Change-point detection using Krylov subspace learning

Tsuyoshi Idé* and Koji Tsuda^

* IBM Research, Tokyo Research Lab.

^ Max Planck Institute for Biological Cybernetics

Summary

Fast algorithm for **PCA** when you want only the **inner product**

Apply this to a **PCA**-based **change-point detection** method

$\boldsymbol{\mu}^{\top}\boldsymbol{u}^{(i)}$

i -th **principal component**

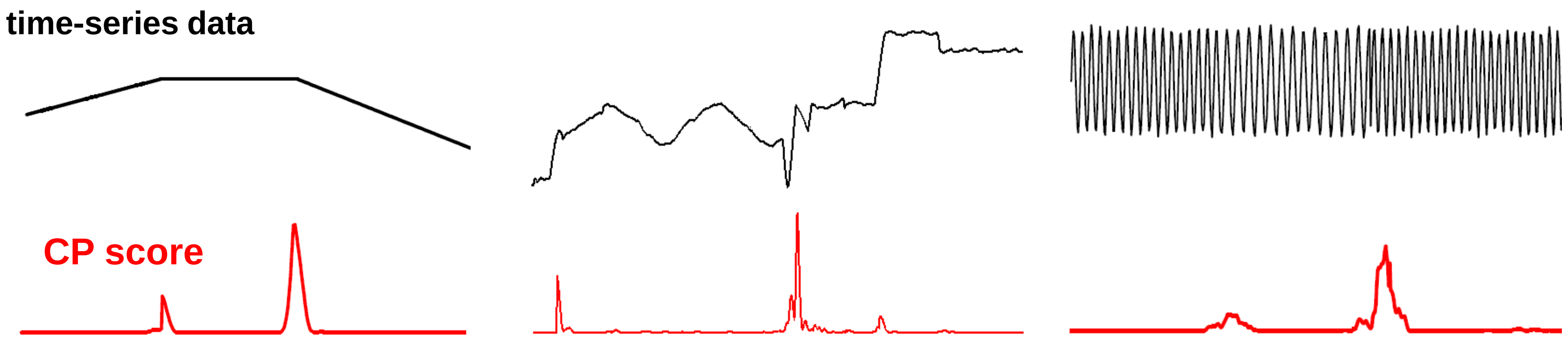
Given vector

Contents

- PCA-based approach to change detection
- Speeding up PCA for the inner product
- Experiment and summary

