Spectral Analysis

Jayotsana Sharma

April 2025

1 Spectral analysis:

1.1 Detection of oscillations with periodograms:

Here is the spectrum and periodogram of the basic and windowed periodograms. We applied the Hamming window and the Welch window to our original signal and examined the periodograms to observe the effects of each window on the signals. We can see the different effects of each window on the signals as follows:

Signal x1

- Periodogram : In a signal's periodogram, the main lobe is narrow, which shows high resolution. On the other hand, high amplitude side lobes indicate significant frequency leakage in the signal.
- Hamming Window: The Hamming Window widens the main lobe of the periodogram. This widening shows that frequency resolution is reduced. The lower amplitude of the side lobes results in less frequency leakage and a smoother estimate.
- Welch: Using this window, we can see that the main lobe is wider than before, which means the resolution is poorer. However, the side lobes are more suppressed, which reduces leakage and creates a smoother estimate.

Conclusion: Welch gives the cleanest and most stable estimate despite lower resolution due to a tradeoff of wider main lobe and reduced side lobes.

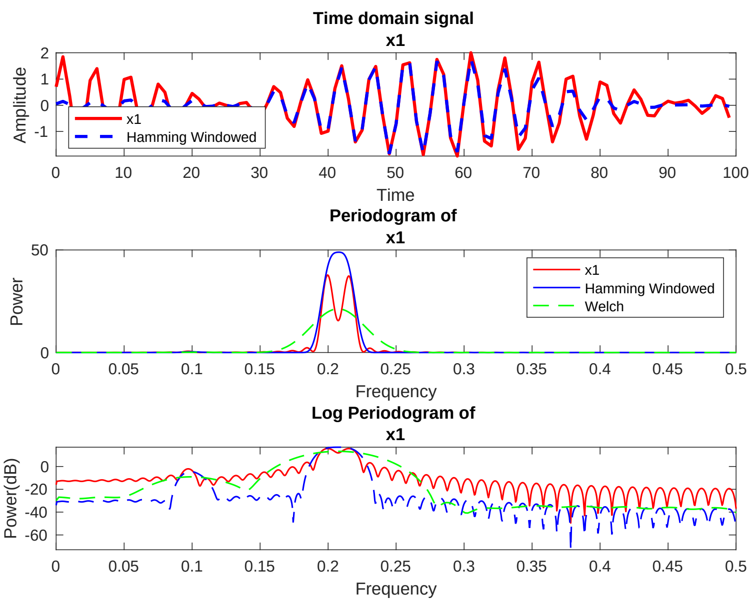


Figure 1: Signal and it's spectrum

Signal x2 Similar behavior to x1, but more noise-dominant.

- Periodogram: Periodograms with narrow main lobes and high side lobes indicate high frequency resolution but also result in a very noisy spectrum.
- Hamming: With the Hamming window, we observe better control of leakage; however, the resolution becomes worse.
- Welch: With the Welch window, we achieve the widest main lobe, but averaging reduces random fluctuations (noise).se)

Conclusion: Welch is best for noisy signals due to low side lobes and noise averaging, despite degraded resolution.

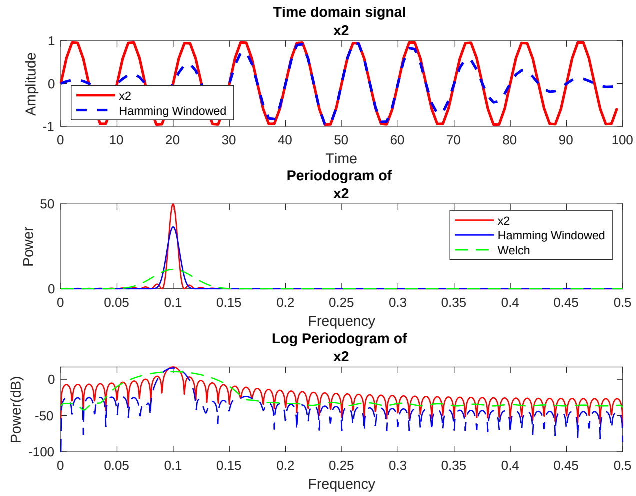


Figure 2: Signal and it's spectrum

Signal x3: Appears to have narrowband components (e.g., sinusoids).

- Periodogram: We observe a narrow main lobe that shows good resolution, but the strong side lobes indicate higher leakage in the spectrum.

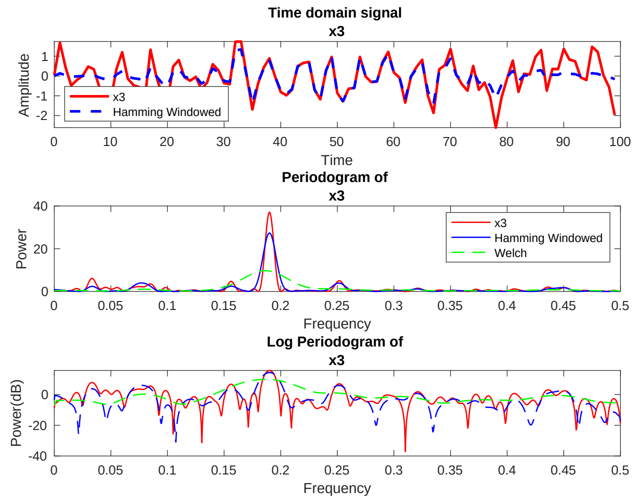


Figure 3: Signal and it's spectrum

- Hamming: This shows a good balance. It reduces side lobes significantly while keeping the main lobes fairly narrow. As a result, it provides better frequency resolution with less leakage.
- Welch: With the Welch method, the main lobes are too wide, and the peaks are blurred, making it difficult to resolve closely spaced frequencies well.

