# Digital Speech and Audio Coding

**Lecture 2**

**Finishing Intro, Signals and Systems**

Mathew Magimai Doss and Petr Motlicek

Idiap Research Institute, Martigny

http://www.idiap.ch/

Ecole Polytechnique Fédérale de Lausanne, Switzerland

# General Information

- Course and Lab materials available on
  - Moodle:

- Contact: {mathew,motlicek}@idiap.ch

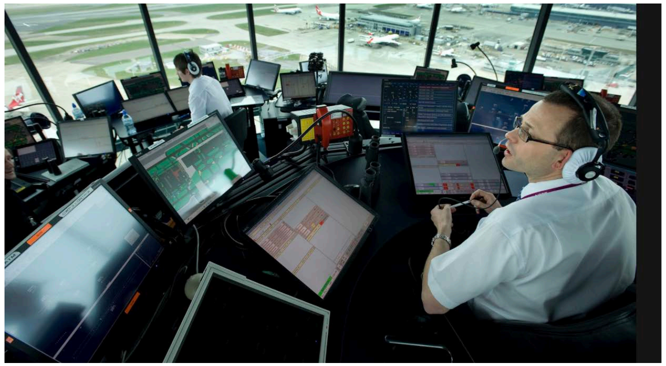
# Intro (repeat)

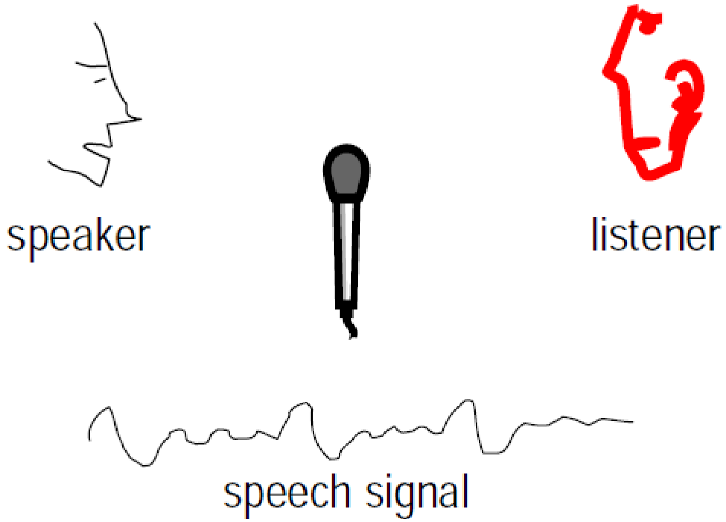
- Speech/Audio signal analysis (applied signal processing)
- Sound production and perception
- Speech coding:
  - LPC (2.4kbps), voiced/unvoiced synthesis ($\sim$ 1kbps)
  - Skype – G.723.1 ($\sim$ 6kbps)
- Audio coding:
  - MP3 – Lame ($\sim$ 32kbps)
  - MPEG4 – HE-AAC, AMR-WB+ ($\sim$ 32kbps)
- Quality assessment
- ...

All starts from speech . . .

# Speech Coding

- Source, receiver
- Purpose: We speak in order to be heard in order to be understood (Roman Jakobson)





# How humans communicate ...

**message**
**linguistic code (~ 50 b/s)**
motor control
*speech production*
**SPEECH SIGNAL (~50 kb/s)**
*speech perception*
cognitive processes
**linguistic code (~ 50 b/s)**
**message**

speech coding

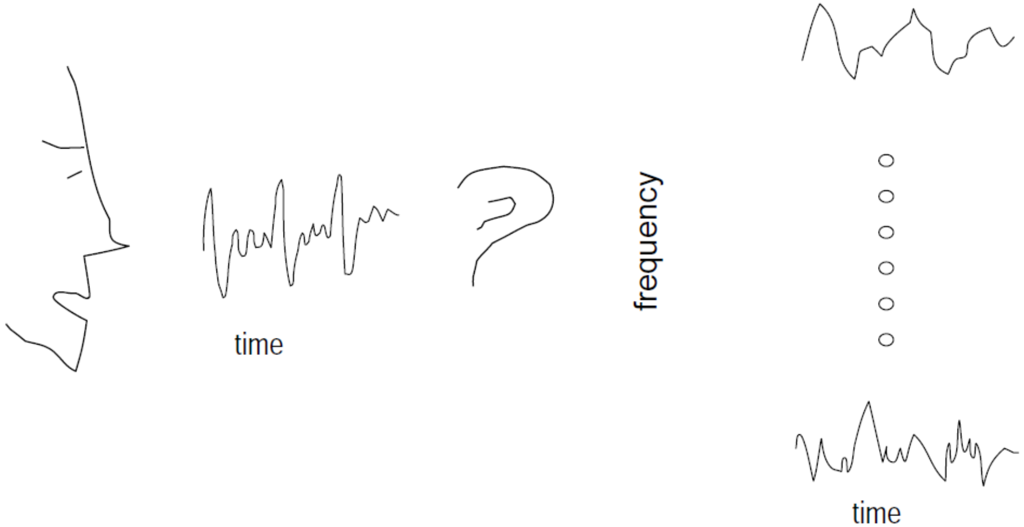
text-to-speech synthesis

automatic speech recognition

Communication:

- classical signal analysis techniques
- production based processing techniques
- perception based processing techniques
- goal oriented processing techniques

# Human hearing ...



# The goal of speech/audio signal processing

To describe the signal, so that it can be:

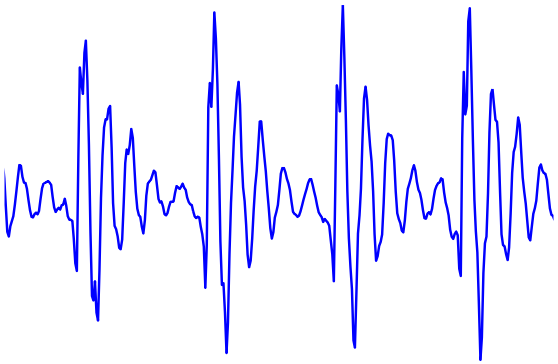
- stored and/or efficiently transmitted and reconstructed
- modified and reconstructed
- possibly used for useful information extraction

Signals and systems:

- relation
- focus on time $\leftrightarrow$ frequency

# Signals

- arbitrary physical values
- one or more independent axis (usually time), one dependent variable.
- Example: acoustic pressure generated by humans, gray scales of B/W video frames, ratio of EURo



# Mathematics view of signals (1 dependent variable)

According to character of $T$, we divide signals:

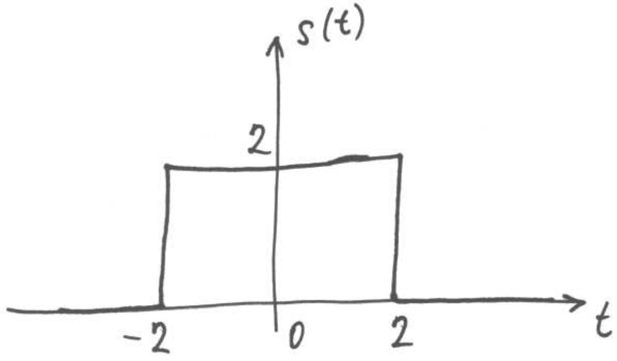
- **Continuous time signals**: $t \in \Re$, $s(t)$.
- **Discrete time signals**: $n \in Z$.

# Deterministic and non-deterministic signals

**Deterministic** signals can be described by equation:

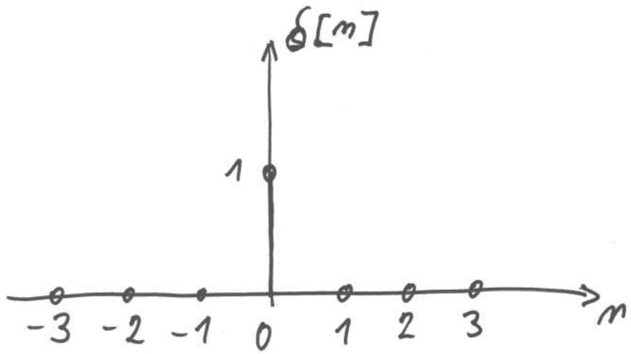
Example 1: square continuous time impulse:

$$x(t) = \begin{cases} 2 & \text{for} \quad -2 \leq t \leq 2 \\ 0 & \text{elsewhere} \end{cases}$$




Example 2: discrete unit impulse:

$$\delta[n] = \begin{cases} 1 & \text{for} \quad n = 0 \\ 0 & \text{elsewhere} \end{cases}$$




**Non-deterministic signals** cannot be described by equation.

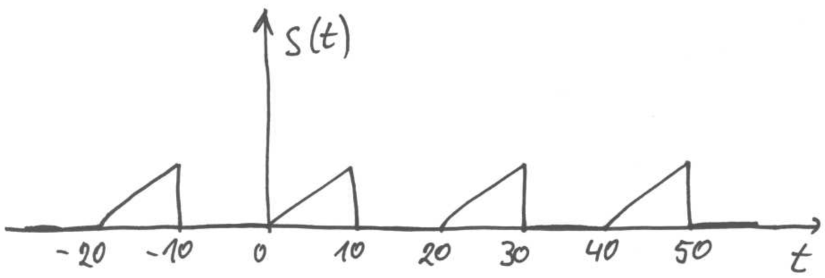
# Periodic and non-periodic signals

For **periodic** signals, we can find such $T$ or $N$:

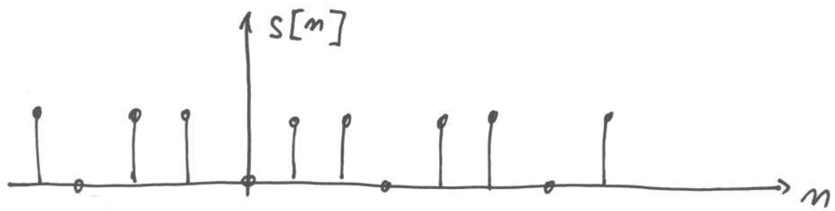
$$s(t+T) = s(t) \quad \text{continuous time}$$

$$s[n+N] = s(n) \quad \text{discrete time}$$

For continuous time:



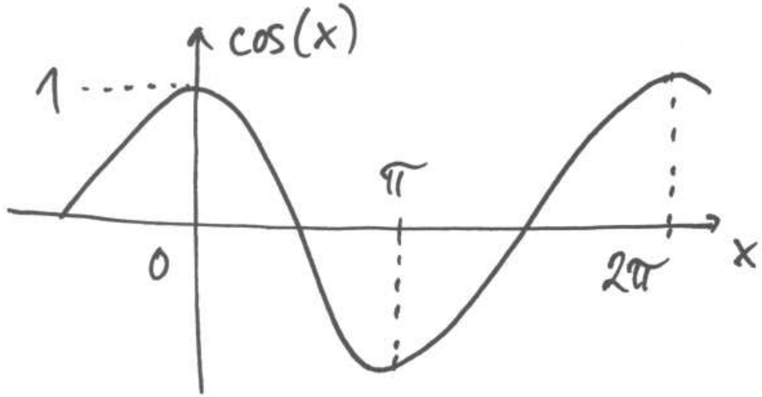

For discrete signals:

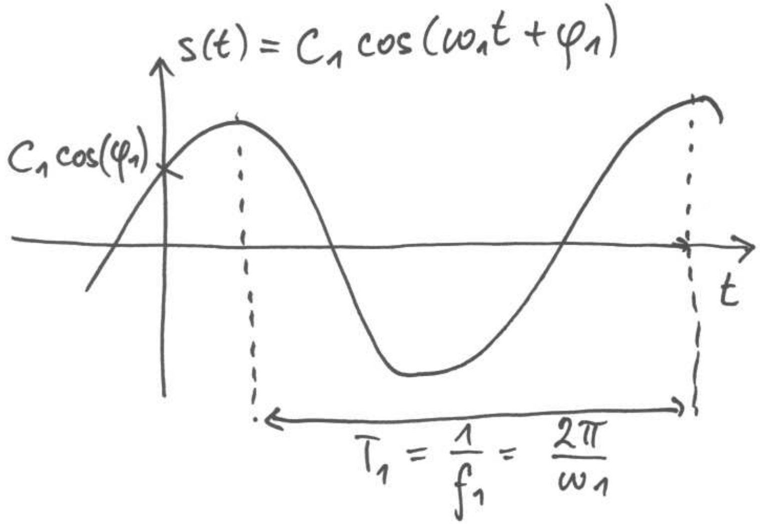


# Harmonic signals continuous time

A harmonic is a signal whose frequency is an integral (whole number) multiple of the frequency of the same reference signal. As part of the harmonic series, the term can also refer to the ratio of the frequency of such a signal to the frequency of the reference signal.

$$s(t) = C_1 \cos(\omega_1 t + \phi_1)$$



# Harmonic signals discrete time

$$s[n] = C_1 \cos(\omega_1 n + \phi_1)$$

- $C_1$ – amplitude,
- $\omega_1$ – frequency [rad], $\phi_1$ phase [rad].

PERIODICITY !!!

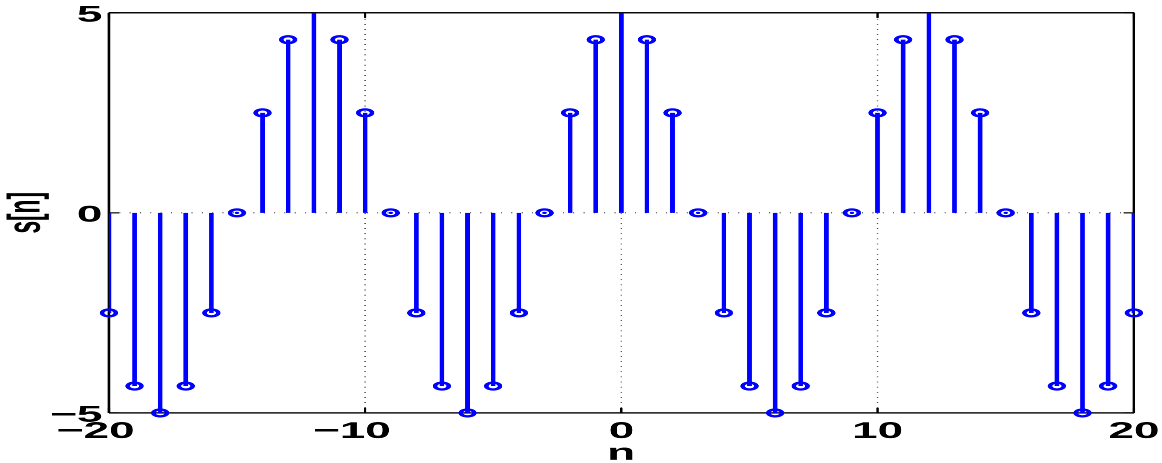
There is a small problem with fundamental period of harmonic sequence, that cannot be computed similarly as for continuous time using: $N_1 = \frac{2\pi}{\omega_1}$. It doesn't have to be integer number. Rule of periodicity must be fulfilled:

$$\cos\left[\omega_1(n + N_1)\right] = \cos \omega_1 n.$$

$$\omega_1(n + N_1) - \omega_1 n = \omega_1 N_1 = k2\pi,$$

# Harmonic signals discrete time

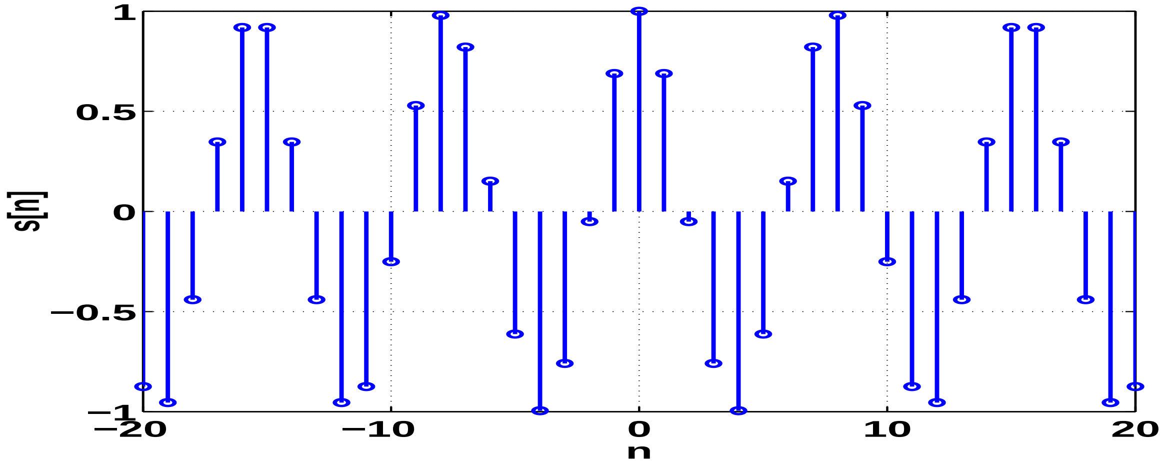
**Example 1**: $s[n] = 5\cos(2\pi n/12)$, $\omega_1 = \pi/6$.




$\frac{\pi}{6}N_1 = k2\pi$, solution: $k = 1$, $N_1 = 12$

**Example 2**: $s[n] = \cos(8\pi n/31)$, $\omega_1 = 8\pi/31$.

$\frac{8\pi}{31}N_1 = k2\pi$, $k = \frac{4}{31}N_1$, $31k = 4N_1$. Solution is: $k = 4$, $N_1 = 31$.