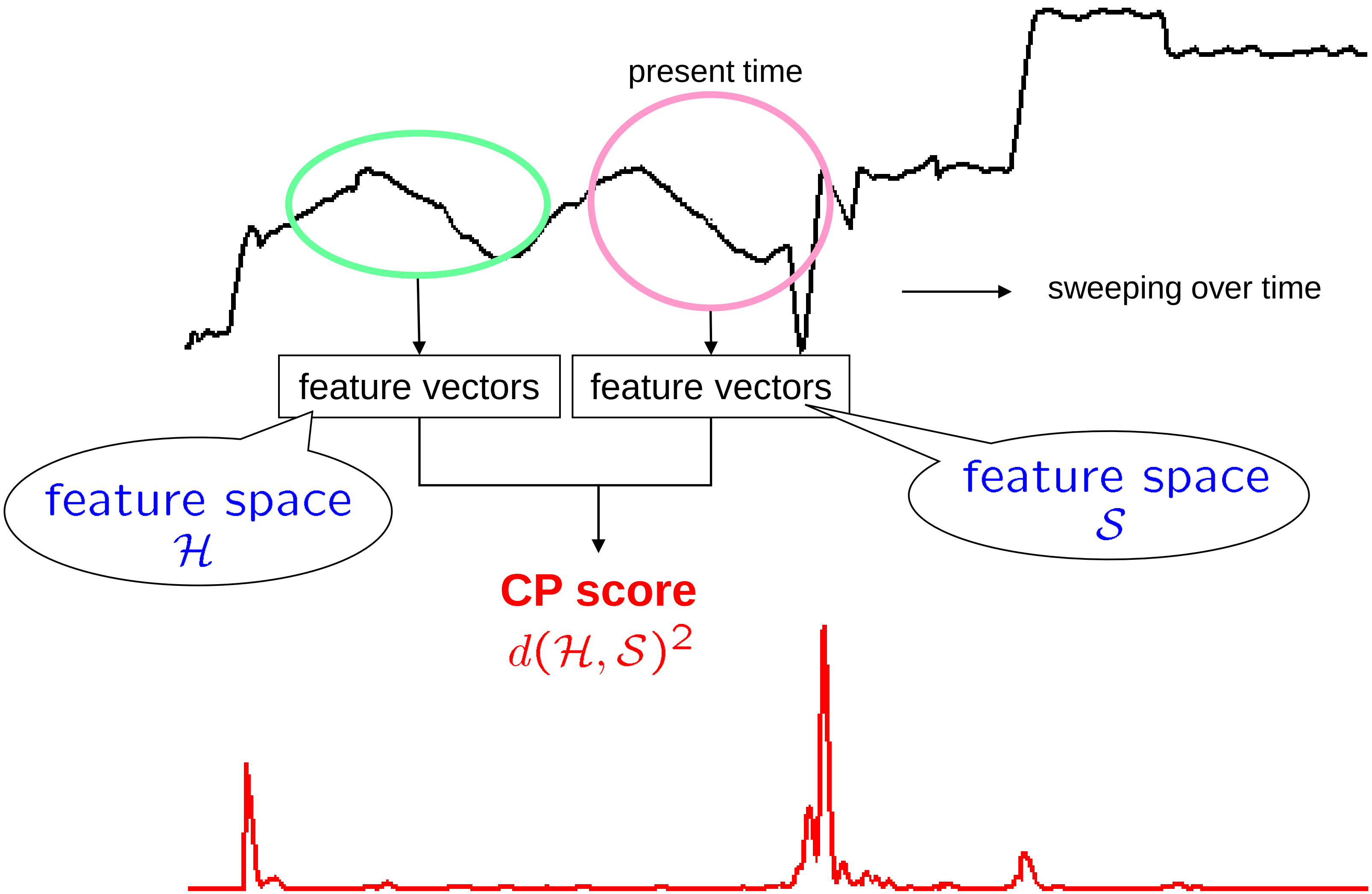
PCA-based approach to change detection

Change-point detection



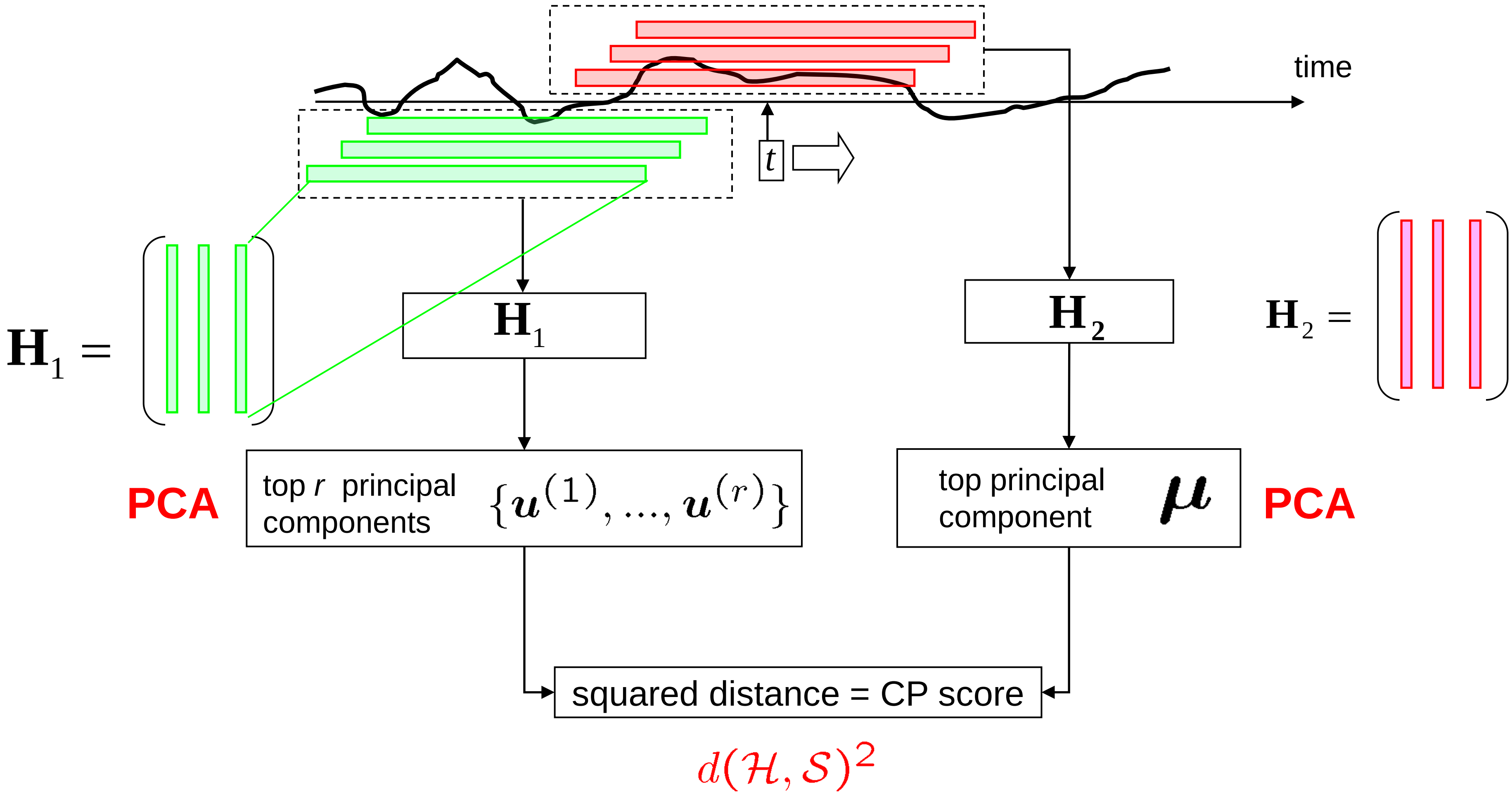
- **CP detection ≒ knowledge discovery**
- **Need to handle the variety of CPs**
  - “model-free” methods are preferable

