Statistical Signal Processing & Inference

Course Introduction

Prof Danilo Mandic

room 813, ext: 46271

Department of Electrical and Electronic Engineering

Imperial College London, UK

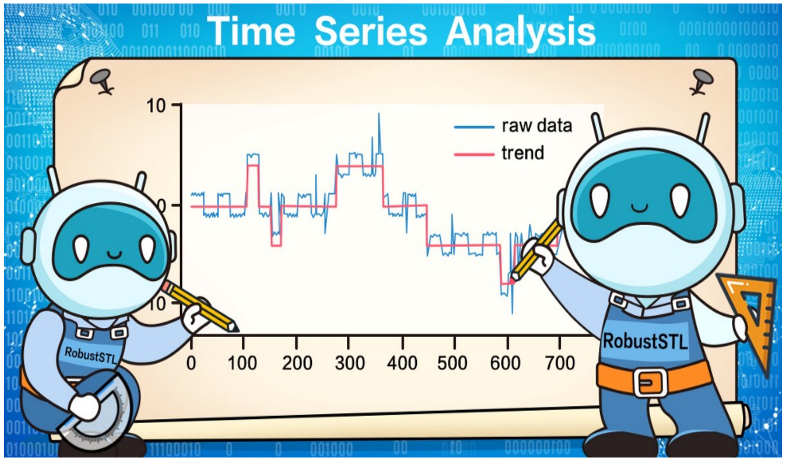
d.mandic@imperial.ac.uk, URL: www.commsp.ee.ic.ac.uk/∼mandic

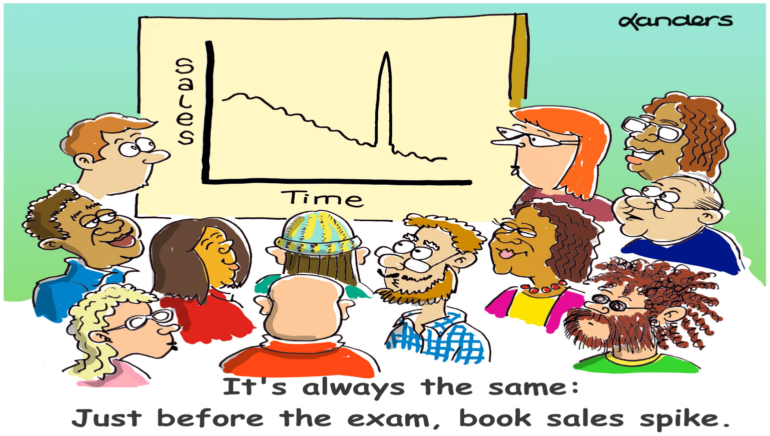
The need for Statistical Signal Processing

Q: Have you ever considered what the following tasks have in common:

- Forecasting of financial data
- Supply-demand modelling (e.g. electricity or air-ticket pricing)
- Modelling of COVID-19 spread
- Person recognition from a set of (noisy) images
- Word generation by Large Language Models such as ChatGPT

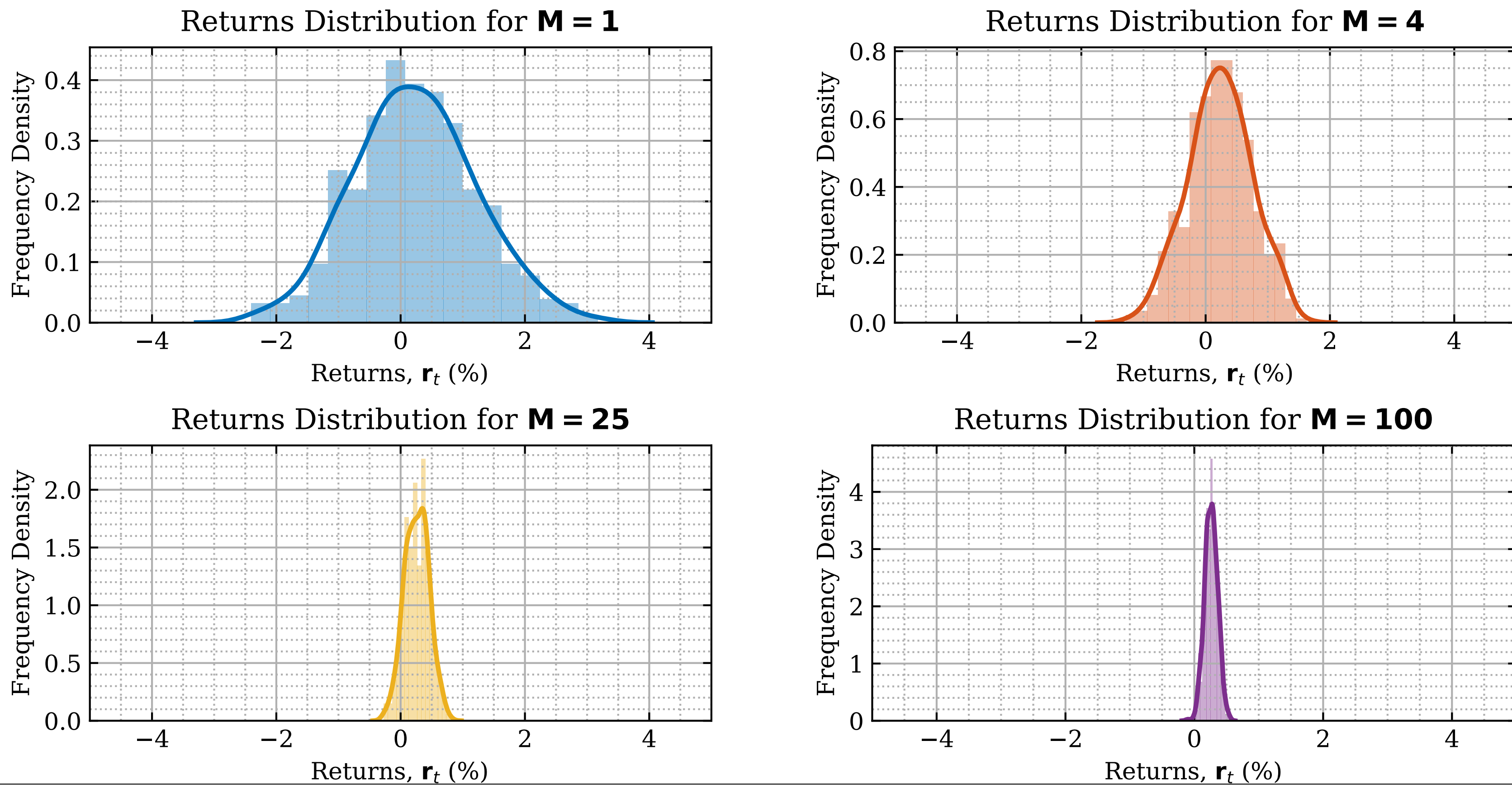
A: These are signals/images of which the signal generating mechanisms are largely unknown or untractable. We need to make sense from such data based on historical observations only $\looparrowright$ subject of **Statistical Inference.**





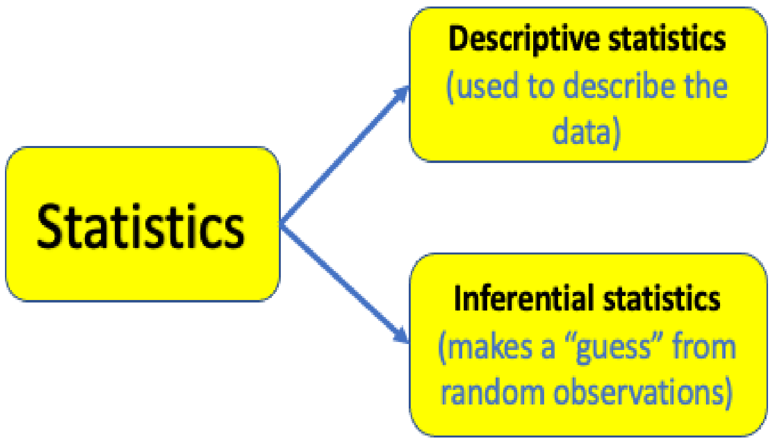
The need for statistical inference: Population modelling

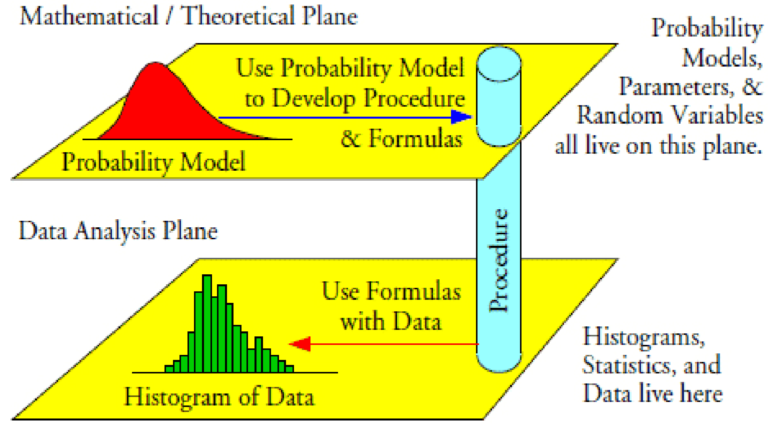
Example from financial modelling: Risk for a single asset and a for a portfolio of uncorrelated assets. Risk is represented by the standard deviation (or the width) of the distribution curves $\looparrowright$ a large portfolio ($M = 100$) can be significantly less risky than a single asset ($M = 1$).



Statistical Inference

From Latin *inferre*, which means "bring into, deduce, conclude"





Inferential statistics: **Statistical Estimation** and **Hypothesis Testing**

☞ In Machine Learning, the term "inference" typically indicates "prediction"

Applications:

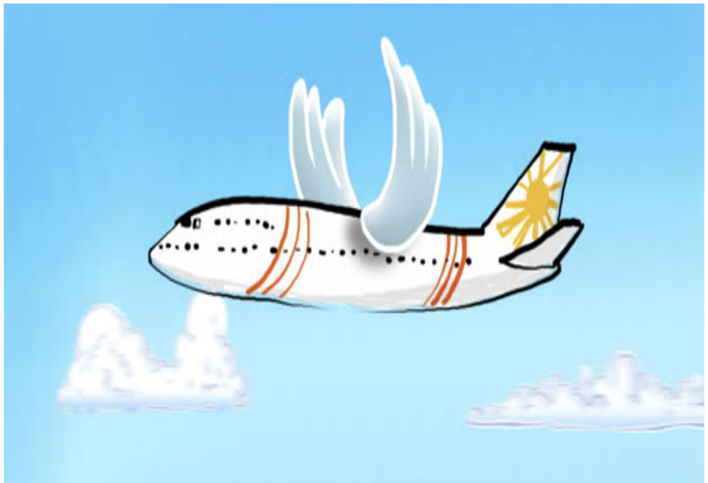
- Adaptive learning algorithms (noise-cancelling headphones, forecasting)
- Neural Networks (e.g. classification, prediction, denoising)
- Communications, power systems, radar, sonar, biomedicine, ...
- Financial modelling, risk estimation, confidence intervals
- Artificial Intelligence (e.g. self-driving cars)

Inferential stats also tells us "what is possible to achieve" (sanity check)

AI and Statistical Signal Processing and Inference

Humans provide a performance 'benchmark' but mimicking human reasoning by AI is suboptimal. Instead, **we should strive to surpass human limitations and not to mimic humans!**

- The 10^11 neurons and 10^15 synapses human brain expend ca. 20 W of power. A digital simulation of an ANN of same size consumes a whopping 7.9 MW.



Engineering solutions do not necessarily mimic the nature

By approaching a problem with an engineering mindset, AI can be considered as a new, human-centric engineering discipline (M. Jordan, "AI – The revolution hasn't happened yet", 2019.)

Claim: Big Data + Deep Learning → General Intelligence

But humans learn very efficiently with little data, not Big Data

Caution: We can no longer train a modern DNN on a personal computer, it would take up to 405 years!
Global share of electricity consumption for digital devices: from 3-4% today to 20% in 2050. We need a convivial technology that is resilient – **a real opportunity for responsible AI, domain knowledge and interpretability.**

Foundations of resilience: Probability vs. Statistics

For discrete RVs, $E\{X\} = \sum_{i=1}^{I} x_i P_X(x_i)$, where P_X is the probability function

Probability: A data modelling view, describes how data **will likely behave**

for example: $$average = E\{X\} = \int_{-\infty}^{\infty} x\, p_X(x)\, dx$$ no data here

Notice that there is no explicit mention of data here $\looparrowright$ x is a dummy variable and p_X is the pdf of a random variable X.

Statistics: A data analysis view, determines how data **did behave**

for example: $$average = \frac{1}{N}\sum_{n=0}^{N-1} x[n]$$ no pdf here

Example: Consider N coarse-quantised data points, $x[0], \ldots, x[N-1]$. The signal has $M \ll N$ possible amplitude values, $V_1, \ldots, V_M$, with the corresponding relative frequencies, $N_1, \ldots, N_M$. Calculate the mean, $\bar{x}$.

Solution:

$$\bar{x} = \frac{1}{N}\sum_{n=0}^{N-1} x[n] \quad = \quad \frac{1}{N}\sum_{m=1}^{M} V_m N_m \quad = \quad \sum_{m=1}^{M} V_m \underbrace{\frac{N_m}{N}}_{\approx\ P(x=V_m)}$$

Aims: To introduce the fundamentals of **statistical estimation theory**, to facilitate the design of signal processing and machine learning algorithms

- The emphasis will be upon:
  - ⊛ random signals, their properties, and statistical descriptors
  - ⊛ linear stochastic models, to generate/describe random signals
  - ⊛ parametric (model based) and nonparametric (data driven) modelling
  - ⊛ optimal estimators for random signals, rigorous performance bounds
  - ⊛ the class of least squares methods, block and sequential LS
  - ⊛ adaptive estimation $\rightsquigarrow$ suitable for nonstationary data

- You will gain **practical experience** through numerous examples on real world signals:
  - ⊛ multimedia (your own speech recorded via PC)
  - ⊛ your own physiological data, some financial data (from *yahoo finance*)

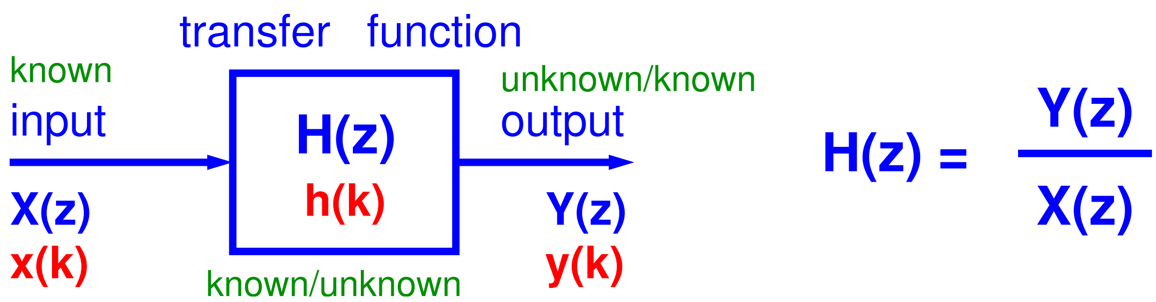
Overall: To gain the know-how and necessary expertise in **statistical inference** from random and non–stationary real world data

