PS # 5 Solutions

Problem 20.1

i) Let $A = \begin{bmatrix} A_{1:k,1:k} & A_{1:k,k+1:m} \\ A_{k+1:m,1:k} & A_{k+1:m,k+1:m} \end{bmatrix}$

$= \begin{bmatrix} L_{1:k,1:k} & 0 \\ L_{k+1:m,1:k} & L_{k+1:m,k+1:m} \end{bmatrix} \begin{bmatrix} U_{1:k,1:k} & - \\ 0 & U_{..} \end{bmatrix}$

so $A_{1:k,1:k} = L_{1:k,1:k} \cdot U_{1:k,1:k}$

Since $U_{1:k,1:k}$ is upper triangular with diagonal elements nonzero we get

$\det(A_{1:k,1:k}) = \det(U_{1:k,1:k}) \neq 0$.

ii) Use induction on $m = \dim A$.

For a 1×1 matrix is obvious.

Assume for matrices of dim $\leq m-1$

Take $\tilde{A} = \begin{bmatrix} A & b \\ c^T & d \end{bmatrix}$ where A is $m-1\times m-1$ matrix. Then try to find L, ℓ^T, U, v, w such that $\tilde{A} = \begin{bmatrix} L & 0 \\ \ell^T & 1 \end{bmatrix} \cdot \begin{bmatrix} U & v \\ 0 & w \end{bmatrix}$, so

$A = LU$, $b = Lv$, $c^T = \ell^T U$, $d = \ell^T v + w$

which has a solution for L & U by induction assumption and then since L & U are invertible (A is invertible) we can solve for v, ℓ, w.

iii) The uniqueness follows since

$L_1 U_1 = L_2 U_2 \Rightarrow L_2^{-1} L_1 = U_2 U_1^{-1}$

and because $L_2^{-1} L_1$ is lower triangular while $U_2 U_1^{-1}$ is upper triangular, they must be diagonal. So since then $L_2^{-1} L_1 = Id$, so $U_2 U_1^{-1} = Id$ hence

$L_1 = L_2$, $U_2 = U_1$.

Problem 21.3

a) Since A is nonsingular it must be an entry nonzero on the first row. Then using a matrix Q_1 we get AQ_1 has the upper left element nonzero. Then proceed with Gaussian elimination on the first column. Then we get a $m-1 \times m-1$ matrix in the lower right which is not invertible and proceed further as above. (remark that this is $m-1\times m-1$ and invertible)

Then we get $L_{m-1} \cdots L_1 A Q_1 \cdots Q_{m-1} = U$ so $AQ = LU$.

b) For example if $A = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$ then we can not write $AQ = LU$. Indeed $AQ = A$ and if $L = \begin{pmatrix} 1 & 0 \\ \ell & 1 \end{pmatrix}$, $U = \begin{pmatrix} u & v \\ 0 & w \end{pmatrix}$ we get $\begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \ell & 1 \end{pmatrix} \begin{pmatrix} u & v \\ 0 & w \end{pmatrix} = \begin{pmatrix} u & v \\ \ell u & \ell v + w \end{pmatrix}$

so $u = v = 0$ and $\ell u = 1$ impossible

Problem 21.6

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

Proceed with Gaussian elimination to arrive to:

$$\begin{pmatrix} a_{11} & A_{12} \\ 0 & A_{22} - \frac{1}{a_{11}} A_{21} A_{12} \end{pmatrix}.$$

(Keep in mind that $|a_{11}| > \sum_{i\neq 1} |a_{i1}| \geq 0$)

Now for $A_{22}-\frac{1}{a_{11}}A_{21}A_{12}$ we show that it has the property of strictly column diagonally dominant.

$$\sum_{j\neq k}\left|\left(A_{22}-\frac{1}{a_{11}}A_{21}A_{12}\right)_{jk}\right| \leq$$

$$\leq \sum_{j\neq k}|(A_{22})_{jk}|+\sum_{j\neq k}\left|\frac{1}{a_{11}}(A_{21})_j(A_{12})_k\right|$$

$$\overset{*}{<} |(A_{22})_{kk}|-|(A_{12})_k|+\frac{|(A_{12})_k|}{|a_{11}|}(|a_{11}|-$$

$$-|(A_{21})_k|)=|(A_{22})_{kk}|-\frac{|(A_{12})_k||(A_{21})_k|}{|a_{11}|}$$

$$\leq \left|(A_{22})_k-\frac{(A_{12})_k(A_{21})_k}{a_{11}}\right|=$$

$$=\left|\left(A_{22}-\frac{1}{a_{11}}A_{21}A_{12}\right)_{kk}\right|$$

where in $*$ we used the original inequality for A.

Hence by induction if the property is true for any matrix of dimension $\leq n-1$ then it is true for any matrix A of dim A $=n$, just using the above property.

Problem 23.3

a) A is symmetric and positive so "\" command solve for Cholesky factorization and two backward substitutions. The time for this is $\sim \frac{1}{3}m^3$ flops.

b) This tests is to make sure that cache effects are eliminated the rest being as above.

c) Here the LU factorization is used instead of Cholesky. So the time is $\sim \frac{2}{3}m^3$ flops.

d) Here 0.9 times the smallest eigenvalue is substracted from diagonal of A, this way the matrix being still positive definite and can be solved by Cholesky.

e) Here the matrix is no longer positive definite so Cholesky does not work. The longer time Matlab needs comes from the fact that the matrix is still symmetric and this takes time for it to realize it.

f) This measures the time for backward solution of a triangular system. It takes $\sim m^2$ flops and the time is much smaller than for factorization

g) Here Matlab uses full LU factorization so it takes twice the time for Cholesky.