CME 302: NUMERICAL LINEAR ALGEBRA
FALL 2005/06
LECTURE 8

GENE H. GOLUB

1. Positive Definite Matrices

A matrix A is *positive definite* if $\mathbf{x}^\top A\mathbf{x} > 0$ for all nonzero $\mathbf{x}$. A positive definite matrix has real and positive eigenvalues, and its leading principal submatrices all have positive determinants. From the definition, it is easy to see that all diagonal elements are positive.

To solve the system $A\mathbf{x} = \mathbf{b}$ where A is positive definite, we can compute the *Cholesky decomposition* $A = F^\top F$ where F is upper triangular. This decomposition exists if and only if A is symmetric and positive definite. In fact, attempting to compute the Cholesky decomposition of A is an efficient method for checking whether A is symmetric positive definite.

It is important to distinguish the Cholesky decomposition from the *square root factorization*. A square root of a matrix A is defined as a matrix S such that

$$S^2 = SS = A.$$

Note that the matrix F in $A = F^\top F$ is not the square root of A, since it does not hold that $F^2 = A$ unless A is a diagonal matrix. The square root of a symmetric positive definite A can be computed by using the fact that A has an eigendecomposition $A = U\Lambda U^\top$ where Λ is a diagonal matrix whose diagonal elements are the positive eigenvalues of A and U is an orthogonal matrix whose columns are the eigenvectors of A. It follows that

$$A = U\Lambda U^\top = (U\Lambda^{1/2}U^\top)(U\Lambda^{1/2}U^\top) = SS$$

and so $S = U\Lambda^{1/2}U^\top$ is a square root of A.

2. The Cholesky Decomposition

The Cholesky decomposition can be computed directly from the matrix equation $A = F^\top F$. Examining this equation on an element-by-element basis yields the equations

$$\begin{aligned}
a_{11} &= f_{11}^2, & & \\
a_{1j} &= f_{11}f_{1j}, & & j = 2, \ldots, n \\
&\vdots & & \\
a_{kk} &= f_{1k}^2 + f_{2k}^2 + \cdots + f_{kk}^2, & & \\
a_{kj} &= f_{1k}f_{1j} + \cdots + f_{kk}f_{kj}, & & j = k+1, \ldots, n
\end{aligned}$$

and the resulting algorithm that runs for $k = 1, \ldots, n$:

$$\begin{aligned}
f_{kk} &= \left(a_{kk} - \textstyle\sum_{j=1}^{k-1} f_{jk}^2\right)^{1/2} & & \\
f_{kj} &= \left(a_{kj} - \textstyle\sum_{\ell=1}^{k-1} f_{\ell k}f_{\ell j}\right) \,/\, f_{kk}, & & j = k+1, \ldots, n.
\end{aligned}$$

This algorithm requires roughly half as many operations as Gaussian elimination.

Date: October 26, 2005, version 1.0.

Notes originally due to James Lambers. Edited by Lek-Heng Lim.

So if A is symmetric positive definite, then we could compute the decomposition

$$A = FF^\top,$$

known as the *Cholesky decompostion*. In fact, there are several ways to write $A = GG^\top$ for some matrix G since

$$A = FF^\top = FQQ^\top F = (FQ)(FQ)^\top = GG^\top$$

for any orthogonal matrix Q, but for the Cholesky decomposition, we require that F is lower triangular, with positive diagonal elements.

