ISTANBUL UNIVERSITY –
JOURNAL OF ELECTRICAL & ELECTRONICS ENGINEERING

YEAR : 2006 (113- 115)
VOLUME : 6
NUMBER : 1

A SHORT NOTE ON THE APPLICATION OF CHOLESKY MATRIX FACTORISATION USING MATLAB

Ali Ümit KESKİN[1] **İmran GÖKER[2]**

[1]Yeditepe University Department of Biomedical Engineering
34755 Kayışdagı Istanbul Turkey
[2]Yeditepe Universitesity School of Applied Sciences
34755 Kayışdagı Istanbul Turkey

[1]E-mail: auk@yeditepe.edu.tr [2]E-mail: goker@yeditepe.edu.tr

ABSTRACT

One must first make sure that a given matrix subject to factorization is symmetric, and then use it in Cholesky algorithm, in MATLAB. This may cost machine time to check for symmetry, however, saves much more of it by preventing errors due to inherent structure of the built-in algorithm.

Keywords: *Matrix Factorisation, Cholesky Algorithm, MATLAB*

1. INTRODUCTION

Nowadays, mathematical software packages have become indispensible tools in signal processing. In most cases, one assumes that no problems exist in their built-in algorithms.
Cholesky factorisation (CF) is a useful method in the analysis and simulation of signals and systems. CF built-in algorithm in MATLAB Software Package uses LAPACK subroutine DPOTRF for real matrices.[1].In fact, MATLAB has an extensive set of commands and application packages (such as DSP block set) for the implementation of CF.

2. CHOLESKY FACTORIZATION

Positive definite matrices are of both theoretical and computational importance in a wide variety of applications. They are used, for example, in optimization algorithms and in the construction of various linear regression models, design of robust adaptive control systems, and non-parametric factor corellated Monte-Carlo simulation.

An (n x n) complex matrix **A** is called positive definite if R [**x*Ax**]>0 for all nonzero complex vectors **x**, where **x*** denotes the conjugate transpose of the vector **x**. In the case of a real matrix **A**, equation R [**x*Ax**]>0 reduces to **x*Ax**>0, where $\mathbf{x}^T$ denotes the transpose. Some other properties of positive definite matrices are briefly given in the appendix section.

Cholesky factorisation applies to any positive definite symmetric matrix **A**, producing the

Received Date : 08.04.2004
Accepted Date: 10.12.2005

lower triangular matrix **L** so that **A=LL**^{T}. This can be stated in more general form: If **A** is a positive definite real symmetric matrix, it can be factorized in Cholesky form as **A=LL**^{T} or **A=UU**^{T} where **L** and **U** are lower and upper real triangular matrices, respectively, and T stands for transpose operation [2-3].

A linear system of equations with a positive definite matrix can be efficiently solved using the so-called Cholesky decomposition. Unlike general matrices, a positive definite matrix has exactly one matrix square root.

If the matrix **A** is symmetric and positive definite, then, denoting

$$\mathbf{A}=\begin{bmatrix} a & \mathbf{v}^T \\ \mathbf{v} & \mathbf{B} \end{bmatrix}$$

the matrix **A** can be expressed by

$$\mathbf{A}=\begin{bmatrix} b & 0 \\ \mathbf{v}/b & \mathbf{I}_{n-1} \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & \mathbf{B}-\mathbf{v}\mathbf{v}^T/a \end{bmatrix}\begin{bmatrix} b & \mathbf{v}^T/b \\ 0 & \mathbf{I}_{n-1} \end{bmatrix}$$

with $b=\sqrt{a}$.

If $\mathbf{B}-\mathbf{v}\mathbf{v}^T/a=\mathbf{G}_1\mathbf{G}_1^T$, then

A=GG^T,

where $\mathbf{G}=\begin{bmatrix} b & 0 \\ \mathbf{v}/b & \mathbf{G}_1 \end{bmatrix}$.

Thus, a recursive computation of Cholesky factorization is possible.

3. MATLAB COMPUTATION

Above given recursive equations show that the manual matrix factorisation of even smaller matrices involves tedious and a time consuming work. This work is greatly simplified using MATLAB which has an extensive set of commands and application packages (such as DSP block set) for the implementation of CF. In fact, this operation consists of a single command: **C**=chol(**A**) which yields the CF of a matrix **A.** Note, however that the matrix **A** must posess certain properties**.**

For example, using an ordinary square test matrix **A** which is neither positive definite nor symmetric, and applying **C**=chol(**A**) in MATLAB, an error message appears as "Error using chol; matrix must be positive definite". However, nothing is said about the symmetry of **A** .

Example: Let

$$\mathbf{A}=\begin{bmatrix} 6 & 3 & 5 & 4 \\ 2 & 5 & 3 & 4 \\ 1 & 2 & 4 & 3 \\ 1 & 2 & 1 & 7 \end{bmatrix} \qquad (1)$$

An attempt to factorize **A** by **U**=chol(**A**) will result in an error message as described above.

