Digital Spectral Analysis Project 1

Abinaya Subramaniam

October 17 2021

1 Introduction

In this project, we take up an ARMA(2,2) process and implement various power spectral density estimation methods. The ARMA(2,2) considered is

$$y(t) = 1.3435y(t-1) - 0.9025y(t-2) + e(t) + 1.3435e(t-1) + 0.9025e(t-2) \tag{1}$$

The arbitrary rational PSD can be associated with a signal obtained by fltering white noise of power $\sigma.^2$ through the rational filter with transfer function H(w) = B(w)/A(w). A signal y(t) satisfying the equation (1) is called an autoregressive moving average (ARMA or ARMA(n,m)) signal. If m = 0 in (3.2.6), then y(t) is an autoregressive (AR or AR(n)) signal; and y(t) is a moving average (MA or MA(m)) signal if n = 0.

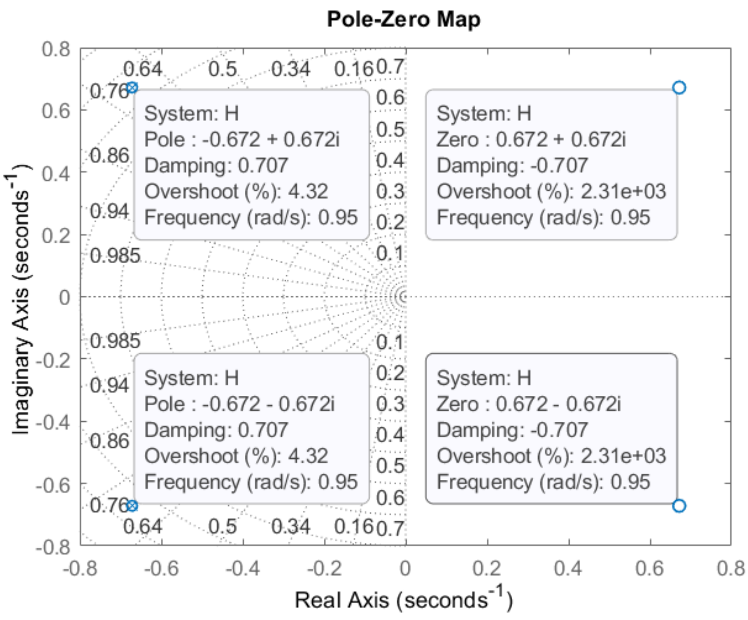

Figure 1: Pole zero map of given ARMA proces

Figure 1 shows the pole-zero map of the ARMA process.Poles are (-0.672+0.672i,-0.672-0.672i). Zeros are (0.672+0.672i,0.672-0.672i). Since both poles are inside the unit circle, the system is stable and since both zeros are inside the unit circle, it is at minimum phase.