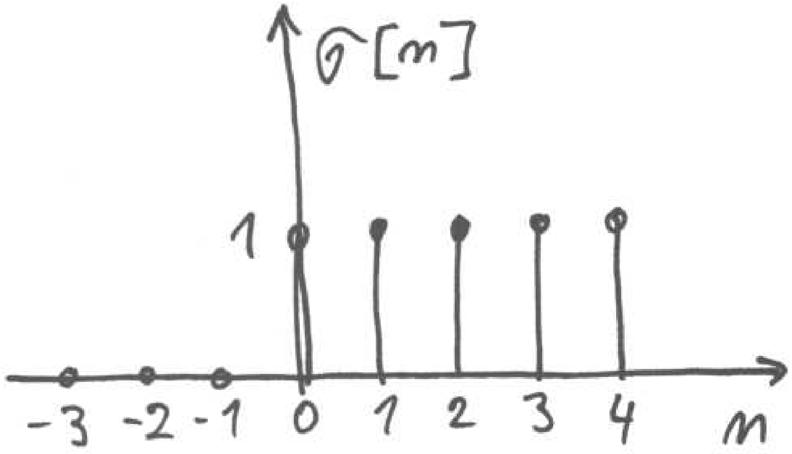

**Example 3**: $s[n] = \cos(n/6)$, $\omega_1 = 1/6$.

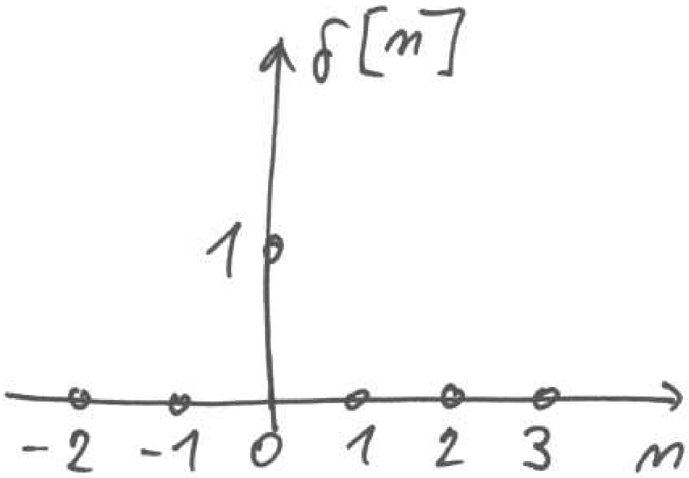
$\frac{1}{6}N_1 = k2\pi$, $N_1 = k12\pi$. No solution, non-periodic signals

# Some interesting signals discrete time

Unit step and impulse:

$$\sigma[n] = \begin{cases} 1 & \text{for} \quad n \geq 0 \\ 0 & \text{elsewhere} \end{cases} \qquad \delta[n] = \begin{cases} 1 & \text{for} \quad n = 0 \\ 0 & \text{elsewhere} \end{cases}$$

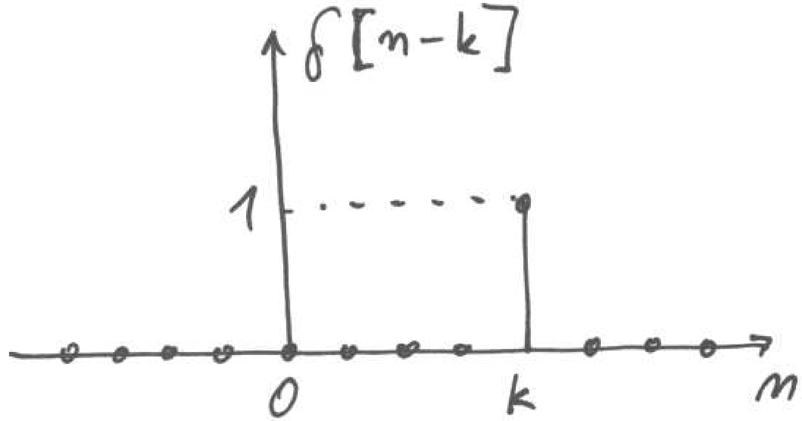



$$\delta[n] = \sigma[n] - \sigma[n-1]$$

# Some interesting signals discrete time

Shifted unit impulse: $\delta[n-k]$:



# Some interesting signals continuous time

Unit step:

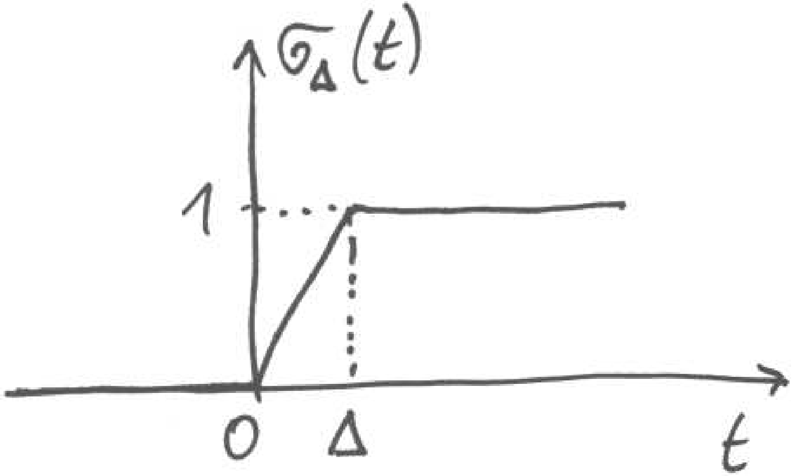
$$\sigma(t) = \begin{cases} 1 & \text{for} \quad t \geq 0 \\ 0 & \text{elsewhere} \end{cases}$$

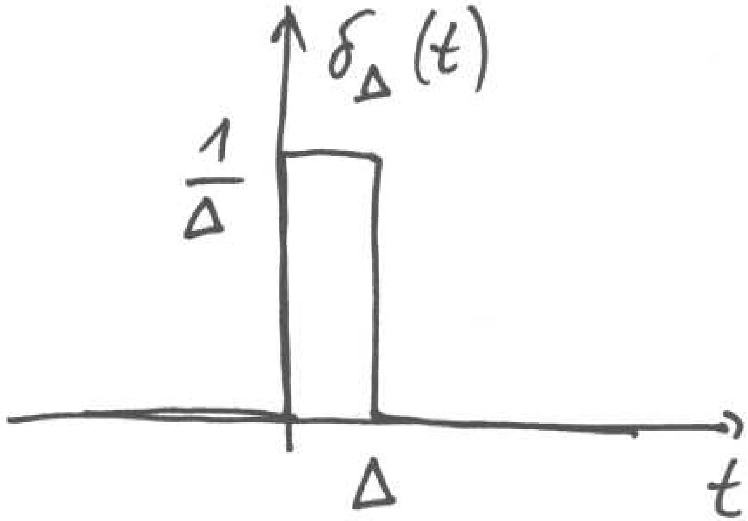
Unit impuls:

$$\delta(t) = \frac{d\sigma(t)}{dt}$$

$$\delta_\Delta(t) = \frac{d\sigma_\Delta(t)}{dt}$$

# Some interesting signals continuous time



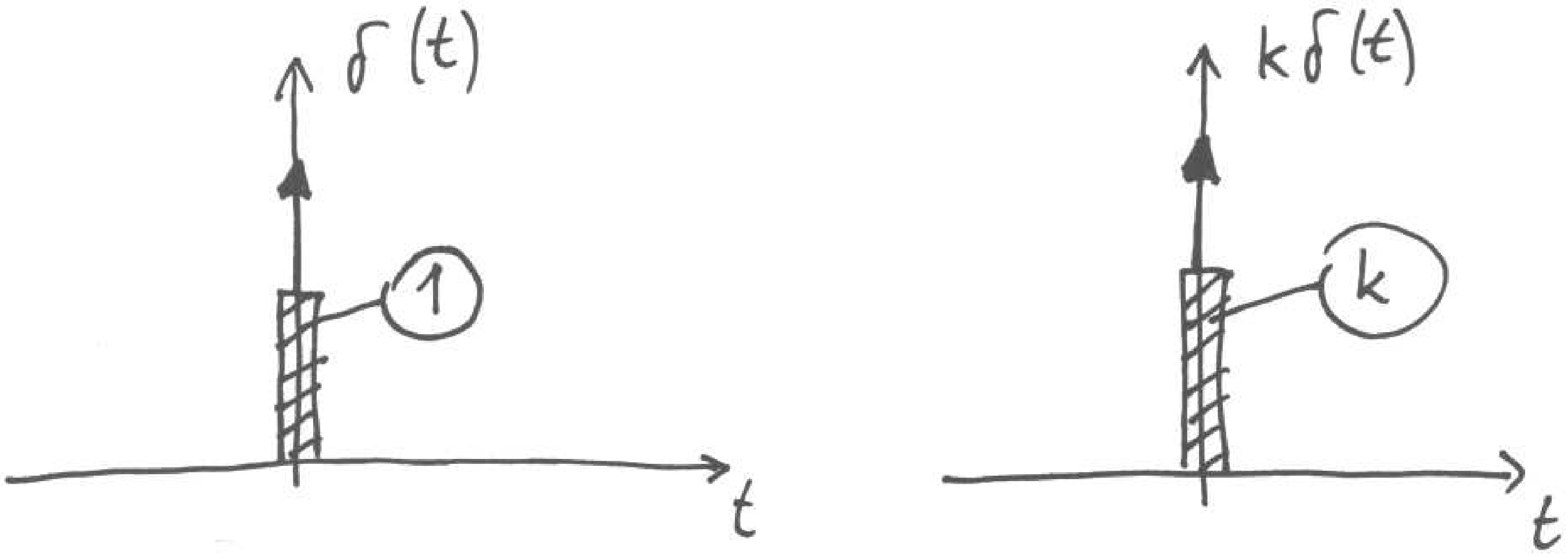

$$\delta(t) = \lim_{\Delta \to 0} \delta_\Delta(t)$$

$$\int_{-\infty}^{+\infty} \delta(t) dt = 1$$

# Some interesting signals continuous time

The Dirac delta can be loosely thought of as a function on the real line which is zero everywhere except at the origin, where it is infinite,

$$\delta(x) = \begin{cases} +\infty, & x = 0 \\ 0, & x \neq 0 \end{cases}$$

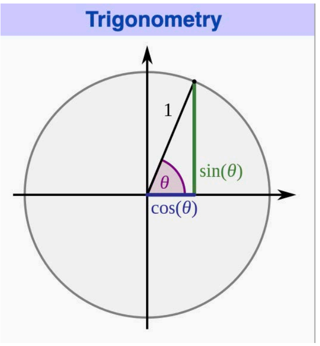


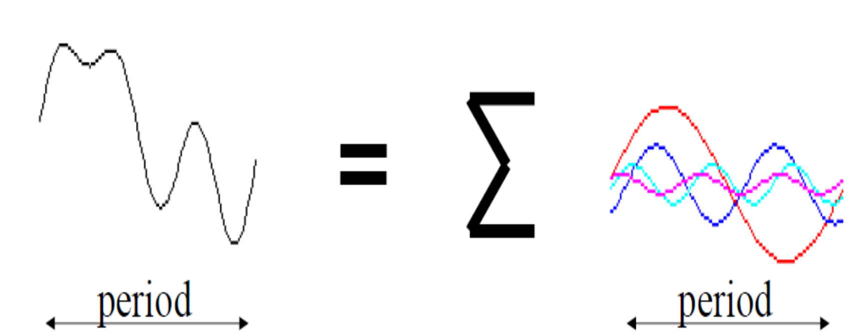

and which is also constrained to satisfy the identity

$$\int_{-\infty}^{\infty} \delta(x)\,dx = 1.$$

# Fourier analysis

...study of the way general functions may be represented or approximated by sums of simpler trigonometric functions. Fourier analysis grew from the study of Fourier series, and is named after Joseph Fourier, who showed that representing a function as a sum of trigonometric functions greatly simplifies the study of heat transfer.




Trigonometric functions = real functions which relate an angle of a right-angled triangle to ratios of two side lengths.

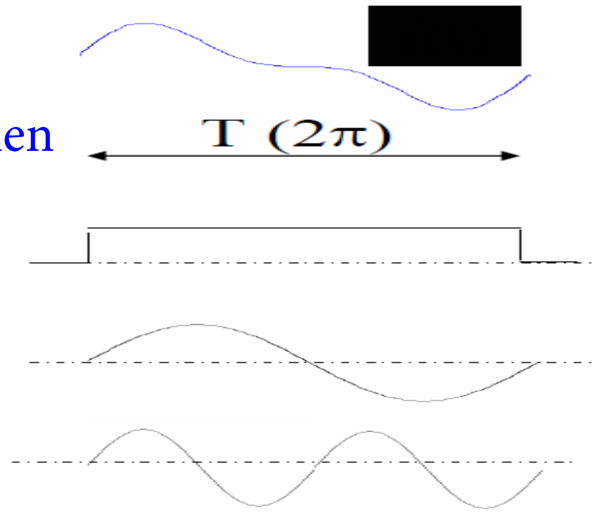
Describe complex function as a weighted sum of simpler functions:

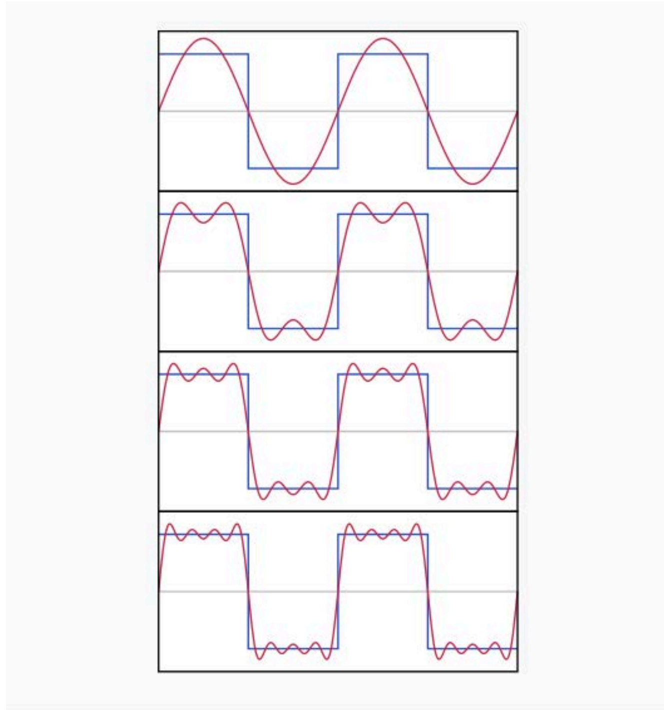
- simpler functions are known - sines and cosines and need to be orthogonal
- weights can be found

# How to get weights - Fourier series

A Fourier series = a sum that represents a periodic function as a sum of sine and cosine waves. The frequency of each wave in the sum, or harmonic, is an integer multiple of the periodic function's fundamental frequency

The functions f and g are orthogonal when this integral is zero, whenever f≠g



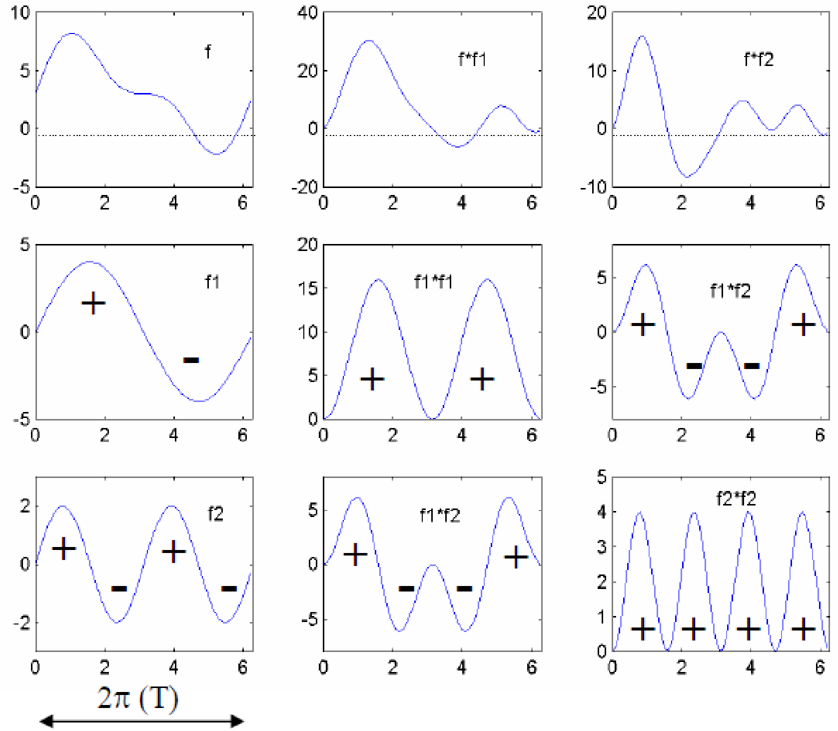



$\int_0^T f(t)\ dt = a_0$

$a_0 = \frac{1}{T} \int_0^T f(t)\ dt$

How to get other weights?

sin(nx) and sin(mx) are orthogonal on the interval $x \in (-\pi, \pi)$ when $m \neq n$ and n and m are positive integers.



