Advanced Research
Intelligent
Embedded
Systems

# Learning in non-stationary environments

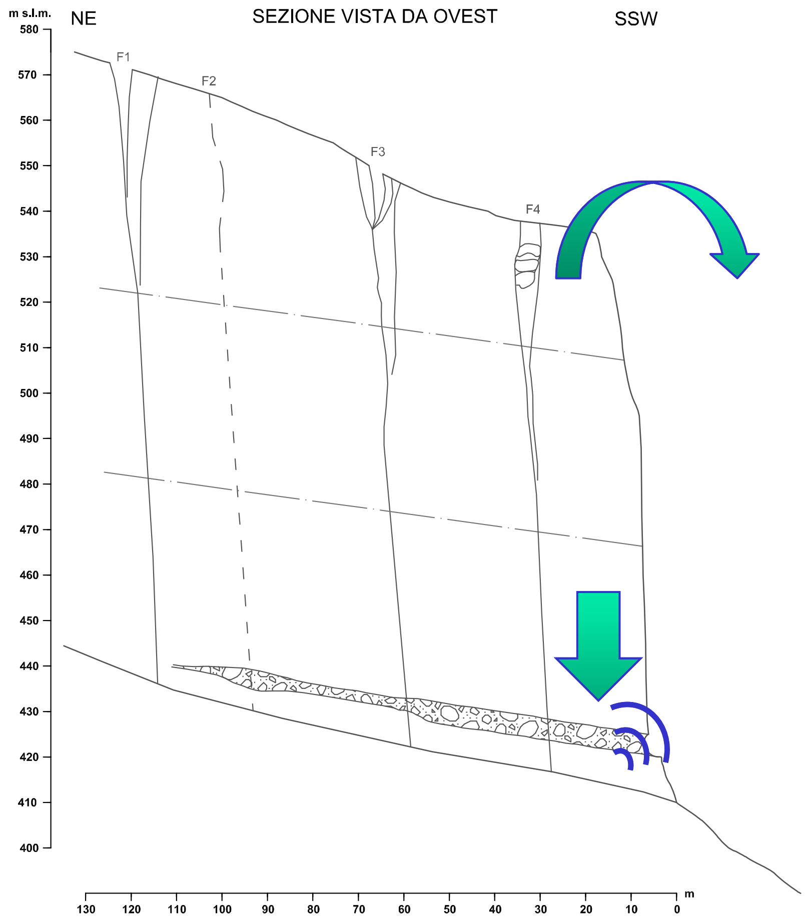
**Cesare Alippi**
Politecnico di Milano, DEIB, Italy

# Presentation Outline

- Why learning in a nonstationary environment?
- Active and passive approaches
  - Focus on active learning
- The Detect& React mechanism

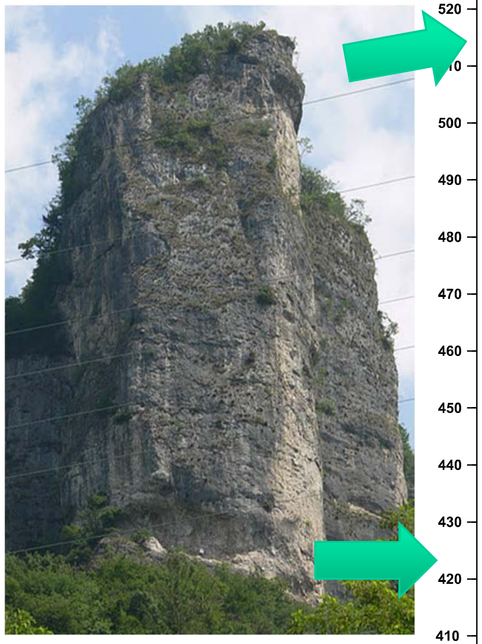
# An example: the Torrioni di Rialba (North Italy)

