On the calculation of a generalized inverse matrix in a domain

Gennadi Malaschonok and Ihor Tchaikovsky

Abstract. We examined the well-known algorithms for calculating the generalized inverse matrix and proposed another algorithm that avoids the accumulation of errors when rounding numbers. Therefore, it is attractive for large matrices.

Introduction

The Moore-Penrose generalized inverse matrix [1]-[3] has many applications in physics, computer science and other fields. A matrix A^+ is called the generalized inverse of the matrix A if the following 4 equalities hold:

$$A^+ = A^+AA^+,\ A = AA^+A,\ (A^+A)^T = A^+A,\ (AA^+)^T = AA^+. \qquad (1)$$

Let a matrix $A \in F^{n\times m}$ be decomposed as follows: $A = B \cdot C$, $B \in F^{n\times k}$, $C \in F^{k\times m}$, $rank(A) = rank(B) = rank(C) = k$.

It is easy to check that matrix

$$A^+ = C^T(CC^T)^{-1}(B^TB)^{-1}B^T, \qquad (2)$$

is the generalized inverse matrix for the matrix A. This idea was first expressed by Vera Kublanovskaya in 1965 [4].

In the case of complex numbers we have to use in (1) and (2) the operation of conjugation:

$$A^+ = C^*(CC^*)^{-1}(B^*B)^{-1}B^*.$$

There are many possibilities to obtain the decomposition (2). For example, you can used the QR decomposition or LU decomposition.

1. SVD algorithm

We can evaluate the complexity of Kublanovskaya algorithm. In total, 5 matrix multiplications, two matrix inversions, and one more decomposition are required. The total number of operations does not exceed $\sim 8max(n,m)^3$.

Unfortunately, classical Gaussian inversion and LU decomposition are not numerically stable for large matrices due to the accumulation of rounding errors.

Today the most popular known computational method used singular value decomposition. This method consists of two stages. In the first stage, due to the Householder reflections (or Givens rotations)[5-7], an initial matrix is reduced to the upper bidiagonal form (the Golub-Kahan bidiagonalization algorithm).

The second stage is known as the Golub-Reinsch algorithm [8]. This is an iterative procedure which with the help of the Givens rotations generates a sequence of bidiagonal matrices converging to a diagonal form. This allows to obtain an iterative approximation to the singular value decomposition of the bidiagonal matrix.

In the paper [9] was presented a new finite recursive numerical algorithm for obtaining explicit rational expressions for the generalized inverse of bidigonal matrix. This rational algorithm has less number of operations. But the problem of stability is the main problem here.

2. New approach

We propose a different approach for calculating the generalized inverse matrix. It guarantees the stability of rational computing. Our approach is based on LDU matrix decomposition like [10],[11]. As in the LU decomposition, we can use equality (2). The main difference is that at each step we operate with some elements of the commutative ring, which are the minors of the original matrix. Therefore, all operations are performed exactly.

Moreover, we can refuse to use the expression (2) if we use the algorithm, which, in addition to the factors L, D, U, calculates their inverse matrices M and W and full rank matrix $\widehat{D}$: $L\widehat{D}M = \mathbf{I}$, $W\widehat{D}U = \mathbf{I}$. We denote by D the weighted truncated permutation matrix. The generalized inverse matrix D^+ can be obtain by transposition of matrix D and inverse each of the non-zero element of D^T. Each of these matrices has rank the same as matrix A.

With these denotes we can write generalized inverse of matrix A as follows:

$$A^+ = U^{-1}D^+L^{-1} = WDM,\ AA^+A = LDD^+DU = A,$$

$$A^+AA^+ = U^{-1}D^+L^{-1}LDUU^{-1}D^+L^{-1} = U^{-1}D^+L^{-1} = A^+$$

$$AA^+ = LDD^+L^{-1} = LI_DL^{-1} = I_D$$

I_D - is a diagonal matrix with unit elements which stand at the non-zero rows of matrix D.

$$A^+A = U^{-1}D^+DU = U^{-1}J_DU = J_D$$

J_D - is a diagonal matrix with unit elements which stand at the non-zero columns of matrix D.

3. Algorithm of dichotomious LDU decomposition

The algorithm is presented without proof. The proof will be published elsewhere.

Let matrix A has size $n \times n$, $n = 2^p$.

$$(L, D, U, M, \widehat{D}, W, \alpha_r) = \mathbf{LDU}(A, \alpha).$$

(1) If $(A = 0)$ then $\{\ \alpha_r = \alpha;\quad D = I = J = 0;$
$L = U = \bar{I} = \bar{J} = \bar{D} = I;\quad M = W = \widehat{D} = \alpha I;\ \}$

(2) If $(n = 1\ \&\ A = [a]\ \&\ a \neq 0)$ then
$\{\ \alpha_r = a;\ L = U = M = W = [a];\ D = [(\alpha * a)^{-1}];\ \widehat{D} = [a^{-2}];$
$J = I = [1];\ \bar{I} = \bar{J} = \bar{D} = [0];\ \}$

(3) If $(n \geq 2\ \&\ A \neq 0)$ then

$\{ \qquad A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}.$

(3.1)

$$(L_{11}, D_{11}, U_{11}, M_{11}, d_{11}, W_{11}, \alpha_k) = \mathbf{LDU}(A_{11}, \alpha),$$

$A^0_{12} = M_{11} * A_{12};\ A^1_{12} = a_k * \widehat{D}_{11} * A^0_{12};$
$A^2_{12} = \bar{D}_{11} * A^0_{12}/\alpha;\ A^0_{21} = A_{21} * W_{11};$
$A^1_{21} = a_k * A^0_{21} * \widehat{D}_{11};\ A^2_{21} = A^0_{21} * \bar{D}_{11}/\alpha;$

