

ICDM 2019

19TH IEEE International Conference on Data Mining

Multi-Aspect Mining of Complex Sensor Sequences

Takato Honda¹, Yasuko Matsubara¹, Ryo Neyama²,
Mutsumi Abe², Yasushi Sakurai¹

¹AIRC-ISIR, Osaka University

²Toyota Motor Corporation



8-11 November 2019
Beijing, China

© 2019 Takato Honda et al.

Motivation

Analysis of IoT sensor data, e.g., car
- Advanced driving assistance service

Risk



Fuel

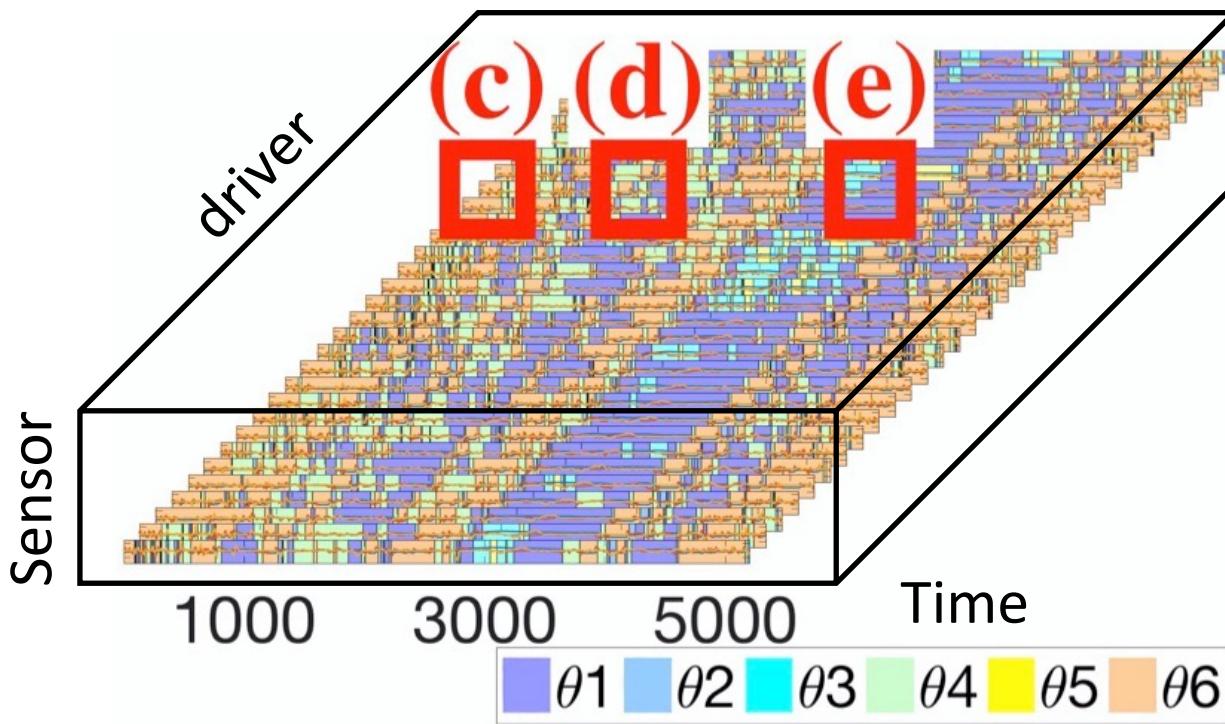


Congestion

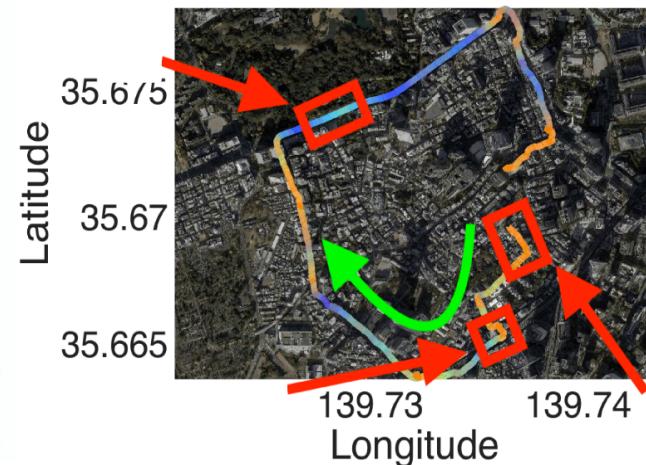


Motivation

IoT sensor data is a tensor
(sensor × driver × time)



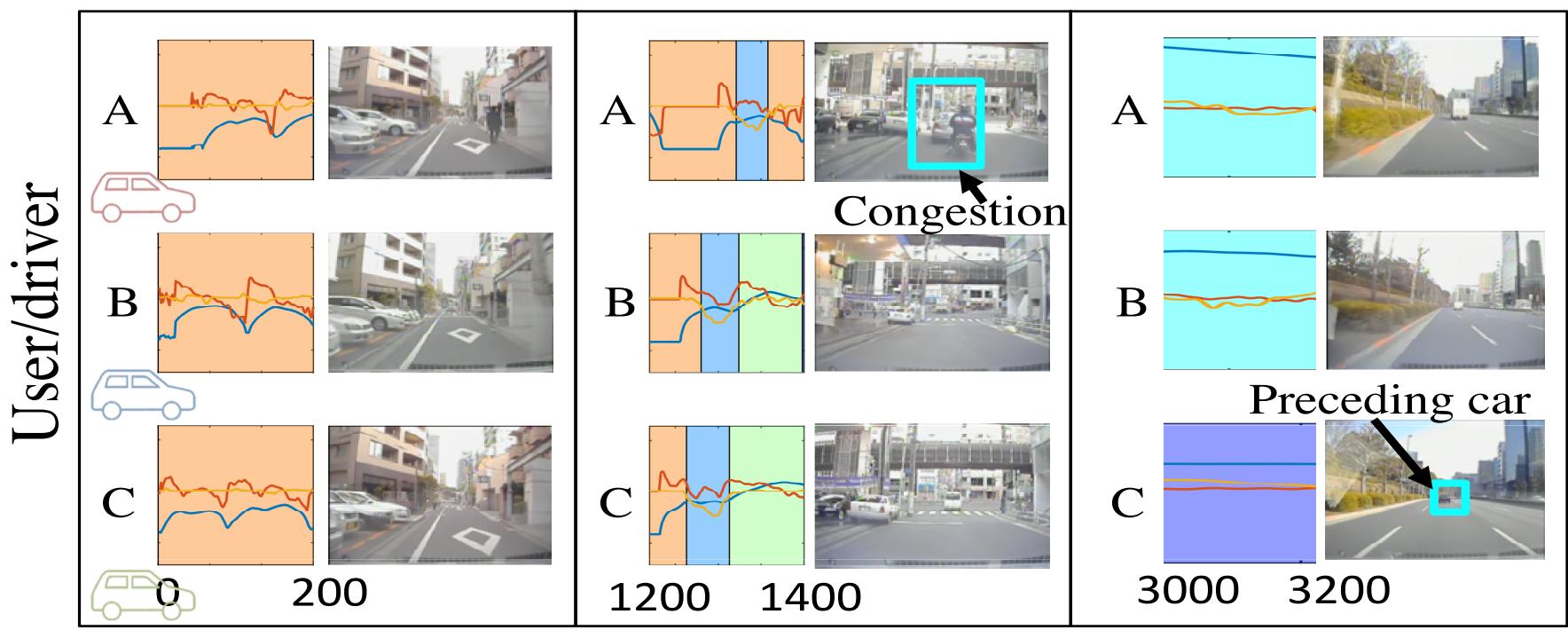
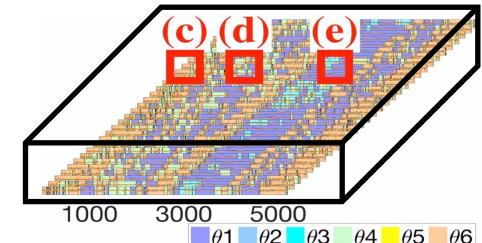
(a) Time series tensor of automobile dataset



(b) On a map

Motivation

Tensor has multi-aspect patterns:
time-aspect and user-aspect



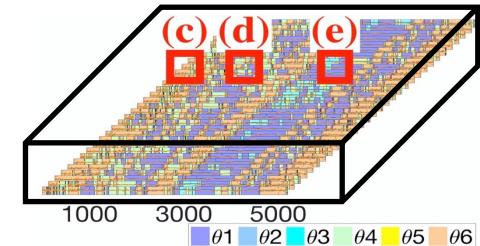
(c) Narrow road

(d) Intersection

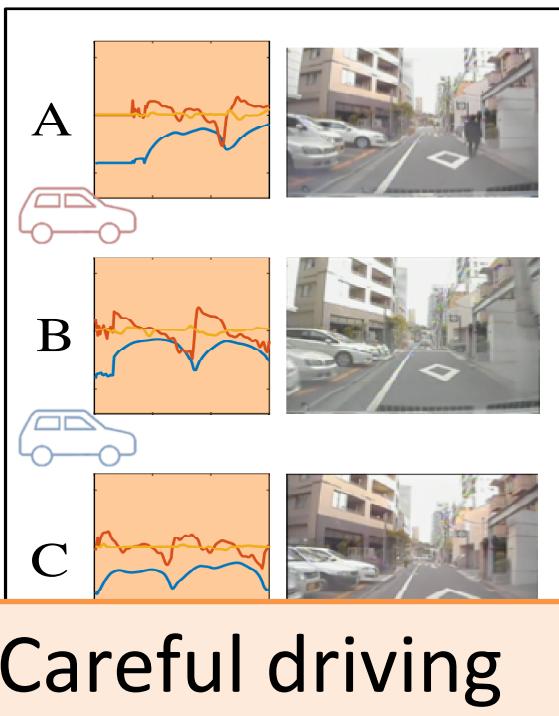
(e) Wide road

Motivation

Tensor has multi-aspect patterns:
time-aspect and user-aspect

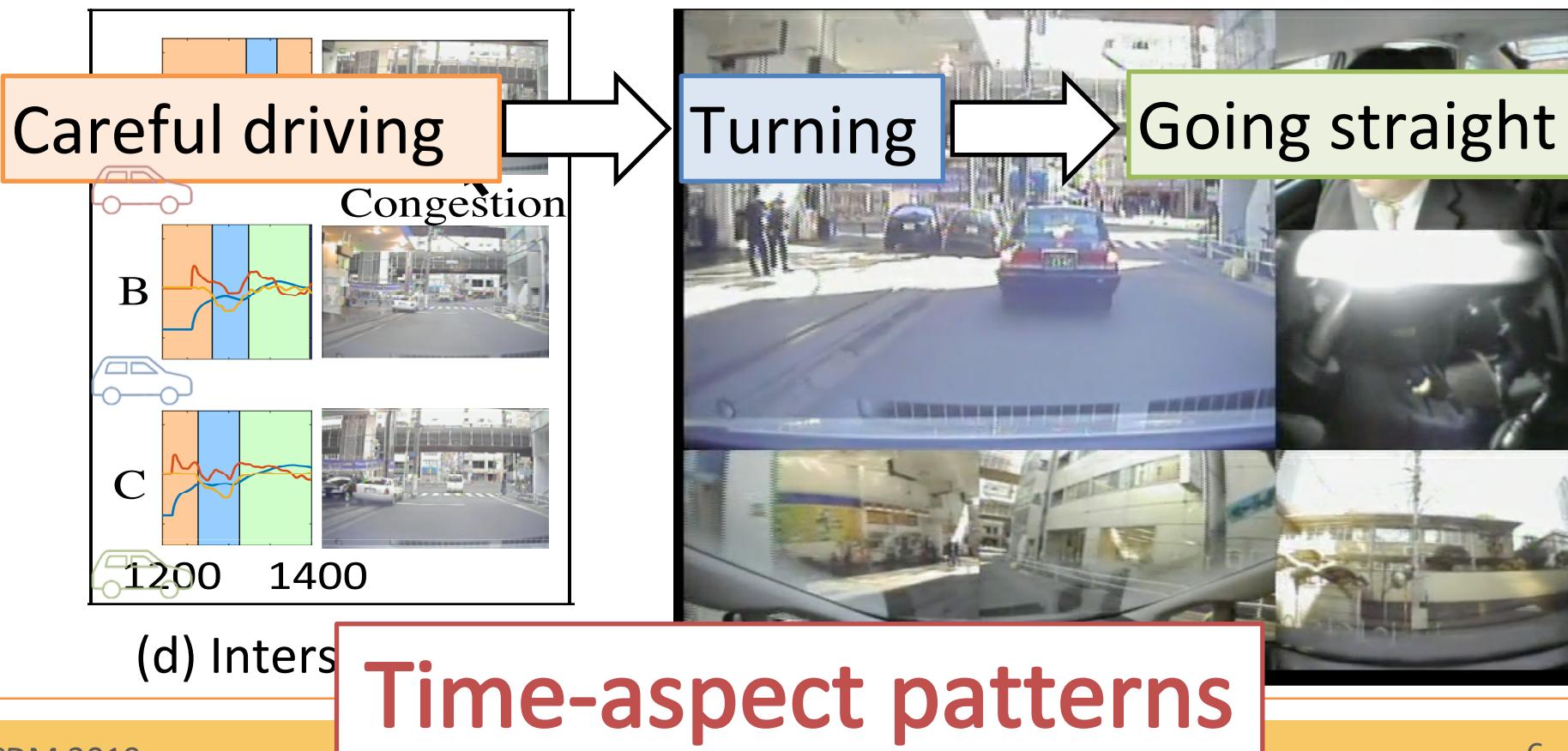
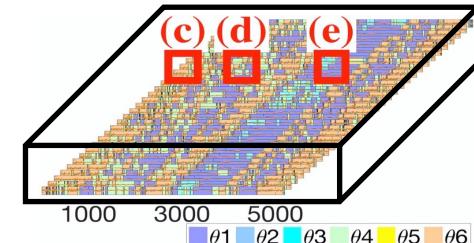


