Table of Contents

PROJECT 1 MATLAB CODE

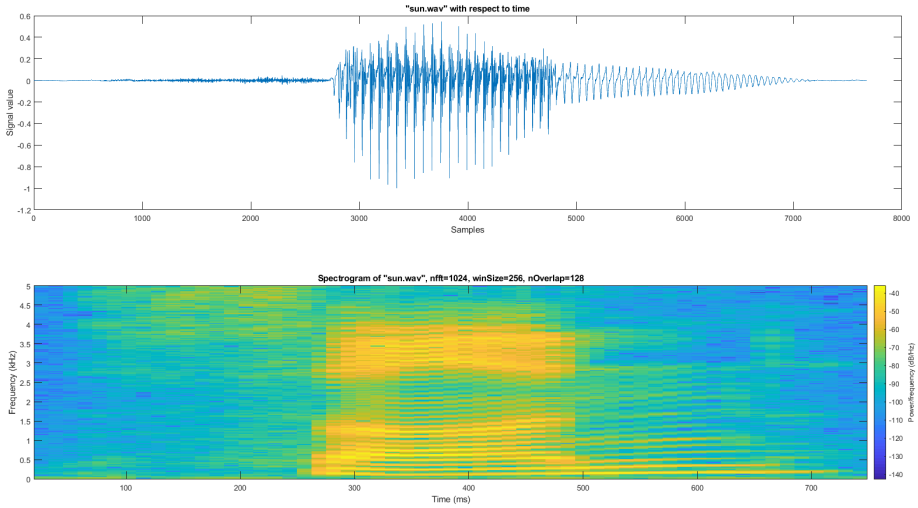
Matt Ruffner EE699 Speech Processing Feb. 9 2019 University of Kentucky

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% INPUT FILE: sun.wav
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Supported frame sized are 256 and 512
```

1. PREPROCESSING

```
%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
% READ SIGNAL AND SAMPLE RATE
X=audioread('sun.wav');
Xinfo=audioinfo('sun.wav');
Fs=Xinfo.SampleRate;

% CHOOSE FRAME SIZE, THESE TWO SUPPORTED BY THE SCRIPT
% 256 -> 25ms CHUNKS
% 512 -> 50ms CHUNKS
Fr=512;
Nfft=Fr;

% ZERO MEAN THE SIGNAL
X=X-mean(X);

% PLOT TIME SERIES AND SPECTROGRAM
figure(1)
subplot(2,1,1)
plot(X)
title('"sun.wav" with respect to time');
xlabel('Samples');
ylabel('Signal value');
subplot(2,1,2)
spectrogram(X,Fr,Fr*.75,Nfft,Fs,'yaxis')
```

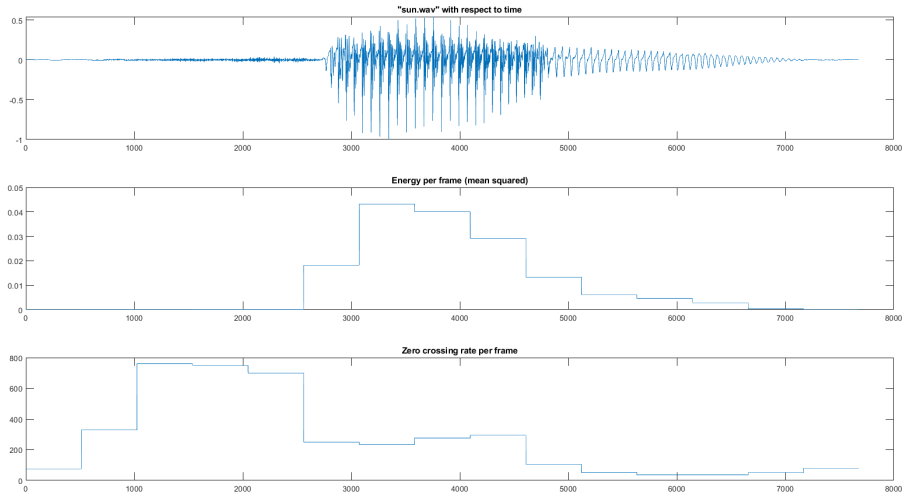
```
title('Spectrogram of "sun.wav", nfft=1024, winSize=256,
 nOverlap=128');
```



2. ENERGY AND ZERO CROSSING RATE

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% ZERO PAD TO MAKE EVEN MULTIPLE OF Fr
X=[X; zeros(length(X)-floor(length(X)/Fr)*Fr,1)];

% RESHAPE TO COLUMNS REPRESENTING Fr SAMPLE CHUNKS
Xk=reshape(X,Fr,length(X)/Fr);

% COMPUTE MEAN SQUARED SIGNAL AMPLITUDE
Xpow=mean(Xk.^2);
% FORMAT FOR PLOTTING
Xpowp=[];
for i=1:length(Xpow)
   Xpowp=[Xpowp; repmat(Xpow(i),Fr,1)];
end

% COMPUTE ZERO CROSSING RATE
Xzcr=[];
for i=1:size(Xk,2)
    a=Xk(:,i);
    Xzcr=[Xzcr; sum(abs(sign(a(2:Fr))-sign(a(1:Fr-1))))];
end
% FORMAT FOR PLOTTING
Xzcrp=[];
for i=1:length(Xzcr)
   Xzcrp=[Xzcrp; repmat(Xzcr(i),Fr,1)];
end
```

