
Table of Contents

Aufgabe 3.13	1
Aufgabe 3.14	2
Aufgabe 3.15	3

Aufgabe 3.13

dbtype [myLRdecomp.m](#)

```
A = [1,3,1;2,9,6;0,6,5];  
[L,R] = myLRdecomp(A);
```

```
disp('A =');  
disp(A);
```

```
disp('L =');  
disp(L);
```

```
disp('R =');  
disp(R);
```

```
1      function [L,R] = myLRdecomp(A)  
2  
3      n=size(A,2);  
4      I=eye(n);  
5  
6      L=I;  
7  
8      e_k = zeros(n,1);  
9      l_k = zeros(n,1);  
10  
11     for k=1:n-1  
12  
13         if(k>1)  
14             e_k(k-1,1) = 0;  
15             l_k(k:n,1) = 0;  
16         end  
17         e_k(k,1) = 1;  
18  
19         a_k = A(:,k);  
20  
21         %Division mit dem Pivotelement  
22         l_k(k+1:n,1) = a_k(k+1:n,1) / a_k(k,1);  
23  
24         A = (I - l_k * e_k' ) * A;  
25  
26         L = L + l_k * e_k';  
27  
28     end  
29
```

```

30     R=A;
31
32     end
A =
     1     3     1
     2     9     6
     0     6     5

L =
     1     0     0
     2     1     0
     0     2     1

R =
     1     3     1
     0     3     4
     0     0    -3

```

Aufgabe 3.14

```

dbtype MatrixA.m
dbtype aufgabe3_14.m

```

```

aufgabe3_14

```

```

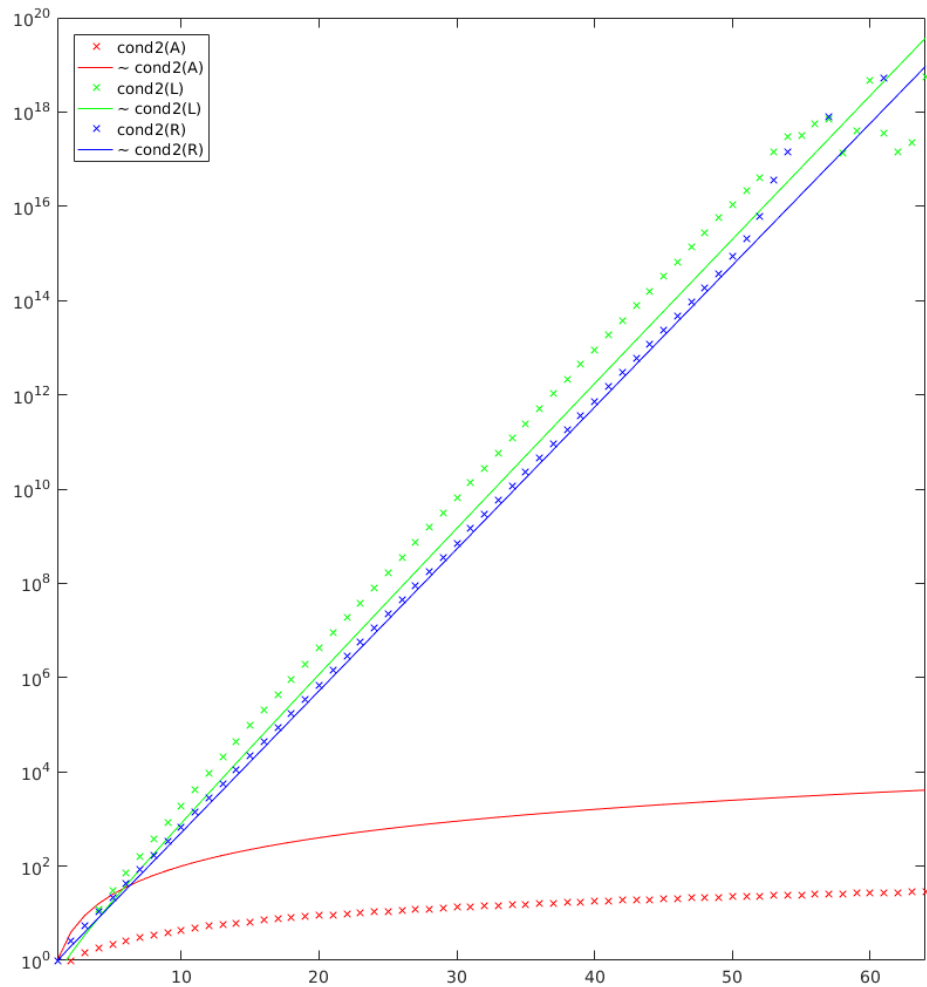
1     function [A] = MatrixA(n)
2
3     A = ones(n) * (-1);
4     A = tril(A);
5     A = A + 2 * eye(n);
6     A(:,n) = ones(n,1);
7
8     end
9
10
11     cA=[];
12     cL=[];
13     cR=[];
14
15     n = 64;
16
17     for i=1:n
18         A=MatrixA(i);
19         cA = [cA; cond(A,2)];
20         [L,R] = myLRdecomp(A);
21         cL = [cL; cond(L,2)];
22         cR = [cR; cond(R,2)];
23     end
24
25

