

Opppg 4

Ser at estimatet for p1 får ut en verdi med relativ error $\text{relerrp1} = 1.9182\text{e-}07$ som er under toleransen.

På p2 så er den relative erroren på $\text{relerrp2} = 1.1180$, dette er pga. at companion matrisen ikke har en dominant egenverdi. Vi ser med funksjonen `roots(p2)` at de største i absolutt verdi egenverdiene er $\pm 2i$. Dermed fungerer ikke Power Method for å finne egenverdien.

Kode:

```
%a = [2 3 4; 0 3 2; 0 0 5];
p1 = [1 3 -1 -3 -1 1]; %car pol1
p2 = [1 -1 4 -4]; %car pol2
rootp1Est = sdrot(p1) %estimated root1
rootp2Est = sdrot(p2) %estimated root2
rootp1 = max(roots(p1)); %assume that max(roots(p)) is exact value
rootp2 = max(roots(p2));
relerrp1 = abs(rootp1-rootp1Est)/abs(rootp1)
relerrp2 = abs(rootp2-rootp2Est)/abs(rootp2)
```

```
function rot = sdrot(p)
    %making the companion-matrix
    numtimes = 100;
    tol = 1e-6;
    n = length(p)-1;
    x = rand(n,1); %random start vector, x =
zeros(n,1); x(end) = 1;
    A = zeros(n,n);
    for i = 1:n
        A(end,i) = -p(n+2-i);
        if i > 1
            A(i-1,i) = 1; %inserting ones on the shifted
diagonal
        end
    end
    %powermethod
    xvals = [];
    muvals = [];
```

```

for r=1:numtimes
    x = A*x;
    [maxval,maxnr]=max(abs(x));
    mu = x(maxnr);
    x = (1/mu)*x;
    % Kunne her brukt R = x'*A*x/(x'*x) i stedet for mu
    muvals = [muvals mu];
    xvals = [xvals x];
    error = max(abs(A*x-mu*x));
    if error<tol
        rot = muvals(end);%adding the last estimate to
root
        break;
    end

end
if error>tol
    sprintf('error = %f > %f = tol, mu = %f ', [error,
tol, muvals(end)])%error message
    rot = muvals(end);%%adding the last estimate to
root
end

    %sprintf('loop ended without giving value with error
less than %f, lambda = %f', [tol, muvals(end)])
End

```

Kjøreeksempel:

```
>> oppg4
```

```
rootp1Est =
```

```
-3.0523
```

```
ans =
```

```
'error = 3.672121 > 0.000001 = tol, mu = 1.000000 '
```

rootp2Est =

1

relerrp1 =

1.9182e-07

relerrp2 =

1.1180

Opppg5

I matrisen a har vi en stor dominans på den dominante egenverdien. Dermed får vi kjapt et godt estimat. Estimattet har en relative error på relerror = 5.3235e-10, Sammenlignet med eig(A)

Kode:

```
A = [1 1 1 1 1 1
      1 2 3 4 5 6
      1 3 6 10 15 21
      1 4 10 20 35 56
      1 5 15 35 70 126
      1 6 21 56 126 252];
polyA = poly(A);%finder det karakteristiske polynomet til
A

rotEst = sdrot(polyA)
rot = max(eig(A))
```

```
relerror = (abs(rot-rotEst))/abs(rot)
```

```
function rot = sdrot(p)
    numtimes = 100;
    tol = 1e-6;
    n = length(p)-1;

    x = rand(n,1);

    A = zeros(n,n);
    for i = 1:n
        A(end,i) = -p(n+2-i);
        if i > 1
            A(i-1,i) = 1;
        end
    end

    xvals = [];
    muvals = [];
    for r=1:numtimes
        x = A*x;
        [maxval,maxnr]=max(abs(x));
        mu = x(maxnr);
        x = (1/mu)*x;
        % Kunne her brukt R = x'*A*x/(x'*x) i stedet for mu
        muvals = [muvals mu];
        xvals = [xvals x];
        error = max(abs(A*x-mu*x));
        if error<tol
            rot = muvals(end);
            break;
        end
    end

    if error>tol
        sprintf('error = %f > %f = tol, mu = %f ', [error,
tol, muvals(end)])
        rot = muvals(end);
    end

    %sprintf('loop ended without giving value with error
less than %f, lambda = %f', [tol, muvals(end)])
```

end

Kjøreeksempel:

>> oppg5

rotEst =

332.8463

rot =

332.8463

relerror =

5.3235e-10