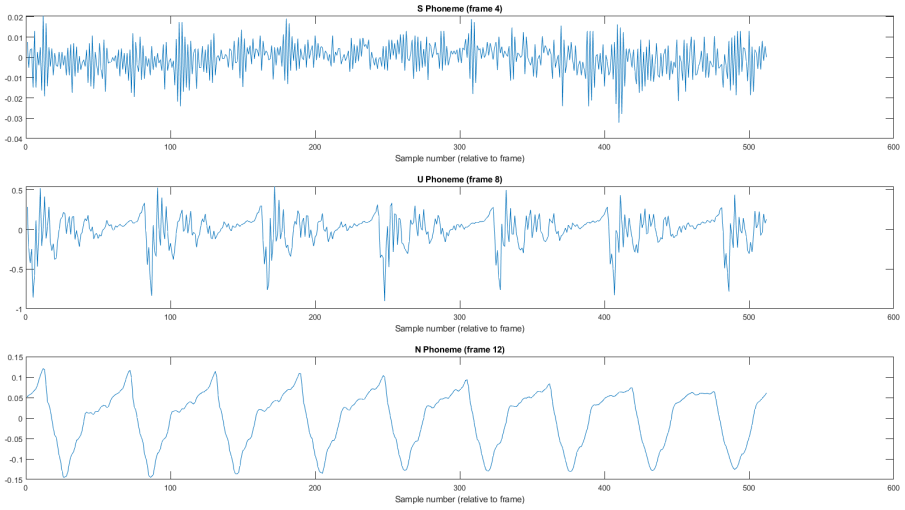
```
figure(2)
subplot(3,1,1)
plot(X)
title('"sun.wav" with respect to time');

subplot(3,1,2)
plot(Xpowp)
title('Energy per frame (mean squared)');

subplot(3,1,3)
plot(Xzcrp)
title('Zero crossing rate per frame')
```



3. PHONEME SELECTION

```
%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%
% PHONEME FRAME CHOICES
if Fr==256
    Sidx=6;
    Uidx=13;
    Nidx=20;
elseif Fr==512
    Sidx=4;
    Uidx=8;
    Nidx=12;
else
    disp('error no Fr (frame size) defined');
end

% PLOT PHONEMES AGAINST TIME (SAMPLE #) AXIS
figure(3)
subplot(3,1,1)
Ph1=Xk(:,Sidx);
```

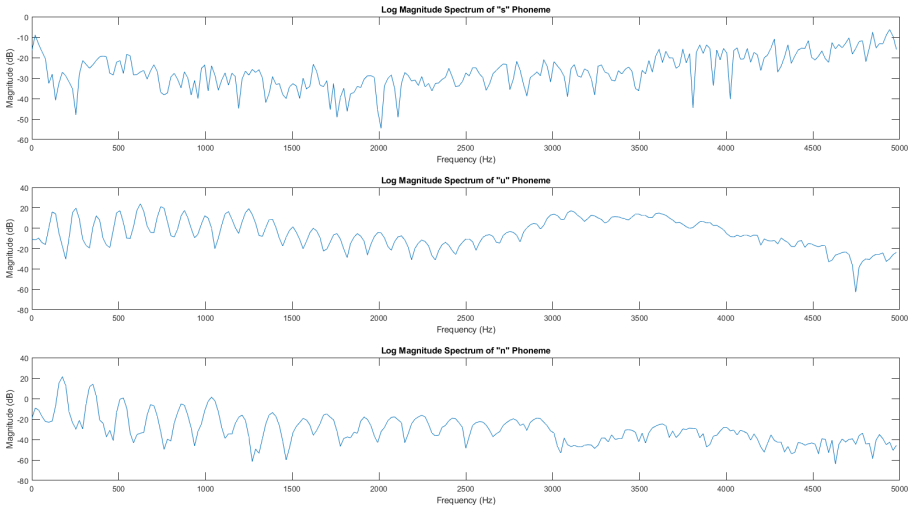
```
plot(Ph1)
title('S Phoneme (frame 4)')
xlabel('Sample number (relative to frame)')

subplot(3,1,2)
Ph2=Xk(:,Uidx);
plot(Ph2)
title('U Phoneme (frame 8)');
xlabel('Sample number (relative to frame)')

subplot(3,1,3)
Ph3=Xk(:,Nidx);
plot(Ph3)
title('N Phoneme (frame 12)');
xlabel('Sample number (relative to frame)')
```



