Numerical Methods 4

Physics 129AL

Zihang Wang
10/18/2023

Plane Wave

The solution to the wave equation,

$$\ddot{u} = c^2 \nabla^2 u, \qquad \frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}.$$

are plane waves.

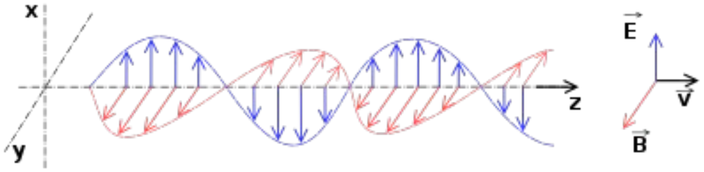
$$Ae^{-i(kx+\omega t)} + Be^{i(kx-\omega t)}$$

$$e^{ix} = \cos x + i \sin x$$

Can we use them as basis to describe functions?

Yes, because they are Orthogonal Functions!

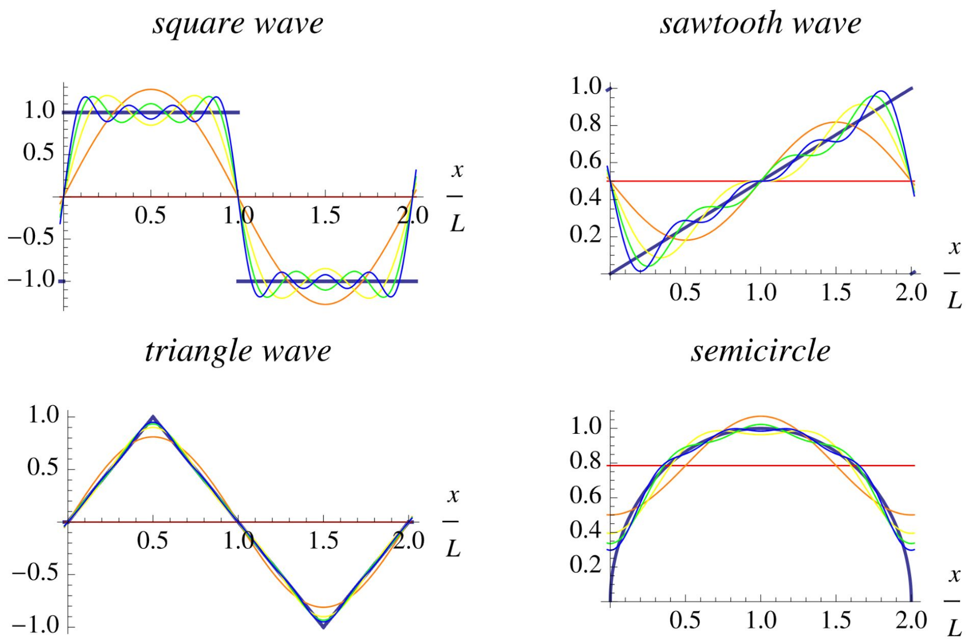
$$\langle f(x) \mid g(x) \rangle \equiv \int_a^b f(x)\, g(x)\, w(x)\, dx = 0.$$



Fourier Series

For periodic function with periodicity L (or function with a finite cutoff, L), we have,

$$f(x) = \sum_{n=0}^{\infty} A_n \cos\left(\frac{n\pi x}{L}\right) + \sum_{n=1}^{\infty} B_n \sin\left(\frac{n\pi x}{L}\right)$$



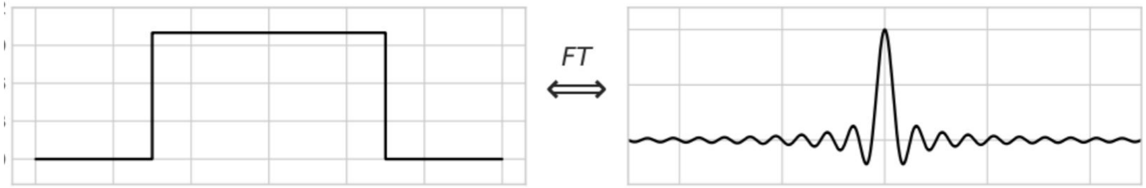
In this case, we only need discrete basis functions, labeled by n, to form a complete basis set.

