Introduction to Signal Processing

INTELLIGENT SYSTEMS FOR PATTERN RECOGNITION (ISPR)

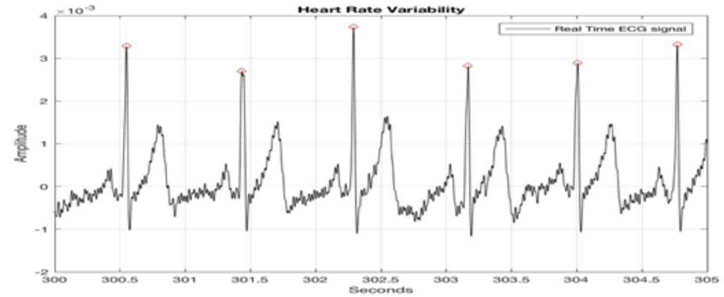
DAVIDE BACCIU – DIPARTIMENTO DI INFORMATICA - UNIVERSITA’ DI PISA

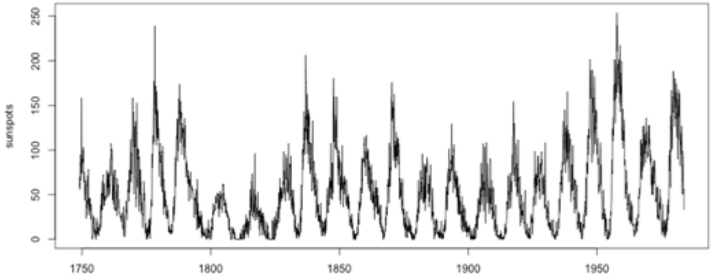
DAVIDE.BACCIU@UNIPI.IT

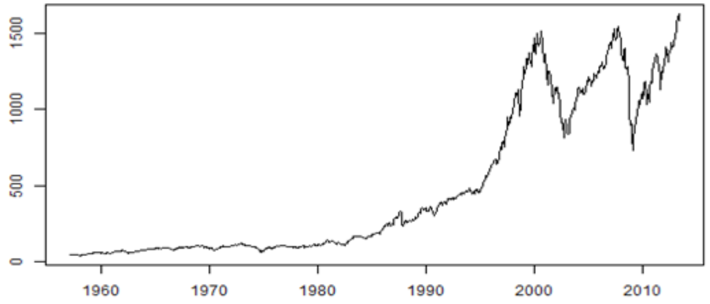
Signals = Time series

A sequence of measurements in time

- Medicine
- Financial
- Meteorology
- Geology
- Biometrics
- Robotics
- IoT
- Biometrics
- ...







Formalization

A time series $\boldsymbol{x}$ is a sequence of measurements in time t

$$\boldsymbol{x} = x_0, x_1, \dots, x_t, \dots, x_N$$

where x_t (or $x(t)$) is the measurement at time t.

- Observations can be observable at irregular time intervals
- Time series analysis assumes weakly stationary (or second-order stationary) data
  - $\mathbb{E}[x_t] = \mu$ for all t
  - $Cov(x_{t+\tau}, x_t) = \gamma_t$ for all t (γ does only depend on lag τ)

IN SUPREMÆ DIGNITATIS 1343 UNIVERSITÀ DI PISA

Goals

- Description - Summary statistics, graphs
- Analysis - Identify and describe dependencies in data
- Prediction - Forecast the next values given information up to time *t*
- Control - Adjust the parameters of the generative process to make the time series fit a target

The goal of this lecture is providing knowledge on some basic techniques that can be useful as

- Baseline
- Preprocessing
- Building blocks

UNIVERSITÀ DI PISA

Key Methods

- Time domain analysis - Assesses how a signal changes over time
  - Correlation and Convolution
  - Autoregressive models

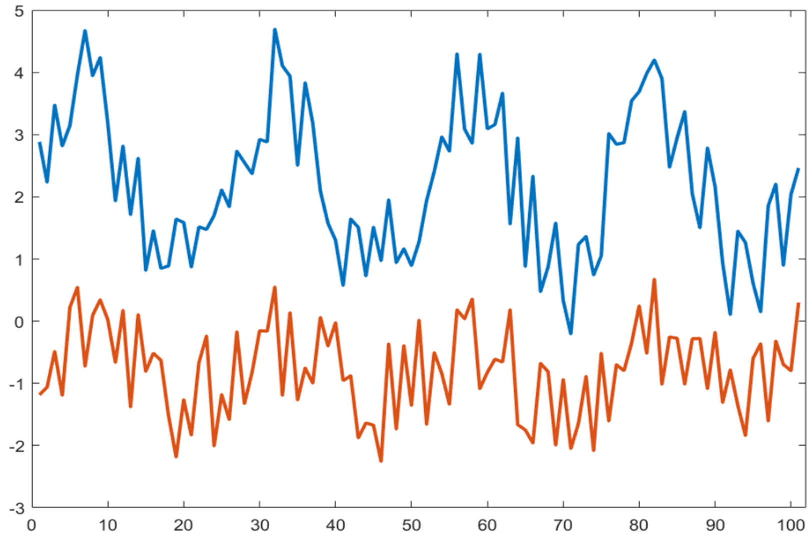
- Spectral domain analysis - Assesses the distribution of the signal over a range of frequencies
  - Fourier Analysis
  - Wavelets (in 2 lectures)

