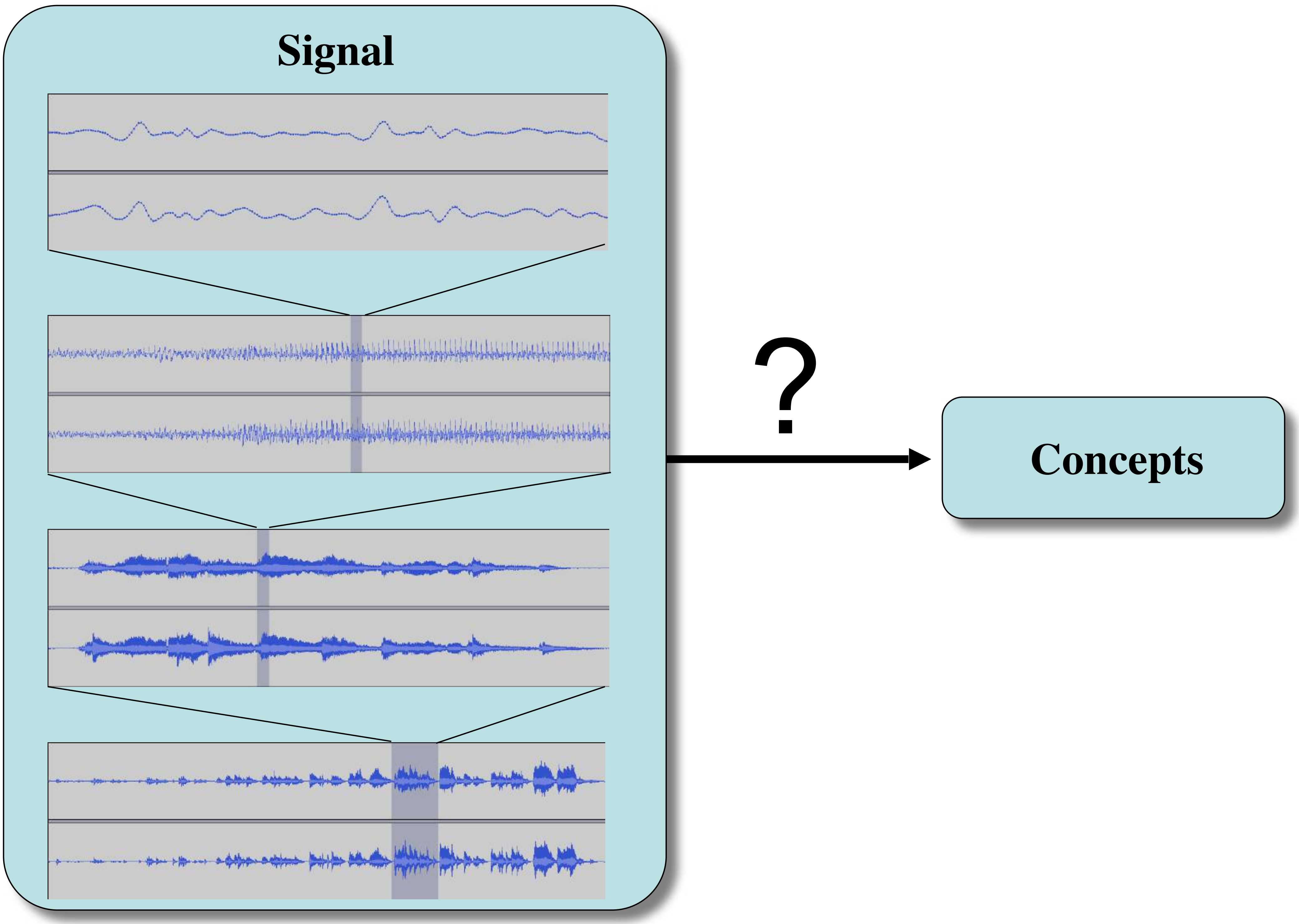
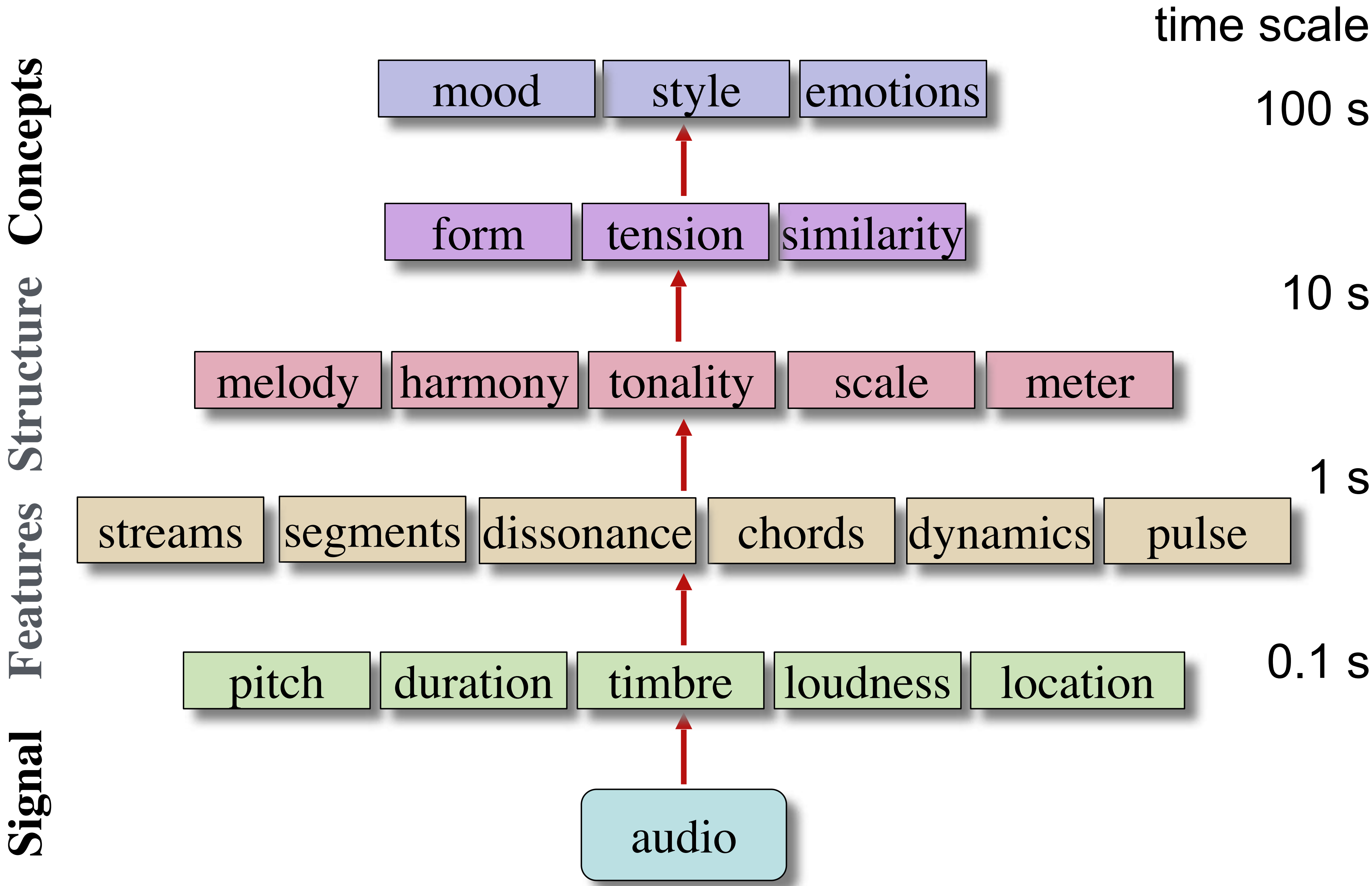
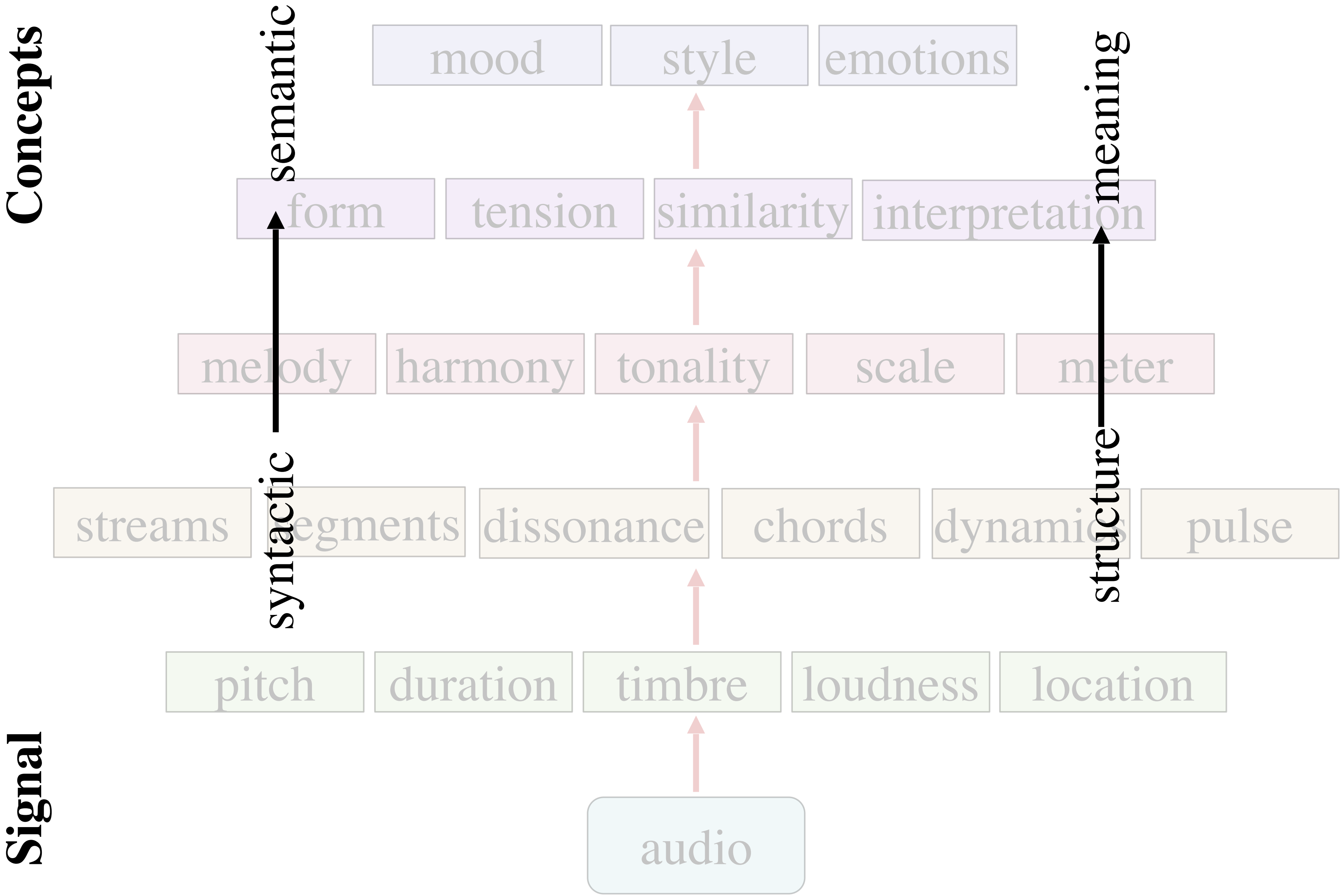

Signal
?
Concepts

Levels of Music Processing



Levels of Music Processing



Some Questions

- Can the syntactic and semantic levels of music be connected?
- Can the meaning of music be predicted from musical structure using computational models?
- Can music similarity be modelled?

Music Information Retrieval (MIR)

- emerged in late 1990s
- multidisciplinary
  - computer science, psychology, cognitive science, musicology, digital library, information science, law, commerce, sociology,…

Why MIR? Practical motivation

- Increasing amount of music being produced
- Need for systems for organization, search, retrieval, classification, recommendation, browsing, dealing with copyright issues, ...

Why MIR? Musicological motivation

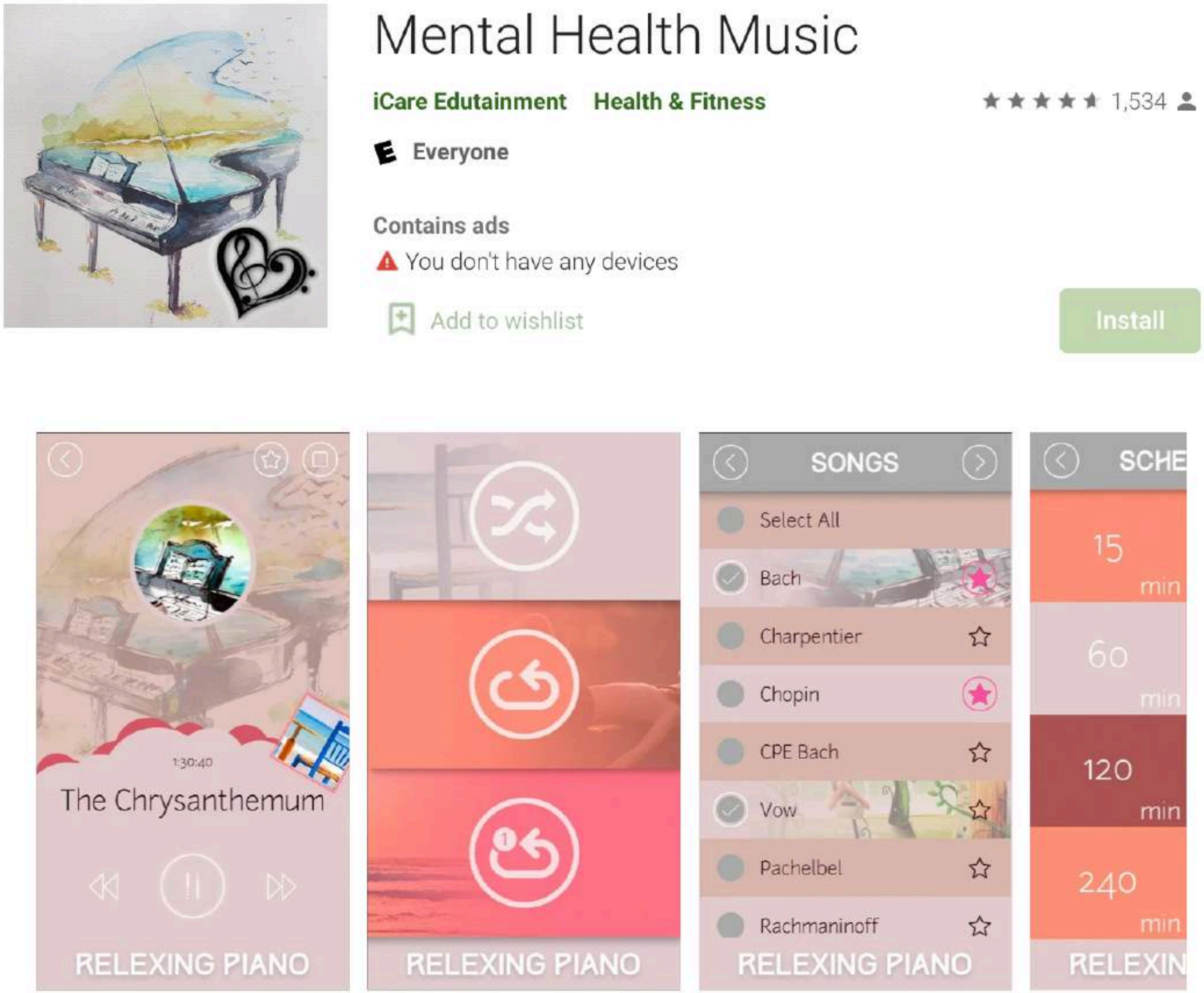
- Traditional music analysis
  - based on personal observation/intuition/speculation
  - theories non-formalized
  - no testing, non-falsifiable theories
  - small corpora

- Computational music analysis
  - empirically-based
  - formalized theories
  - testable, falsifiable
  - large corpora

Applications of MIR

- Content-based music search (query by example, similarity, mood)
- Music recommendation
- Music browsing
- Automatic playlist generation
- Automatic music transcription
- Automatic accompaniment
- Music summarization (watermarking, fingerprinting)
- Music generation
- **what else????**

Theory/Function-driven Organisation and Retrieval for Well-Being



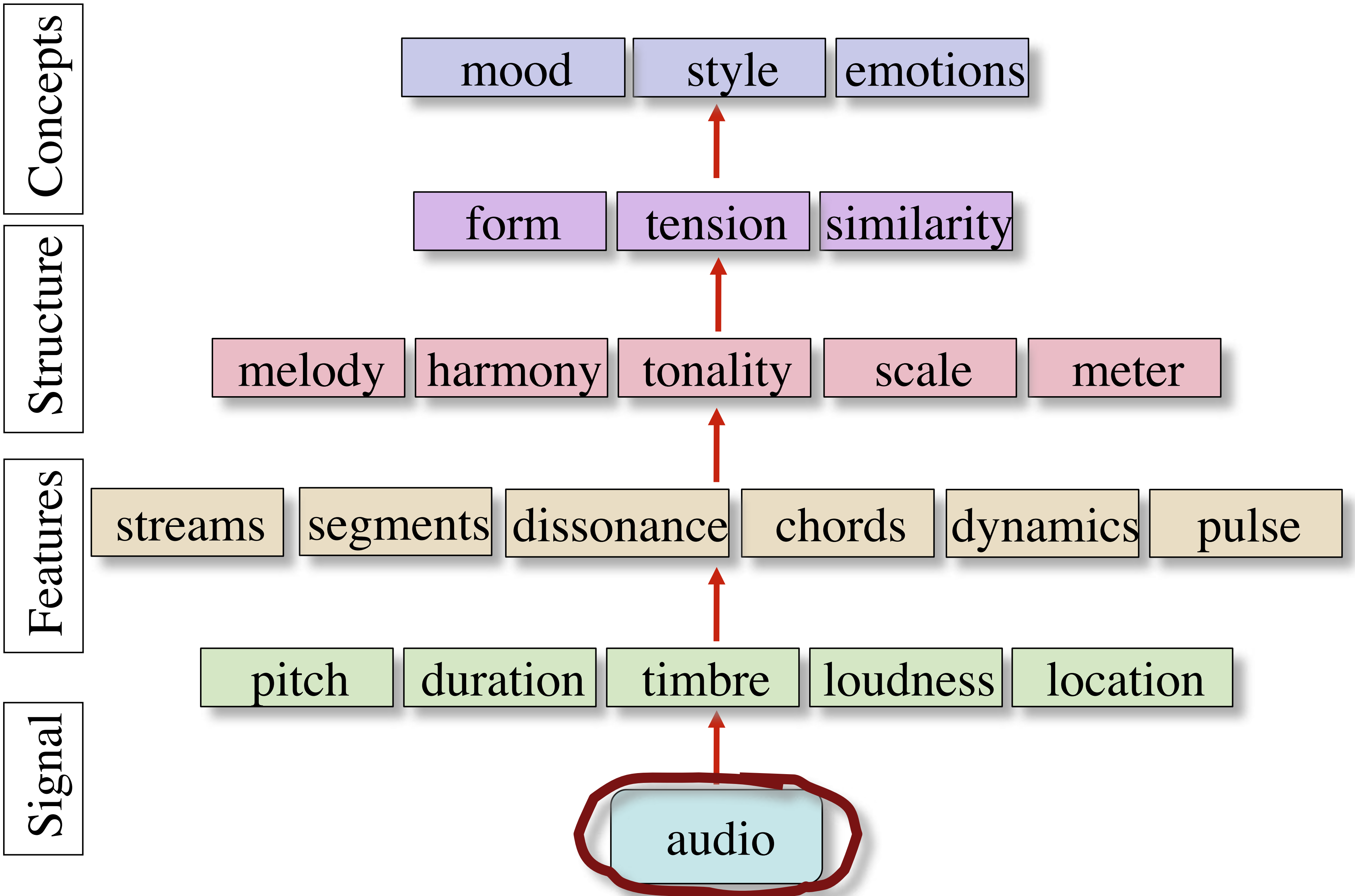
Theory/Function-driven Organisation and Retrieval for Well-Being

can you organise music into these categories?