If the value of $A_{13}=5$ term is replaced with $A_{13}=4$, the algorithm "computes" the CF of **A** as

$$\mathbf{U}=\begin{bmatrix} 2.4495 & 1.2247 & 1.6330 & 1.6330 \\ 0 & 1.8708 & 1.5345 & 1.0690 \\ 0 & 0 & 1.0235 & -.2326 \\ 0 & 0 & 0 & 1.7710 \end{bmatrix} \quad (2)$$

and with **B=UU**^{T},

$$\mathbf{B}=\begin{bmatrix} 6 & 3 & 4 & 4 \\ 3 & 5 & 3 & 4 \\ 4 & 3 & 4 & 3 \\ 4 & 4 & 3 & 7 \end{bmatrix} \qquad (3)$$

Apparently, $\mathbf{A}\neq\mathbf{B}$.

Moreover, when "[**U,**p]=chol(**A**)" command is input, one will obtain **U** as given before, with the flag value p=0, meaning that algorithm treats **A** as a positive definite matrix! The same situation can be observed for other values of A_{11} or A_{33} (e.g., $A_{11}=7$, $A_{33}=5$).

The reason for this is that, MATLAB does not check whether a given matrix is real symmetric or complex Hamiltonian. It takes the upper triangular part of the matrix under consideration and assume WITHOUT CHECK that the strictly lower part of it will conform.

Therefore, one must make sure that a matrix is symmetric before using it in the CF algorithm, by imposing symmetry testing conditions (especially for larger matrices). This may cost time to check for symmetry, however, saves much more of it by preventing errors.

APPENDIX

A necessary and sufficient condition for a complex matrix **A** to be positive definite is that the Hermitian part $\mathbf{A_H=(A+A^H)/2}$, where $\mathbf{A}^H$ denotes the conjugate transpose, be positive definite. This means that a real matrix **A** is positive definite if and only if the symmetric part $\mathbf{A_S=(A+A^H)/2}$ where $\mathbf{A}^T$ is the transpose, is positive definite [4-6].

A Hermitian (or symmetric) matrix is positive definite if and only if all its eigenvalues are positive. Therefore, a general complex (respectively, real) matrix is positive definite if and only if its Hermitian (or symmetric) part has all positive eigenvalues.

The determinant of a positive definite matrix is always positive, so a positive definite matrix is always non-singular.

If **A** and **B** are positive definite, then so is their matrix sum. The matrix inverse of a positive definite matrix is also positive definite.

The definition of positive definiteness is equivalent to the requirement that the determinants associated with *all* upper-left sub matrices are positive.

The following are necessary (but not sufficient) conditions for a Hermitian matrix **A** (which by definition has real diagonal elements a_{ii}) to be positive definite:

1. $a_{ii} > 0$ for all i ,
2. $a_{ii}+a_{jj} > 2\, R[a_{ij}]$, for $i \neq j$,
3. The element with largest modulus lies on the main diagonal,
4. det (**A**) >0.

Here, R[z] is the real part of z.
A real symmetric matrix **A** is positive definite if and only if there exists a real non-singular matrix **L** such that $\mathbf{A=LL}^T$ where $\mathbf{L}^T$ is the transpose.

REFERENCES

[1]. MATLAB function reference, version 6, release 12, The Mathworks Inc., November 2000, pp: 207-208.

[2]. Horn, R.A. and Johnson, C.R., Matrix Analysis. Cambridge University Press, Reprinted 1991, pp: 397-398.

[3]. Moon, T.K. and Stirling W.C., Mathematical methods and algorithms for signal processing. Prentice Hall, N. J., 2000, pp:275-283.

[4]. Eric W. Weisstein. "Positive Definite Matrix." From *MathWorld*--A Wolfram Web Resource.

[5]. Ayres, F. Jr. *Schaum's Outline of Theory and Problems of Matrices.* New York: Schaum, p. 134, 1962.

[6]. Pease, M. C. *Methods of Matrix Algebra.* New York: Academic Press, 1965.

Ali Ümit Keskin received his BSEE degree from Bogazici University in 1978, MSEE from Yildiz University (1980) and and PhD degree from the Institute of Science and Technology, Istanbul Technical University (1984), respectively. In 1985, he joined Siemens AG, Turkey. He was engaged in the field of Medical Imaging Technology throughout his professional career in Siemens AG. Since September 2002, he is affiliated as an assistant professor in Department of Electical Engineering, Yeditepe University, Istanbul. His main research interests are analog signal processing, sensors and transducers , medical imaging theory and its applications.

İmran Göker Born in Ankara Turkey, in 1969. He received his M.D. Degree from Medical Faculty in Ege University in 1996. He received his MSc Degree in Biomedical Engineering from the Institute of Biomedical Engineering in Boğaziçi University in 2000. He is continuing his PhD studies at the same Institute. He is also the head of department of Dental Health in the School of Vocational Studies at Yeditepe University.