UNIVERSITY OF MORATUWA
Faculty of Engineering

MATLAB Assignment 3
Continuous and Discrete Wavelet Transforms

November 3, 2023

BM4151 Biosignal processing

Ranathunga R.A.C.D.
190501V
Department of Electronic and Telecommunication Engineering

Contents

1 Continuous Wavelet Transform **2**

1.1 Introduction 2

1.2 Wavelet properties 2

1.3 Continuous Wavelet Decomposition 8

2 Discrete Wavelet Transform **10**

2.1 Introduction 10

2.2 Applying DWT with the Wavelet Toolbox in MATLAB 10

2.3 Signal Denoising with DWT 12

2.4 Signal Compression with DWT 14

1 Continuous Wavelet Transform

1.1 Introduction

Nothing to perform under this subsection.

1.2 Wavelet properties

i) $g(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2}$

since $\mu = 0$ and $\sigma = 1$,

$$g(t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-0}{1}\right)^2} = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}t^2}$$

$$\frac{d}{dt}[g(t)] = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}t^2} \cdot \left(\frac{-1}{2}\right) 2t = \frac{-t e^{-\frac{1}{2}t^2}}{\sqrt{2\pi}}$$

$$\therefore \frac{d^2}{dt^2}[g(t)] = \frac{d}{dt}\left[\frac{-t e^{-\frac{1}{2}t^2}}{\sqrt{2\pi}}\right]$$

$$= \frac{-1}{\sqrt{2\pi}}\left[t e^{-\frac{1}{2}t^2}\left(\frac{-1}{2}\right)2t + e^{-\frac{1}{2}t^2}\right]$$

$$= \frac{-e^{-\frac{1}{2}t^2}}{\sqrt{2\pi}}\left[-t^2 + 1\right]$$

since $m(t) = -\frac{d^2}{dt^2} g(t)$, $m(t) = \frac{-e^{-\frac{1}{2}t^2}(t^2-1)}{\sqrt{2\pi}}$

ii) $E = \int_{-\infty}^{+\infty} m^2(t)\, dt = \int_{-\infty}^{+\infty}\left[\frac{-e^{-\frac{1}{2}t^2}(t^2-1)}{\sqrt{2\pi}}\right]^2 dt = 0.21157$

$\therefore$ if $m(t)_{norm} \rightarrow K \cdot m(t)$, $E_{norm} \rightarrow K^2 \cdot E$

when we want $E_{norm} = 1 \rightarrow$, $K^2 = \frac{1}{E} \Rightarrow K = \frac{1}{\sqrt{E}}$

$K = \frac{1}{\sqrt{0.21157}} = 2.174$

$\therefore$ Normalizing factor $= \underline{2.174}$

iii) ψ is the wavelet function. Since $\int_{-\infty}^{+\infty} \psi(t)\, dt = 0$ and $\int_{-\infty}^{+\infty} \psi^2(t)\, dt = 1$

$\psi(t) = K\, m(t)$ is a good suggestion.

$$= 2.174\left[\frac{-e^{-\frac{1}{2}t^2}(t^2-1)}{\sqrt{2\pi}}\right]$$

Hence, $\frac{2.174}{\sqrt{2\pi}} = \frac{1}{\sqrt{S}}$

$S = 1.33$

$\therefore \psi(t) = \frac{1}{\sqrt{S}}\left[-e^{-\frac{1}{2}t^2}(t^2-1)\right]$ where $S = 1.33$

$\therefore \psi(t) = \frac{1}{\sqrt{1.33}}\left[-e^{-\frac{1}{2}t^2}(t^2-1)\right] = -(0.8671)\, e^{-\frac{1}{2}t^2}(t^2-1)$

Figure 1: Answers for the i,ii and iii parts of the 1.2 subsection

Mother wavelet for "Mexican hat" wavelet :

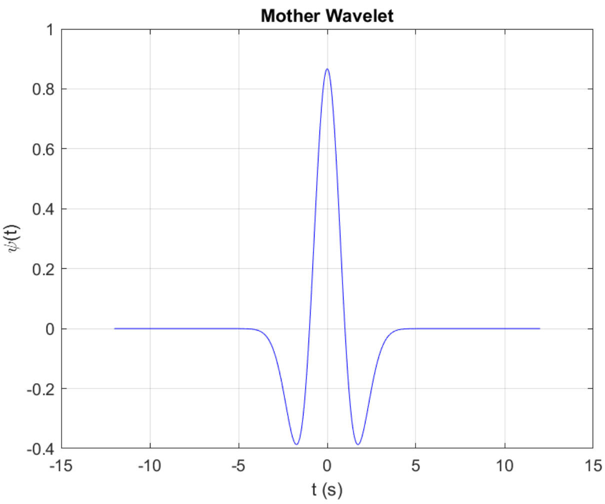
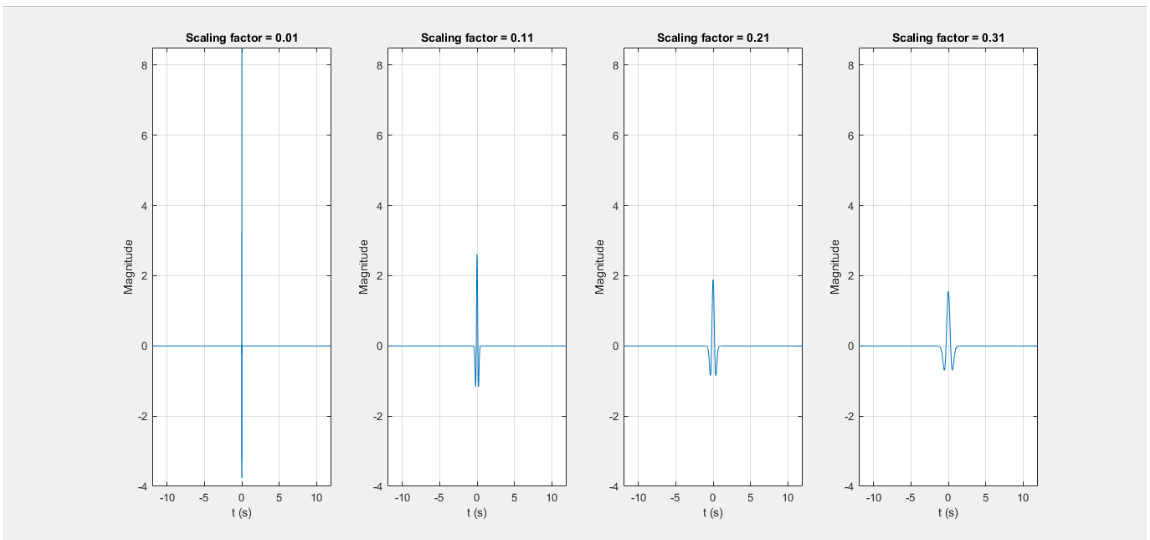
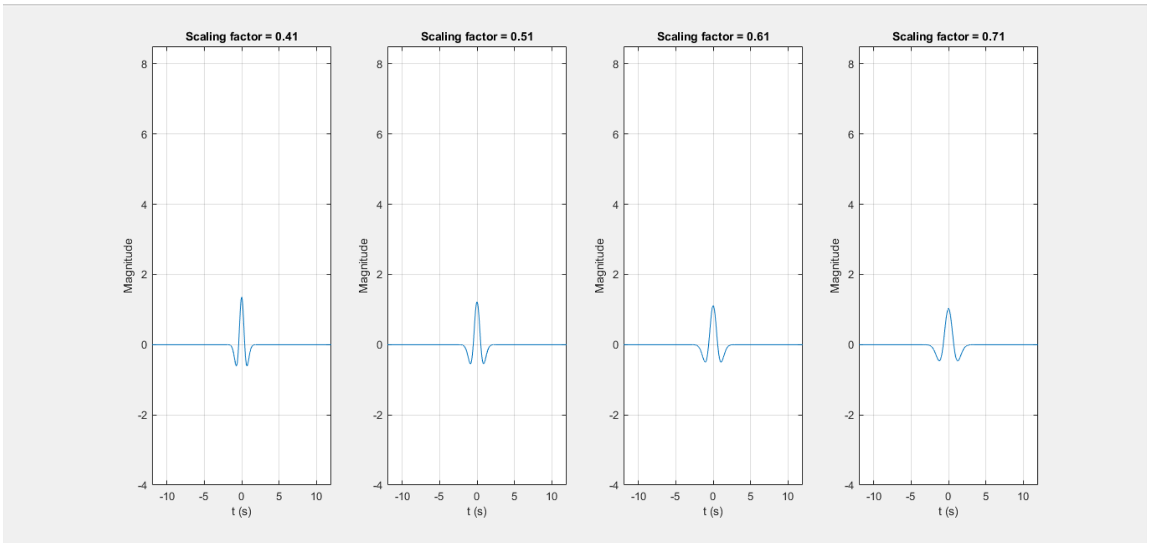
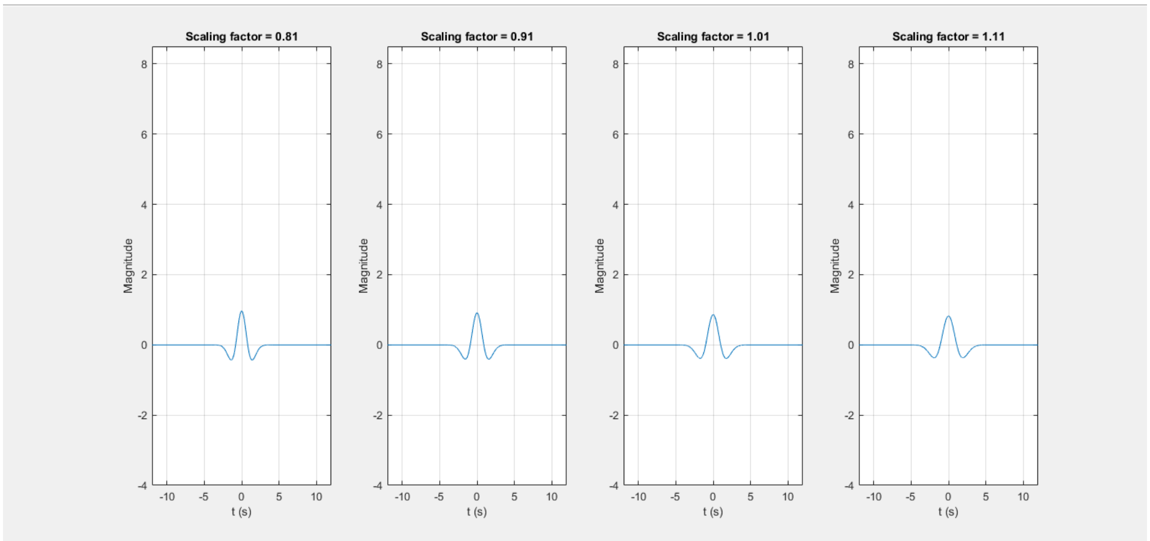