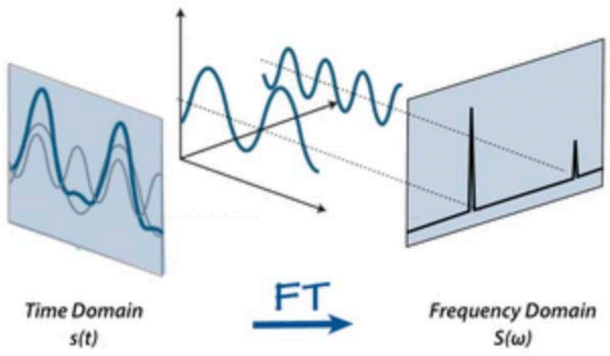

$$f(t) = a_0 + f_1(t) + f_2(t) = a_0 + a_1 \cdot sin\omega t + a_2 \cdot sin2\omega t$$

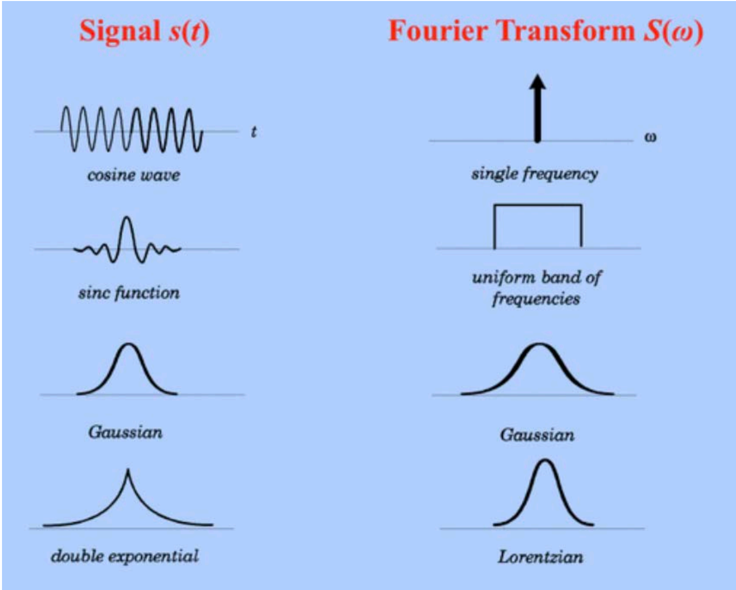
$$\int_0^T f(t) \cdot sin\omega t \; dt = a_1$$

$$\int_0^T f(t) \cdot sin2\omega t \; dt = a_2$$

$$\int_0^T sin^2 \omega t \; dt = \frac{T}{2}$$

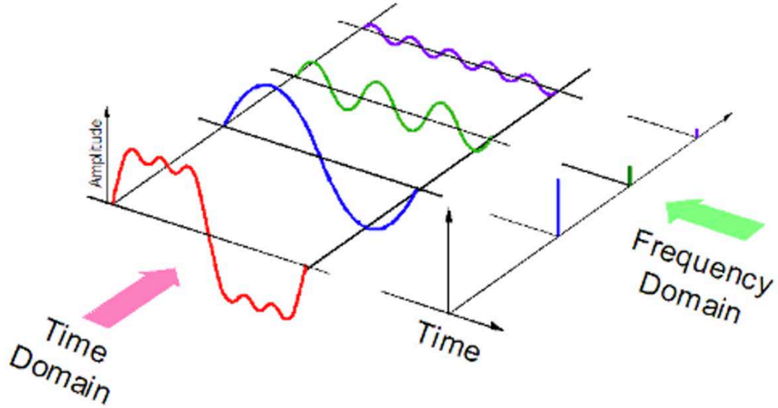
# Fourier Transform




$$X(j\omega) = \int_{-\infty}^{+\infty} x(t) e^{-j\omega t}\, dt,$$

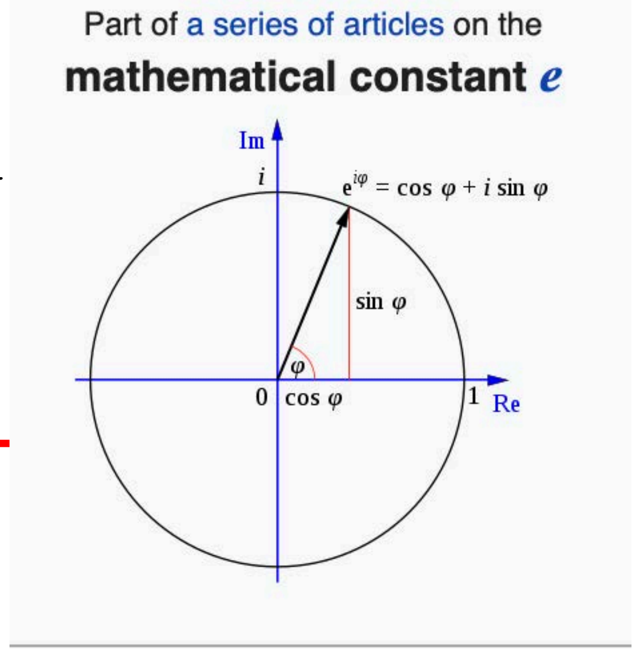
$X(j\omega)$ will be called **Fourier projection/image** of signal $x(t)$.

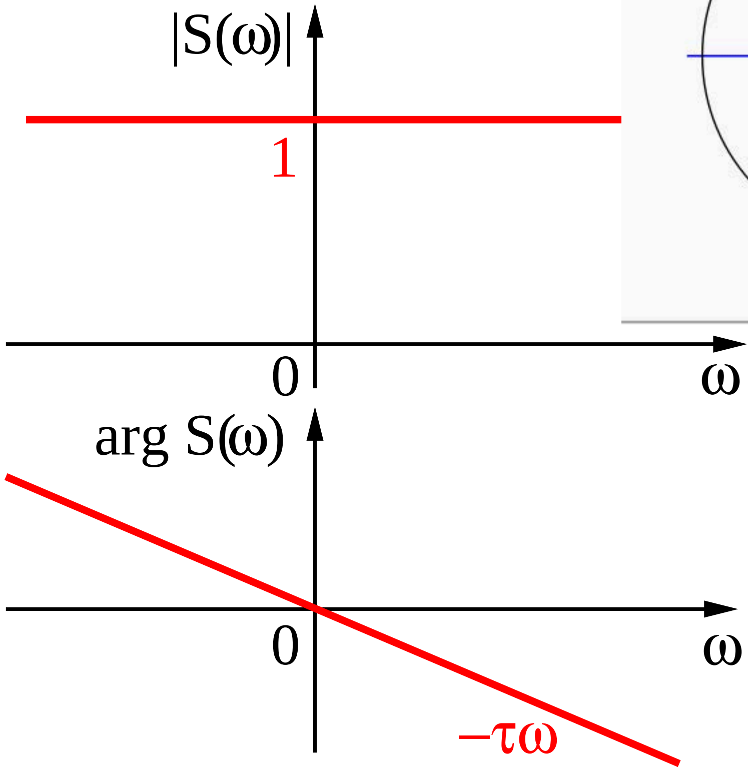


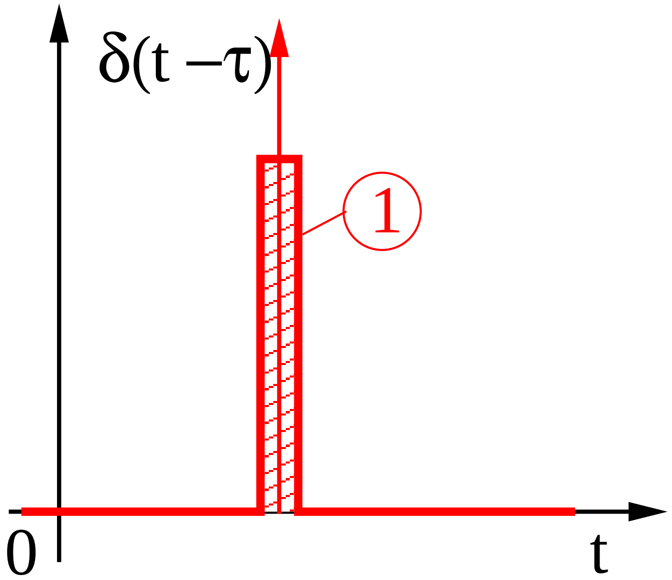
# Fourier Transform

**Spectral function of unity impulse:**

$$X(j\omega) = \int_{-\infty}^{+\infty} \delta(t-\tau)e^{-j\omega t}dt = e^{-j\omega\tau}$$



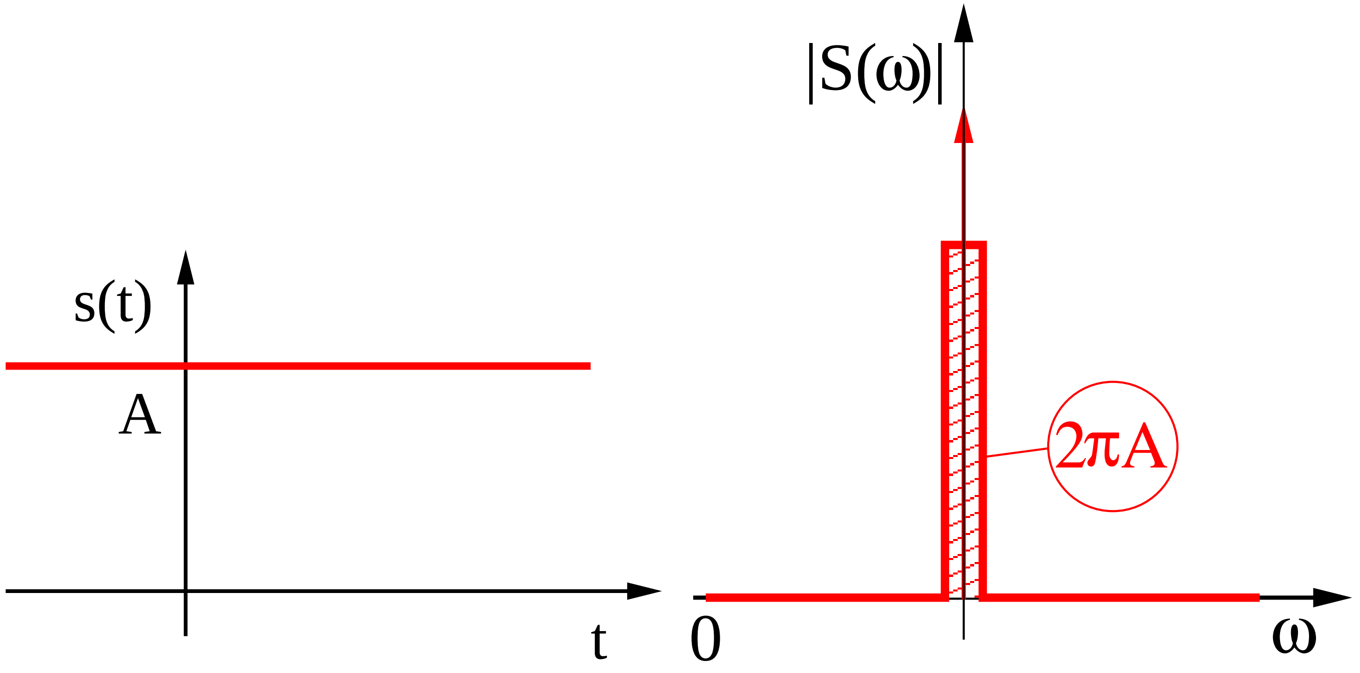



**Direct signal:**

$$X(j\omega) = 2\pi A\delta(\omega)$$

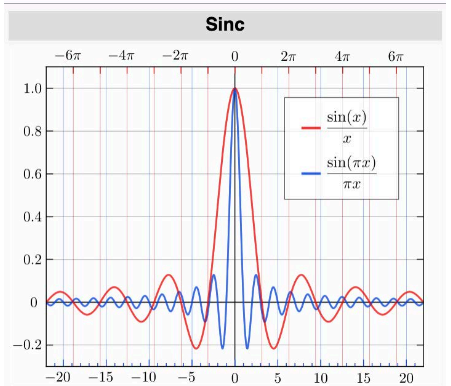
$$x(t) = \frac{1}{2\pi} 2\pi A \int_{-\infty}^{+\infty} \delta(\omega) e^{j\omega t} d\omega = A.$$



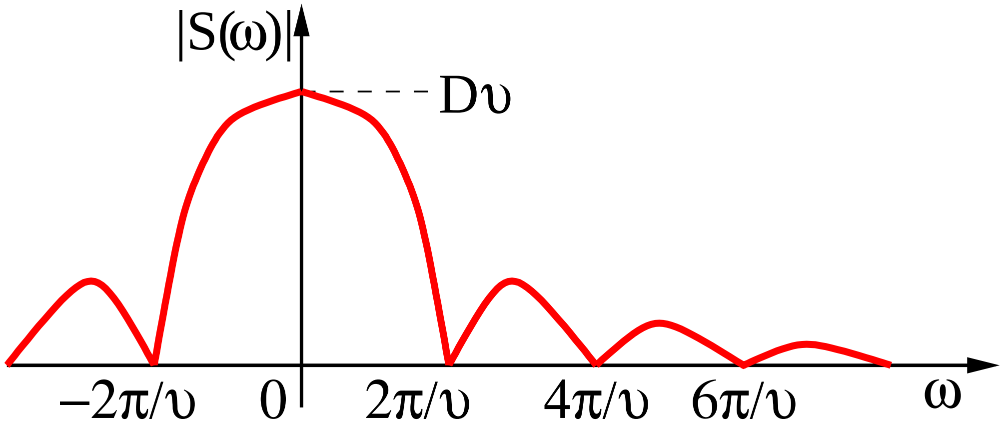
# Inverse projection of square spectral

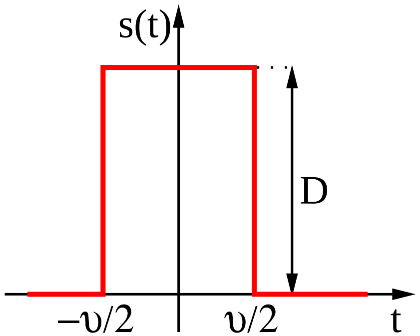
**Square impuls:** We know: $\int_{-b}^{b} e^{\pm jxy} dy = 2b \,\mathrm{sinc}(bx)$. Let's have $b = \frac{\vartheta}{2}$, $\quad y = t, \quad x = \omega$, we obtain:

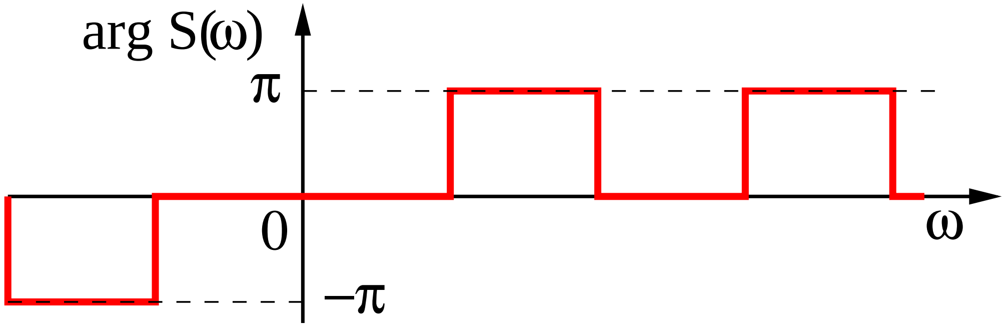
$$X(j\omega) = D \int_{-\frac{\vartheta}{2}}^{+\frac{\vartheta}{2}} e^{-j\omega t} dt = D2\frac{\vartheta}{2}\mathrm{sinc}\left(\frac{\vartheta}{2}\omega\right) = D\vartheta\mathrm{sinc}\left(\frac{\vartheta}{2}\omega\right)$$



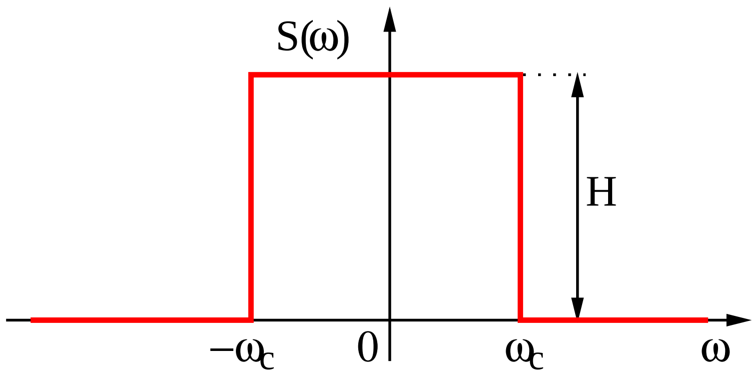
Part of the normalized sinc (blue) and unnormalized sinc function (red) shown on the same scale