Mean and Autocovariance

Some interesting estimators for time series statistics are

Sample mean

$$\hat{\mu} = \frac{1}{N}\sum_{t=1}^{N} x_t$$

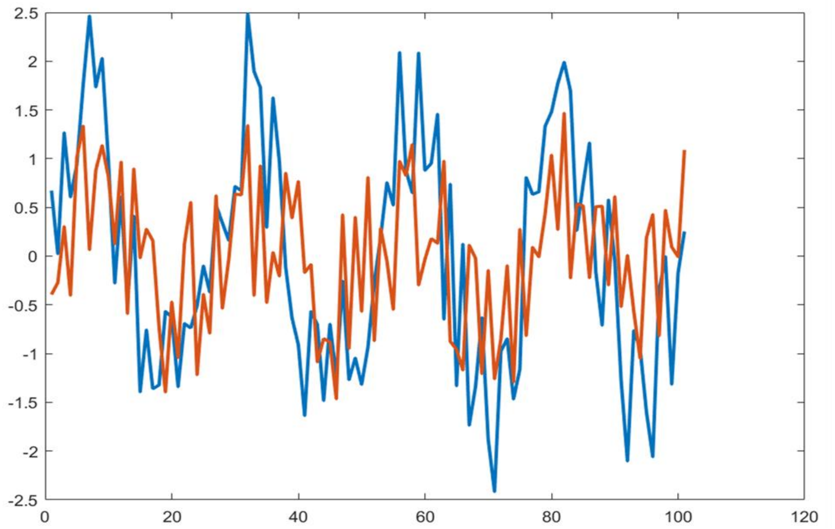


Mean and Autocovariance

Some interesting estimators for time series statistics are

Sample mean

$$\hat{\mu} = \frac{1}{N}\sum_{t=1}^{N} x_t$$



(Sample) Autocovariance for lag $-N \leq \tau \leq N$

$$\hat{\gamma}_x(\tau) = \frac{1}{N}\sum_{t=1}^{N-|\tau|} (x_{t+|\tau|} - \hat{\mu})(x_t - \hat{\mu})$$

Autocorrelation

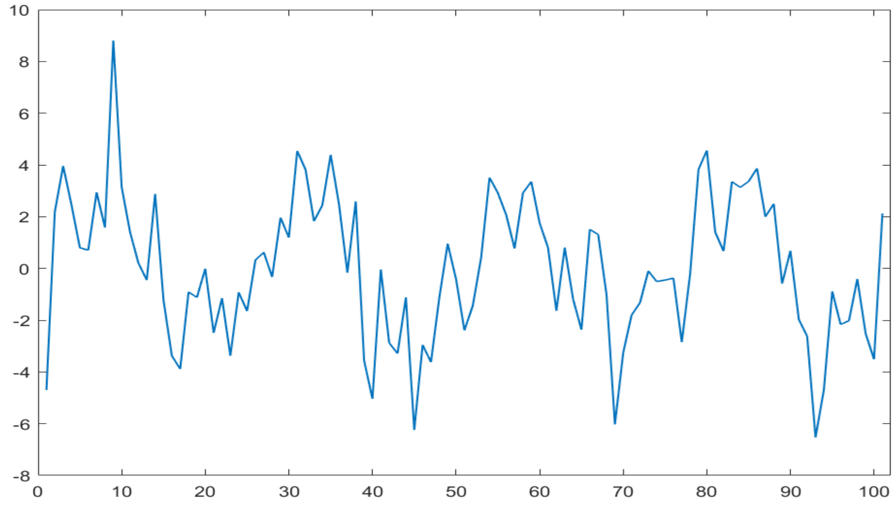
Autocovariance serves to compute autocorrelation, i.e. the correlation of a signal with itself

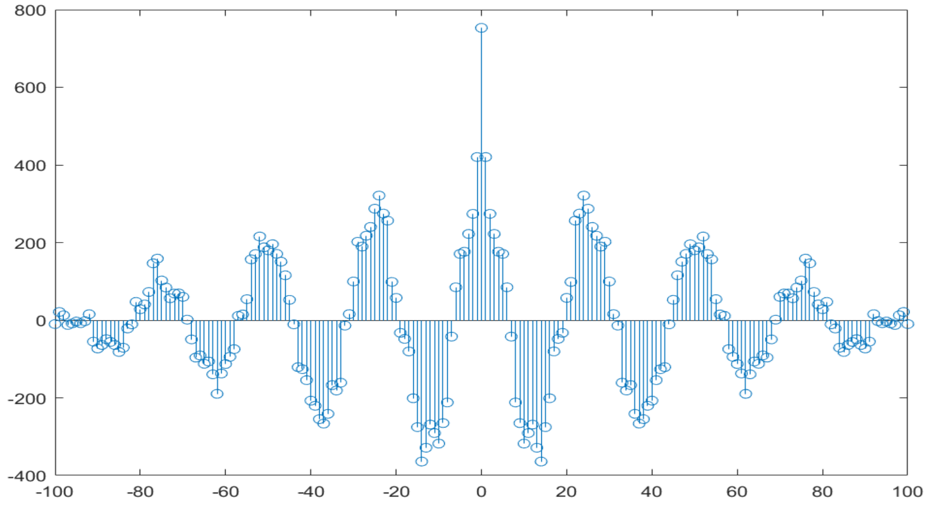
$$\hat{\rho}_{\boldsymbol{x}}(\tau) = \frac{\hat{\gamma}_{\boldsymbol{x}}(\tau)}{\hat{\gamma}_{\boldsymbol{x}}(0)}$$

Autocorrelation analysis can reveal repeating patterns such as the presence of a periodic signal hidden by noise

Autocorrelation Plot

A revealing view on time series statistics





What do you see in this time series?

