EXAMPLE 6 It can be verified that the matrix

$$A = \begin{bmatrix} .8 & .5 \\ -.1 & 1.0 \end{bmatrix}$$

has eigenvalues $.9 \pm .2i$, with eigenvectors $\begin{bmatrix} 1 \mp 2i \\ 1 \end{bmatrix}$. Figure 5 shows three trajectories of the system $\mathbf{x}_{k+1} = A\mathbf{x}_k$, with initial vectors $\begin{bmatrix} 0 \\ 2.5 \end{bmatrix}$, $\begin{bmatrix} 3 \\ 0 \end{bmatrix}$, and $\begin{bmatrix} 0 \\ -2.5 \end{bmatrix}$. ■

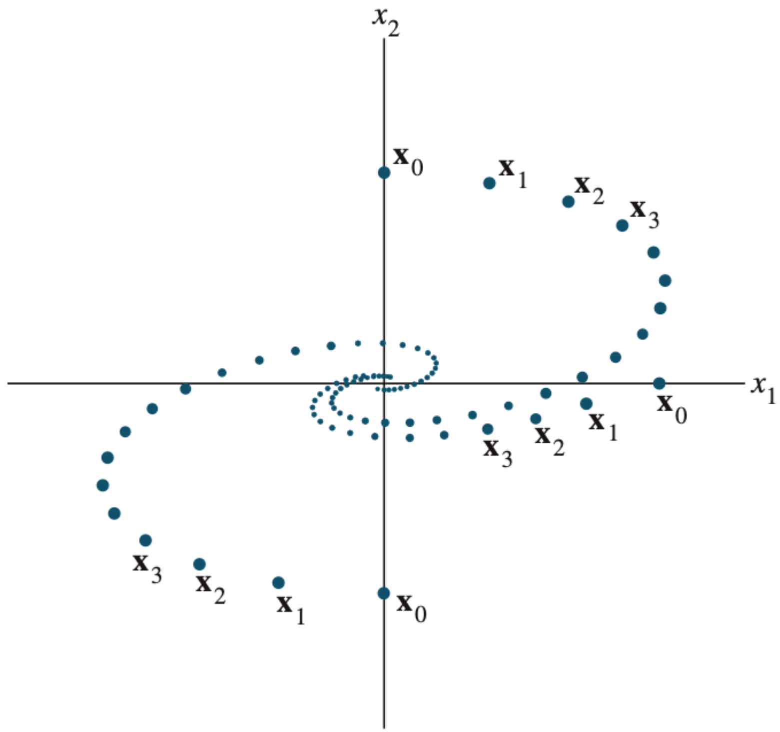
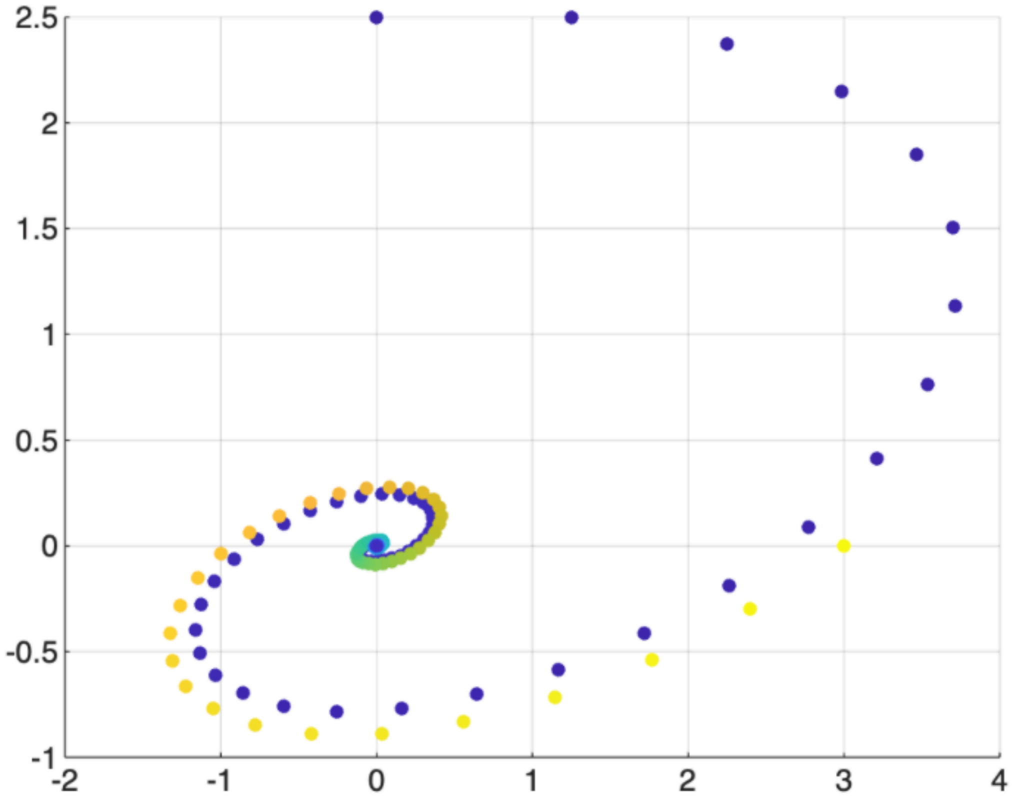


