Life Sciences and
Facility Management

Institute of
Computational Life Sciences

Filtering II

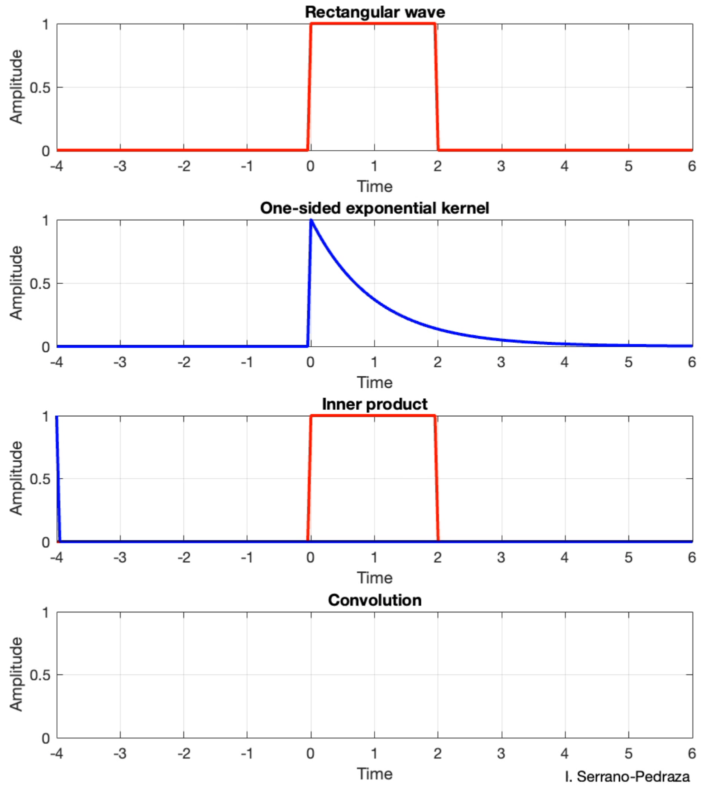
Image and Signal Processing

Norman Juchler

Cross-correlation vs. convolution

Comparison: Convolution

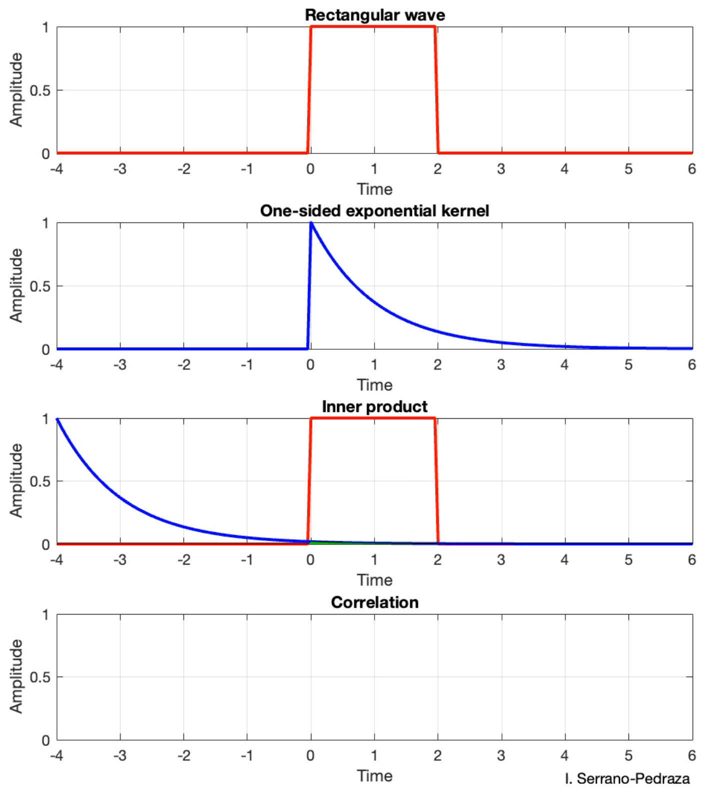
- Definitions:
  - Continuous-time: $$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau) \cdot g(t-\tau)\, d\tau$$
  - Discrete-time: $$(f * g)[n] = \sum_{k=-\infty}^{\infty} f[k] \cdot g[n-k]$$
- Interpretation:
  - Flip one signal, shift it, multiply, and sum or integrate.
  - Think of convolution as "blending" one signal with another.
  - Used to describe how a system modifies an input signal.
  - Why flip one signal? To reflect the directional flow of cause and effect in system theory.



- Applications:
  - Audio processing: Reverb, echo, ...
  - Image processing: Blurring, sharpening, edge detection, ...
  - System modeling: Computing output from input via impulse response

Comparison: Cross-correlation

- Definitions:
  - Continuous-time: $$(f \star g)(t) = \int_{-\infty}^{\infty} f(\tau) \cdot g(t+\tau)\, d\tau$$
  - Discrete-time: $$(f \star g)[n] = \sum_{k=-\infty}^{\infty} f[k] \cdot g[n+k]$$



