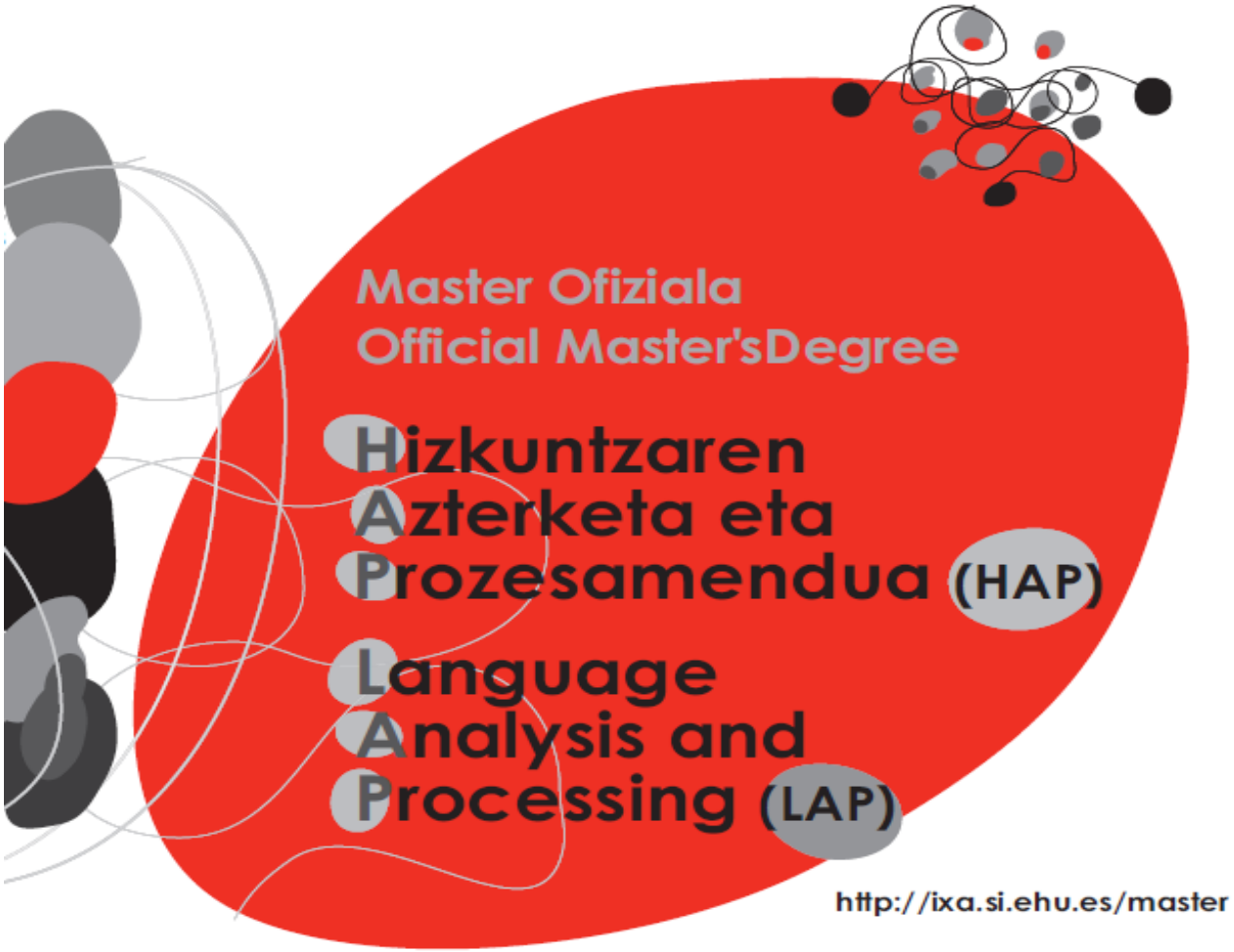
Erasmus Mundus



Language and Communication Technologies

Universidad del País Vasco · Euskal Herriko Unibertsitatea

MDE Master eta Doktorego Eskola · Escuela de Máster y Doctorado · Master and Doctoral School

lct

Erasmus Mundus

LT-M4 Speech Processing (and speech technologies)

Speech signal: representations

Universidad del País Vasco
Euskal Herriko Unibertsitatea
MDE
Master eta Doktorego Eskola
Escuela de Máster y Doctorado
Master and Doctoral School

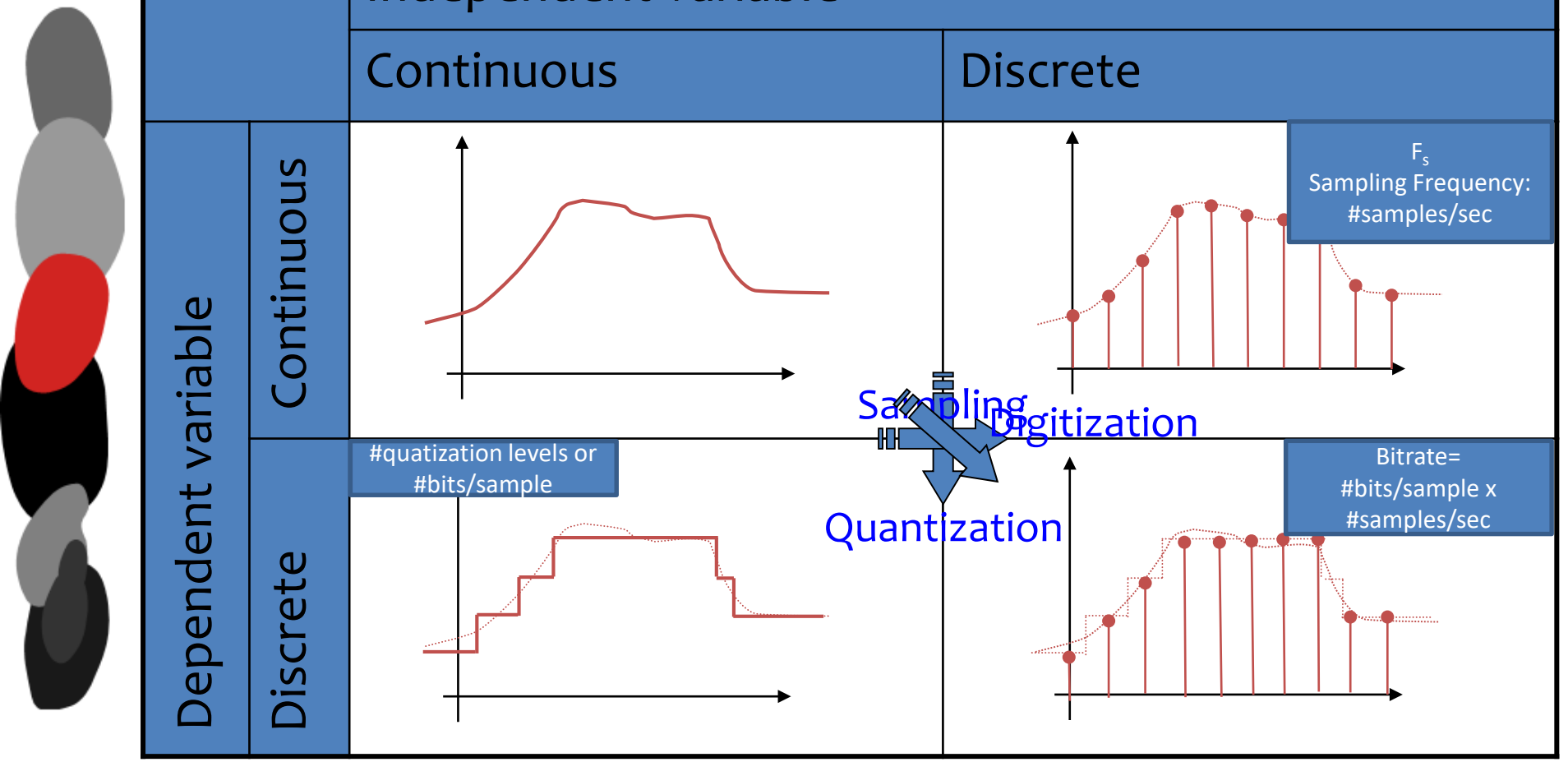
Syllabus

- Lesson 1: Speech production and perception
- Lesson 2: Basic concepts about signals & systems
- Lesson 3: Speech signal: representations

Outline

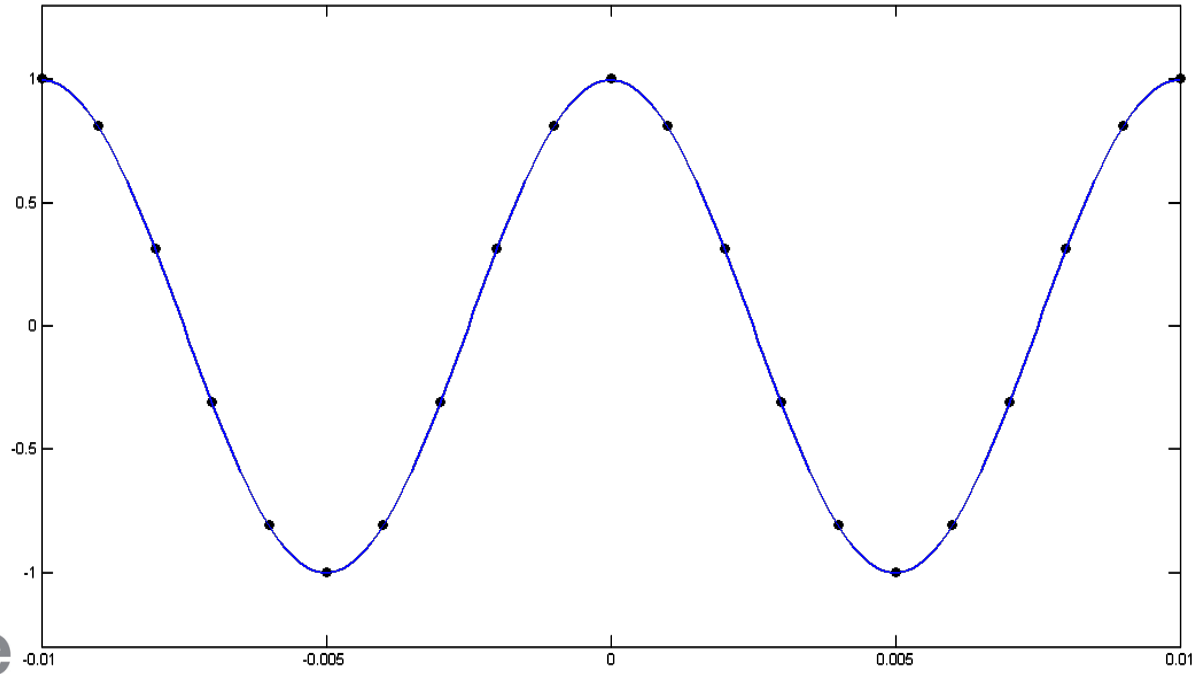
1. Digitization
2. Supra-segmental acoustic features
3. Segmental acoustic features
4. Mathematical modeling of the speech sequence

Analog vs. digital signals



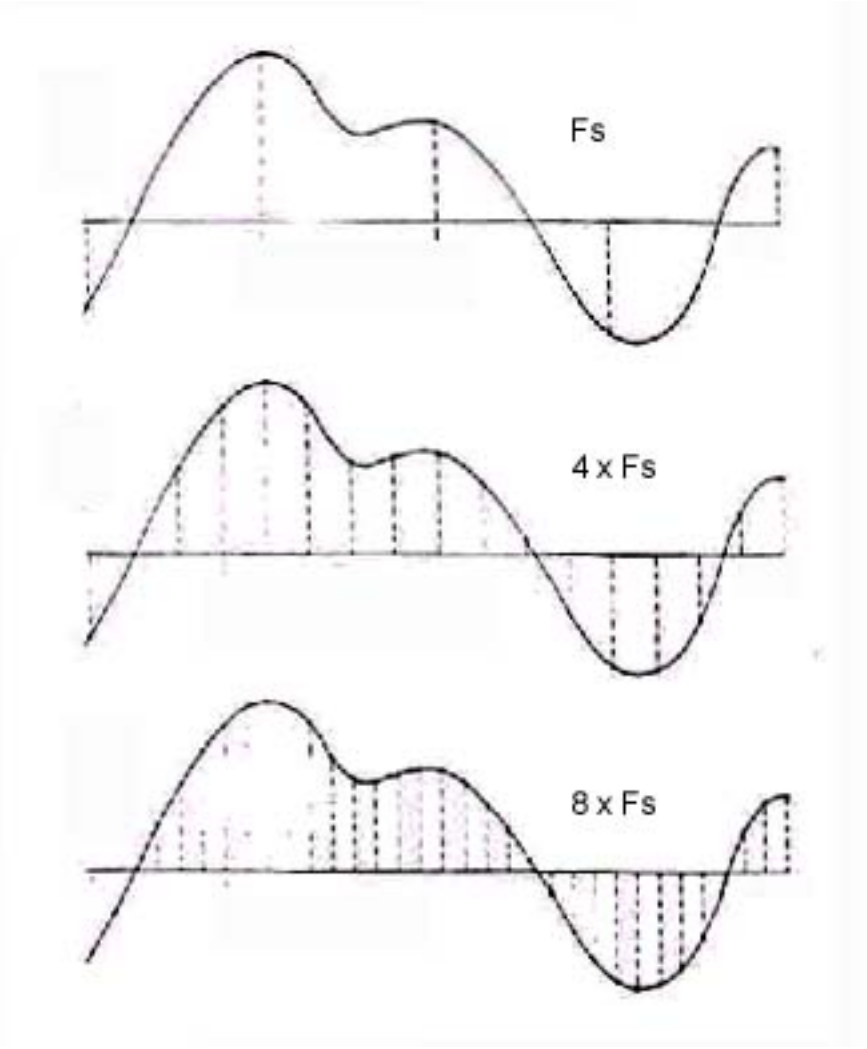
Digitizing

- Digitization is the representation of an analog signal by a series of numbers corresponding to points of the signal (samples).



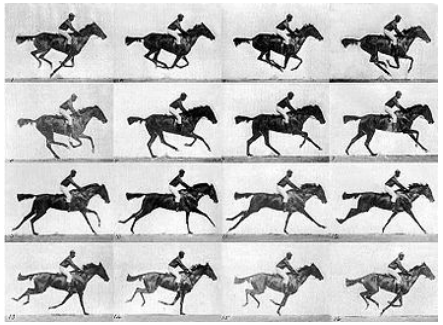
Sampling process

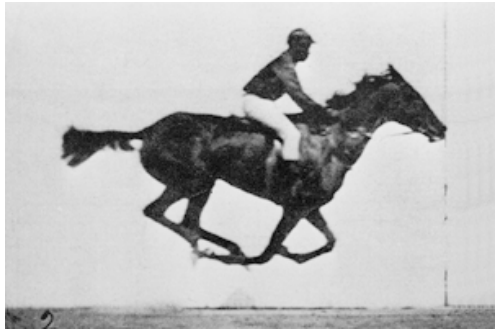
- Sampling frequency:
  - Sampling period Ts=1/ f_s
  - Sampling frequency:
    - f_s -> samples/s
    - Samples taken by time unit
- The faster the signal, the higher the sampling frequency.
- Which is the suitable criterion to choose f_s?



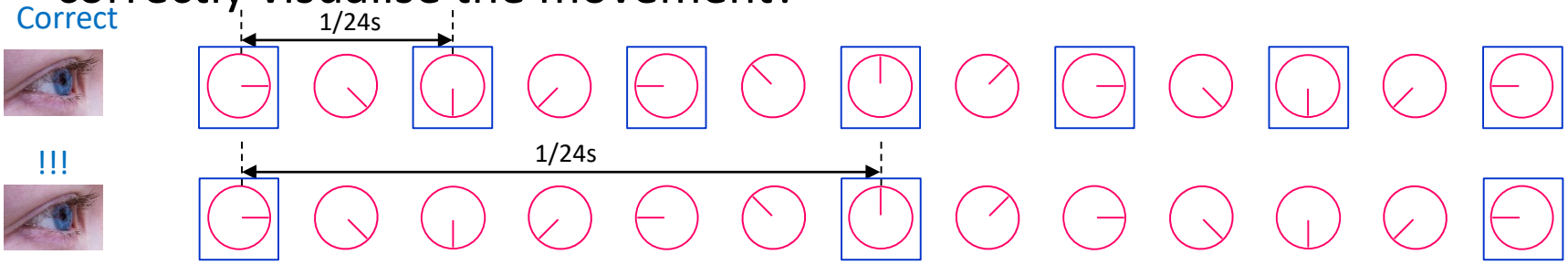
Sampling

- Video: A film captures a image every 1/24 seconds.





- Which is the minimum number of images per second required to correctly visualise the movement?



- In each turn at least 2 images are required to see the wheel rolling forward.

Sampling theorem

- To be able to sample and then recover a signal from its samples:
  - The signal must be band limited

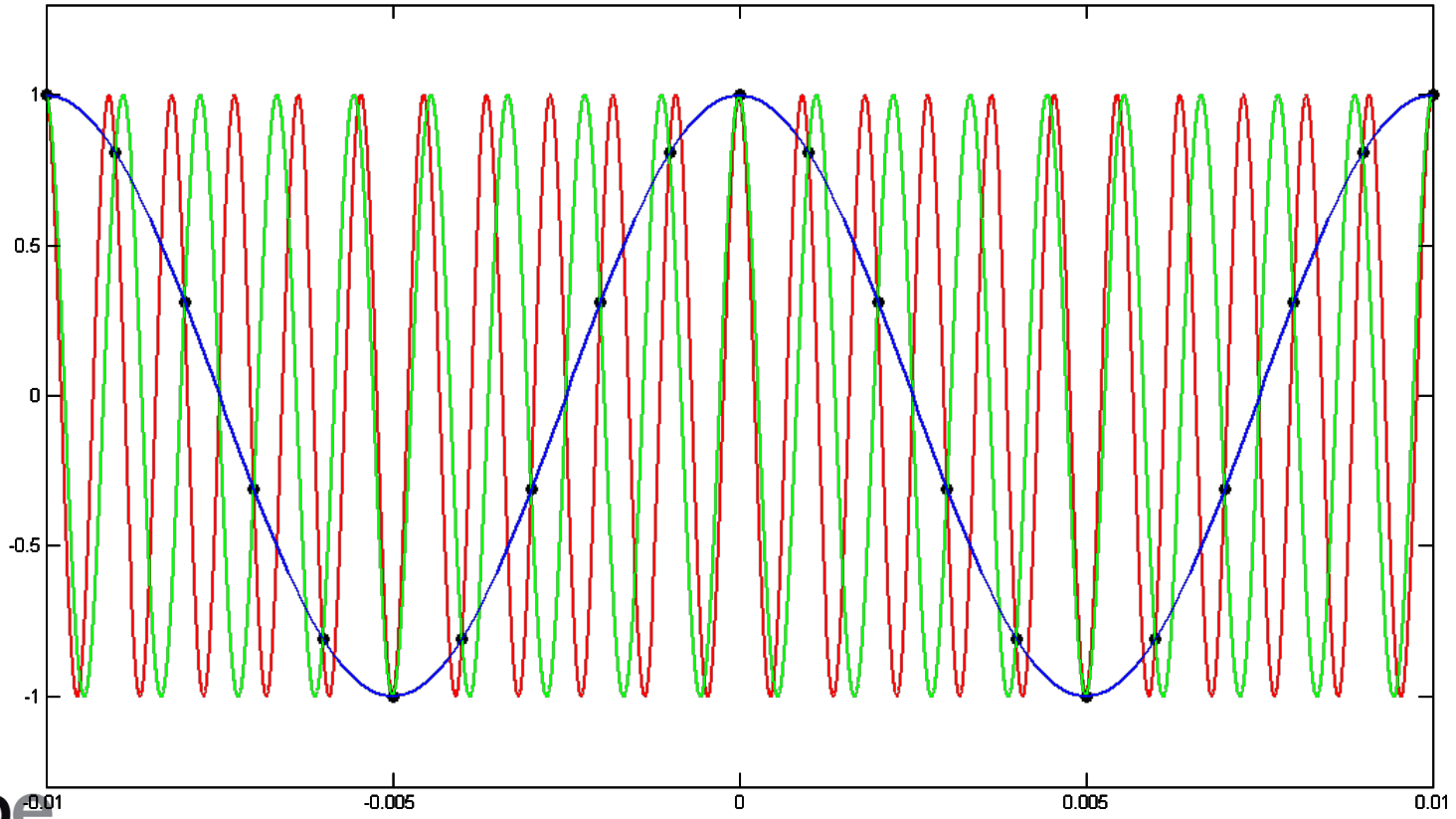
$$X(f)=0 \quad |f| \leq B \text{ Hz}$$

  - The sampling frequency must be at least twice the value of the signal bandwidth

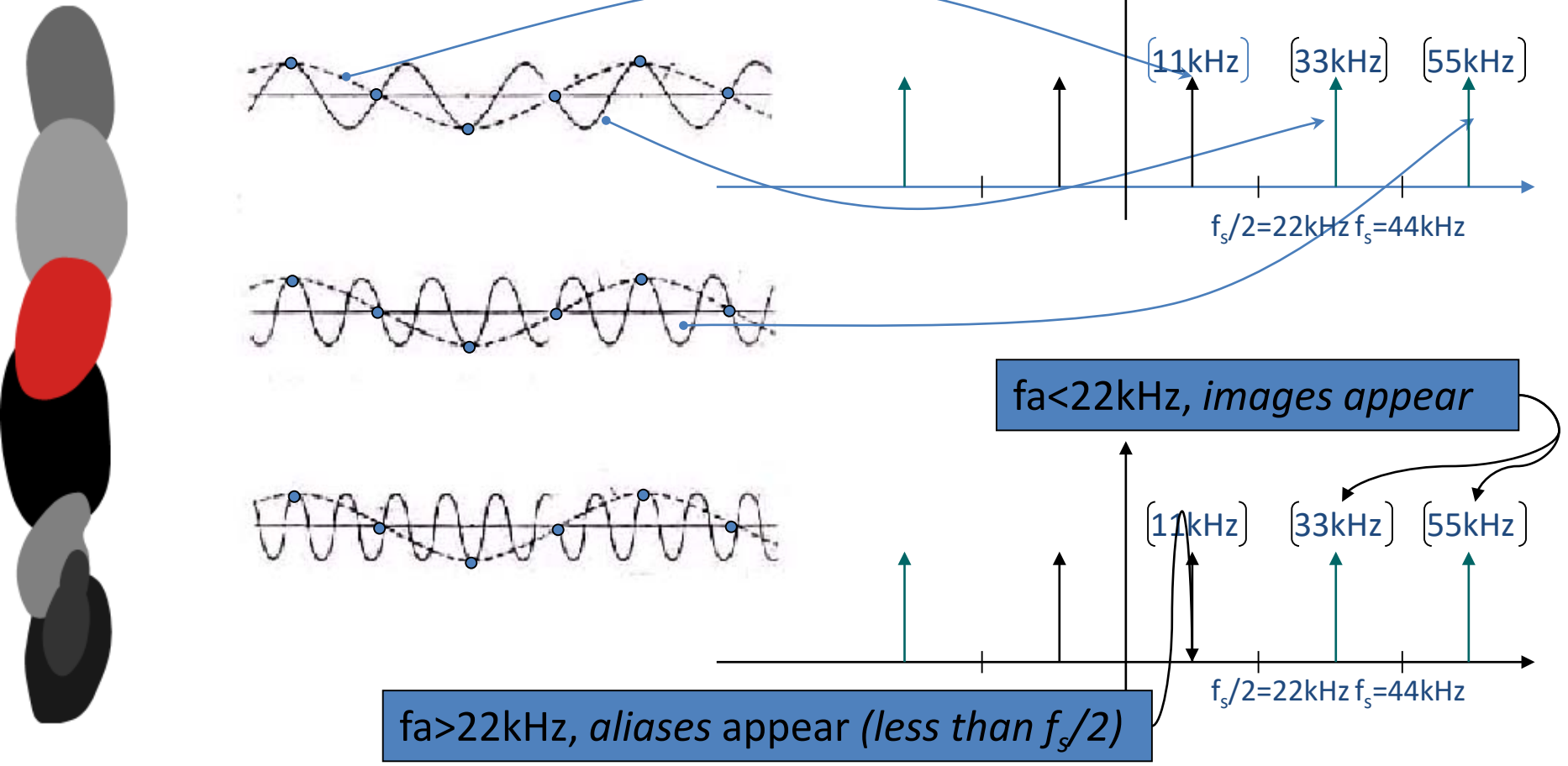
$$f_s \geq 2B \text{ Hz} (\textit{Nyquist frequency})$$