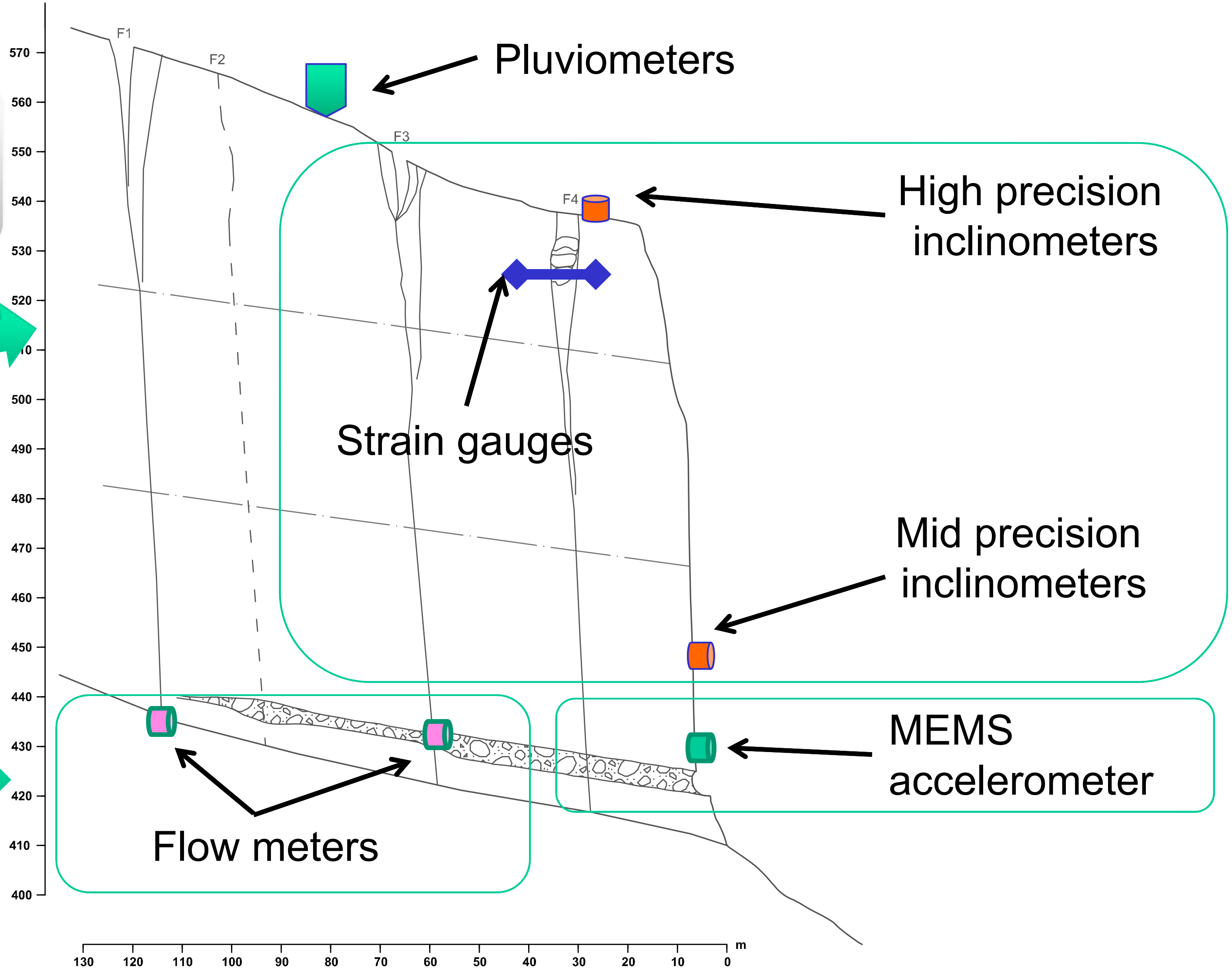
## Rock-toppling & collapse




# The Torrioni di Rialba Monitoring system

**In addition:**
Many temperature sensors





# Harsh conditions


The system needs to detect changes and adapt:

- Sensor Calibration
- Adaptive sampling
- Adaptive filtering
- Adaptive thresholds for event detection

# Stationarity and time invariance

- Stationarity
  - We say that a data generating process is stationary when generated data are i.i.d. realizations of a unique random variable whose distribution does not change with time

- Time invariance
  - We say that a process is time invariant when its outputs do not explicitely depend on time

$$y(t) = a_1(e^{t_0 - t})y(t-1) + a_2 y(t-2) + \eta, \eta = N(0, \sigma^2)$$

# The quest for adaptation

How

Active approach

Passive approach


Always (compulsive)

When needed (lazy)

When

# Passive learning in the traditional statistical learning framework

## Online (incremental) learning

$$V_N(\theta,\{(x_i,y_i)\}) = L(y_i, f(\theta,x_i))$$

$$\theta_{i+1} = \theta_i - \eta \frac{\partial L(y_i, f(\theta,x_i))}{\partial \theta}\Big|_{\theta_i}$$