4. LOG MAGNITUDE FFT

```
%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%
% APPLY A HAMMING WINDOW TO EACH PHONEME AND COMPUTE THE LOG MAGNITUDE
W=hamming(Fr);
Nfft=Fr;
%Fax=Fs/2*[0:1/Nfft:1-1/Nfft];
Fax=Fs*[0:Nfft-1]/Nfft;

% COMPUTE PHONEME LOG MAGNITUDE
Ph1mag=20*log10(abs(fft(Ph1.*W,Nfft)));
Ph2mag=20*log10(abs(fft(Ph2.*W,Nfft)));
Ph3mag=20*log10(abs(fft(Ph3.*W,Nfft)));

% PLOT LOG MAGNITUDES OF PHONEMES
figure(4)
subplot(3,1,1)
```

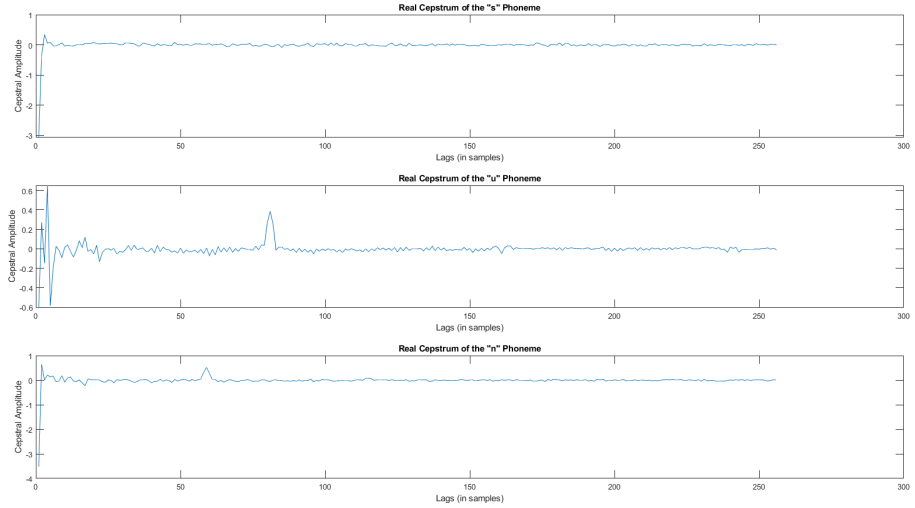
```
plot(Fax(1:length(Fax)/2),Ph1mag(1:length(Ph1mag)/2))
title('Log Magnitude Spectrum of "s" Phoneme')
xlabel('Frequency (Hz)')
ylabel('Magnitude (dB)')
subplot(3,1,2)
plot(Fax(1:length(Fax)/2),Ph2mag(1:length(Ph2mag)/2))
title('Log Magnitude Spectrum of "u" Phoneme')
xlabel('Frequency (Hz)')
ylabel('Magnitude (dB)')
subplot(3,1,3)
plot(Fax(1:length(Fax)/2),Ph3mag(1:length(Ph3mag)/2))
title('Log Magnitude Spectrum of "n" Phoneme')
xlabel('Frequency (Hz)')
ylabel('Magnitude (dB)')
```



5. CEPSTRUM

```
%%%%%%%%%%%%%
%%%%%%%%%%%%%

Ph1cep=ifft(log(abs(fft(W.*Ph1,Fr))));
Ph2cep=ifft(log(abs(fft(W.*Ph2,Fr))));
Ph3cep=ifft(log(abs(fft(W.*Ph3,Fr))));

figure(5)
subplot(3,1,1)
plot(Ph1cep(1:length(Ph1cep)/2))
title('Real Cepstrum of the "s" Phoneme')
xlabel('Lags (in samples)')
ylabel('Cepstral Amplitude')
subplot(3,1,2)
plot(Ph2cep(1:length(Ph2cep)/2))
title('Real Cepstrum of the "u" Phoneme')
xlabel('Lags (in samples)')
```