Sampling process

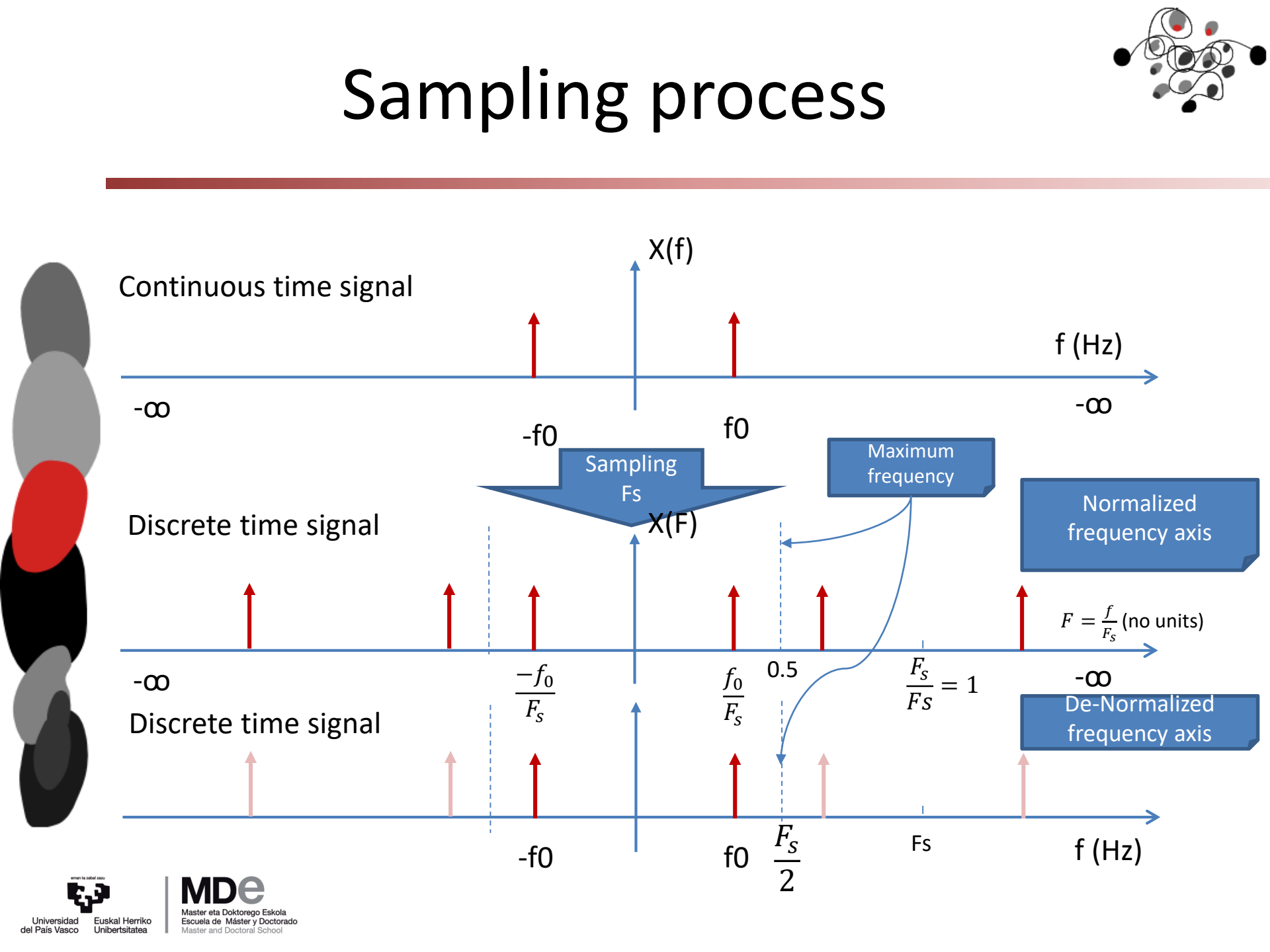
- A set of samples may correspond to different analog signals



Sampling process

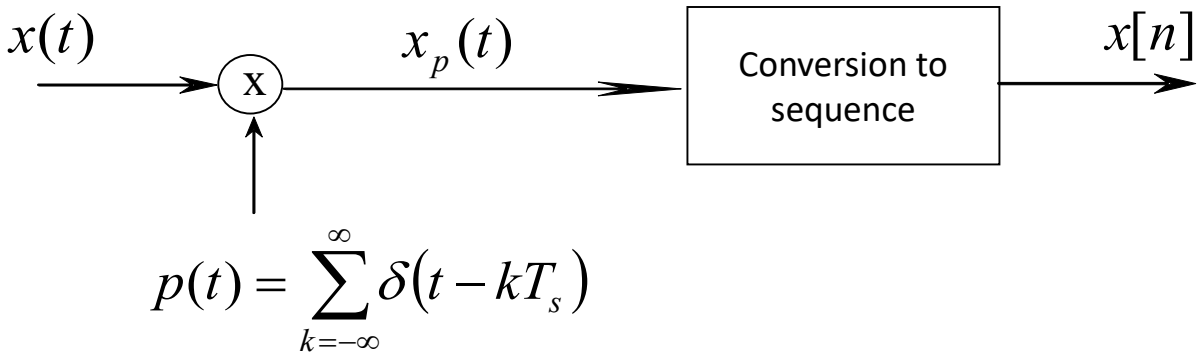


Sampling process



Sampling process

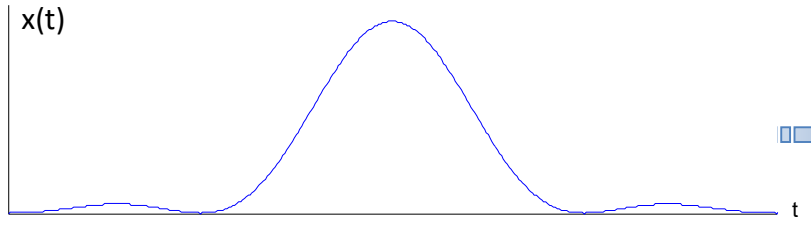
- Block representation of the sampling process



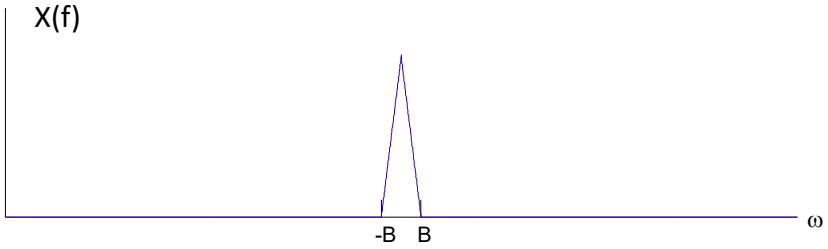
$$x_p(t) = x(t) \sum_{k=-\infty}^{\infty} \delta(t - kT_s)$$

$$x[n] = x(nT_s)$$

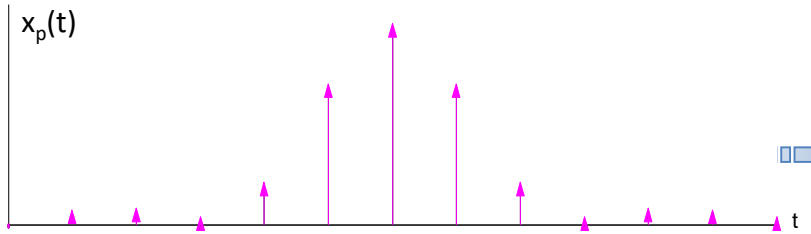
Sampling process



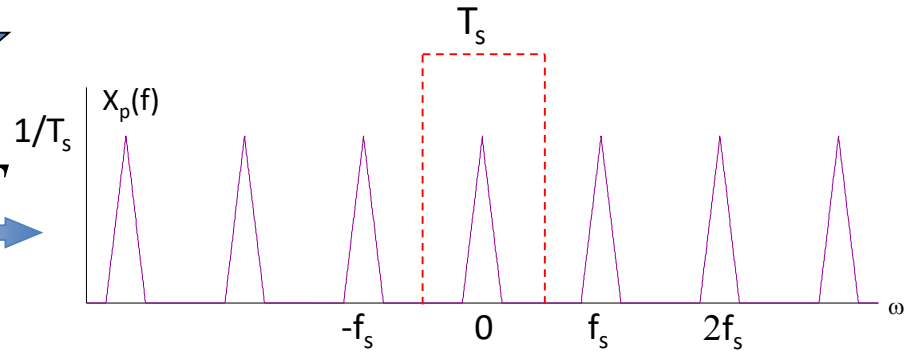
F



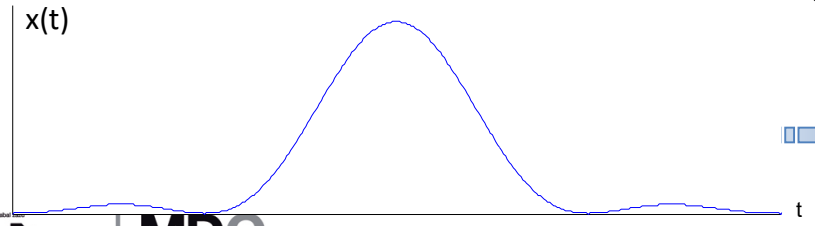
SAMPLING



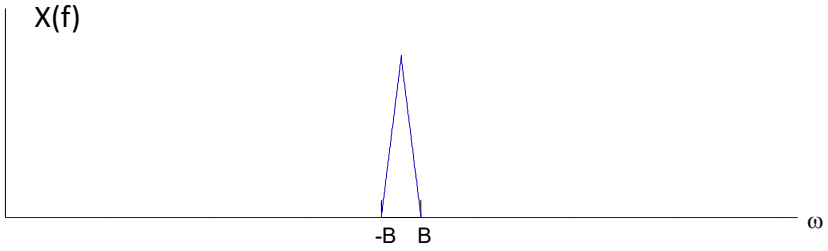
F



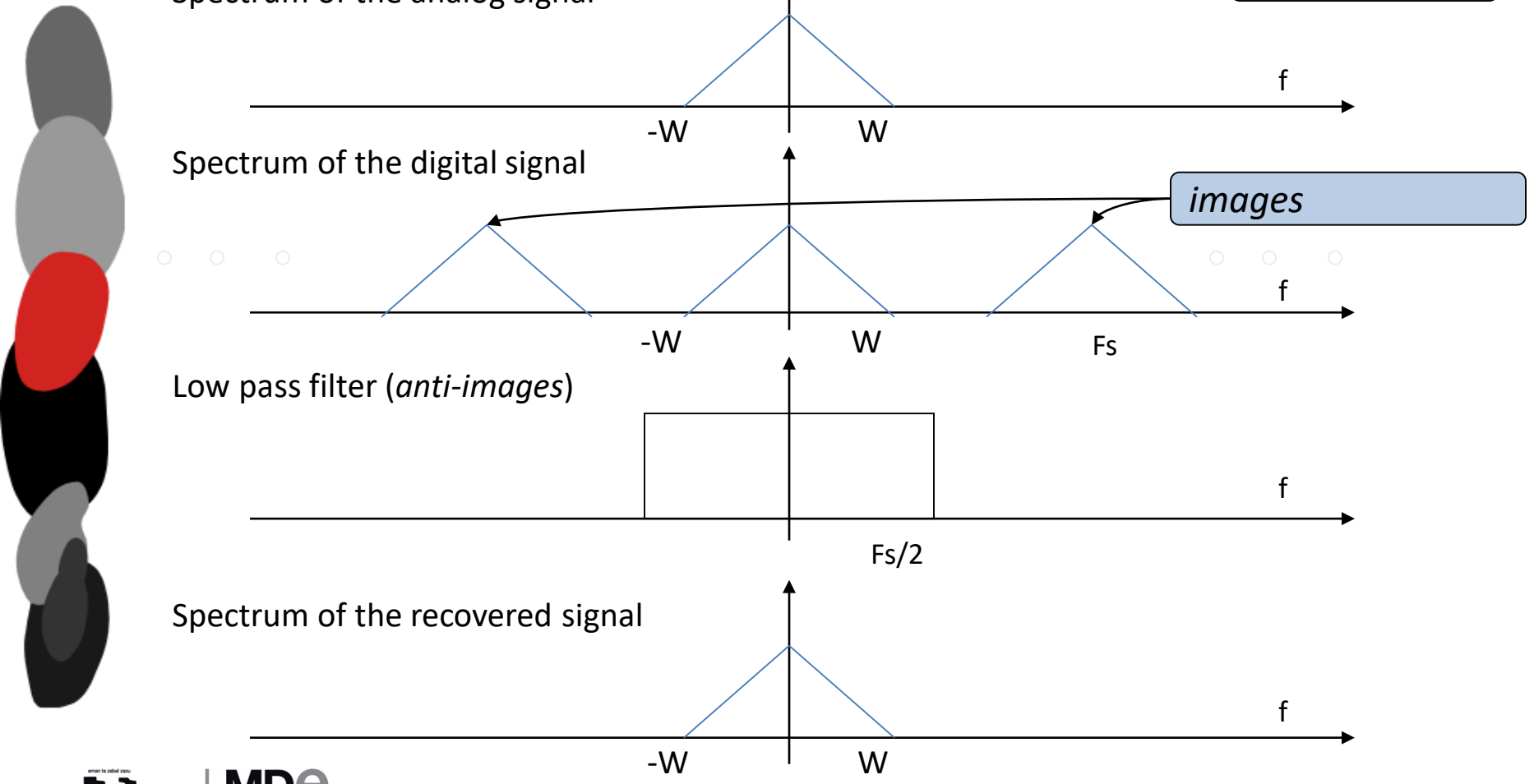
RECOVERING



F



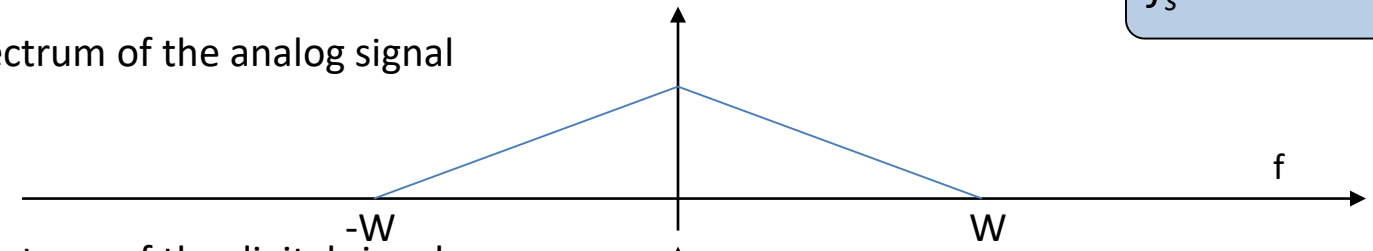
Sampling process



Sampling process

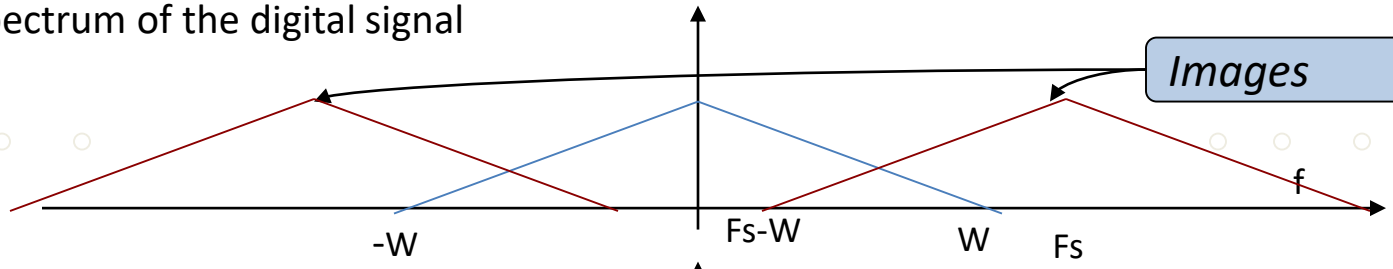
$f_s<2W$

Spectrum of the analog signal

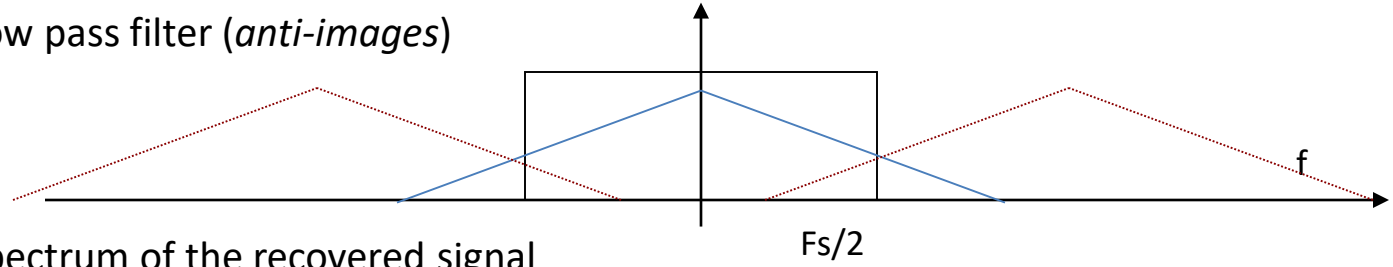


Spectrum of the digital signal

Images

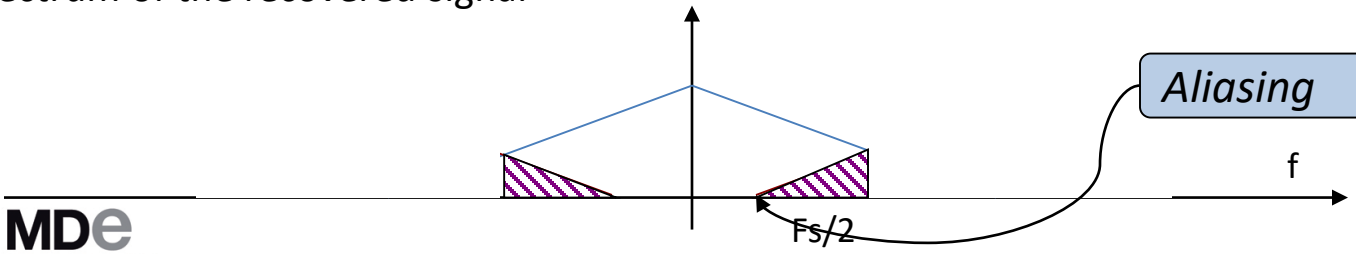


Low pass filter (*anti-images*)

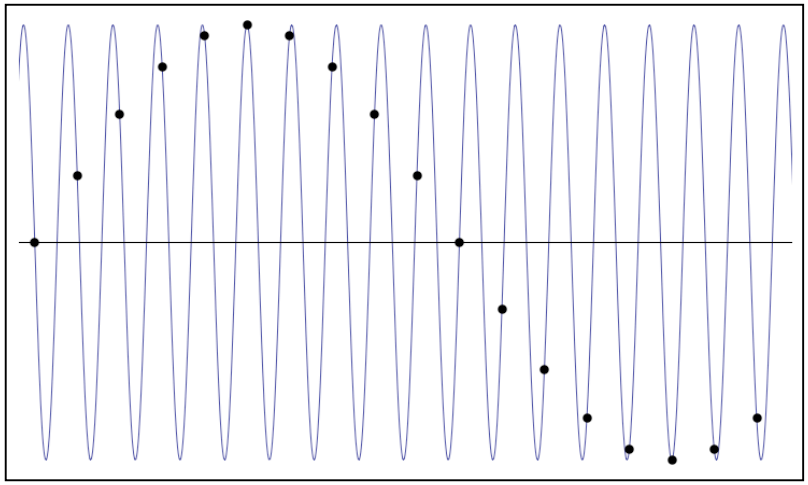


Spectrum of the recovered signal

Aliasing



Sampling



https://www.youtube.com/watch?v=xTo3gxsZOWo
One image each 55 minutes (aliasing in the hour hand)
http://www.youtube.com/watch?v=yvEAo6Z-g6k
The helicopter is flying
https://www.youtube.com/watch?v=A-19SxqZ8Qs
Rotating string
https://www.youtube.com/watch?v=SFbINinFsxk
Why wheels rotate backwards in films?

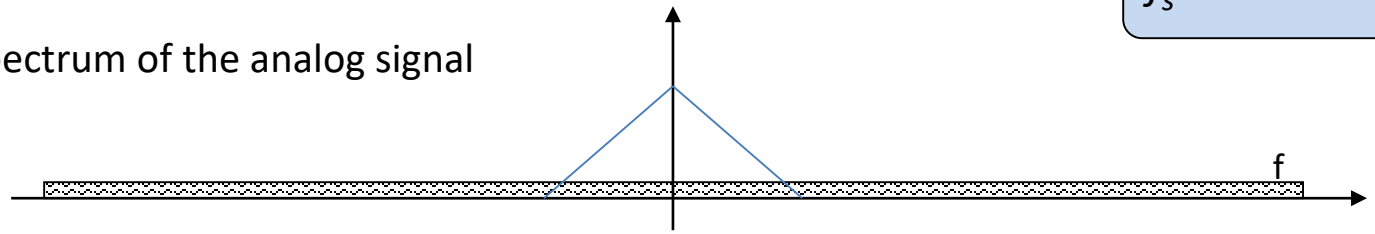
Aliasing

- If the sampling is applied without complying with the sampling theorem spectrums overlap (*aliasing*)
- Once aliasing is produced, the signal cannot be recovered from its samples

Antialiasing filtering

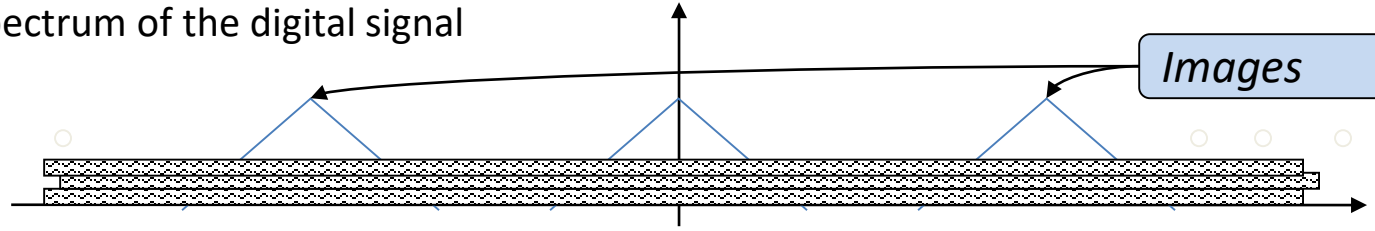
$f_s<2W$

Spectrum of the analog signal

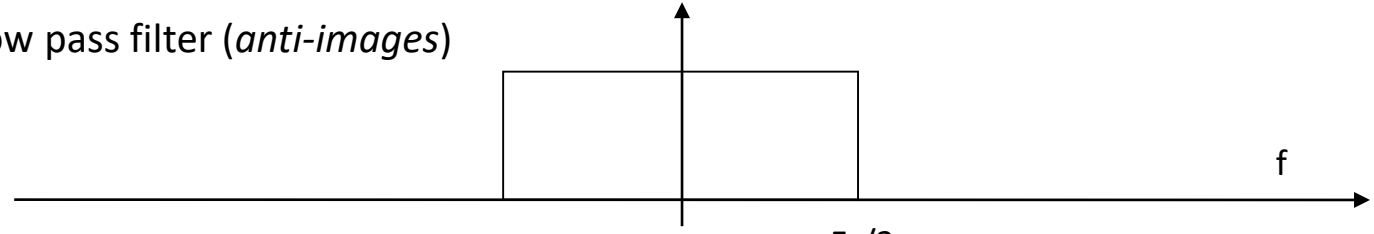


Spectrum of the digital signal

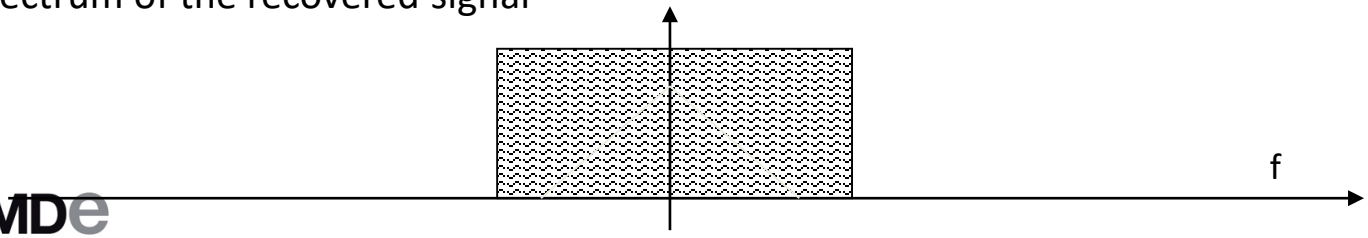
Images



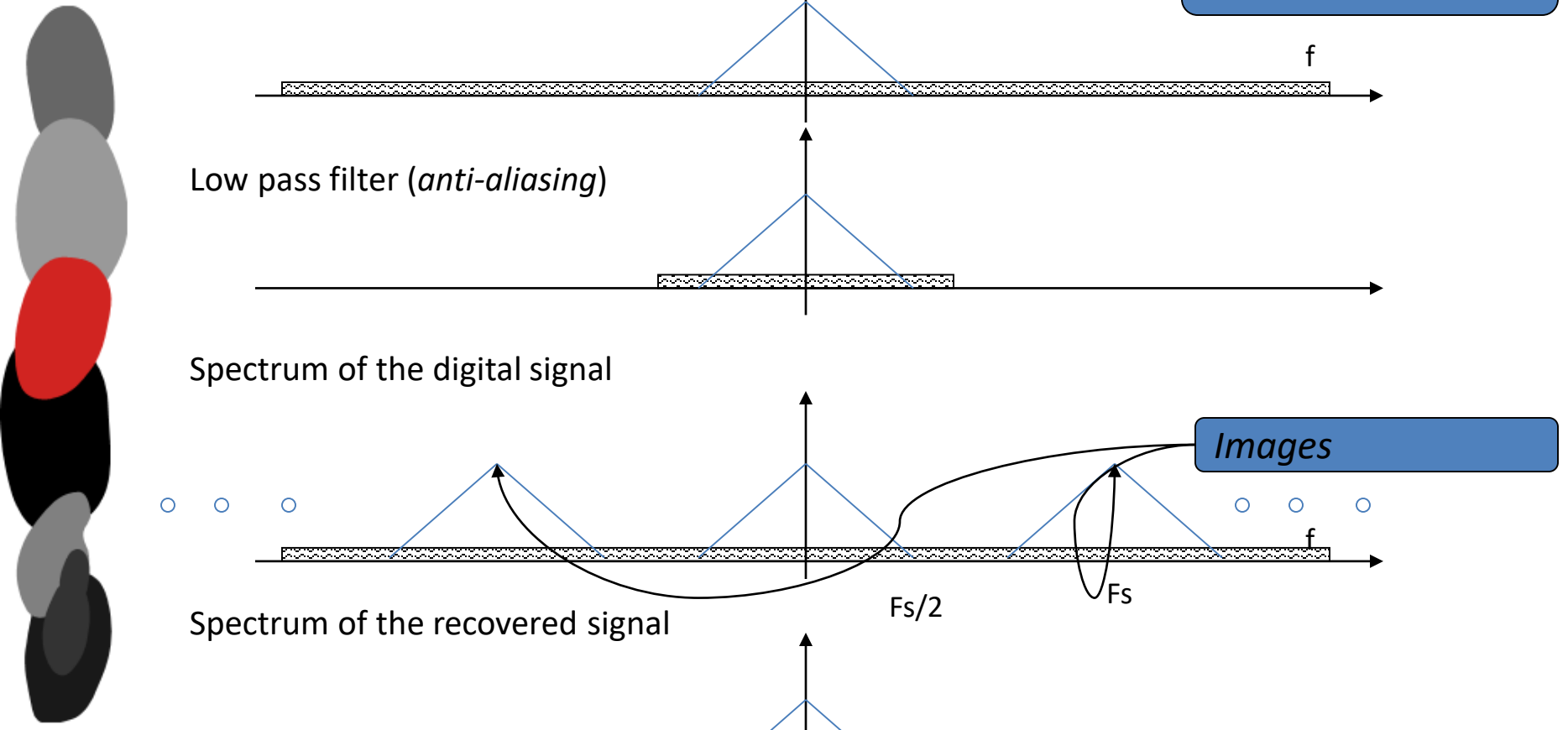
Low pass filter (*anti-images*)



