AutoPlait: Automatic Mining of Co-evolving Time Sequences

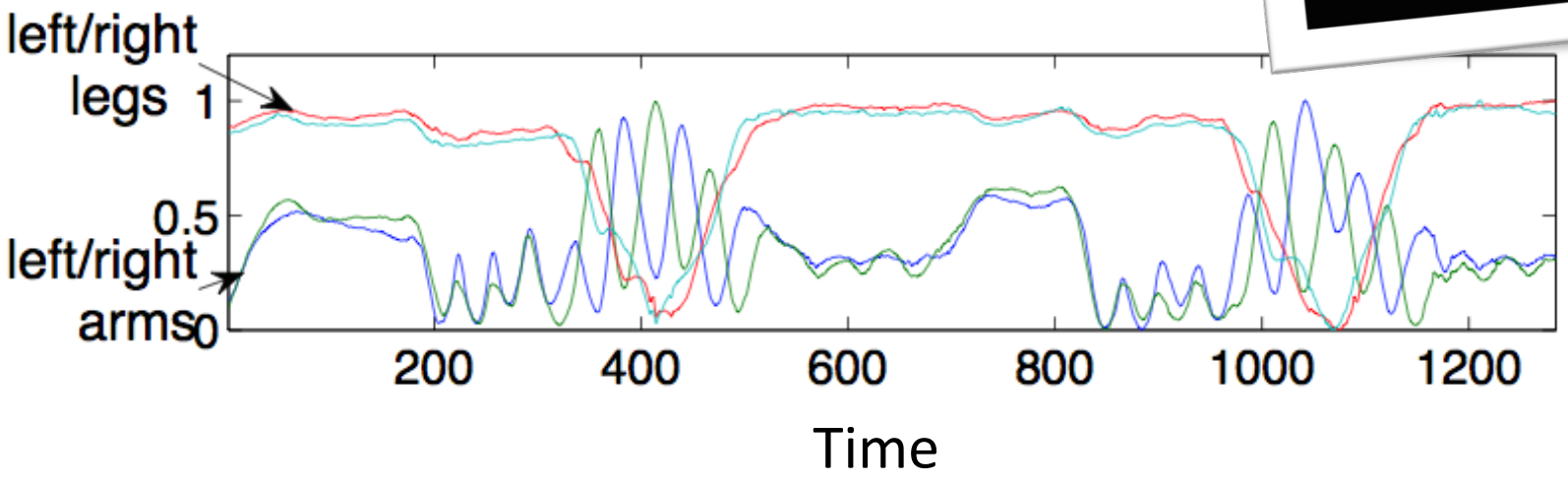
Yasuko Matsubara (Kumamoto University)
Yasushi Sakurai (Kumamoto University)
Christos Faloutsos (CMU)

Motivation

Given: co-evolving time-series

– e.g., MoCap (leg/arm sensors)

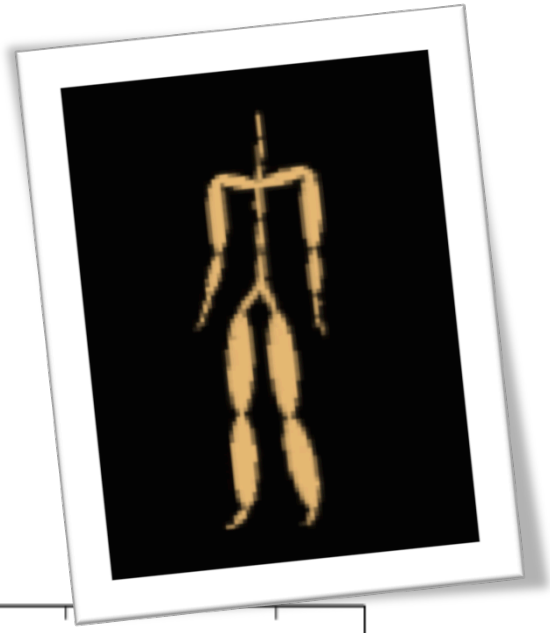
"Chicken dance"



Motivation

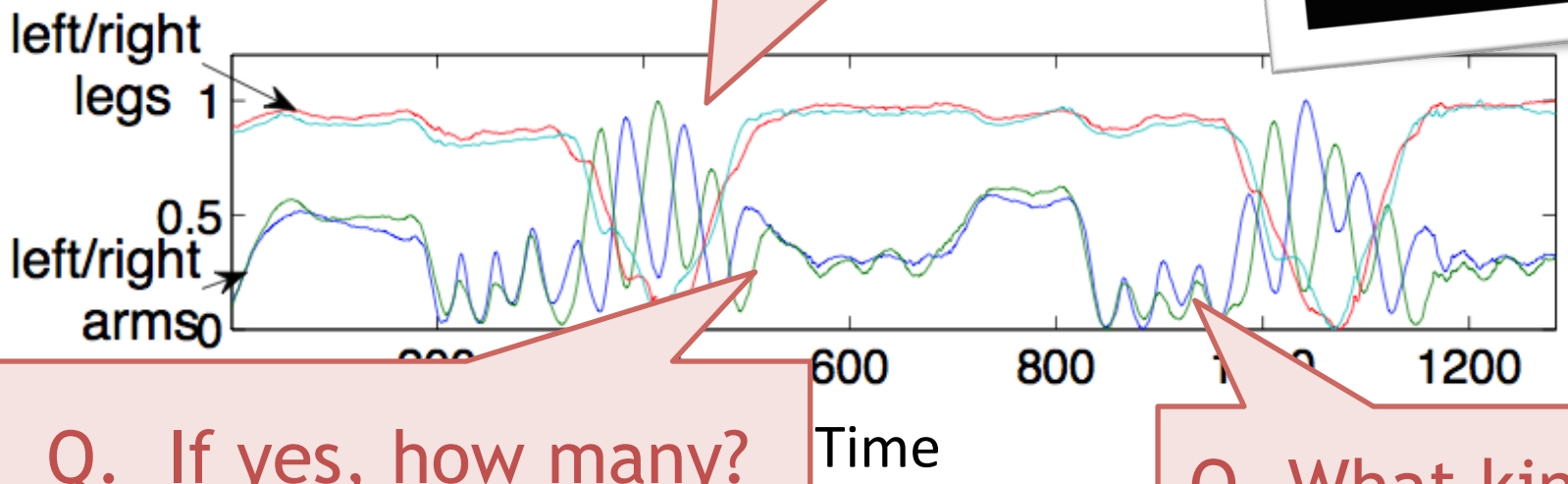
Given: co-evolving time-series

– e.g., MoCap (leg/arm sensors)



"Chicken dance"

Q. Any distinct patterns?



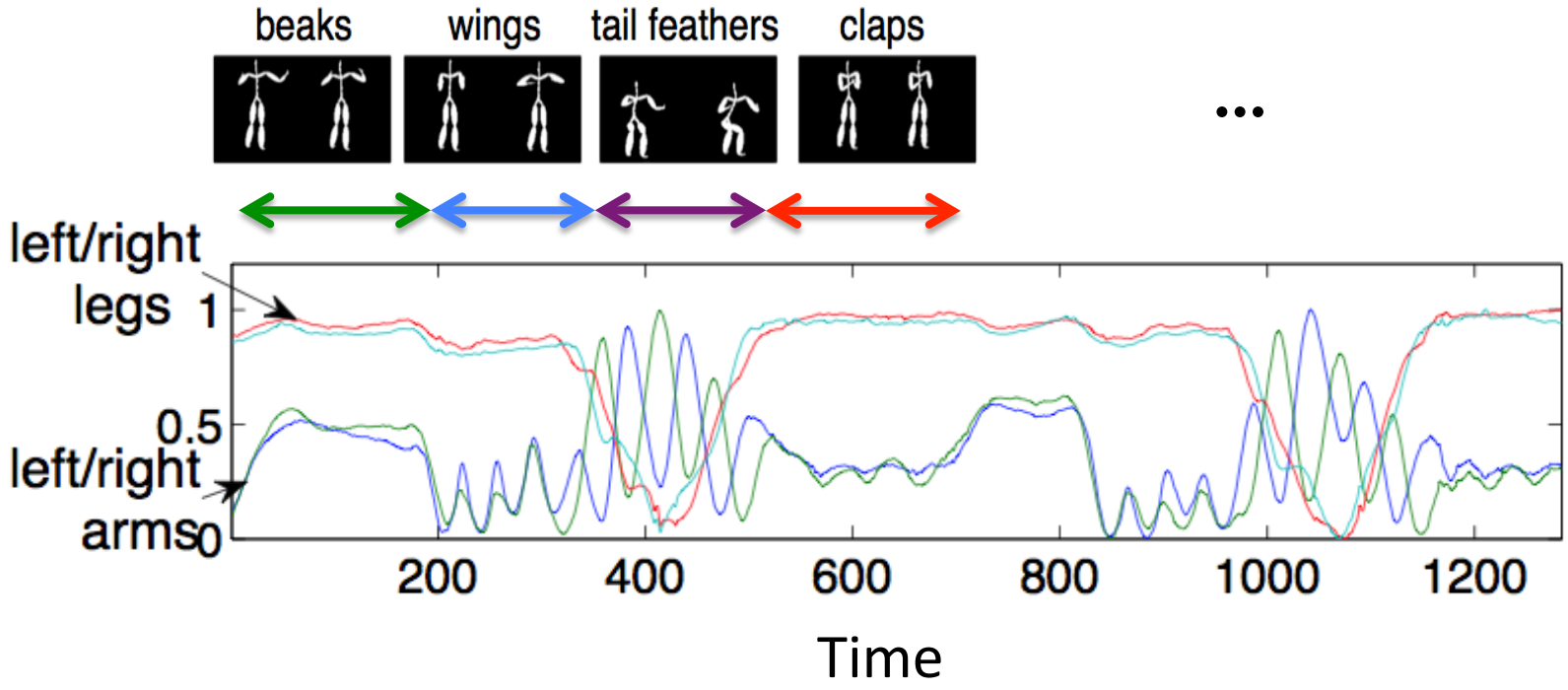
Q. If yes, how many?

Q. What kind?

Motivation

Challenges: co-evolving sequences

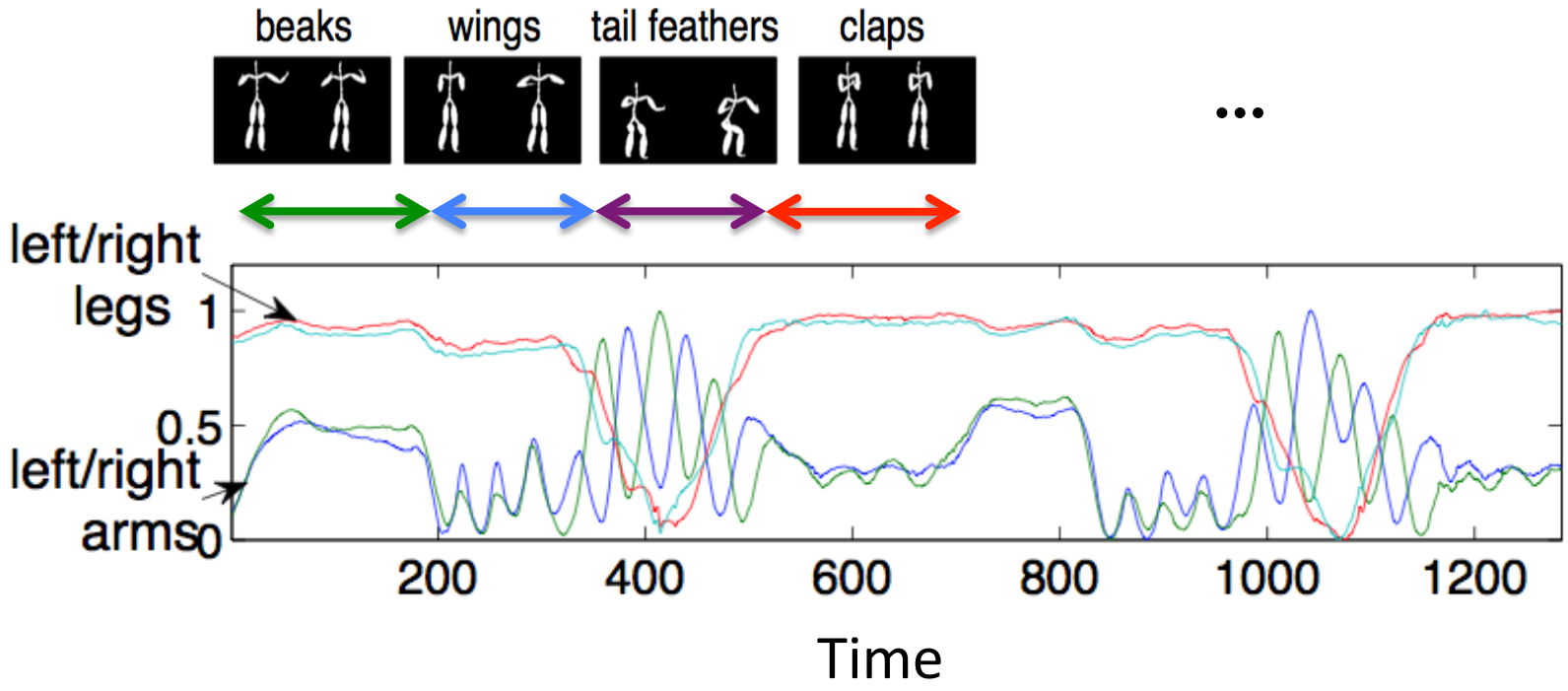
- Unknown # of patterns (e.g., beaks)
- Different durations



