CR09 - HMM for time series classification and filtering

Introduction

Stéphane Derrode, stephane.derrode@ec-lyon.fr
http://perso.ec-lyon.fr/derrode.stephane/

École Centrale de Lyon, https://www.ec-lyon.fr/en
LIRIS Lab, https://liris.cnrs.fr/en

Let me introduce myself!

PhD in image processing
in Brest

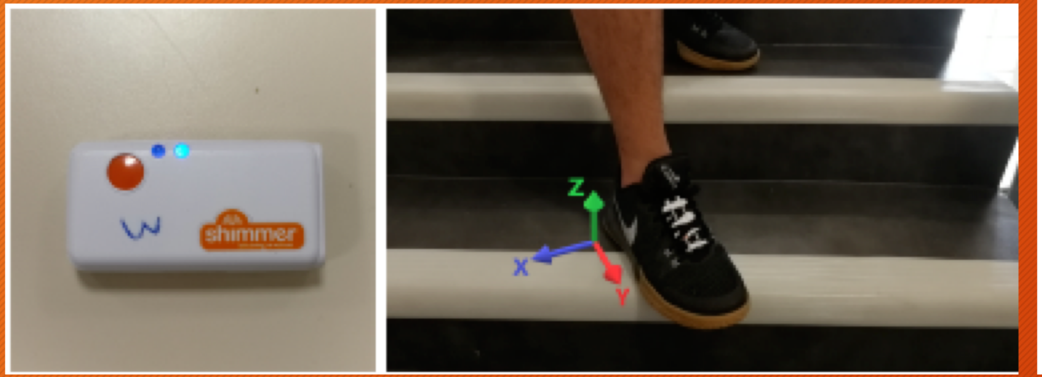
Master in telecommunication
in Lille

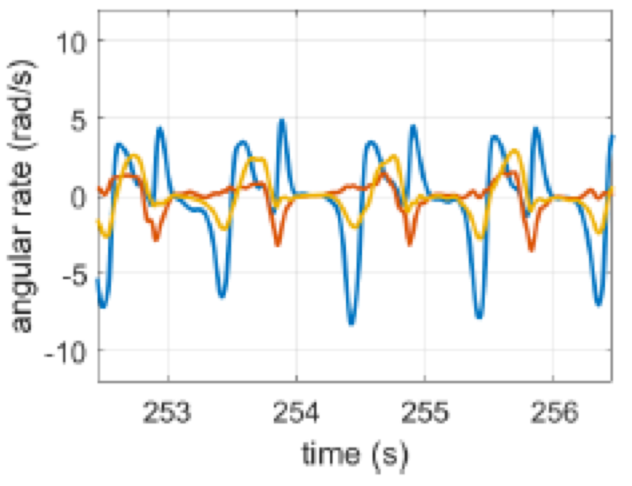
Full professor
in Computer Sciences
In Lyon

