ARSIG : Spectral Analysis Report

Supervisor : Sebastien Bourguignon
Name : Chaitanya Krishna VIRIYALA

SPECTRAL ANALYSIS

1 Detection in noise and separation of close frequencies

The first part of this subject studies the performance of different spectral analysis methods on simulated data in order to perceive their strengths and limitations. We are interested in the detection and estimation of multiple oscillations in the presence of noise.

The methods will be evaluated on signals composed of three sinusoıdes and noise:

$$y[n] = \underbrace{\alpha_1 \sin(2\pi\lambda_1 n + \varphi_1) + \alpha_2 \sin(2\pi\lambda_2 n + \varphi_2) + \alpha_3 \sin(2\pi\lambda_3 n + \varphi_3)}_{x[n]} + b[n], n = 0, \ldots, N-1, \text{ où :} \tag{1}$$

- the amplitudes are $\alpha_1 = 0.1$, $\alpha_2 = 1$, $\alpha_3 = 1$;
- the frequencies are $\lambda_1 = 0.1$, $\lambda_2 = 0.2$ et $\lambda_3 = 0.2 + \delta\lambda$, or the gap will be varied $\delta\lambda$ (by default, $\lambda_3 = 0.22$) ;

- the phases ϕ_1, ϕ_2, ϕ_3 can be chosen zero;
- the noise samples b[n] are independent, of the same Gaussian law : $\forall n$, $b[n] \sim N(0, \sigma^2)$.

TO DO 1 :

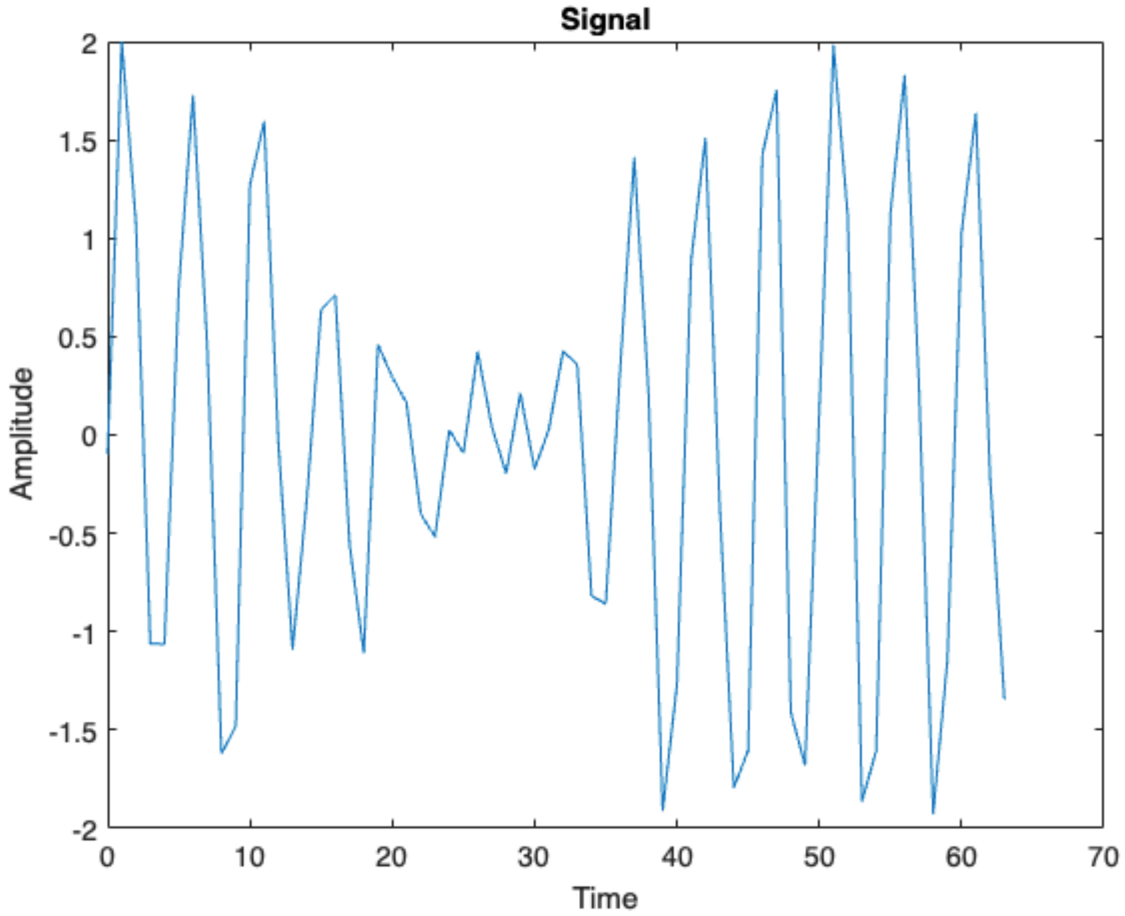
1. What do you think are the difficulties in estimating these three components?

There are a few difficulties in estimating these three components. Some of these are :

- The three signal frequencies are very close, which will raise the level of the difficulty of separating them as we need to get the signal from the noise.

- For autoregressive (AR) modeling and the MUSIC method, we need to choose the proper number P in order to obtain the right results.

2. Write in Matlab a function **x = generate_signal(a, lambda, phi, N)** generating a signal multisinusoıdal x according to equation (1), where the vectors a, lambda, phi contain the values of the corresponding parameters.



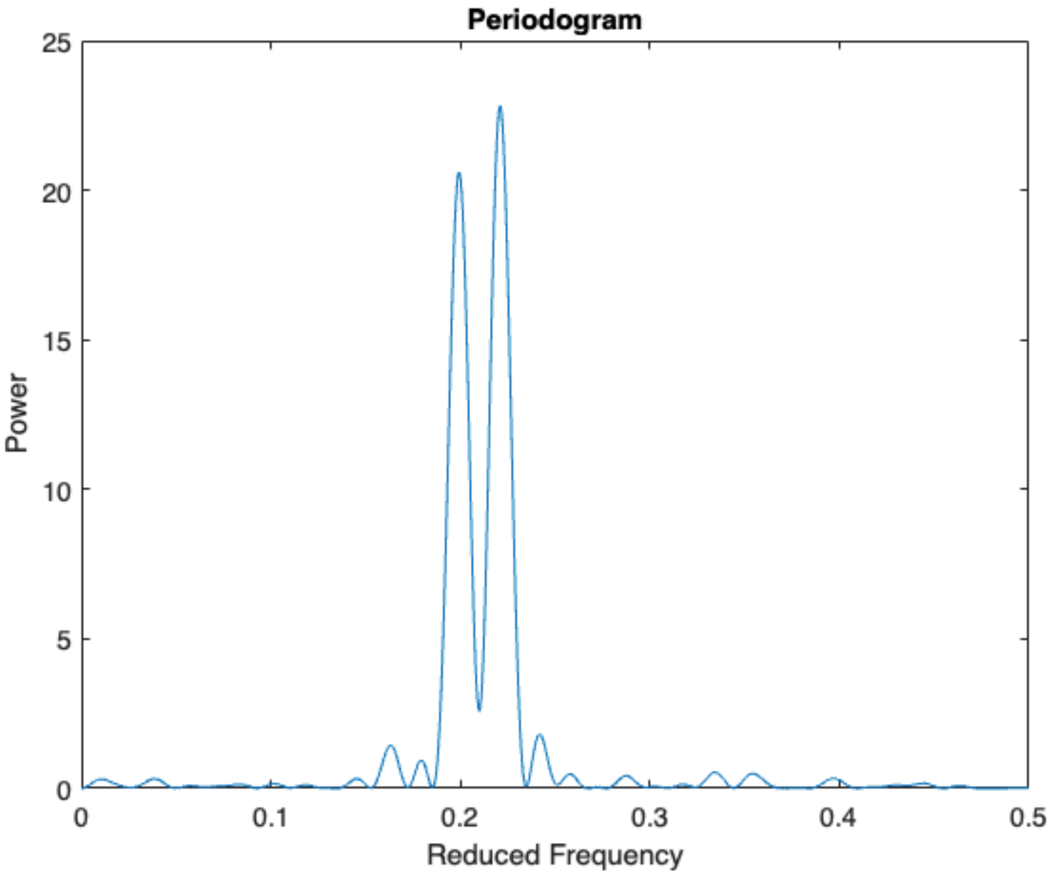
Signal has 3 sinusoids. From the graph, we can see that the amplitude is changing over time.

We will compare the performance of different spectral analysis methods.

For each method, an axis of Nf = 1024 frequencies distributed between 0 and 1 will be used and the display will be restricted to the frequencies of the interval [0.0.5] : ***set(gca,’xlim’,[0,0.5])***

We first consider a noise-free signal with N = 64 points with λ3 = 0.22***.***

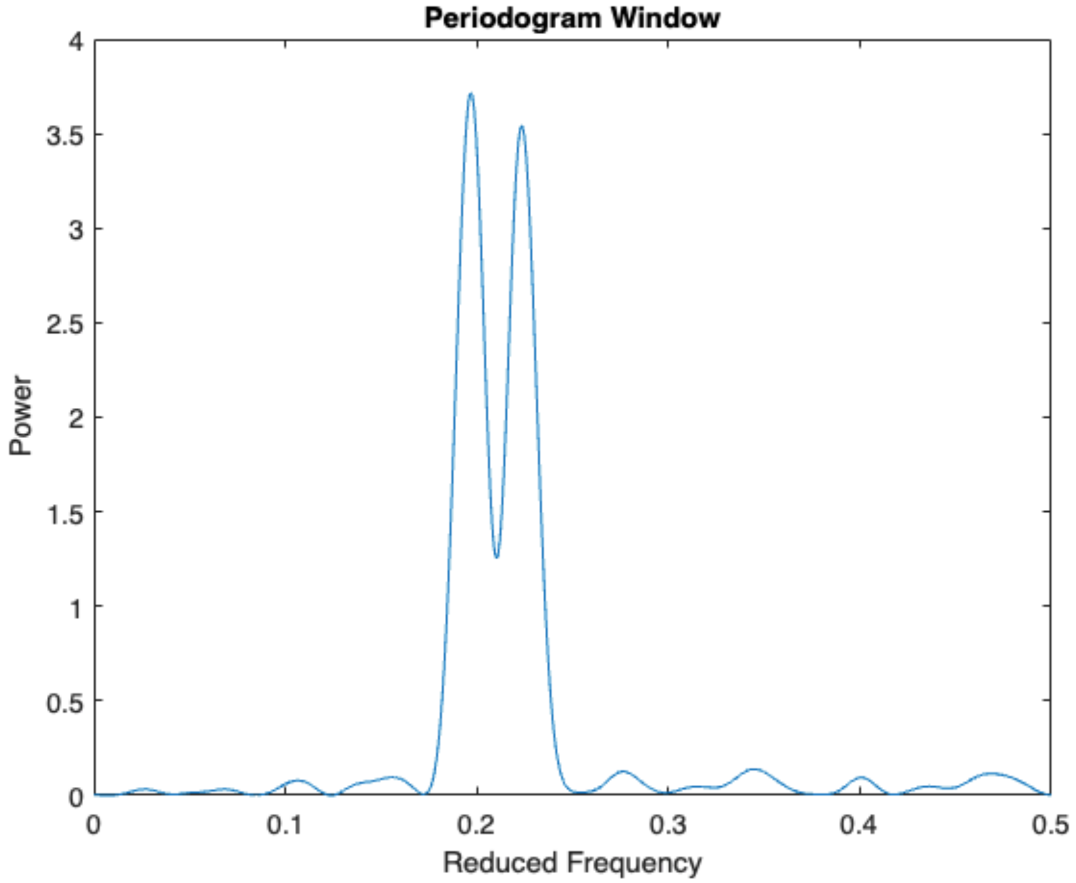
3. Calculate and display the classic periodogram. *Do not use the periodogram function of Matlab... It is calculated in two lines!*



This is the graph of Periodogram. on the X axis, we have Reduced Frequency and Power on the Y axis. The power of a signal is Px = 1/ N×abs(fft(x,Nf)).^{2}

4. Calculate and display the window periodogram using a Hamming window: consider the signal x[n]h[n], where **h = hamming(N)**;