Fourier Transform

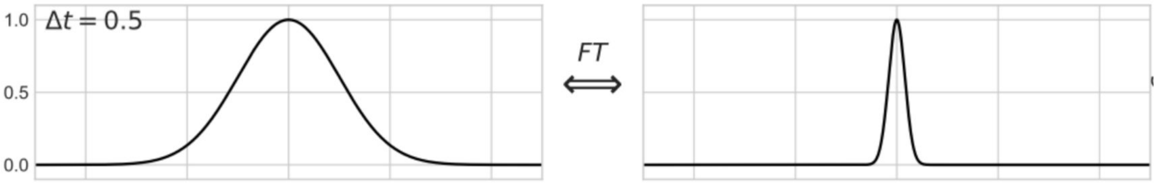
For a general function, we have,

$$F(k) = \mathcal{F}_x[f(x)](k) = \int_{-\infty}^{\infty} f(x)\, e^{-2\pi i k x}\, dx$$

In this case, we must use **all** plane waves to form a complete basis set.



Square pulse



Gaussian

Fourier Transform

- Complete basis

$$\omega \equiv 2\pi f \qquad H(\omega) \equiv [H(f)]_{f=\omega/2\pi}$$

$$H(f) = \int_{-\infty}^{\infty} h(t) e^{2\pi i f t} \, dt \qquad H(\omega) = \int_{-\infty}^{\infty} h(t) e^{i\omega t} \, dt$$

$$h(t) = \int_{-\infty}^{\infty} H(f) e^{-2\pi i f t} \, df \qquad h(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} H(\omega) e^{-i\omega t} \, d\omega$$

If . . .	then . . .
$h(t)$ is real	$H(-f) = [H(f)]^*$
$h(t)$ is imaginary	$H(-f) = -[H(f)]^*$
$h(t)$ is even	$H(-f) = H(f)$ [i.e., $H(f)$ is even]
$h(t)$ is odd	$H(-f) = -H(f)$ [i.e., $H(f)$ is odd]
$h(t)$ is real and even	$H(f)$ is real and even
$h(t)$ is real and odd	$H(f)$ is imaginary and odd
$h(t)$ is imaginary and even	$H(f)$ is imaginary and even
$h(t)$ is imaginary and odd	$H(f)$ is real and odd

Properties of Fourier Transforms

$h(at) \Longleftrightarrow \frac{1}{\lvert a\rvert} H(\frac{f}{a})$		time scaling
$\frac{1}{\lvert b\rvert} h(\frac{t}{b}) \Longleftrightarrow H(bf)$		frequency scaling
$h(t - t_0) \Longleftrightarrow H(f)\, e^{2\pi i f t_0}$		time shifting
$h(t)\, e^{-2\pi i f_0 t} \Longleftrightarrow H(f - f_0)$		frequency shifting

- Convolution theorem: $g * h \equiv \int_{-\infty}^{\infty} g(\tau) h(t-\tau)\, d\tau$

$$\int g(\tau)h(t-\tau)d\tau$$

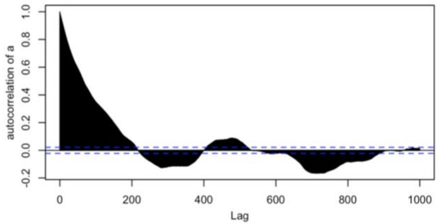
$$= \int \frac{dw}{2\pi} e^{-iw\tau} g(w) \int \frac{dw'}{2\pi} e^{-iw'(t-\tau)} h(w') d\tau$$

$$\int e^{-i(w-w')\tau} d\tau = 2\pi\delta_D(w-w')$$

$$= \int \frac{dw}{2\pi} e^{iwt} g(w) h(w)$$

$$\int \frac{dw'}{2\pi} 2\pi\delta_D(w-w') = 1$$

Correlation Function and Power Spectrum



$$\mathrm{Corr}(g,h)(t) = \int g(\tau+t)h(\tau)d\tau$$

$$= \int\int \frac{dw}{2\pi} e^{-iw(\tau+t)} g(w) \frac{dw'}{2\pi} e^{-iw'\tau} h(w') d\tau$$

$$\int e^{-i\tau(w+w')} d\tau = 2\pi\delta_D(w+w')$$

$$= \int \frac{dw}{2\pi} g(w)h(-w)e^{-iwt} = \mathrm{FT}\Big(g(w)h^*(w)\Big)$$

POWER SPECTRUM $\quad P(w) = |g(w)|^2$

$$t=0, \quad \int |g(w)|^2 \frac{dw}{2\pi} = \int g(\tau)^2 d\tau$$

Parseval's theorem

Power Spectrum in higher dimensions

$$g(\vec{x}) \quad \mathrm{FT}: g(\vec{k}) = \int d^\mu \vec{x} \cdot g(\vec{x}) e^{i\vec{k}\cdot\vec{x}}$$