MIR Community

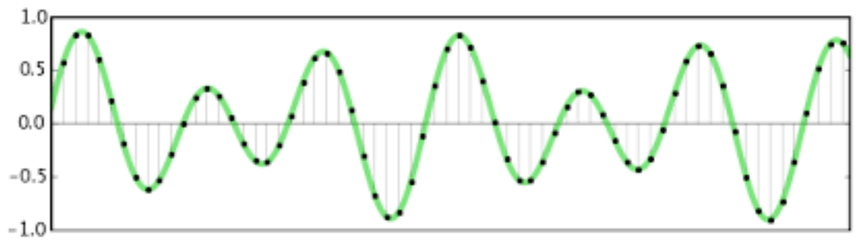
- Conferences
  - International Conference on Music Information Retrieval (ISMIR): 1999- (annual)
    - all papers online: http://www.ismir.net/proceedings/
  - CogMIR (2011-)
  - International Computer Music Conference
  - CMMR
- Journals
  - Computer Music Journal
  - Journal of New Music Research
  - Transactions of ISMIR
  - IEEE Transactions on Audio, Speech, and Language Processing
  - IEEE Transactions on Knowledge and Data Engineering
- Contests
  - MIREX (Music Information Retrieval Evaluation eXchange)

Levels of Music Processing



Audio Signal?

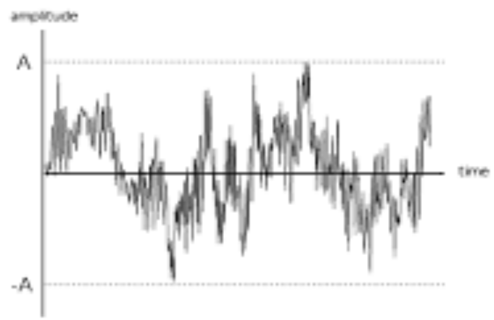
- Audio signal represents the air pressure variations that constitute sound
  - e.g. microphone output
- Digital audio signal is a sampled representation of analog audio signal

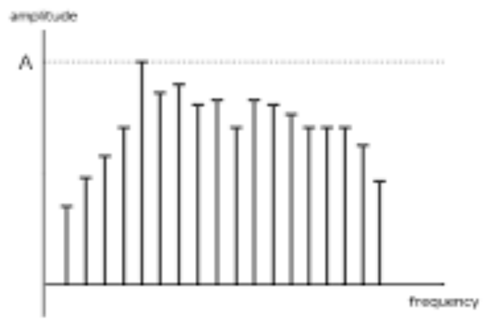


Representations of sound signals

- Two ways to represent a sound signal
  - **time domain**: signal waveform
  - **frequency domain**: spectrum

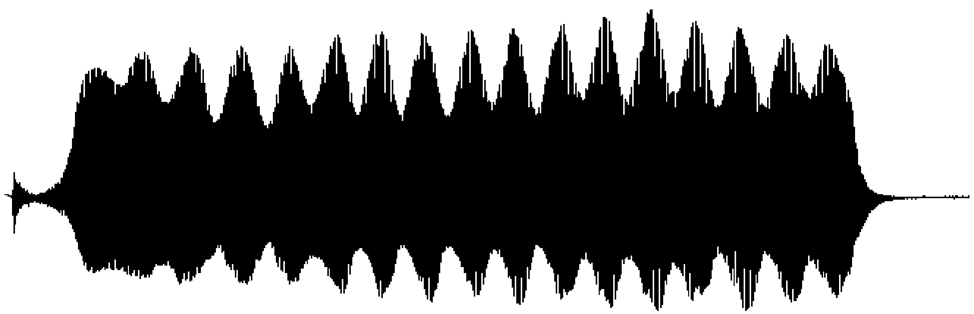
- Any given signal can be described in either of these domains





Time domain representation

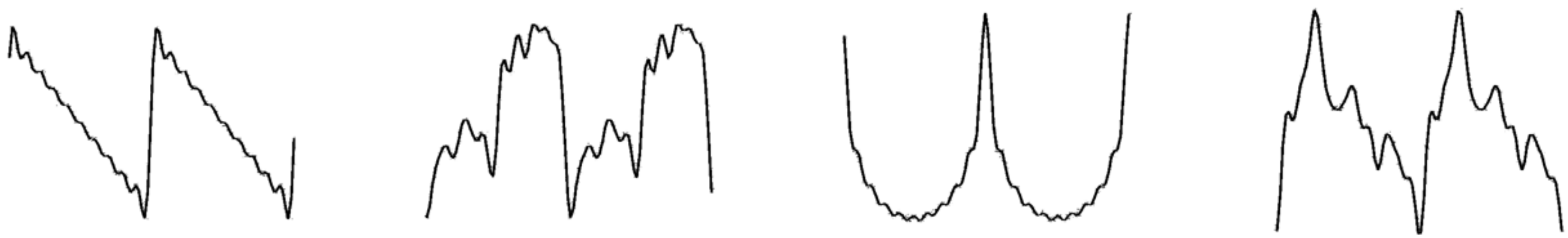
- Sound signal: variation of sound amplitude as a function of time
  - pressure, voltage
- What does it tell about the sound?
  - temporal features: attack time, amplitude envelope
  - periodicity



Time domain representation

- Waveforms of Four synthetic sounds

1 2 3 4

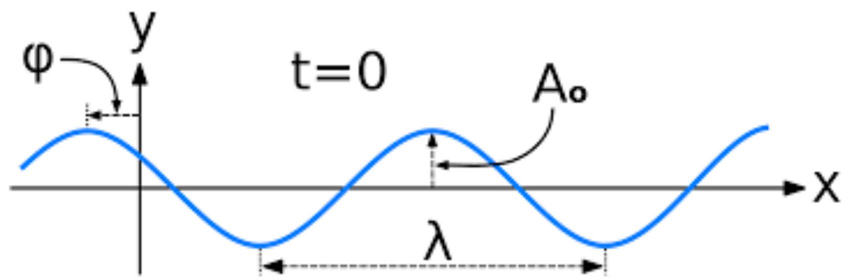


Time domain representation

- Sounds with different time domain representations may sound similar
- Representation in time domain is not always perceptually relevant
- More relevant representation is obtained in frequency domain

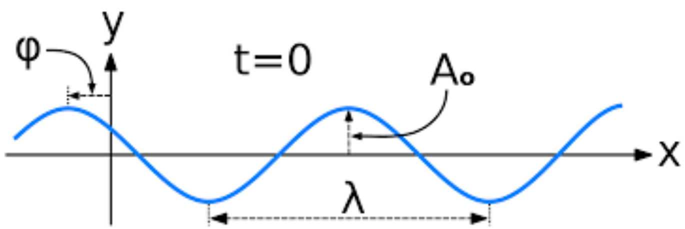
Frequency domain representation

- Representation of a waveform as **amplitudes/phases** of the individual frequency components vs. frequency
- Transformation between time and frequency domains:
  - Fourier transform
  - Wavelet transform

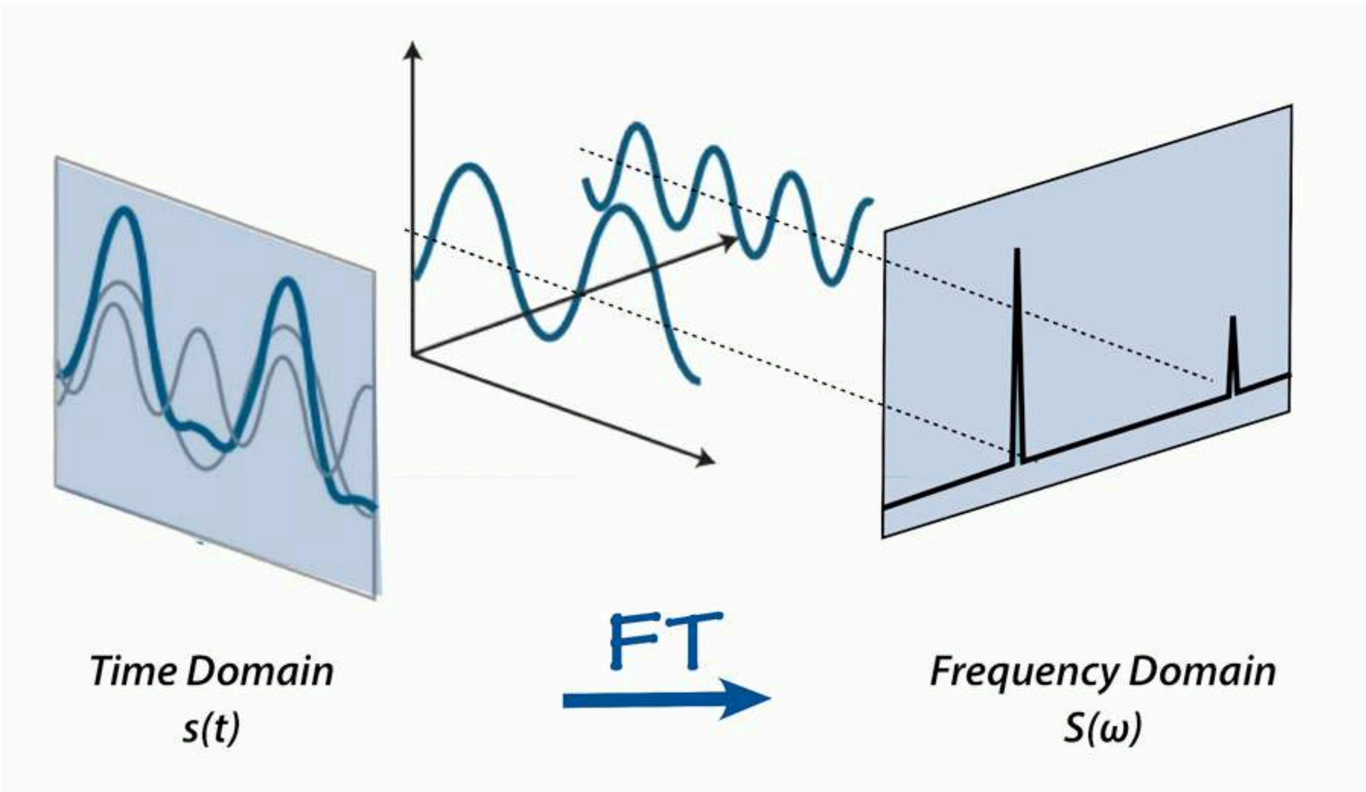


Fourier analysis

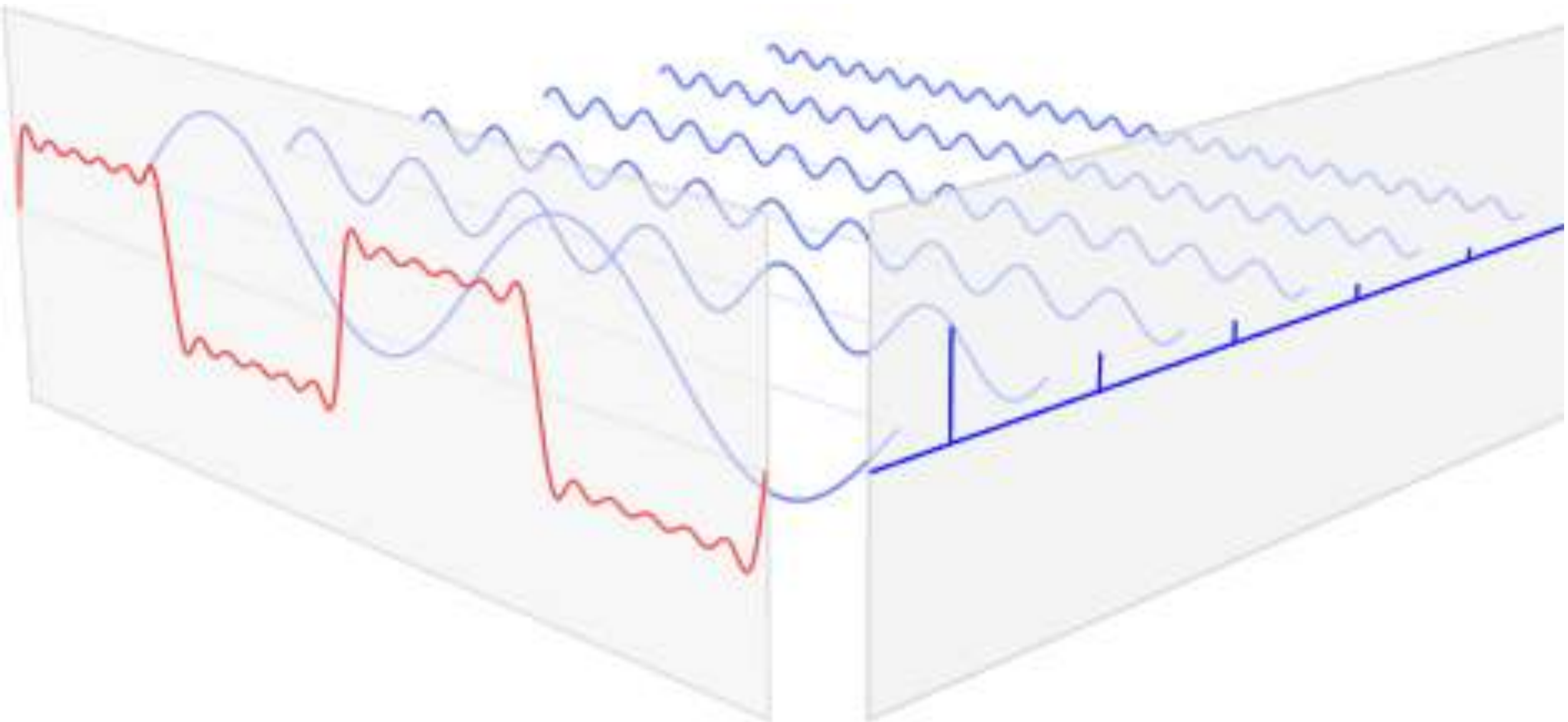
- Jean Baptiste Joseph Fourier (1768-1830)
- any continuous and periodic function can be represented as the sum of sine and cosine waves oscillating at different frequencies
- Each sine wave is characterized by
  - frequency
  - amplitude
  - phase



Fourier Transform (FT)



Fourier Transform (FT)



Fourier Transform (FT)

- Used for continuous (analog) signals

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{i2\pi\omega t} dt$$

$$A(\omega) = \left| X(\omega) \right| \qquad \text{amplitude spectrum}$$

$$\phi(\omega) = \angle X(\omega) \qquad \text{phase spectrum}$$

- continuous spectrum

Discrete Fourier Transform (DFT)