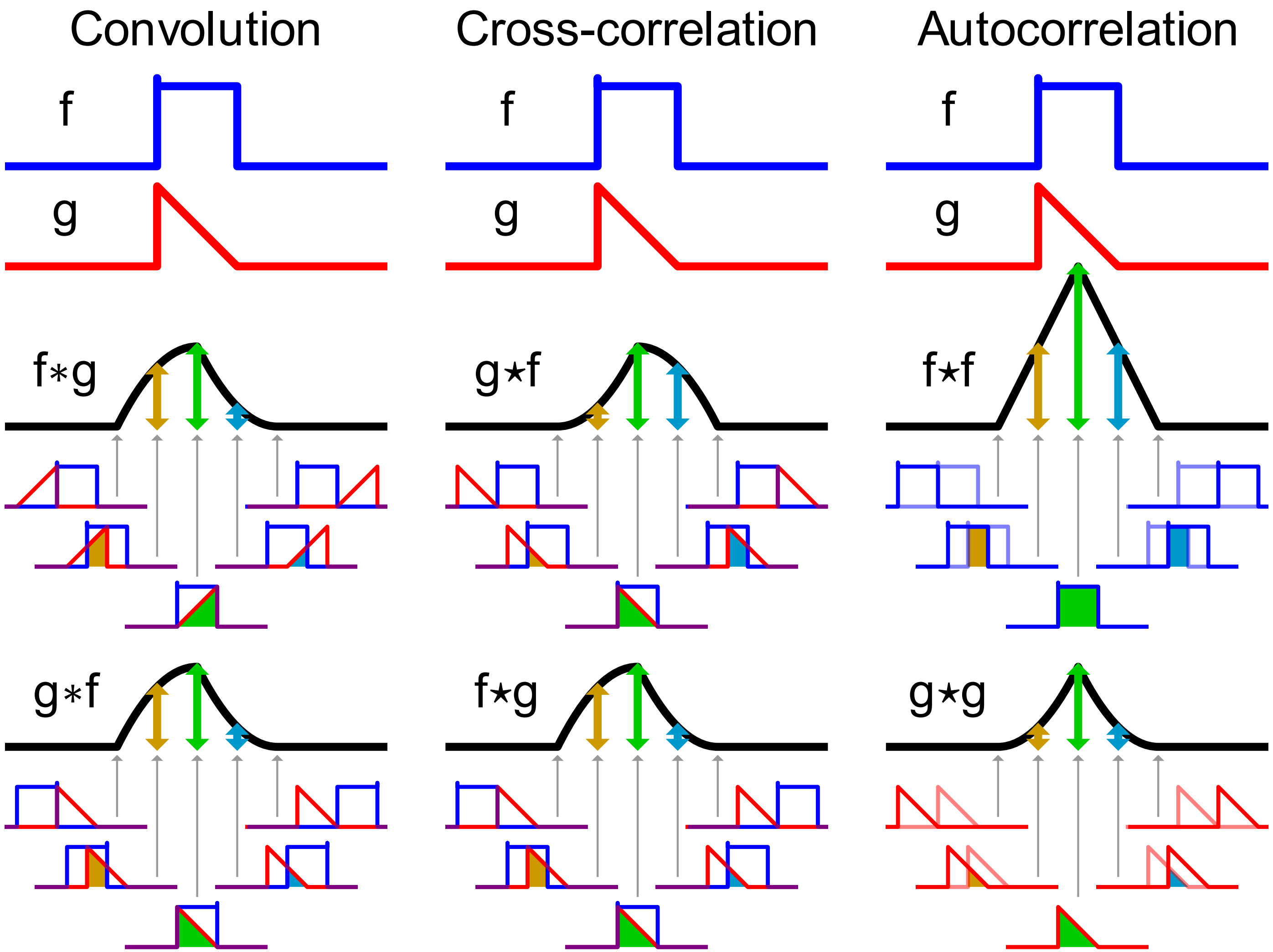
- Interpretation:
  - Measures similarity between two signals as one is shifted over the other.
  - Unlike convolution, no flipping is applied.
  - Peak in the output → best alignment between f(t) and g(t).
- Applications:
  - Signal synchronization: Aligning signals in time (e.g. audio/video sync)
  - Feature detection: Locating repeating events or bursts in time series
  - Time delay estimation: Finding the lag between transmitted and received signals

Comparison: Autocorrelation

- Definitions:
  - Continuous-time: $$R_{ff}(t) = \int_{-\infty}^{\infty} f(\tau) \cdot f(\tau + t)\, d\tau$$
  - Discrete-time: $$R_{ff}[n] = \sum_{k=-\infty}^{\infty} f[k] \cdot f[k+n]$$
- Interpretation:
  - It is the cross-correlation of a signal with itself.
  - Measures how much a signal resembles itself at different time shifts (lags).
  - High values → strong self-similarity at that lag.
  - Peaks in autocorrelation indicate periodicity or cyclic behavior.
- Applications:
  - Pitch detection (audio signals): Detect repeating waveforms like musical notes or voiced speech.
  - Pattern detection: Identify repeated behavior in time series data.

https://www.youtube.com/watch?v=K7cn9b7BNyk

Visual comparison



Visual comparison of **convolution, cross-correlation & autocorrelation**. For the operations involving function f, and assuming the height of f is 1.0, the value of the result at 5 different points is indicated by the shaded area below each point. Note that placing f★g next to g∗f, instead of next to f∗g facilitates comparing the 5 snapshots below the graphs. They are identical sets, except for the orientation of function f, as denoted by the little bump on its left-hand corner. But since f is symmetric, the orientation does not matter, which explains why g∗f and f★g are identical in this example. (Source: Wikipedia)

Source: Wikipedia / Wikimedia (Link)