Autocorrelogram reveals a sine wave

Cross-Correlation (Discrete)

A measure of similarity of $\boldsymbol{x}^1$ and $\boldsymbol{x}^2$ as a function of a time lag τ

$$\phi_{\boldsymbol{x}^1\boldsymbol{x}^2}(\tau) = \sum_{t=\max\{0,\tau\}}^{\min\{(T^1-1+\tau),(T^2-1)\}} x^1(t-\tau)\cdot x^2(t)$$

- $\tau \in [-(T^1-1), \ldots, 0, \ldots, (T^1-1)]$
- The maximum $\phi_{\boldsymbol{x}^1\boldsymbol{x}^2}(\tau)$ w.r.t. τ identifies the displacement of $\boldsymbol{x}^1$ vs $\boldsymbol{x}^2$

Cross-Correlation (Discrete)

Normalized cross-correlation returns an amplitude independent value

$$\bar{\phi}_{\boldsymbol{x}^1\boldsymbol{x}^2}(\tau) = \frac{\phi_{\boldsymbol{x}^1\boldsymbol{x}^2}}{\sqrt{\sum_{t=0}^{T^1-1}\left(x^1(t)\right)^2 \sum_{t=0}^{T^2-1}\left(x^2(t)\right)^2}} \in [-1,+1]$$

- $\bar{\phi}_{\boldsymbol{x}^1\boldsymbol{x}^2}(\tau) = +1 \Rightarrow$ The two time-series have the exact same shape if aligned at time τ
- $\bar{\phi}_{\boldsymbol{x}^1\boldsymbol{x}^2}(\tau) = -1 \Rightarrow$ The two time-series have the exact same shape but opposite sign if aligned at time τ
- $\bar{\phi}_{\boldsymbol{x}^1\boldsymbol{x}^2}(\tau) = 0 \Rightarrow$ Completely uncorrelated signals

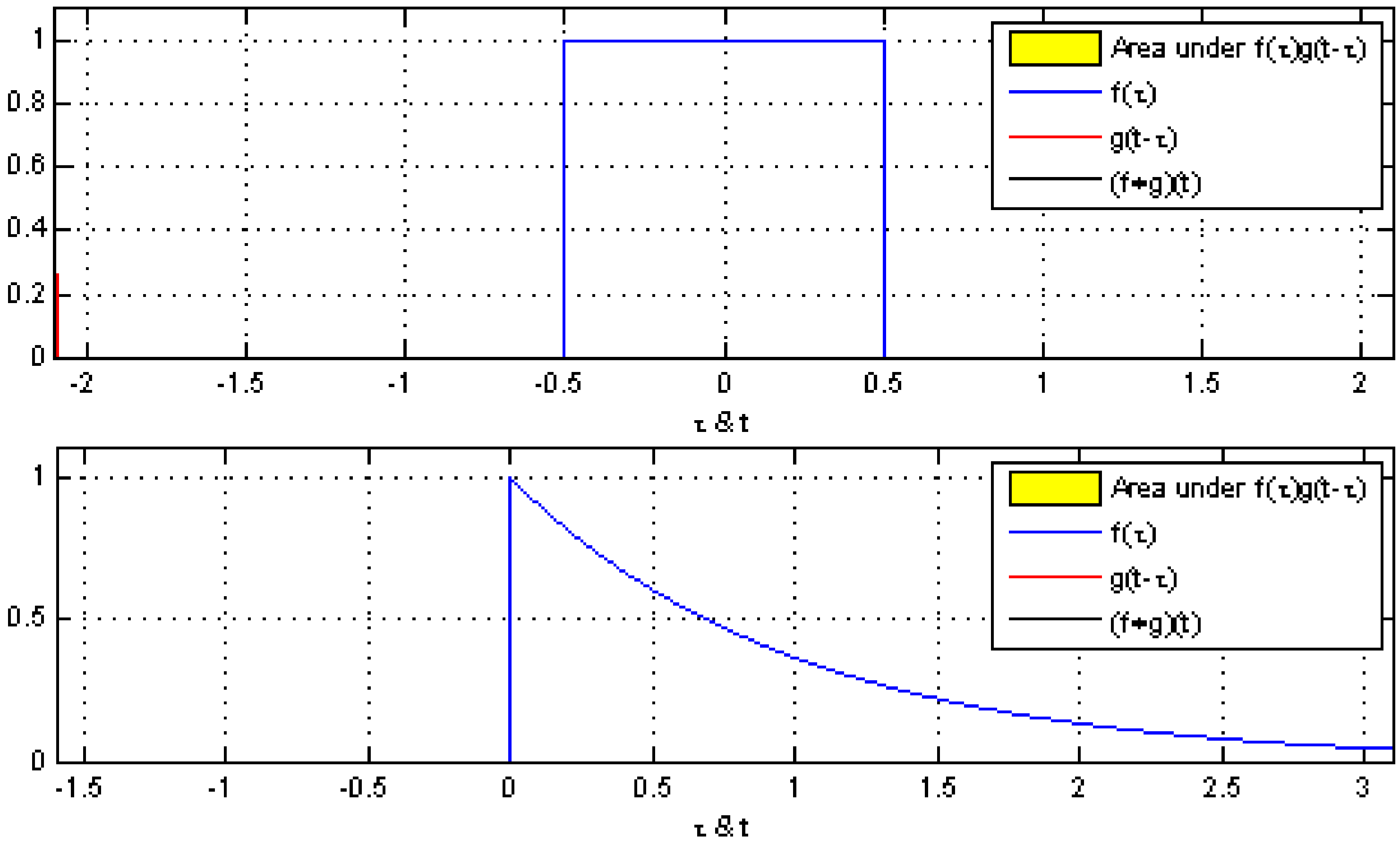
Cross-Correlation - Something already seen...