Assistant professor
in Computer Sciences
in Marseilles

Application : Human activity monitoring

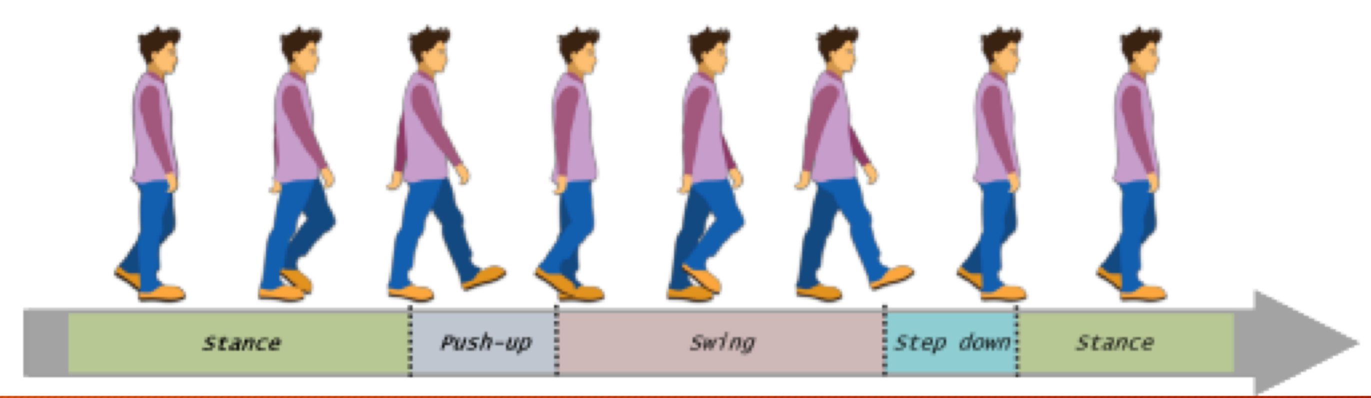
Inertial Measurment UInit

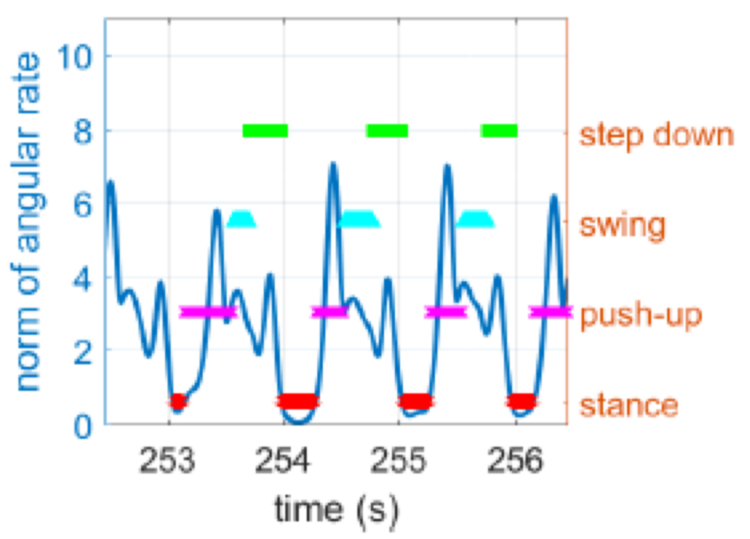




(d) Stair descent.

