PN Logbook (Speech Signal Processing)

Demonstrator: Neo, Vincent

Name: Liu Jiaoyang

CID: 01867787

PN – Speech Signal Processing

Demonstrator: Neo, Vincent

Academic Supervisor: Dr. Ling

$\boxed{\text{26th October}}$

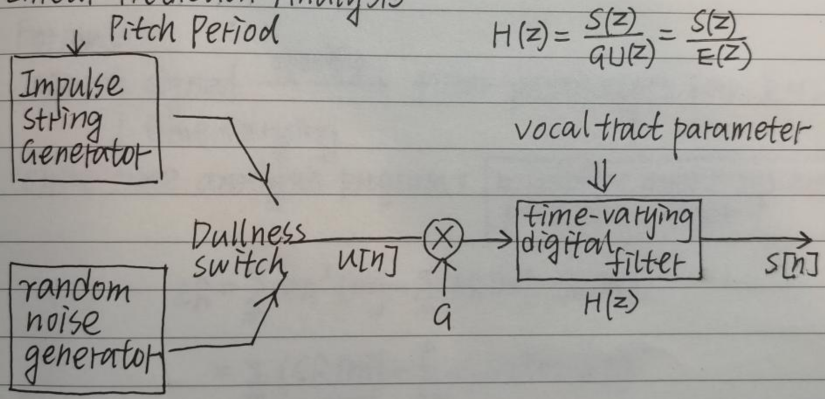
I. Aims

- Study the method of linear prediction
- Learn the Autocorrelation method

II. Principle

- Linear Prediction Analysis

$$H(z) = \frac{S(z)}{GU(z)} = \frac{S(z)}{E(z)}$$



$$s[n] = \sum_{k=1}^{p} a_k s[n-k] + G u[n]$$

$\{a_k, k=1,2,\cdots p\} \rightarrow$ system

input $s[n]$

output $\tilde{s}[n]$

$$\tilde{s}[n] = \sum_{k=1}^{p} \alpha_k s[n-k]$$

z-transform

Predictor Polynomial $\leftarrow P(z) = \sum_{k=1}^{p} \alpha_k z^{-k} = \frac{\tilde{S}(z)}{S(z)}$

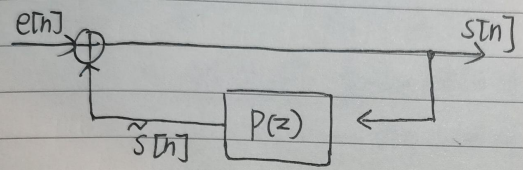
$$e[n] = s[n] - \tilde{s}[n] = s[n] - \sum_{k=1}^{p} \alpha_k s[n-k]$$

$$A(z)=\frac{E(z)}{S(z)}=1-P(z)$$

LPC Polynomial ←

if $\{a_k\}=\{\alpha_k\}$, then $e[n]=Gu[n] \Rightarrow A(z)=\frac{GU(z)}{S(z)}=\frac{1}{H(z)}$

$$s[n] \longrightarrow \boxed{A(z)} \xrightarrow{e[n]} \boxed{H(z)} \xrightarrow{s[n]}$$



• Formula

speech signal $\xrightarrow{\text{specify}}$ filter parameters $\{\alpha_k, k=1,2,\cdots p\}$

↓ time-varying ↑

short-time analysis program: $\boxed{\text{minimize mean square prediction error}}$

$$\mathcal{E}_{\hat{n}}=\sum_m e_{\hat{n}}^2[m]=\sum_m(s_{\hat{n}}[m]-\tilde{s}_{\hat{n}}[m])^2$$

$$=\sum_m\Big(\underbrace{s_{\hat{n}}[m]}_{\text{speech segment near } \hat{n}}-\sum_{k=1}^{p}\alpha_k s_{\hat{n}}[m-k]\Big)^2$$

speech segment near $\hat{n}$ i.e. $s_{\hat{n}}[m]=s[m+\hat{n}]$

$$\frac{\partial \mathcal{E}_{\hat{n}}}{\partial \alpha_i}=0,\ i=1,2,\cdots,p$$

$$\Downarrow$$

$$\sum_m s_{\hat{n}}[m-i]s_{\hat{n}}[m]=\sum_{k=1}^{p}\hat{\alpha}_k\sum_m s_{\hat{n}}[m-i]s_{\hat{n}}[m-k],\ 1\le i\le p$$

$\rightarrow \{\hat{\alpha}_k : \min(\mathcal{E}_{\hat{n}})\}$

define: $\varphi_{\hat{n}}[i,k]=\sum_m s_{\hat{n}}[m-i]s_{\hat{n}}[m-k]$

$$\Downarrow$$

$$\sum_{k=1}^{p}\alpha_k\varphi_{\hat{n}}[i,k]=\varphi_{\hat{n}}[i,0]$$

That is:
$$\begin{cases} \alpha_1 \phi_{\hat{n}}[1,1] + \alpha_2 \phi_{\hat{n}}[1,2] + \cdots + \alpha_p \phi_{\hat{n}}[1,p] = \phi_{\hat{n}}[1,0] \\ \alpha_1 \phi_{\hat{n}}[2,1] + \alpha_2 \phi_{\hat{n}}[2,2] + \cdots + \alpha_p \phi_{\hat{n}}[2,p] = \phi_{\hat{n}}[2,0] \\ \quad \vdots \qquad\qquad\qquad \vdots \qquad\qquad\qquad \vdots \\ \alpha_p \phi_{\hat{n}}[p,1] + \alpha_p \phi_{\hat{n}}[p,2] + \cdots + \alpha_p \phi_{\hat{n}}[p,p] = \phi_{\hat{n}}[p,0] \end{cases}$$

The least mean square prediction error:

$$\varepsilon_{\hat{n}} = \sum_m S_{\hat{n}}^2[m] - \sum_{k=1}^{p} \alpha_k \sum_m S_{\hat{n}}[m] S_{\hat{n}}[m-k]$$

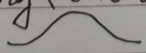
$$= \underbrace{\phi_{\hat{n}}[0,0]}_{\text{the sum of sample energy}} - \sum_{k=1}^{p} \alpha_k \phi_{\hat{n}}[0,k]$$

27th October

- Autocorrelation method

If: $S_{\hat{n}}[m] = S[m+n] \underbrace{w[m]}_{\text{Finite window function (Hamming)}}, \quad 0 \le m \le L-1$

$$\varepsilon_{\hat{n}} = \sum_{m=0}^{L-1+p} e_{\hat{n}}^2[m] = \sum_{m=-\infty}^{\infty} e_{\hat{n}}^2[m]$$

NB: At the beginning & end of the window, the error is relatively larger. So:  Hamming Window.

$\forall p, \ \varepsilon_{\hat{n}} > 0$

$$\phi_{\hat{n}}[i,k] = \sum_{m=0}^{L-1+p} S_{\hat{n}}[m-i] S_{\hat{n}}[m-k] \quad \begin{cases} 1 \le i \le p \\ 0 \le k \le p \end{cases}$$ can be represented

as: $$\phi_{\hat{n}}[i,k] = \sum_{m=0}^{L-1-(i-k)} S_{\hat{n}}[m] S_{\hat{n}}[m+(i-k)] \quad \begin{cases} 1 \le i \le p \\ 0 \le k \le p \end{cases}$$

$$\because \quad R_{\hat{n}}[k] = \sum_{m=0}^{L-1-k} S_{\hat{n}}[m] S_{\hat{n}}[m+k]$$

$$\Rightarrow \varphi_{\hat{n}}[i,k] = R_{\hat{n}}[i-k]$$

$$\therefore \sum_{k=1}^{p} \alpha_k R_{\hat{n}}[|i-k|] = R_{\hat{n}}[i], \quad 1 \le i \le p$$

$$\mathcal{E}_{\hat{n}} = R_{\hat{n}}[0] - \sum_{k=1}^{p} \alpha_k R_{\hat{n}}[k].$$

$$\begin{bmatrix} R_{\hat{n}}[0] & R_{\hat{n}}[1] & \cdots & R_{\hat{n}}[p-1] \\ R_{\hat{n}}[1] & R_{\hat{n}}[0] & \ddots & R_{\hat{n}}[p-2] \\ R_{\hat{n}}[2] & R_{\hat{n}}[1] & \ddots & R_{\hat{n}}[p-3] \\ \vdots & \vdots & \ddots & \vdots \\ R_{\hat{n}}[p] & R_{\hat{n}}[p-2] & \cdots & R_{\hat{n}}[0] \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_{\hat{n}}[1] \\ R_{\hat{n}}[2] \\ R_{\hat{n}}[3] \\ \vdots \\ R_{\hat{n}}[p] \end{bmatrix} \quad (*)$$

$P \times P$: symmetric and same diagnal

- Levinson-Durbin Algorithm

Because of symmetric, we have:

$$\sum_{k=1}^{p} \alpha_k R[|i-k|] = R[i], \quad 1 \le i \le p \qquad (1)$$

$$R\alpha = r$$

the least mean square prediction error: (p order)

$$R[0] - \sum_{k=1}^{p} \alpha_k R[k] = \mathcal{E}^{(p)}$$

add to (1):

$$\begin{bmatrix} R(0) & R(1) & \cdots & R[p] \\ R[1] & R[0] & \cdots & R[p-1] \\ \vdots & & & \vdots \\ R[p] & R[p-1] & \cdots & R[0] \end{bmatrix} \begin{bmatrix} 1 \\ -\alpha_1^{(p)} \\ \vdots \\ -\alpha_p^{(p)} \end{bmatrix} = \begin{bmatrix} \mathcal{E}^{(p)} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad p+1 \times p+1$$

for ith order:

$$R^{(i)} \alpha^{(i)} = e^{(i)} \leftarrow R^{(i-1)} \alpha^{(i-1)} = e^{(i-1)}$$

$i-1$:

$$\begin{bmatrix} R(0) & R[1] & R[2] & \cdots & R[i-1] \\ R[1] & R[2] & R[1] & \cdots & R[i-2] \\ \vdots & \vdots & \vdots & & \vdots \\ R[i-1] & R[i-2] & R[i-3] & \cdots & R[0] \end{bmatrix} \begin{bmatrix} 1 \\ -\alpha_1^{(i-1)} \\ \vdots \\ -\alpha_{i-1}^{(i-1)} \end{bmatrix} = \begin{bmatrix} \mathcal{E}^{(i-1)} \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

$i+1$:
$$\begin{bmatrix} R[0] & R[1] & \cdots & R[i] \\ R[1] & R[0] & \cdots & R[i-1] \\ \vdots & \vdots & \ddots & \vdots \\ R[i-1] & R[i-2] & \cdots & R[1] \\ R[i] & R[i-1] & \cdots & R[0] \end{bmatrix} \begin{bmatrix} 1 \\ -\alpha_1^{(i-1)} \\ \vdots \\ -\alpha_{i-1}^{(i-1)} \\ 0 \end{bmatrix} = \begin{bmatrix} \varepsilon^{(i-1)} \\ 0 \\ \vdots \\ 0 \\ \gamma^{(i-1)} \end{bmatrix} \qquad (2)$$

change the sequence

$$\gamma^{(i-1)} = \boxed{R[i]} - \sum_{j=1}^{i-1} \alpha_j^{(i-1)} R[i-j]$$

?

$$\begin{bmatrix} R[0] & R[1] & \cdots & R[i] \\ R[1] & R[0] & \cdots & R[i-1] \\ \vdots & \vdots & \ddots & \vdots \\ R[i-1] & R[i-2] & \cdots & R[1] \\ R[i] & R[i-1] & \cdots & R[0] \end{bmatrix} \begin{bmatrix} 0 \\ -\alpha_{i-1}^{(i-1)} \\ \vdots \\ -\alpha_1^{(i-1)} \\ 1 \end{bmatrix} = \begin{bmatrix} \gamma^{(i-1)} \\ 0 \\ \vdots \\ 0 \\ \varepsilon^{(i-1)} \end{bmatrix} \qquad (3)$$

From (2) and (3):

$$R[i] \left[\begin{bmatrix} 1 \\ -\alpha_1^{i-1} \\ \vdots \\ -\alpha_{i-1}^{i-1} \\ 0 \end{bmatrix} - k_i \begin{bmatrix} 0 \\ -\alpha_{i-1}^{i-1} \\ \vdots \\ -\alpha_1^{i-1} \\ 1 \end{bmatrix} \right] = \left[\begin{bmatrix} \varepsilon^{(i-1)} \\ 0 \\ \vdots \\ 0 \\ \boxed{\gamma^{(i-1)}} \end{bmatrix} - k_i \begin{bmatrix} \gamma^{(i-1)} \\ 0 \\ \vdots \\ 0 \\ \varepsilon^{(i-1)} \end{bmatrix} \right]$$

choose appropriate $r(i-1)$ to make the RHS ~~be~~ have only one nonzero number.