What is this?

$$(f * g)[\tau] = \sum_{t=-M}^{M} f(\tau - t)g(t)$$

- Discrete convolution on finite support $[-M, +M]$
- Similar to cross-correlation but one of the signals is flipped on y-axis (i.e. $-t$ in place of t)
- Convolution can be seen as a smoothing operator (commutative!)

Convolution - Graphically



The area under f when weighted by a displaced and flipped version of g

Autoregressive Process

A timeseries Autoregressive process (AR) of order *K* is the linear system

$$x_t = \sum_{k=1}^{K} \alpha_k x_{t-k} + \epsilon_t$$

- autoregressive ⇒ x_t regresses on itself
- α_k ⇒ linear coefficients s.t. $|\alpha| < 1$
- ϵ_t ⇒ sequence of i.i.d. values with mean 0 and fixed variance

ARMA

Autoregressive with Moving Average process (ARMA)

$$x_t = \sum_{k=1}^{K} \alpha_k x_{t-k} + \sum_{q=1}^{Q} \beta_q \epsilon_{t-q} + \epsilon_t$$

- $\epsilon_t \Rightarrow$ Random white noise (again)
- The current timeseries value is the result of a regression on its past values plus a term that depends on a combination of stochastically uncorrelated information
- Denotes new information or shocks at time t

Estimating Autoregressive Models

- Need to estimate
  - The values of the linear coefficients α_t (and β_t)
  - The order of the autoregressor K (and Q)
- Estimation of the α is performed with the Levinson-Durbin recursion
  - Native Matlab: `a = levinson(x,K)`
  - Included in several Python modules: statsmodels, Spectrum, …
- The order is often estimated with a Bayesian model selection criterion, e.g. BIC, AIC, etc.

The set of autoregressive parameters $\alpha_1^i, \dots, \alpha_K^i$ fitted to a specific timeseries $\boldsymbol{x}^i$ is used to confront it with other timeseries

UNIVERSITÀ DI PISA

Comparing Timeseries by AR

- Timeseries clustering

$$d(\boldsymbol{x}^1, \boldsymbol{x}^2) = \|\alpha^1 - \alpha^2\|_M^2$$

- Novelty/anomaly detection

$$Test\ Err(x_t, \hat{x}_t) < \xi$$

where $\hat{x}_t$ is the AR predicted value

- Encode time series as a set of α^i vectors and feed them to a flat ML model

Spectral Analysis

Analyzing time series in the frequency domain

Key Idea
Decompose a time series into a linear combination of sinusoids (and cosines) with random and uncorrelated coefficients

- Time domain - Regression on past values of the time series
- Frequency domain - Regression on sinusoids

Use the framework of Fourier Analysis

Fourier Transform

- Discrete Fourier transform (DFT)
- Transforms a time series from the time domain to the frequency domain
- Can be easily inverted (back to the time domain)
- Useful to handle periodicity in the time series
  - Seasonal trends
  - Cyclic processes

Representing Functions

We (should) know that, given an orthonormal system $\{\boldsymbol{e}_1;\ \boldsymbol{e}_2, ...\}$ for E, we can represent any function $f \in E$ by a linear combination of the basis

$$\sum_{k=1}^{\infty} \langle f, \boldsymbol{e}_k \rangle \boldsymbol{e}_k$$

Given the orthonormal system

$$\{\frac{1}{\sqrt{2}}, \sin(x), \cos(x), \sin(2x), \cos(2x), \dots\}$$

the linear combination above becomes the Fourier Series

$$\frac{a_0}{2} + \sum_{k=1}^{\infty} [a_k \cos(kx) + b_k \sin(kx)]$$

with a_k, b_k being coefficients resulting from integrating $f(x)$ with the sin and cos functions

IN SUPREMÆ DIGNITATIS 1343 UNIVERSITÀ DI PISA

Representing Functions in Complex Space

Using $\cos(kx) - i\, sin(kx) = e^{-ikx}$ with $i = \sqrt{-1}$ we can rewrite the Fourier series as

$$\sum_{k=-\infty}^{\infty} X_k e^{-ikx}$$

on the orthonormal system

$$\{1, e^{ix}, e^{-ix}, e^{2ix}, e^{-2ix}, \dots\}$$

and X_k integrates $f(x)$ with e^{-ikx} .