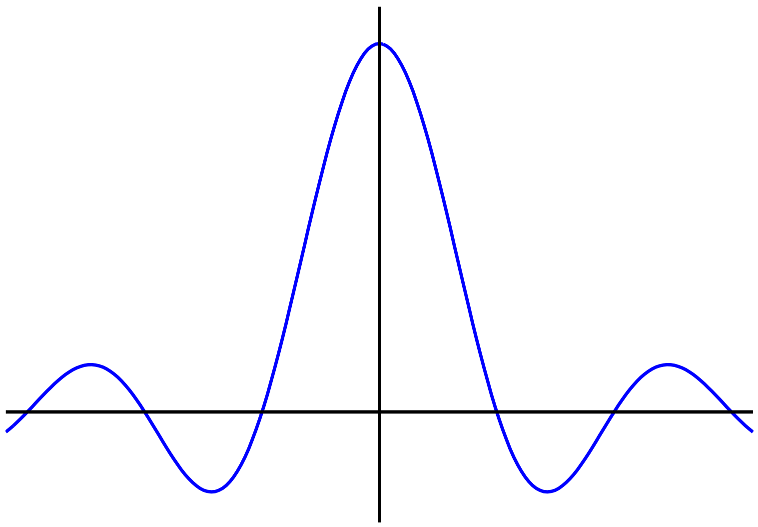


# Inverse projection of square spectral



$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(j\omega)e^{+j\omega t}d\omega = \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} He^{+j\omega t}d\omega = \frac{H}{2\pi}\int_{-\omega_c}^{\omega_c} e^{+j\omega t}d\omega =$$

$$= \frac{H}{2\pi}2\omega_c \mathrm{sinc}(\omega_c t) = \frac{H\omega_c}{\pi}\mathrm{sinc}(\omega_c t)$$

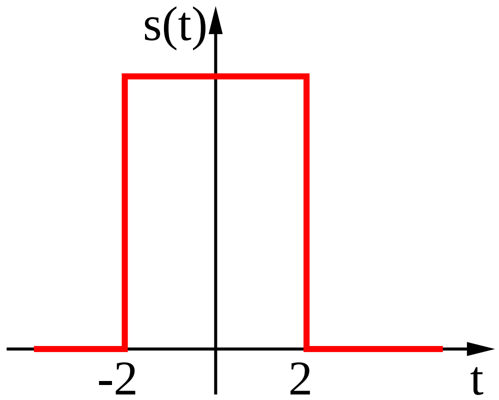
# Hints about spectra of non periodic signals

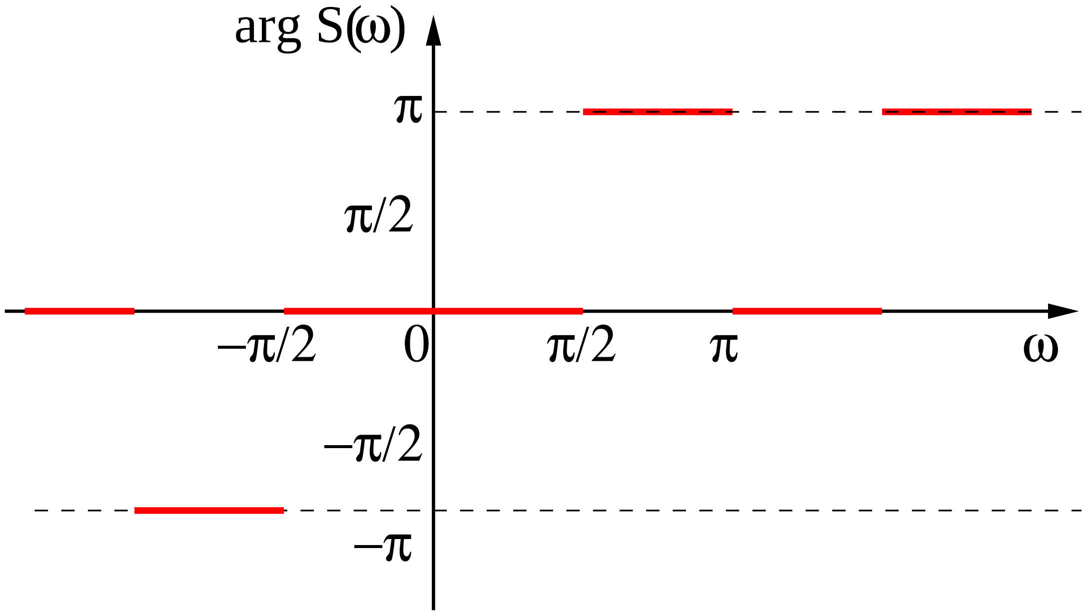
| | $x(t)$ | $X(j\omega)$ |
|---|---|---|
| linearity | $ax_a(t) + bx_b(t)$ | $aX_a(j\omega) + bX_b(j\omega)$ |
| shift in time | $x(t-\tau)$ | $X(j\omega)e^{-j\omega\tau}$ |
| change of scale | $s(mt) \quad m > 0$ | $\frac{1}{m}X\left(\frac{\omega}{m}\right)$ |
| convolution | $x_1(t) \star x_2(t) = \int_{-\infty}^{\infty} x_1(\tau)x_2(t-\tau)d\tau$ | $X_1(j\omega)X_2(j\omega)$ |

# Time-shift

just argument/phase of spectral function will be changed: $-\omega\tau$.

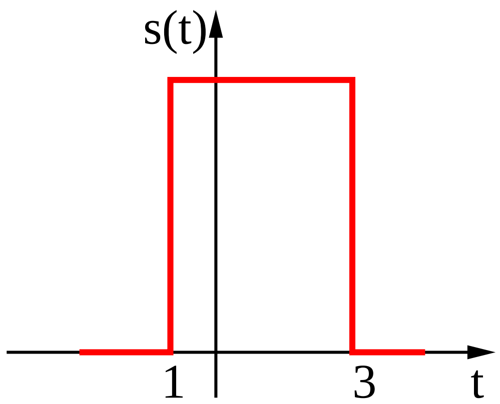
Example:

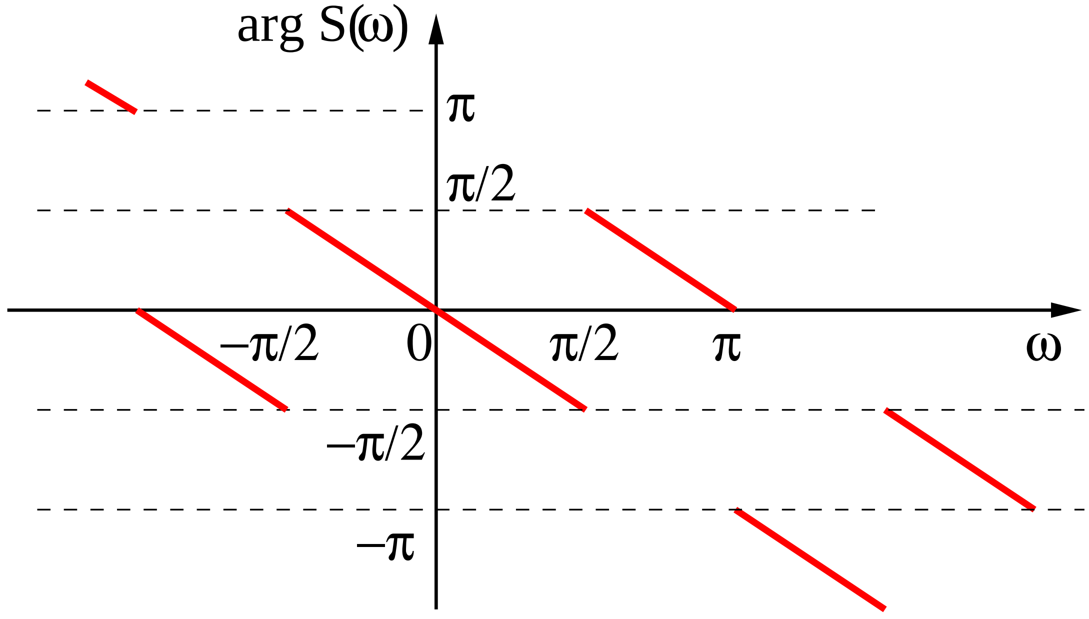


# Time-shift

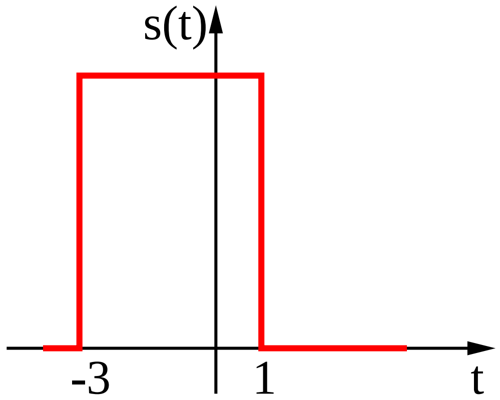
1 s delay: $X(j\omega)$ multiplied by function $e^{-j\omega}$, thus $\omega$ will be subtracted from argument:

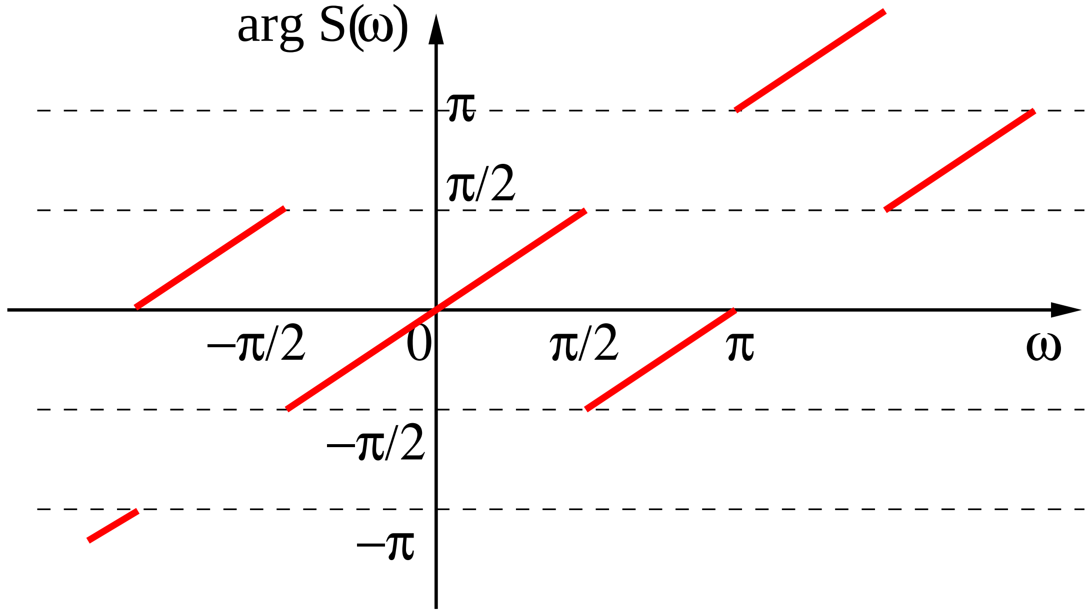


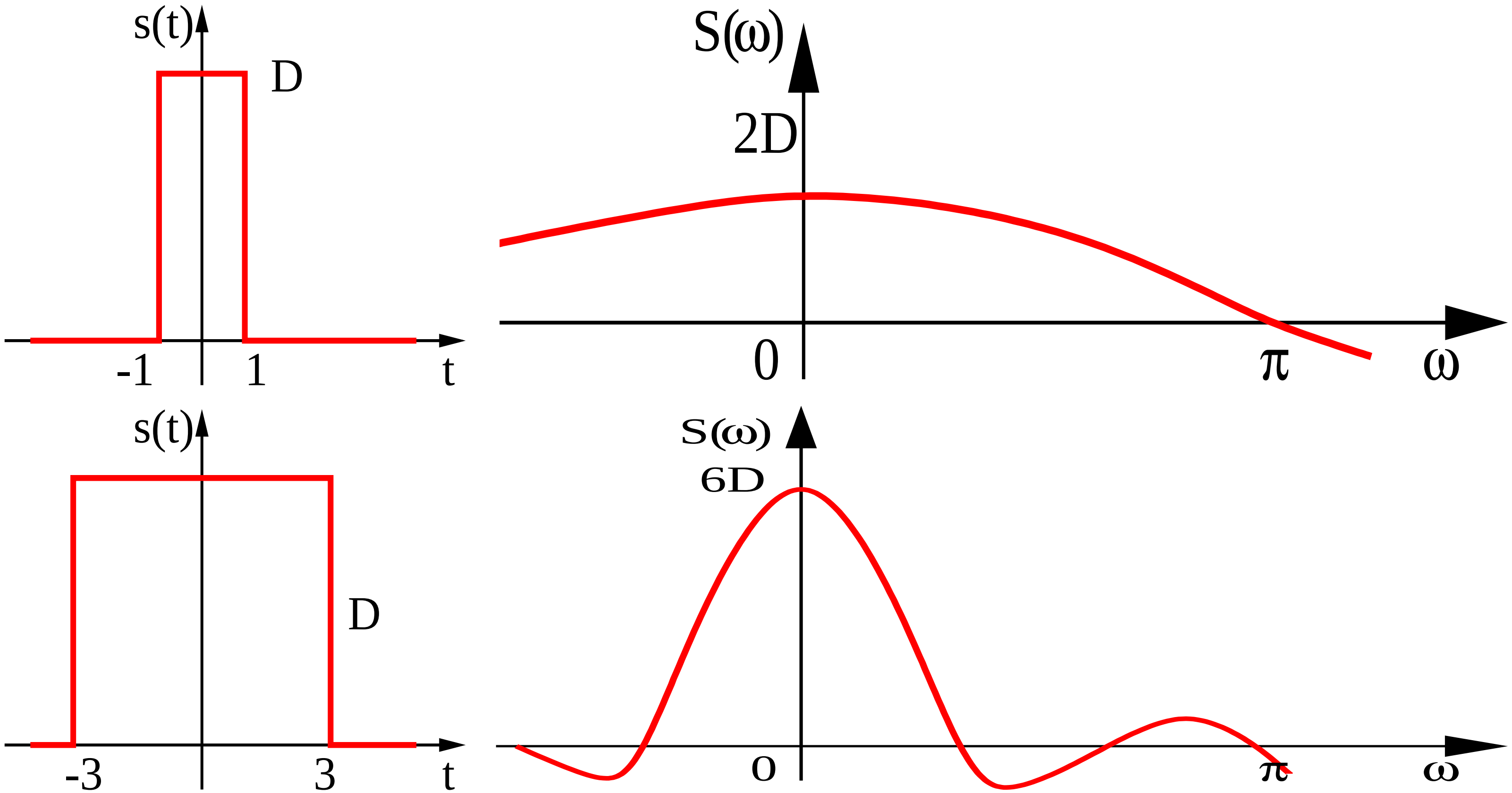
# Time-shift

1 s take over: $X(j\omega)$ multiplied by function $e^{j\omega}$, thus $\omega$ will be added to the argument:



# Examples of modification of time axis

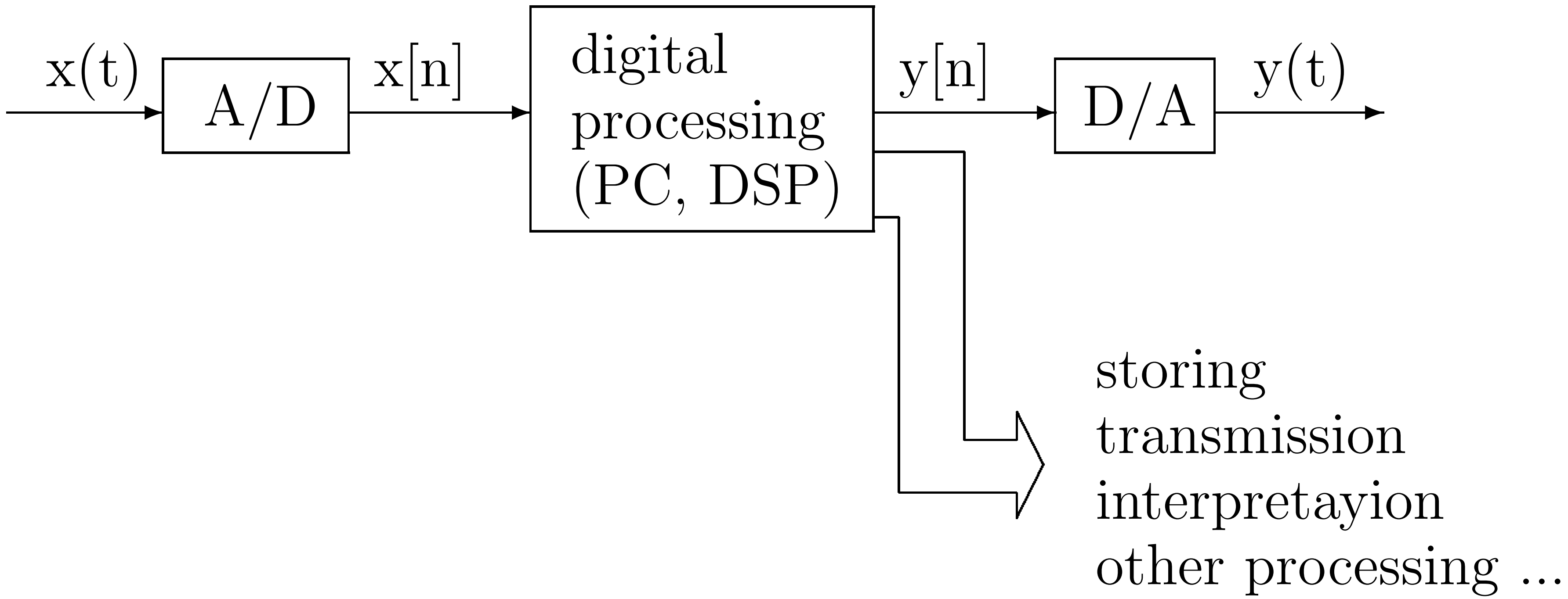


Now on ... only discrete signal processing

# Why discrete signal processing ?

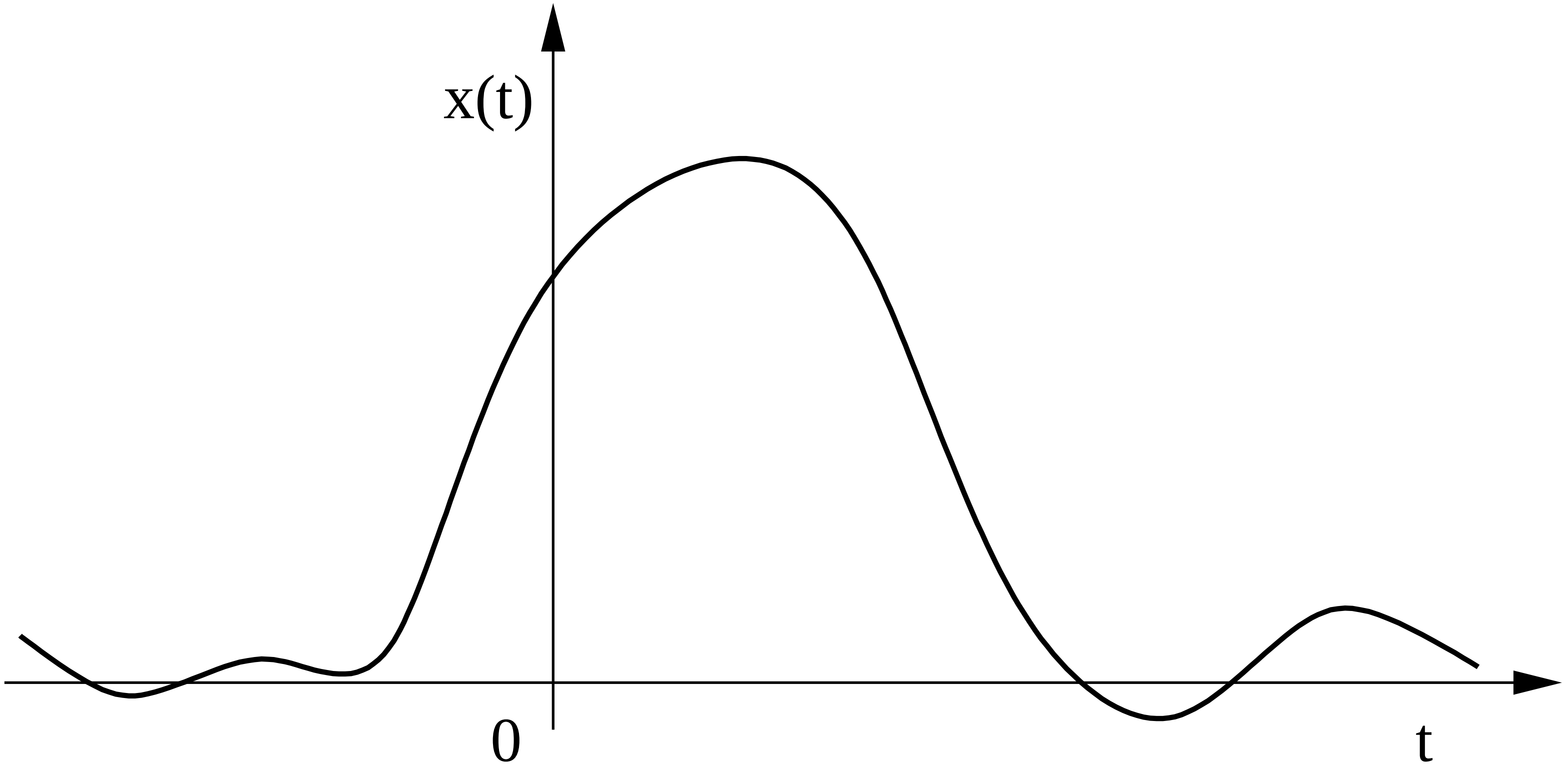
- reproducibility
- no changes due to material and temperature
- no setting and calibration
- possibility of adaptive processing
- simulation = application
- compatible with the boom of computer technology, Internet, mobile communication

# A/D conversion




At the input – continuous time signal: defined for $t \in (-\infty, +\infty)$, time has $\infty$ values

# A/D conversion




To represent the signal in spectral domain, we use Fourier transform:

$$X(f) = \int_{-\infty}^{-\infty} x(t) e^{-j2\pi f t} dt,$$

where $X(f)$ is denoted to as spectral function. $X(f)$ is defined for $\forall f$

# A/D conversion

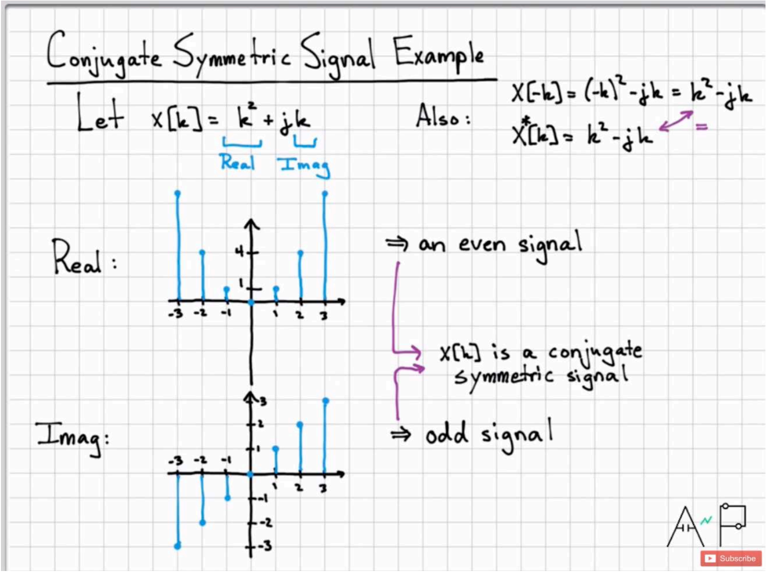
from $-\infty$ to $\infty$ and X(f) is complex:

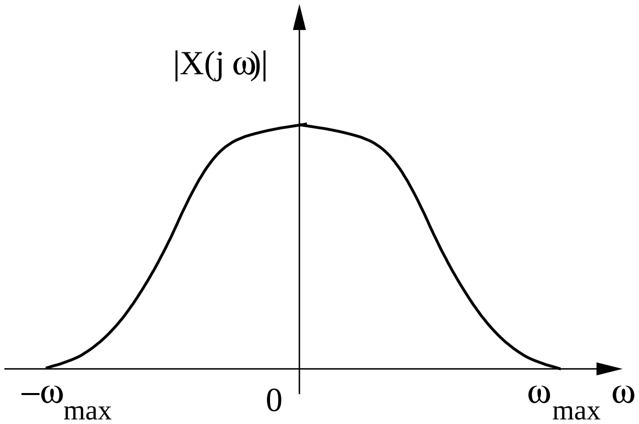
- $|X(f)|$ and $\angle X(f)$ are called module and argument of $X(f)$.

For real signals – we can just remember the right part of $X(f)$ for $f > 0$:

$$X(f) = X^{\star}(-f).$$

This means $|X(f)| = |X(-f)|$ and $\arg X(f) = -\arg X(-f)$.



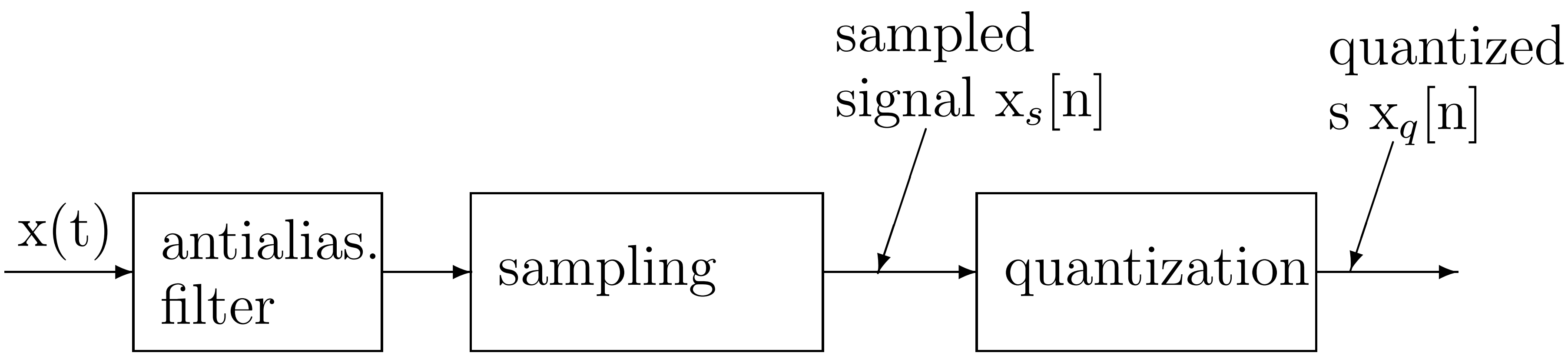

**Definition**

A complex-valued signal $x[k]$ is conjugate symmetric if $x[-k] = x^*[k]$ for all $k$

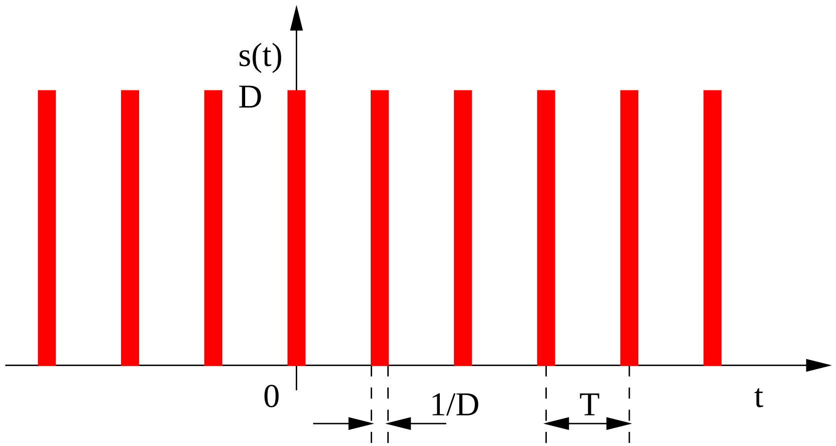
- The Asterisk Denotes Complex Conjugation
- If $x[k] = a[k] + jb[k]$, then $x^*[k] = a[k] - jb[k]$
- A Conjugate Symmetric Complex-Valued Signal Has An Even Real Part and Odd Imaginary Part

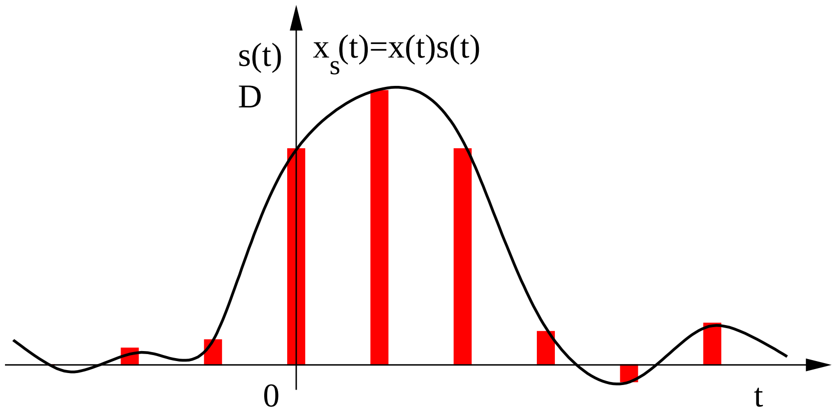
$\Rightarrow$ Intelligent signals are frequency constrained: energy lies in $(0, f_{max})$.

# A/D conversion




Sampled signal is obtained in such a way that the original signal is multiplied with time-periodic sequence of Dirac impulses.

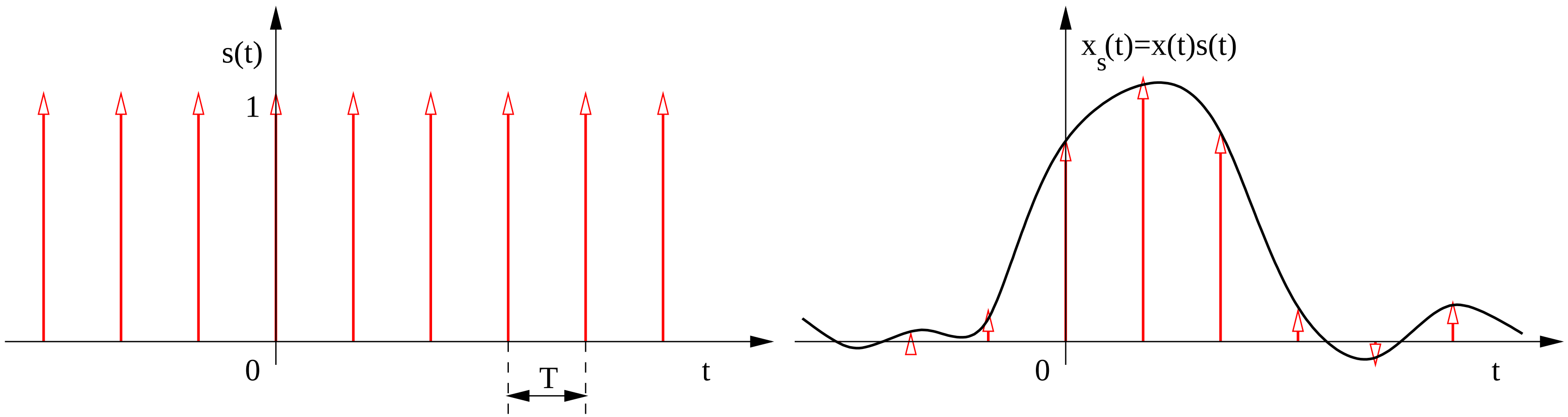


# A/D conversion

In theory Dirac impulses have: height $\rightarrow \infty$, width $\rightarrow 0$, area/mass $\rightarrow 1$.

After multiplication: again sequence of Dirac impulses, but mass is equal to the values of original signal in samples $nT$.




$T$ is sampling period

$$F_s = \frac{1}{T} \text{ is sampling frequency}$$

# A/D conversion

What happens with spectrum of a sampled signal? It becomes **periodic !!!**

$$X_s(f) = \frac{1}{T} \sum_{n=-\infty}^{+\infty} X\left(f - \frac{n}{T}\right) = \frac{1}{T} \sum_{n=-\infty}^{+\infty} X\left(f - nF_s\right)$$

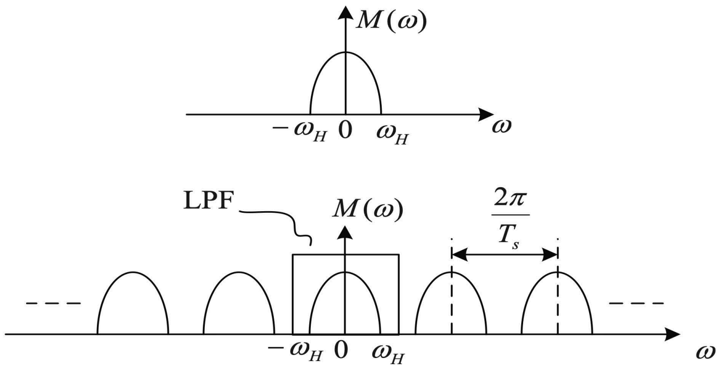
According to the ratio of $f_{max}$ and $F_s$, two possibilities can appear:
**1)** $F_s > 2f_{max}$: Original signal can be ideally reconstructed, individual copies of the spectra do NOT overlap.
**2)** $F_s \leq 2f_{max}$: Original signal can NOT be anyhow reconstructed, individual copies of the spectra DO overlap. **Shannon – Nyquist sampling theorem:**

$$\boxed{F_s > 2f_{max}}$$

# Reconstruction

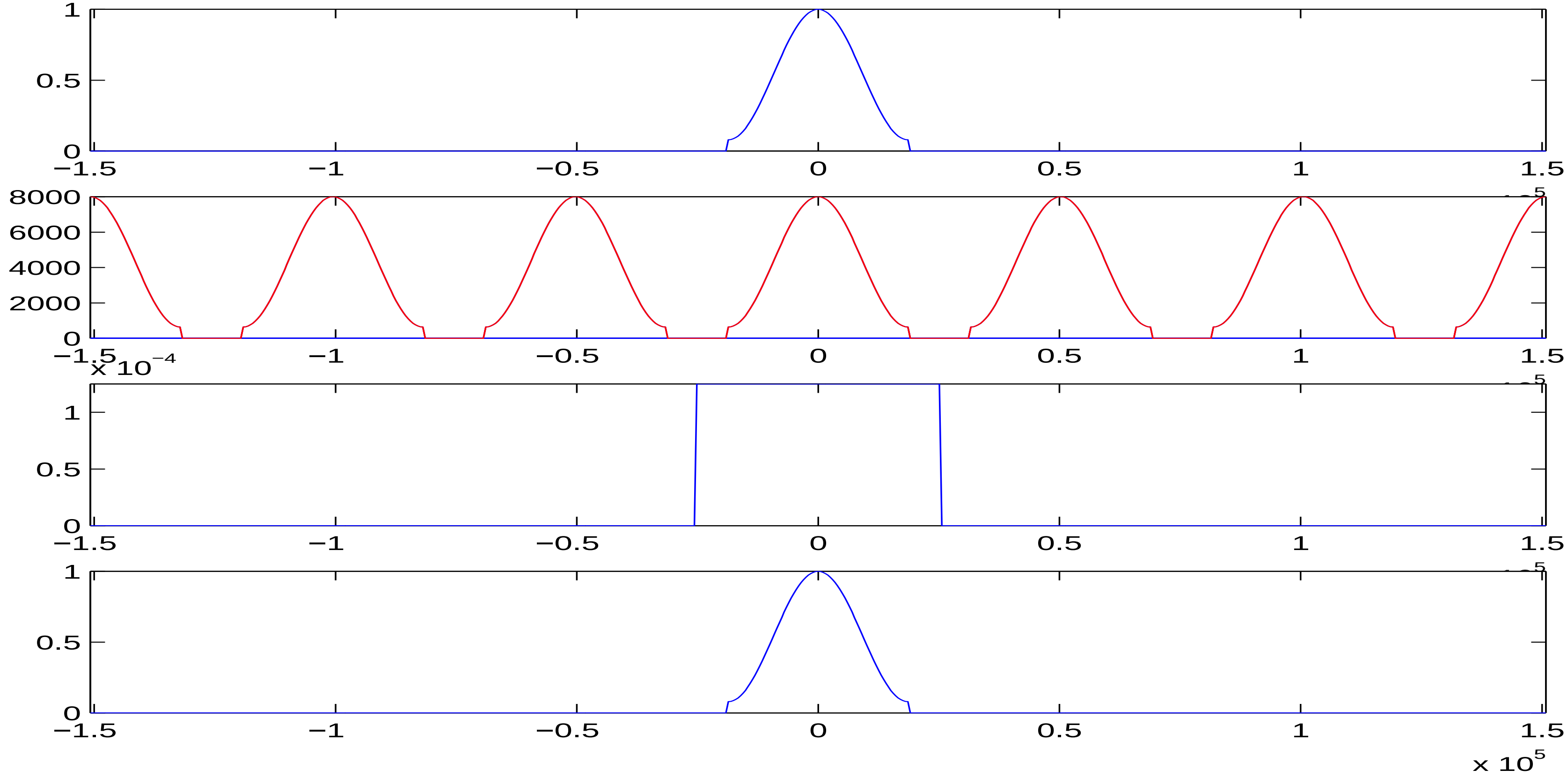
Low-pass filter with cutting frequency $\Omega_s/2$:

$$H_r(j\omega) = \begin{cases} T & \text{for} \quad -\Omega_s/2 < \omega < \Omega_s/2 \\ 0 & \text{else} \end{cases}$$