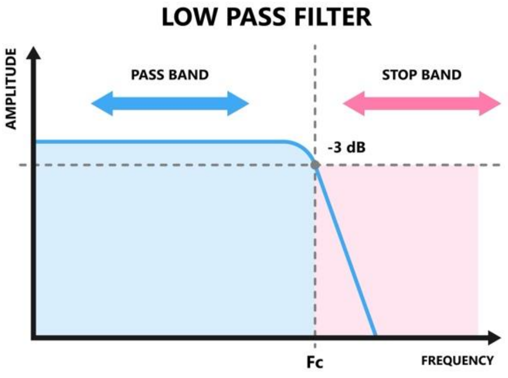
Further reading / watching

- Convolution vs. Correlation / WolfSound
- Geeks for Geeks: Convolution and Cross-Correlation in CNN
- Discrete Convolution: Digital Signals Theory, Brian McFee

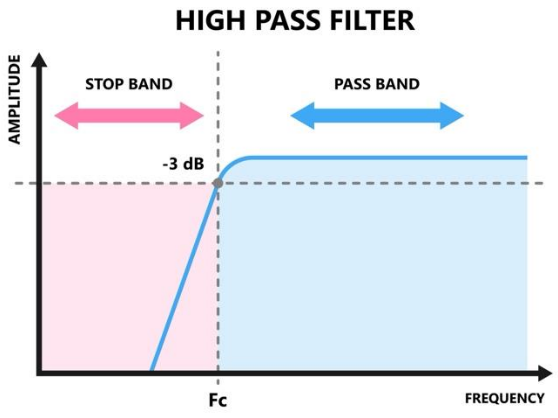
Signal filtering

Recapitulation

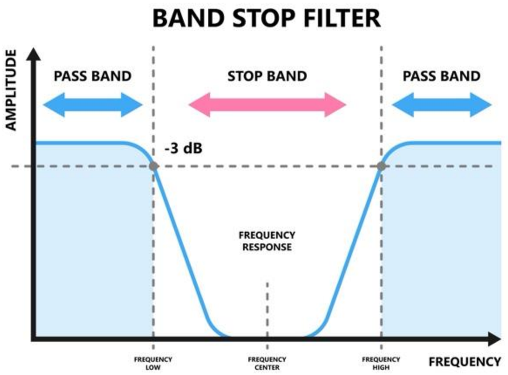
Overview: Types of filters



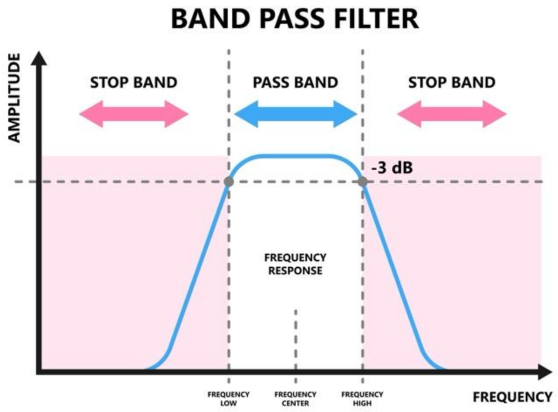
- Allows frequencies below a cutoff frequency to pass through while attenuating higher frequencies.
- Used to remove high-frequency noise, smooth signals, or extract low-frequency components.



- Allows frequencies above a cutoff frequency to pass through while attenuating lower frequencies.
- Used to remove low-frequency noise, extract high-frequency components, or emphasize high-frequency features.



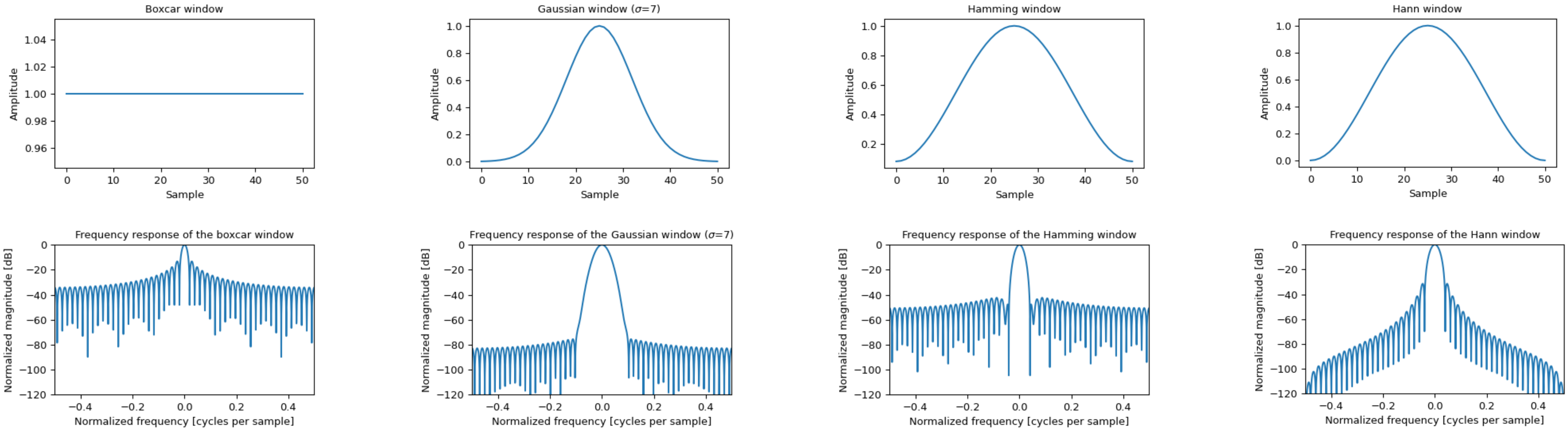
- Attenuates frequencies within a specific range, known as the stopband, while allowing frequencies outside this range to pass through.
- Used to remove unwanted interference or noise within a certain frequency range while preserving the rest of the signal.



