Inversion of Symmetric Coefficient Matrix of Positive-Definite Quadratic Form

1. Introduction. This paper presents a method of inverting a positive-definite symmetric matrix which has been coded by the author for the IBM 704 digital computer at Los Alamos. The code is now running so that timing and accuracy descriptions are included.

For brevity we shall call a real n by n symmetric matrix a *positive-definite* if the quadratic form $x^T a x$ is positive-definite, x real. We shall invert a by constructing an upper triangular matrix H subject to the conditions $|H| = 1$ and $H^T a H = d^{-1}$, where d^{-1} is *diagonal*. It then follows that $a^{-1} = HdH^T$, $|a| = |d^{-1}|$, and $x^T a x = \xi^T d^{-1} \xi$, where $x = H\xi$. This gives the

Criterion: The necessary and sufficient conditions that $x^T a x$ be positive-definite are that *all* n terms of the diagonal matrix d^{-1}, which the code forms for printing, be *positive*.

It should be remarked that, while the positive-definite condition ensures no attempted machine division by zero, there exists a large class of *indefinite* matrices which can be inverted by our method.

2. The choice of H. The method of constructing H so as to diagonalize a according to $H^T a H = d^{-1}$ will be described for the case $n = 4$, the generalization to any n being immediate. We employ Gauss elimination to define

$$h_{1j} = -a_{1j}/a_{11}, \quad a^{(2)}_{jk} = (a_{11}a_{jk} - a_{1k}a_{j1})/a_{11}, \quad j, k = 2, 3, 4, \tag{2.1}$$

the division being justified by our positive-definite hypothesis on a, and form

$$H_1 = \begin{Vmatrix} 1 & h_{12} & h_{13} & h_{14} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{Vmatrix} \qquad a^{(2)} = H_1{}^T a H_1 = \begin{Vmatrix} a_{11} & 0 & 0 & 0 \\ 0 & a^{(2)}_{22} & a^{(2)}_{23} & a^{(2)}_{24} \\ 0 & a^{(2)}_{32} & a^{(2)}_{33} & a^{(2)}_{34} \\ 0 & a^{(2)}_{42} & a^{(2)}_{43} & a^{(2)}_{44} \end{Vmatrix}$$

for which $|H_1| = 1$. Since $x^T a x = y^T a^{(2)} y$, where $x = H_1 y$, the right-hand form is likewise positive-definite.

Continuing with the diagonalization by defining

$$h_{2j} = -a^{(2)}_{2j}/a^{(2)}_{22}, \quad a^{(3)}_{jk} = (a^{(2)}_{22}a^{(2)}_{jk} - a^{(2)}_{2k}a^{(2)}_{j2})/a^{(2)}_{22},$$

for $j, k = 3, 4$, and so on, we finally obtain

$$H = H_1H_2H_3 = \begin{Vmatrix} 1 & h_{12} & h_{13} & h_{14} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{Vmatrix} \cdot \begin{Vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & h_{23} & h_{24} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{Vmatrix} \cdot \begin{Vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & h_{34} \\ 0 & 0 & 0 & 1 \end{Vmatrix}$$

$$(2.2)\quad H = \begin{Vmatrix} 1 & h_{12}\cdot 1 & h_{12}\cdot h_{23} + h_{13}\cdot 1 & h_{12}\cdot(h_{23}h_{34} + h_{24}) + h_{13}\cdot h_{34} + h_{14}\cdot 1 \\ 0 & 1 & h_{23}\cdot 1 & h_{23}\cdot h_{34} + h_{24}\cdot 1 \\ 0 & 0 & 1 & h_{34}\cdot 1 \\ 0 & 0 & 0 & 1 \end{Vmatrix}$$

d^{-1} = diag. matrix of elements $(a_{11}, a_{22}^{(2)}, a_{33}^{(3)}, a_{44}^{(4)})$.

An inspection of the inner product composition of the columns of H shows how this matrix generalizes for arbitrary n. H is generated from the triangular array of the h's by first forming the elements of the last column, proceeding from the bottom up, then passing to the adjacent column, etc.