FIGURE 5 Rotation associated with complex eigenvalues.

```
clear
A=[0.8 0.5; -0.1 1];
[v,D]=eig(A);
x0=[0; 2.5];
x1=[3; 0];

 for k=1:1:100
     y1(:,:,k)=(A^(k-1))*x0;
     z1(k)=y1(1,1,k);
     z2(k)=y1(2,1,k);
     y2(:,:,k)=(A^(k-1))*x1;
     w1(k)=y2(1,1,k);
     w2(k)=y2(2,1,k);
 end

sz = 20;
c = linspace(1,10,length(z1));
r = linspace(20,5,length(w1));
scatter(z1,z2,sz,c,'filled')
hold on
scatter(w1,w2,sz,r,'filled')
```



Final Project: deadline 2025.6.13 FRI. pm12:00

Please use MATLAB to simulate the Moving Average Filter.

Consider the discrete-time signal $x[n]$:

$$x[n] = (1.02)^n + \frac{1}{2}\cos(\frac{2\pi n}{8} + \frac{\pi}{4})$$

(a) Plotting the Discrete-Time Signal:

generate the discrete-time signal $x[n]$ in MATLAB for $0 \le n \le 40$.

(b) Implementing Moving Average Filters:

3-point moving average: $y_3[n] = \frac{1}{3}(x[n-1] + x[n] + x[n+1])$

7-point moving average:

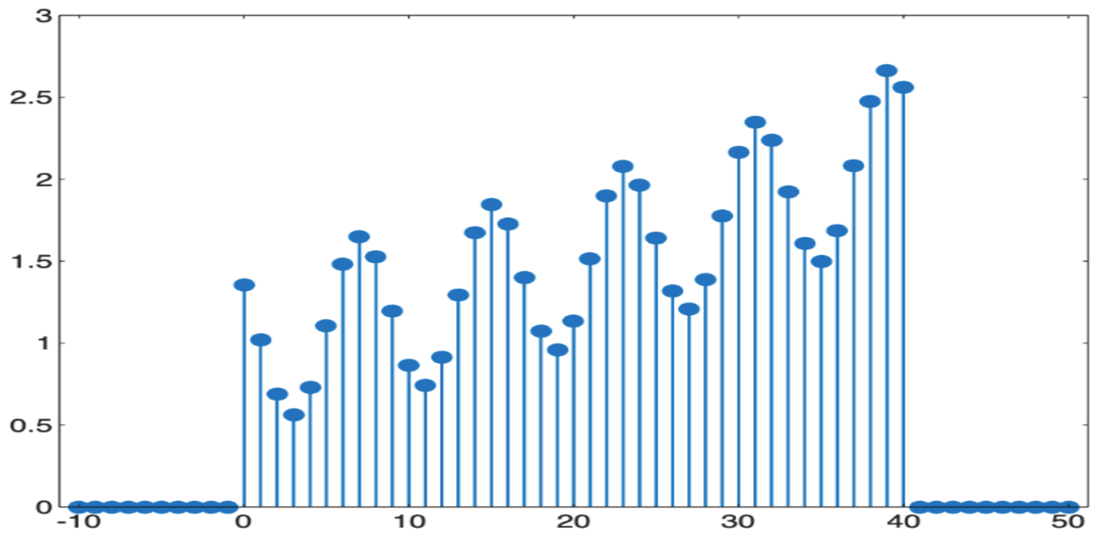
$$y_7[n] = \frac{1}{7}(x[n-3] + x[n-2] + x[n-1] + x[n] + x[n+1] + x[n+2] + x[n+3])$$