Simple approach to change detection



PCA-based approach to CP detection: SST *

SST = singular spectrum transformation



Definition of subspace distance

$$d(\mathcal{H},\mathcal{S})^2 \equiv \min_{\boldsymbol{x}\in\mathcal{H},\ ||\boldsymbol{x}||=1} ||\mathsf{P}_{\mathcal{H}}\boldsymbol{x} - \mathsf{P}_{\mathcal{S}}\boldsymbol{x}||^2.$$

projection onto $\mathcal{H}$

projection onto $\mathcal{S}$

general properties

1. $d(\mathcal{H},\mathcal{S}) = d(\mathcal{S},\mathcal{H})$
2. $0 \leq d(\mathcal{H},\mathcal{S}) \leq 1$

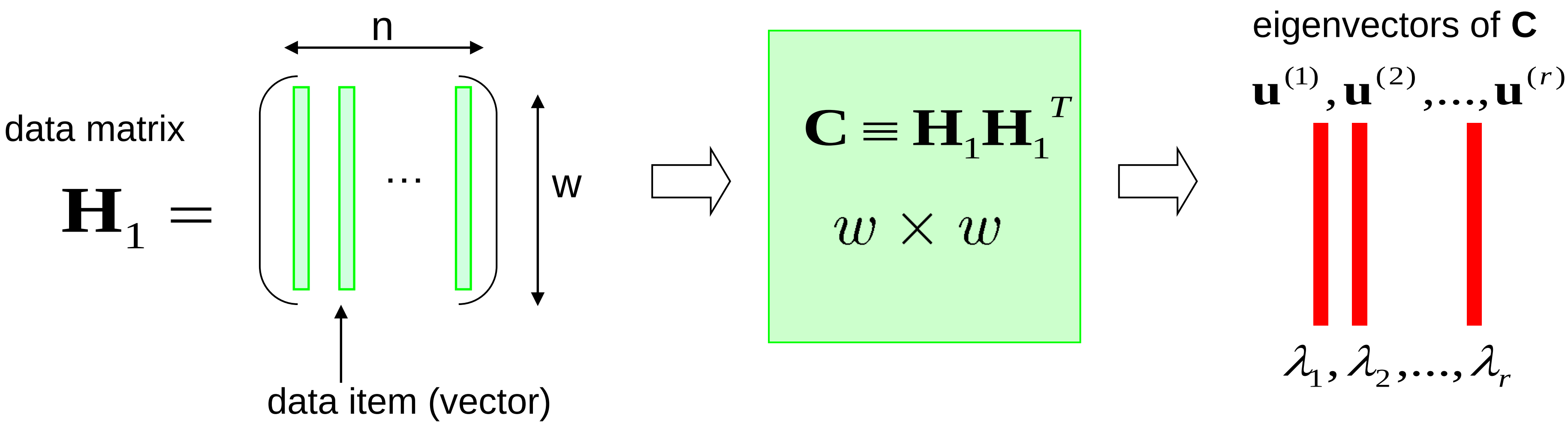
If we restrict to dim($\mathcal{S}$)=1, it reads

$$d(\mathcal{H},\mathcal{S})^2 = 1 - \sum_{i=1}^{r} \boldsymbol{\mu}^\top \boldsymbol{u}^{(i)}$$

where

$$\mathcal{S} = \mathrm{span}\{\boldsymbol{\mu}\} \quad \text{and} \quad \mathcal{H} = \mathrm{span}\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}, ..., \boldsymbol{u}^{(r)}\}$$

Principal component analysis (PCA) review



- **Eigenvectors**
  - the most popular directions among the column vectors in H_1
- **Eigenvalues**
  - the degree of popularity

SST is very slow due to repetitive PCA

- Most of the computation time is spent in finding $\{\boldsymbol{u}^{(2)}, \boldsymbol{u}^{(3)}, ..., \boldsymbol{u}^{(r)}\}$
- The top components $\boldsymbol{\mu}$ can be found efficiently
- Our problem is

> Compute the **inner product** $\boldsymbol{\mu}^\top \boldsymbol{u}^{(i)}$ for a given $\boldsymbol{\mu}$ at each t as fast as possible

C

$w \times w$

The size is moderate (~ O(100)).

But **dense** !