- Allows a specific range of frequencies, known as the passband, to pass through while attenuating frequencies outside this range.
- Used to isolate or extract signals within a certain frequency band while rejecting others.

Filters have properties in the time- and frequency domain

- Comparison of different window filters



$$h_{box}[n] = 1$$

$$h_{gauss}[n] = e^{\frac{1}{2}\left(\frac{n}{\sigma}\right)^2}$$

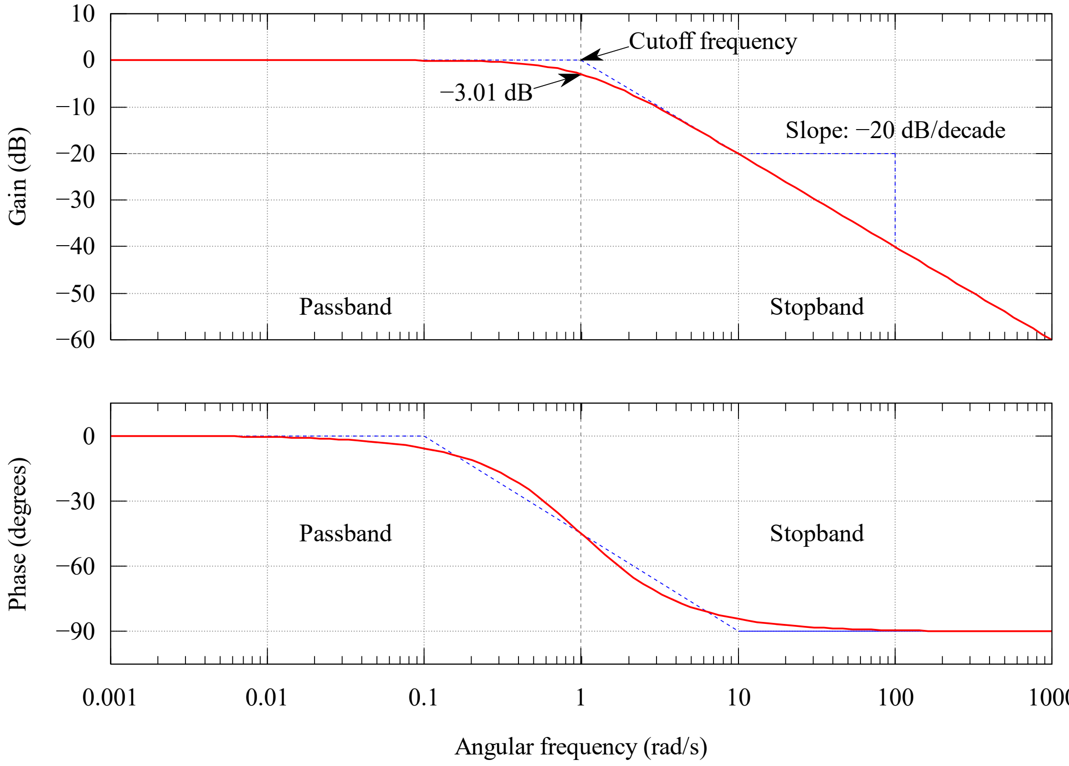
$$h_{lanc}[n] = sinc_{\pi}\left(\frac{2n}{M-1} - 1\right)$$

$$h_{hann}[n] = 0.5 - 0.5\cos\left(\frac{2\pi n}{M-1}\right)$$

Example: Butterworth filter

- A common type of filter used in signal processing
- Can be implemented as an analog or digital filter
- **Design goal**: Achieve a frequency response that is as flat as possible in the passband
- Also known as a “maximally flat magnitude filter”
- Typically implemented as an **Infinite Impulse Response** (IIR) filter
- Can be modeled as a **Linear Time-Invariant (LTI) system**: Example:

$$H(s) = \frac{V_o(s)}{V_i(s)} = \frac{1}{1 + 2s + 2s^2 + s^3}$$



Common filtering methods

- FIR Filtering (Finite Impulse Response)
  - Uses a filter with a finite-duration impulse response (i.e., it settles to zero in finite time).
  - The output depends only on current and past inputs (no feedback).
  - Provides linear phase response and stable filtering.
  - Typically requires more coefficients for sharp filtering, increasing computation.

$$y[n] = \sum_{k=0}^{N} b_k \cdot x[n-k]$$

- IIR Filtering (Infinite Impulse Response)
  - Impulse response theoretically extends infinitely due to internal feedback.
  - The output depends on both past inputs and past outputs.
  - Can achieve sharp filtering with fewer coefficients, making it more computationally efficient.
  - May be unstable if not carefully designed.

$$y[n] = \sum_{k=0}^{M} b_k \cdot x[n-k] - \sum_{k=1}^{N} a_k \cdot y[n-k]$$

Common filtering methods

- Special Filter Types
  - Butterworth filter: Maximally flat in the passband (smooth response, no ripples).
  - Chebyshev filters: Sharper cutoff but allow ripples in passband or stopband.
  - Windowed FIR filters: Designed using a window function (e.g., Hamming, Blackman).
  - Adaptive filters: Adjust their coefficients in real-time based on input (e.g., LMS algorithm) – used in noise cancellation, echo suppression.