Representing Discrete Time series

1. Consider a discrete time series $\boldsymbol{x} = x_0, x_1, \dots, x_{N-1}$ of length N and $x_n \in \mathbb{R}$

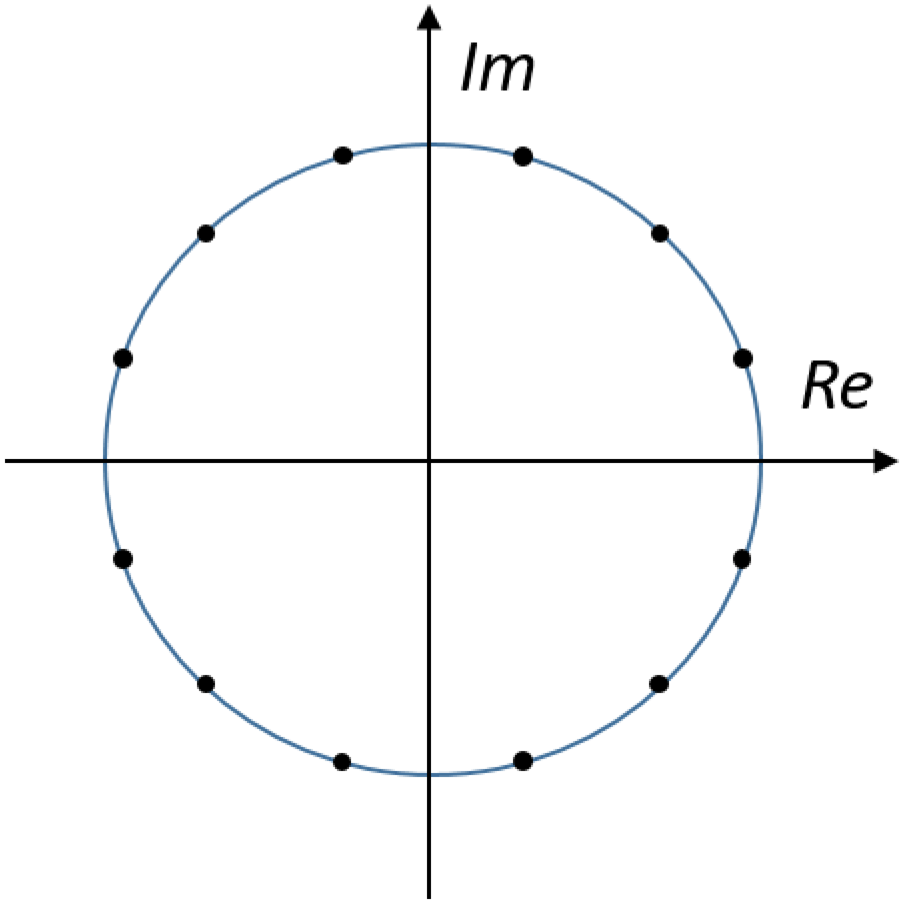
2. Using the exponential formulation, the orthonormal system is made of $\{\boldsymbol{e}_0, \boldsymbol{e}_1, \dots, \boldsymbol{e}_{N-1}\}$ vectors $\boldsymbol{e}_k \in \mathbb{C}^N$

3. The n-th component of the k-th vector is

$$[\boldsymbol{e}_k]_n = e^{\frac{-2\,\pi\, ink}{N}}$$

UNIVERSITÀ DI PISA

Graphically



A basis $\boldsymbol{e}_k$ at frequency k has N elements sampled from the roots of the unitary circle in imaginary-real space

Discreet Fourier Transform

Given a time series $\boldsymbol{x} = x_0, x_1, \dots, x_{N-1}$ its Discrete Fourier Transform (DFT) is the sequence (in frequency domain)

$$X_k = \sum_{n=1}^{N-1} x_n e^{\frac{-2\,\pi\, ink}{N}}$$

The DFT has an inverse transform

$$x_n = \frac{1}{N} \sum_{k=1}^{N-1} X_k e^{\frac{2\,\pi\, ink}{N}}$$

to go back to the time domain.

Basic Spectral Quantities in DFT

We would like to measure relevance/strength/contribution of a target frequency bin k

- Amplitude

$$A_k = |X_k| = \sqrt{Re^2(X_k) + Im^2(X_k)}$$

(you can also compute phase)

