## Table of Contents



# Problem 1

Consider an r-dimensional column vector x whose entries are random real numbers between ?5 and 5. For an integer r between 15 and 25 (inclusive), (a) compute |x|_inf using a user-defined function norminf (b) compute |x|_1 using a user-defined function norm1 (c) compute |x|_2 using a user-defined function norm2 (d) compute |x|_p for p = 3 using a user-defined function normp

```
fprintf('Problem 1\n')
r = 14 + randi(11);

x = randi([-5,5], r, 1);

% (a)
xinf = Norminf(x)

% (b)
x1norm = Norm1(x)

% (c)
x2norm = Norm2(x)

% (d)
x3norm = Pnorm(x,3)
```

*Problem 1*

# Problem 2

Redo Problem 1 (for the same vector x) using the built-in function norm.

```
fprintf('Problem 2\n')
% (a)
xinf2 = norm(x, Inf)
% (b)
x1norm2 = norm(x,1)
% (c)
x2norm2 = norm(x,2)
```

```
% (d)
x3norm3 = norm(x,3)
```

```
Problem 2

xinf2 =

     5


x1norm2 =

    59


x2norm2 =

   14.1774


x3norm3 =

    9.1057
```

# Problem 3

The following linear systems Ax = b have x as the actual solution and xtilde as an approximate solution. Compute |`x ? xtilde`|_? and |`Axtilde ? b`|_?.

```
fprintf('Problem 3\n')
% (a)
x = [1/7, -1/6]';
xtilde = [0.142, -0.166]';

diff1 = norm(x-xtilde, Inf)

A = [1/2, 1/3;
     1/3, 1/4];

b = [1/63, 1/168]';

diff2 = norm(A*xtilde - b, Inf)

% (b)
x = [1.827586, 0.6551724, 1.965517]';
xtilde = [1.8, 0.64, 1.9]';

diff1 = norm(x-xtilde, Inf)

A = [0.04, 0.01, -0.01;
     0.2 , 0.5 , -0.2 ;
     1   , 2   ,  4   ];
```

```
b = [0.06, 0.3, 11]';

diff2 = norm(A*xtilde - b, Inf)
```

```
Problem 3

diff1 =

   8.5714e-04


diff2 =

   2.0635e-04


diff1 =

    0.0655


diff2 =

    0.3200
```

# Problem 4

Find the eigenvalues and corresponding eigenvectors for

```
A = [1 ,0;
     0, 2];

%(a) by hand
%       det(A-lam*I) = 0
% => (1-lam)(2-lam) = 0
% => lam in {1,2}
% => If lam = 1 then eigVec1 = [1,0]'
%    If lam = 2 then eigVec2 = [0,1]'

[V, D] = eig(A)
```

```

V =

     1     0
     0     1


D =

     1     0
     0     2
```

## Problem 5

Repeat Problem 4 for

```
A = [ 0, 1;
      1, 0];

%(a) by hand
%          det(A-lam*I) = 0
% => (0-lam)(0-lam)-1*1 = 0
% =>        lam^2 - 1^2 = 0
% =>     (lam-1)(lam+1) = 0
% => lam in {-1,+1}
% => If lam = -1 then
    %       (A-lam*I)x = 0
    %    => [ 1 1 | 0
    %         1 1 | 0 ]
    %    => [ 1 1 | 0
    %         0 0 | 0 ]
    %    => [ x1, x2 ]' = [ -p, p ]' where p in R. If p = 1 then
    %    => [ x1, x2 ]' = [ -1, 1 ]'
%    If lam =  1 then
%        (A-lam*I)x = 0
    %    => [ -1  1 | 0
    %          1 -1 | 0 ]
    %    => [ -1  1 | 0
    %         -1  1 | 0 ]
    %    => [  1 -1 | 0
    %          0  0 | 0 ]
    %    => [ x1, x2 ]' = [ p, p ]' where p in R. If p = 1 then
    %    => [ x1, x2 ]' = [ 1, 1 ]'

[V, D] = eig(A)

% The entries in V are scalar multiples of the eignvectors I chose.


V =

   -0.7071    0.7071
    0.7071    0.7071


D =

    -1     0
     0     1
```

## Infinity norm

```
function norminf = Norminf(x)
norminf = max( abs(x) );
```

```
end
```

```
xinf =

     5
```

# 1-norm

```
function norm1 = Norm1(x)
norm1 = sum( abs(x) );
end
```

```
x1norm =

    59
```

# 2-norm

```
function norm2 = Norm2(x)
norm2 = ( sum( (abs(x)).^2 ) ) ^ (1/2);
end
```

```
x2norm =

   14.1774
```

# p-norm

```
function pnorm = Pnorm(x,p)
pnorm = ( sum( (abs(x)).^p ) ) ^ (1/p);
end
```

```
x3norm =

    9.1057
```

*Published with MATLAB® R2018b*