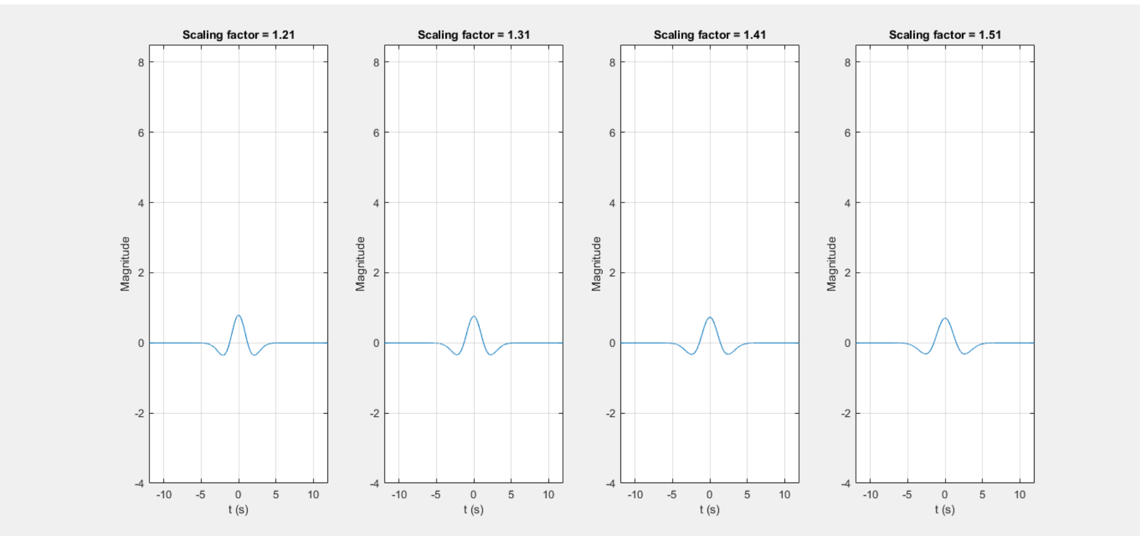
Figure 2: Mother wavelet for "Mexican hat" wavelet

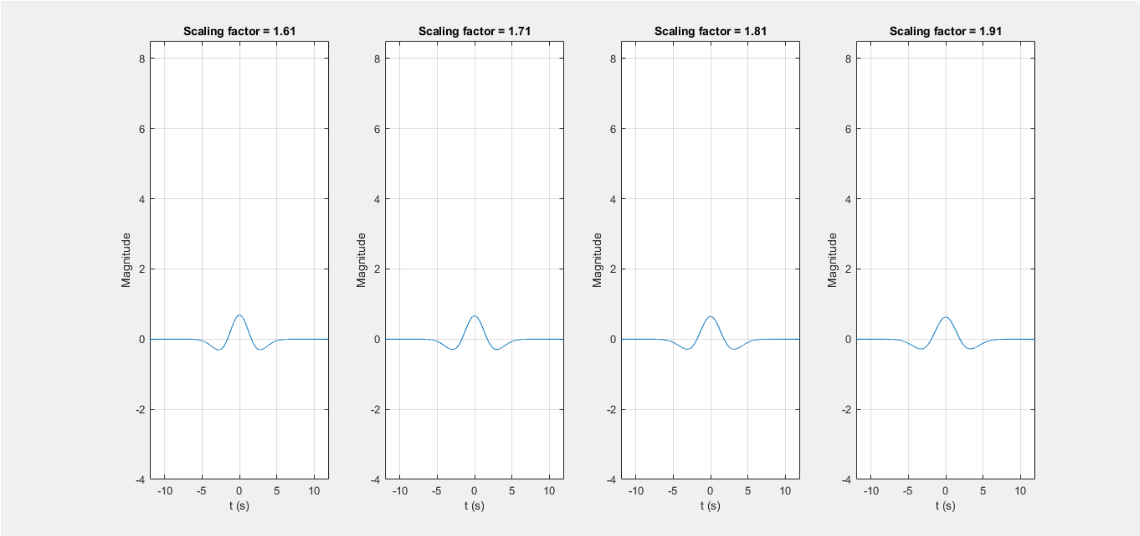
Now let's analyse the time-domain waveforms for the Mexican hat daughter wavelets for scaling factors of 0.01:0.1:2.




