Solving Upper/Lower Triangular Systems

Tommy Luckner
Department of Mathematics

Overview

We develop MATLAB functions to solve upper and lower triangular systems and then solve a more general system with LU factorization.

Activities

1. Suppose we have a 4×4 upper triangular system $A\mathbf{x} = \mathbf{b}$ as below:

$$A = \begin{pmatrix} -5 & 5 & 0 & -2 \\ 0 & 6 & 5 & 7 \\ 0 & 0 & 9 & -7 \\ 0 & 0 & 0 & -4 \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} -2 \\ -9 \\ 3 \\ -7 \end{pmatrix}.$$

   In the command window, define these matrices.

2. To solve for $\mathbf{x}$ using backward substitution, type

```
>> % Initialize the solution.
>> x = zeros(4,1)
>> % Solve A(4,4)*x(4) = b(4).
>> x(4) = b(4)/A(4,4)
>> % Solve A(3,3)*x(3) + A(3,4)*x(4) = b(3).
>> x(3) = (b(3) - A(3,4)*x(4))/A(3,3)
>> % Solve A(2,2)*x(2) + A(2,3)*x(3) + A(2,4)*x(4) = b(2).
>> x(2) = (b(2) - A(2,3)*x(3) - A(2,4)*x(4))/A(2,2)
>> % Solve A(1,1)*x(1) + A(1,2)*x(2)
>> % + A(1,3)*x(3) + A(1,4)*x(4) = b(1).
>> x(1) = (b(1) - A(1,2)*x(2) - A(1,3)*x(3) - A(1,4)*x(4))/A(1,1)
```

3. To perform the same task using matrix multiplication, type

```
>> clear x
>> x = zeros(4,1)
>> x(4) = b(4)/A(4,4)
>> x(3) = (b(3) - A(3,4:4)*x(4:4))/A(3,3)
>> x(2) = (b(2) - A(2,3:4)*x(3:4))/A(2,2)
>> x(1) = (b(1) - A(1,2:4)*x(2:4))/A(1,1)
```

   Remark: When performing matrix multiplication, make sure the matrices have proper dimension.

In-Class Exercise

4. Finish the following m-file.

```
% This function performs backward substitution for 4 x 4 upper
% triangular systems. It solves A*x = b.
% Input: upper triangular matrix A, column vector b
% Output: column vector x that solves A*x = b
function x = backward4(A,b)
    x = zeros(4,1);
    x(...) = ...
    for i = 3:-1:1
        x(...) = ...
    end
end
```

5. Run your function on the same upper triangular system as given above.

6. Write a function `forward4` to solve 4×4 (or $n \times n$ if you want) lower triangular systems (using forward substitution). Test your program with $B\mathbf{x} = \mathbf{b}$, where $B = A^T$, and A and b are given above.

7. We now have `backward4.m` and `forward4.m`, as well as `MYLU.m` from last week's lab. Combine these to solve the 4×4 system $A\mathbf{x} = \mathbf{b}$ as below:

$$A = \begin{pmatrix} -8 & 1 & 5 & 9 \\ -6 & 9 & 3 & -4 \\ -5 & -2 & 9 & -9 \\ 8 & -4 & 3 & -3 \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} -2 \\ -7 \\ 9 \\ -5 \end{pmatrix}.$$

Remark on MATLAB Functions

Inside of a function in MATLAB, you can call and use any other function that you have, so long as you do one of the following: (a) the function that you are calling is in the same folder as the function that is using it; (b) the function that you are calling is beneath the function that is using it in the same script. For example, below `made_up_function.m` uses `other_function.m`. `other_function.m` will have to be in the same folder as `made_up_function.m`, or written beneath `made_up_function.m` in the same script.

```
function [W,f] = made_up_function(z,p)
x = sin(z)*p/2;
y = other_function(x);
f = y^(1/3);
W = f*x;
end
```