☞ **This underpins in-depth understanding and interpretability statistical signal processing and machine learning tools (performance bounds).**

The difference in this course ↬ it gives a big picture of statistical modelling, with rigorous performance bounds

So far, you are familiar with problems characterised by:

- **A well defined transfer function** in the form



- **Deterministic** signals (assuming a mathematically tractable model)
- Rigorous analysis through the notions of **impulse response**, **step response**, **frequency response**, based on $y(n) = \sum_m h(m)x(n-m)$
- **Operation in noise-free & statistically stationary environments**

In this course we will consider more realistic situations where:

⊛ **Signals are random**, and we only known their statistical properties

⊛ **Models/descriptors are derived from data**, and operate even for nonstationary and streaming data sources, and in the presence of noise

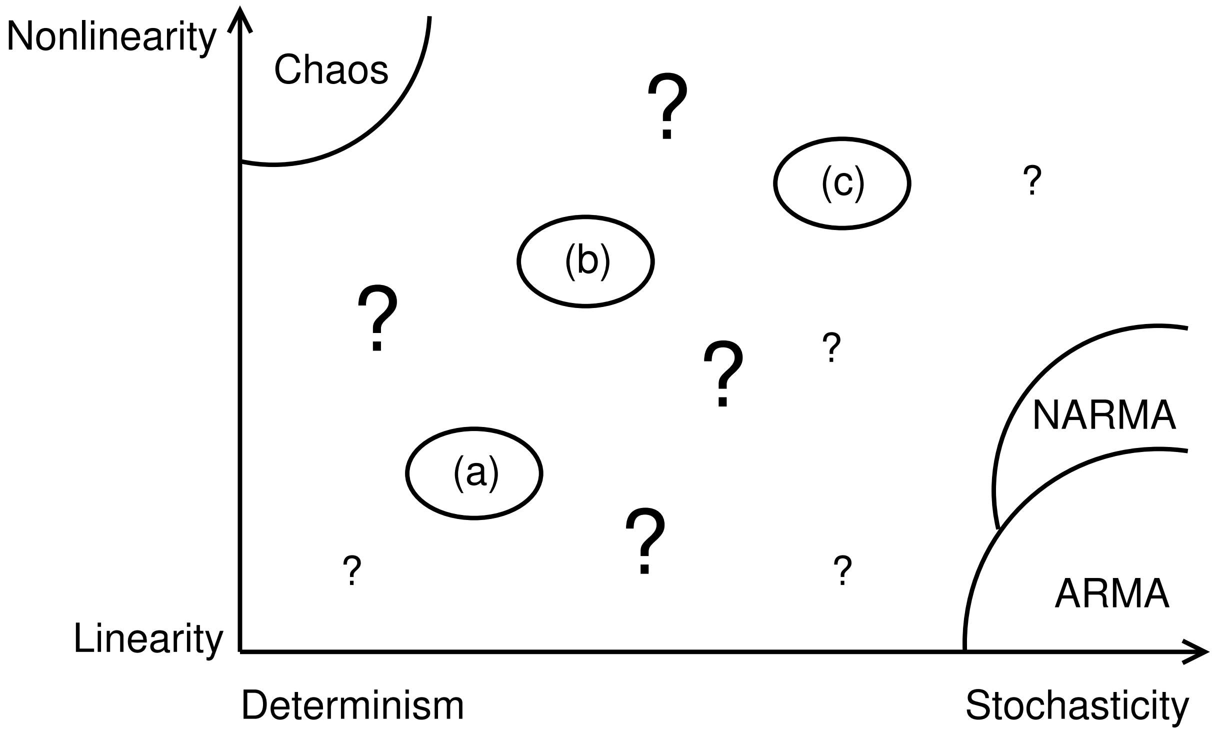
In a nutshell ⇝ basis for adaptive detection, estimation, prediction

You will learn how to make sense from real-world data

where would you place a DC level in WGN, $x[n] = A + w[n], \quad w \sim \mathcal{N}(0, \sigma_w^2)$

(a) Noisy oscillations, (b) Nonlinearity and noisy oscillations, (c) Random nonlinear process

(? left) Route to chaos, (? top) stochastic chaos, (? middle) mixture of sources

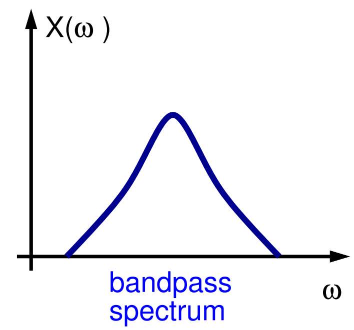


In terms of time series, we will cover linear and nonlinear stochastic models

How about observing the signal through a nonlinear sensor?

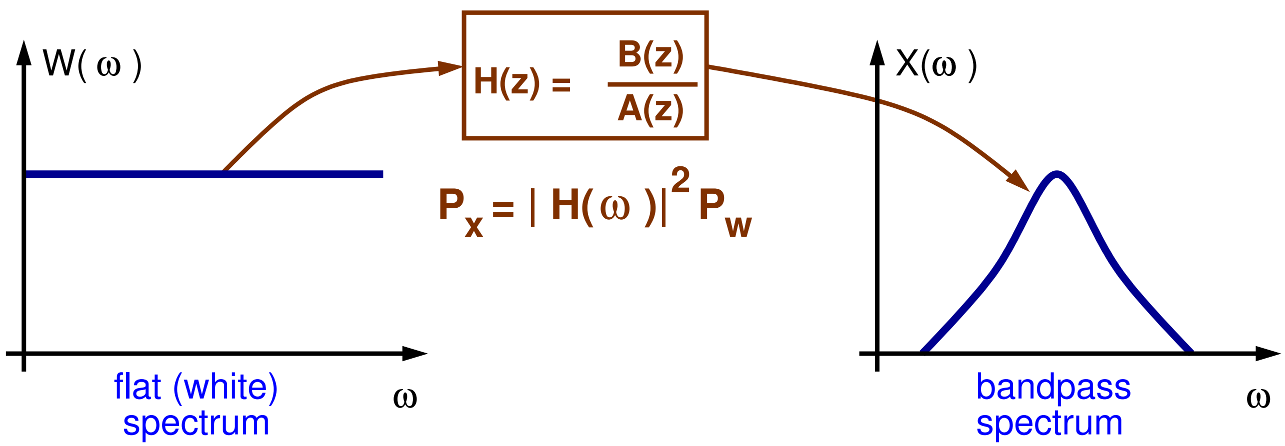
Can we model a complicated and random real world signal with only a few parameters?

Suppose the measured real world signal has a bandpass power spectrum, see figure $\longrightarrow$



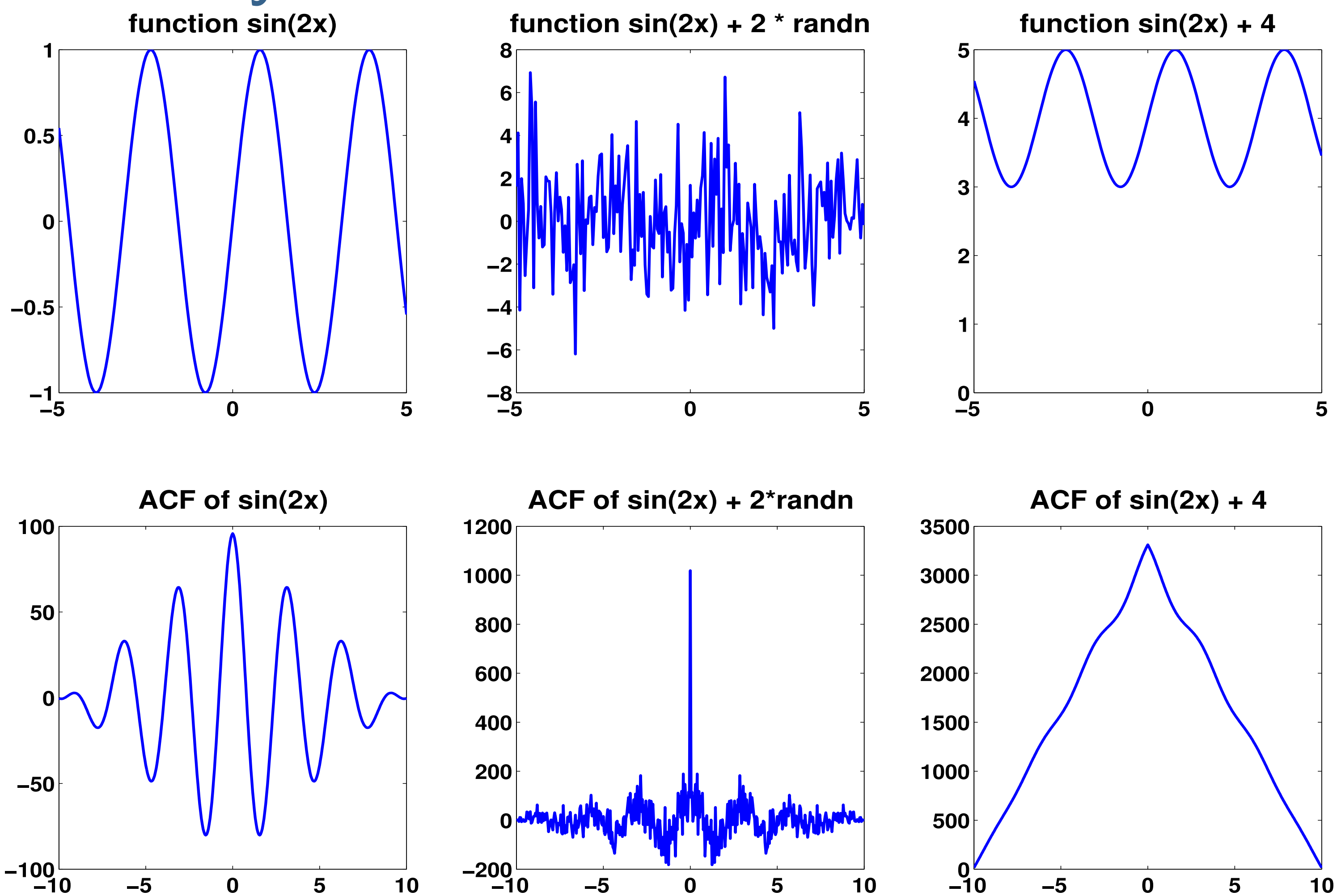
We wish to uniquely describe the whole signal with only very few parameters

1. Can we model first and second statistics of real world signal by shaping the white noise spectrum using some transfer function?
2. Does this produce the same second order properties (mean, variance, ACF, spectrum) for any white noise input?



Can we use this linear stochastic model for prediction?

Example 1: The autocorrelation function (ACF) $\looparrowright$ the basis for many statistical estimators



The figure above $\rightarrow$ *Top panel:* Original signals. *Bottom:* Their ACFs

$\rightsquigarrow$ useful information becomes obscured in noise or DC offset

Example 2: What can we learn from second order stats?

X-corr = matched filter $\looparrowright$ explains and interprets the operation of CNNs

Detection of Tones in Noise:

Consider tone $x = A\cos(\omega n + \theta)$ in noise

$$y[n] = A\cos(\omega n + \theta) + w[n]$$

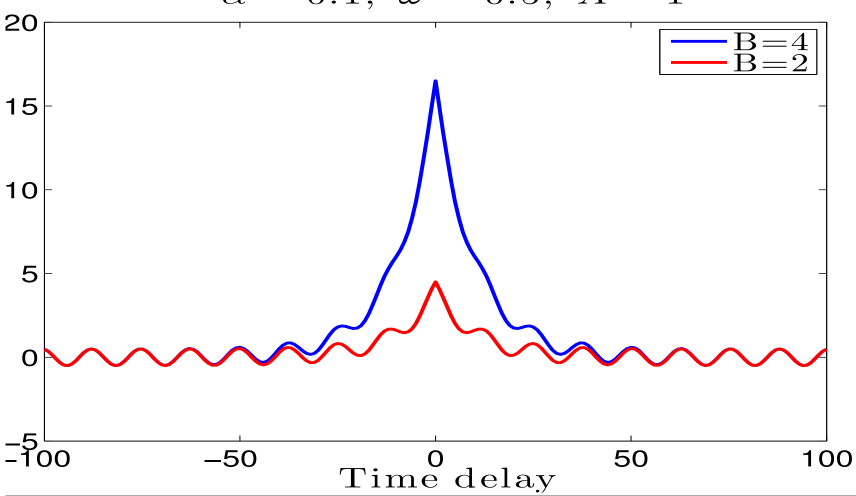
$$\textbf{ACF}: \ R(m) = E\big[y[n]y[n+m]\big] =$$

$$= R_x(m) + R_w(m) + R_{xw}(m) + R_{wx}(m)$$

For $R_w = B^2 \exp(-\alpha|m|)$ & $x \perp w$, then

$$R_y(m) = \frac{1}{2}A^2\cos(\omega m) + B^2\exp(-\alpha|m|)$$

- for large m, the ACF $\propto$ the signal
- $\exists$ extract tiny signal from large noise



Principle of Radar (matched filter):

The received signal (see previous slide)

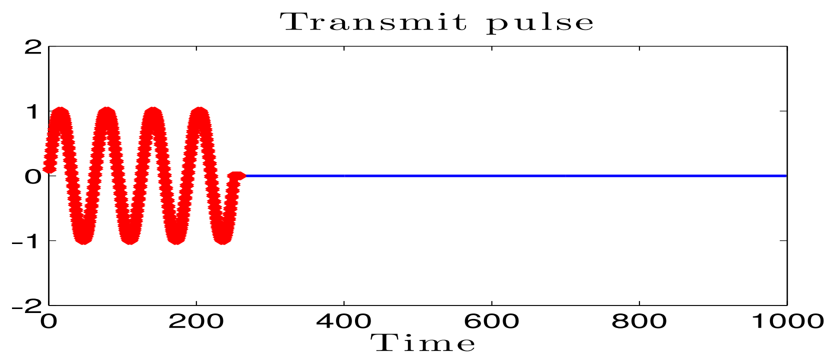
$$y[n] = ax[n - T_0] + w[n], \quad \text{so that}$$

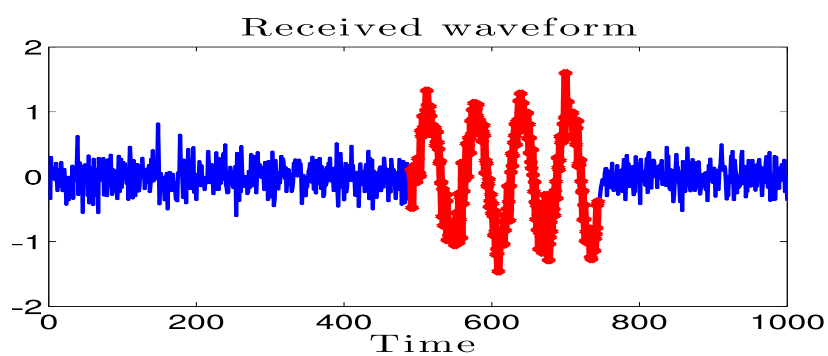
$$R_{xy}(\tau) = E\{x(n)y(n+\tau)\}$$

$$= aR_x(\tau - T_0) + R_{xw}(\tau)$$

Since

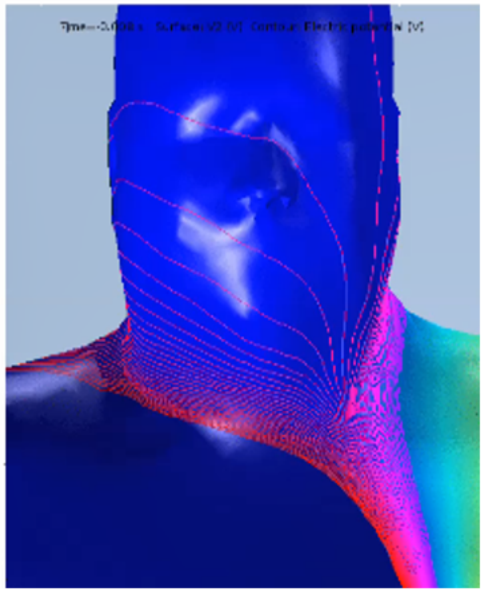
$$\boldsymbol{x \perp w \rightsquigarrow R_{xy}(\tau) = aR_x(\tau - T_0)}$$