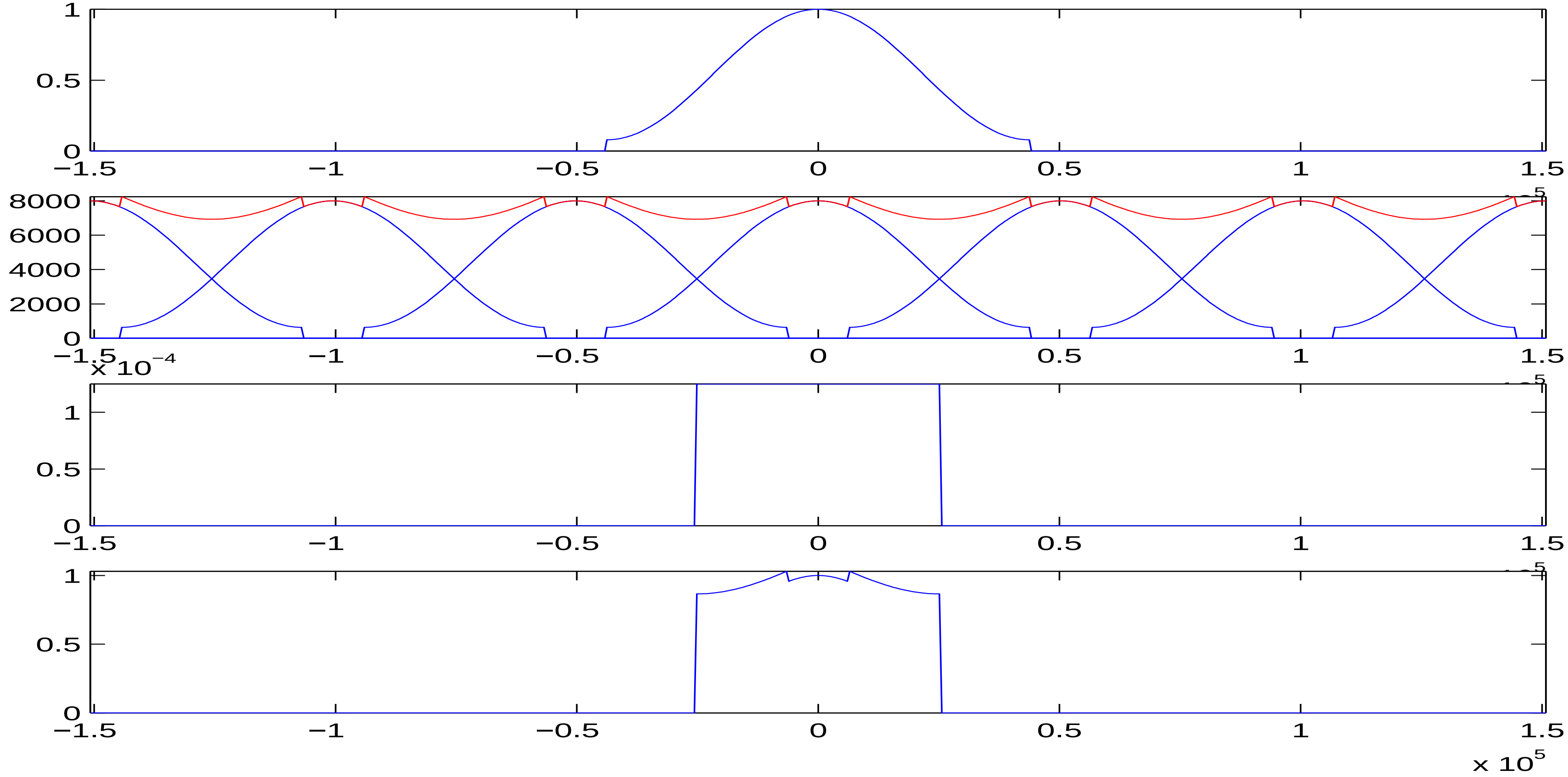
# 1. Exercise sampling and reconstruction – OK

$F_s = 8000\ Hz$, $f_{max} = 3000\ Hz$, thus $\Omega_s = 16000\pi$ rad/s, $\omega_{max} = 6000\pi$ rad/s. $T = \frac{1}{8000}$ s

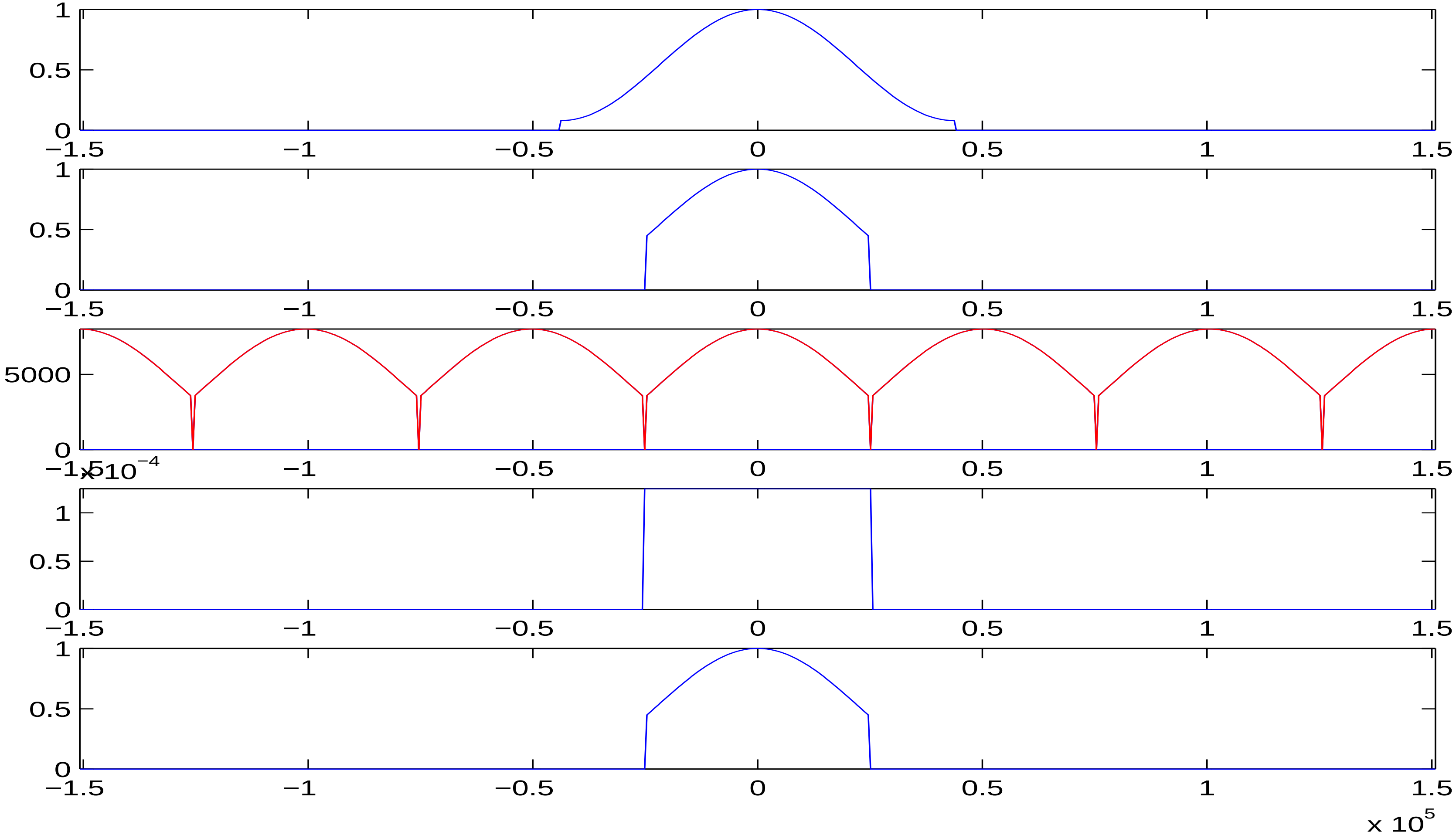


# 2. Exercise sampling and reconstruction – BAD

$F_s = 8000\ Hz$, $f_{max} = 7000\ Hz$, thus $\Omega_s = 16000\pi$ rad/s, $\omega_{max} = 14000\pi$ rad/s. $T = \frac{1}{8000}$ s



# Antialiasing filter – restriction for $[-F_s/2,\ F_s/2]$



# Spectrum of discrete-time sampled signals

Discrete Fourier transform – DFT – definition:

$$X(k) = \sum_{n=0}^{N-1} x[n] e^{-j2\pi \frac{nk}{N}} \quad \text{pro } k \in < 0, N-1 >$$

How to apply DFT on discrete signal:

- analysis of the "window" of length equal to $N$ samples.
- Output? If $X(k)$ are multiplied by sampling period $T$, we obtain an approximation of spectral function in frequency samples $k\Delta f$, where $\Delta f = \dfrac{F_s}{N}$

$$\hat{X}(k\Delta f) = T \sum_{n=0}^{N-1} x[n] e^{-j2\pi \frac{nk}{N}}$$

$x[n]$ in the equation can be replaced by $x(nT)$.

# Spectrum of discrete-time sampled signals

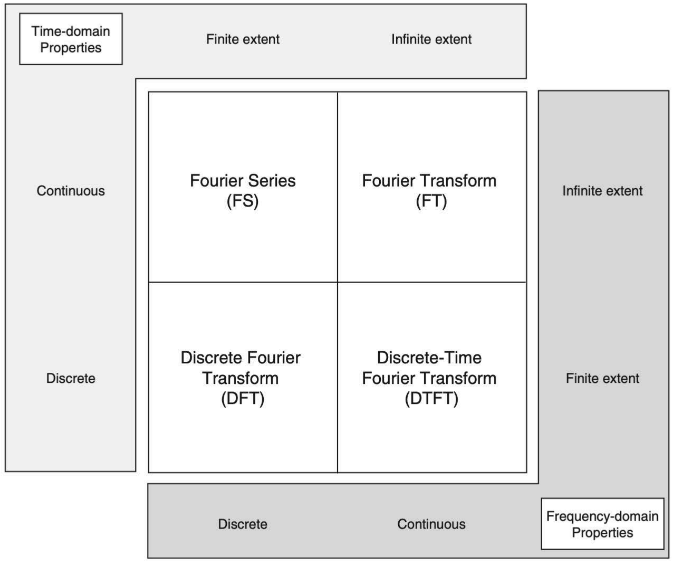
Comparison with spectral function obtained using analog FT: different results !!!

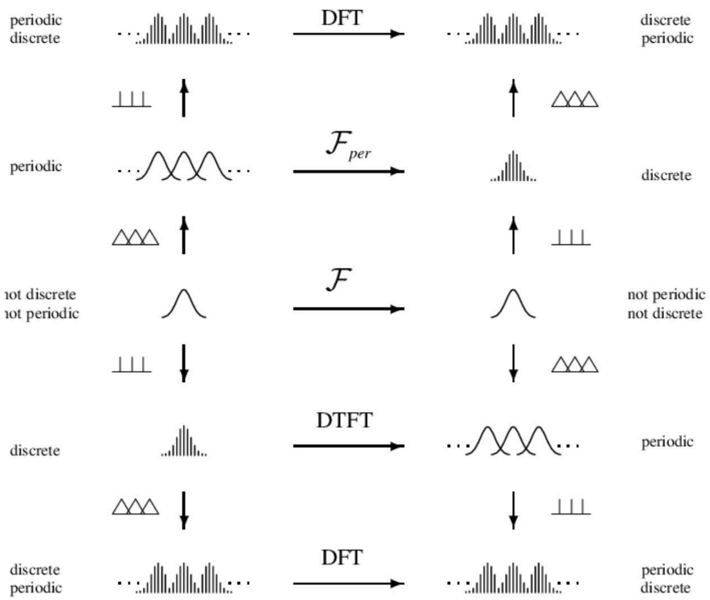
1. Spectrum of sampled signal is computed $\rightarrow$ periodic spectrum ($N$ samples $\sim F_s$). If we let $k \in (-\infty, +\infty)$, we find out that $\hat{X}(k\Delta f)$ is periodic with period $N$.

2. Signal was segmented using a "window". Properties of such the window influence $\hat{X}(k\Delta f)$. In time domain, the window is multiplied with the original signal $\Longleftrightarrow$ spectrum of the window is convolved with $\hat{X}(k\Delta f)$.

3. $\hat{X}(k\Delta f)$ is discrete $\Rightarrow$ spectrum of the periodic signal was computed. We can imagine that the signal's window is repeating $\infty$-times.

# Spectrum of discrete-time sampled signals




DF DCT

| time | frequency |
|---|---|
| sampling | periodic |
| periodic | discrete |

**Fourier Series (FS)**

For $x(t)$ of duration $T$, set $\omega_0 = \frac{2\pi}{T}$.

$x(t): \quad 0 \le t \le T$
$X[k]: \quad k = \ldots, -2, -1, 0, 1, 2, \ldots$

$$X[k] = \frac{1}{T}\int_{t=0}^{T} x(t)\, e^{-jk\omega_0 t} dt$$

$$x(t) = \sum_{k=-\infty}^{\infty} X[k]\, e^{jk\omega_0 t}$$

**Fourier Transform (FT)**

$x(t): \quad -\infty < t < \infty$
$X(\omega): \quad -\infty < \omega < \infty$

$$X(\omega) = \int_{t=-\infty}^{\infty} x(t)\, e^{-j\omega t} dt$$

$$x(t) = \frac{1}{2\pi}\int_{\omega=-\infty}^{\infty} X(\omega)\, e^{j\omega t} d\omega$$

**Discrete Fourier Transform (DFT)**

For $x[n]$ of length $N$, set $\omega_0 = \frac{2\pi}{N}$.