We can compute F by examining the matrix equation $A = FF^\top$ on an element-by-element basis, writing

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} f_{11} & & & \\ f_{21} & f_{22} & & \\ \vdots & \vdots & \ddots & \\ f_{n1} & f_{n2} & \cdots & f_{nn} \end{bmatrix} \begin{bmatrix} f_{11} & f_{21} & \cdots & f_{n1} \\ & f_{22} & & \vdots \\ & & \ddots & \vdots \\ & & & f_{nn} \end{bmatrix}.$$

From the above matrix multiplication we see that $f_{11}^2 = a_{11}$, from which it follows that

$$f_{11} = \sqrt{a_{11}}.$$

From the relationship $f_{11}f_{i1} = a_{i1}$ and the fact that we already know f_{11}, we obtain

$$f_{i1} = \frac{a_{i1}}{f_{11}}, \quad i = 2, \ldots, n.$$

Proceeding to the second column of F, we see that $f_{21}^2 + f_{22}^2 = a_{22}$. Since we already know f_{21}, we have

$$f_{22} = \sqrt{a_{22} - f_{21}^2}.$$

Next, we use the relation $f_{21}f_{i1} + f_{22}f_{i2} = a_{2i}$ to compute

$$f_{i1} = \frac{a_{2i} - f_{21}f_{i1}}{f_{22}}.$$

In general, we can use the relationship $a_{ij} = \mathbf{f}_i^\top \mathbf{f}_j$ to compute f_{ij}, where $\mathbf{f}_i$ is the ith column of F.

Another method for computing the Cholesky decomposition is to compute

$$\mathbf{f}_1 = \frac{1}{\sqrt{a_{11}}}\mathbf{a}_1$$

where $\mathbf{a}_i$ is the ith column of A. Then we set $A^{(1)} = A$ and compute

$$A^{(2)} = A^{(1)} - \mathbf{f}_1\mathbf{f}_1^\top = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & A_2 & \\ 0 & & & \end{bmatrix}.$$

Note that

$$A^{(1)} = B \begin{bmatrix} 1 & 0 \\ 0 & A_2 \end{bmatrix} B^\top$$

where B is the identity matrix with its first column replaced by $\mathbf{f}_1$. Writing $C = B^{-1}$, we see that A_2 is positive definite since

$$\begin{bmatrix} 1 & 0 \\ 0 & A_2 \end{bmatrix} = CAC^\top$$

is positive definite. So we may repeat the process on A_2.

We partition the matrix A_2 into columns, writing $A_2 = \begin{bmatrix} \mathbf{a}_2^{(2)} & \mathbf{a}_3^{(2)} & \cdots & \mathbf{a}_n^{(2)} \end{bmatrix}$ and then compute

$$\mathbf{f}_2 = \frac{1}{\sqrt{a_{22}^{(2)}}} \begin{bmatrix} 0 \\ \mathbf{a}_2^{(2)} \end{bmatrix}.$$

We then compute

$$A_3 = A^{(2)} - \mathbf{f}_2\mathbf{f}_2^\top$$

and so on.

Note that

$$a_{kk} = f_{k1}^2 + f_{k2}^2 + \cdots + f_{kk}^2,$$

which implies that

$$|f_{ki}| \leq |a_{kk}|.$$

In other words, the elements of F are bounded. We also have the relationship

$$\det A = \det F \det F^\top = (\det F)^2 = f_{11}^2 f_{22}^2 \cdots f_{nn}^2.$$

Is the Cholesky decompositon unique? Employing a similar approach to the one used to prove the uniquess of the LU decomposition, we assume that A has two Cholesky decompositions

$$A = F_1F_1^\top = F_2F_2^\top.$$

Then

$$F_2^{-1}F_1 = F_2^\top F_1^{-\top},$$

but since F_1 and F_2 are lower triangular, both matrices must be diagonal. Let

$$F_2^{-1}F_1 = D = F_2^\top F_1^{-\top}.$$

So $F_1 = F_2D$ and thus $F_1^\top = DF_2^\top$ and we get $D^{-1} = F_2^\top F_1^{-\top}$. In other words, $D^{-1} = D$ or $D^2 = I$. Hence D must have diagonal elements equal to ± 1. Since we require that the diagonal elements be positive, it follows that the decomposition is unique.

In computing the Cholesky decomposition, no row interchanges are necessary because A is positive definite, so the number of operations required to compute F is approximately $n^3/3$.

