Fast Algorithms for Mining Co-evolving Time Series

Lei Li
Computer Science Department
Carnegie Mellon University

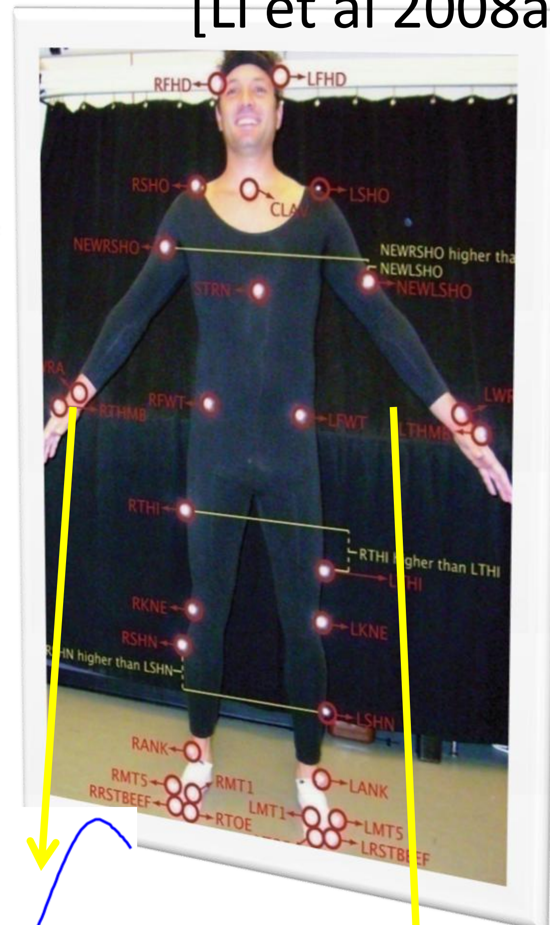
Committee:
Christos Faloutsos (chair)
Nancy Pollard
Eric Xing
Jiawei Han (UIUC)

Why study co-evolving time series?

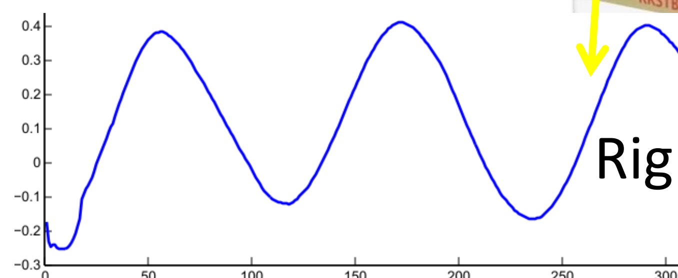
Correlated multidimensional time sequences with joint temporal dynamics

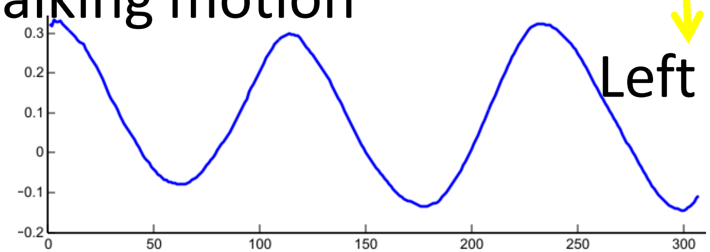
Motion Capture

[Li et al 2008a]



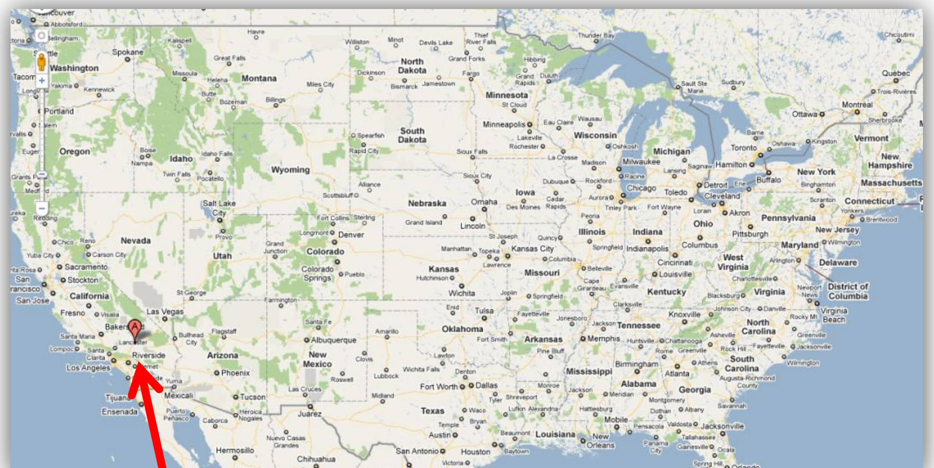
- Goal: generate natural human motion
  - Game ($57B)
  - Movie industry
- Challenge:
  - Missing values
  - "naturalness"

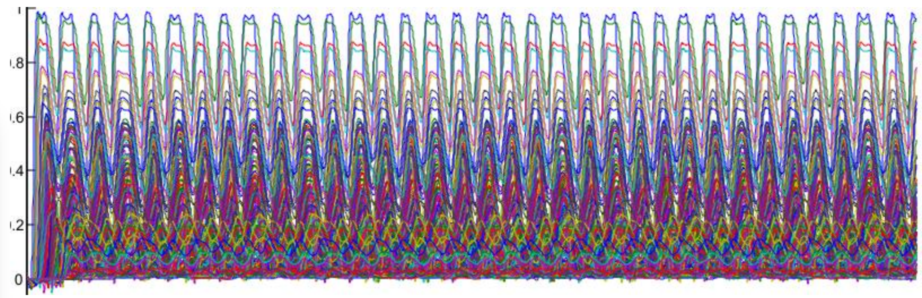




Environmental Monitoring

- Problem: early detection of leakage & pollution
- Challenge: noise & large data





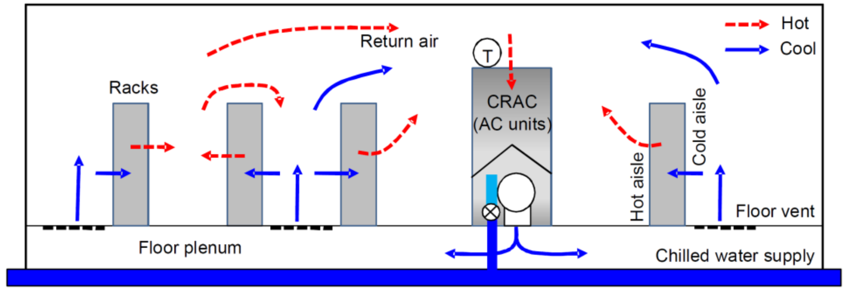
Chlorine level in drinking water systems [Li et al 2009]

Barstow residents advised not to drink tap water because of possible contamination

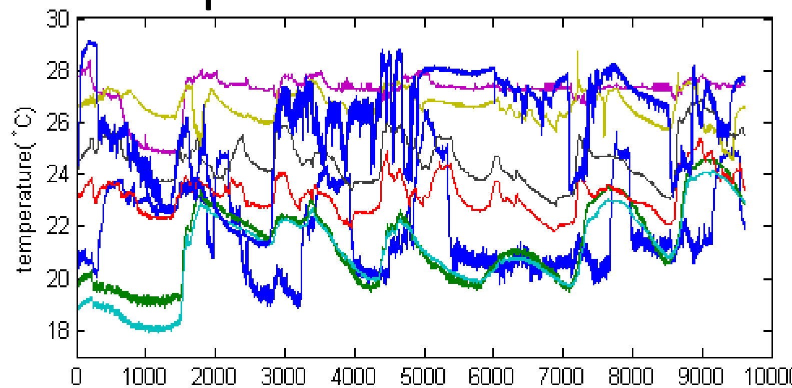
November 19, 2010 | 5:54 pm

Datacenter Monitoring & Management

- Goal: save energy in data centers
  - US alone, $7.4B power consumption (2011)
- Challenge:
  - Huge data (1TB per day)
  - Complex cyber physical systems



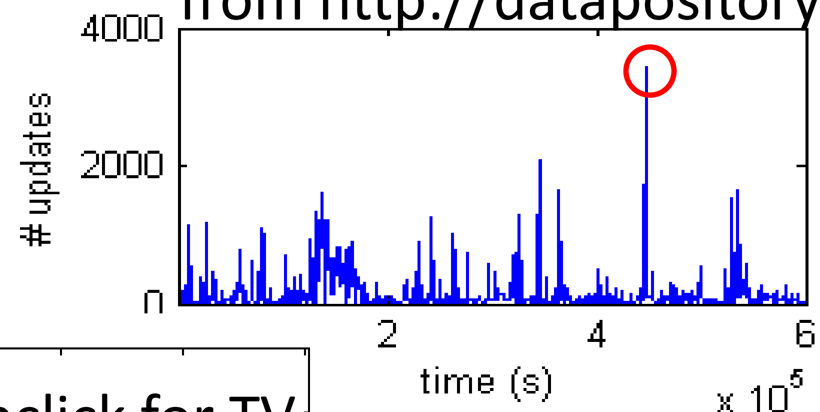
Temperature in datacenter

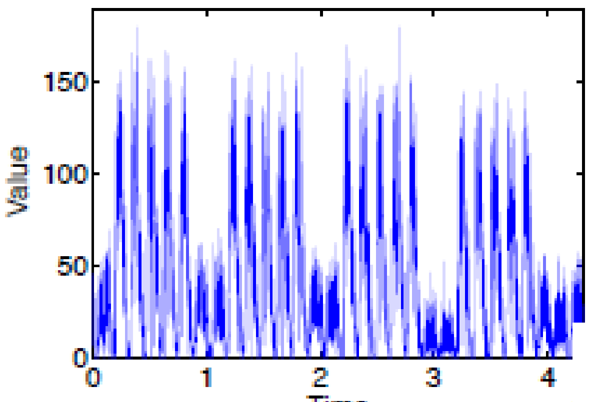


Network Security

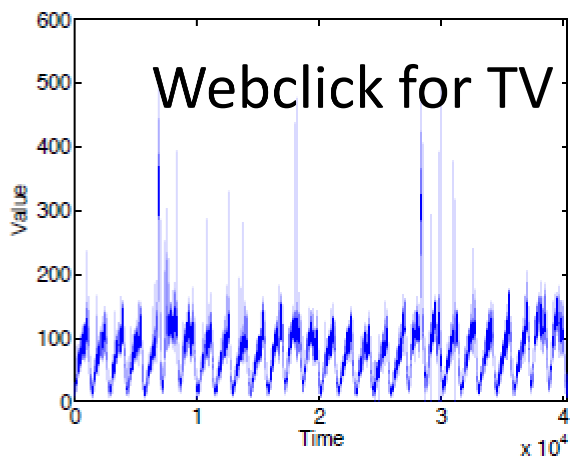
- Challenge: Anomaly detection in computer network & online activity

BGP # updates on backbone from http://datapository.net/





Webclick for news from NTT



Webclick for TV

BIG Challenges

in mining co-evolving time series

Pattern discovery

1. Imputation
2. Compression
3. Segmentation
4. Anomaly

Feature extraction

5. Clustering
6. Visualization
7. Indexing
8. Similarity search

Parallel algorithm

9. Parallel learning algorithms on SMP/multicore

BIG Challenges and Solutions

in mining co-evolving time series

Pattern discovery

1. Imputation
2. Compression
3. Segmentation
4. Anomaly

Feature extraction

5. Clustering
6. Visualization
7. Indexing
8. Similarity search

Parallel algorithm

9. Parallel learning algorithms on SMP/multicore

- DynaMMo [Li 09]
- BoLeRO [Li 10a]
- ThermoCast [Li 11b]
- LazinessScore [Li08a]

- PLiF [Li 10b]
- CLDS [Li 11a]

- Cut-And-Stitch [Li 08b]
- WindMine [Sakurai 11]

Contributions & Results

Pattern discovery

- DynaMMo [Li 09]
- BoLeRO [Li 10a]
- ThermoCast [Li 11b]
- LazinessScore [Li08a]

Feature extraction

- PLiF [Li 10b]
- CLDS [Li 11a]

Parallel algorithm

- Cut-And-Stitch [Li 08b]
- WindMine [Sakurai 11]

Contributions:

1. Most accurate missing value recovery/summarization
2. Most effective clustering on TS
3. Fast algorithms: linear to length
4. Parallel algorithms: linear speed up on multicore

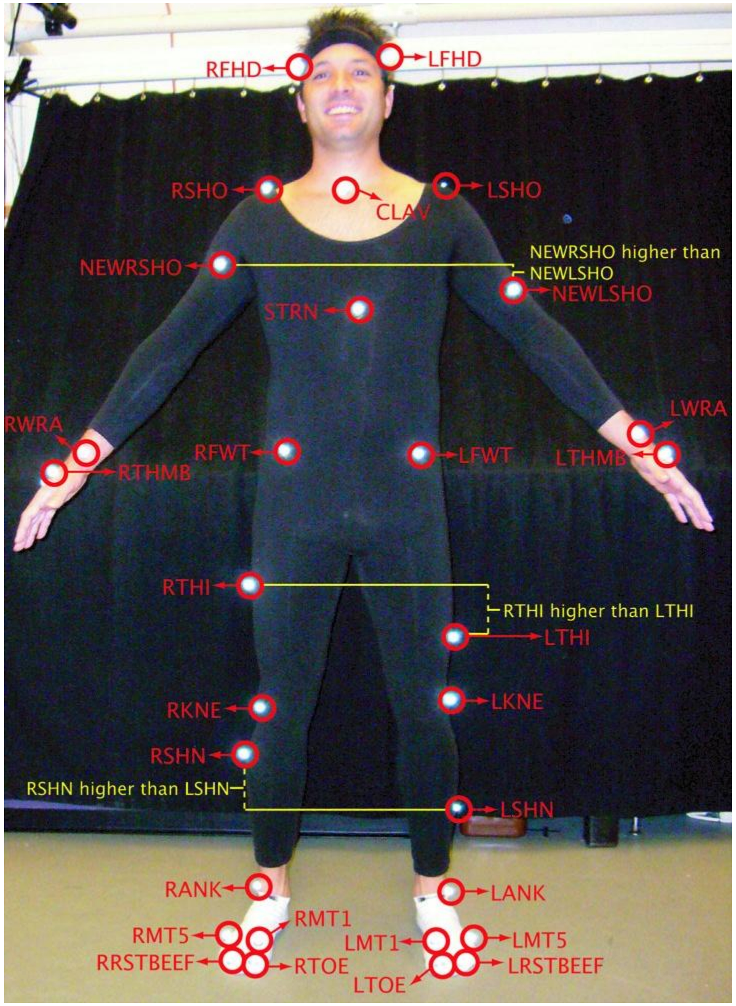
Outline

- Motivation
- Mining w/ Missing Values [Li+ 09, Li+10a]
- Feature Learning for Time Series [Li+10b, Li+11a]
- Summary of the remaining chapters
- Conclusion and Future Directions

Missing Values in Time Series

- Motion Capture:
  - Markers
  - Cameras track 3D positions
  - 93 dimensional body-local coordinates(31-joints)
  - Occlusion
- Sensor data

  missing values due to
  - Low battery
  - RF error

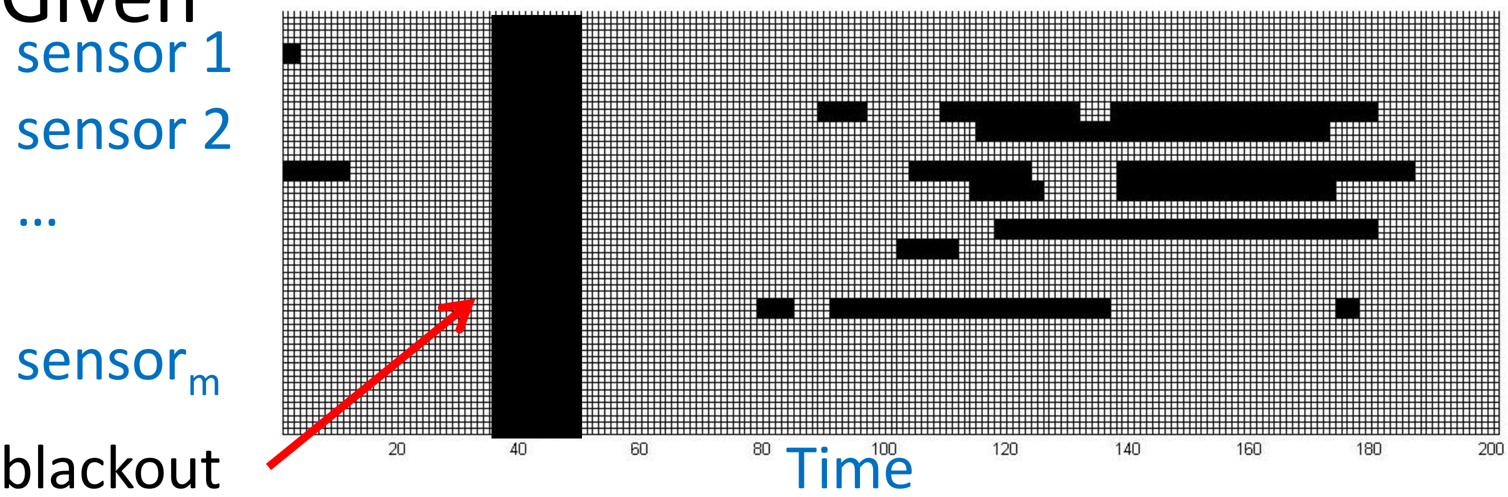


From mocap.cs.cmu.edu

[Li et al, KDD 2009, Li et al SCA 2010a]