$$\therefore \quad k_i = \frac{\gamma^{(i-1)}}{\varepsilon^{(i-1)}}$$

$$\varepsilon^{(i)} = \varepsilon^{(i-1)} - k_i \gamma^{(i-1)} = \varepsilon^{(i-1)}(1-k_i^2)$$

$$\therefore \quad \begin{bmatrix} 1 \\ -\alpha_1^{(i)} \\ -\alpha_2^{(i)} \\ \vdots \\ -\alpha_i^{(i)} \end{bmatrix} = \begin{bmatrix} 1 \\ -\alpha_1^{(i-1)} \\ -\alpha_2^{(i-1)} \\ \vdots \\ 0 \end{bmatrix} - k_i \begin{bmatrix} 0 \\ -\alpha_{i-1}^{(i-1)} \\ -\alpha_{i-2}^{(i-1)} \\ \vdots \\ 1 \end{bmatrix}$$

$$\alpha_j^{(i)} = \alpha_j^{(i-1)} - k_i \alpha_{i-j}^{(i-1)}, \quad j = 1, 2, \cdots, i-1$$

$$\alpha_i^{(i)} = k_i$$

29th October

- pseudo-program

$$\varepsilon^{(0)} = R(0)$$

for $i = 1, 2, \dots p$

$$k_i = \left(R[i] - \sum_{j=1}^{i-1} \alpha_j^{(i-1)} R[i-j]\right) / \varepsilon^{(i-1)}$$

$$\alpha_i^{(i)} = k_i$$

if $i > 1$, then for $j = 1, 2, \dots, i-1$

$$\alpha_j^{(i)} = \alpha_j^{(i-1)} - k_i \alpha_{i-j}^{(i-1)}$$

end

$$\varepsilon^{(i)} = (1 - k_i^2)\, \varepsilon^{(i-1)}$$

end

$$\alpha_j = \alpha_j^{(p)} \quad j = 1, 2, \dots, p$$

- G

$$G u(n) = s(n) - \sum_{i=1}^{p} a_i s(n-i)$$

$$\Downarrow \times s(n)$$

$$E\left[\left(s(n) - \sum_{i=1}^{p} a_i s(n-i)\right) s(n)\right] = E[s^2(n)] - \sum_{i=1}^{p} a_i E[s(n-i) s(n)]$$

$$= R(0) - \sum_{i=1}^{p} a_i R(i)$$

$$G\, E[u(n) s(n)] = E\left[G u(n)\left(G u(n) + \sum_{i=1}^{p} a_i s(n-i)\right)\right]$$

$$= G^2 E[u^2(n)] + G \sum_{i=1}^{p} a_i E[u(n) s(n-i)]$$

if $u[n] \sim N(0,1)$

$$G^2 = R(0) - \sum_{i=1}^{p} \alpha_i R(i)$$

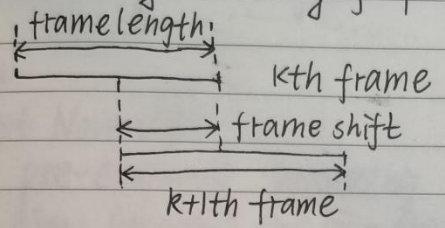
$$E_p = E[e(n) s(n)] = R(0) - \sum_{i=1}^{p} a_i R(i)$$

$$\Rightarrow G^2 = E_p$$

$$G = \sqrt{E_p}$$

30th October

- Windowing Processing of speech signal.



- Rectangular Window

$$w[n] = \begin{cases} 1, & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases}$$

- Hamming Window

$$w[n] = \begin{cases} 0.54 - 0.46\cos[2\pi n/(N-1)], & 0 \le n \le N \\ 0, & \text{otherwise} . \end{cases}$$

31th October

- Repeat Experiment

SH	~~consonant~~		
AA	vowel		centre
AO ~~ow~~	vonsonant	/ɔ/	centre
IY	consonant	/i/	front

$\begin{cases} \text{SH} & \text{vioceless Fricative-consonant} \\ \text{AA} & \text{vowel center} \end{cases}$

$\begin{cases} \text{AO} & \text{vowel center} \\ \text{IY} & \text{vowel front} \end{cases}$

vowel: last a long time

The vibration of the vocal cords produces a quasi-periodic pulse of air, the impulse of the empty air excites vocal tract and then causes vowe

- Vroiceless Fricative:
  steady stream of air that excites vocal tract.

77

1st November

- pre-emphasis processing

  Aim: Emphasis the high frequency part of the speech, remove the effects of lip radiation, increase the high frequency resolution of speech

$$H(z)=1-\alpha z^{-1},\ 0.9<\alpha<1.0$$

Experiment Result and Analysis

Exercise1

```
function [A,G,r,a]=autolpc(x,p)
    % Usage: [A, G, r, a] = autolpc(x, p)
    % x : input samples
    % p : order of LPC
    % A : prediction error filter, (A = [1; -a])
    % G : rms prediction error
    % r : autocorrelation coefficients
    % a : predictor coefficients
    n=length(x);
    s=x; %convert input row vector to column vector

    Rp=zeros(p,1);
    %autocorrelation function
    for i=1:p
        Rp(i,1)=sum(s(i+1:n).*s(1:n-i));
    end

    Rp=Rp(:);
    Rp_0=s'*s; %Rn(0)
    G=zeros(p,1);
    r=zeros(p,1);
    a1=zeros(p,p);

    %p=1
    G_0=Rp_0;
    r(1,1)=Rp(1,1)/Rp_0;
    a1(1,1)=r(1,1);
    G(1,1)=(1-r(1,1)^2)*G_0;

    %i>=2
    if p>1
        for i=2:p
            r(i,1)=(Rp(i,1)-sum(a1(1:i-1,i-1).*Rp(i-1:-1:1)))/G(i-1,1);
            a1(i,i)=r(i,1);
            G(i,1)=(1-r(i,1)^2)*G(i-1,1);
            for j=1:i-1
                a1(j,i)=a1(j,i-1)-r(i,1)*a1(i-j,i-1);
            end
        end
    end
    a=a1(:,p);
    A=zeros(1,p+1);
    A(1)=1.0;
    for i=2:p+1
```

```
        A(1,i)=-a(i-1,1);
    end

end
```

Test:

```
clear all;
w=randn(10000,1);
A=[1;-2.213;2.940;-2.170;0.961];
B=1;
x=filter(B,A,w);
[A1,G,r,a]=autolpc(x,4);
```

Result:

A1

1x5 double

	1	2	3	4	5
1	1	-2.215971054972604	2.940011464268123	-2.166616019857339	0.958898864227972

Analysis:

The prediction filter data A1 is approximately equal to the input A, which implies the lpc function perform correctly.

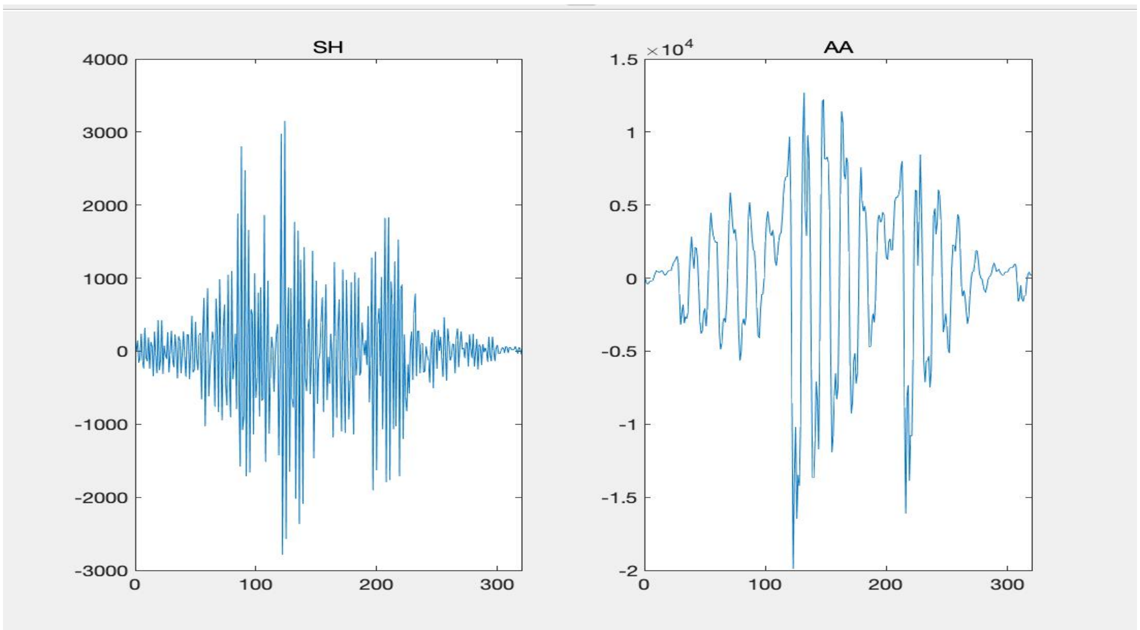
Exercise2:

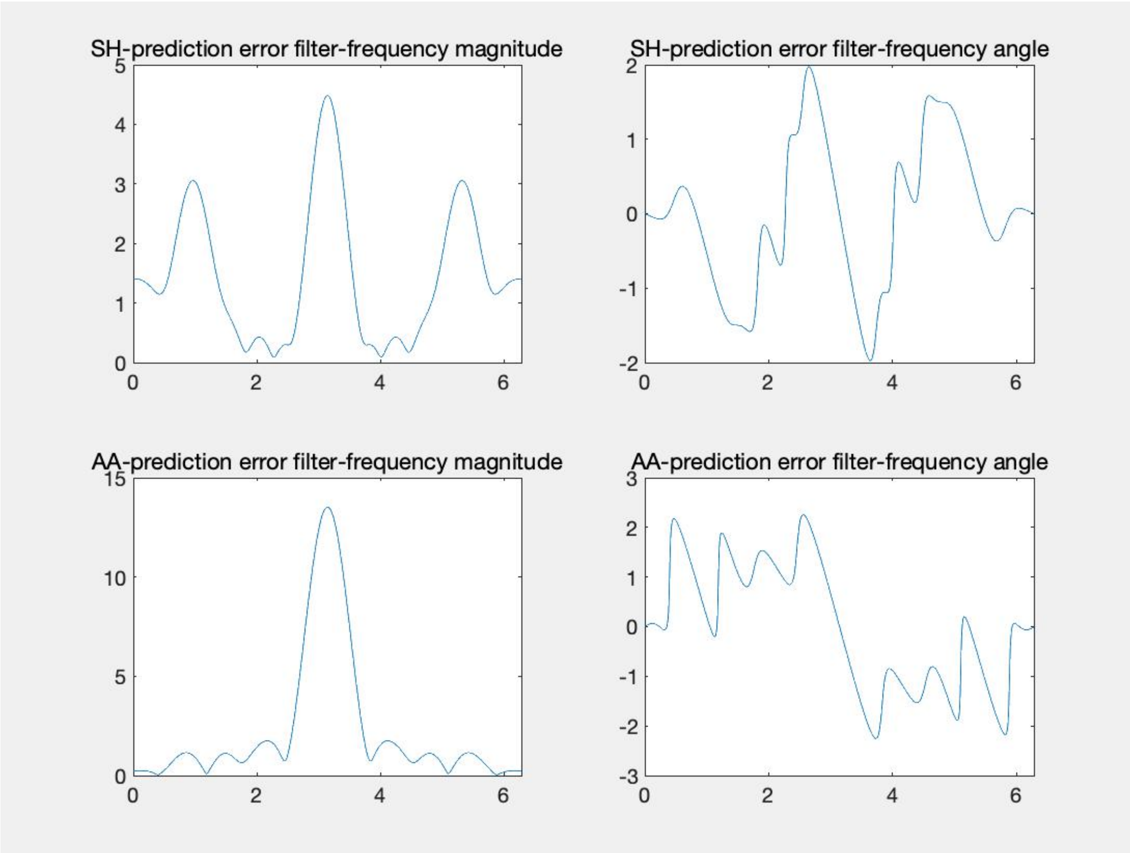
```
clear all;
load('s5.mat');
%soundsc(s5);

SH=s5(16100:17100);
AA=s5(17000:18000);
%soundsc(SH);

%hamming window
N=320;
x=linspace(1,320,N);
h=hamming(N);

 %enframe and windowed
x1=enframe(SH,hamming(N),160);
figure(1);
subplot(1,2,1);
plot(x1(3,:));
title('SH');

x2=enframe(AA,hamming(N),160);
subplot(1,2,2);
```