2 True Power Spectral Density

True power spectral density is calculated using the following equation

$$\phi(w) = |\frac{B(w)}{A(w)}|^2 \sigma^2 \tag{2}$$

where

$$A(w) = 1 - 1.3435e^{-iw} + 0.9025e^{-2iw} \tag{3}$$

$$B(w) = 1 + 1.3435e^{-iw} + 0.9025e^{-2iw} \tag{4}$$

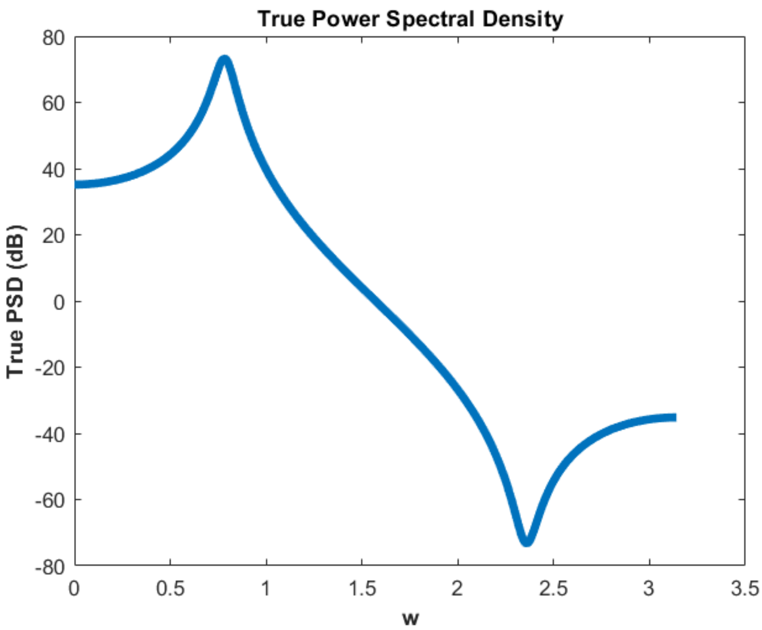

Figure 2: True power spectral density

3 Non-parametric Methods to estimate PSD

3.1 Periodogram

Periodogram is calculated as follows

$$\phi_p(w) = \frac{1}{N} | \sum_{t=1}^{N} y(t) e^{-iwt} |^2 \quad (5)$$

Figure 3 shows periodogram estimate for 3 different values of N

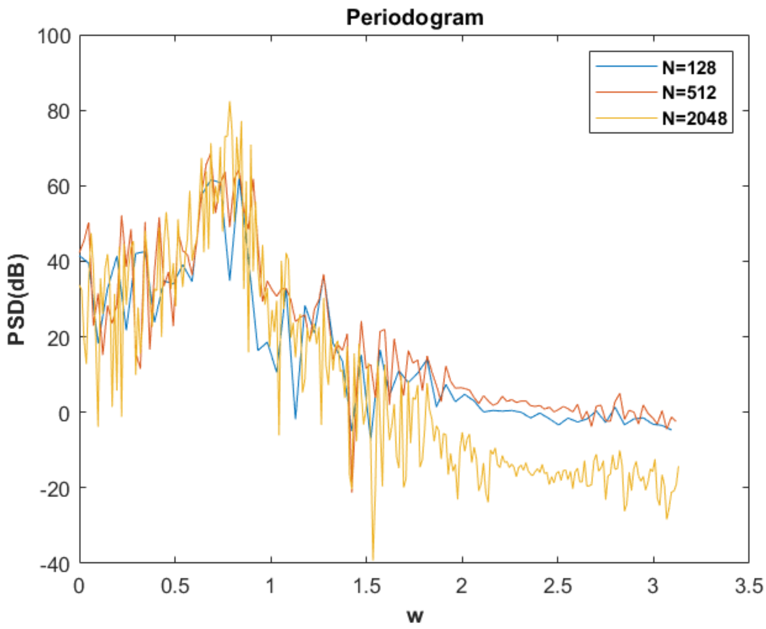

Figure 3: Periodogram estimate

3.2 Bartlett Periogram

Mathematically, the idea behind Bartlett estimate is

$$y_j(t) = y((j-1)M + t), \qquad t \ = 1, \ldots, M$$

$j = 1, \ldots, L$ denote the observations of the j th subsample, and let

$$\hat{\phi}_j(\omega) = \frac{1}{M} \left| \sum_{t=1}^{M} y_j(t) e^{-i\omega t} \right|^2$$

denote the corresponding periodogram. The Bartlett spectral estimate is then given by

$$\hat{\phi}_B(\omega) = \frac{1}{L} \sum_{j=1}^{L} \hat{\phi}_j(\omega)$$

In figure 4, we can see that as M increases, the PSD becomes noisier. This can be reasoned easily. Resolution is of the order 1/M. As M increases, resolution decreases. As resolution decreases, variance increases. So this implies that PSD becomes noisier as M increases. Since the variance increases with M, bias reduces according to the bias-variance tradeoff.