Supplementary knowledge

Frequency bands

- A signal is called **band-limited** if its frequency components are limited within frequencies:

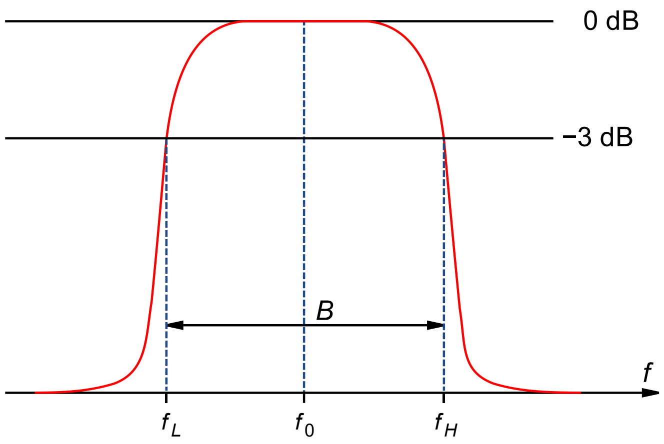
$$f_L \leq f \leq f_H$$

- The **bandwidth** of a signal is defined as the difference between the extreme frequencies:

$$B = \Delta f = f_H - f_L$$

- A strictly band-limited signal does not carry energy at frequencies outside the band limits

- In practice, a signal is considered band-limited if its energy outside of a frequency range is low enough to be considered negligible



Amplitude spectrum of a bandlimited signal. Source of illustration: Wikimedia

The decibel scale

- Decibels (dB) are often used to measure the power of a signal relative to some reference level.
- Formally, a **bel** is defined as a logarithmic ratio between two power quantities, a measured and reference value:
- The power carried by a sinusoidal signal is proportional to the square of the signal amplitude, so one can also write: (c is a proportionality constant)
- Decibels are commonly used in **power spectra**, which are like amplitude spectra, but use a decibel scale.
- Example:
  - Acoustic signals = pressure waves traveling through a medium
  - The standard reference amplitude used for acoustic signals $A_{ref} = p_0$ is 20 μPa, which corresponds approximately to the quietest sound that a person can hear.

$$1\mathrm{B} = \log_{10}\left(\frac{P_{meas}}{P_{ref}}\right)$$

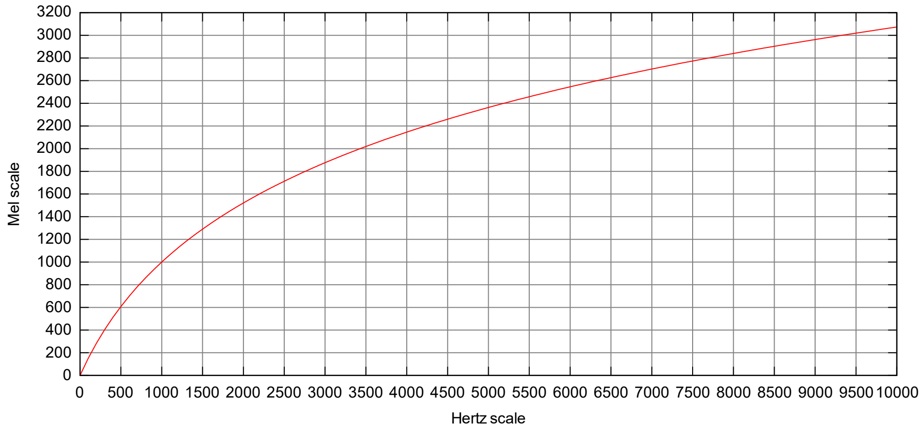
$$1\mathrm{dB} = 10\mathrm{B} = 10\log_{10}\left(\frac{P_{meas}}{P_{ref}}\right)$$

$$= 10\log_{10}\left(\frac{cA_{meas}^2}{cA_{ref}^2}\right)$$

$$= 20\log_{10}\left(\frac{A_{meas}}{A_{ref}}\right)$$

The Mel scale

- Main idea:
  - The Mel scale is a frequency scale designed to match human auditory perception.
  - Humans do not perceive frequencies linearly; we are more sensitive to lower frequencies than higher ones.



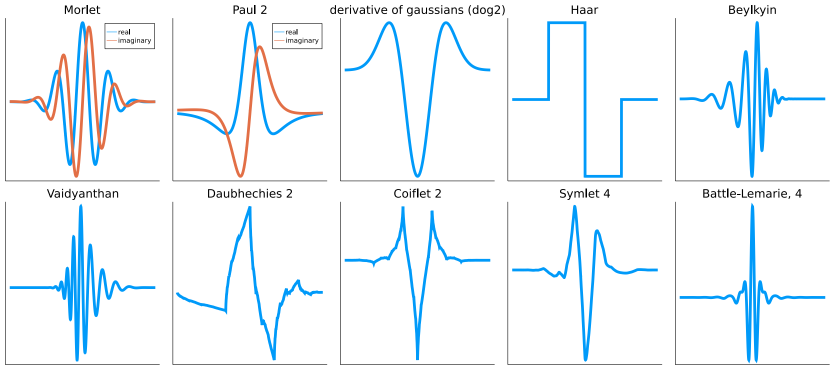
- The Mel scale:
  - Spaces frequency bins non-linearly: lower frequencies have higher resolution and higher frequencies have lower resolution.
  - Transformation from Hertz (Hz) to Mel