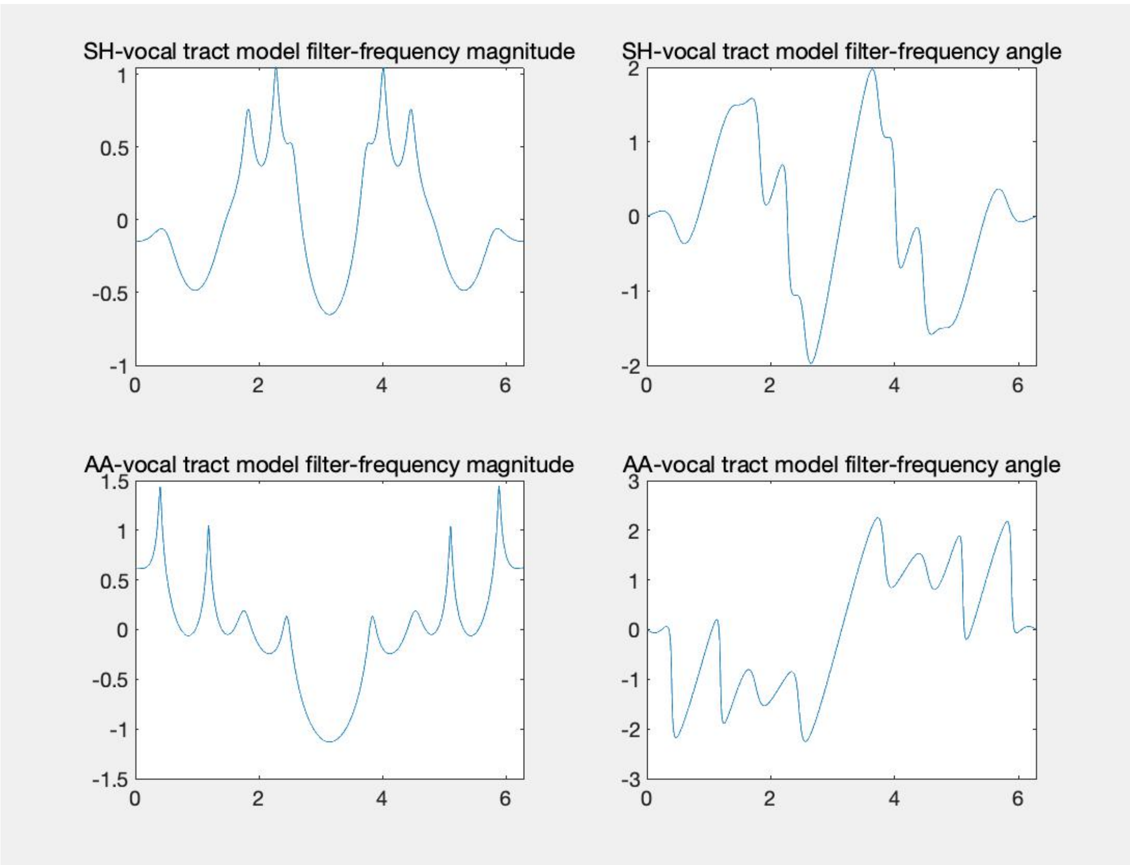
```
plot(x2(2,:));
title('AA');
%lpc
p=12;
[A1,G1,r1,a1]=autolpc(x1(3,:)',p);
[A2,G2,r2,a2]=autolpc(x2(2,:)',p);
%frequency response of prediction error filter
B=1;
[H1,w1]=freqz(A1,B,1024,'whole');
[H2,w2]=freqz(A2,B,1024,'whole');
Hf1=abs(H1);
Hx1=angle(H1);
Hf2=abs(H2);
Hx2=angle(H2);
%plot for prediction error filter
figure(2);
subplot(2,2,1);
plot(w1,Hf1);
title('SH-prediction error filter-frequency magnitude');
subplot(2,2,2);
plot(w1,Hx1);
title('SH-prediction error filter-frequency angle');
subplot(2,2,3);
plot(w2,Hf2);
title('AA-prediction error filter-frequency magnitude');
subplot(2,2,4);
plot(w2,Hx2);
title('AA-prediction error filter-frequency angle');
 %frequency response of vocal tract filter
[H3,w3]=freqz(B,A1,1024,'whole');
[H4,w4]=freqz(B,A2,1024,'whole');
Hf3=log10(abs(H3));
Hx3=angle(H3);
Hf4=log10(abs(H4));
Hx4=angle(H4);
figure(3);
subplot(2,2,1);
plot(w3,Hf3);
title('SH-vocal tract model filter-frequency magnitude');
subplot(2,2,2);
plot(w3,Hx3);
title('SH-vocal tract model filter-frequency angle');
subplot(2,2,3);
plot(w4,Hf4);
title('AA-vocal tract model filter-frequency magnitude');
subplot(2,2,4);
plot(w4,Hx4);
```

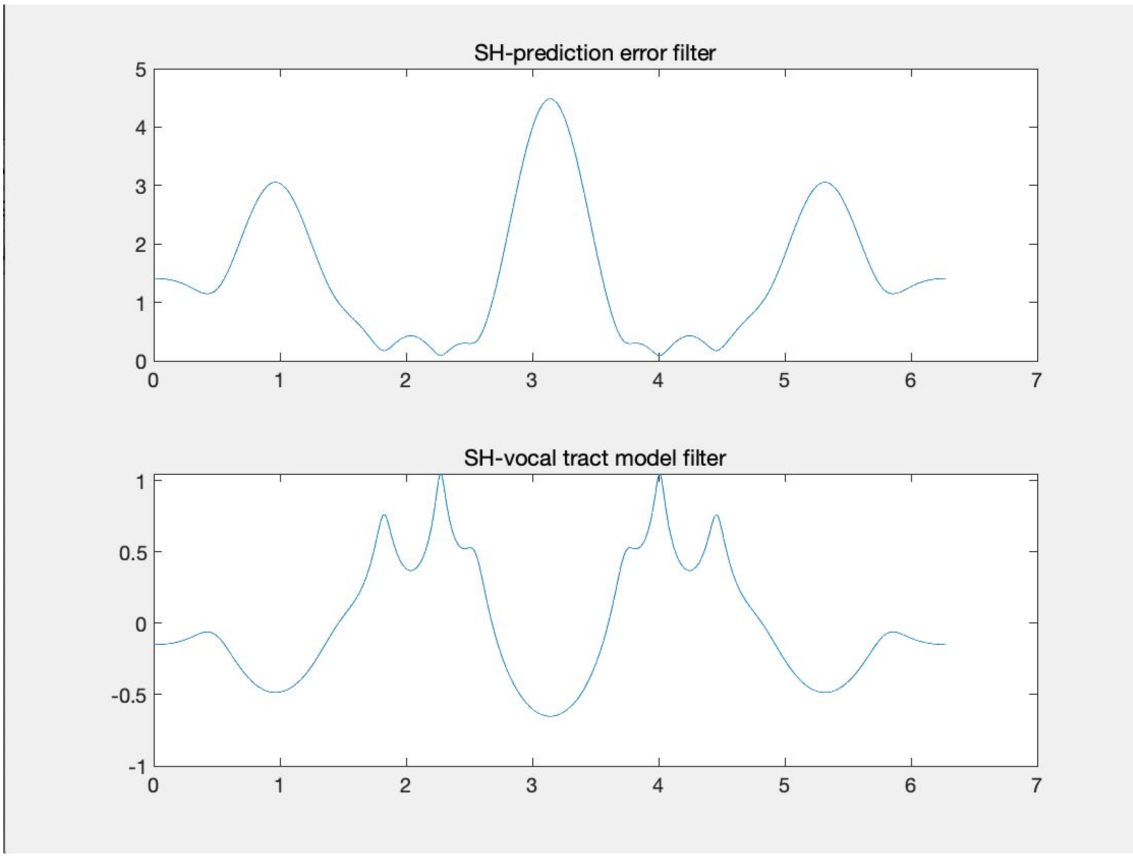
```
title('AA-vocal tract model filter-frequency angle');
%plot for vocal tract filter
figure(4);
subplot(2,1,1);
plot(w1,Hf1);
title('SH-prediction error filter');
subplot(2,1,2);
plot(w3,Hf3);
title('SH-vocal tract model filter');
figure(5);
subplot(2,1,1);
plot(w2,Hf2);
title('AA-prediction error filter');
subplot(2,1,2);
plot(w4,Hf4);
title('AA-vocal tract model filter');
%zplane for SH and AA
figure(6);
subplot(1,2,1);
zplane(A1);
title('SH');
subplot(1,2,2);
zplane(A2);
title('AA');
```

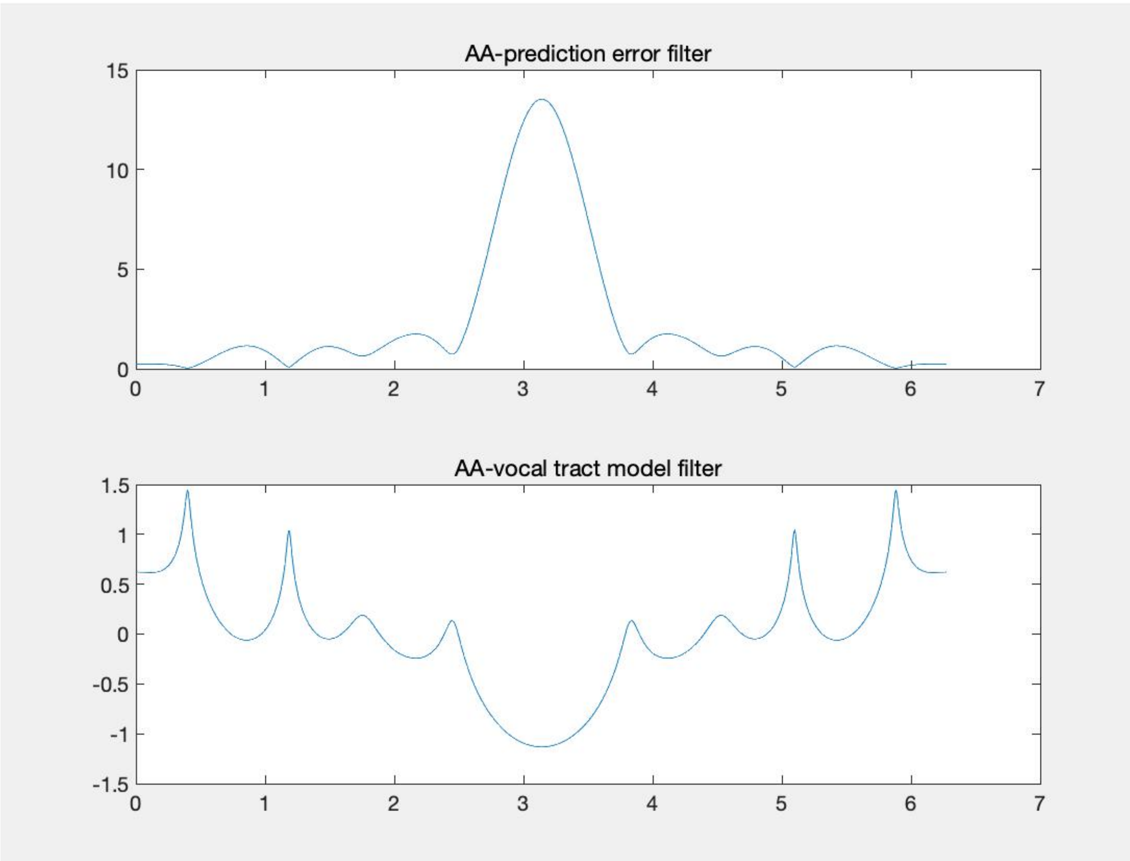
Result:

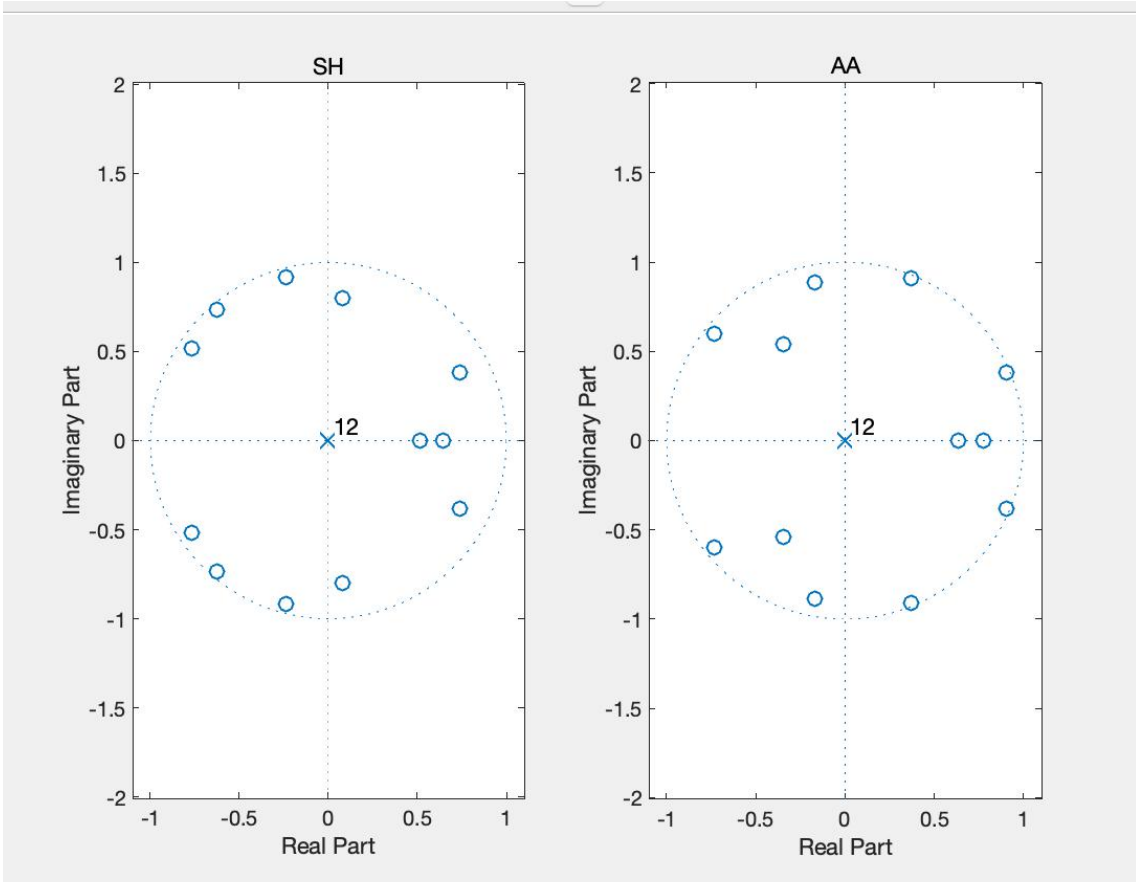



SH-prediction error filter-frequency magnitude
SH-prediction error filter-frequency angle
AA-prediction error filter-frequency magnitude
AA-prediction error filter-frequency angle


SH-vocal tract model filter-frequency magnitude
SH-vocal tract model filter-frequency angle
AA-vocal tract model filter-frequency magnitude
AA-vocal tract model filter-frequency angle


SH-prediction error filter
SH-vocal tract model filter


AA-prediction error filter
AA-vocal tract model filter



Analysis:

a. The zeros of prediction error filter are just corresponding to the poles of vocal tract filter and the number are equal to the filter order. The system functions of these two filters are derivatives of each other.

b. The number of peaks in the frequency response of the vocal tract model is equal to the number of zeros of the prediction error filter which close to the unit circle.

c. The number of dips in the frequency response of the prediction error model is equal to the number of zeros of the prediction error filter which close to the unit circle.