Motivation

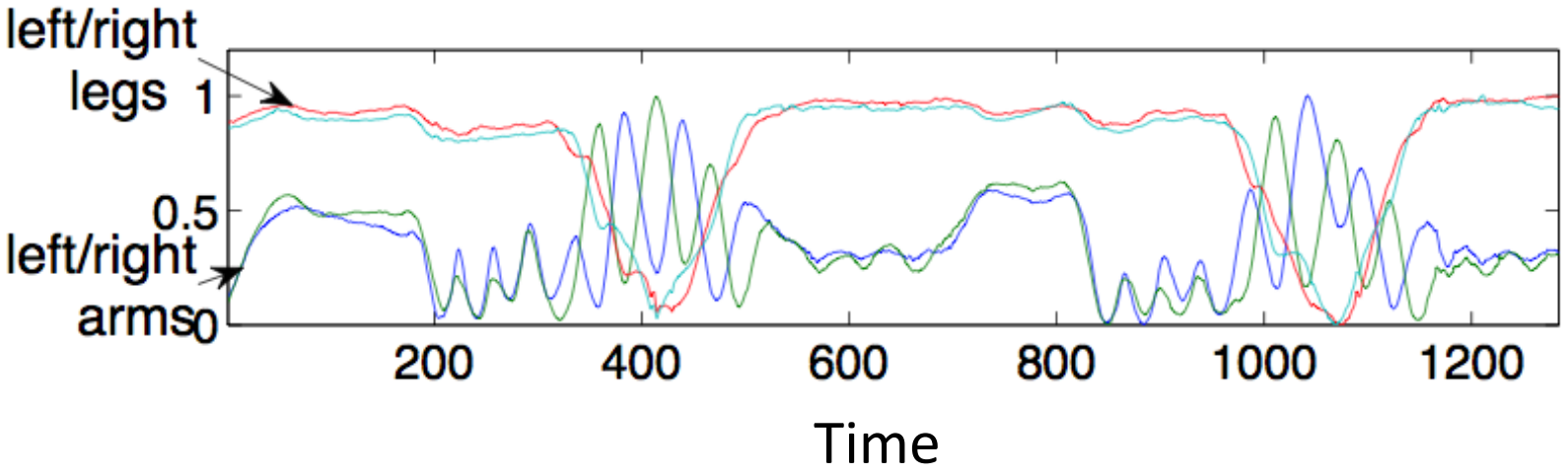
Challenges: co-evolving sequences

Q. Can we summarize it ***automatically*** ?



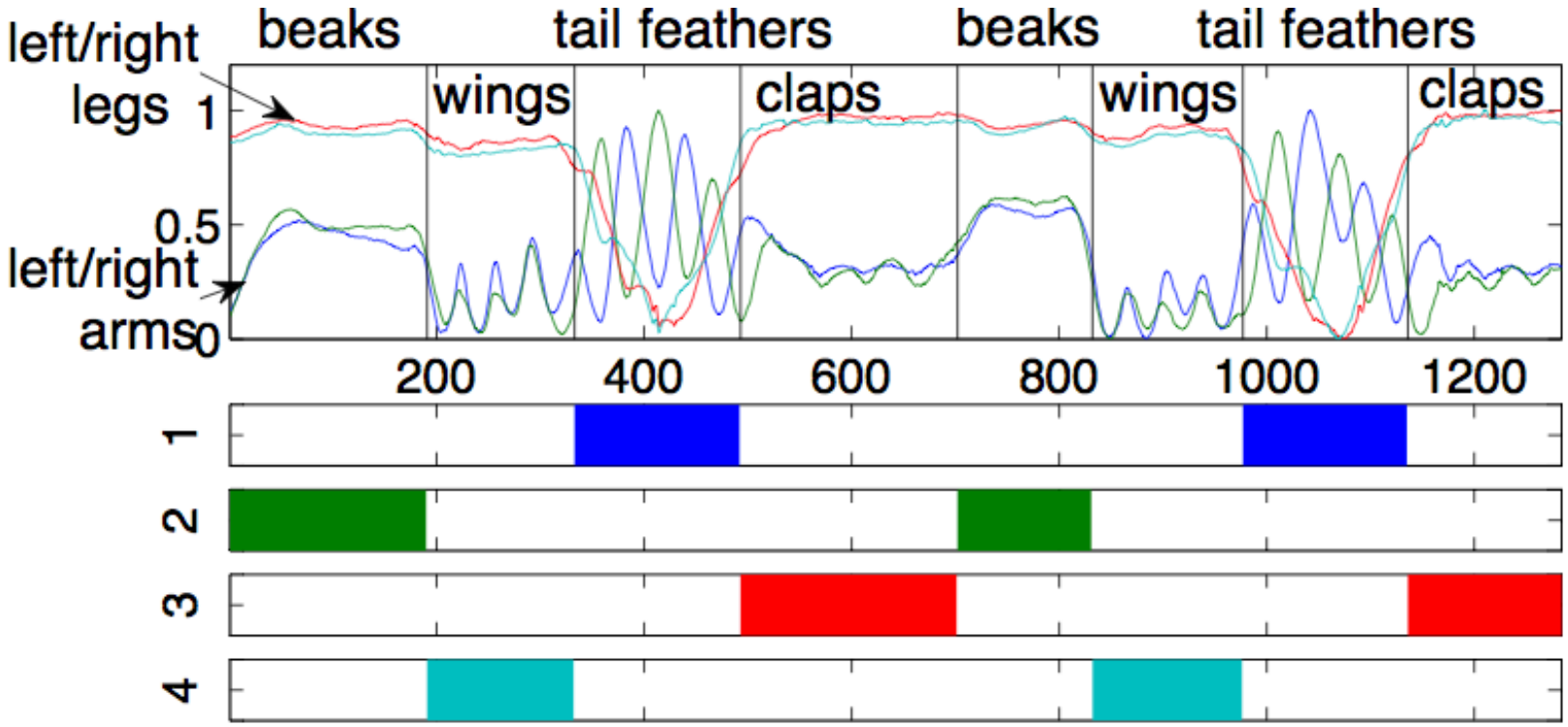
Motivation

Goal: find patterns that agree with human intuition



Motivation

Goal: find patterns that agree with human intuition



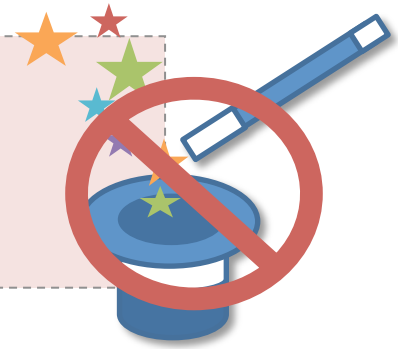
AutoPlait: "**fully-automatic**" mining algorithm

Importance of “fully-automatic”

No magic numbers! ... because,

Manual

- sensitive to the parameter tuning
- long tuning steps (hours, days, …)



Automatic (no magic numbers)

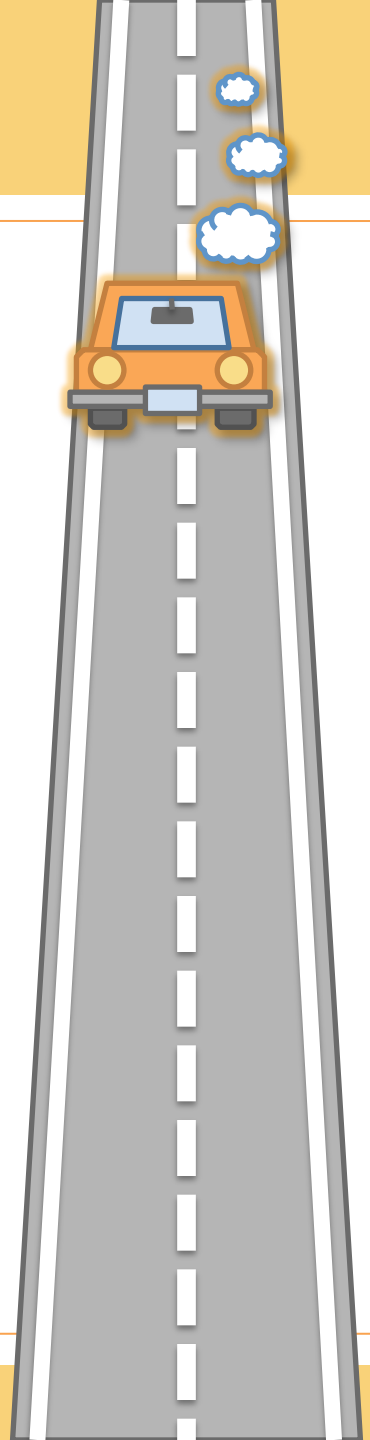
- no expert tuning required

Big data mining:

-> we cannot afford human intervention!!

Outline

- Motivation
- Problem definition
- Compression & summarization
- Algorithms
- Experiments
- AutoPlait at work
- Conclusions



Problem definition

Key concepts

- Bundle: X — given
- Segment: S — hidden
- Regime: Θ — hidden
- Segment-membership: F — hidden

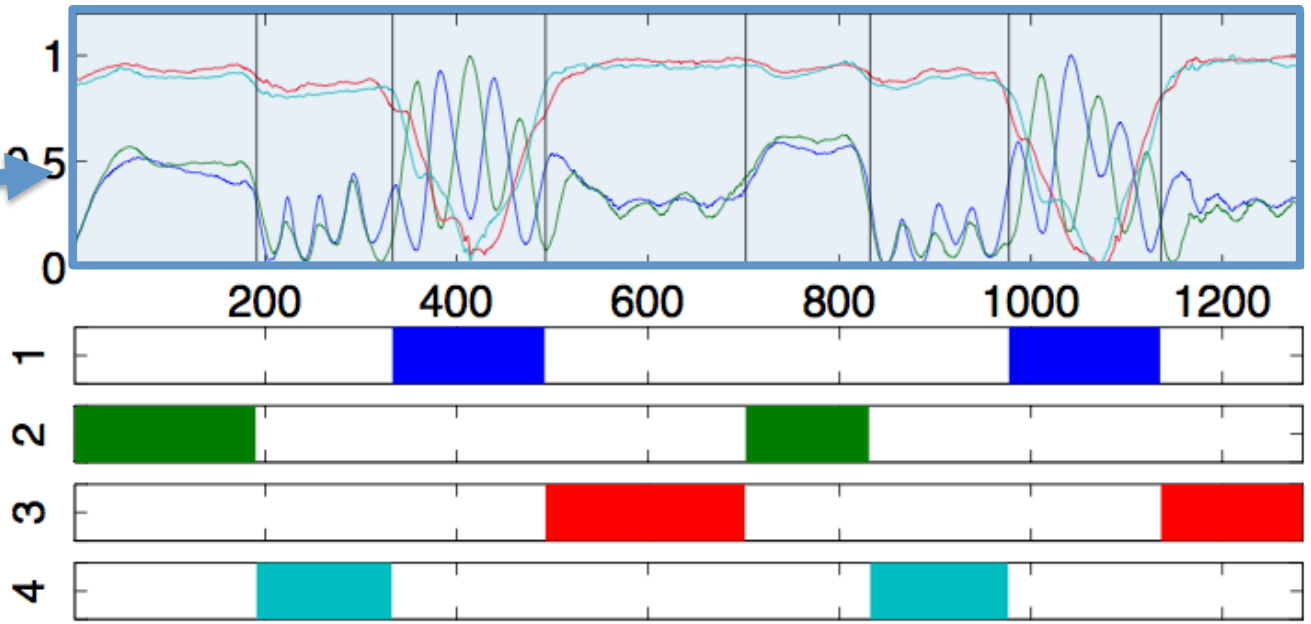
Problem definition

- Bundle : set of d co-evolving sequences

given

$$\underset{d \times n}{X} = \{x_1, \ldots, x_n\}$$

Bundle X (d=4)



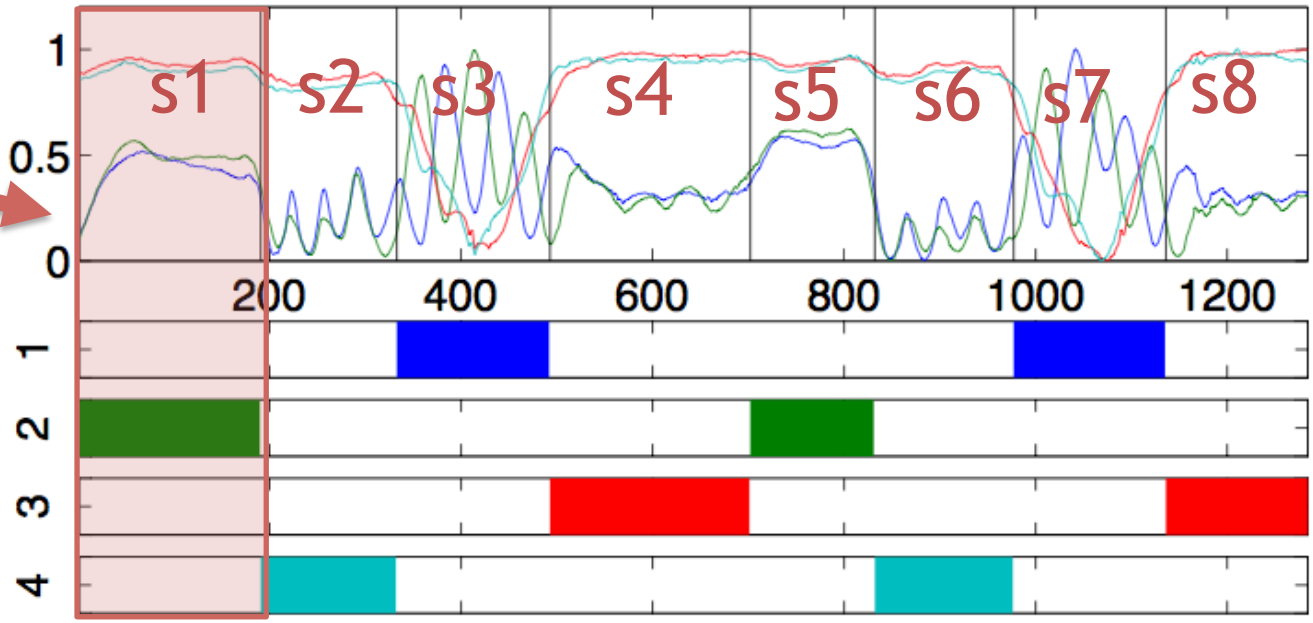
Problem definition

- Segment: convert X -> m segments, S

hidden

$$S = \{s_1, \ldots, s_m\}$$

Segment (m=8)



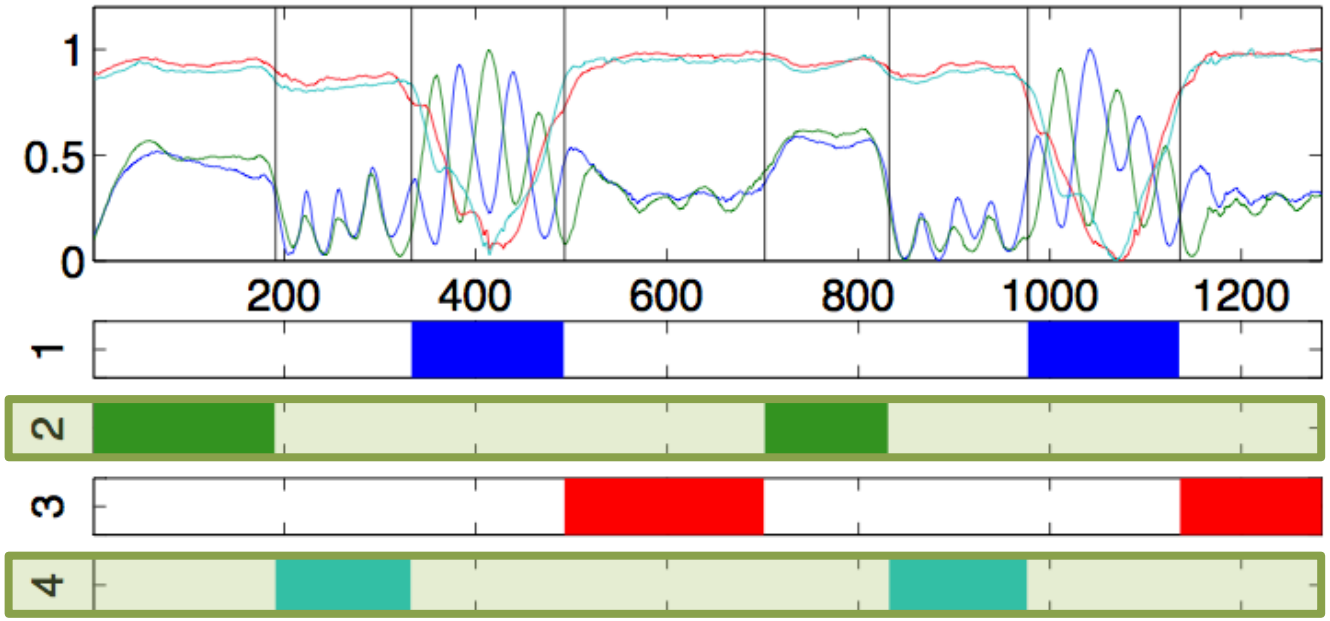
Problem definition

- Regime: segment groups: $\Theta = \{\theta_1, \theta_2, \ldots \theta_r, \Delta_{r \times r}\}$

hidden

θ_r : model of regime r

Regimes (r=4)