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right)$$

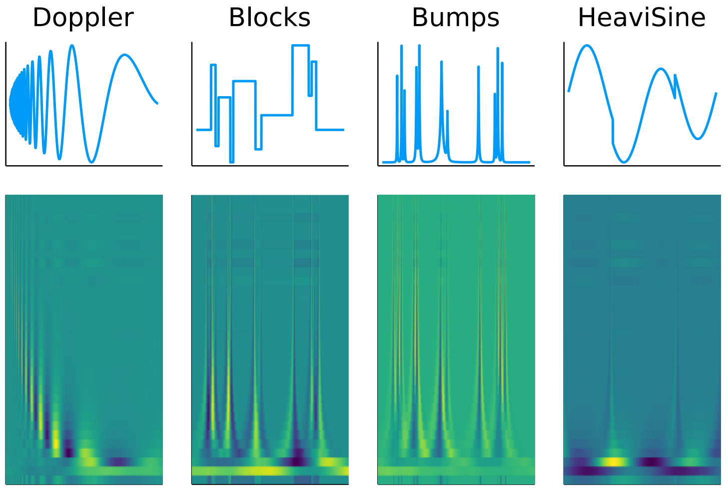
- Why Use Mel-Scaled Spectrograms?
  - Closer to human hearing → Useful in speech/audio processing.
  - More compact representation → Reduces unnecessary high-frequency details.
  - Foundation for MFCCs (Mel Frequency Cepstral Coefficients) → Common in speech recognition.

Wavelet decomposition

- Problems with Fourier transform:
  - Fourier representation: Functions are expressed as linear combinations of sinusoidal functions.
  - Sinusoids: They have infinite extent in the time domain – they have no distinct start or end.
  - Localization issue: The Fourier transform provides frequency information but lacks time localization
  - Limitations: It is not well-suited for capturing transient features or localized events in a signal.
- Solution: Wavelet transform
  - Introduce new basis functions that are localized in both time and frequency domains.
  - Decompose a signal using scaled and shifted versions of these basis functions.
  - Advantage: Wavelets provide good frequency resolution while maintaining good time localization.



Exemplary mother functions for Wavelets, from which the base functions are derived by scaling in x- and y-direction.



Wavelet decomposition result: Like an STFT representation, but achieved without requiring multiple transformations.

Compression

Compression

- Signal compression refers to the process of reducing the size (in bytes) of a signal while preserving essential information.
- Goal:
  - Eliminate redundancy or irrelevant information.
  - Trade-off: Compression ratio / information loss / method complexity
- Methods:
  - **Lossless compression**: Exact reconstruction is possible
    - Run-length encoding (RLE)
    - Huffmann coding
    - Lempel-Ziv-Welch (LZW)

Example: Run length encoding (RLE).

Input:

```
WWWWWWWWWWWWBWWWWWWWWWW
WWWBBBWWWWWWWWWWWWWWWWW
WWWWWWWWBWWWWWWWWWWWWWW
```

Output: Compressed data

```
12W1B12W3B24W1B13W
```

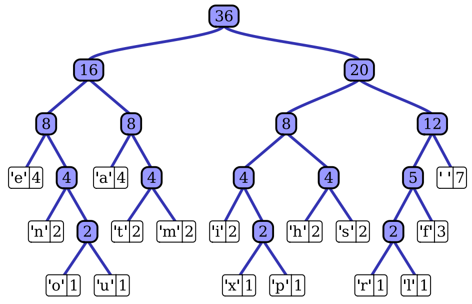
Compression

- Signal compression refers to the process of reducing the size (in bytes) of a signal while preserving essential information.
- Goal:
  - Eliminate redundancy or irrelevant information.
  - Trade-off: Compression ratio / information loss / method complexity
- Methods:
  - **Lossless compression**: Exact reconstruction is possible
    - Run-length encoding (RLE)
    - Huffmann coding
    - Lempel-Ziv-Welch (LZW)

Example: Huffmann coding.

Idea: Use a variable-length code for different tokens (or byte sequences), assigning shorter codes to the most frequent tokens.

Input: “this is an example of a huffman tree”. Tokens by frequency:



Output: Huffmann code table

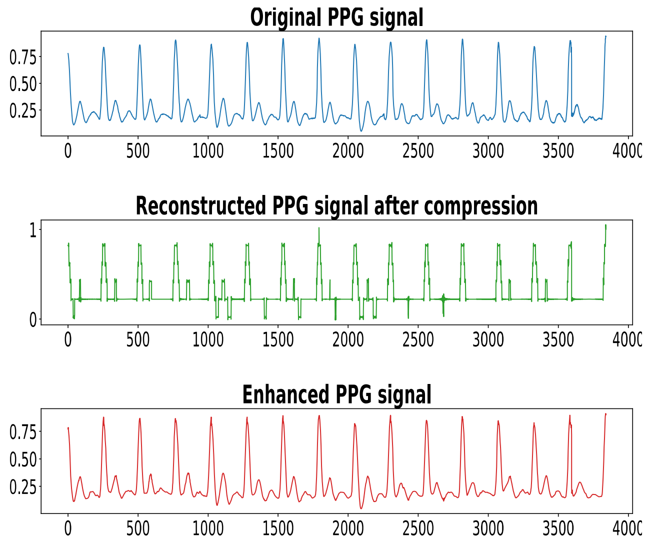
Char	Freq	Code	Char	Freq	Code
space	7	111	s	2	1011
a	4	010	t	2	0110
e	4	000	l	1	11001
f	3	1101	o	1	00110
h	2	1010	p	1	10011
i	2	1000	r	1	11000
m	2	0111	u	1	00111
n	2	0010	x	1	10010

Compression

- Signal compression refers to the process of reducing the size (in bytes) of a signal while preserving essential information.
- Goal:
  - Eliminate redundancy or irrelevant information.
  - Trade-off: Compression ratio / information loss / method complexity
- Methods:
  - **Lossless compression**: Exact reconstruction is possible
    - Run-length encoding (RLE)
    - Huffmann coding
    - Lempel-Ziv-Welch (LZW)
  - **Lossy Compression**: Higher compression ratios by discarding some signal information. Exact reconstruction is not possible.
    - Subsampling / down-sampling
    - Transform coding (e.g., discrete cosine transform, discrete wavelet tr.)
  - **Hybrid compression**: Lossless + lossy. Examples: MP3, JPEG...