- Power

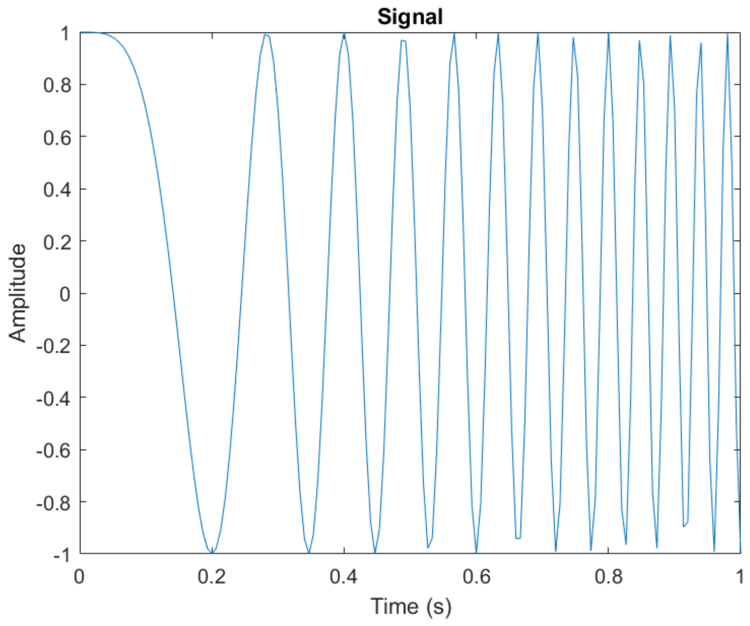
$$P_k = \frac{|X_k|^2}{N}$$

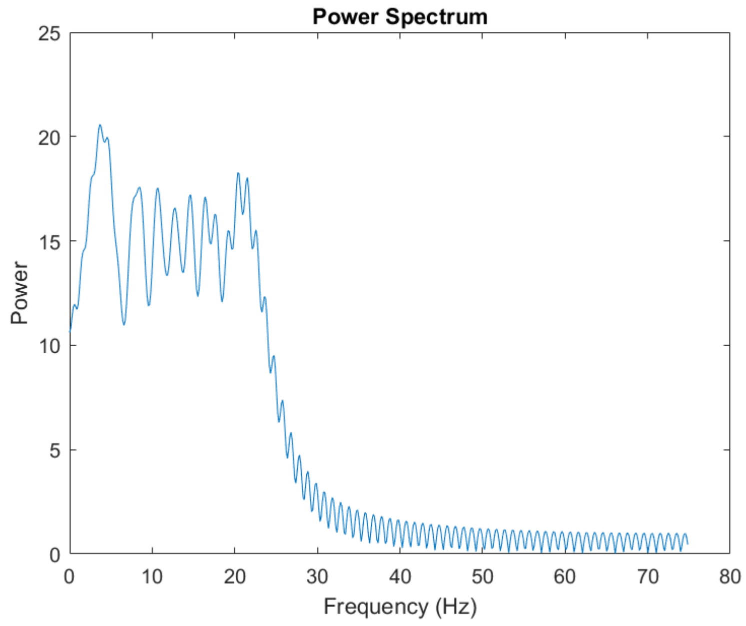
(under some conditions this is a more-or-less reasonable estimate of the power spectral density)

UNIVERSITÀ DI PISA

DFT Power spectrum in use

```
X = fft(x);                 % x - sample signal
n = length(x);
f = (0:n-1)*(fs/n);         % fs - sample frequency (Hz)  / f – frequency range
power = abs(X).^2/n;
```

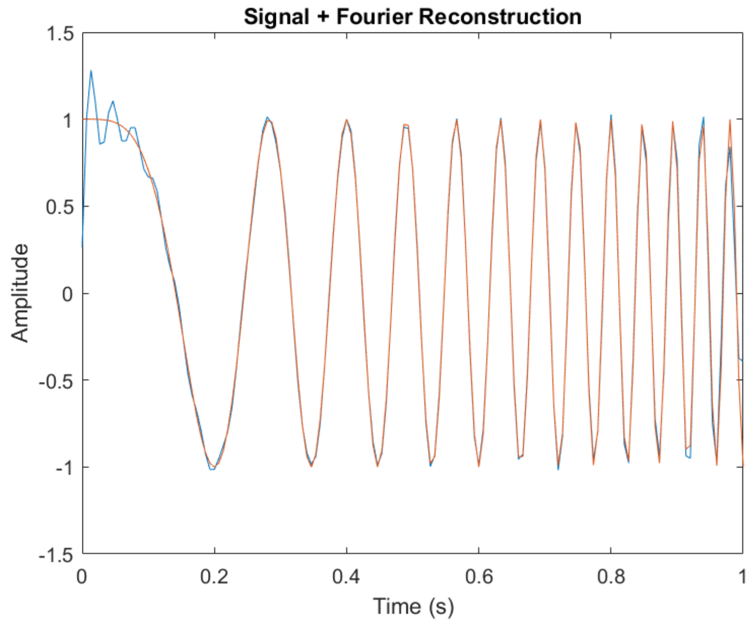


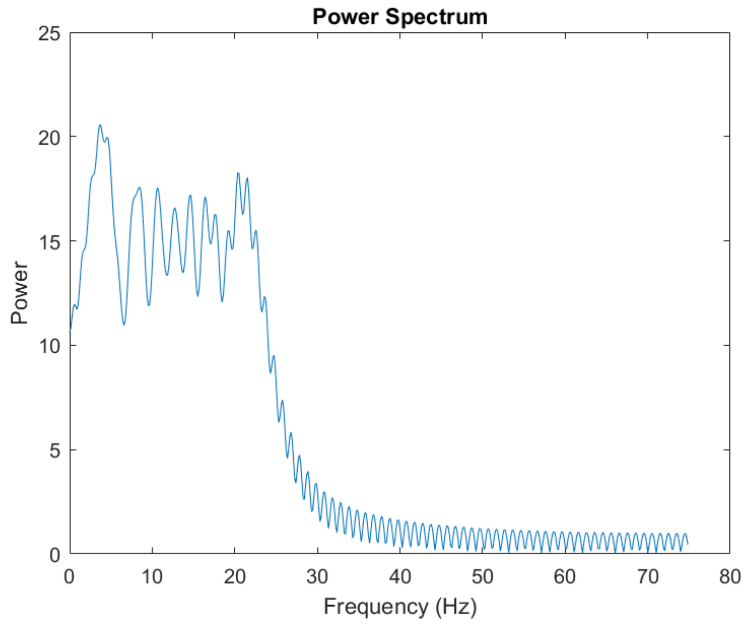


UNIVERSITÀ DI PISA

DFT Power spectrum in use

Back to the time domain (keeping only relevant frequencies)