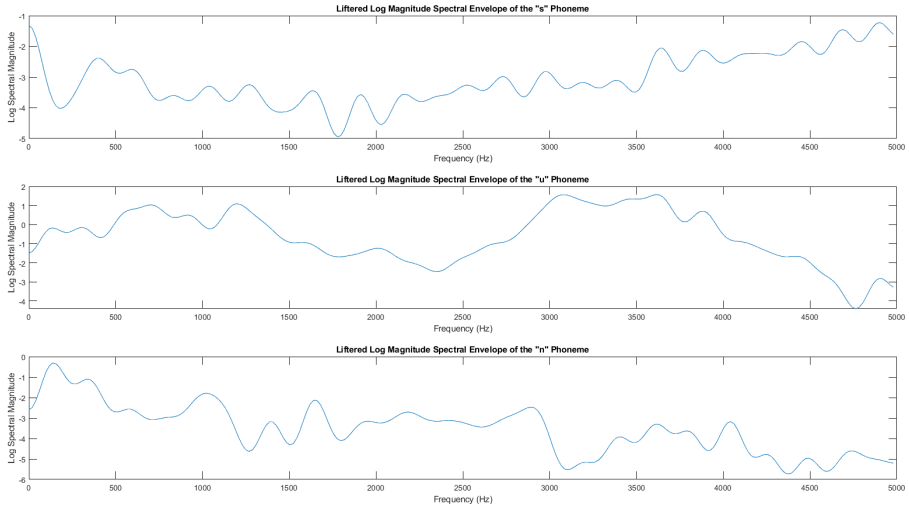
```
ylabel('Cepstral Amplitude')
subplot(3,1,3)
plot(Ph3cep(1:length(Ph3cep)/2))
title('Real Cepstrum of the "n" Phoneme')
xlabel('Lags (in samples)')
ylabel('Cepstral Amplitude')
```



6. LIFTERED SPECTRUM

```
%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%

% SELECT A SINGLE CEPSTRUM CUTOFF TO SEPARATE VOCAL TRACT AND
 EXCITATION
if Fr==256
    Cepc=30;
else
    Cepc=50;
end

% SAVE ONLY LOW FREQ VOCAL TRACT INFO, ZERO OUT THE REST SYMMETRICALLY
Ph1cep=[Ph1cep(1:50);zeros(Fr-Cepc*2-1,1);Ph1cep(length(Ph1cep)-
Cepc:end)];
Ph2cep=[Ph2cep(1:50);zeros(Fr-Cepc*2-1,1);Ph2cep(length(Ph2cep)-
Cepc:end)];
Ph3cep=[Ph3cep(1:50);zeros(Fr-Cepc*2-1,1);Ph3cep(length(Ph3cep)-
Cepc:end)];

% LIFTER THE SPECTRA
Ph1Lft=fft(Ph1cep);
Ph2Lft=fft(Ph2cep);
Ph3Lft=fft(Ph3cep);

% PLOT THE LIFTERED SPECTRA
```

```
figure(6)
subplot(3,1,1)
plot(Fax(1:length(Fax)/2),real(Ph1Lft(1:length(Ph1Lft)/2)))
title('Liftered Log Magnitude Spectral Envelope of the "s" Phoneme')
xlabel('Frequency (Hz)')
ylabel('Log Spectral Magnitude')
subplot(3,1,2)
plot(Fax(1:length(Fax)/2),real(Ph2Lft(1:length(Ph2Lft)/2)))
title('Liftered Log Magnitude Spectral Envelope of the "u" Phoneme')
xlabel('Frequency (Hz)')
ylabel('Log Spectral Magnitude')
subplot(3,1,3)
plot(Fax(1:length(Fax)/2),real(Ph3Lft(1:length(Ph3Lft)/2)))
title('Liftered Log Magnitude Spectral Envelope of the "n" Phoneme')
xlabel('Frequency (Hz)')
ylabel('Log Spectral Magnitude')
```