$x[n]: \quad n = 0, 1, \ldots, N-1$
$X[k]: \quad k = 0, 1, \ldots, N-1$

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-jk\omega_0 n}$$

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]\, e^{jk\omega_0 n}$$

**Discrete-Time Fourier Transform (DTFT)**

$x[n]: \quad n = \ldots, -2, -1, 0, 1, 2, \ldots$
$X(\omega): \quad -\pi \le \omega \le \pi$

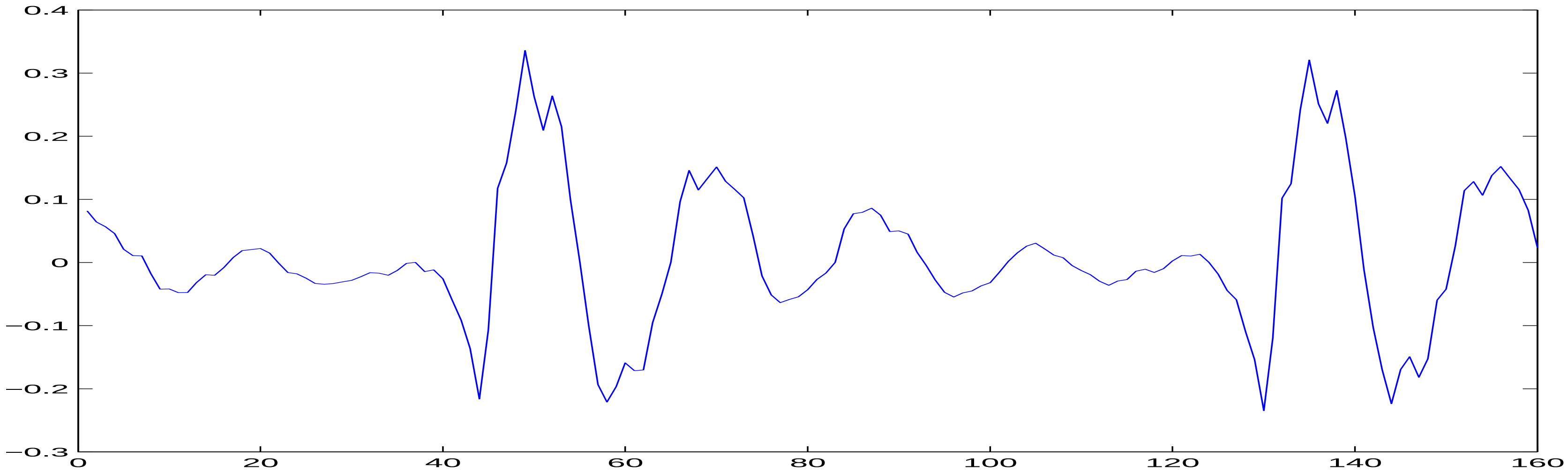
$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n]\, e^{-j\omega n}$$

$$x[n] = \frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} X(\omega)\, e^{j\omega n} d\omega$$

# Example – Real signal!

Analysis of a voiced segment of a speech signal:

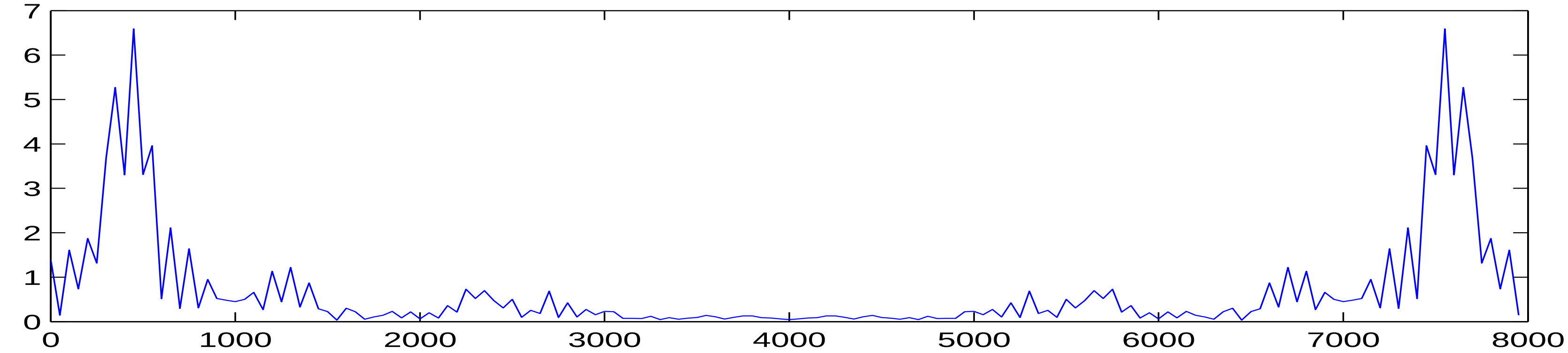
```
s = wavread('test.wav')';
sfr = frame (s,160,80);
x = sfr(:,13);
plot (x);
```

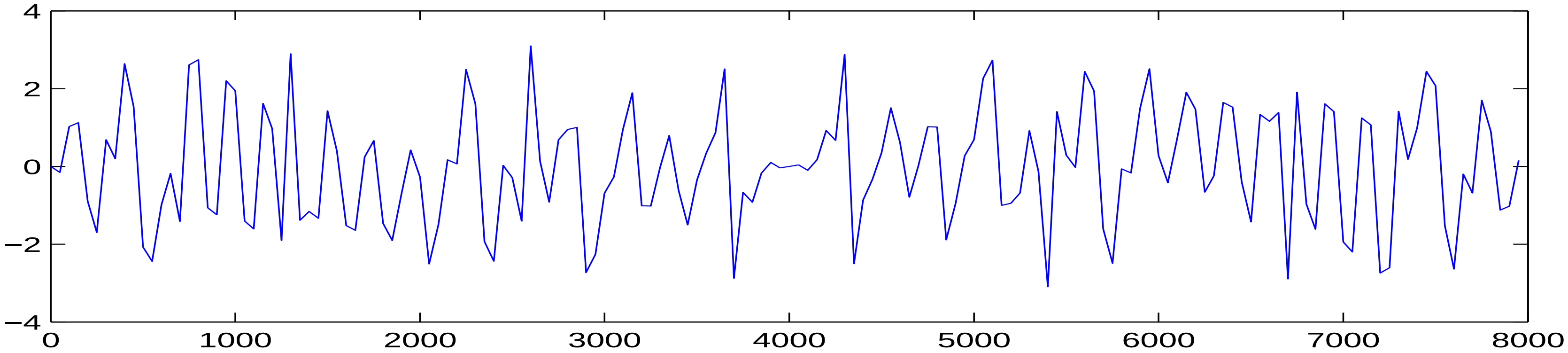


# Example – Real signal!

Only DFT –

```
Fs = 8000; f = (0:159) / 160 * Fs; X = fft(x);
subplot (211); plot(f,abs(X)); subplot (212); plot(f,angle(X));
```

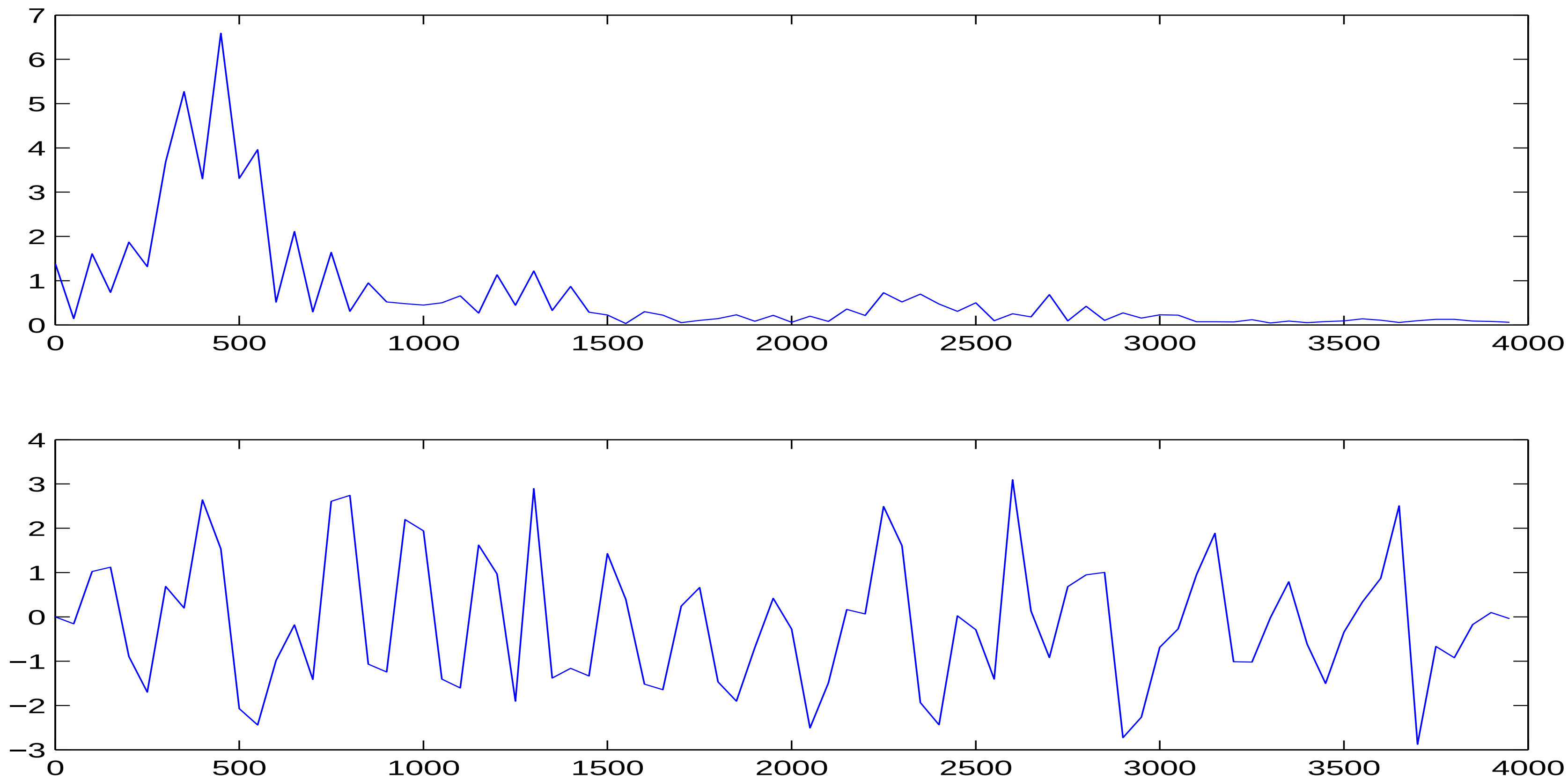


# Example – Real signal!

Only DFT – upper part of the spectrum is symmetric to the lower part

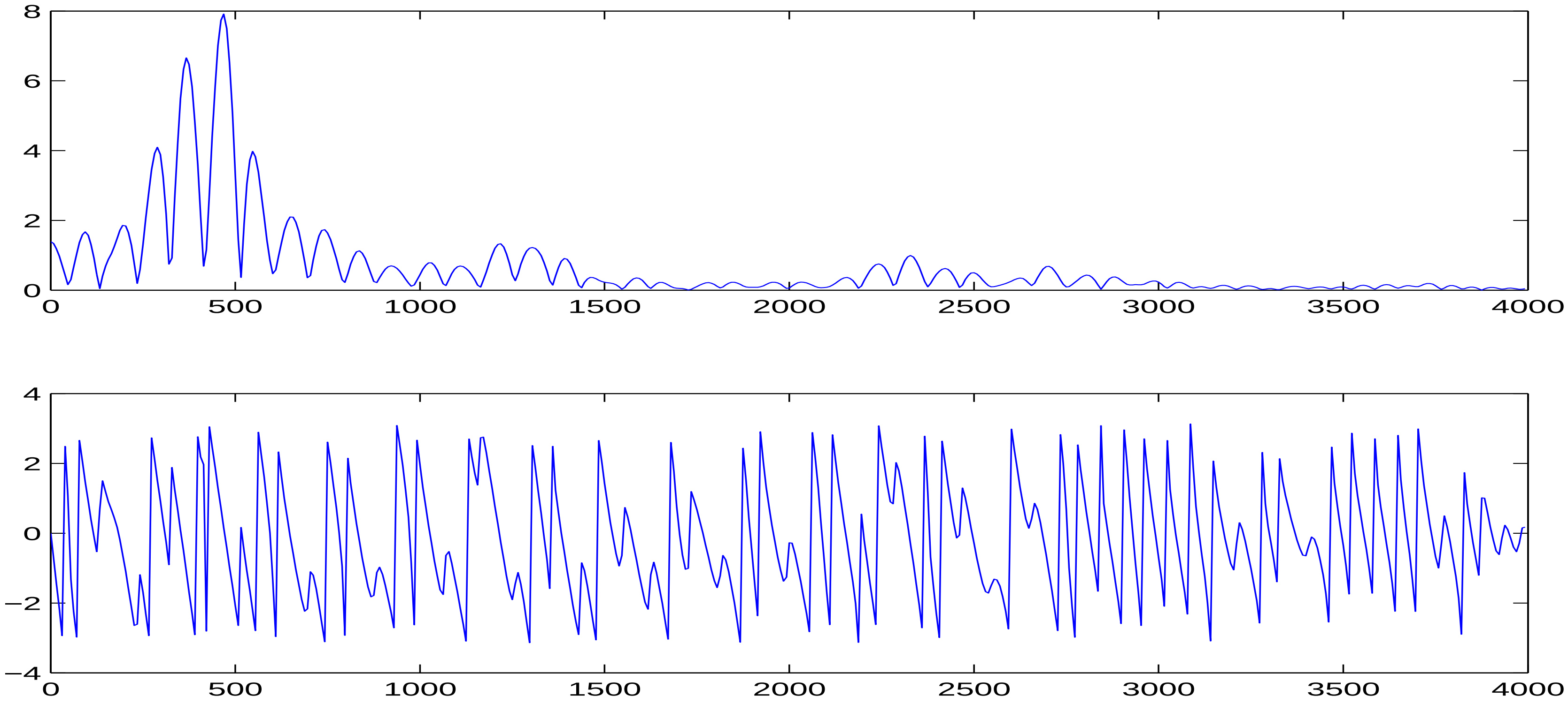
```
Fs = 8000; f = (0:79) / 160 * Fs; X = fft(x); X = X(1:80);
subplot (211); plot(f,abs(X)); subplot (212); plot(f,angle(X));
```



# Example – Real signal!

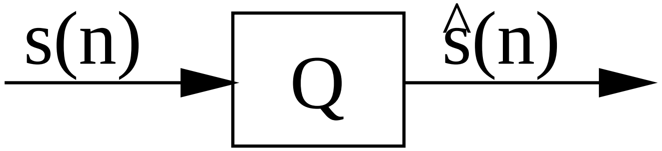
More samples in the spectrum – the frame cannot be enlarged ⇒ zero padding:

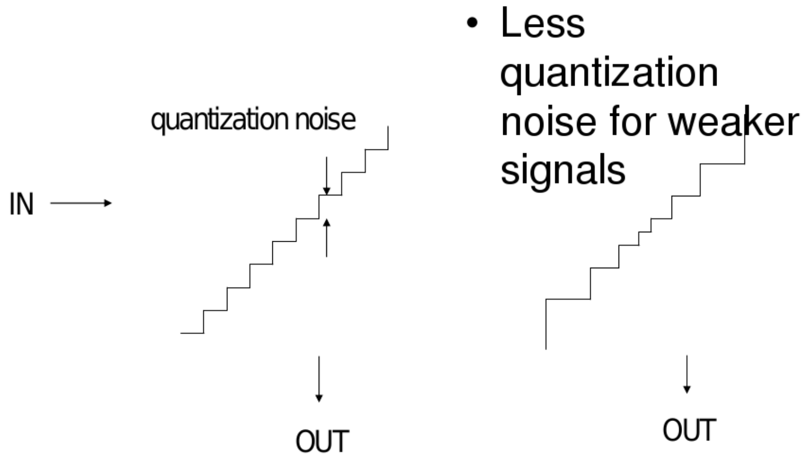
```
Fs = 8000; f = (0:511) / 1024 * Fs;
X = fft([x' zeros(1,1024-160)]); X = X(1:512);
subplot (211); plot(f,abs(X)); subplot (212); plot(f,angle(X));
```



# Quantization

if quantization bits are increased by 1, the SNR increases by 6dB.




Pulse Code Modulation (PCM):

- dynamic range of speech is about 50-60dB – 11bits/sample
- maximum frequency in telephone speech is 3.4kHz – $F_s = 8$kHz

$\Rightarrow$ 88 kbps – simple/universal but not efficient in case of linear quantization:

$$SNR = 6B + K \qquad\qquad (\frac{S}{N_q})_{\mathrm{dB}} = 20 \log M = 6N$$

in dB, where B = number of bits, K - const. depending on the

# Quantization

character of the signal.