Related work

- ***Other fast EVD engines***
  - our algorithm is specially designed to compute the **inner product**
  - it is much faster than when a fastest general-purpose PCA engine is simply used
    - orthogonal iteration (power method)
    - EM-PCA [Roweis, NIPS 98]
- ***Online SVD algorithms***
  - extensively studied in information retrieval
    - “folding-in” and its variant
      - Zha-Simon, SIAM J. Sc. Com. 1999
  - unacceptable assumptions
    - document DB is **stable**
    - matrix is high-dimensional but ***sparse***
- ***Sampling-based approaches***
  - not useful unless the matrix is huge
    - Williams-Seeger, NIPS 00
      - Nyström method
    - Channubhotla-Jepson, NIPS 04
      - a renormalization technique
- ***Different from standard Krylov subspace methods?***
  - We point out that
    - the Krylov subspace is very useful in computing the inner products
    - and applicable to ***dense*** matrices if you want only the inner products
  - one common sense in numerical analysis:
    - Krylov subspace methods are unstable for dense matrices

Speeding up PCA for the inner product

A trick for computing the inner product

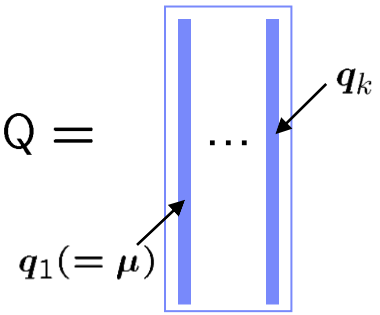
Suppose we have an **orthonormal** set $\{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_k\}$ satisfying $\boldsymbol{q}_1 = \boldsymbol{\mu}$, to approximate $\boldsymbol{u}^{(i)}$ as

$$\boldsymbol{u}^{(i)} \approx \sum_{\alpha=1}^{k} x_\alpha^{(i)} \boldsymbol{q}_\alpha$$

matrix compression

$\boldsymbol{x}^{(i)} = i$-th eigenvector of $\mathsf{Q}^\top \mathsf{C} \mathsf{Q}$

$\mathsf{Q} =$

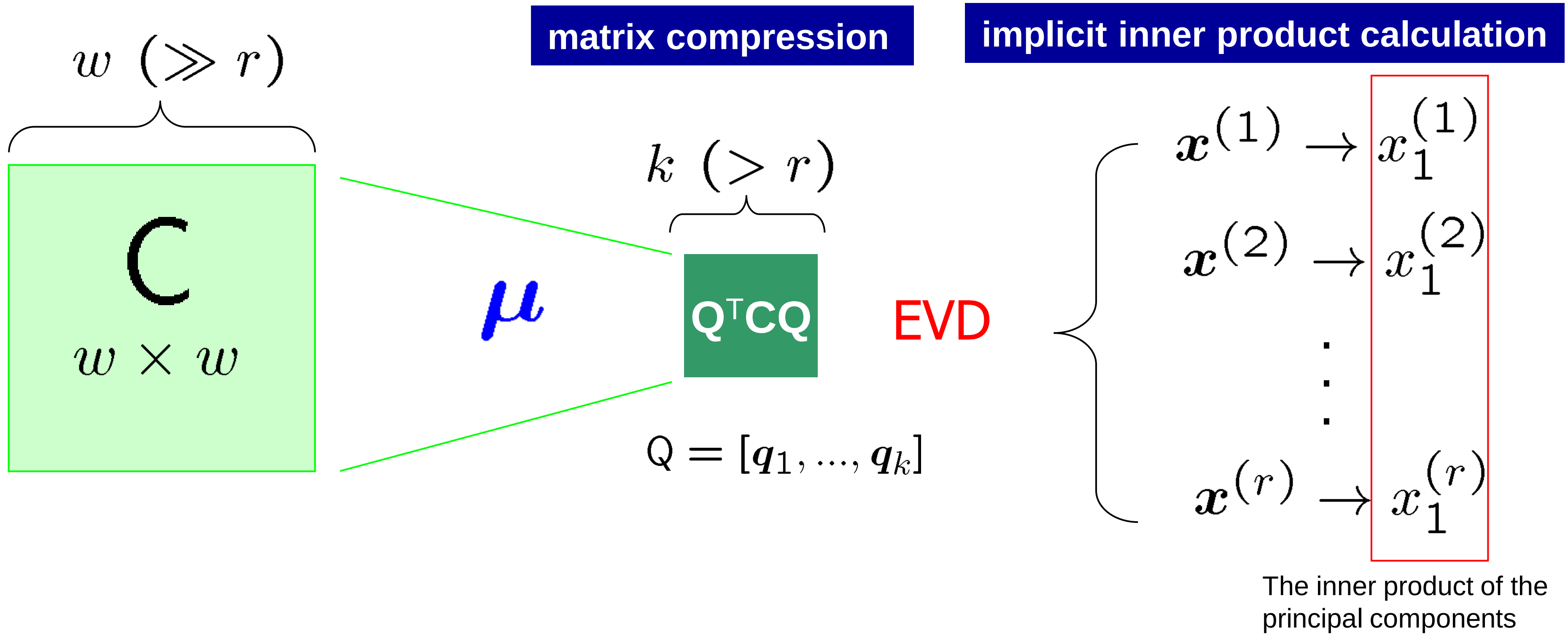


implicit inner product calculation

$$\boldsymbol{\mu}^\top \boldsymbol{u}^{(i)} \approx \sum_{\alpha=1}^{k} x_\alpha^{(i)} \boldsymbol{\mu}^\top \boldsymbol{q}_\alpha = x_1^{(i)}$$

Find the $k \times k$ matrix $\mathsf{Q}^\top \mathsf{C} \mathsf{Q}$, and perform EVD.