- Used for time-discrete (sampled) signals

$$X(\omega_m) = \sum_{k=-\infty}^{\infty} x(t_k) e^{i2\pi\omega_m t_k}$$

$$A(\omega_m) = \left|X(\omega_m)\right| \qquad \text{amplitude spectrum}$$

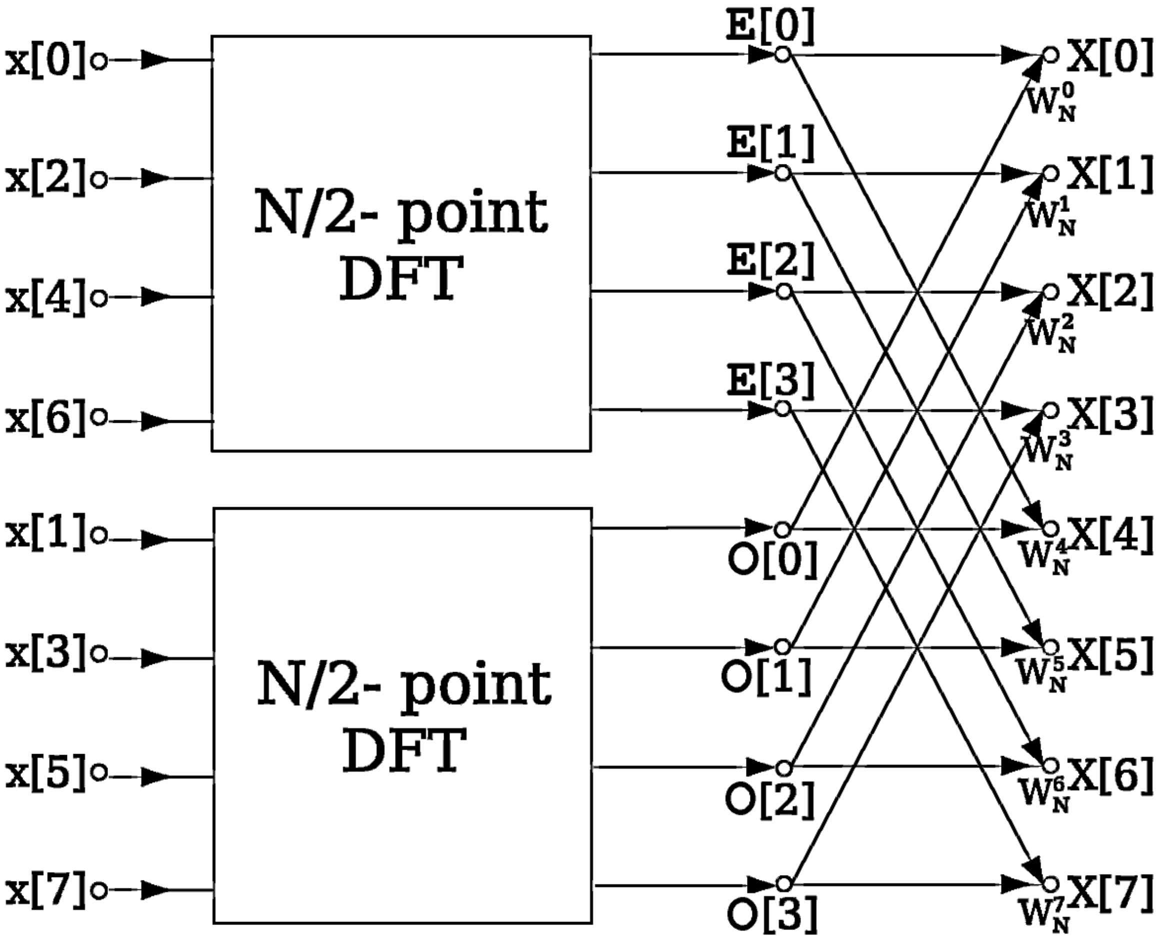
$$\phi(\omega_m) = \angle X(\omega_m) \qquad \text{phase spectrum}$$

- Discrete spectrum (frequency bins)
- Usually only amplitude spectrum is considered

DFT and FFT

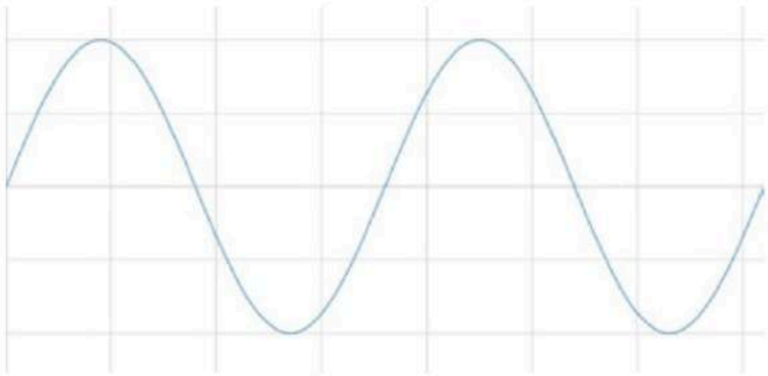
- Fast Fourier Transform (FFT) efficient algorithm for computing DFT
  - (Cooley & Tukey, 1965)
- Computational complexity
  - DFT: $O(N^2)$
  - FFT: $O(N \log N)$

FFT



x[0]
x[2]
x[4]
x[6]
x[1]
x[3]
x[5]
x[7]
N/2- point DFT
N/2- point DFT
E[0]
E[1]
E[2]
E[3]
O[0]
O[1]
O[2]
O[3]
X[0]
X[1]
X[2]
X[3]
X[4]
X[5]
X[6]
X[7]
W_N^0
W_N^1
W_N^2
W_N^3
W_N^4
W_N^5
W_N^6
W_N^7

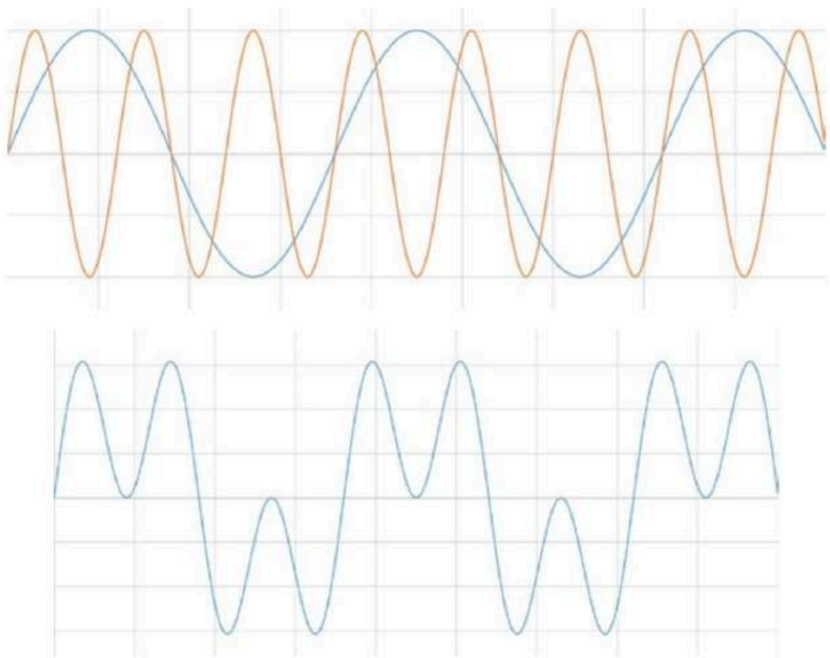
How does a sine wave sound?



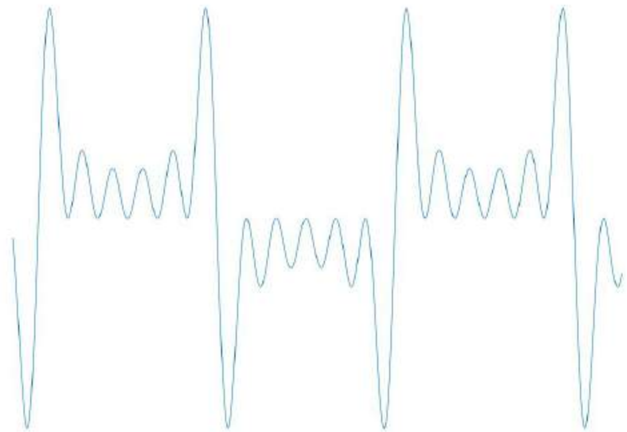
150 Hz

Sine wave + **Odd** Harmonics

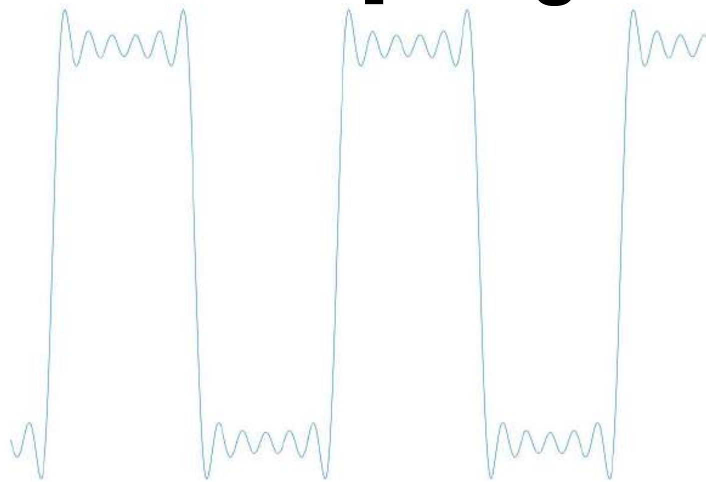
150 Hz + 450 Hz



[1 3 5 7 9 11]

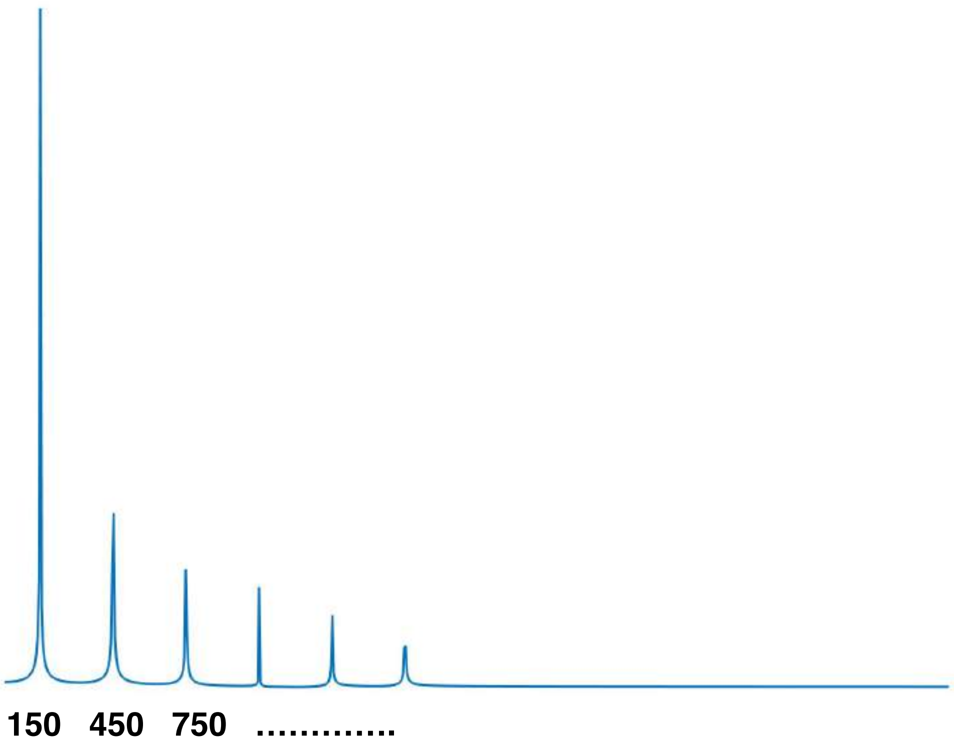


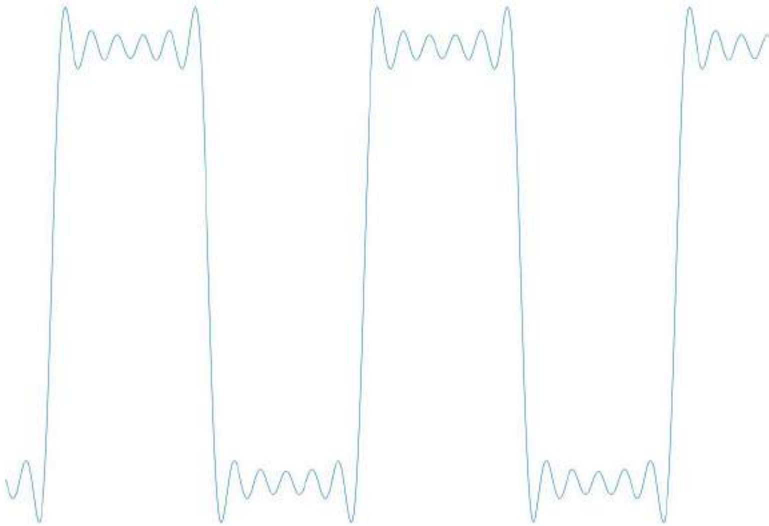
[1 3 5 7 9 11]-weighted



Sine wave + **Odd** Harmonics

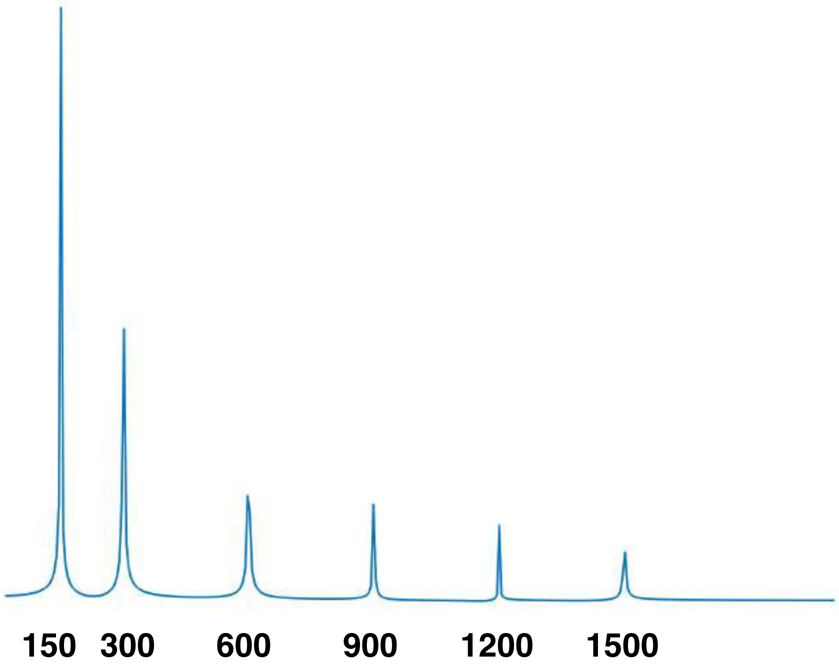
spectrum?

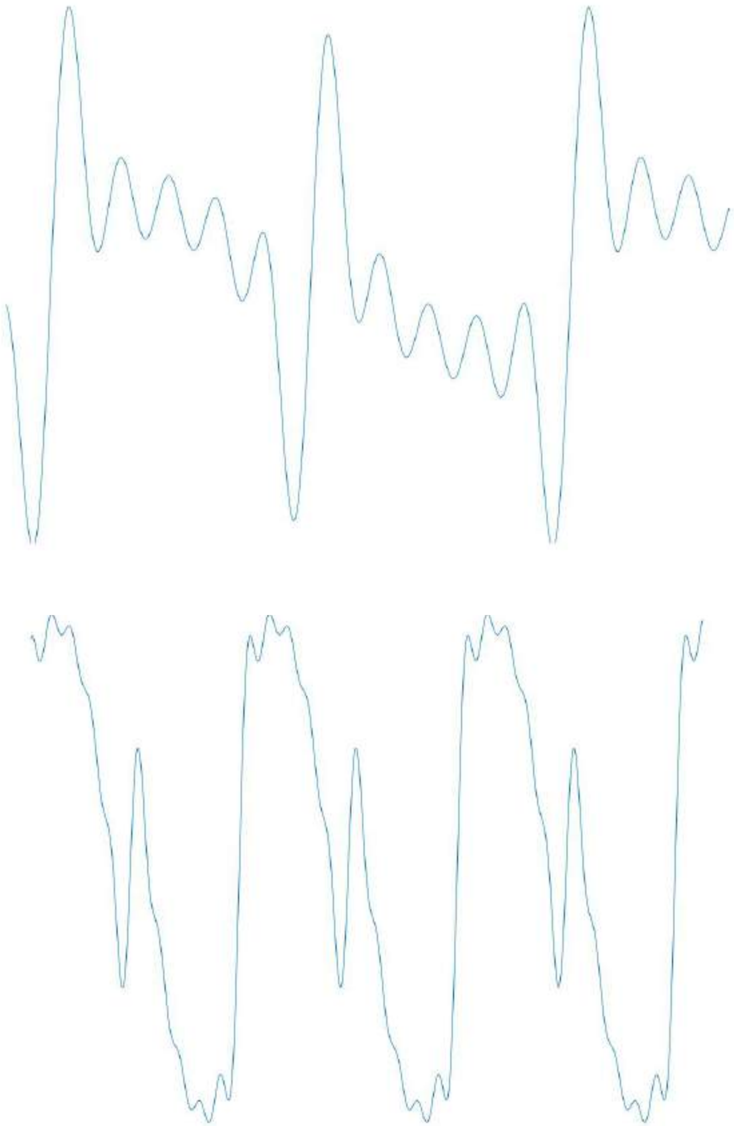




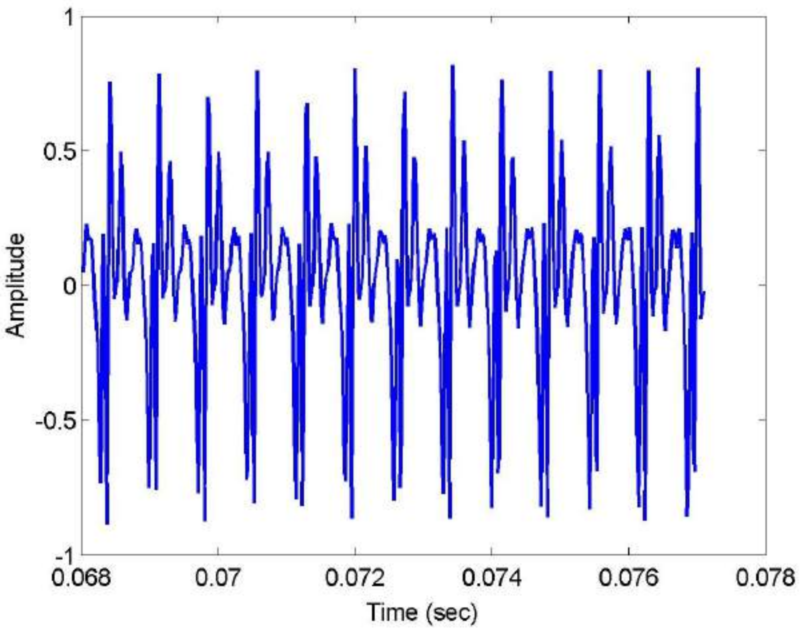
Sine wave + **Even** Harmonics

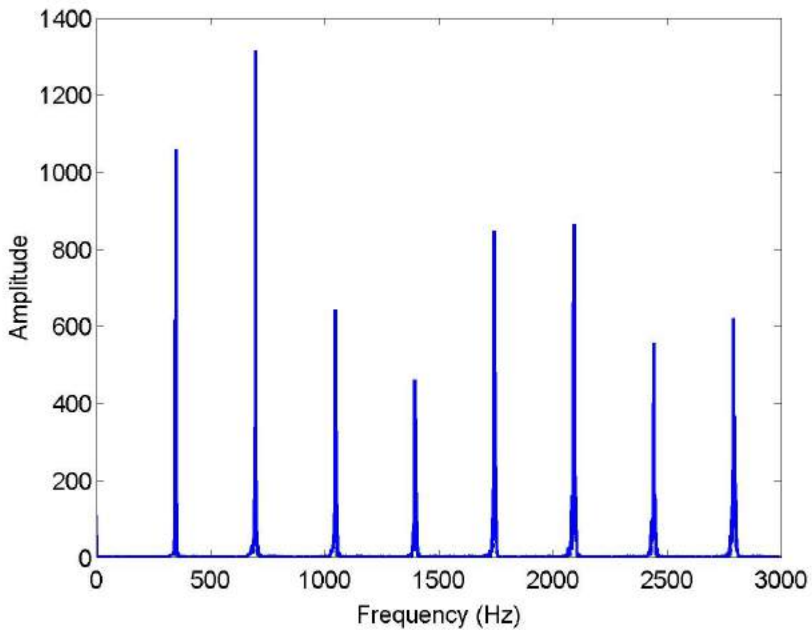
spectrum?