Scaling factor = 0.41
Scaling factor = 0.51
Scaling factor = 0.61
Scaling factor = 0.71
Magnitude
t (s)


Scaling factor = 0.81
Scaling factor = 0.91
Scaling factor = 1.01
Scaling factor = 1.11
Magnitude
t (s)


Scaling factor = 1.21
Scaling factor = 1.31
Scaling factor = 1.41
Scaling factor = 1.51
Magnitude
t (s)



The results verifies the facts that, when increasing the scaling factor,

- Window length in the time domain increase.
- The highest amplitude (peak) in the time domain decrease.

Here the 34th line calculates mean and the energy of each of the daughter wavelets and print it.

```
21  k = 1;
22  for i = 1:5
23      figure;
24      for j = 1:4
25          subplot(1,4,j);
26          s_wavelt = (1/sqrt(s(k)))* (-0.8671 .* exp(-0.5*(t/s(k)).^2).*((t/s(k)).^2-1));
27          plot(t, s_wavelt);
28          ylim([-4,8.5]);
29          xlabel("t (s)");
30          ylabel('Magnitude')
31          title_string = sprintf("Scaling factor = %.2f",s(k));
32          title(title_string);
33          grid on;
34          fprintf("When scaling factor is %.2f, mean = %.2f , energy = %.2f\n",s(k),mean(s_wavelt),round(trapz(t, wavelt.^2),2));
35          k=k+1;
36      end
37  end
```

So below is the result. This results verify the wavelet properties of zero mean, unity energy. Compact support property can be seen from the spectras.

```
Command Window
When scaling factor is 0.01, mean = -0.00 , energy = 1.00
When scaling factor is 0.11, mean = 0.00 , energy = 1.00
When scaling factor is 0.21, mean = -0.00 , energy = 1.00
When scaling factor is 0.31, mean = 0.00 , energy = 1.00
When scaling factor is 0.41, mean = 0.00 , energy = 1.00
When scaling factor is 0.51, mean = 0.00 , energy = 1.00
When scaling factor is 0.61, mean = -0.00 , energy = 1.00
When scaling factor is 0.71, mean = -0.00 , energy = 1.00
When scaling factor is 0.81, mean = -0.00 , energy = 1.00
When scaling factor is 0.91, mean = -0.00 , energy = 1.00
When scaling factor is 1.01, mean = 0.00 , energy = 1.00
When scaling factor is 1.11, mean = -0.00 , energy = 1.00
When scaling factor is 1.21, mean = -0.00 , energy = 1.00
When scaling factor is 1.31, mean = -0.00 , energy = 1.00
When scaling factor is 1.41, mean = 0.00 , energy = 1.00
When scaling factor is 1.51, mean = 0.00 , energy = 1.00
When scaling factor is 1.61, mean = 0.00 , energy = 1.00
When scaling factor is 1.71, mean = 0.00 , energy = 1.00
When scaling factor is 1.81, mean = 0.00 , energy = 1.00
When scaling factor is 1.91, mean = 0.00 , energy = 1.00
fx >>
```

Let's analyse the spectra of each.

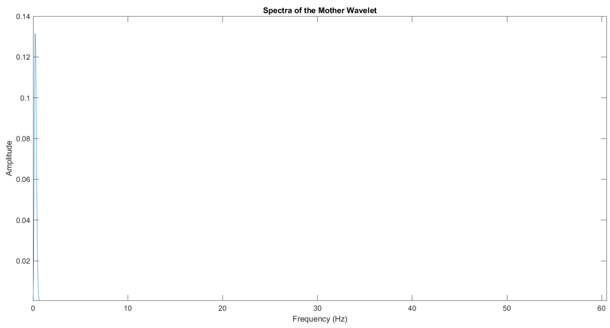
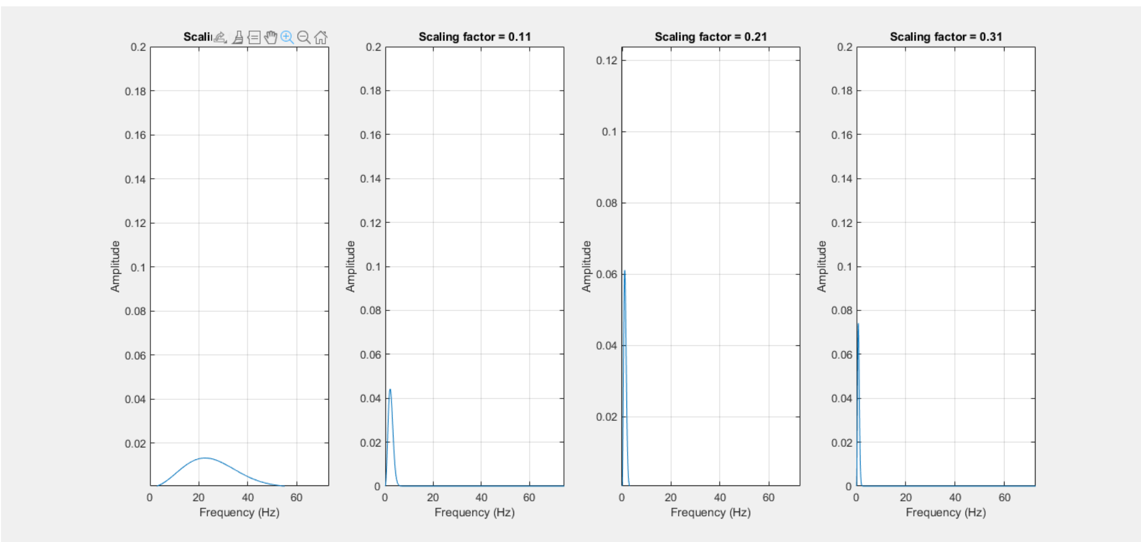
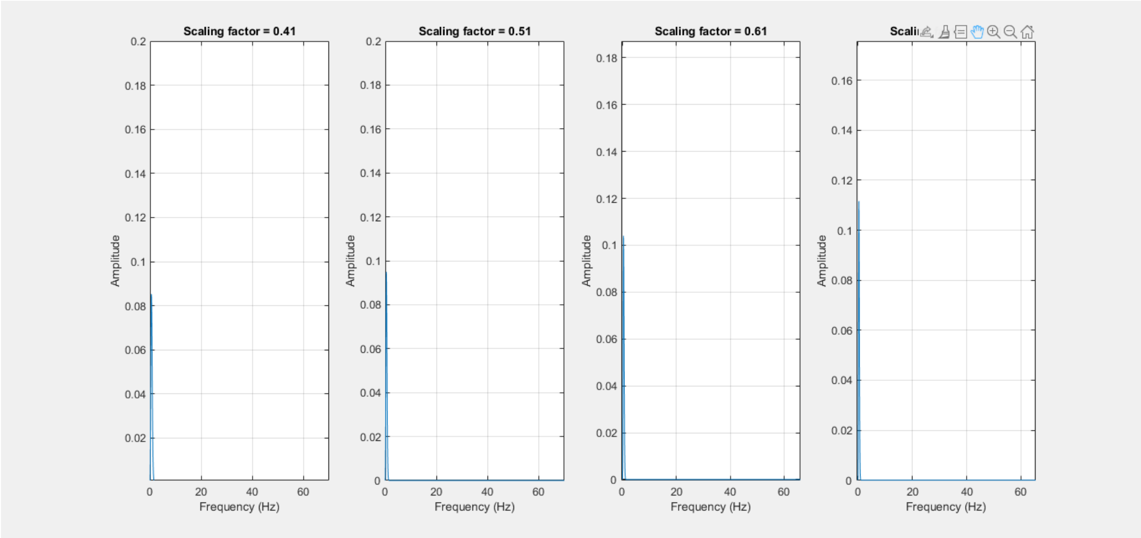