θ_1

beaks → θ_2

wings → θ_3

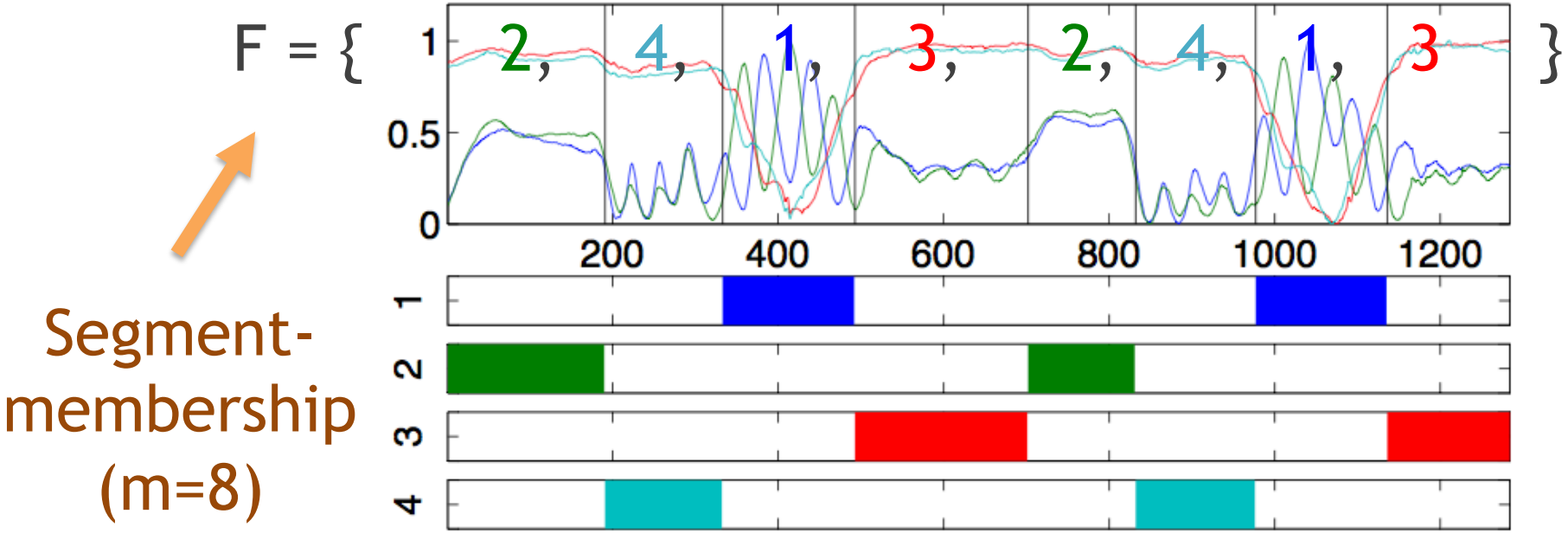
θ_4

Problem definition

- Segment-membership: assignment

hidden

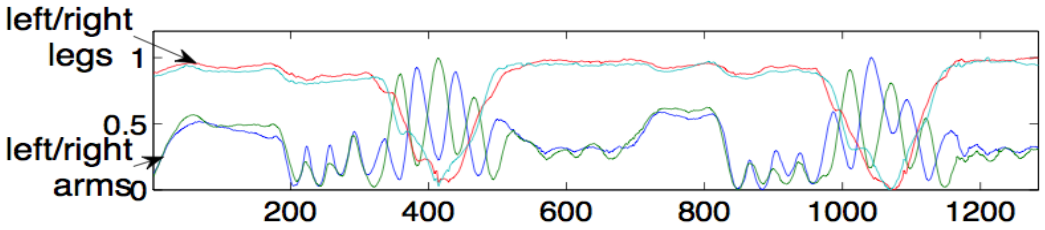
$$F = \{f_1, \ldots, f_m\}$$



Problem definition

- Given: bundle X

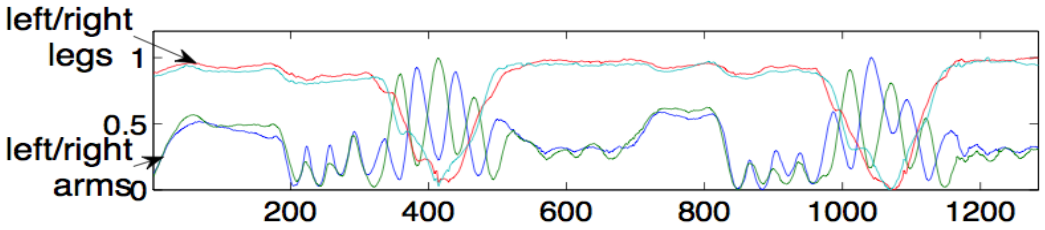
$$X = \{x_1, \ldots, x_n\}$$



Problem definition

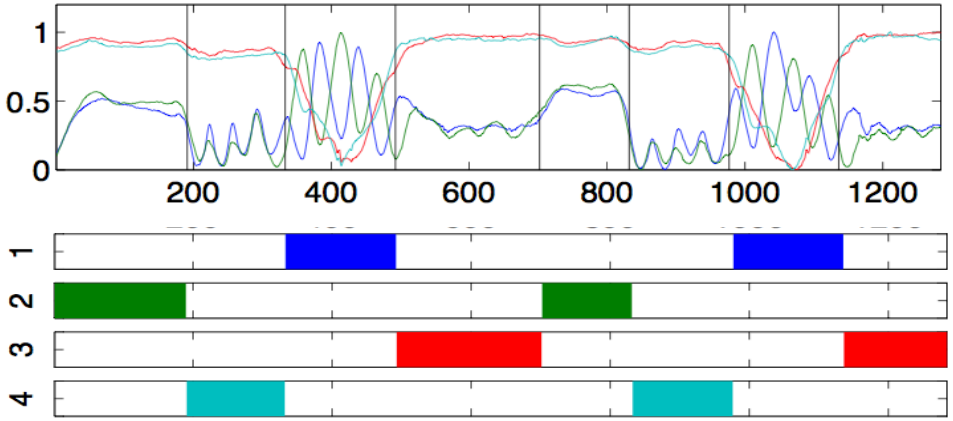
- Given: bundle X

$$X = \{x_1, \ldots, x_n\}$$



- Find: compact description C of X

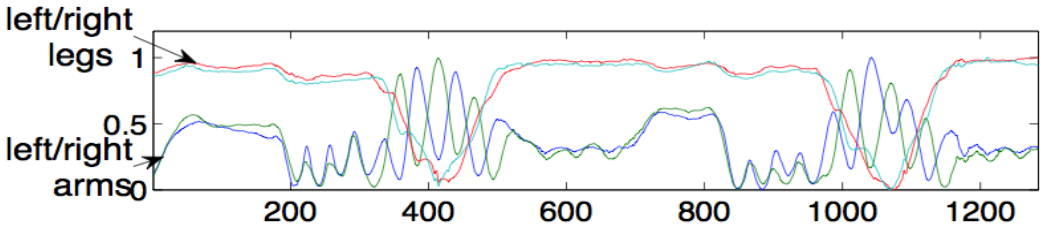
$$C = \{m, r, S, \Theta, F\}$$



Problem definition

- Given: bundle X

$$X = \{x_1, \ldots, x_n\}$$

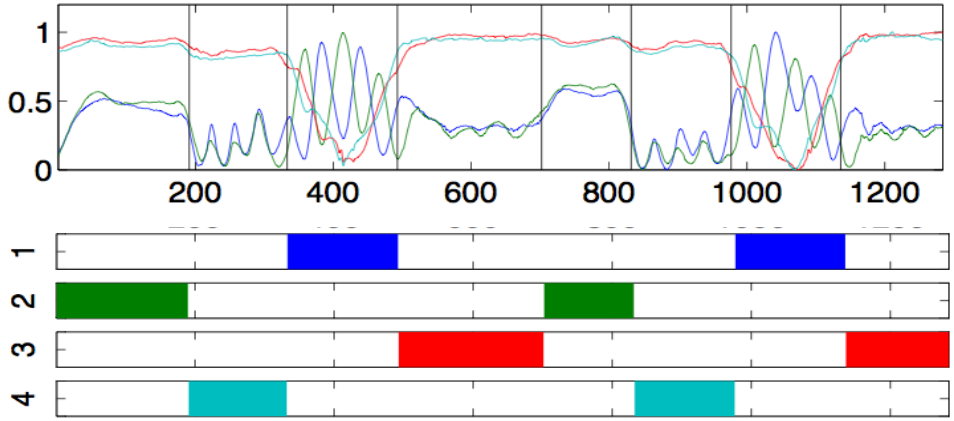


- Find: compact description C of X

$$C = \{m, r, S, \Theta, F\}$$

m segments

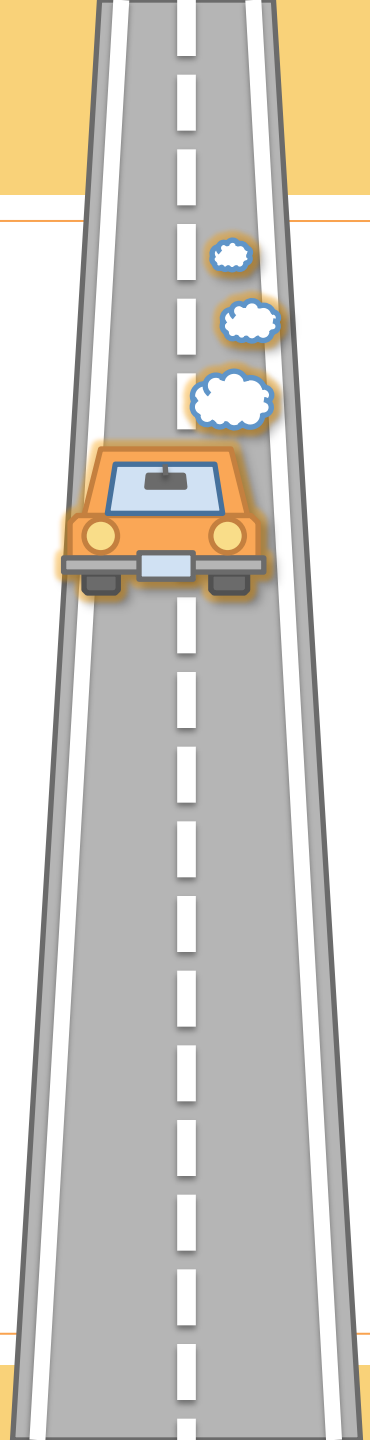
r regimes



Segment-membership

Outline

- Motivation
- Problem definition
- Compression & summarization
- Algorithms
- Experiments
- AutoPlait at work
- Conclusions



Main ideas

Goal: compact description of X

$$C = \{m, r, S, \Theta, F\}$$

without user intervention!!

Challenges:

Q1. How to generate 'informative' regimes ?

Q2. How to decide # of regimes/segments ?

Main ideas

Goal: compact description of X

$$C = \{m, r, S, \Theta, F\}$$

without user intervention!!

Challenges:

Q1. How to generate 'informative' regimes ?

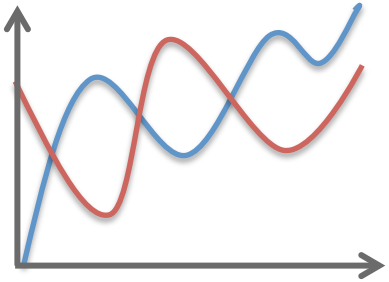
Idea (1): Multi-level chain model

Q2. How to decide # of regimes/segments ?

Idea (2): Model description cost

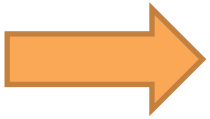
Idea (1): MLCM: multi-level chain model

Q1. How to generate ‘informative’ regimes ?



Sequences

Model



beaks

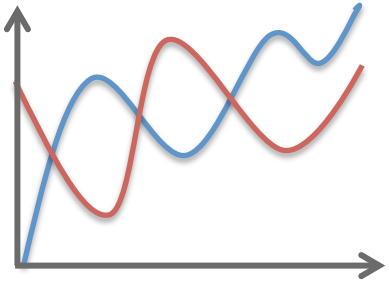
claps

wings

Regimes

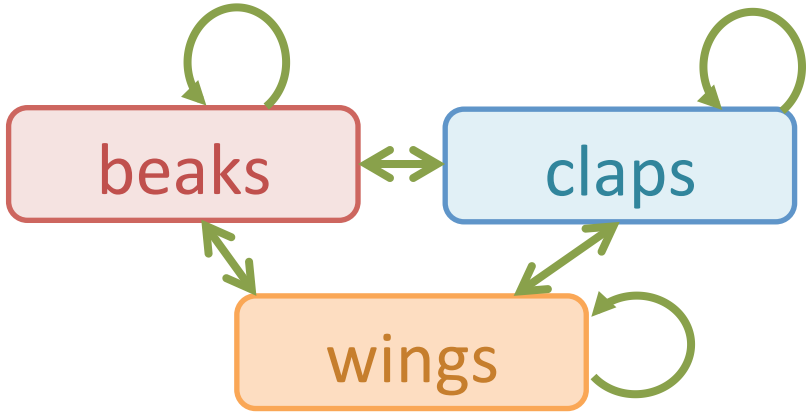
Idea (1): MLCM: multi-level chain model

Q1. How to generate 'informative' regimes ?



Sequences

Model



Regimes

Idea (1): Multi-level chain model

–HMM-based probabilistic model

–with "across-regime" transitions

Idea (1): MLCM: multi-level chain model

Details

$$\Theta = \{\underbrace{\theta_1, \theta_2, \ldots \theta_r}_{\text{r regimes (HMMs)}}, \underbrace{\Delta_{r \times r}}_{\text{across-regime transition prob.}}\} \qquad (\underbrace{\theta_i = \{\pi, A, B\}}_{\text{Single HMM parameters}})$$

Idea (1): MLCM: multi-level chain model

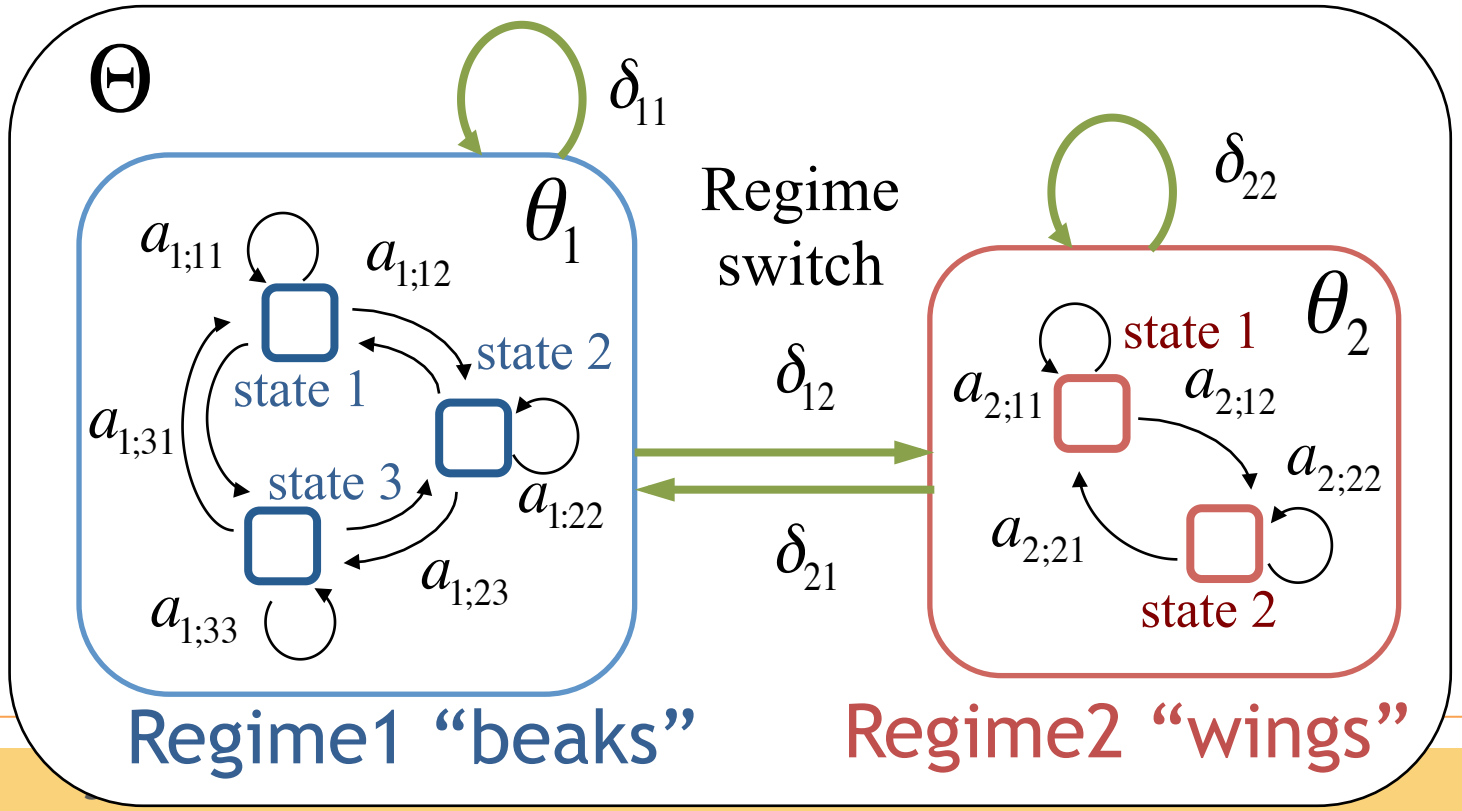
Details

$$\Theta = \{\theta_1, \theta_2, \ldots \theta_r, \Delta_{r \times r}\} \qquad (\theta_i = \{\pi, A, B\})$$

r regimes (HMMs)

across-regime transition prob.

Single HMM parameters



Regimes r=2

Regime 1 (k=3)

Regime 2 (k=2)

Idea (2): model description cost

Q2. How to decide # of regimes/segments ?