A variant of the Cholesky decomposition is known as the *square-root-free Cholesky decomposition*, and has the form

$$A = LDL^\top$$

where L is a unit lower triangular matrix, and D is a diagonal matrix with positive diagonal elements. This is a special case of the $A = LDM^\top$ factorization previously discussed. The $LDL^\top$ and Cholesky decompositions are related by

$$F = LD^{1/2}.$$

3. Banded Matrices

A *banded* matrix has all of its nonzero elements contained within a "band" consisting of select diagonals. Specifically, a matrix A that has *upper bandwidth* p and *lower bandwidth* q has the form

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1,p+1} & & \\ a_{21} & \ddots & a_{2,p+1} & a_{2,p+2} & \\ \vdots & & & & \\ a_{q+1,1} & \cdots & a_{q+1,q+1} & \cdots & a_{q+1,n} \\ & & \ddots & \ddots & \vdots \end{bmatrix}.$$

Matrices of this form arise frequently from discretization of partial differential equations.

The simplest banded matrix is a *tridiagonal* matrix, which has upper bandwidth 1 and lower bandwidth 1. Such a matrix can be stored using only three vectors instead of a two-dimensional array. Computing the LU decomposition of a tridiagonal matrix without pivoting requires only $O(n)$ operations, and produces bidiagonal L and U. When pivoting is used, this desirable structure is lost, and the process as a whole is more expensive in terms of computation time and storage space.

Various applications, such as the solution of partial differential equations in two or more space dimensions, yield symmetric *block tridiagonal* matrices, which have a block Cholesky decomposition:

$$\begin{bmatrix} A_1 & B_2^\top & & \\ B_2 & \ddots & \ddots & \\ & \ddots & \ddots & B_n^\top \\ & & B_n & A_n \end{bmatrix} = \begin{bmatrix} F_1 & & & \\ G_2 & \ddots & & \\ & \ddots & \ddots & \\ & & G_n & F_n \end{bmatrix} \begin{bmatrix} F_1^\top & G_2^\top & & \\ & \ddots & \ddots & \\ & & \ddots & G_n^\top \\ & & & F_n^\top \end{bmatrix}.$$

From the above matrix equation, we determine that

$$A_1 = F_1 F_1^\top, \quad B_2 = G_2 F_1^\top$$

from which it follows that we can compute the Cholesky decomposition of A_1 to obtain F_1, and then compute $G_2 = B_2(F_1^\top)^{-1}$. Next, we use the relationship $A_2 = G_2 G_2^\top + F_2 F_2^\top$ to obtain

$$F_2 F_2^\top = A_2 - G_2 G_2^\top = A_2 - B_2(F_1^\top)^{-1} F_1^{-1} B_2^\top = A_2 - B_2 A_1^{-1} B_2.$$

It is interesting to note that in the case of $n = 2$, the matrix $A_2 - B_2 A_1^{-1} B_2$. is known as the *Schur complement* of A_1.

Continuing with the block tridiagonal case with $n = 2$, suppose that we wish to compute the factorization

$$\begin{bmatrix} A & B \\ B^\top & 0 \end{bmatrix} = \begin{bmatrix} F \\ G \end{bmatrix} \begin{bmatrix} F^\top & G^\top \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & X \end{bmatrix}.$$

It is easy to see that $X = -B^\top A^{-1} B$, but this matrix is *negative definite*. Therefore, we cannot compute a block Cholesky decomposition, but we can achieve the factorization

$$\begin{bmatrix} A & B \\ B^\top & 0 \end{bmatrix} = \begin{bmatrix} F & 0 \\ G & K \end{bmatrix} \begin{bmatrix} F^\top & G^\top \\ 0 & -K^\top \end{bmatrix}$$

where K is the Cholesky factor of the positive definite matrix $B^\top A^{-1} B$.