Figure 3: Spectra of the Mother Wavelet

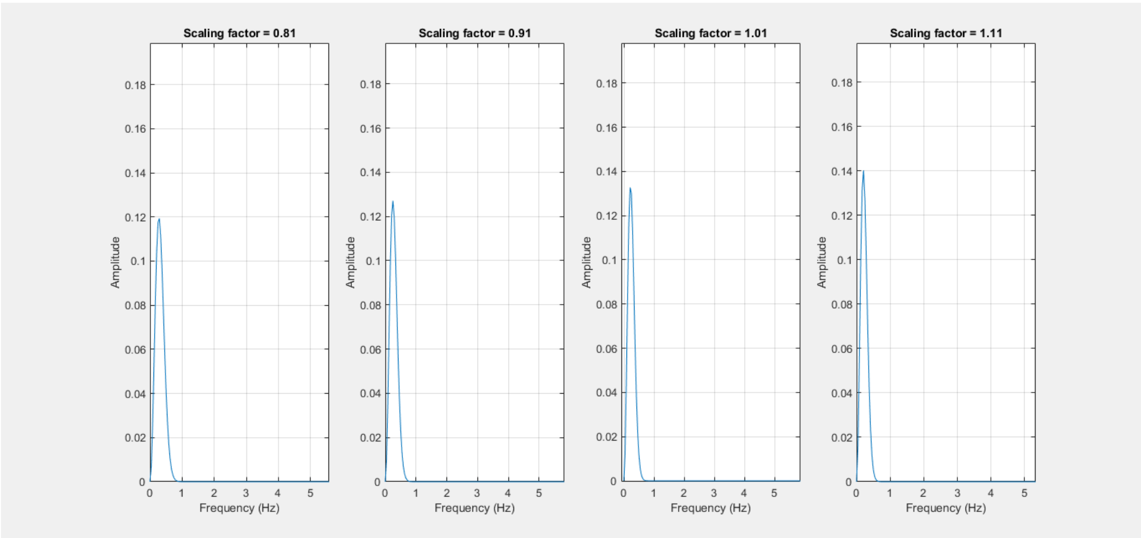
As you can see, almost all the energy is compact in the very low frequencies for the mother wavelet.

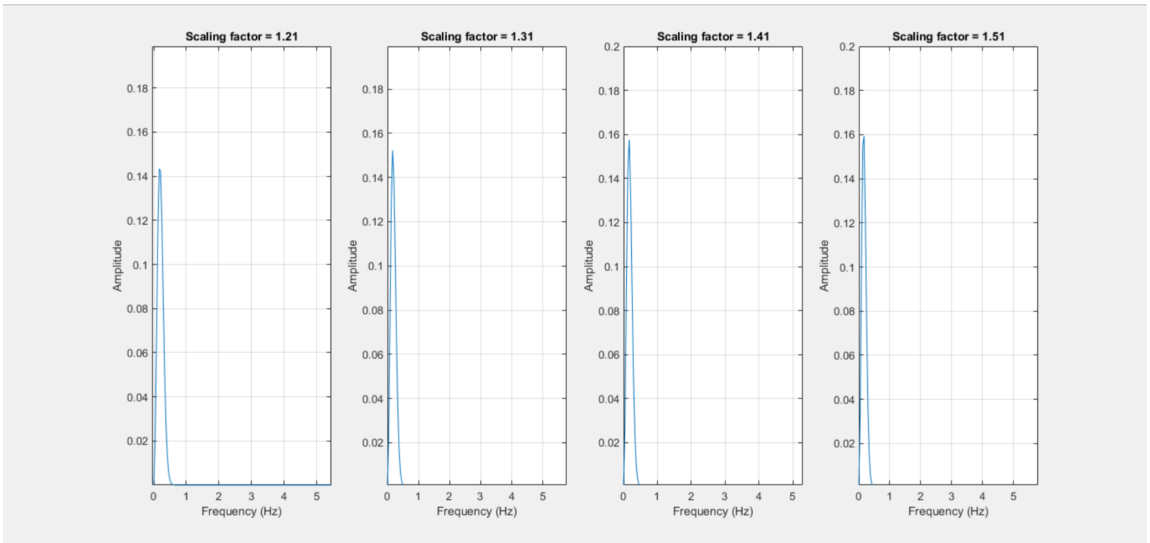
Now take a look at the affect of scaling on the spectras.

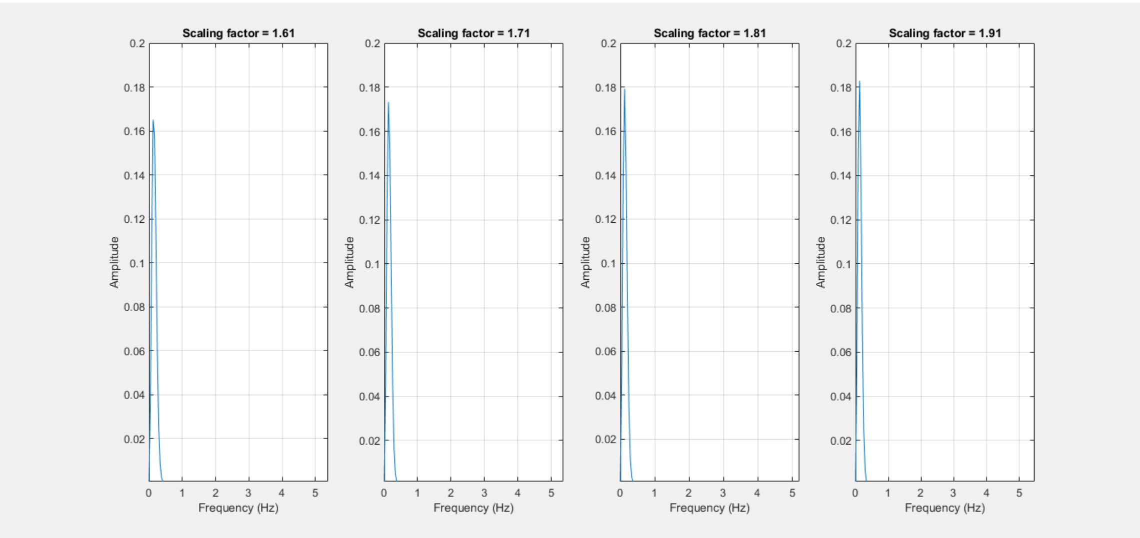




As you can see, when we increase th scaling factor, energy get's compacted to low frequencies. Therefore, for better visualization purposes, x axis is shortened in the following plots.







From this result, we can come to these conclusions.