Hamming window can be calculate as h = hamming(N);



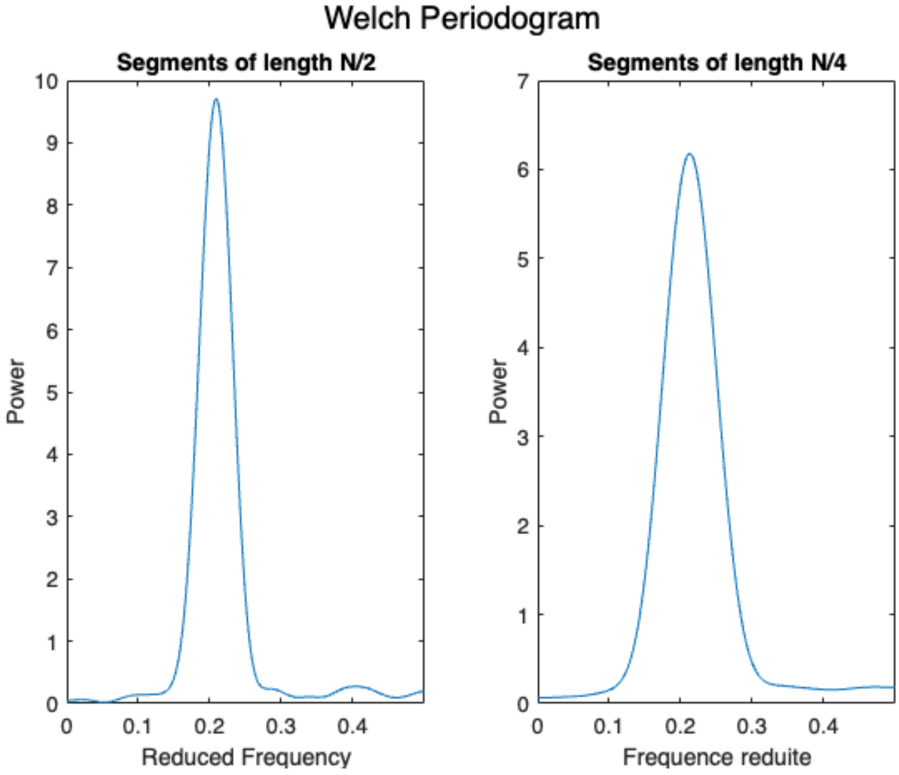
By multiplying the vector h obtained by a Hamming window, we can obtain the window signal x [n] h [n]. Then we calculate its periodogram like the classic periodogram above.

5. Calculate and display the average periodogram and window on a signal split into several segments (Welch periodogram). The Matlab function :

dsp_welch = pwelch(x, window, noverlap, Nf, Fs, 'twosided');

performs this calculation, where window is the size of each segment, no overlap is the number of overlapping points between two segments, Nf is the number of points on the frequency axis and Fs is the sampling frequency (here Fs = 1).

Take, for example, segments of size N/2 or N/4, and an overlap of a quarter of the segment size.



When we take window = N / 2, no overlap is a quarter of a window, which is therefore N / 8, and then by taking window = N / 4, no overlap = N / 8, we get the figure above.

6. Spectral analysis based on auto-regressive (AR) modelling.

- The parameters of the AR model are first estimated with the Matlab function (which minimises error of prediction in the sense of the least squares) :

```
[a_est,sigma2_est] = arcov(x,P);
```

where P is the order of the prediction model.

Attention : The power spectral density model is here :

$$S_x(\lambda, \boldsymbol{a}) = \frac{\sigma^2}{|1 + a_1 e^{-j2\pi\lambda} + \cdots + a_P e^{-j2\pi P\lambda}|^2}$$

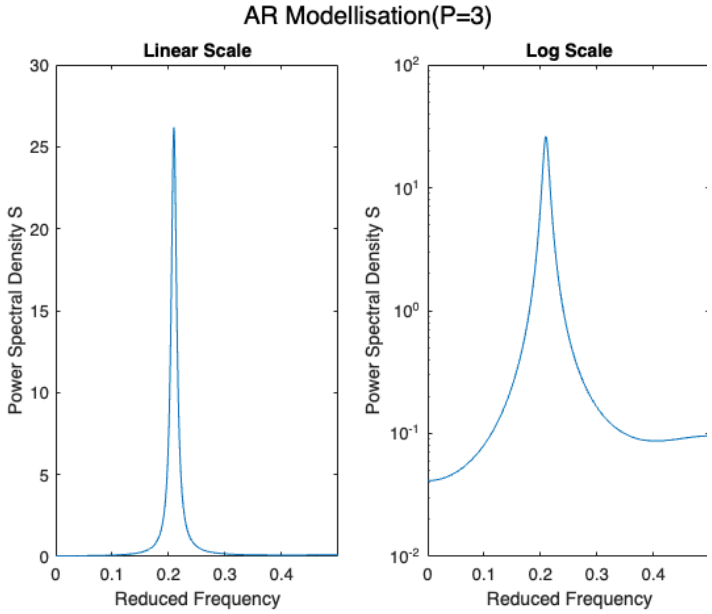
and the vector a is returned´e by Matlab contains the coefficients [1, a1, ... , ... ,, aP]

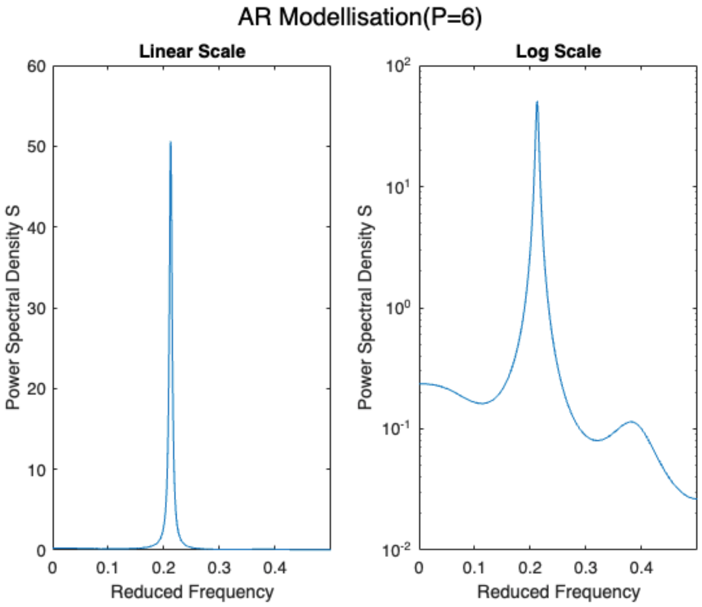
- How to numerically calculate the power spectral density Sx(λ,a) on a frequency {λk} once the parameters have been estimated ?

To calculate the denominator of Sx (lambda, a) once the parameters a have been estimated, we can first consider to calculate them one by one and then to cumulate them, it is therefore d 'use two for loops which iterate over the frequencies and P. Another simpler way is to directly calculate the Fourier transform of a (S = sigma2 est./(abs(fft(a est, Nf))). 2).

- Display the estimated power spectral density.

We vary the value of P in order to get different results, the images on the left are on a linear scale, and the others are on a logarithmic scale.



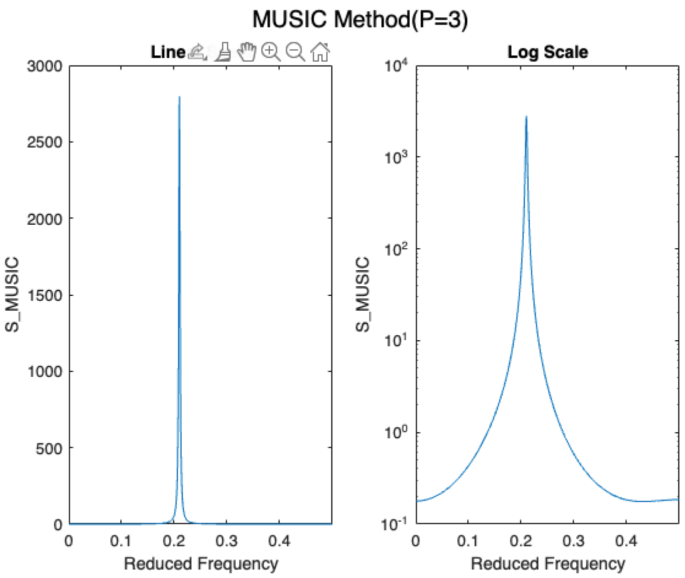


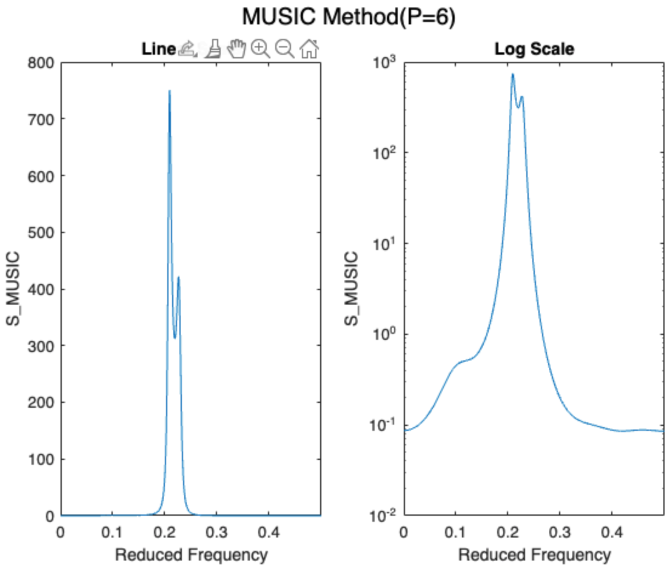
We have calculated the AR modernisation for two different values of P = 3 and 5.

7. Spectral analysis using the MUSIC method. For this method, the Matlab function will be used directly:

```
S_MUSIC = pmusic(x,P,axe_freq,Fs);
```

where P is the number of complex exponentials sought in the model.





We can see that if P = 4, the peak is not clear enough, but for P = 6, the peaks are at lambda = 0.1, 0.2 and somewhere at around 0.2

8. **Estimation of a parsimonious spectrum.** This method is based on the resolution of a shape optimization problem:

$$\widehat{\boldsymbol{u}} = \arg \min_{\boldsymbol{u} \in \mathbb{C}^{Nf}} \frac{1}{2} \|\boldsymbol{x} - \mathbf{W}\boldsymbol{u}\|^2 + \mu_{\text{regul}} \|\boldsymbol{u}\|_1,$$

where the penalty in norm l1 requires that the vector u⍰ (the amplitude spectrum discretized on the grid of Nf frequencies) be parsimonious, i.e., contain many zero values. Function :

umin = minl1 Fourier (x, freq axis, mu regul)

carries out the optimization of the corresponding least squares criterion penalized by the l1 norm of Fourier coefficients. We test at several values of the parameter μregul> 0 until we get a satisfactory solution. Comment.

We test at several values of the parameter μregul> 0 until a satisfactory solution is obtained. After several tests, we find that μregul = 0.6 leading to a minimum of the number of iterations, which is 500. The satisfactory solution is indeed 0.

9. Repeat the previous analysis in the following cases:

- Frequencies λ2 and λ3 closer: λ3 = 0.21 ;
- Presence of noise in the data with a noise level of 20 dB and then 10 dB. Consider first λ3 = 0.22, then λ3 = 0.21 ;

In the presence of noise, restart the simulation several times: due to the random nature of the noise, the methods do not provide the same result from one realisation to the next.