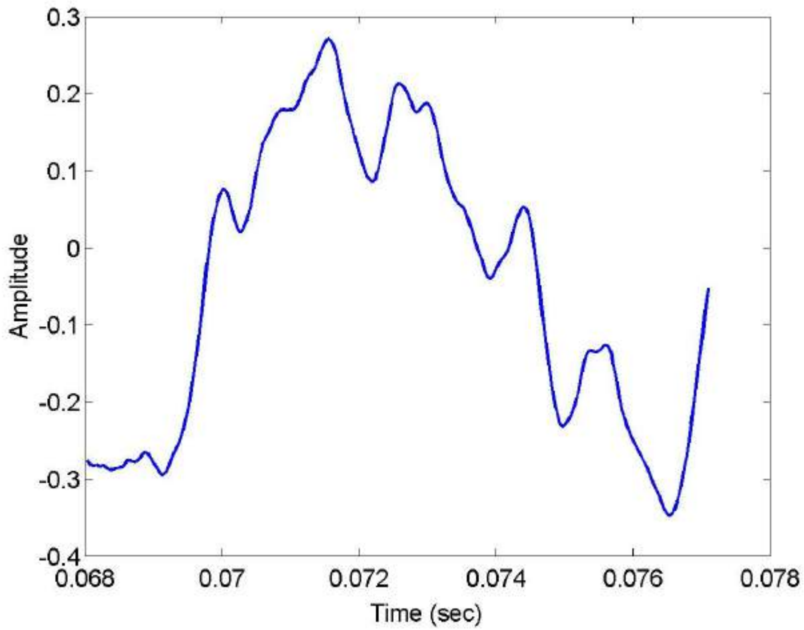
example 1

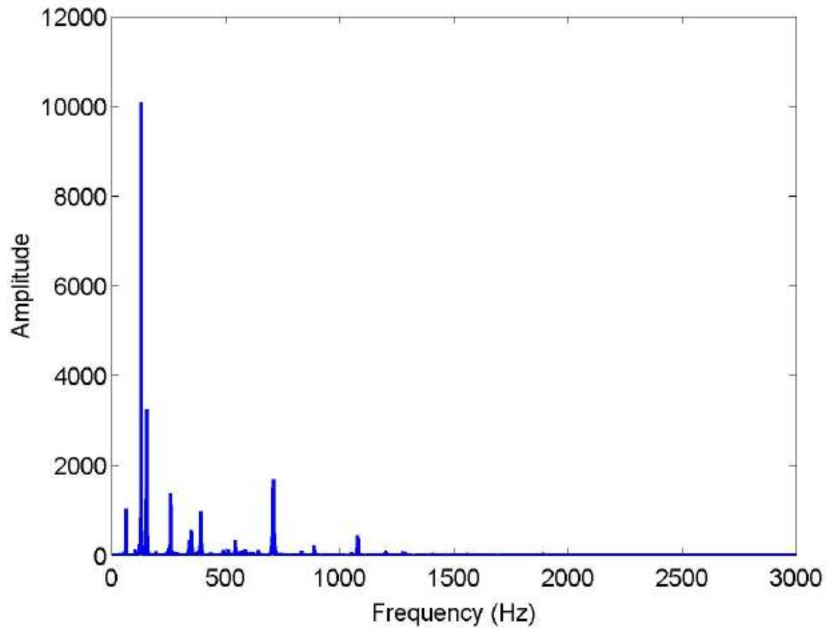




- harmonic sound

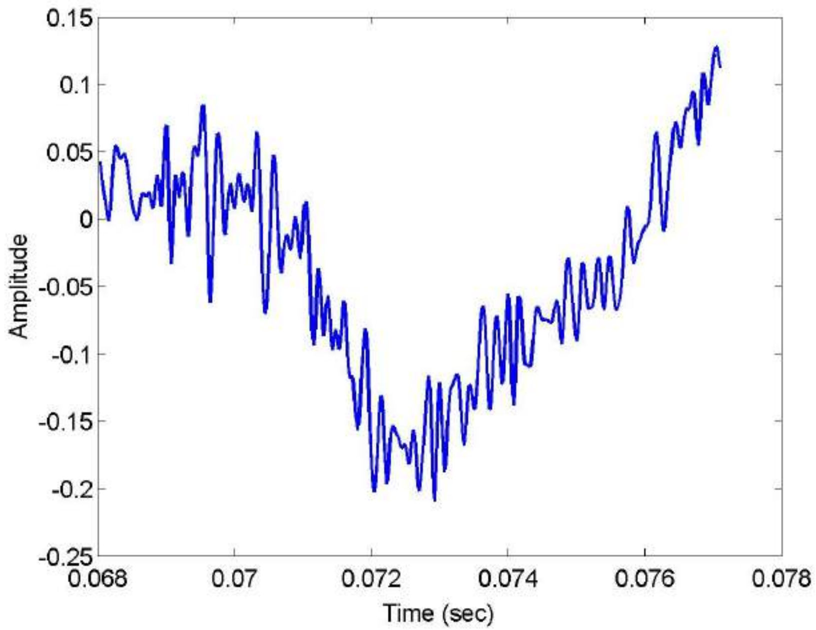
example 2

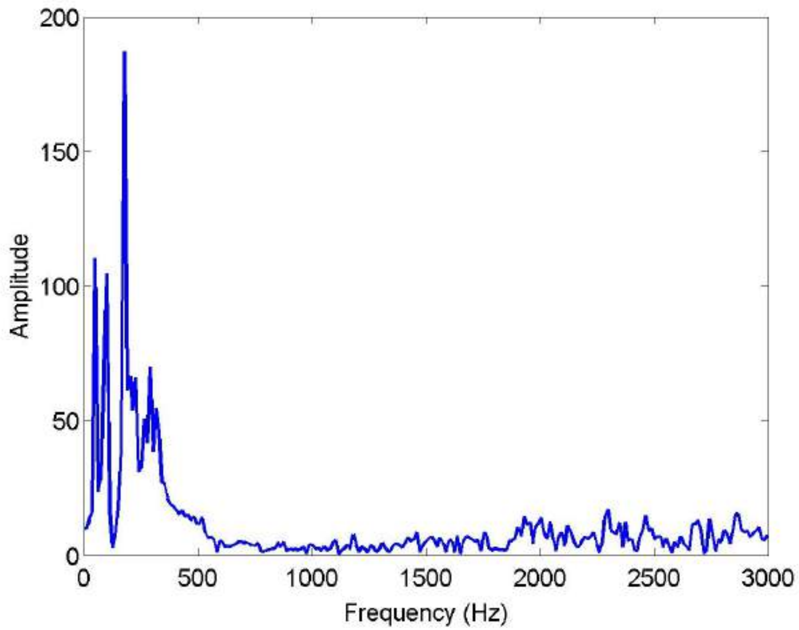




- non-harmonic sound

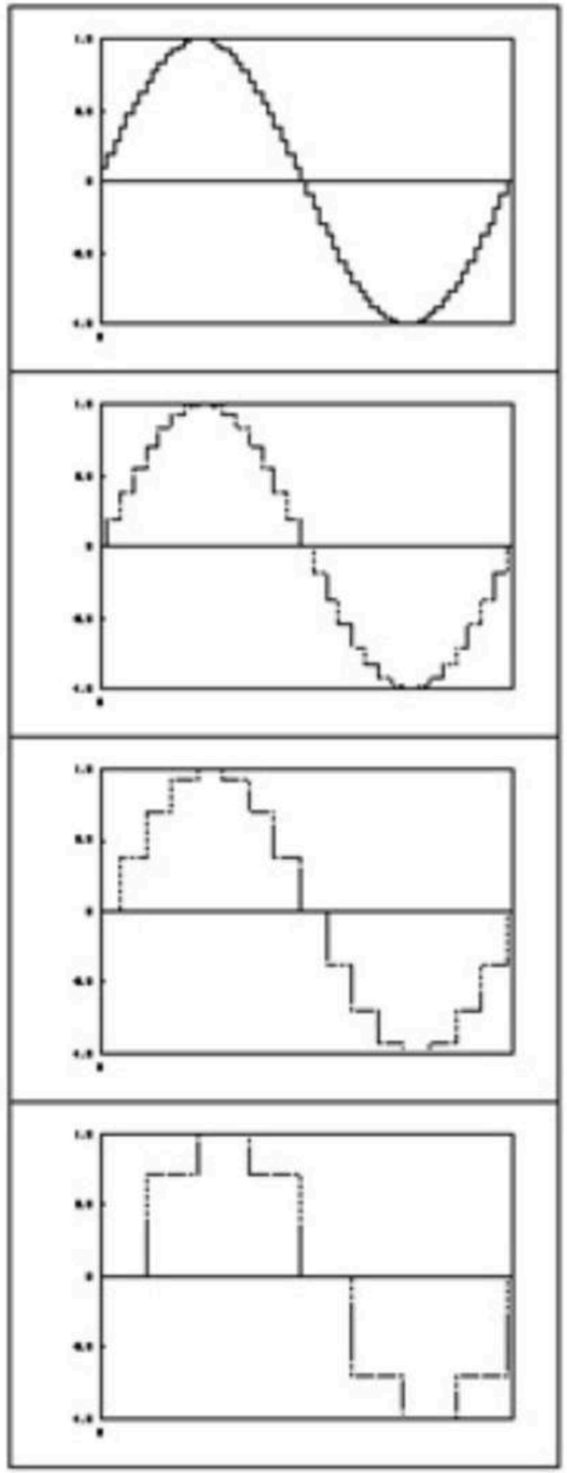
example 3



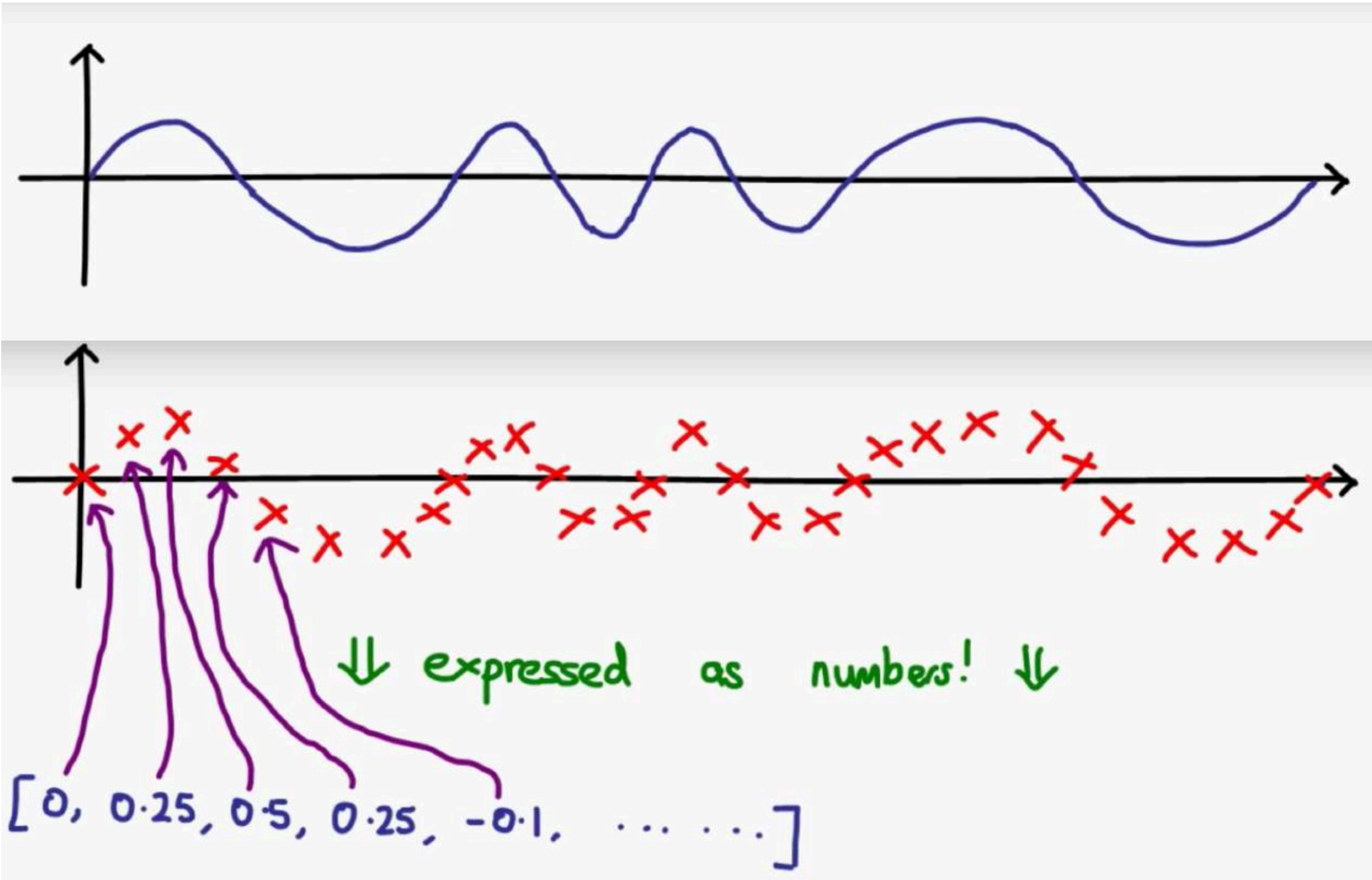


- noisy sound

Sampling, Aliasing, Nyquist Theorem



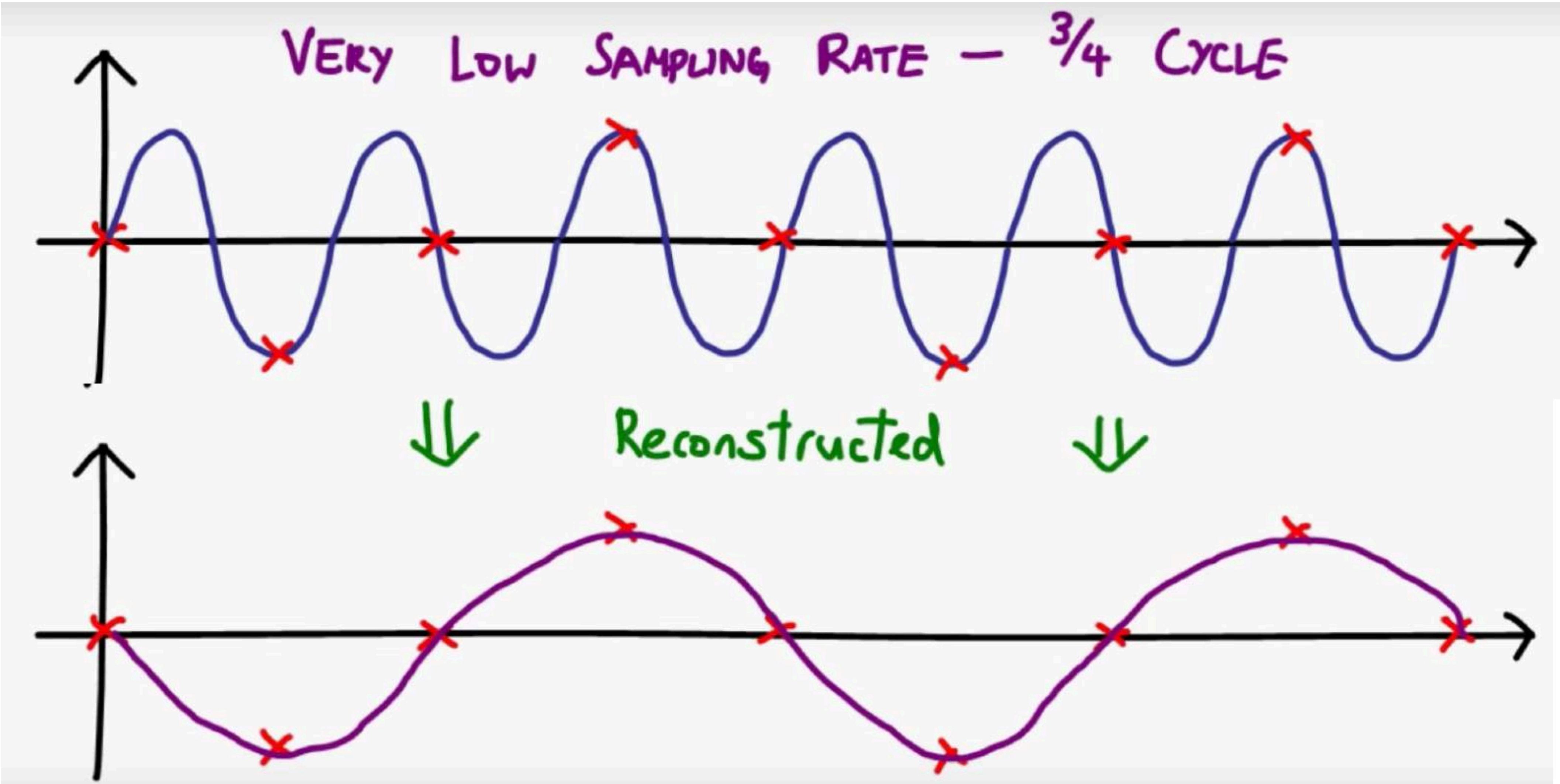
Sampling



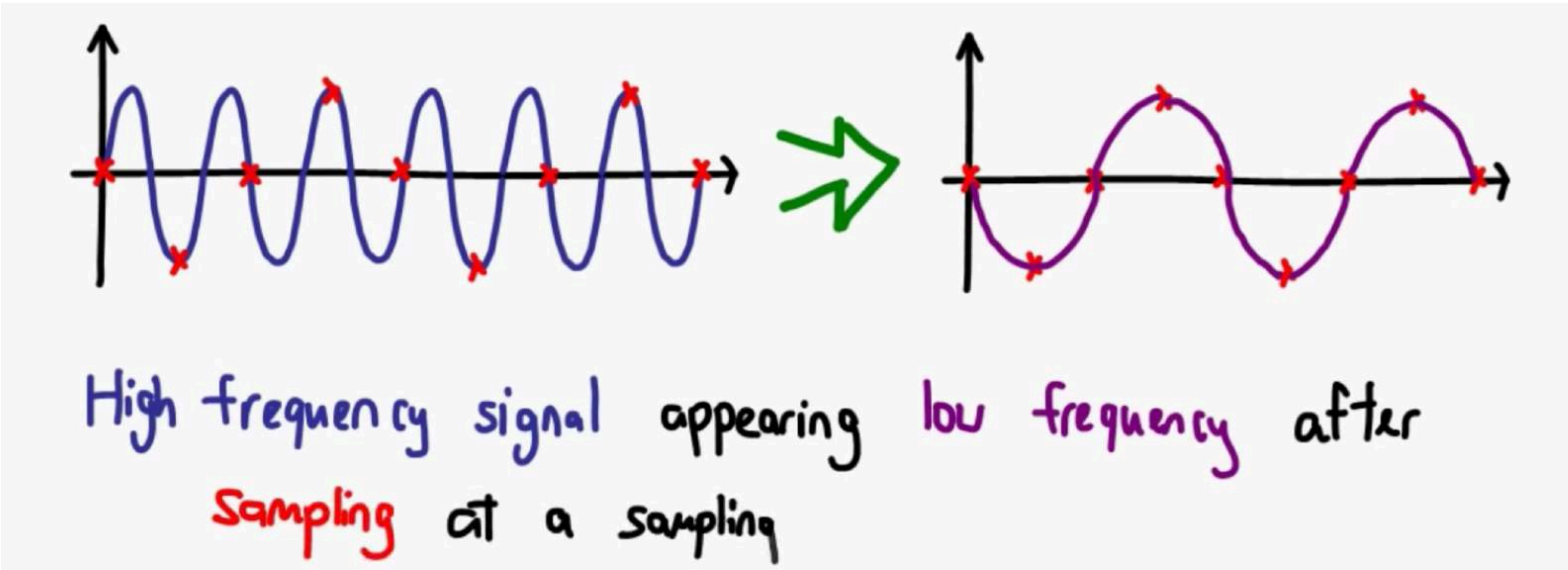
Sampling Theorem

- a fundamental bridge between continuous-time signals (often called "analog signals") and discrete-time signals
- establishes a sufficient condition for a sample rate that permits a discrete sequence of samples to capture all the information from a continuous-time signal of finite bandwidth
- Nyquist-Shannon Sampling Theorem
- for a given sample rate fs, perfect reconstruction is possible for a bandlimit B < fs/2.