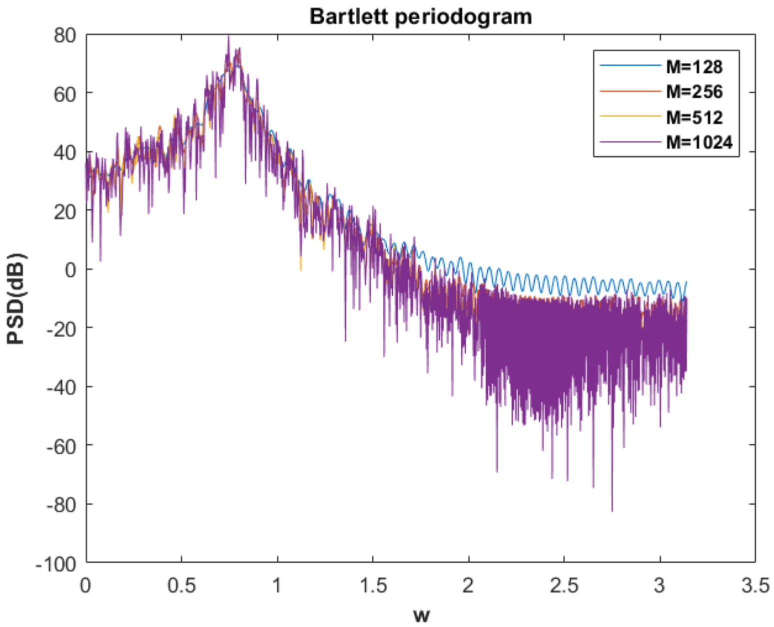

Figure 4: Bartlett Periodogram

3.3 Welch periodogram

Mathematically, Welch method can be described as

$$y_j(t) = y((j-1)K + t), \qquad \begin{matrix} t = 1, \ldots, M \\ j = 1, \ldots, S \end{matrix}$$

denote the j th data segment. $(j-1)K$ is the starting point for the j th sequence of observations. If $K = M$, then the sequences do not overlap (but are contiguous) and we get the sample splitting used by the Bartlett method (which leads to $S = L = N/M$ data subsamples). However, the value recommended for K in the Welch method is $K = M/2$, in which case $S \simeq 2M/N$ data segments (with 50% overlap between successive segments) are obtained. The windowed periodogram corresponding to $y_j(t)$ is computed as

$$\hat{\phi}_j(\omega) = \frac{1}{MP} \left| \sum_{t=1}^{M} v(t) y_j(t) e^{-i\omega t} \right|^2$$

where P denotes the "power" of the temporal window $\{v(t)\}$:

$$P = \frac{1}{M} \sum_{t=1}^{M} |v(t)|^2$$

The Welch estimate of PSD is determined by averaging the windowed periodograms in:

$$\hat{\phi}_W(\omega) = \frac{1}{S}\sum_{j=1}^{S}\hat{\phi}_j(\omega)$$

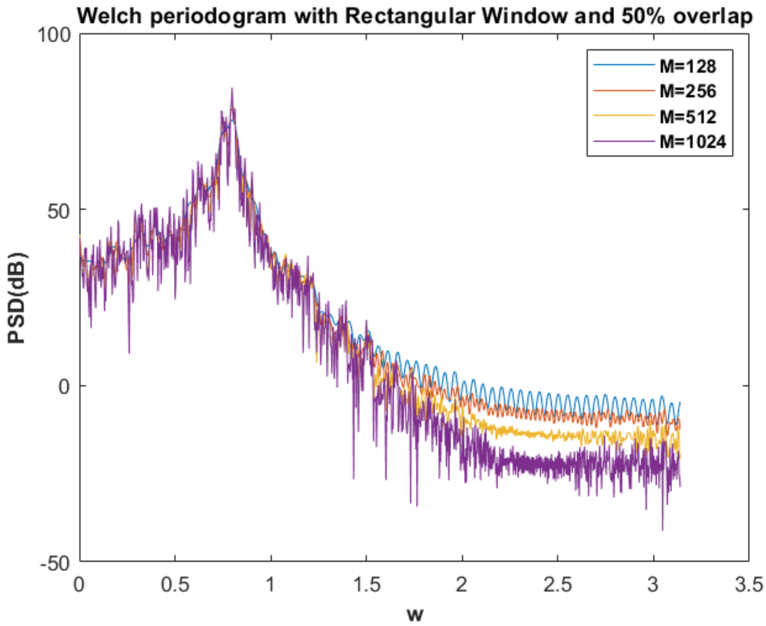

Figure 5: Welch periodogram with Rectangular Window

In figure 6 we can see that here is not much shifting of PSD across different values of M. The use of windowed periodograms in the Welch method, as contrasted to the unwindowed periodograms in the Bartlett method, indeed offers more fexibility in controlling the bias properties of the estimated spectrum. By allowing overlap between the data segments and hence by getting more periodograms to be averaged, we hope to decrease the variance of the estimated PSD. Additionally, the temporal window may be used to give less weight to the data samples at the ends of each subsample, hence making the consecutive subsample sequences less correlated to one another, even though they are overlapping. The principal effect of this decorrelation should be a more effective reduction of variance via the averaging.