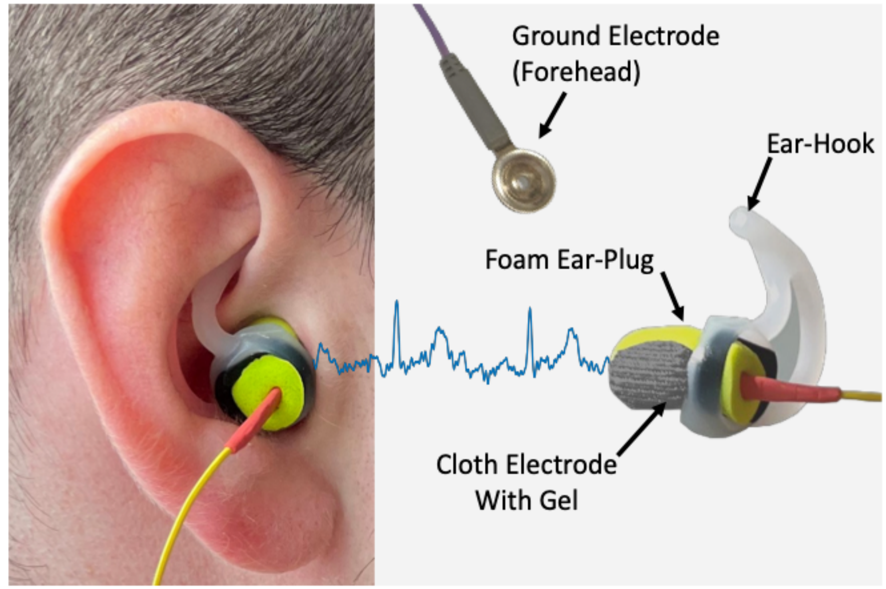




Design starting from first principles

A CNN interpretation through deep matched filters yields ear-electrocardiogram





3614 IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, VOL. 53, NO. 6, JUNE 20

Convolutional Neural Networks Demystified: A Matched Filtering Perspective-Based Tutorial

Ljubiša Stanković, *Fellow, IEEE*, and Danilo Mandic, *Fellow, IEEE*

***Abstract*—Deep neural networks (DNNs) and especially convolutional neural networks (CNNs) have revolutionized the way we approach the analysis of large quantities of data. However, the**

multimedia communication and social networks, and increas ingly from Internet-enabled autonomous electronic device e.g., the Internet of Things (IoT). The usefulness of these dat

Learning from data $\looparrowright$ mathematical formalism of the statistical estimation paradigm

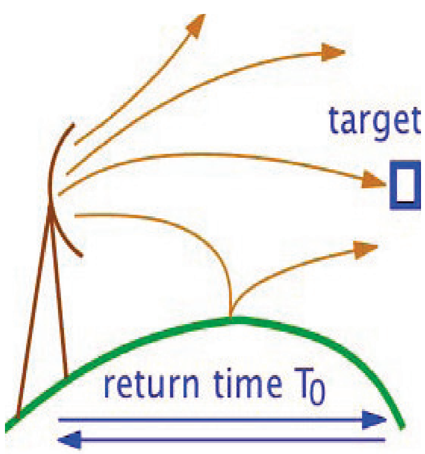
Problem: Based on an N-point dataset $\mathbf{x} = \left[x[0], x[1], \ldots, x[N-1]\right]^T$

Task: Find an **unknown parameter**, θ, based on the data $\mathbf{x}$, in order to define a ***statistical estimator*** (e.g. $\hat{\theta}$ can be the sinewave frequency)

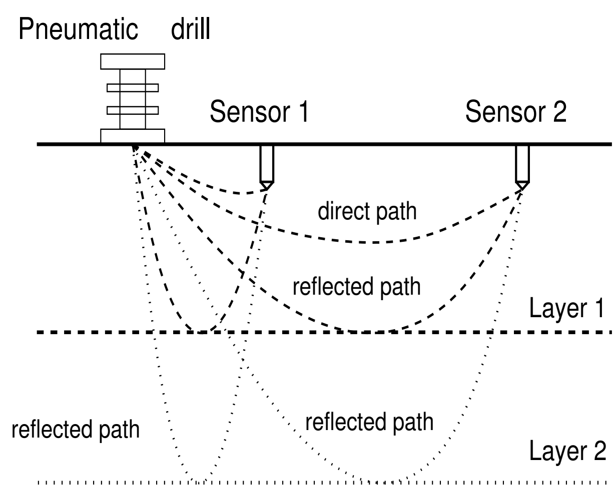
$$\hat{\theta} = g\big(x[0], x[1], \ldots, x[N-1]\big), \qquad g \text{ is some function}$$

This is formalised as **parameter estimation from random signals**

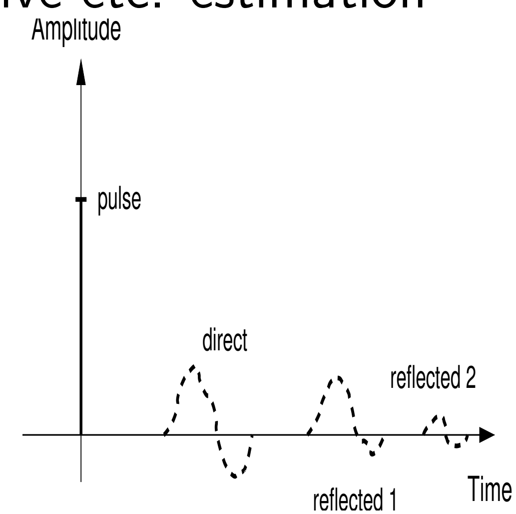
Depending on the choice of g we can talk about: ⊛ linear, ⊛ nonlinear, ⊛ maximum likelihood, ⊛ minimum variance, ⊛ adaptive etc. estimation



Radar



Seismics



Seismic impulse response

Example 3: Estimating spectral peaks ↬ statistical way

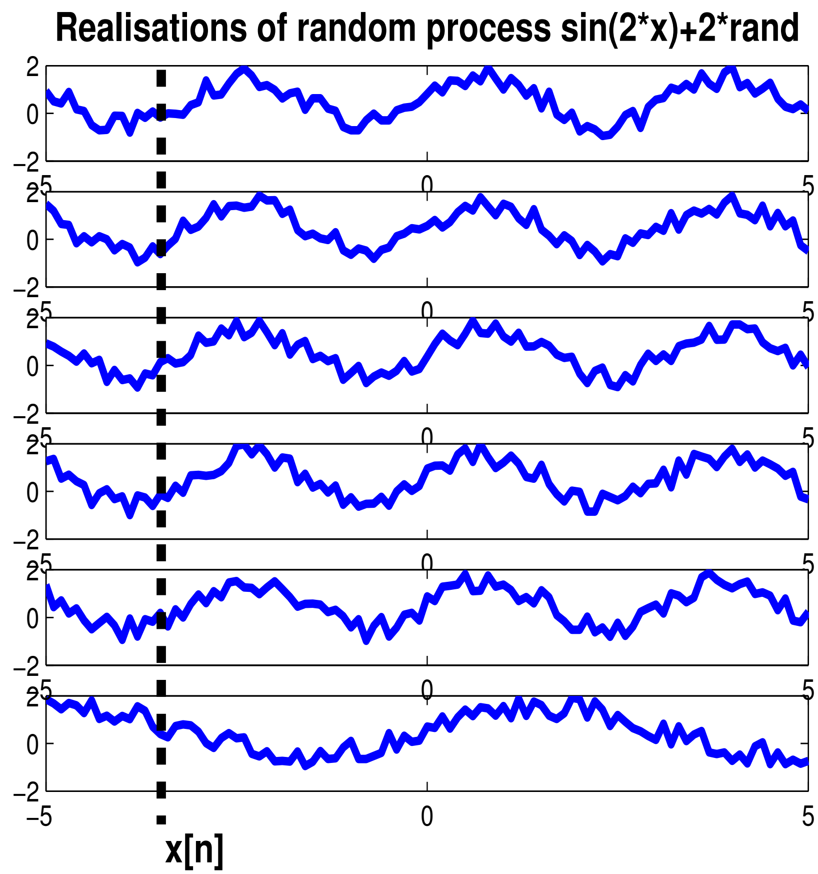
Ensemble ↬ collection of **all possible realisations** of a **random signal**

Consider 6 realisations of the process

$$y = \sin(x) + rand \;\; \Leftrightarrow \;\; '\mathrm{det}' + '\mathrm{stoch}'$$

- our aim is to **estimate** frequency f
- sinusoid ↬ ***deterministic***
- noise ↬ ***stochastic***

☞ We need to use a **statistical** estimator, which will be ***unbiased*** and will have ***minimum variance***



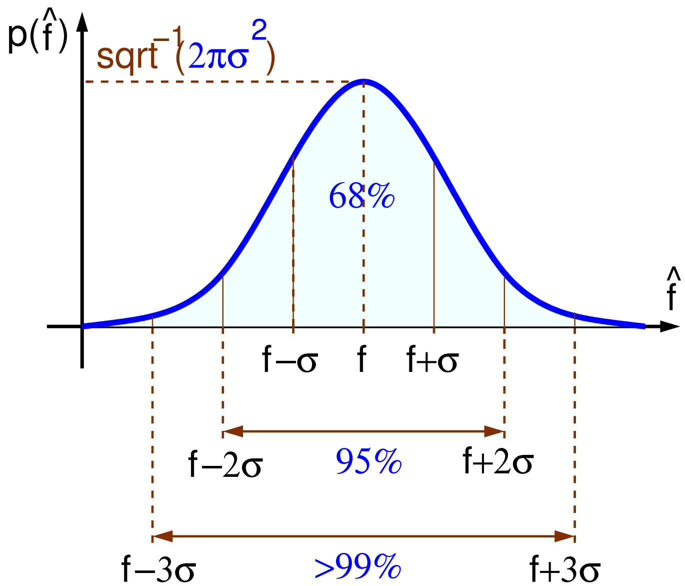
Can we average both **along** one and **across** all realisations?

Discrete–time estimation problem

We almost always work with samples of the *observed signal*, $x[n]$, that is, *signal*, $s[n]$, + *noise*, $w[n]$.

For example, when estimating an unknown frequency, f, we have

$$x[n] + s[n; f] + w[n] \qquad w[n] \text{ is random, e.g. } w \sim \mathcal{N}(0, \sigma^2)$$



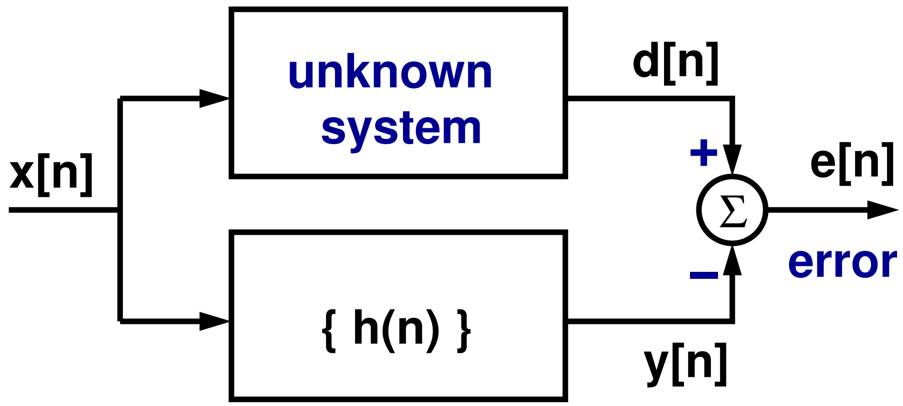
Task: Given a dataset, $x[0], x[1], \ldots, x[N-1]$, find *estimators* (functions) which map the observed data into the estimates

$$\hat{f} = g(x[0], x[1], \ldots, x[N-1])$$

Our thought process: Each time we observe $x[n]$, it contains same $s[n; f]$ but a different realisation of noise, $w[n]$, so that $\hat{f}$ **is also a random variable** (it has a *pdf*).

Course goal: Find optimal estimators, with $E\{\hat{f}\} = f$, and $\sigma^2_{\hat{f}}$ small.

Example 4: Use of estimation in system identification



Task: Find the set of coefficients, $\{h(n)\}$, such that the output of our assumed system model, $y[n]$, is as similar as possible to the output of the original system, $d[n]$. *This similarity is measured in some statistical or probabilistic sense, for example through error power,* $E\{e^2\}$.

This error is then used to update the estimates of the coefficients, $\{h(n)\}$.

Measure of "goodness" is the distribution of the error $\{e[n]\}$.