User/driver



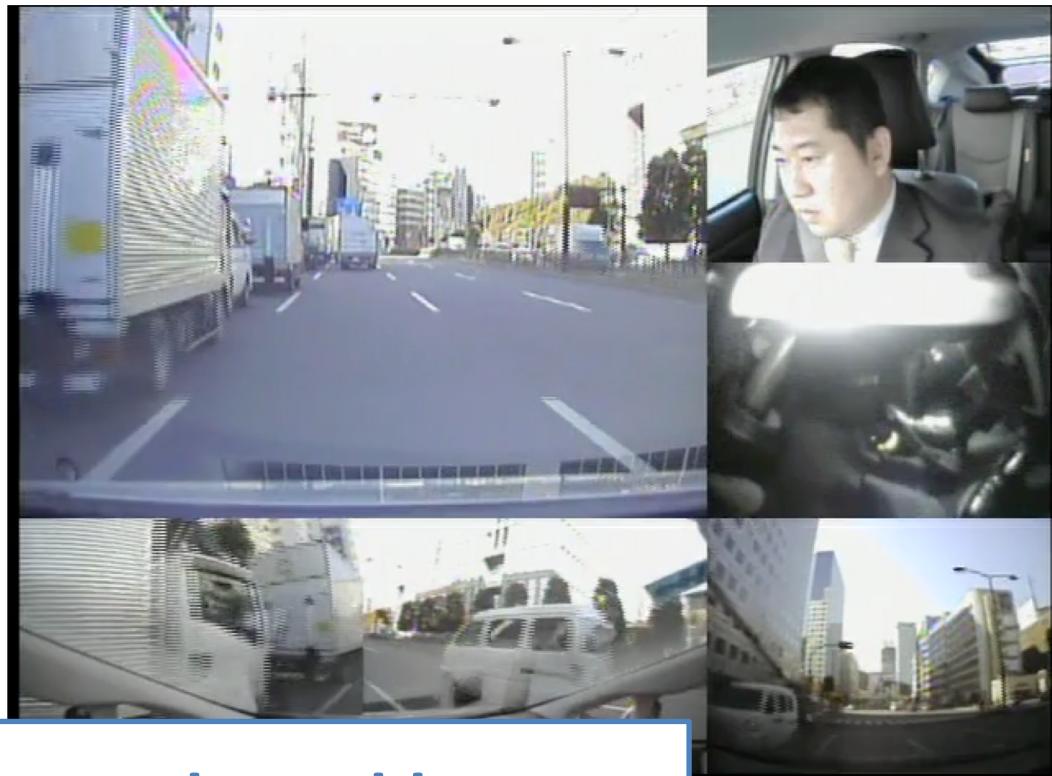
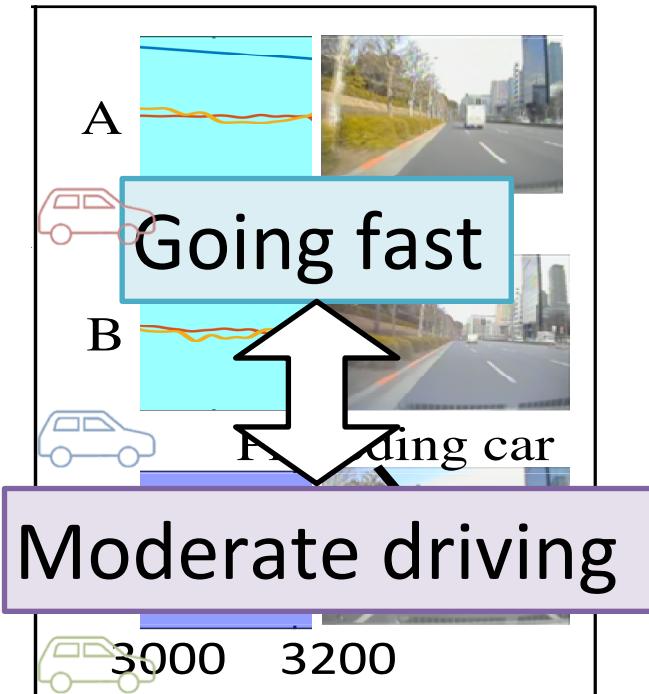
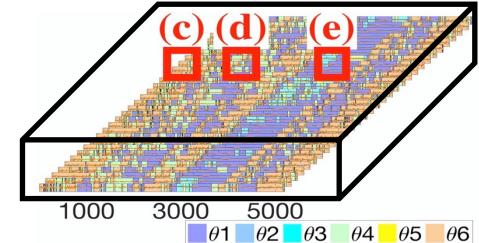
Motivation

Tensor has multi-aspect patterns:
time-aspect and **user-aspect**



Motivation

Tensor has multi-aspect patterns:
time-aspect and **user-aspect**

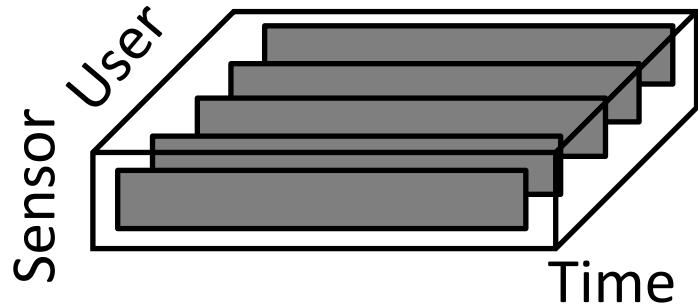
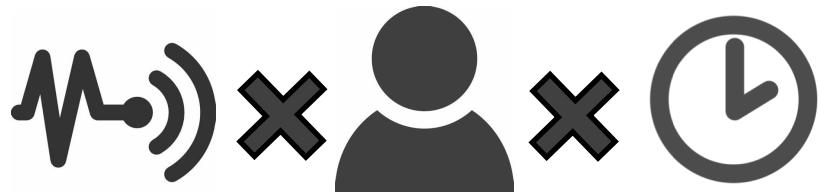


(e) Wide

User-aspect patterns

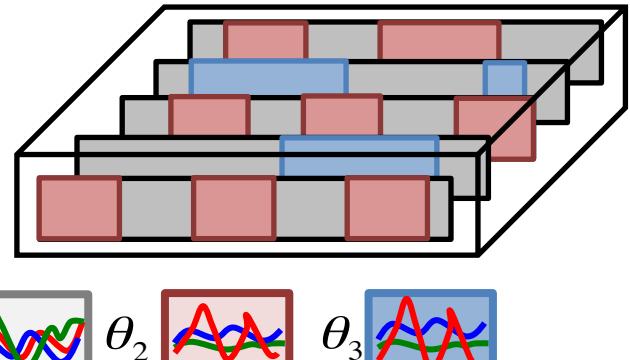
Motivation

Given: Time-series tensor
(sensor × user × time)



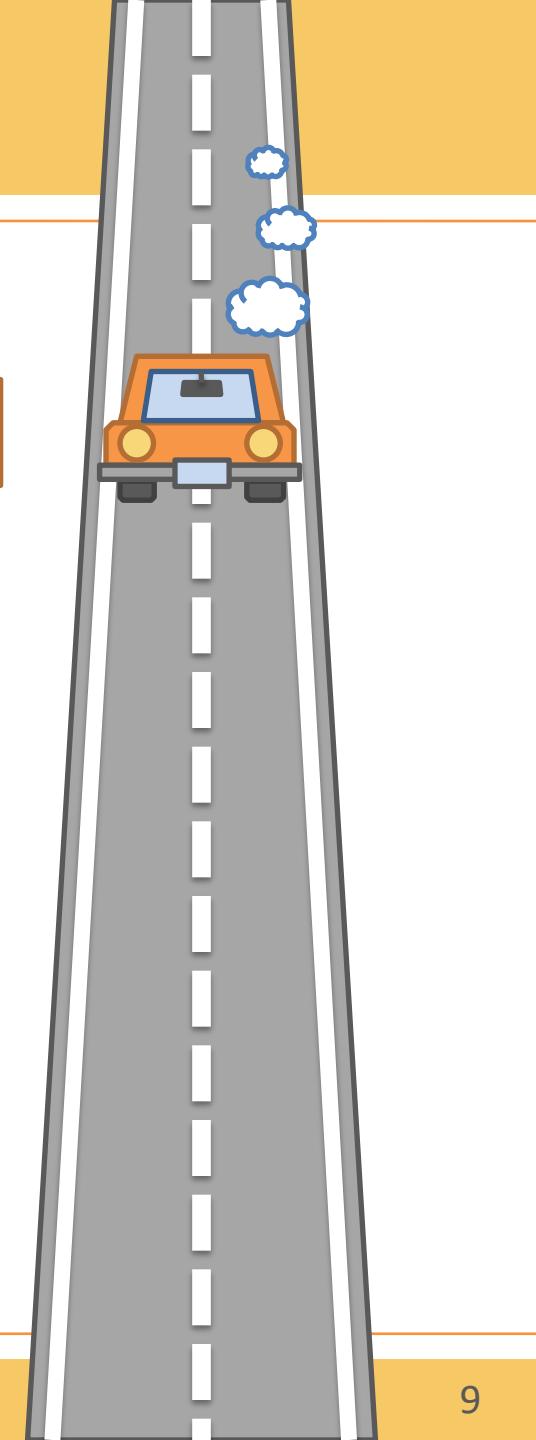
Find: Multi-aspect patterns
(time and user-aspect)

Automatically & quickly



Outline

- Motivation
- Problem definition
- Main ideas
- Algorithms
- Experiments
- Conclusions



Problem definition

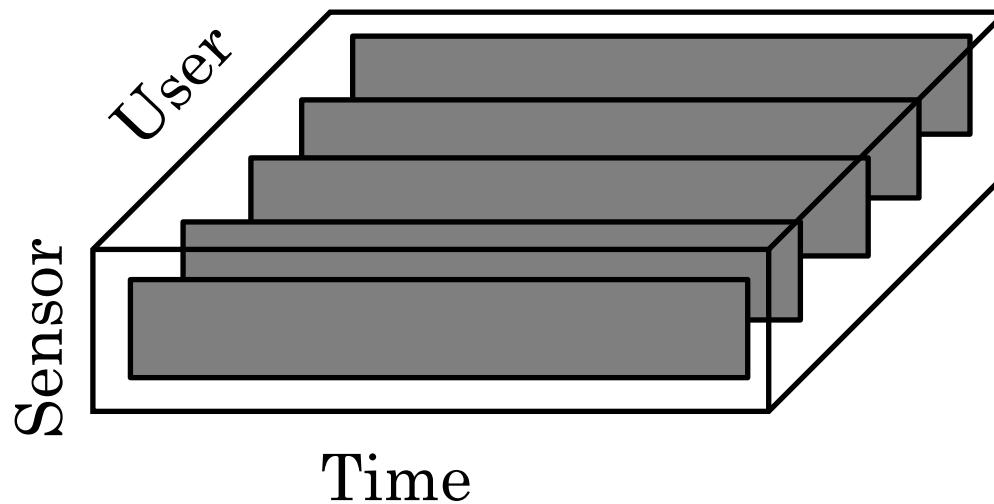
Key concepts

- **Tensor:** x given
- **Segment:** S hidden
- **Regime:** Θ hidden
- **Segment-membership:** F hidden

Problem definition

Tensor : $\mathcal{X} \in R^{d \times w \times n} = \{X_1, \dots, X_w\}$

given



Problem definition

Segment : $S = \{S_1, \dots, S_m\}$

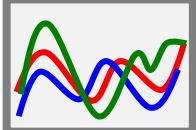
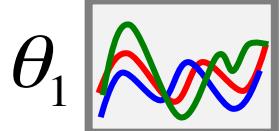
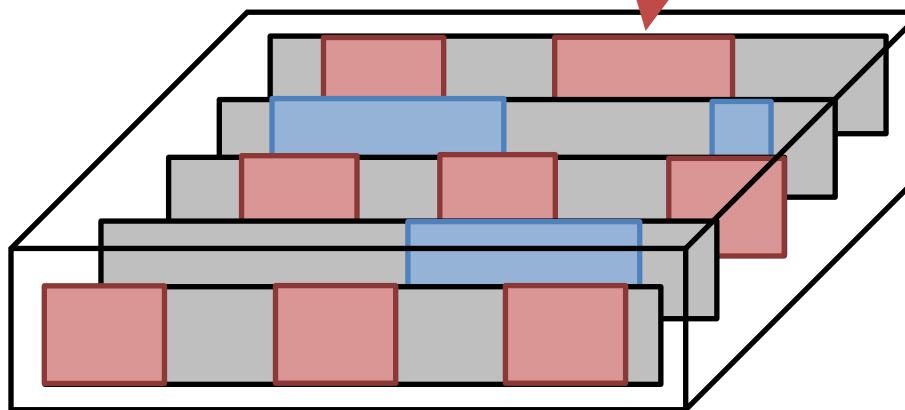
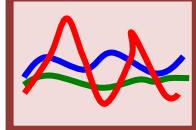
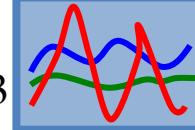
hidden

$s_i = \{t_s, t_e, userID\}$

start
position

end
position

$m = 25$ segments

 θ_2  θ_3 

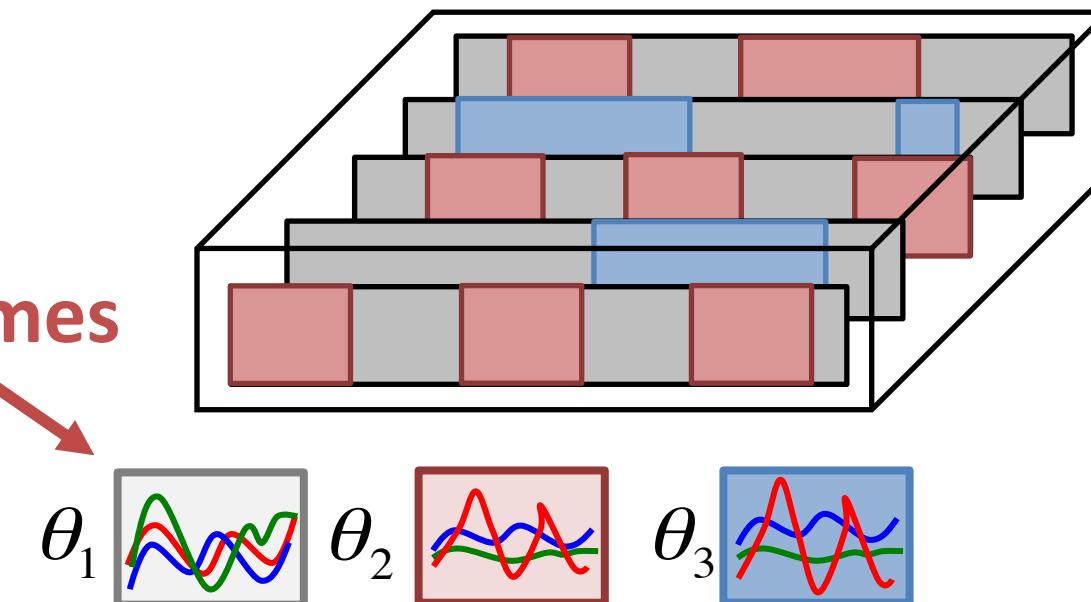
Problem definition

Regime: $\Theta = \{\theta_1, \theta_2, \dots, \theta_r, \Delta_{r \times r}\}$

hidden

$\theta_i = \{\pi, A, B\}$ (hidden Markov model)
Initial prob. transition prob. output prob.