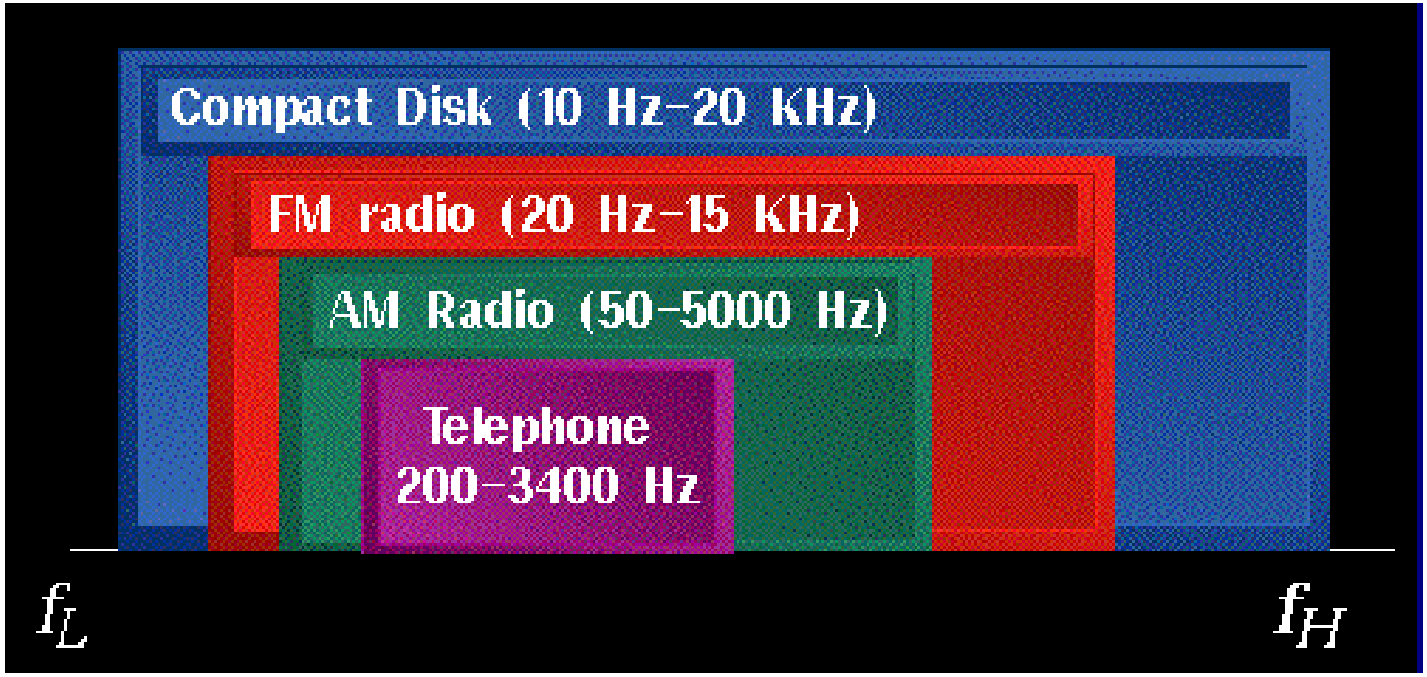
Spectrum of the recovered signal



Antialiasing filtering



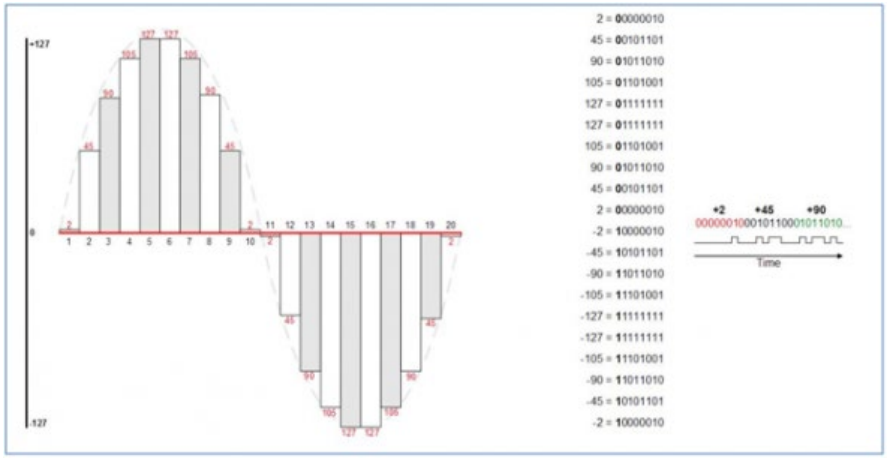
Sampling

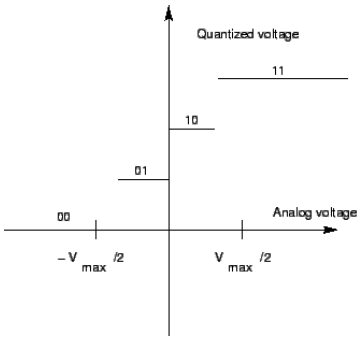


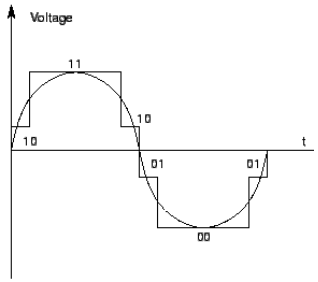
Telephone	Radio	Audio (DAT)	Audio CD	Audio Prof.
8kHz	16kHz	32kHz	44,1kHz	48kHz

Signal quantization

- To save signals the amplitudes have also to be digitized: samples are represented with a finite amount of bits.
- This process introduces error: *quantization noise*





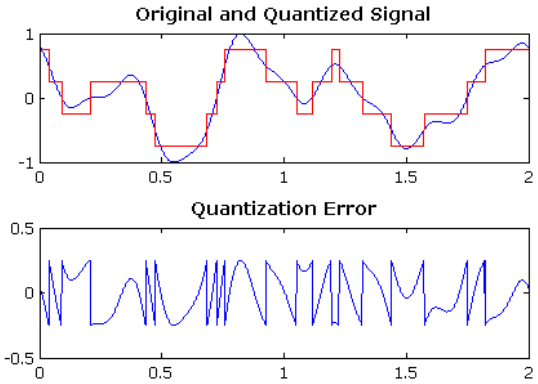


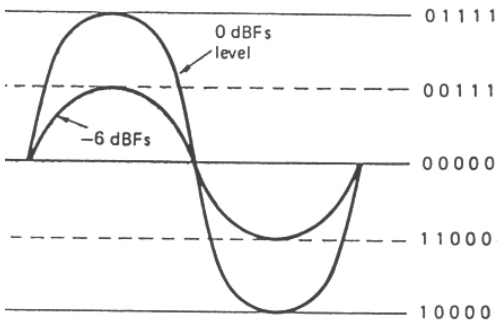
Signal quantization

- Quantization noise: limited to ½ LSB.
- It is white noise (scattered in all the spectrum), if the signals:
  - Fills all the range of the quantifier.
  - Has a complex spectrum.
- If the quantization is uniform: (all the intervals are equal)

  $S/N \sim 6\mathbf{n}+4.8-20\log K_L$

  - $K_L=x_{max}/\sigma_x$ (*crest factor*)
  - Maximum $S/N \sim 6n+4.8$
  - **1 bit more: +6dB improvement**
  - 16 bits (CD), sinusoid ($K_L=\sqrt{2}$): 98dB
  - 12 bits, speech ($K_L=\sqrt{7}$): 60dB





Signal quantization

- Bit rate (R): bit per second (transmission speed, bps, bit/s)
- Bit rate=f_s (*samples/s*)* n(*bit/sample*)
- Example:
  - Speech (telephone quality):
    - f_s=8000 samples/s (8kHz)
    - n=12bit/sample
    - R=8000*12 bps=96kbps.
    - Space needed to store one second 96000bit=96000/8 byte=12000byte=12000byte/1000 byte/kB=12kB.
    - Space needed to store one hour: 43.2 MB
  - Audio (CD quality)
    - f_s=44100 samples/s
    - n=16bit/sample
    - R=44100*16 bps=705600bps.
    - Space needed to store a second (one channel) 705600bit=705600/8 byte=88200byte=88200byte/1000byte/kb=88.2kB
    - Space needed to store a second (stereo) = 176.4 kB
    - Space needed to store one hour (stereo) = 635040 kb ~ 635 MB

Outline

1. Digitization
2. Supra-segmental acoustic features
3. Segmental acoustic features

Outline

1. Digitization
2. Supra-segmental acoustic features
3. Segmental acoustic features

Speech signal: representation

- Acoustic features:
  - Supra-segmental (units > phoneme)
    - Speech prosody: intonation, rhythm, pauses, intensity, voice quality, articulation degree…
    - Influenced by socio-cultural factors (“software”)
  - Segmental (units < phoneme)
    - Voice timbre: spectral info for each phoneme
    - Influenced by physical/physiological aspects (“hardware”)

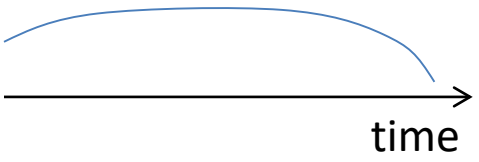
Supra-segmental features

- Related to “units” longer than the phoneme
- Known as prosody
- Main dimensions of prosody:
  - Intonation
  - Rhythm
  - Intensity
- Other dimensions:
  - Voice quality
  - Degree of articulation

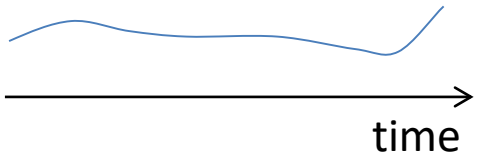
Supra-segmental features

- Intonation
  - The “melody” of voice
  - Characterized by the fundamental frequency evolution over time
  - There are multiple correct ways of intoning a sentence

How are you?

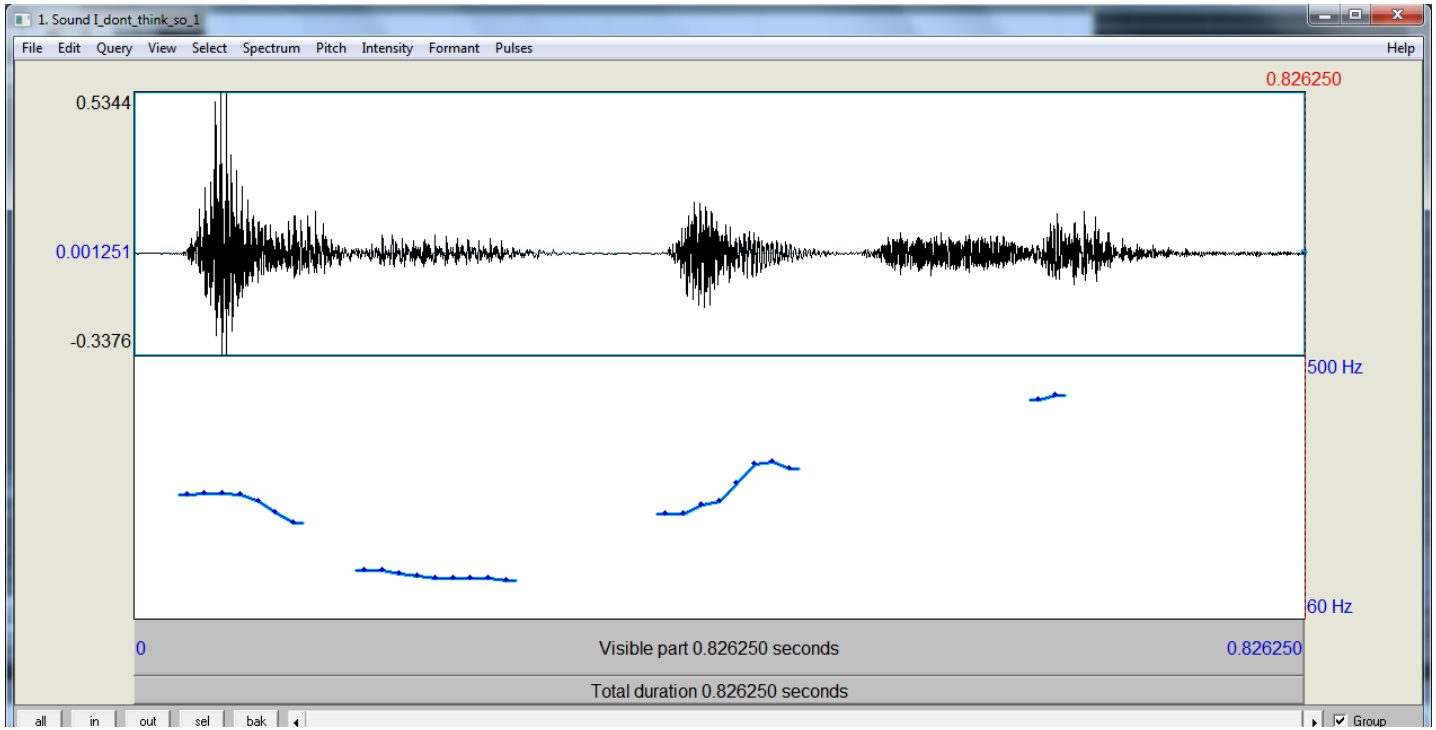


How are you?



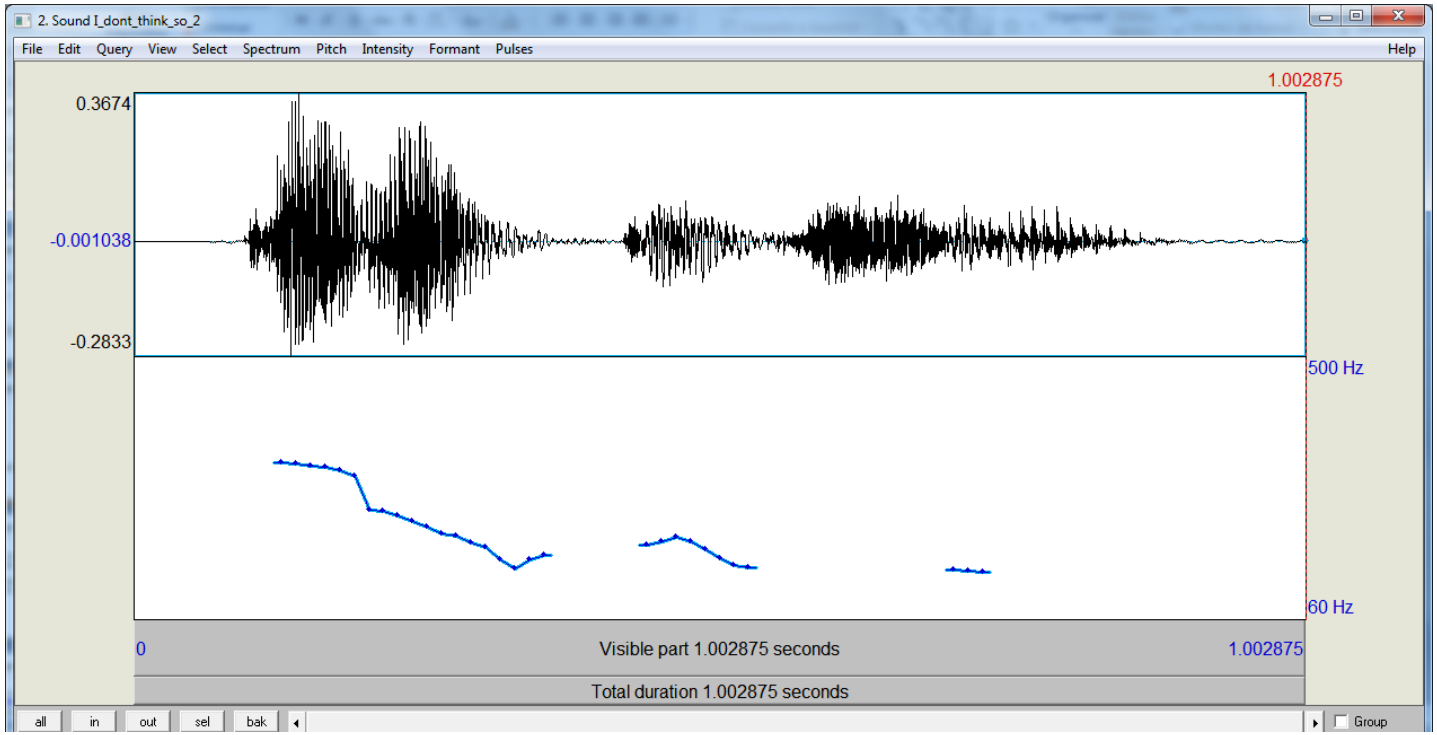
Supra-segmental features

- Intonation



Supra-segmental features

- Intonation

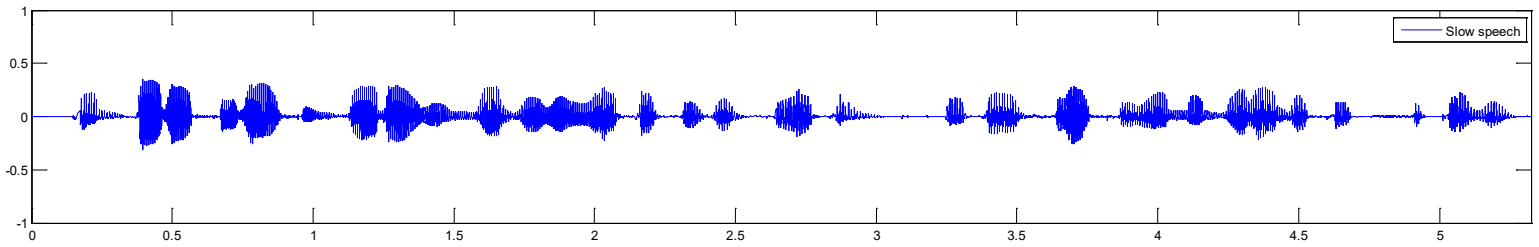


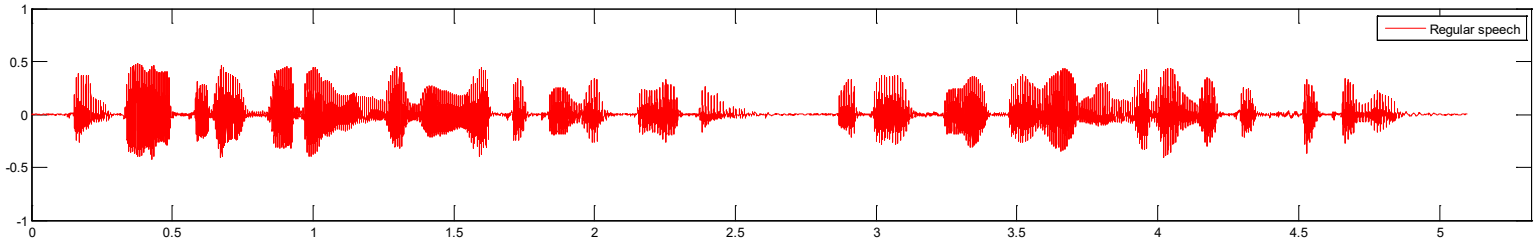
Supra-segmental features

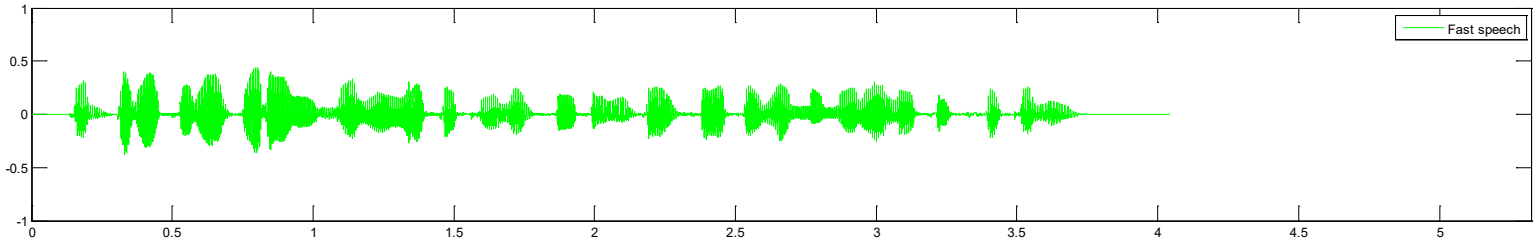
- Rhythm
  - The “speed” of speech
  - Not only fast/slow, but also variations within a sentence
  - How many pauses, where, and how long
  - Characterized by the duration of phonemes and pauses

Supra-segmental features

- Rhythm







Supra-segmental features

- Intensity
  - The “loudness” of speech
  - Characterized by the local signal power and its evolution over time

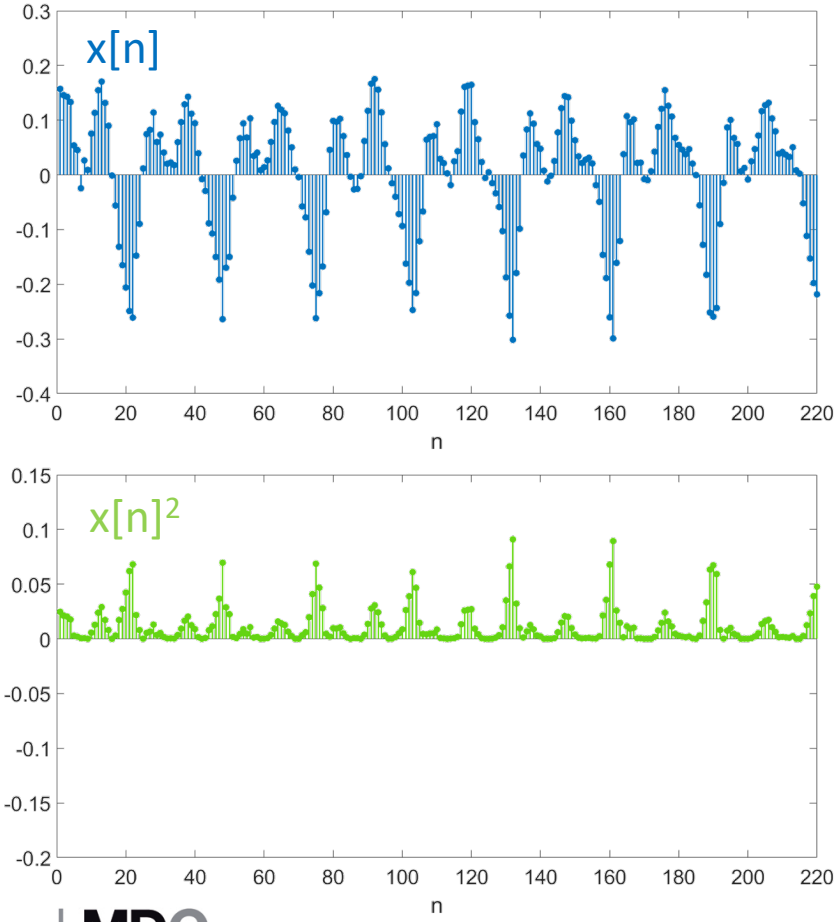
$$P[n] = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} s[k]^2 \qquad P_{dB}[n] = 10 \cdot \log_{10} P[n]$$