- **Nyquist frequency** = $f_s/2$

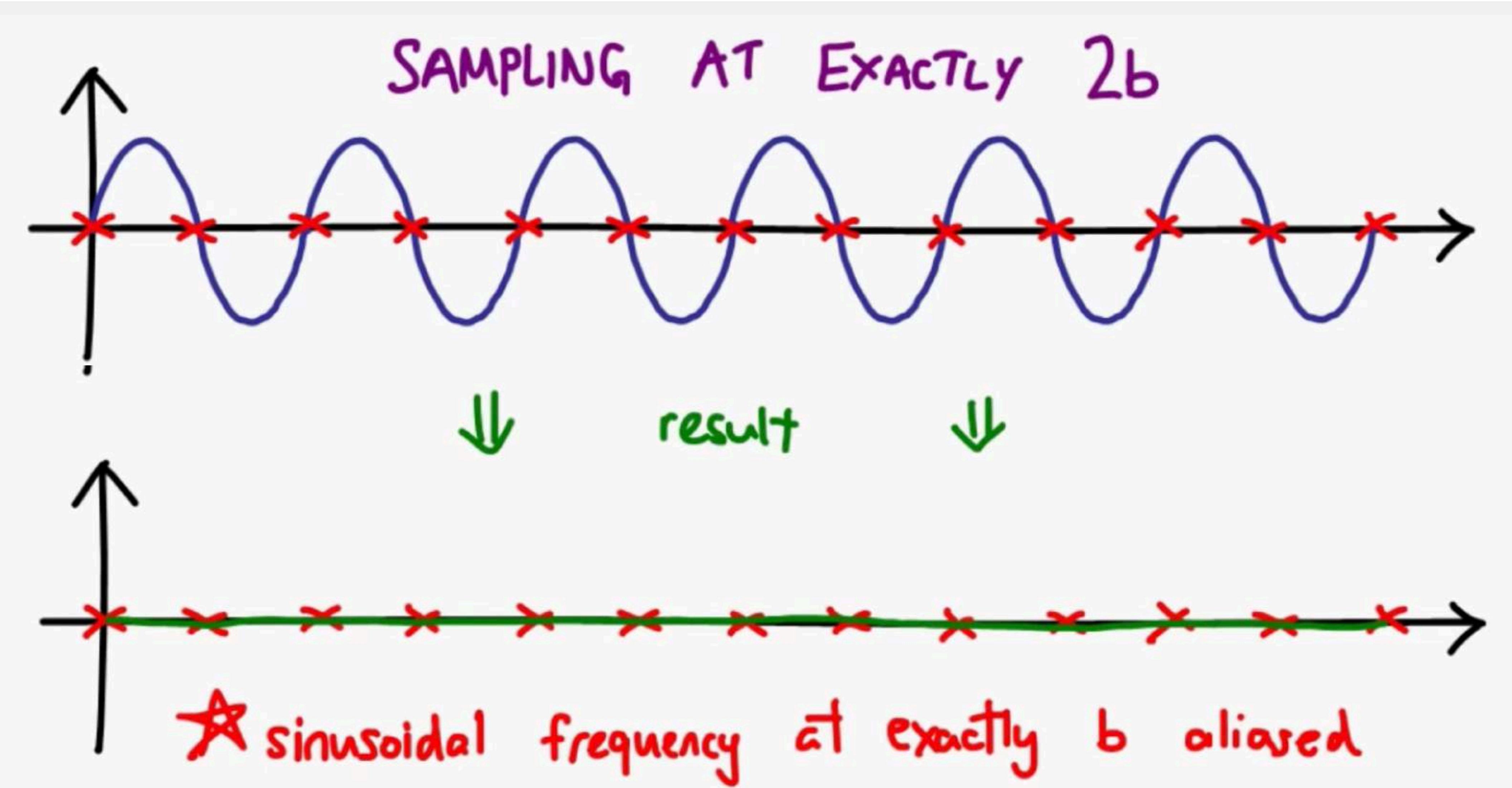
Sampling Theorem



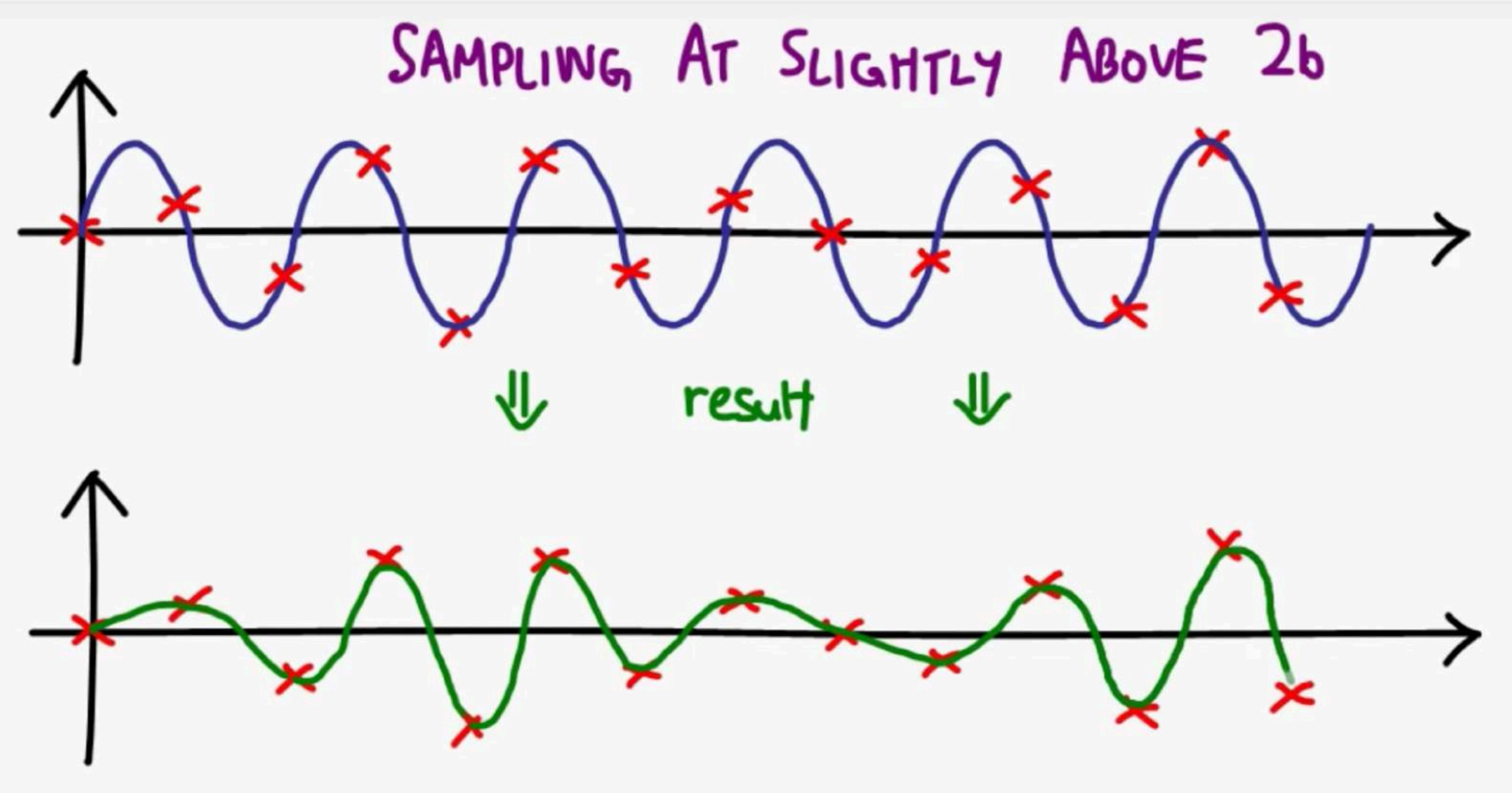
Aliasing


High frequency signal appearing low frequency after
sampling at a sampling

Sampling Rate



Sampling Rate



Nyquist Rate vs Nyquist frequency

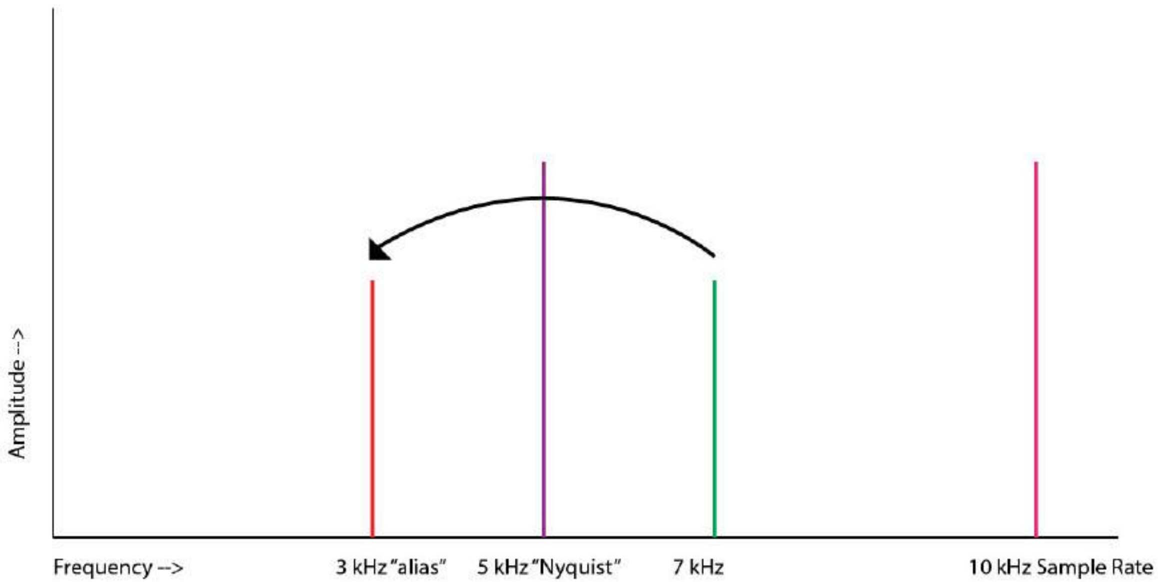
- min. sampling rate to prevent aliasing

- max. frequency that will not alias given a sampling rate

- Why are audio files sampled at 44.1 KHz?
- What is the Nyquist rate and Nyquist frequency for such audio files?

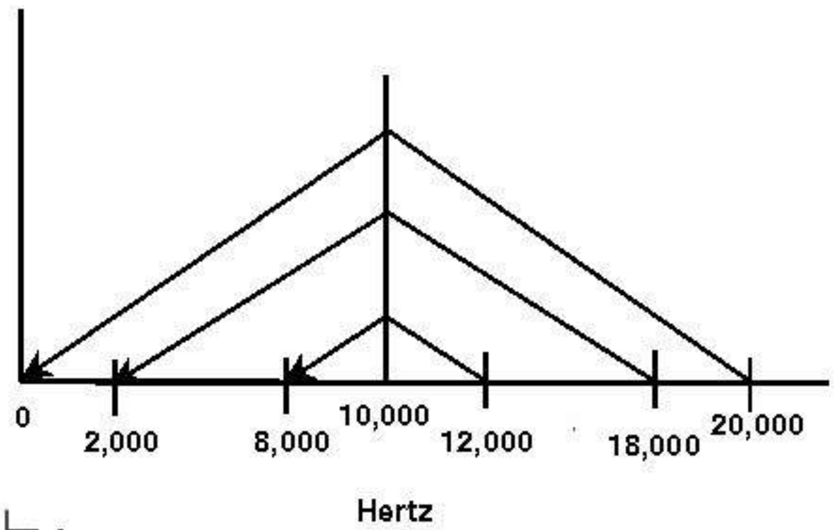
Aliasing

Foldover



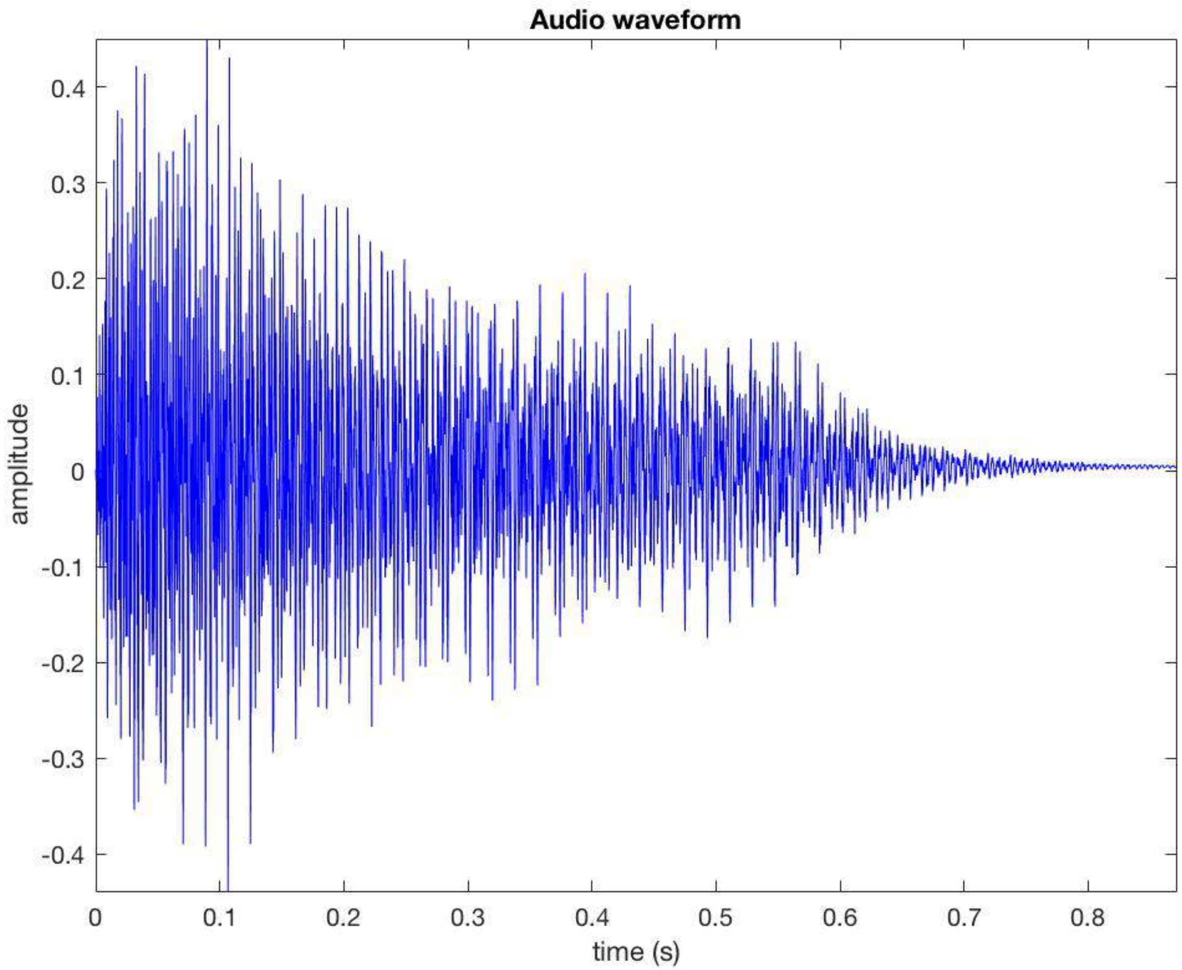
Aliasing

$$f' = |f - SR|$$



- Example:
  - SR = 20,000 Hz
  - Nyquist Frequency = 10,000 Hz
  - f = 12,000 Hz --> f' = 8,000 Hz
  - f = 18,000 Hz --> f' = 2,000 Hz
  - f = 20,000 Hz --> f' = 0 Hz