$r = 3$ regimes



Problem definition

Membership:

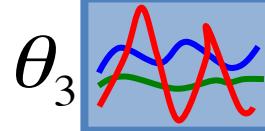
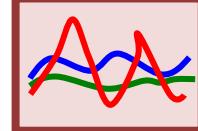
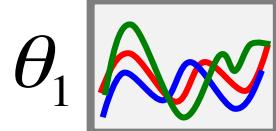
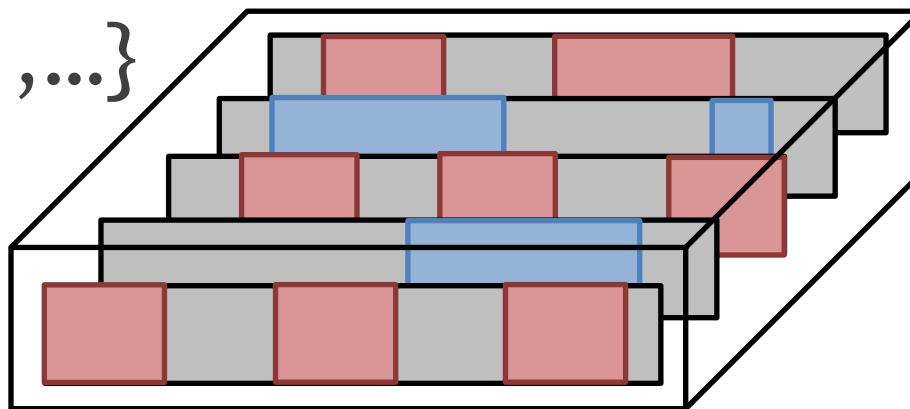
$$F = \{f_1, f_2, \dots, f_m\}$$

hidden

$$1 \leq f_i \leq r$$

Example:

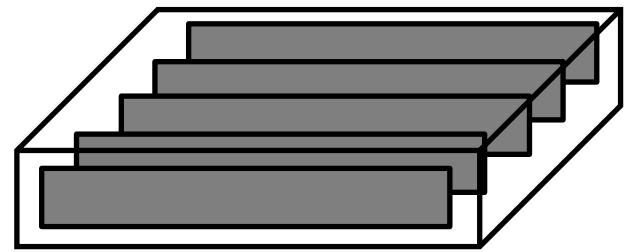
$$F = \{1, 2, 1, 2, 1, \dots\}$$



Problem definition

Given: tensor \mathcal{X}

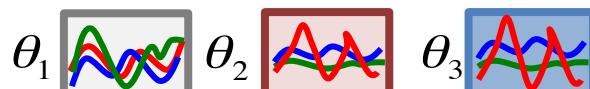
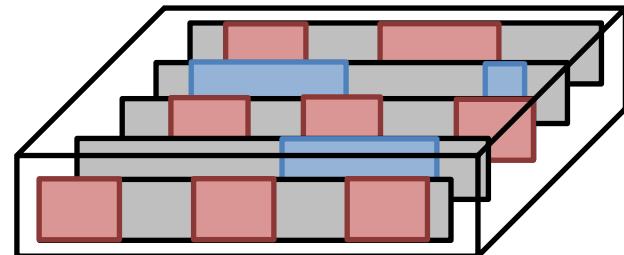
$$\mathcal{X} = \{X_1, \dots, X_w\}$$



Find: compact description C of \mathcal{X}

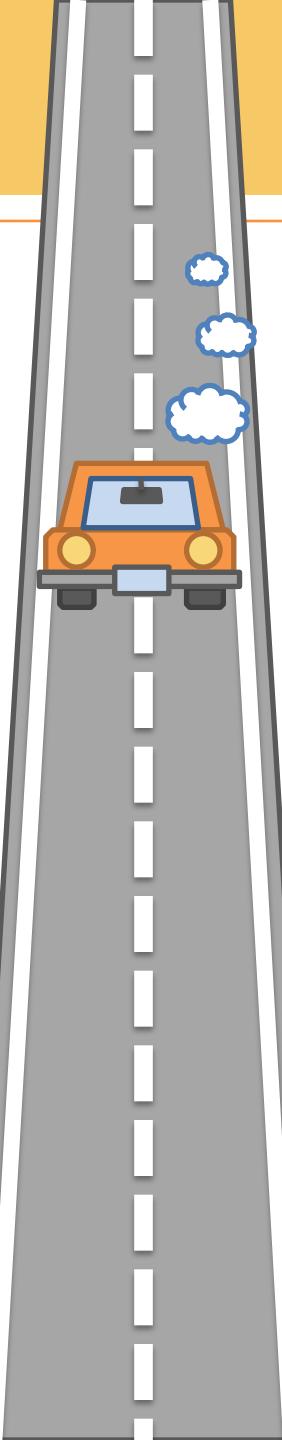
$$C = \{m, r, S, \Theta, F\}$$

Automatically & quickly



Outline

- Motivation
- Problem definition
- Main ideas
- Algorithms
- Experiments
- Conclusions



Main ideas

Goal: compact description of

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

without user intervention

Challenges:

Q1. How to decide m and r **automatically**

Q2. How to find **multi-aspect regimes**

Main ideas

Goal: compact description of

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

without user intervention

Challenges:

Q1. How to decide m and r automatically

Idea 1: Model description cost

Q2. How to find multi-aspect regimes

Idea 2: Multi-splitting algorithm

(1): model description cost

Q1. How to decide # of regimes/segments?

Idea 1: Model description cost

- Minimize coding cost
- Optimal # of segments/regimes

Good
compression



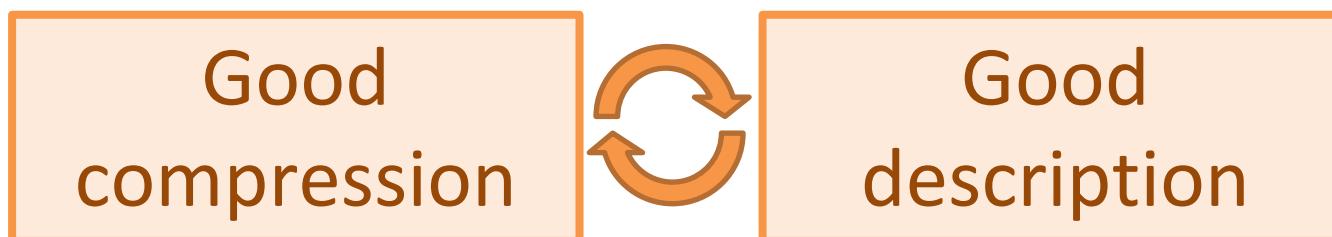
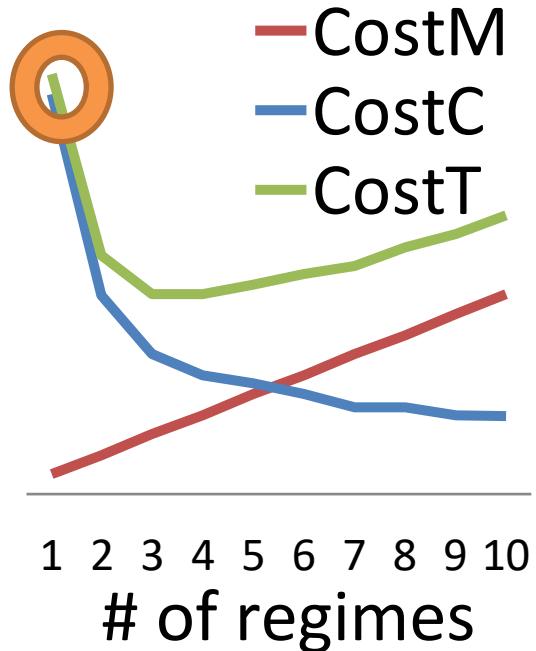
Good
description

(1): model description cost

Idea: Minimize total cost

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

Model cost **Coding cost**

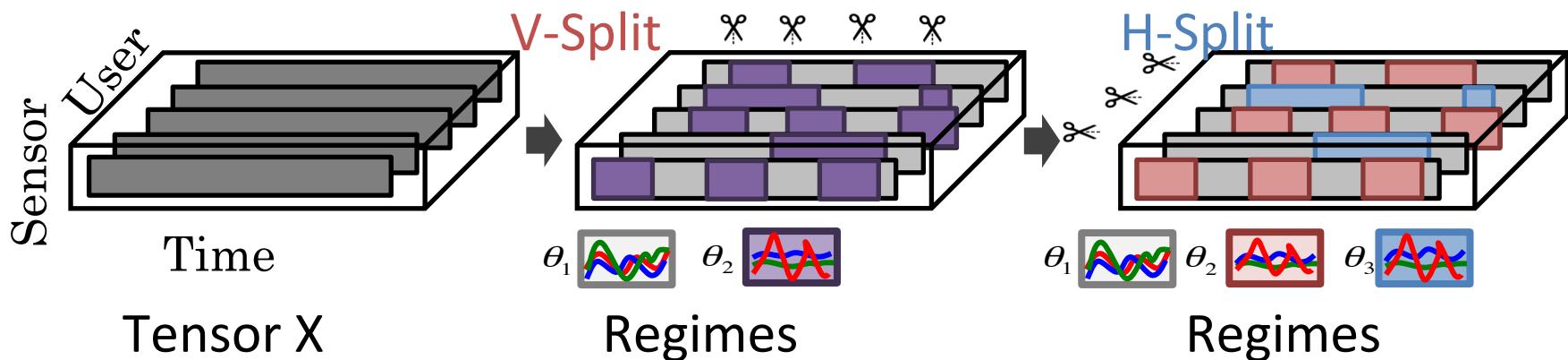


(2): Multi-aspect mining

Q2. How to find multi-aspect regimes?

Idea 2: Multi-aspect splitting algorithm

- Find **time-aspect** transitions
- And their differences between **users**



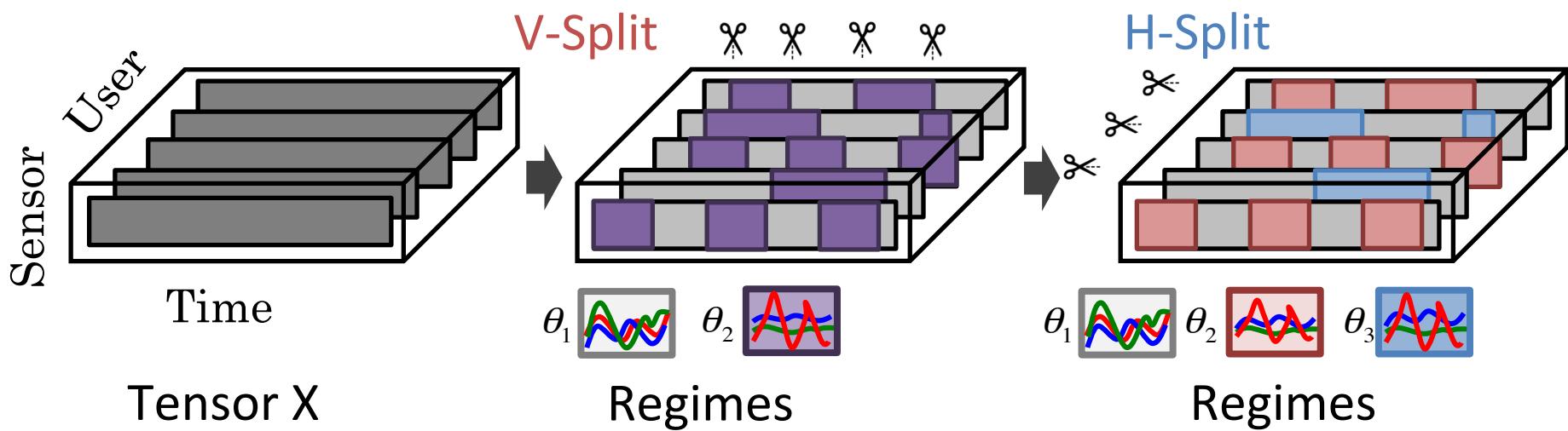
(2): Multi-aspect mining

V-Split (vertical):