- When we increase the scaling factor, almost all of the energy get compacts in the lower frequencies. Hence, becomes low pass filters.
- This result can verify the compact support property of the wavelets.
- Also the peak increases with the scaling factor.

1.3 Continuous Wavelet Decomposition

Following is a stem plot without markers for the created x[n] signal.

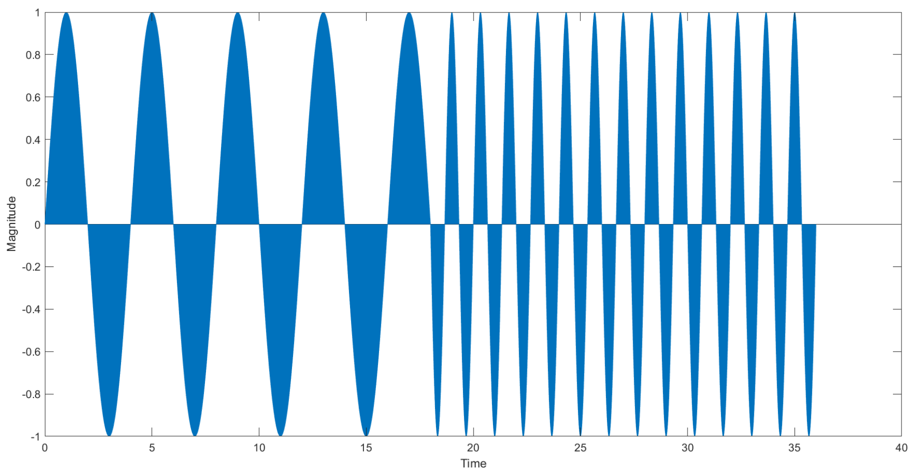


Figure 4: Stem plot of the signal x[n] without markers

Below is the spectrogram of the signal when the scaled Mexican hat wavelets are applied.

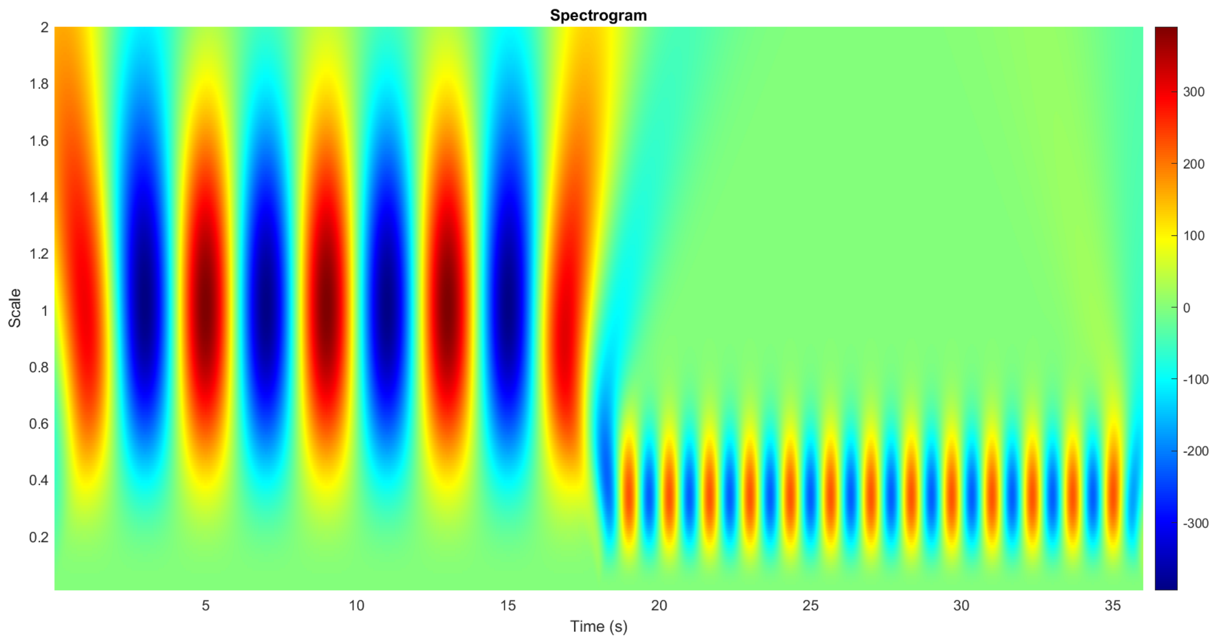


Figure 5: Spectrogram of x[n] when using Mexican hat wavelets

From this spectrogram we can derive this information.

- In the beginning of the signal, higher scales have high magnitudes and in the later part of the signal, lower scales have high magnitudes.
- When the scaling factor is high, it highlights low frequency components. Hence, we can say that the beginning of the signal is having a lower frequency.

- Similarly, when the scaling factor is low, it keeps information about high frequencies. Therefore the frequency in the signal is high in the later part of the time domain.
- We can verify this behaviour by looking at our signal.

2 Discrete Wavelet Transform

2.1 Introduction

Nothing to perform under this subsection.

2.2 Applying DWT with the Wavelet Toolbox in MATLAB

Here is the created x1[n], x2[n] signals with their respective noise added versions y1[n] and y2[n].

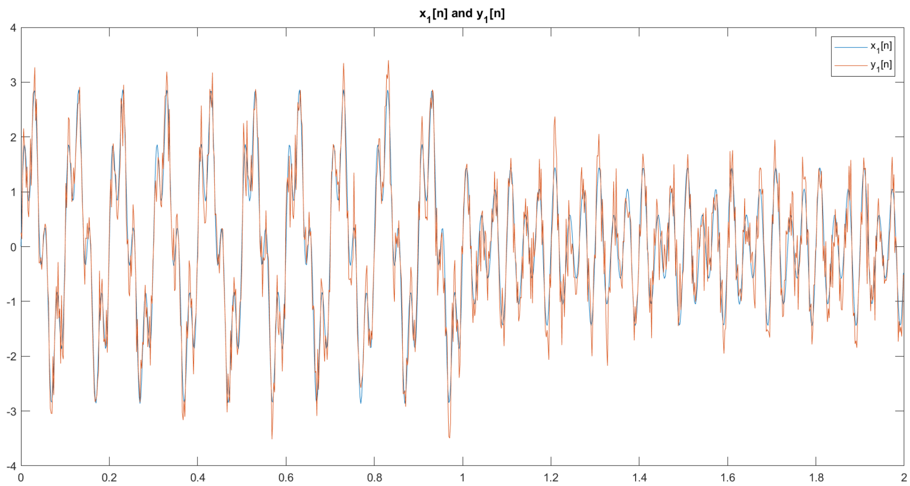


Figure 6: x1[n] and y1[n]

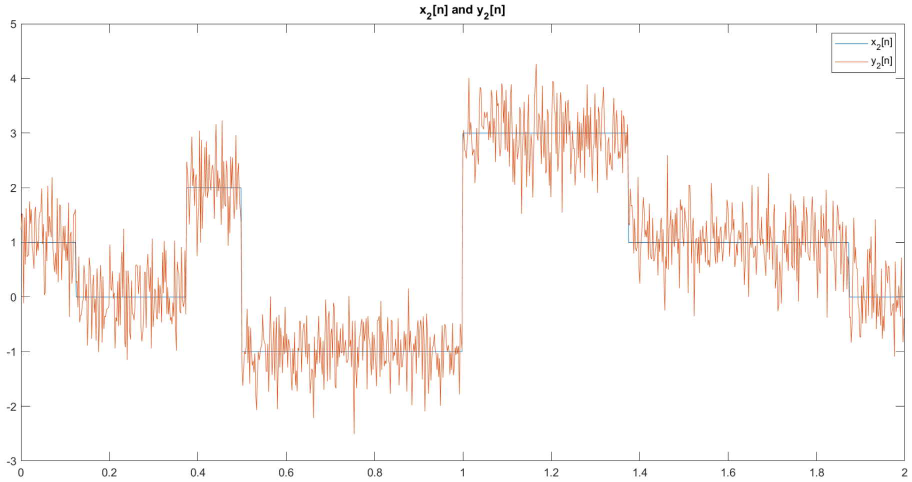


Figure 7: x2[n] and y2[n]

To denoise, the plan is to try Haar and Daubechies tap 9 wavlets and their scaling functions. Below is the morphology of those.