Idea (2): Model description cost

- Minimize coding cost
- find "optimal" # of segments/regimes

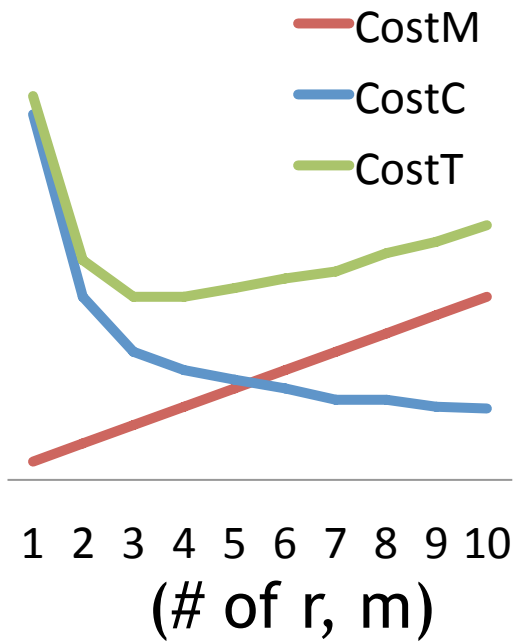
Idea (2): model description cost

Idea: Minimize encoding cost!

$$\min \left(\text{Cost}_M(M) + \text{Cost}_c(X \mid M) \right)$$

Model cost

Coding cost



Good compression ⇄ Good description

Idea (2): model description cost

Details

Total cost of bundle X, given C

$$C = \{m, r, S, \Theta, F\}$$

$$\begin{aligned} Cost_T(\boldsymbol{X}; \mathcal{C}) &= Cost_T(\boldsymbol{X}; m, r, \mathcal{S}, \boldsymbol{\Theta}, \mathcal{F}) \\ &= \log^*(n) + \log^*(d) + \log^*(m) + \log^*(r) + m\log(r) \\ &\quad + \sum_{i=1}^{m-1} \log^* |s_i| + Cost_M(\boldsymbol{\Theta}) + Cost_C(\boldsymbol{X}|\boldsymbol{\Theta}) \end{aligned} \quad (6)$$

Idea (2): model description cost

Details

Total cost of bundle X, given C

$$C = \{m, r, S, \Theta, F\}$$

duration/ dimensions

# of segments/ regimes

segment-membership F

$$Cost_T(\boldsymbol{X}; \mathcal{C}) = Cost_T(\boldsymbol{X}; m, r, \mathcal{S}, \boldsymbol{\Theta}, \mathcal{F})$$

$$= \log^*(n) + \log^*(d) + \log^*(m) + \log^*(r) + m \log(r)$$

$$+ \sum_{i=1}^{m-1} \log^* |s_i| + Cost_M(\boldsymbol{\Theta}) + Cost_C(\boldsymbol{X}|\boldsymbol{\Theta}) \quad (6)$$

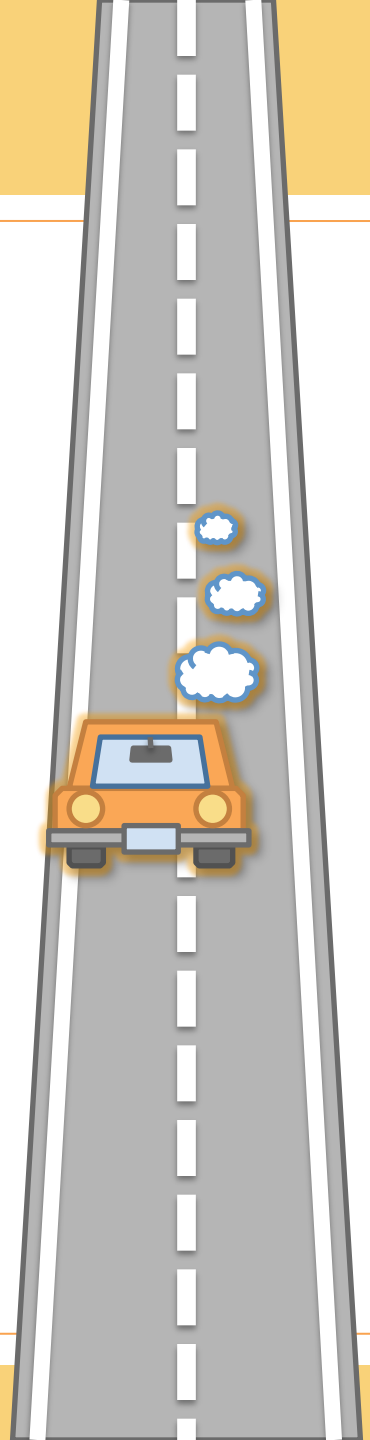
segment lengths

Model description cost of Θ

Coding cost of X given Θ

Outline

- Motivation
- Problem definition
- Compression & summarization
- Algorithms
- Experiments
- AutoPlait at work
- Conclusions

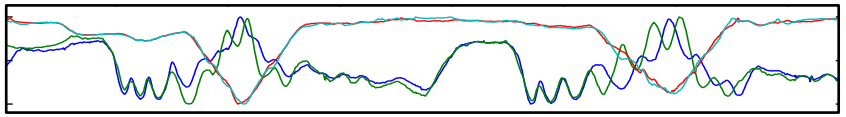


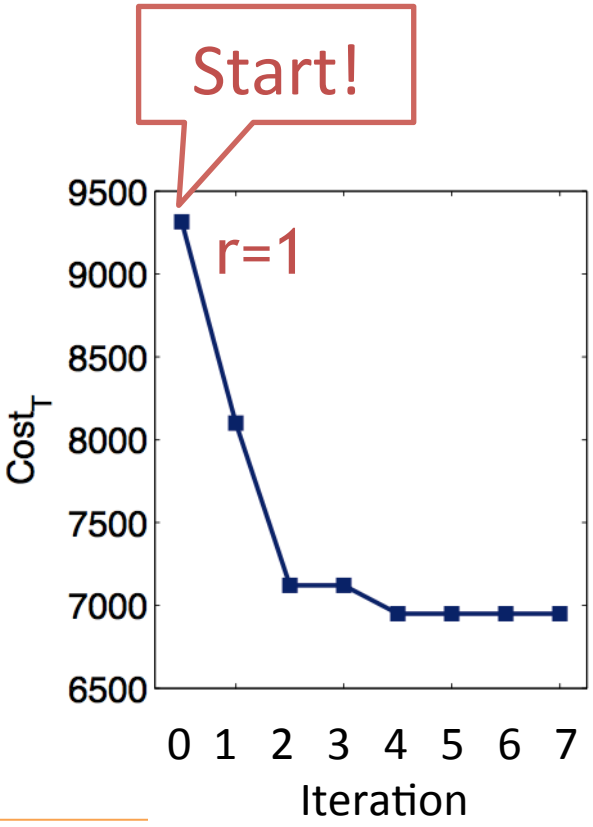
AutoPlait

Overview

Iteration 0
r=1, m=1

X

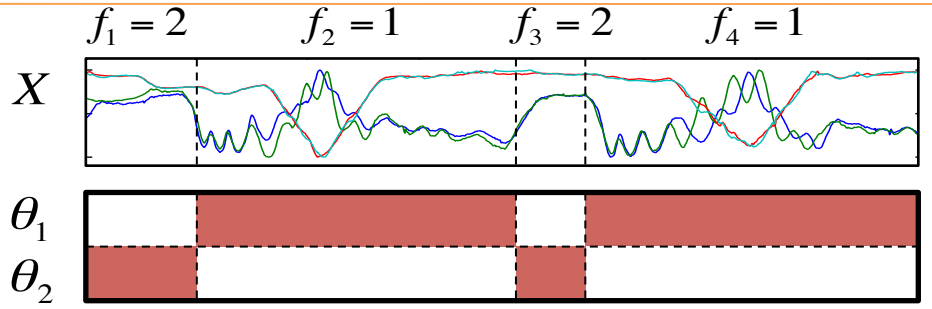


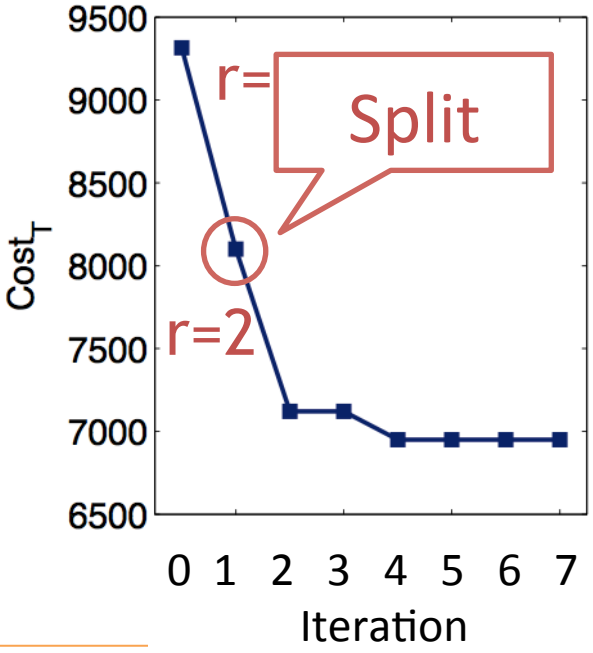


AutoPlait

Overview

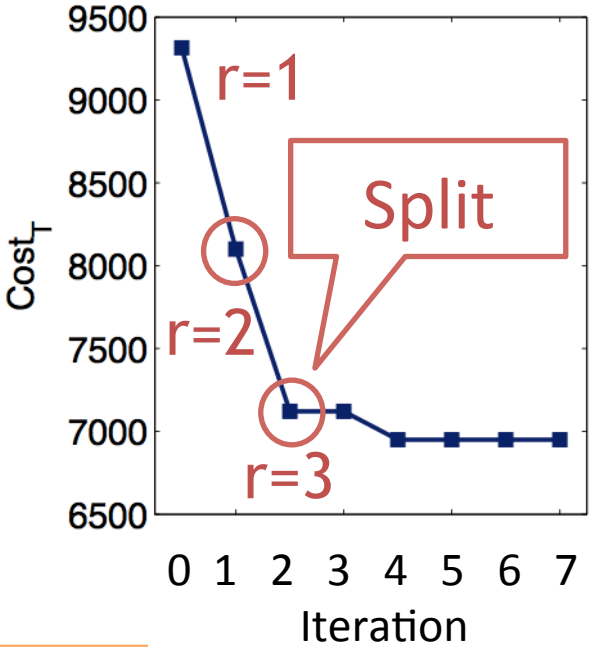
Iteration 1
r=2, m=4



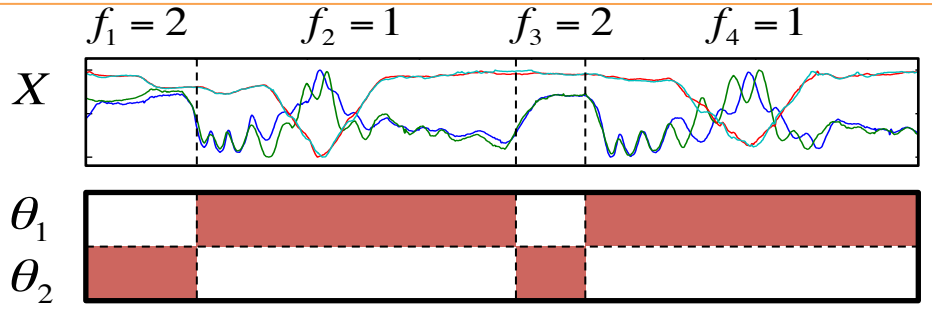


AutoPlait

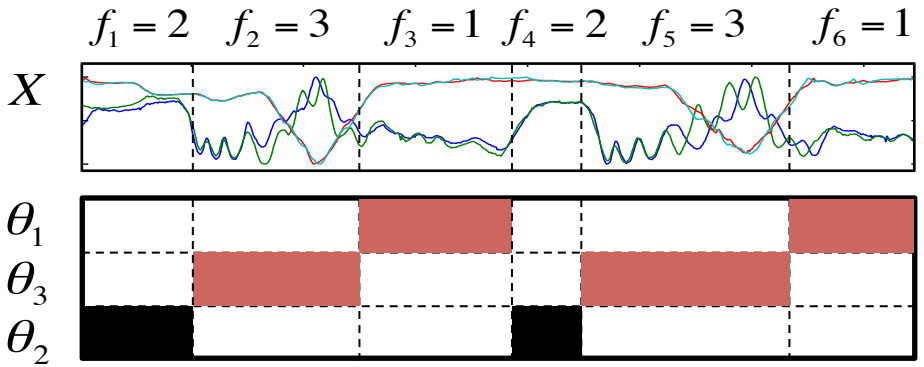
Overview



Iteration 1
r=2, m=4

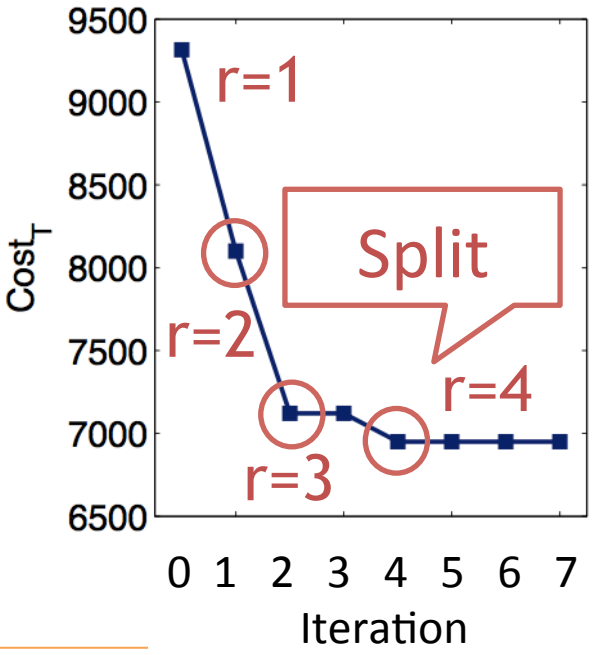


Iteration 2
r=3, m=6



AutoPlait

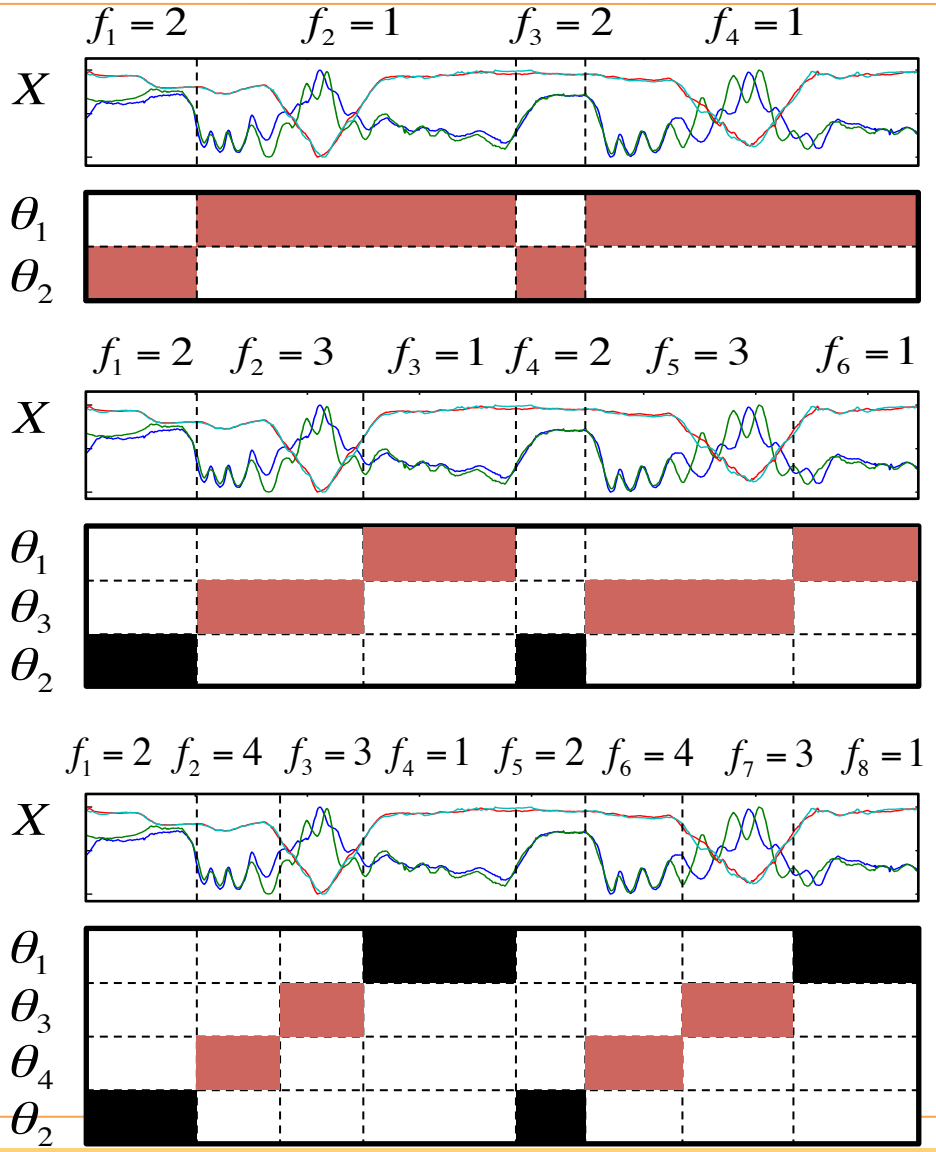
Overview



Iteration 1
r=2, m=4

Iteration 2
r=3, m=6

Iteration 4
r=4, m=8



AutoPlait

Algorithms

1. CutPointSearch — Inner-most loop

   Find good cut-points/segments

2. RegimeSplit — Inner loop

   Estimate good regime parameters **Θ**

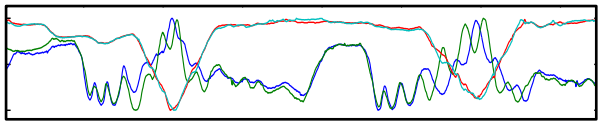
3. AutoPlait — Outer loop

   Search for the best number of regimes (r=2,3,4…)