split \mathcal{X} into time-aspect

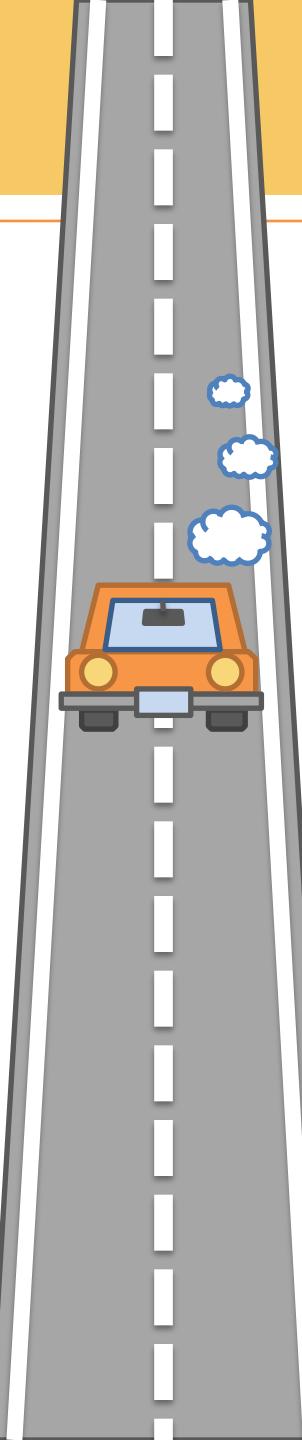
H-Split (horizontal):

split \mathcal{X} into user-aspect



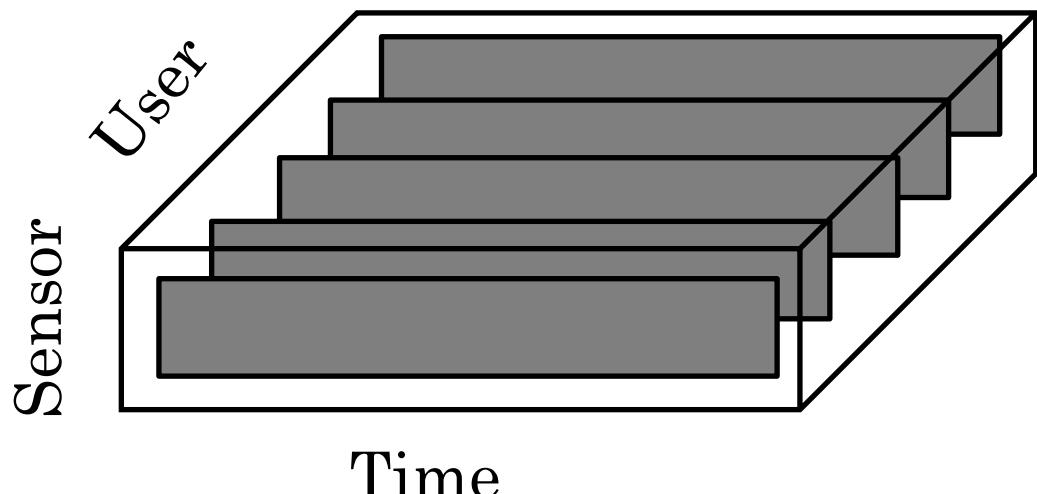
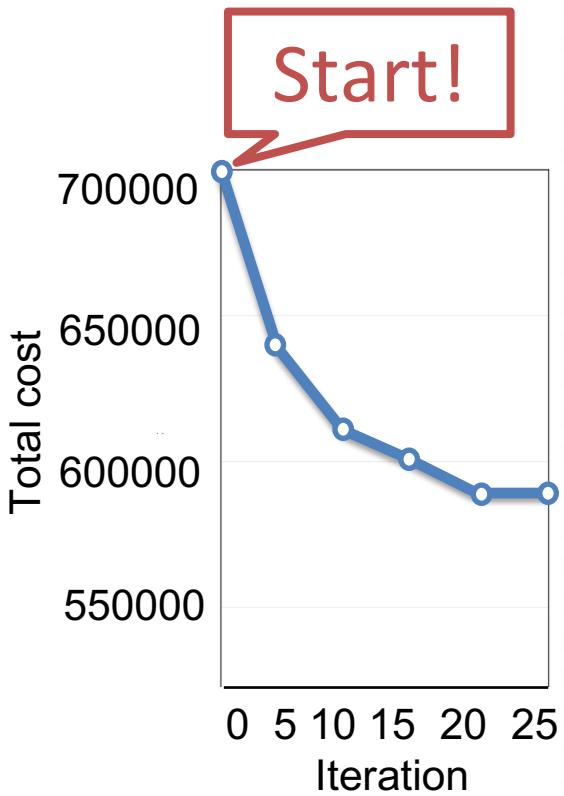
Outline

- Motivation
- Problem definition
- Main ideas
- Algorithms
- Experiments
- Conclusions



Proposed algorithm

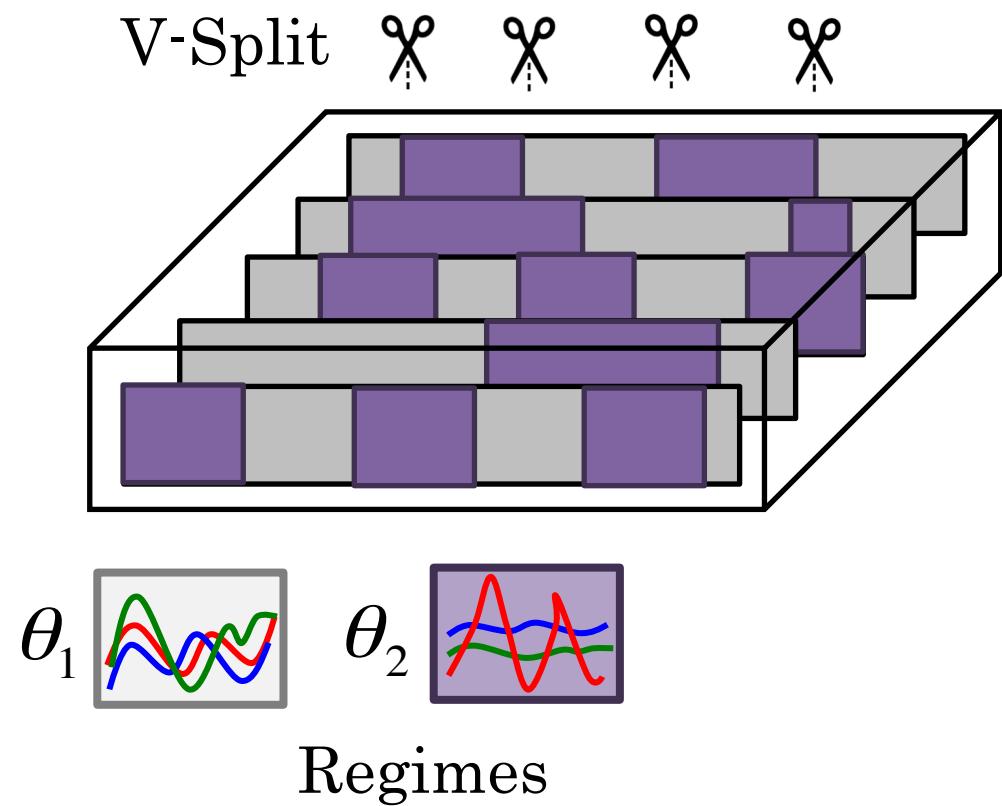
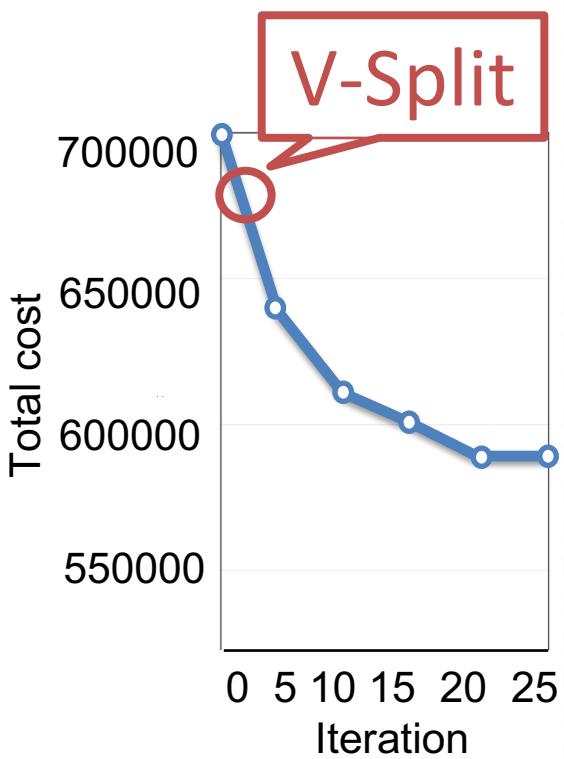
Overview



Iteration 0 ($r=1$)

Proposed algorithm

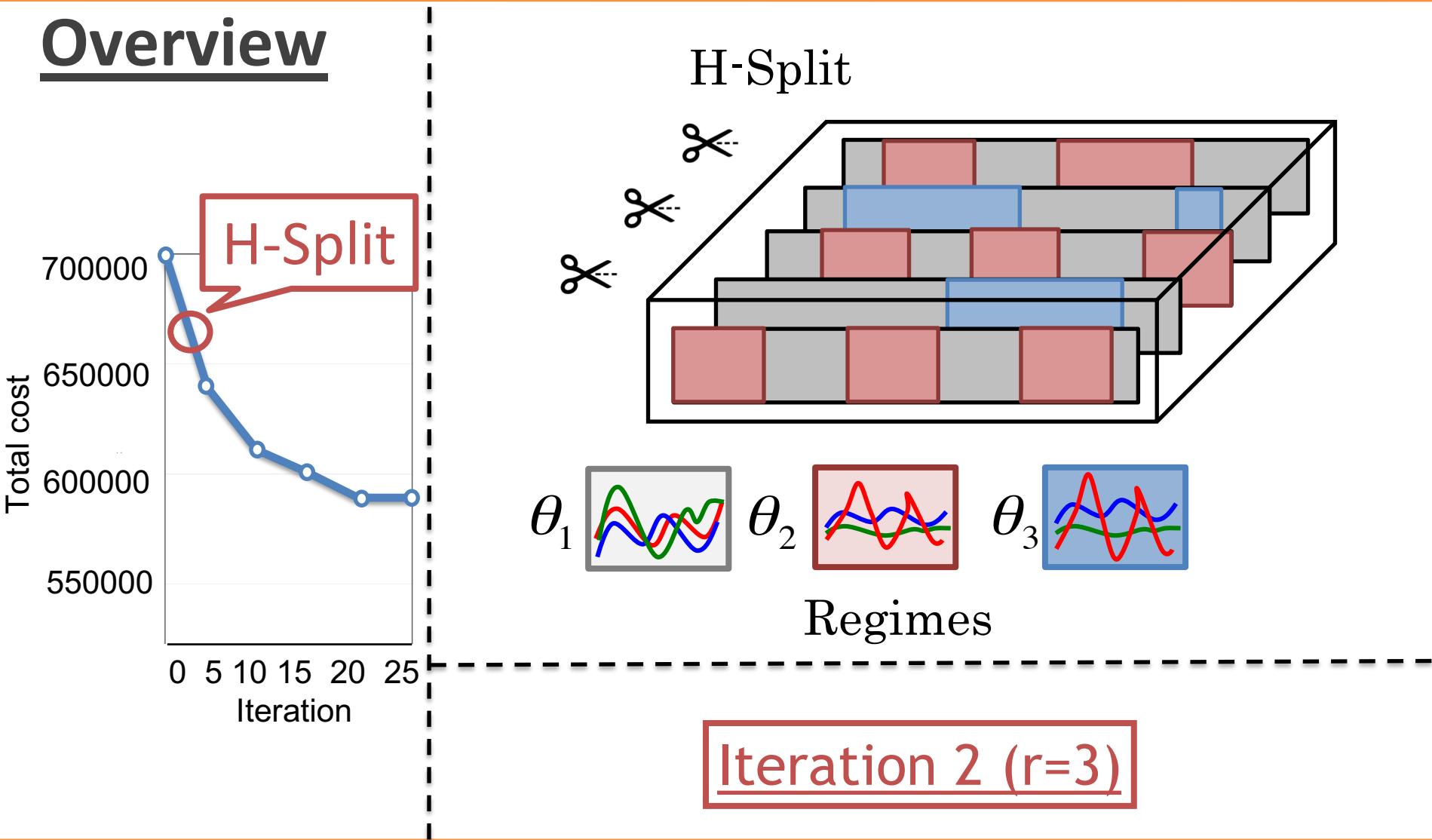
Overview



Iteration 1 ($r=2$)

Proposed algorithm

Overview



Algorithms

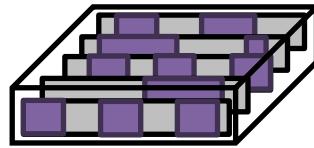
Algorithms of our method

CubeMarker

V-Split

Inner loop

- V-Assignment
- ModelEstimation



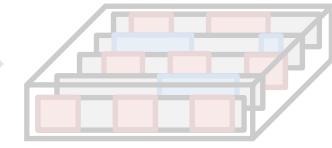
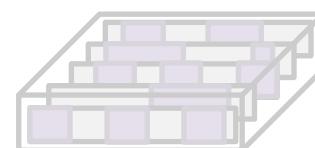
Find time-aspect regime