Ideally, the error should be zero mean, white, and uncorrelated with the output signal

Course aims: More specifically

- To introduce fundamentals of the analysis of *real-world discrete-time random signals*, their properties and representations
- To introduce linear stochastic models for *time series analysis*
- To provide a grounding in *linear estimation theory*, to facilitate the design and analysis of *statistical statistical signal processing and machine learning* algorithms
- Based upon these concepts, we will:
  - ⊛ Explain the notion of signal modelling, its applications, and its relations to parametric spectral estimation
  - ⊛ Describe the need for statistical and adaptive learning theories
- To illuminate the application of statistical estimation theory (inference) in prediction, equalisation, echo and noise cancellation, biomedical eng.
- To introduce and verify theoretical and practical bounds on the performance of any statistical estimation and learning algorithm, from linear regression to nonlinear DNNs

Course structure

The course is divided roughly into four parts:

1. Introduction to Statistical Estimation Theory

   discrete random signals, moments, bias-variance dilemma, curse of dimensionality, sufficient statistics

2. Statistical Modelling, Estimation Theory and Performance Bounds

   linear stochastic models, ARMA model, properties of estimators, Cramer Rao performance bound, minimum variance unbiased (MVU) estimator

3. Practical Statistical Estimators and Inference

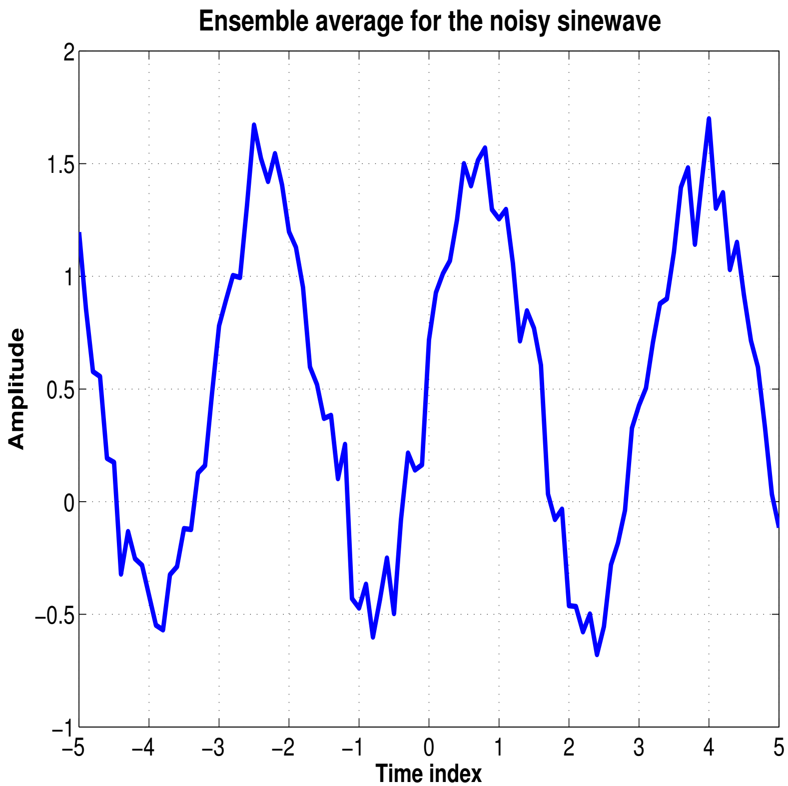
   best linear unbiased estimator (BLUE), maximum likelihood (ML) estimation, multivariate estimators, Bayesian estimation (optional)

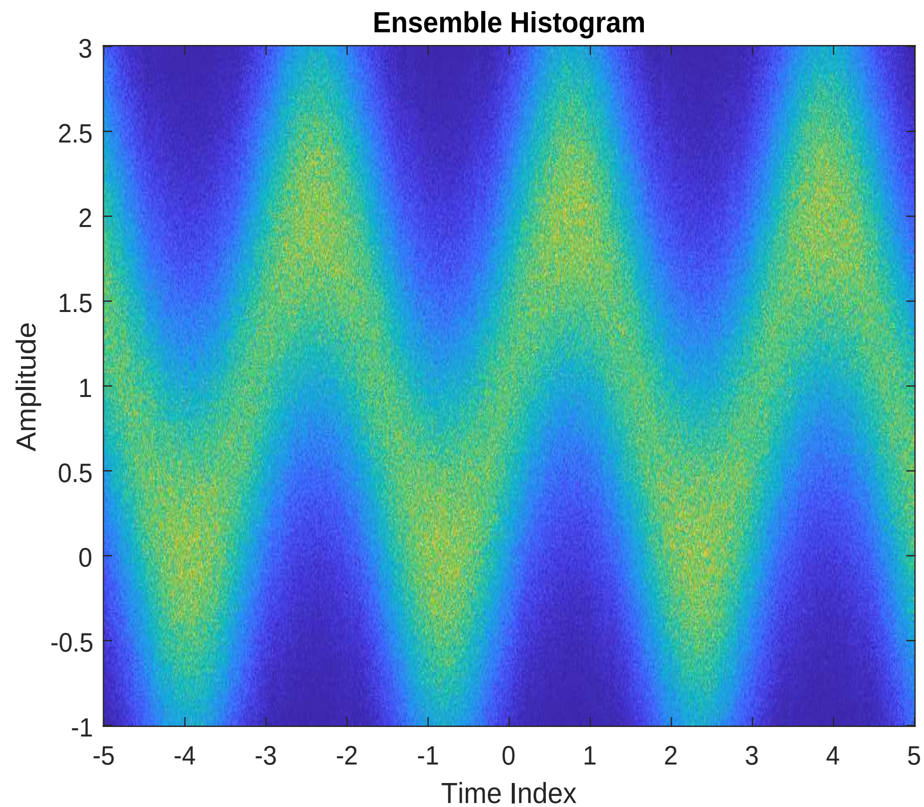
4. Mean Square Error (MSE) based Estimation (block and adaptive)

   orthogonality principle, block and sequential forms of Least Squares, Wiener filter, adaptive filtering, concept of an artificial neuron

Lecture 1: Background on random signals

(for illustration, consider the noisy sinewave from Slide 13)





The pdf at time instant n is different from that at m, in particular:

$$\mu(n) \neq \mu(m) \qquad m \neq n$$

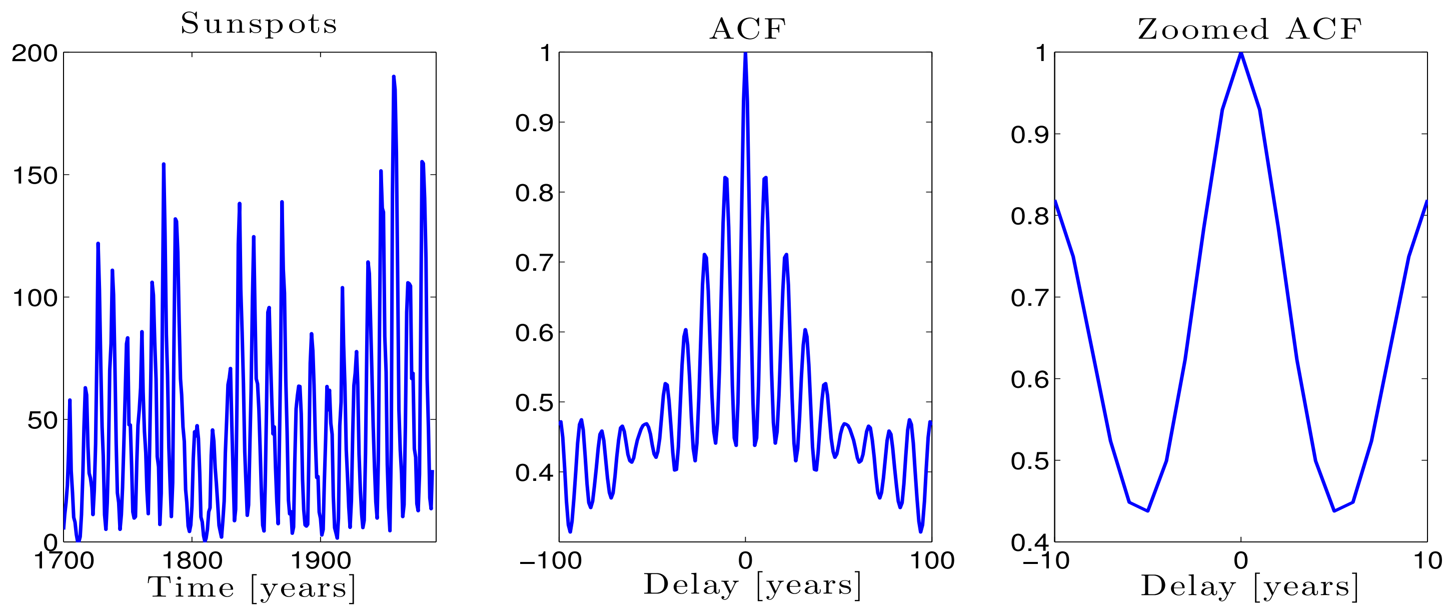
Left & Right: Ensemble average

$$\sin(2x) + 2 * randn + 1$$

Left: 6 realisations, **Right:** 100 realisations (and the overlay plot)

Example 5: Sunspot number estimation

The power of $x(n) = a_1 x(n-1) + a_2 x(n-2) + \cdots + a_p x(n-1) + w(n)$



$\mathbf{a}_1 = [0.9295]$ $\quad$ $\mathbf{a}_2 = [1.4740, -0.5857]$

$\mathbf{a}_3 = [1.5492, -0.7750, 0.1284]$

$\mathbf{a}_4 = [1.5167, -0.5788, -0.2638, 0.2532]$

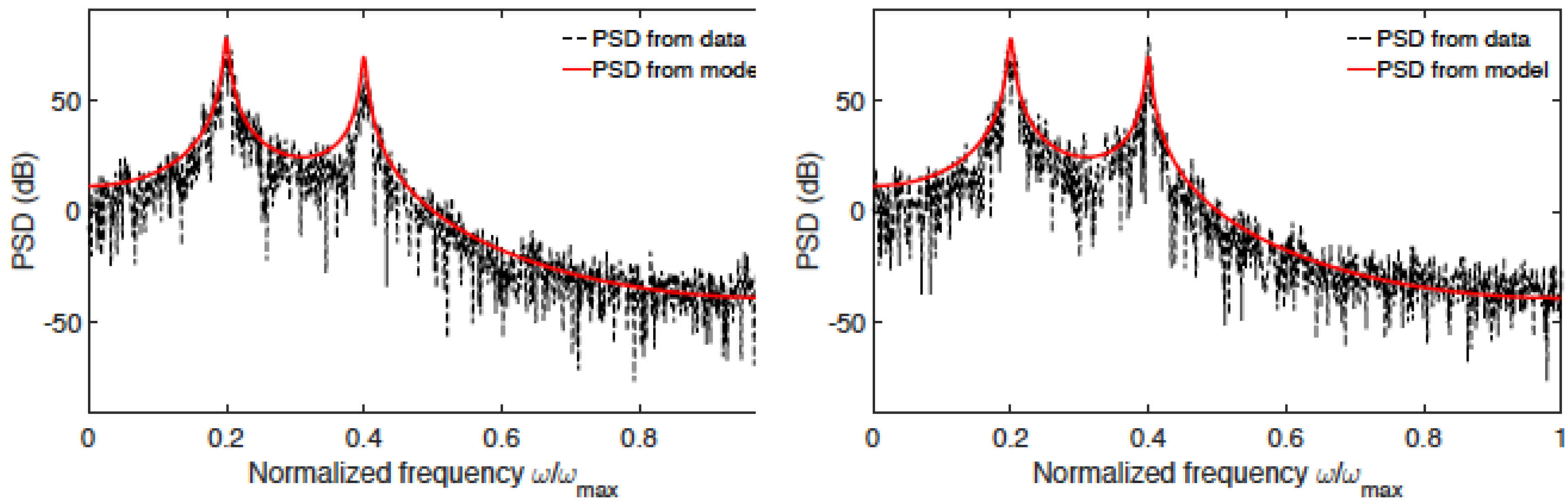
$\mathbf{a}_5 = [1.4773, -0.5377, -0.1739, 0.0174, 0.1555]$

$\mathbf{a}_6 = [1.4373, -0.5422, -0.1291, 0.1558, -0.2248, 0.2574]$

$\looparrowright$ The sunspots model is $\quad x[n] = 1.474\, x[n-1] - 0.5857\, x[n-2] + w[n]$

Lecture 2: Time series analysis $\rightsquigarrow$ linear stoch. models

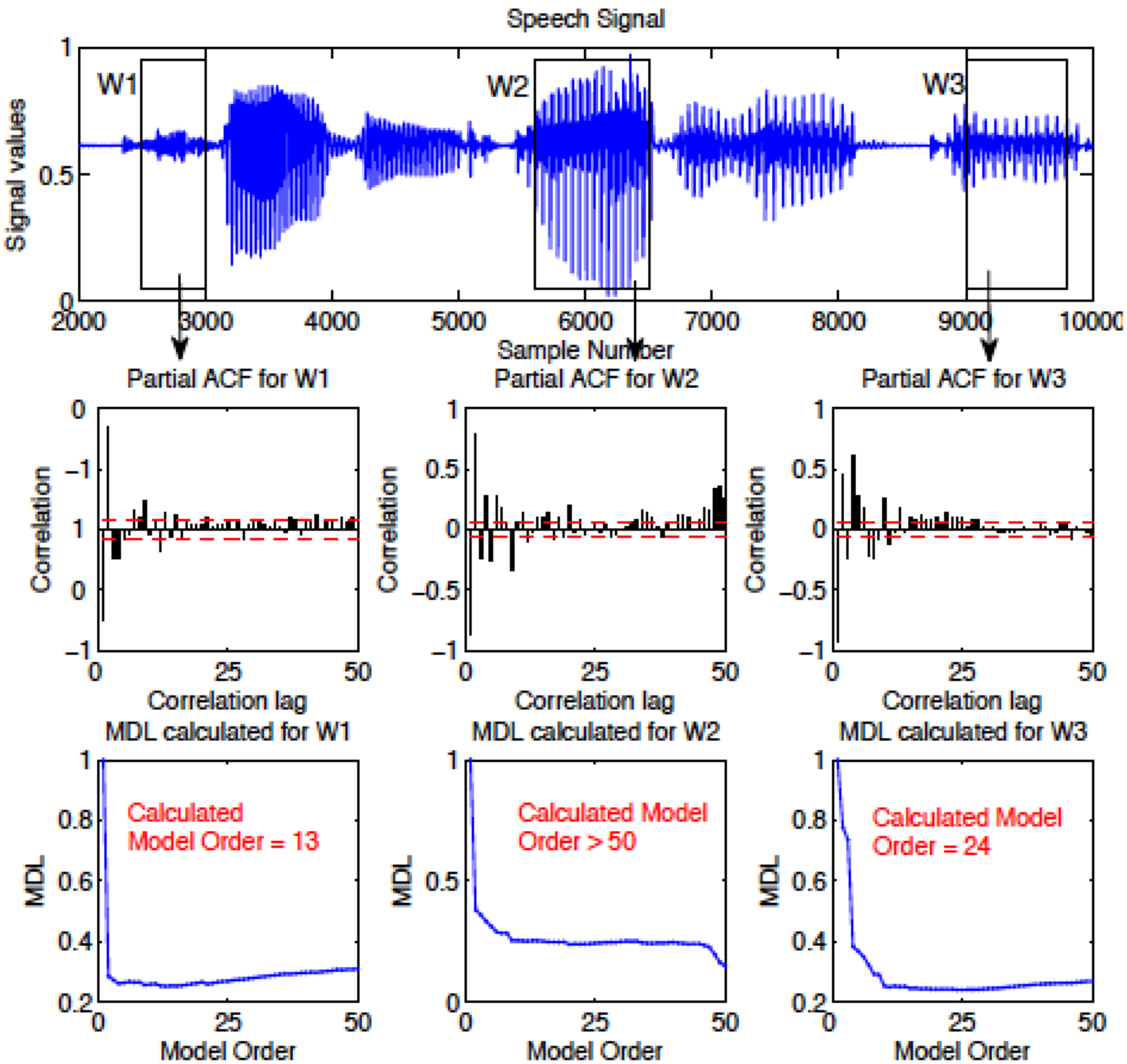
Is it possible to represent a very long random signal with only a few parameters?