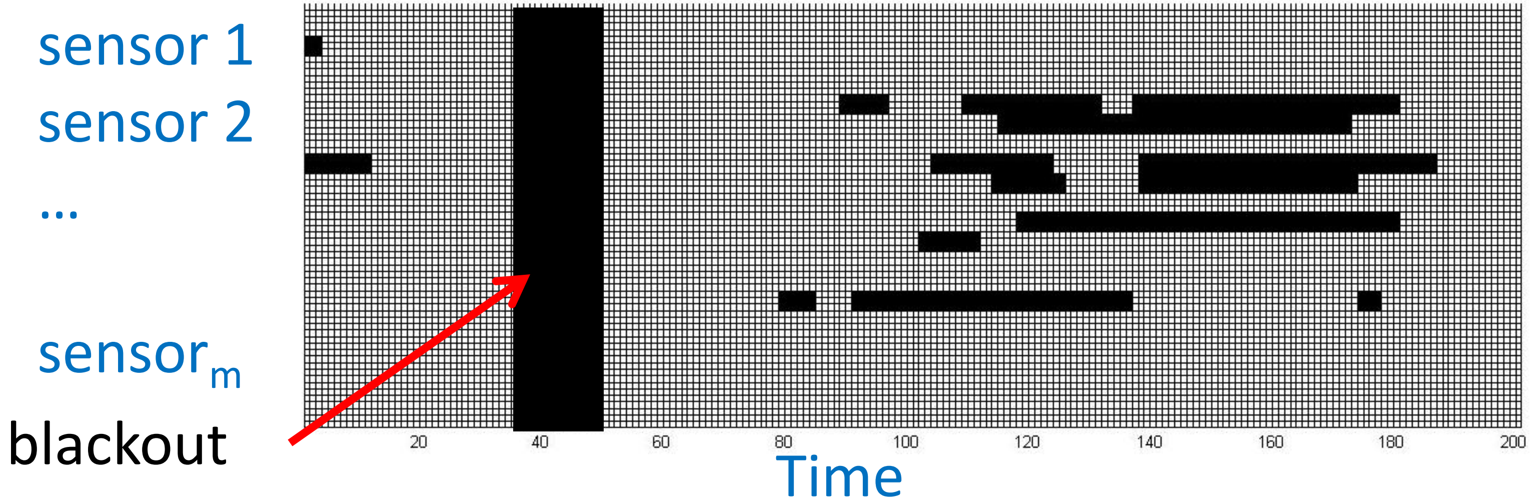
Problem Definition

Given



- Find algorithms for:
  - Task 1: Recovering missing values/imputation
  - Task 2: Compression/summarization
  - Task 3: Segmentation

Problem Definition (cont')

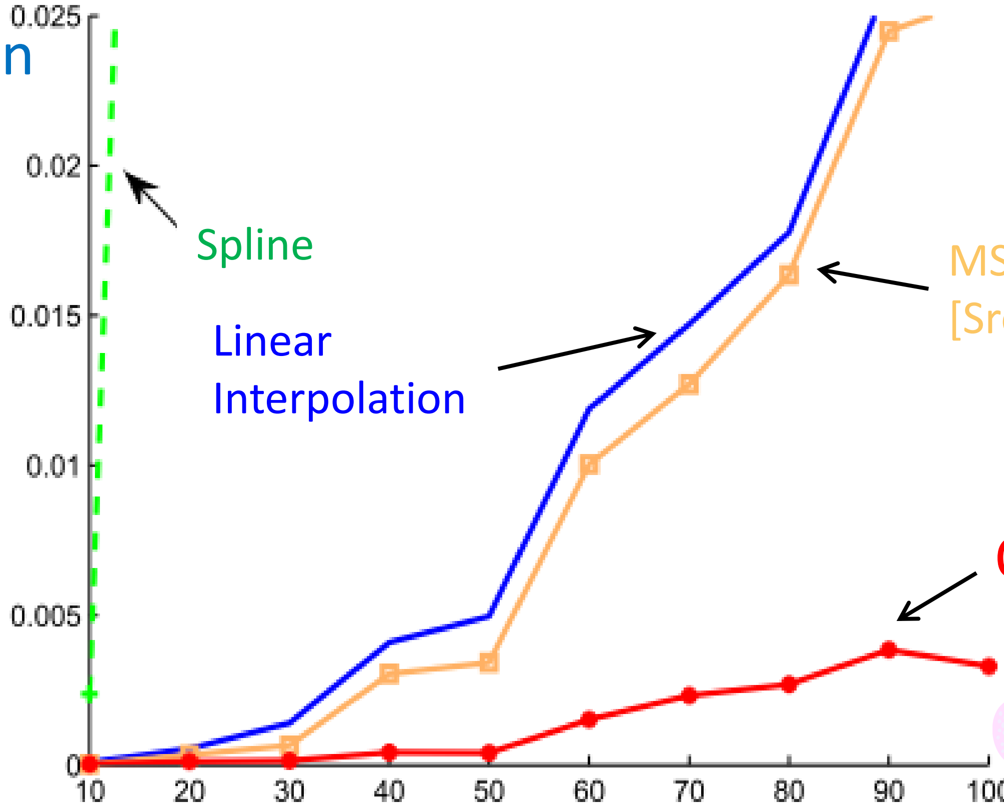


- Ideal algorithm:
  - Goal 1: *Effective*
  - Goal 2: *Scalable:* to duration of sequences

Preview – “DynaMMo”

Reconstruction error

better



Spline

Linear Interpolation

MSVD [Srebro’03]

Our DynaMMo

Ideal

Average missing length

harder

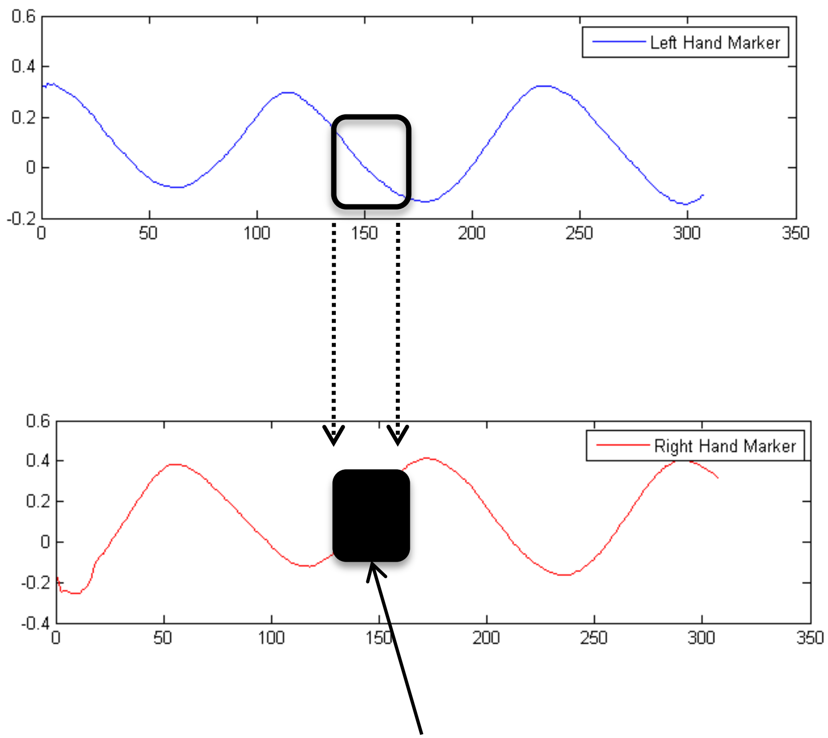
Dataset:
CMU Mocap #16
mocap.cs.cmu.edu

more results in [Li et al, KDD 2009]

Proposed Method: DynaMMo Intuition

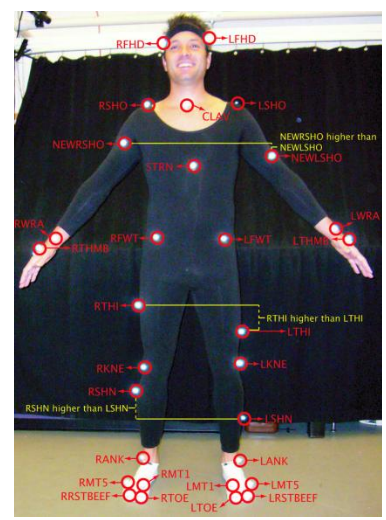
Position of Left hand marker

Position of right hand marker



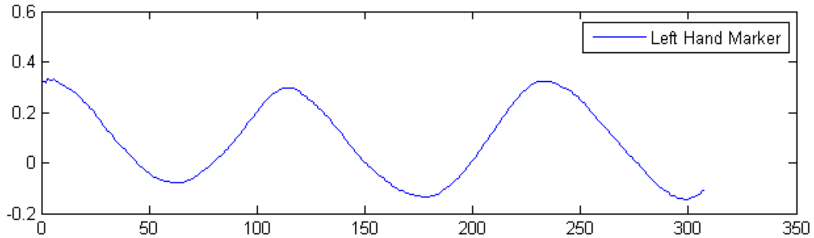
missing

Recover using **(a) Correlation** among multiple sequences

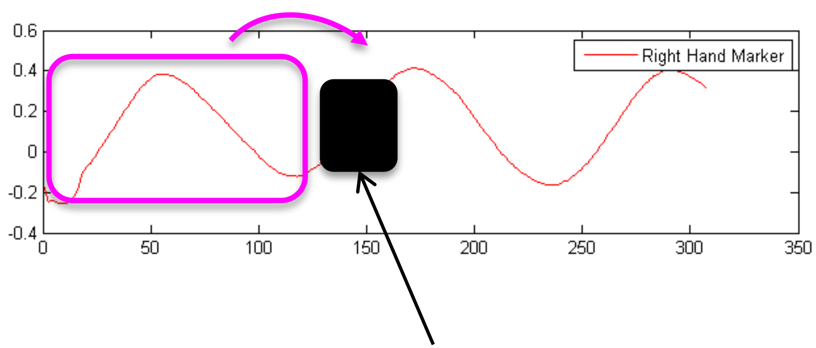


Proposed Method: DynaMMo Intuition

Position of
Left hand
marker



Position of
right hand
marker



missing

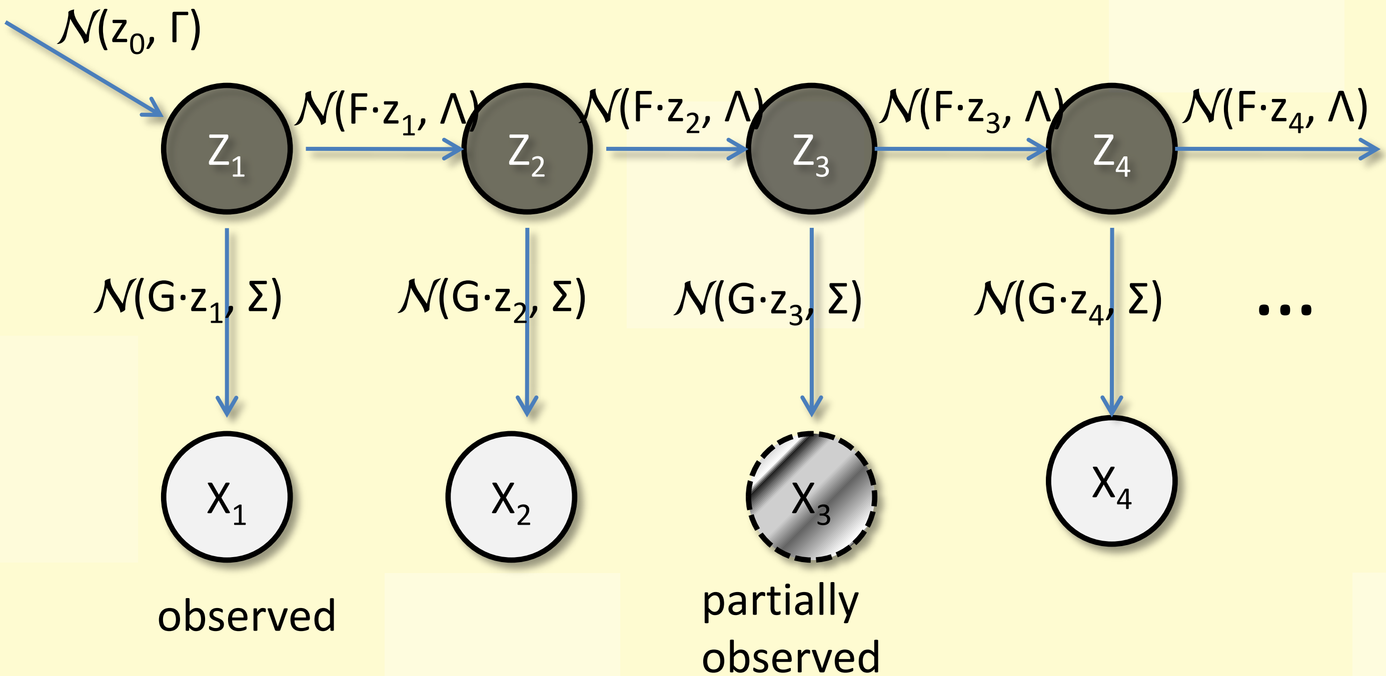
Recover using
(a) Correlation
among multiple
sequences

and
(b) Dynamics
temporal moving
pattern

(details)

DynaMMo Underlying Model

Use *Linear Dynamical Systems* to model whole sequence.



Model parameters:

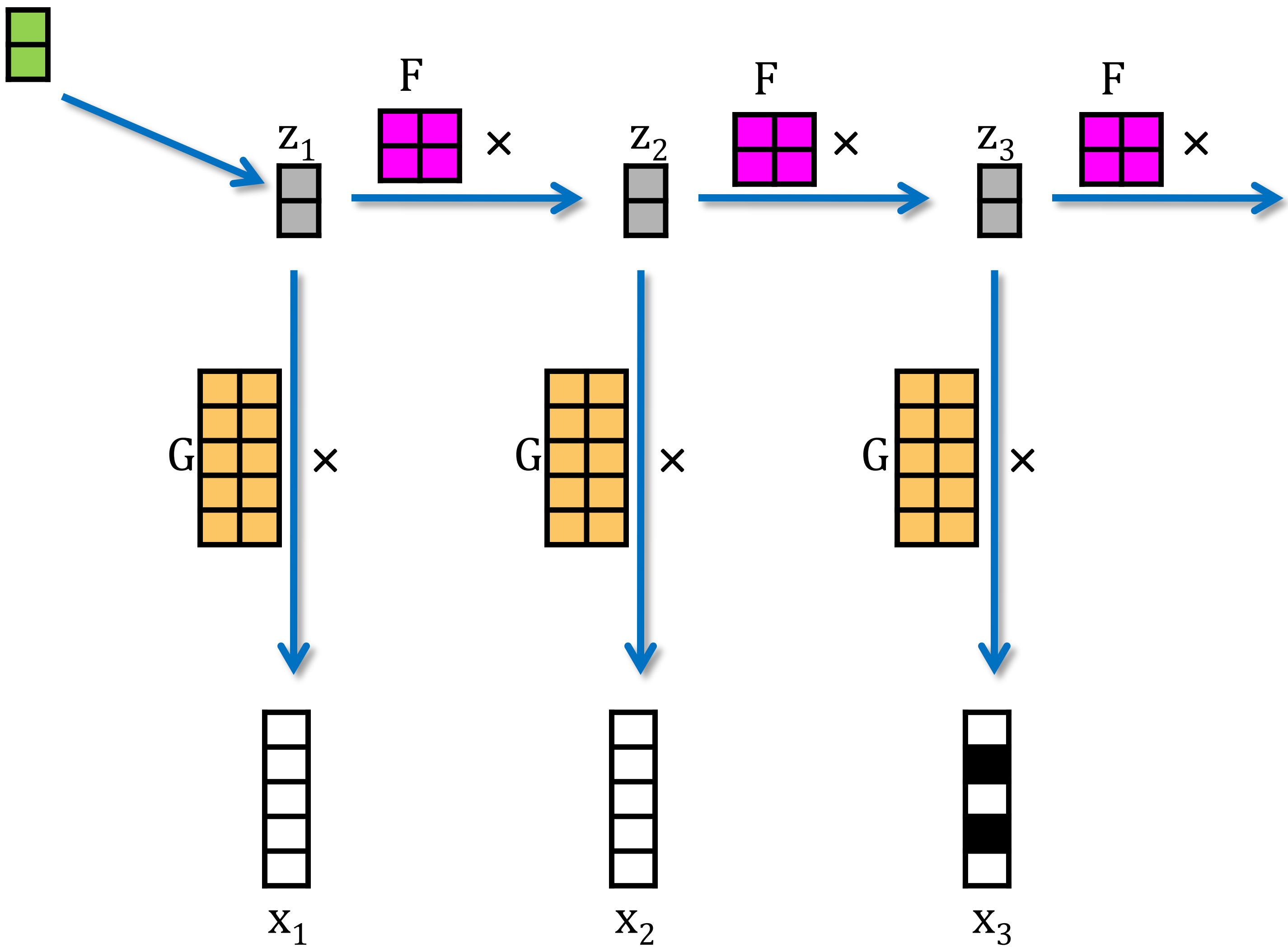
$\theta = \{z_0, \Gamma, F, \Lambda, G, \Sigma\}$

$$z_1 = z_0 + \omega_0$$

$$z_{n+1} = F\cdot z_n + \omega_n$$

$$x_n = G\cdot z_n + \varepsilon_n$$

Learning problem:
estimate all colored elements



Details in [Li+2009]