(revisited later)

  - P(speech) > P(silence) → Voice activity detection
  - P(voiced) > P(unvoiced) → Voiced/unvoiced decision

Supra-segmental features

- Intensity



$$P[n] = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} s[k]^2$$

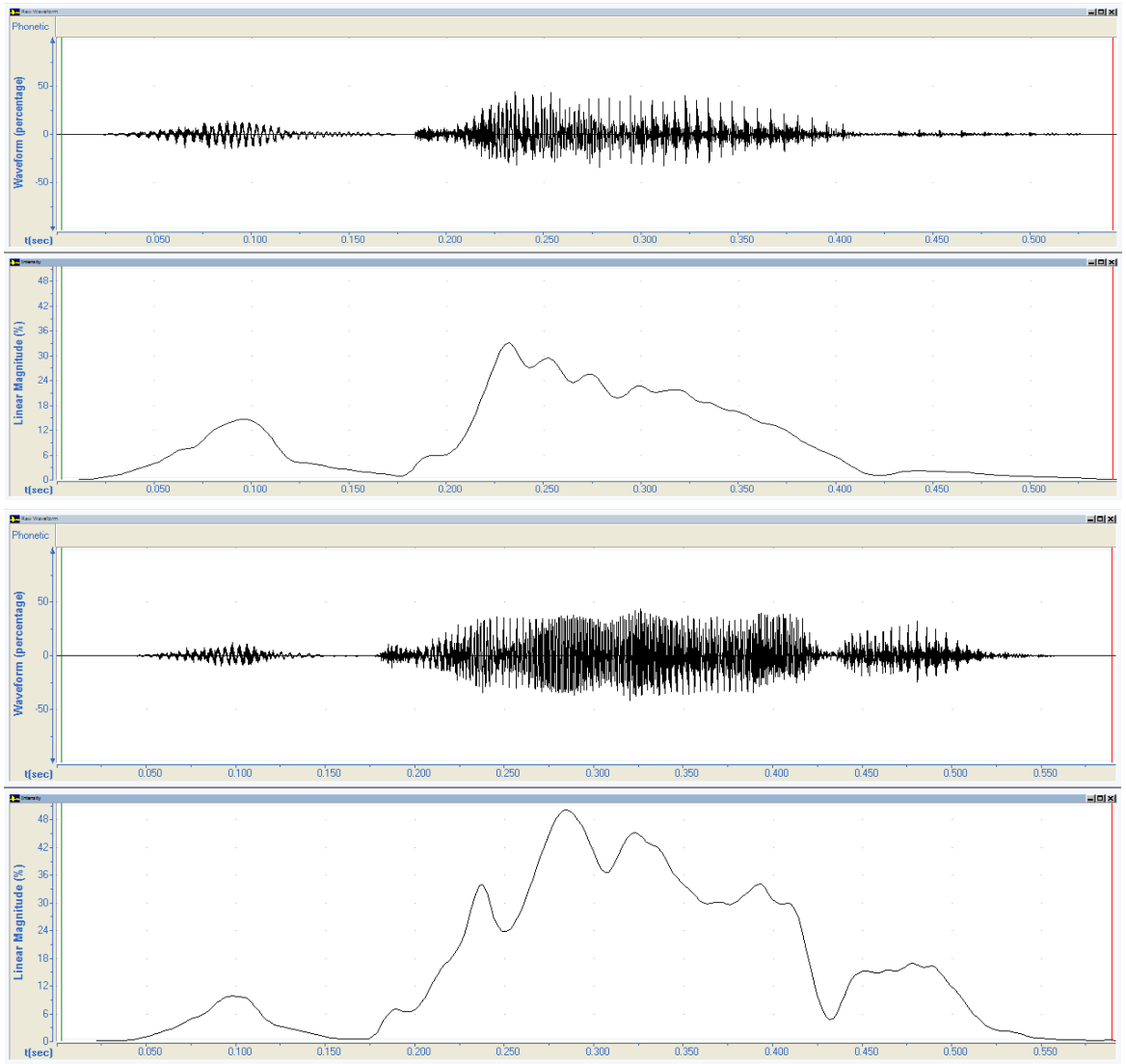
Sum(x[n]²)=2.8755

$$P_{dB}[n] = 10 \cdot \log_{10} P[n]$$

P=10.log$_{10}$(2.8755)=4.5871dB

Supra-segmental features

- Intensity



Supra-segmental features

- Voice quality
  - Related to the way of phonation
  - More difficult to characterize

- Degree of articulation
  - Hypo- vs. hyper-articulated speech
  - Also difficult to characterize

Outline

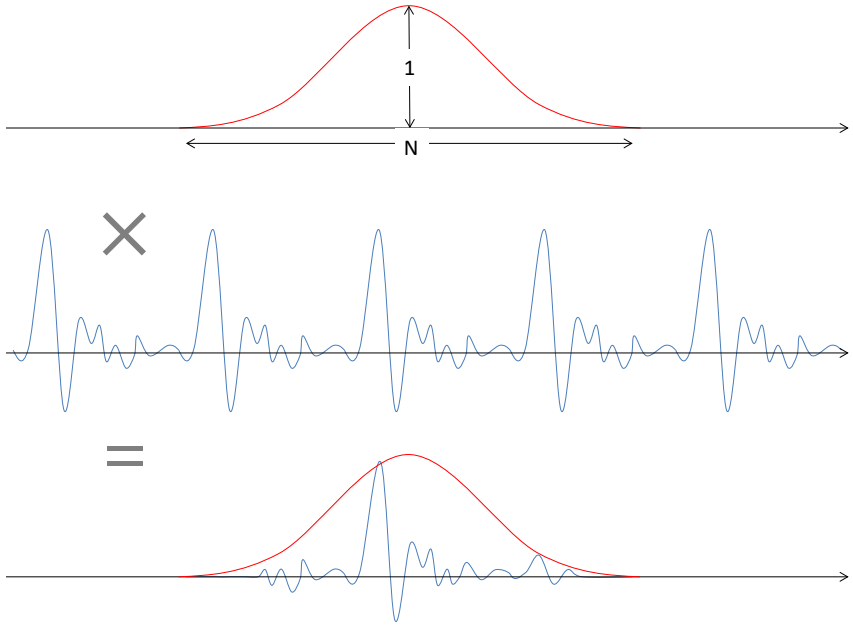
- Introduction
- Supra-segmental acoustic features
- Segmental acoustic features

Outline

- Introduction
- Supra-segmental acoustic features
- Segmental acoustic features
  - Windows
  - Short term Fourier Transform
    - Windowing effect
    - Spectrogram representation
    - Power Spectral Density
  - Cepstrum, MFCCs, mel filterbanks

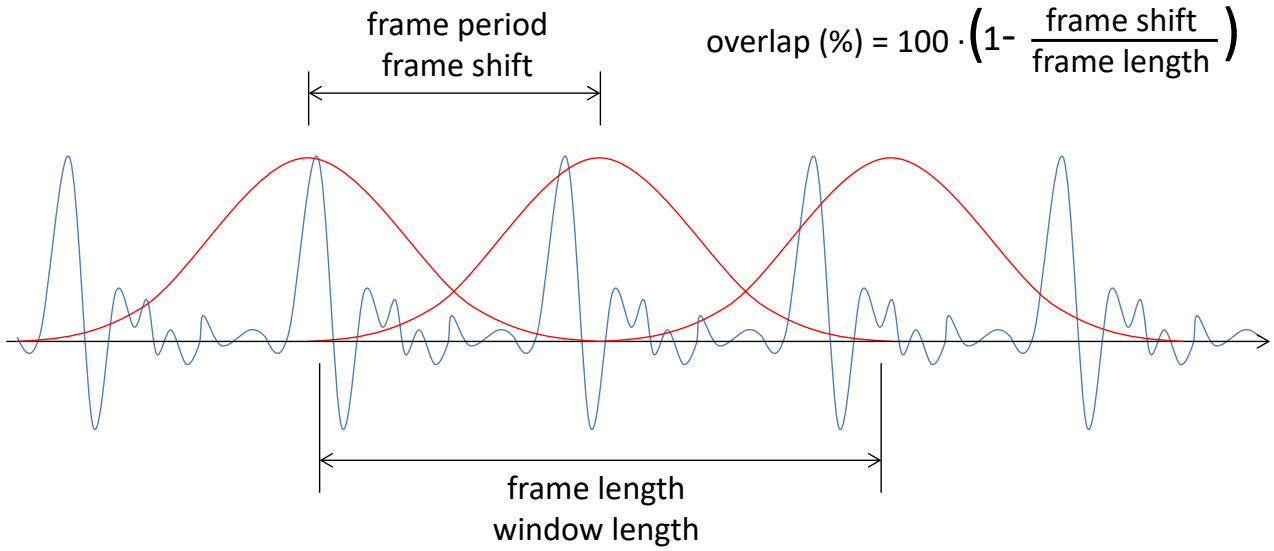
Segmental features

- Related to “units” shorter than the phoneme
- Speech signals are non-stationary → to extract reliable information, we must use short-time analysis



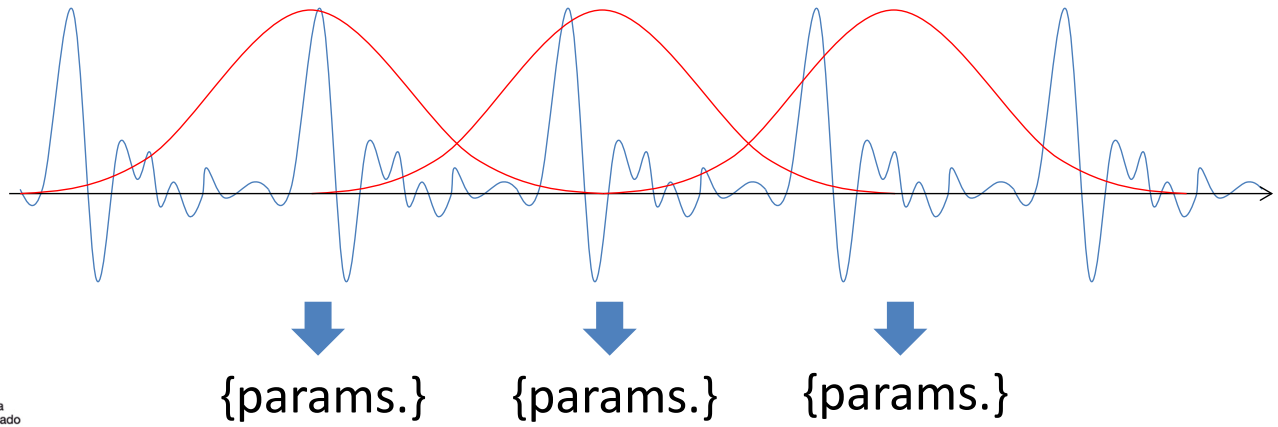
Segmental features

- Related to “units” shorter than the phoneme
- Speech signals are non-stationary → to extract reliable information, we must use short-time analysis



Segmental features

- Related to “units” shorter than the phoneme
- Speech signals are non-stationary → to extract reliable information, we must use short-time analysis



Windowing for supra-segmental features

- Intensity
  - The “loudness” of speech
  - Characterized by the local signal power and its evolution over time

use a window here!!

$$P[n] = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} s[k]^2$$

$$P_{dB}[n] = 10 \cdot \log_{10} P[n]$$

  - P(speech) > P(silence) → Voice activity detection
  - P(voiced) > P(unvoiced) → Voiced/unvoiced decision

Windowing for supra-segmental features

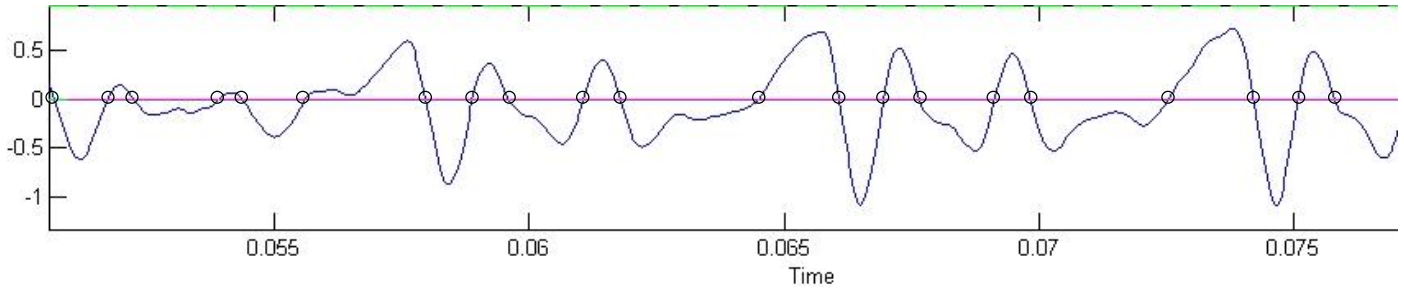
- Intensity
  - The “loudness” of speech
  - Characterized by the local signal power and its evolution over time

$$P[n] = \frac{1}{N}\sum_{k=0}^{N-1}(s[k]w[k-n])^2 \qquad P_{dB}[n] = 10log_{10}(P[n])$$

  - P(speech) > P(silence) → Voice activity detection
  - P(voiced) > P(unvoiced) → Voiced/unvoiced decision

Zero Crossing Rate

- ZCR: number of zero crossings per time unit



$$ZCR(n) = \frac{1}{N} \sum_{m=-\frac{N}{2}}^{\frac{N}{2}} \frac{1}{2} \left| \text{sgn}(x[m]) - \text{sgn}(x[m-1]) \right| w(n-m)$$

- ZCR(voiced) < ZCR(unvoiced) → V/UV decision
- ZCR(silence) < ZCR(speech) → VAD
  - Voiced: ~1400 zcr/s
  - Unvoiced: ~4900 zcr/s

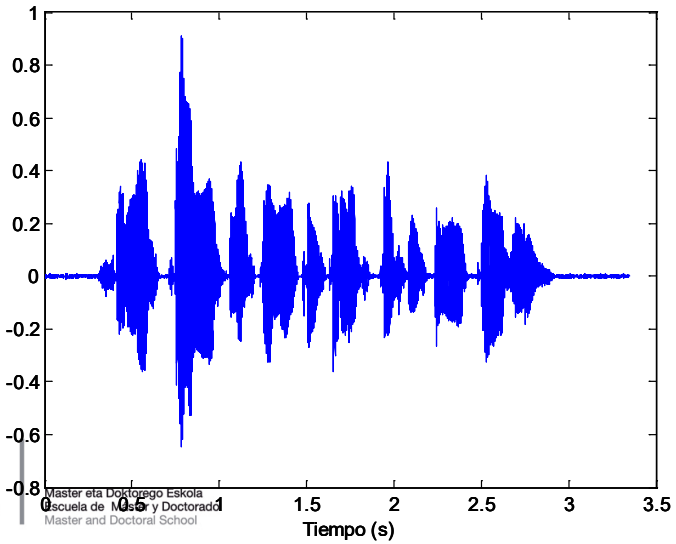
Supra-segmental features

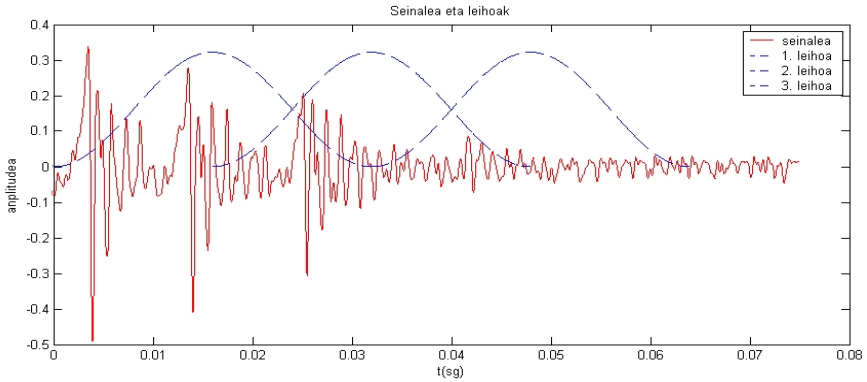
- Intensity and ZCR
  - Silence: Intensity low, ZCR low to medium
  - Voiced speech: Intensity high, ZCR Low
  - Unvoiced speech: Intensity low to medium, ZCR high

Guided exercise: plot power and ZCR contours of a speech signal in Matlab®. How can we implement a basic voiced/unvoiced classifier?

Short-time Fourier Transform

- If the signal is not stationary its spectral characteristics change with time
  - Localized Fourier transform is applied (*Short Time Fourier Transform*)
  - Selection of time segment is made with a window to reduce border effects



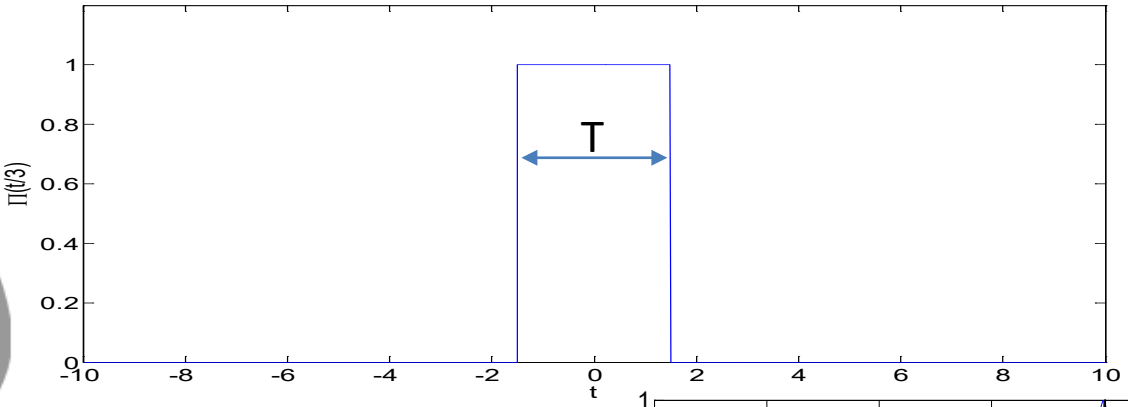


Windowing property of the Fourier transform

- Multiplication

$$x_1(t)x_2(t) \xrightarrow{F} X_1(f)*X_2(f)$$

Some usual Fourier transforms

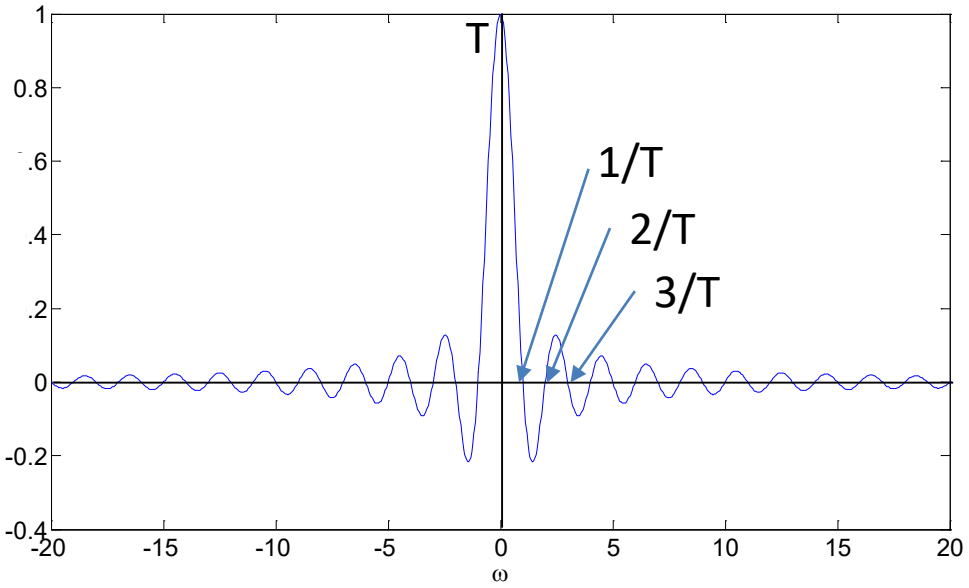


Time signal

$x(t)=\Pi(t/T)$

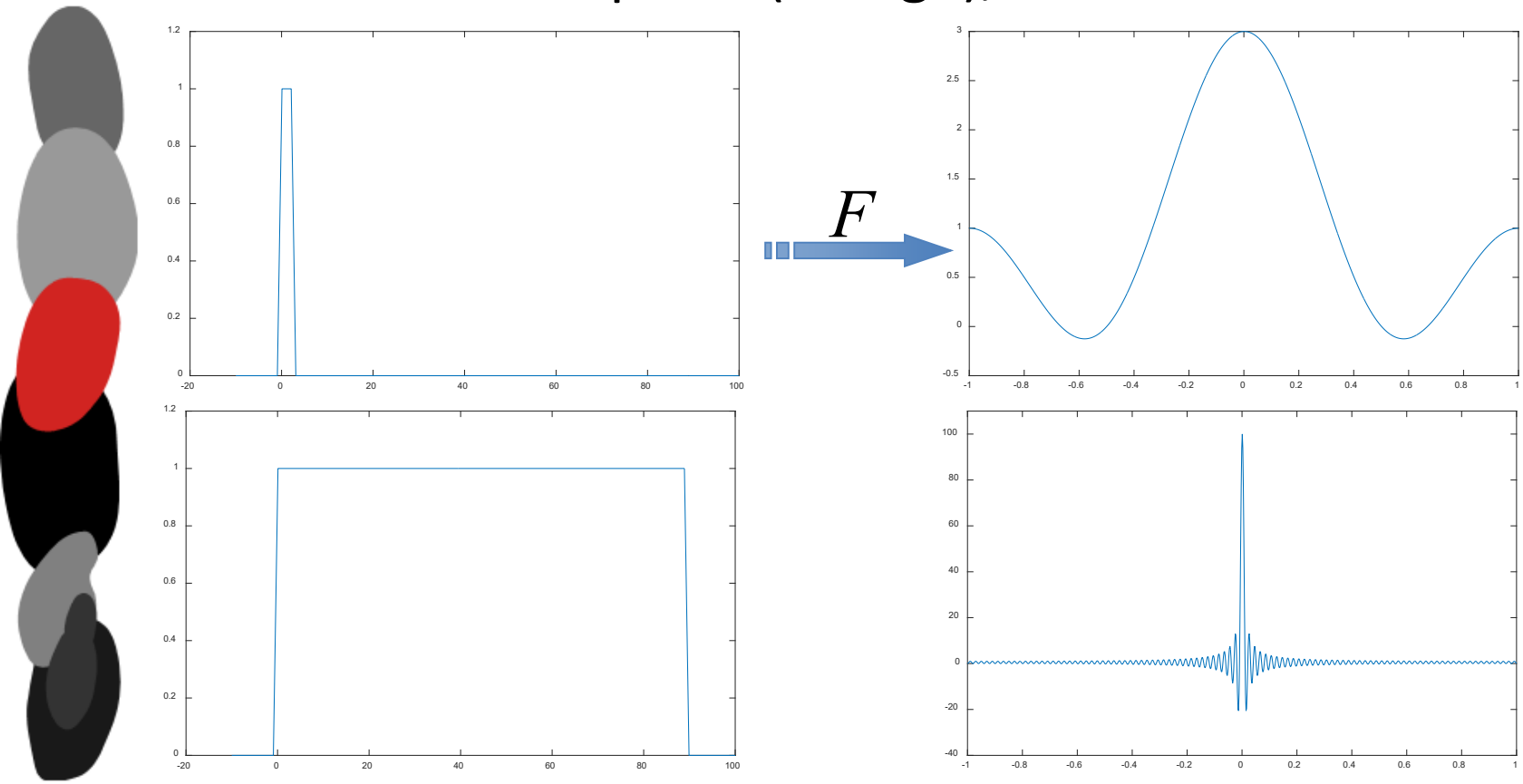
Transform: sinc(x) function

$$X(f)=T\,\frac{\sin(\pi fT)}{\pi fT}$$



Some usual Fourier transforms

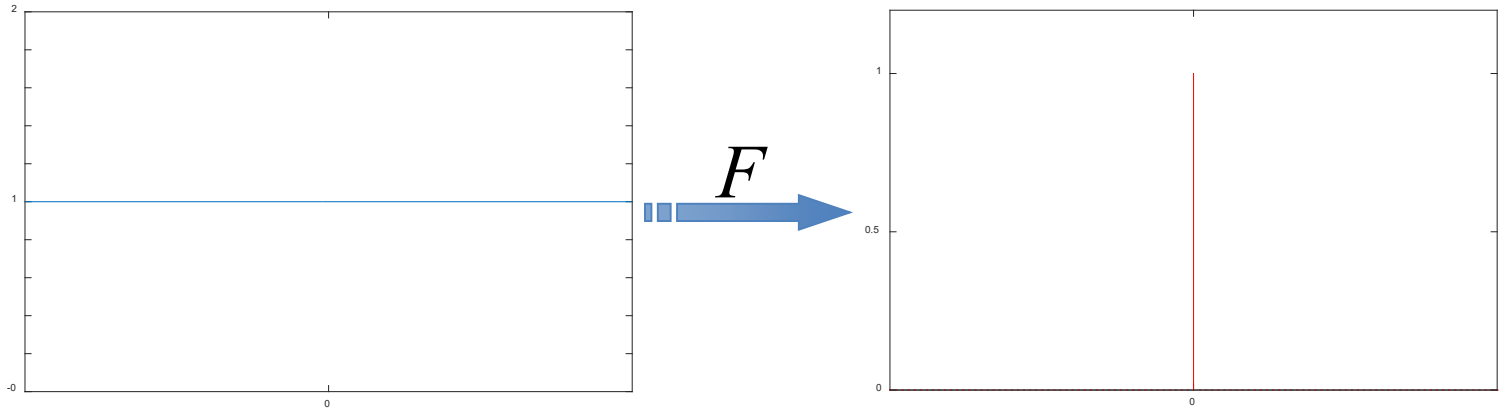
- The wider the pulse (T large), the narrower the sinc.