A trick for computing the inner product



Find the $k \times k$ matrix $Q^T CQ$, and perform EVD.

How to find the orthonormal vectors $\{\boldsymbol{q}_\alpha\}$

- $\boldsymbol{q}_1 = \boldsymbol{\mu}$
- Large overlap with the principal subspace

Solution: **Perform Gram-Schmidt orthgonalization of** $\{\boldsymbol{\mu}, \mathsf{C}\boldsymbol{\mu}, ..., \mathsf{C}^{k-1}\boldsymbol{\mu}\}$ **to get** $\boldsymbol{q}_1(=\boldsymbol{\mu}), \boldsymbol{q}_2, .., \boldsymbol{q}_k$

known as *Krylov subspace*

Intuition

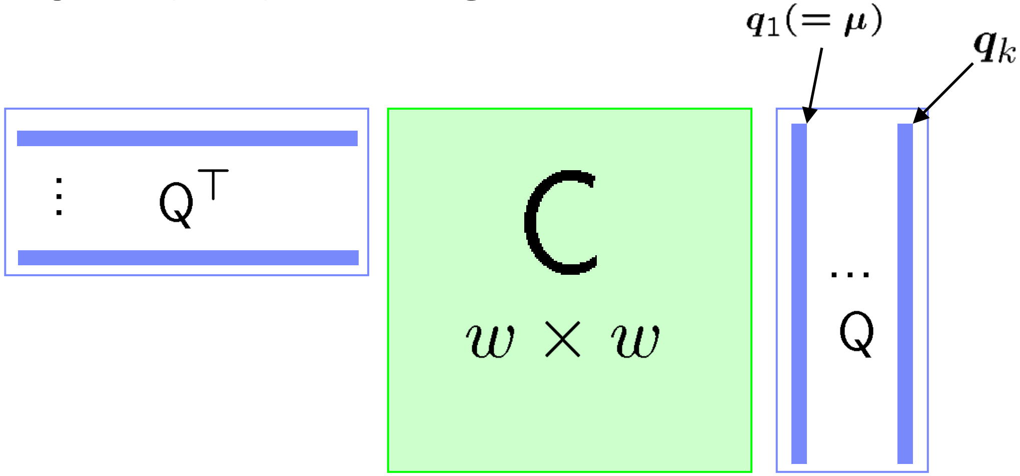
$$\mathsf{C}\boldsymbol{\mu} = \sum_{l=1}^{w} \lambda_l \boldsymbol{u}^{(l)} \boldsymbol{u}^{(l)\top} \boldsymbol{\mu}, \qquad \mathsf{C}^2\boldsymbol{\mu} = \sum_{l=1}^{w} \lambda_l^{\,2} \boldsymbol{u}^{(l)} \boldsymbol{u}^{(l)\top} \boldsymbol{\mu}, \quad ...$$

spectral expansion of **C**

multiplying by C ⇒ more focus on larger eigenvectors, or **principal components**

Useful properties of Krylov subspace

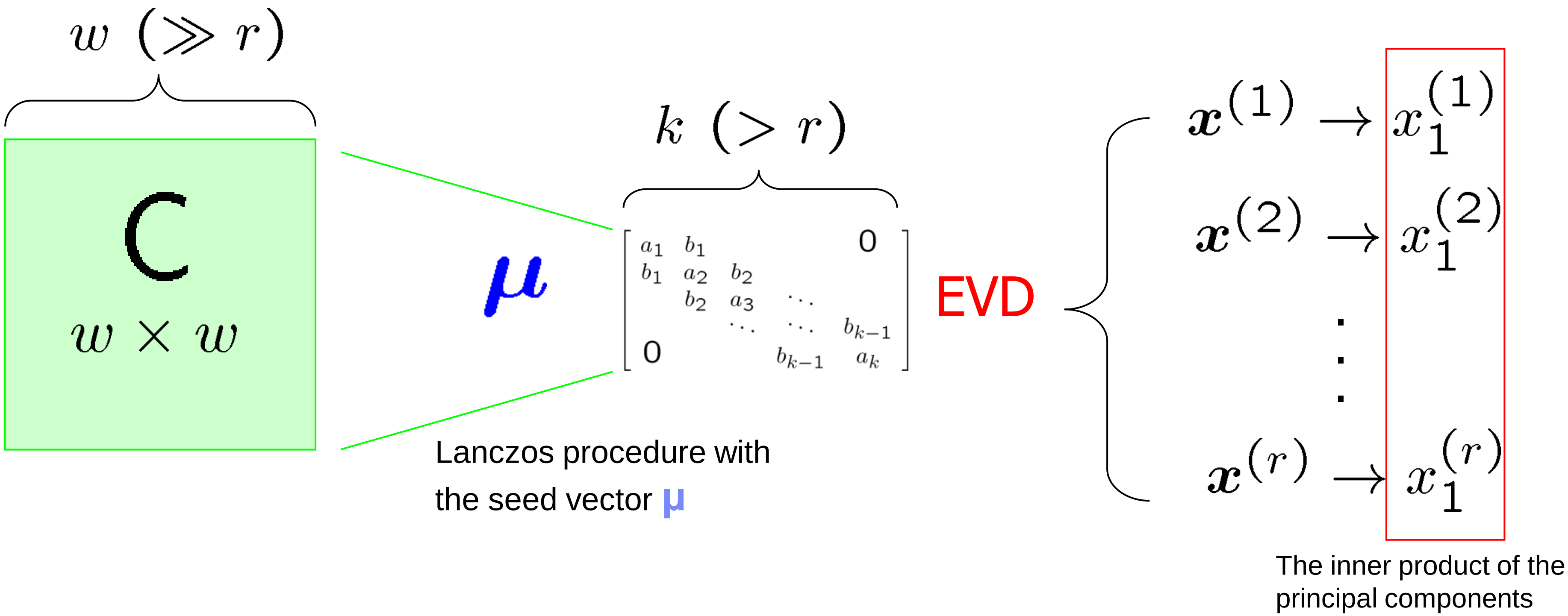
- **Property 1: Q^TCQ is *tridiagonal***