It may be helpful to point out the relation which the method of this section bears to a familiar procedure for reducing a positive-definite quadratic form x^Tax to a sum of squares. Namely, one employs the algorithm of Gauss-Banachiewicz (see Bodewig [2], p. 105, or Turing [3], p. 289), to factor the coefficient matrix according to the form

$$(2.3)\qquad a = (I + U)^T D(I + U),$$

where $(I + U)$ is a uniquely determined upper triangular matrix with diagonal elements unity and D is a diagonal matrix of positive terms. Comparison with our factorization $H^TaH = d^{-1}$ identifies H with $(I + U)^{-1}$ and d^{-1} with D. Then the transformation $H^{-1}x = \xi$ gives the desired reduction to a sum of squares, $x^Tax = \xi^TD\xi$. Thus, if we had chosen to employ the algorithm (2.3) to form $H^{-1} = (I + U)$, we would then have been confronted, as was Banachiewicz [4], p. 398, with the *additional* task of inverting the transformation $H^{-1}x = \xi$ to express it in the solved form $x = H\xi$ necessary for our matrix inversion scheme.

It can be shown [2], p. 91, that the elements d_i^{-1} of the diagonal matrix d^{-1} are given by the quotients of successive principal minors of a,

$$d_1^{-1} = |a_{11}|, \quad d_2^{-1} = |a_{11}a_{22}|/|a_{11}|, \quad d_3^{-1} = |a_{11}a_{22}a_{33}|/|a_{11}a_{22}|, \cdots.$$

3. The inverse matrix a^{-1}. The matrix H of (2.2) is of the upper triangular form

$$(3.1)\qquad H = \begin{Vmatrix} \eta_{11} & \eta_{12} & \eta_{13} & \eta_{14} \\ 0 & \eta_{22} & \eta_{23} & \eta_{24} \\ 0 & 0 & \eta_{33} & \eta_{34} \\ 0 & 0 & 0 & \eta_{44} \end{Vmatrix}, \quad \eta_{11} = \eta_{22} = \eta_{33} = \eta_{44} = 1,$$

so that the distinct elements of the symmetric inverse matrix $a^{-1} = HdH^T$ are given by

$$(3.2)\qquad a^{-1} = \begin{Vmatrix} \eta_{1r}d_r\eta_{1r} & \eta_{1s}d_s\eta_{2s} & \eta_{1t}d_t\eta_{3t} & \eta_{14}d_4\eta_{44} \\ & \eta_{2s}d_s\eta_{2s} & \eta_{2t}d_t\eta_{3t} & \eta_{24}d_4\eta_{44} \\ & & \eta_{3t}d_t\eta_{3t} & \eta_{34}d_4\eta_{44} \\ & & & \eta_{44}d_4\eta_{44} \end{Vmatrix},$$

$$r = 1, 2, 3, 4 \quad s = 2, 3, 4 \quad t = 3, 4.$$

In writing (3.2) the convention of summing the repeated indices r, s, t, over their respective ranges has been observed.

4. Storage arrangement for coding. The input array is illustrated for $n = 3$:

$$(4.1)\qquad \left\|\begin{array}{llll} G_0(a_{11}) & G_1(a_{12}) & G_2(a_{13}) & \\ & G_3(a_{22}) & G_4(a_{23}) & \\ & & G_5(a_{33}) & \\ G_6(b_1) & G_7(b_2) & G_8(b_3) & \\ G_9(\ ) & G_{10}(\ ) & G_{11}(\ ) & G_{12}(\ ) \end{array}\right\| \begin{array}{l} \\ (\tfrac{1}{2}n(n+1) \text{ words}) \\ \\ (n \text{ words}) \\ (n+1 \text{ words reserved for output}) \end{array}$$

where the G's represent $\frac{1}{2}n(n+1) + 2n + 1$ consecutive storage locations The b's are the right-hand sides of any linear system $ax = b$, while the contents of the last $n + 1$ words at the time of input are immaterial.

A code of 263 orders has been written which requires a fixed block of 13 and a variable block of $2n$ erasable words in the consecutive locations $D_0, \cdots, D_{2n-1}$. Since the computations of the type (2.1) involve only two rows at a time, these two rows can be placed in the D-block for ease in addressing and the results of the computation stored appropriately in the G-block, destroying data no longer required.

Proceeding in this way, the triangular array (3.1) for H is generated in the first $\frac{1}{2}n(n+1)$ locations of the G-block, the former non-unity diagonal terms which define d^{-1} having been stored for printing and further use in the first n locations of the last line of (4.1).

Inspection of the matrix (3.2) defining a^{-1} reveals that again only one pair of rows at a time is involved in forming a^{-1}. Once more the D-block serves to store this pair for easy addressing and clearance in storing a^{-1} in the first $\frac{1}{2}n(n+1)$ locations of the G-block, data no longer needed being destroyed.