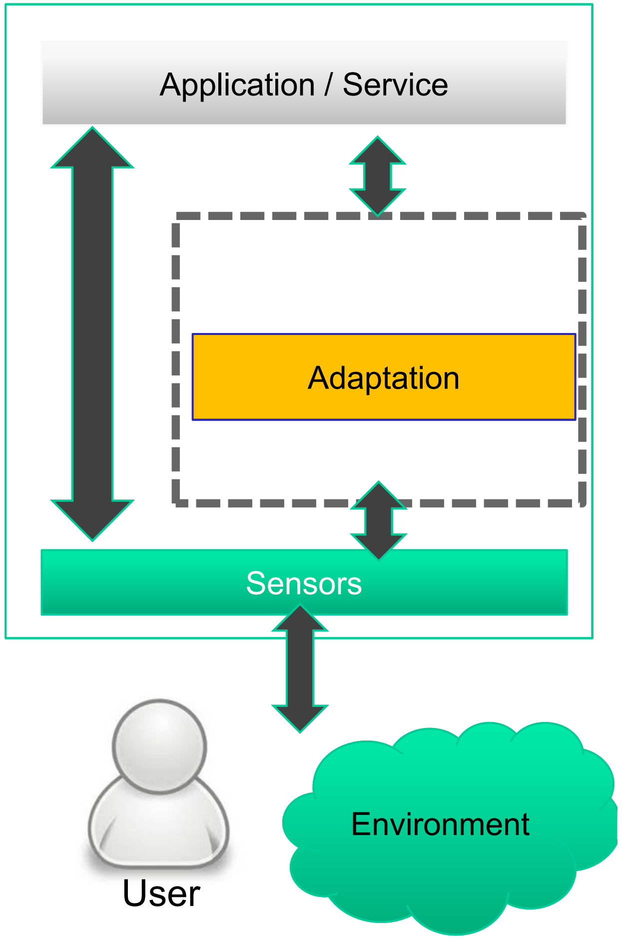
## Batch learning

$$Z_{n,i} = \{(x_i,y_i),(x_{i-1},y_{i-1}),\cdots(x_{i-n+1},y_{i-n+1})\}$$

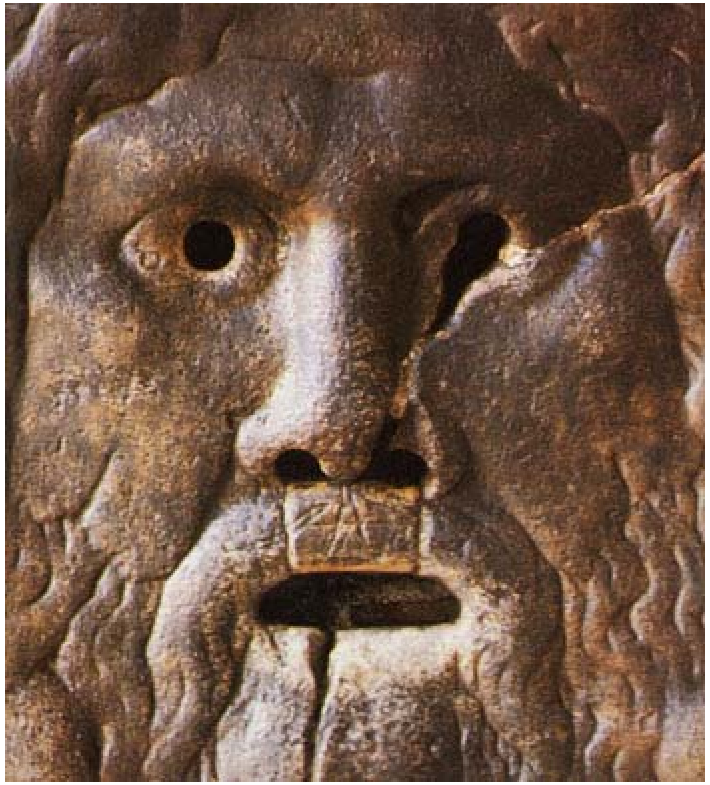
$$\theta_{i+1} = \theta_i - \eta \frac{\partial V_N(\theta,Z_{n,i})}{\partial \theta}\Big|_{\theta_i}$$