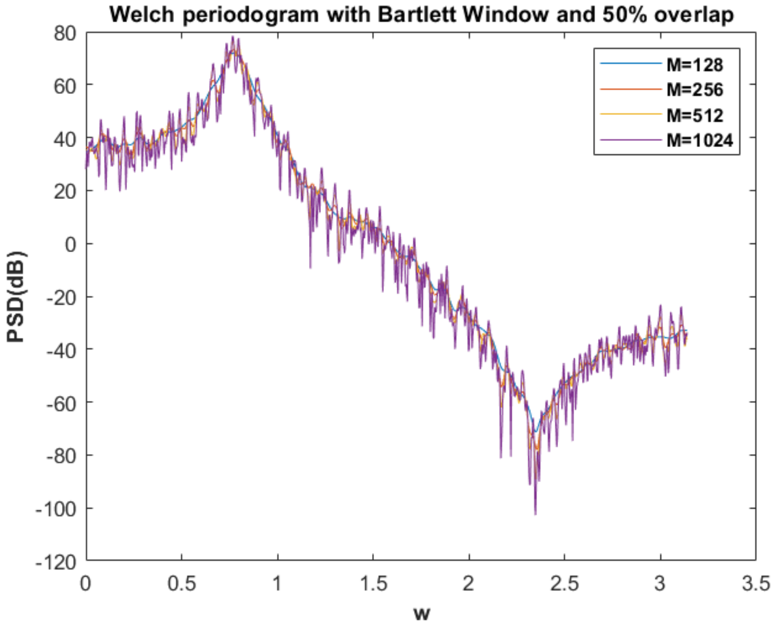

Figure 6: Welch periodogram with Bartlett Window

3.4 Blackman Tukey estimation

In Figure 7, the Blackman Tukey estimate for different values of M is shown. It is clearly seen that this is almost same as the Barlett estimate in figure 4. But here PSD becomes noisier as M decreases i.e variance decreases with increasing M and bias increases. This is because Bartlett method matches a Blackman Tuckey estimate with rectangular window of length 1/LM.

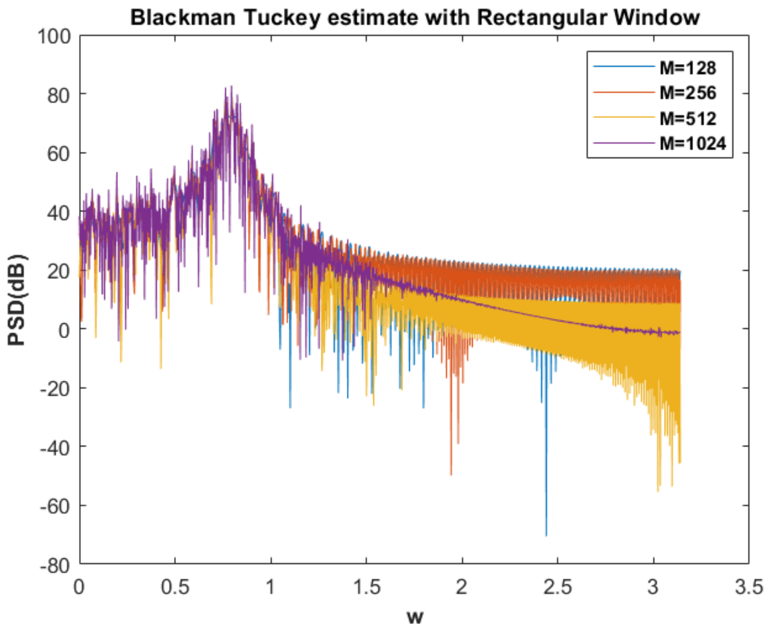

Figure 7: Blackman Tukey with rectangular window

$$\hat{\phi}_B(\omega) = \frac{1}{L}\sum_{j=1}^{L}\hat{\phi}_j(\omega)$$

where

$$\hat{\phi}_j(\omega) = \frac{1}{M}\left|\sum_{t=1}^{M} y_j(t)e^{-i\omega t}\right|^2 = \sum_{k=-M+1}^{M-1}\hat{r}_j(k)e^{-i\omega k}$$

with

$$\hat{r}_j(k) = \frac{1}{M}\sum_{t=k+1}^{M} y_j(t)y_j^*(t-k), \quad k \geq 0$$

and with $\hat{r}_j(-k) = \hat{r}_j^*(k)$. Thus

$$\hat{\phi}_B(\omega) = \frac{1}{L}\sum_{j=1}^{L}\sum_{k=-M+1}^{M-1}\hat{r}_j(k)e^{-i\omega k} = \sum_{k=-M+1}^{M-1}\underbrace{\left(\frac{1}{L}\sum_{j=1}^{L}\hat{r}_j(k)\right)}_{\tilde{r}(k)}e^{-i\omega k}$$