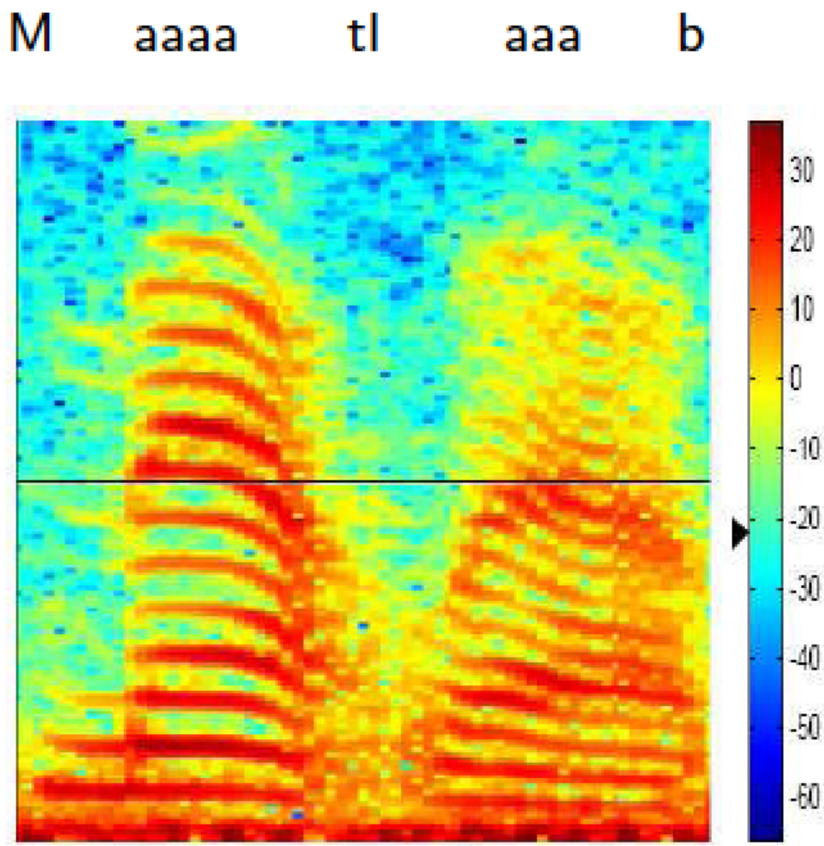
- The different realisations lead to different Emprical PSD's (in thin black)
- The theoretical PSD from the model is consistent regardless of the data (in thick red)

```
N = 1024;
w = wgn(N,1,1);
a = [2.2137, -2.9403, 2.1697, -0.9606];          % Coefficients of AR(4) process
a = [1 -a];
x = filter(1,a,w);
xacf = xcorr(x);                                   % Autocorrelation of AR(4) process
dft = fft(xacf);
EmpPSD = abs(dft/length(dft)).^ 2;                 % Empirical PSD obtained from data
ThePSD = abs(freqz(1,a,N,1)).^ 2 ;                 % Theoretical PSD obtained from model
```

Example 6: Dealing with nonstationary signals



- Consider a real–world speech signal, and thee different segments with different statistical properties
- **Different AR model orders required for different segments of speech $\hookrightarrow$ opportunity for content analysis!**
- To deal with nonstationarity we need short sliding data windows



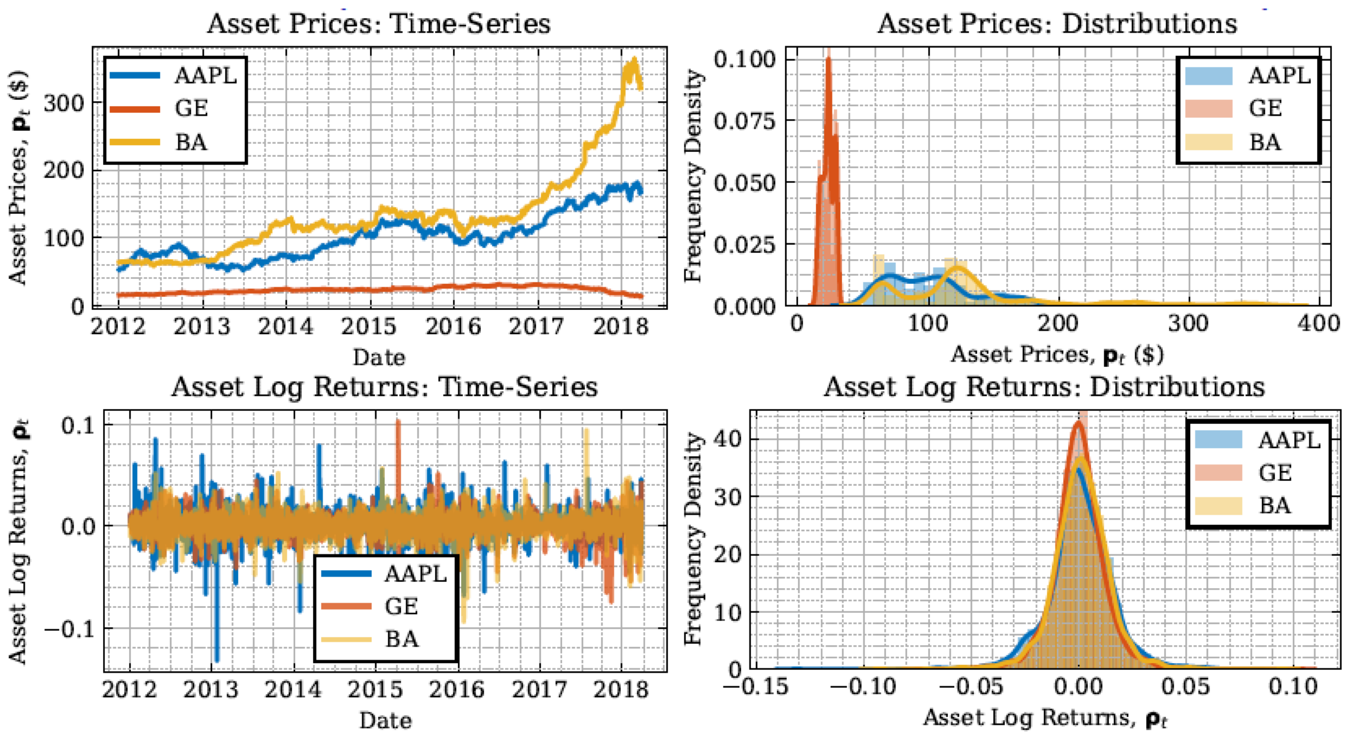
Horizontal axis: time
Vertical axis: frequency

An enabling technology in many DSP applications

- Radar and sonar: estimation of the range and azimuth
- Image analysis: motion estimation, segmentation
- Speech: features used in recognition and speaker verification
- Seismics: oil reservoirs
- Communications: equalization, symbol detection
- Biomedicine: various applications

Often we can resort to (approximately) Gaussian distrib.

Top panel. Share prices, p_n, of Apple (AAPL), General Electric (GE) and Boeing (BA) and their histogram (right). **Bottom panel.** Logarithmic returns for these assets, $ln(p_n/p_{n-1})$, that is, the log of price differences at consecutive days (left) and the histogram of log returns (right).



Clearly, by a suitable data transformation, we may arrive at symmetric distributions which are more amenable to analysis (bottom right).

Importance of establishing optimum performance bounds

A typical artefact in teleconferencing, where an algorithm which provides artificial background cannot cope with movement

You will learn how to establish the optimal theoretic performance bounds in both block-based and real-time adaptive data analysis.

These will serve to:

- Indicate the quality of your algorithm/strategy against the best achievable performance for that class of estimators

- Help identify an error in your algorithm, if its performance appears better than the optimal performance bound.

Lecture 4: Bias-variance dilemma $\rightsquigarrow$ Minimum Variance Unbiased estimation, rigorous performance bounds

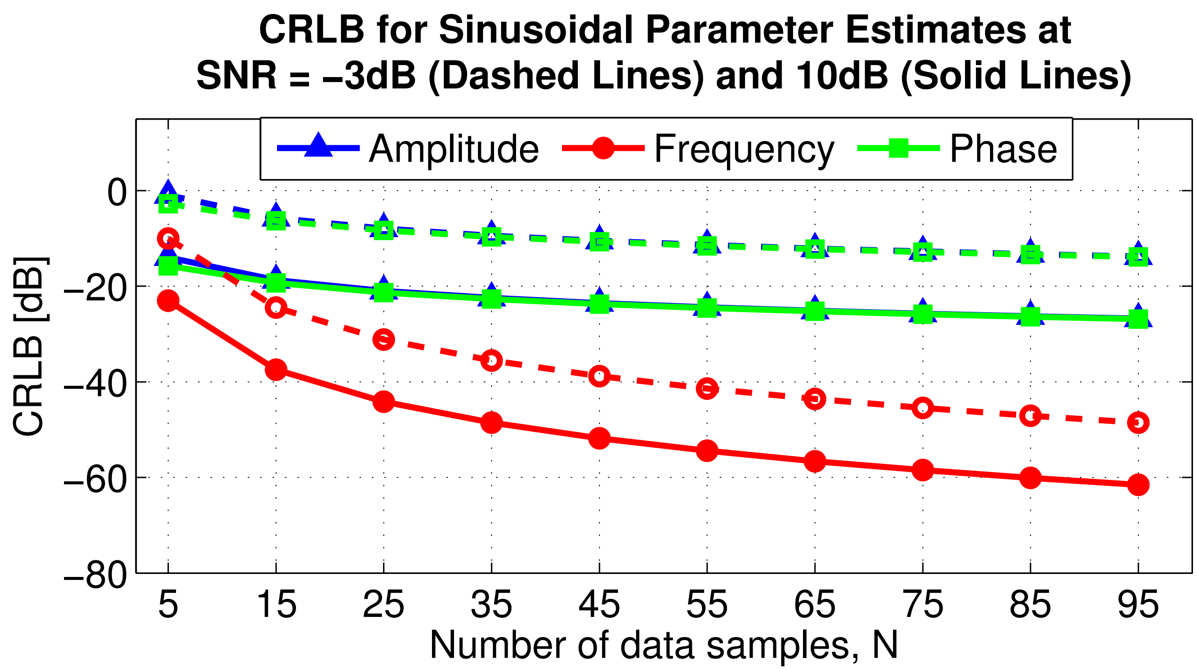
☞ variance of the estimated parameters is sensitive to data length

Consider a sinusoid $x[n] = A\,cos(2\pi f_0 n + \Phi) + w[n], \quad w[n] \sim \mathcal{N}(0, \sigma^2)$

Task: Find the parameters A, f_0, Φ, from the noisy measurements $x[n]$

We will show that the optimal estimators obey (where $\eta = \frac{A^2}{2\sigma^2}$ is SNR):

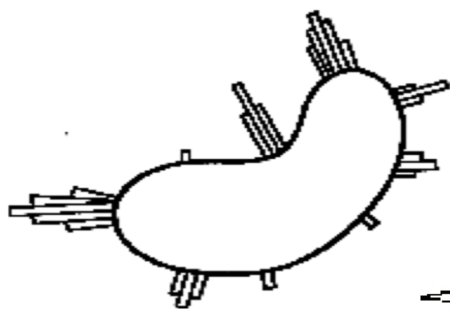
$$var(\hat{A}) \geq \frac{2\sigma^2}{N} \qquad var(\hat{f}_0) \geq \frac{12}{(2\pi)^2 \eta N(N^2-1)} \qquad var(\hat{\Phi}) \geq \frac{2(2N-1)}{\eta N(N+1)}$$



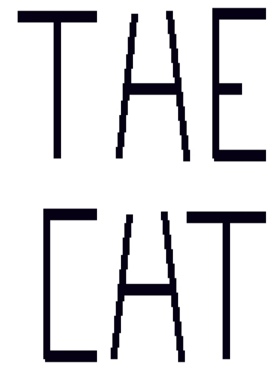
Sufficient statistics, goodness of an estimator

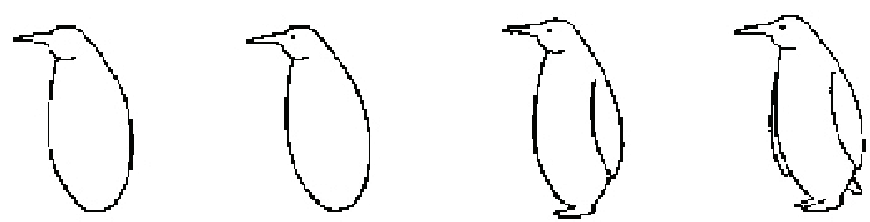
Example 7a: The drawing of a bean (top) and the histogram of eye dwellings (bottom)

A simple closed curve.



Example 7b: Read the words below ... now read letter by letter ... are you still sure?

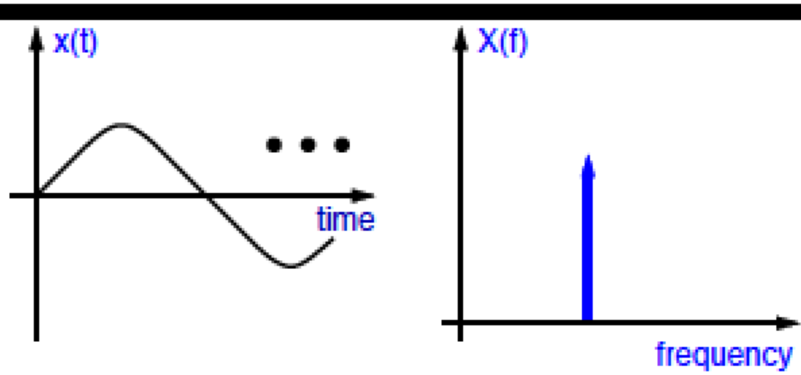


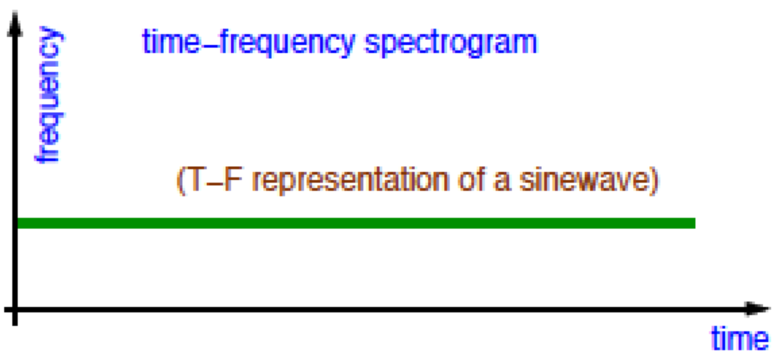


Example 7c: Is the drawing on the left still a penguin?

So, what is the sufficient information to 'estimate' an object?

