Nicole Deyerl

Numerical Analysis

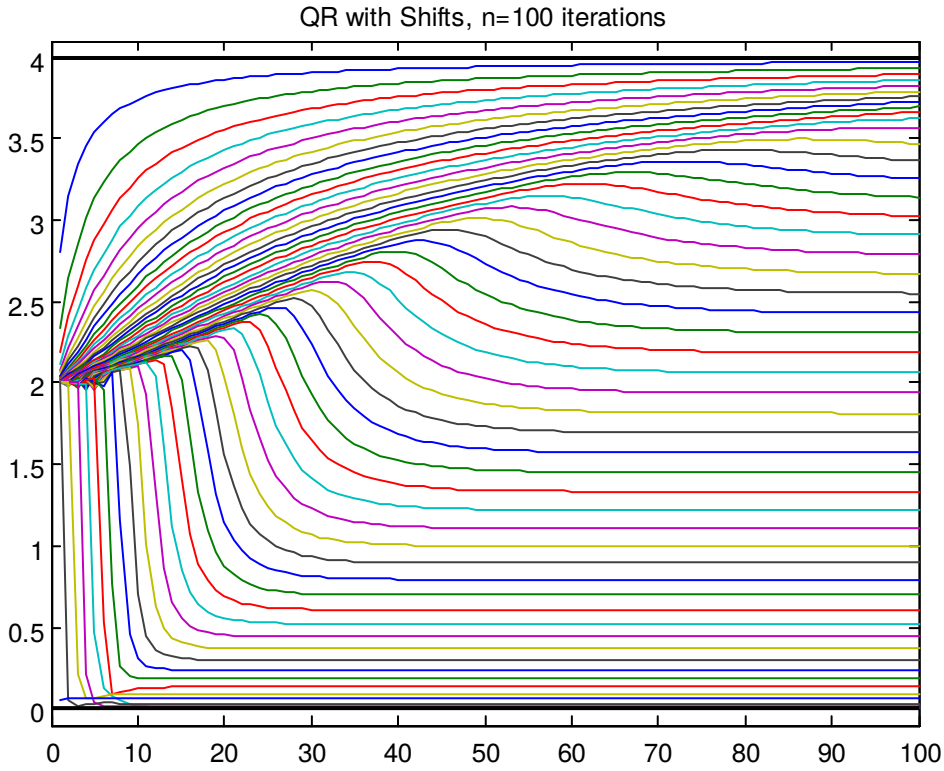
Project 3

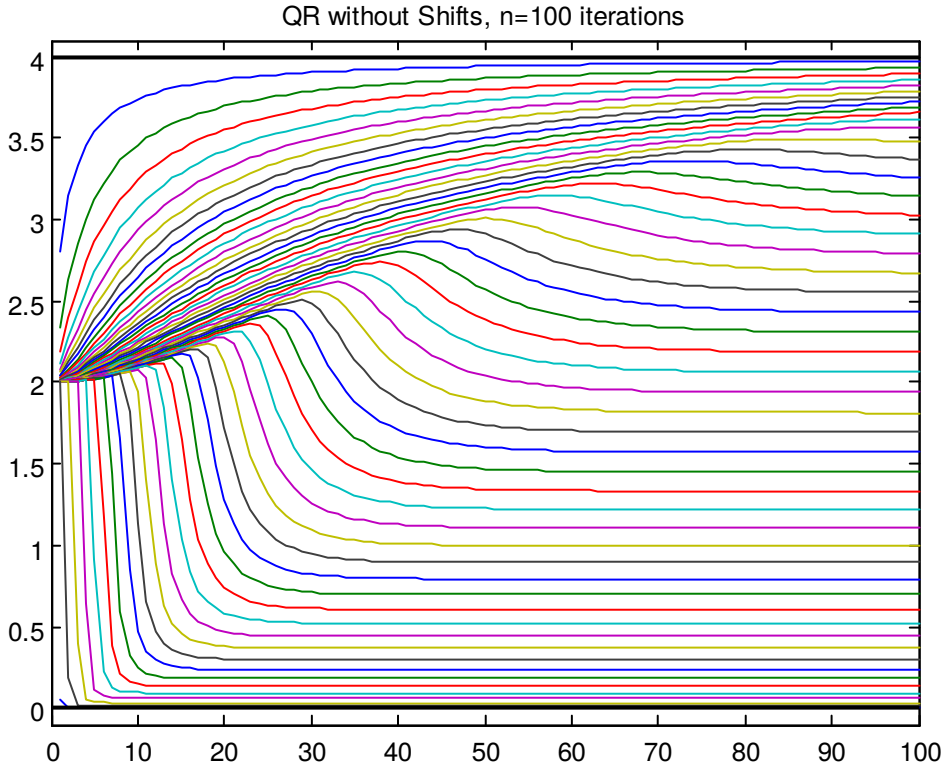
Problem 1

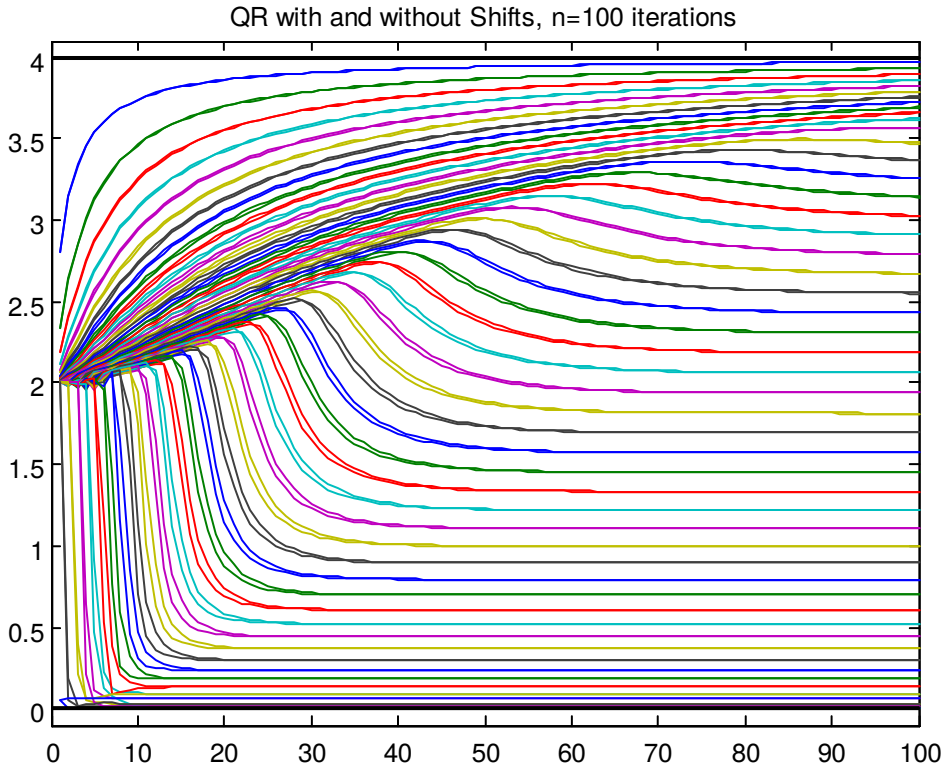
```
%with shifts
m=50;
a=[2,-1,zeros(1,m-2)];
A0=toeplitz(a);
A=A0;
n=100;
dVec=[];
sVec=[];
for k=1:n
    if k==1
    mu=0;
    [Q,R]=qr(A-mu*eye(m));
    A=R*Q+mu*eye(m);
    dVec(:,k)=diag(A);
    sVec(:,k)=diag(A,-1);
    else
    mu=A(m,m);
    [Q,R]=qr(A-mu*eye(m));
    A=R*Q+mu*eye(m);
    dVec(:,k)=diag(A);
    sVec(:,k)=diag(A,-1);
    end
end

%without shifts
m=50;
a=[2,-1,zeros(1,m-2)];
A0=toeplitz(a);
A=A0;
n=100;
dVec=[];
sVec=[];
for k=1:n
    mu=0;
    [Q,R]=qr(A-mu*eye(m));
    A=R*Q+mu*eye(m);
    dVec(:,k)=diag(A);
    sVec(:,k)=diag(A,-1);
end
```