So, for $k \geq 0$

$$
\begin{aligned}
\tilde{r}(k) &= \frac{1}{L}\sum_{j=1}^{L}\frac{1}{M}\sum_{t=k+1}^{M} y_j(t)y_j^*(t-k) \\
&= \frac{1}{LM}\sum_{j=1}^{L}\sum_{t=k+1}^{M} y((j-1)M+t)y^*((j-1)M+t-k) \\
&= \sum_{t=k+1}^{N} \alpha(k,t)y(t)y^*(t-k)
\end{aligned}
$$

where

$$
\alpha(k,t) = \begin{cases} 1/N, & t \in [rM+k+1,(r+1)M], \quad ,r = 0,\ldots,L-1 \\ 0, & \text{otherwise} \end{cases}
$$

This shows that Bartlett estimate is equivalent to Blackman Tukey estimate with a rectangular window of length 1/N

4 Parametric Methods to estimate PSD

In figure 8, the modified Yule Walker estimate for the ARMA(2,2) process is given. Comparing this figure to true PSD that is shown in figure 2, we can say that peak estimation is better for M=1024 and valley estimation is better for M=128. For a higher value of M, AR estimates are more accurate and for a lower value of M, MA estimates are more accurate.

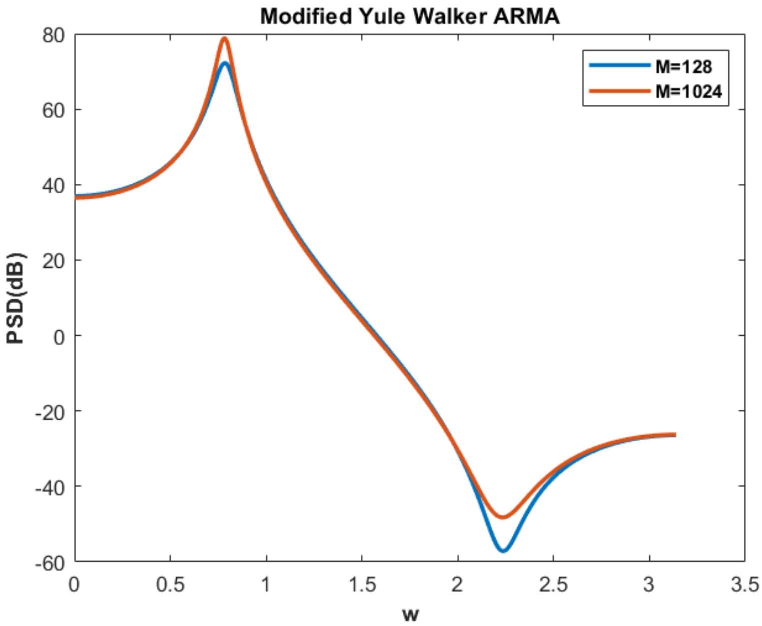

Figure 8: Modified Yule Walker estimate

In figure 9, the two-stage least squares method for ARMA(2,2) is shown. While peak detection is same for both values of K, valley detection is better for K=20. This implies that there is no change in AR parameter estimation for different values of K and MA parameter estimation is better for a higher value of K.