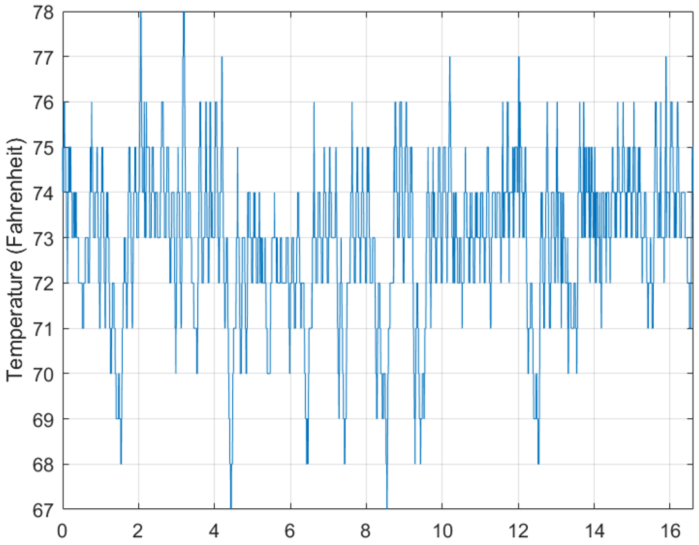


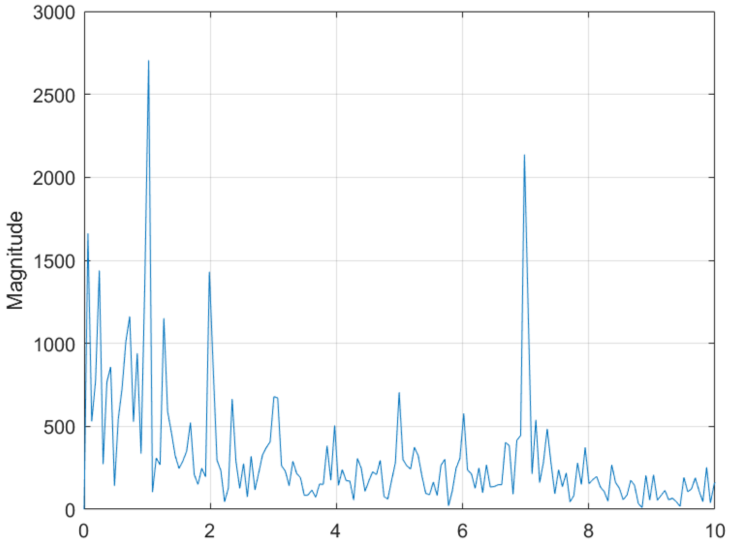


UNIVERSITÀ DI PISA

DFT in Action

- Use the DFT elements $X_1, ..., X_K$ as representation of the signal to train predictor/classifier
- Representation in spectral domain can reveal patterns that are not clear in time domain





Some less basic spectral descriptors

- Spectral Centroid
- Spectral Spread
- Spectral Skewness
- Spectral Kurtosis
- Spectral Entropy
- Spectral flatness
- Spectral crest
- Spectral flux
- Spectral slope
- ….

Spectral Centroid

Spectral-weighted average frequency (between frequency bands b_1 and b_2)

$$\mu = \frac{\sum_{k=b_1}^{b_2} f_k s_k}{\sum_{k=b_1}^{b_2} s_k}$$

- f_k is the k-th frequency (in Hz)
- s_k is the corresponding spectral weight (e.g. amplitude A_k or power spectrum P_k)

Higher-order moments

- Spread - Standard deviation around the spectral centroid μ

$$\sigma = \sqrt{\frac{\sum_{k=b_1}^{b_2} (f_k - \mu)^2 s_k}{\sum_{k=b_1}^{b_2} s_k}}$$

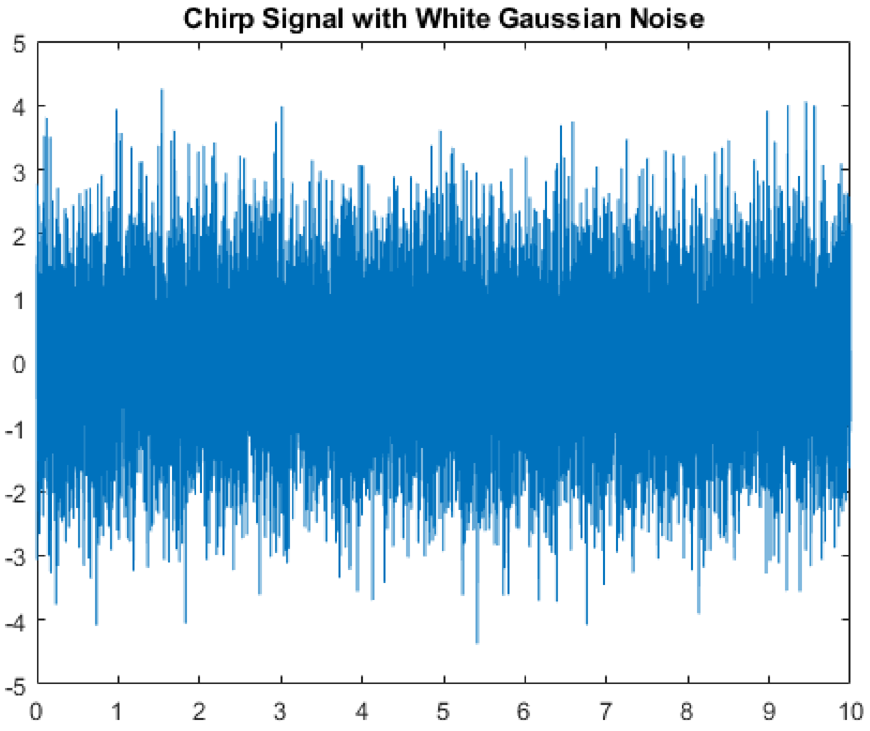
- Kurtosis – (4th order moment) Measures flatness or non-Gaussianity of the spectrum around the centroid μ

$$K = \frac{\sum_{k=b_1}^{b_2} (f_k - \mu)^4 s_k}{\sigma^4 \sum_{k=b_1}^{b_2} s_k}$$

k = pkurtosis(x)