## Ensemble learning

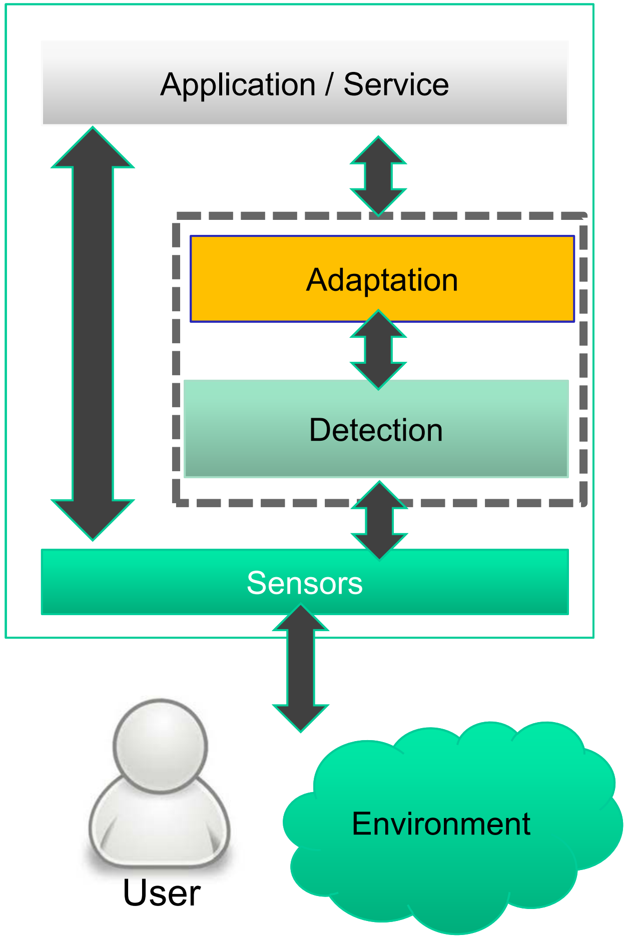
$$y(x) = \sum_{i=1}^{k} w_i M_i(x)$$



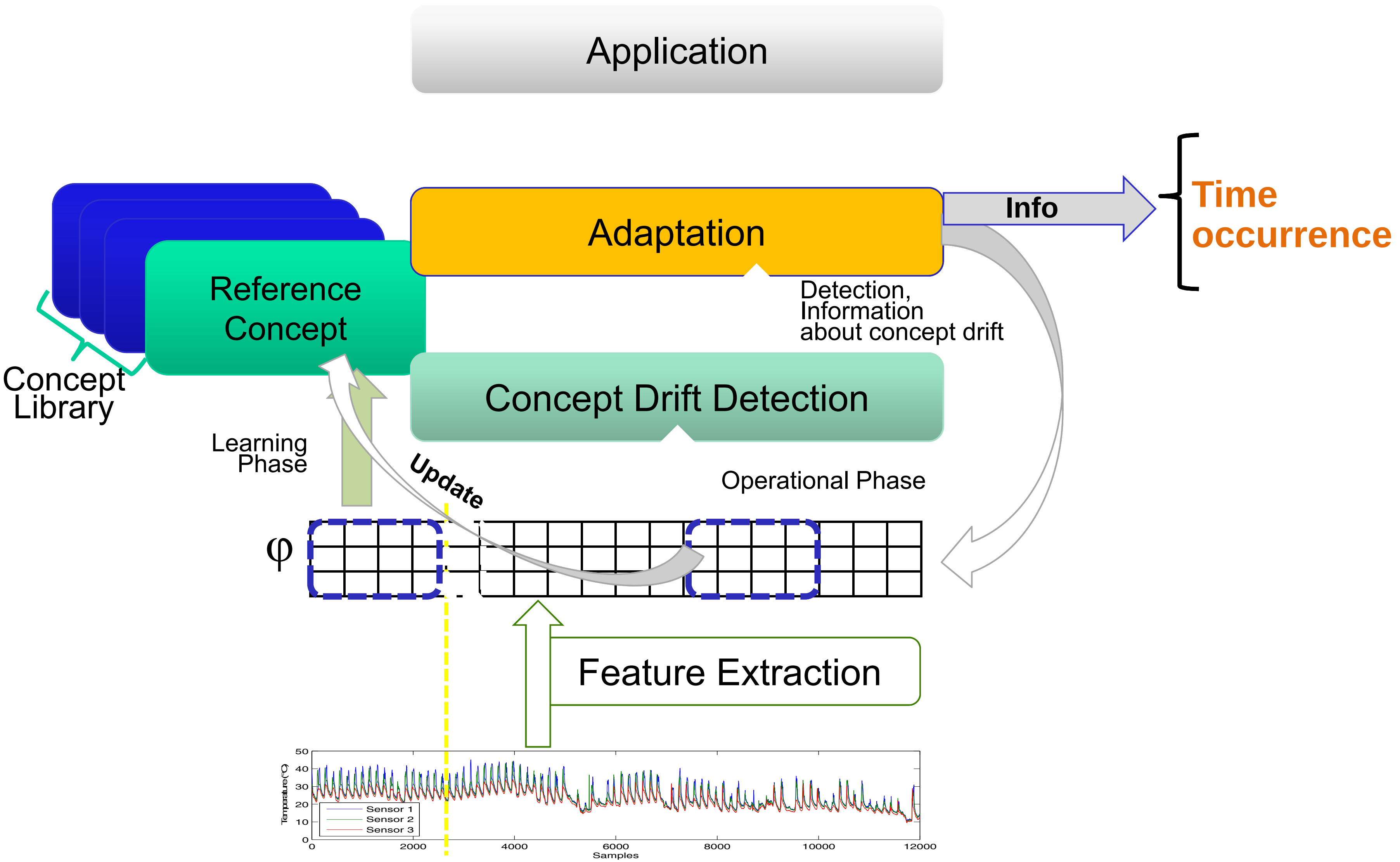
# Active learning



The Oracle provides information about an event, e.g., the occurrence of concept drift