Gait cycle



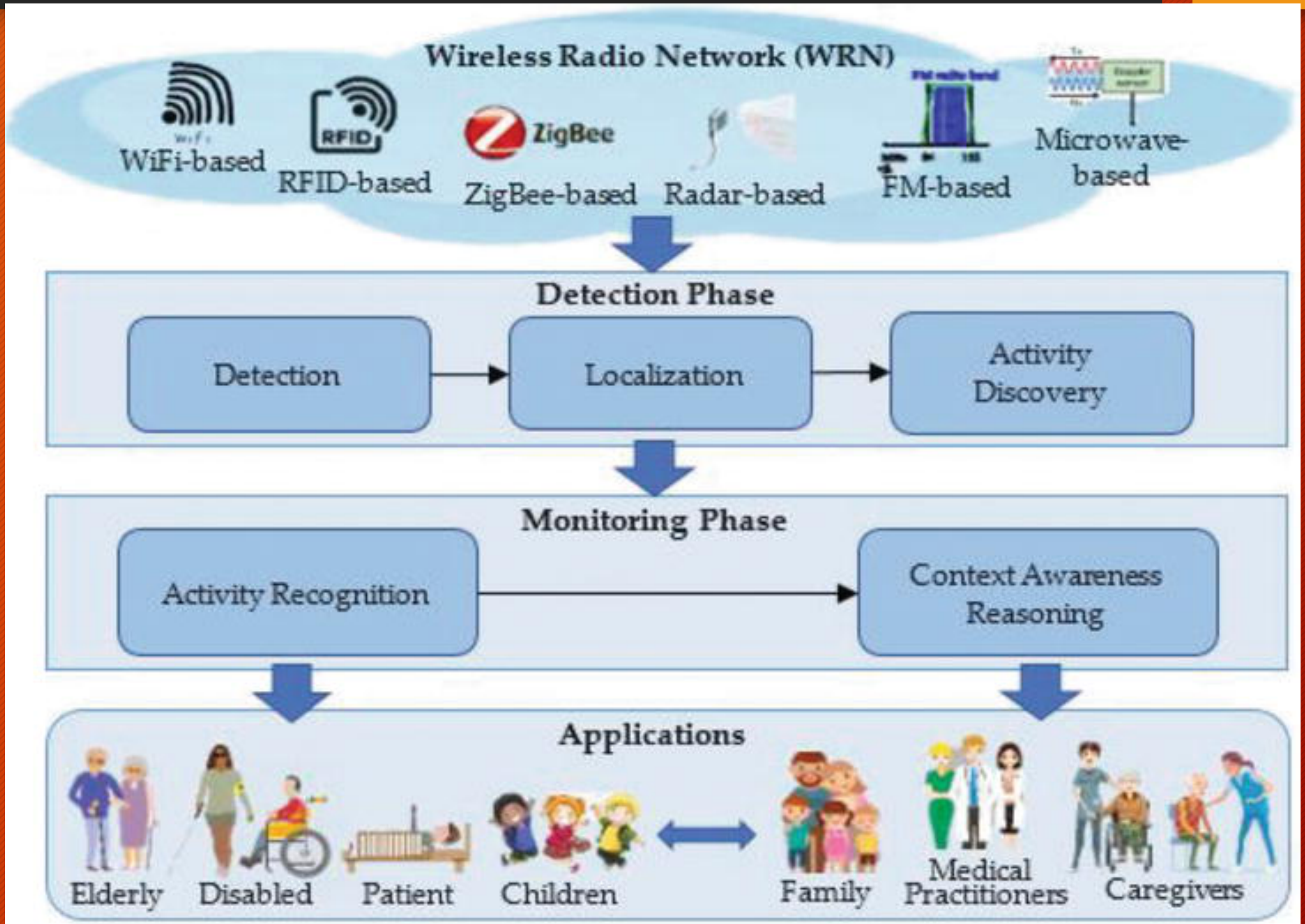


(d) Stair descent.

Application : Human activity monitoring using IoT network

Home automation / domotics.

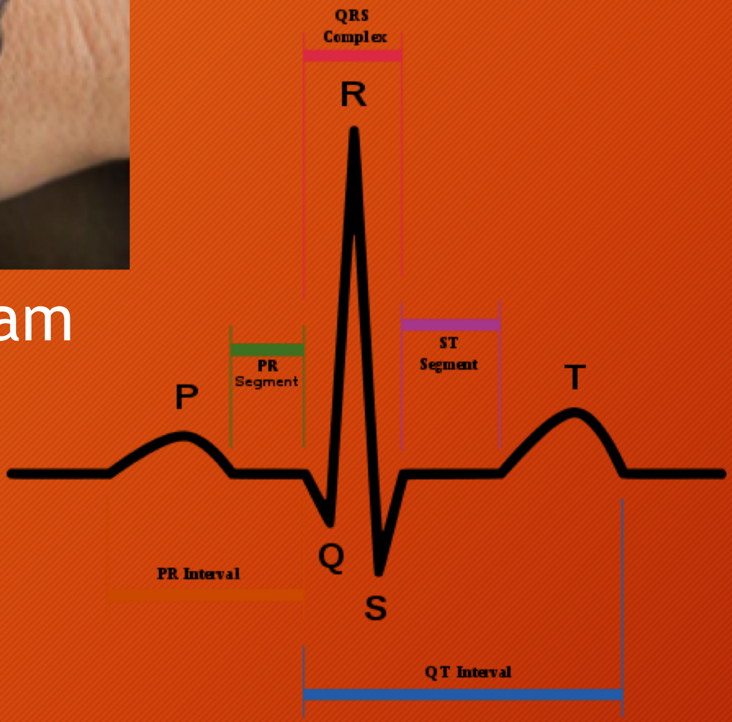
Example: Devlopment of surveillance system for the elderly using wireless beacon signal analysis, such as WIFI, Bluetooth, ZigBee sensors…