1.CutPointSearch

Inner-most loop

Given:

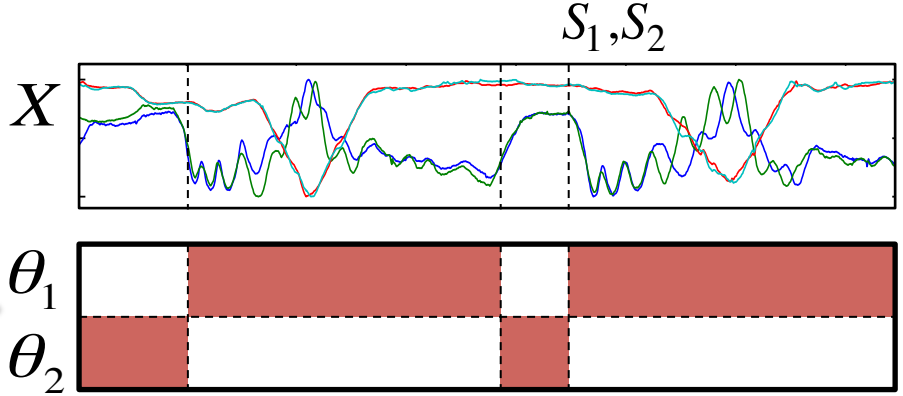
- bundle X 
- parameters of two regimes $\Theta = \{\theta_1, \theta_2, \Delta\}$

Find: cut-points of segment sets S_1, S_2,

$$\{S_1, S_2\} = \mathrm{argmax}\, P(X \mid S_1, S_2, \Theta)$$

X

$\{\theta_1, \theta_2, \Delta\}$



$S_1 = \{s_2, s_4\}$
$S_2 = \{s_1, s_3\}$

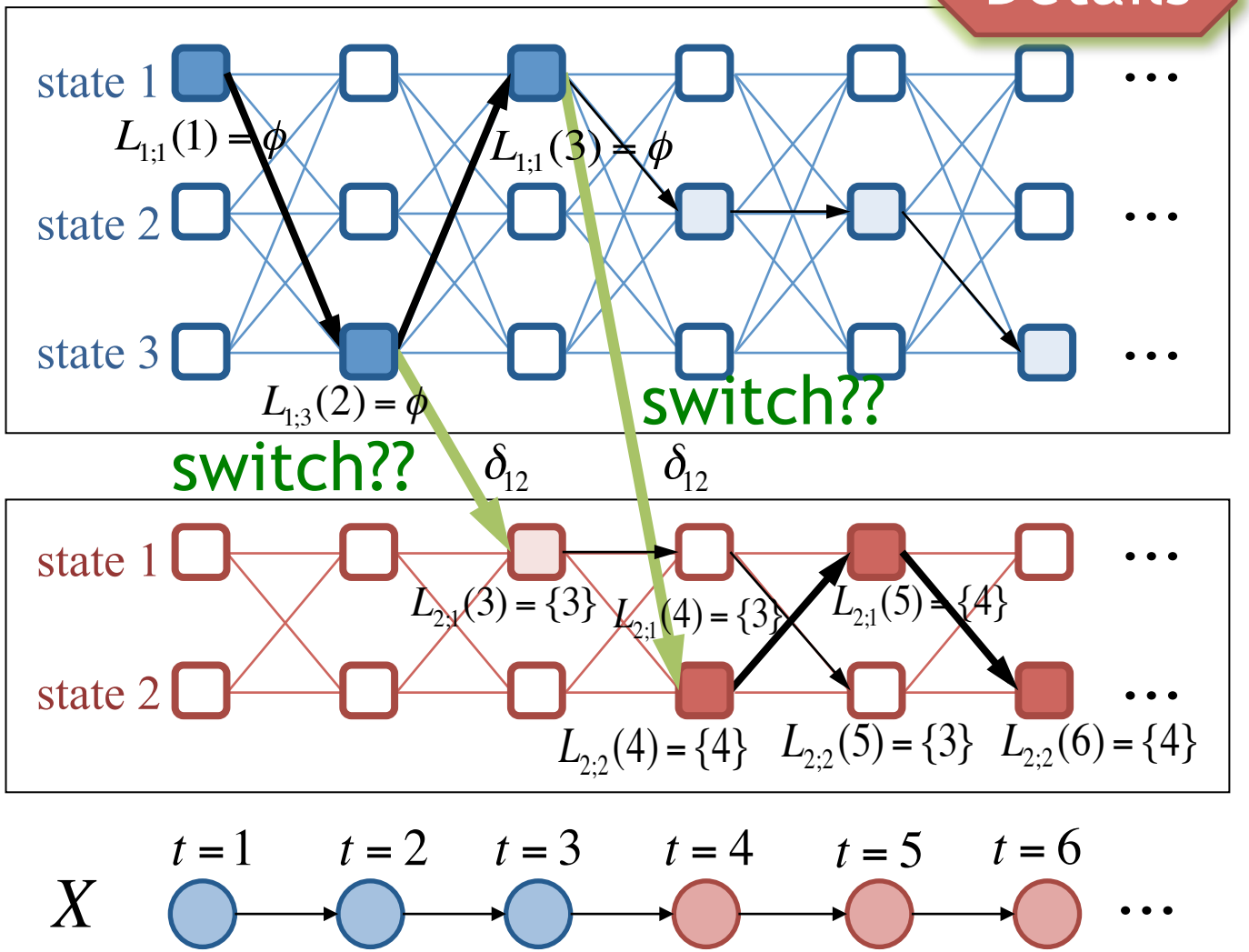
1.CutPointSearch

Inner-most loop

Details

DP algorithm to compute likelihood:

$$P(X \mid \Theta)$$



1.CutPointSearch

Inner-most loop

Details

Theoretical analysis

Scalability

- It takes $O(ndk^2)$ time (only single scan)
  - n: length of X
  - d: dimension of X
  - k: # of hidden states in regime

Accuracy

It guarantees the optimal cut points

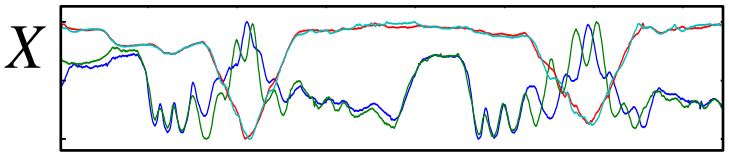
- (Details in paper)

2.RegimeSplit

Inner loop

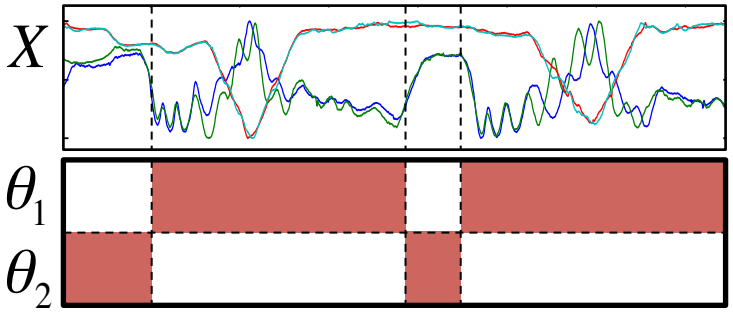
Given:

- bundle X



Find: two regimes

1. find cut-points of segment sets: S_1, S_2
2. estimate parameters of two regimes:



$$\Theta = \{\theta_1, \theta_2, \Delta\}$$

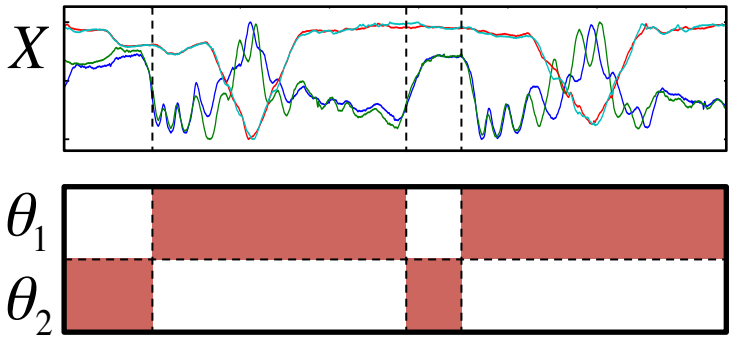
2.RegimeSplit

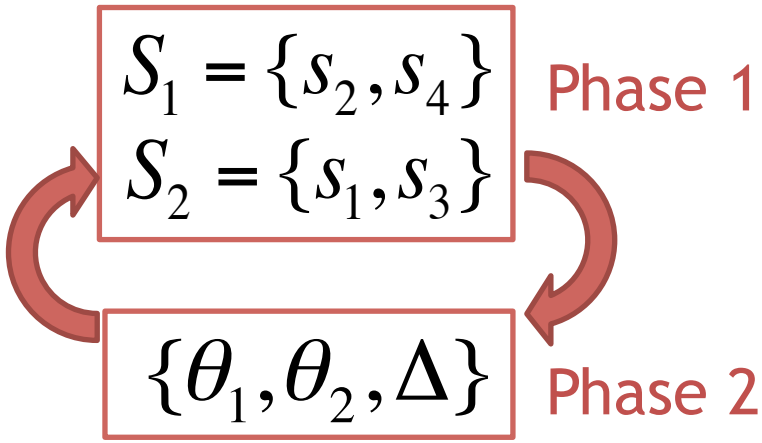
Inner loop

Details

Two-phase iterative approach

- Phase 1: (CutPointSearch)
  - Split segments into two groups : S_1, S_2
- Phase 2: (BaumWelch)
  - Update model parameters: $\Theta = \{\theta_1, \theta_2, \Delta\}$



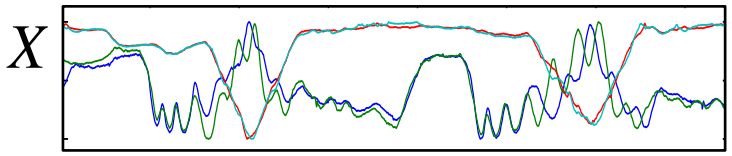


3.AutoPlait

Outer loop

Given:

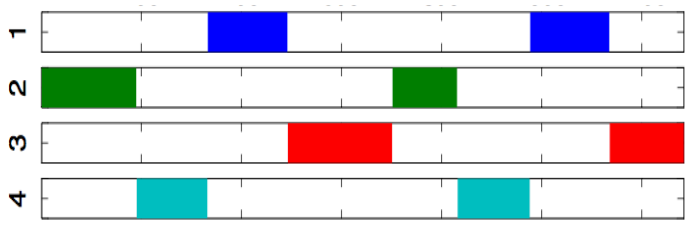
- bundle X



Find: r regimes (r=2, 3, 4, …)

- i.e., find full parameter set

$$C = \{m, r, S, \Theta, F\}$$



3.AutoPlait

Outer loop

Split regimes r=2,3,…, as long as cost keeps decreasing

- Find appropriate # of regimes

$$r = \min_{r} Cost_T(X; m, r, S, \Theta, F)$$

3.AutoPlait

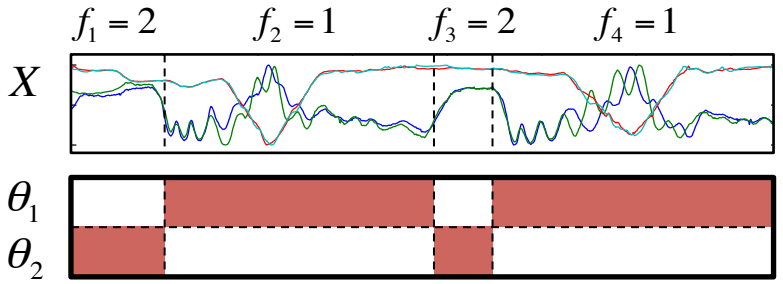
Outer loop

Split regimes r=2,3,…, as long as cost keeps decreasing

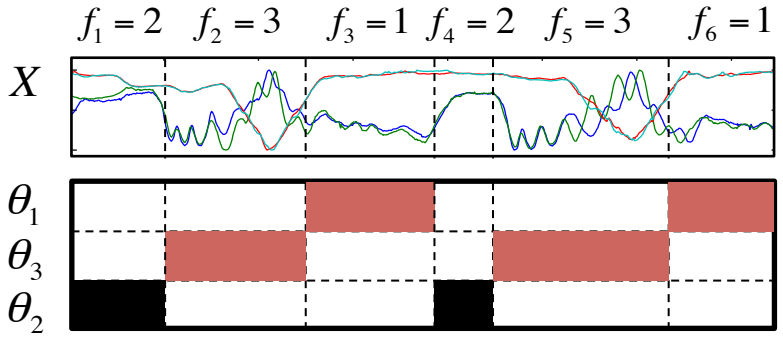
- Find appropriate # of regimes

$$r = \min_{r} Cost_T(X; m, r, S, \Theta, F)$$

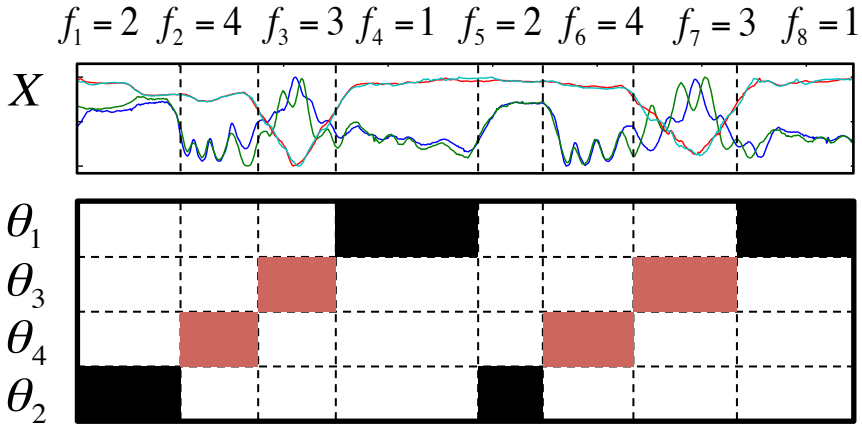
r=2, m=4



r=3, m=6

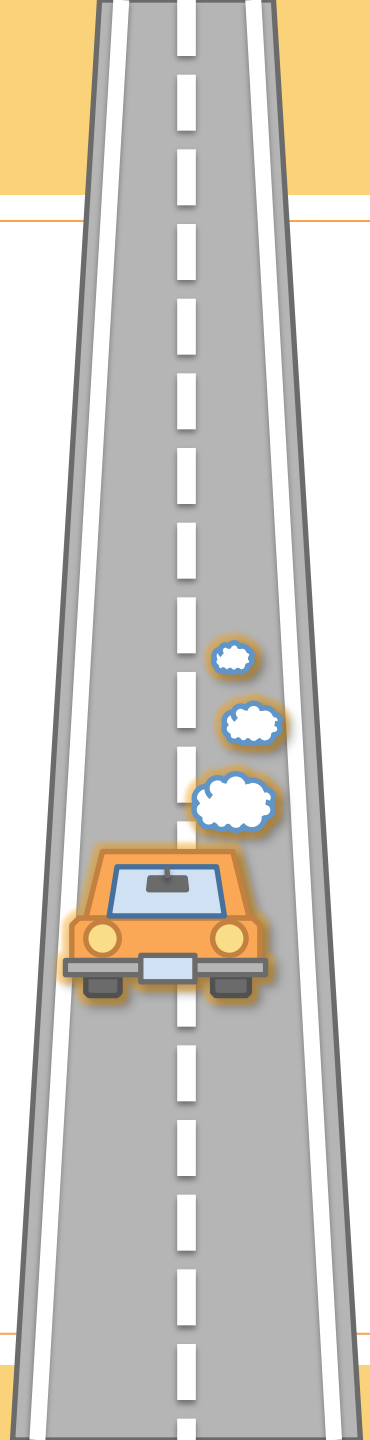


r=4, m=8



Outline

- Motivation
- Problem definition
- Compression & summarization
- Algorithms
- Experiments
- AutoPlait at work
- Conclusions



Experiments

We answer the following questions…

Q1. Sense-making

Can it help us understand the given bundles?