Conclusion: Hamming is best here. It maintains good resolution with significantly reduced leakage.

Signal x4: Appears to be wideband or high-dynamic-range.

- Periodogram: Shows too much leakage due to high side lobes.
- Hamming: This window improved leakage control, but still shows variance.
- Welch: This gives the broadest main lobe, but excellent side lobe suppression and noise smoothing.

Conclusion: Welch is most suitable due to strong leakage suppression, despite loss of frequency resolution.

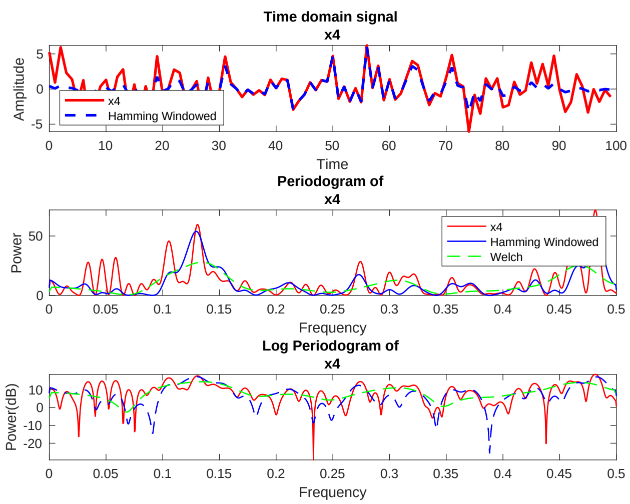


Figure 4: Signal and it's spectrum

1.2 High-resolution methods

1. The autoregressive (AR) spectral analysis method can be computed as follows:
(a) First, define the order P of the model (tip: one complex exponential corresponds to one pole in the transfer function.)
We have choosen P=4
An AR model is exactly an all-pole filter in addition of white noise. It shows we have 4 poles.
(b) Then, estimate the AR coefficients. Among the several possible methods, you may use the covariance one:

$$[a, sigma2] = arcov(x, P);$$

AR Spectral Estimation via the Covariance Method: Let $x[n]$ be an AR(P) process defined by the difference equation

$$x[n] = \sum_{k=1}^{P} a_k \, x[n-k] \; + \; b[n], \tag{1}$$

where $b[n]$ is white noise with variance σ^2. Equivalently, $x[n]$ can be seen as the output of the all-pole filter

$$H(z) \;=\; \frac{1}{A(z)}, \quad A(z) \;=\; 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_P z^{-P}, \tag{2}$$

driven by $b[n]$.

Estimation of AR coefficients and noise variance Using the covariance method (`[a,`σ^2`] = arcov(x,P)` in MATLAB) one obtains

$$\boldsymbol{a} = [\,1,\ a_1,\ \ldots,\ a_P\,]^{\mathsf{T}}, \quad \sigma^2 = \mathrm{Var}\{b[n]\}.$$

(c) Finally, the spectrum is computed by

$$ARspectrum = sigma2./abs((fft(a, Nf))).b2$$

Explain why.

Explaination:

Frequency response of the AR polynomial Sampling $A(e^{j\omega})$ on an N_f-point grid is done via the FFT:

$$A\big(e^{j\omega_m}\big) \;\approx\; A_{\text{fft}}[m] = \;\text{fft}\big(\boldsymbol{a},\, N_f\big), \quad \omega_m = \frac{2\pi m}{N_f},\; m = 0, \ldots, N_f - 1. \tag{3}$$

Power spectral density White noise $b[n]$ has flat PSD $S_b(\omega) = \sigma^2$. The output PSD of $x[n]$ is

$$S_x(\omega) = S_b(\omega)\,\big|H(e^{j\omega})\big|^2 = \sigma^2 \left|\frac{1}{A(e^{j\omega})}\right|^2 = \frac{\sigma^2}{\big|A(e^{j\omega})\big|^2}. \tag{4}$$

Discretely,

$$\widehat{S}_x(\omega_m) = \frac{\sigma^2}{\big|A_{\text{fft}}[m]\big|^2} \quad \Longrightarrow \quad \texttt{ARspectrum = } sigma2./abs(fft(a, Nf)).^2; \tag{5}$$

2. The MUSIC method will be directly implemented from the Matlab function: S MUSIC = pmusic(x,P,freq,Fs); where P is the number of complex exponentials that are searched and freq is the corresponding frequency axis on which the MUSIC indicator value is computed.

Impact of AR Method and MUSIC Method on Signals

In Figure 5, we compared high-resolution (HR) methods—Auto-Regressive (AR) and MUSIC. Our findings are summarized below:

Signal x_1

The AR method shows two peaks with some bandwidth, while the MUSIC method gives two perfect impulse-like spikes when the model order is set to two.