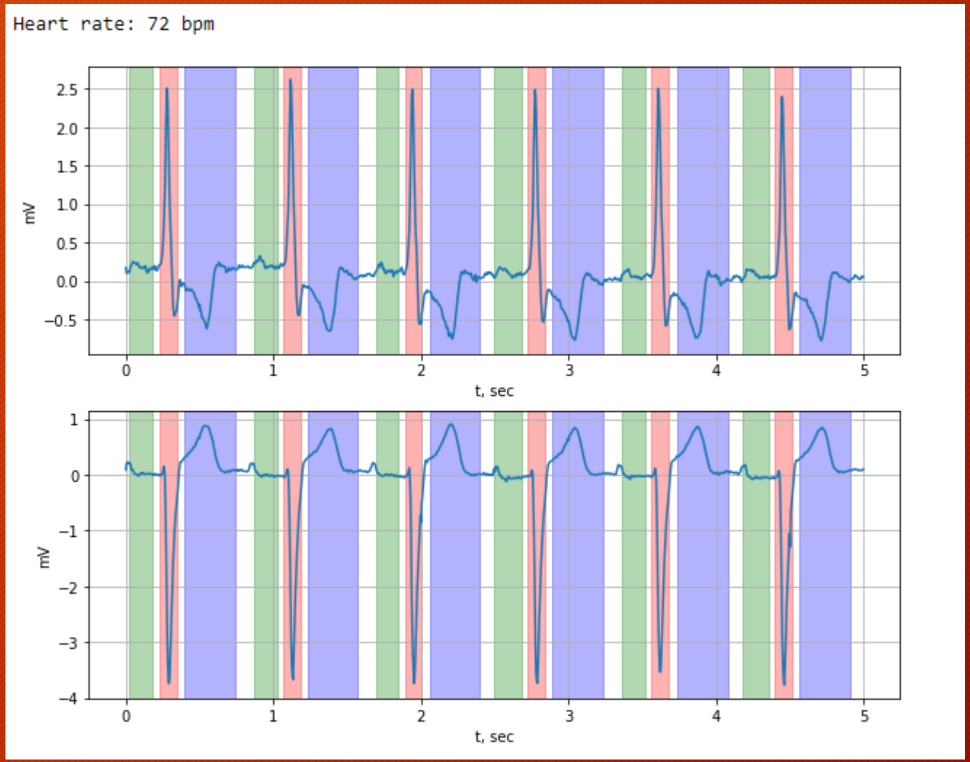


https://www.intechopen.com/books/intelligent-video-surveillance/device-free-localization-for-human-activity-monitoring

Markov models everywhere!