$$g(\vec{x}) = \int \frac{d^\mu \vec{k}}{(2\pi)^N} e^{-i\vec{k}\cdot\vec{x}} g(\vec{k})$$

$$\mathrm{Corr}\left[g(\vec{x}_1) g(\vec{x}_2 = \vec{x}_1 + \vec{x})\right] = \int \frac{d^\mu \vec{k}}{(2\pi)^N} P(\vec{k}) e^{-i\vec{k}\cdot\vec{x}}$$

$$P(\vec{k}) = g(\vec{k}) g^*(\vec{k})$$

If isotropic, only depends on *k* length

$$P(\vec{k}) = P(k)$$

$$\mathrm{Corr}(\vec{x}) = \int \frac{d^\mu \vec{k}}{(2\pi)^N} P(\vec{k}) e^{-i\vec{k}\cdot\vec{x}}$$

Discrete Sampling: Sampling Theorem

- We sample interval in points of length Δ: $h_n = h(n\Delta)$
- Nyquist frequency: $f_c = 1/2\Delta$
- Sampling theorem: if the function h(t) does not have frequencies above f_c ($h(f) = 0$ for $f > f_c$) it is bandwidth limited. Then $h(t)$ is completely determined by h_n:

$$h(t) = \Delta \sum_{n=-\infty}^{+\infty} h_n \frac{\sin[2\pi f_c(t - n\Delta)]}{\pi(t - n\Delta)}$$

- This says that the information content is limited. If we know the maximum bandwidth frequency then we know how to sample the function using $f_c = 1/2D$ to get the full information content

Beyond Nyquist frequency: Aliasing

- All the power outside $-f_c < f < f_c$ is moved into this bandwidth by the process of discrete sampling
- Take $\exp(2\pi i f_1 t)$ and $\exp(2\pi i f_2 t)$: if $f_1 - f_2 = m/\Delta$ then the values at $h_n = h(n\Delta)$ differ by $\exp(2\pi i n m) = 1$: we cannot distinguish between these frequencies using discrete sampling
- In practice we thus want to look at the power spectrum as a function of f: if it goes to 0 as f approaches f_c then aliasing is small, otherwise we need to decrease the sampling Δ, ie increase $f_c = 1/2\Delta$

Interval

Beyond Nyquist frequency: Aliasing

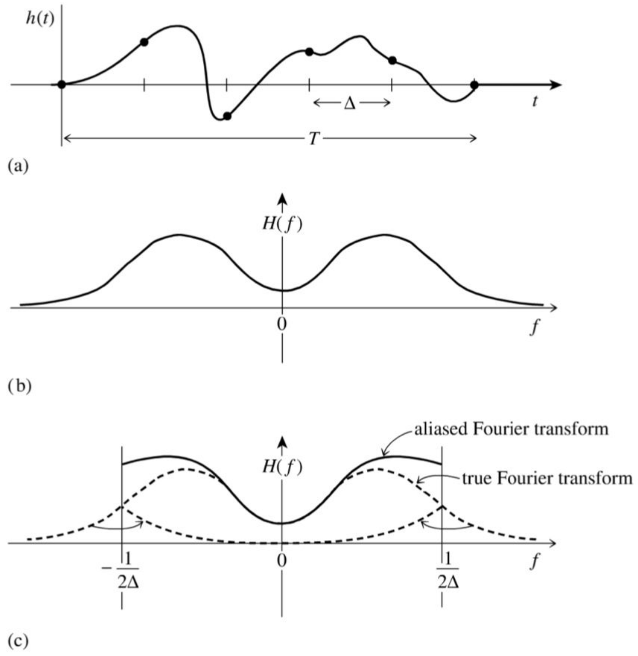


Figure 12.1.1. The continuous function shown in (a) is nonzero only for a finite interval of time T. It follows that its Fourier transform, whose modulus is shown schematically in (b), is not bandwidth limited but has finite amplitude for all frequencies. If the original function is sampled with a sampling interval Δ, as in (a), then the Fourier transform (c) is defined only between plus and minus the Nyquist critical frequency. Power outside that range is folded over or "aliased" into the range. The effect can be eliminated only by low-pass filtering the original function *before sampling*.

