



UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
ÁREA DE ANÁLISIS FUNCIONAL
MÉTODOS NUMÉRICOS

Integrantes: Jefferson David Yépez Morán
Fecha: 9 de julio de 2024

NRC:14188

Tarea 2.2

- Realice el programa que resuelva un sistema de ecuaciones $n \times n$ por medio del método de descomposición LU. Debe incluir una sección que detenga el cálculo en el caso que algún elemento de la diagonal sea nulo.

- Corra una prueba con el programa para un sistema de 10×10 .

Para esta tarea, debe realizar una entrega:

Aula virtual: PDF realizado en LaTeX con los respaldos pertinentes. Además, adjuntar en el mismo documento el código fuente del algoritmo con el script solicitado. (No hay entrega en Gradescope).

Funcion:

```
1 function [L, U, x] = descomposicionLU(A, b)
2     % Verificar si la matriz es cuadrada
3     [n, m] = size(A);
4     if n ~= m
5         error('La matriz A debe ser cuadrada.');
```

16

```
6     end
7
8     % Inicializar matrices L y U
9     L = zeros(n);
10    U = zeros(n);
11
12    % Descomposici n LU
13    for i = 1:n
14        % Verificar si el elemento diagonal es nulo
15        if A(i, i) == 0
16            error('Elemento nulo en la diagonal. No se puede continuar con la descomposici n LU.');
```

18

```
17        end
18
19        for j = i:n
20            U(i, j) = A(i, j) - sum(L(i, 1:i-1) .* U(1:i-1, j));
21        end
22
23        for j = i:n
24            if i == j
25                L(i, i) = 1;
26            else
27                L(j, i) = (A(j, i) - sum(L(j, 1:i-1) .* U(1:i-1, i))) / U(i, i);
28            end
29        end
30    end
31 end
```

```

32 % Sustituci n progresiva para resolver Lc = b
33 c = zeros(n, 1);
34 for i = 1:n
35     c(i) = b(i) - sum(L(i, 1:i-1) .* c(1:i-1)');
36 end
37
38 % Sustituci n regresiva para resolver Ux = c
39 x = zeros(n, 1);
40 for i = n:-1:1
41     x(i) = (c(i) - sum(U(i, i+1:n) .* x(i+1:n)')) / U(i, i);
42 end
43 end

```

Ejecuci3n:

```

1 %Autor: Jefferson Y pez
2
3 % Prueba del programa con un sistema de 10x10
4 A = rand(10, 10);
5 b = rand(10, 1);
6 disp("A:");
7 disp(A);
8 disp("B: ");
9 disp(b);
10 [L, U, x] = descomposicionLU(A, b);
11 disp('Soluci n x:');
12 disp(x);

```

OUTPUT:

A:

```

0.1239 0.9479 0.7378 0.1339 0.5590 0.9884 0.1781 0.9121 0.8949 0.4561
0.4904 0.0821 0.0634 0.0309 0.8541 0.5400 0.3596 0.1040 0.0715 0.1017
0.8530 0.1057 0.8604 0.9391 0.3479 0.7069 0.0567 0.7455 0.2425 0.9954
0.8739 0.1420 0.9344 0.3013 0.4460 0.9995 0.5219 0.7363 0.0538 0.3321
0.2703 0.1665 0.9844 0.2955 0.0542 0.2878 0.3358 0.5619 0.4417 0.2973
0.2085 0.6210 0.8589 0.3329 0.1771 0.4145 0.1757 0.1842 0.0133 0.0620
0.5650 0.5737 0.7856 0.4671 0.6628 0.4648 0.2089 0.5972 0.8972 0.2982
0.6403 0.0521 0.5134 0.6482 0.3308 0.7640 0.9052 0.2999 0.1967 0.0464
0.4170 0.9312 0.1776 0.0252 0.8985 0.8182 0.6754 0.1341 0.0934 0.5054
0.2060 0.7287 0.3986 0.8422 0.1182 0.1002 0.4685 0.2126 0.3074 0.7614

```

B:

```

0.6311
0.0899
0.0809
0.7772
0.9051
0.5338
0.1092
0.8258
0.3381
0.2940

```

Soluci3n x:

```

-1.1357
-0.3443
0.7436
-0.0470

```

0.1860
0.0844
1.0997
0.5848
-0.3829
-0.0398