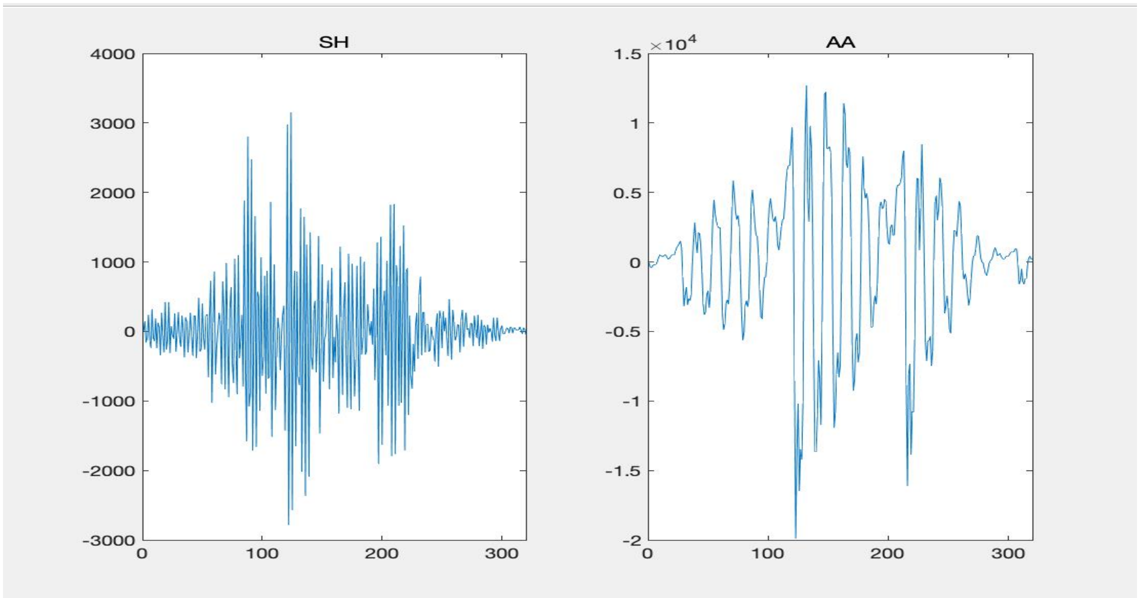
Exercise3:

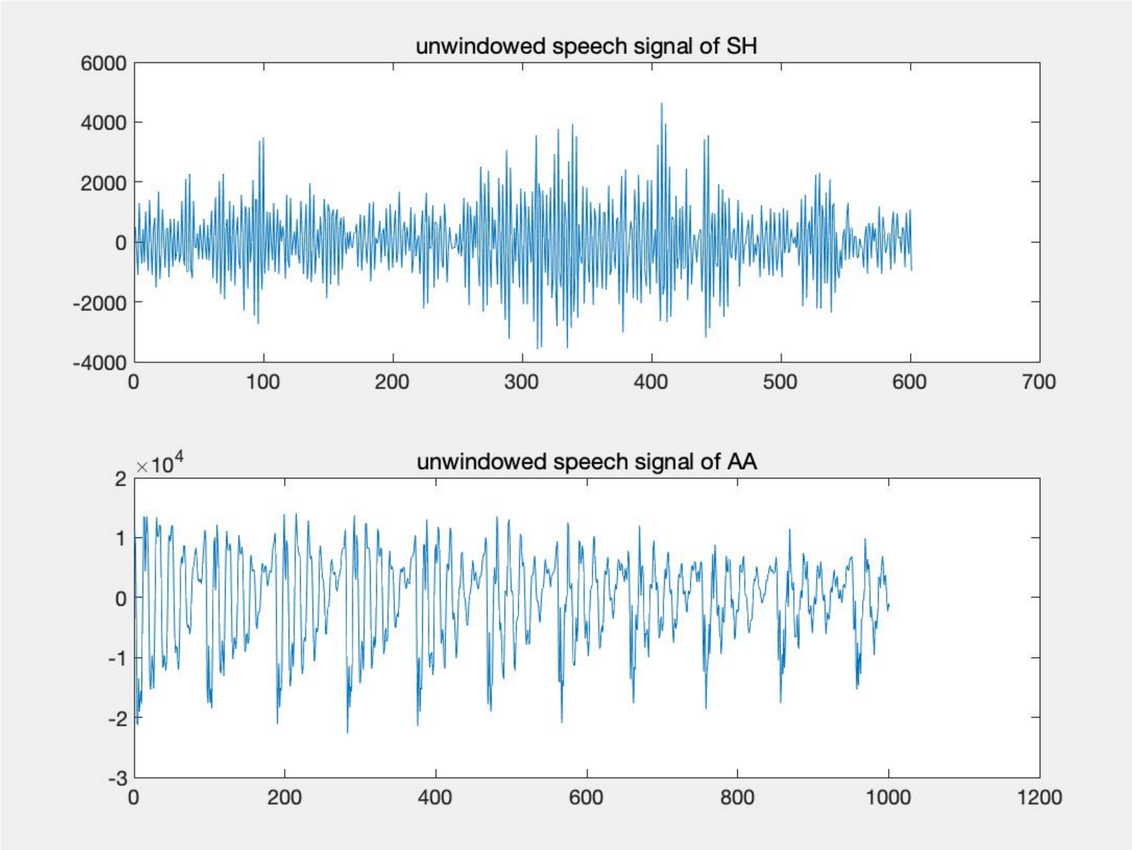
```
clear all;
load('s5.mat');
%soundsc(s5);

SH=s5(16100:17100);
AA=s5(17000:18000);
%soundsc(SH);

N=320;
x=linspace(1,320,N);
h=hamming(N);
```

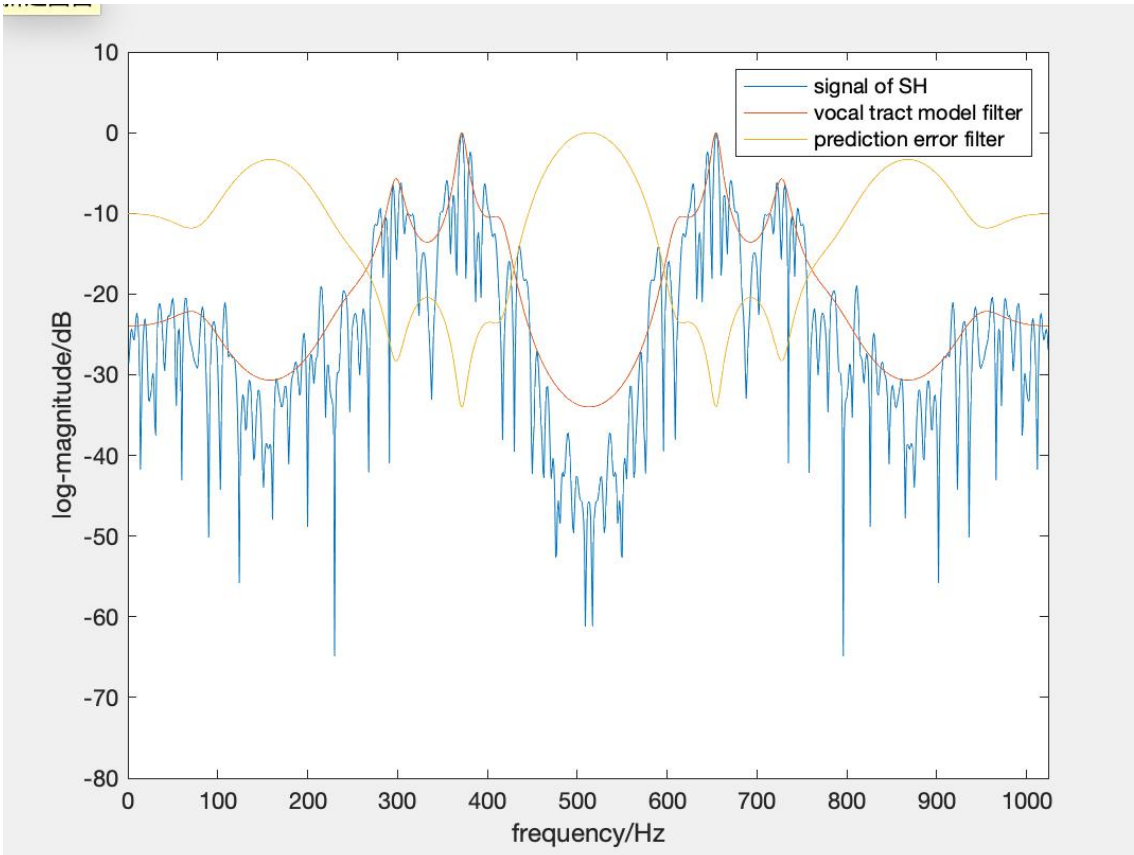
```
x1=enframe(SH,hamming(N),160);
figure(1);
subplot(1,2,1);
s1=x1(3,:);
plot(s1);
title('SH');

x2=enframe(AA,hamming(N),160);
subplot(1,2,2);
s2=x2(2,:);
plot(s2);
title('AA');

s1_fft=fft(s1,1024);
s1_fft1=abs(s1_fft);
s1_fft1=s1_fft1/max(s1_fft1);
s1_mag=20*log10(s1_fft1);

s2_fft=fft(s2,1024);
s2_fft1=abs(s2_fft);
s2_fft1=s2_fft1/max(s2_fft1);
s2_mag=20*log10(s2_fft1);
%pinlv=(0:1:255)*8000/512;

%LPC
p=12;
[A1,G1,r1,a1]=autolpc(x1(3,:)',p);
[A2,G2,r2,a2]=autolpc(x2(2,:)',p);
B=1;

[H1,w1]=freqz(A1,B,1024,'whole');
H1=abs(H1);
H1=H1/max(H1);
H1_dB=20*log10(abs(H1));

%vocal tract filter
[H3,w3]=freqz(B,A1,1024,'whole');
H3=abs(H3);
H3=H3/max(H3);
H3_dB=20*log10(abs(H3));
H3sig(1:256)=s1_mag(1:256);

figure(2);
plot(s1_mag);
xlabel('frequency/Hz');
ylabel('log-magnitude/dB');
axis([0,1024,-80,15]);
```

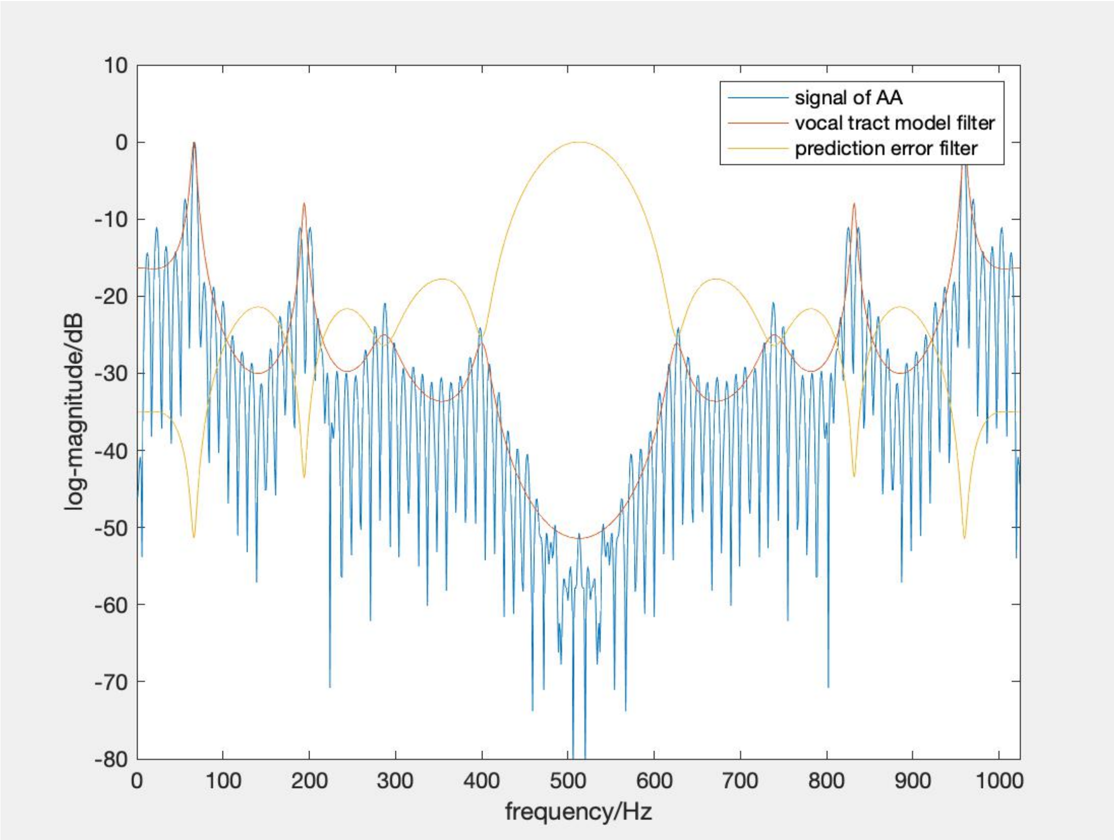
```
hold on;
plot(H3_dB);
axis([0,1024,-80,10]);
hold on;
plot(H1_dB);
axis([0,1024,-80,10]);
legend('signal','vocal tract model filter','prediction error filter');

[H2,w2]=freqz(A2,B,1024,'whole');
H2=abs(H2);
H2=H2/max(H2);
H2_dB=20*log10(abs(H2));
[H4,w4]=freqz(B,A2,1024,'whole');
H4=abs(H4);
H4=H4/max(H4);
H4_dB=20*log10(abs(H4));
figure(3);
plot(s2_mag);
xlabel('frequency/Hz');
ylabel('log-magnitude/dB');
axis([0,1024,-80,15]);
hold on;
plot(H4_dB);
axis([0,1024,-80,10]);
hold on;
plot(H2_dB);
axis([0,1024,-80,10]);
legend('signal','vocal tract model filter','prediction error filter');
```

Result:




unwindowed speech signal of SH
unwindowed speech signal of AA
$\times 10^4$


signal of SH
vocal tract model filter
prediction error filter
log-magnitude/dB
frequency/Hz



Analysis:

a. The waveform of AA has the property of quasi-period and can be found easily, while the waveform of SH is live noise and hard to be recognized. The energy of AA is larger than SH. So the phoneme AA is voiced and SH is unvoiced.

b. Voiced sound is produced by forcing airflow through the glottis and adjusting the vocal cords to a proper tension. At this time, the vocal cords are relaxed and tense alternately, producing quasi-periodic air pulses to stimulate the vocal tract, and finally resulting in a quasi-periodic waveform.
Unvoiced sound is produced by contraction in certain positions of the vocal tract, forcing air to be disturbed by the contraction point at a sufficiently high speed. This creates a wideband noise source to excite the channel.