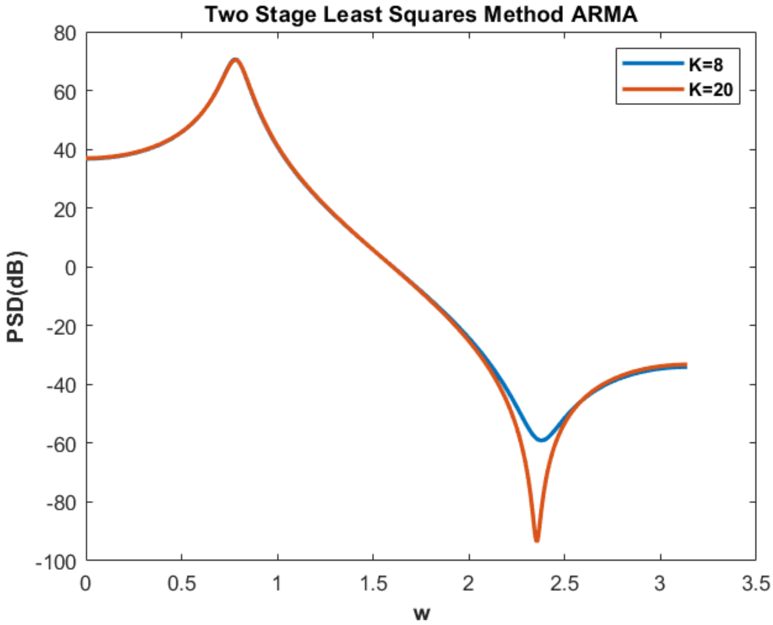

Figure 9: Two stage Least squares method for ARMA

If we assume ARMA as just AR, we equate all coefficients of MA to zero. So B(w) will be equal to 1. Consider this ARMA process as just AR, PSD is shown in figure 10. As expected, only the peaks (i.e) AR estimates have been detected and the MA estimates have not been detected. This a very poor estimation of PSD.

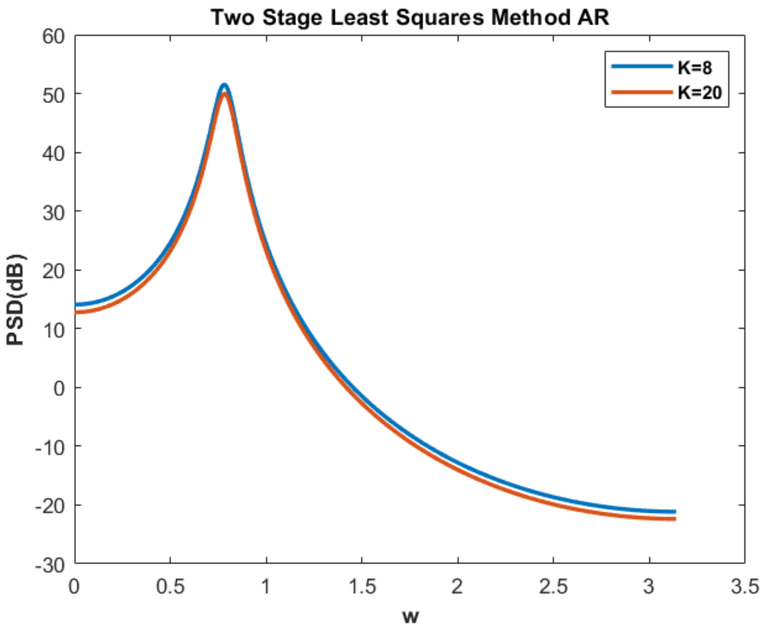

Figure 10: Two stage Least square method for AR

5 Appendix

5.1 Question 1 and 2

```
clc
clear all
close all

H = tf([1 -1.3435 0.9025],[1 1.3435 0.9025]);

figure(1)
pzmap(H)
grid on
a = [1 -1.3435 0.9025]; % AR coeffs
b = [1 1.3435 0.9025]; % MA coeffs


w = 0:0.01:pi; % frequencies to compute density
h = freqz(b,a,w); % returns frequency response
sd = abs(h).^2; % make into density

figure(2)
plot(w,10*log(sd),'LineWidth',4);
xlabel('w','fontweight','bold');ylabel('True PSD (dB)','fontweight','bold');
title('True Power Spectral Density')
set(gcf,'color','w');
```