Signal x_2

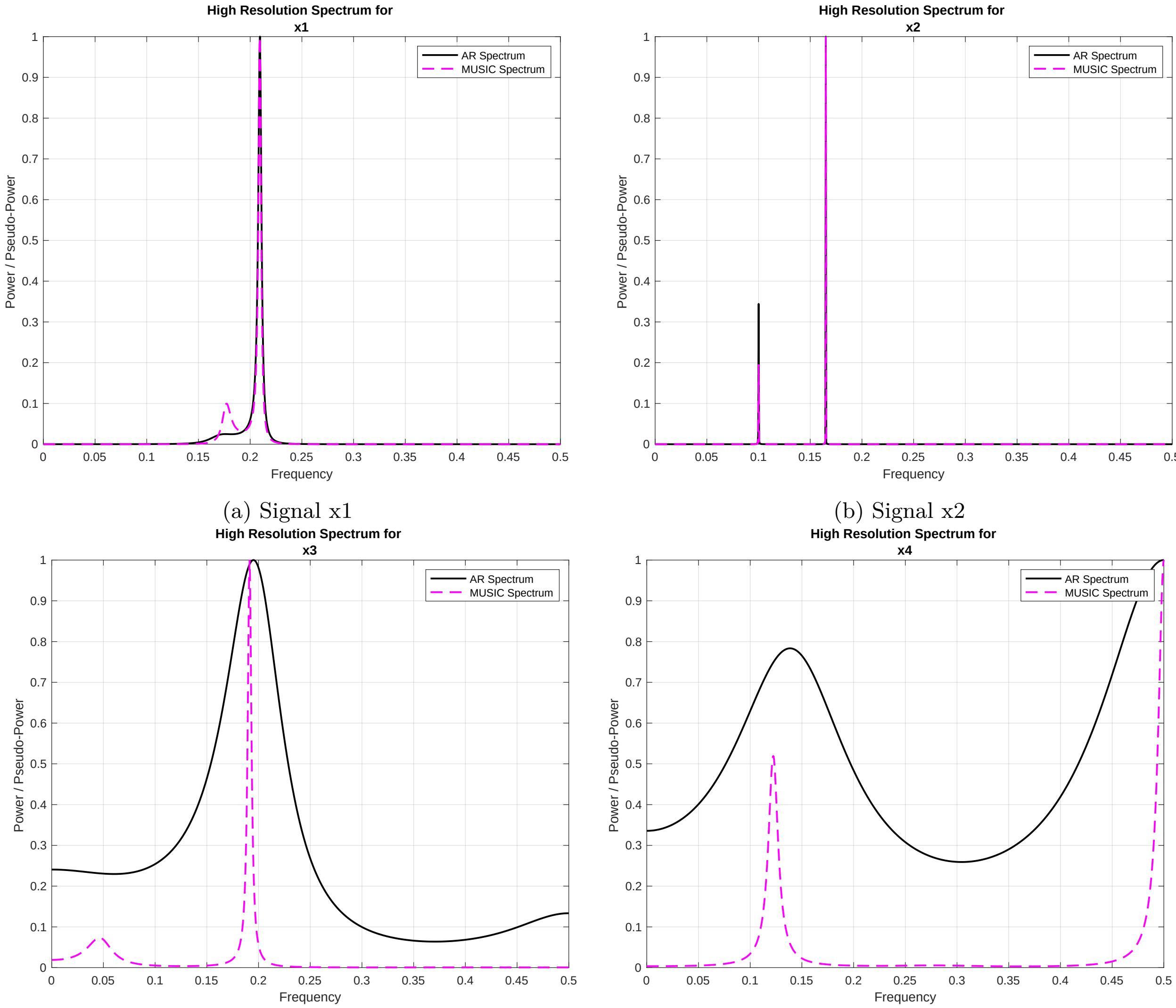
Both methods produce two well-defined, impulse-like peaks.

Signal x_3

The AR method gives two peaks—one sharp and one with a broader, lower-power bandwidth. The MUSIC method gives two sharp, well-defined peaks.

Signal x_4

The AR method shows two broad peaks. The MUSIC method gives two sharp peaks with different amplitudes.



(a) Signal x1

(b) Signal x2

(c) Signal x3

(d) Signal x4

Figure 5: High-Resolution Spectra (AR and MUSIC) for Signals x1–x4

3. Sparsity-Regularized Fourier Spectrum

The sparsity-regularized Fourier spectrum method is implemented via the function available on Hippocampus:

$$u = \texttt{minl1\_Fourier}(x,\ \mathrm{freq},\ \mu),$$

which computes the minimizer of the least-squares problem penalized by an ℓ_1-norm term (see course slide 44). Here, `freq` is the frequency axis and μ is the weight of the ℓ_1 penalty, to be tuned for each case.

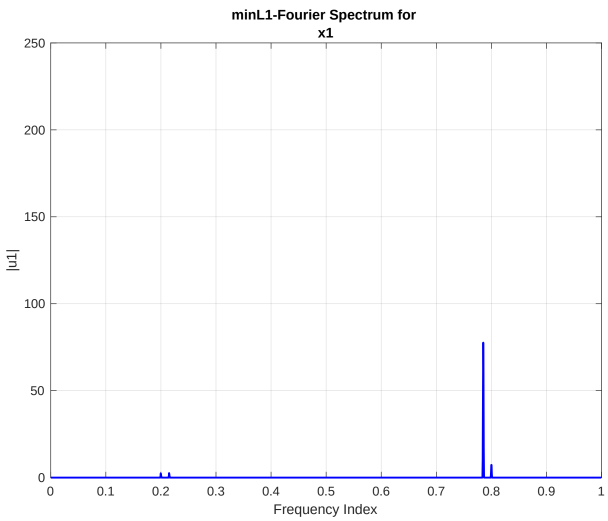
In figure 6

- **Signal x_1:**
  The spectrum shows exactly two sharp nonzero spikes at the signal frequencies, while all other bins are nearly zero. This confirms that ℓ_1 regularization has perfectly identified the two frequencies without leakage.

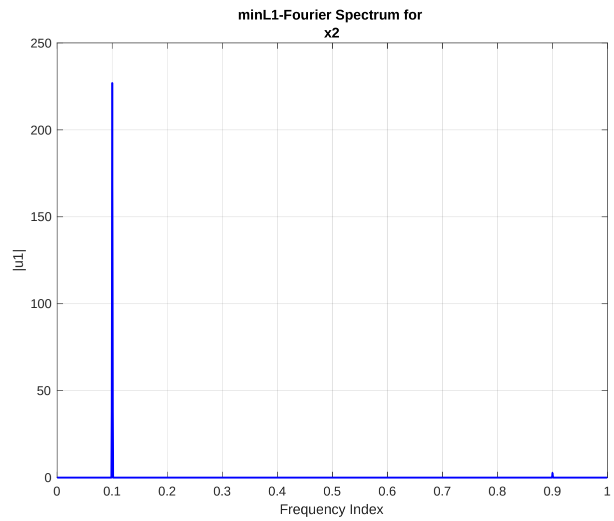
- **Signal x_2:**
  Only one coefficient is nonzero, located exactly at the signal's frequency. The rest of the spectrum is zero, showing ideal sparse recovery of a single sinusoid.