Sinusoidal frequency estimation

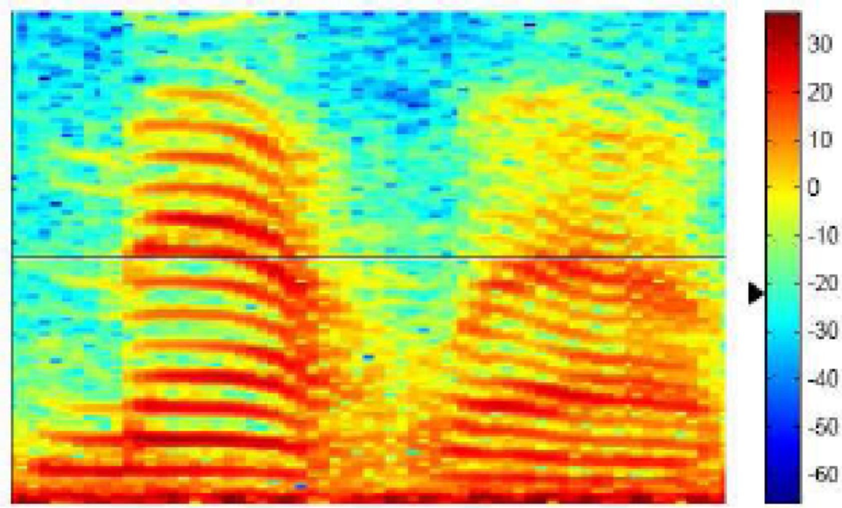




- Ramp in time $\looparrowright$ DC level in time (via differentiation)
- Chirp in time $\looparrowright$ ramp in T-F

Transforming other problems

time-frequency representation

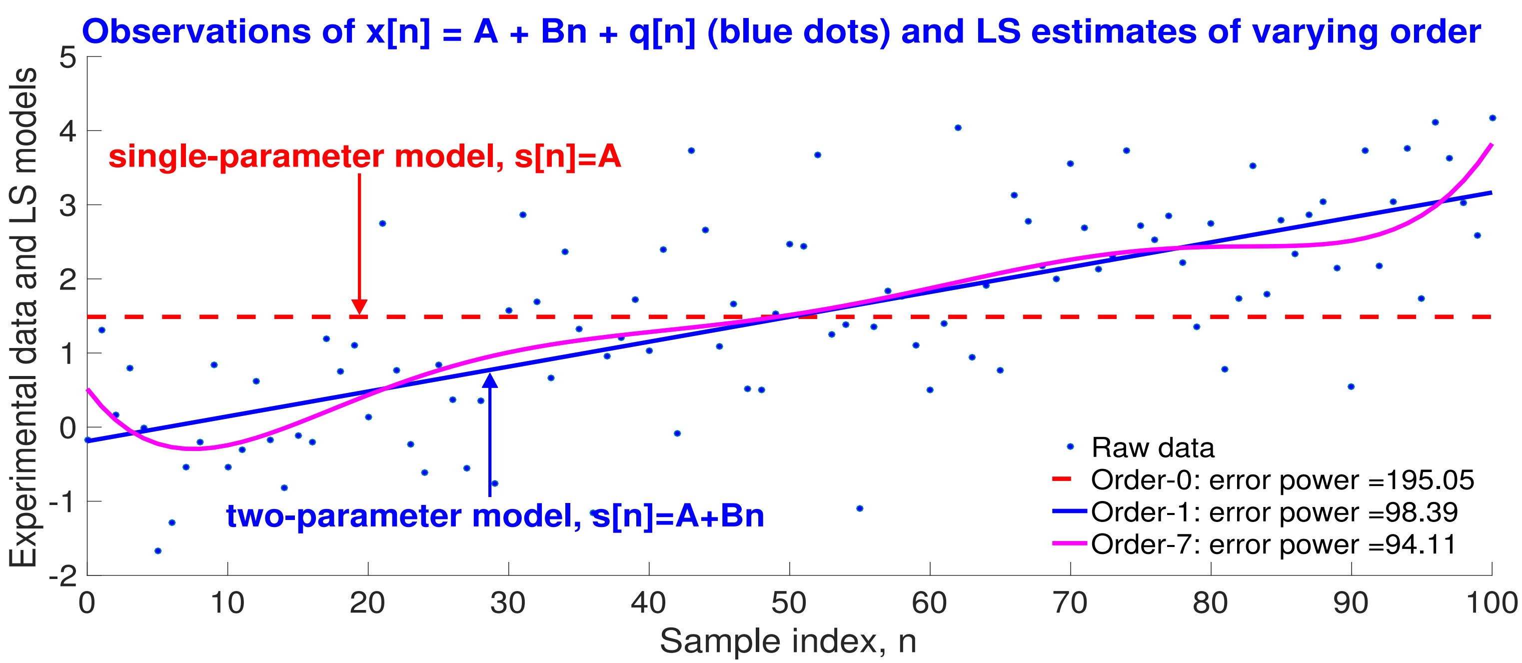


horizontal: time vertical: frequency

This is a T-F representation of a waveform of the word "matlab"

DC-level like harmonics for "a"

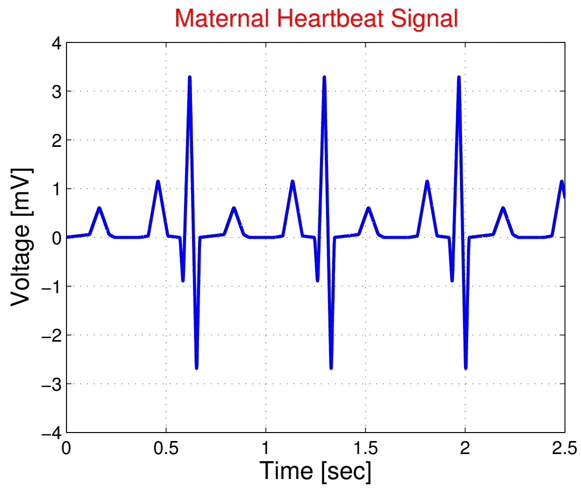
- The LS approach can be interpreted as the problem of approximating a data vector $\mathbf{x} \in \mathbb{R}^N$ by another vector $\hat{\mathbf{s}}$ which is a linear combination of vectors $\{\mathbf{h}_1, \ldots, \mathbf{h}_p\}$ that lie in a p-dimensional subspace $S \in \mathbb{R}^p \subset \mathbb{R}^N$
- The problem is solved by choosing $\hat{\mathbf{s}}$ so as to be an orthogonal projection of $\mathbf{x}$ on the subspace spanned by $\mathbf{h}_i, i = 1, \ldots, p$
- The LS estimator is very sensitive to the correct deterministic model of $\mathbf{s}$, as shown in the figure below for the LS fit of $x[n] = A + Bn + q[n]$.

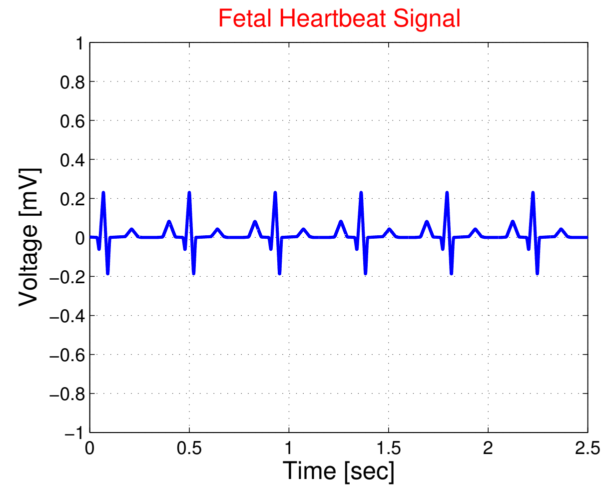


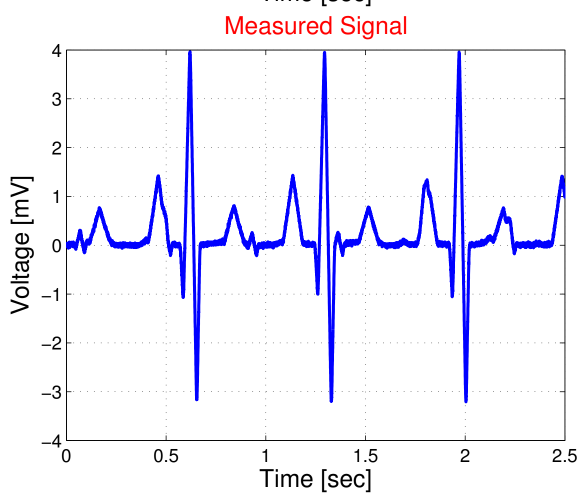
Example 8: Least squares and sequential LS in action

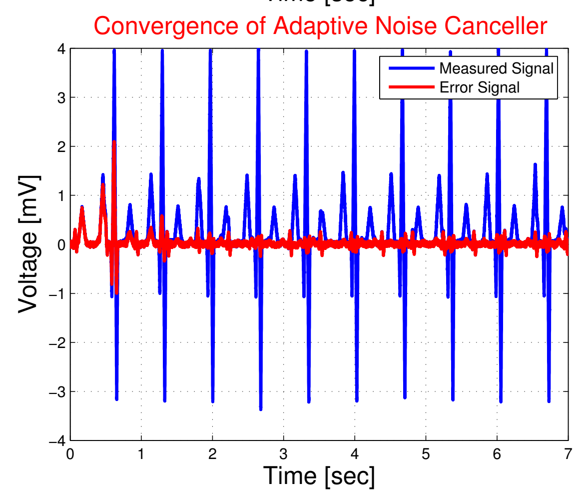
Maternal ECG signal

Foetal heartbeat





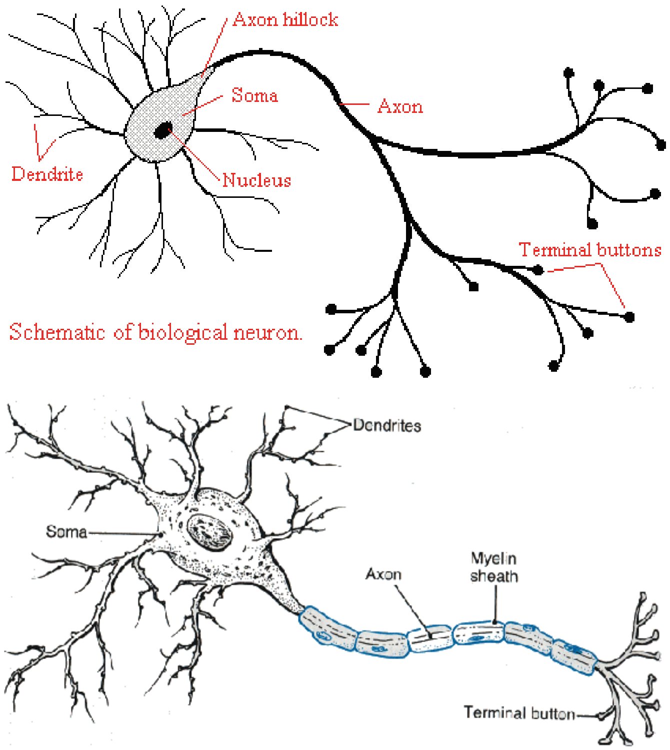




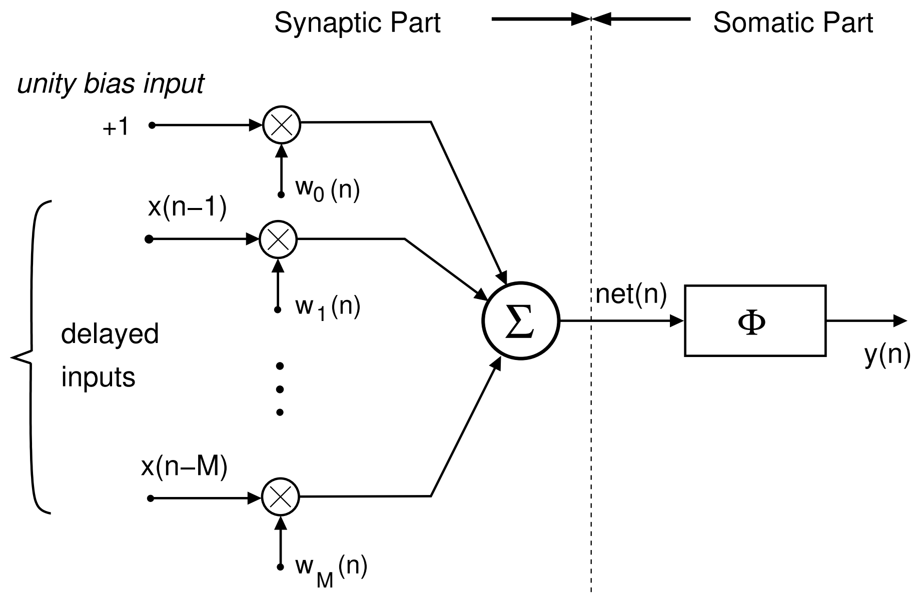
Measured foetal ECG

Maternal and foetal ECG

Lecture 7: Adaptive systems
Linear and neural adaptive filters



Biological neuron

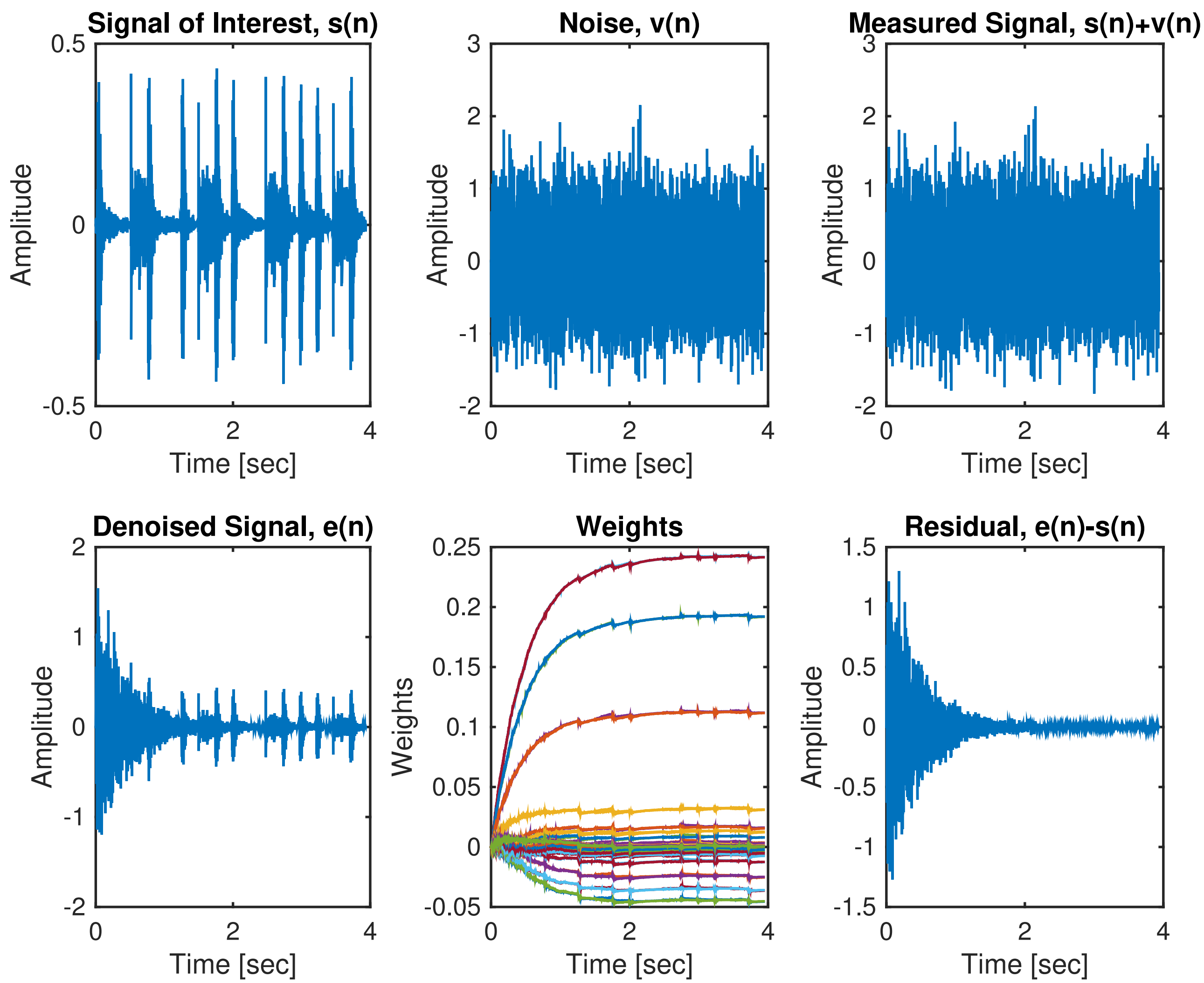


Model of an artificial neuron

- delayed inputs $\mathbf{x}$
- bias input with unity value
- sumer and multipliers
- output nonlinearity