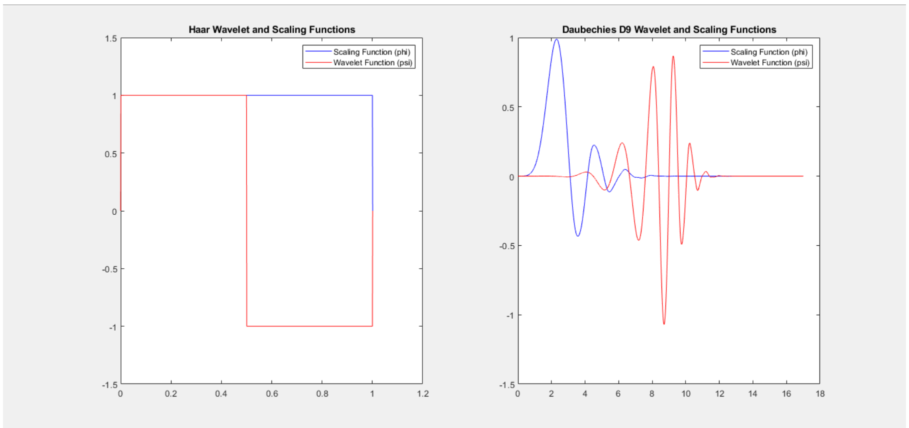


Figure 8: Morphology of wavlets and scaling functions for Haar and Daubechies tap 9

After decomposing the signal using the above mentioned wavelets and their scaling functions, here is the reconstructed signal with inverse DWT.

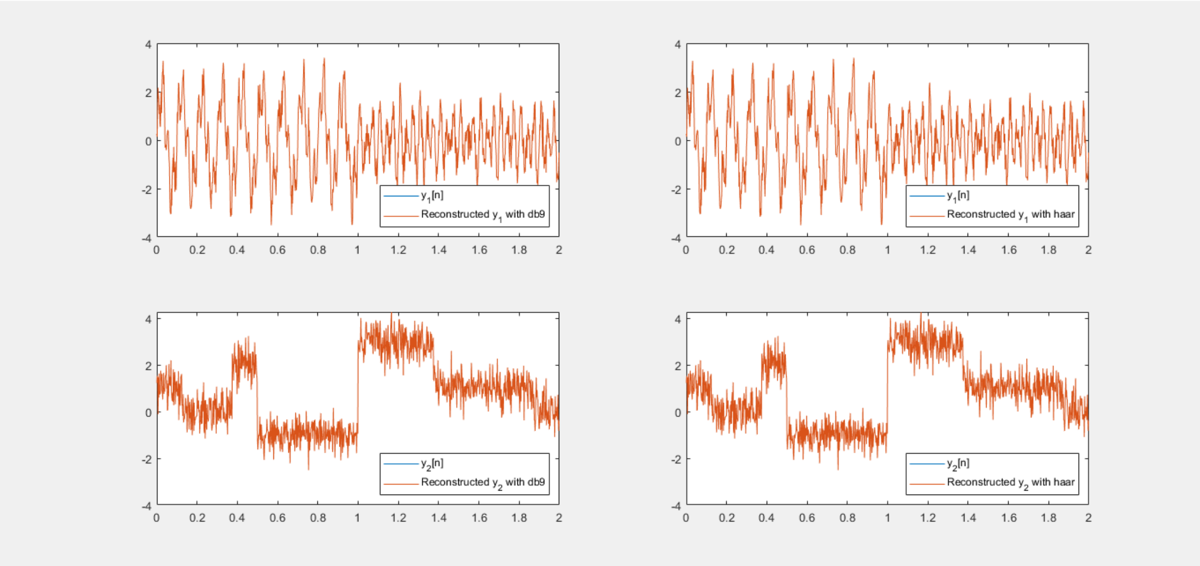


Figure 9: Reconstructed signals and their original versions

Here for the all 4 plots, original signal is not visible since the reconstructed signal has overwritten it. It means the reconstruction is very accurate. We can ensure it by looking for the energy differences.

```
%% Using inverse DWT to reconstruct the signal

reconstructed_db9_y1 = waverec(c_db9_y1, l_db9_y1, 'db9');
energy_db9_y1 = sum((y1 - reconstructed_db9_y1).^2);

reconstructed_haar_y1 = waverec(c_haar_y1, l_haar_y1, 'haar');
energy_haar_y1 = sum((y1 - reconstructed_haar_y1).^2);

reconstructed_db9_y2 = waverec(c_db9_y2, l_db9_y2, 'db9');
energy_db9_y2 = sum((y2 - reconstructed_db9_y2).^2);

reconstructed_haar_y2 = waverec(c_haar_y2, l_haar_y2, 'haar');
energy_haar_y2 = sum((y2 - reconstructed_haar_y2).^2);
```

Figure 10: Code snippet to calculate the energy differences in between reconstructed signal and the original

energy_db9_y1	1.5021e-17
energy_db9_y2	2.1275e-17
energy_haar_y1	7.1042e-28
energy_haar_y2	4.0053e-27

Figure 11: Calculated energy differences

It shows that the energy difference is almost zero, hence can ignore. It verifies that, $Y = \sum_i D^i + A$

2.3 Signal Denoising with DWT

For the both, signals y1 and y2, below are the magnitudes of wavelet coefficients in the descending order.