From Numerical Recipes, Press etal

Discrete Fourier Transform (DFT)

- We measure a function on an interval $N\Delta$. Maybe the function is 0 outside the interval, otherwise we assume periodicity (periodic boundary conditions), because sin and cos are periodic

$$h_k \equiv h(t_k), \qquad t_k \equiv k\Delta, \qquad k = 0, 1, 2, \ldots, N-1$$

- Frequency range from $-f_c$ to f_c $\quad f_n \equiv \frac{n}{N\Delta}, \qquad n = -\frac{N}{2}, \ldots, \frac{N}{2}$

- DFT: $\quad H(f_n) = \int_{-\infty}^{\infty} h(t) e^{2\pi i f_n t} dt \approx \sum_{k=0}^{N-1} h_k \, e^{2\pi i f_n t_k} \Delta = \Delta \sum_{k=0}^{N-1} h_k \, e^{2\pi i k n/N}$

- $H(-f_c) = H(f_c)$ $\quad H(f_n) \approx \Delta H_n \qquad H_n \equiv \sum_{k=0}^{N-1} h_k \, e^{2\pi i k n/N}$

- Inverse DFT $\quad h_k = \frac{1}{N} \sum_{n=0}^{N-1} H_n \, e^{-2\pi i k n/N}$

- Parseval's theorem $\quad \sum_{k=0}^{N-1} |h_k|^2 = \frac{1}{N} \sum_{n=0}^{N-1} |H_n|^2$

Fast Fourier Transform (FFT)

- How expensive is to do FT? Naively it appears to be a matrix multiplication, hence $O(N^2)$

$$W \equiv e^{2\pi i/N} \qquad H_n = \sum_{k=0}^{N-1} W^{nk} h_k$$

- FFT: $O(N\log_2 N)$
- The difference is enormous: these days we can have $N > 10^{10}$. FFT is one of the most important algorithms of numerical analysis
- FFT existence became widely known in 1960s (Cooley & Tukey), but known (and forgotten) since Gauss (1805)

Fast Fourier Transform (FFT)

- Assume for now $N = 2^M$: one can show that FT of length N can be rewritten as the sum of 2 FTs of length N/2 (Danielson & Lanczos 1942): even (e) and odd (o)

$$
\begin{aligned}
F_k &= \sum_{j=0}^{N-1} e^{2\pi i jk/N} f_j \\
&= \sum_{j=0}^{N/2-1} e^{2\pi i k(2j)/N} f_{2j} + \sum_{j=0}^{N/2-1} e^{2\pi i k(2j+1)/N} f_{2j+1} \\
&= \sum_{j=0}^{N/2-1} e^{2\pi i kj/(N/2)} f_{2j} + W^k \sum_{j=0}^{N/2-1} e^{2\pi i kj/(N/2)} f_{2j+1} \\
&= F_k^e + W^k \; F_k^o
\end{aligned}
$$

- This can be recursively repeated, until we reach the length of 1, at which point we have for some n

$$F_k^{eoeeoeo\cdots oee} = f_n$$

Fast Fourier Transform (FFT)

- To determine n we take the sequence eoeeoo…oee, reverse the pattern of e and o, assign e=0 and o=1, and we get n in binary form.

$$F_k^{eoeeeoeo\cdots oee} = f_n$$

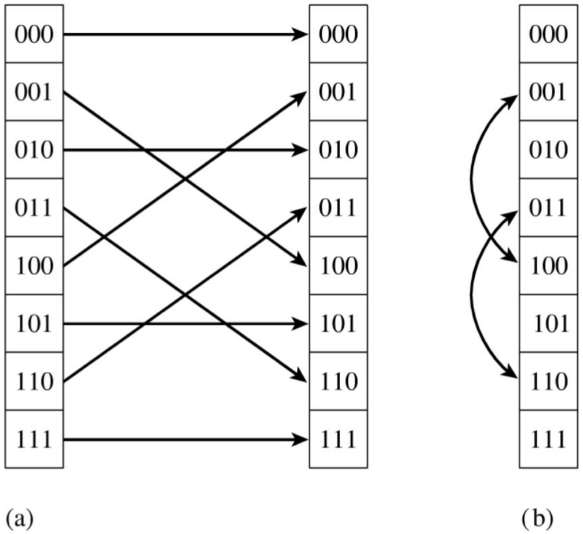


Figure 12.2.1. Reordering an array (here of length 8) by bit reversal, (a) between two arrays, versus (b) in place. Bit-reversal reordering is a necessary part of the fast Fourier transform (FFT) algorithm.