Electrocardiogram





ECG signal annotation

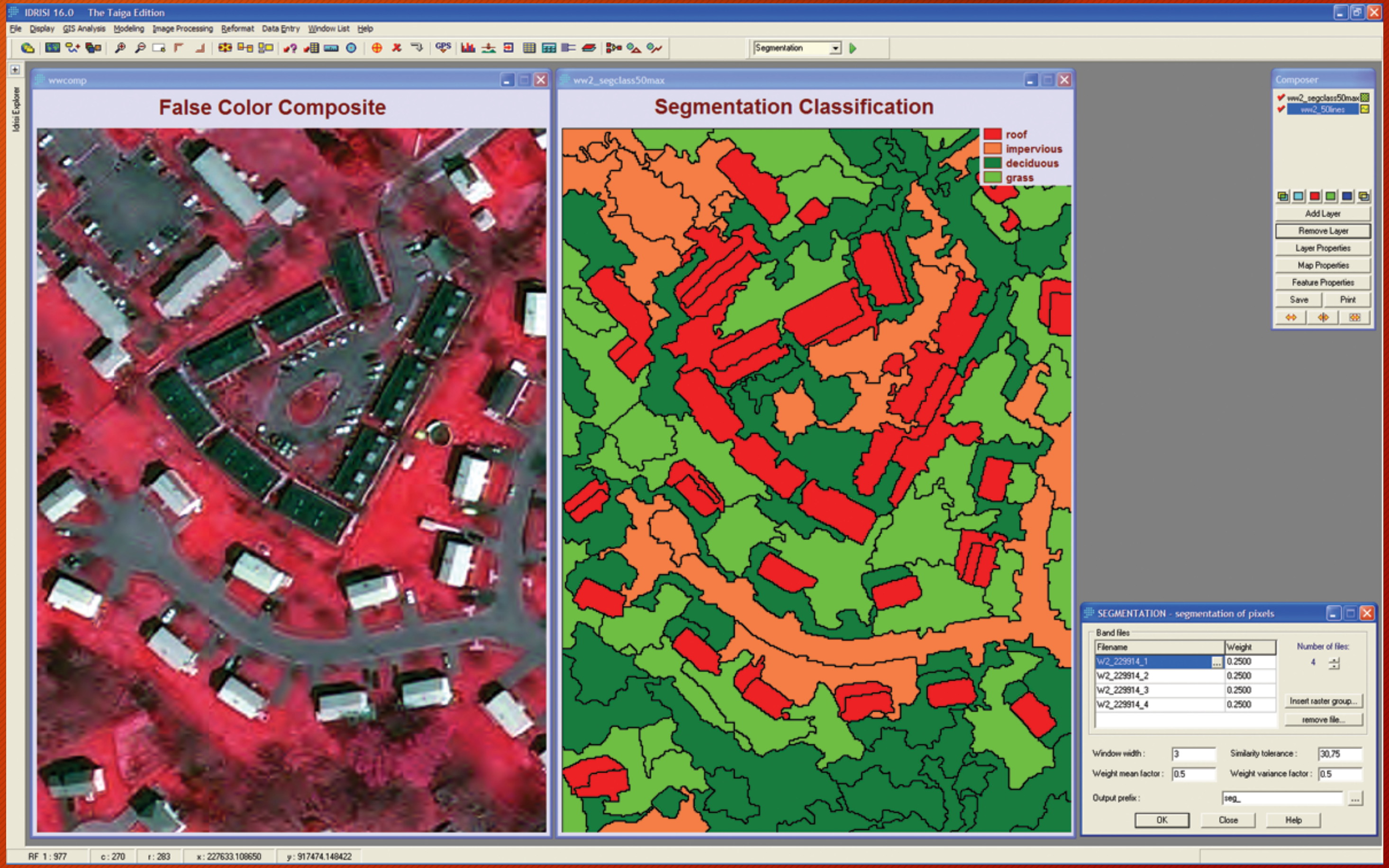
https://medium.com/data-analysis-center/annotating-ecg-signals-with-hidden-markov-model-56f8b9abd83a

Markov models everywhere!

Hidden Markov Random Field Model

Satellite image segmentation

Aerial photography for Woburn, Massachusetts in 2005.