Outer loop

H-Split

Inner loop

- H-Assignment
- ModelEstimation



Find user-aspect regime

Decide splitting algorithm

V-Split

Inner loop

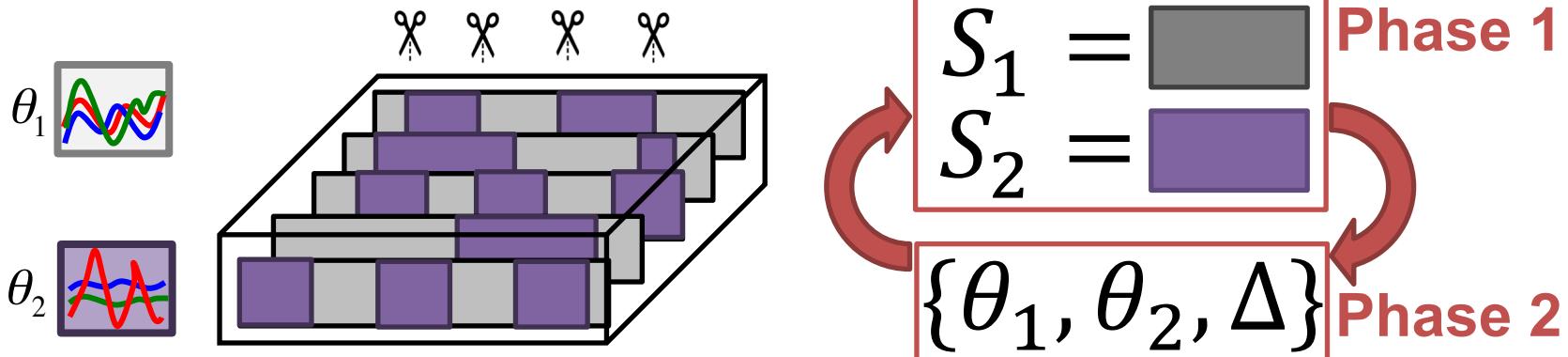
Two phase iterative approach

- Phase 1: (V-Assignment)

- Split segments into two groups: S_1, S_2

- Phase 2: (ModelEstimation)

- Update model parameters: $\Theta = \{\theta_1, \theta_2, \Delta\}$



V-Assignment

Inner loop

Given:

- tensor \mathcal{X}

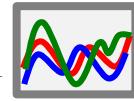
- model parameter set

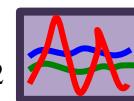
$$\Theta = \{\theta_1, \theta_2, \Delta\}$$

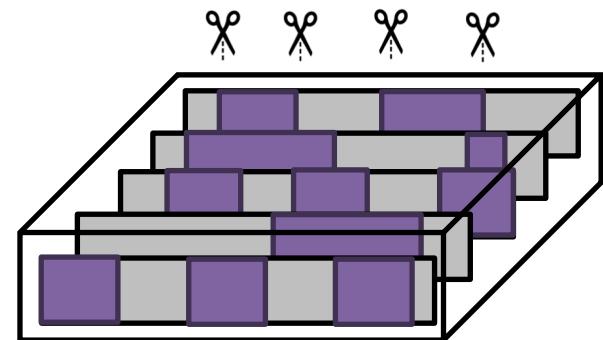
Find: segment set S_1, S_2

$$\{S_1, S_2\} = \operatorname{argmax} P(\mathcal{X}|S_1, S_2, \Theta)$$

$$\left. \begin{array}{c} \mathcal{X} \\ \Theta = \{\theta_1, \theta_2, \Delta\} \end{array} \right\}$$

θ_1 

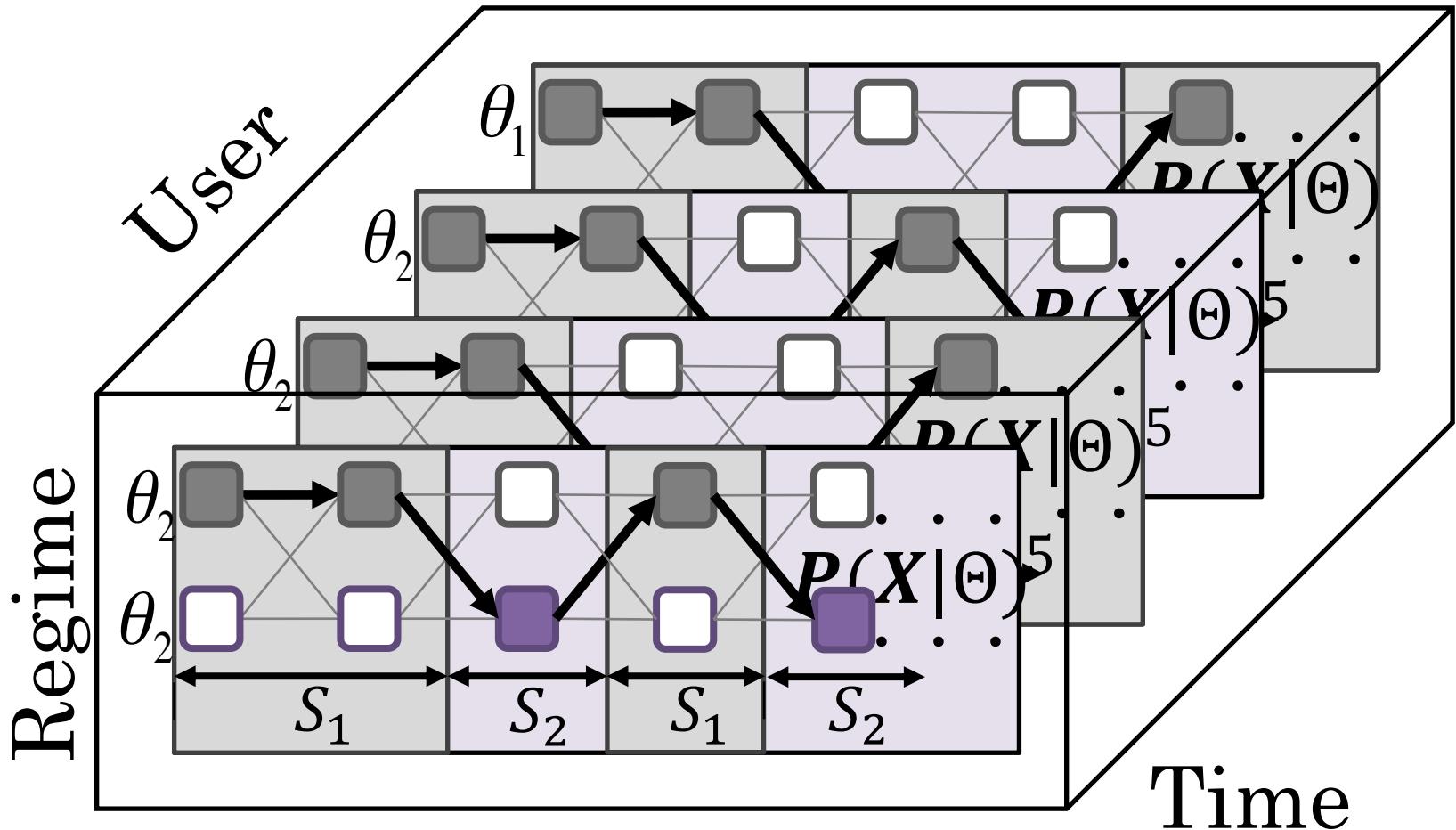
θ_2 



V-Assignment

Details

Inner loop



Algorithms

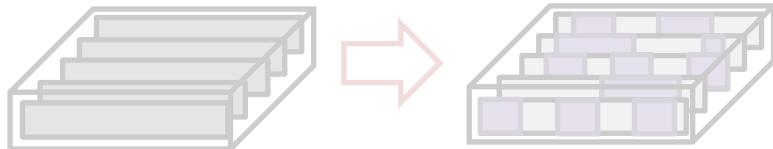
Algorithms of our method

CubeMarker

V-Split

Inner loop

- V-Assignment
- ModelEstimation



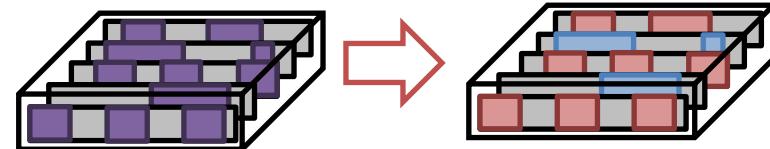
Find time-aspect regime

Outer loop

H-Split

Inner loop

- H-Assignment
- ModelEstimation



Find user-aspect regime

Decide splitting algorithm

H-Split

Inner loop

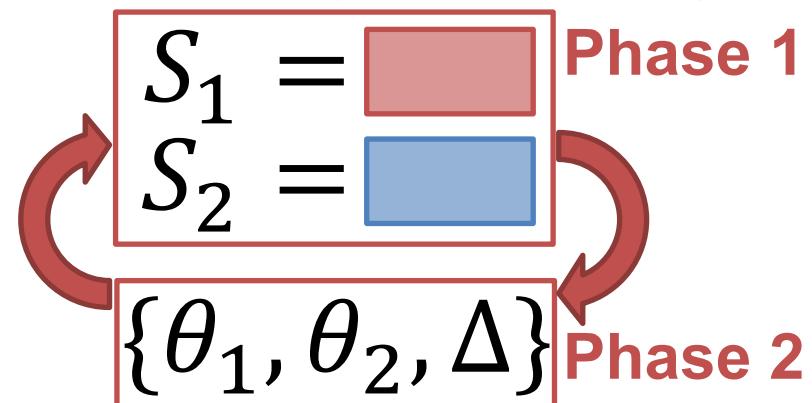
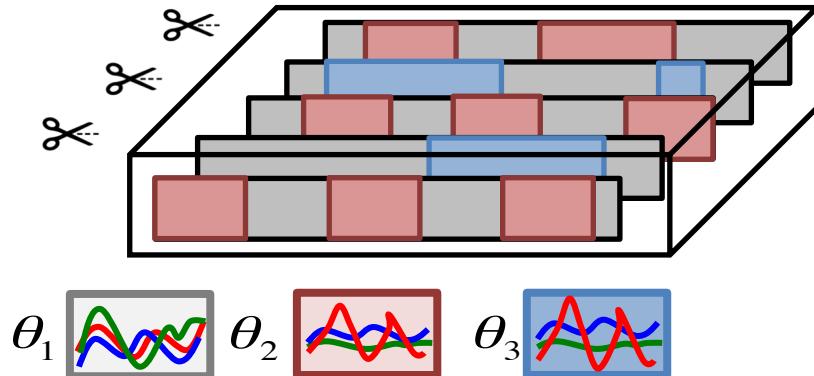
Two phase iterative approach

- Phase 1: (H-Assignment)

- Split segments into two groups: S_1, S_2

- Phase 2: (ModelEstimation)

- Update model parameters: $\Theta = \{\theta_1, \theta_2, \Delta\}$



H-Split

Inner loop

Given:

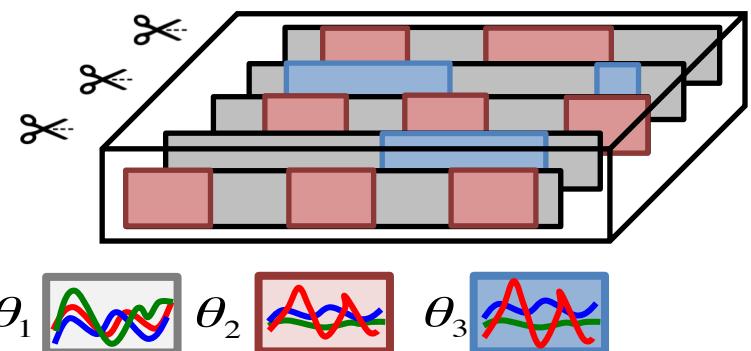
- tensor \mathcal{X}

- model parameter set

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

Find: two user-aspect regimes based on the similarity: $Cost_C(X_i | \theta_j)$

$$\left. \begin{matrix} \mathcal{X} \\ \{\theta_1, \theta_2, \Delta\} \end{matrix} \right\}$$