Example 9: Acoustic noise cancellation (e.g. on airplane)

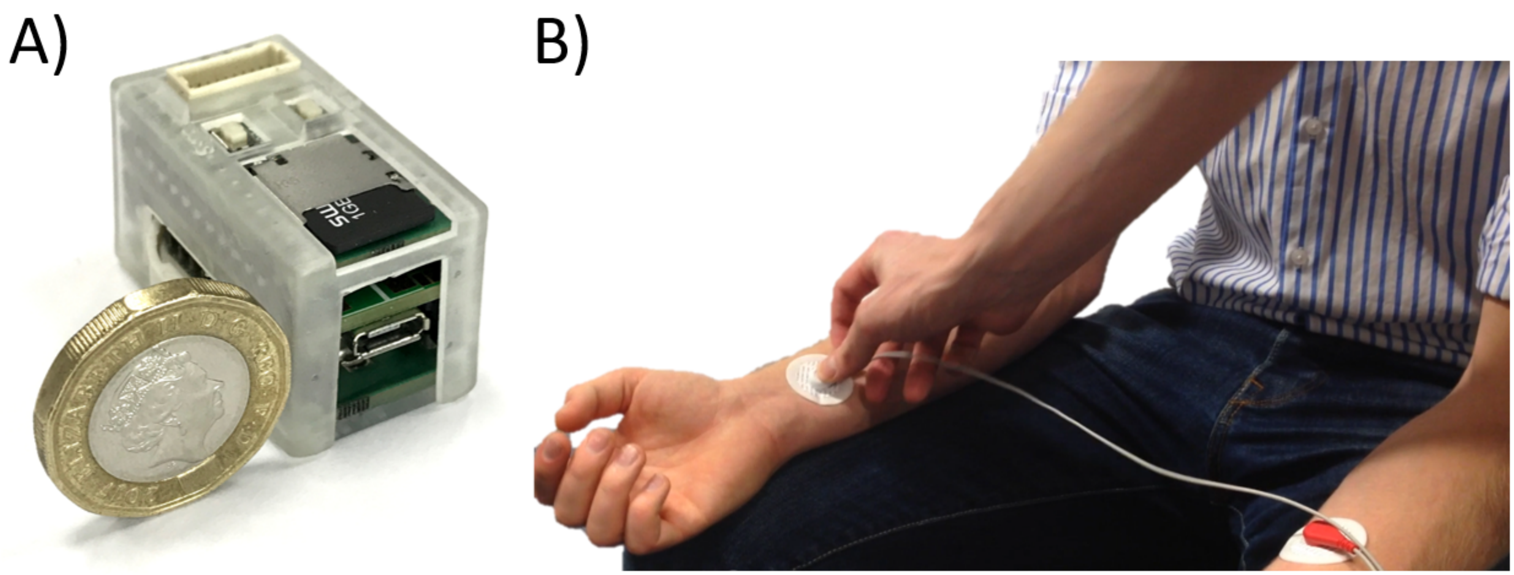
Denoising_Reference_Drum



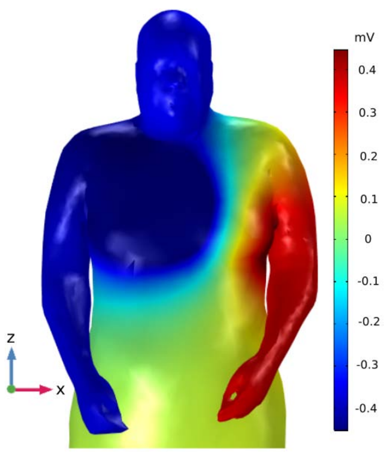
Coursework: Your own speech and biosignal recordings

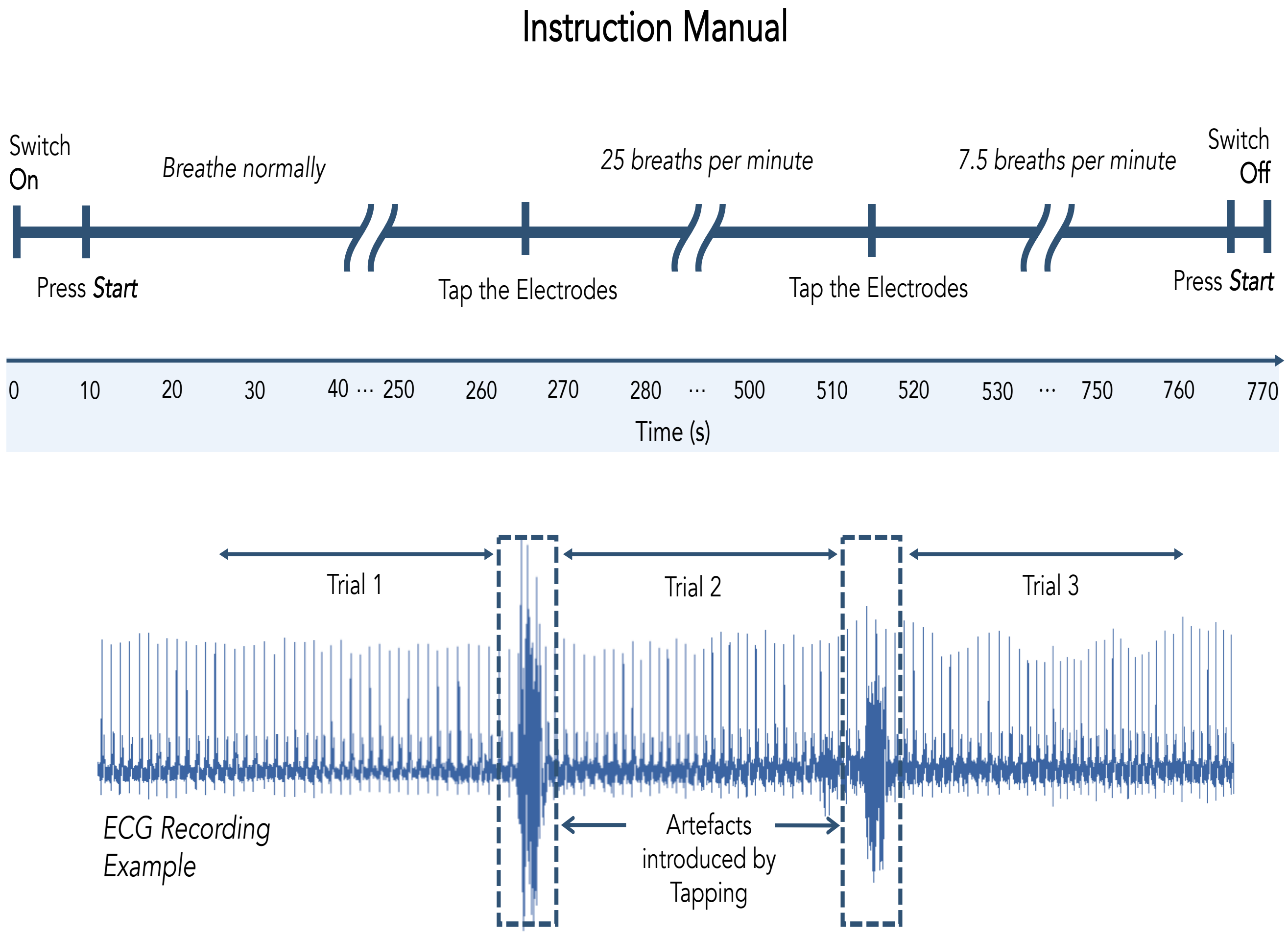
- Our own custom-made portable signal acquisition device – the iAMP – is designed to record any biopotentials (e.g. ECG, EEG, and EMG) from up to eight channels

- It consists of an analogue-to-digital converter (ADC), a microprocessor and a secure digital (SD) card slot to store the data



Coursework: Recording your own ECG

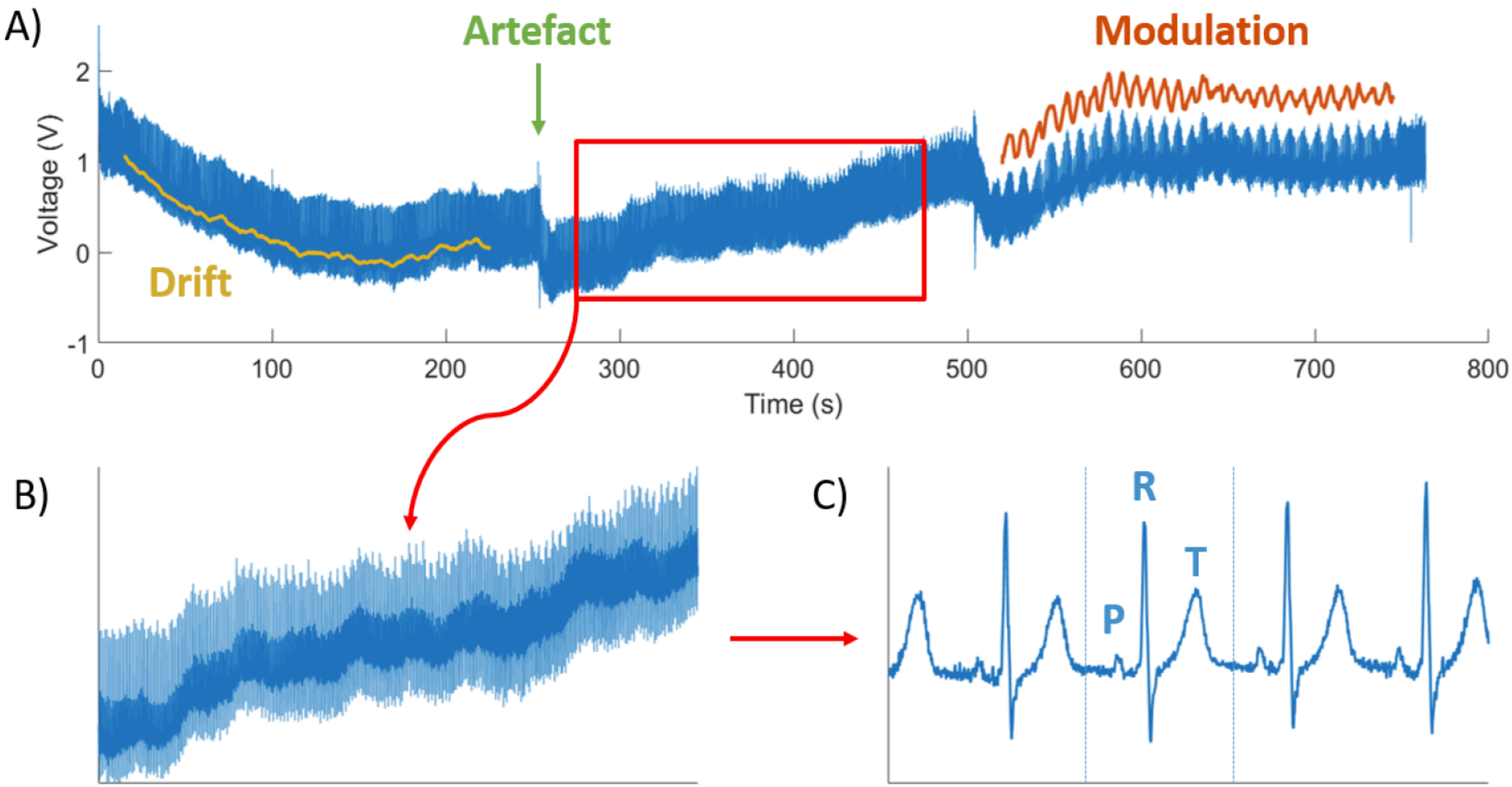




Left: Electric heart potentials on human body. Right: Experiment protocol

Coursework: Gain experience with real–world data

Example relevant for eHealth: Estimate your own ECG from your wrists.



Course format

Lecture notes with problem/answer sets and coursework.

- Coursework involves the implementation of the algorithms we discuss in the class
- We will regularly discuss coursework and Matlab implementation

  Prerequisites:

  ⊛ There are no prerequisites, although DSP and basic probability would be useful

  ⊛ The course is aimed to be self-contained

  ⊛ Due to algorithm implementation, knowledge of Matlab is important

Assessment:

100% Coursework assignments. There are 5 Assignments (from random signals to audio denoising) $\looparrowright$ **Matlab based**

Feedback: $\looparrowright$ **after completing Assignment 1**

Reference material

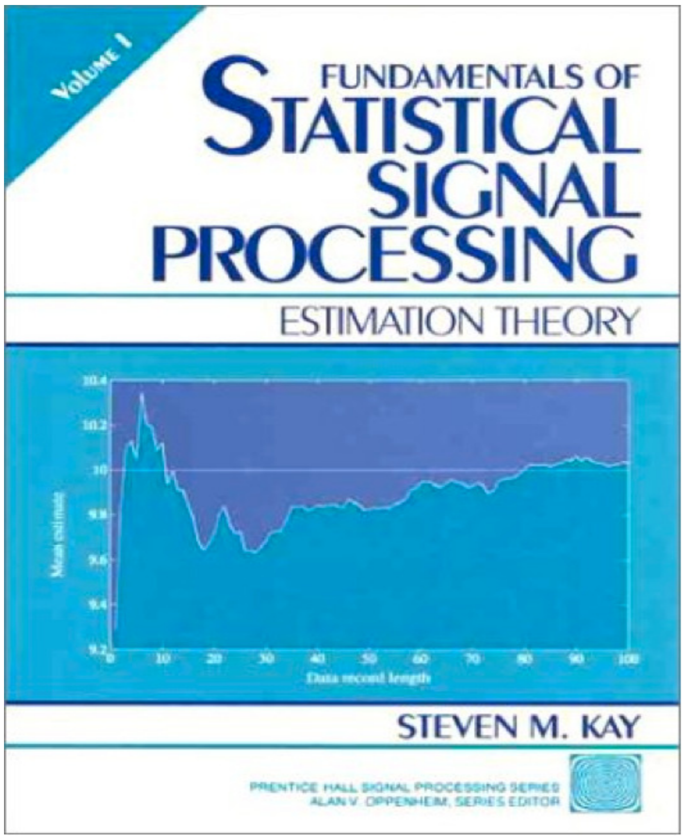
Course notes and problem sets: Prof D. Mandic

- There is no single textbook that covers all the material in the course
- We will use S. Kay's book for the first part of the course (an excellent text, covers most of the estimation theory, well worked-out examples, highly recommended, has many editions)
- For parametric modelling we will use the Box & Jenkins book (a 'bible' for time series analysis, easy to read, excellent examples, used by people in engineering, physics, finance, has many editions)
- For the least squares part, we will use M. Hayes' book (wider scope than Kay's book, less detailed derivations, a must have for practitioners)
- For more background and further reading, the book by S. Haykin (Adaptive Filters) and D. Mandic & J. Chambers (Recurrent Neural Networks)