DynaMMo learning

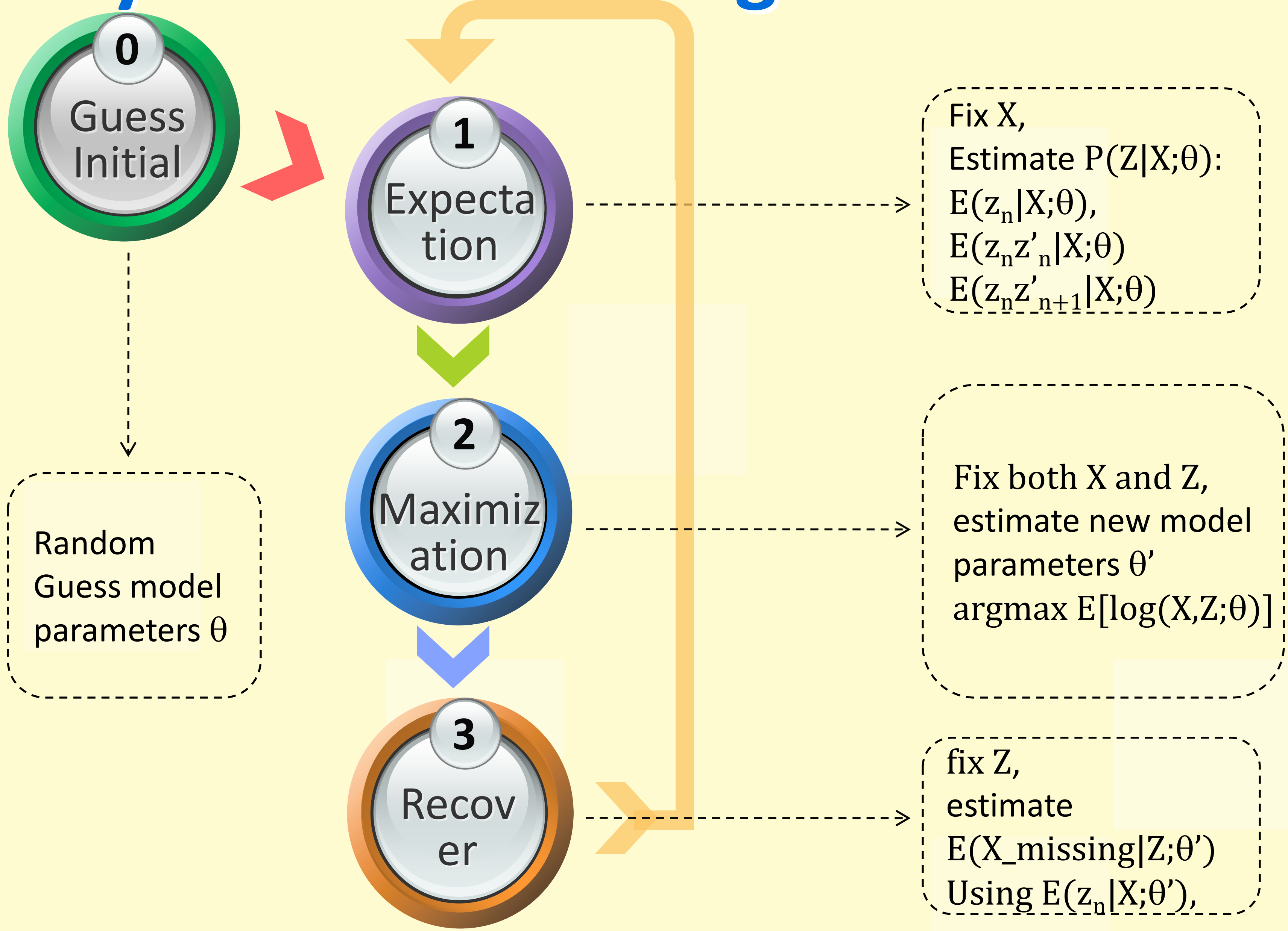
- Finding the best model parameters (θ) and missing values for X to maximize the expected log-likelihood:

$$Q(\theta) = E_{X_m, Z|X_g;\theta}\Big[-(z_1 - z_0)^T \Gamma^{-1} (z_1 - z_0) - \sum_{n=2}^{N} (z_n - F \cdot z_{n-1})^T \Lambda^{-1} (z_n - F \cdot z_{n-1}) - \sum_{n=1}^{N} (x_n - G \cdot z_n)^T \Sigma^{-1} (x_n - G \cdot z_n)\Big]$$

- Proposed optimization method:
  - **Expectation-Maximization-Recover**

DynaMMo Learning

(details)



Outline

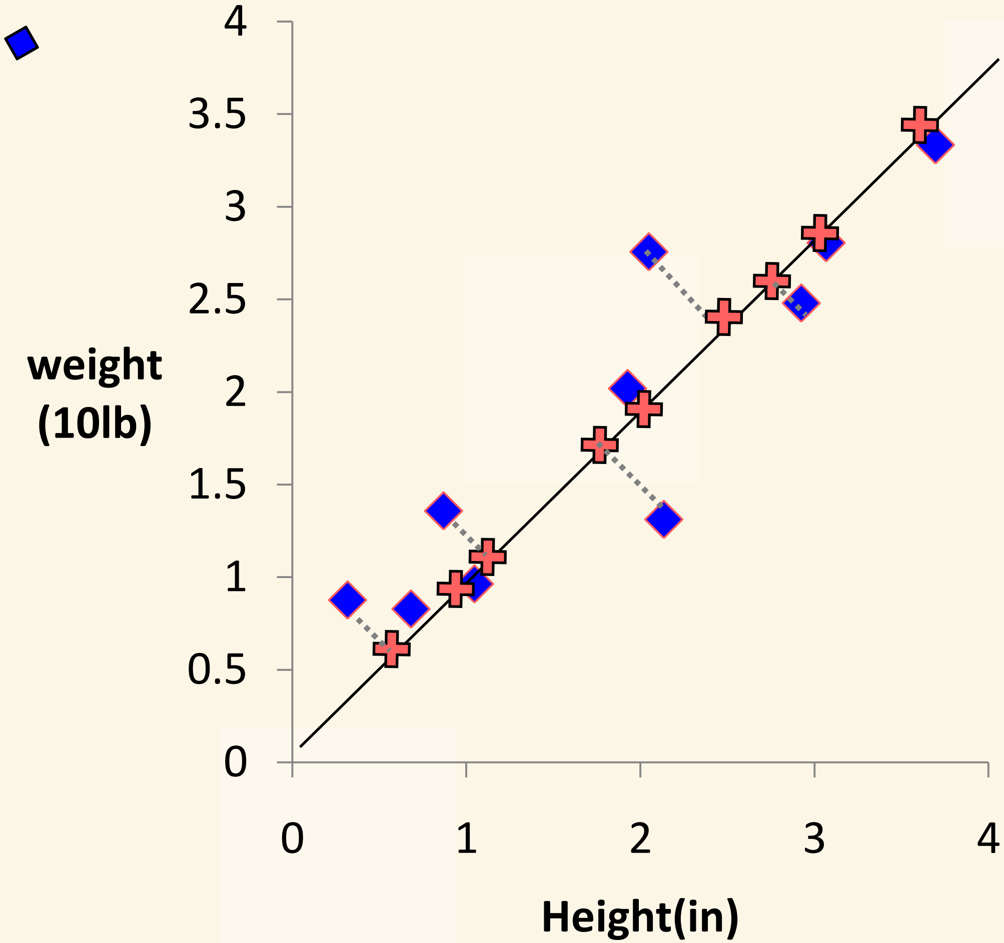
- Motivation
- Mining w/ Missing Values [Li+ 09, Li+10a]
  - Problem Definition
  - Proposed Method
    - T1: recovering
    - T2: compression
    - T3: segmentation
  - Results
- Feature Learning for Time Series [Li+10b , Li+11a]
- Summary of the remaining chapters
- Conclusion and Future Directions

(Background)

How to Compress? Traditional Approach: PCA/SVD

original data

height	weight
0.32	0.88
0.68	0.83
0.87	1.36
1.05	0.96
2.13	1.31
1.93	2.02
2.05	2.76
2.92	2.48
3.07	2.81
3.70	3.34



PC1

−1.84
−1.58
−1.10
−1.21
−0.15
0.13
0.69
1.20
1.52
2.34

PCA: general data matrix

data:
column
centered

Loading matrix

$$X = U \bullet V^T$$

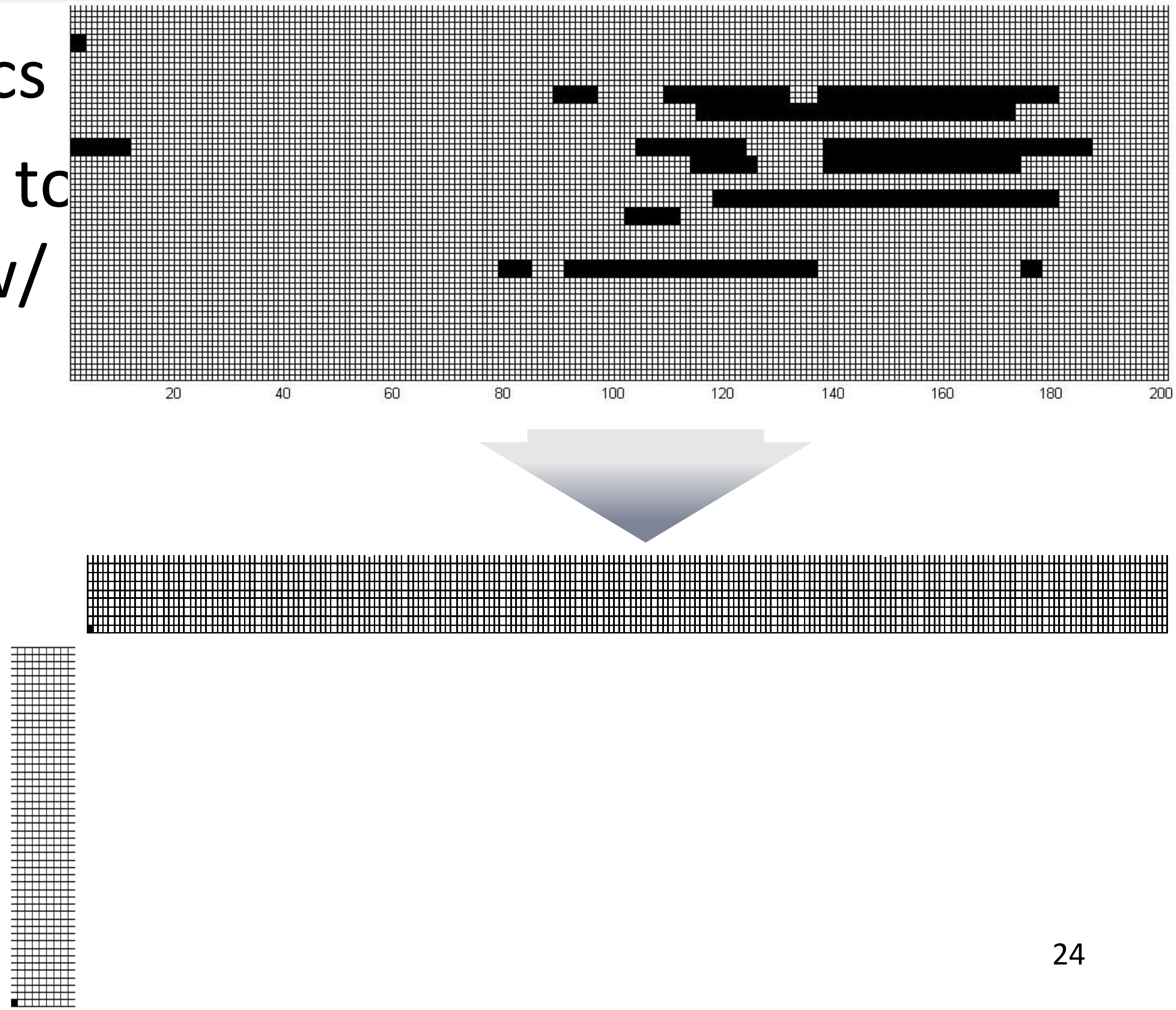
X

U
Score
matrix

V^T
PC1, 2..k

Why Not PCA/SVD?

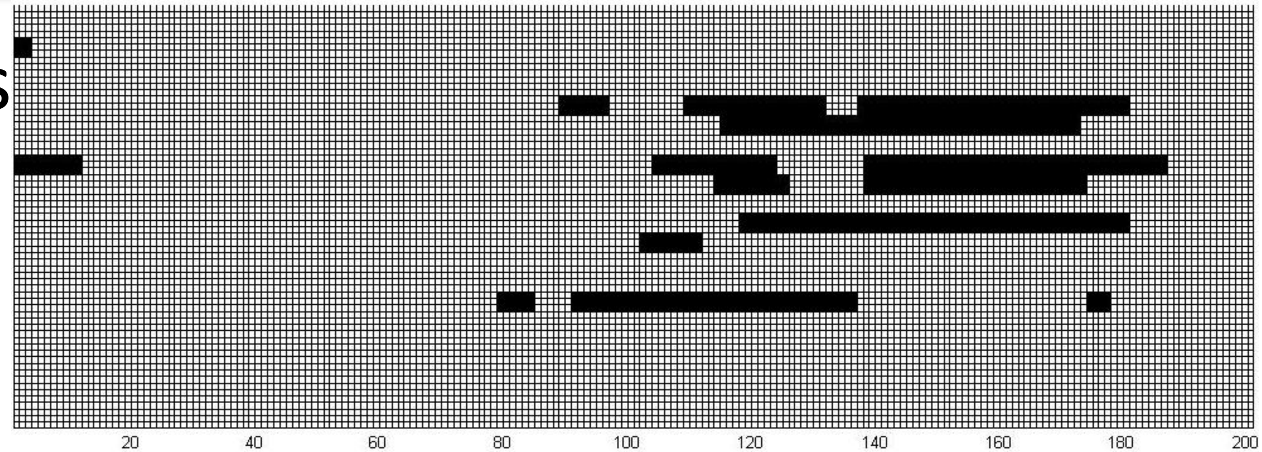
- No dynamics
- Need more to compress w/ same accuracy