4. Parallelism of Gaussian Elimination

Suppose that we wish to perform Gaussian elimination on the matrix $A = \begin{bmatrix} \mathbf{a}_1 & \cdots & \mathbf{a}_\mathbf{n} \end{bmatrix}$. During the first step of the elimination, we compute

$$P^{(1)}\Pi_1 A = \begin{bmatrix} P^{(1)}\Pi_1 \mathbf{a}_1 & \cdots & P^{(1)}\Pi_1 \mathbf{a}_n \end{bmatrix}.$$

Clearly we can work on each column independently, leading to a parallel algorithm. As the elimination proceeds, we obtain less benefit from parallelism since fewer columns are being modified at each step.

5. Error Analysis of Gaussian Elimination

Suppose that we wish to solve the system $A\mathbf{x} = \mathbf{b}$. Our computed solution $\tilde{\mathbf{x}}$ satisfies a perturbed system $(A + \Delta)\tilde{\mathbf{x}} = \mathbf{b}$. It can be shown that

$$\begin{aligned} \frac{\|\mathbf{x} - \tilde{\mathbf{x}}\|}{\|\mathbf{x}\|} &\leq \frac{\|A^{-1}\|\|\Delta\|}{1 - \|A^{-1}\|\|\Delta\|} \\ &\leq \frac{\|A\|\|A^{-1}\|\frac{\|\Delta\|}{\|A\|}}{1 - \|A\|\|A^{-1}\|\frac{\|\Delta\|}{\|A\|}} \\ &\leq \frac{\kappa(A) r}{1 - \kappa(A) r} \end{aligned}$$

where $\kappa(A) = \|A\|\|A^{-1}\|$ is the *condition number* of A and $r = \|\Delta\|/\|A\|$.. The condition number has the following properties:

- $\kappa(\alpha A) = \kappa(A)$ where α is a nonzero scalar.

- $\kappa(I) = 1$
- $\kappa(Q) = 1$ when $Q^\top Q = I$.

The perturbation matrix Δ is typically a function of the algorithm used to solve $A\mathbf{x} = \mathbf{b}$.

In this section, we will consider the case of Gaussian elimination and perform a detailed error analysis, illustrating the analysis originally carried out by J.H. Wilkinson. The process of solving $A\mathbf{x} = \mathbf{b}$ consists of three stages:

(1) Factoring $A = LU$, resulting in an approximate LU decomposition $A + E = \bar{L}\bar{U}$

(2) Solving $L\mathbf{y} = \mathbf{b}$, or, numerically, computing $\mathbf{y}$ such that

$$(\bar{L} + \delta\bar{L})(\mathbf{y} + \delta\mathbf{y}) = \mathbf{b}$$

(3) Solving $U\mathbf{x} = \mathbf{y}$, or, numerically, computing $\mathbf{x}$ such that

$$(\bar{U} + \delta\bar{U})(\mathbf{x} + \delta\mathbf{x}) = \mathbf{y} + \delta\mathbf{y}.$$

Combining these stages, we see that

$$\begin{aligned}
\mathbf{b} &= (\bar{L} + \delta\bar{L})(\bar{U} + \delta\bar{U})(\mathbf{x} + \delta\mathbf{x}) \\
&= (\bar{L}\bar{U} + \delta\bar{L}\bar{U} + \bar{L}\delta\bar{U} + \delta\bar{L}\delta\bar{U})(\mathbf{x} + \delta\mathbf{x}) \\
&= (A + E + \delta\bar{L}\bar{U} + \bar{L}\delta\bar{U} + \delta\bar{L}\delta\bar{U})(\mathbf{x} + \delta\mathbf{x}) \\
&= (A + \Delta)(\mathbf{x} + \delta\mathbf{x})
\end{aligned}$$

where $\Delta = \delta\bar{L}\bar{U} + \bar{L}\delta\bar{U} + \delta\bar{L}\delta\bar{U}$.

Department of Computer Science, Gates Building 2B, Room 280, Stanford, CA 94305-9025

E-mail address: `golub@stanford.edu`