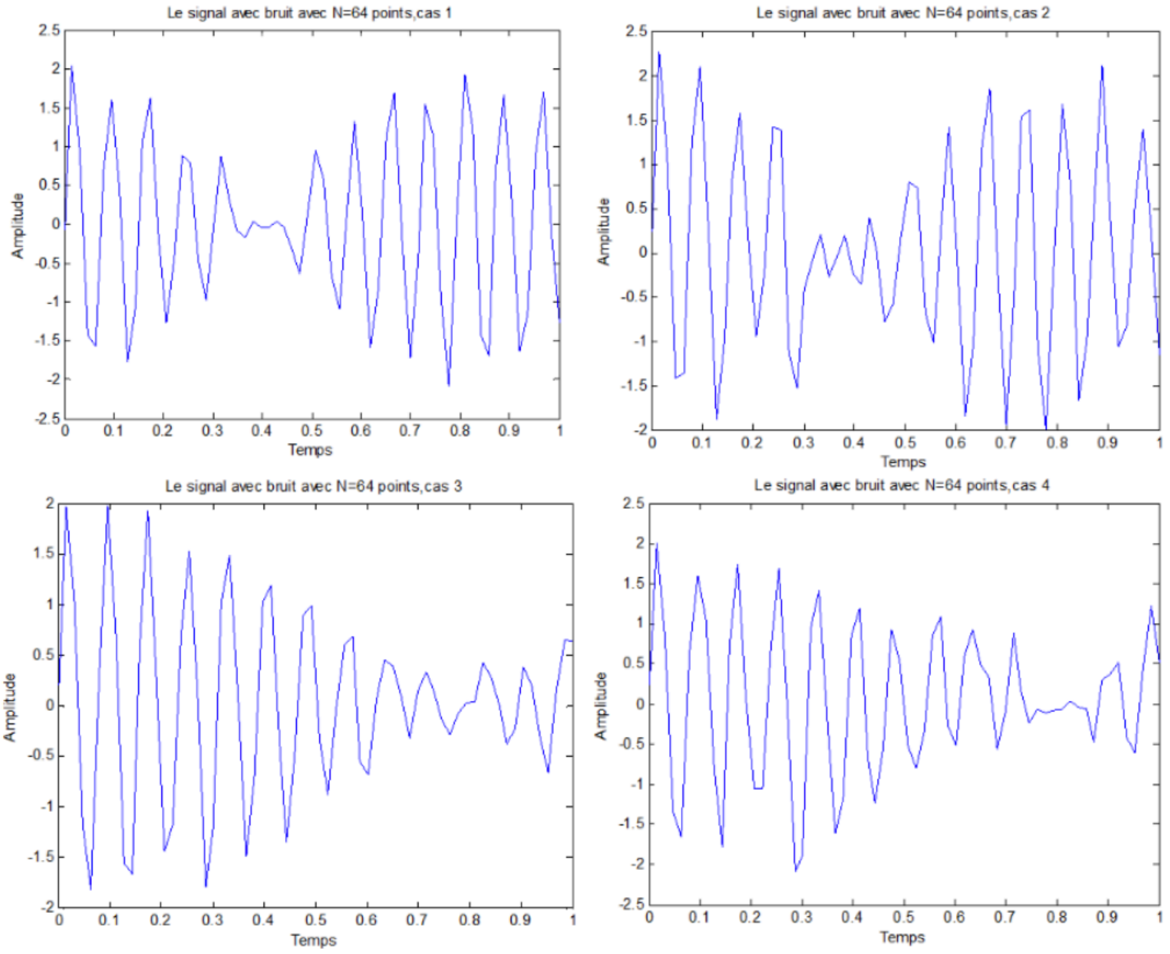
Useful :

% Adds white Gaussian noise to a signal x, of defined power to achieve a given signal-to-noise ratio

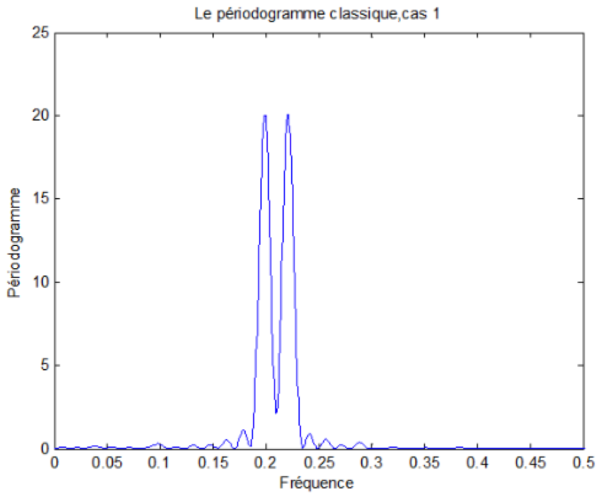
function y = add_noise(x,RSB)

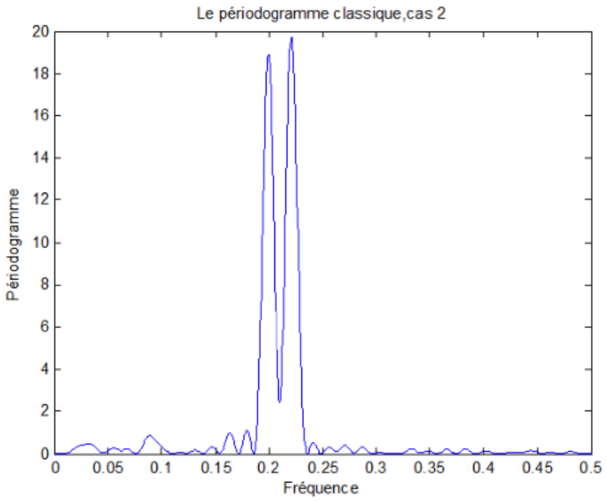
```
N = length(x);

% Signal energy x

E_signal = sum(x.^2);

% noise variance sigma2 such that RSB = 10 log10(E_x/E_noise)

var_noise = E_signal*10^(-RSB/10)/N;

% addition of a random realisation of a Gaussian noise vector iid, of

% variance var_noise

y = x + sqrt(var_noise)*randn(size(x));
```

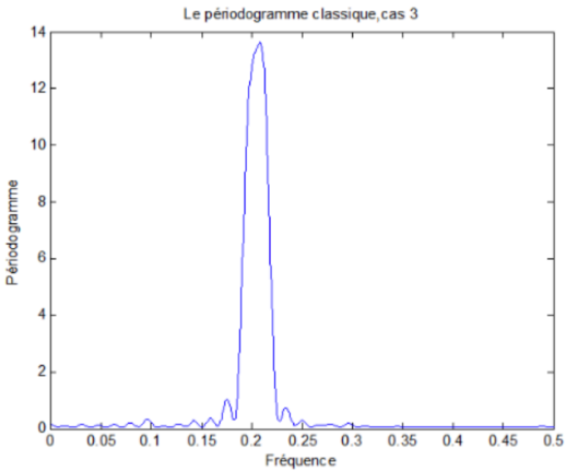
- Repeat the last simulation (RSB=10 dB, λ3 = 0.21) with now N = 512 points.