# The active learning framework within an evolving environment



# Concept drift detection

**Ad hoc triggers designed to detect changes by inspecting sequences of data or derived features**

- **Change-point methods**

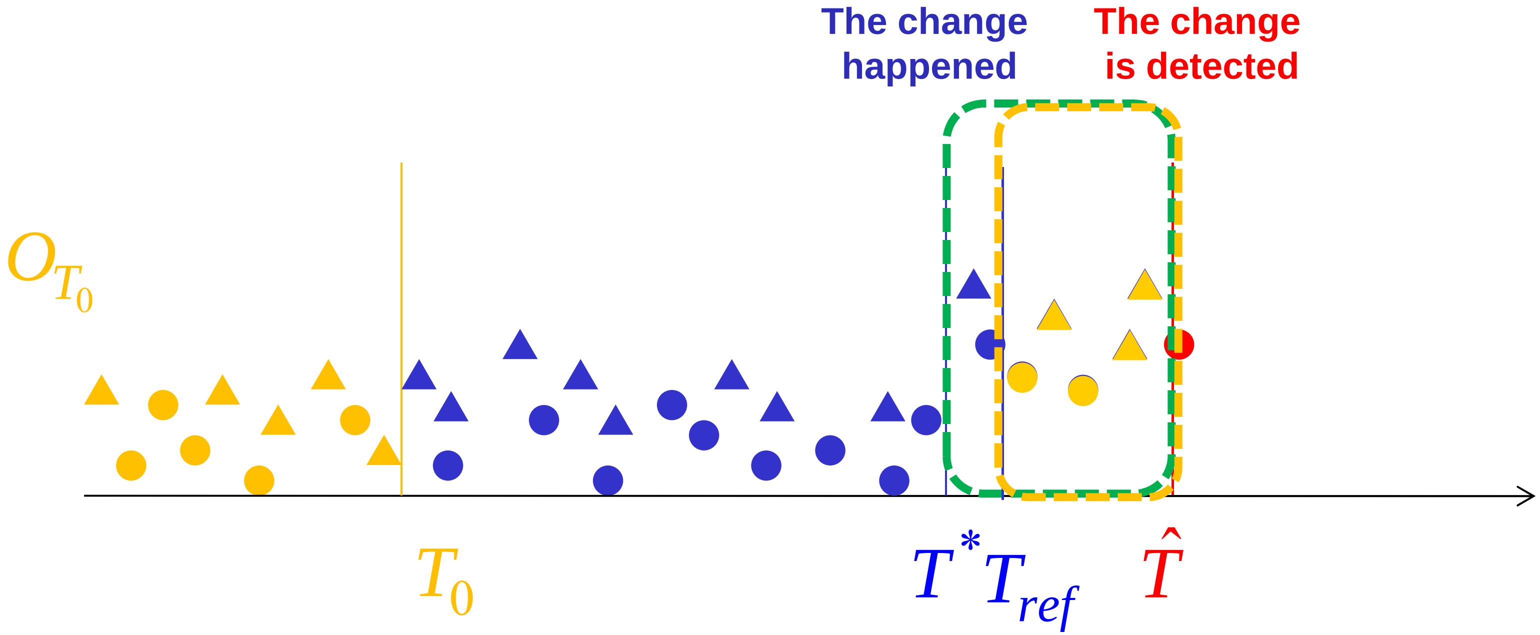
  Inspect a fixed sequence

- **Change detection tests** are designed for sequential use, e.g.,
  - ✓ CI-CUSUM test
  - ✓ ICI-based change detection test
  - ✓ Hierarchical change detection test