Exercise 4:

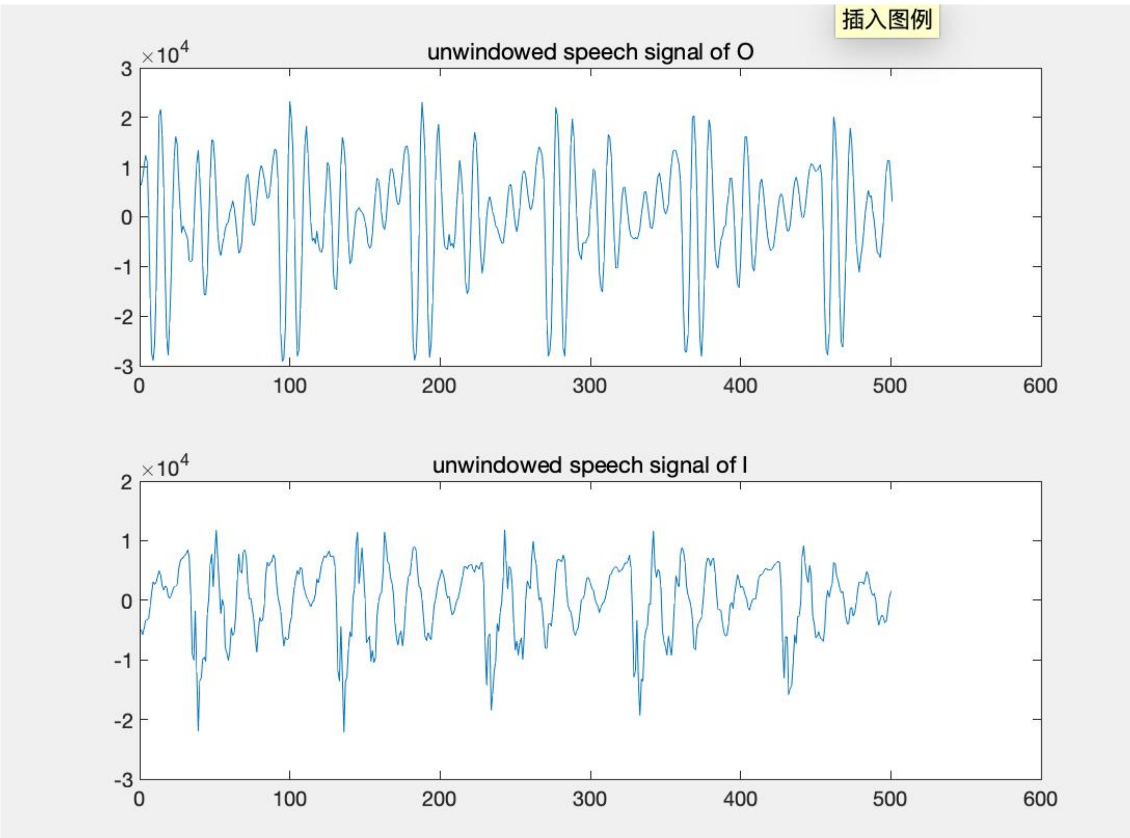
```
clear all;
load('s5.mat');
O=s5(7000:7500);
I=s5(14500:15000);
%soundsc(I);

N=320;
x=linspace(1,320,N);
h=hamming(N);
figure(1);
```

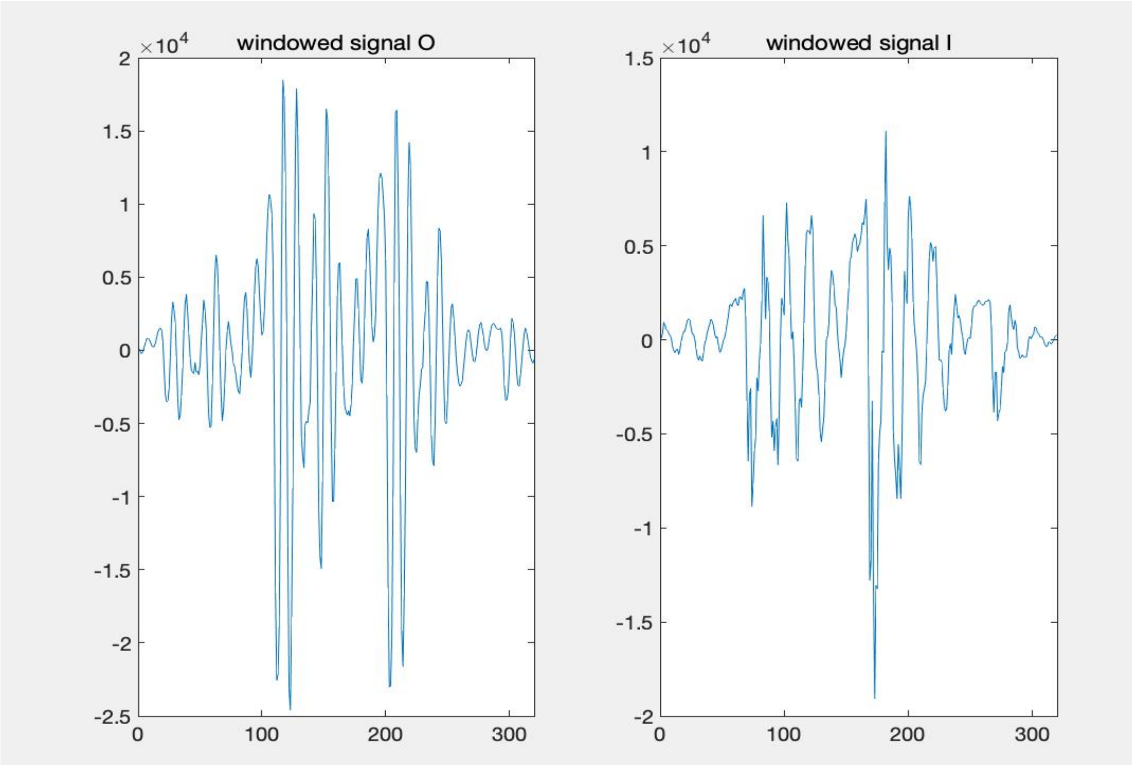
```
plot(O);
hold on;
plot(I);
legend('O','I');

x1=enframe(O,hamming(N),160);
figure(2);
subplot(1,2,1);
plot(x1(2,:));
title('O');

x2=enframe(I,hamming(N),160);
subplot(1,2,2);
plot(x2(2,:));
title('I');

%lpc
p=12;
[A1,G1,r1,a1]=autolpc(x1(2,:)',p);
[A2,G2,r2,a2]=autolpc(x2(2,:)',p);

%frequency response of prediction error filter
B=1;
[H1,w1]=freqz(A1,B,1024,'whole');
[H2,w2]=freqz(A2,B,1024,'whole');
%magnitude & angle
Hf1=abs(H1);
Hf1=Hf1/max(Hf1);
Hf1=20*log10(Hf1);
Hx1=angle(H1);
Hf2=abs(H2);
Hf2=Hf2/max(Hf2);
Hf2=20*log10(Hf2);
Hx2=angle(H2);

%frequency response of vocal tract model filter
[H3,w3]=freqz(B,A1,1024,'whole');
[H4,w4]=freqz(B,A2,1024,'whole');
%log magnitude and angle
Hf3=20*log10(abs(H3)/max(abs(H3)));
Hx3=angle(H3);
Hf4=20*log10(abs(H4)/max(abs(H4)));
Hx4=angle(H4);

%plot for O
figure(3);
plot(Hf1);
```

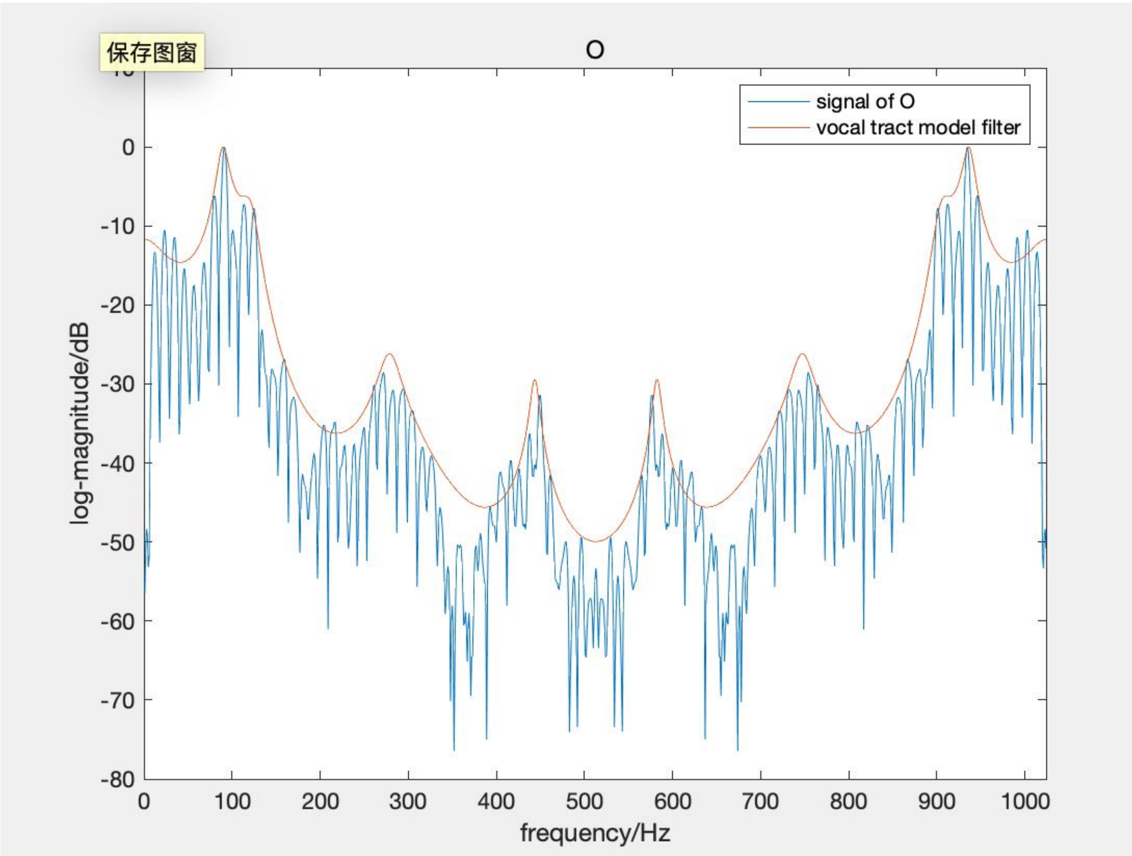
```
hold on;
title('filter magnitude frequency response of O');
xlabel('frequency/Hz');
ylabel('magnitude/dB');
axis([0,1024,-80,15]);
plot(Hf3);
axis([0,1024,-80,15]);
legend('prediction error filter','vocal tract model filter');

%plot for I
figure(4);
plot(Hf2);
hold on;
title('filter magnitude frequency response of I');
xlabel('frequency/Hz');
ylabel('magnitude/dB');
axis([0,1024,-50,15]);
plot(Hf4);
axis([0,1024,-50,15]);
legend('prediction error filter','vocal tract model filter');

%plot for zeros
figure(5);
subplot(1,2,1);
zplane(a1);
title('the zeros of the prediction error filter of O');
subplot(1,2,2);
zplane(a2);
title('the zeros of the prediction error filter of I');

%DFT of windowed segment of speech
s1=x1(2,:);
s1_fft=fft(s1,1024);
s1_fft1=abs(s1_fft);
s1_fft1=s1_fft1/max(s1_fft1);
s1_mag=20*log10(s1_fft1);

s2=x2(2,:);
s2_fft=fft(s2,1024);
s2_fft1=abs(s2_fft);
s2_fft1=s2_fft1/max(s2_fft1);
s2_mag=20*log10(s2_fft1);

%plot for O
figure(6);
plot(s1_mag);
xlabel('frequency/Hz');
```