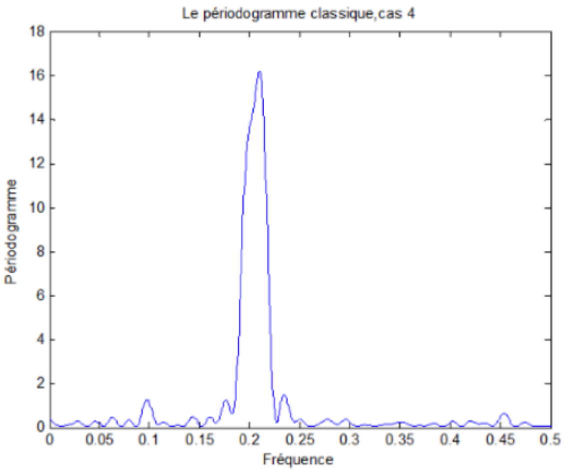


1) Classical Periodogram

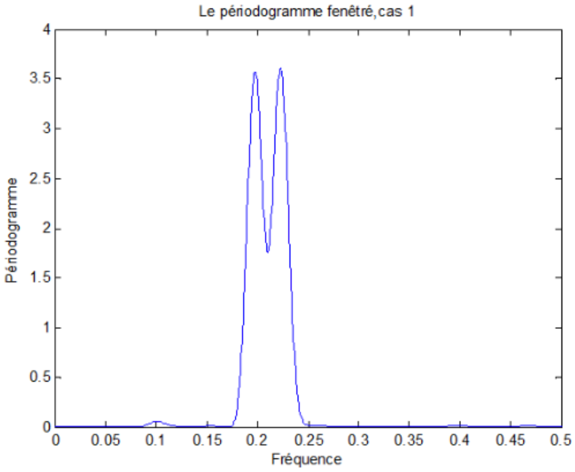


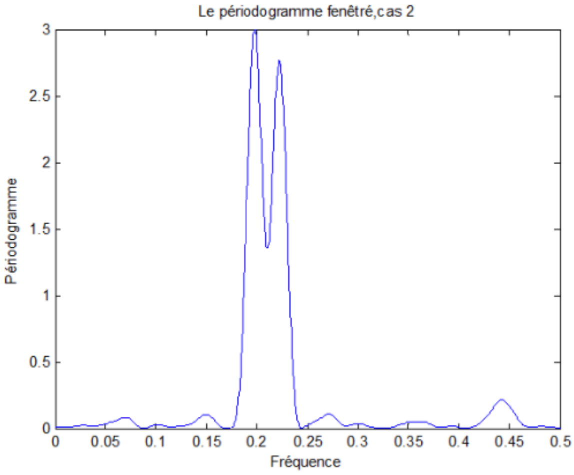


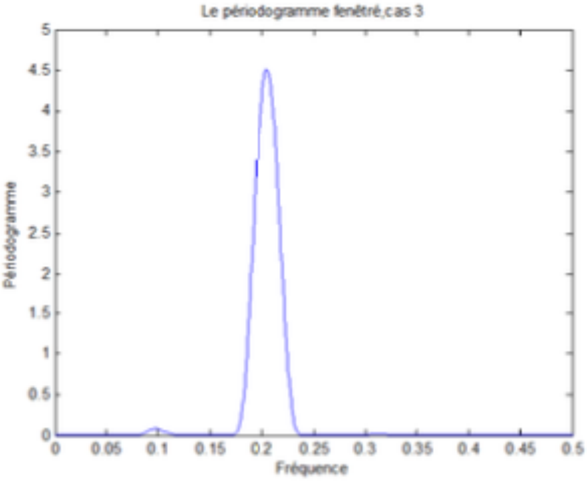


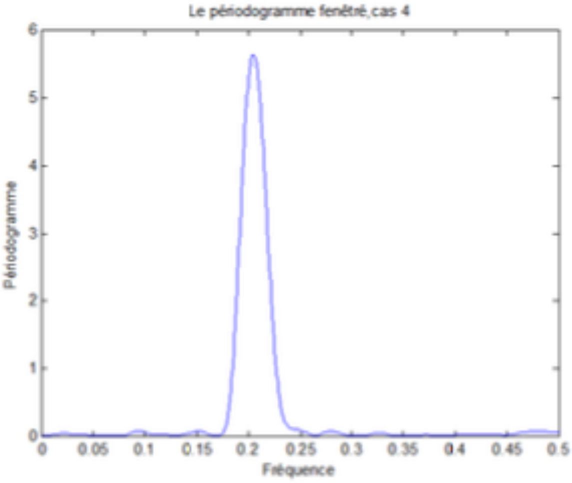


2) With Periodogram window



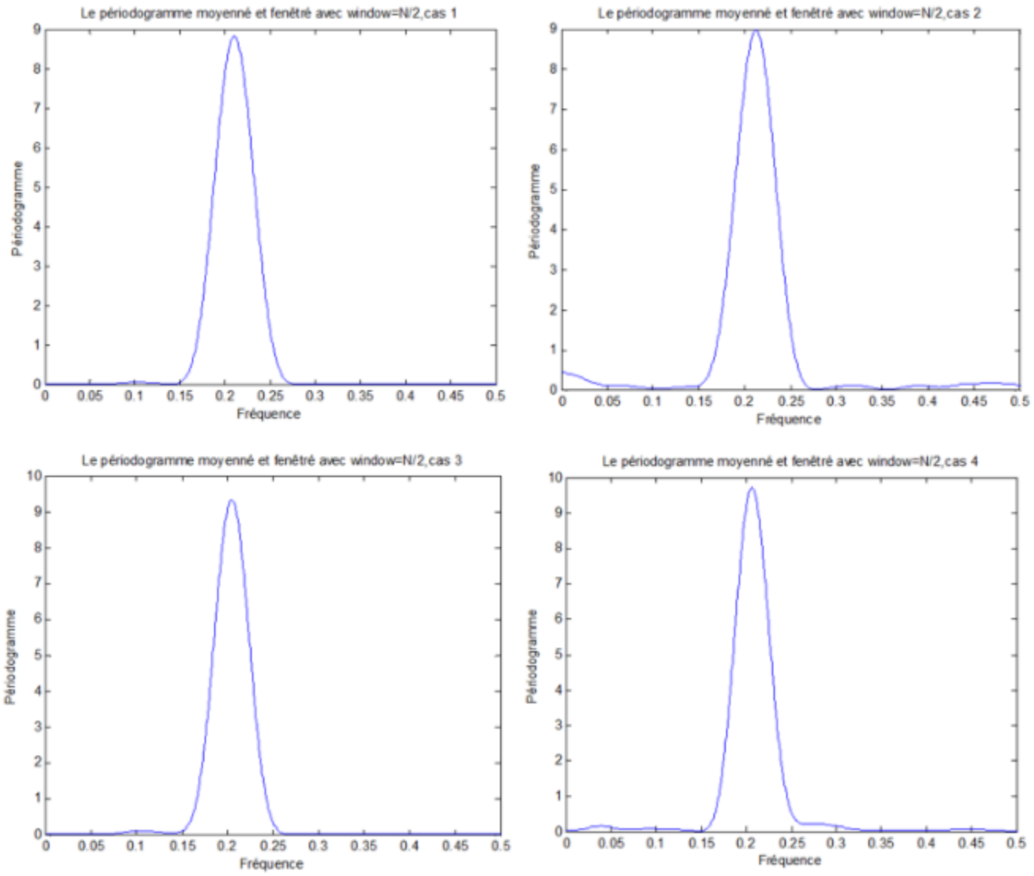




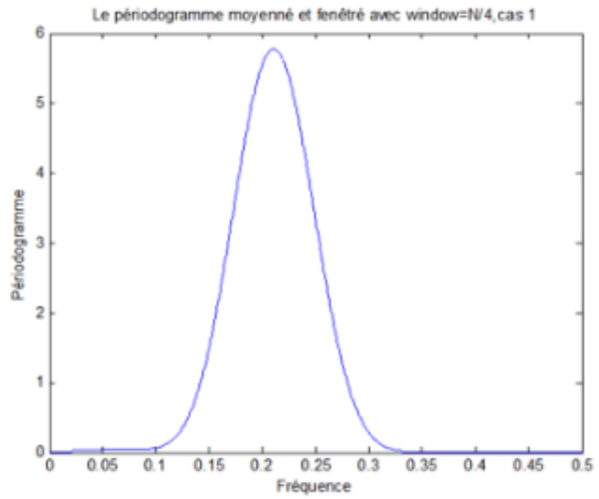


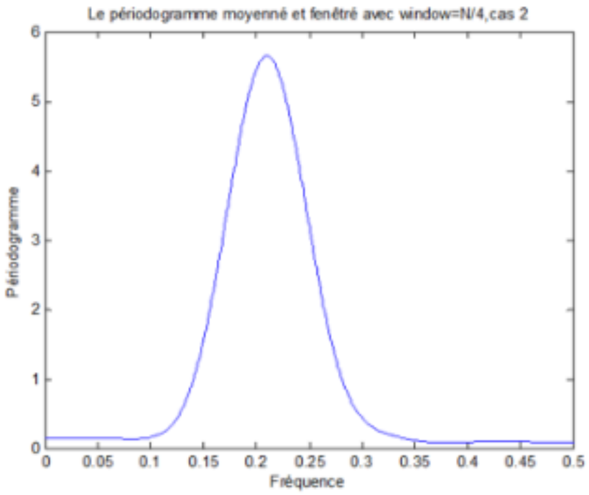
3) Calculation and display of the averaged periodogram and window:

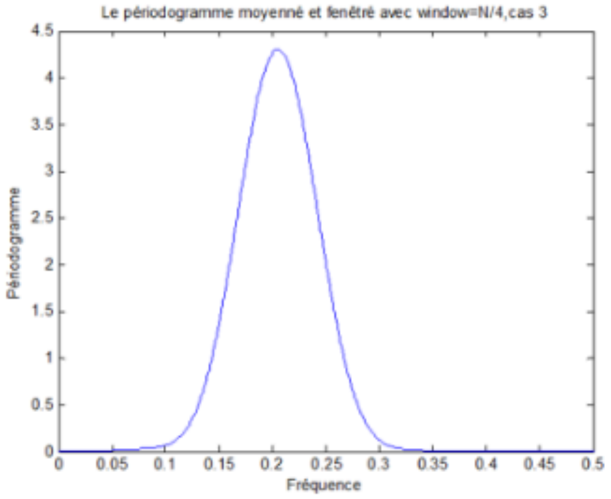
In the case: the size of each segment is N / 2.

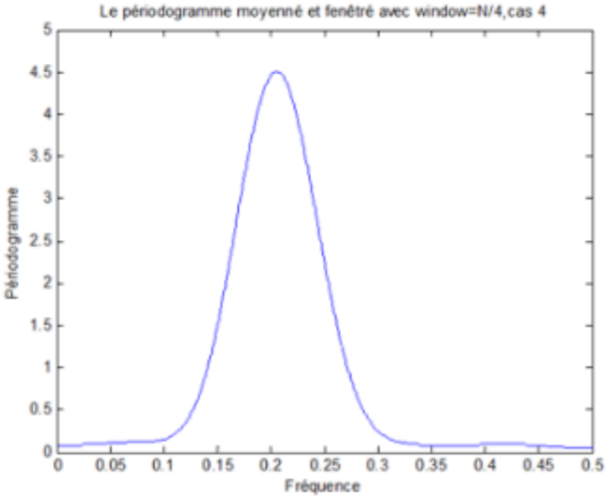


In the case: the size of each segment is N / 4.

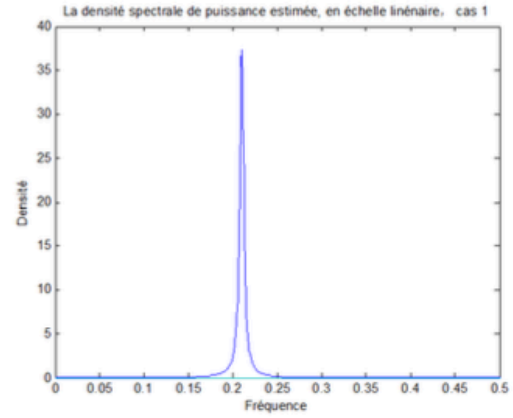


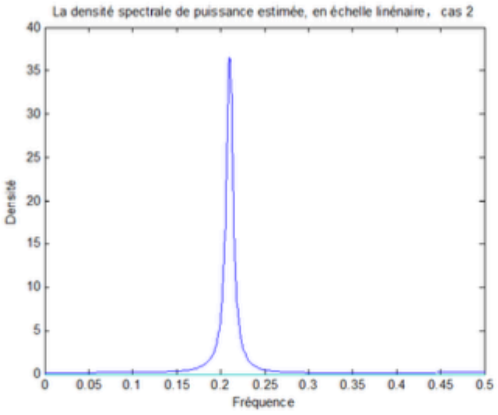


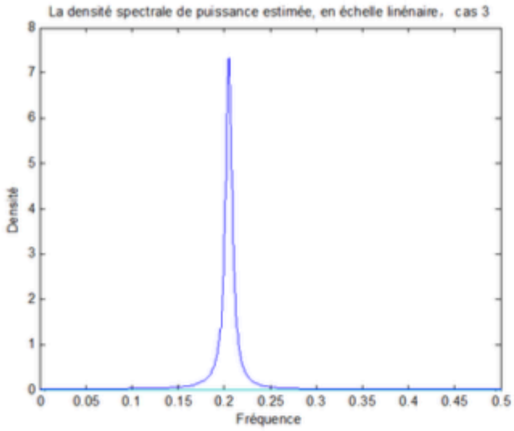


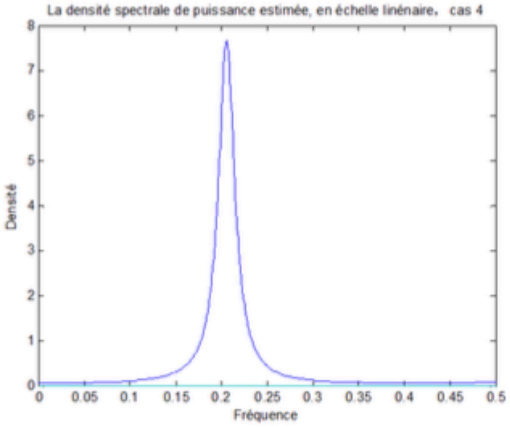


4) Analysis based on a self-regressive (AR) modeling:

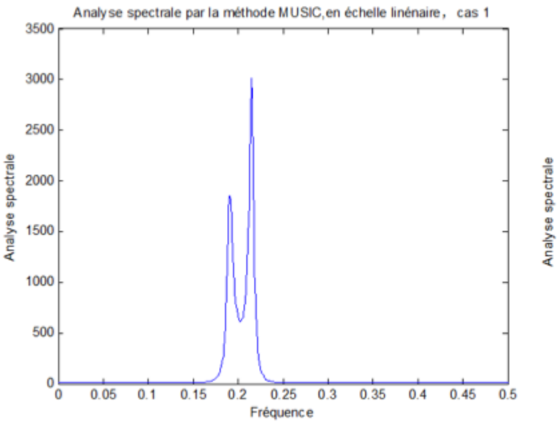


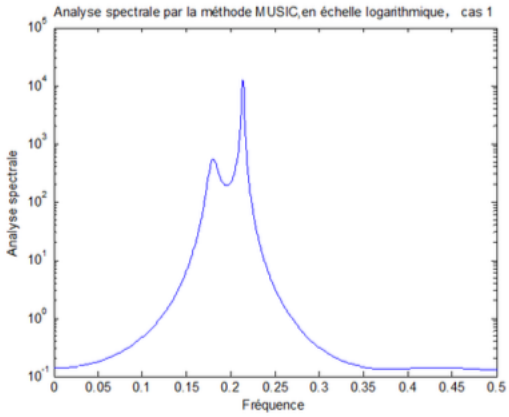


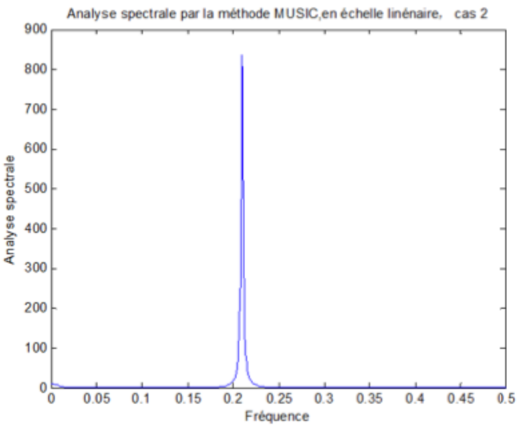


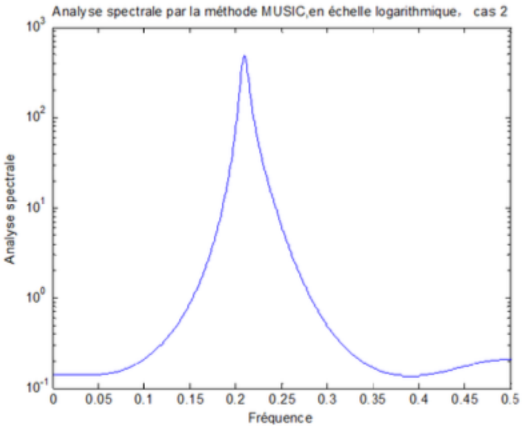


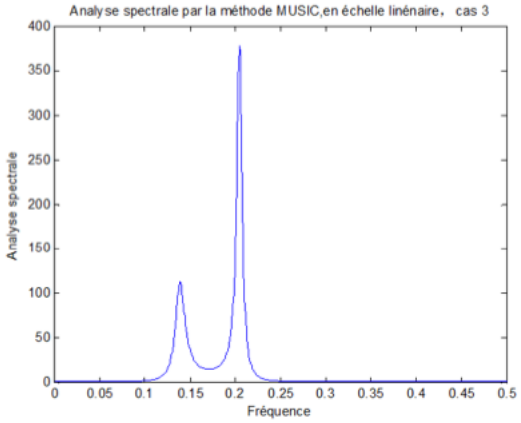
5) Spectral Analysis by MUSIC method :