# Which data are consistent with the current status?

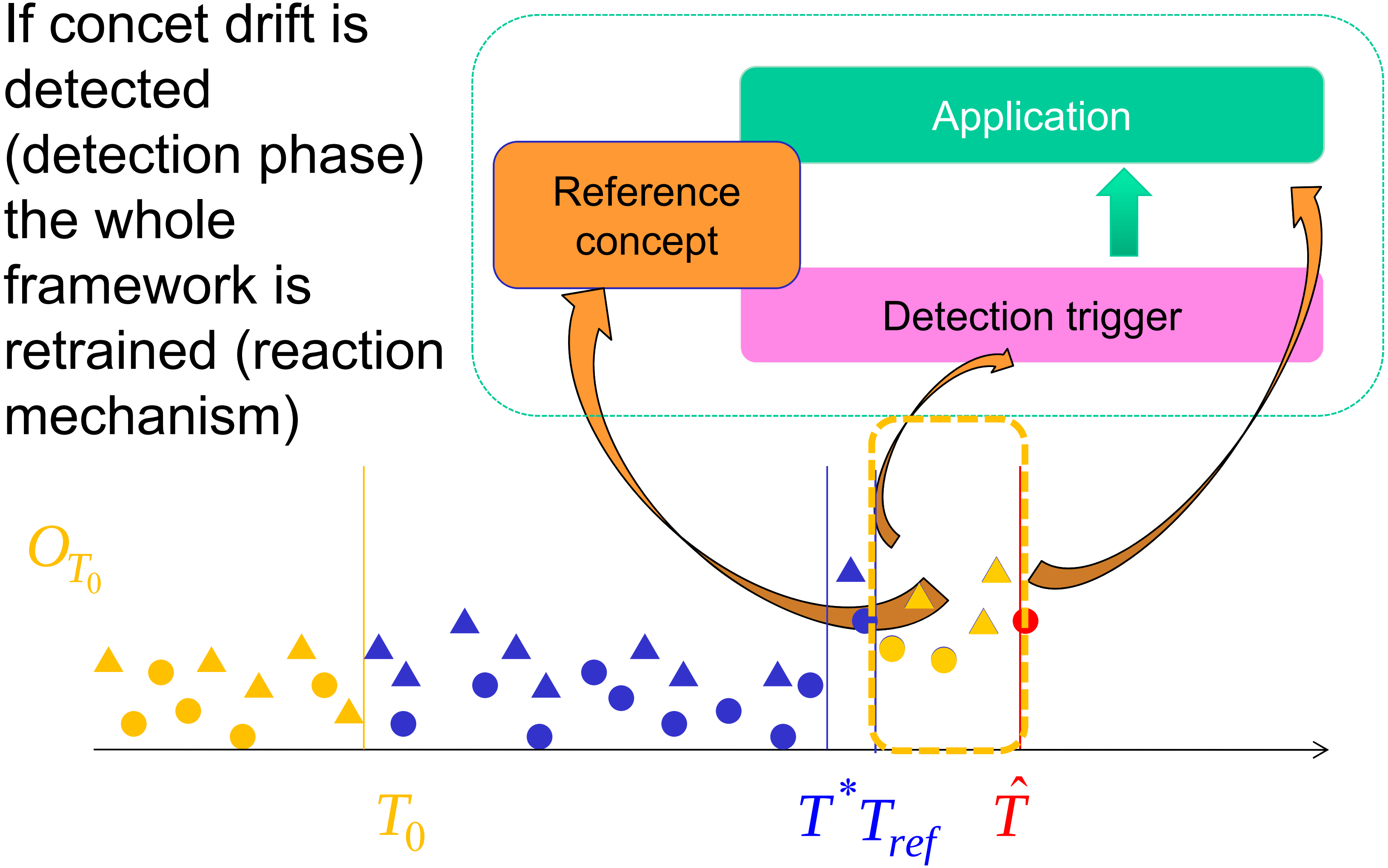
- Instances: between $T^*$ and $\hat{T}$