Example: Lossy data compression with AI-based signal reconstruction. Reference: Link

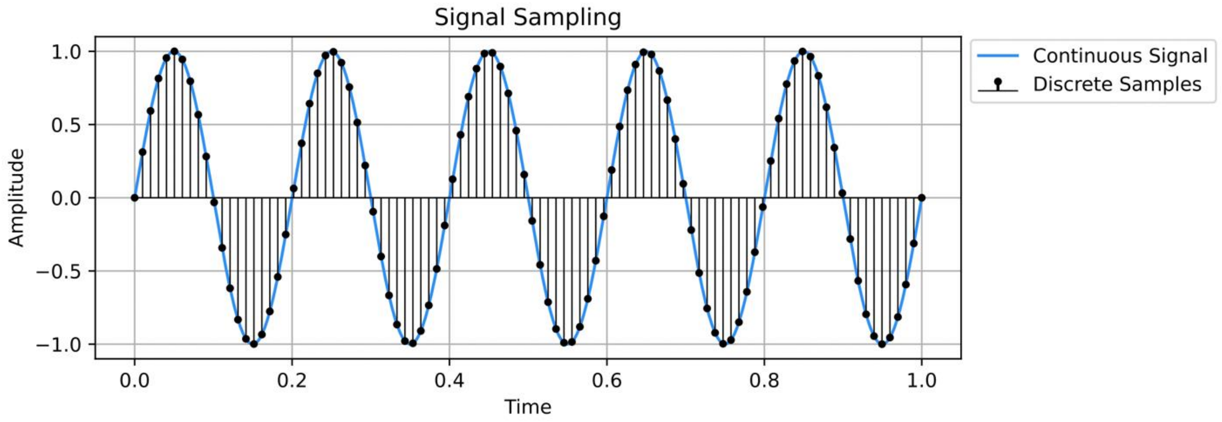
Data: PPG sensor data (top), after SZ compression (middle), and after enhanced reconstruction (bottom)