$$= \begin{bmatrix} a_1 & b_1 & & & 0 \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{k-1} \\ 0 & & & b_{k-1} & a_k \end{bmatrix}$$

- **Property 2: There exists an algorithm that finds $\{a_\alpha\}$ and $\{b_\beta\}$ directly from C**
  - i.e., you can skip to find **Q**
  - the Lanczos procedure [see, Golub-van Loan, Chap.9]

Final algorithm: implicit Krylov approximation

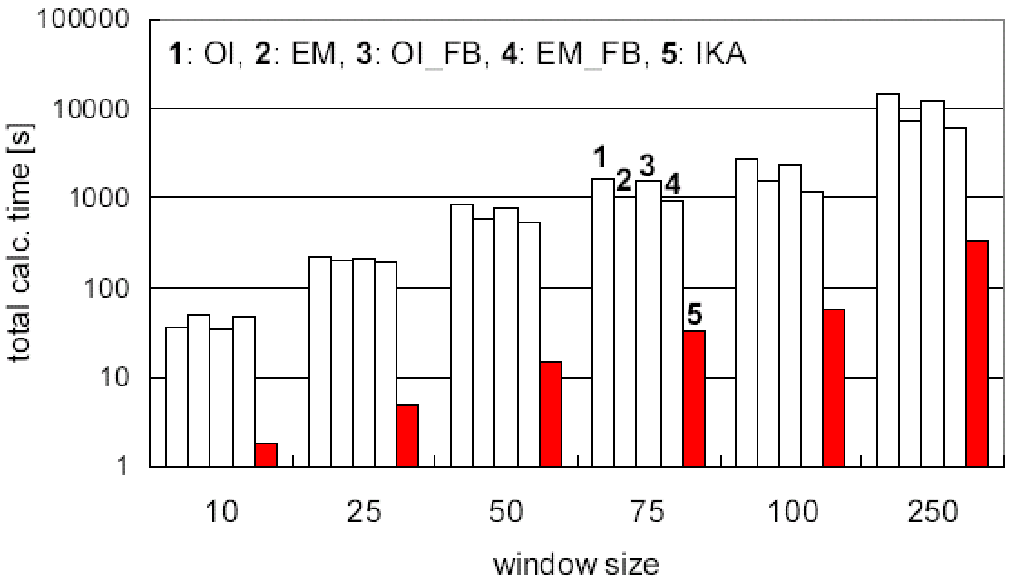


- Typical size: w ~100, r ~ 3, k ~ 5
- EVD of tridiagonalized matrices can be done extremely efficiently using QR iteration

Experiment and summary

50 times faster than standard iterative methods

- **CP detection task for the "phone1" data**
  - available in the UCR archive
- **Compared to standard iterative methods**
  - complexity = (# of time points) $\times$ $O(rw^2)$