Algorithms

Algorithms of our method

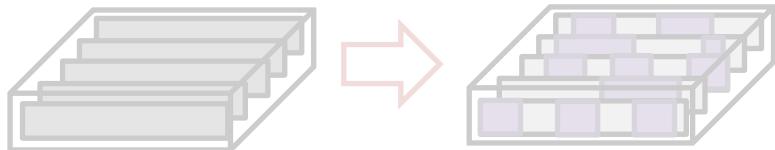
CubeMarker

Outer loop

V-Split

Inner loop

- V-Assignment
- ModelEstimation

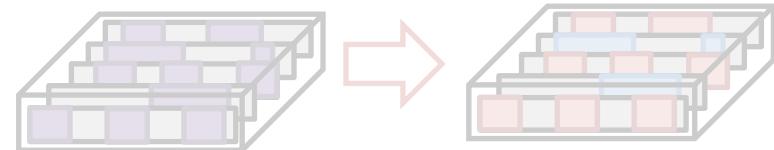


Find time-aspect regime

H-Split

Inner loop

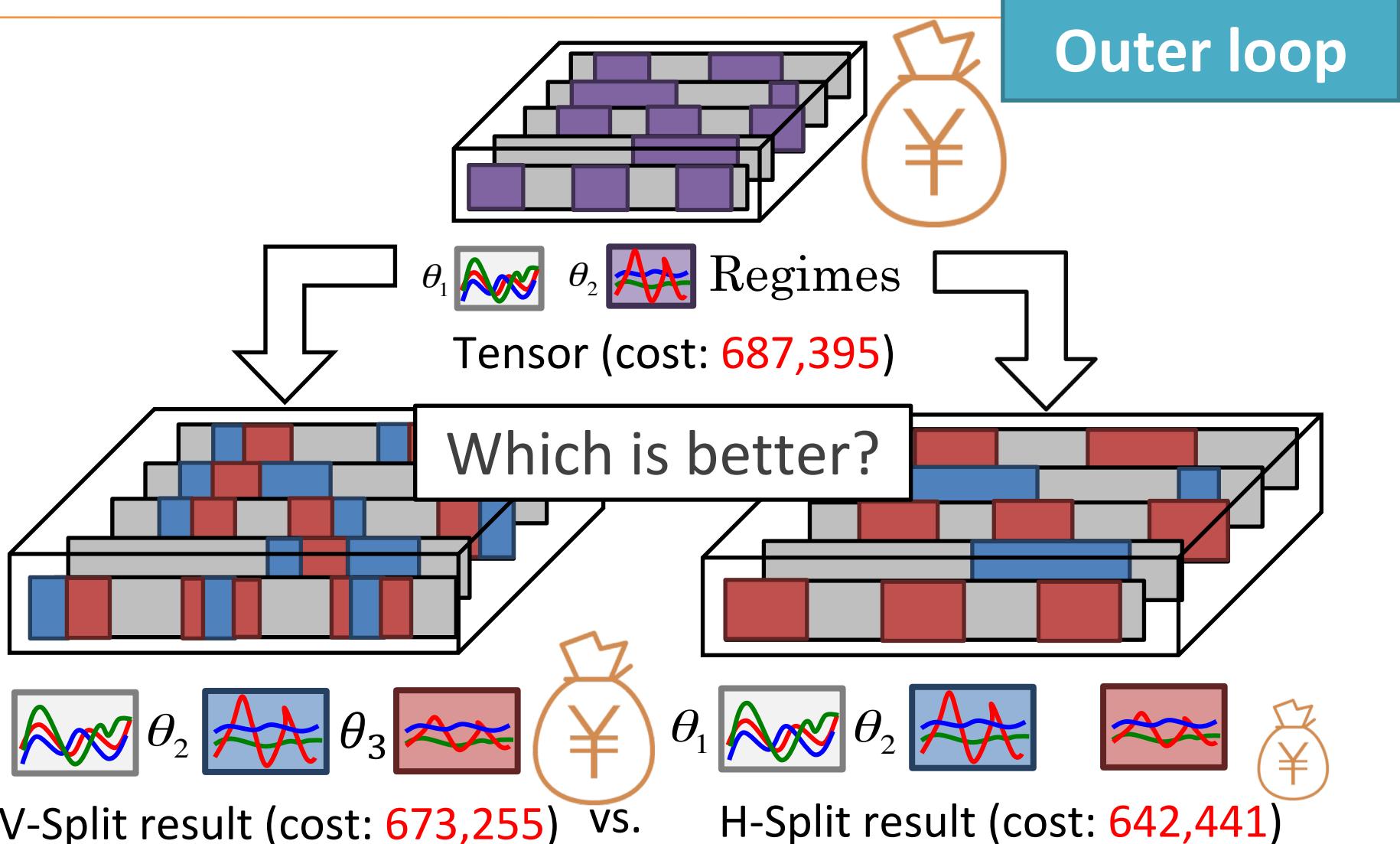
- H-Assignment
- ModelEstimation



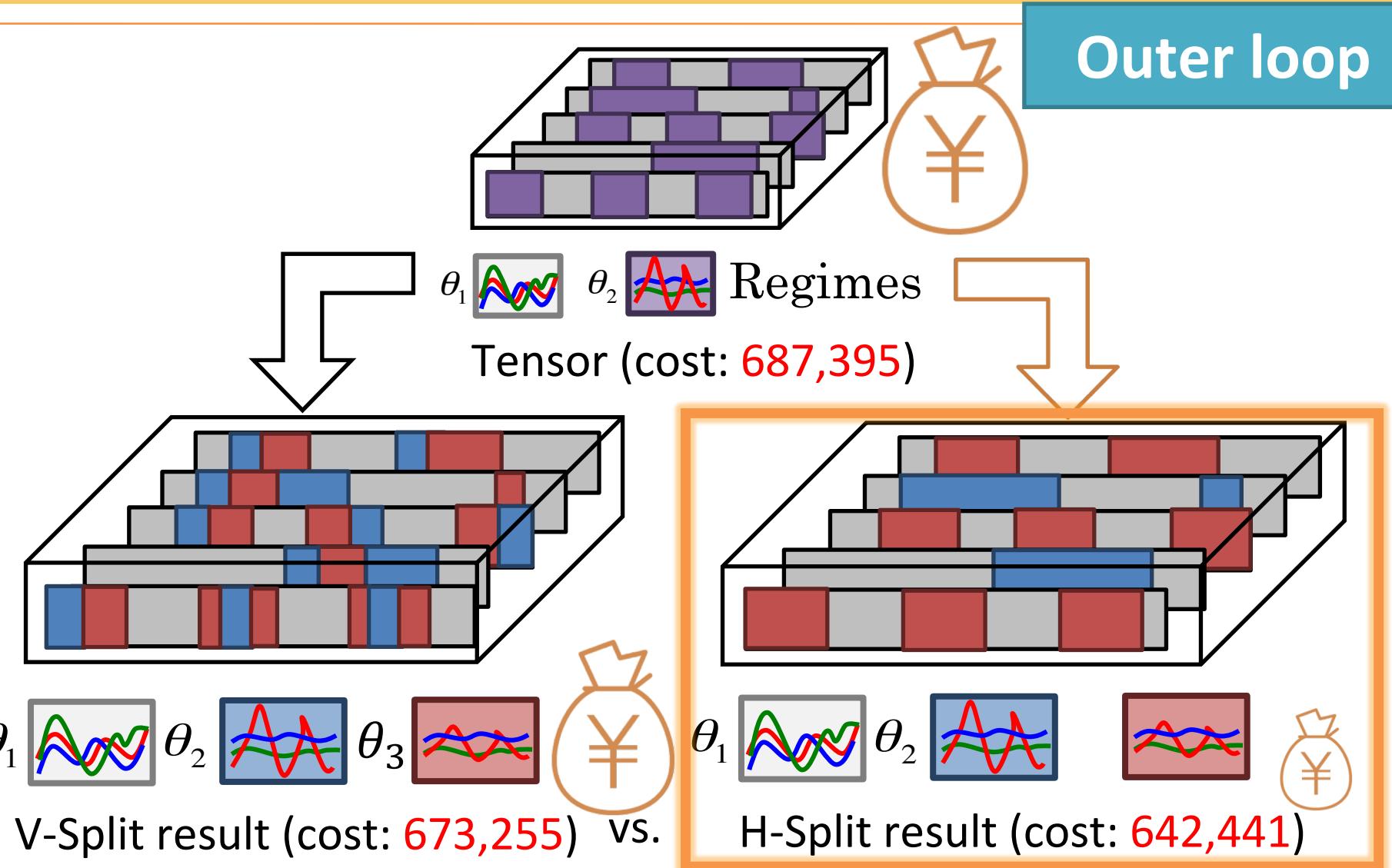
Find user-aspect regime

Decide splitting algorithm

CubeMarker



CubeMarker



Outline

- Motivation
- Problem definition
- Main ideas
- Algorithms
- Experiments
- Conclusions



Experiments

Q1. Effectiveness

Can it help us understand the given tensor?

Q2. Scalability

How does it scale in terms of computational cost?

Q3. Accuracy

How well does it find segments and regimes?

Competitors:

pHMM (SIGMOD'11)

AutoPlait (SIGMOD'14)

TICC (KDD'17)

CubeMarker-V (naïve ver. of our method)

Datasets

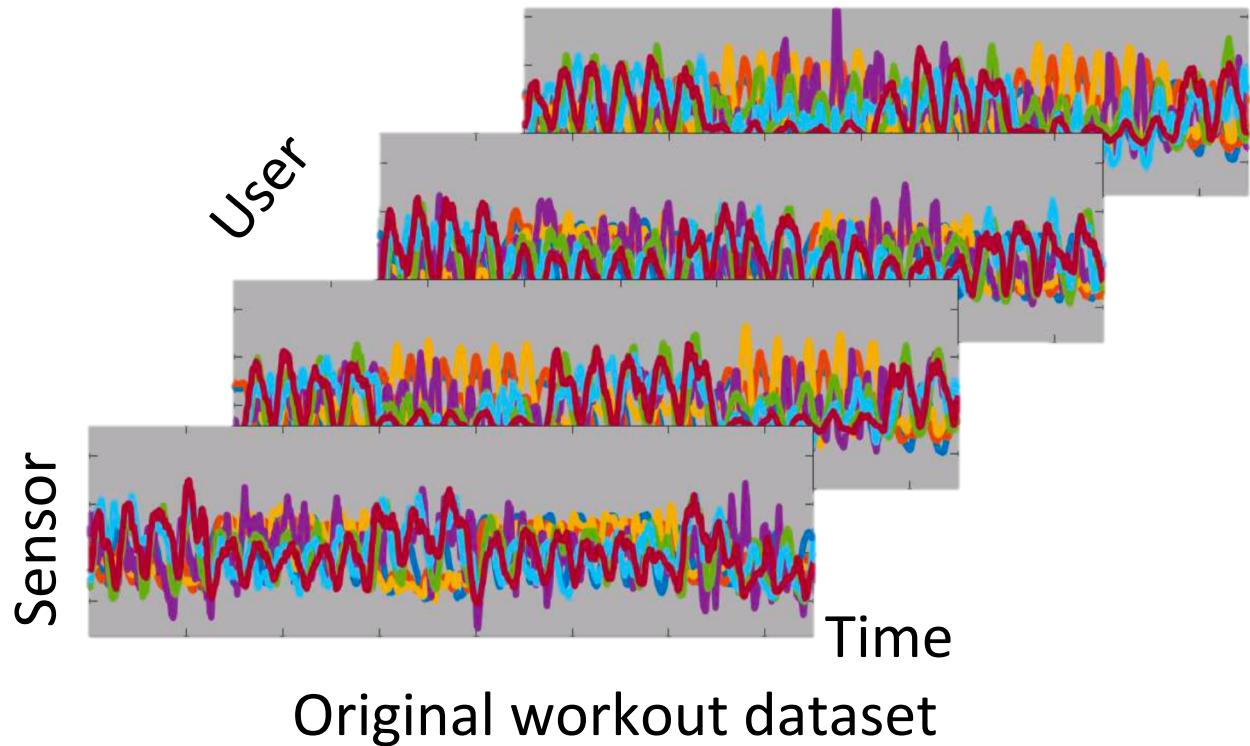
Experiments on the 8 real-world datasets:

| Dataset | Data size ($w \times n \times d$) |
|-----------------------------------|-------------------------------------|
| (#1) <i>Workout</i> | $182 \times 4000 \times 7$ |
| (#2) <i>Tennis</i> | $100 \times 4500 \times 7$ |
| (#3) <i>Factory</i> | $60 \times 3000 \times 7$ |
| (#4) <i>Reading</i> | $71 \times 10000 \times 5$ |
| (#5) <i>Free throw</i> | $170 \times 2000 \times 7$ |
| (#6) <i>Automobile-Tokyo</i> | $171 \times 2400 \times 3$ |
| (#7) <i>Automobile-Expressway</i> | $13 \times 9100 \times 3$ |
| (#8) <i>Automobile-Togu</i> | $32 \times 5200 \times 3$ |