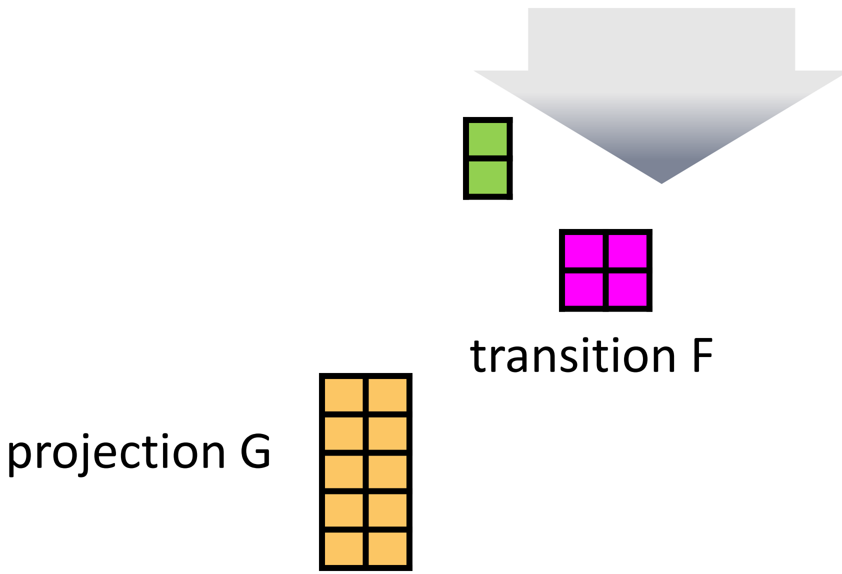


A higher compression ratio

Store parameters of DynaMMo

But bad reconstruction

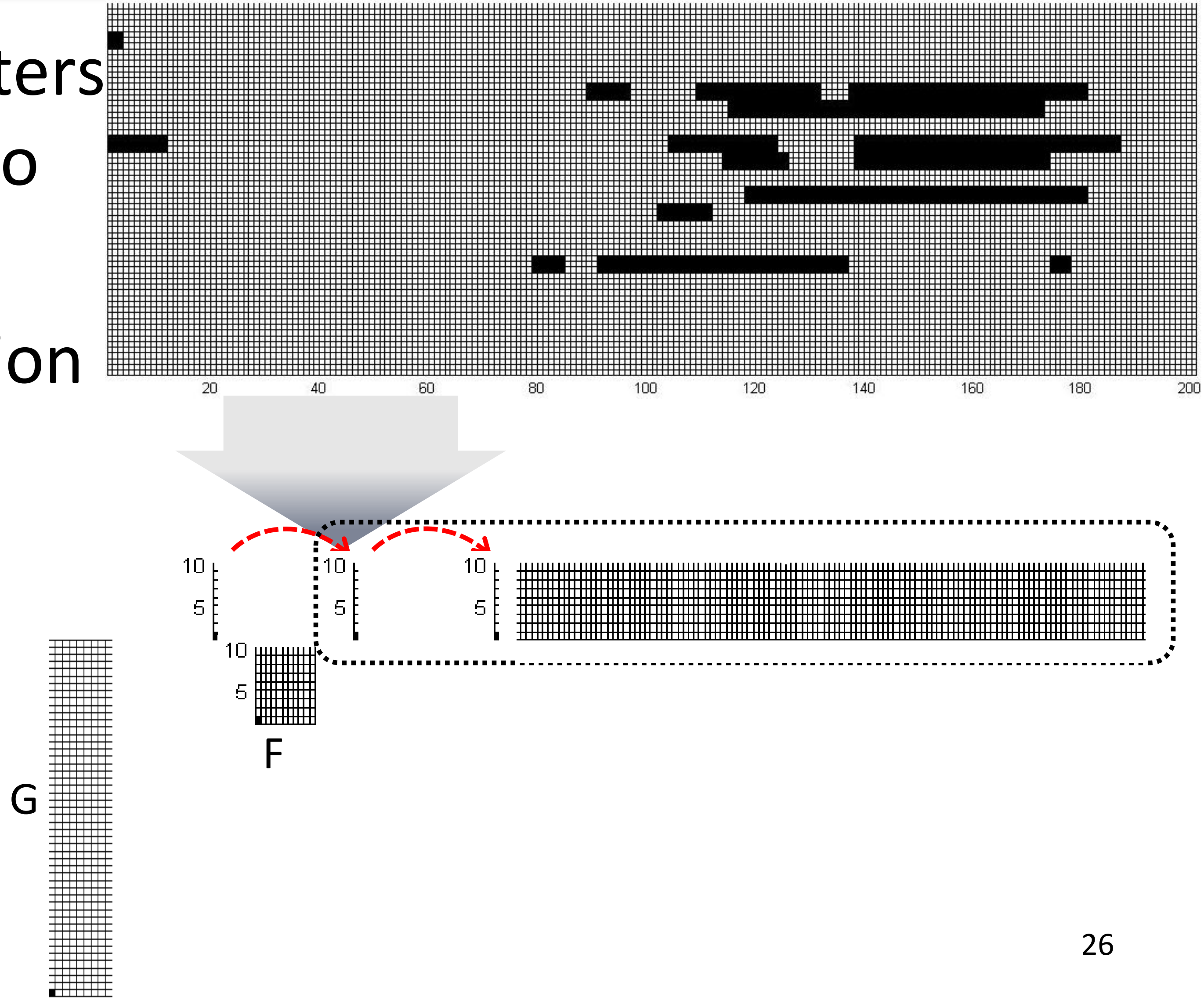




Is there a better tradeoff?

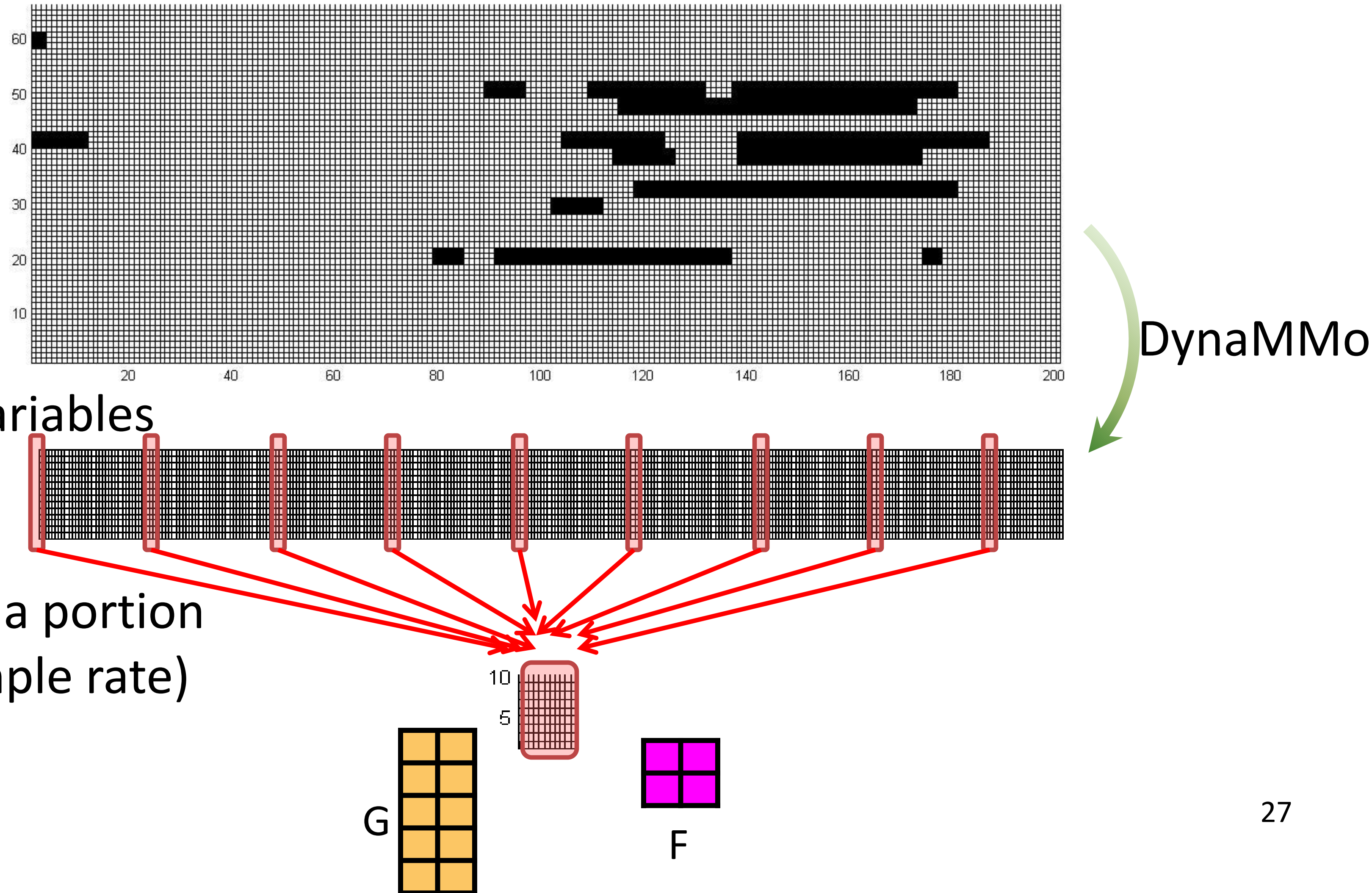
Store parameters of DynaMMo

But bad reconstruction



DynaMMo Compression:
sample & sync

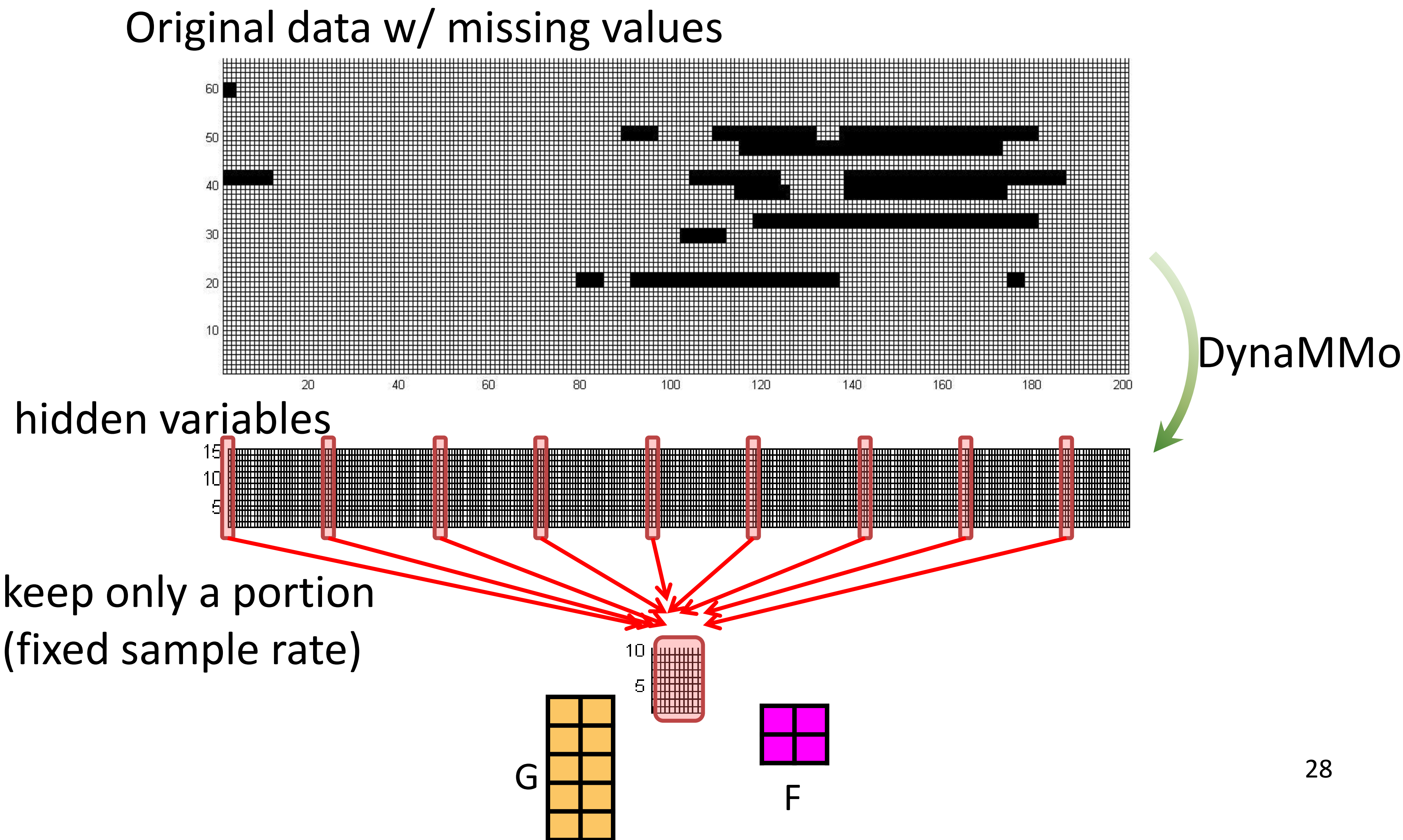
Original data w/ missing values



hidden variables

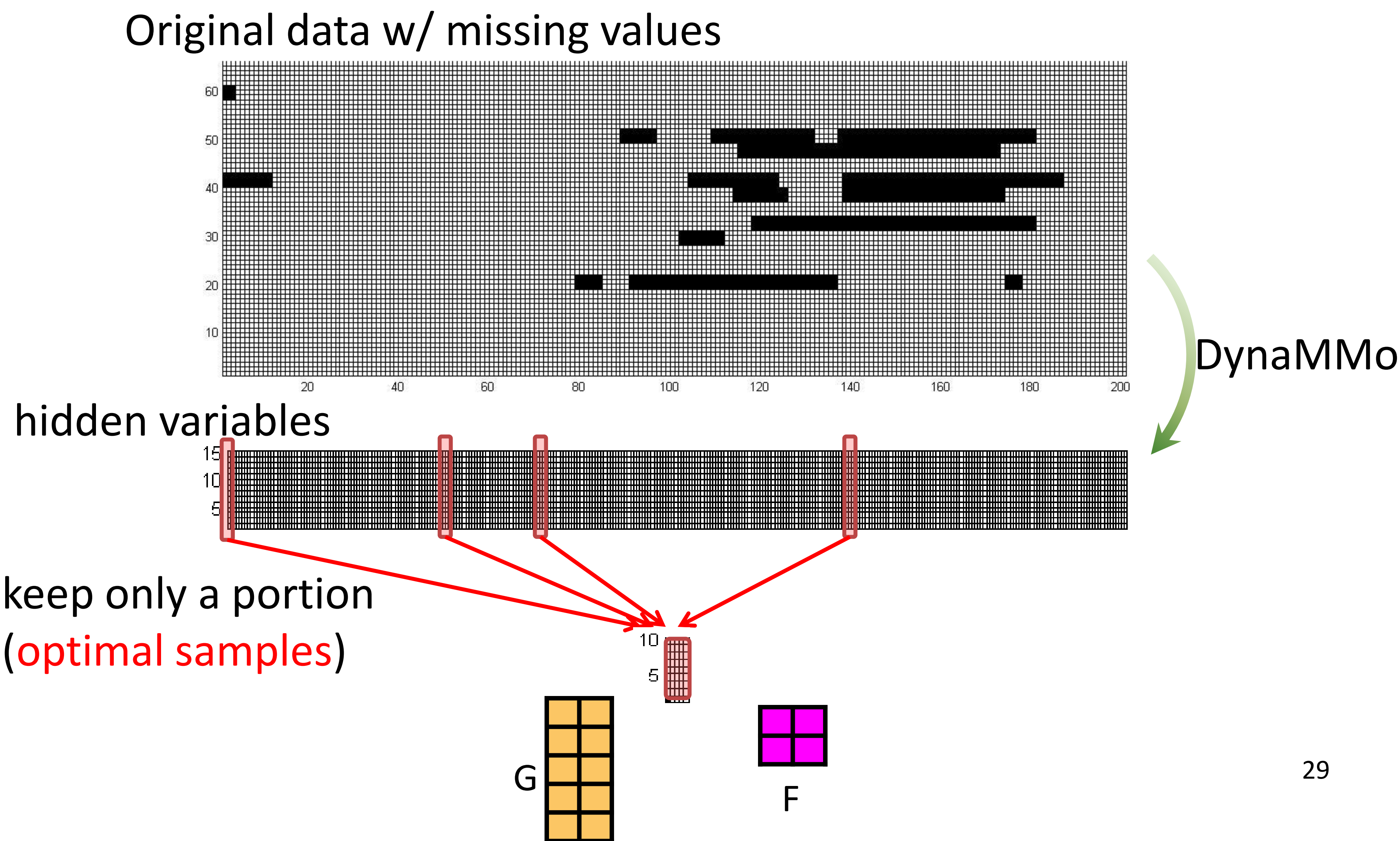
keep only a portion
(fixed sample rate)

Q: Can we do even better?