QR	n=50 iterations	n=100 iterations	n=1000 iterations
Shifts: max eigenval	3.94260531	3.97088411	3.9960689
No Shifts: max eigenval	3.94174757	3.97044335	3.9960565
Shifts: min eigenval	0.003793343	0.003793343	0.00379334
No Shifts: min eigenval	0.003793343	0.003793343	0.00379334

Matlab built-in	max	min
Eig(A)	3.996206657	0.00379334


QR with Shifts, n=100 iterations
4
3.5
3
2.5
2
1.5
1
0.5
0
0
10
20
30
40
50
60
70
80
90
100


QR without Shifts, n=100 iterations
4
3.5
3
2.5
2
1.5
1
0.5
0
0
10
20
30
40
50
60
70
80
90
100

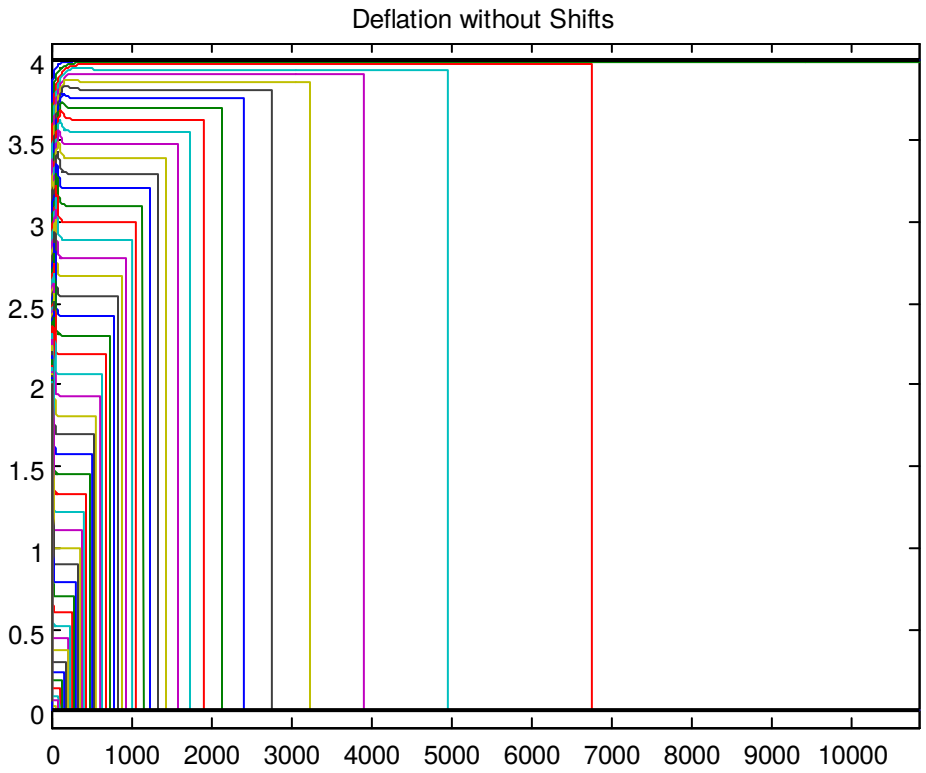

QR with and without Shifts, n=100 iterations
4
3.5
3
2.5
2
1.5
1
0.5
0
0
10
20
30
40
50
60
70
80
90
100