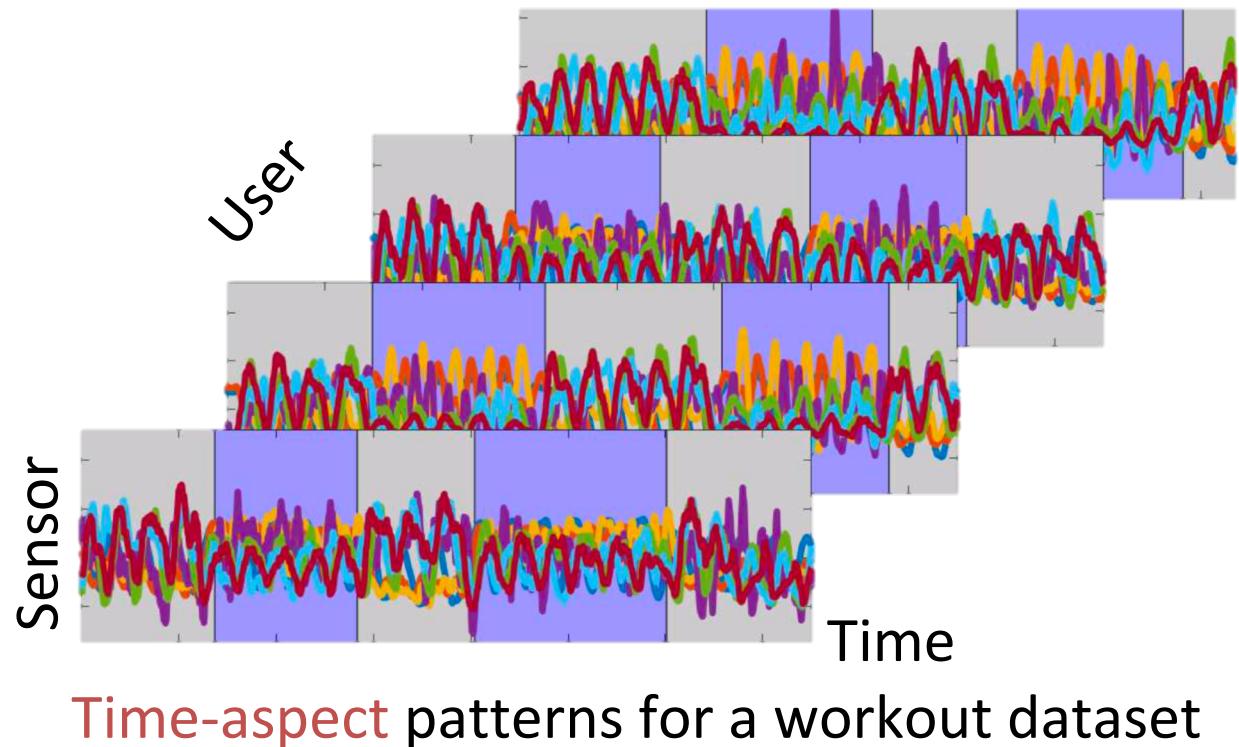
Summary of the datasets

Q1. Effectiveness - Workout

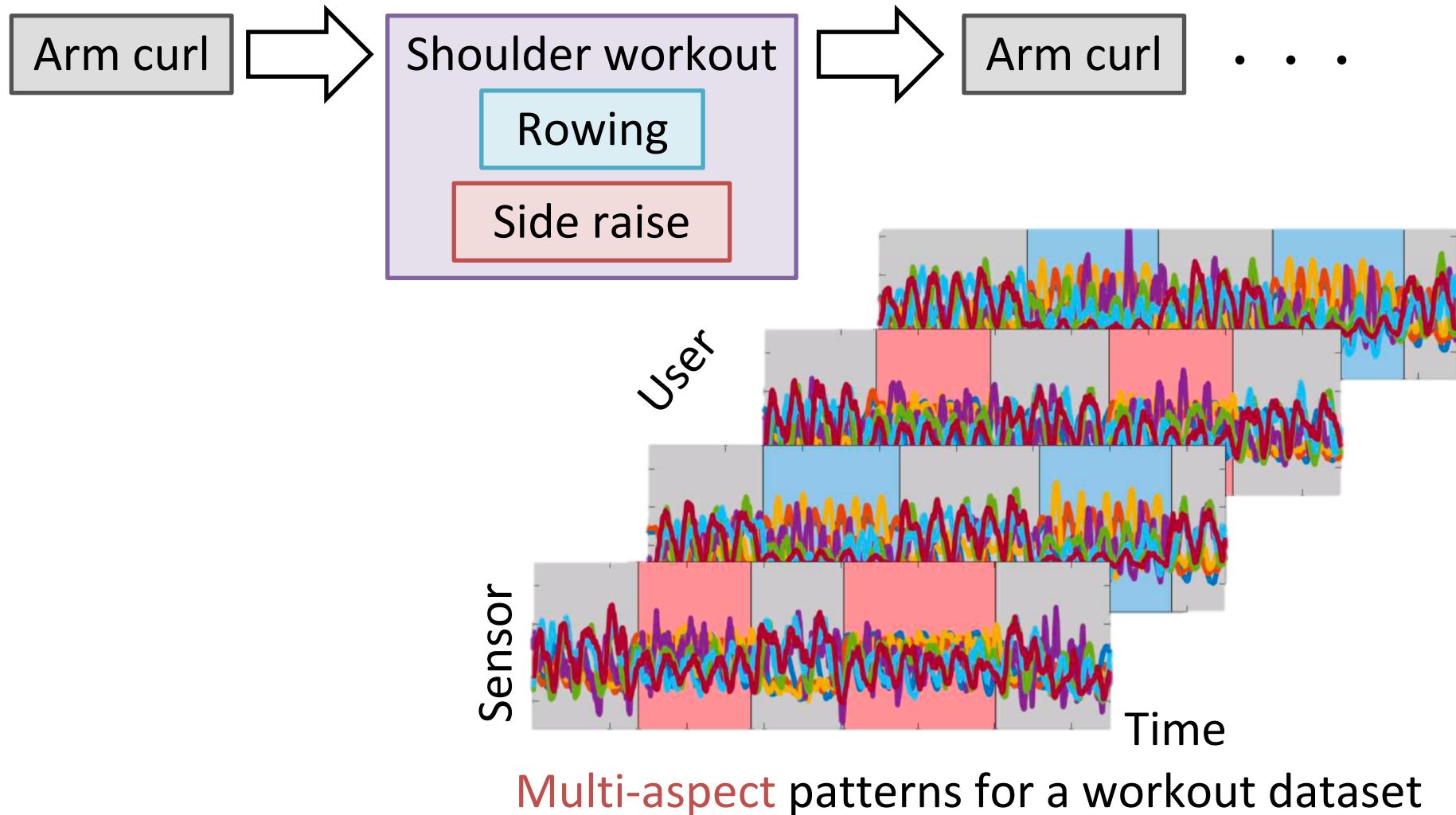
How many and what kind of patterns does it include?



Q1. Effectiveness - Workout



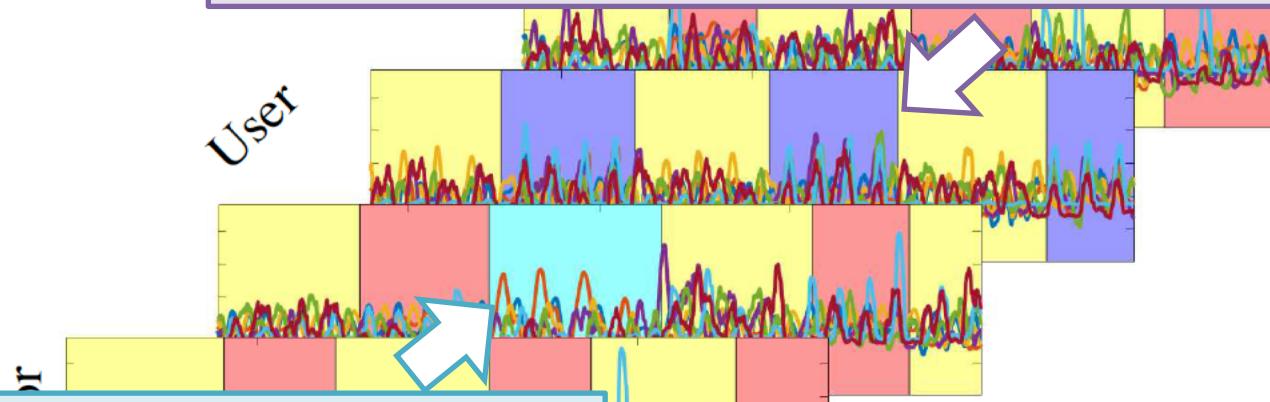
Q1. Effectiveness - Workout



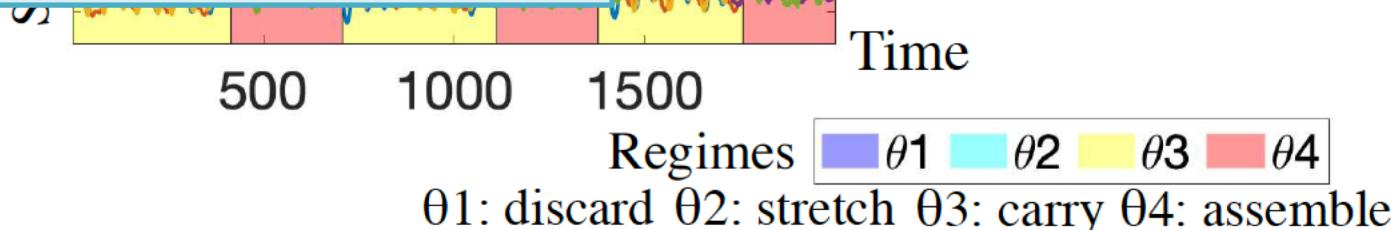
Q1. Effectiveness - Factory worker

Basic pattern transitions: carrying → assembling → · · ·

User-aspect pattern: Discarding defective products



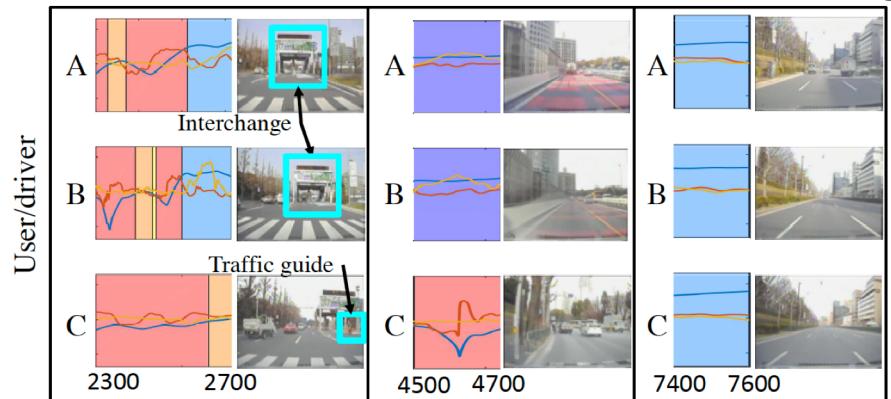
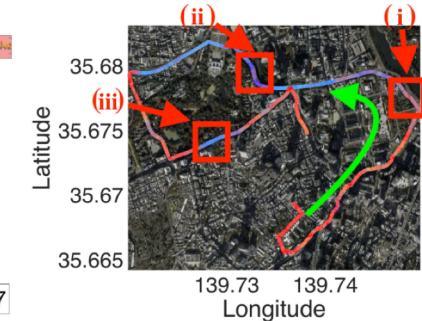
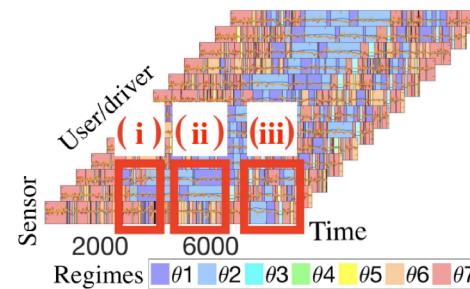
One-shot outlier: stretch arms



Multi-aspect patterns for a factory workers

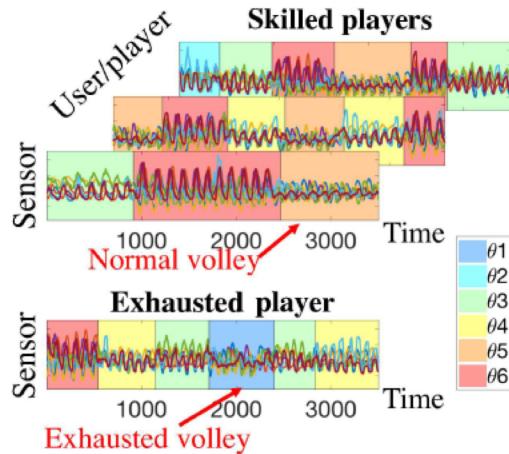
Q1. Effectiveness - Automobile

Our method finds
multi-aspect regimes,
i.e.,
time transition
and
user-specific patterns

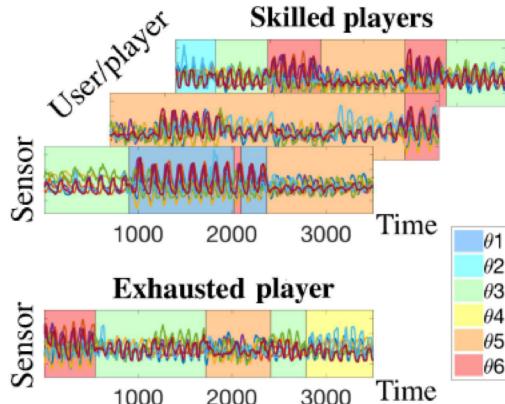


Result for an automobile dataset

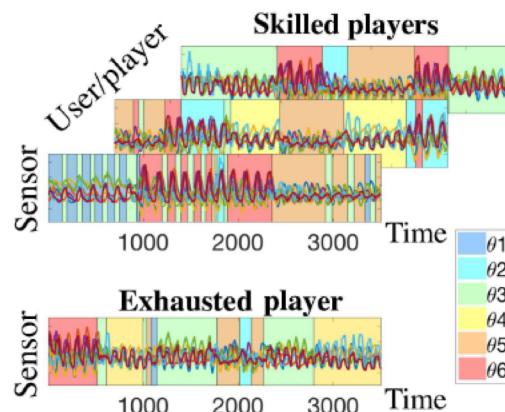
Q1. Effectiveness - Tennis



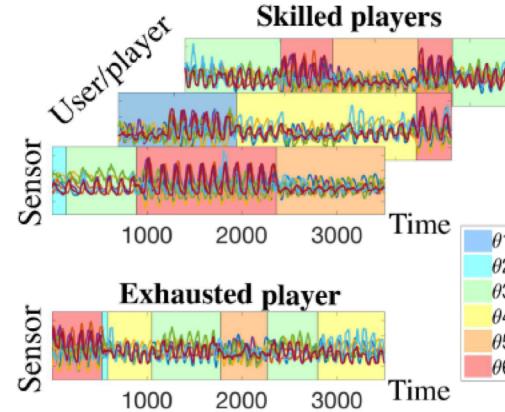
(a) CUBE MARKER
(no parameter setting)



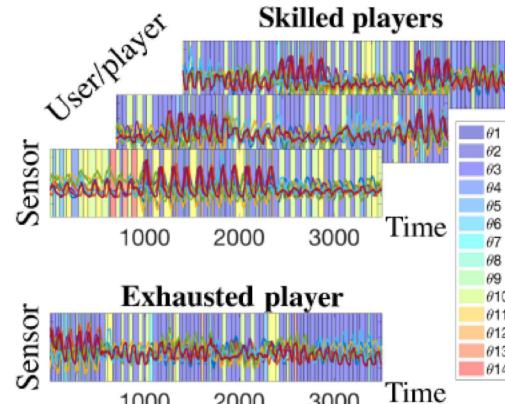
(c) AutoPlait
(no parameter setting)



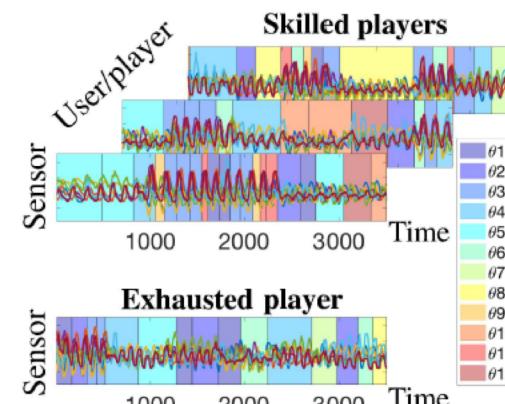
(b-1) TICC ($\beta = 100, \lambda = 1000$)
(need parameter setting)



(b-2) TICC ($\beta = 600, \lambda = 1000$)
(need parameter setting)