Filtering applications

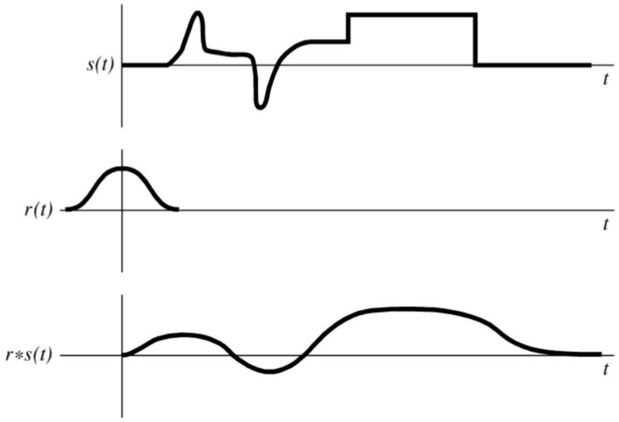
Figure 12.6.2. Fourier processing of an image. Upper left: Original image. Upper right: Blurred by low-pass filtering. Lower left: Sharpened by enhancing high frequency components. Lower right: Magnitude of the derivative operator as computed in Fourier space.

Low pass filter: smoothly set high f/ω to 0
High frequency sharpening: increase high f/ω

Derivative operator in FT: multiply Fourier modes with $i\omega$ (edge detection)

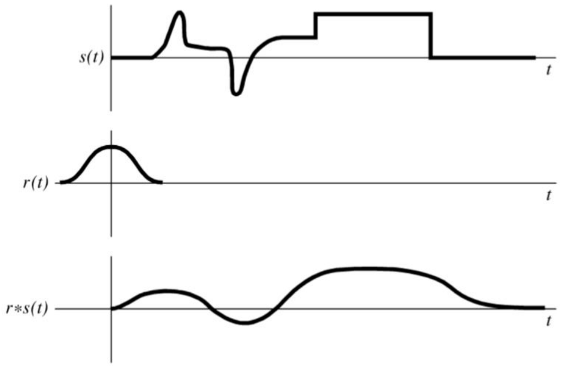
Convolutions with DFT

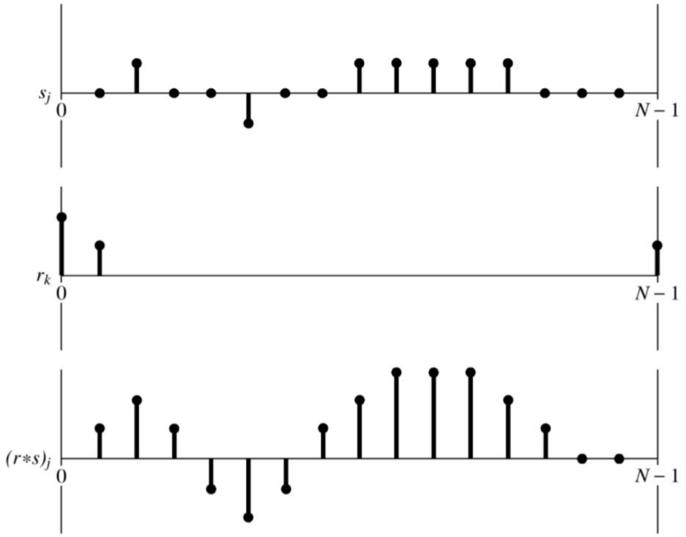
- We have data that have been convolved with some response. For example, we observe the sky through the telescope which does not resolve angles below λ/R, where R is its diameter. Here response is Airy function. To simulate the process we convolve data $s(t)$ with $r(t)$, by multiplying their FTs.
- If the data are perfect (no noise) we can deconvolve the process: we FT the data $r*s(t)$ to get $r(f)s(f)$ and divide by the convolution term $r(f)$ to get $s(f)$, then inverse FT to get $s(t)$.