A: Yes, sample adaptively

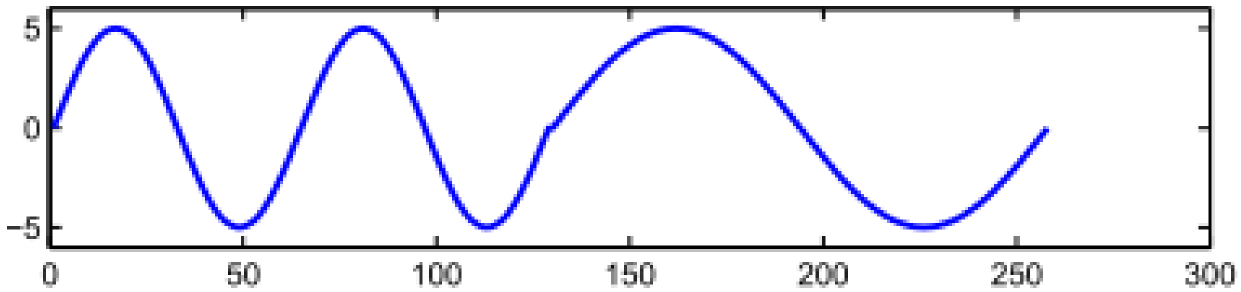
DynaMMo$_d$ Compression



How to Segment

- Segment by threshold on reconstruction error

original data

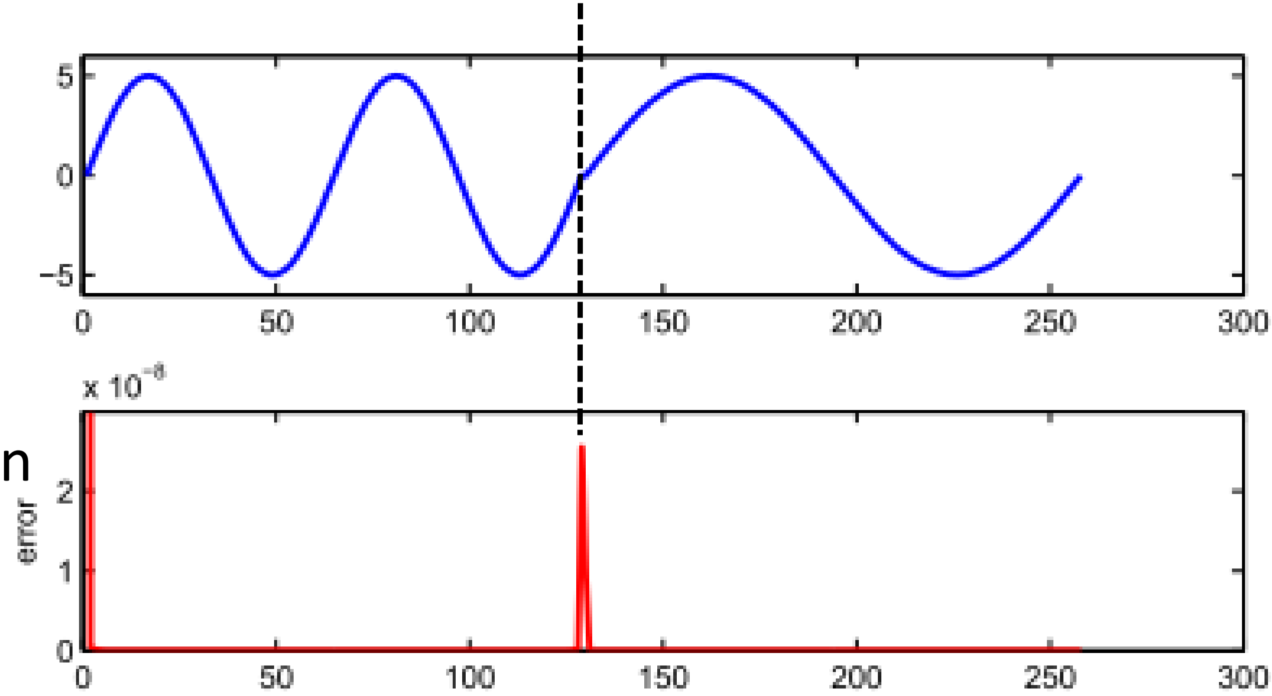


How to Segment

- Segment by threshold on reconstruction error

original data

reconstruction error



Outline

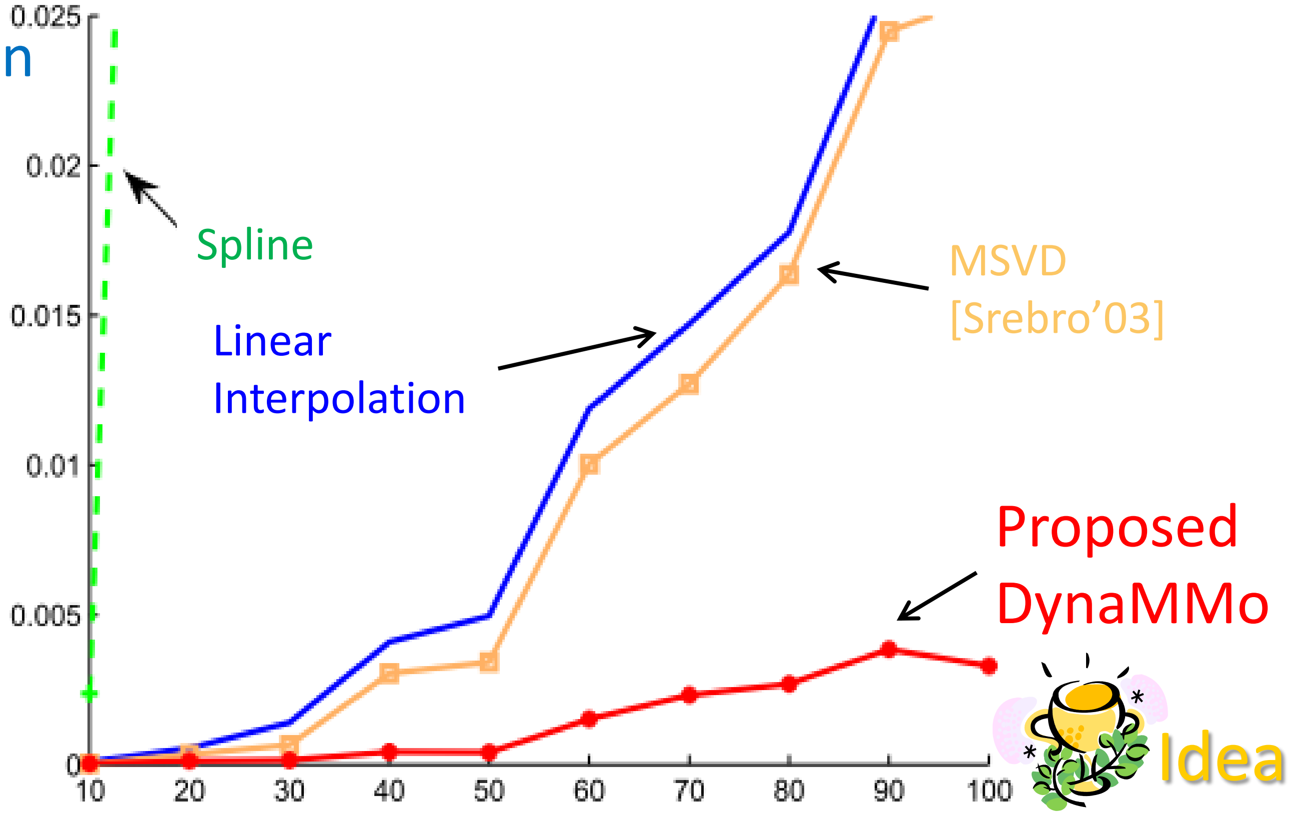
- Motivation
- Mining w/ Missing Values [Li+ 09, Li+10a]
  - Problem Definition
  - Proposed Method
    - T1: recovering
    - T2: compression
    - T3: segmentation
  - Results
- Feature Learning for Time Series [Li+10b , Li+11a]
- Summary of the remaining chapters
- Conclusion and Future Directions

Results

– Better Recovery of missing values

Reconstruction error

better



Average missing length

harder

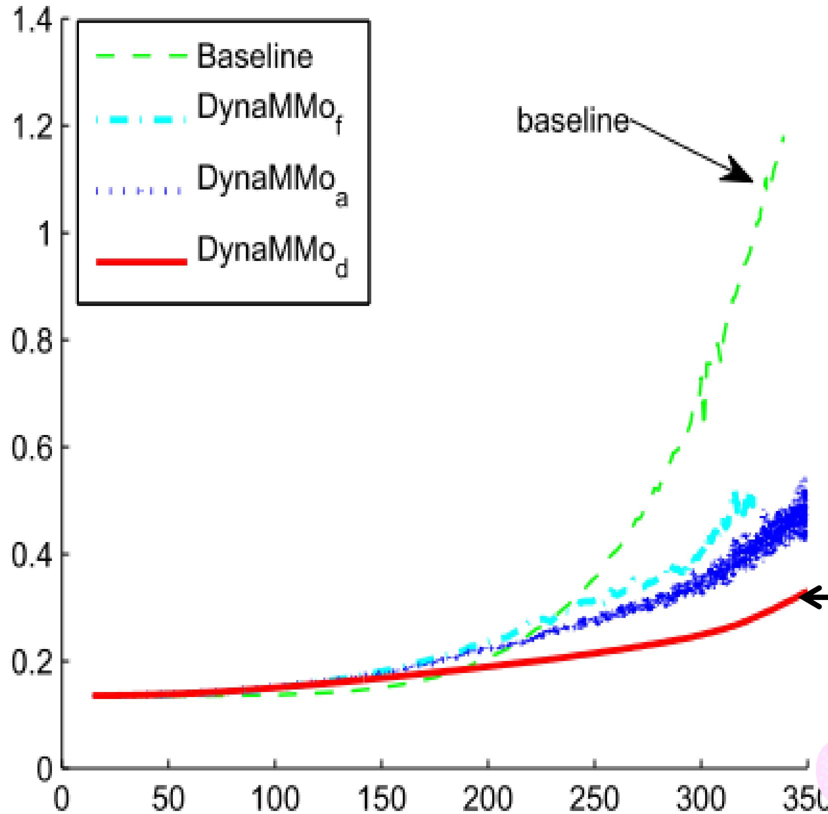
Dataset:
CMU Mocap #16
mocap.cs.cmu.edu

more results in [Li+2009]

Results – Better Compression

error

lower better



DynaMMo w/ optimal compression

Ideal

Compression ratio

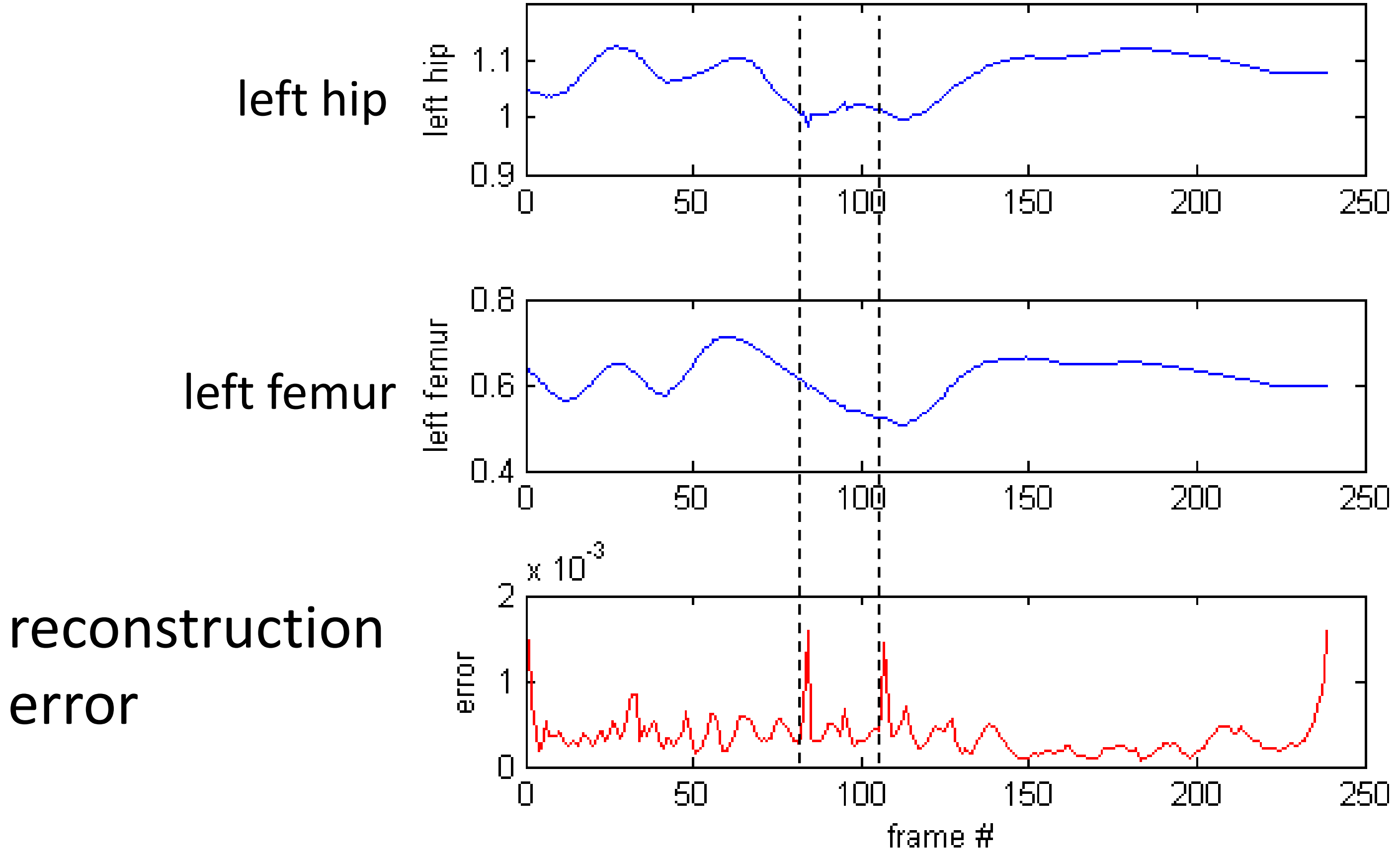
better

Dataset:
Chlorine levels

more results in [Li+2009]

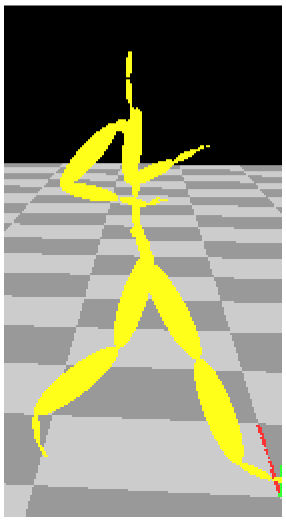
Results – Segmentation

- Find the *transition* during “running” to “stop”.



Results – Segmentation

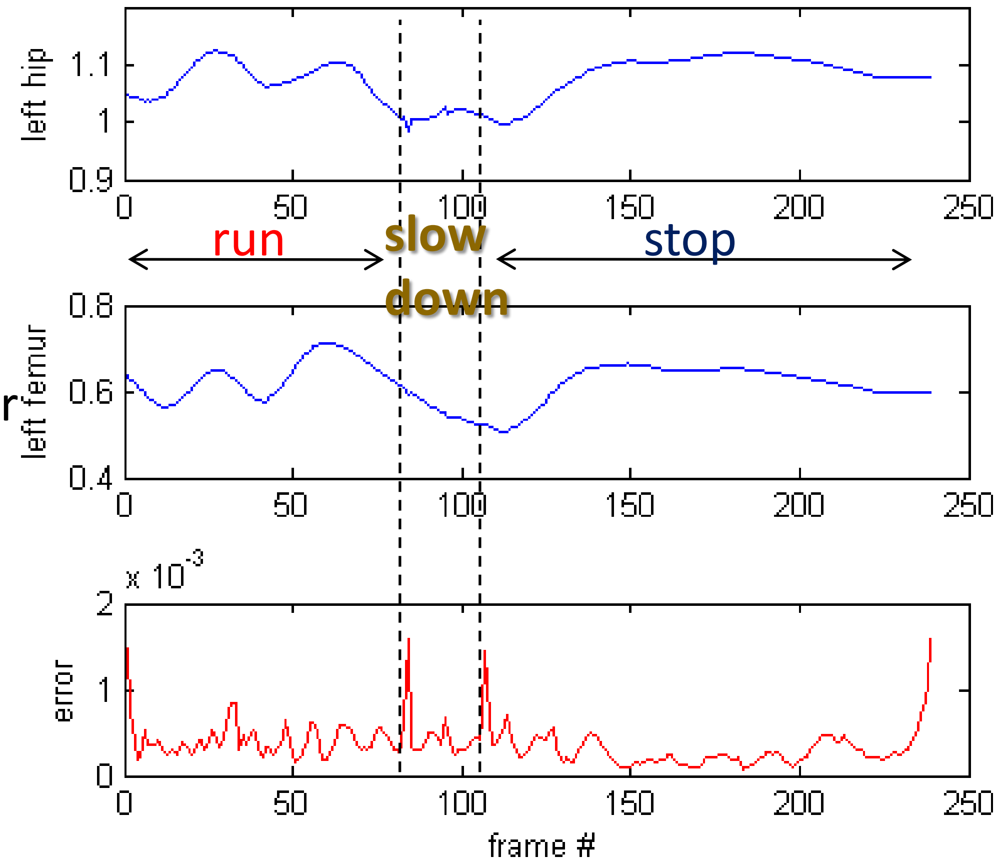
- Find the *transition* during “running” to “stop”.