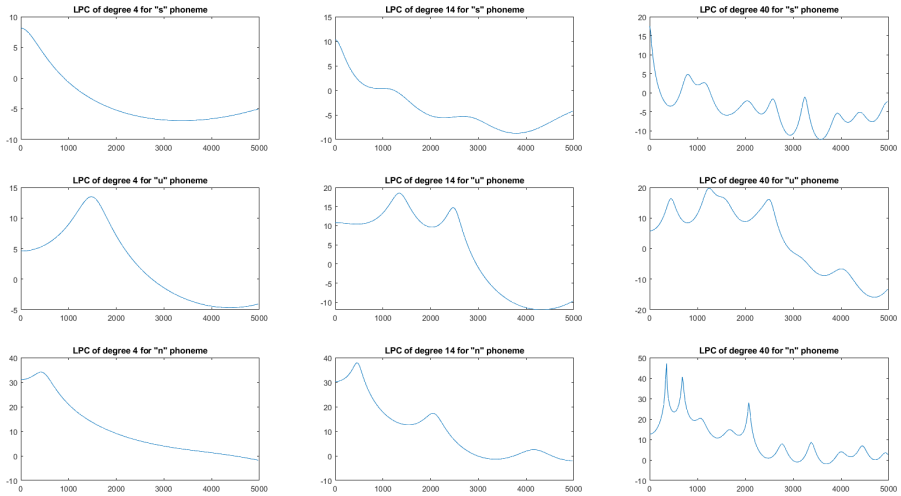


8. LPC SPECTRA

```
%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%

Ph1LPC4= lpc(W.*Ph1,4);
Ph1LPC14=lpc(W.*Ph1,14);
Ph1LPC40=lpc(W.*Ph1,40);

Ph2LPC4= lpc(W.*Ph2,4);
Ph2LPC14=lpc(W.*Ph2,14);
Ph2LPC40=lpc(W.*Ph2,40);

Ph3LPC4=lpc(W.*Ph3,4);
Ph3LPC14=lpc(W.*Ph3,14);
Ph3LPC40=lpc(W.*Ph3,40);
```

```
figure(7)

subplot(3,3,1)
lpcr=20*log10(abs(freqz(1,Ph1LPC4)));
lpcr=lpcr(1:length(lpcr)/2);
plot(Fax(1:length(Fax)/2),lpcr)
title('LPC of degree 4 for "s" phoneme')

subplot(3,3,2)
lpcr=20*log10(abs(freqz(1,Ph1LPC14)));
lpcr=lpcr(1:length(lpcr)/2);
plot(Fax(1:length(Fax)/2),lpcr)
title('LPC of degree 14 for "s" phoneme')

subplot(3,3,3)
lpcr=20*log10(abs(freqz(1,Ph1LPC40)));
lpcr=lpcr(1:length(lpcr)/2);
plot(Fax(1:length(Fax)/2),lpcr)
title('LPC of degree 40 for "s" phoneme')

subplot(3,3,4)
lpcr=20*log10(abs(freqz(1,Ph2LPC4)));
lpcr=lpcr(1:length(lpcr)/2);
plot(Fax(1:length(Fax)/2),lpcr)
title('LPC of degree 4 for "u" phoneme')

subplot(3,3,5)
lpcr=20*log10(abs(freqz(1,Ph2LPC14)));
lpcr=lpcr(1:length(lpcr)/2);
plot(Fax(1:length(Fax)/2),lpcr)
title('LPC of degree 14 for "u" phoneme')

subplot(3,3,6)
lpcr=20*log10(abs(freqz(1,Ph2LPC40)));
lpcr=lpcr(1:length(lpcr)/2);
plot(Fax(1:length(Fax)/2),lpcr)
title('LPC of degree 40 for "u" phoneme')

subplot(3,3,7)
lpcr=20*log10(abs(freqz(1,Ph3LPC4)));
lpcr=lpcr(1:length(lpcr)/2);
plot(Fax(1:length(Fax)/2),lpcr)
title('LPC of degree 4 for "n" phoneme')

subplot(3,3,8)
lpcr=20*log10(abs(freqz(1,Ph3LPC14)));
lpcr=lpcr(1:length(lpcr)/2);
plot(Fax(1:length(Fax)/2),lpcr)
title('LPC of degree 14 for "n" phoneme')

subplot(3,3,9)
lpcr=20*log10(abs(freqz(1,Ph3LPC40)));
lpcr=lpcr(1:length(lpcr)/2);
plot(Fax(1:length(Fax)/2),lpcr)
```