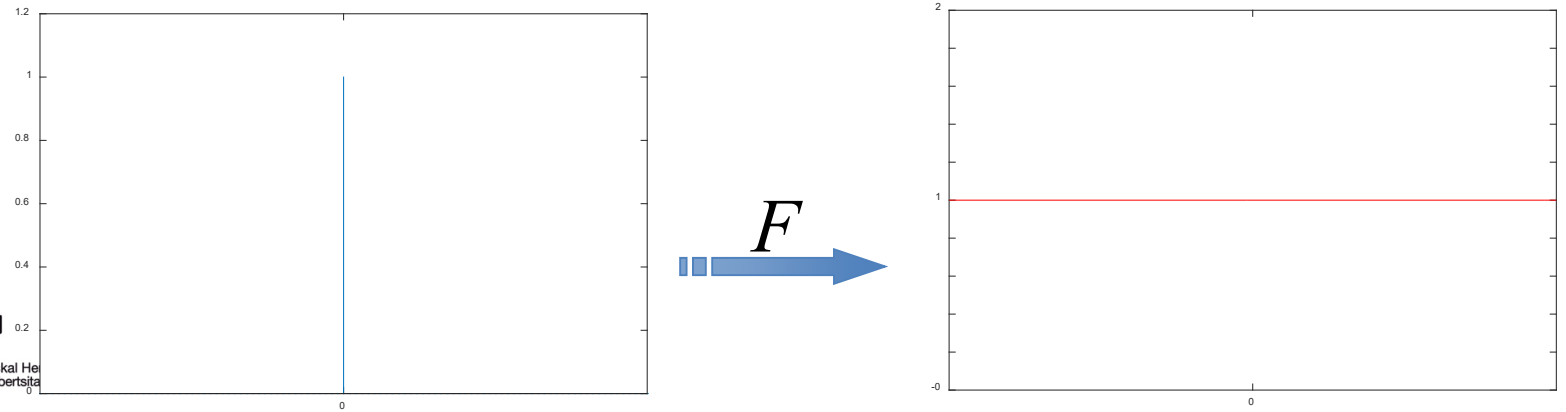
- In the limit when T-> ∞ we get a delta in frequency δ(f).

Some usual Fourier transforms

- The FT of y(t)=1 (pulse of T= ∞) is δ(f)



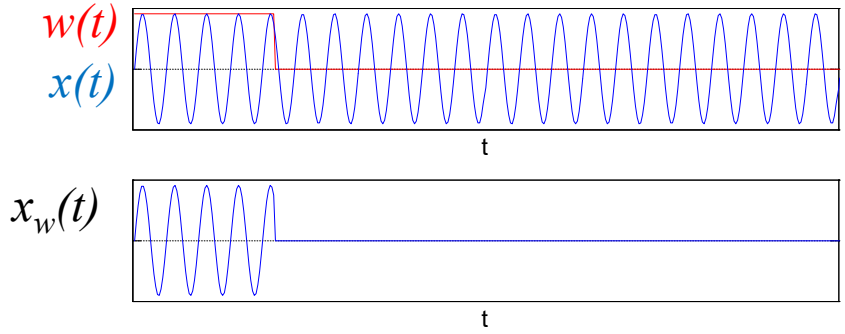
- Duality property: if y(t)<->Y(f) then Y(t)<->y(-f), so the FT of δ(t) is Y(f)=1 for every frequency



Short-time Fourier Transform

- To select finite length segments in a signal finite duration windows are applied:

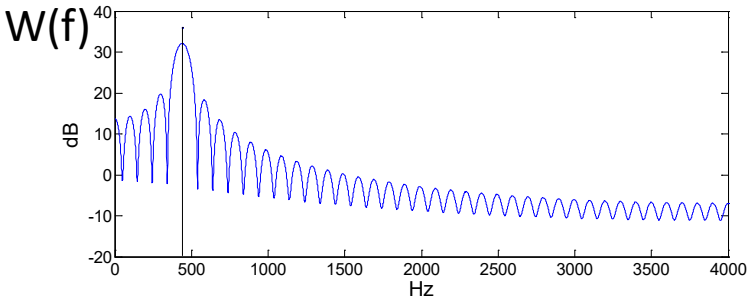
$$x_w(t) = x(t)\,w(t)$$



- The multiplication property of the FT:

$x_1(t)x_2(t)$ $\xrightarrow{F}$ $X_1(f)*X_2(f)$

$$X_w(f) = X(f) * W(f)$$

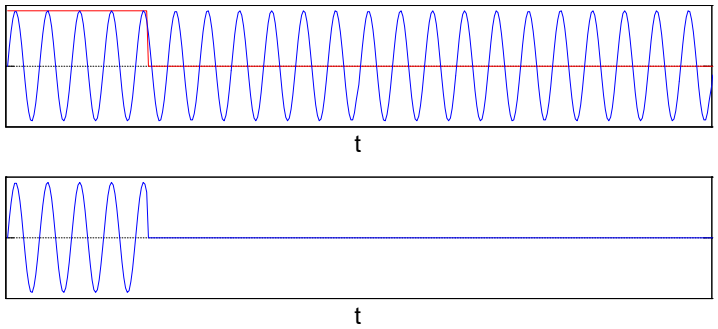


Short-time Fourier Transform

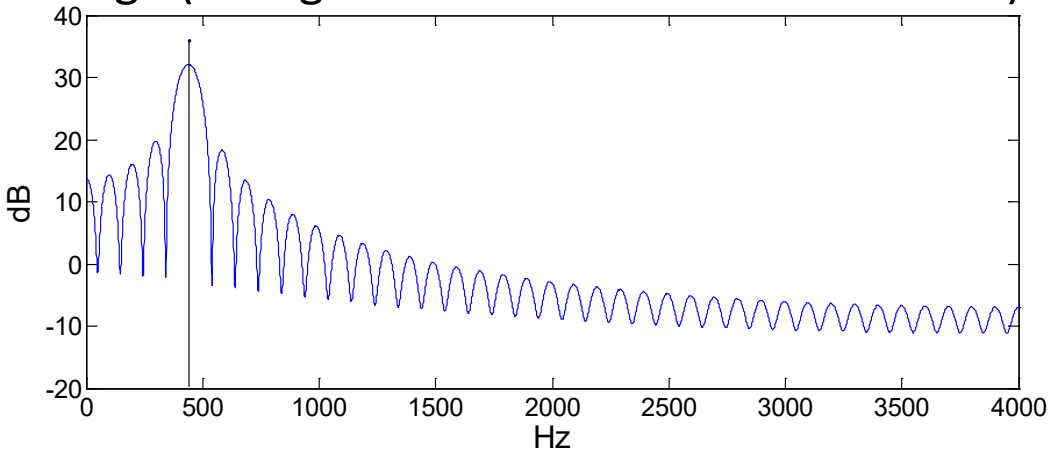
- To select finite length segments in a signal finite duration windows are applied.

$$x_w(t) = x(t)\,w(t)$$

$$X_w(f) = X(f) * W(f)$$

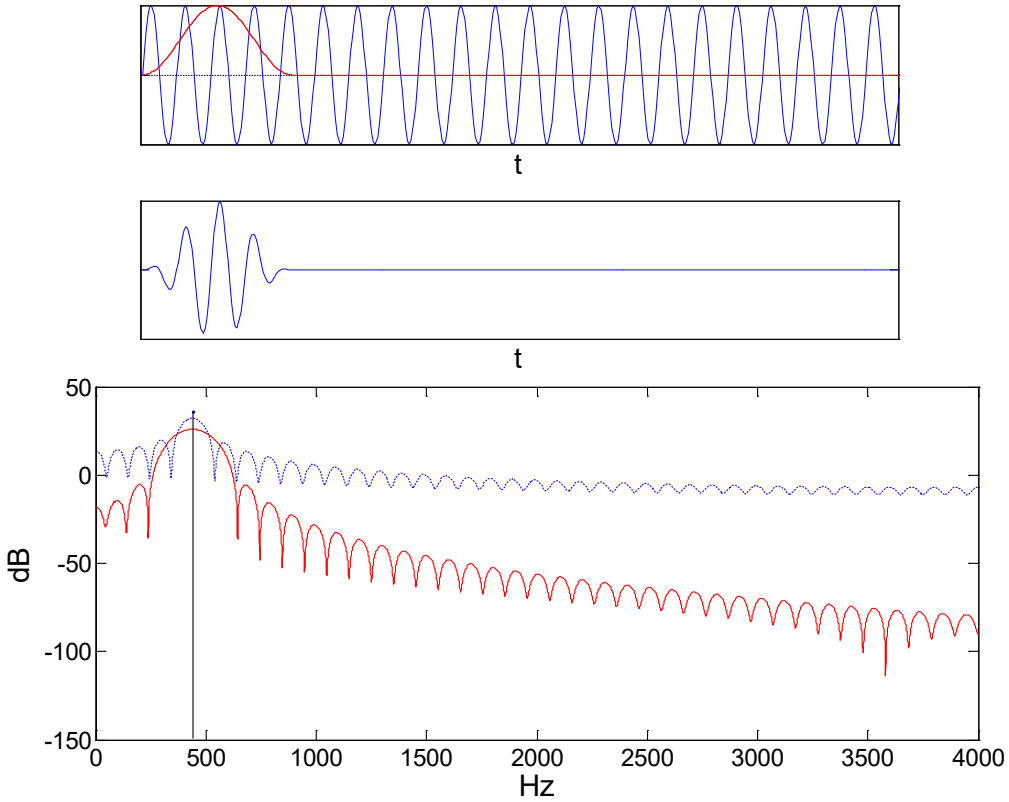


- The windowing process has effects in FT:
  - Frequency Resolution (worse when the window is shorter in time)
  - Spectral leakage (changes with the form of the window)



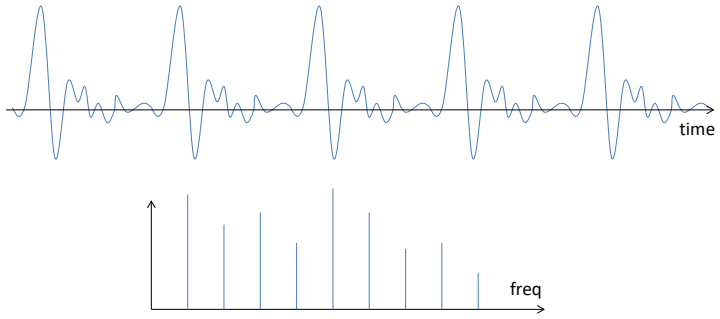
Short-time Fourier Transform: window effects

- Special window functions are used to reduce its effect in frequency (Hanning window for instance)



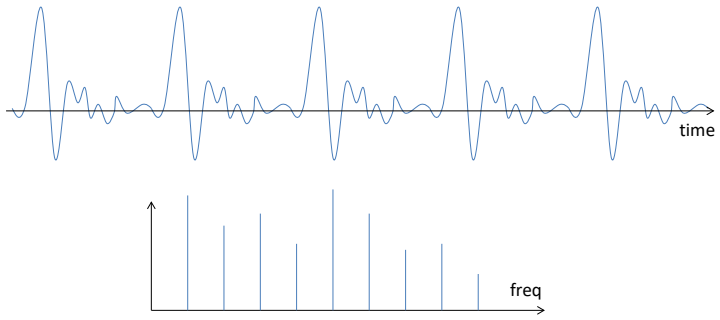
Segmental features

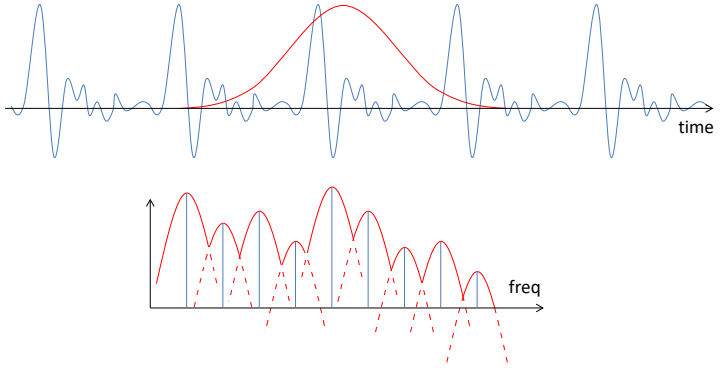
- How short?



Segmental features

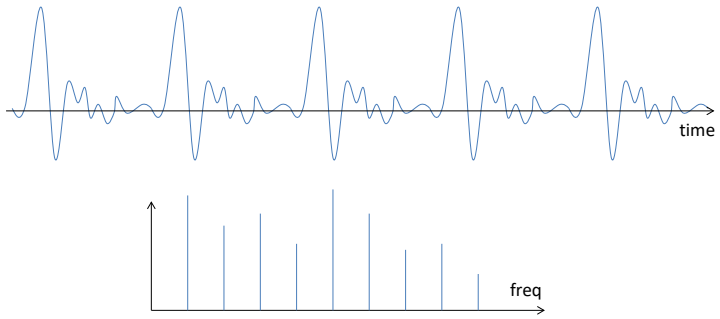
- How short?

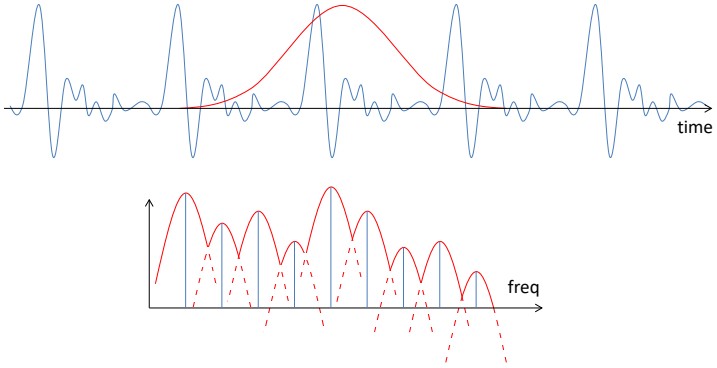


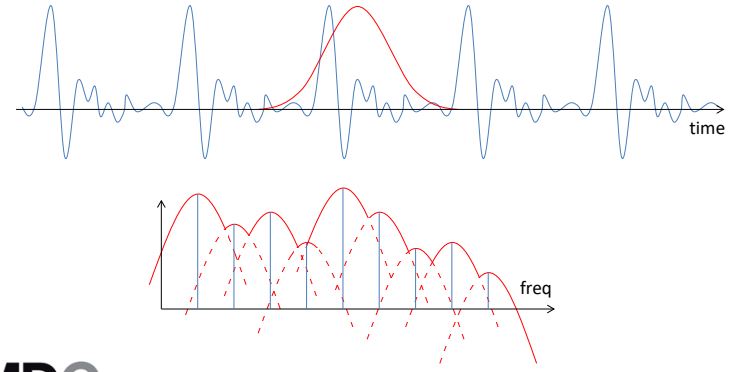


Segmental features

- How short?

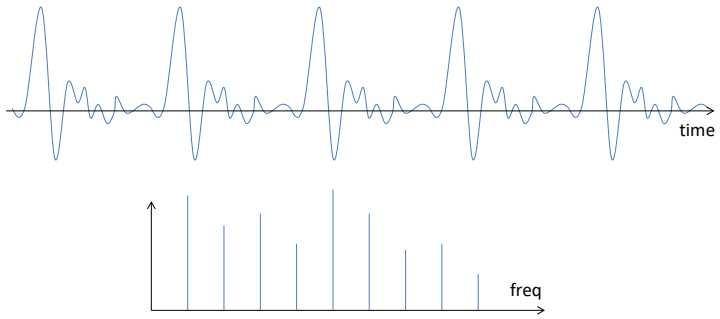


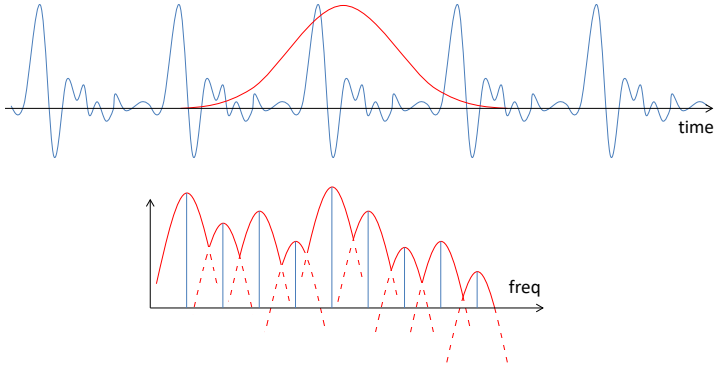


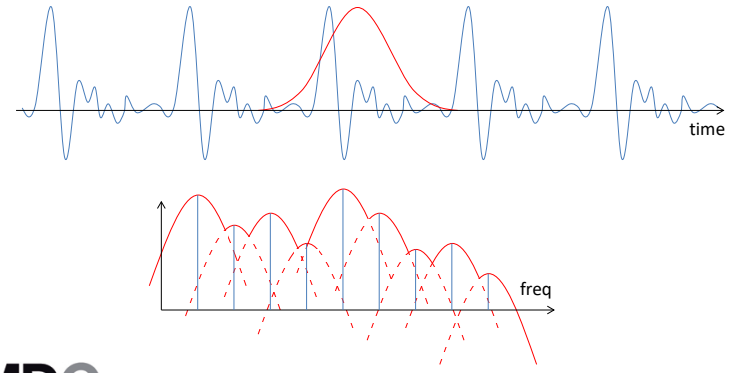


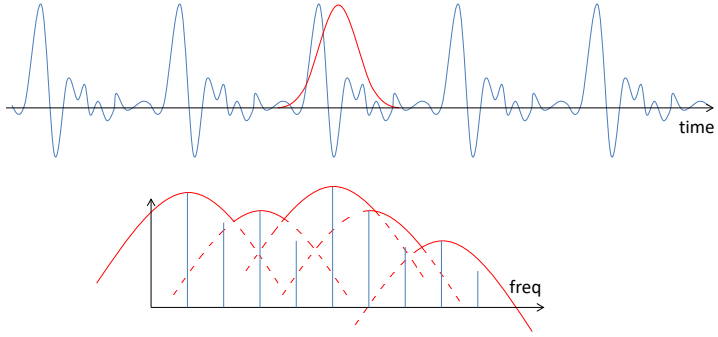
Segmental features

- How short?









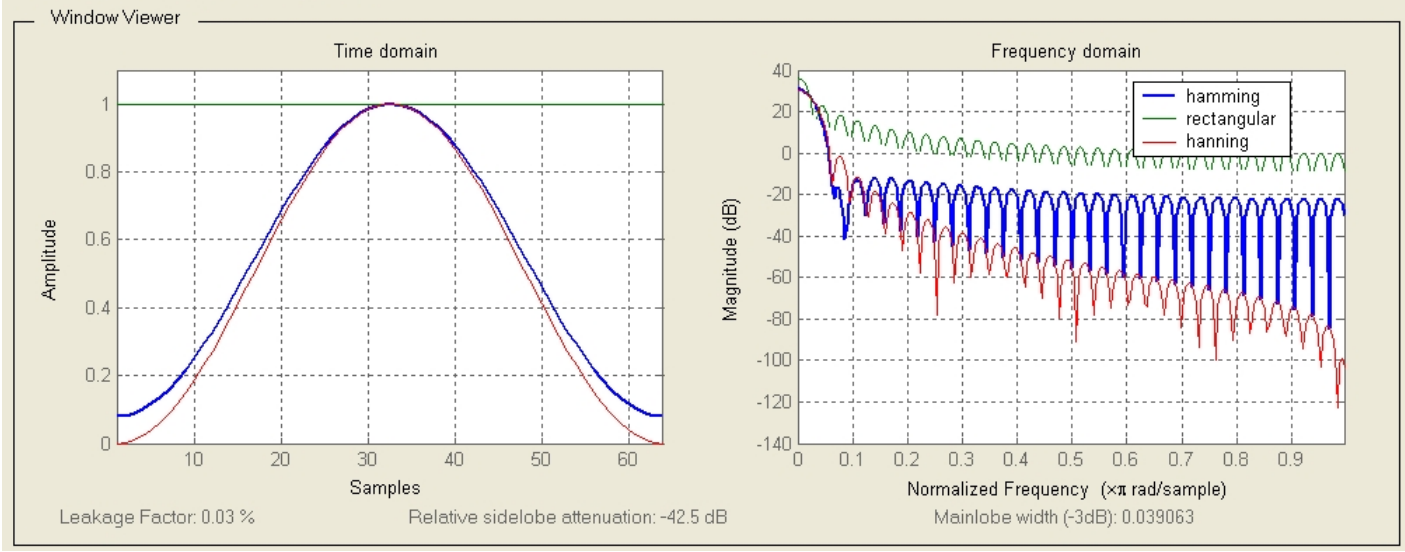
Short-time Fourier Transform: window effects

- The width of the main lobule of the window (bandwidth) limits the frequency resolution:
  - Rectangular window $\Rightarrow$ $\Delta f_{max} \sim 1/T$
  - Other windows $\Rightarrow$ $\sim \Delta f_{max} \sim 2/T$

- The second level lobules produce losses (spectral leakage)
- Ideal window:
  - High frequency resolution: main lobule narrow and sharp
  - Low losses: second level lobules with high attenuation

Short-time Fourier Transform: window effects

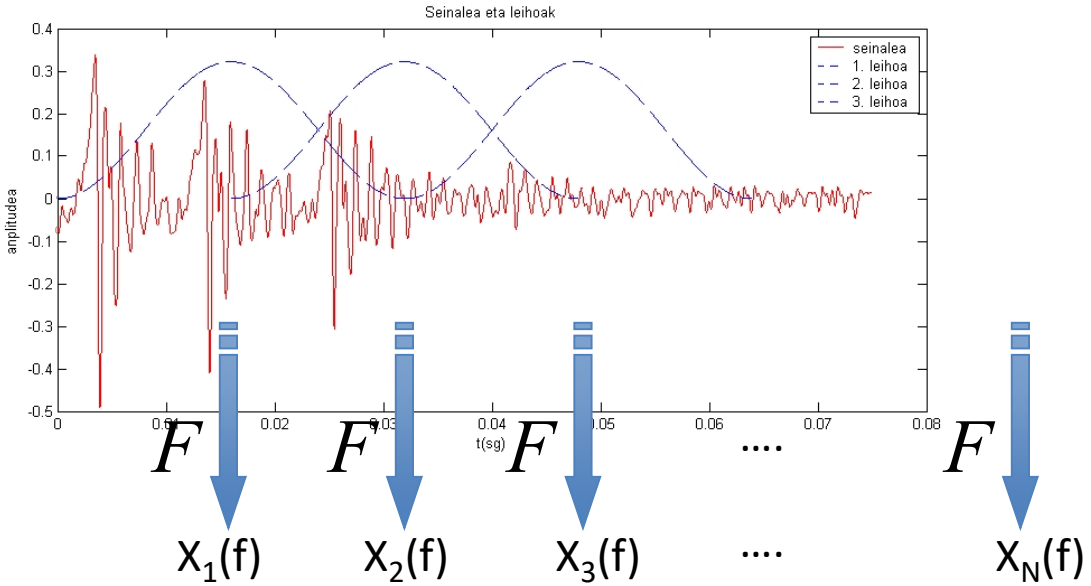
- The effective length of the window is reduced: overlapped segments
- Usually:
  - Window length=30ms, frame =10ms.
    - Frame Rate=1/10ms = 100 Hz
- To know more about windows: wintool in Matlab



Short-time Fourier Transform

- For non stationary signals windowing is applied to analyze them frame by frame. The FT of every frame is calculated: Short-time Fourier transform (STFT).