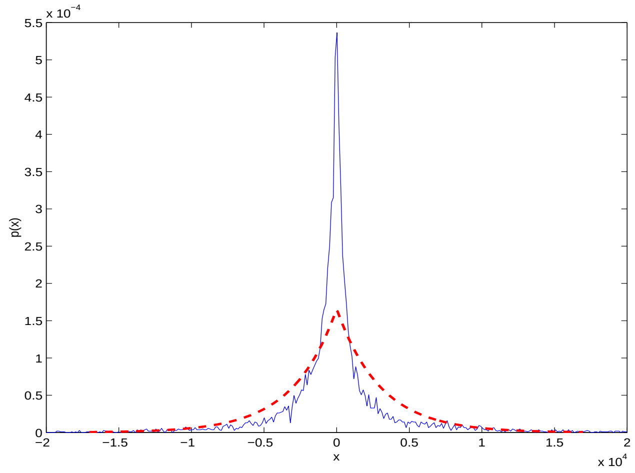
# Quantization

For speech signal - linear quantization is not optimal:

- speech contains many "small values", PDF can be approximated by Laplace distribution:

$$p(x) = \frac{1}{\sqrt{2}\sigma_x} e^{-\frac{\sqrt{2}|x|}{\sigma_x}},$$

where $\sigma_x$ is the standard deviation. Example - speech without silence:



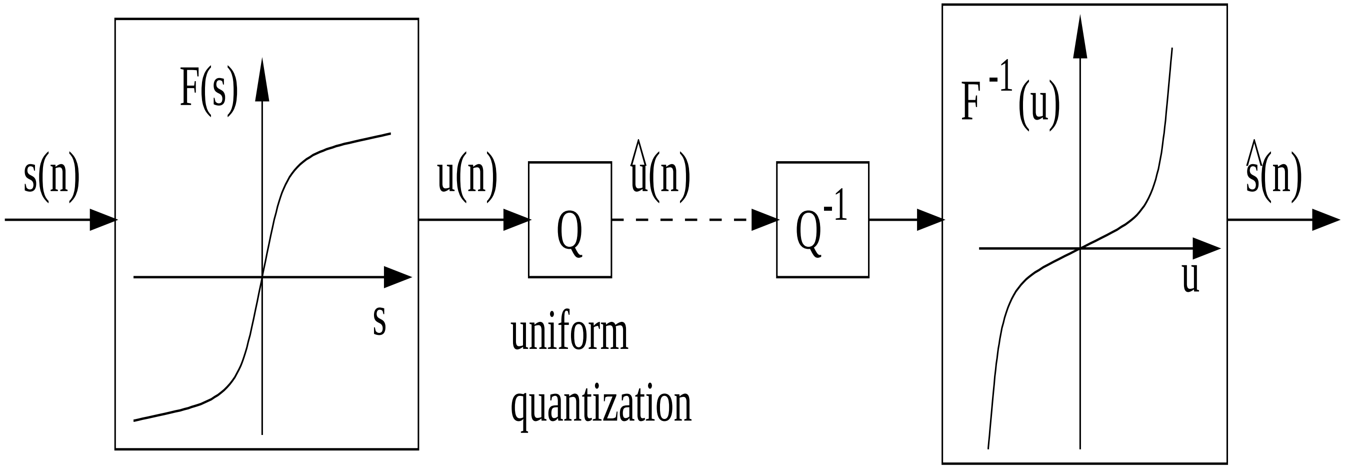

- Perceptual properties of human auditory system: logarithmic

# Quantization

sensitivity to the amplitude of acoustic signal

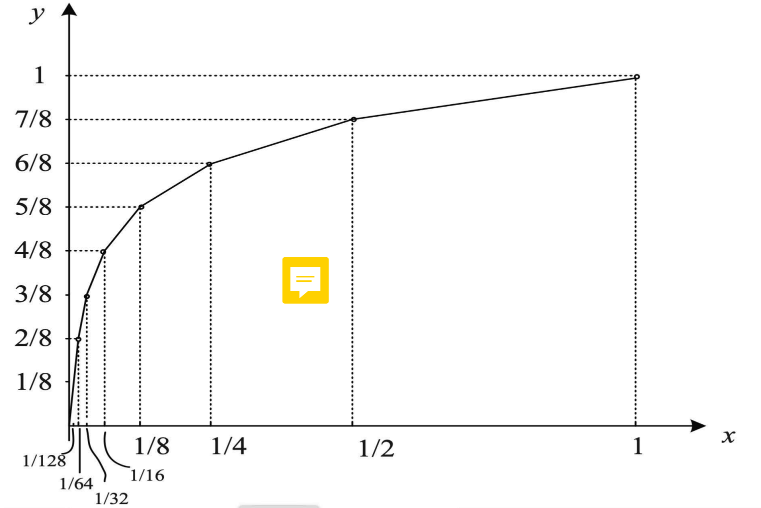
# Logarithmic PCM

Compression in coder $\Longleftrightarrow$ expansion in decoder



# Logarithmic PCM

Non-linearity cannot be log-directly: $(\log(0) = -\infty) \Rightarrow$ approximation:




- Europe: **A-law**:

$$u(n) = S_{max} \frac{1 + \ln A \frac{|s(n)|}{S_{max}}}{1 + lnA} sign[s(n)], \quad \text{where} \quad A = 87.56.$$

- USA: $\mu$-law:

$$u(n) = S_{max} \frac{ln\left(1 + \mu \frac{|s(n)|}{S_{max}}\right)}{ln(1+\mu)} sign[s(n)],$$

The PCM consists of three steps: sampling, quantization, and **coding**.

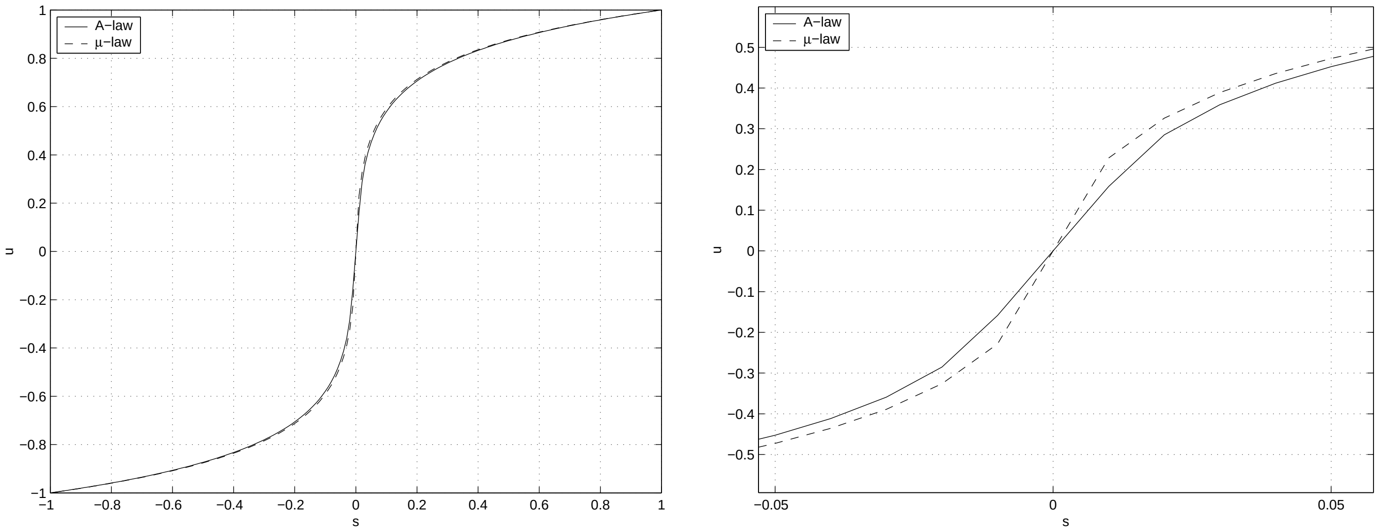
- A-law 13 broke line: the input range $[0, 1]$ is non-uniformly divided into 8 segments, [0, 1/128], ..., [1/4, 1/2], [1/2, 1].
- Each segment is uniformly divided into 16 smaller segments.

Table: Folded binary code and nature binary code

| Polarity | Nature | Folded | Levels |
|---|---|---|---|
| + | 1111 | 1111 | 15 |
| | 1110 | 1110 | 14 |
| | 1101 | 1101 | 13 |
| | 1100 | 1100 | 12 |
| | 1011 | 1011 | 11 |
| | 1010 | 1010 | 10 |
| | 1001 | 1001 | 9 |
| | 1000 | 1000 | 8 |
| − | 0111 | 0000 | 7 |
| | 0110 | 0001 | 6 |
| | 0101 | 0010 | 5 |
| | 0100 | 0011 | 4 |
| | 0011 | 0100 | 3 |
| | 0010 | 0101 | 2 |
| | 0001 | 0110 | 1 |
| | 0000 | 0111 | 0 |

# Logarithmic PCM

Comparison of A-law a $\mu$-law:



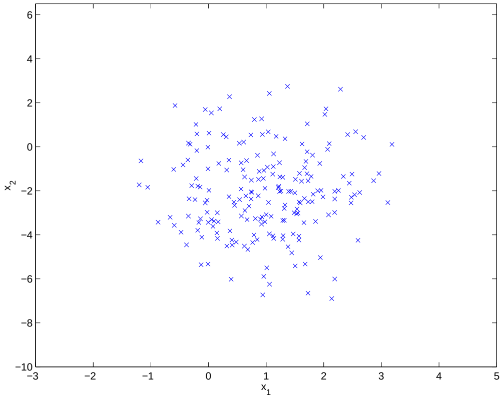

$\Rightarrow$ both practically similar, the both improve qualities for "small signals" by 12dB. For telephone applications, log-PCM with 8bits reaches same quality as lin-PCM with 13bits – CCITT G.711.

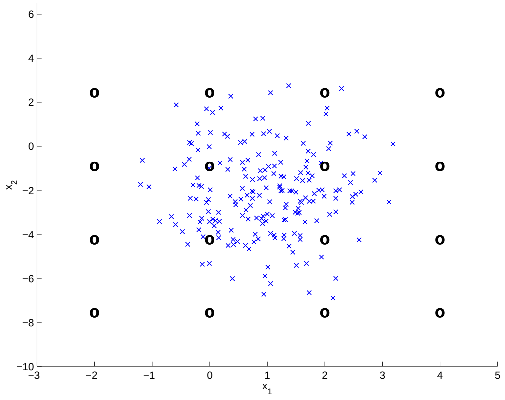
# Vector quantization

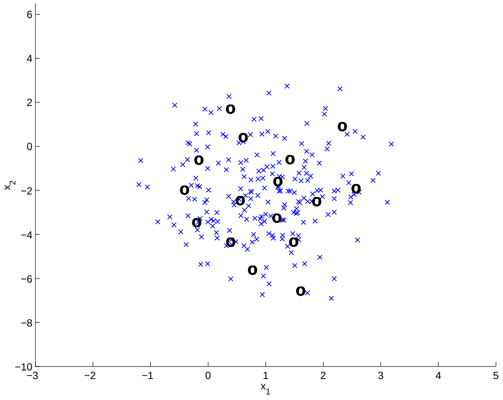
Why:

- Usually, processing of speech/acoustic vectors, vectors are correlated.
- Scalar quantization - inefficient utilization of bits
- How about spread into P-dimensional space “typical vectors – centroids” and quantize new vectors according to these centroids.
- Variants of VQ: split-VQ, algebraic VQ, random codebook, tree-structured VQ, multi-stage VQ

# Vector quantization

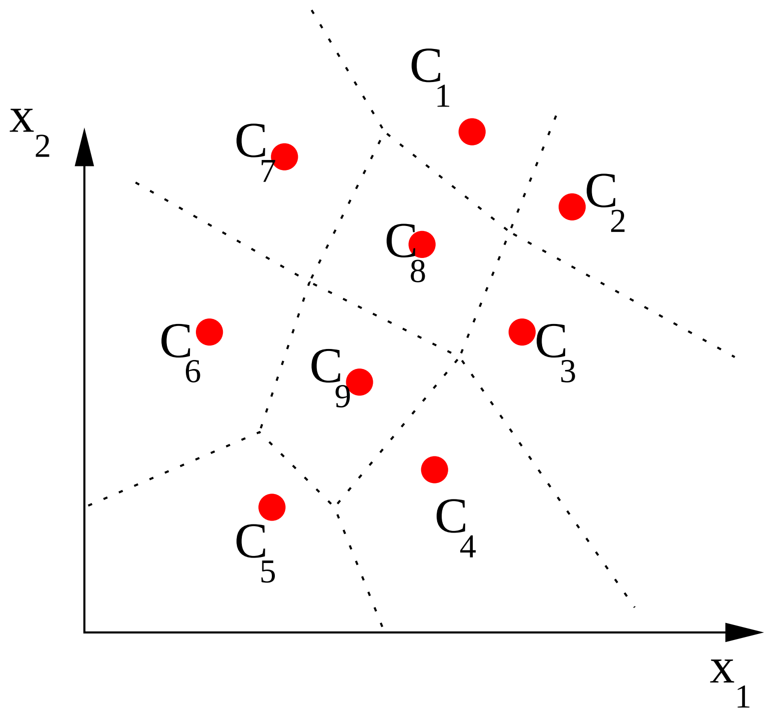


# Vector quantization

**Code vectors, centroids, Voronoi's regions ...**



# Training of code vectors

Code vectors need to be trained on data: N-training vectors, expected codebook $\mathbf{Y}$ of size K: a) $K$-means:

- **Inicialisation:** $k = 0$, define $\mathbf{Y}(0)$.
- **Step 1:** asigning vectors to centroids:

  $$Q[\mathbf{x}] = \mathbf{y}_i(k) \text{ if } d(\mathbf{x}, \mathbf{y}_i(k)) \leq d(\mathbf{x}, \mathbf{y}_j(k)) \text{ pro } j \neq i, \; j \in 1 \ldots K$$

  As $d(\mathbf{x}, \mathbf{y}_j)$, Euclidean distance can be used:

  $$d(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T (\mathbf{x} - \mathbf{y})} = \sqrt{\sum_{k=1}^{P} |x_k - y_k|^2}.$$

- **Step 2:** evaluation of quality of codebook:

  $$D_{VQ} = \frac{1}{N} \sum_{n=1}^{N} d\left(\mathbf{x}(n), Q[\mathbf{x}(n)]\right).$$

# Training of code vectors

If $\dfrac{D_{VQ}(k-1) - D_{VQ}(k)}{D_{VQ}(k)} \leq \varepsilon$, STOP training.

- **Step 4:** New codebook:

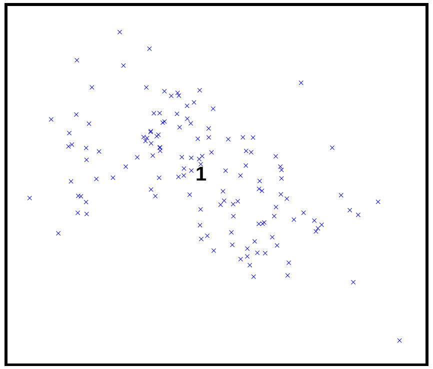
$$\mathbf{y}_i(k+1) = Cent(C_i(k)) = \frac{1}{M_i(k)} \sum_{\mathbf{x} \in C_i(k)} \mathbf{x},$$

where $M_i(k)$ is number of training vectors in k-iteration to be asigned for centroid $i$.

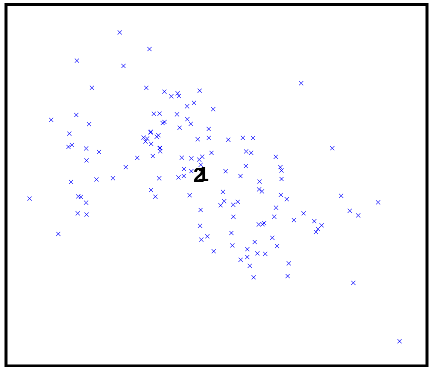
b) LBG – Linde-Buzo-Gray: $K$-means has problems with initialization ($\mathbf{Y}(0)$), no training vector for some centroid !!! LBG - sequential incrementation of size of codebook:

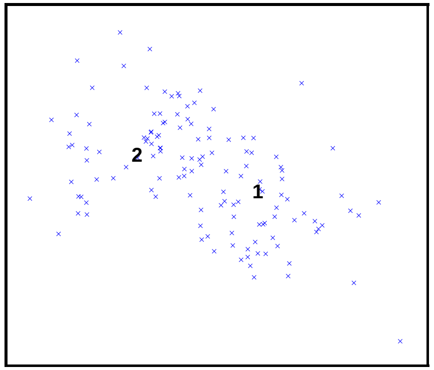
# Training of code vectors

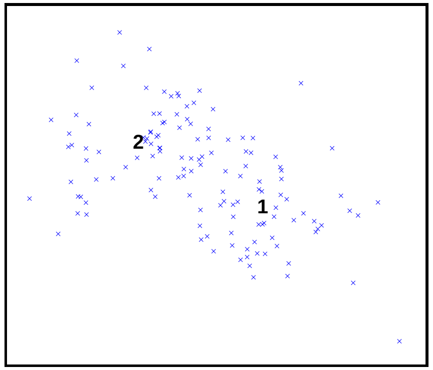
$r = 0, L = 1$




$r = 1, L = 2$

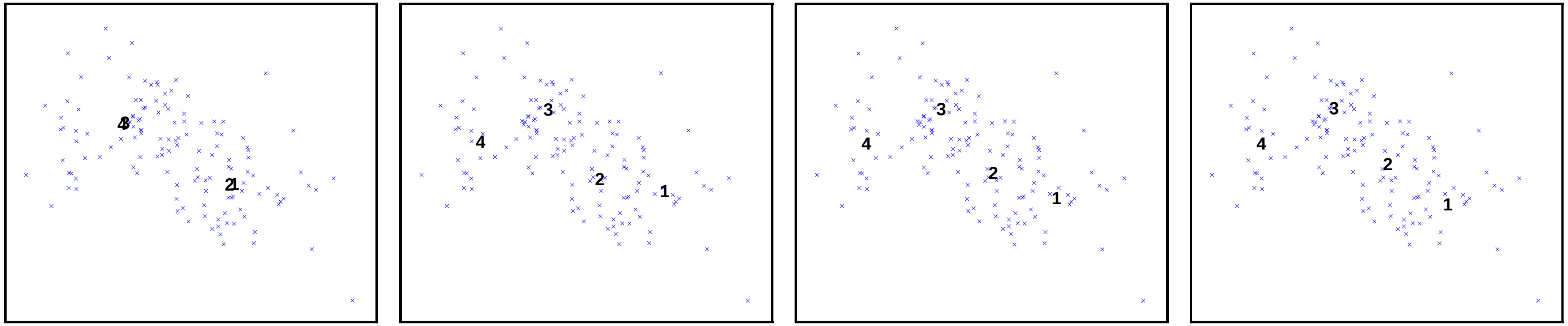


# Training of code vectors

$r = 2, L = 4$



$r = 3, L = 8$