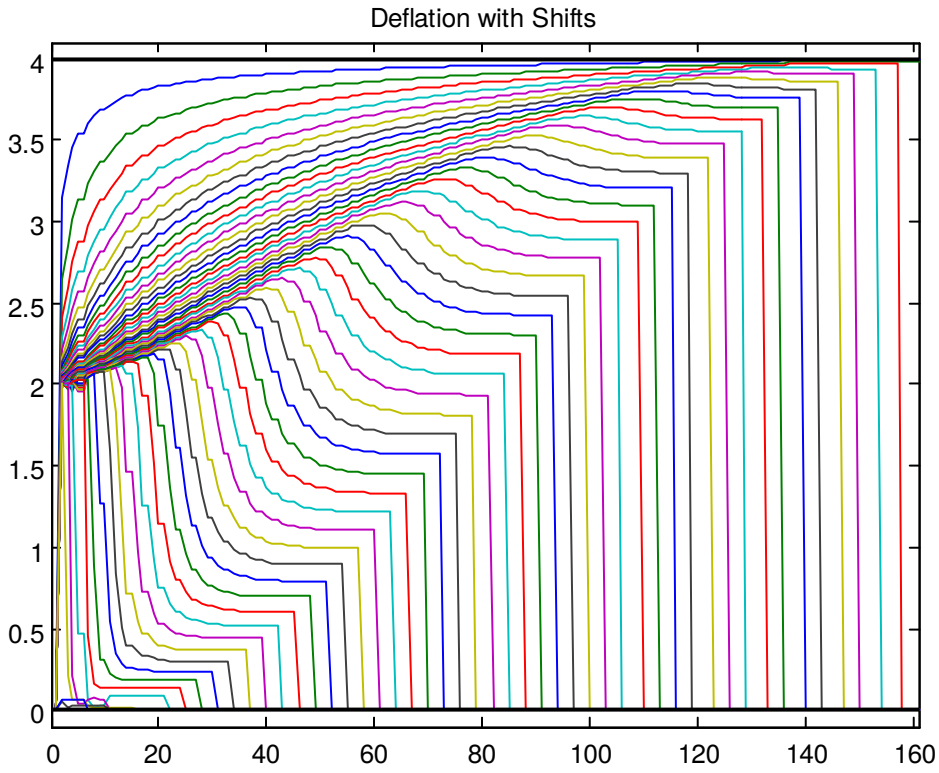
Problem 2

```
%with shifts
m=50;
a=[2,-1,zeros(1,m-2)];
A0=toeplitz(a);
A=A0;
dVec=[];
sVec=[];
tol=1e-15; %epsilon machine
counter=0;
while m>1
        if counter==0
            mu=0; %first iteration, need to do mu=0
            [Q,R]=qr(A-mu*eye(m));
            A=R*Q+mu*eye(m);
            counter=counter+1; %add one to exit this loop
        else
            if abs(A(m,m-1))>tol
                mu=A(m,m);
                [Q,R]=qr(A-mu*eye(m));
                A=R*Q+mu*eye(m);
                counter=counter+1;
                dVec(1:m,counter)=diag(A);
                sVec(1:m-1,counter)=diag(A,-1);
            else %the A(m,m-1) entry is below tolerance here
                counter=counter+1; %add one to make space in dVec/sVec
                dVec(1:m,counter)=diag(A);
                sVec(1:m-1,counter)=diag(A,-1);
                m=m-1;
                A=A(1:m,1:m);
            end
        end
end

%without shifts
m=50;
a=[2,-1,zeros(1,m-2)];
A0=toeplitz(a);
A=A0;
dVec=[];
sVec=[];
tol=1e-15; %epsilon machine
counter=0;
while m>1
            if abs(A(m,m-1))>tol
                mu=0;
                [Q,R]=qr(A-mu*eye(m));
                A=R*Q+mu*eye(m);
                counter=counter+1;
                dVec(1:m,counter)=diag(A);
                sVec(1:m-1,counter)=diag(A,-1);
            else %the A(m,m-1) entry is below tolerance here
                counter=counter+1; %add one to make space in dVec/sVec
                dVec(1:m,counter)=diag(A);
                sVec(1:m-1,counter)=diag(A,-1);
                m=m-1;
                A=A(1:m,1:m);
            end
end
```

Deflation Method	Number of iterations	Max eigenvalue	Min eigenvalue	Final size of A
Shifts	161	3.996206657474089	0.003793342525912	1x1
No Shifts	10847	3.996206657474071	0.003793342525912	1x1

Note: using the deflation method for no shifts, 4000 of the final iterations (6000 to 10000) were to deflate A from a 3x3 to a 1x1 with no noticeable change of the eigenvalues, 3000 of the iterations before that were used to deflate A from a 4x4 to a 3x3 with a change of 1*10-14.


Deflation without Shifts
4
3.5
3
2.5
2
1.5
1
0.5
0
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000


Deflation with Shifts
4
3.5
3
2.5
2
1.5
1
0.5
0
0
20
40
60
80
100
120
140
160