Q2. Accuracy

How well does it find cut-points and regimes?

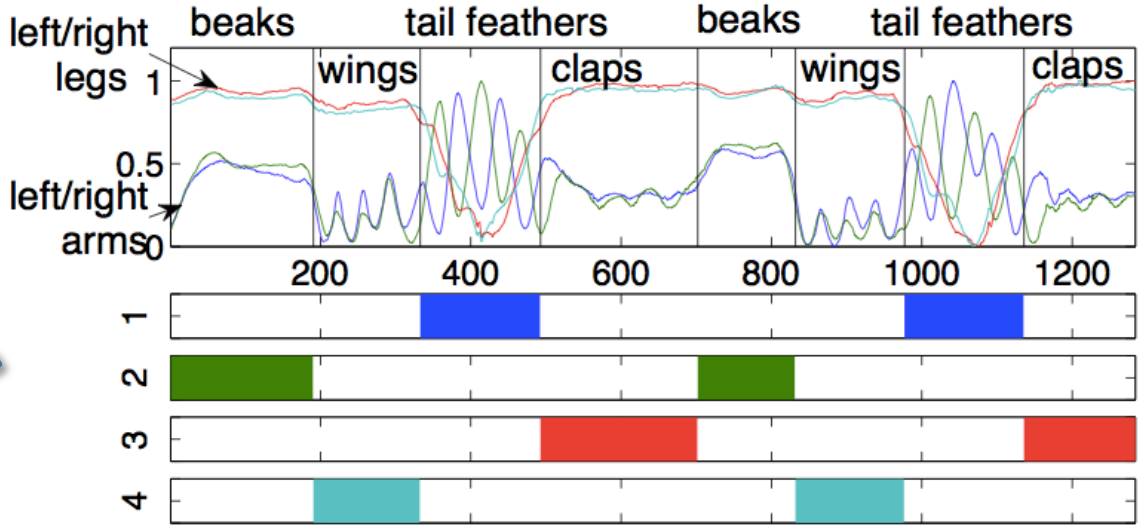
Q3. Scalability

How does it scale in terms of computational time?

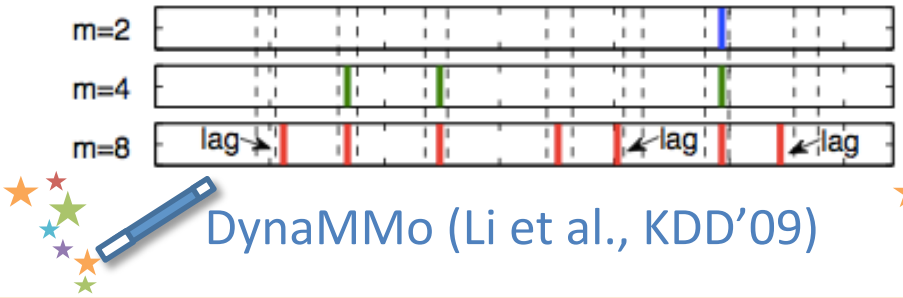
Q1. Sense-making

MoCap data

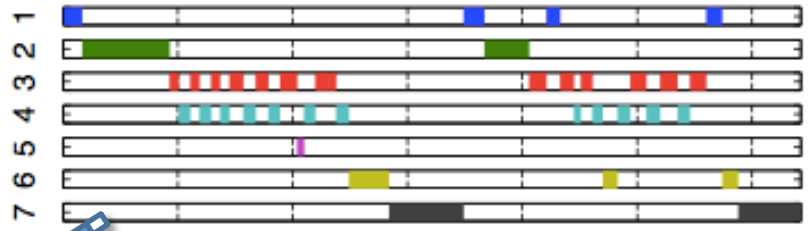
AutoPlait (NO magic numbers)



(a) AUTOPLAIT (**no** user defined parameters)



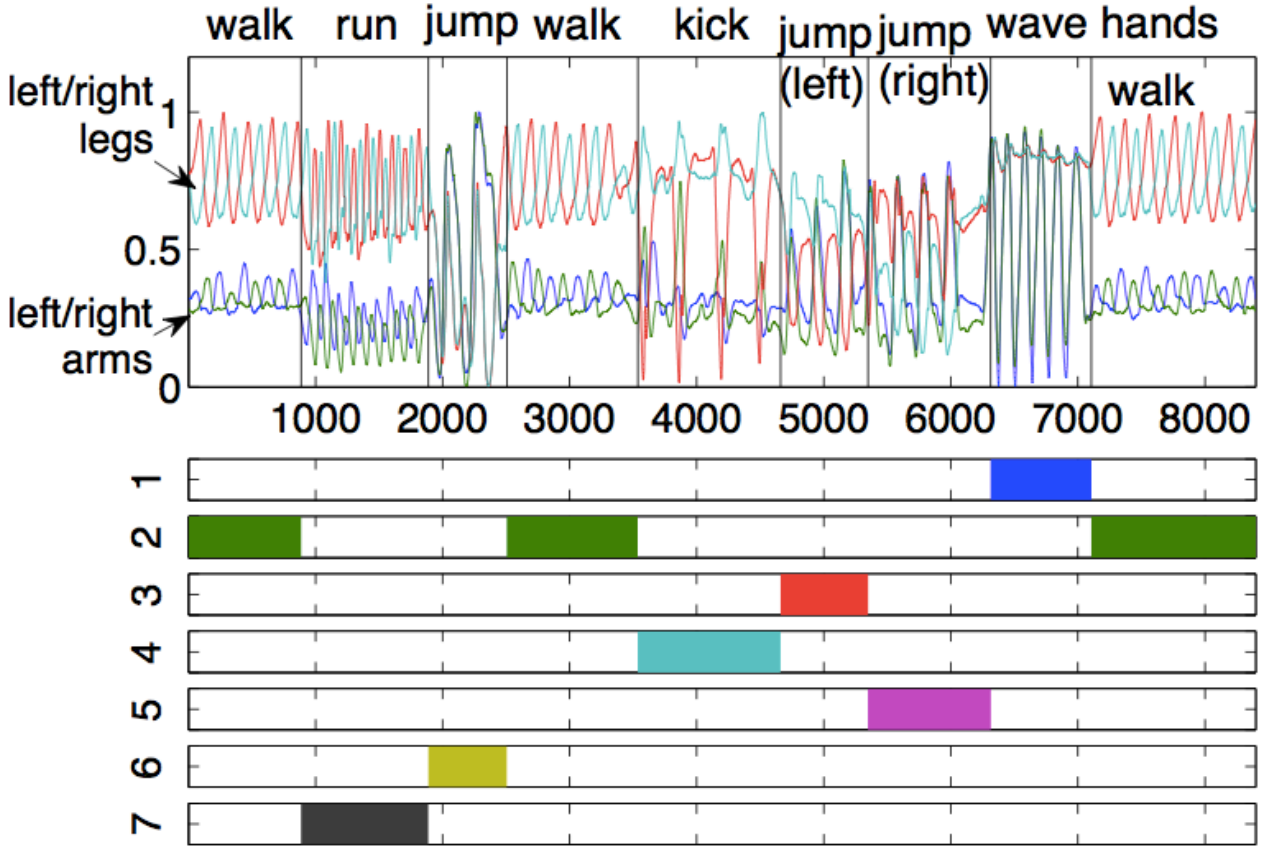
DynaMMo (Li et al., KDD'09)



pHMM (Wang et al., SIGMOD'11)

Q1. Sense-making

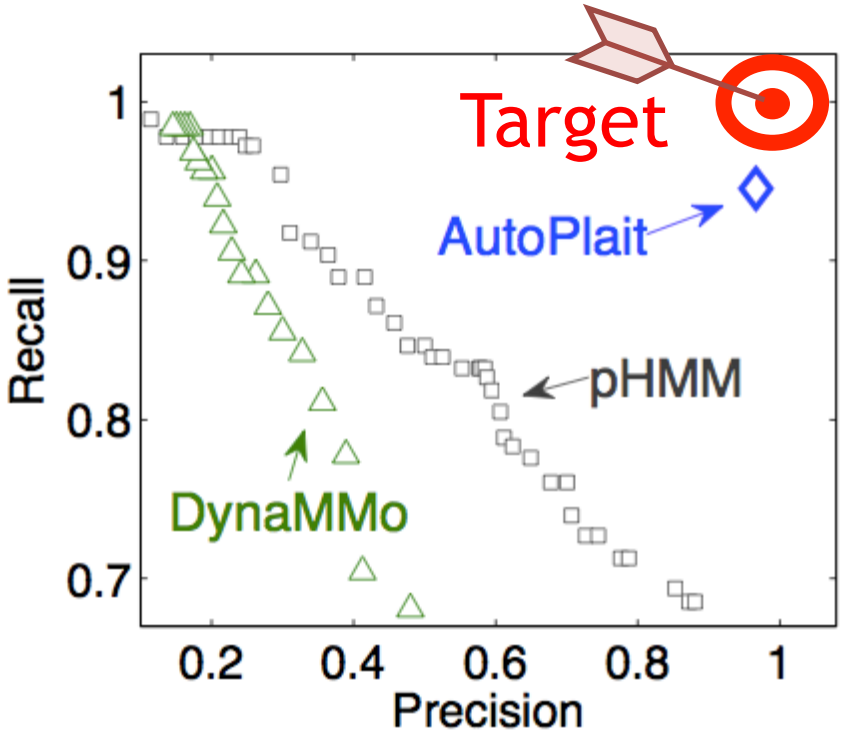
MoCap data



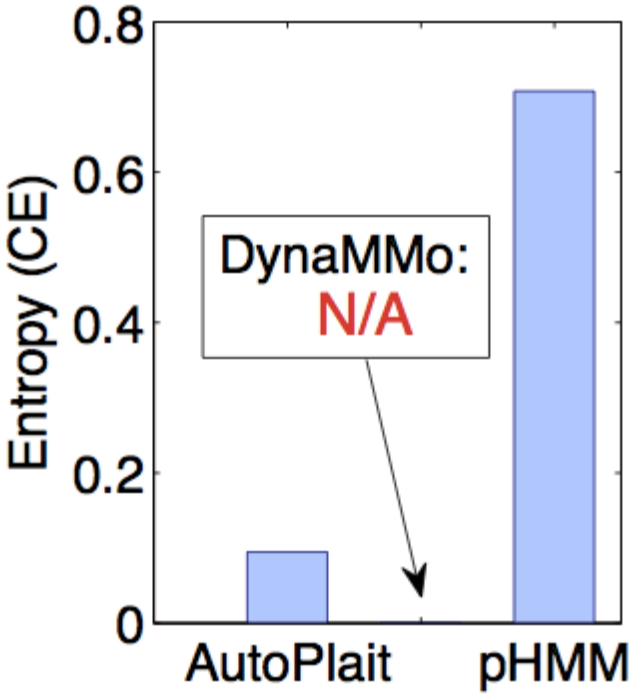
AutoPlait (NO magic numbers)

Q2. Accuracy

(a) Segmentation



(b) Clustering



(a) Precision and recall (higher is better)

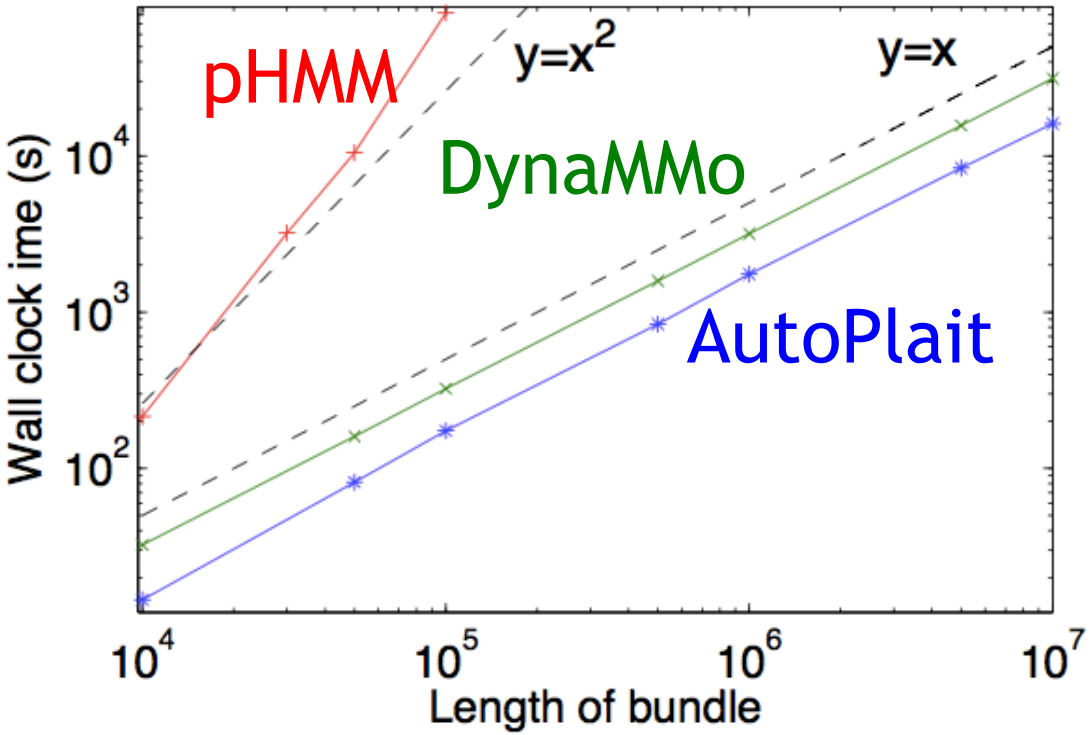
(b) CE score (lower is better)

AutoPlait needs "no magic numbers"

Q3. Scalability

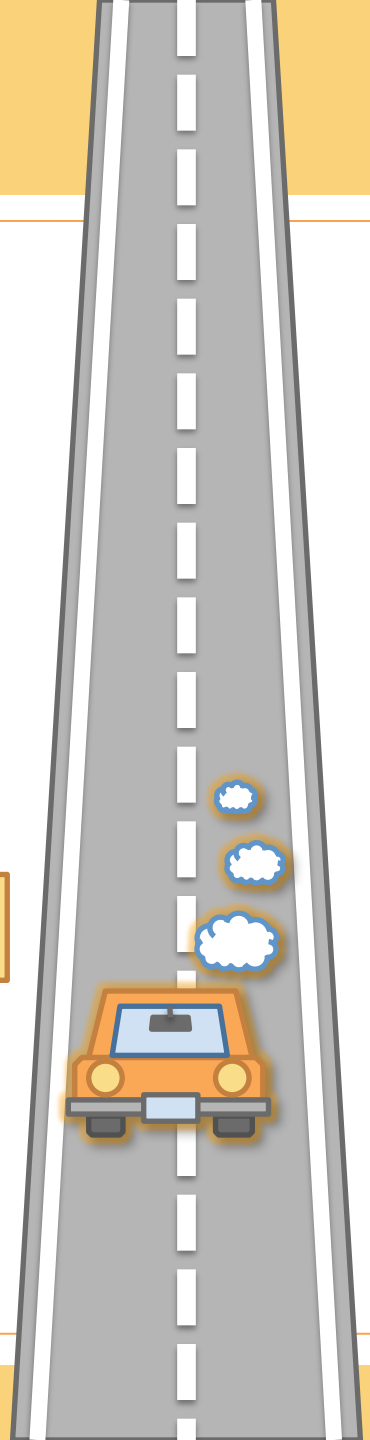
Wall clock time vs. data size (length) : n

AutoPlait scales linearly, i.e., O(n)



Outline

- Motivation
- Problem definition
- Compression & summarization
- Algorithms
- Experiments
- AutoPlait at work
- Conclusions