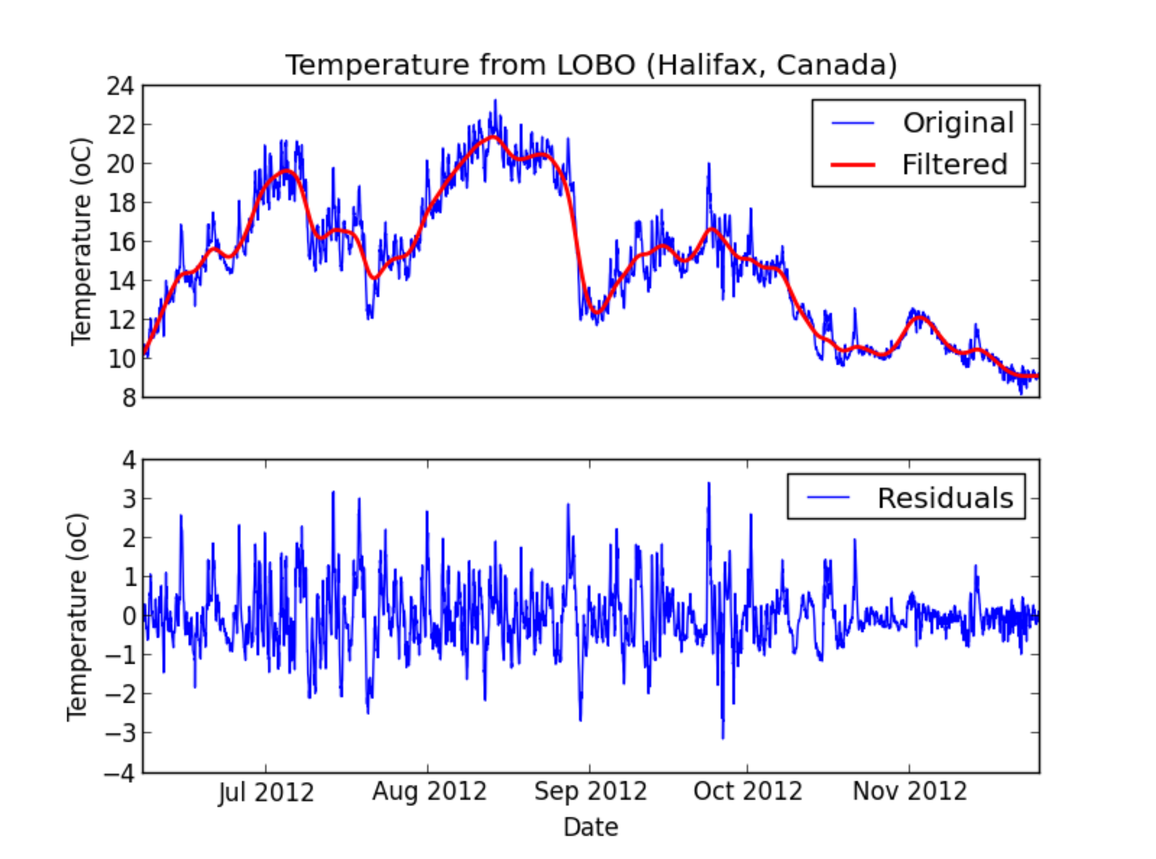
https://clarklabs.org/segmentation-and-segment-based-classification/

Markov models everywhere!

Kalman-like filtering

LOBO instrument (Land/Ocean Biogeochemical Observator)

Temperature data measured in the southwest edge of peninsular Halifax (Canada)



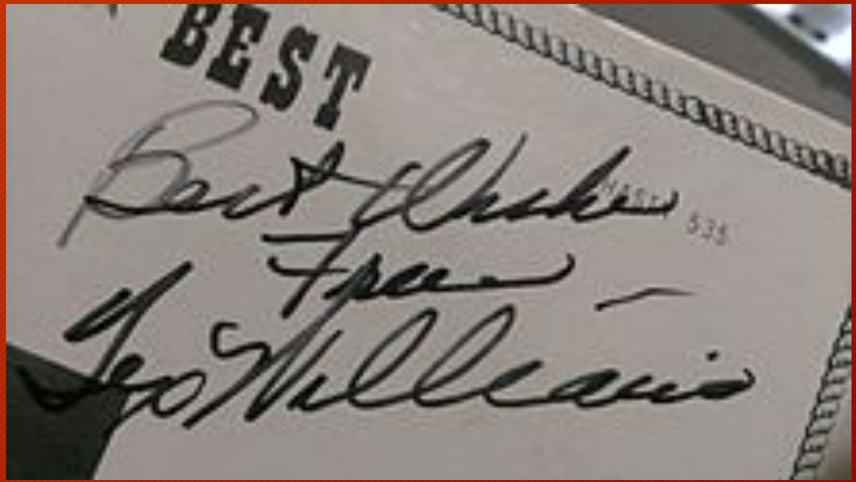
https://oceanpython.org/2013/03/11/signal-filtering-but

Markov chain models everywhere!