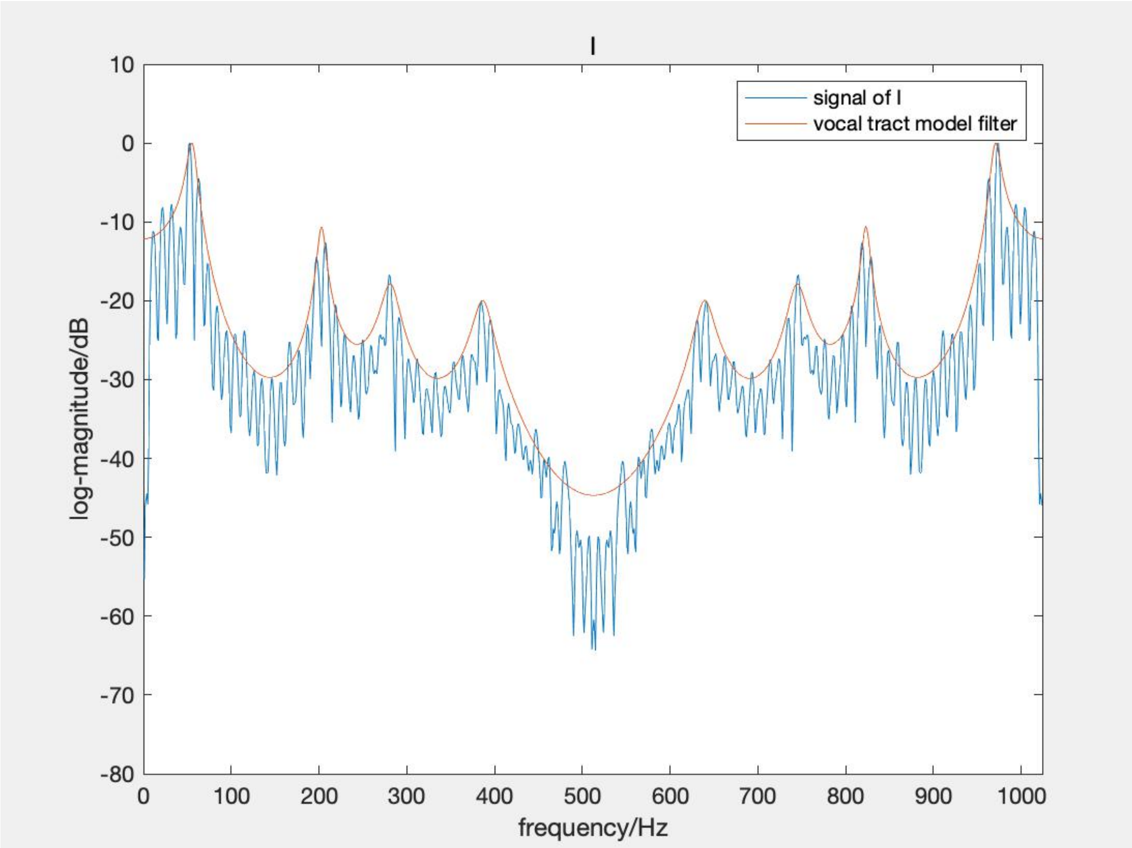
```
ylabel('log-magnitude/dB');
axis([0,1024,-80,15]);
title('O');
hold on;
plot(Hf3);
axis([0,1024,-80,10]);
hold on;
legend('signal','vocal tract model filter');

%plot for I
figure(7);
plot(s2_mag);
xlabel('frequency/Hz');
ylabel('log-magnitude/dB');
axis([0,1024,-80,15]);
hold on;
plot(Hf4);
axis([0,1024,-80,10]);
hold on;
legend('signal','vocal tract model filter');
title('I');
```

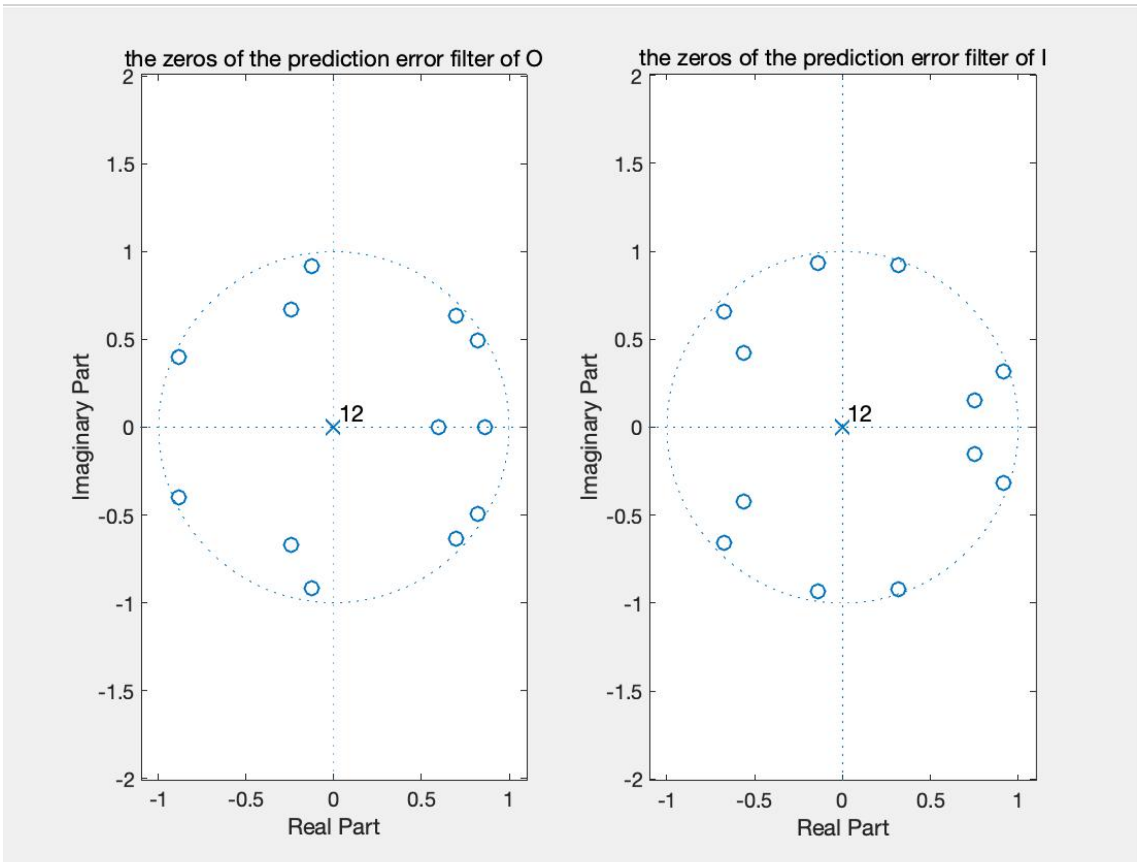
Result:

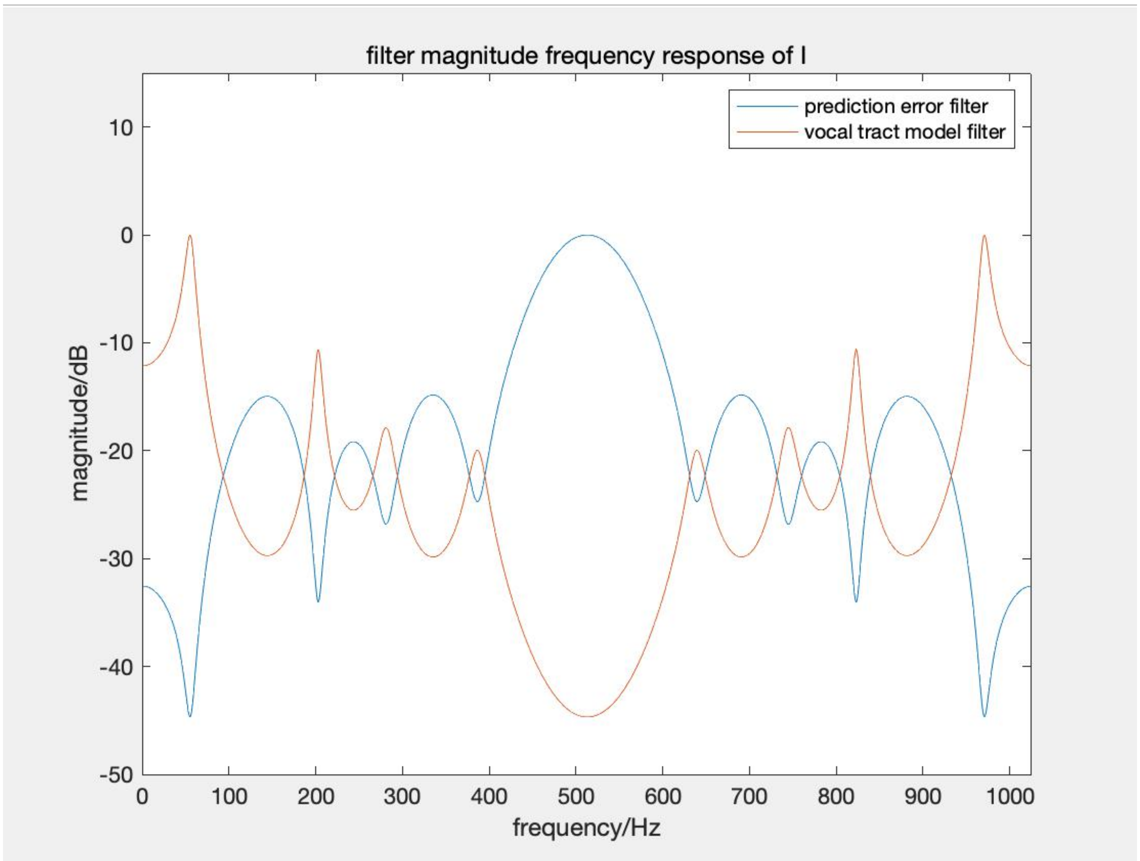


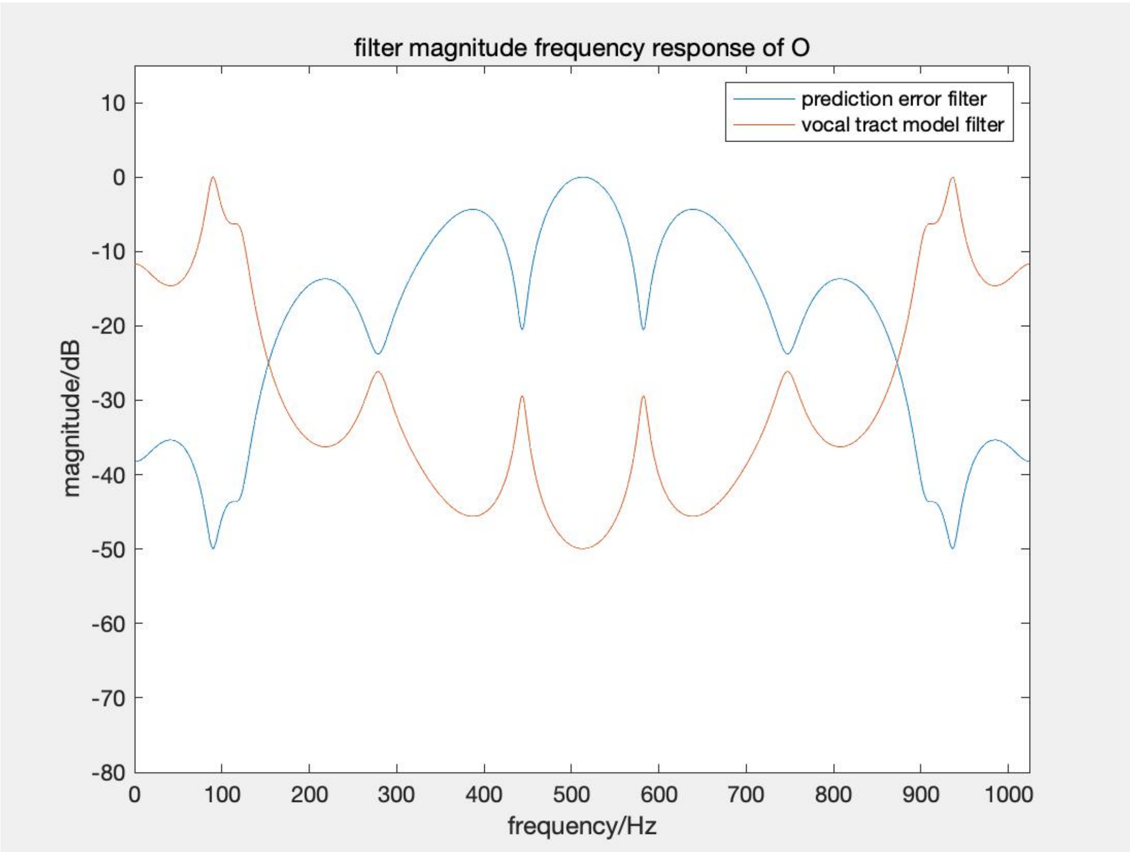

windowed signal O
windowed signal I
×10^4


保存图窗
O
signal of O
vocal tract model filter
log-magnitude/dB
frequency/Hz


I
signal of I
vocal tract model filter
log-magnitude/dB
frequency/Hz


the zeros of the prediction error filter of O
the zeros of the prediction error filter of I
Imaginary Part
Real Part
12





Analysis:

a. The 'o' in 'strong' corresponds to the phoneme 'AO' and the 'i' in 'gives' corresponds to 'IY'.

They both belong to voiced phonemes and have quasi-period. However, there are still some differences between them. The magnitude of the AO is bigger than the IY and the peaks of DFT of the two signals are different.

b. AO is so-called middle vowel, while IY is front vowel. When the front vowel is uttered, the vocal tract is contracted by the tongue bulging forward, with a lower first resonance frequency and a higher second resonance frequency. The middle vowel has a higher first resonance frequency and a lower second resonance frequency.

