

Windowing

Why we use windowing?

When you use the FFT to convert a signal from the time domain to the time-frequency domain you are basing it on a finite number of samples.

- FFT is based on a finite number of samples
- FFT assumes that sample is one period of a periodic signal
- when there are an integer number of periods in the sample this works out very nicely and data matches FFT's assumption.
 - this usually is not the case :(
 - we use window functions to reduce the aliasing this causes
 - the finiteness of the measured signal may result in a truncated waveform
- when number of periods in the sample is not an integer its endpoints are discontinuous
- artificial discontinuities show up in FFT as high frequency aliasing not found in original signal
 - aliasing is often FAR higher than the nyquist limit
- Spectrum the FFT returns is smeared
 - energy from one frequency appears to leak over into neighboring frequencies
 - called *spectral leakage*

Windowing helps minimize the effects of performing an FFT over a noninteger number of cycles!

Basic idea of windowing

- windowing reduces the amplitudes of the samples at the boundaries of the period
- the idea is this reduces the aliasing of the artificial discontinuities in the FFT
- windows generally approach zero towards their edges
- windows general have a single maxima at the center

Considerations when windowing

was tired when writing this so it needs editing!

Windowing of a sample waveform i.e. $\cos(\omega t)$ causes its Fourier transform to develop non-zero values (spectral leakage) at frequencies other than ω . This interference tends to be greatest at frequencies near ω and decreases as you get farther from ω .

This becomes more of a problem when you have multiple waveforms in the same sample that are dissimilar and one component is weaker than the others. Leakage from the stronger component can obscure the weaker frequency's presence. A rectangular window is good for sinusoids of comparable strength, but it is a poor choice for sinusoids of disparate amplitudes. This is sometimes described as sinusoids with a low-dynamic-range.

On the other end of the spectrum windows with the poorest resolution and sensitivity. These can reveal relatively weak sinusoids in the presence of additive random noise. That is because noise produces a stronger response with high-dynamic-range windows than with high-resolution windows. Therefore, high-dynamic range windows are often justified in wideband applications where the spectrum being analyzed is expected to contain components of various amplitudes.

In between are moderate windows, such as the Hamming and Hann windows.

General Types

Low-Dynamic-Range Windows

- rectangular window is good for sinusoids of comparable strength
- poor results with sinusoids of disparate amplitudes
- low dynamic range

High-Dynamic-Range Windows

- these are used in wide-band applications. spectrum being analyzed contains many different components of various amplitudes.
- reveal relatively weak sinusoids in the presence of additive random noise.
- noise produces a stronger response with high-dynamic-range windows than with high-resolution windows.

When selecting a type of window function

todo: indepth analysis of types of window functions

- in general the Hanning (Hann) window is satisfactory in almost all cases
 - has good frequency resolution
 - reduced spectral leakage
the tendency for energy in one frequency to appear to smear into neighboring frequencies
- if

Moderate Windows

such as hamming and Hann windows

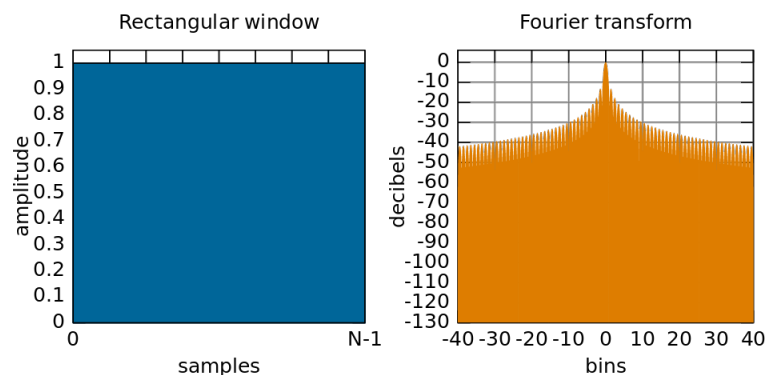
- commonly used in narrowband applications
- tradeoff between components of similar strengths and frequencies and dissimilar strengths and frequencies

Window Functions

B-Spline window: Rectangular Window

This is the simplest window. it makes it appear as though the wave form simply turns on and off.

$$w(n) = 1$$

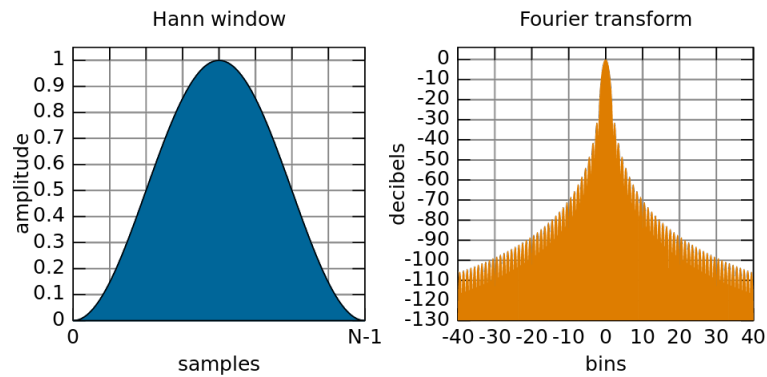


B-Spline Window: Triangular Window

$$w(n) = 1 - \left| \frac{n - \frac{N-1}{2}}{\frac{L}{2}} \right|$$

Hann (Hanning Window)

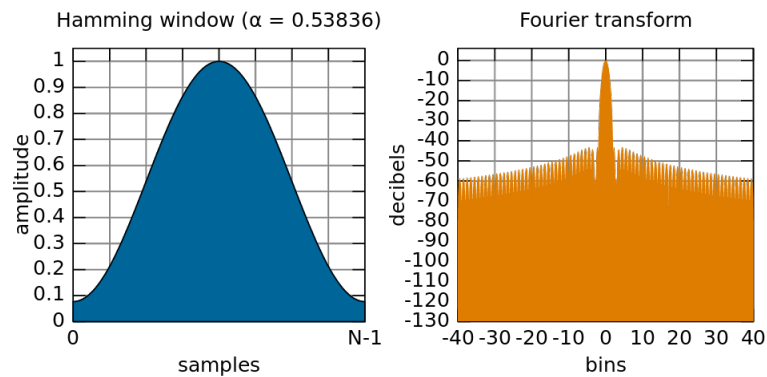
$$w(n) = 0.5 \left(1 - \cos\left(\frac{2\pi n}{N-1}\right) \right) = \text{sinc}\left(\frac{2\pi n}{N-1}\right)$$



Hamming Window

The window is optimized to minimize the maximum (nearest) side lobe, giving it a height of about one-fifth that of the Hann window.

$$w(n) = \alpha - \beta \cos\left(\frac{2\pi n}{N-1}\right)$$



Sources

<http://www.ni.com/white-paper/4844/en/>
https://en.wikipedia.org/wiki/Window_function
https://en.wikipedia.org/wiki/Hann_function