Also (wikipedia):

- Finance and econometry (stock exchange),
- Cryptanalysis (used to breach cryptographic security systems),
- Speech coding and synthesis, handwriting recognition,
- Biology : Gene prediction, bio-sequence alignment, DNA motif recovery…

Handwritten Text Recognition (HTR), is the ability of a computer to interpret intelligible handwritten input from sources such as paper documents, photographs, touch-screens and other devices.



https://en.wikipedia.org/wiki/Handwriting_recognition

Our stochastic point of view on TS analysis

- What we observe through an instrument or a sensor is degraded by many noises: Sensor noise (electronic), Transmission noise (coding error)…

- The noise is stochastic: it cannot be modelled by a deterministic equation. But its behaviour can be characterized statistically, through a probability law.

- Consequently, if the same experiment is repeated twice, the resulting time series will be different, ie samples will not be exactly the same.

Notations : the observations

- One observation emitted by a sensor (or whatever) will be denoted by the lower-case letter y.

- A series of observations of length N will be denoted by

$$\boldsymbol{y} = \boldsymbol{y}_1^N = \{y_1, y_2, \ldots, y_n, \ldots, y_N\}$$

- The series is modeled by a stochastic process with as many random variables as there are samples:

$$\boldsymbol{Y} = \boldsymbol{Y}_1^N = \{Y_1, Y_2, \ldots, Y_n, \ldots, Y_N\}$$

Each random variable Y_n is assumed to be real-valued (for the requirment of our course only) and characterized by a probability density function (pdf, e.g. Gaussian).

Notations : the states (or classes)

- The series of states / labels will be modeled by a stochastic process with as many random variables as there are samples:

$$\boldsymbol{X} = \boldsymbol{X}_1^N = \{X_1, X_2, \ldots, X_n, \ldots, X_N\}$$

  Each random variable X_n is assumed to be
  - Discrete-valued for data classification $X_n \in \Omega = \{1, \ldots, K\}$
  - Real-valued for data filtering

- A realization of the stocastic process will be denoted by

$$\boldsymbol{x} = \boldsymbol{x}_1^N = \{x_1, x_2, \ldots, x_n, \ldots, x_N\}$$

Preparation to upcoming lab and homework

- Install Python 3.x through the Anaconda distribution:
  https://www.anaconda.com/distribution/

- The distribution includes a text editor to write programs called « Spyder ».

- Beginners can follow Spyder tutorial from:
  https://www.youtube.com/watch?v=a1P_9fGrfnU

CR09 Lesson agenda

- **Session 1 (8h): Bayesian Decision (BD) theory and Mixture Model (MM)**
  - One 2h lab session with Python
  - Grading: a 1h sitting exam, no document (weight: 25%)
- **Session 2 (14h): Hidden Markov chain (and beyond)**
  - Two 2h lab session with Python
  - A 1h conference about a self-quantified application of HMM
  - Grading: a 2h sitting-exam, no document (weight : 50%)
- **Session 3 (8h): from Kalman filter to particle filter**
  - Grading : A 2h lab session with Python (weight : 25%)

Slides (completed and updated periodically), exercise and lab statements collected at: http://perso.ec-lyon.fr/derrode.stephane/Teaching.php

Classical textbooks

- Pattern recognition and Machine Learning, Christopher M. Bishop, Springer
- Bayesian smoothing and filtering, Simo Sarkka, Cambridge University Press, https://users.aalto.fi/~ssarkka/pub/cup_book_online_20131111.pdf

Keywords : Bayesian decision theory, mixture model, HMM, Kalman filtering, particle filtering