- **Signal x_3:**
  Two spikes appear: a taller one and a smaller one, both slightly higher than the rest of the spectrum. This indicates a reasonably good sparse solution.

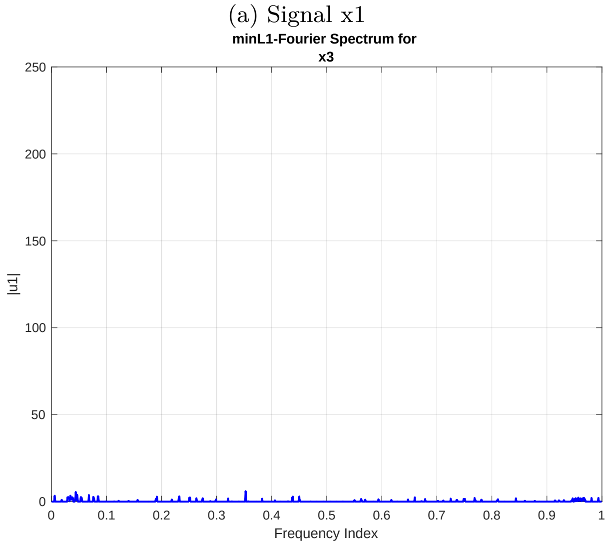
- **Signal x_4:**
  A few isolated spikes are scattered across the frequency range, showing that sparse ℓ_1 regularization does not work well for wide-band signals.



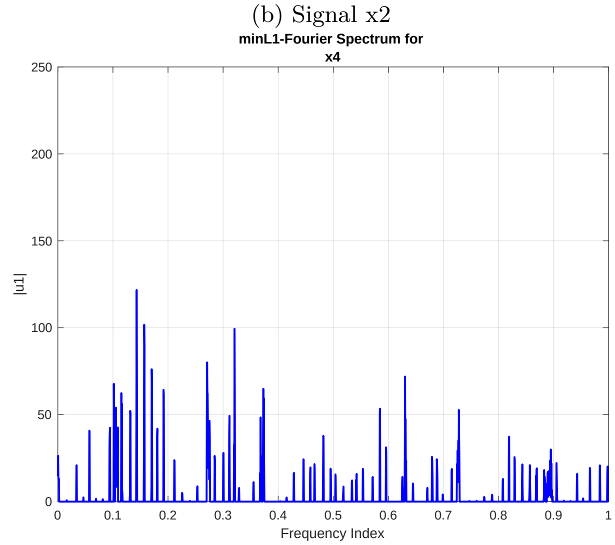
(a) Signal x1



(b) Signal x2



(c) Signal x3



(d) Signal x4

Figure 6: Frequency Index

Comparison Of output spactrum of high resolution methods from periodogram: In figure 7:

Signal x_1

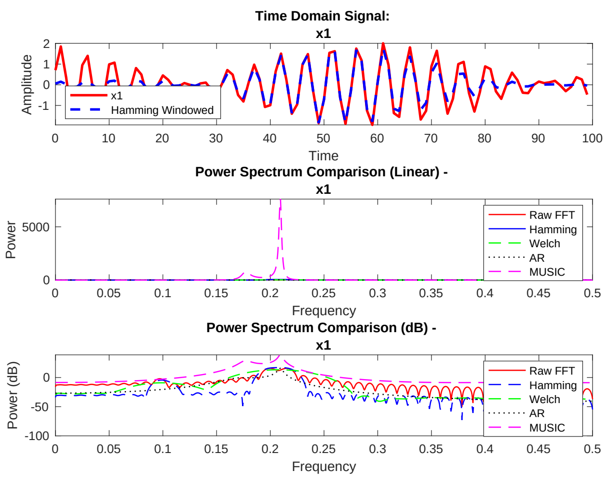
Window methods show leakage when two frequencies are close, but HR methods clearly separate them and give two sharp spikes exactly where they should be.

Signal x_2

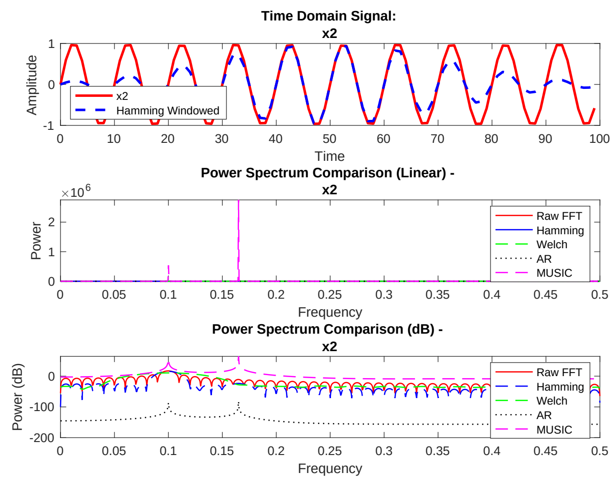
Window methods find the peak, but it is spread over a few bins. HR methods give an almost perfect single line at the correct frequency.

