Math 426 Intro to Numerical Analysis and Algorithmic Computing 03F

R. J. Braun

Assignment 4, due Wednesday 10/29/03, 11:15am

1. Ch. 6, §2: Solve problem 6.1.3(a) using Gaussian Elimination with Partial Pivoting. You may use GEPPandBS.m if you like.

2. Ch. 6, §5: 3(a),4(a).

3. Ch. 6, §6: 1(acf),6(c).

4. Count the operations required for Crout’s method as in Algorithm 6.7 of your text.

Solutions

Problem 6.1.3(a) with partial pivoting

MATLAB is used to do this problem.

>> % Math 426 Intro to Numerical Analysis and Algorithmic Computation
>> % R Braun
>> % Homework 4
>> %
>> % Problem 6.1.3(a), using the matlab function file GEPPandBS.m
>> % to do partial pivoting with GE
>> % First define A and b, then call the function
>> % (lots of output from function follows)
>> A = [1 -1 3; 3 -3 1; 1 1 0]
A =
   1 -1  3
   3 -3  1
   1  1  0
>> b=[2; -1; 3]
    b =
         2
         -1
         3
>> [x,U,b,piv]=GEPPandBS(A,b)
Row switching?
piv =
   2  1  3
A =
   3 -3  1
   1 -1  3
   1  1  0
b =
   -1
    2
    3
After modification with row 1
A =
  3.0000 -3.0000  1.0000
  0.3333  0  2.6667
  0.3333  2.0000 -0.3333
b =
Row switching?
piv = 

A =

b =

After modification with row 2
A =

b =

% final output from function now...

x =

U =

b =

piv =

>> % Two row switches were required.
>> %

Problem 6.5.3(a) and 4(a)

Using Matlab with no partial pivoting, the factorization and solution are as follows.

>> !cat GE1b.m
% Script: GE1b.m
% basic LU with row switching for zero pivots only
% use matlab functions for triangular solves
% A= [1 2 3; 4 5 6; 7 8 0];
% b=[6 15 15]';
A= [2 -1 1; 3 3 9; 3 3 5];
b=[-1 0 4]';
n = 3;
piv=1:n;
x=zeros(n,1);
m=eye(n);
for ii=1:n-1
  p = 1;
  if A(ii,ii)==0
    while A(ii+p,ii)==0 & p<n
      p = p+1
    end
    if p==n
      disp('no unique solution')
      break
    end
    A([ii ii+p],:) = A([ii+p ii],:);
piv([ii ii+p]) = piv([ii+p ii]);
  end
  for jj=ii+1:n
    A(jj,ii) = A(jj,ii)/A(ii,ii);
    for k=ii+1:n
      A(jj,k) = A(jj,k) - A(jj,ii)*A(ii,k);
    end
  end
end
A
piv
L = tril(A,-1)+eye(size(A))
U = triu(A)
y = L\b(piv)
x= U\y
>> GE1b
A =
  2.0000  -1.0000   1.0000
  1.5000   4.5000   7.5000
  1.5000   4.5000   3.5000
piv =
   1   2   3
A =
  2.0000  -1.0000   1.0000
  1.5000   4.5000   7.5000
  1.5000   1.0000  -4.0000
piv =
   1   2   3
L =
  1.0000    0    0
  1.5000   1.0000    0
  1.5000   1.0000   1.0000
U =
  2.0000  -1.0000   1.0000
   0  4.5000   7.5000
   0    0  -4.0000
y =
 -1.0000
  1.5000
  4.0000

3
Using Matlab and partial pivoting, the factorization and solution are as follows.

\[
A = \begin{bmatrix}
2 & -1 & 1 \\
3 & 3 & 9 \\
3 & 3 & 5 \\
\end{bmatrix}
\]

\[
[L, U, P] = lu(A)
\]

\[
L = \begin{bmatrix}
1 & 0 & 0 \\
0.6667 & 1 & 0 \\
1 & 0 & 1 \\
\end{bmatrix}
\]

\[
U = \begin{bmatrix}
3 & 3 & 9 \\
0 & -3 & -5 \\
0 & 0 & -4 \\
\end{bmatrix}
\]

\[
P = \begin{bmatrix}
0 & 1 & 0 \\
1 & 0 & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\]

\[
b = \begin{bmatrix}
-1 \\
0 \\
4 \\
\end{bmatrix}
\]

\[
y = L \backslash (P \cdot b)
\]

\[
x = \begin{bmatrix}
1 \\
2 \\
-1 \\
\end{bmatrix}
\]

**Problems 6.6.1 (acf)**

(a) symmetric, strictly diagonally dominant, positive definite.

(c) strictly diagonally dominant.

To test for positive definiteness, compute \(x^T A x\).

\[
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 
\end{bmatrix}
= \begin{bmatrix}
2 & 1 & 0 \\
0 & 3 & 0 \\
1 & 0 & 4 
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 
\end{bmatrix}
= \begin{bmatrix}
2x_1 + x_2 \\
3x_2 \\
x_1 + 4x_3 
\end{bmatrix}
= 2x_1^2 + x_1x_2 + 3x_2^2 + x_1x_3 + 4x_3^2
\]

We can complete the square to obtain

\[
x^T A x = \left( x_1 + \frac{x_2}{2} \right)^2 + \frac{11}{4} x_2^2 + \left( x_1 + \frac{x_3}{2} \right)^2 + \frac{15}{4} x_3^2 \geq 0
\]
Now if the matrix did not need to be symmetric, then this would show that the matrix is positive definite; however, your book requires both this and symmetry for the $A$ to be positive definite.

(f) symmetric, positive definite.

Problem 6.6.6(c)

The factorization may be computed as in the table.

| $i$ | result
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$l_{11} = a_{11} = 2$</td>
</tr>
<tr>
<td>1</td>
<td>$u_{12} = a_{12}/l_{11} = -1/2$</td>
</tr>
<tr>
<td>2</td>
<td>$l_{21} = a_{21} = -1$</td>
</tr>
<tr>
<td>2</td>
<td>$l_{22} = a_{22} - a_{21}u_{12} = 3/2$</td>
</tr>
<tr>
<td>2</td>
<td>$u_{23} = a_{23}/a_{22} = -2/3$</td>
</tr>
<tr>
<td>3</td>
<td>$l_{32} = a_{32} = -1$</td>
</tr>
<tr>
<td>3</td>
<td>$l_{33} = a_{33} - a_{n,n-1}u_{n-1} = 4/3$</td>
</tr>
</tbody>
</table>

We then have

$$L = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 3/2 & 0 \\ 0 & -1 & 4/3 \end{bmatrix} \text{ and } U = \begin{bmatrix} 1 & -1/2 & 0 \\ 0 & 1 & -2/3 \\ 0 & 0 & 1 \end{bmatrix}$$

Use forward substitution, solving $Lz = b$, to get $z = [3 - 3/2 0]^T$ then use back substitution, solving $Ux = z$ to get $x = [1 - 1 0]^T$.

Operation count for Crout’s method

The following is an operation count for Crout’s method algorithm in text.

<table>
<thead>
<tr>
<th>Step</th>
<th>addn</th>
<th>mult</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>2(n-2)</td>
<td>4(n-2)</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>n-1</td>
<td>n-1</td>
</tr>
<tr>
<td>total</td>
<td>$3n - 3$</td>
<td>$5n - 4$</td>
</tr>
</tbody>
</table>