IN SUPREMÆ DIGNITATIS 1343
UNIVERSITÀ DI PISA

Kurtosis Example



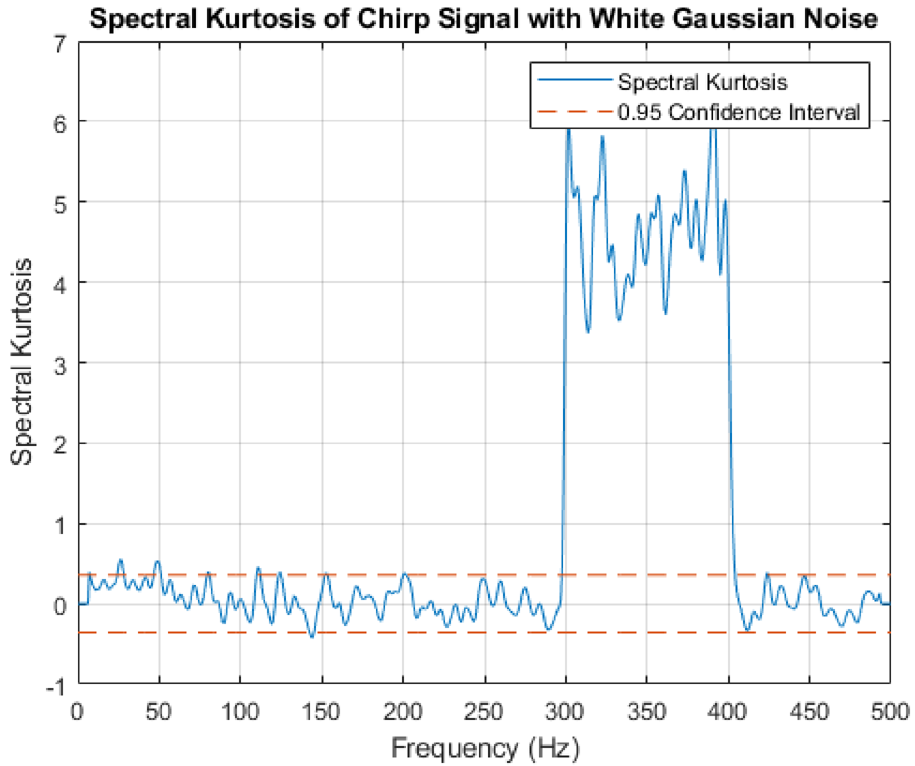
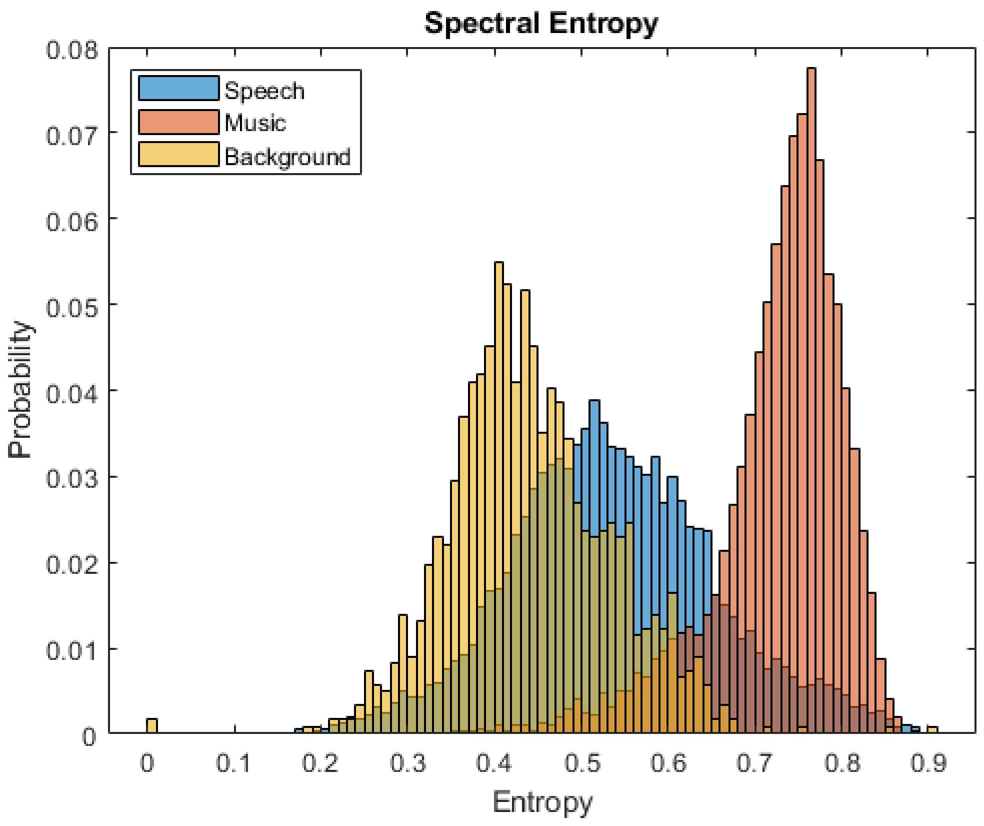


Image from pkurtosis @ Matlab

Spectral Entropy



- Represents peak-ness of the spectrum

$$H = \frac{-\sum_{k=b_1}^{b_2} s_k \log s_k}{\log(b_2 - b_1)}$$

- e.g. discriminate between music and speech

H = pentropy(x)

Università di Pisa

Take Home Messages

- Old-school pattern recognition on timeseries is about learning coefficients that describe properties of the time series
  - Autoregressive coefficients (time domain)
  - Fourier coefficient (frequency domain)
- Often linear methods
  - Autocorrelation reveals similitude of a signal with delayed versions of itself
  - Cross-correlation provides hints on time series similarity and how to align them
- Fourier analysis allows to identify recurring patterns and key frequencies in the signal (and represent this information through spectral descriptors)

Next Lecture

Introduction to image processing (I)

- Representing images and visual content
- Intensity gradients and histograms
- Filters
- Spatial descriptors: SIFT
- Spectral analysis in 2D