Convolutions with DFT

- Discrete convolution assumes periodic signal on the interval





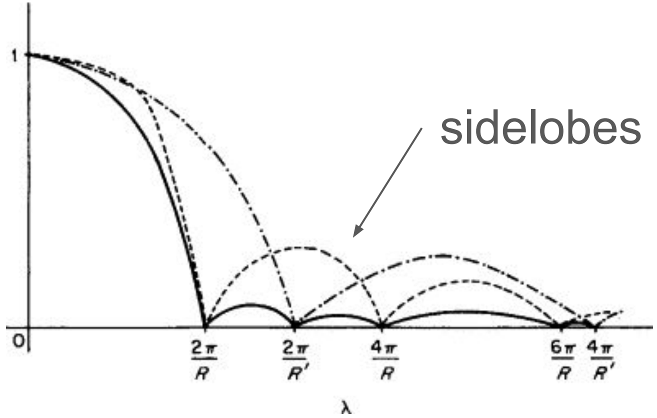
Centered at zero

$$(r * s)_j \equiv \sum_{k=-M/2+1}^{M/2} s_{j-k}\, r_k$$

Convolutions with DFT: Window Functions

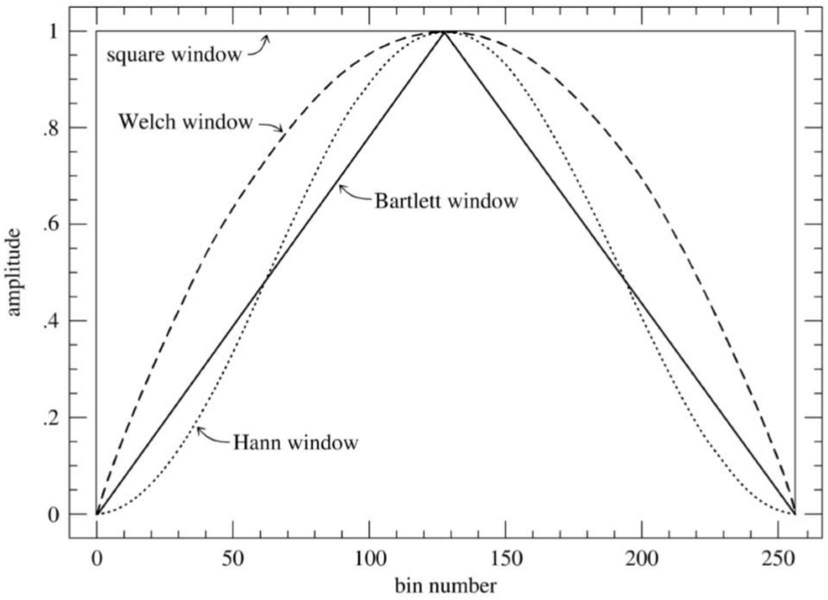
- Many signals in real-world applications have finite durations, and implicitly assumed that the signal repeats infinitely. This could lead to artifacts.
- Window functions are applied to signals before taking the Fourier Transform to reduce the abrupt cutoff of the signal at its edges.

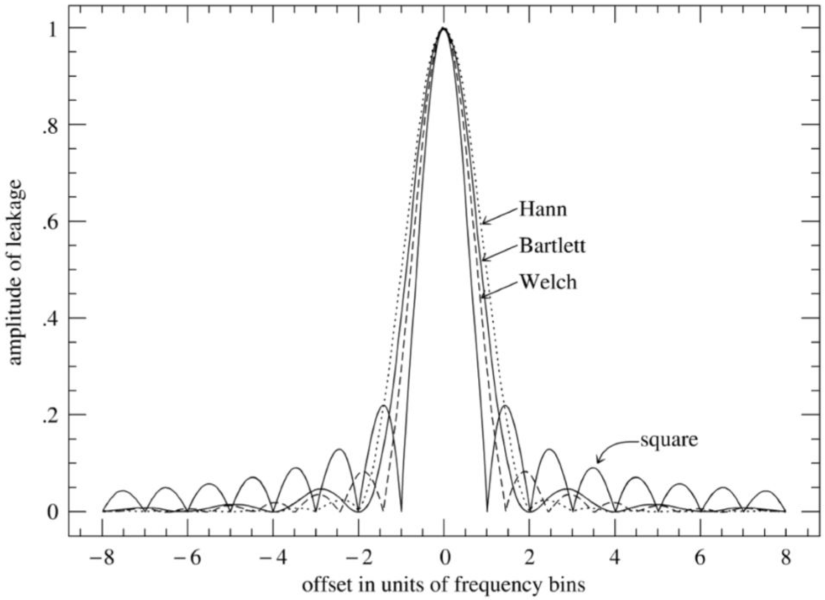
Window functions are commonly used to **minimize sidelobes**, by having a product in time domain (convolution in frequency space).