5.2 Question 3 and 4

```
clc
clear all;
close all;

a = [1 -1.3435 0.9025]; % AR coeffs
b = [1 1.3435 0.9025]; % MA coeffs
T = 3000;
e=normrnd(0,1,[1,3000]);

y = filter(b,a,e); % generate y
L=[128, 256, 512, 1024];
y=y(953:3000);

v1=rectwin(L(1));
v2=rectwin(L(2));
v3=rectwin(L(3));

phi1=periodogramse(y,v1,L(1));
phi2=periodogramse(y,v2,L(2));
phi3=periodogramse(y,v3,L(3));


w1=0:2*pi/L(1):2*pi*(L(1)-1)/L(1);
w2=0:2*pi/L(2):2*pi*(L(2)-1)/L(2);
w3=0:2*pi/L(3):2*pi*(L(3)-1)/L(3);

w1=w1(1:L(1)/2);
w2=w2(1:L(2)/2);
w3=w3(1:L(3)/2);

phi1=phi1(1:L(1)/2);
phi2=phi2(1:L(2)/2);
phi3=phi3(1:L(3)/2);

figure(1)
plot(w1,10*log(phi1))
hold on
plot(w2,10*log(phi2))
hold on
plot(w3,10*log(phi3))

xlabel('w','fontweight','bold'); ylabel('PSD(dB)','fontweight','bold');
legend('N=128','N=512','N=2048','fontweight','bold');
title('Periodogram');
```

5.3 Question 5

```
clc
clear all;
close all;

a = [1 -1.3435 0.9025]; % AR coeffs
b = [1 1.3435 0.9025]; % MA coeffs
T = 3000;
e = randn(T,1); % generate gaussian white noise
y = filter(b,a,e); % generate y

L=2048;
M=[128,256,512,1024];
y=y(953:3000);
v=rectwin(L);
phi1=bartlettse(y,M(1),L);
phi2=bartlettse(y,M(2),L);
phi3=bartlettse(y,M(3),L);
phi4=bartlettse(y,M(4),L);

w=0:2*pi/L:2*pi*(L-1)/L;
w=w(1:L/2);
phi1=phi1(1:L/2);
phi2=phi2(1:L/2);
phi3=phi3(1:L/2);
phi4=phi4(1:L/2);

figure(1)
plot(w,10*log(phi1))
hold on
plot(w,10*log(phi2))
hold on
plot(w,10*log(phi3))
hold on
plot(w,10*log(phi4))
xlabel('w','fontweight','bold'); ylabel('PSD(dB)','fontweight','bold');
legend('M=128','M=256','M=512','M=1024','fontweight','bold');
title('Bartlett periodogram');
```

5.4 Question 6

```
clc
clear all;
close all;

a = [1 -1.3435 0.9025]; % AR coeffs
b = [1 1.3435 0.9025]; % MA coeffs
T = 3000;
e = randn(T,1); % generate gaussian white noise
y = filter(b,a,e); % generate y
L=2048;
M=[128,256,512,1024];
y=y(953:3000);
% v1=rectwin(M(1));
% v2=rectwin(M(2));
% v3=rectwin(M(3));
% v4=rectwin(M(4));

v1=bartlett(M(1));
v2=bartlett(M(2));
v3=bartlett(M(3));
v4=bartlett(M(4));
K=M/2;
phi1=welchse(y,v1,K(1),L);
phi2=welchse(y,v2,K(2),L);
phi3=welchse(y,v3,K(3),L);
phi4=welchse(y,v4,K(4),L);

w=0:2*pi/L:2*pi*(L-1)/L;
w=w(1:L/2);
phi1=phi1(1:L/2);
phi2=phi2(1:L/2);
phi3=phi3(1:L/2);
phi4=phi4(1:L/2);

figure(1)
plot(w,10*log(phi1))
hold on
plot(w,10*log(phi2))
hold on
plot(w,10*log(phi3))
hold on
plot(w,10*log(phi4))
xlabel('w','fontweight','bold'); ylabel('PSD(dB)','fontweight','bold');
legend('M=128','M=256','M=512','M=1024','fontweight','bold');
title('Welch periodogram with Bartlett Window and 50% overlap');
```

5.5 Question 7

```
clc
clear all;
close all;


a = [1 -1.3435 0.9025]; % AR coeffs
b = [1 1.3435 0.9025]; % MA coeffs
T = 3000;
e = randn(T,1); % generate gaussian white noise
y = filter(b,a,e); % generate y

L=2048;
M=[128,256,512,1024];
y=y(953:3000);
v1=rectwin(M(1));
v2=rectwin(M(2));
v3=rectwin(M(3));
v4=rectwin(M(4));

phi1=btse(y,v1,L);
phi2=btse(y,v2,L);
phi3=btse(y,v3,L);
phi4=btse(y,v4,L);

w=0:2*pi/L:2*pi*(L-1)/L;
w=w(1:L/2);
phi1=phi1(1:L/2);
phi2=phi2(1:L/2);
phi3=phi3(1:L/2);
phi4=phi4(1:L/2);

figure(1)
plot(w,10*log(phi1))
hold on
plot(w,10*log(phi2))
hold on
plot(w,10*log(phi3))
hold on
plot(w,10*log(phi4))
xlabel('w','fontweight','bold'); ylabel('PSD(dB)','fontweight','bold');
legend('M=128','M=256','M=512','M=1024','fontweight','bold');
title('Blackman Tuckey estimate with Rectangular Window');
```