The result is the output array, illustrated for $n = 3$, available for printing or further computing:

$$(4.2)\qquad \left\|\begin{array}{llll} G_0(\alpha_{11}) & G_1(\alpha_{12}) & G_2(\alpha_{13}) & \\ & G_3(\alpha_{22}) & G_4(\alpha_{23}) & \\ & & G_5(\alpha_{33}) & \\ G_6(x_1) & G_7(x_2) & G_8(x_3) & \\ G_9(d_1^{-1}) & G_{10}(d_2^{-1}) & G_{11}(d_3^{-1}) & G_{12}(|a|) \end{array}\right\| \begin{array}{l} \\ (\text{inverse matrix } a^{-1}) \\ \\ (\text{solution of } ax = b) \\ (\text{pos.-def. criterion and det.}). \end{array}$$

5. Computing with a^{-1}. It is anticipated that the elements α_{ij} of a^{-1} may be subsequently employed to form inner products of the type $\sum_{r=1}^{n} \alpha_{jr}y_r$, where y is an arbitrary vector. To facilitate the addressing of this sum, one may load the jth row of a^{-1} into $D_0, \cdots, D_{n-1}$ and the y's into $D_n, \cdots, D_{2n-1}$. The inversion code contains a block of 20 consecutive orders which assemble the jth row of a^{-1} from the triangular output array of (4.2) and store in $D_0, \cdots, D_{n-1}$. After the inversion these 20 orders may be retained in memory for this purpose.

6. Accuracy, timing and limitations on n. The Hilbert matrix H_n, $h_{ij} = (i + j - 1)^{-1}$, $i, j = 1, 2, \cdots, n$, being somewhat a champion among ill-conditioned matrices, imposes a severe test upon the accuracy of any method of inversion. The true inverse of H_n has been tabulated by I. R. Savage and E. Lukacs

([1], p. 105), through $n = 10$. A floating-point nine-digit approximation to H_n, which took account of all digits in the IBM 704, was loaded and the computed inverse of this approximate H_n was compared with the true inverse of H_n. The following table of comparisons was obtained:

n	$\lvert H_n \rvert$	*Agreement with true* H_n^{-1}
4	$1.6534(10^{-7})$	5 digits
5	$3.75(10^{-12})$	3 digits
6	$5.3(10^{-18})$	2 digits
7	$4(10^{-25})$	max. discrepancy 20%
8	$2(10^{-33}$	max. discrepancy 50%
9	10^{-42}	inversion failed.

J. Todd reported in 1954 ([1], p. 113), that SEAC failed to invert H_6 when using a single precision fixed point routine.

A less stringent test matrix is Γ_n, $\gamma_{ij} = \gamma_{ji} = -i(n + 1 - j)/(n + 1)$, $i \leq j$, whose inverse ([1], p. 112), is $c_{ii} = -2$, $c_{ij} = 1$, $|i - j| = 1$, $c_{ij} = 0$, $|i - j| > 1$, $i, j = 1, 2, \cdots, n$. A floating-point, eight-digit approximation to Γ_n was loaded and the computed inverse of this approximate Γ_n was compared with the true Γ_n^{-1}. The following two runs were made:

n	$\lvert \Gamma_n \rvert$	*Agreement with true* Γ_n^{-1}	*Computing time*
49	−0.020000	6 digits	1 min 33 sec
115	−0.00862	5 digits	19 min 30 sec.

The computing time for inversion seems to be proportional to n^3. Comparison with SEAC's time of about three hours to invert Γ_{49}, as reported by J. Todd in 1954 [1], p. 113, gives an idea of the progress being made in the design of high-speed computers.

For a machine with 2^q words of core storage, the inequality which limits the size of n is $\frac{1}{2}n(n + 9) + 1 \leq 2^q - 295$, where 295 is the storage required by the Los Alamos print program. This gives the table:

2^q	*Max. n*	*Approx. running time*
4,096	82	7 min
8,192	121	23 min
32,768	250	200 min

THOMAS C. DOYLE

University of California
Los Alamos Scientific Laboratory

Work performed under the auspices of the U. S. Atomic Energy Commission.

1. NBS, Applied Mathematics Series, No. 39, *Contributions to the Solution of Systems of Linear Equations and the Determination of Eigenvalues*, U. S. Gov. Printing Office, Washington, D. C., 1954.
2. E. BODEWIG, *Matrix Calculus*, Interscience Publishers, Inc., New York, 1956.
3. A. M. TURING, "Rounding-off errors in matrix processes," *Quart. Jn. Mech. Appl. Math.*, v. 1, 1948, p. 287–308.
4. T. BANACHIEWICZ, "Méthode de résolution numérique des équations linéaires, du calcul des déterminants et des inverses, et de réduction des formes quadratiques," Internat. Acad. Polonaise Sci. Lett., *Bull.*, A, 1938, p. 393–404.