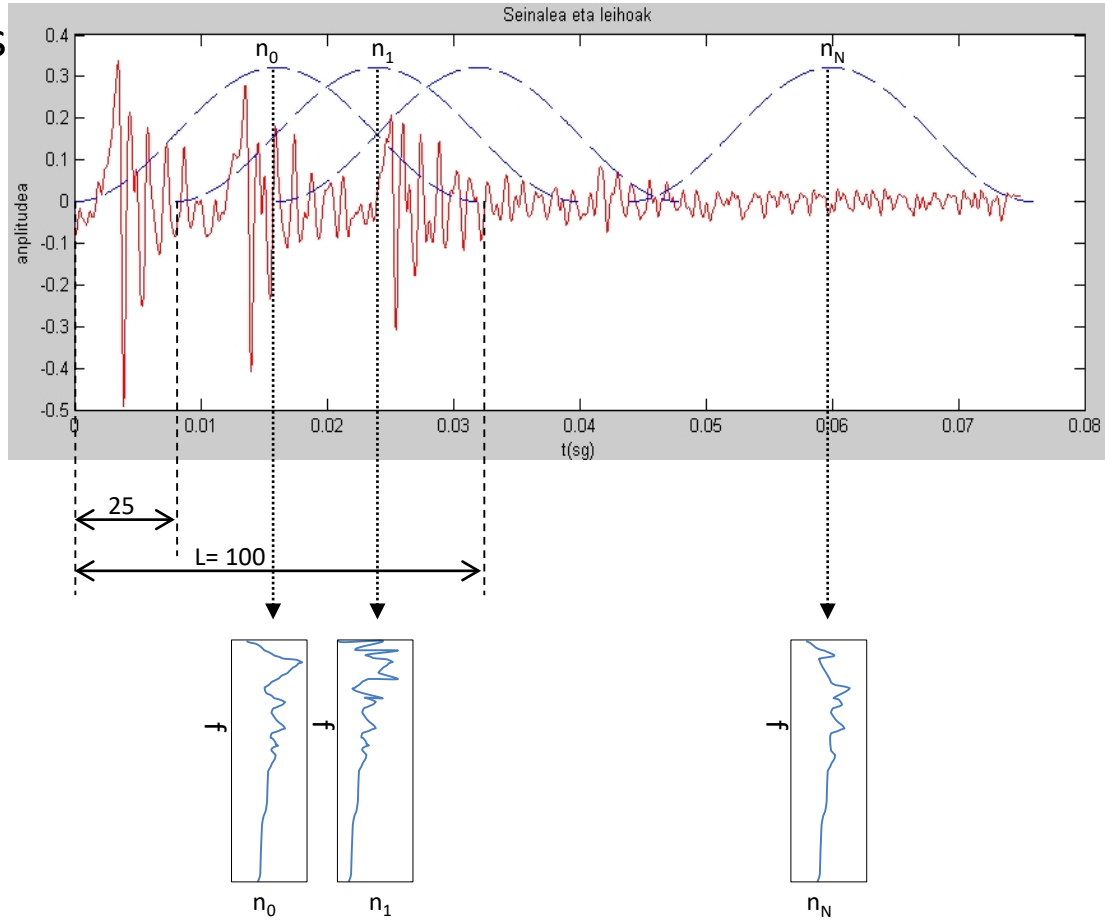
$$S_n(f) = \sum_{m=-\infty}^{\infty} s(m)w(m-n)e^{-j2\pi fm}$$



Short-time Fourier Transform

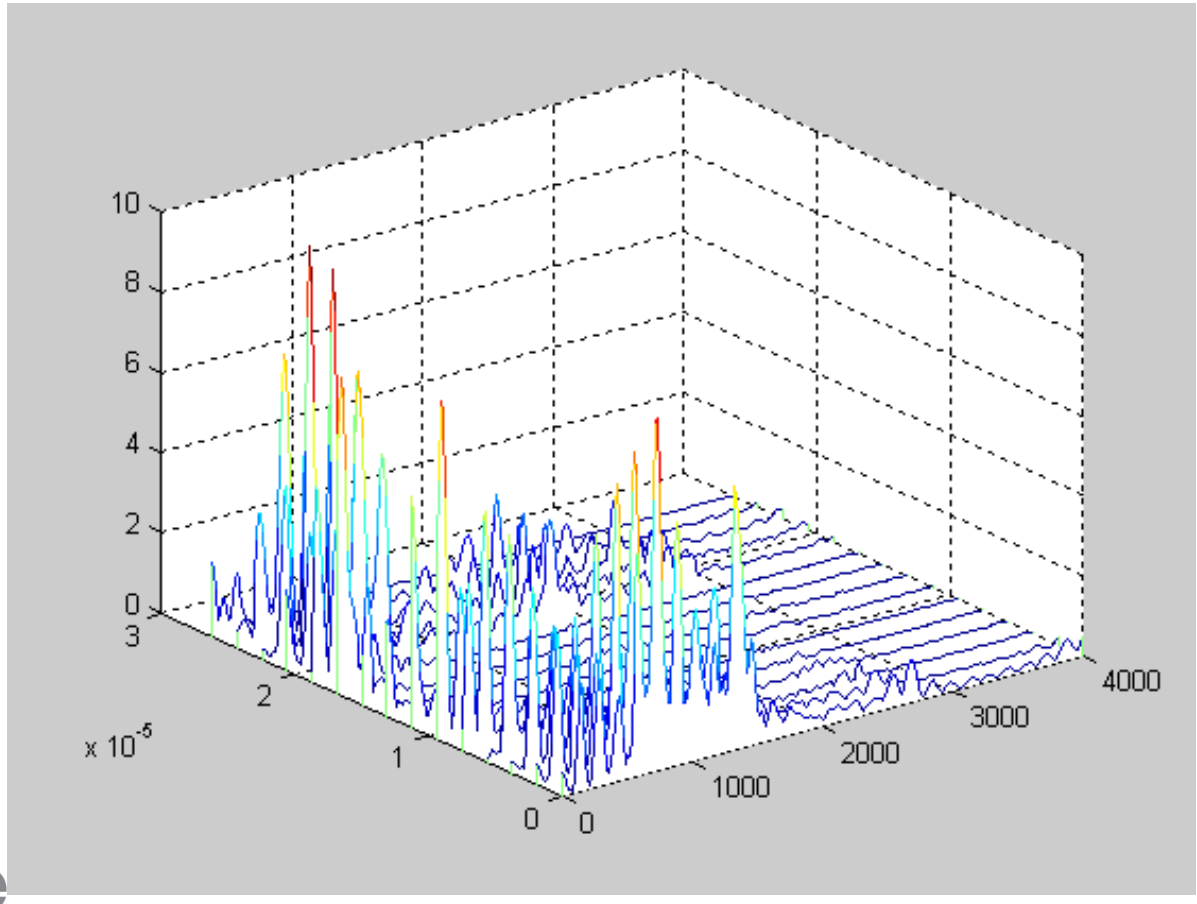
Example ⟹ 100 samples, *Hamming* window, f_s=10kHz

- The window length has influence in the frequency resolution.
- The window shape will influence the precision of the measurement.
- The amount of overlapping will determine the time resolution



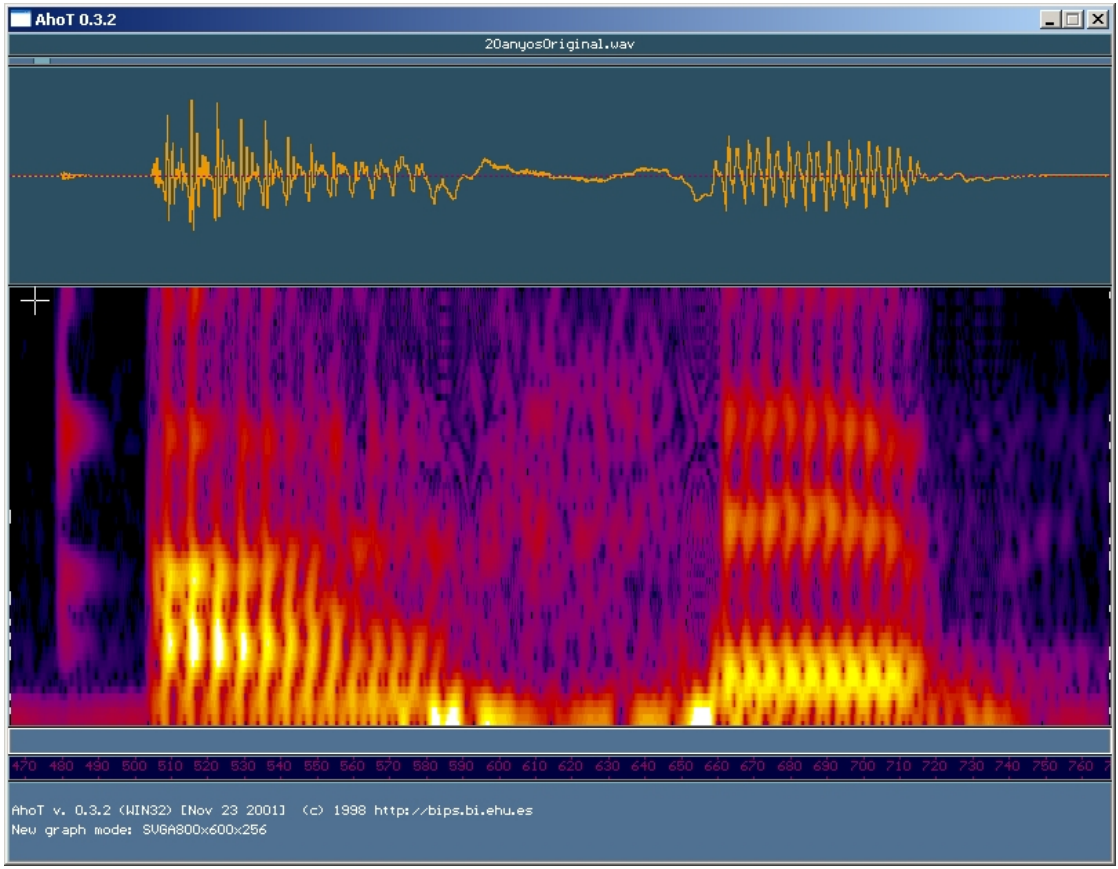
Spectrogram

- 3D Spectrogram



Spectrogram

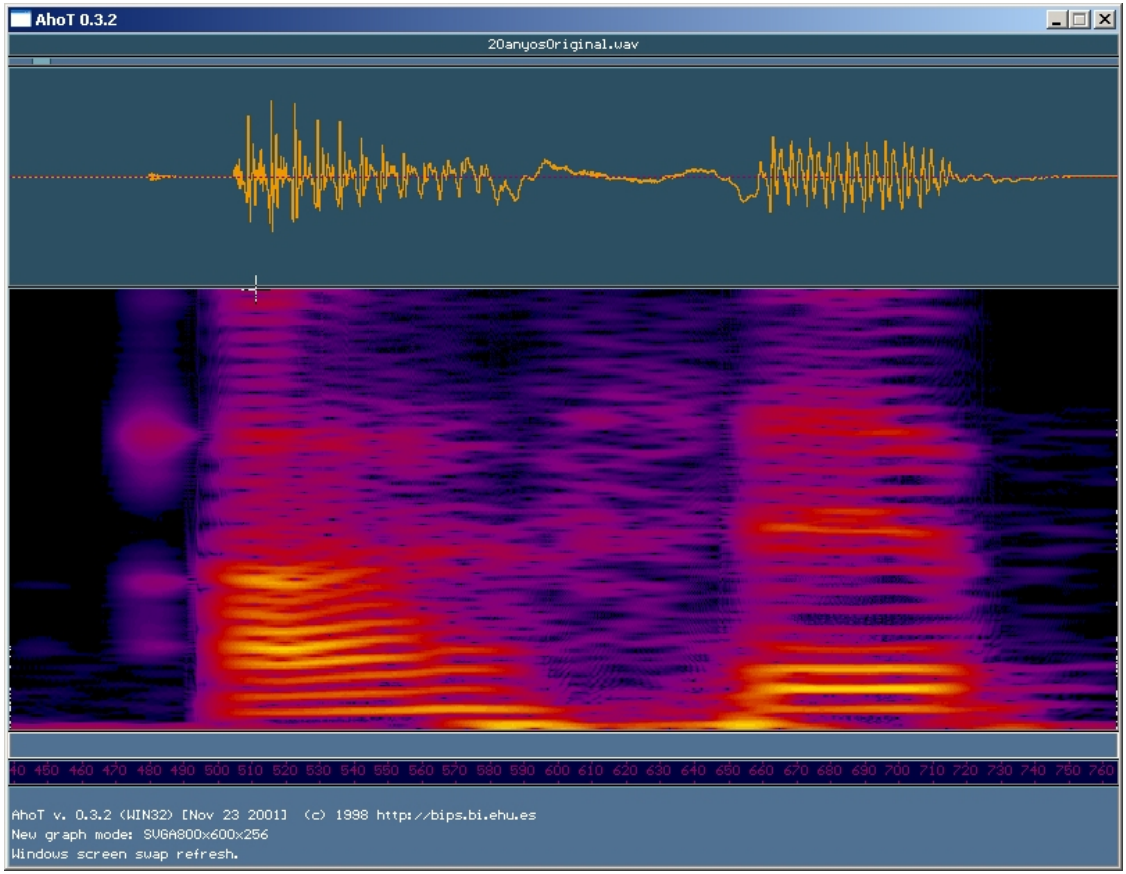
- If the window is short
  - Good time resolution
  - Worse frequency resolution
  - Wide band spectrogram



Wide band Spectrogram. 5 ms time window

Spectrogram

- If the window is long
  - Good frequency resolution
  - Worse time resolution
  - Narrow band spectrogram

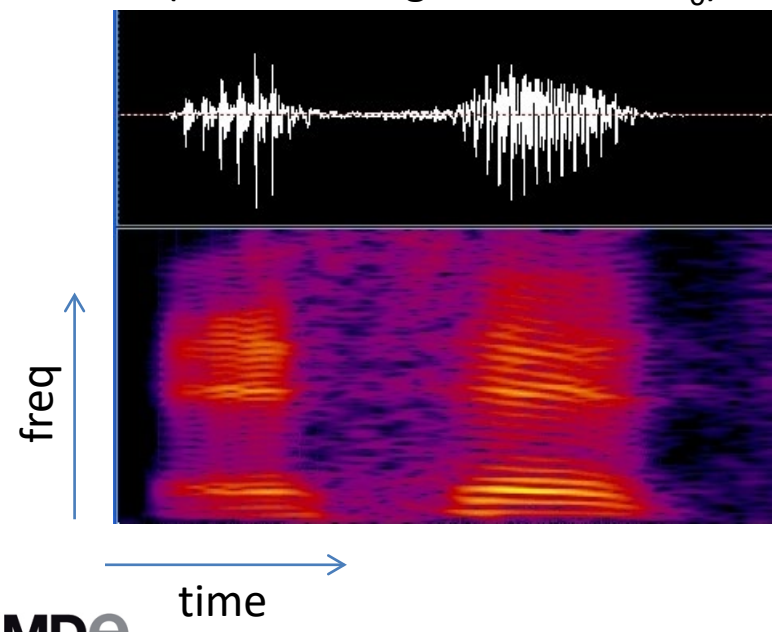


Narrow band spectrogram. 30ms time window

Segmental features

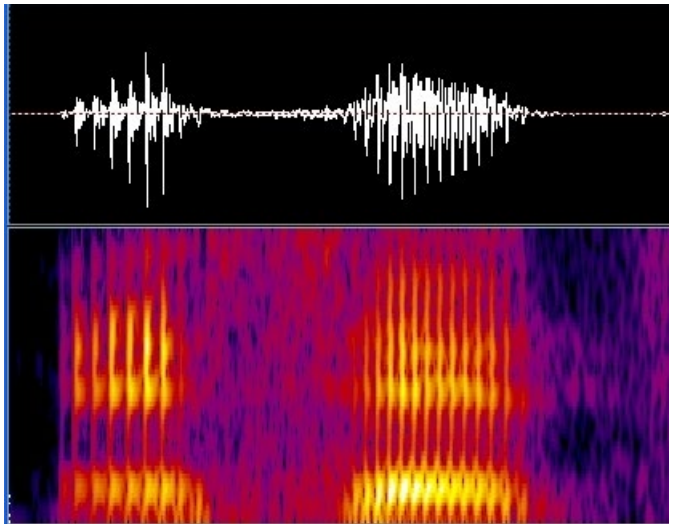
- Spectrogram
  - 2D representation of spectral (log-)amplitude as a function of time and frequency

(window length = 30ms > T_0)



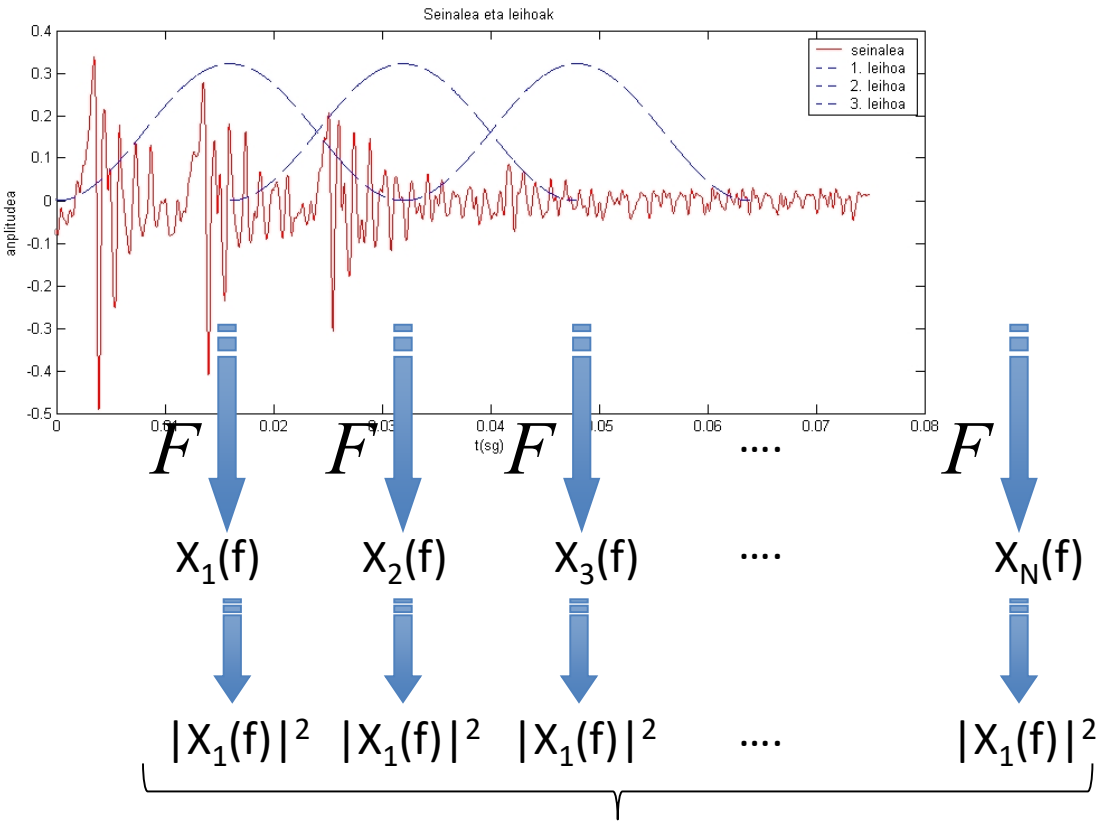
Narrow-band spectrogram

(window length = 5ms < T_0)



Wide-band spectrogram

Power spectral density

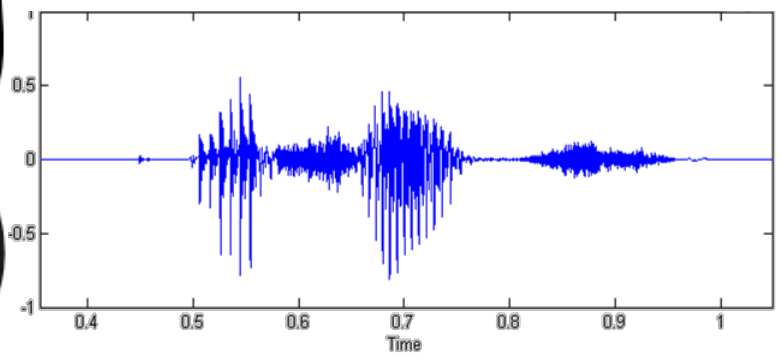


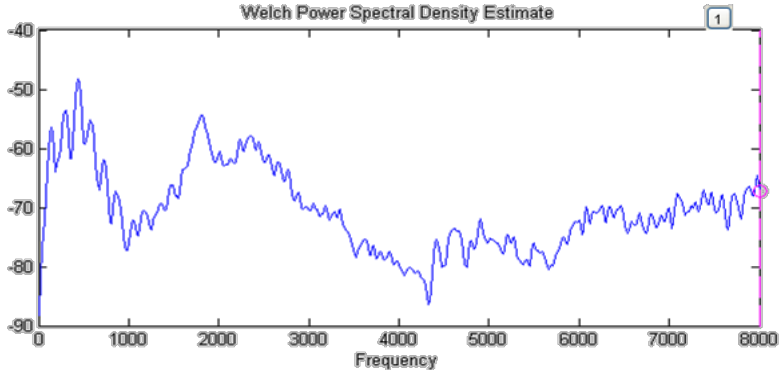
$$\frac{1}{N}\sum_{i=1}^{N}\left|X_i(f)\right|^2$$

Basic estimation of the signal power distribution in frequency

Power spectral density

$$G_x(f) \approx \frac{1}{N}\sum_{i=1}^{N}\left|X_i(f)\right|^2$$





Segmental features

- Spectrogram

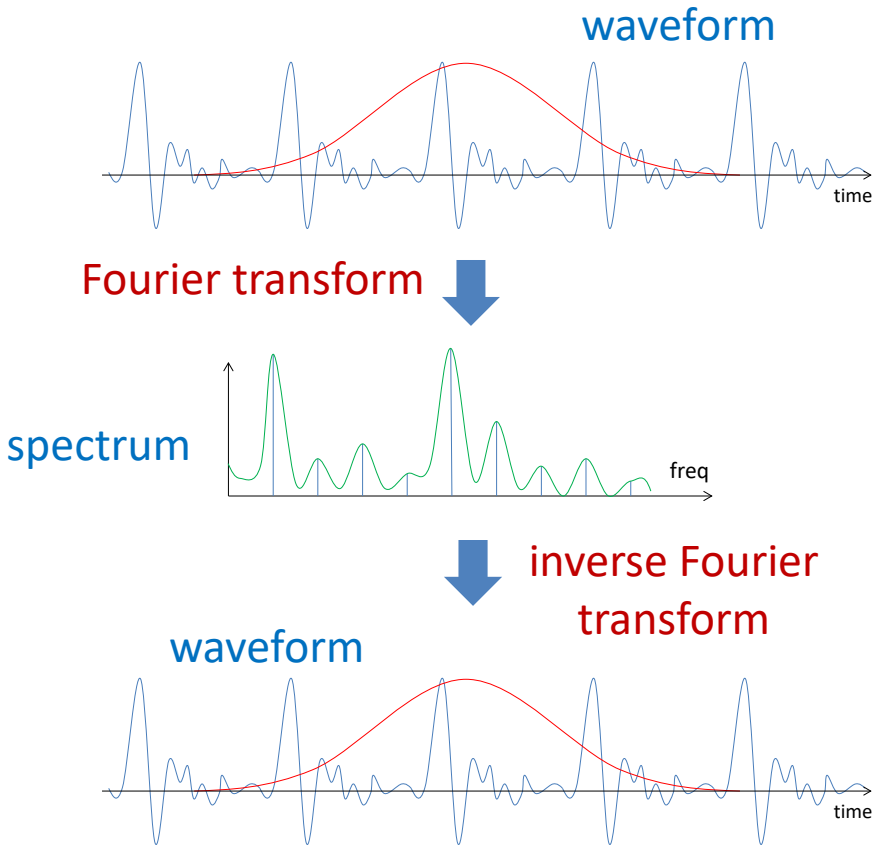
Guided exercise: plot the spectrogram of a speech signal for different window lengths in Matlab® and check the effect of the length in frequency

Outline

- Introduction
- Supra-segmental acoustic features
- Segmental acoustic features
  - Short term analysis and spectrogram
  - Cepstrum, MFCCs, mel filterbanks