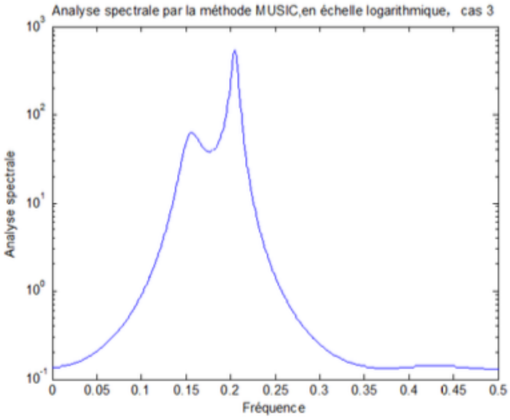


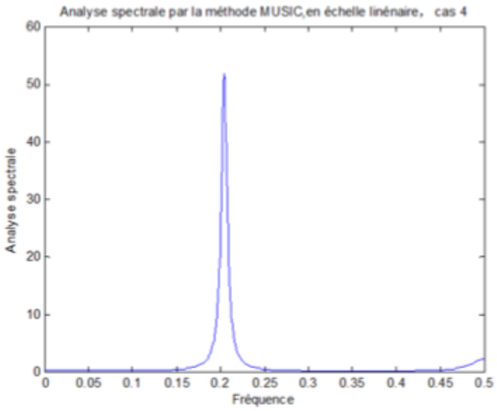


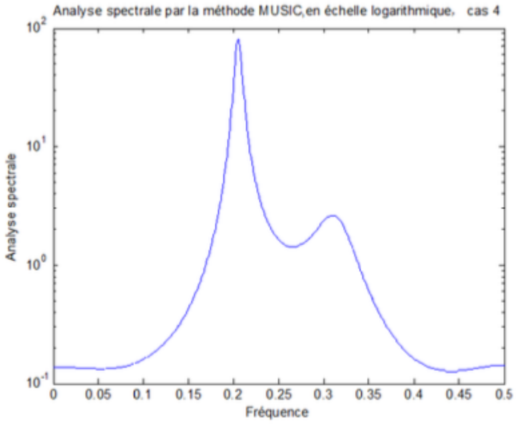










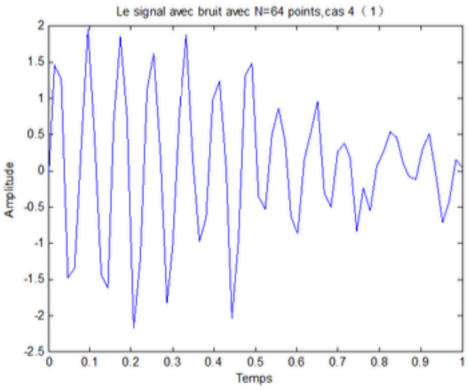


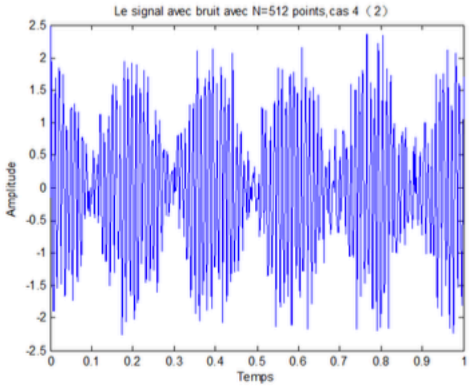
Repeat the last simulation (RSB = 10dB, 3 = 0.21), with N = 512 points. Comparison in case 4 with different points.

Case 4 (1): RSB = 10dB, 3 = 0.21, N = 64 points

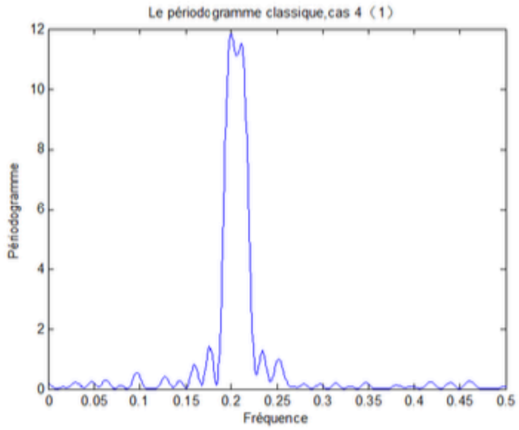
Case 4 (2): RSB = 10dB, 3 = 0.21, N = 512 points

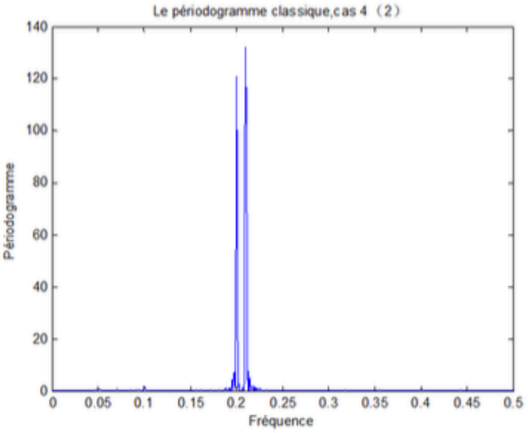
The signal figure is displayed below:



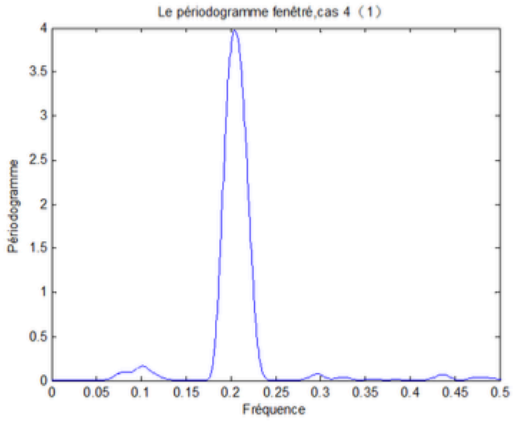


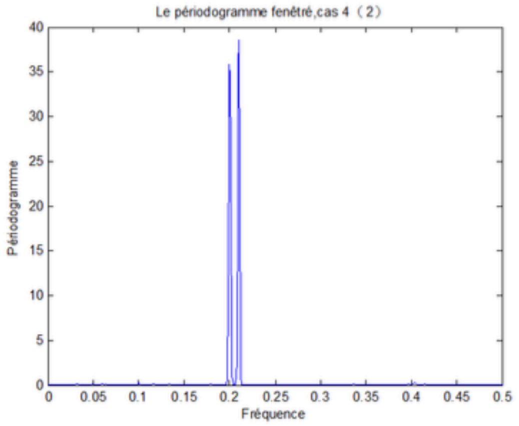
1) Classical Periodogram





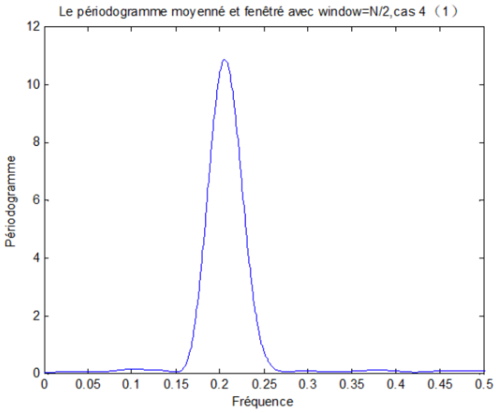
2) Calculation and display of the windowed periodogram:

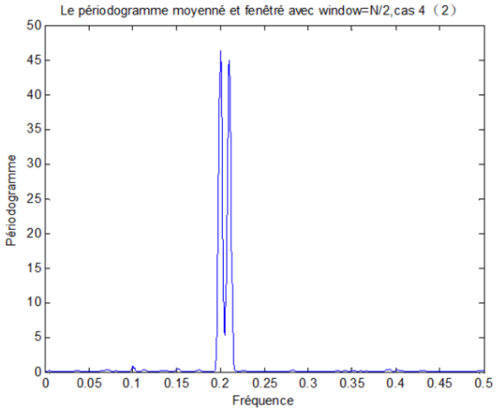




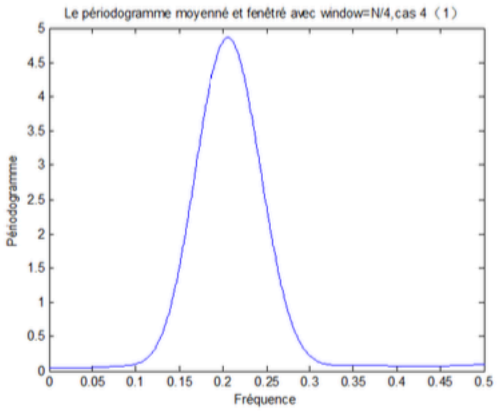
3) Calculation and display of the averaged periodogram and window:

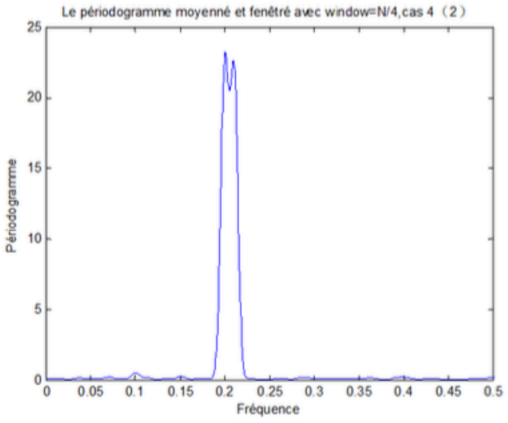
In the case: the size of each segment is N / 2.



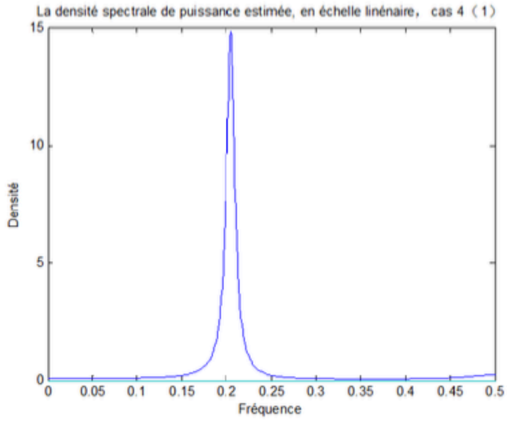


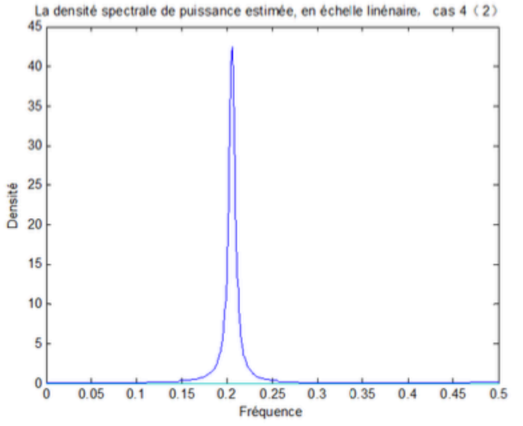
In this case : the size of each segment is N / 4.