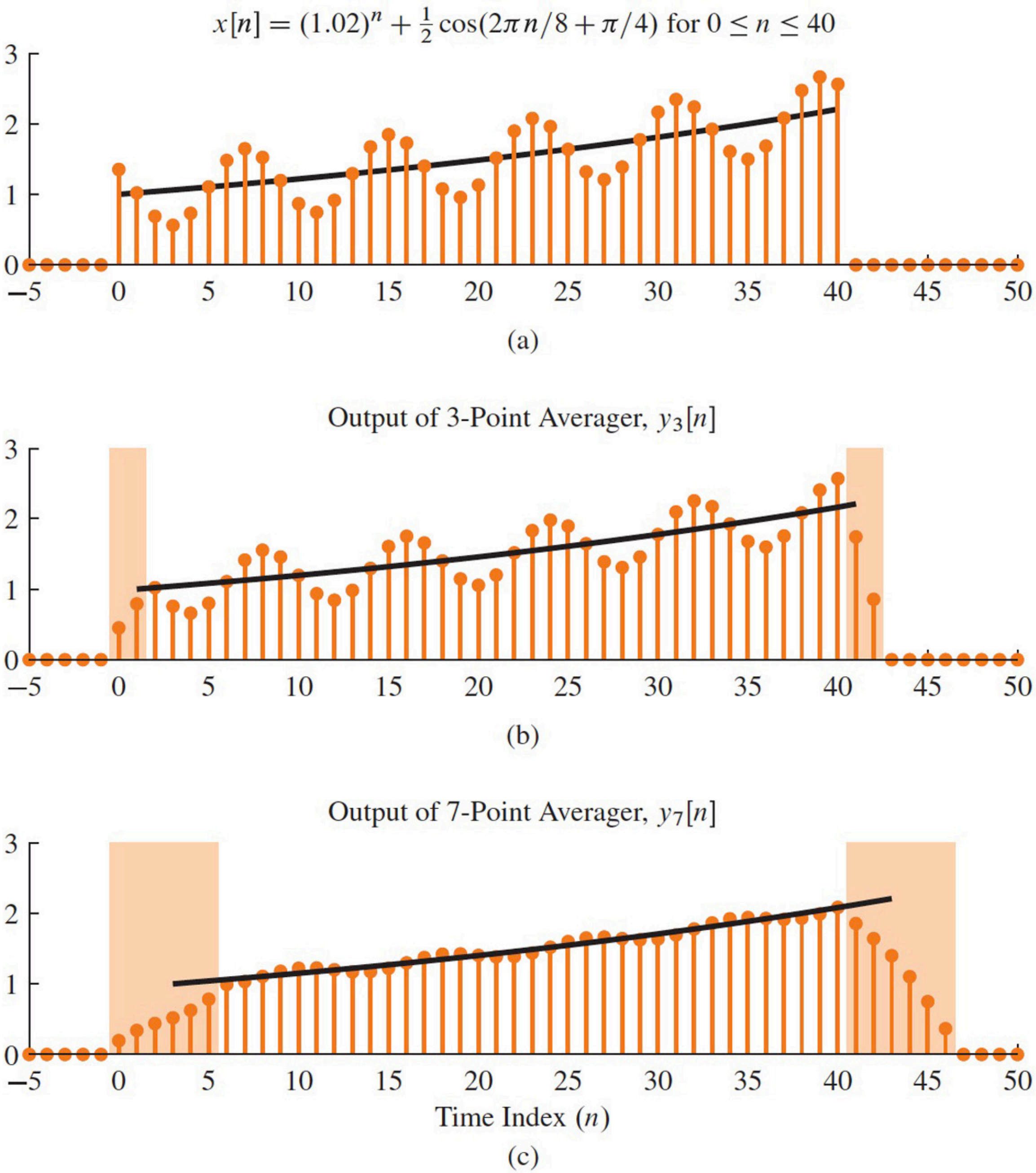
(c) Comparing Simulation Results:

After plotting the original and filtered signals, compare them and describe the effect of the moving average filter, particularly how the window size (3-point vs. 7-point) influences the filtering.

```
clear
for n=1:1:61
   N(n)=n-11;
    if (N(n)>40) | (N(n)<0)
        x(n)=0;
    else
        x(n)=((1.02)^N(n))+0.5*cos(2*pi*N(n)/8+pi/4);
    end

end
stem(N,x,'filled',LineWidth=1)
```




$x[n] = (1.02)^n + \frac{1}{2}\cos(2\pi n/8 + \pi/4)$ for $0 \le n \le 40$
3
2
1
0
−5 0 5 10 15 20 25 30 35 40 45 50
(a)
Output of 3-Point Averager, $y_3[n]$
3
2
1
0
−5 0 5 10 15 20 25 30 35 40 45 50
(b)
Output of 7-Point Averager, $y_7[n]$
3
2
1
0
−5 0 5 10 15 20 25 30 35 40 45 50
Time Index (n)
(c)