Exercise 5:

```
clear all;
load('s5.mat');
O=s5(7000:7500);
I=s5(14500:15000);

%generate hamming window
N=320;
x=linspace(1,320,N);
h=hamming(N);

%enframe O and I
x1=enframe(O,hamming(N),160);
x2=enframe(I,hamming(N),160);

%lpc for O, p=8,10,12,20
so=x1(2,:)';
p1=8;
p2=10;
p3=12;
p4=20;
[AO1,GO1,rO1,aO1]=autolpc(so,p1);
[AO2,GO2,rO2,aO2]=autolpc(so,p2);
[AO3,GO3,rO3,aO3]=autolpc(so,p3);
[AO4,GO4,rO4,aO4]=autolpc(so,p4);

%DFT of windowed segment of speech O
so_fft=fft(so,1024);
so_fft1=abs(so_fft);
so_fft1=so_fft1/max(so_fft1);
so_mag=20*log10(so_fft1);

%frequency response of prediction error filter of O
B=1;
%p=8
[HO1,wO1]=freqz(AO1,B,1024,'whole');
```

```
%magnitude
HfO1=abs(HO1);
HfO1=HfO1/max(HfO1);
HfO1=20*log10(HfO1);
%p=10
[HO2,wO2]=freqz(AO2,B,1024,'whole');
%magnitude
HfO2=abs(HO2);
HfO2=HfO2/max(HfO2);
HfO2=20*log10(HfO2);
%p=12
[HO3,wO3]=freqz(AO3,B,1024,'whole');
%magnitude
HfO3=abs(HO3);
HfO3=HfO3/max(HfO3);
HfO3=20*log10(HfO3);
%p=20
[HO4,wO4]=freqz(AO4,B,1024,'whole');
%magnitude
HfO4=abs(HO4);
HfO4=HfO4/max(HfO4);
HfO4=20*log10(HfO4);

%plot for O
figure(1);
title('O');
plot(so_mag);
xlabel('frequency/Hz');
ylabel('log-magnitude/dB');
axis([0,1024,-80,15]);
hold on;
plot(HfO1);
hold on;
plot(HfO2);
hold on;
plot(HfO3);
hold on;
plot(HfO4);
hold on;
legend('DFT of windowed speech segment','prediction error filter p=8','prediction error filter p=10','prediction error filter p=12','prediction error filter p=20');

%frequency response of vocal tract filter of O
B=1;
%p=8
[HO11,wO11]=freqz(B,AO1,1024,'whole');
%magnitude
```

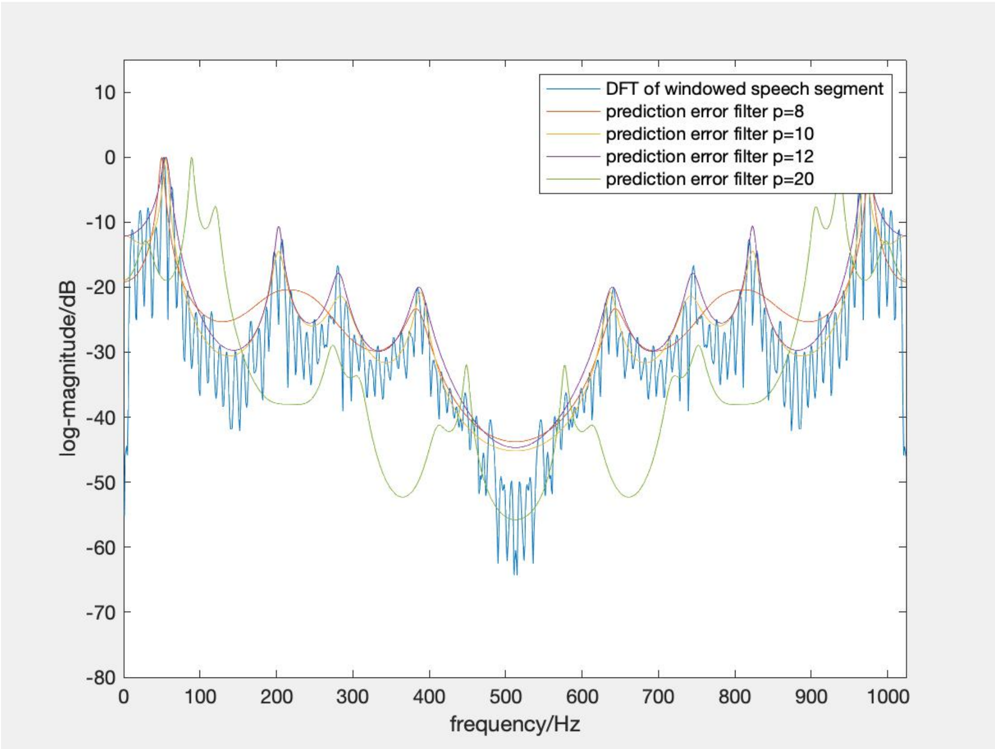
```
HfO11=abs(HO11);
HfO11=HfO11/max(HfO11);
HfO11=20*log10(HfO11);
%p=10
[HO21,wO21]=freqz(B,AO2,1024,'whole');
%magnitude
HfO21=abs(HO21);
HfO21=HfO21/max(HfO21);
HfO21=20*log10(HfO21);
%p=12
[HO31,wO31]=freqz(B,AO3,1024,'whole');
%magnitude
HfO31=abs(HO31);
HfO31=HfO31/max(HfO31);
HfO31=20*log10(HfO31);
%p=20
[HO41,wO41]=freqz(B,AO4,1024,'whole');
%magnitude
HfO41=abs(HO41);
HfO41=HfO41/max(HfO41);
HfO41=20*log10(HfO41);
%plot for O
figure(2);
title('O');
plot(so_mag);
xlabel('frequency/Hz');
ylabel('log-magnitude/dB');
axis([0,1024,-80,15]);
hold on;
plot(HfO11);
hold on;
plot(HfO21);
hold on;
plot(HfO31);
hold on;
plot(HfO41);
hold on;
legend('DFT of windowed speech segment','vocal tract filter p=8','vocal tract filter
p=10','vocal tract filter p=12','vocal tract filter p=20');

%lpc for I, p=8,10,12,20
si=x2(2,:)';
p1=8;
p2=10;
p3=12;
p4=20;
[AI1,GI1,rI1,aI1]=autolpc(si,p1);
```

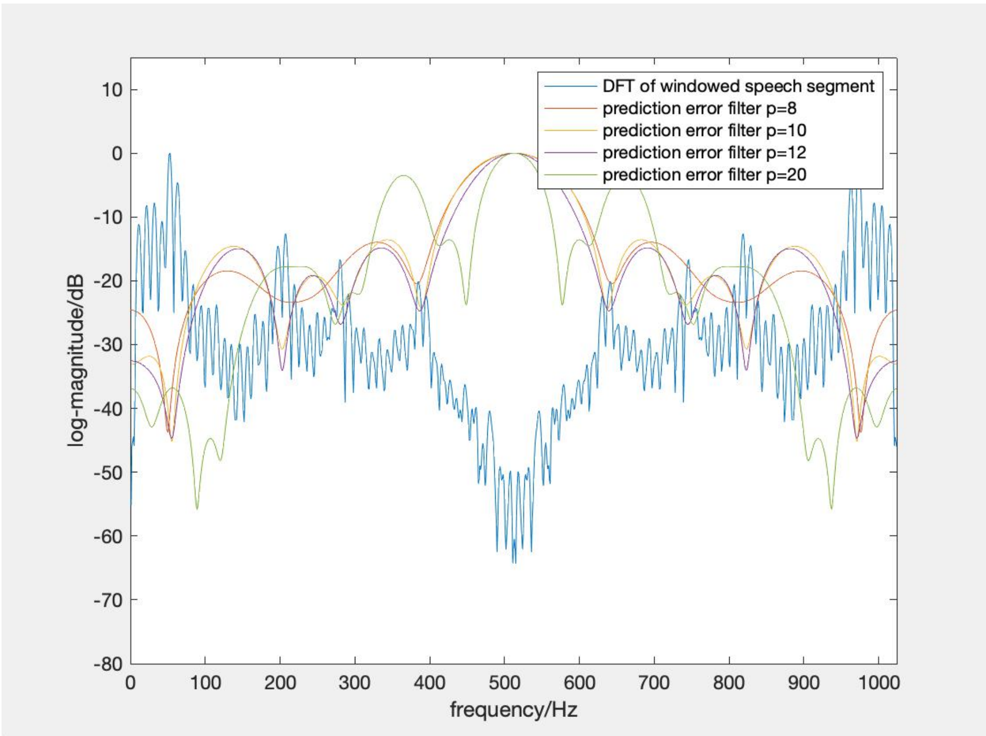
```
[Al2,Gl2,rl2,al2]=autolpc(si,p2);
[Al3,Gl3,rl3,al3]=autolpc(si,p3);
[Al4,Gl4,rl4,al4]=autolpc(si,p4);
%DFT of windowed segment of speech O
si_fft=fft(si,1024);
si_fft1=abs(si_fft);
si_fft1=si_fft1/max(si_fft1);
si_mag=20*log10(si_fft1);

%frequency response of prediction error filter of O
B=1;
%p=8
[Hl1,wl1]=freqz(Al1,B,1024,'whole');
%magnitude
Hfl1=abs(Hl1);
Hfl1=Hfl1/max(Hfl1);
Hfl1=20*log10(Hfl1);
%p=10
[Hl2,wl2]=freqz(Al2,B,1024,'whole');
%magnitude
Hfl2=abs(Hl2);
Hfl2=Hfl2/max(Hfl2);
Hfl2=20*log10(Hfl2);
%p=12
[Hl3,wl3]=freqz(Al3,B,1024,'whole');
%magnitude
Hfl3=abs(Hl3);
Hfl3=Hfl3/max(Hfl3);
Hfl3=20*log10(Hfl3);
%p=20
[HO4,wO4]=freqz(AO4,B,1024,'whole');
%magnitude
Hfl4=abs(HO4);
Hfl4=Hfl4/max(Hfl4);
Hfl4=20*log10(Hfl4);

%plot for O
figure(3);
title('I');
plot(si_mag);
xlabel('frequency/Hz');
ylabel('log-magnitude/dB');
axis([0,1024,-80,15]);
hold on;
plot(Hfl1);
hold on;
plot(Hfl2);
```

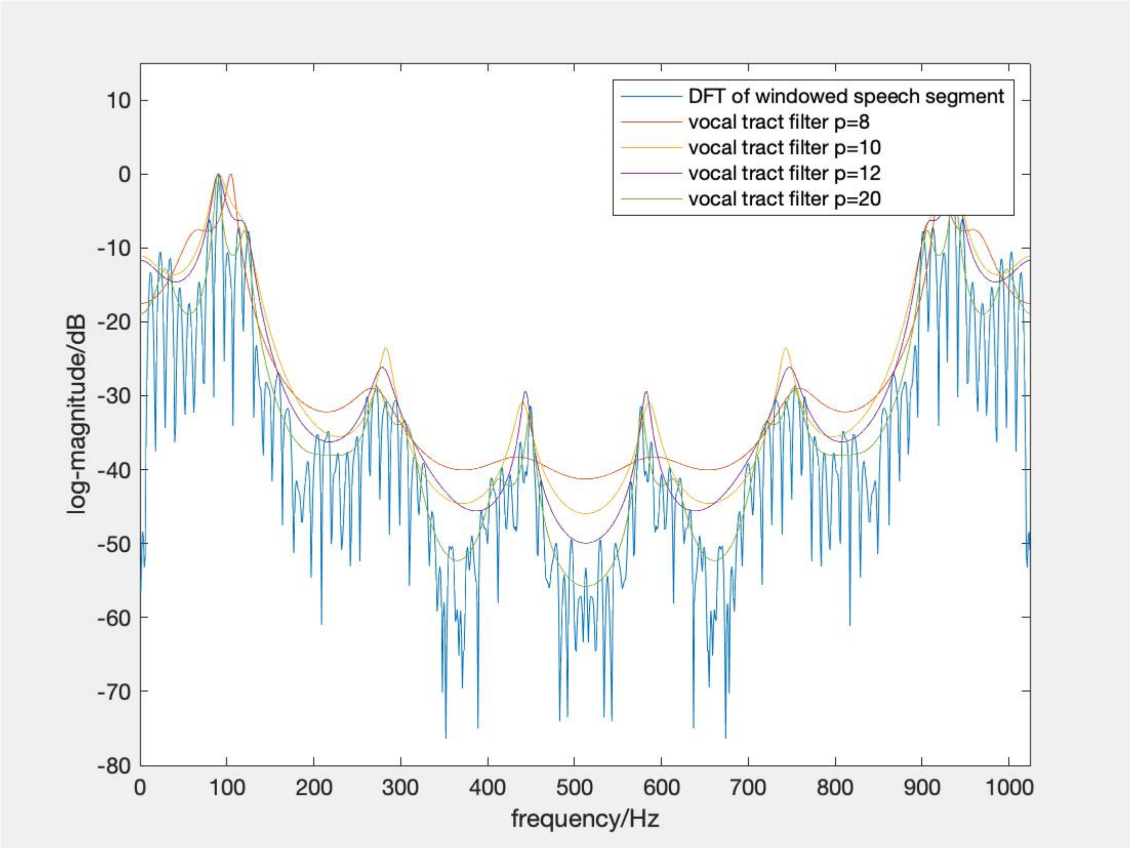
```
hold on;
plot(Hfl3);
hold on;
plot(Hfl4);
hold on;
legend('DFT of windowed speech segment','prediction error filter p=8','prediction error filter p=10','prediction error filter p=12','prediction error filter p=20');

%frequency response of vocal tract filter of O
B=1;
%p=8
[Hl11,wl11]=freqz(B,Al1,1024,'whole');
%magnitude
Hfl11=abs(Hl11);
Hfl11=Hfl11/max(Hfl11);
Hfl11=20*log10(Hfl11);
%p=10
[Hl21,wl21]=freqz(B,Al2,1024,'whole');
%magnitude
Hfl21=abs(Hl21);
Hfl21=Hfl21/max(Hfl21);
Hfl21=20*log10(Hfl21);
%p=12
[Hl31,wl31]=freqz(B,Al3,1024,'whole');
%magnitude
Hfl31=abs(Hl31);
Hfl31=Hfl31/max(Hfl31);
Hfl31=20*log10(Hfl31);
%p=20
[HO41,wO41]=freqz(B,AO4,1024,'whole');
%magnitude
Hfl41=abs(HO41);
Hfl41=Hfl41/max(Hfl41);
Hfl41=20*log10(Hfl41);

%plot for O
figure(4);
title('I');
plot(si_mag);
xlabel('frequency/Hz');
ylabel('log-magnitude/dB');
axis([0,1024,-80,15]);
hold on;
plot(Hfl11);
hold on;
plot(Hfl21);
hold on;
```