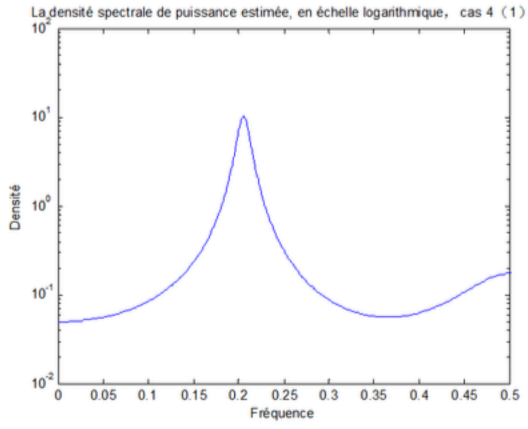


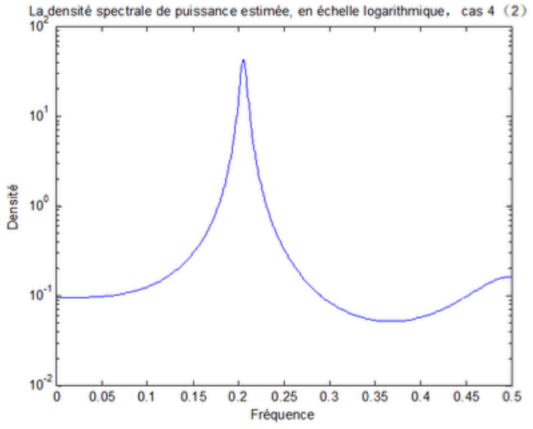


4) Analysis based on a self-regressive (AR) modeling:

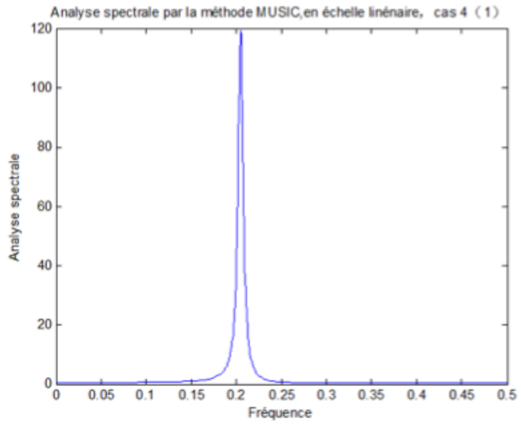


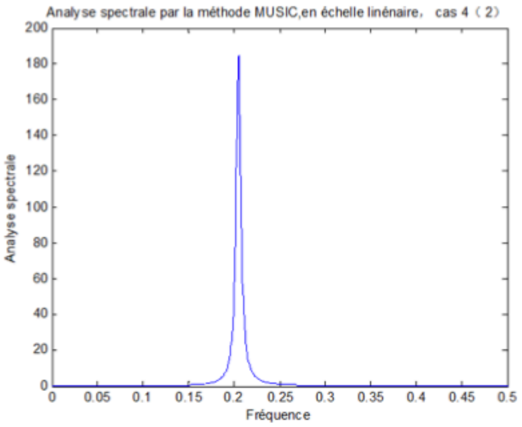


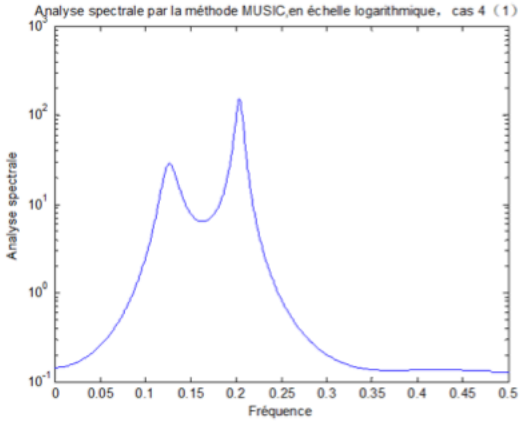


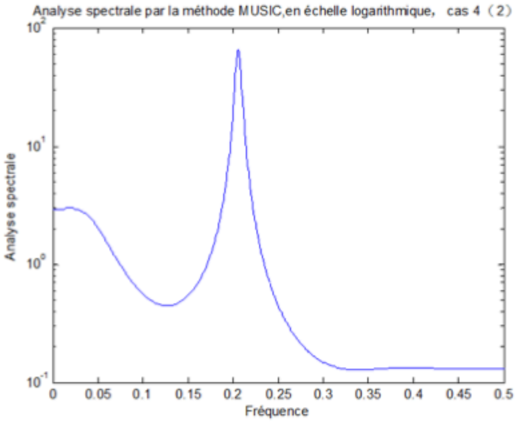


5) Spectral Analysis by MUSIC method







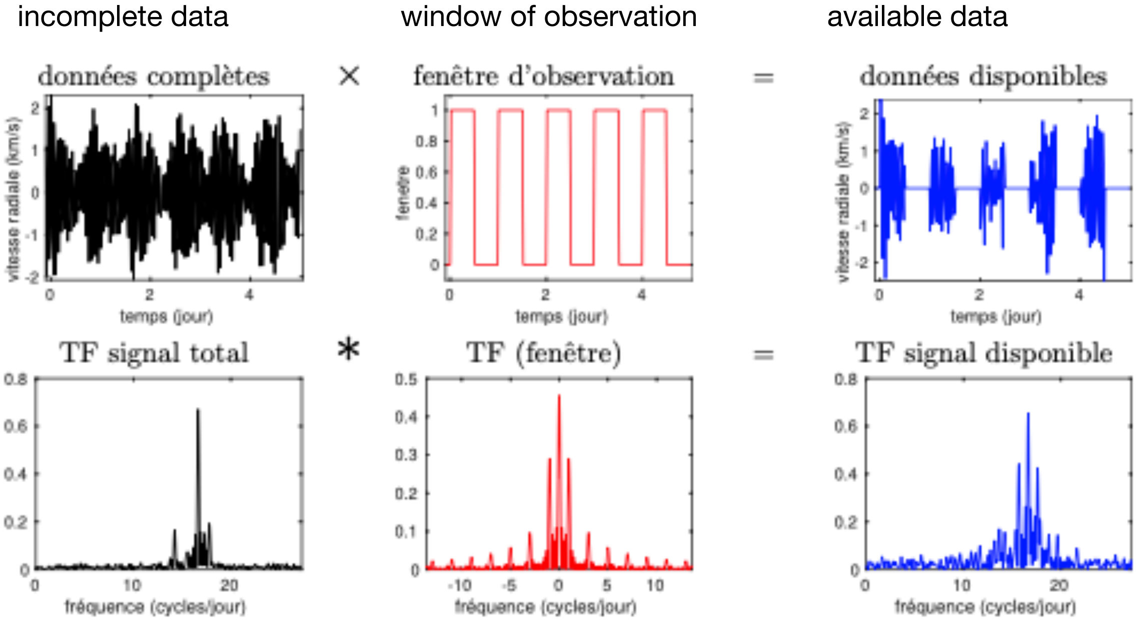


10. Summarise the advantages and limitations of each method in a table.

Method	Advantages	Limitations
Classical Periodogram	Not much, can reduce the oscillations to some extent	Capacity of PSD distinguishes different frequency composantes decreases
Windowed Periodogram	Almost same, can reduce the random oscillations, a bit better than Classical Periodogram	Important limitation because of using windowed periodogram is to reduce the resolution of periodogram
Average and Windowed Periodogram	With the help of Average and Windowed Periodogram, we can reduce the random oscillations plus assure a spectral resolution which is sufficient	For N = 64, we see no clear improvement with the windowed periodogram. We can see a slight difference when N = 512
Auto Regressive	AR could guarantee us a spectrum in high resolution	Performs poor for short data records and possible frequency bias for estimates of sinusoids in noise
MUSIC	Used for multi signal classification	When P is not large, MUSIC method a bit badly

2 Detection of exoplanetes by time-series spectral analysis

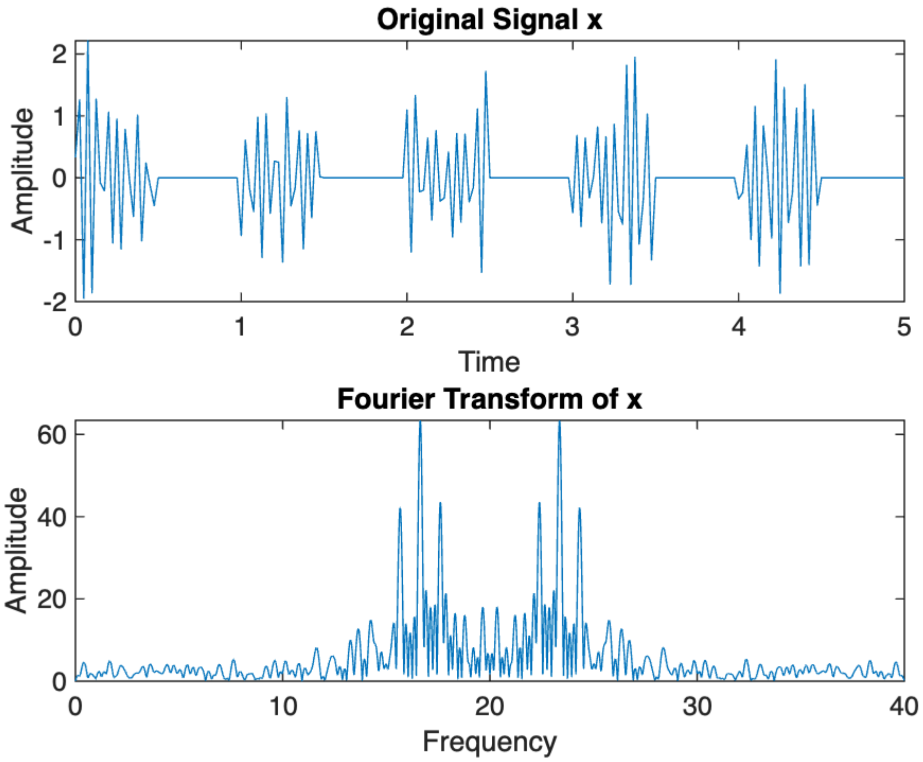
In astrophysics, the detection of extrasolar planets can be carried out by searching for periodic components in the temporal variations of an observable quantity, such as luminosity or the apparent speed of the star. A specific feature of observations from the earth is their irregular nature: since the object under study is only visible at night, the observations contain holes. The absence of data means that the null signal at the missing moments is considered to be zero, i.e. the complete (unknown) signal is multiplied by the observation window coding by 1 (by 0) the presence (absence) of the measurement. Consequently, the Fourier transform of the data is written as the convolution of the theoretical spectrum by the Fourier transform of this observation window.

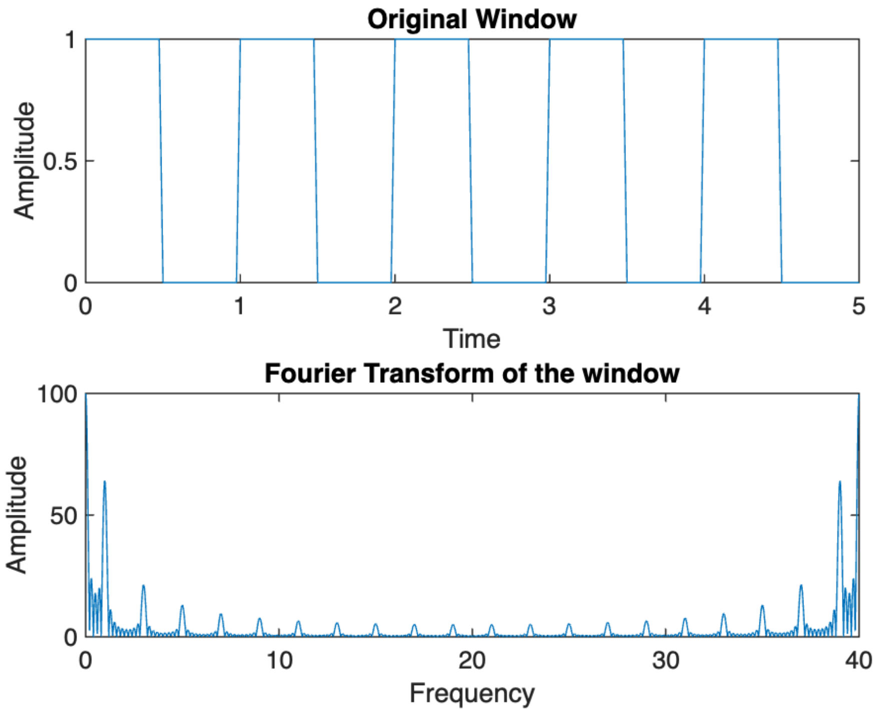


The astro.mat data file contains simulated data (t and x variables), typical of this kind of problem, shown in the figure above. The data represent five nights of observation of the apparent speed of a star. The sampling period is Te = 0.025 day (36 min).