Signal x_3

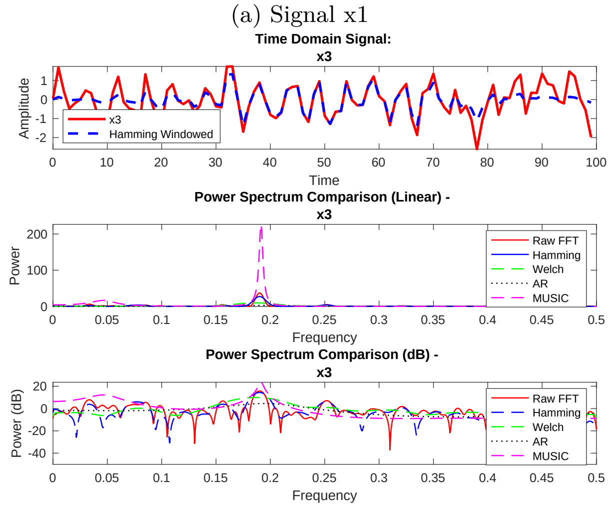
Because of sidelobes, the weak signal is hidden under the strong one when using windows. HR methods show both peaks clearly and at the right relative heights.



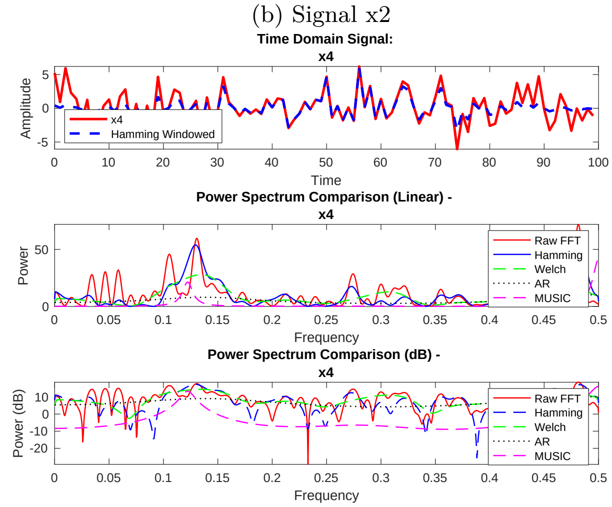
(a) Signal x1



(b) Signal x2



(c) Signal x3



(d) Signal x4

Figure 7: High-Resolution Spectra vs Windowed Spectra

Signal x_4

HR methods give many sharp spikes. They are not good for smooth, wide-band signals. Welch or other window methods work better in this case.

Overall, HR methods are more useful in resolving closely–spaced or low–power sinusoids, delivering superior frequency discrimination when their model assumptions are met. However, they can mislead on broadband or transient data, where classical windowed FFT approaches remain the most reliable choice.'

1.3 Exoplanet detection from the spectral analysis of time series

1. Variable win contains the observation window, and t contains the corresponding time axis. Compute and graph its Fourier transform in magnitude (use 4096 frequencies between 0 and Fs). Comment its shape.