5.6 Question 8, 9 and 10

```
clc
clear all;
close all;

a = [1 -1.3435 0.9025]; % AR coeffs
b = [1 1.3435 0.9025]; % MA coeffs
T = 3000;
e = randn(T,1); % generate gaussian white noise
y = filter(b,a,e); % generate y
L=2048;
M=[128,1024];
y=y(953:3000);
w=0:2*pi/L:2*pi*(L-1)/L;
[a1,gamma1]=mywarma(y,2,2,M(1));
[a2,gamma2]=mywarma(y,2,2,M(2));
for i=1:length(w)
    phi1(i)=(gamma1(1)+gamma1(2)*exp(-1i*w(i))+conj(gamma1(2))*exp(1i*w(i))+gamma1(3)*exp(-1i*2*w(i))+conj(gamma1(3))*exp(1i*2*w(i)))/abs(a1(1)+a1(2)*exp(-1i*w(i))+a1(3)*exp(-1i*2*w(i))).^2;
    phi2(i)=(gamma2(1)+gamma2(2)*exp(-1i*w(i))+conj(gamma2(2))*exp(1i*w(i))+gamma2(3)*exp(-1i*2*w(i))+conj(gamma2(3))*exp(1i*2*w(i)))/abs(a2(1)+a2(2)*exp(-1i*w(i))+a2(3)*exp(-1i*2*w(i))).^2;
end
w=w(1:L/2);
phi1=phi1(1:L/2);
phi2=phi2(1:L/2);

figure(1)
plot(w,10*log(phi1),'LineWidth',2)
hold on
plot(w,10*log(phi2),'LineWidth',2)
xlabel('w','fontweight','bold'); ylabel('PSD(dB)','fontweight','bold');
title('Modified Yule Walker ARMA')
legend('M=128','M=1024','fontweight','bold');

[a3,b3,sig3]=lsarma(y,2,2,8);
[a4,b4,sig4]=lsarma(y,2,2,20);

for i=1:length(w)
    phi3(i)=(abs(b3(1)+b3(2)*exp(-1i*w(i))+b3(3)*exp(-1i*2*w(i))).^2/abs(a3(1)+a3(2)*exp(-1i*w(i))+a3(3)*exp(-1i*2*w(i))).^2)*sig3;
    phi4(i)=(abs(b4(1)+b4(2)*exp(-1i*w(i))+b4(3)*exp(-1i*2*w(i))).^2/abs(a4(1)+a4(2)*exp(-1i*w(i))+a4(3)*exp(-1i*2*w(i))).^2)*sig4;
end

phi3=phi3(1:L/2);
phi4=phi4(1:L/2);

figure(2)
plot(w,10*log(phi3),'LineWidth',2)
hold on
plot(w,10*log(phi4),'LineWidth',2)
xlabel('w','fontweight','bold'); ylabel('PSD(dB)','fontweight','bold');
legend('K=8','K=20','fontweight','bold');
title('Two Stage Least Squares Method ARMA');

for i=1:length(w)
    phi5(i)=(1/abs(a3(1)+a3(2)*exp(-1i*w(i))+a3(3)*exp(-1i*2*w(i))).^2)*sig3;
```

```
    phi6(i)=(1/abs(a4(1)+a4(2)*exp(-1i*w(i))+a4(3)*exp(-1i*2*w(i))).^2)*sig4;
end

phi5=phi5(1:L/2);
phi6=phi6(1:L/2);

figure(3)
plot(w,10*log(phi5),'LineWidth',2)
hold on
plot(w,10*log(phi6),'LineWidth',2)
xlabel('w','fontweight','bold'); ylabel('PSD(dB)','fontweight','bold');
legend('K=8','K=20','fontweight','bold');
title('Two Stage Least Squares Method AR');
```