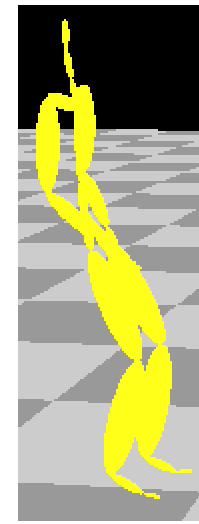


left hip

left femur

reconstruction error





A summary of my work on time series

Pattern discovery

- DynaMMo [Li 09]
- BoLeRO [Li 10a]
- ThermoCast [Li 11a]
- LazinessScore [Li08a]

Feature extraction

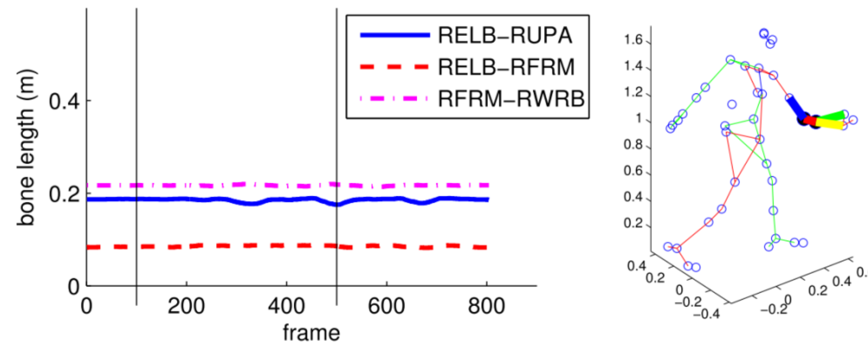
- PLiF [Li 10b]
- CLDS [Li 11a]

Parallel algorithm

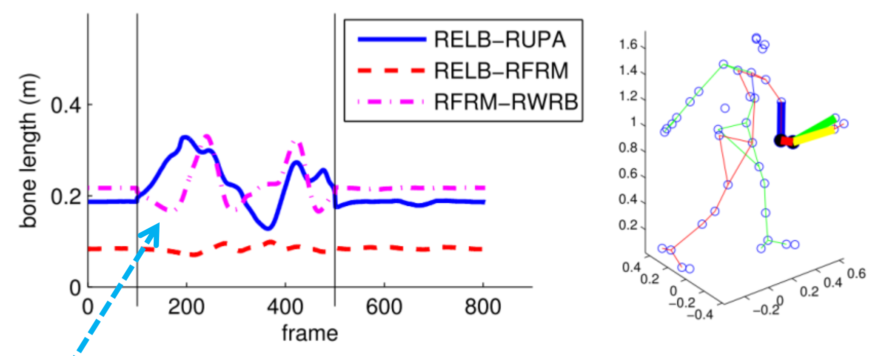
- Cut-And-Stitch [Li 08b]
- WindMine [Sakurai 11]

BoLeRO: including domain knowledge

- How to handle VERY LONG occlusions?
- Bone Length Constrained Occlusion filling in motion capture
  - Exploiting the skeleton of human body
  - [Lei Li et al, 2010a]



Original



LDS/DynaMMo

violation of bone length

(details)

BoLeRO

BoLeRO-Hard Constraint

$$\min \quad Q(X_m, \Theta)$$

$$\text{subject to} \|x_t^{(i)} - x_t^{(j)}\|^2 - d_{i,j}^2 = 0 \quad \forall \langle i, j, d_{i,j} \rangle \in B$$

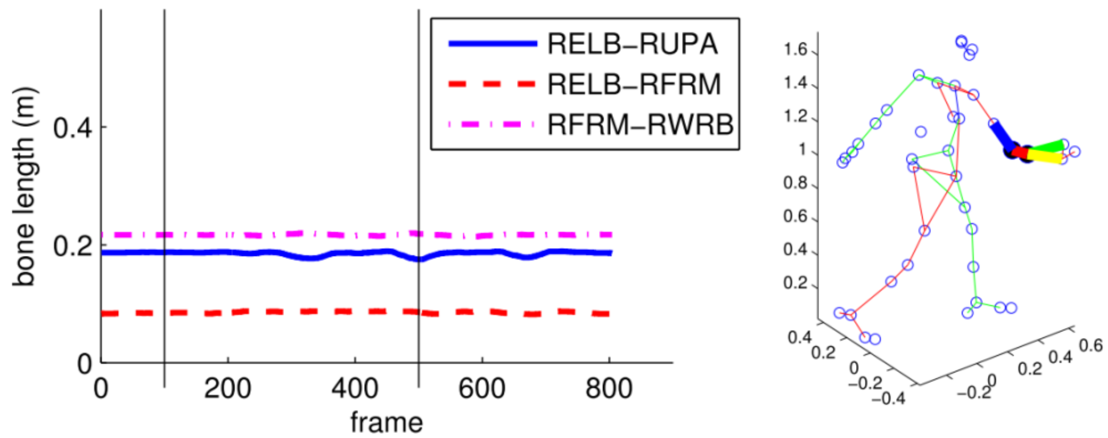
$$\begin{aligned} Q(X_m, \Theta) = &\frac{1}{2} \mathbb{E}[(\mathbf{z}_1 - \mu_0)^T \Gamma^{-1} (\mathbf{z}_1 - \mu_0) \\ &+ \sum_{t=2}^{T} (\mathbf{z}_t - \mathbf{F} \cdot \mathbf{z}_{t-1})^T \Lambda^{-1} (\mathbf{z}_t - \mathbf{F} \cdot \mathbf{z}_{t-1}) \\ &+ \sum_{t=1}^{T} (\mathbf{x}_t - \mathbf{G} \cdot \mathbf{z}_t)^T \Sigma^{-1} (\mathbf{x}_t - \mathbf{G} \cdot \mathbf{z}_t)] \\ &+ \frac{\log|\Gamma|}{2} + \frac{T-1}{2} \log|\Lambda| + \frac{T}{2} \log|\Sigma| \end{aligned}$$

BoLeRO-Soft Constraint

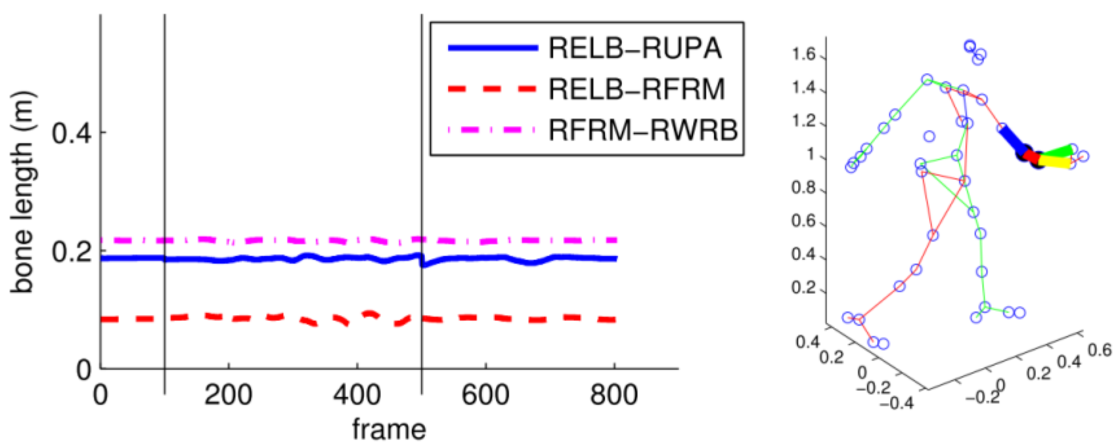
$$\begin{aligned} \min \quad & f(X_m, \Theta) \\ = & \frac{1}{2} \mathbb{E}\Big[(\mathbf{z}_1 - \mu_0)^T \Gamma^{-1} (\mathbf{z}_1 - \mu_0) \\ & + \sum_{t=2}^{T} (\mathbf{z}_t - \mathbf{F} \cdot \mathbf{z}_{t-1})^T \Lambda^{-1} (\mathbf{z}_t - \mathbf{F} \cdot \mathbf{z}_{t-1}) \\ & + \sum_{t=1}^{T} (\mathbf{x}_t - \mathbf{G} \cdot \mathbf{z}_t)^T \Sigma^{-1} (\mathbf{x}_t - \mathbf{G} \cdot \mathbf{z}_t)\Big] \\ & + \frac{\log|\Gamma|}{2} + \frac{T-1}{2} \log|\Lambda| + \frac{T}{2} \log|\Sigma| \\ & + \frac{\lambda}{2} \sum_{t=1}^{T} \sum_{\langle i,j,d_{i,j} \rangle \in B} (W_{t,i} | W_{t,j}) (\|\mathbf{x}_t^{(i)} - \mathbf{x}_t^{(j)}\|^2 - d_{i,j}^2)^2 \end{aligned}$$

where $W_{t,i} | W_{t,j} = W_{t,i} + W_{t,j} - W_{t,i} W_{t,j}$.

BoLeRO Results



Original



BoLeRO

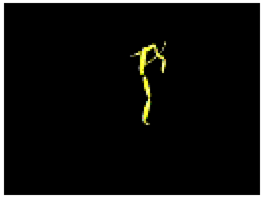
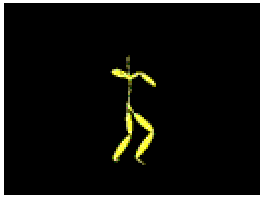
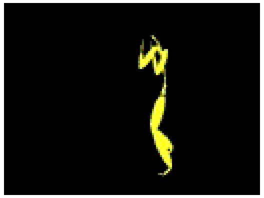
video

Outline

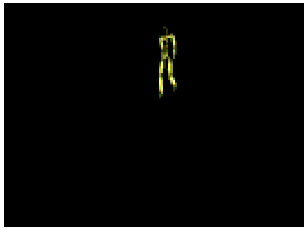
- Motivation
- Mining w/ Missing Values [Li+ 09, Li+10a]
- Feature Learning for Time Series [Li+10b, Li+11a]
  - Motivation and intuition
  - Complex-valued Linear Dynamical System
  - CLDS Clustering and interpretation
  - Experiments
- Summary of the remaining chapters
- Conclusion and Future Directions

Answering similarity queries

[Li et al, VLDB 2010]

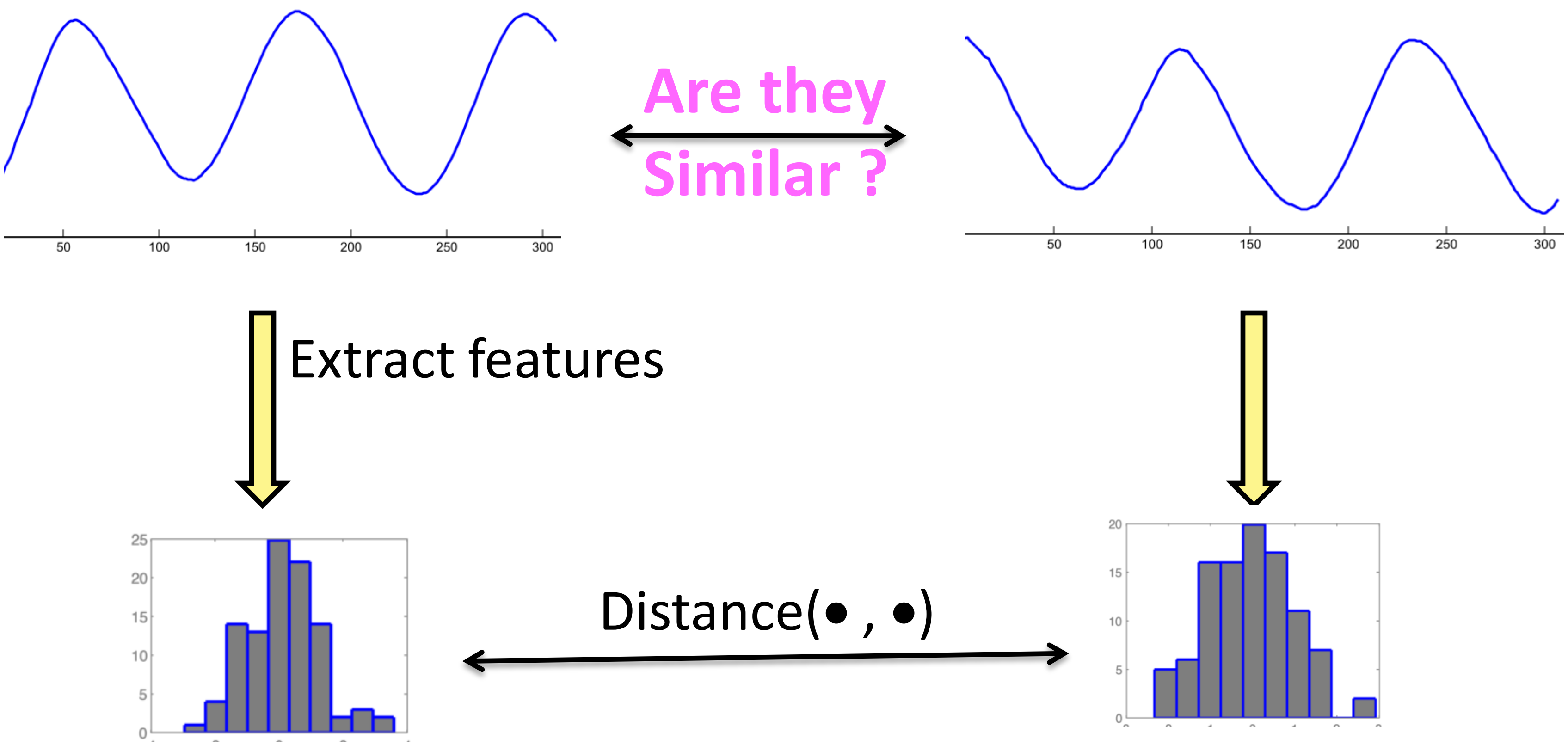
SELECT * **FROM**    

WHERE time_seq.

LIKE 

Central Problem

- Estimate “Similarity” among time sequences

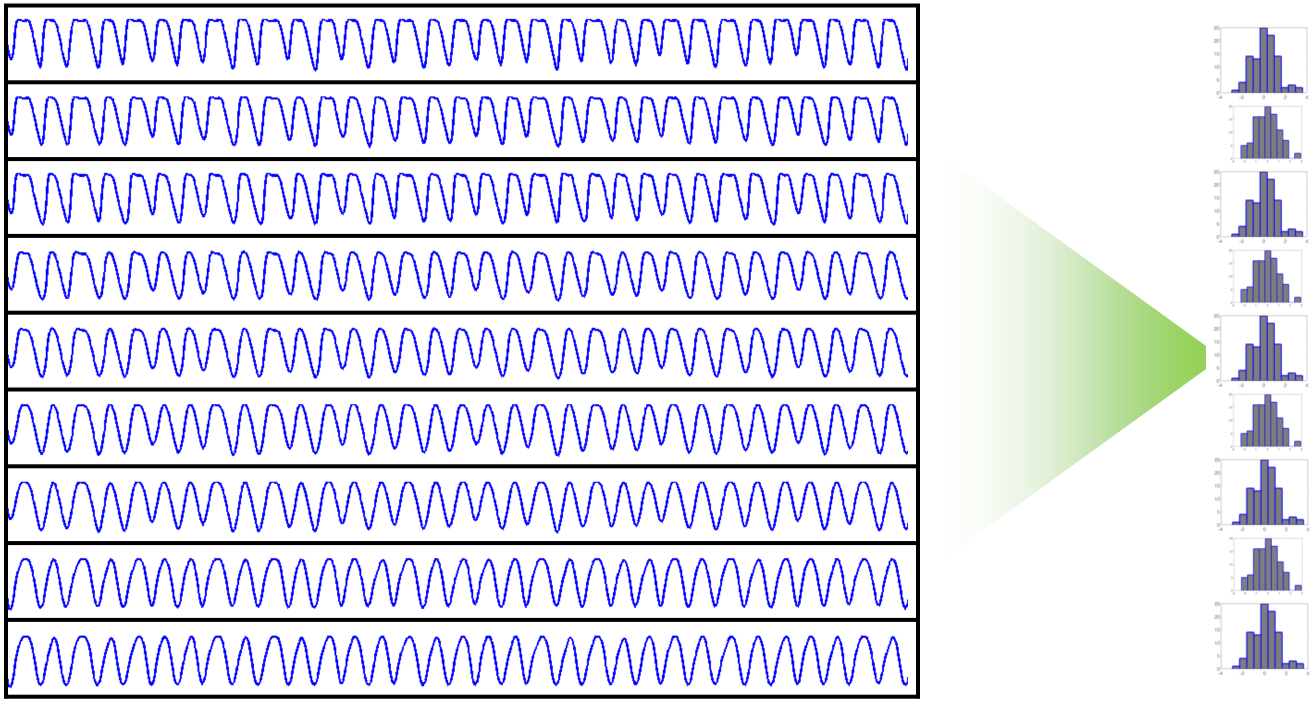


Features (e.g. average, Fourier)

features

What are good features?