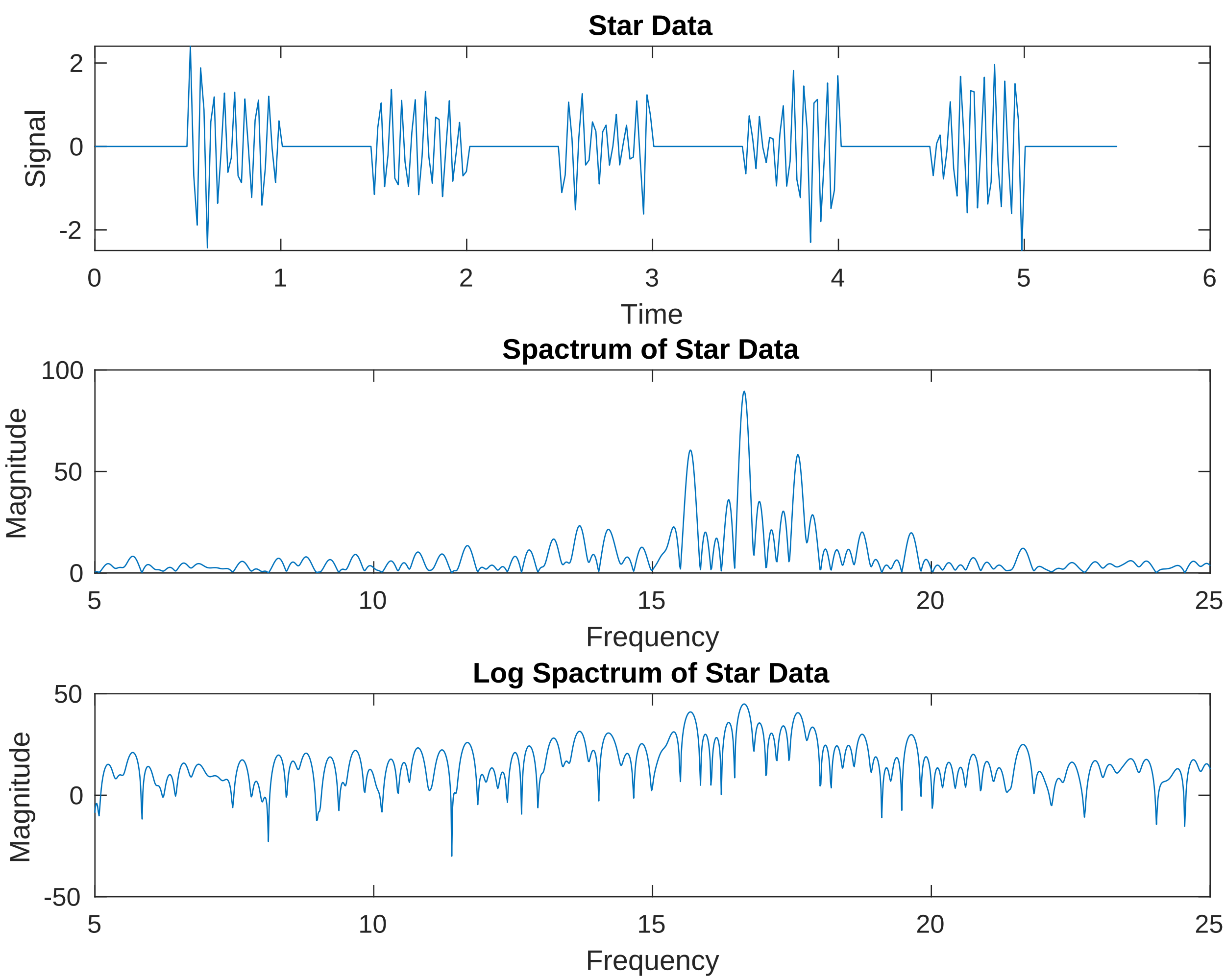


Figure 8: Signal and it's spectrum

Spactrum Shape: The window's spectrum has a narrow main lobe (High frequency resolution), slowly decaying sidelobes that cause significant leakage.

2. Compute and graph the periodogram of the data (variable x), and try to interpret it.

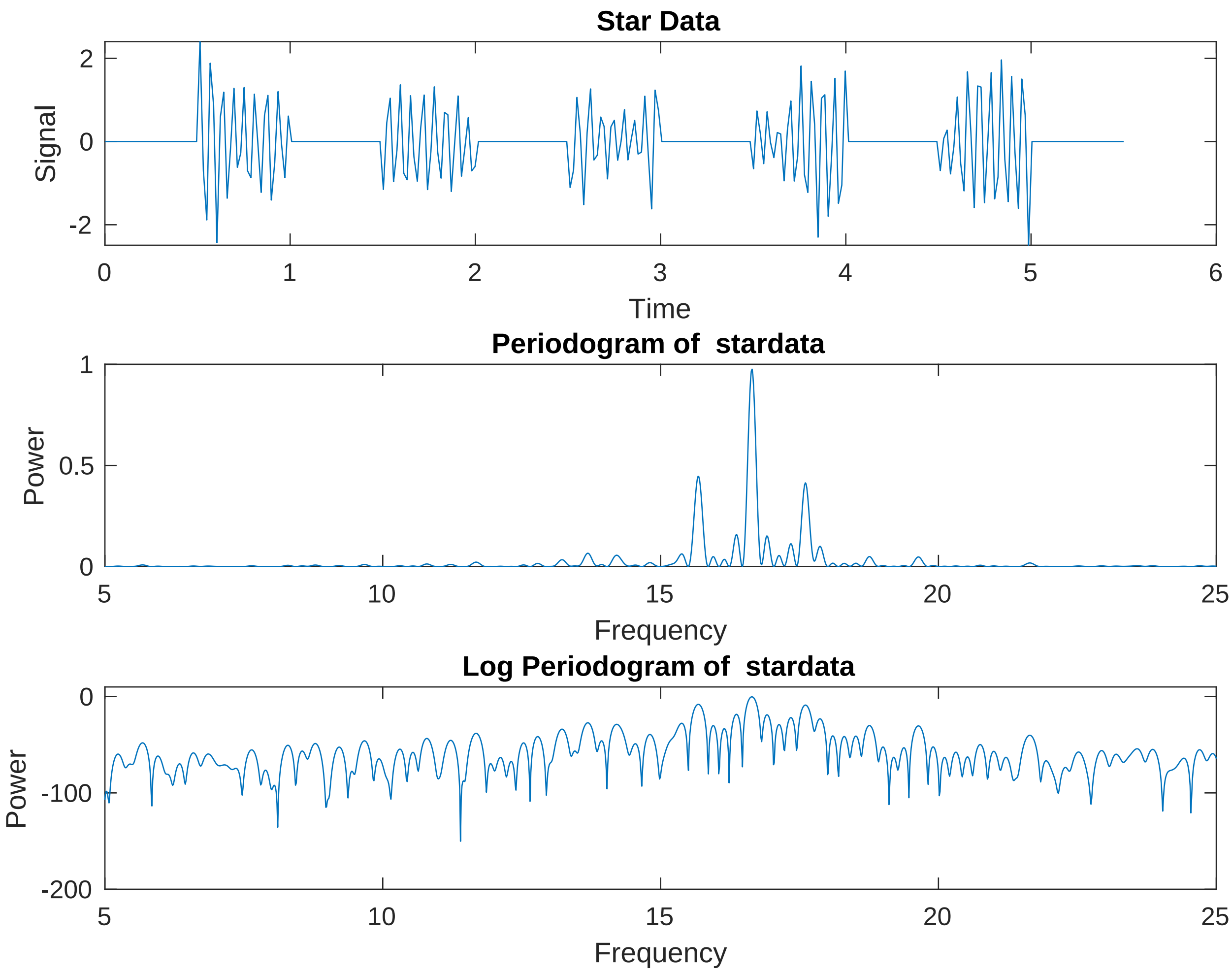


Figure 9: Signal and it's Periodogram

Periodogram: Shows a big, sharp spike shows the main oscillation and a smaller spike its overtone. Outside those peaks the spectrum is mostly flat, though on a log scale you can see tiny bumps from weaker modes or noise.

3. In order to reduce artifacts, a classical method1 consists in the following iterative procedure: the main frequency is estimated by the Fourier transform, then its contribution to the data is removed. The residual signal is then processed again and a new frequency is searched in its Fourier transform, until no significant component can be found in the residual.
   Implement this method where, at each iteration:
   i) the main frequency f_i is estimated at the maximum of the periodogram of the residual signal;
   ii) the corresponding cosine and sine components are obtained by least-squares estimation. This is a trivial operation, which is performed by the given function:

$$[alpha, beta] = estimcossin(residual, t, f_i)$$

   The contribution of the corresponding sine wave is then

$$cos(2f_i t) + sin(2f_i t)$$

   iii) The analysis continues or terminates, depending on whether relevant features are found in the periodogram.