T* is unknown: use estimates $T_{ref}$ and $\hat{T}$



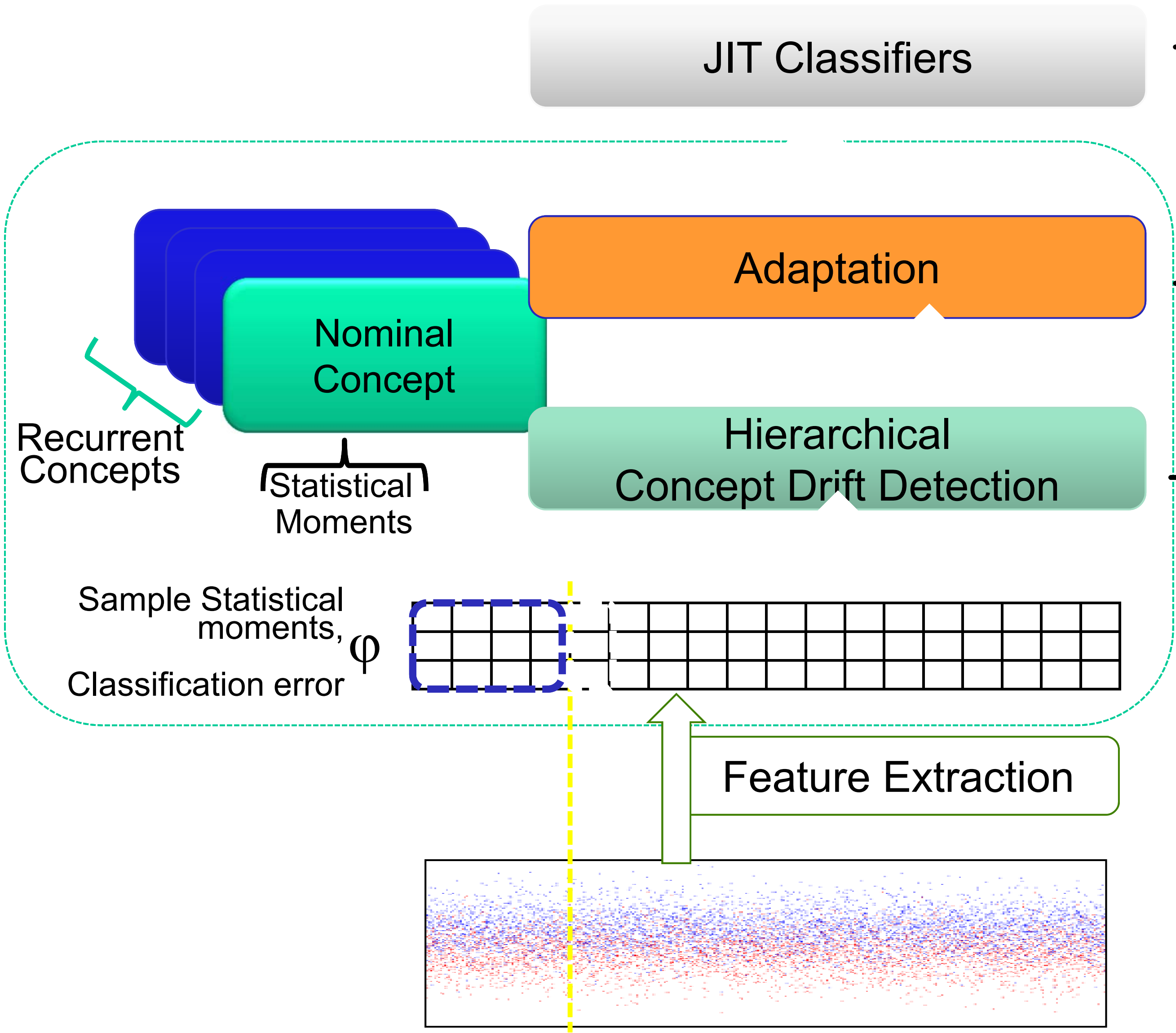
# The Detect&React approach

- If concet drift is detected (detection phase) the whole framework is retrained (reaction mechanism)