```

```

16    %% print graph %%
17    x_space = linspace(1,n,n);
18    semilogy(x_space, cA, 'rx', x_space, x_space.^2, 'r',...
19            x_space, cL, 'gx', x_space, sqrt(x_space).*2.^(x_space-2),
20            'g',...
21            x_space, cR, 'bx', x_space, 2.^(x_space-1), 'b');
22    legend('cond2(A)', '~ cond2(A)', 'cond2(L)', '~
23          cond2(L)', 'cond2(R)', '~ cond2(R)', 'Location', 'NorthWest')
24    axis([1, 64, 1, 10^20])

```



Aufgabe 3.15

```

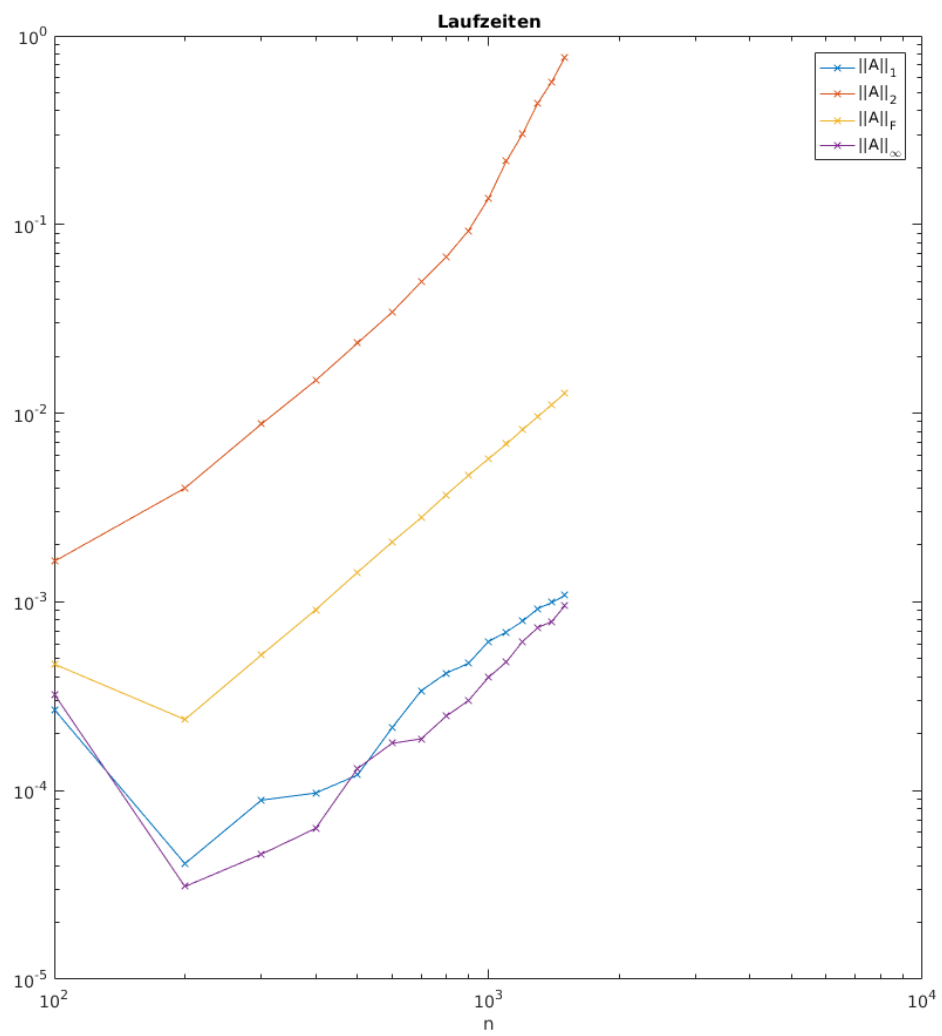
dbtype normtime.m
normtime

```

```

1    v = 100:100:1500;
2    n = length(v);
3
4    tSpek = ones(n,1) * inf;
5    tFrob = ones(n,1) * inf;
6    tZeilen = ones(n,1) * inf;
7    tSpalten = ones(n,1) * inf;
8
9    for k=1:n
10       A = randn( v(k) );
11
12       %spektralnorm  $\|A\|_1$ 
13       tic
14       norm(A,2);
15       tSpek(k,1) = toc;
16
17       %spaltensummennorm  $\|A\|_2$ 
18       tic
19       norm(A,1);
20       tSpalten(k,1) = toc;
21
22
23       %frobenius
24       tic
25       norm(A, 'fro' );
26       tFrob(k,1) = toc;
27
28       %zeilensummennorm
29       tic
30       norm(A, inf );
31       tZeilen(k,1) = toc;
32    end
33
34
35    T=[tSpalten,tSpek,tFrob,tZeilen];
36
37    loglog(v,T.','x-');
38    title('Laufzeiten')
39    xlabel('n')
40    legend('  $\|A\|_1$  ', '  $\|A\|_2$  ', '  $\|A\|_F$  ', '  $\|A\|_{\{\infty\}}$  ')

```



Published with MATLAB® R2016a