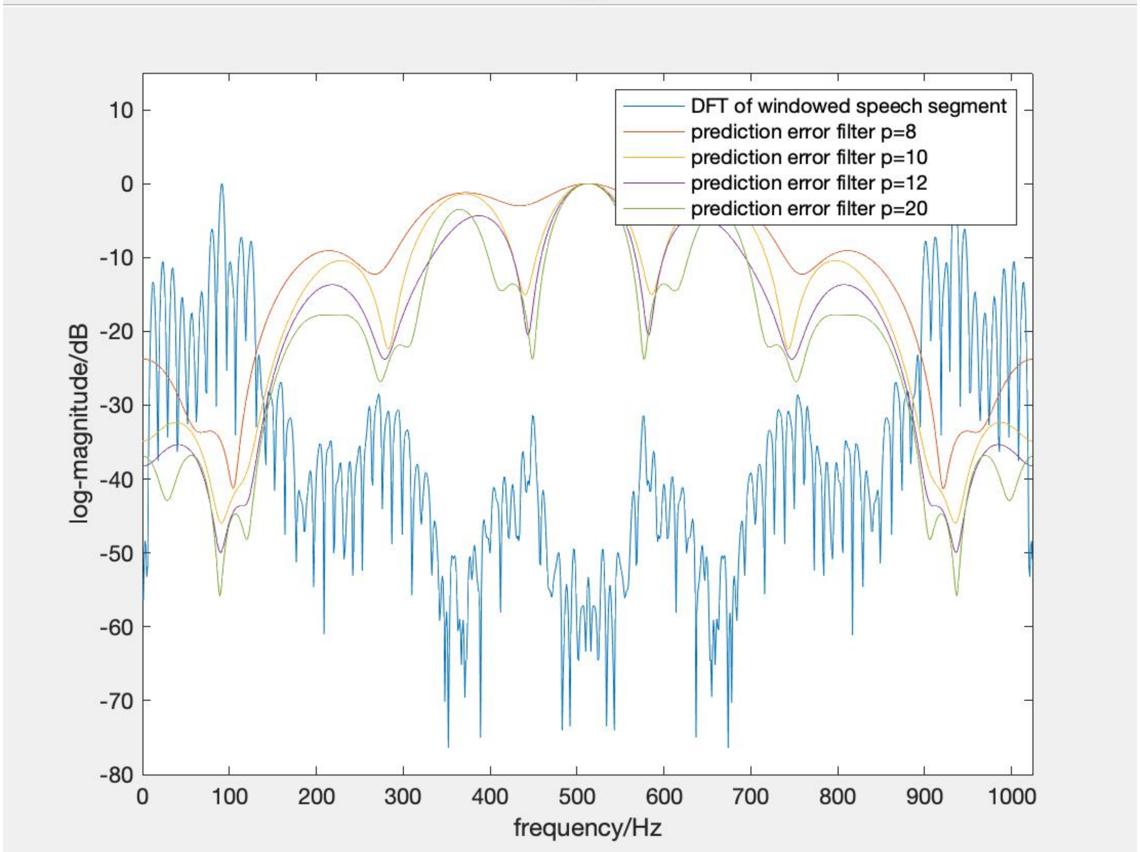
```
plot(Hfl31);
hold on;
plot(Hfl41);
hold on;
legend('DFT of windowed speech segment','prediction error filter p=8','prediction error filter p=10','prediction error filter p=12','prediction error filter p=20');
```

Result:






DFT of windowed speech segment
vocal tract filter p=8
vocal tract filter p=10
vocal tract filter p=12
vocal tract filter p=20
log-magnitude/dB
frequency/Hz
10
0
-10
-20
-30
-40
-50
-60
-70
-80
0
100
200
300
400
500
600
700
800
900
1000


DFT of windowed speech segment
prediction error filter p=8
prediction error filter p=10
prediction error filter p=12
prediction error filter p=20
log-magnitude/dB
frequency/Hz
10
0
-10
-20
-30
-40
-50
-60
-70
-80
0
100
200
300
400
500
600
700
800
900
1000

Analysis:

a. As the model order p increases, the frequency response of the vocal tract model is getting closer and closer to the short-time Fourier transform frequency response of the signal. This means that if p is large enough, the frequency response of the all-pole model can approximate the signal spectrum with arbitrarily small errors.

b. The order p of linear prediction can effectively control the smoothness of the generated spectrum. As p increases, more spectral details are retained.

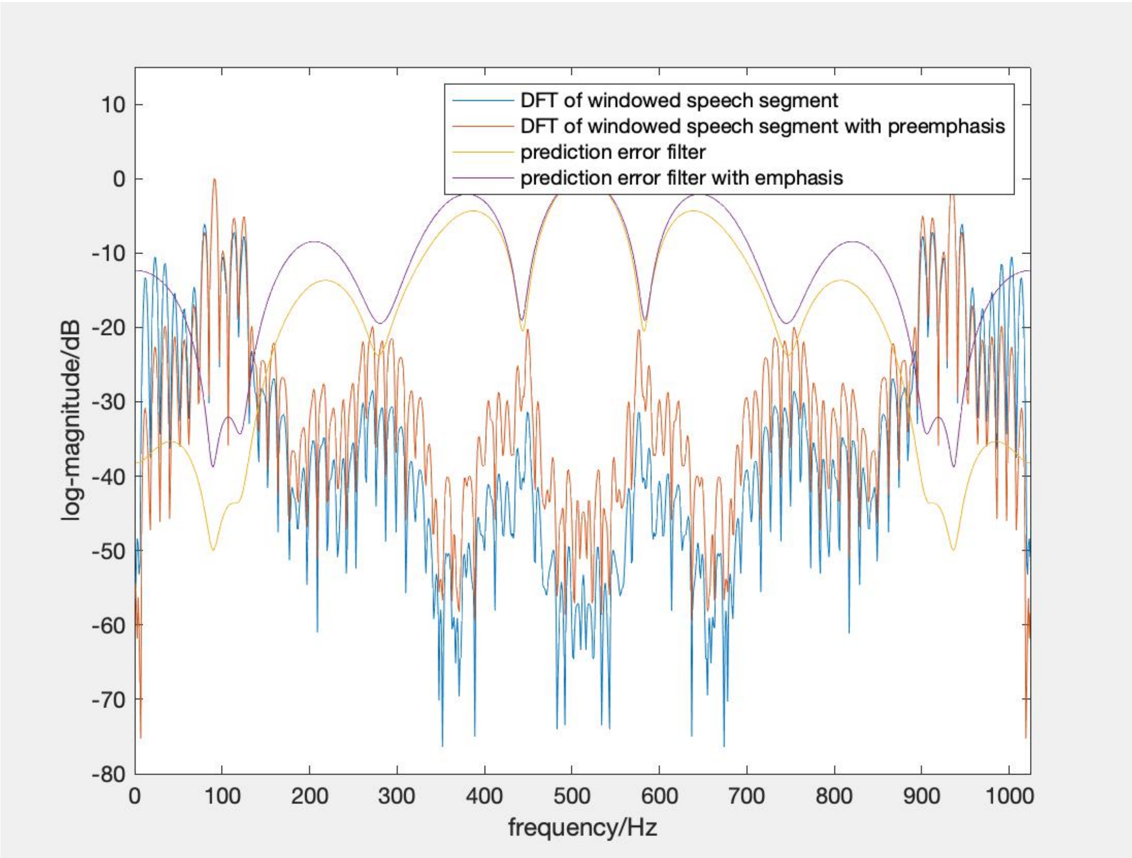
Exercise6

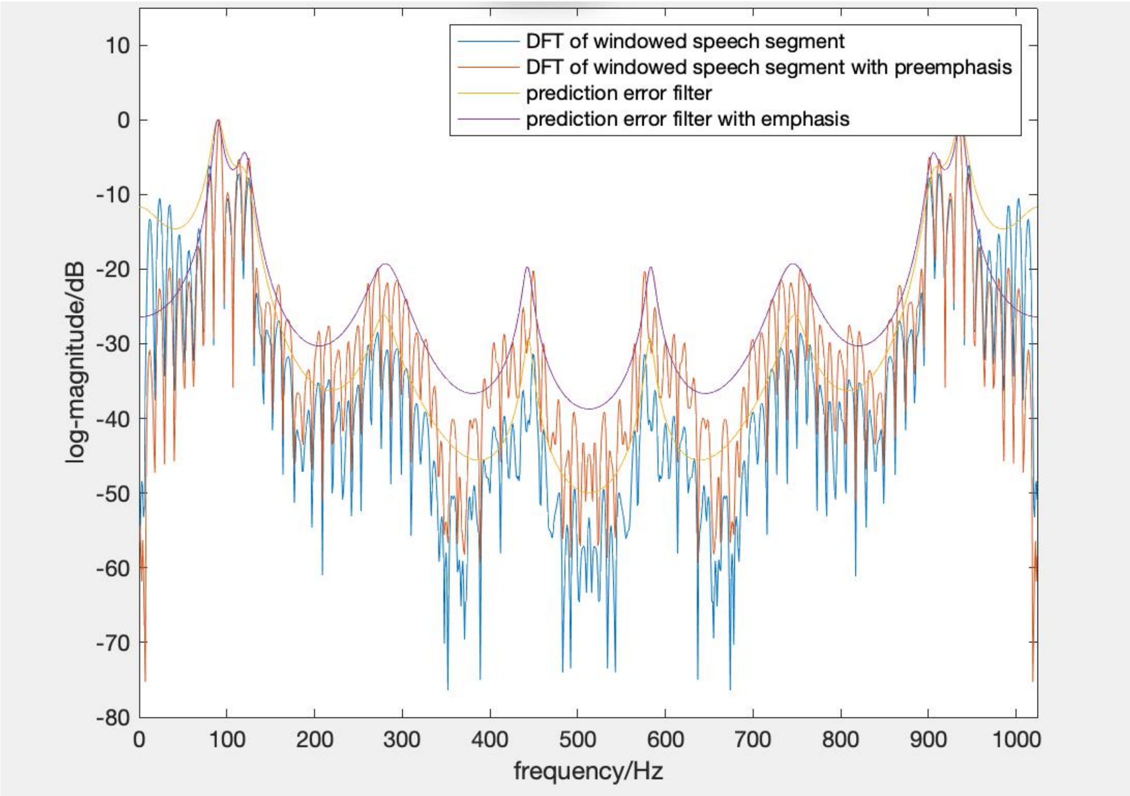
```
clear all;
load('s5.mat');
O=s5(7000:7500);

N=320;
x=linspace(1,320,N);
h=hamming(N);

%enframe O
x1=enframe(O,hamming(N),160);
so=x1(2,:)';
p=12;
[AO1,GO1,rO1,aO1]=autolpc(so,p);
%frequency response of prediction error filter of O
B=1;
%p=12
[HO1,wO1]=freqz(AO1,B,1024,'whole');
%magnitude
HfO1=abs(HO1);
HfO1=HfO1/max(HfO1);
HfO1=20*log10(HfO1);

%frequency response of vocal tract filter of O
B=1;
%p=12
[HO11,wO11]=freqz(B,AO1,1024,'whole');
%magnitude
HfO11=abs(HO11);
HfO11=HfO11/max(HfO11);
HfO11=20*log10(HfO11);

%preemphasis
Opre=filter([1,-0.98],1,O);
x2=enframe(Opre,hamming(N),160);
sopre=x2(2,:)';
p=12;
```

```
[AO2,GO2,rO2,aO2]=autolpc(sopre,p);
%frequency response of prediction error filter of O
B=1;
%p=12
[HO2,wO2]=freqz(AO2,B,1024,'whole');
%magnitude
HfO2=abs(HO2);
HfO2=HfO2/max(HfO2);
HfO2=20*log10(HfO2);
%frequency response of prediction error filter of O
B=1;
%p=12
[HO21,wO21]=freqz(B,AO2,1024,'whole');
%magnitude
HfO21=abs(HO21);
HfO21=HfO21/max(HfO21);
HfO21=20*log10(HfO21);

%DFT of windowed segment of speech O
so_fft=fft(so,1024);
so_fft1=abs(so_fft);
so_fft1=so_fft1/max(so_fft1);
so_mag=20*log10(so_fft1);

%DFT of windowed segment of speech O with preemphasis
sopre_fft=fft(sopre,1024);
sopre_fft1=abs(sopre_fft);
sopre_fft1=sopre_fft1/max(sopre_fft1);
sopre_mag=20*log10(sopre_fft1);

%plot
figure(1);
title('O');
plot(so_mag);
xlabel('frequency/Hz');
ylabel('log-magnitude/dB');
axis([0,1024,-80,15]);
hold on;
plot(sopre_mag);
hold on;
plot(HfO1);
hold on;
plot(HfO2);
hold on;
legend('DFT of windowed speech segment','DFT of windowed speech segment with
preemphasis','prediction error filter','prediction error filter with emphasis');
```

```
%plot
figure(2);
title('O');
plot(so_mag);
xlabel('frequency/Hz');
ylabel('log-magnitude/dB');
axis([0,1024,-80,15]);
hold on;
plot(sopre_mag);
hold on;
plot(HfO11);
hold on;
plot(HfO21);
hold on;
legend('DFT of windowed speech segment','DFT of windowed speech segment with preemphasis','prediction error filter','prediction error filter with emphasis');
```

Result:





Analysis:

a. The filter is a first-order FIR high-pass digital filter, where α is pre-emphasis coefficient and $0.9 < \alpha < 1$.

b. Pre-emphasis on the input digital voice signal is to emphasize the high-frequency part of the voice, remove the influence of lip radiation, and increase the high-frequency resolution of the voice. The frequency spectrum after pre-emphasis has been improved in the high frequency part.