# An example:
# Just-in-Time Adaptive Classifiers

# Just-in-Time Adaptive Classifiers



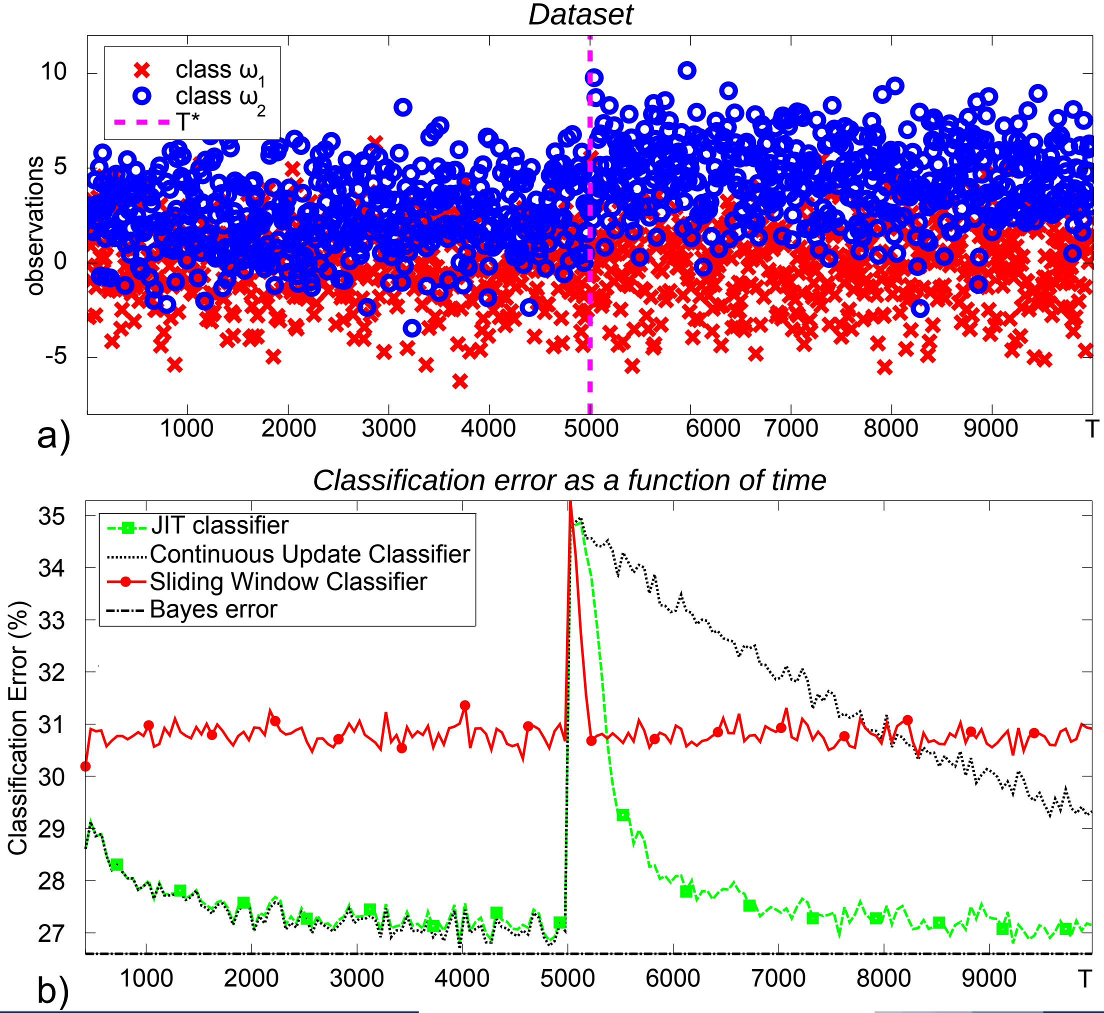

- K-NN
- SVMs
- Neural networks

- Dynamic knowledge base management
- Estimate of change time

- ICI-based CDT on the observations and the errors
- Hypothesis tests, Change-Point Methods

# Asymptotic optimality with JIT classifiers

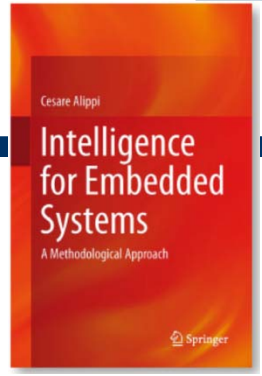
JIT adaptive classifiers grant asymptotic optimality when the process generating the data is affected by a sequence of abrupt concept drift



# Conclusions

- Concept drift occur, we cannot ignore their existence
- Most of time they are harmless (i.e., no fault) but the application has to react and udergo adaptation
- False positives occur in any detection method if data are affected by uncertainty characterized by a infinite support pdf. They have a computational cost
- In *Big Data* the quality of data is a main issue (a fault is a type of concept drift)

# Selected references




- **Monograph** C.Alippi, Intelligenge for Embedded Systems: a Methodological approach, Springer, pp.284, 2014

- **Change Detection Tests**
  - C.Alippi, G.Boracchi, M. Roveri, "Ensembles of Change-Point Methods to Estimate the Change Point in Residual Sequences", Soft Computing, Volume 17, Issue 11, pp 1971-1981, November 2013.
  - C. Alippi, S. Ntalampiras, and M. Roveri, “A cognitive fault diagnosis system for sensor networks,” IEEE Transactions on Neural Networks and Learning Systems, Vol.24, No.8., pp.1213-1226, August, 2013
  - C. Alippi, G. Boracchi, M. Roveri, A just-in-time adaptive classification system based on the intersection of confidence intervals rule, Neural Networks, Elsevier, vol. 24 , pp. 791-800, (2011)
  - C.Alippi, M.Roveri: Just-in-time Adaptive Classifiers. Part I. Detecting non-stationary Changes, IEEE-Transactions on Neural Networks, Vol. 19, No. 7, July 2008, pp.1145–1153
  - C. Alippi, G. Boracchi, M. Roveri, A Hierarchical, Nonparametric Sequential Change-Detection Test, in Proc. of IJCNN 2011, San Jose, USA Jul 31 - Aug 5, 2011.

- **JIT Adaptation**
  - C. Alippi, D. Liu, L. Bu, D. Zhao, "*Detecting and Reacting to Changes in Sensing Units: the Active Classifier Case*", IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 2014
  - C.Alippi. M.Roveri. F.Trovo’, *A Self-building and Cluster-based Cognitive Fault Diagnosis System for Sensor Networks*, IEEE Transactions on Neural Networks and Learning Systems, accepted for publication, 2014
  - C. Alippi, G. Boracchi, M. Roveri, (2012) "*Just In Time Classifiers for Recurrent Concepts*", IEEE Special issue on Learning in Nonstationary and Evolving Environments, IEEE Transactions on Neural Networks and Learning Systems, Vol.24, No.4., pp.620-634, April, 2013.
  - C.Alippi, G.Boracchi, G.Ditzler, R.Polikar, M.Roveri, Adaptive Classifiers for Nonstationary Environments, Contemporary Issues in Systems Science and Engineering, IEEE/Wiley Press, 2012
  - C.Alippi, M.Roveri: *Just-in-time Adaptive Classifiers. Part II. Designing the Classifier, IEEE-Transactions on Neural Networks.* IEEE Transactions on Neural Networks, Volume 19, Issue 12, December 2008, pp.2053 - 2064.