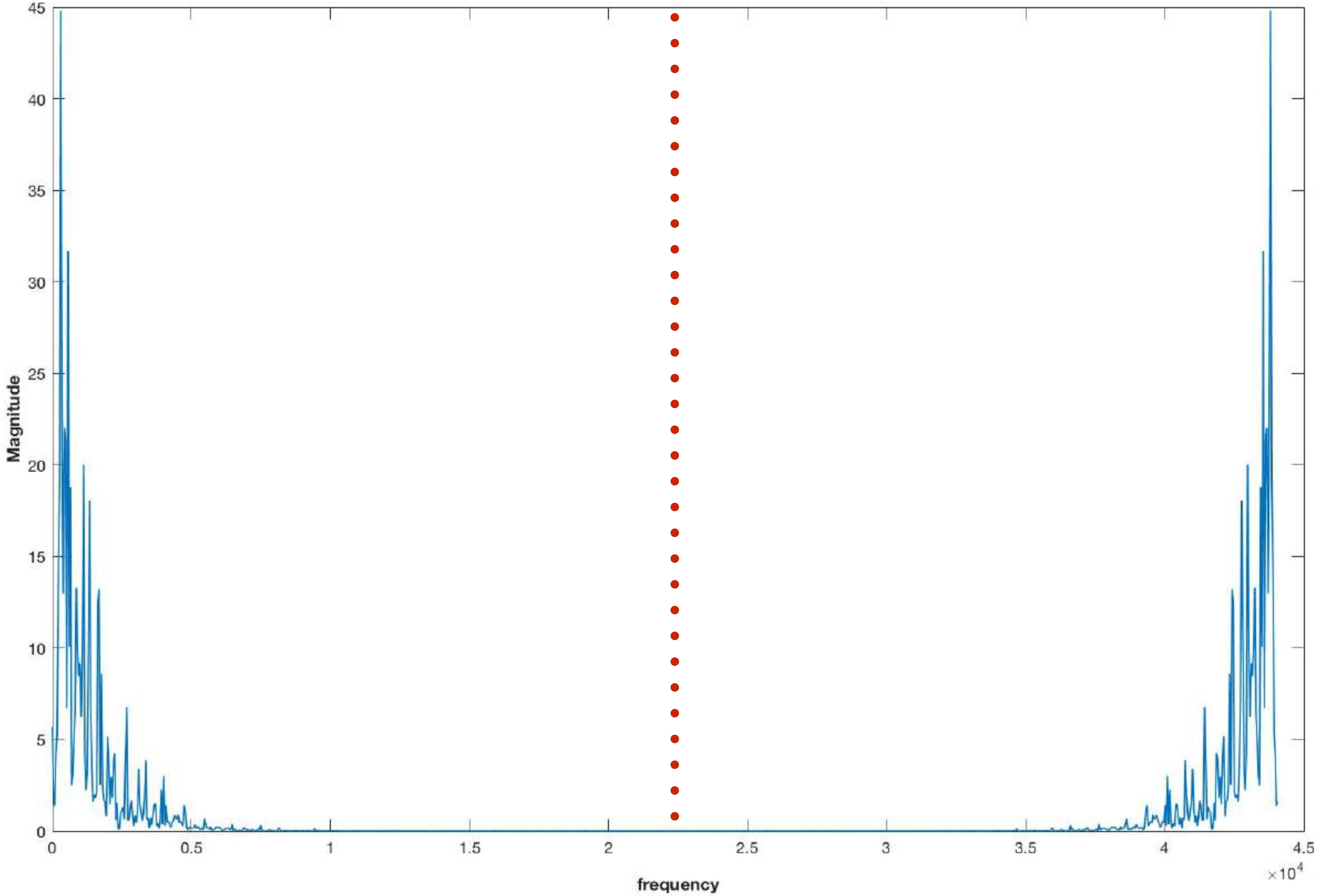
Aliasing



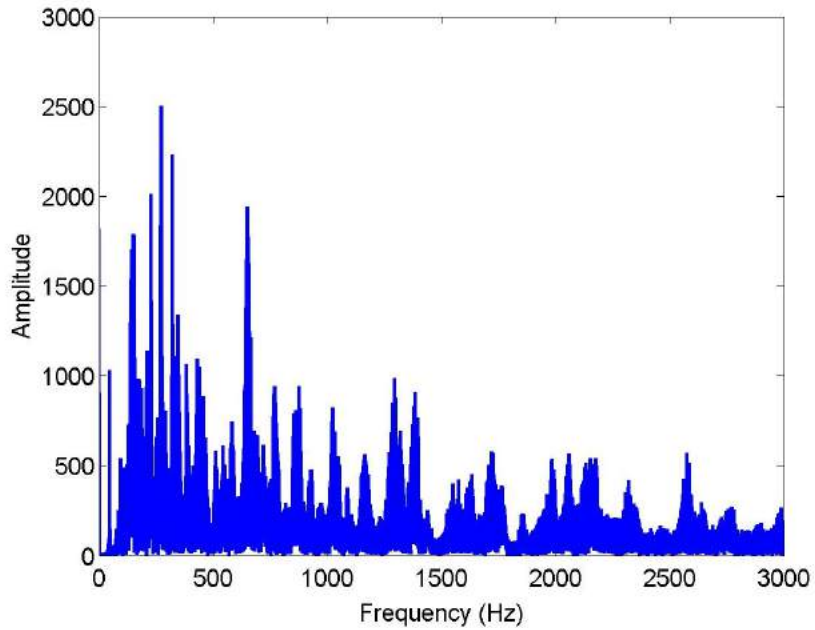
Nyquist rate = 44100 Hz

Aliasing

Nyquist frequency = 22050 Hz



FFT for music?



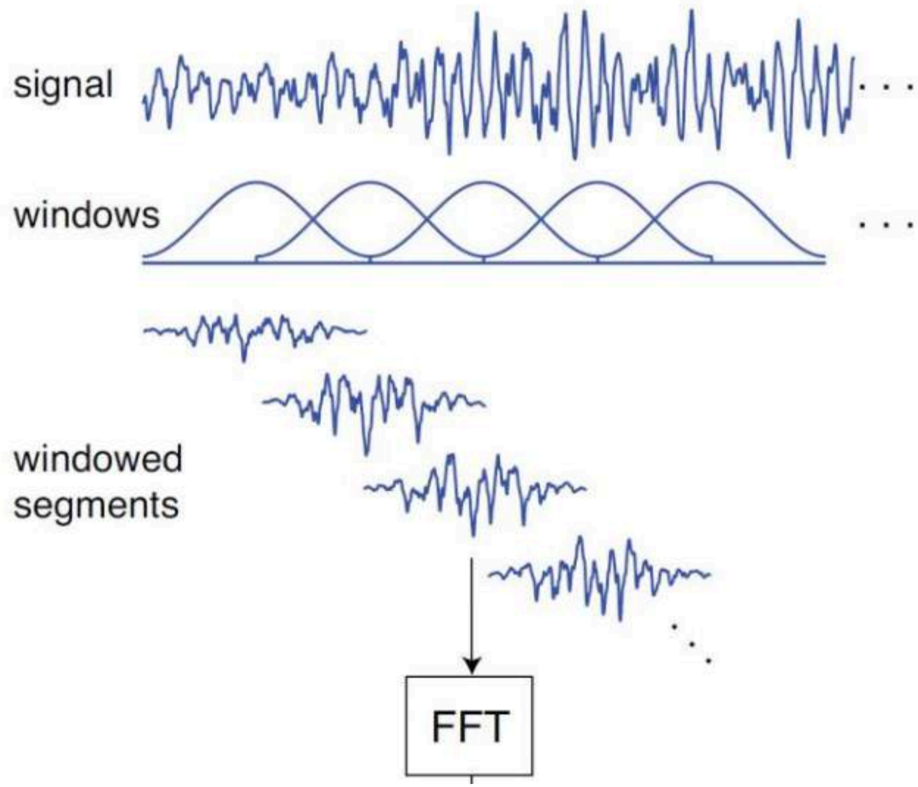
FFT & STFT

- For long sound signals Fourier transform is not meaningful, because it does not display temporal evolution

- Solution: short-time Fourier transform (STFT)

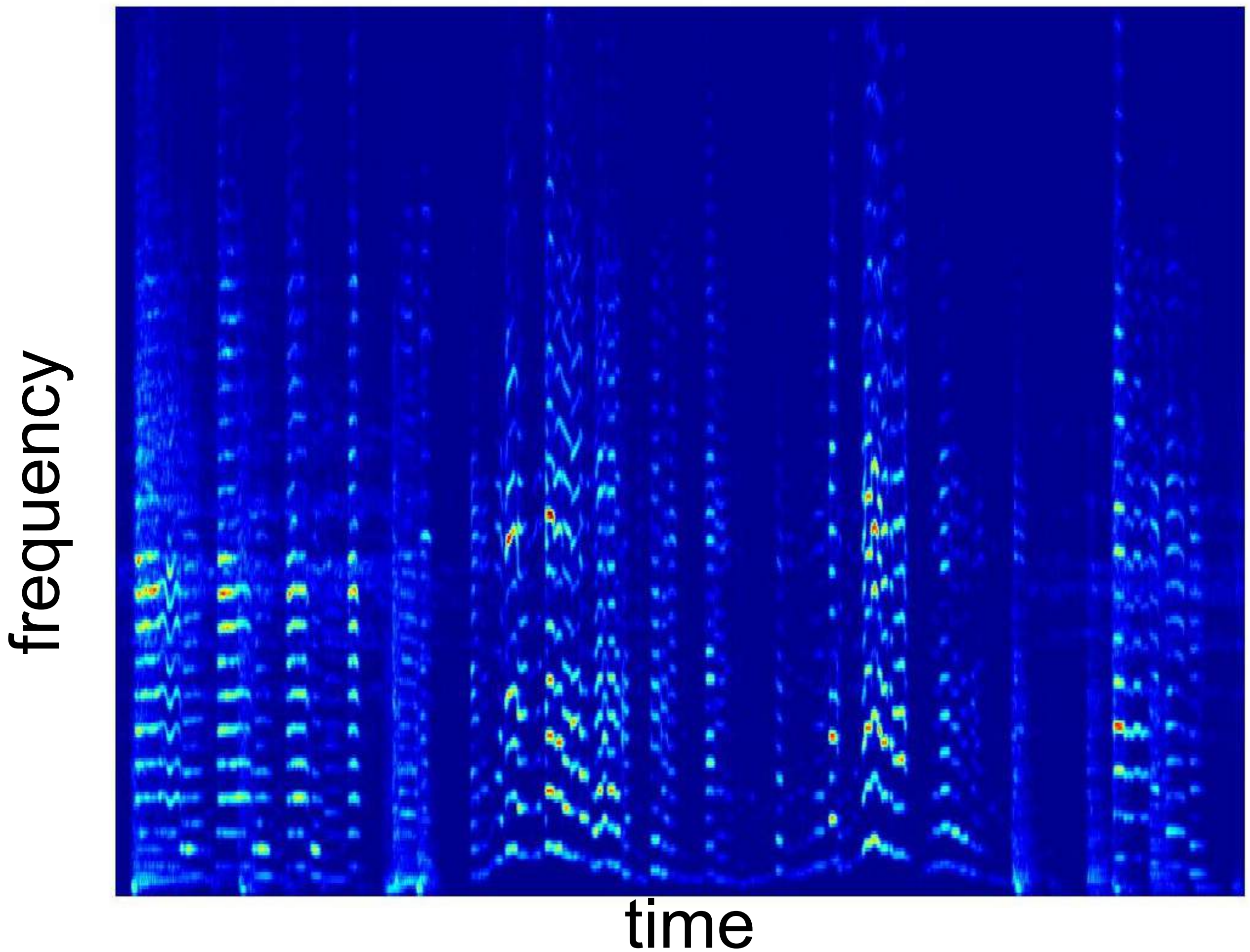
STFT

- Long sound signal is divided into **short** segments
- Segments overlap (Hop factor/size)
- Segments are **windowed**
- FFT is computed for each segment



Spectrogram

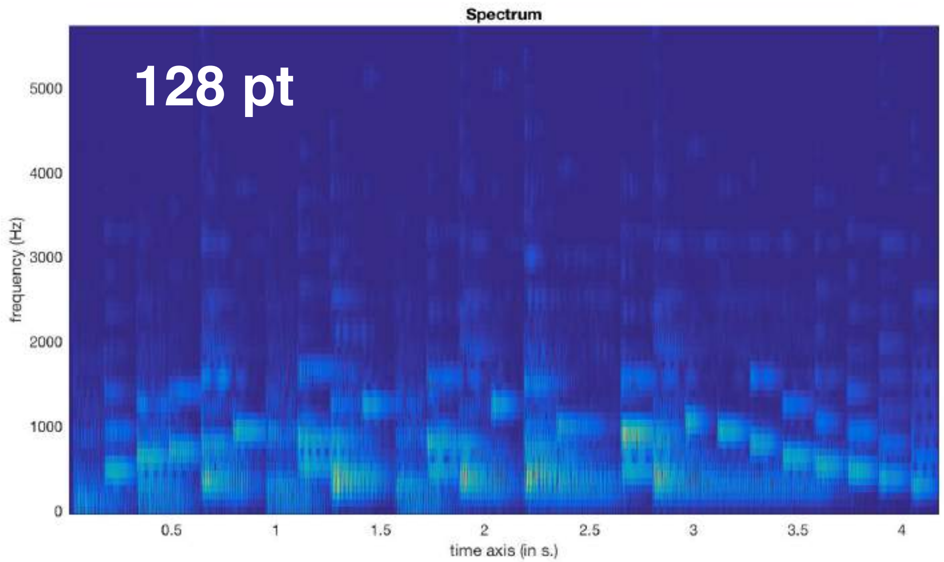
- A visual display of STFT

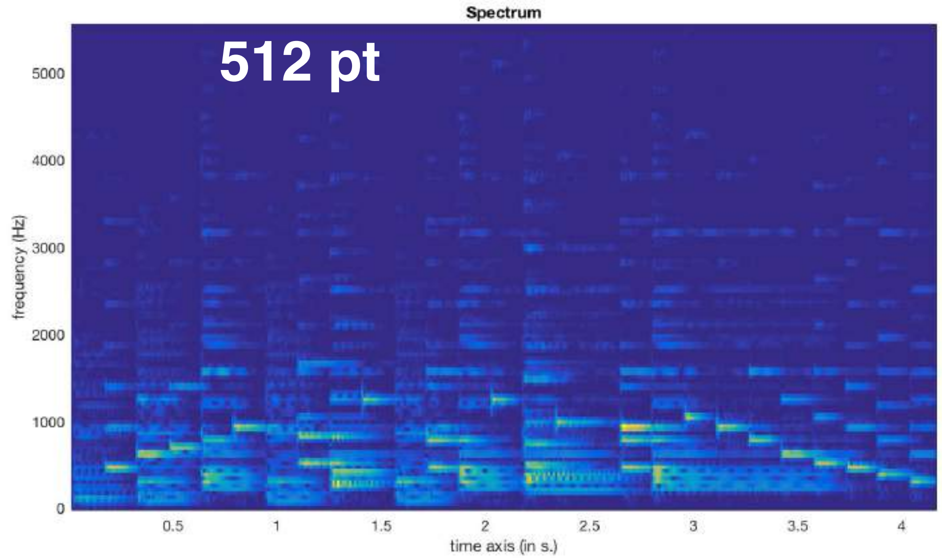


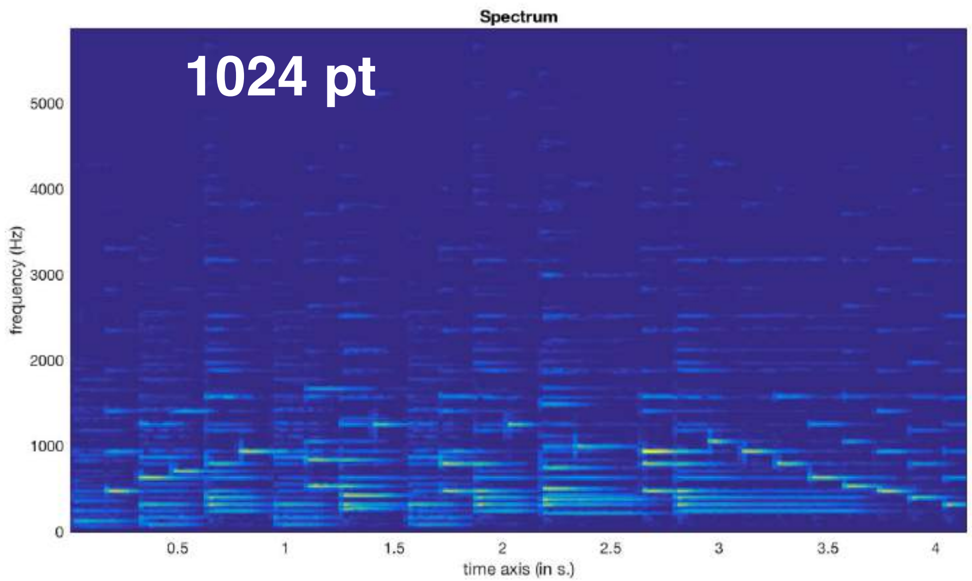
STFT

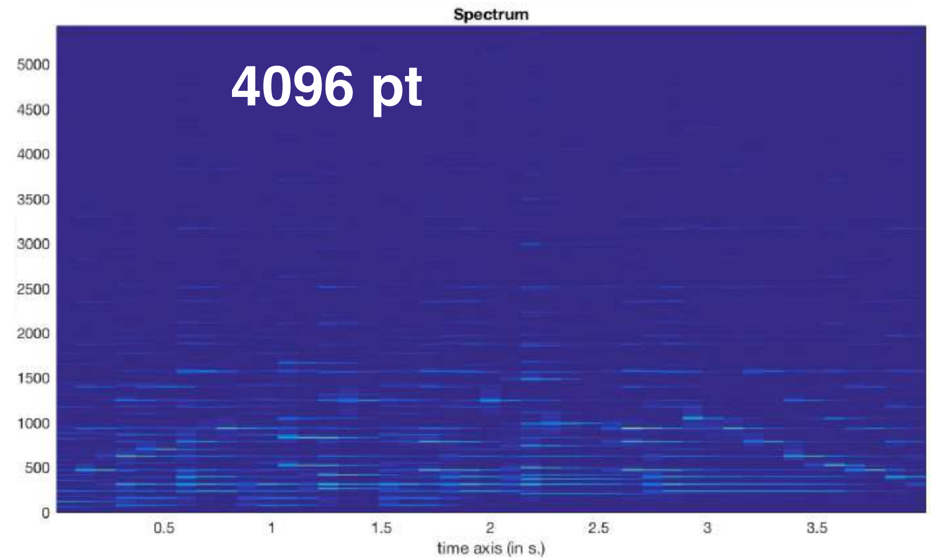
- For a given sample rate (**f_s**), the corresponding frequency range of the representation is split into a number of bins
- the frequency resolution (**FR**) is the frequency band of a bin
- the more bins, the more slices of frequency range we get, and the more precise these slices are