AutoPlait at work

AutoPlait is capable of various applications, e.g.,

App1. Model analysis

- Web-click sequences

App2. Event discovery

- Google Trend data

AutoPlait at work

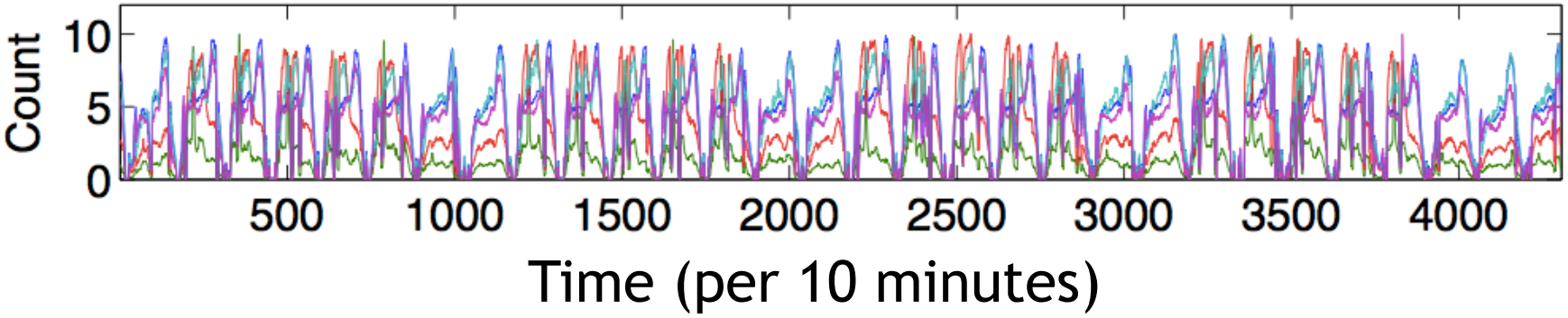
AutoPlait is capable of various applications, e.g.,

App1. Model analysis
- Web-click sequences

App2. Event discovery
- Google Trend data

App1. Model analysis (WebClick)

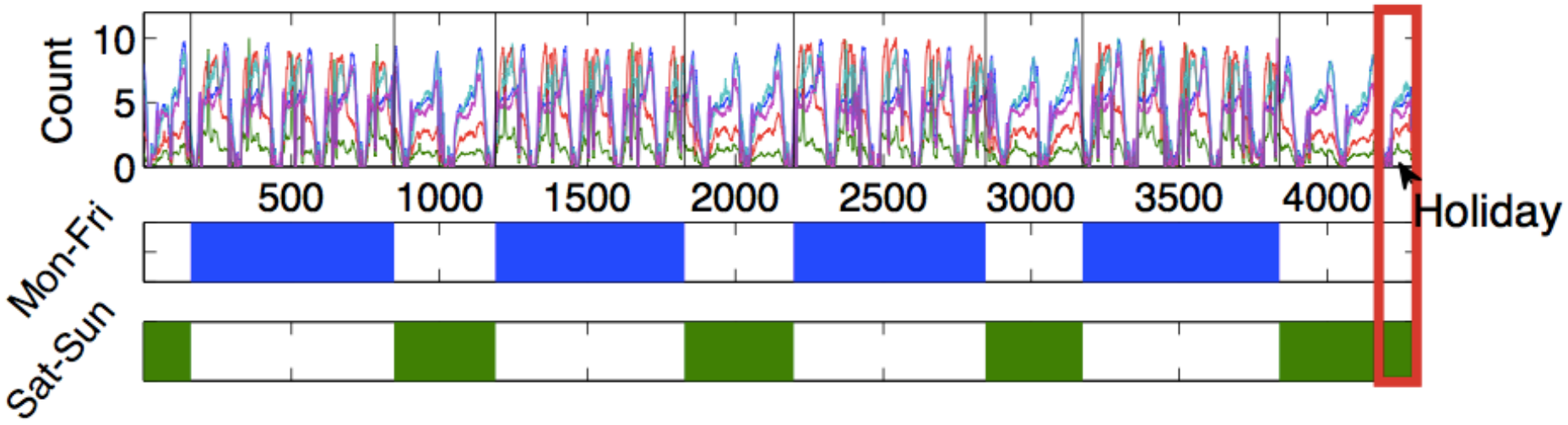
Web-click sequences (1 month, 5urls)



– 5urls: blog, news, dictionary, Q&A, mail
– every 10 minutes

App1. Model analysis (WebClick)

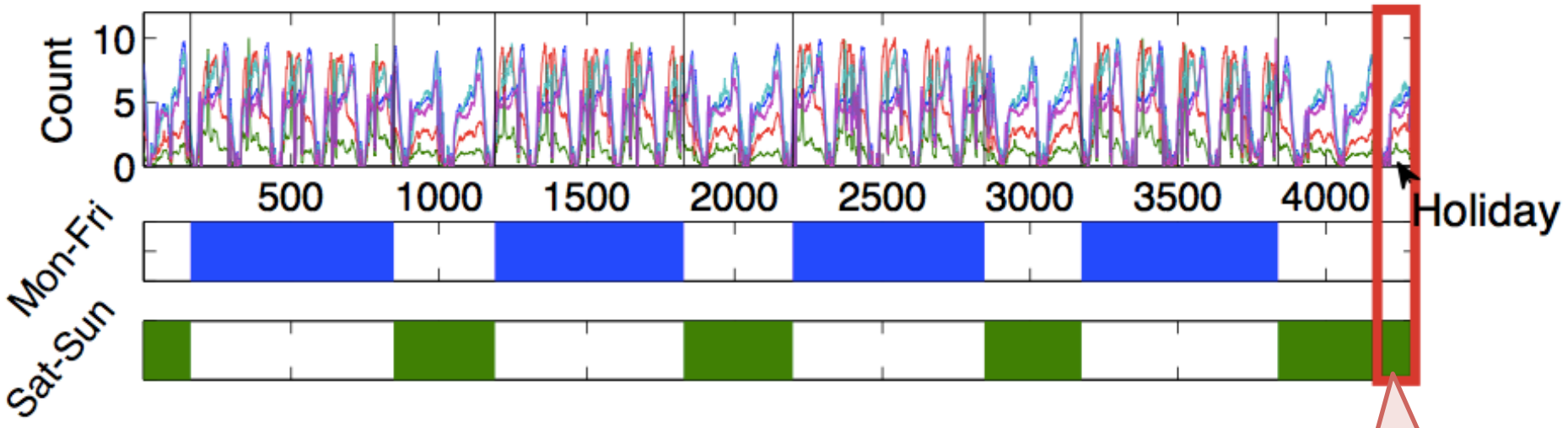
Web-click sequences (1 month, 5urls)



AutoPlait finds 2 patterns: weekday/weekend !

App1. Model analysis (WebClick)

Web-click sequences (1 month, 5urls)



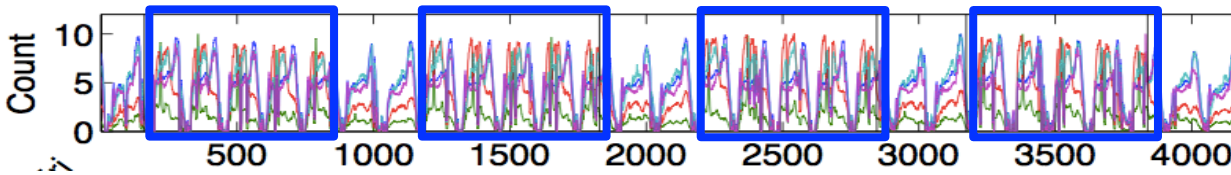
AutoPlait finds 2 patterns: weekday

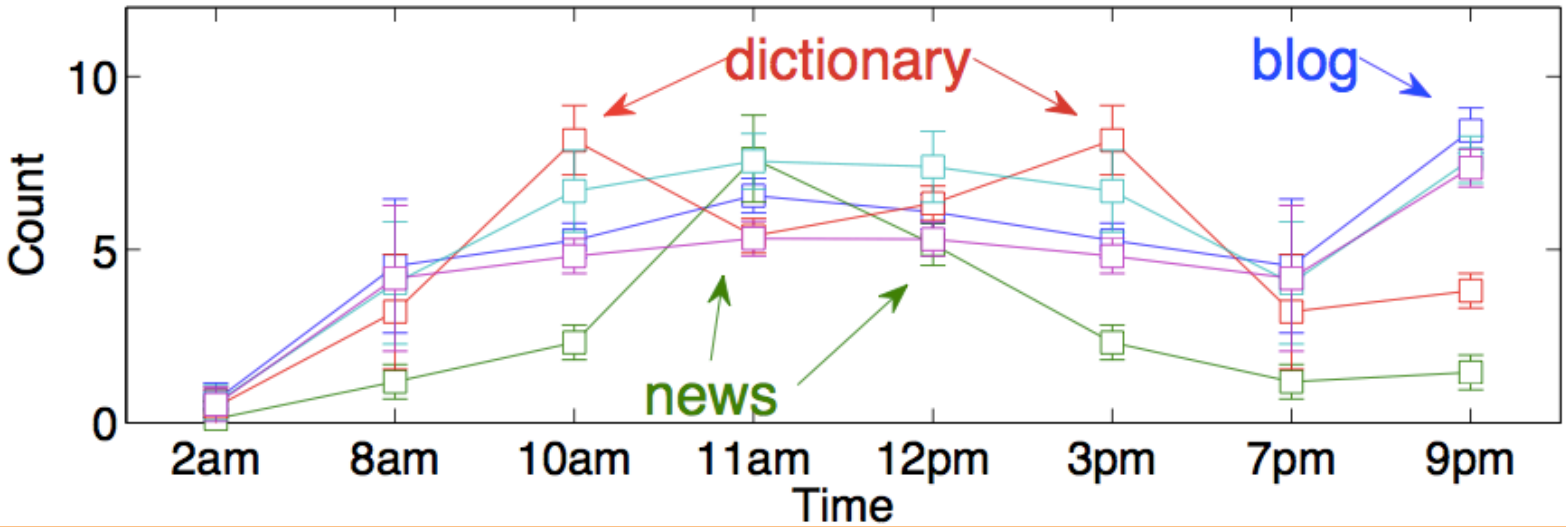
Monday
(but holiday)

App1. Model analysis (WebClick)

Details

Pattern of weekday regime



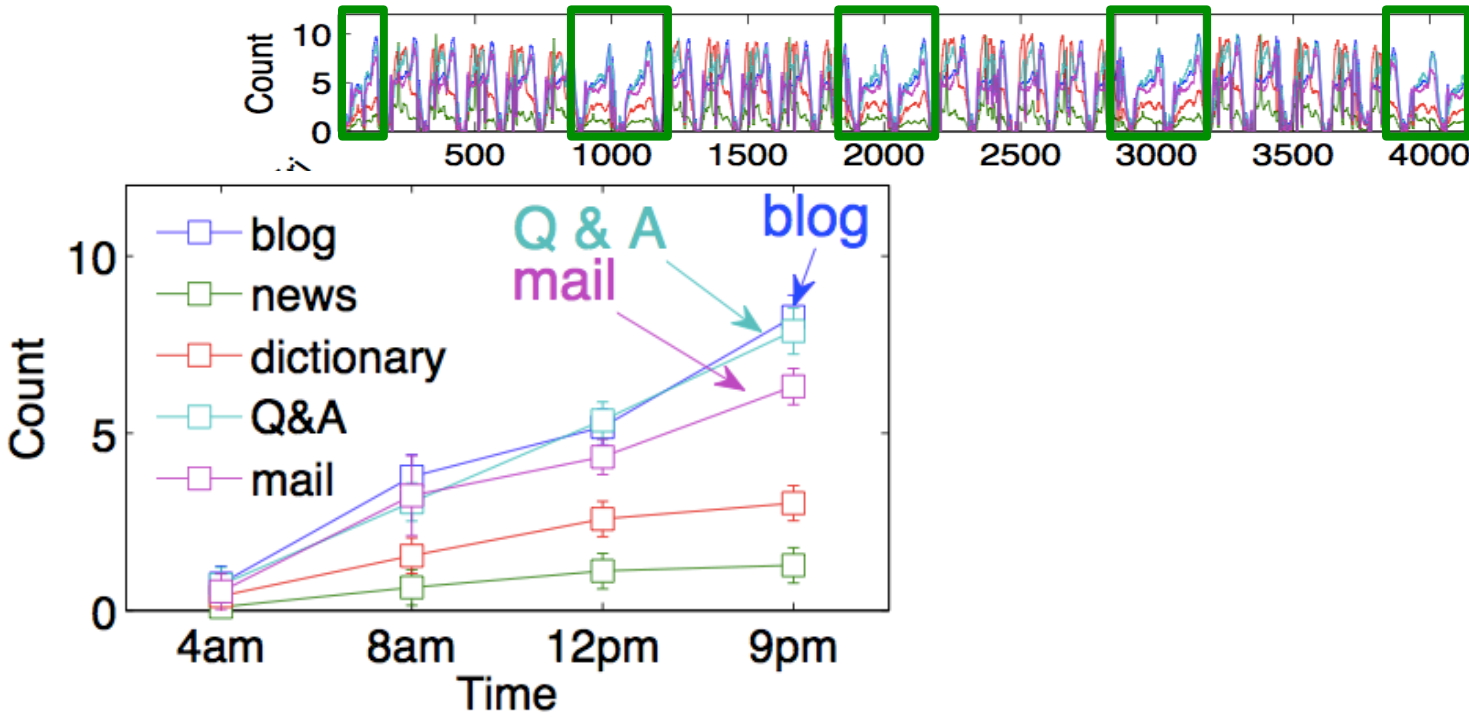


Observation: Working hard every weekday
(i.e., using dictionary, news sites)

App1. Model analysis (WebClick)

Details

Pattern of weekend regime



Observation: No more work on weekend
(i.e., blog, mail, Q&A for non-business purposes)

AutoPlait at work

AutoPlait is capable of various applications, e.g.,

App1. Model analysis

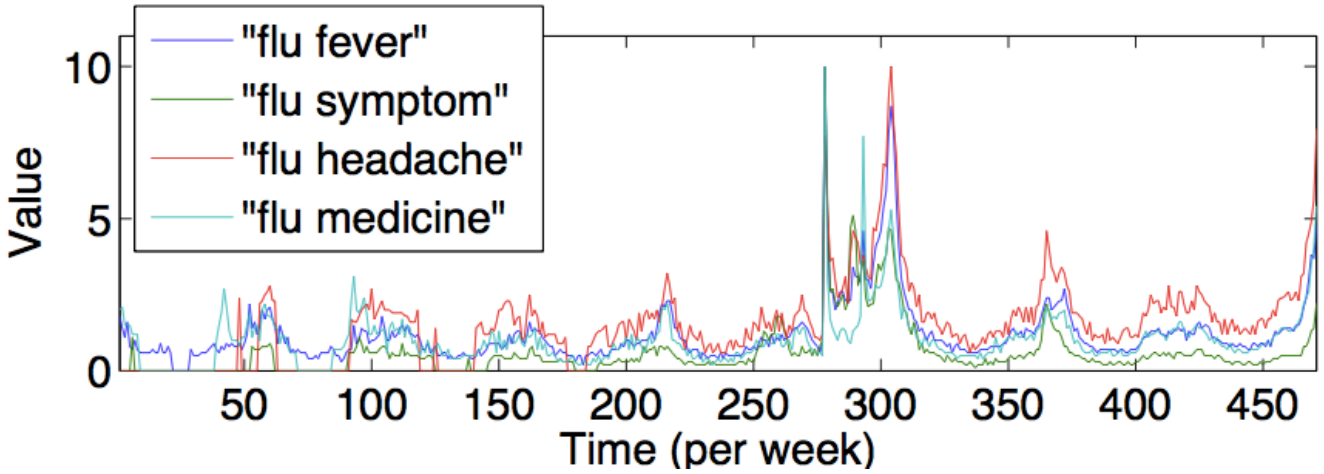
- Web-click sequences

App2. Event discovery

- Google Trend data

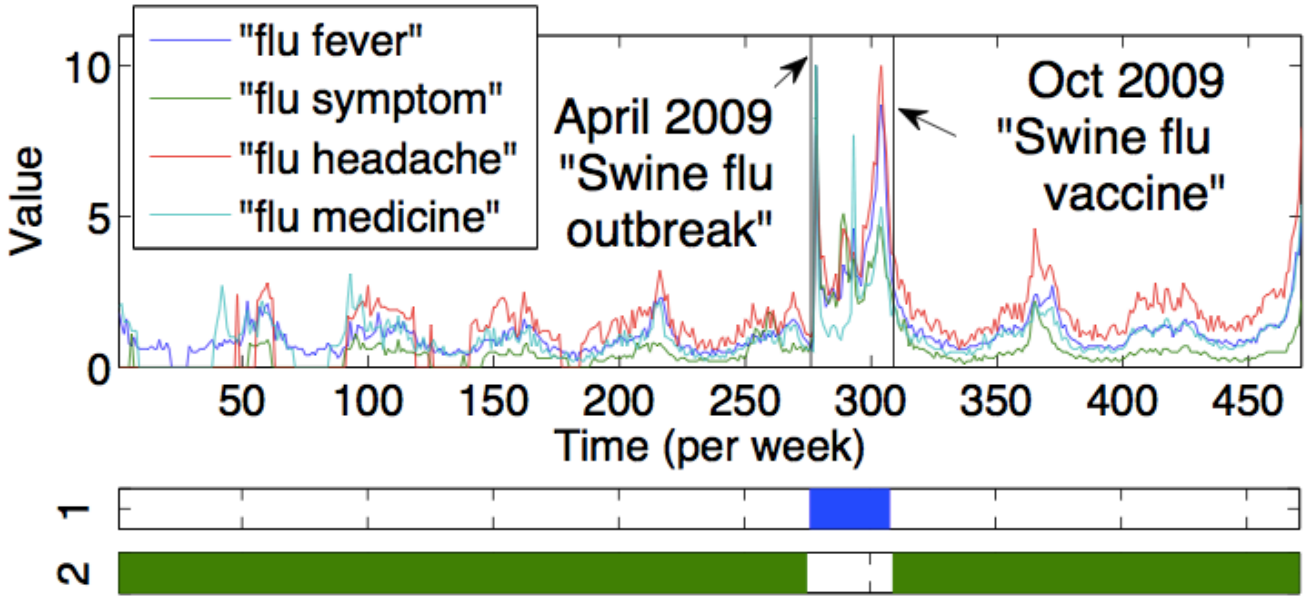
App2. Event discovery (GoogleTrend)