4. Implement this algorithm and display, at each iteration, the residual data as a function of time, its Fourier transform and the detected components.

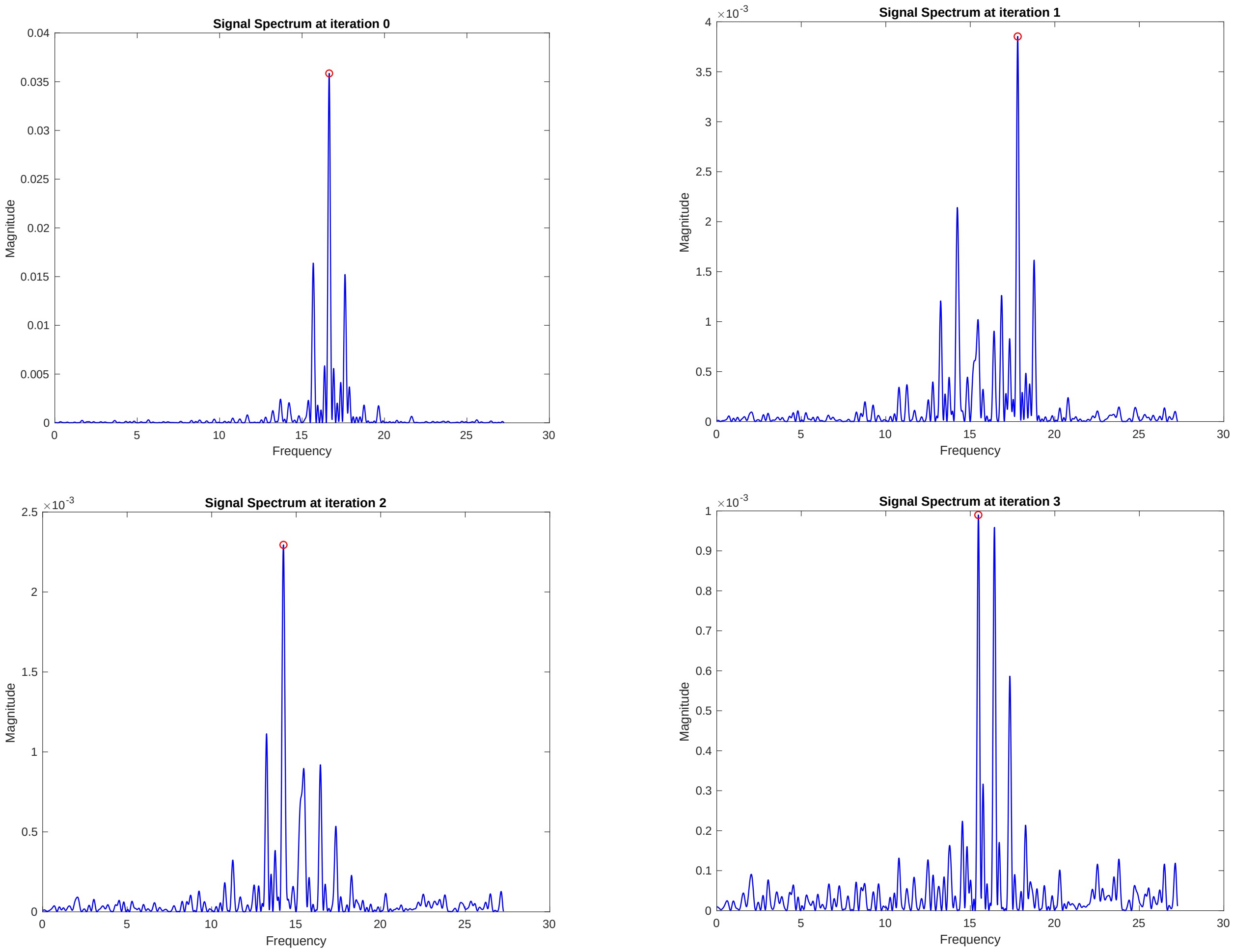


Figure 10: Signal Peaks at each iteration to be removed

Iterative Method: After implementing the method on each iteration we got the peaks to be removed as shown in figure 10 and we also got the residual signal after each iteration as shown in figure 11. Here used threshold value is 0.01

In 1st iteration one clear peak at around 17 Hz rise well above the noise floor, indicating the algorithm's first targets. We now see only one prominent spike around 18 Hz; the former spike is gone, and everything else sits at or near the noise level. This process is going on till all significant frequencies been removed.

Residual Signal: After each iteration, we can observe the residual signal in the blue plot of Figure 11. By the final iteration, the residual signal no longer contains any significant frequency components and appears almost flat.

.