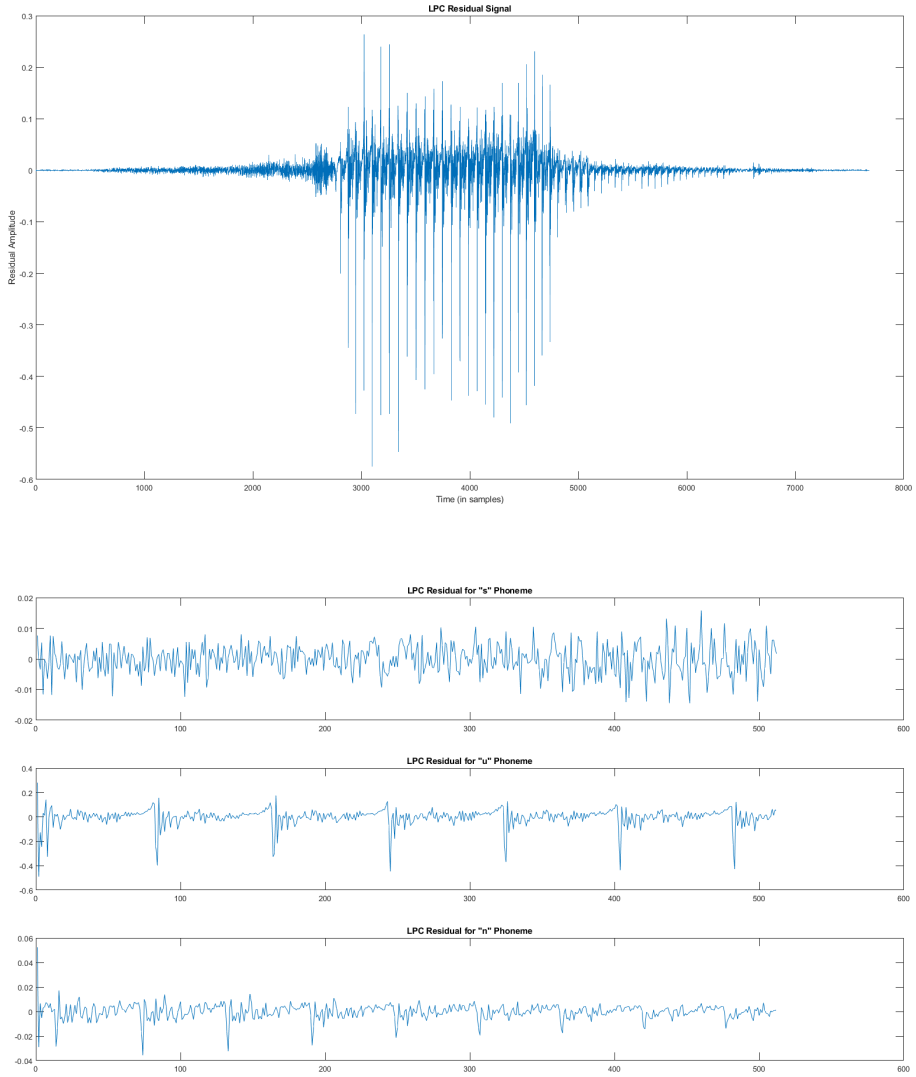
```
title('LPC of degree 40 for "n" phoneme')
```



8. LPC RESIDUAL

```
%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%


Nfrms=length(X)/Fr;
Xres=[];
for i=1:Nfrms
    res=lpc(Xk(:,i),14);
    if i==1
        temp=filter(res,1,Xk(:,i));
        Xres(i,:)=temp;
    else
        temp=filter(res,1,[Xk(:,i-1);Xk(:,i)]);
        Xres(i,:)=temp(Fr+1:end);
    end
end

figure(8);
plot(reshape(Xres',Fr*Nfrms,1))
title('LPC Residual Signal')
xlabel('Time (in samples)')
ylabel('Residual Amplitude')

Ph1r=lpc(Ph1,14);
Ph2r=lpc(Ph2,14);
Ph3r=lpc(Ph3,14);

Ph1rf=filter(Ph1r,1,Ph1);
Ph2rf=filter(Ph2r,1,Ph2);
Ph3rf=filter(Ph3r,1,Ph3);
```

```
figure(9)
subplot(3,1,1)
plot(Ph1rf)
title('LPC Residual for "s" Phoneme')

subplot(3,1,2)
plot(Ph2rf)
title('LPC Residual for "u" Phoneme')

subplot(3,1,3)
plot(Ph3rf)
title('LPC Residual for "n" Phoneme')
```



Published with MATLAB® R2018b