A look back

Signal processing in a nutshell

Signal representations



Discretization
Sampling, quantization

Continuous signal

Discrete signal

Interpolation

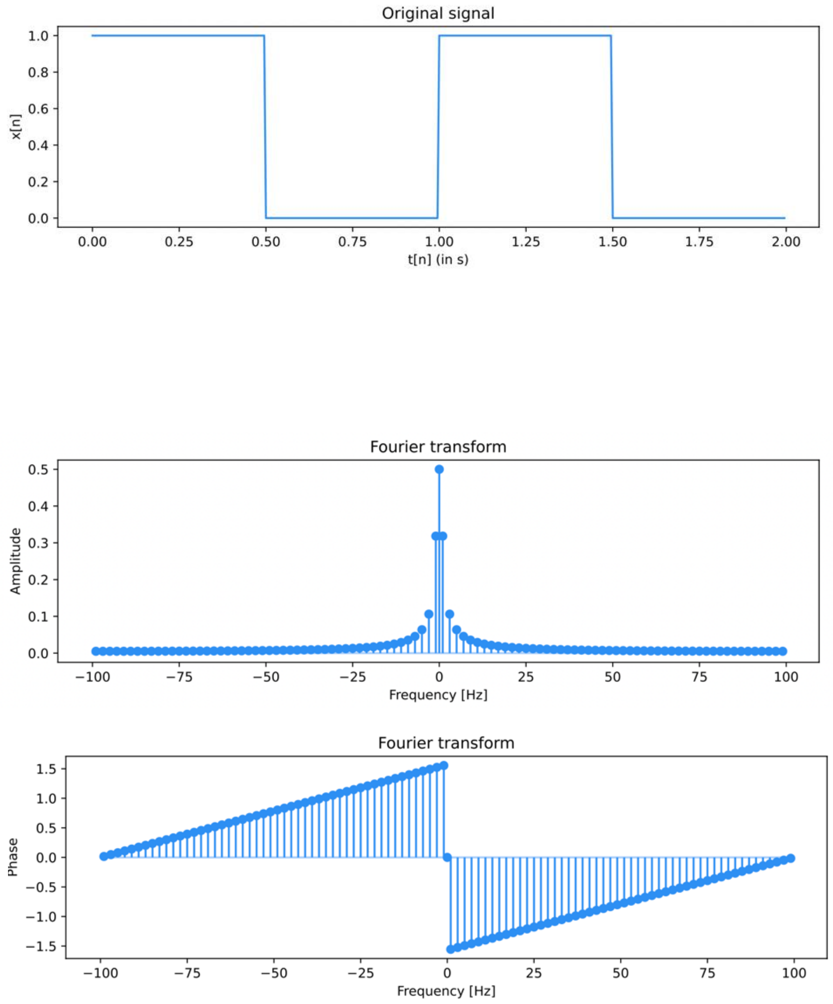
Signal representations

Fourier transformation
DFT/FFT

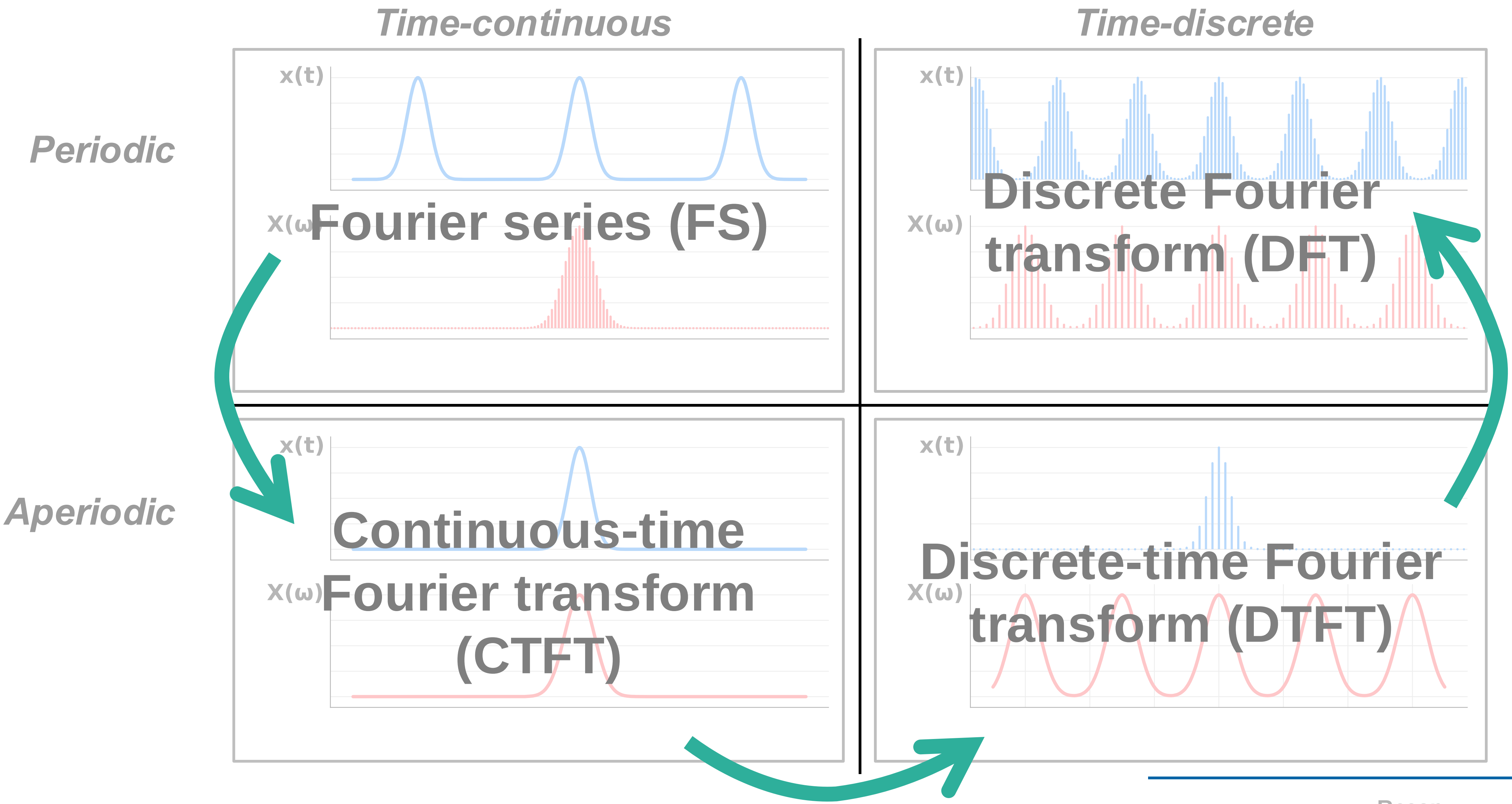
Time domain: $x(t)$

Frequency domain: $X(\omega)$

Inverse Fourier transformation
IDFT/IFFT



Fourier’s landscape



Signal representations

- Single- and multi-channel data
  - Mono audio (1 channel)
  - Stereo audio (2 channels)
  - Surround sound (5.1, 7.1, ambisonic)
  - Accelerometer and gyroscope data (3 or 6 channels)
  - EEG (16+ channels)
  - ...
  - (Note: Multi-channel data may require specialized synchronization mechanisms.)

Observations

- Frequency representation of time-series data is universal, and not limited to signals that are inherently periodic!

- This is useful for many applications, such as
  - Filtering
  - Feature extraction
  - Classification
  - ...

- With the convolution operation, we apply linear filters

What can we do with a signal?

- **Acquisition**: Capture signals using sensors or measurement systems.
- **Processing**: Filter, enhance, or manipulate signals
- (**Transmission**: Send signals over communication channels.)
- (**Storage**: Store signals in digital or analog formats for later use.)
- **Analysis**: Extract meaningful information or identify patterns within the signal.
- **Modeling**: Develop mathematical models to represent and understand signals.
- **Visualization**: Graphically display signals for easier interpretation and analysis.
- (**Control**: Use signals as inputs for feedback and control systems.)
- **Communication**: Encode information into signals for reliable transmission.
- (**Interpretation**: Understand signals within the context of specific applications.)