Anomaly detection (flu-related topics,10 years)



App2. Event discovery (GoogleTrend)

Anomaly detection (flu-related topics,10 years)

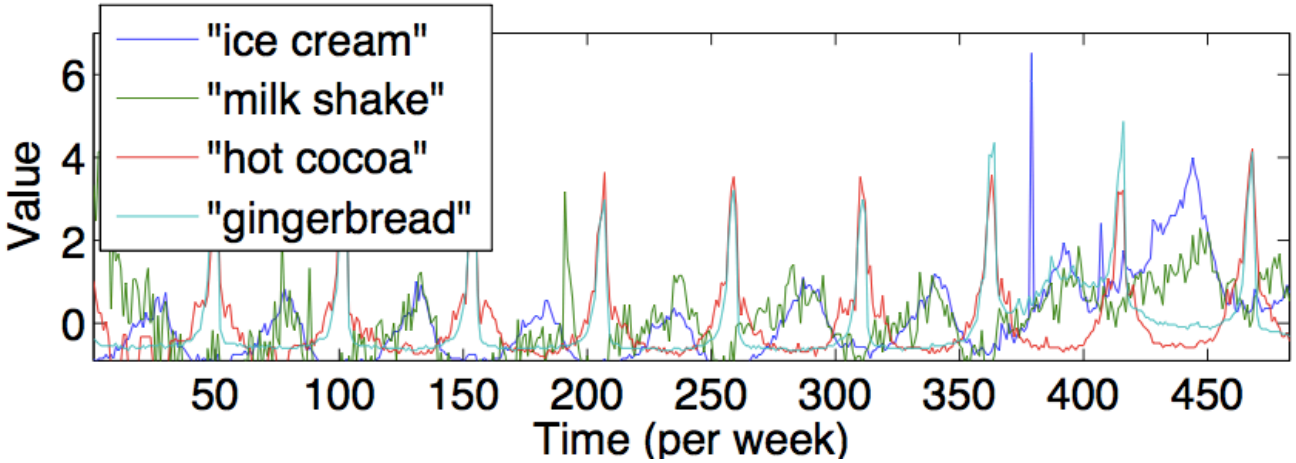


(a) Flu-related topics (regimes $r = 2$)

AutoPlait detects 1 unusual spike in 2009 (i.e., swine flu)

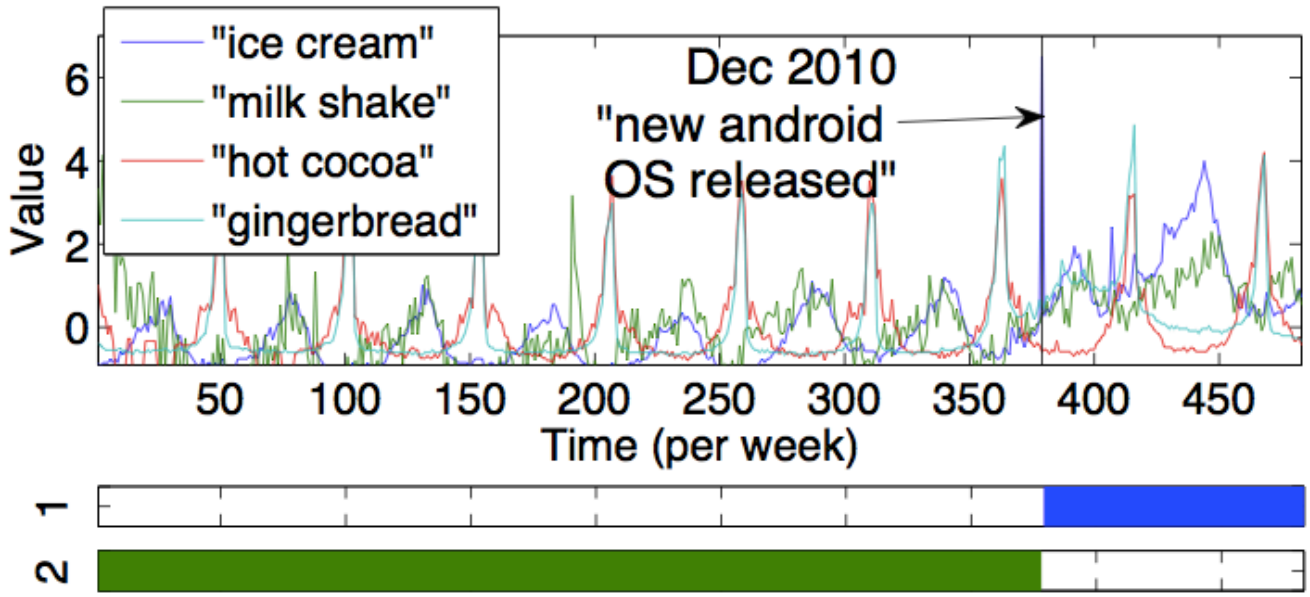
App2. Event discovery (GoogleTrend)

Turning point detection (seasonal sweets topics)



App2. Event discovery (GoogleTrend)

Turning point detection (seasonal sweets topics)

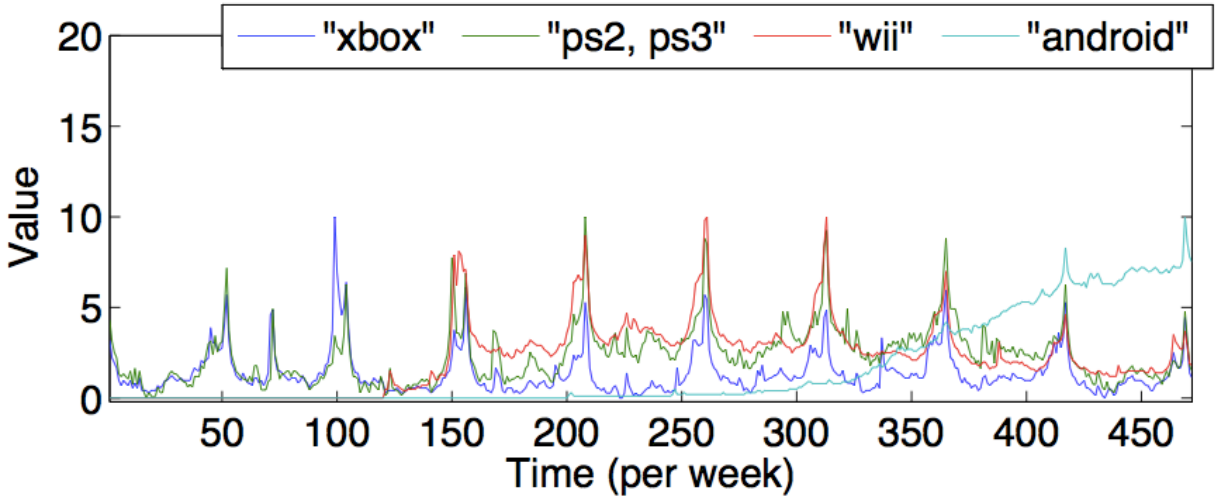


(b) Seasonal sweets topics (regimes $r = 2$)

Trend suddenly changed in 2010 (release of android OS "Ginger bread", "Ice Cream Sandwich")

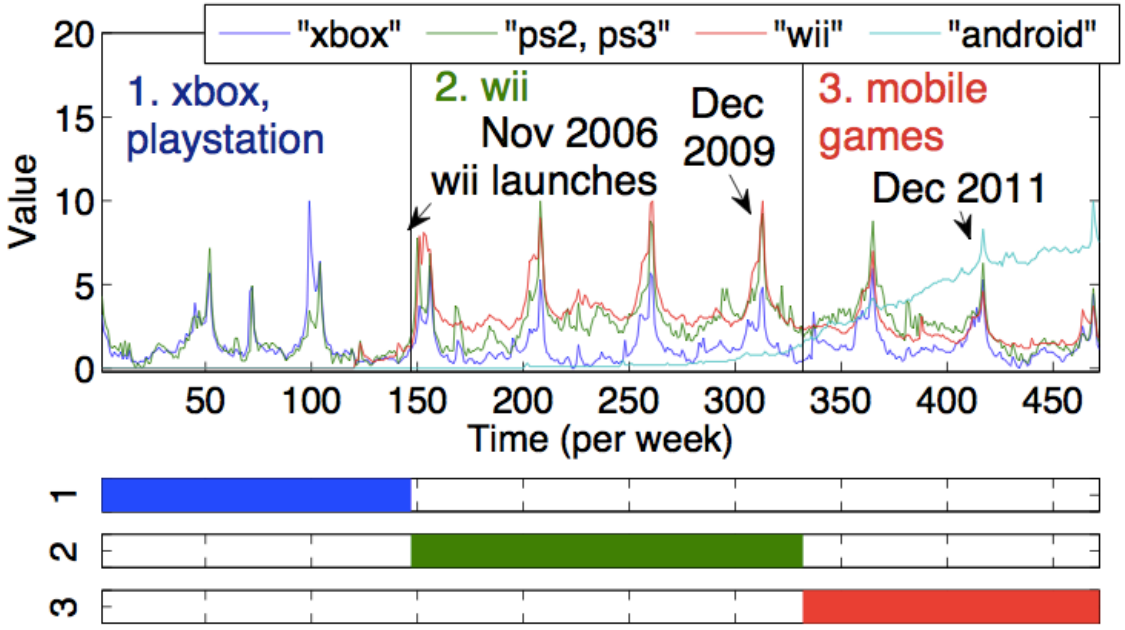
App2. Event discovery (GoogleTrend)

Trend discovery (game-related topics)



App2. Event discovery (GoogleTrend)

Trend discovery (game-related topics)

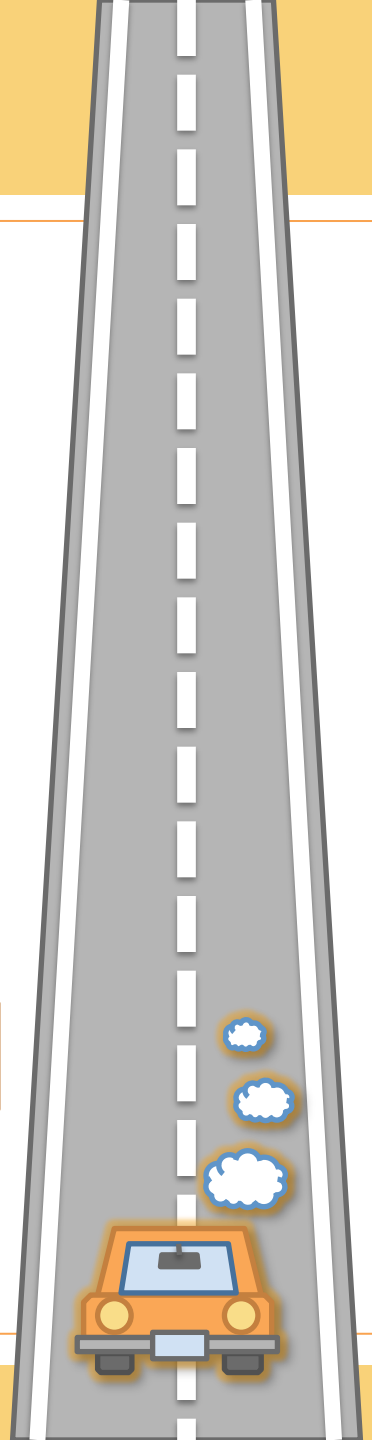


(c) Game-related topics (regimes $r = 3$)

It discovers 3 phases of "game console war" (Xbox&PlayStation/Wii/Mobile social games)

Outline

- Motivation
- Problem definition
- Compression & summarization
- Algorithms
- Experiments
- AutoPlait at work
- Conclusions



Conclusions

AutoPlait has the following properties

- Effective ✔

  Find optimal segments/regimes

- Sense-making ✔

  Reasonable regimes

- Fully-automatic ✔

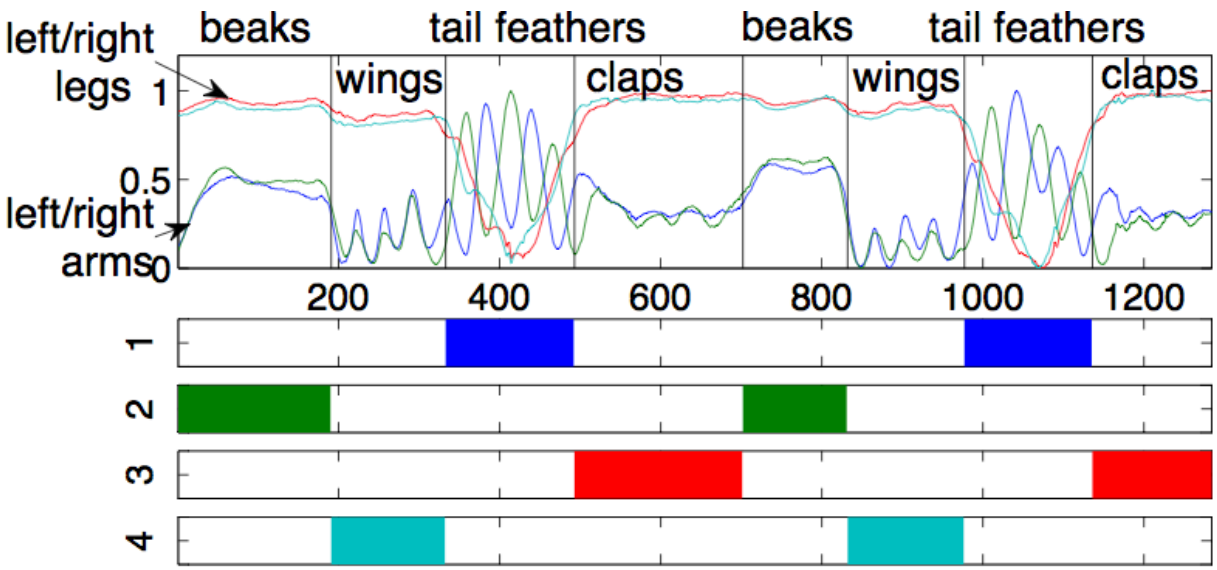
  No magic numbers

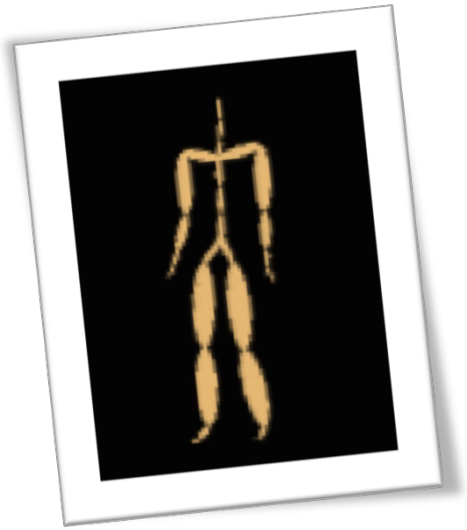
- Scalable ✔

  It scales linearly

Thank you!







Code: http://www.cs.kumamoto-u.ac.jp/~yasuko/software.html

Mail: yasuko@cs.kumamoto-u.ac.jp