TO DO 2

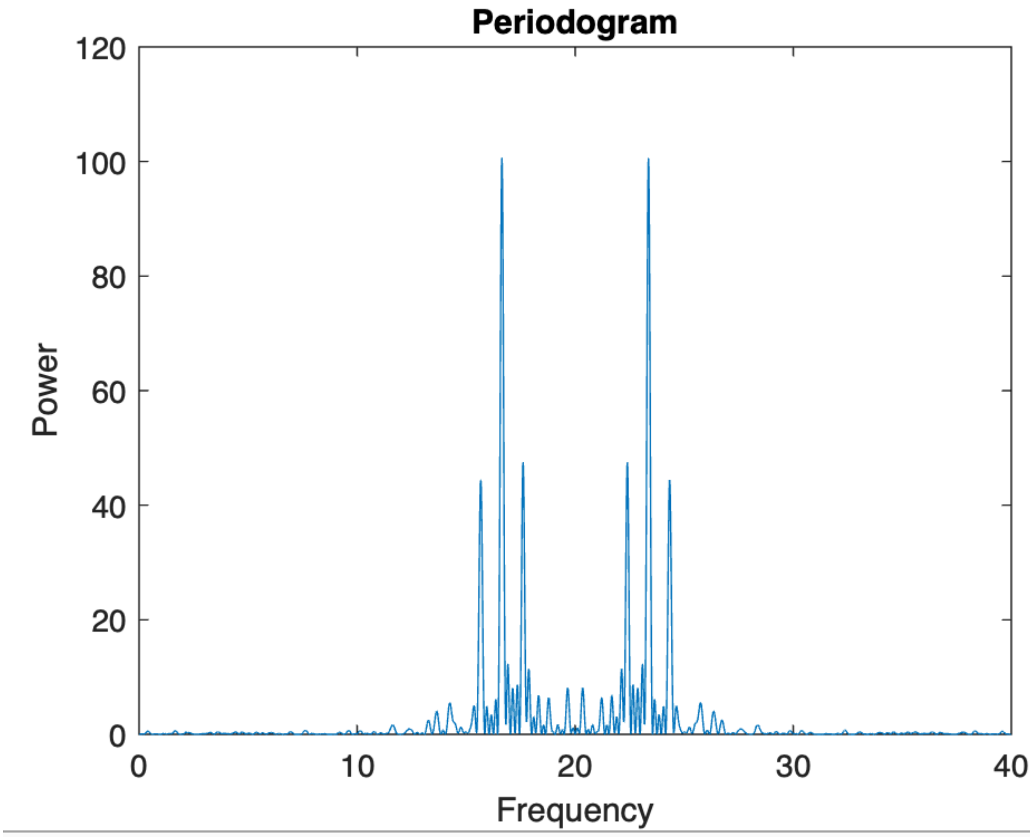
1. The observation window is contained in the variable fen. Calculate and represent (in module) its Fourier transform (it will be calculated by FFT on 4096 frequencies between 0 and Fe = 1/Te). Comment on its appearance.





2. Calculate and represent the periodogram of the signal. Comment

The real parameters of each sinusoıde for this example are contained in the variables f_th, amp_th and phi_th.



3. *To reduce the impact of parasitic peaks, a common method 4 is based on the following principle: the main frequency is estimated and then its contribution to the signal is removed. It is then iterated over the residue, and so on until nothing significant remains.*

Implement this method, or, at each iteration :

i) the main frequency is estimated at the maximum of the residu's periodogram;

ii) the estimation of the associated amplitude and phase is then carried out in smaller squares. The solution can indeed be calculated analytically, from the function provided:

[amp_est,phi_est] = estim_amp_phase(residu,t,f_est);

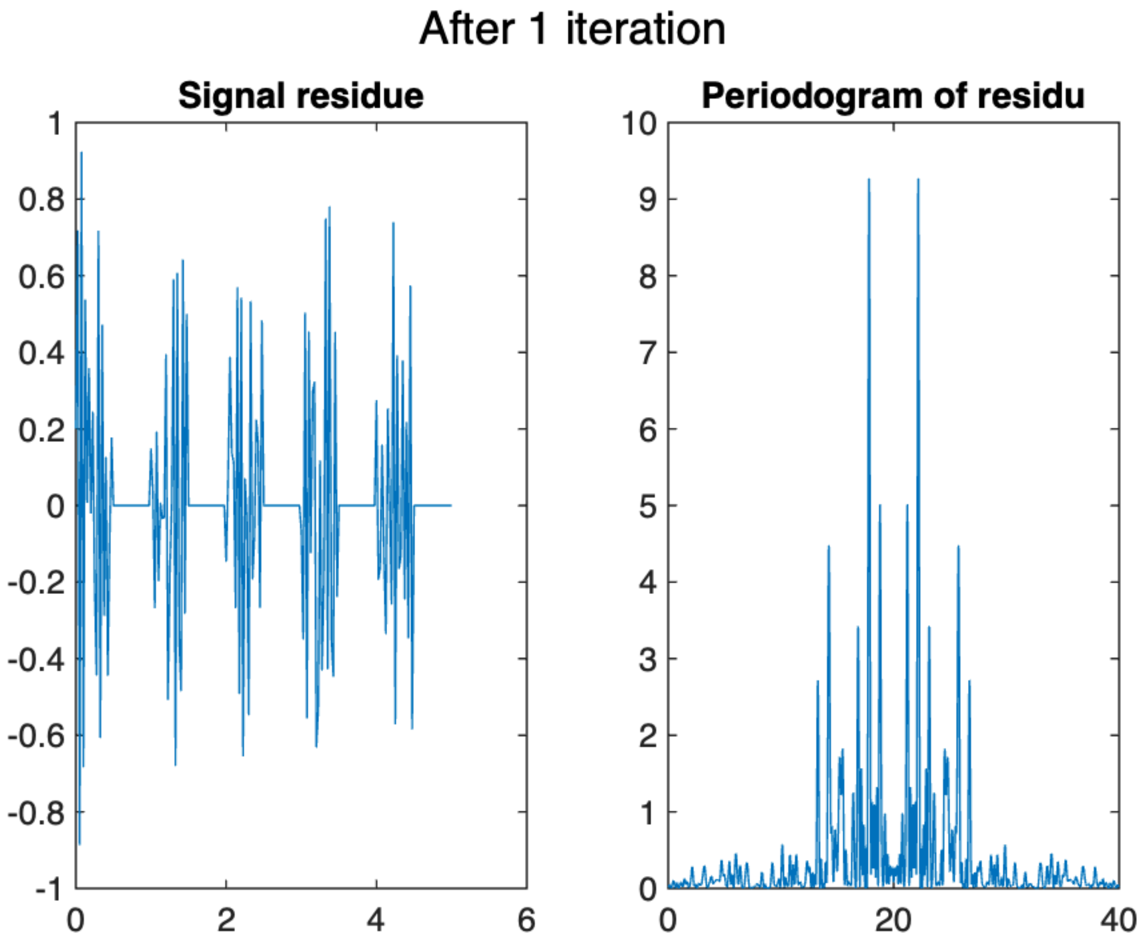
The contribution corresponding to sinusoıde is then amp is × sin(2πf is t + phi is), which is subtracted to form a new residue.

```
% Least Squares Estimation of amplitude and phase parameters of a sinusoid of given frequency

function [amp_est,phi_est] = estim_amp_phase(signal,t,f_est)


ind_NZ = find(signal~=0);
t_NZ = t(ind_NZ); t_NZ = t_NZ(:);
signal = signal(ind_NZ); signal = signal(:);


R = [cos(2*pi*f_est*t_NZ), sin(2*pi*f_est*t_NZ)];
amp_cossin = (R'*R)\(R'*signal);


if amp_cossin(2)>=0
    phi_est = atan(amp_cossin(1)/amp_cossin(2));
else
    phi_est = atan(amp_cossin(1)/amp_cossin(2)) + pi;
end

amp_est = sqrt(amp_cossin(1)^2+amp_cossin(2)^2);
```

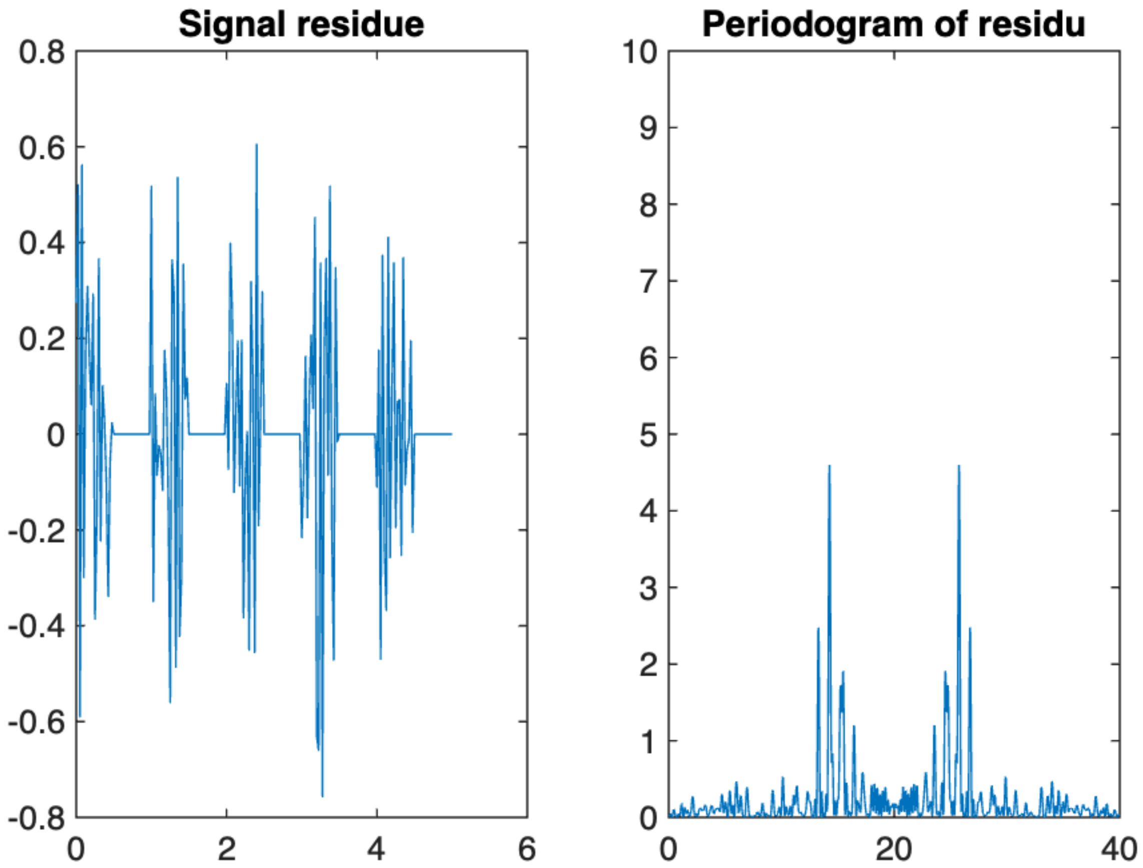
iii) After analysing the periodogram of the residence, it is decided whether or not to continue the iterations.

We take the maximum of periodogram <1 as the stop condition to decide whether or not to continue the iterations. After 6 times iteration, we get a residue periodogram as below, where there is no main peak left. In this case, it is believed that sinusoidal signals with periodogram amplitudes less than 0.1 no longer remain significant.