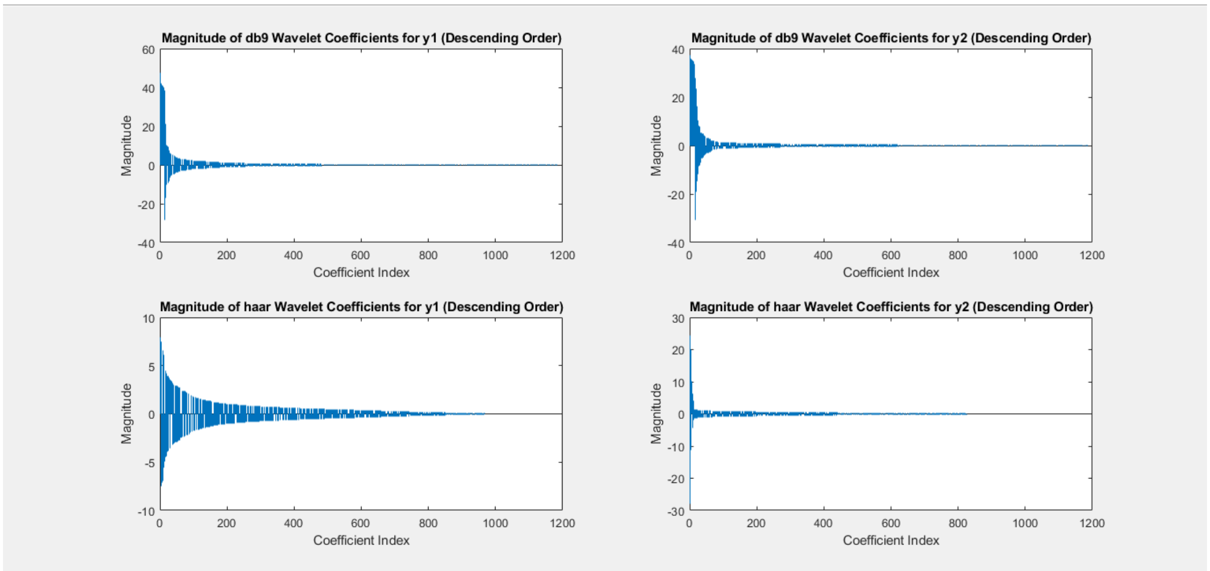


Figure 12: Magnitudes of wavelet coefficients in the descending order

Since the signal y1[n] is an addition of sinusoids + noise, db9 is a good wavelet to choose for y1[n] over the haar wavelet. The magnitudes are higher in the case of using haar wavelet for y1[n]. But for the signal y2[n], haar wavelet is good since it is having straight lines + noise. In that case also, db9 has higher magnitudes for the coefficients.

Below are the denoised signals by "hard thresholding" with thresholds determined by iterative inspections.

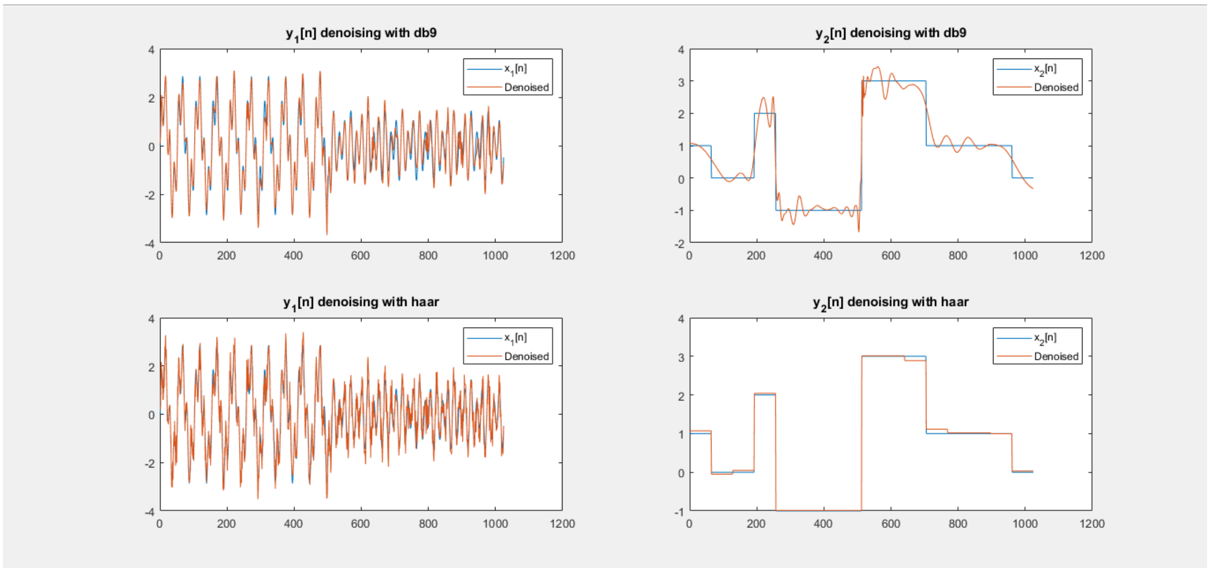
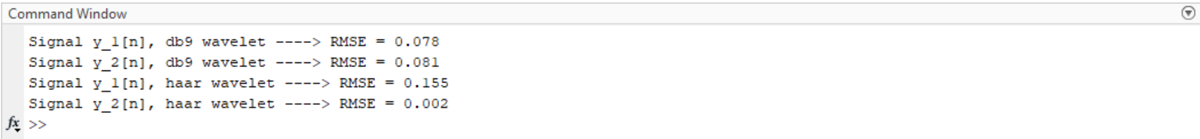


Figure 13: Denoised signals with the selected wavelets and their scaling functions

As we can see visually, db9 denoised version of signal y1[n] is better than the haar denoised version of signal y1[n]. Similarly, signal y2[n] is better denoised using the haar, but bad with db9.



```
Command Window
Signal y_1[n], db9 wavelet ----> RMSE = 0.078
Signal y_2[n], db9 wavelet ----> RMSE = 0.081
Signal y_1[n], haar wavelet ----> RMSE = 0.155
Signal y_2[n], haar wavelet ----> RMSE = 0.002
>>
```

Figure 14: Root mean square errors for each

So this confirms the above sayings.

2.4 Signal Compression with DWT

This is the given aVr lead ECG signal.

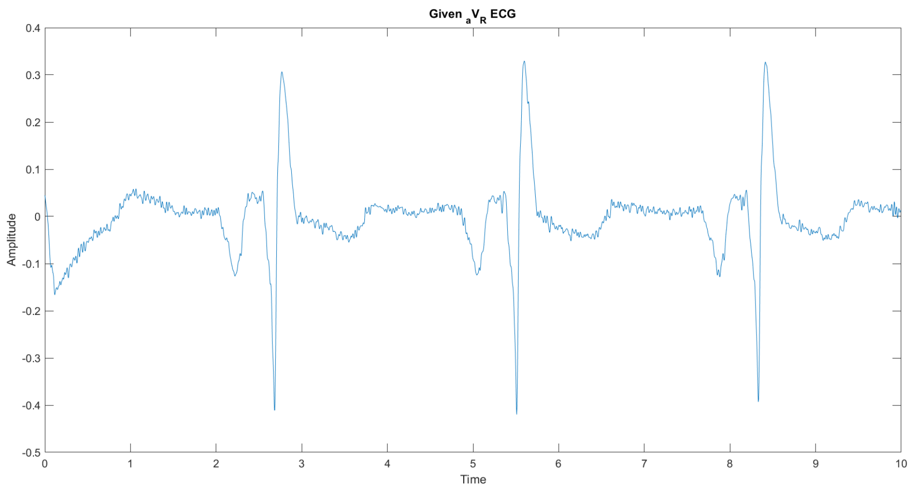


Figure 15: aVr lead ECG

After arranging the coefficients in the descending order, calculated the number of coefficients needed to construct the signal again retaining 99% of it's energy.