Temporal vs Frequency resolution







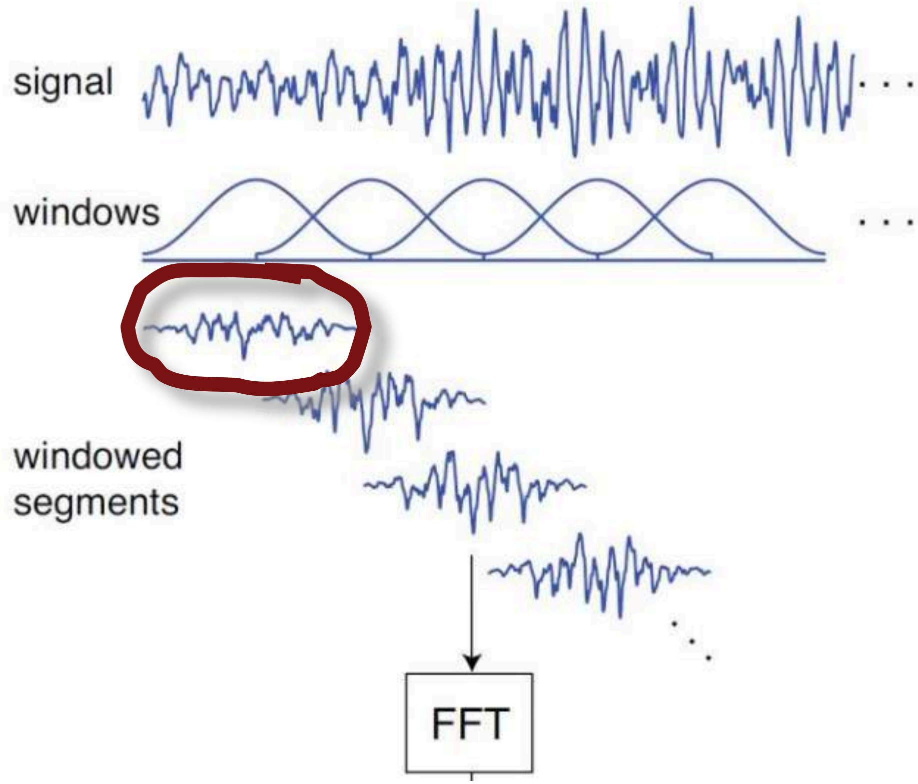


How short a window?

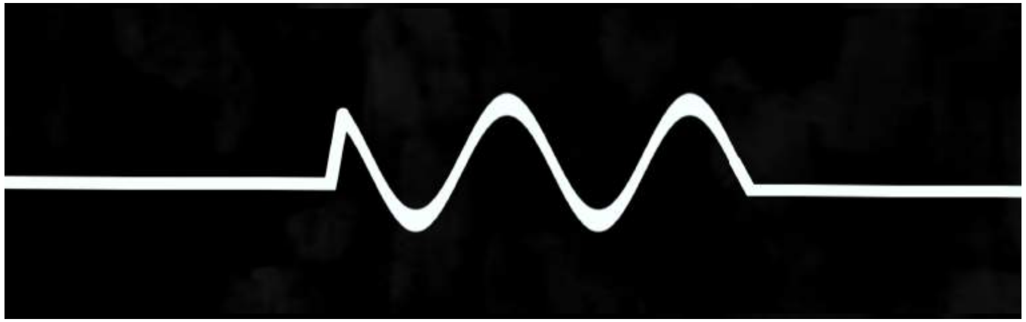
- STFT can have either a good temporal resolution or a good frequency resolution

- Short window -> high temporal resolution

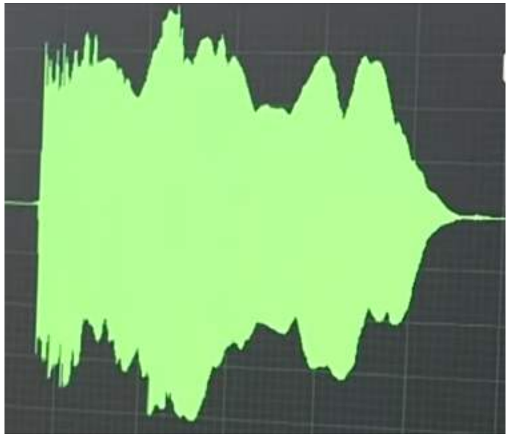
- Long window -> high frequency resolution



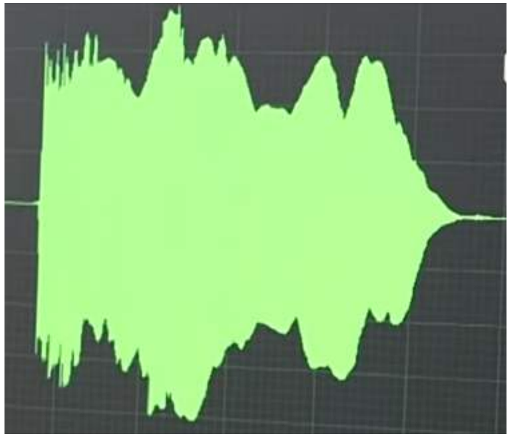
Uncertainty principle

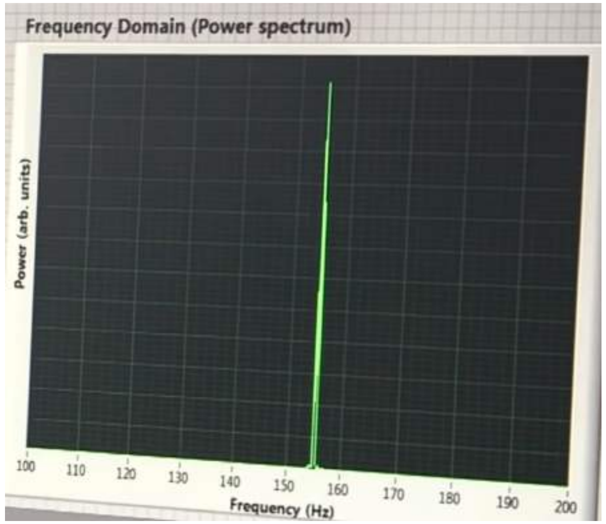


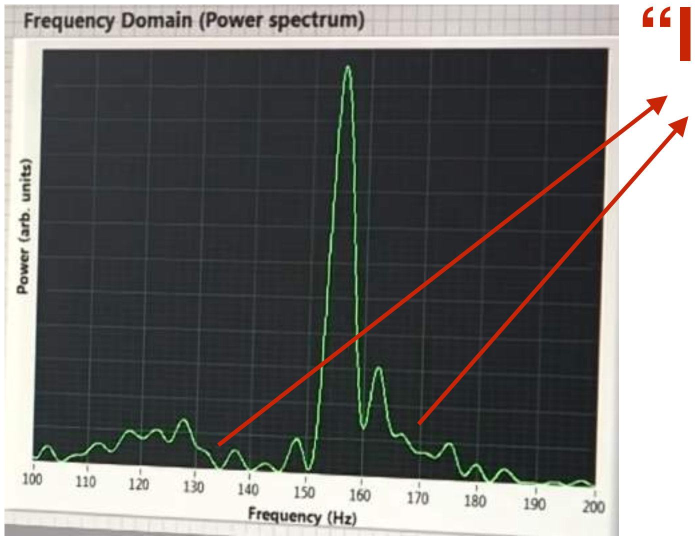
Uncertainty principle



Uncertainty principle







“leakage”

Temporal vs. Frequency Resolution

- f_s = 44100 Hz **Nyquist rate**
- f_{max} = 22050 Hz **Nyquist frequency**
- desired freq resolution (FR) = **10** Hz
- With a 1024 window size, we get
  - FR = 44100/1024 = 43.066
  - The spectrum is equally split into bins of 43.066 Hz width
- With a 4096 window size, we get
  - FR = 44100/4096 = **10,76**
  - **this is more precise**

temporal window (in ms) used?

4096/44100 = ~93ms

Temporal vs Frequency resolution

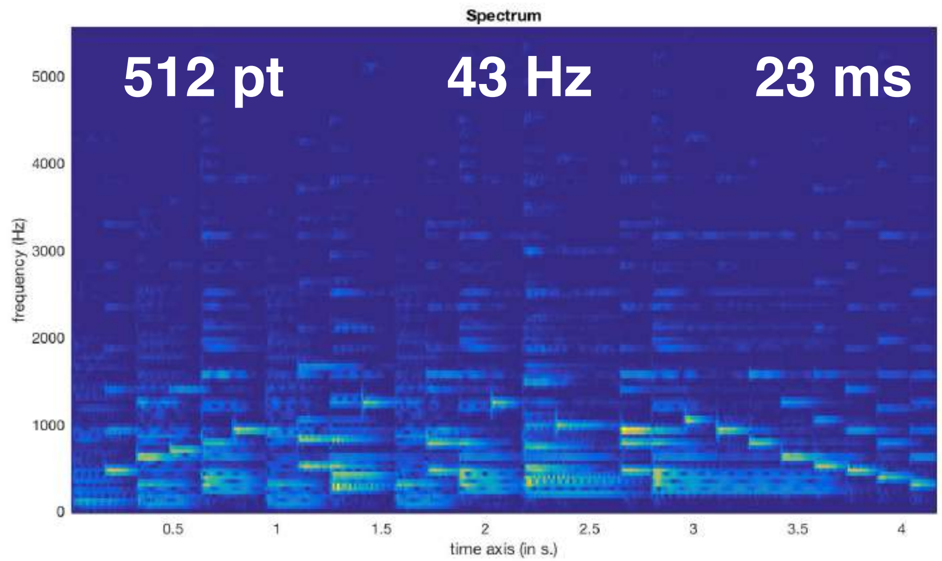
f_s = 44100 Hz

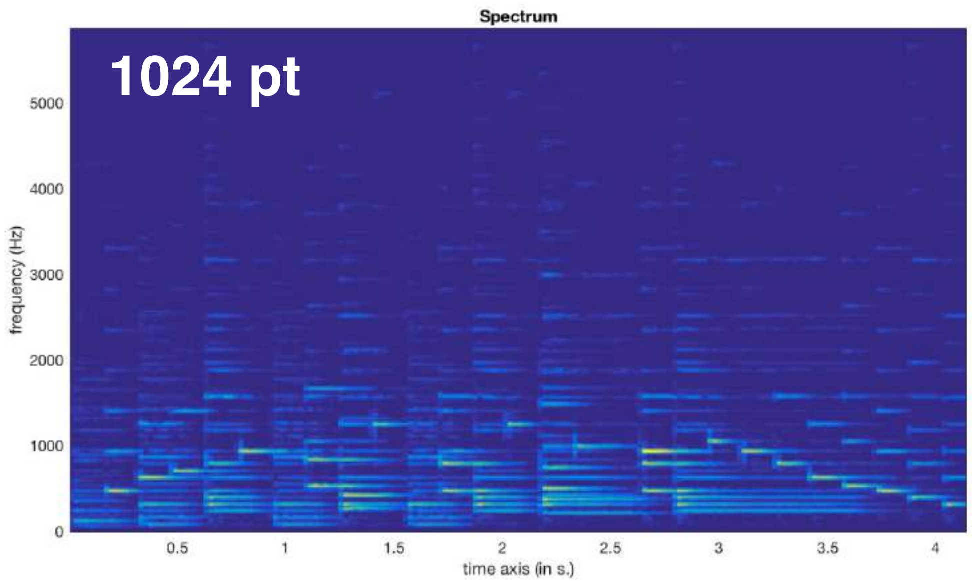
512 pt = spectrum until f_{max} divided into 512 bins

or f_s divided into 1024 bins

each bin = ____ Hz?

temporal window (in ms) used?
1 sec = 44100 samples

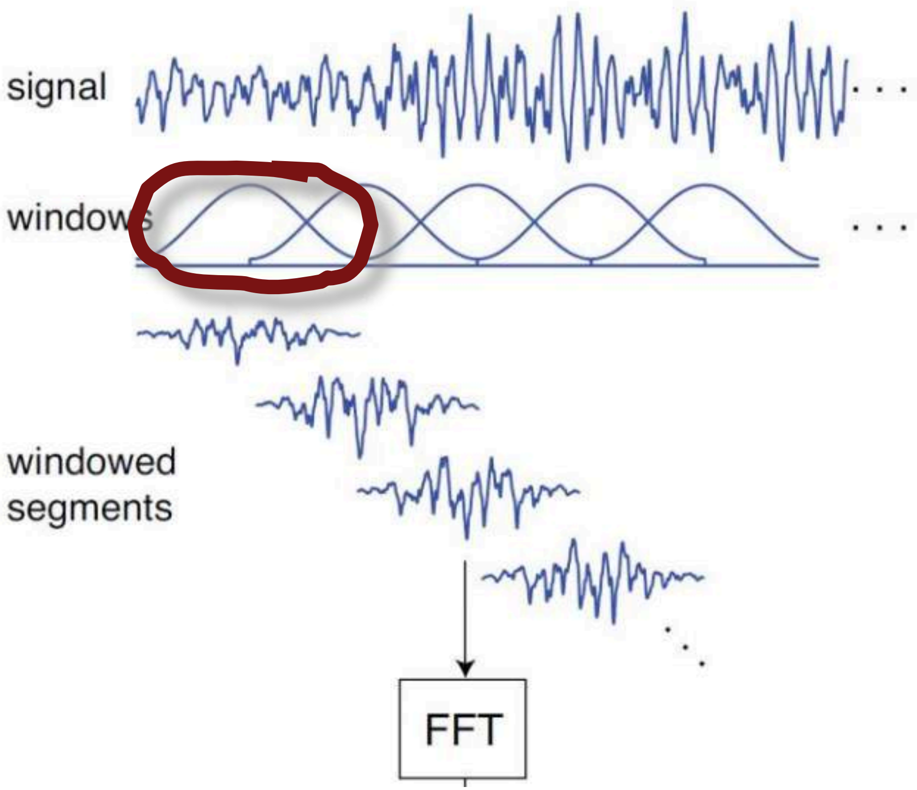




1024 pt = spectrum until f_{max} divided into 1024 bins
freq resolution?
temporal window (in ms) used?

freq resolution = 21.53 Hz
temporal window = 46 ms

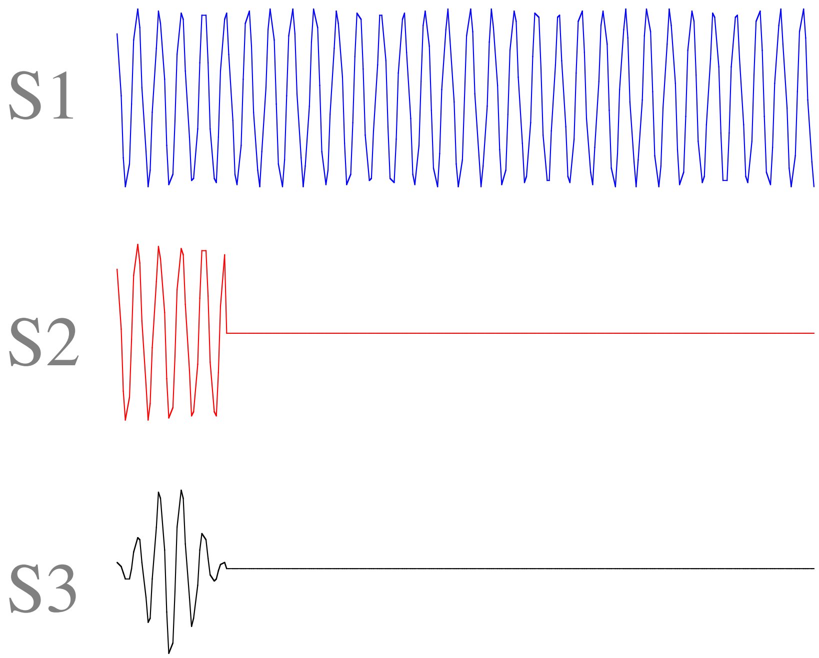
Why is windowing necessary?