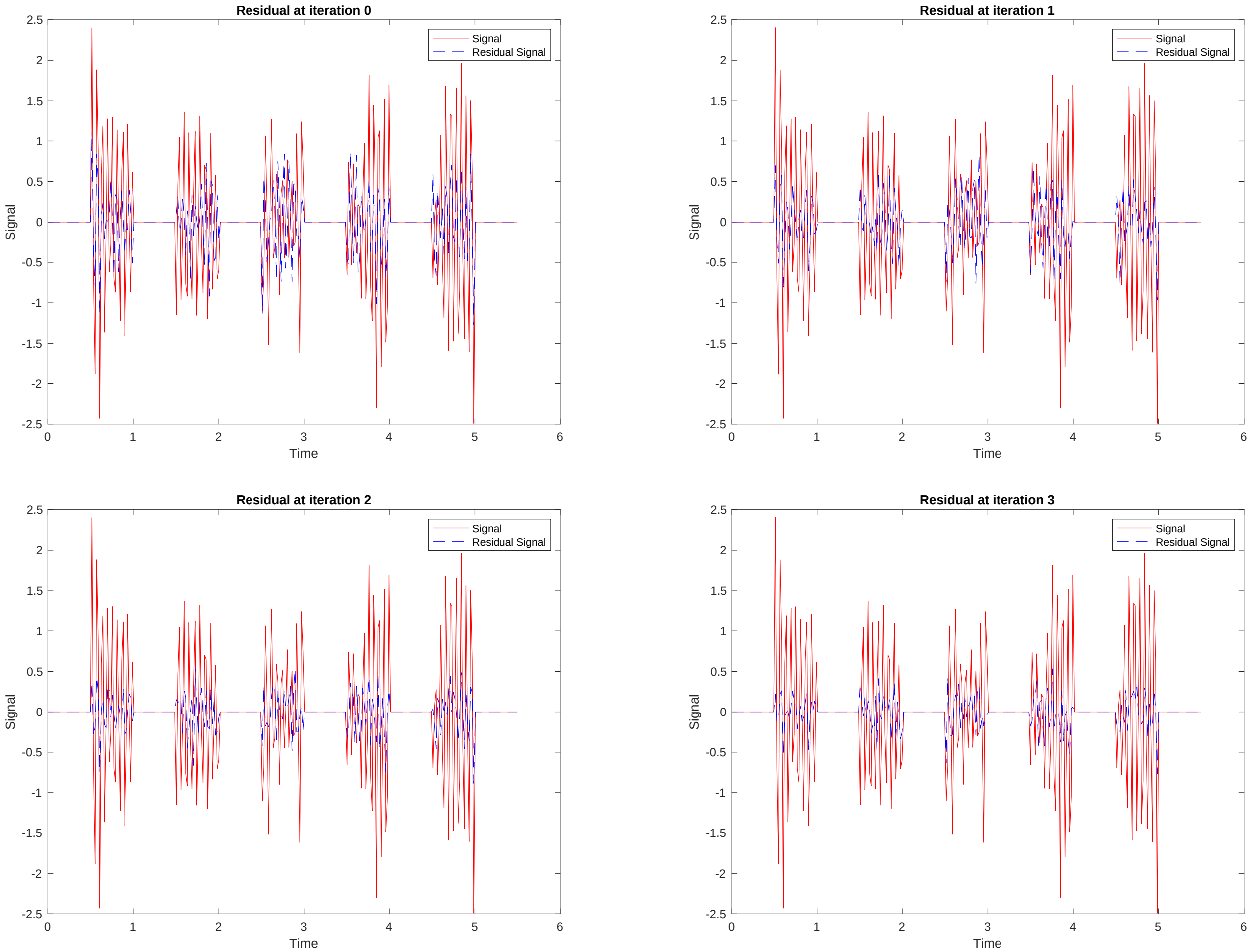


Figure 11: Residual Signal at each iteration

5. Comment on the results. How many frequencies (and therefore, exoplanets) can you detect?

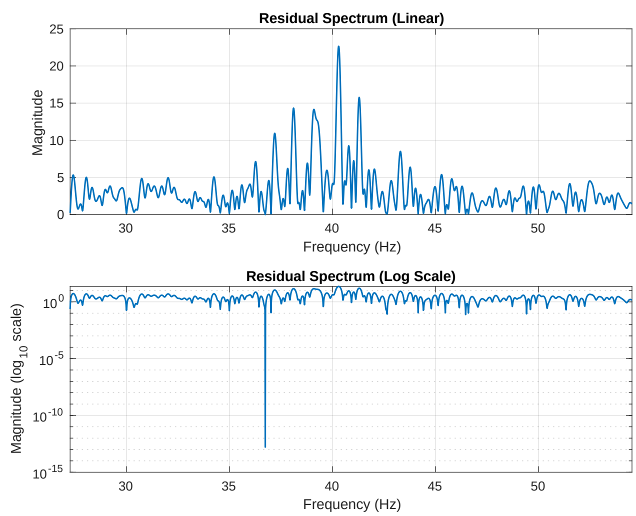


Figure 12: Reidual Signal Spectrum

In Figure 13, after subtracting the detected sinusoid(s), the residual periodogram is essentially flat—no peaks exceed the noise level. All that remains are broadband fluctuations and small artifacts at the window's sidelobe frequencies, with the main planet signature completely removed.

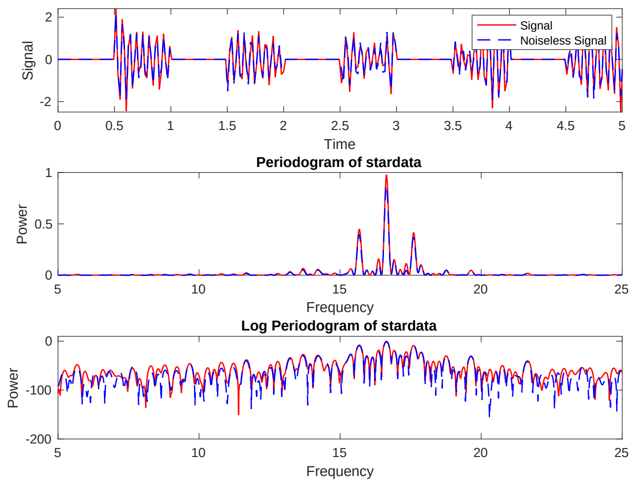


Figure 13: Periodogram of Original Signal and Signal afer Removing the Noise