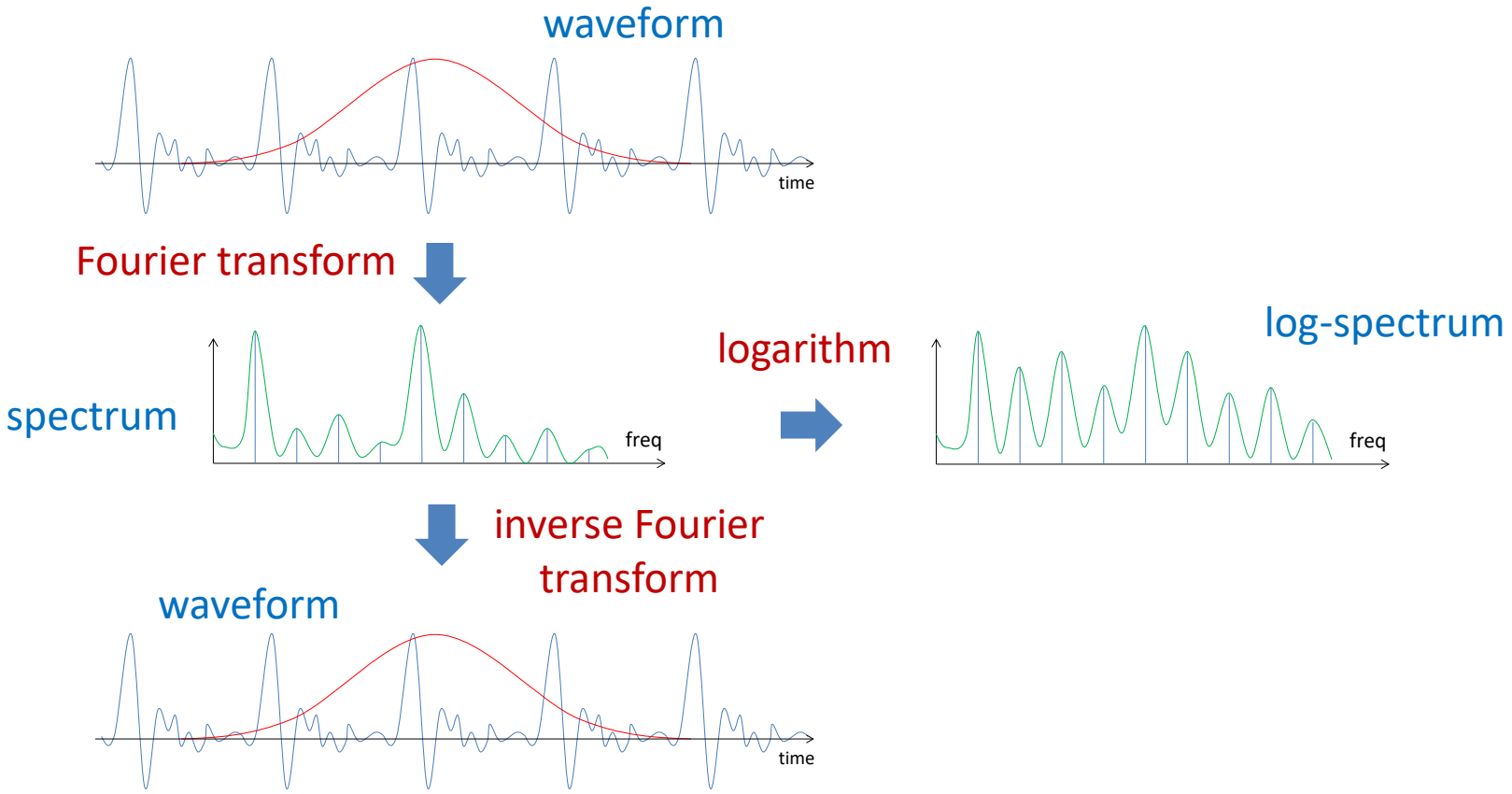
Cepstrum coefficients

- Cepstrum



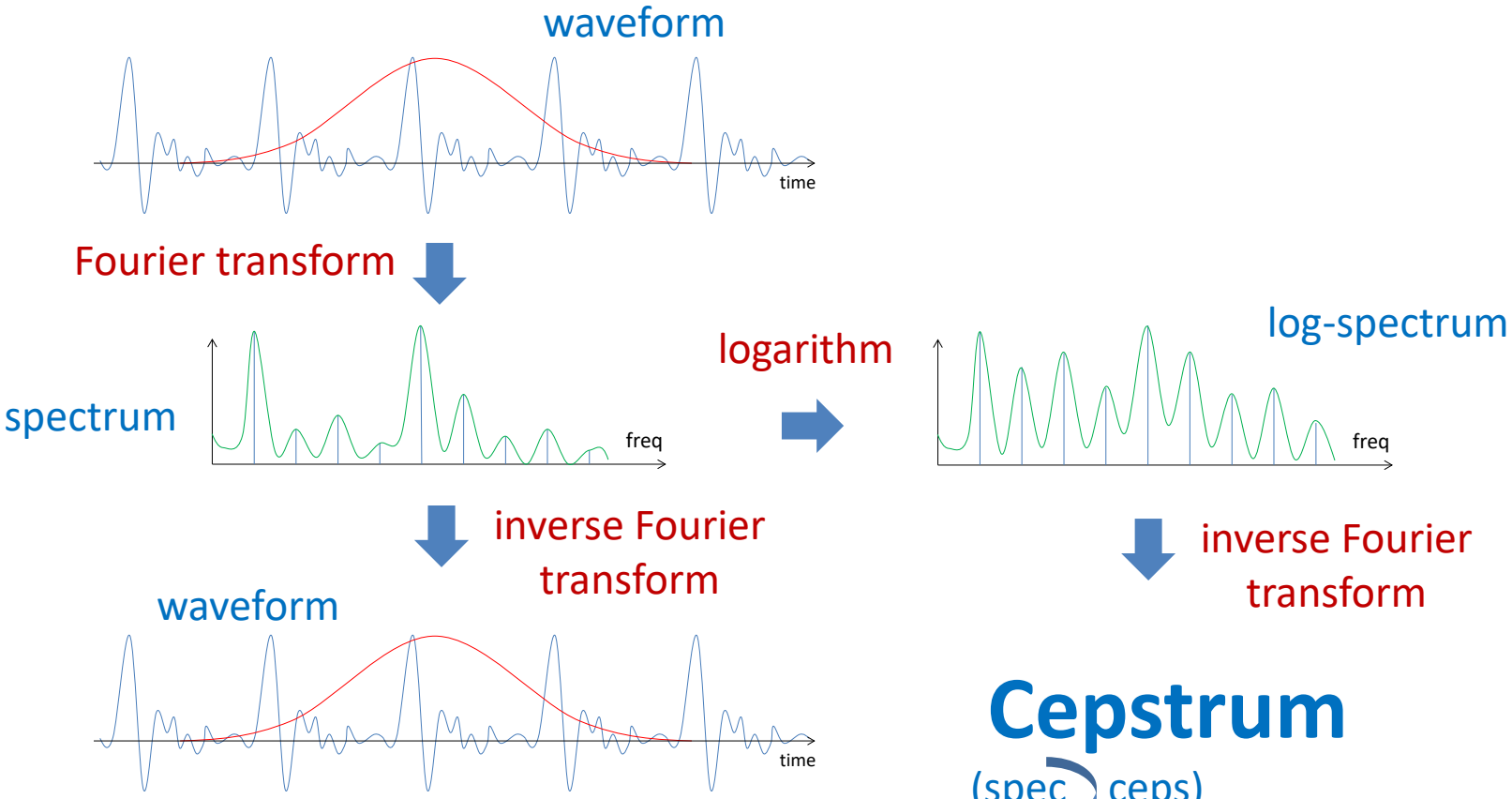
Cepstrum coefficients

- Cepstrum



Cepstrum coefficients

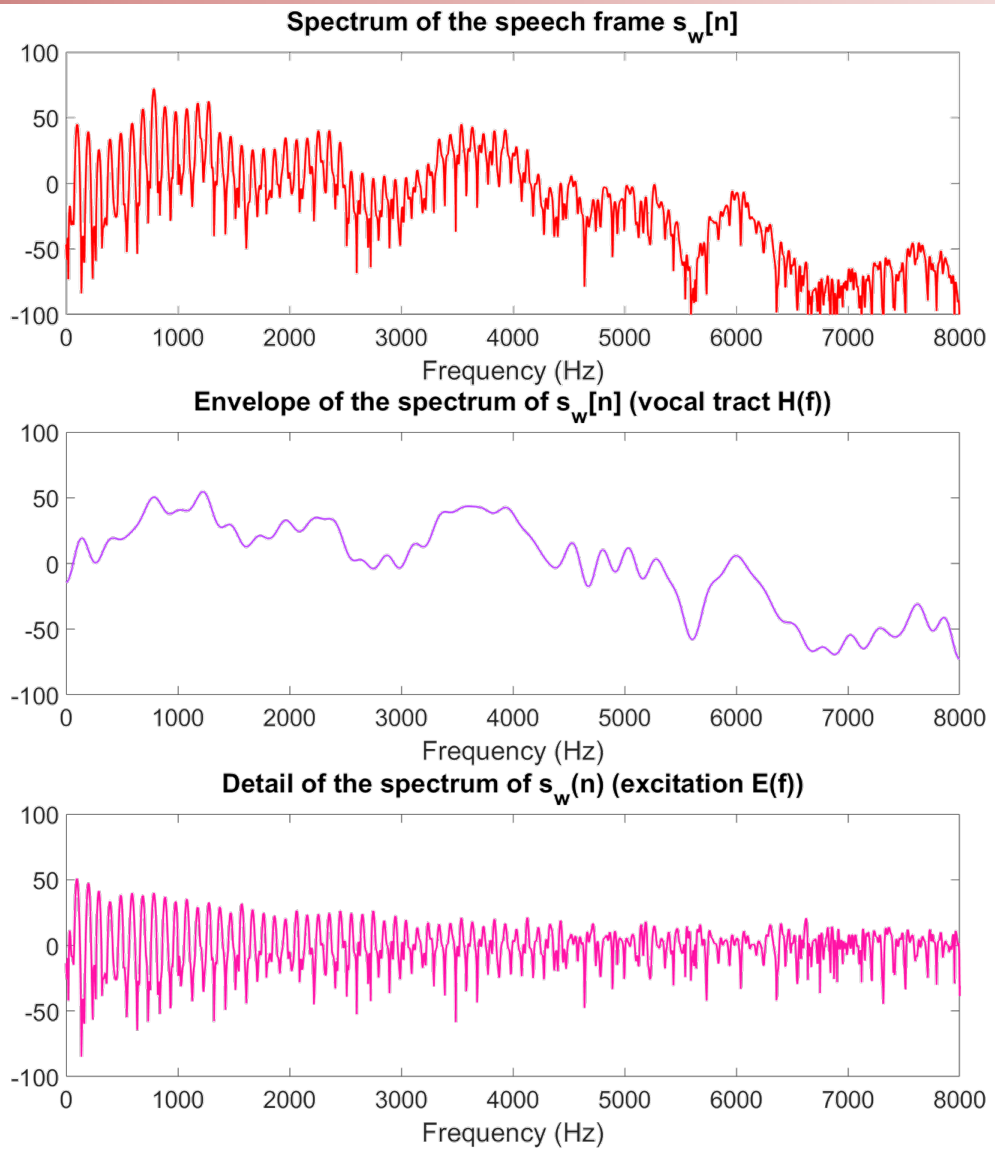
- Cepstrum



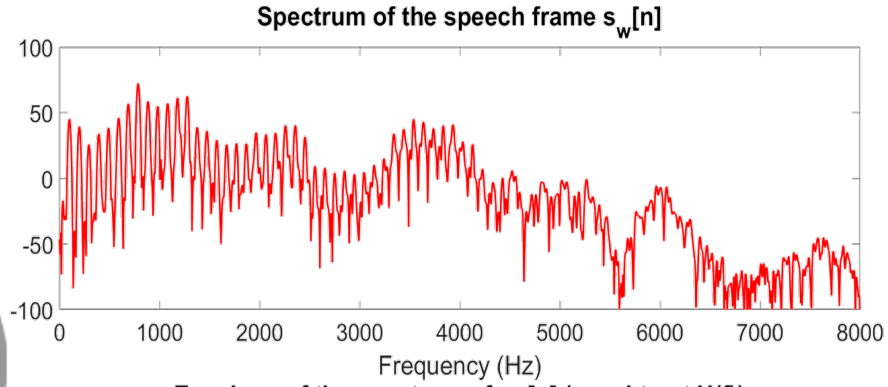
Cepstrum coefficients

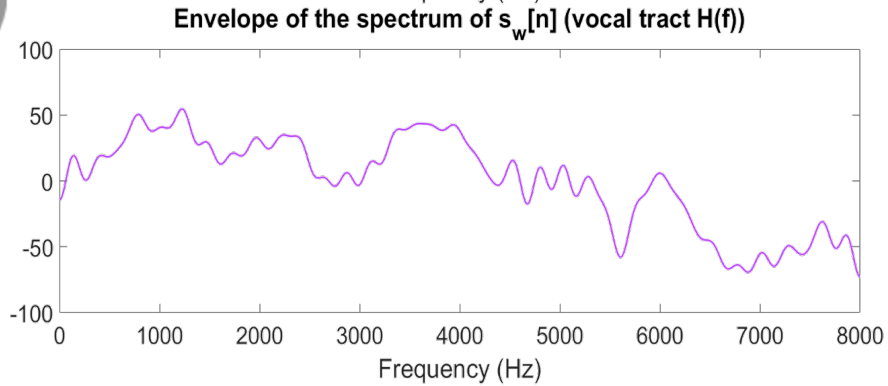
Spectrum envelope varies slowly while excitation varies fast

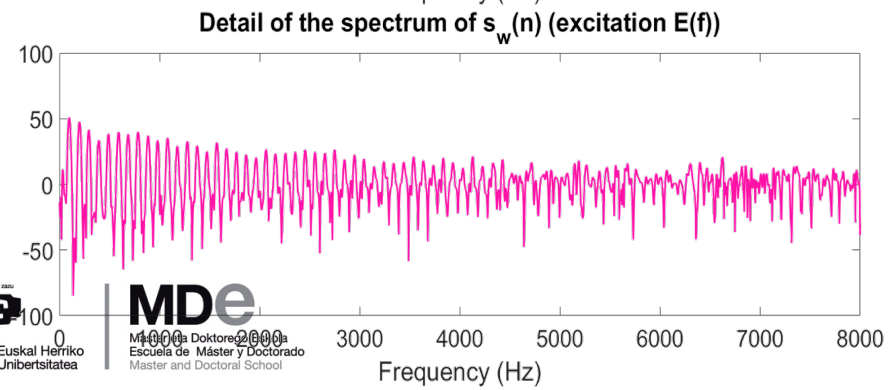
They could be separated by frequency analysis

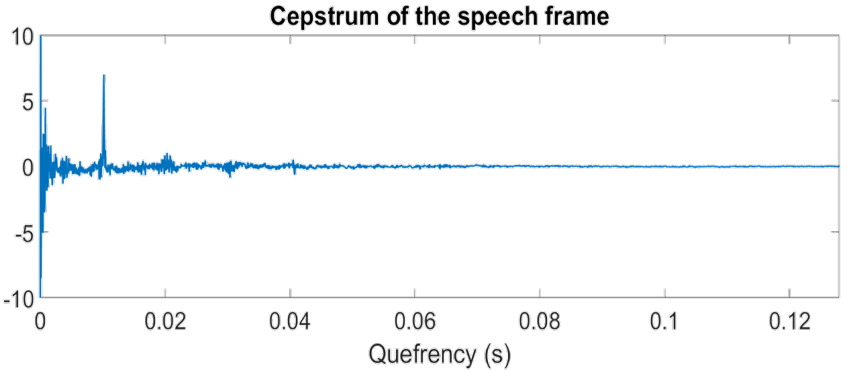


Cepstrum coefficients

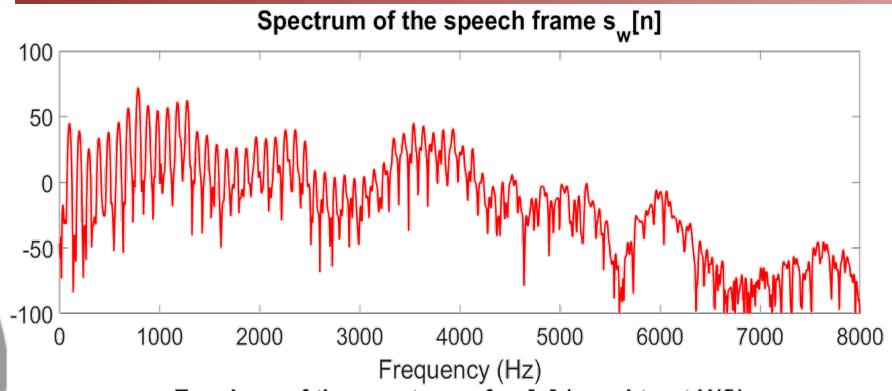


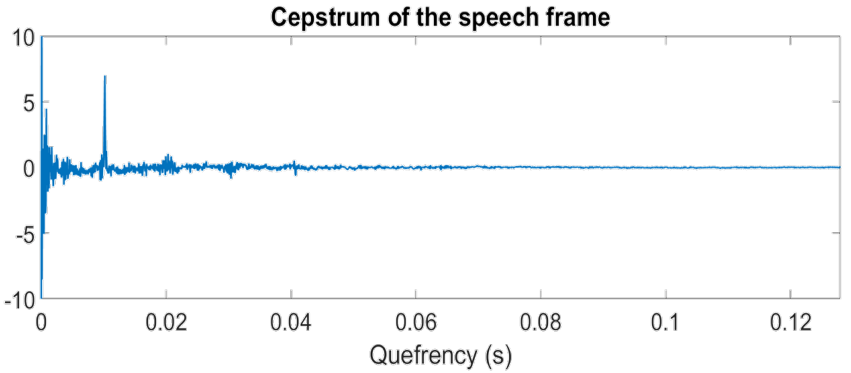


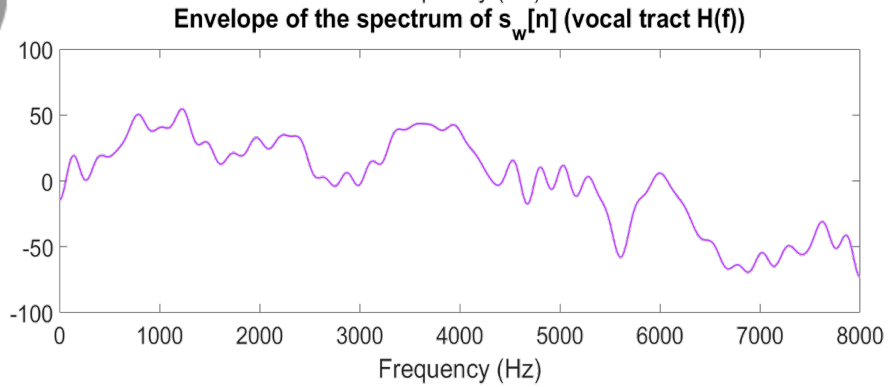


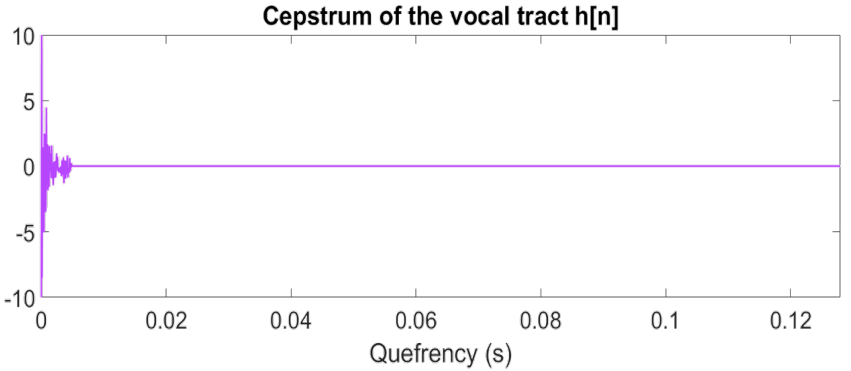


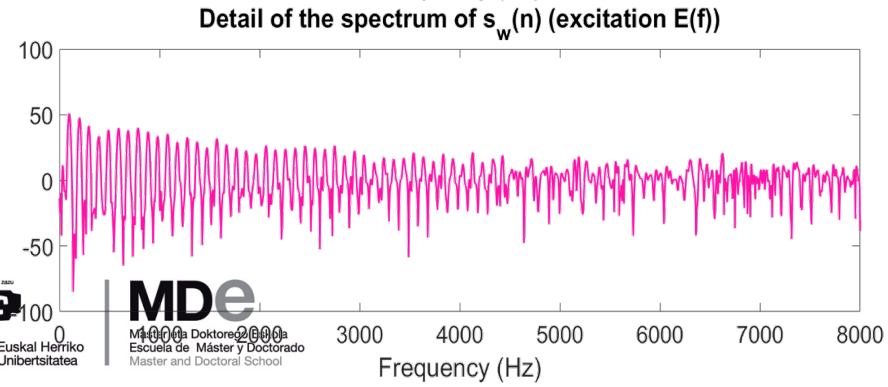
Cepstrum coefficients

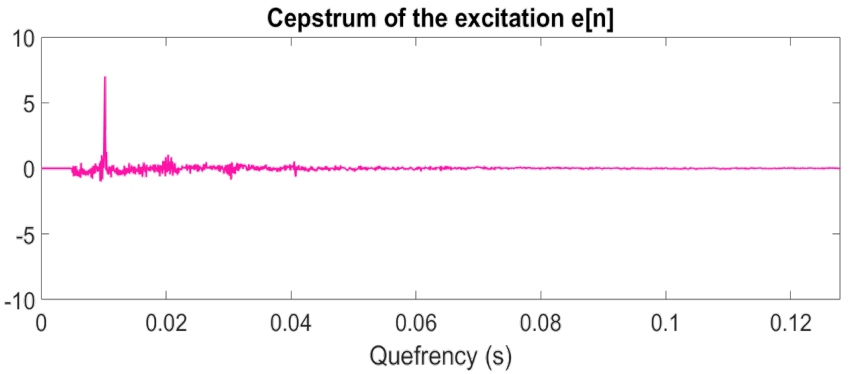




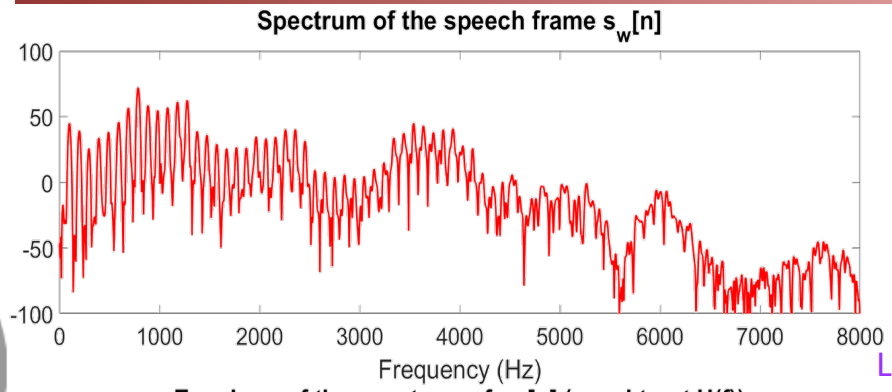


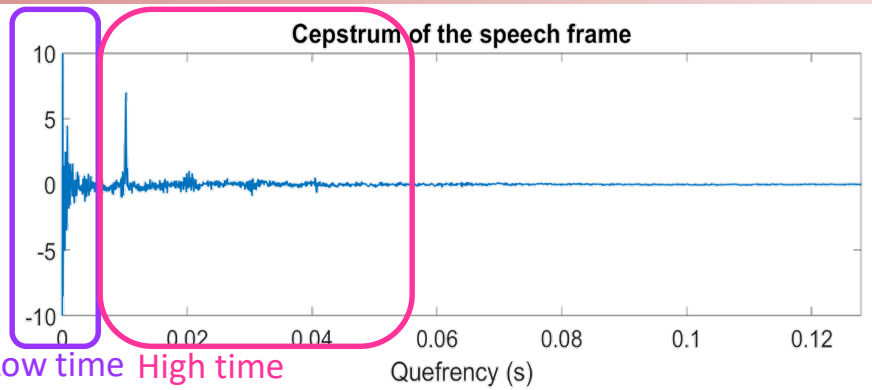






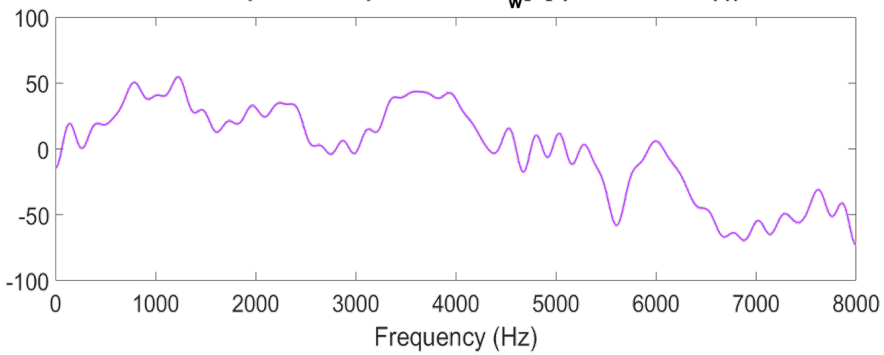
Cepstrum coefficients

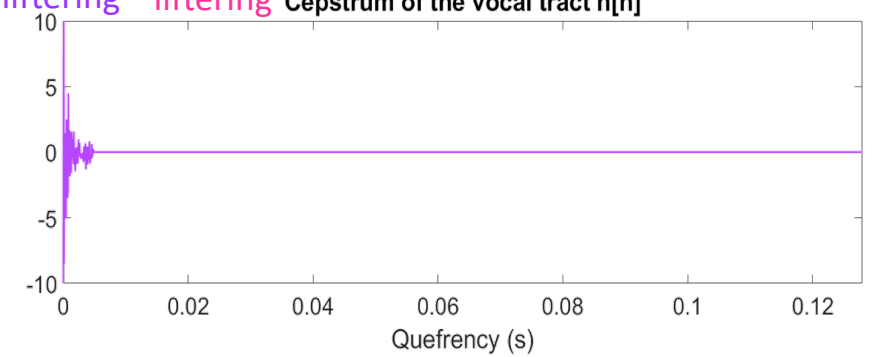


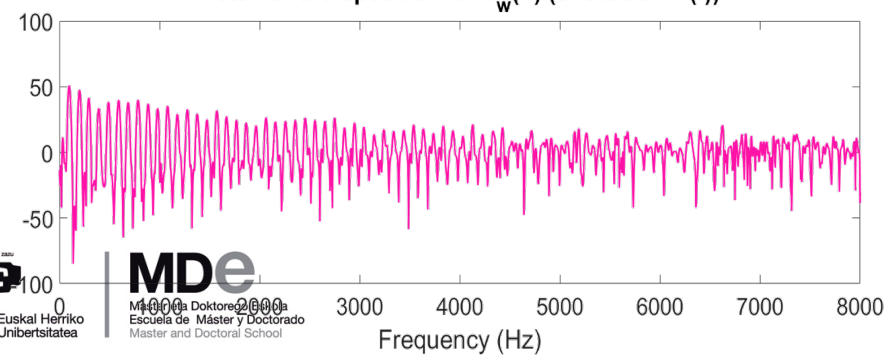


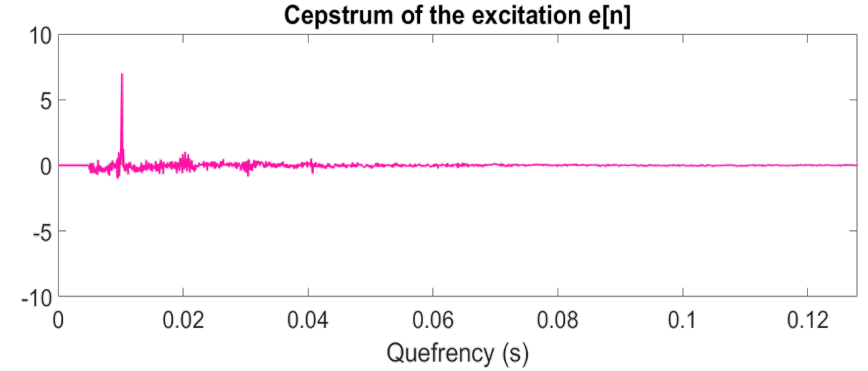
Low time "liftering"