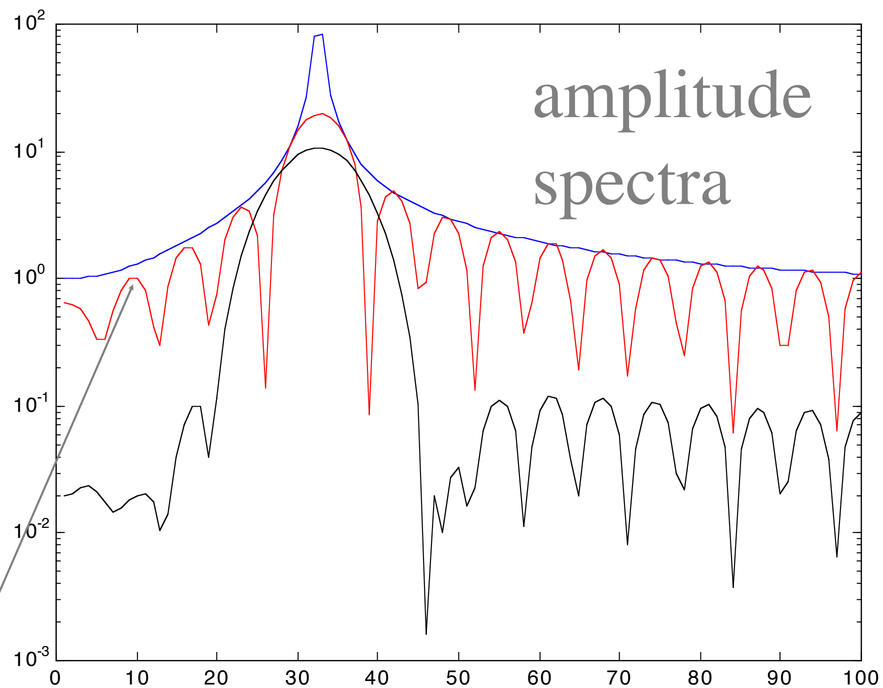


Why is windowing necessary?

- to generally taper the short-term signal at its ends to avoid unnatural discontinuities
- any window effects the spectral estimate computed on it
- the window is selected to trade off the width of its main lobe and attenuation of its side lobes

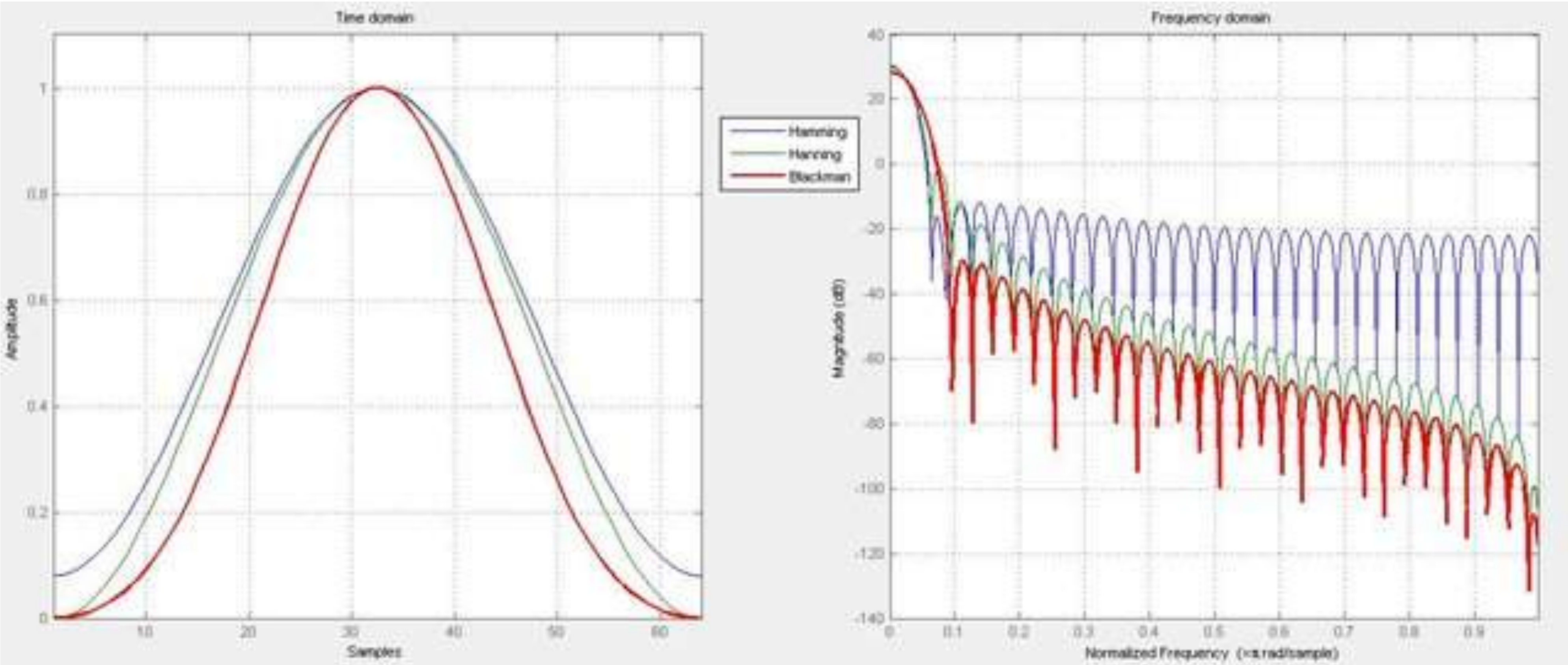
Why is windowing necessary?





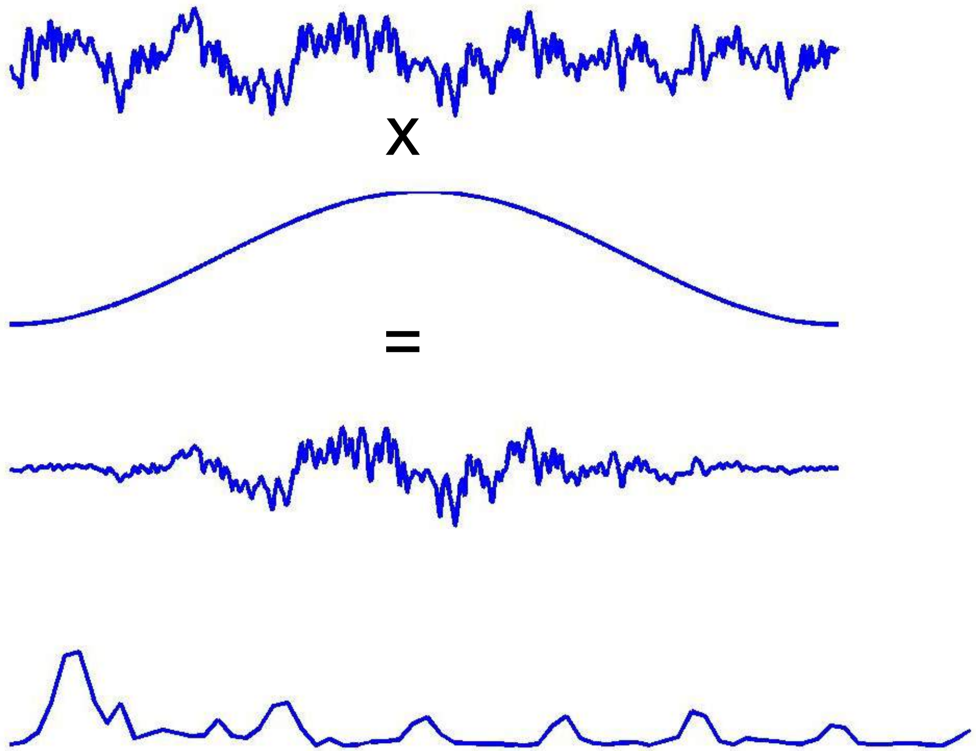
leakage into side lobes

Why is windowing necessary?

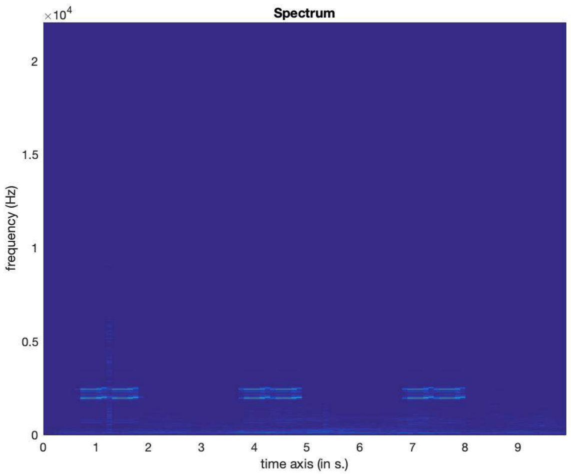


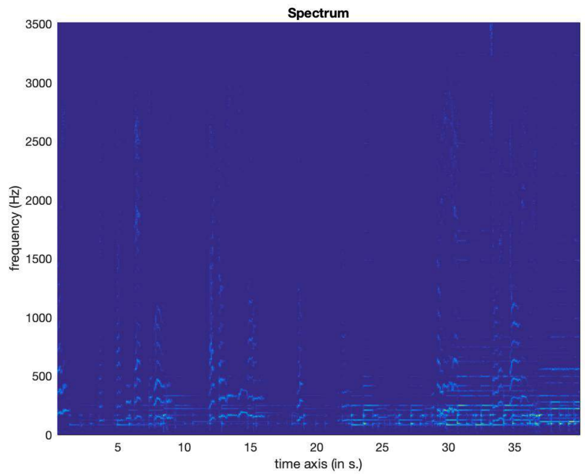
Recap:Computing the STFT

- Divide signal into frames
  - frame length e.g. 128 ... 1024 samples
  - hop size e.g. 16 ... 512
- Window each frame
  - e.g. with Hamming window
- Compute the FFT for each frame

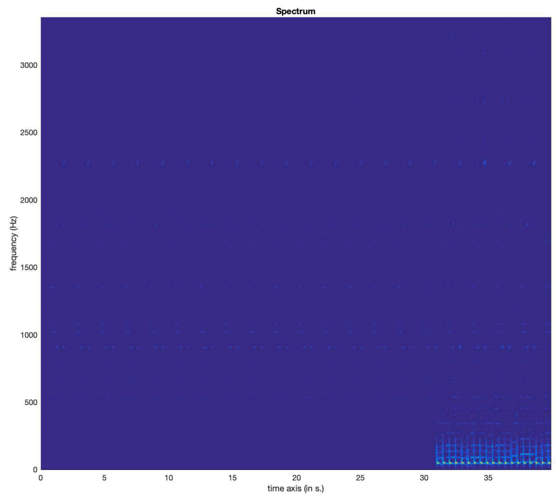


Match the Spectrogram

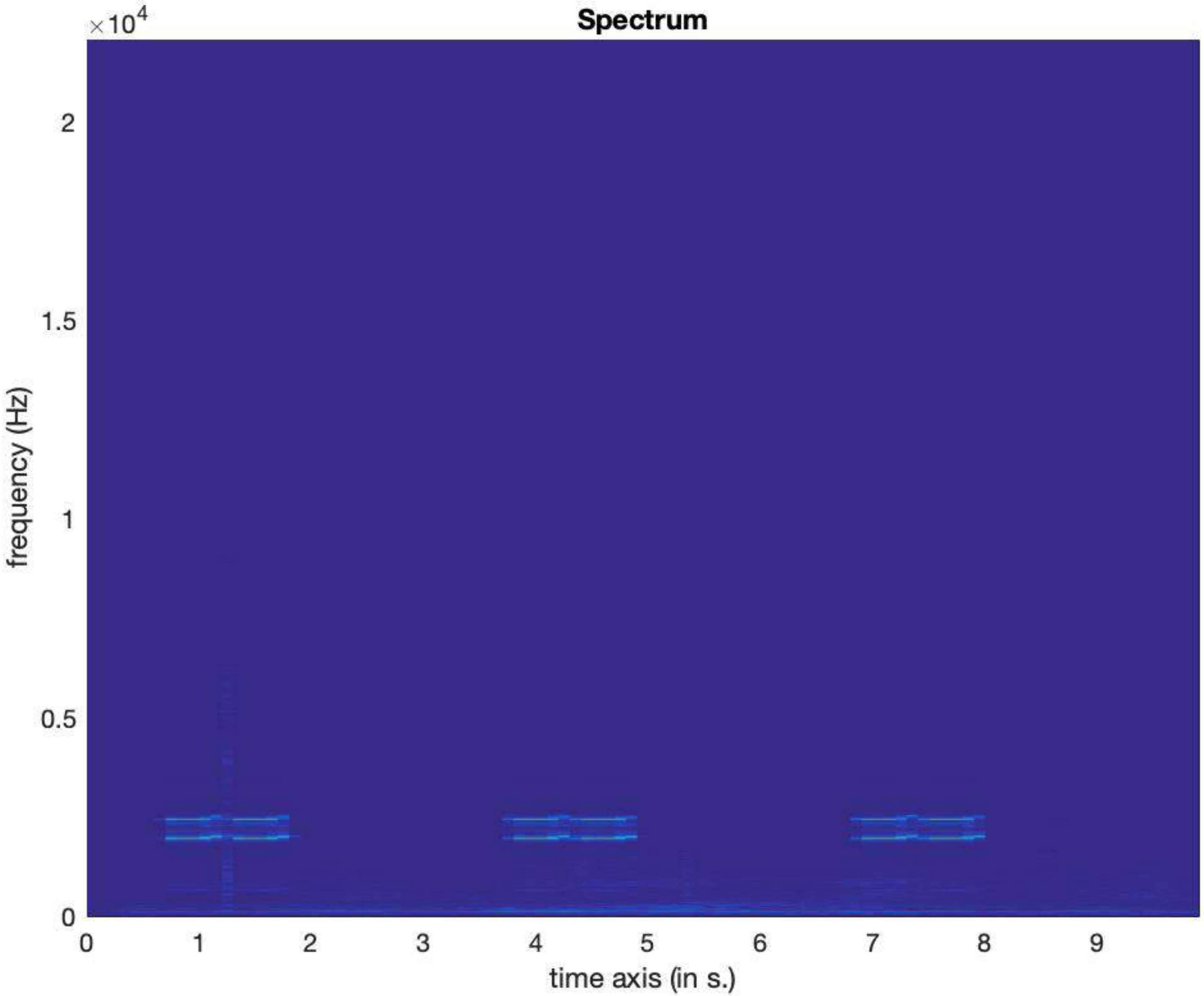




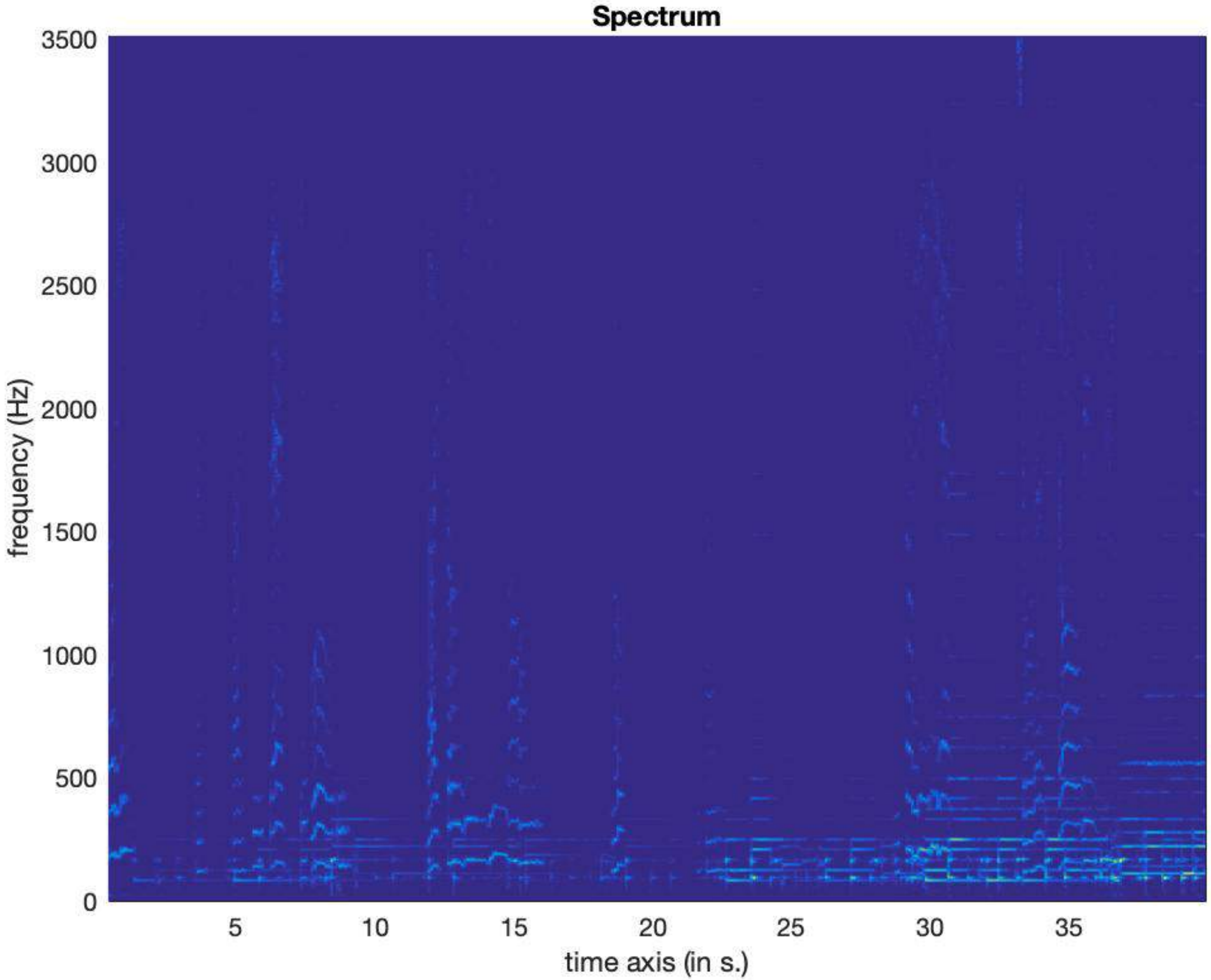
https://1drv.ms/u/s!AtoLT6JDyxo-iVyRF4V6bqWwjcSa?e=tBlXeX



Match the Spectrogram



Match the Spectrogram



Match the Spectrogram