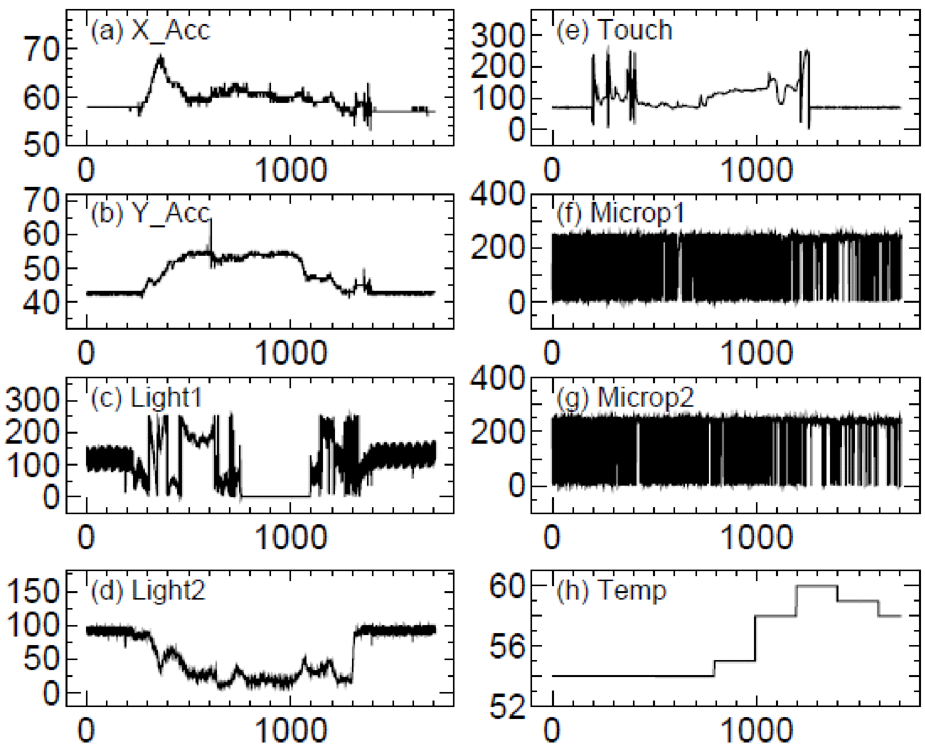


Methods compared

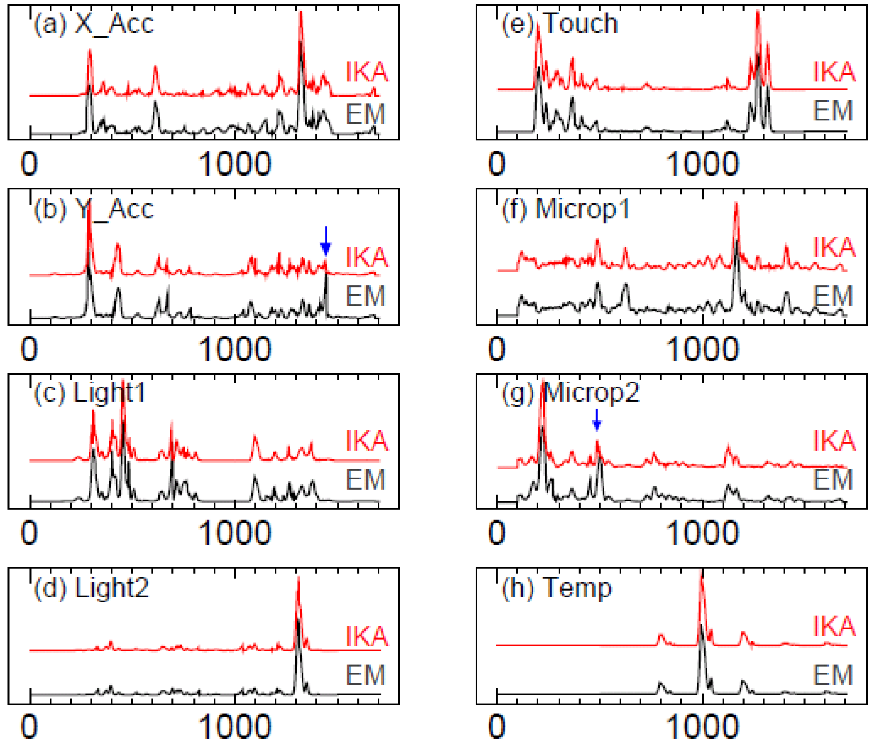
OI	orthogonal iteration
EM	EM-PCA [Roweis, 1998]
OI_FB	OI with feedback
EM_FB	EM with feedback
IKA	Implicit Krylov aprxn. (our method)

Numerical errors exist, but small

raw signal



change-point score



Summary

- **Presented an approximated algorithm to compute the inner product of the principal components**

- **Showed Krylov subspace learning works well for *dense* matrices when finding the inner product**

- **Applied our algorithm to a PCA-based CP detection method (SST) to get dramatic seed up**

- **Future work: apply this technique to other areas**
  - will be useful in computing Markov transition probabilities

Thank you !!

Other CP detection methods (1/2)

- **Single Gaussian model**
  - detects the change in mean and variance
  - implicitly assumes step-like changes
    - See, e.g., M.Basseville and I. V. Nikiforov, “Detection of Abrupt Changes”, Prentice-Hall, 1993.