High time "liftering"



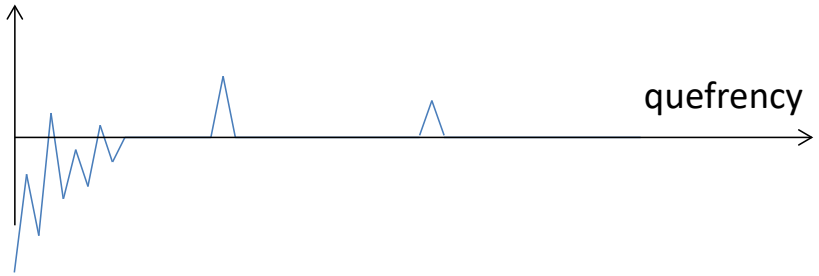






Cepstrum coefficients

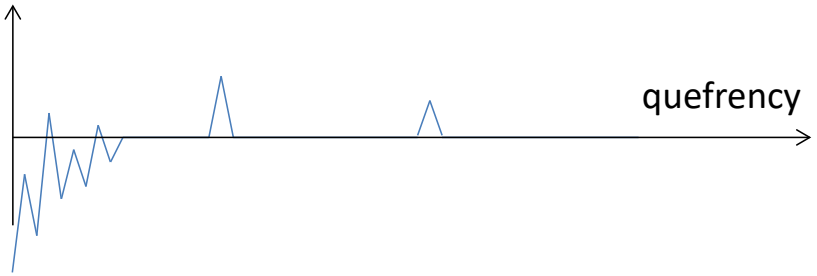
- Cepstrum
  - How does it look like?



(quefrency ≅ time)

Cepstrum coefficients

- Cepstrum
  - How does it look like?



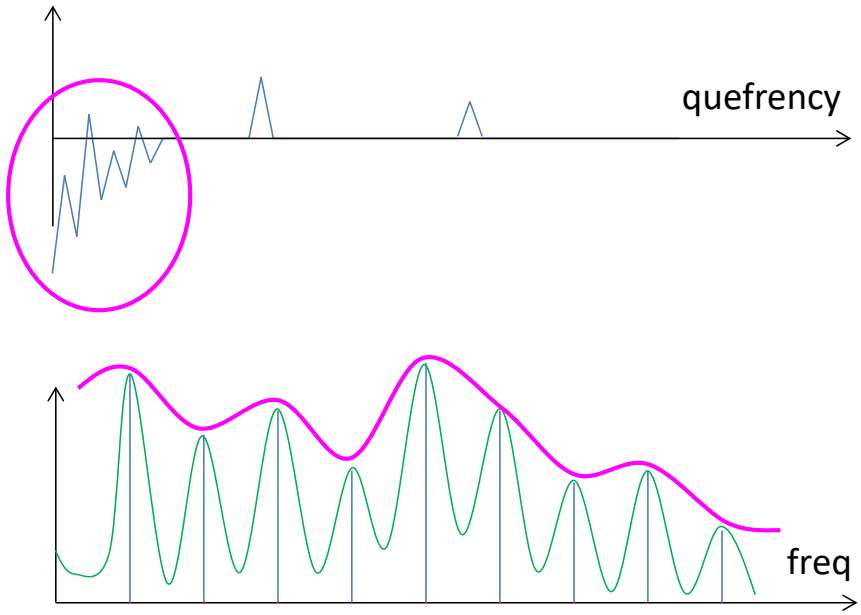
Guided exercise: plot the cepstrum of a speech signal in Matlab® and check that it looks like this

(quefrency ≅ time)

Cepstrum coefficients

- Cepstrum
  - How does it look like?

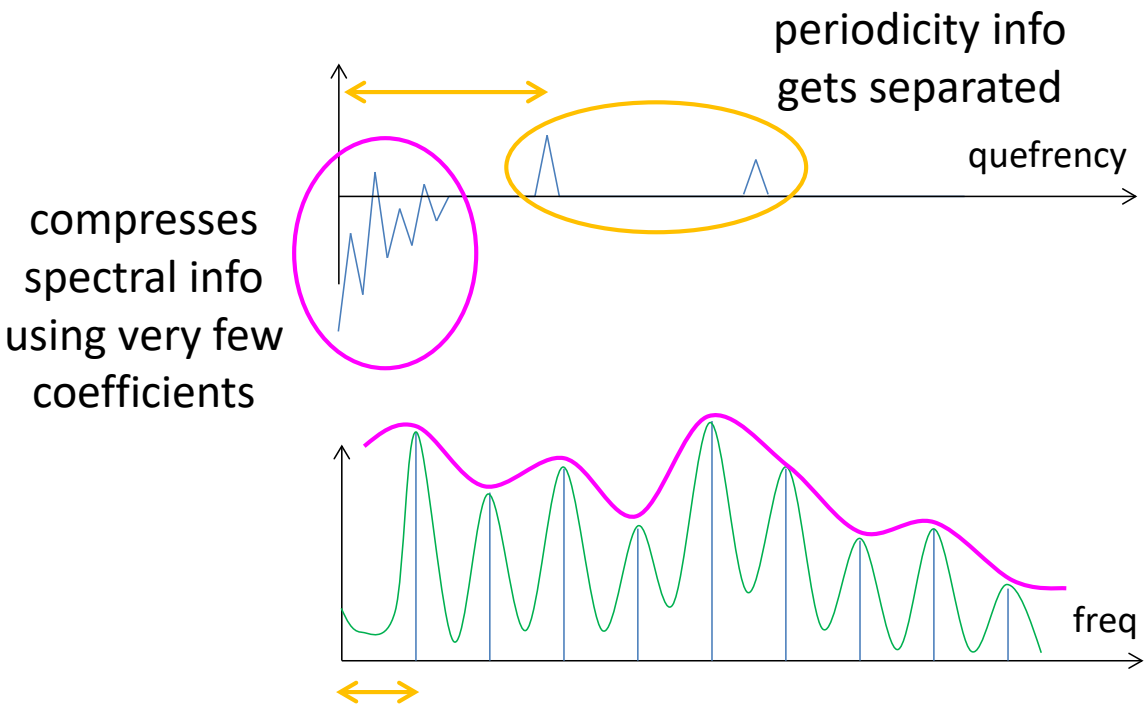
compresses spectral info using very few coefficients



(quefrency ≅ time)

Cepstrum coefficients

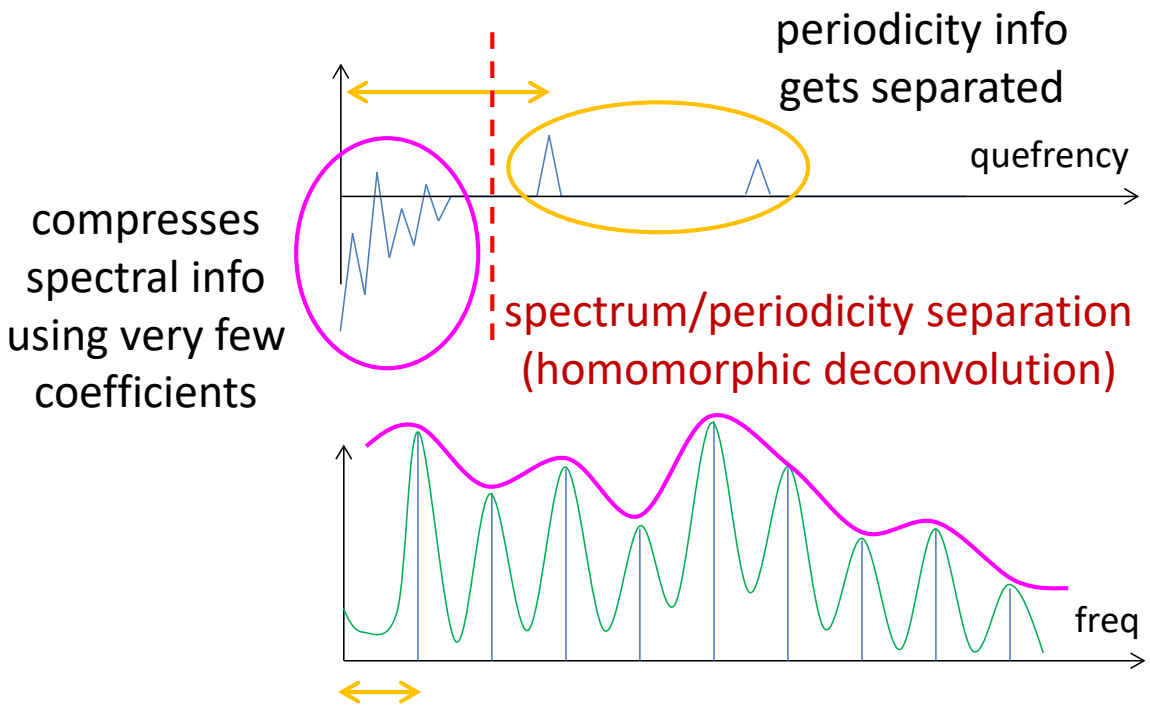
- Cepstrum
  - How does it look like?



(quefrency ≅ time)

Cepstrum coefficients

- Cepstrum
  - How does it look like?



(quefrency ≅ time)

Cepstrum coefficients

- Cepstrum

Guided exercise: create a basic pitch detection algorithm in Matlab®

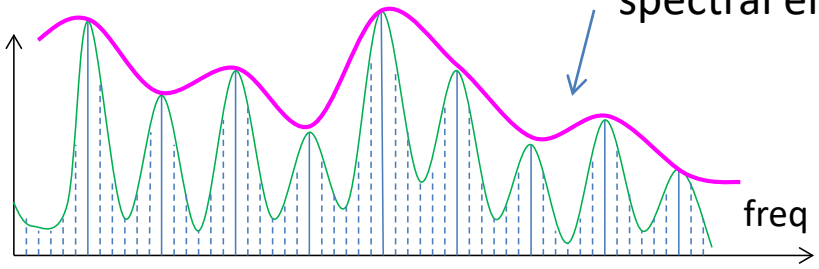
- Periodicity of voiced frames → max{cepstrum}
- Voice activity detection → signal power > P_0
- Discriminate voiced/unvoiced → ZCR < Z_0

MFCCs

- Mel-frequency cepstral coefficients (MFCCs)

we have uniformly-spaced samples of the spectrum

we want a compact representation of the spectral envelope



According to our ear's response, the resolution at low frequencies is perceptually more important

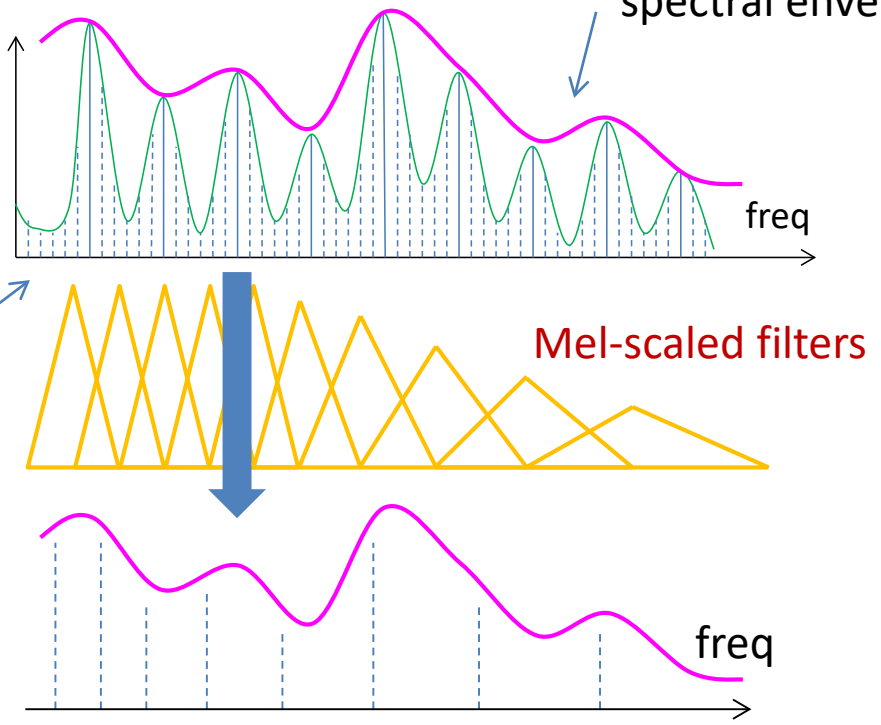
MFCCs

- Mel-frequency cepstral coefficients (MFCCs)

we have uniformly-spaced samples of the spectrum

we want a compact representation of the spectral envelope

According to our ear's response, the resolution at low frequencies is perceptually more important

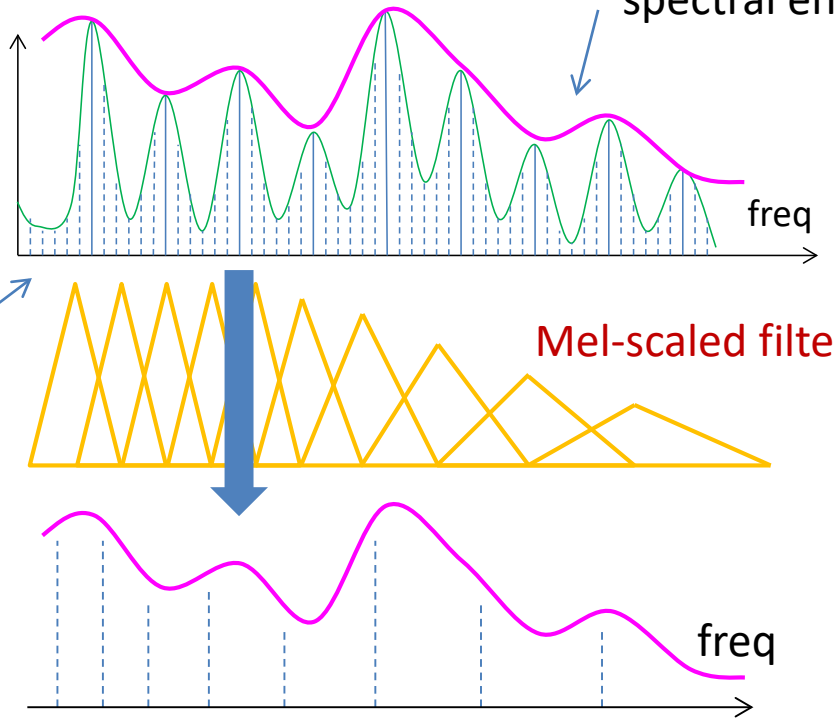


MFCCs

- Mel-frequency cepstral coefficients (MFCCs)

we have uniformly-spaced samples of the spectrum

we want a compact representation of the spectral envelope



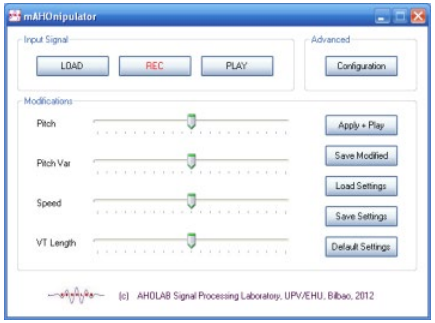
According to our ear's response, the resolution at low frequencies is perceptually more important

discrete cosine transform → **MFCCs**

(retain first coeffs usually 13)

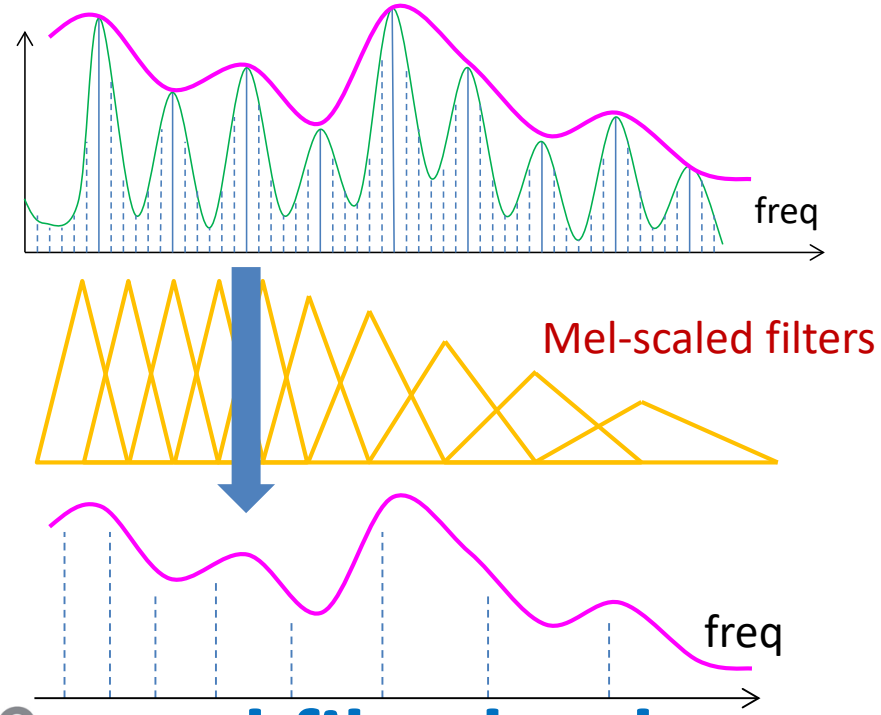
MFCCs

- Mel-frequency cepstral coefficients (MFCCs)
  - Rapid analysis → real-time applications
  - Very popular wherever speech analysis is involved: speech, speaker or [whatever] recognition
  - No way back → not adequate for synthesis (there are alternative ways of calculating similar coefficients: http://aholab.ehu.eus/ahocoder)



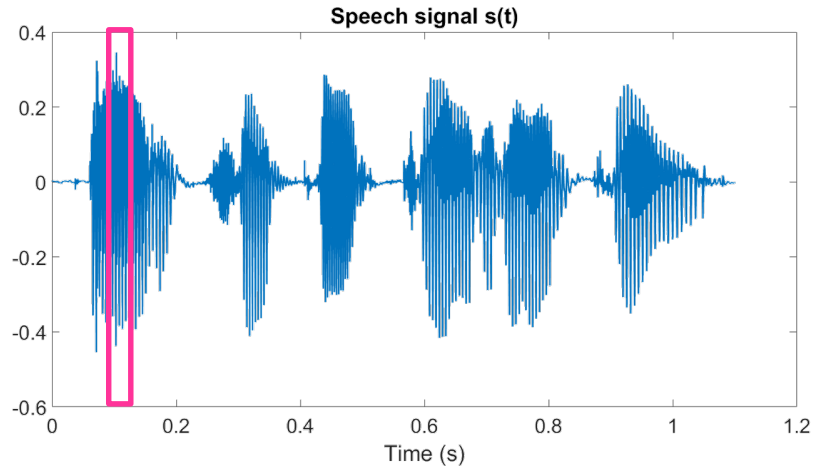
Mel filterbanks

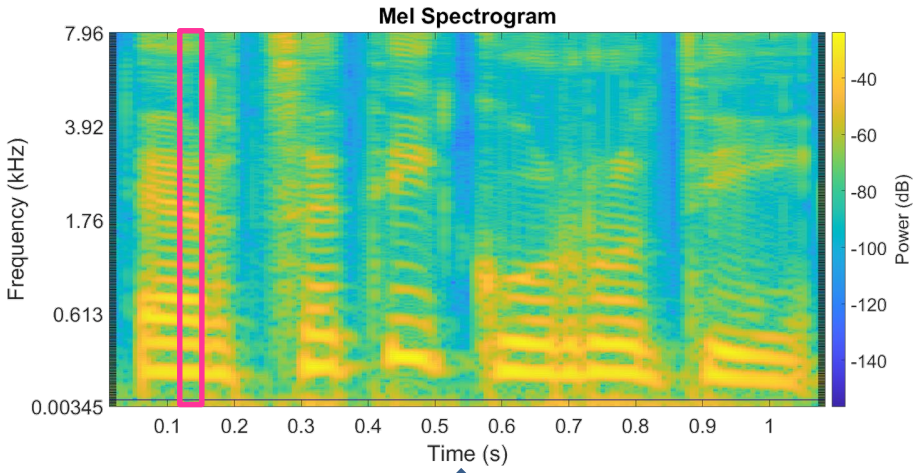
- Filter banks
  - Neural networks do not need coefficient decorrelation



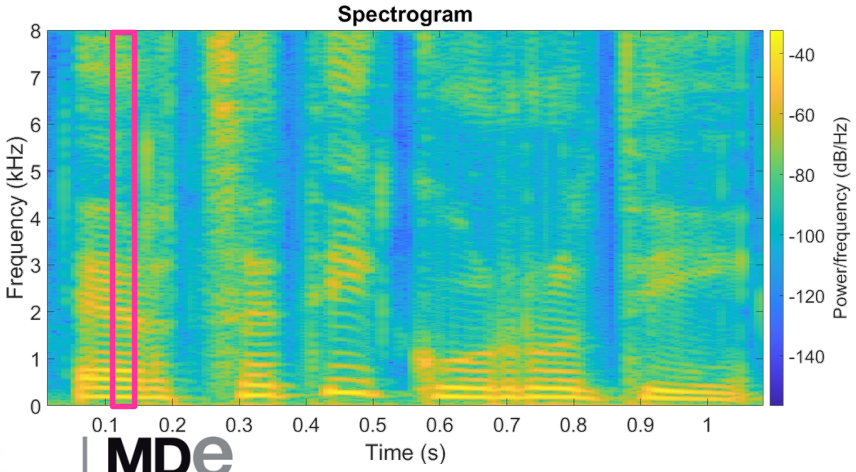
mel filterbanks

Mel filterbanks



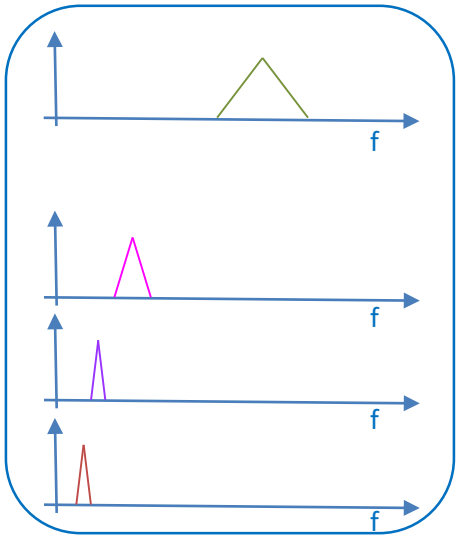


Fourier transform



Logarithm

MEL filterbank



Recommended lectures