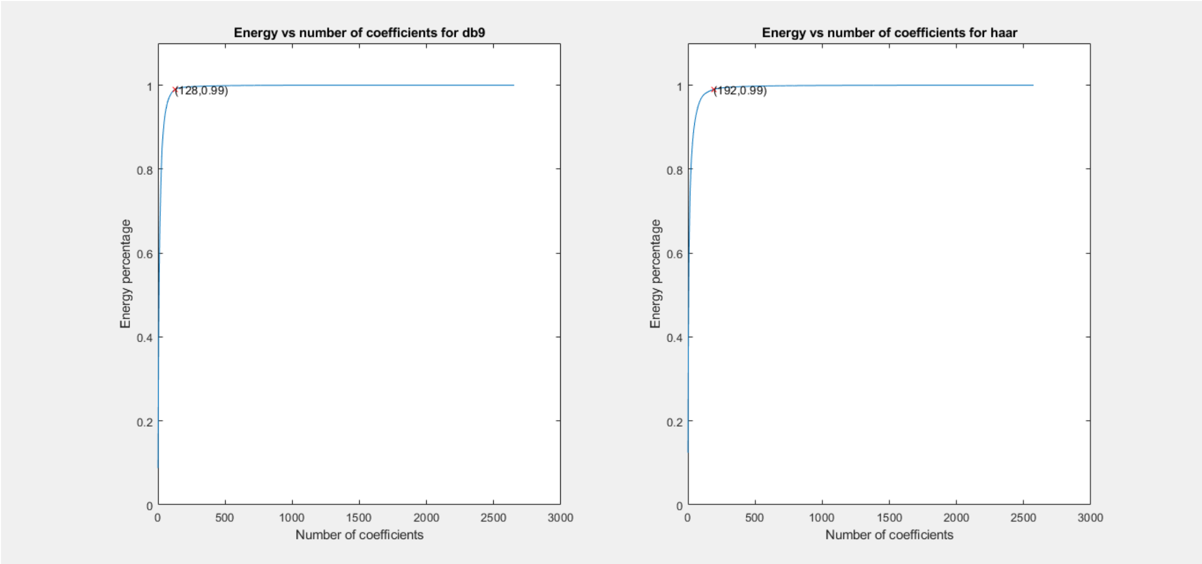


Figure 16: Energy vs number of coefficients

So when using db9, we require the largest (in magnitude) 128 coefficients while haar requires 192.

Below is the compressed signals, plot in the same graph.

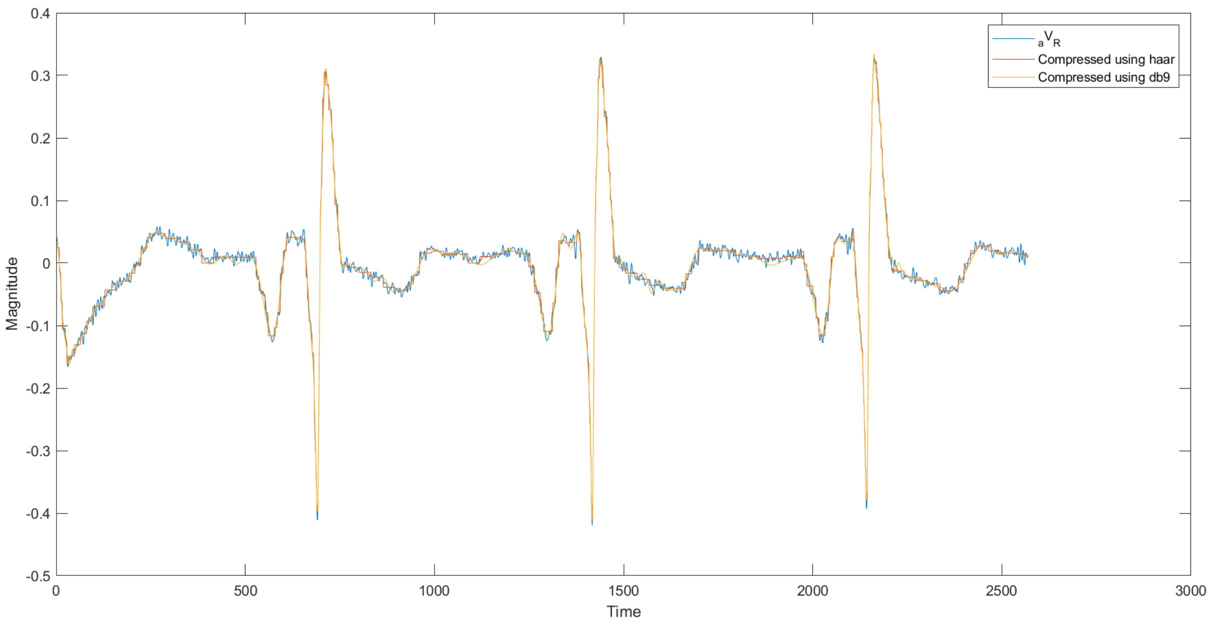


Figure 17: Compressed results

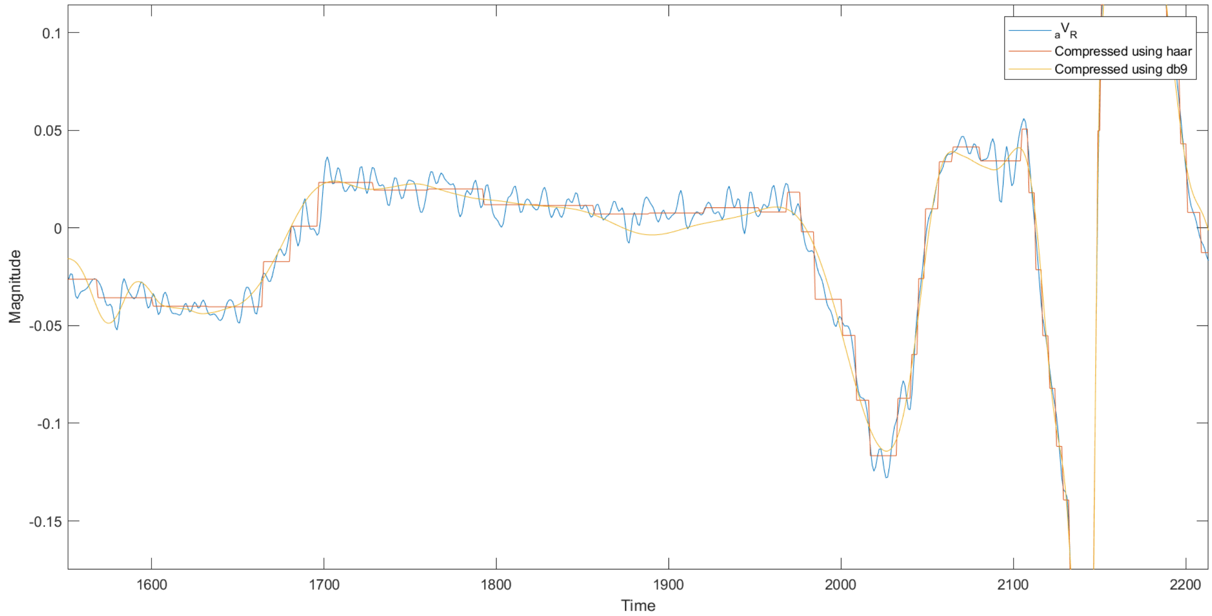


Figure 18: A zoomed view on a selected part

The compression ratio when using db9 is better when compared with the haar. Also the accuracy.

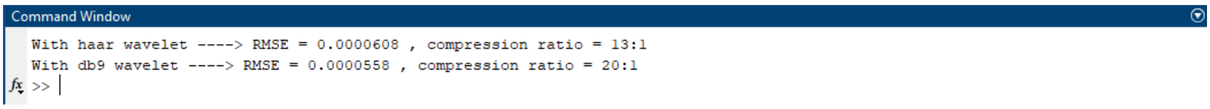


Figure 19: Compression ratios

Since the ECG signal is not consisted of straight lines like in haar, it's better to go with db9 in this case.