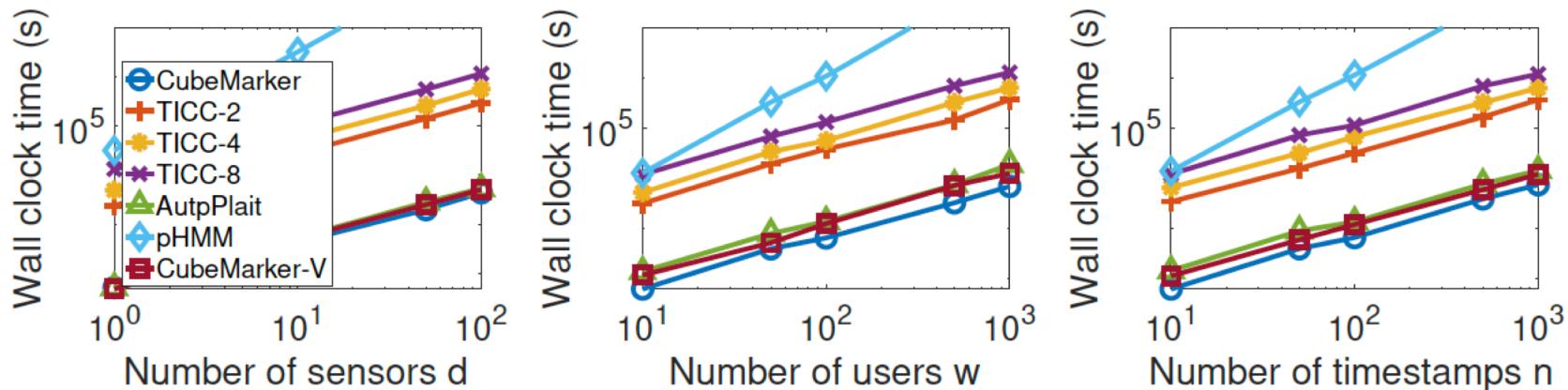


(d-1) pHMM ($\epsilon_r = 0.1, \epsilon_c = 0.8$)
(need parameter setting)

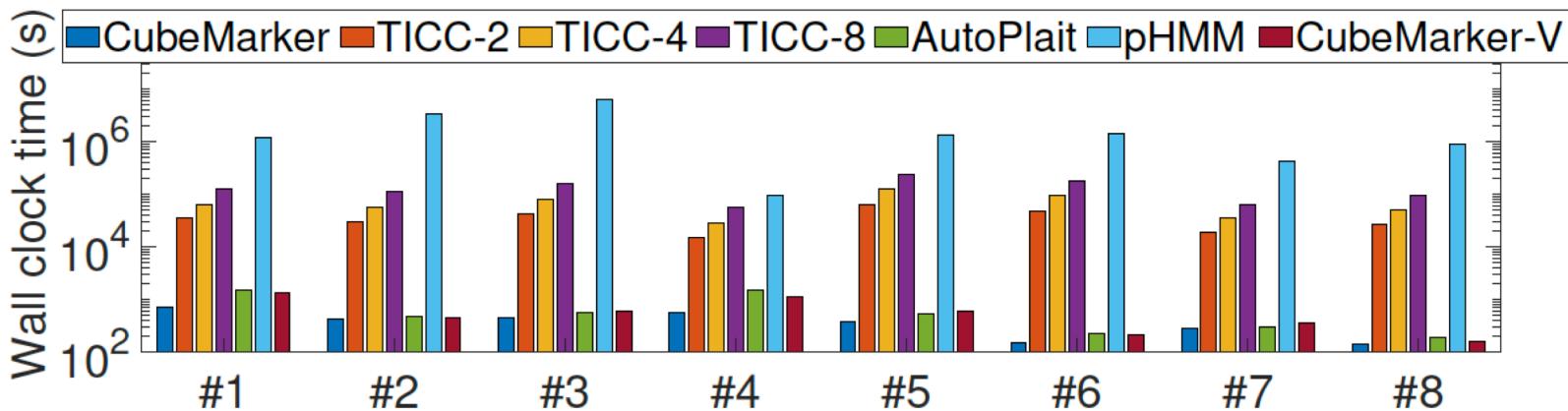


(d-2) pHMM ($\epsilon_r = 10, \epsilon_c = 0.8$)
(need parameter setting)

Q2. Scalability

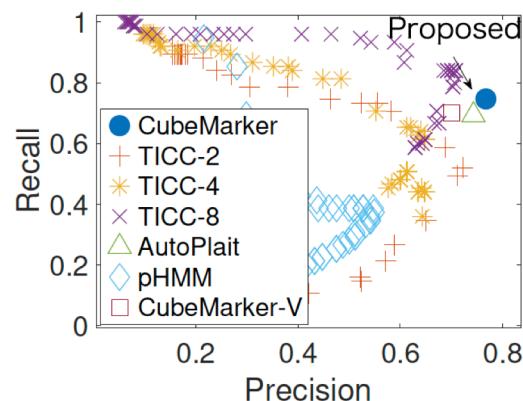


Wall clock time v.s. dataset size for (#1) Workout ($O(dwn)$)

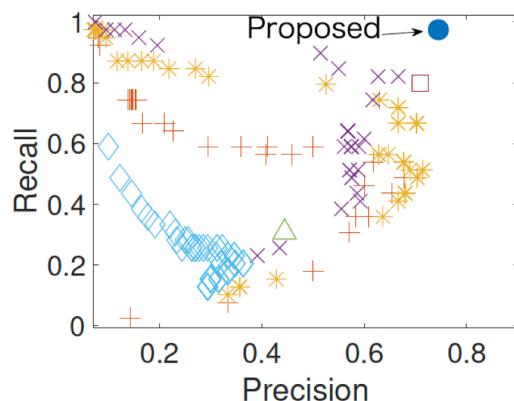


Wall clock time for each dataset (**1700x faster** than pHMM)

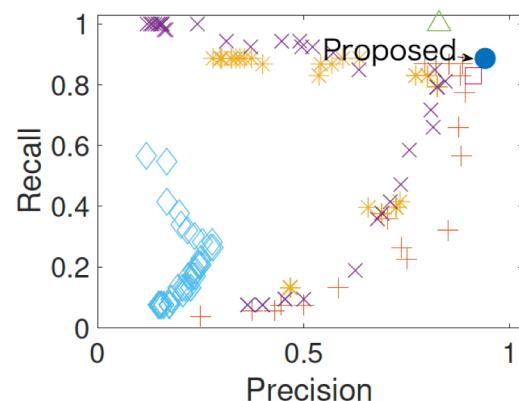
Q3. Accuracy (segment/regime)



(a) (#1) *Workout*

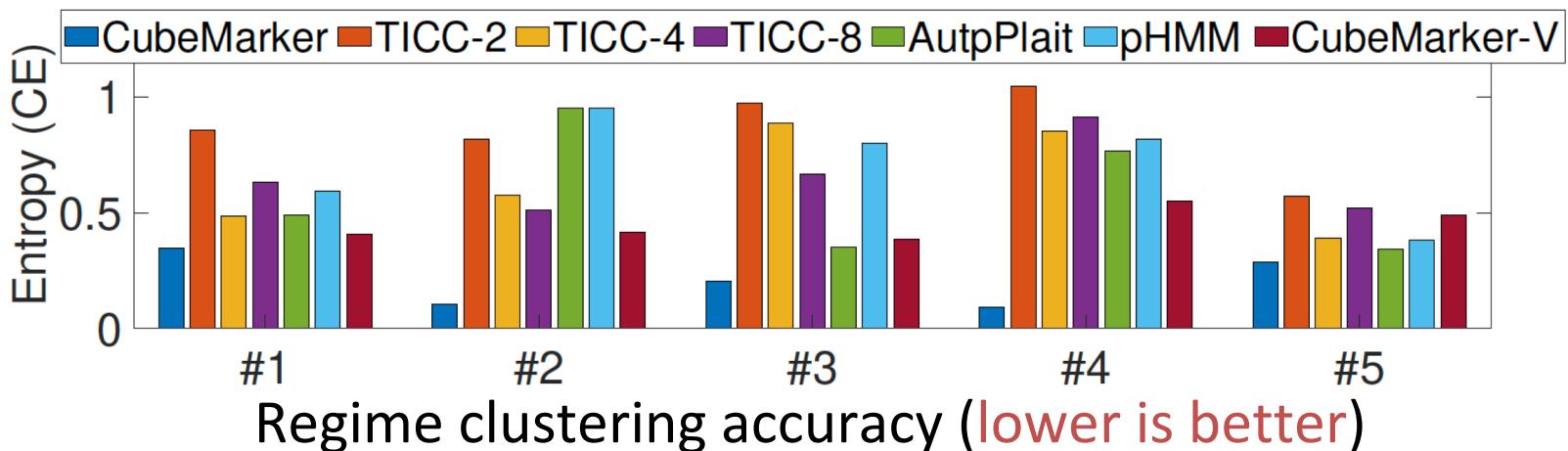


(b) (#2) *Tennis*



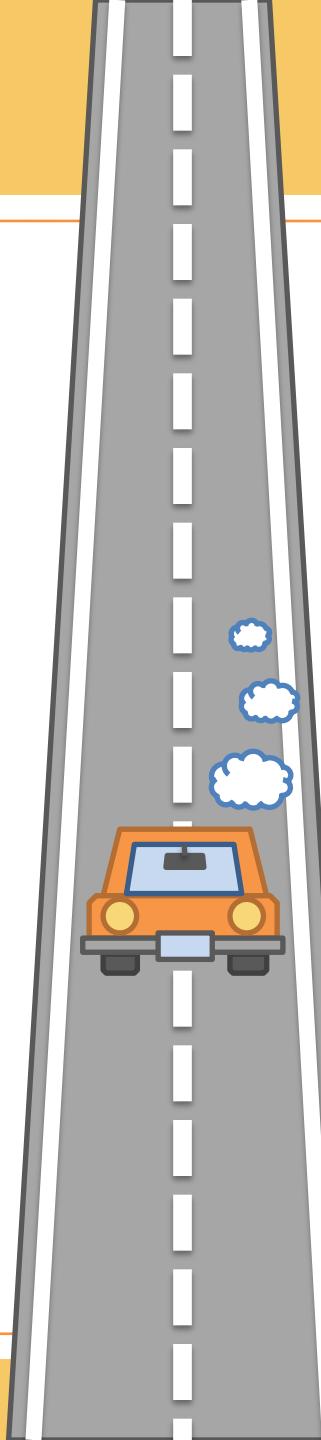
(c) (#3) *Factory*

Segmentation accuracy (top righter is better)



Outline

- Motivation
- Problem definition
- Main ideas
- Algorithms
- Experiments
- Conclusions



Conclusions

Our method has the following properties:

- **Effective**

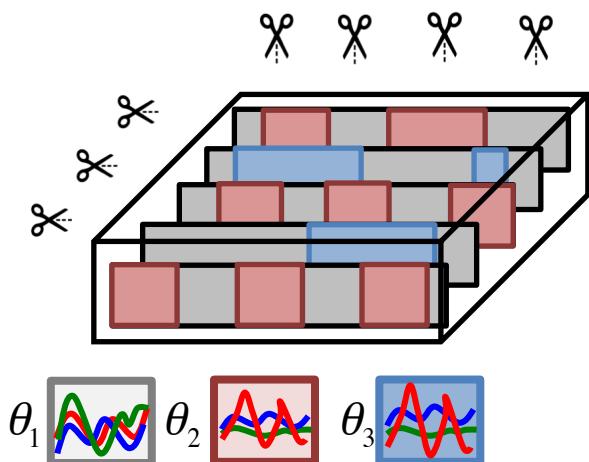
Find multi-aspect segments/regimes

- **Automatic**

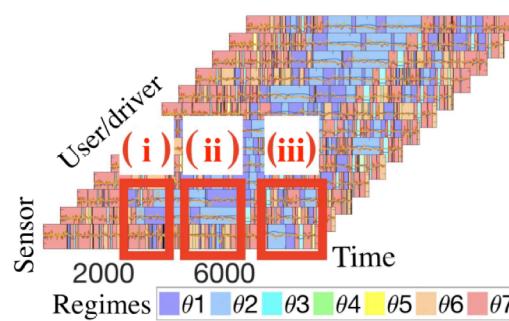
No magic numbers

- **Scalable**

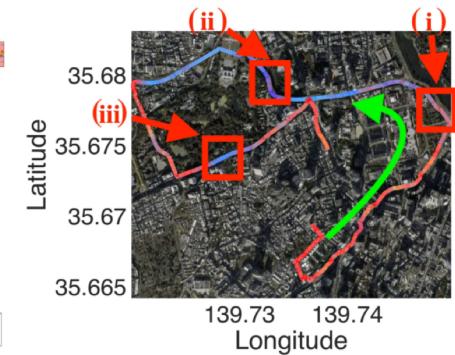
It scales linearly to the data size



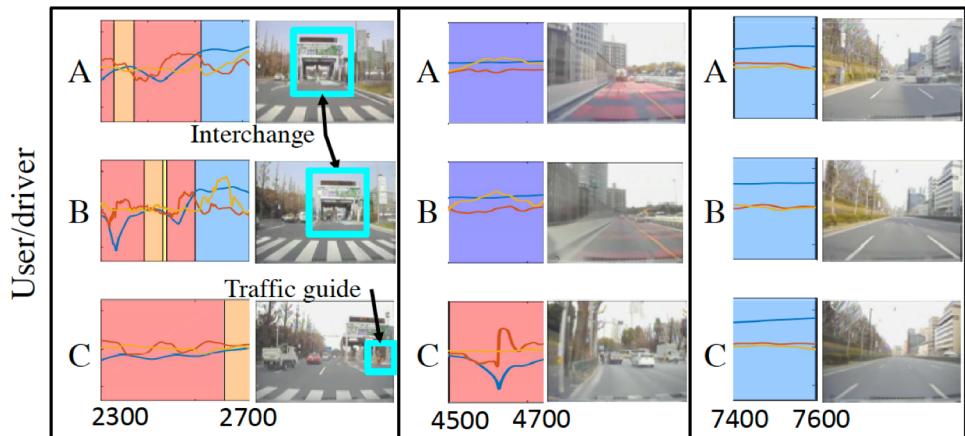
Thank you!



(a) Multi-aspect segmentation and summarization



(b) Representative driving behavior on a map



(c-i) Interchange (c-ii) Expressway (c-iii) Wide road
(c) User/driver-specific behavior at three different locations