Good features should agree with **human intuition**

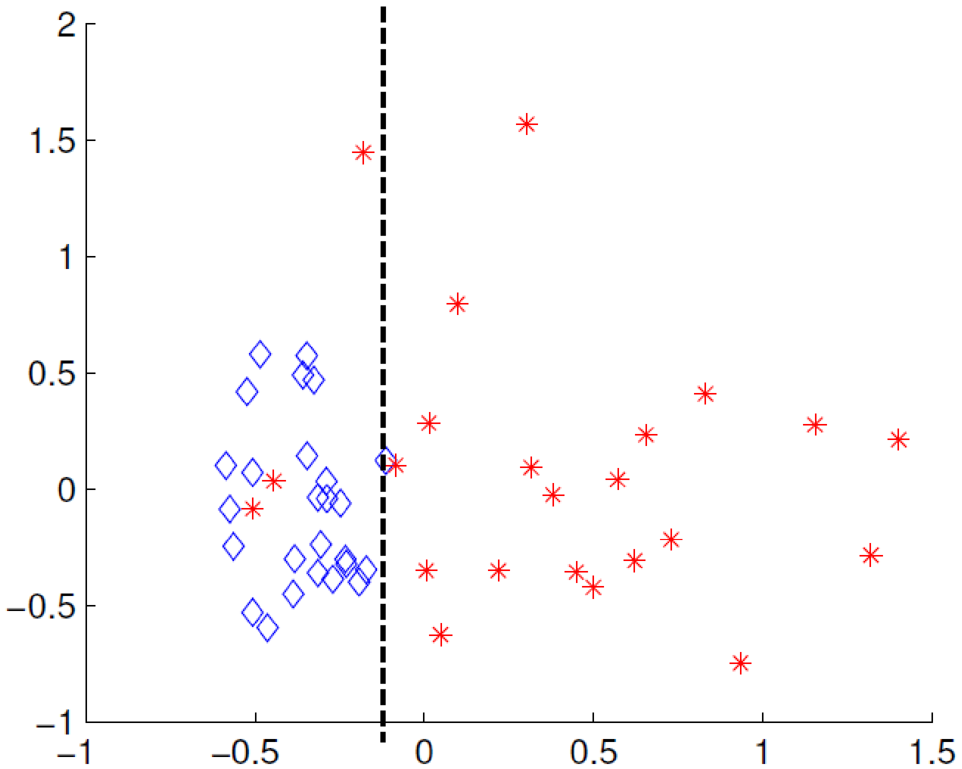


Requirements of good features:

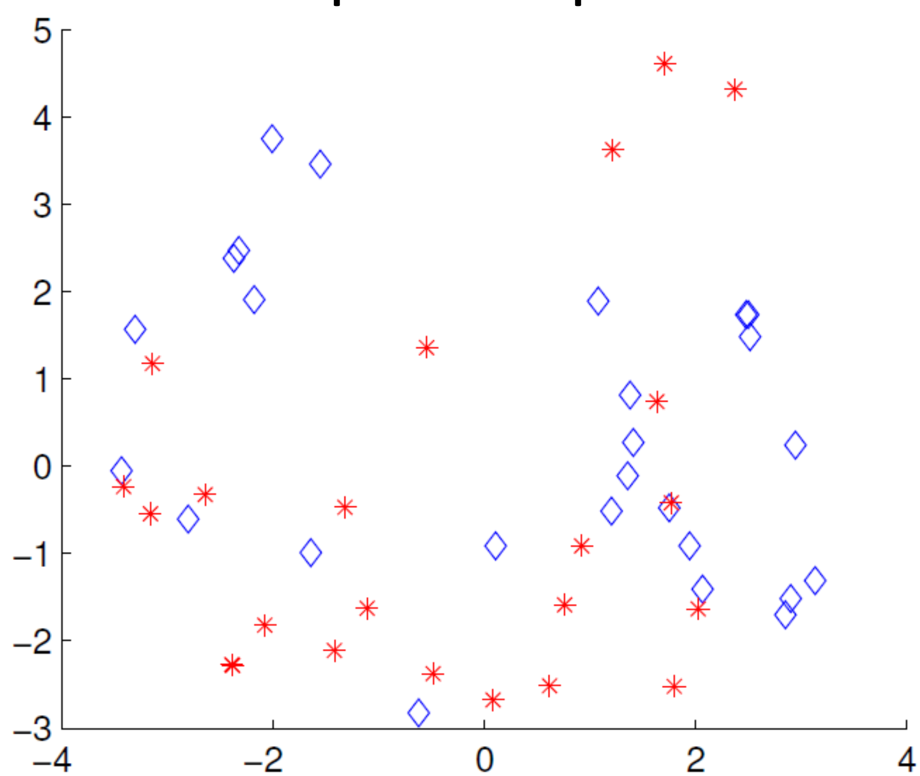
1. Time Shift
2. Frequency Proximity
3. Grouping Harmonics

Preview

CLDS two features



PCA top 2 components



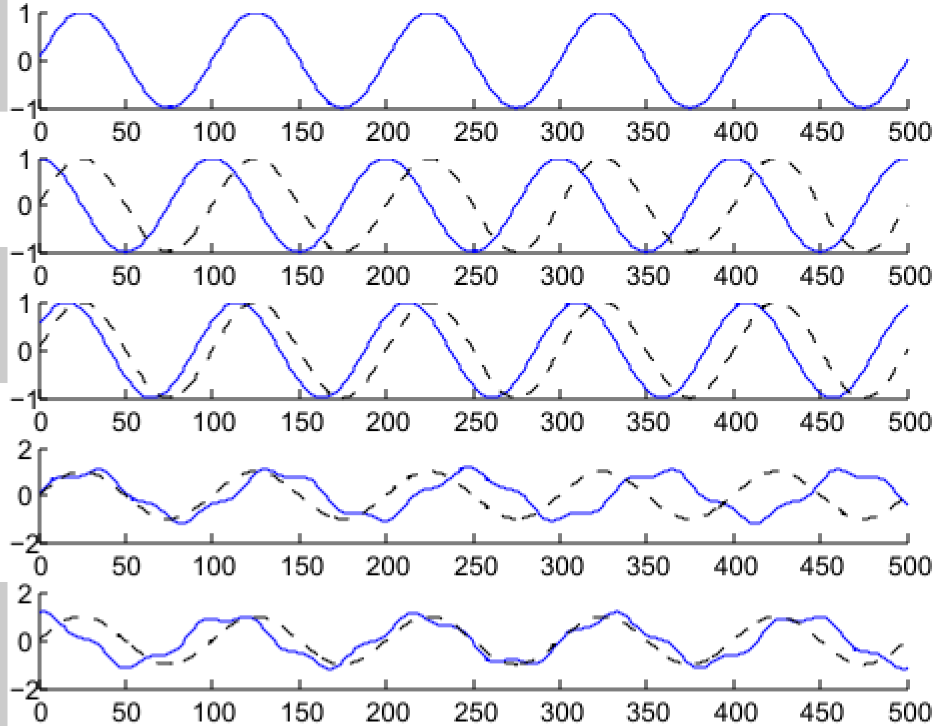
Accuracy = 93.9%

Accuracy = 51.0%

◊ walking motion ✳ running motion

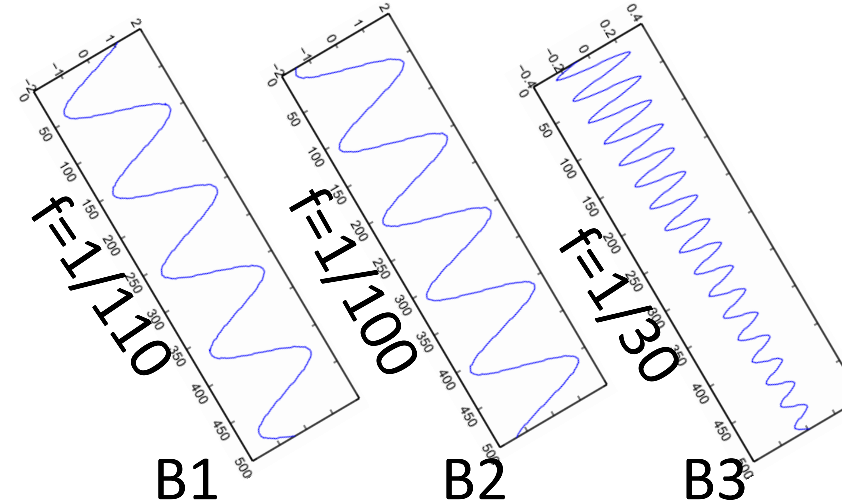
Example: synthetic signals

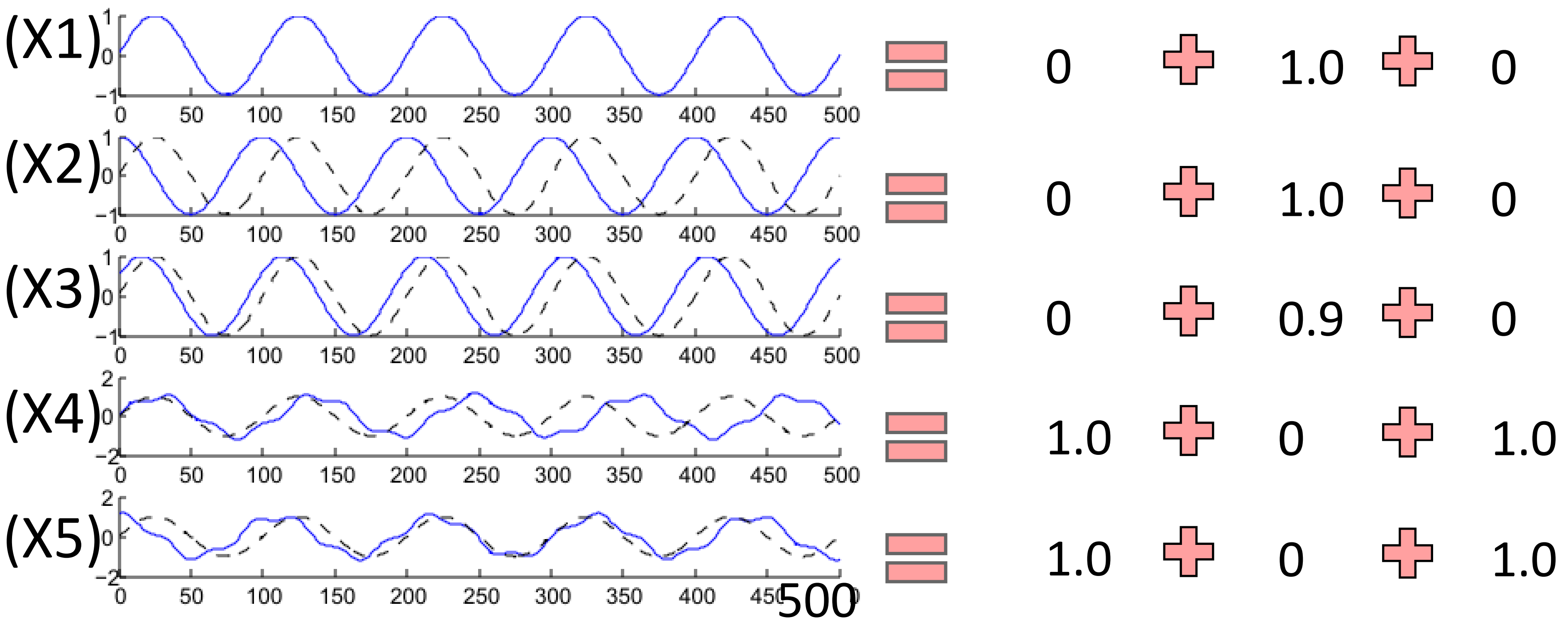
Equations	
(X1)	$\sin(2\pi t/100)$
(X2)	$\cos(2\pi t/100)$
(X3)	$\sin(2\pi t/98 + \pi/6)$
(X4)	$\sin(2\pi t/110) + 0.2\sin(2\pi t/30)$
(X5)	$\cos(2\pi t/110) + 0.2\sin(2\pi t/30 + \pi/4)$



Basic idea

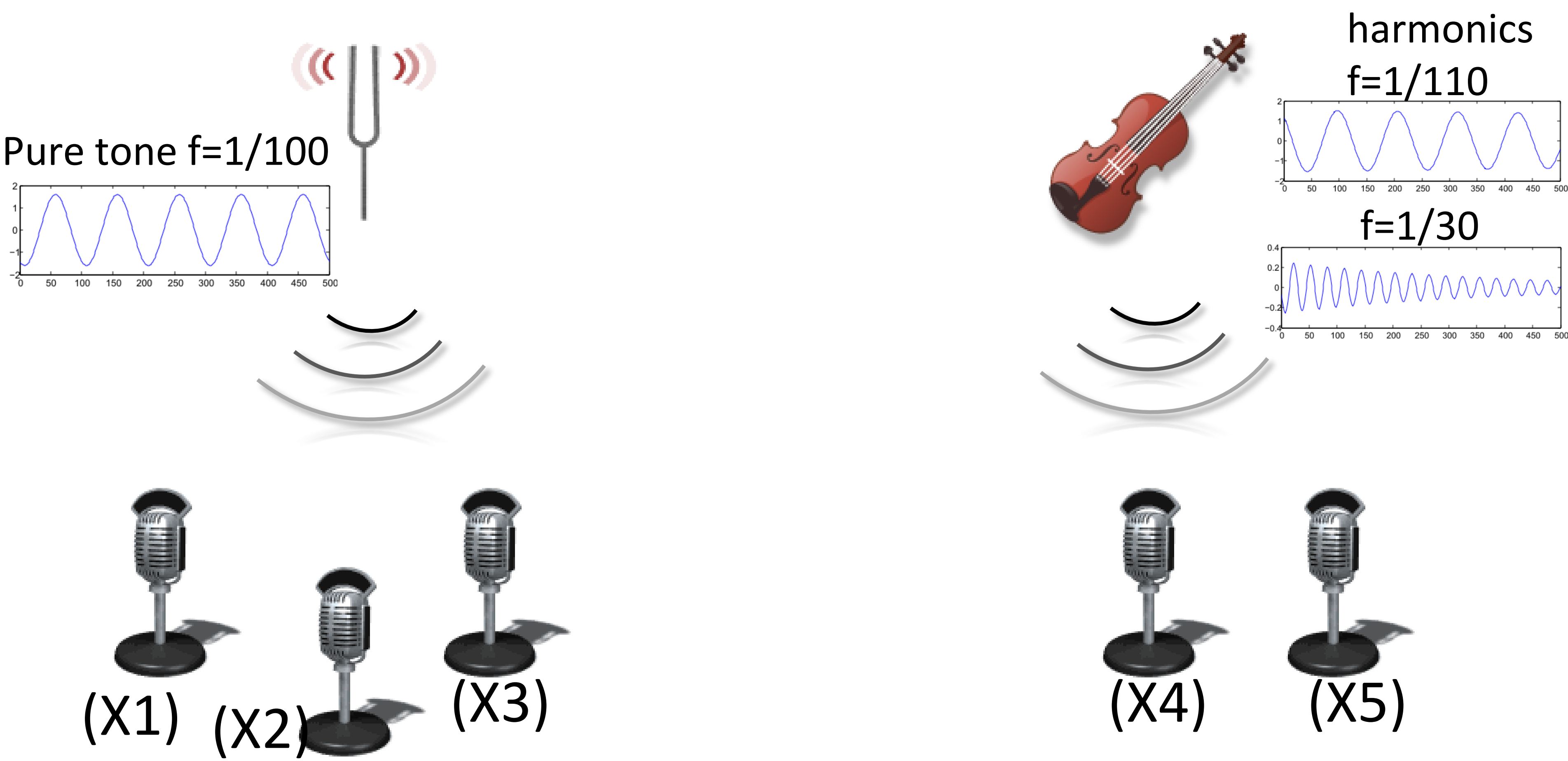
learning basis/harmonics





Mixing weights

Intuition of Basis

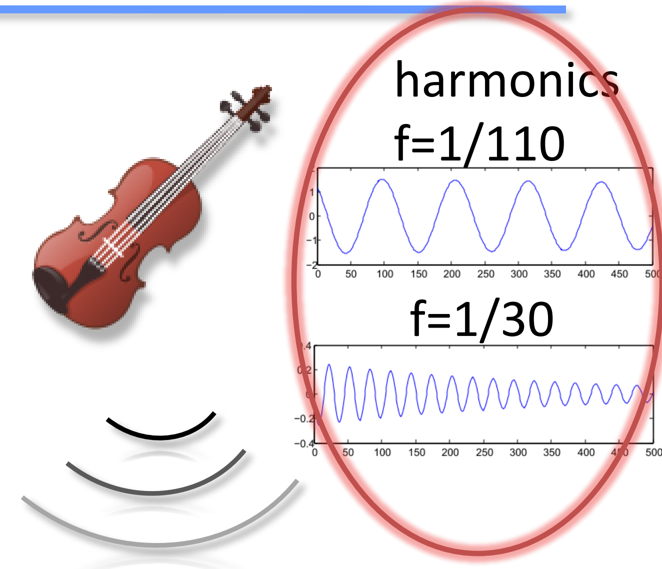


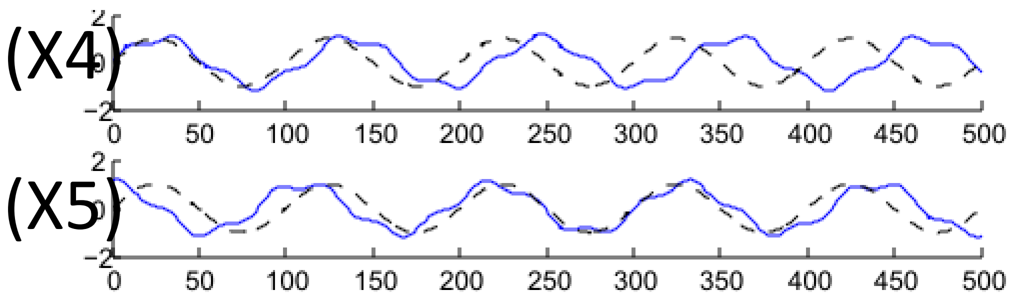
Mixing weights = participation strength of sound sources in observation (mic.)