- **Wavelet transform**
  - can be viewed as generalized derivative
  - practical difficulty in choosing the base function
    - See, e.g., S. Mallat and W-. L. Hwang, "Singularity Detection And Processing With Wavelets", IEEE Trans. Information Theory, 38 (1992) 617-643.

time series data

cumulative summation

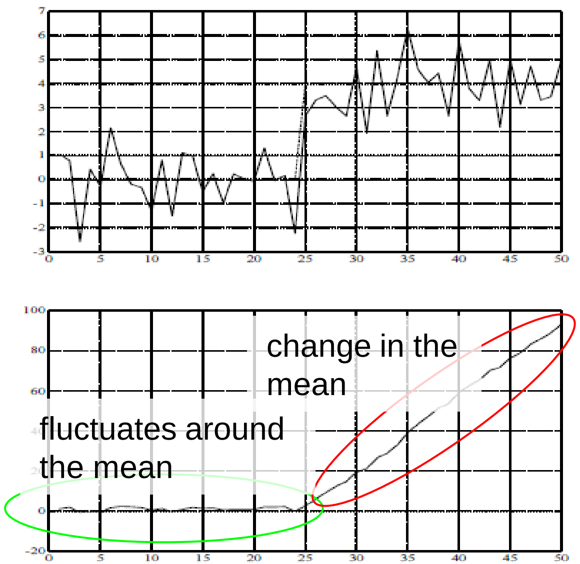
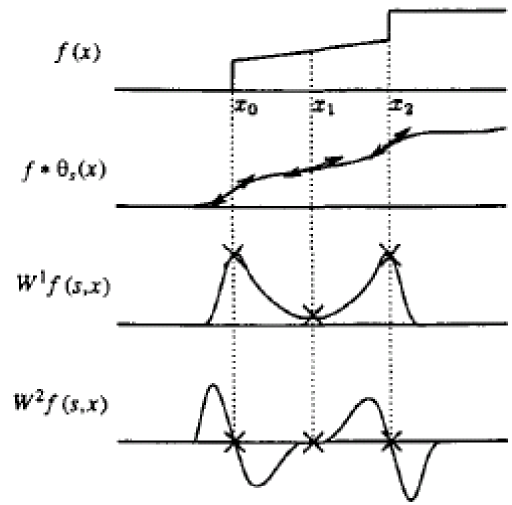


Figure 1.1 Increase in mean with constant variance and the typical behavior of the decision function in quality control.



time series data

smoothing

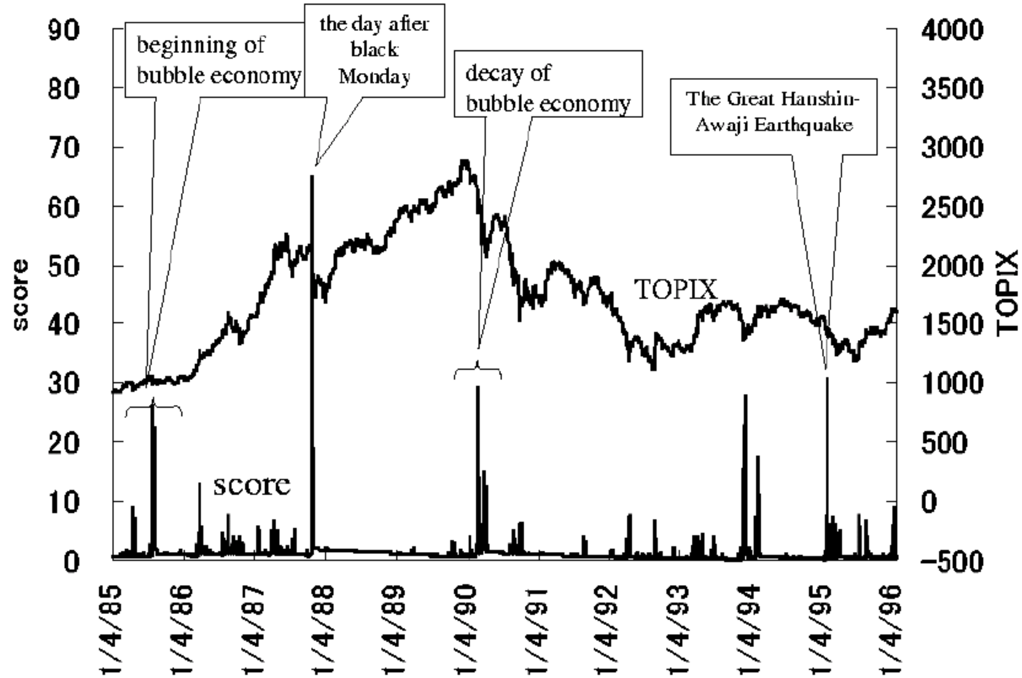
1st order WT

2nd order WT

Other CP detection methods (2/2)

- **Combination of Gaussian mixture and auto-regressive models**
  - Yamanishi et al., KDD 02
  - applicable to nonstationary and noisy signals
  - online-algorithm

  - not very suitable to mixtures of different types of signals
    - → our scope: rough CP detection for heterogeneous mixtures of signals



K. Yamanishi and J. Takeuchi, "A Unifying Framework for Detecting Outliers and Change Points from Non-Stationary Time Series Data", Proc. SIGKDD, pp.676-681, 2002.