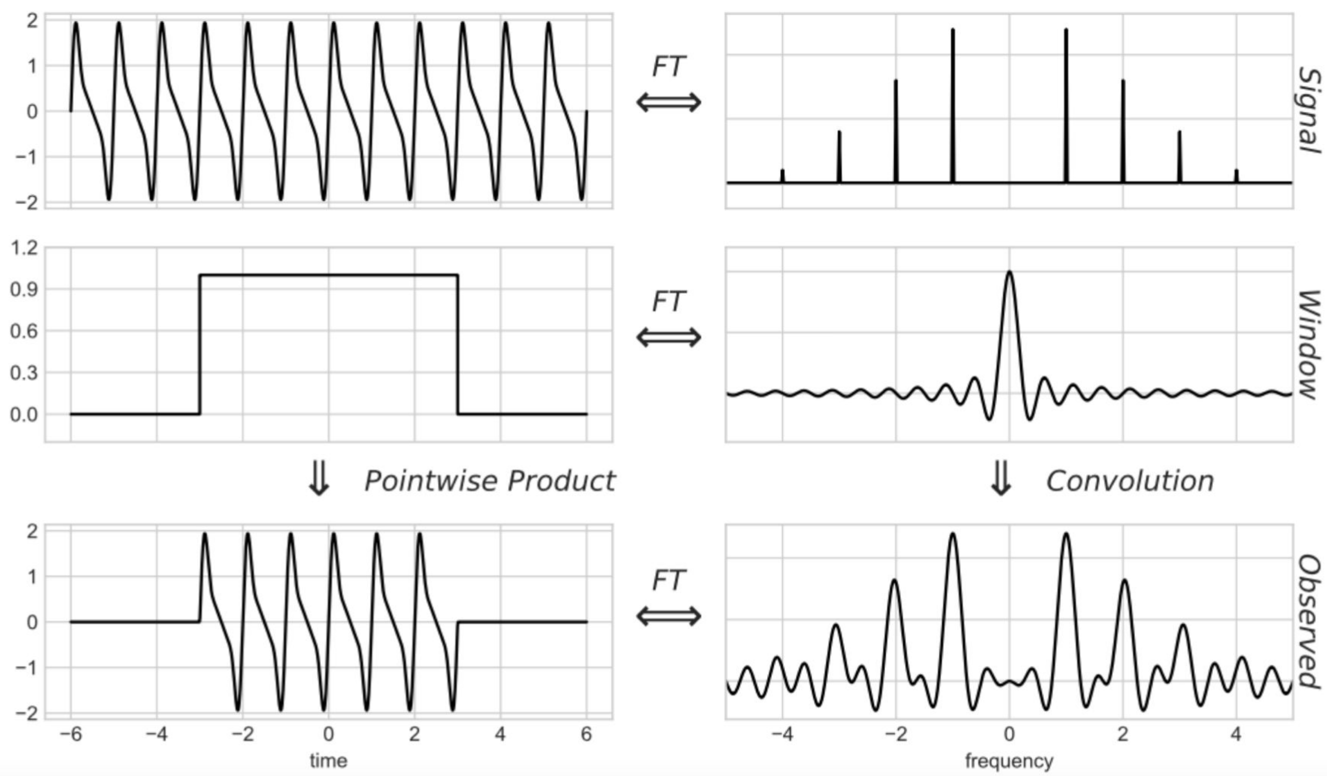
Convolutions with DFT: Window Functions

Example: Window functions in time and frequency domain.