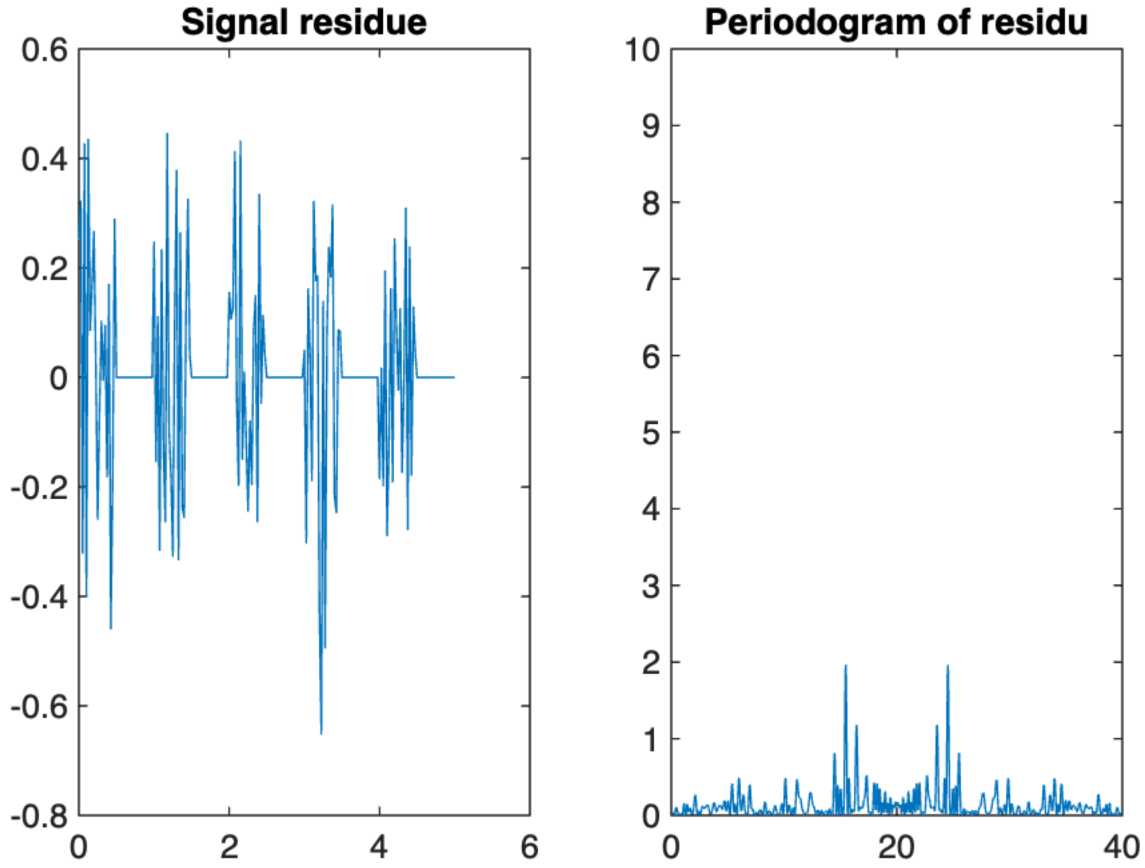
4. Run the algorithm displaying the residue, its Fourier transform and the components of˝etectees at each iteration. Comment and conclude on the d´etection of the oscillations.

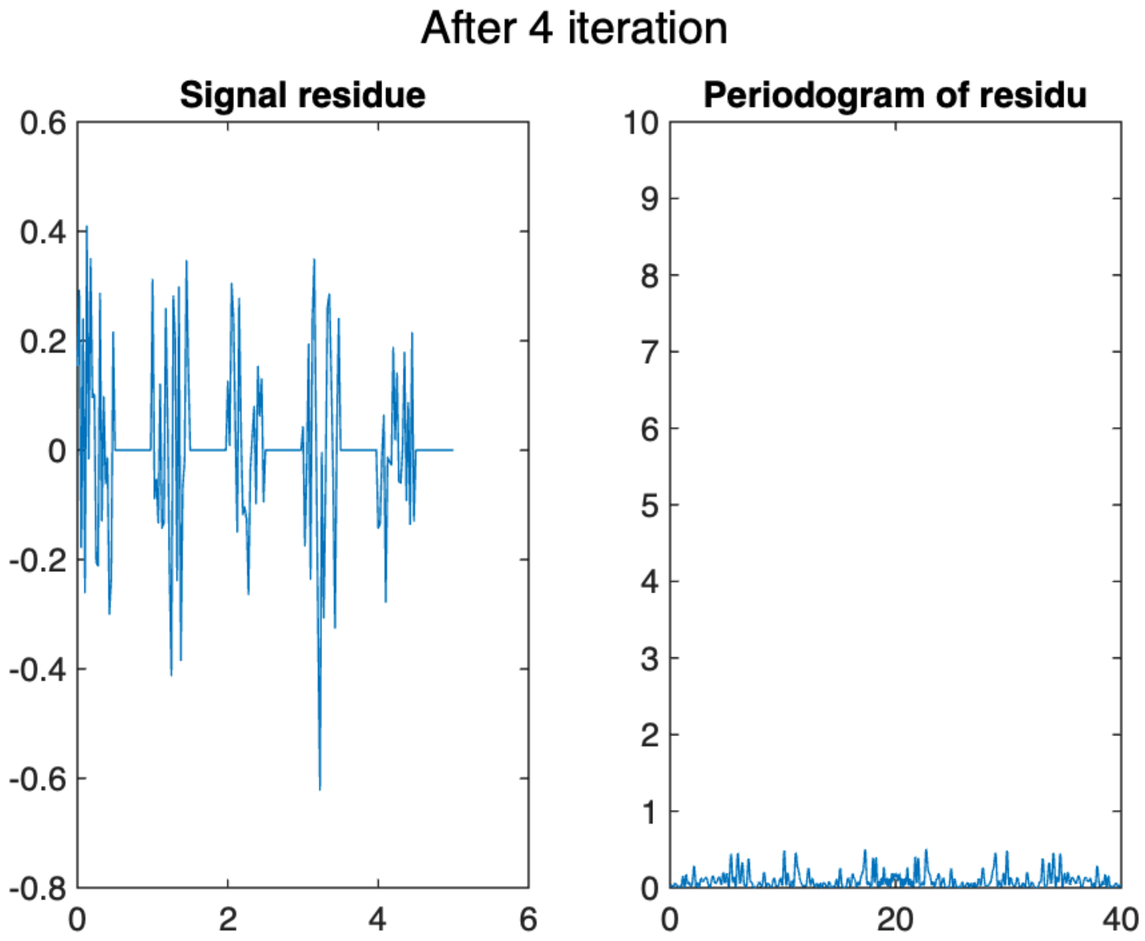


After 2 iteration



After 3 iteration




After 4 iteration
Signal residue
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
0
2
4
6
Periodogram of residu
10
9
8
7
6
5
4
3
2
1
0
0
20
40

Annexed Code :

Question 1

```
%% Initialisation
clc;
clear;
N = 64;
% N = 512;
% Define the variables as given in the question
a=[0.1 1 1];
lambda=[0.1 0.2 0.22];
phi=[0 0 0];
rsb =inf;



%% Generate Signal
x = genere_signal(a,lambda,phi,N,rsb);
% Change 10 to 20 if needed
x = ajoute_bruit(x,10);
n = 0:N-1;
figure(1)
plot(n,x)
title('Signal');
xlabel('Time');ylabel('Amplitude');


%% Classical Periodogram
Nf = 2048;
Axe_freq  = linspace(0,1-1/Nf,Nf);
% Power of DFT
Px = 1/N*abs(fft(x,Nf)).^2;
figure(2)
plot(Axe_freq,Px);
set(gca,'xlim',[0,0.5])
title('Periodogram');
xlabel('Reduced Frequency');ylabel('Power');

%% Windowed Periodogram
% given in the question
h = hamming(N);
```

```matlab
% element wise multiplication
xnhn = x.*h;
Pxn = 1/N*abs(fft(xnhn,Nf)).^2;
figure(3)
plot(Axe_freq,Pxn)
set(gca,'xlim',[0,0.5])
title('Periodogram Window');
xlabel('Reduced Frequency');ylabel('Power');


%% Welch Periodogram
Fs = 1;
dsp_welch = pwelch(x,N/2,N/8,Nf,Fs,'twosided');
figure(4)
subplot(121)
plot(Axe_freq,dsp_welch)
set(gca,'xlim',[0,0.5])
title('Segments of length N/2');
xlabel('Reduced Frequency');ylabel('Power');
dsp_welch = pwelch(x,N/4,N/16,Nf,Fs,'twosided');
subplot(122)
plot(Axe_freq,dsp_welch)
set(gca,'xlim',[0,0.5])
title('Segments of length N/4');
xlabel('Frequence reduite');ylabel('Power');
sgtitle('Welch Periodogram');


%% AR method
P = 3; % P is chosen
% Method 1: In total
p = linspace(0,P,P+1);
lambda = linspace(0,1,Nf);
[a_est,sigma2_est] = arcov(x,P);
sum = zeros(Nf);S = sum;
for k = 1 : Nf
    for j = 1 : P+1
        sum(k) = sum(k) +
a_est(j)*exp(-2*pi*p(j)*lambda(k)*1i);
    end
    S(k) = sigma2_est/abs(sum(k))^2;
end


% Method 2: Using FFT directly
```

```
S = sigma2_est./(abs(fft(a_est,Nf))).^2;


figure(5)
subplot(121)
plot(Axe_freq,S)
set(gca,'xlim',[0,0.5])
title('Linear Scale');
xlabel('Reduced Frequency');ylabel('Power Spectral Density
S');
subplot(122)
plot(Axe_freq,S)
set(gca,'xlim',[0,0.5])
set(gca,'yscale','log')
title('Log Scale');
xlabel('Reduced Frequency');ylabel('Power Spectral Density
S');
sgtitle(['AR Modelisation(P=',num2str(P),')']);



%% MUSIC Method
P = 6;
% Fs = 1 according to Q6
S_MUSIC = pmusic(x,P,Axe_freq,Fs); figure(6)
subplot(121)
plot(Axe_freq,S_MUSIC)
set(gca,'xlim',[0,0.5])
title('Linear Scale');
xlabel('Reduced Frequency');ylabel('S\_MUSIC');
subplot(122)
plot(Axe_freq,S_MUSIC)
set(gca,'xlim',[0,0.5])
set(gca,'yscale','log')
title('Log Scale');
xlabel('Reduced Frequency');ylabel('S\_MUSIC');


sgtitle(['MUSIC Method(P=',num2str(P),')']);




%% Estimation of a parcimoniex signal
mu_regul = 0.6;
```

```
umin = minl1_Fourier(x,Axe_freq,mu_regul);
```

Question 2 :

```
% Load the Data
clc;
clear all;

load donnees_astro.mat;
Nf = 4096;
Axe_freq = linspace(0,Fe-Fe/Nf,Nf);
TF_x = abs(fft(x,Nf));
TF_fen = abs(fft(fen,Nf));


%% 2.1 a Signal and its Fourier Transform
figure(1)
subplot(211)
plot(t,x)
title('Original Signal x');
xlabel('Time');ylabel('Amplitude');
subplot(212)
plot(Axe_freq,TF_x);
title('Fourier Transform of x');
xlabel('Frequency');ylabel('Amplitude');


%% 2.1 b Window and its Fourier Transform
figure(2)
subplot(211)
plot(t,fen)
title('Original Window');
xlabel('Time');ylabel('Amplitude');


subplot(212)
plot(Axe_freq,TF_fen);
title('Fourier Transform of the window');
xlabel('Frequency');ylabel('Amplitude');
```

```
%% 2.2 Periodogram of Signal
% DFT Power
Px = 1/Fe*abs(fft(x,Nf)).^2;
figure(3)
plot(Axe_freq,Px);
title('Periodogram');
xlabel('Frequency');ylabel('Power');



%% 2.3 Reduce the impacts of peaks
residu = x;
nb_iter = 1;
periodresidu = Px;
while (max(periodresidu) >= 1)
    [~,i_est] = max(periodresidu);
    [amp_est,phi_est] =
estim_amp_phase(residu,t,Axe_freq(i_est));
    contrib = amp_est * sin(2*pi*Axe_freq(i_est)*t +
phi_est);
    residu = residu - fen.*contrib;
    periodresidu = 1/Fe * abs(fft(residu,Nf)).^2;

    figure(nb_iter+3)
    subplot(121)
    plot(t,residu)
    title('Signal residue')
    subplot(122)
    plot(Axe_freq,periodresidu);
    ylim([0 10])
    title('Périodogram of résidu')
    sgtitle(['After ',num2str(nb_iter),' iteration']);
    nb_iter = nb_iter + 1;
end
```