The course is self-contained: most of the material is already in course notes

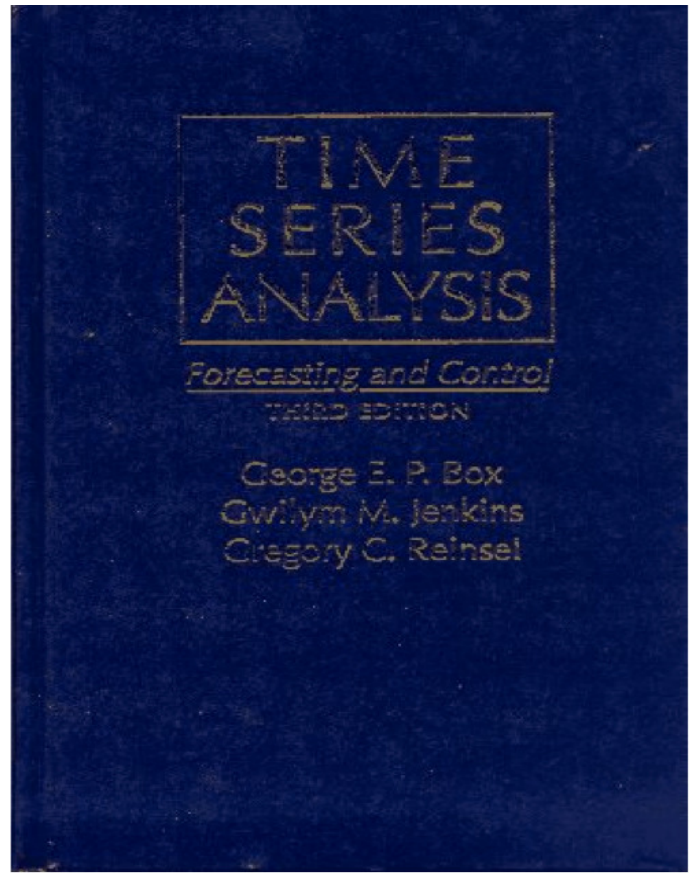
Textbooks: Recommended

S. Kay (*Estimation Theory*, several editions)



a comprehensive account of estimation theory

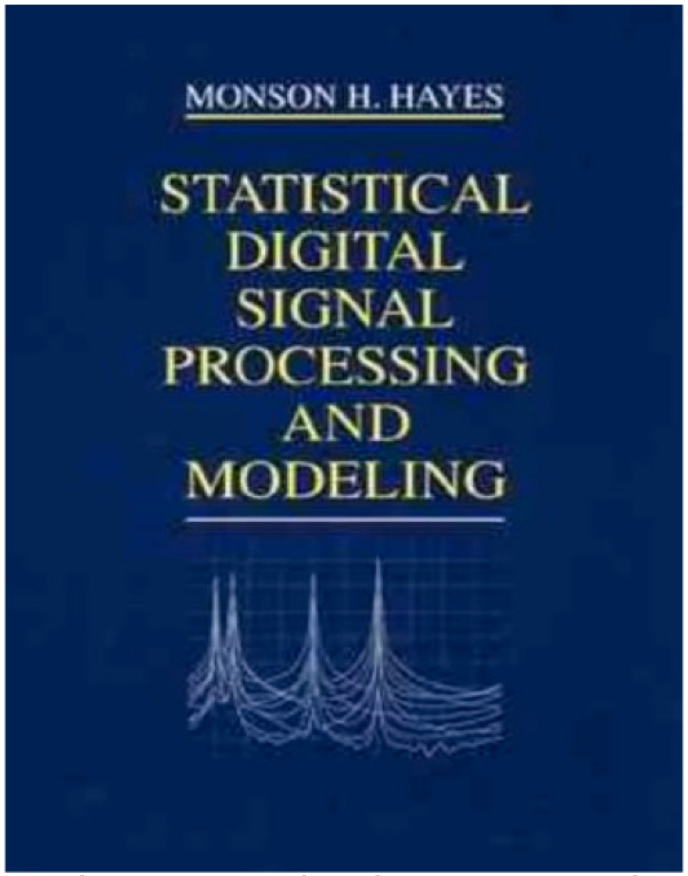
G. Box and G. Jenkins (*Time Series Analysis*, several editions)



linear stochastic models

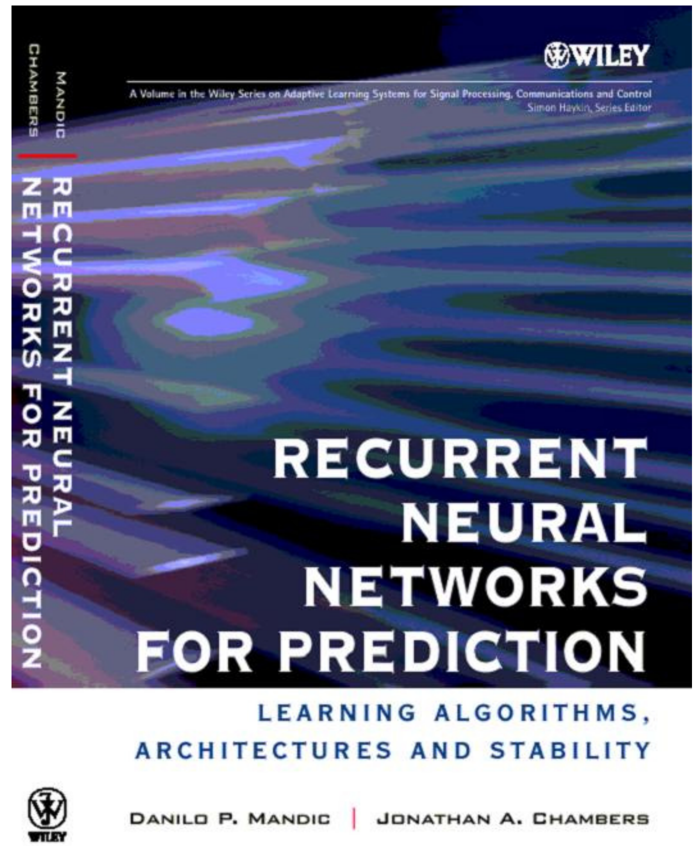
Textbooks: Additional reading (optional)

M. Hayes (*Statistical Signal Processing and Modeling*, several editions)



stochastic and adaptive models

D. Mandic and J. Chambers (*Recurrent Neural Networks*, Wiley 2001.)



(what can I say) - neural models

Course plan

2 Lect: Week 2: Course introduction and motivation, background

3 Lect: Week 2-3: Discrete time random signals, linear stochastic (ARMA) models

4 Lect: Week 3-5: Minimum variance unbiased estimation, Cramer-Rao bound

4 Lect: Week 6-7: Constrained estimators, BLUE, Maximum likelihood

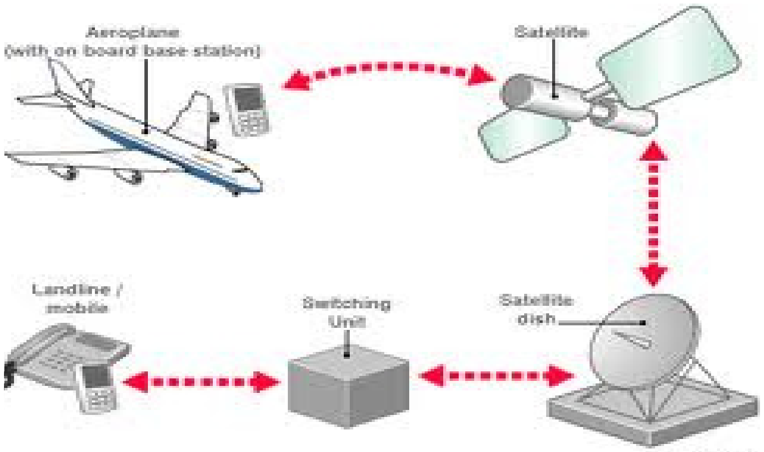
6 Lect: Week 8-9: Block, sequential and adaptive estimators

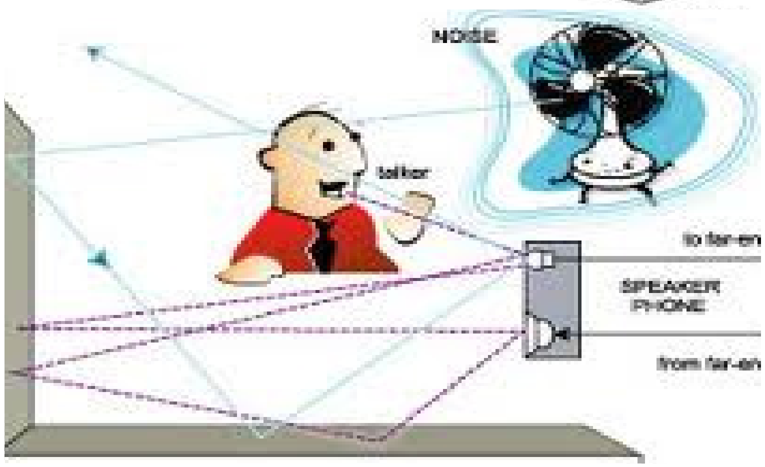
1 Lect: Week 10: Consolidation and research directions

Course web page: www.commsp.ee.ic.ac.uk/∼mandic/Teaching

Lectures, additional reading, homework, problem sets, and other material will be put on course webpage

Statistical Sig Proc & Inference $\looparrowright$ A stealth technology





- **There will always be signals**
- **They always need processing**
- **There will always be new mathematics for processing them**

$\leadsto$ **Guaranteed job security**

Appendix: Probability vs. Statistics

For discrete RVs, $E\{X\} = \sum_{i=1}^{I} x_i P_X(x_i)$, where P_X is the probability function

Probability: A data modelling view, describes how data **will likely behave**

for example: $$average = E\{X\} = \int_{-\infty}^{\infty} x\, p_X(x)\, dx$$ no data here

Notice that there is no explicit mention of data here $\looparrowright$ x is a dummy variable and p_X is the pdf of a random variable X.

Statistics: A data analysis view, determines how data **did behave**

for example: $$average = \frac{1}{N}\sum_{n=0}^{N-1} x[n]$$ no pdf here

Example: Consider N coarse-quantised data points, $x[0], \dots, x[N-1]$. The signal has $M \ll N$ possible amplitude values, $V_1, \dots, V_M$, with the corresponding relative frequencies, $N_1, \dots, N_M$. Calculate the mean, $\bar{x}$.

Solution:

$$\bar{x} = \frac{1}{N}\sum_{n=0}^{N-1} x[n] \quad = \quad \frac{1}{N}\sum_{m=1}^{M} V_m N_m \quad = \quad \sum_{m=1}^{M} V_m \underbrace{\frac{N_m}{N}}_{\approx\ P(x=V_m)}$$

Appendix: Probability vs. Statistics

(for discrete RVs, $E\{X\} = \sum_{i=1}^{I} x_i P_X(x_i)$, where P_X is the probability function)

Probability: A data modelling view, describes how data **will likely behave**

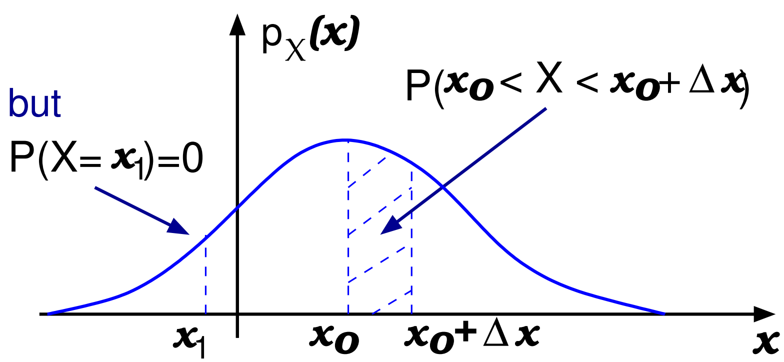
for example: $$average = E\{X\} = \int_{-\infty}^{\infty} x\, p_X(x)\, dx$$ no data here

Notice that there is no explicit mention of data here $\looparrowright$ x is a dummy variable and p_X is the pdf of a random variable X.

Statistics: A data analysis view, determines how data **did behave**

for example: $$average = \frac{1}{N}\sum_{n=0}^{N-1} x[n]$$ no pdf here

Vagaries of probability: $P(x_0 < X < x_0 + \Delta x) = \int_{x_0}^{x_0+\Delta x} p_X(x)dx$



Notice that

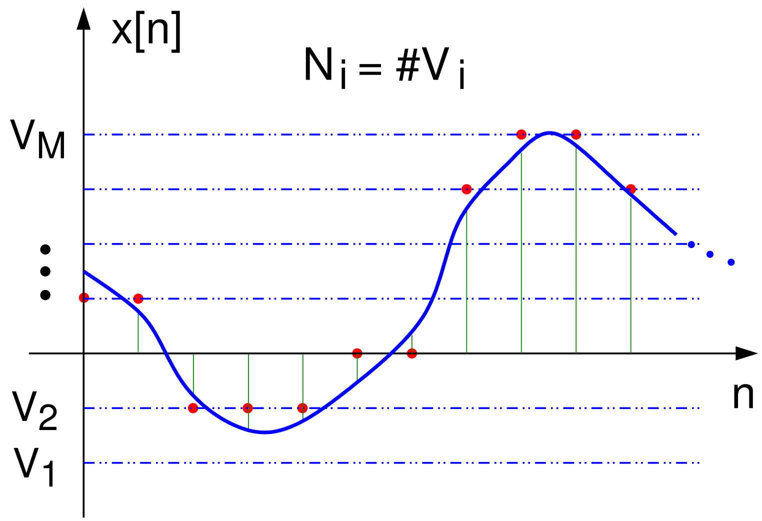
$$P(X = x_1) = 0$$

This appears odd, but otherwise the probabilities sum up to ∞

Appendix: Statistics vs. Probability

Statistical inference ↬ based on the observed data and supported by prob. theory

Vagaries of statistics: Consider N coarse-quantised data points, $x[0], \ldots, x[N-1]$. The quantised signal has $M \ll N$ possible amplitude values, $V_1, \ldots, V_M$, for which the corresponding relative frequencies are, $N_1 = \#V_1, \ldots, N_M = \#V_M$. Calculate the mean, $\bar{x}$.



Solution:

$$\bar{x} = \frac{1}{N}\sum_{n=0}^{N-1} x[n] \quad = \quad \frac{1}{N}\sum_{m=1}^{M} V_m N_m \quad = \quad \sum_{m=1}^{M} V_m \underbrace{\frac{N_m}{N}}_{\approx\, P(x=V_m)}$$

☞ Clearly, the factor $1/N$ does not imply "uniform distribution"

Statistical inference

Chinese for statistics is 统计 (summarizing & counting) and

probability is 概率(论) ((theory of) randomness & chances), ·

Probability: Assumes perfect knowledge about the "population" of random data (through the pdf).

Typical question: There are 100 books on a bookshelf, 40 with red cover, 30 with blue cover, and 20 with green cover. What is the probability to randomly draw a blue book from the shelf?

Statistics: No knowledge about the types of books on the shelf, we need to infer properties about the "population" based on random samples of "objects" on the shelf $\looparrowright$ **statistical inference**.

Typical question: A random sampling of 20 books from the bookshelf produced X red books, Y blue books and Z green books. What is the total proportion of red, blue, and green books on the shelf?

Statistical inference is applied in many different contexts under the names of: data analysis, data mining, machine learning, classification, pattern recognition, clustering, regression, classification

Notes:

○

Notes:

∘

Notes:

○