to a rich literature. In many cases, solutions to the generalised eigenproblem

\[ A x = e B x \]  \hfill (2.63)

are wanted, where \( B \) is another \( n \) by \( n \) matrix. For matrices which are of a size that the computer can accommodate, it is usual to transform (2.63) into type (2.62) if this is possible. For large matrices, an attempt is usually made to solve (2.63) itself for one or more eigensolutions. In all the cases where the author has encountered equation (2.63) with large matrices, \( A \) and \( B \) have fortunately been symmetric, which provides several convenient simplifications, both theoretical and computational.

**Example 2.5. Illustration of the matrix eigenvalue problem**

In quantum mechanics, the use of the variation method to determine approximate energy states of physical systems gives rise to matrix eigenvalue problems if the trial functions used are linear combinations of some basis functions (see, for instance, Pauling and Wilson 1935, p 180ff).

If the trial function is \( F \), and the energy of the physical system in question is described by the Hamiltonian operator \( H \), then the variation principle seeks stationary values of the energy functional

\[ C = \frac{(F, HF)}{(E, F)} \]  \hfill (2.64)

subject to the normalisation condition

\[ (F, F) = 1 \]  \hfill (2.65)

where the symbol \(( , )\) represents an inner product between the elements separated by the comma within the parentheses. This is usually an integral over all
the dimensions of the system. If a linear combination of some functions $f_i, j = 1, 2, \ldots, n$, is used for $F$, that is

$$ F = \sum_{i=1}^{n} x_i f_i $$  \hspace{1cm} (2.66)

then the variation method gives rise to the eigenvalue problem

$$ Ax = \lambda Bx $$  \hspace{1cm} (2.63)

with

$$ A_{ij} = (f_i, Hf_j) $$  \hspace{1cm} (2.67)

and

$$ B_{ij} = (f_i, f_j) $$  \hspace{1cm} (2.68)

It is obvious that if $B$ is a unit matrix, that is, if

$$ (f_i, f_j) = \delta_{ij} = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases} $$  \hspace{1cm} (2.69)

a problem of type (2.56) arises. A specific example of such a problem is equation (11.1).