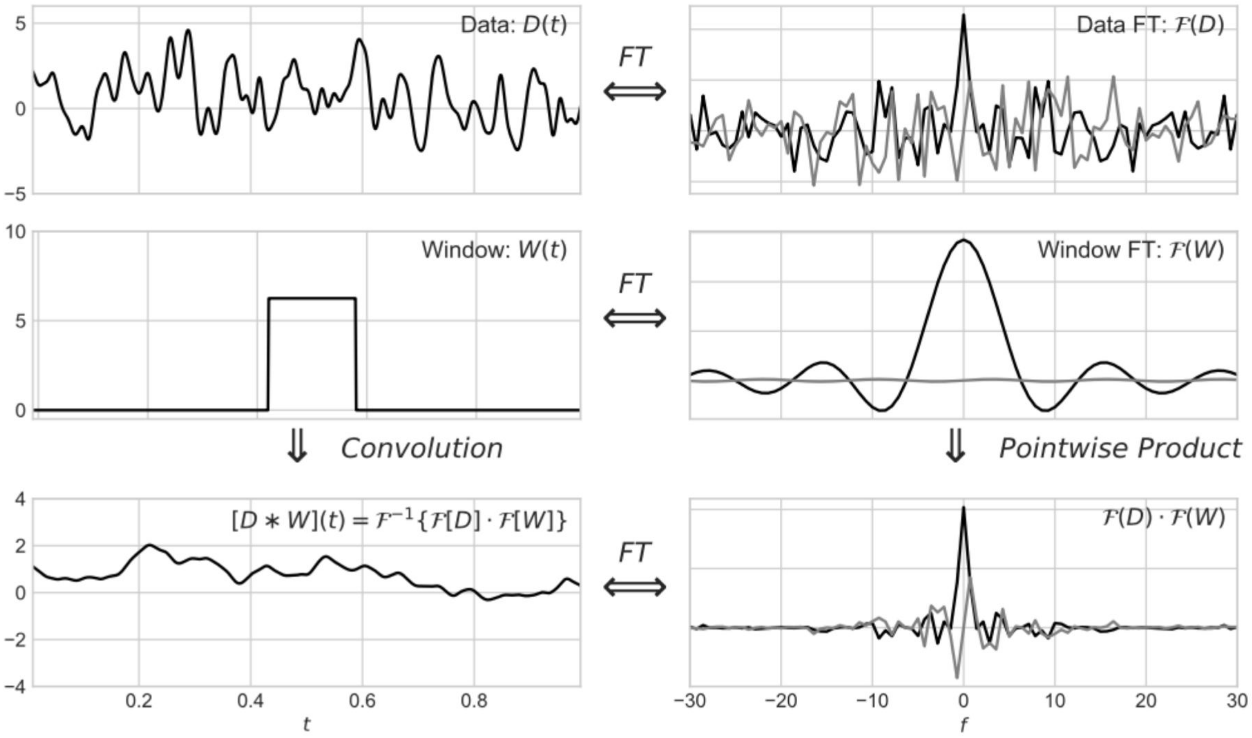




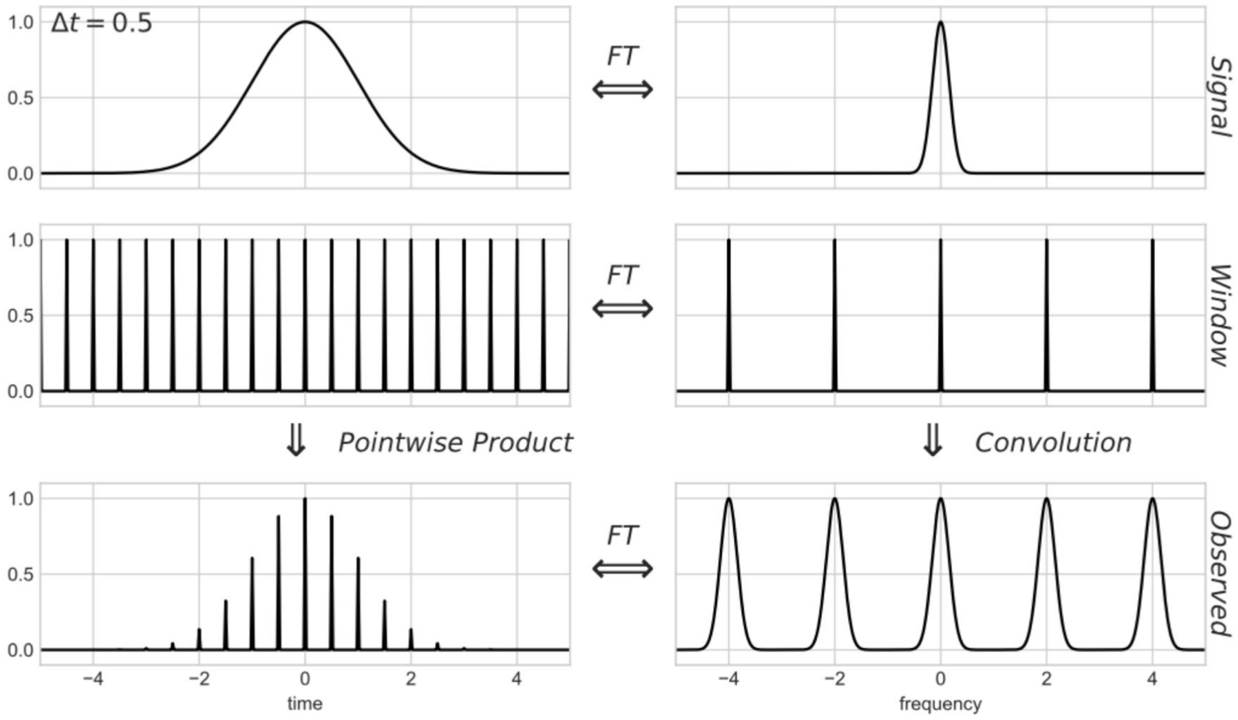
Convolutions with DFT: Window Functions



Convolutions with DFT: Window Functions



Convolutions with DFT: Window Functions



Application with DFT: Filtering

- Suppose we have noisy and convolved time stream of gaussian data and we wish to determine its best reconstruction in absence of noise and smearing
- In absence of noise we would deconvolve, but in presence of noise this would amplify noise

Although we will not work in detailed with filtering techniques, it is an essential component for modern data analysis.

Acknowledgement

The slides are partially developed or inspired by Professor Uros Seljak at UC Berkeley. For more information, please visit the github page.

A short story: When I was doing my undergraduate, I took his class (the exact same one, linked below).

https://phy151-ucb.github.io/seljak-phy151-fall-2018/#course-syllabus

Uros Seljak (Berkeley)