(3.2)

$$(L_{21}, D_{21}, U_{21}, M_{21}, d_{21}, W_{21}, \alpha_l) = \mathbf{LDU}(A^2_{21}, a_k),$$

(3.3)

$$(L_{12}, D_{12}, U_{12}, M_{12}, d_{12}, W_{12}, \alpha_m) = \mathbf{LDU}(A^2_{12}, a_k),$$

$\lambda = \frac{a_l}{a_k};\ a_s = \lambda * a_m;\ A^0_{22} = A^1_{21} * D^+_{11} * A^1_{12};$
$A^1_{22} = (\alpha a^2_k * A_{22} - A^0_{22})/(\alpha a_k);$
$A^2_{22} = \bar{D}_{21} * M_{21} * A^1_{22} * W_{12} * \bar{D}_{12};\ A^3_{22} = A^2_{22}/(a^2_k \alpha);$

(3.4)

$$(L_{22}, D_{22}, U_{22}, M_{22}, d_{22}, W_{22}, \alpha_r) = \mathbf{LDU}(A^3_{22}, a_s),$$

$J^\lambda_{12} = \lambda * J_{12} + \bar{J}_{12};\quad I^\lambda_{12} = \lambda I_{12} + \bar{I}_{12};$
$L_{\widetilde{12}} = L_{12} * I^\lambda_{12};\quad U_{\widetilde{12}} = J^\lambda_{12} * U_{12};$
$U_2 = J_{11} * M_{11} * A_{12}/a_k + J_{21} * M_{21} * A^1_{22}/(a_l * \alpha);$
$L_3 = A_{21} * W_{11} * I_{11}/a_k + \bar{D}_{21} * M_{21} * A^1_{22} * W_{12} * I_{12}/(a_m * a_k * \alpha);$

$$D = \begin{pmatrix} D_{11} & \lambda^{-2} D_{12} \\ D_{21} & D_{22} \end{pmatrix},\quad \widehat{D} = \alpha(\alpha_r)^{-1} D + (\alpha_r)^{-1} \bar{D},$$

$$L = \begin{pmatrix} L_{11} L_{\widetilde{12}} & 0 \\ L_3 & L_{21} L_{22} \end{pmatrix},\quad U = \begin{pmatrix} U_{21} U_{11} & U_2 \\ 0 & U_{22} U_{\widetilde{12}} \end{pmatrix},$$

$$M = \widehat{D}^{-1}\begin{pmatrix} I_{12}^{\lambda^{-1}}\widehat{D}_{12}M_{12}\widehat{D}_{11}M_{11} & 0 \\ -\widehat{D}_{22}M_{22}\widehat{D}_{21}M_{21}L_3 I_{12}^{\lambda^{-1}}\widehat{D}_{12}M_{12}\widehat{D}_{11}M_{11} & \widehat{D}_{22}M_{22}\widehat{D}_{21}M_{21} \end{pmatrix},$$

$$W = \begin{pmatrix} W_{11}\widehat{D}_{11}W_{21}\widehat{D}_{21} & -W_{11}\widehat{D}_{11}W_{21}U_2W_{12}\widehat{D}_{12}J_{12}^{\lambda^{-1}}W_{22}\widehat{D}_{22} \\ 0 & W_{12}\widehat{D}_{12}J_{12}^{\lambda^{-1}}W_{22}\widehat{D}_{22} \end{pmatrix}\widehat{D}^{-1}.$$

}

References

[1] B. Penrose A generalized inverse for matrices. Proe. Cambridge Philos. Soc, 51 (1955).

[2] A. Ben-Israel and T. N. E. Greville, Generalized Inverses. Theory and Applications, 2nd ed. Springer, New York, (2003).

[3] A. S. Hauscholder . The theory of matrices in numerical analysis. Boston, Ginn and Co., (1963).

[4] V.N.Kublanovskaya. Evaluation of a generalized inverse matrix and projector. USSR Computational Mathematics and Mathematical Physics, Volume 6, Issue 2, 179-188, (1966)

[5] G. H. Golub and C. Reinsch, Singular Value Decomposition and Least Squares Solutions. Numer. Math. 14, 403–420, (1970).

[6] G. H. Golub and Ch. F. van Loan. Matrix Computations, 3rd ed. The John Hopkins University Press, (1996).

[7] J. W. Lewis, Invertion of tridiagonal matrices. Numer.Math., 38, 333–345 (1982).

[8] G. H. Golub and W. Kahan., Calculating the Singular Values and Pseudo-Inverse of a Matrix. SIAM J. Num. Anal., 2, 205–224. (1965)

[9] Yu. Hakopian. Computing the Moor-Penrose inverse for bidiagonal matrices.Mohyla Mathematical Journal. National University of Kyiv-Mohyla Academy. V.2, 11-23, (2019)

[10] G. Malashonok, Generalized Bruhat decomposition in commutative domains. Computer Algebra in Scientific Computing, CASC'2013, LNCS 8136, Springer, Heidelberg, 2013, 231–242, (2013)

[11] G. Malashonok, A. Scherbinin. Triangular Decomposition of Matrices in a Domain, Computer Algebra in Scientific Computing, LNCS 9301, Springer, Switzerland, 2015, 290–304, (2015)

Gennadi Malaschonok
National University of Kyiv-Mohyla Academy
Kiev, Ukraine
e-mail: `malaschonok@gmail.com`

Ihor Tchaikovsky
Lviv Polytechnic National University
Institute of Telecommunications, Radioelectronics and Electronic Engineering
Lviv, Ukraine
e-mail: `ihortch@yahoo.com`