Grouping Correlated Harmonics

Through PCA/SVD

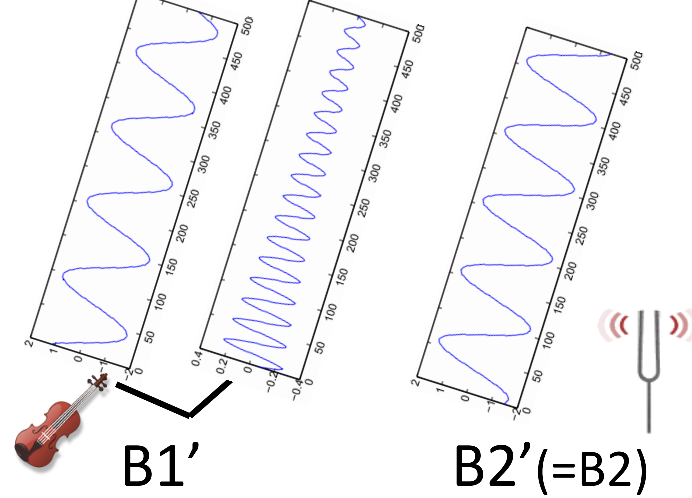
B1’ = {B1, B3}

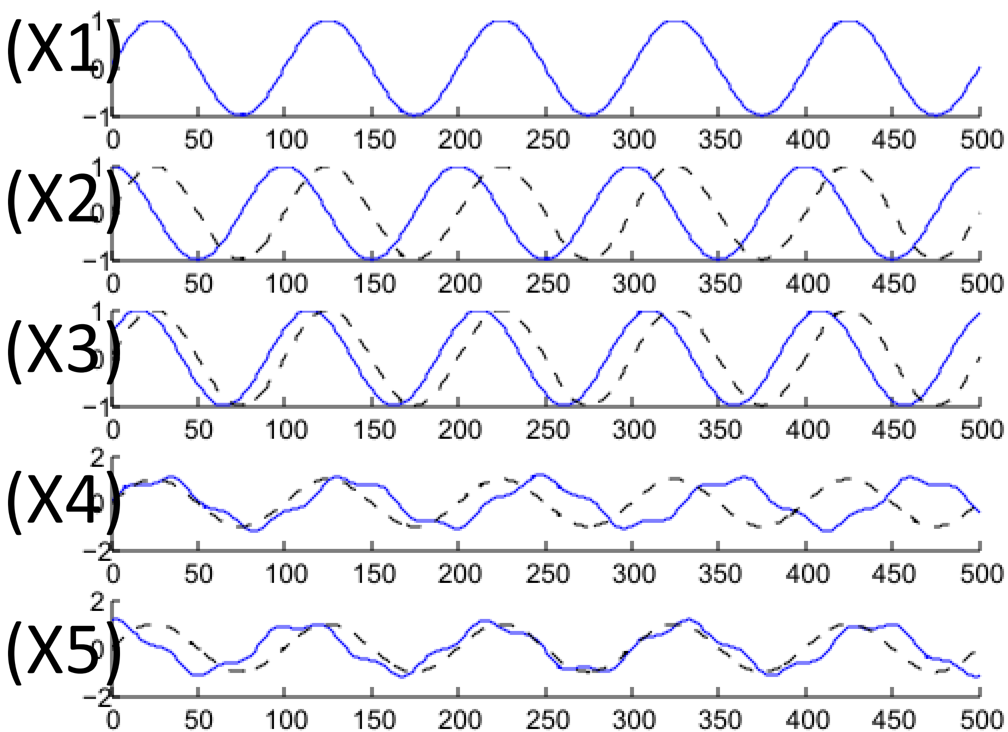




(X4) (X5)

Fingerprints





(X1) = 0 + 1.0

(X2) = 0 + 1.0

(X3) = 0 + 0.9

(X4) = 1.0 + 0

(X5) = 1.0 + 0

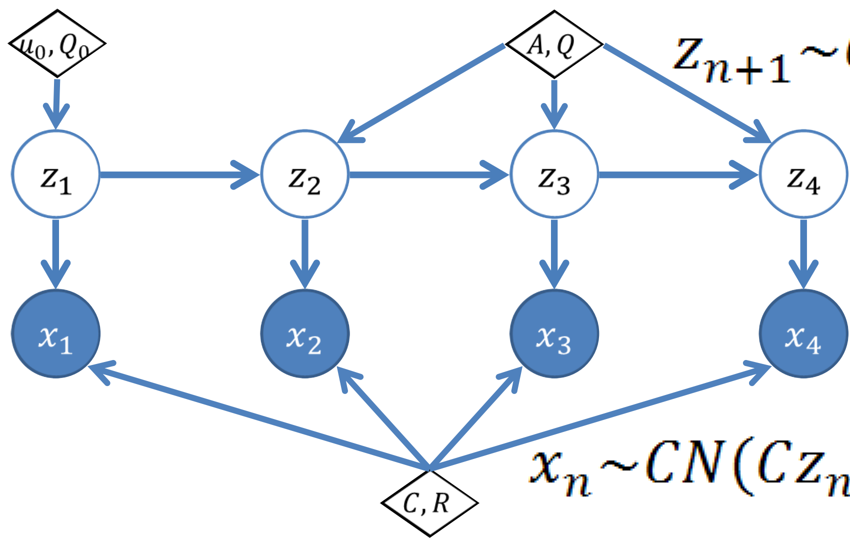
Outline

- Motivation
- Mining w/ Missing Values [Li+ 09, Li+10a]
- Feature Learning for Time Series [Li+10b, Li+11a]
  - Motivation and intuition
  - Complex-valued Linear Dynamical System
  - CLDS Clustering and interpretation
  - Experiments
- Summary of the remaining chapters
- Conclusion and Future Directions

How to learn the basis? Complex Linear Dynamical Systems

$z_1 \sim CN(\mu_0, Q_0)$

$z_{n+1} \sim CN(Az_n, Q)$



$x_n \sim CN(Cz_n, R)$

$$\mu_0 = \begin{pmatrix} \square \\ \square \\ \square \end{pmatrix} \qquad A = \begin{pmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{pmatrix} \qquad C = \begin{pmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{pmatrix}$$

$$Q_0 = \begin{pmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{pmatrix} \qquad Q = \begin{pmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{pmatrix} \qquad R = \begin{pmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{pmatrix}$$

[Li et al, ICML 2011]

Complex Normal Distribution

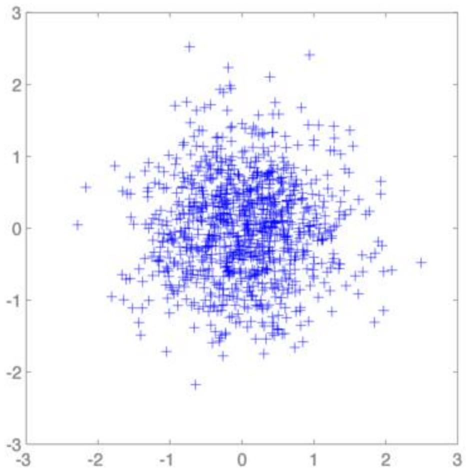
- Example: $x = a + ib$

  standard complex normal distribution

$$x \sim CN(0,1) \longleftrightarrow p(x) = \frac{1}{\pi} e^{-|x|^2}$$

$$\begin{pmatrix} a \\ b \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \frac{1}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right) \longleftrightarrow p(a,b)$$

$$= (2\pi)^{-1} |\Sigma|^{-\frac{1}{2}} e^{-\frac{1}{2}\left(\begin{pmatrix} a \\ b \end{pmatrix} - \mu\right)' \Sigma^{-1} \left(\begin{pmatrix} a \\ b \end{pmatrix} - \mu\right)}$$



(details)

Complex Normal Distribution

- x is said to follow the complex normal distribution, if its p.d.f

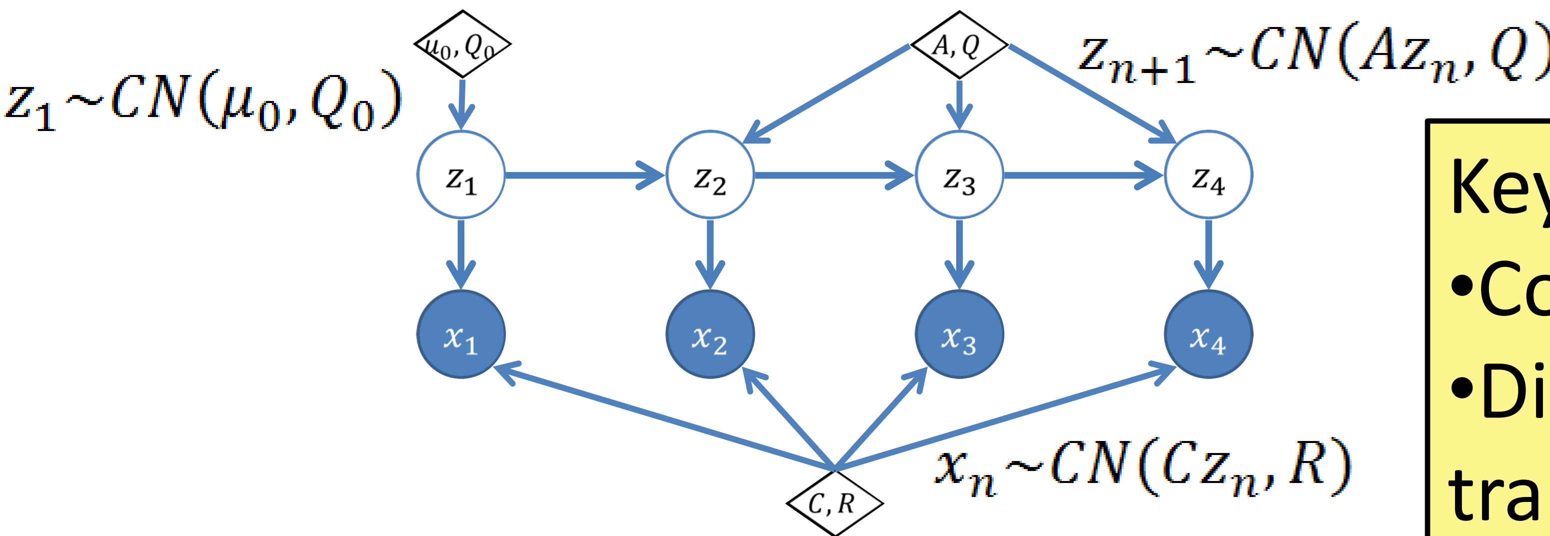
$\boldsymbol{x} \sim \mathcal{CN}(\mu, H),\ if\ its\ p.d.f\ is$

$$p(\boldsymbol{x}) = \pi^{-m}|H|^{-1}\exp(-(\boldsymbol{x}-\mu)^* H^{-1}(\boldsymbol{x}-\mu))$$

H is hermitian matrix, $(\cdot)^*$ is conjugate transpose

[Goodman, 1963]

Complex Linear Dynamical Systems



Key points
- Complex
- Diagonal transition

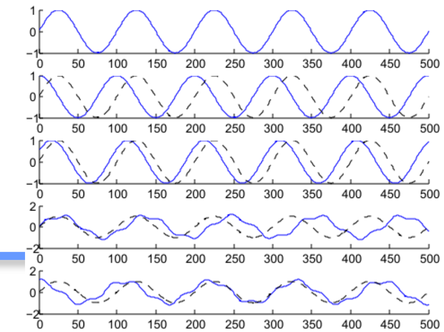
$$\mu_0 = \begin{pmatrix} \square \\ \square \\ \square \end{pmatrix} \qquad A = \begin{pmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{pmatrix} \qquad C = \begin{pmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{pmatrix}$$

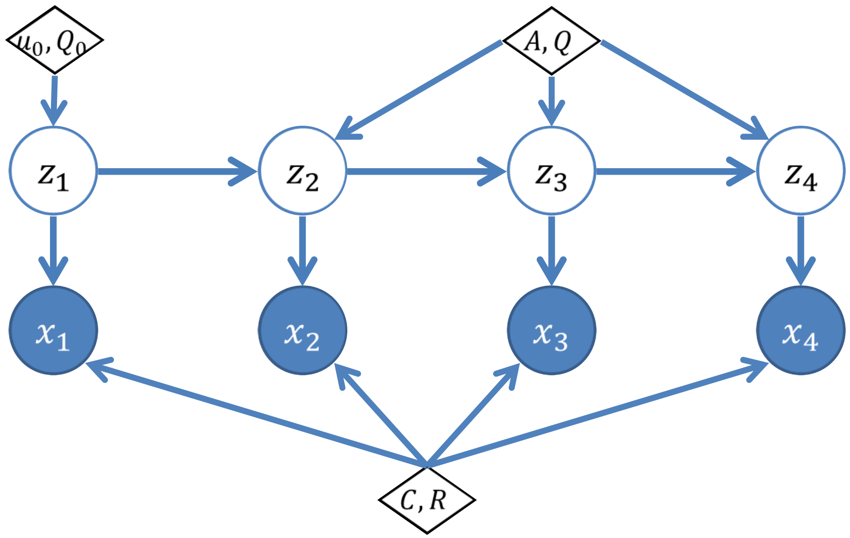
$$Q_0 = \begin{pmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{pmatrix} \qquad Q = \begin{pmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{pmatrix} \qquad R = \begin{pmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{pmatrix}$$

Rationale:
- Faster
- More robust
- Better clustering

Feature=output matrix

Example





$$z_1 \sim CN(\mu_0, Q_0)$$
$$z_{n+1} \sim CN(Az_n, Q)$$
$$x_n \sim CN(Cz_n, R)$$

A: transition matrix

C: output matrix

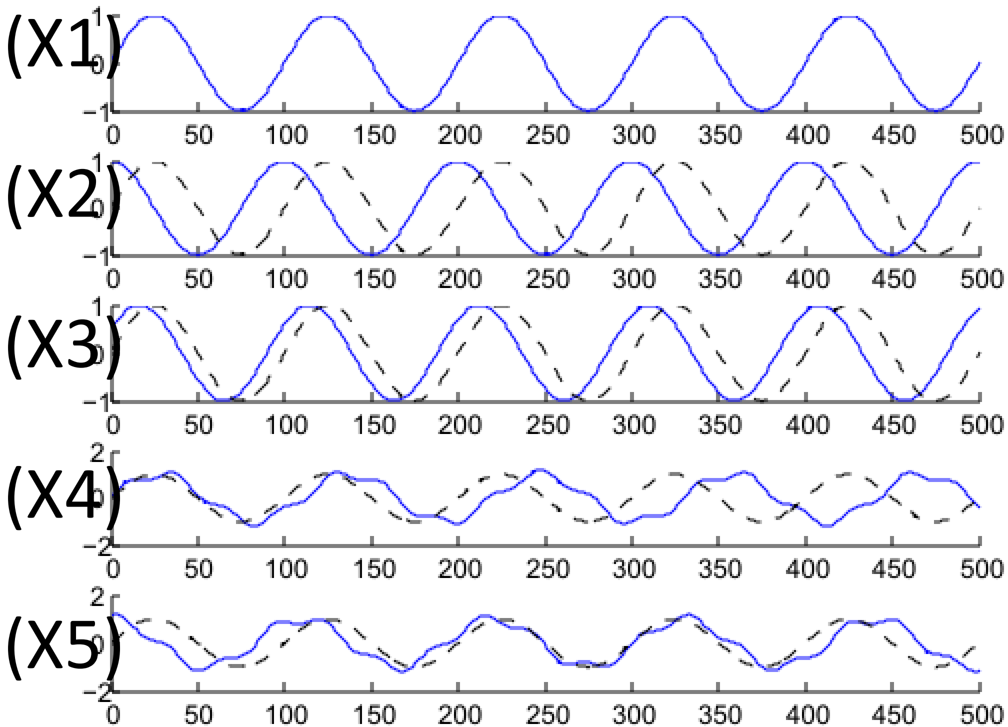
$$A = \begin{pmatrix} 0.9984 + 0.0571i & 0 & 0 \\ 0 & 0.9980 + 0.0628i & 0 \\ 0 & 0 & 0.9781 + 0.2079i \end{pmatrix}$$

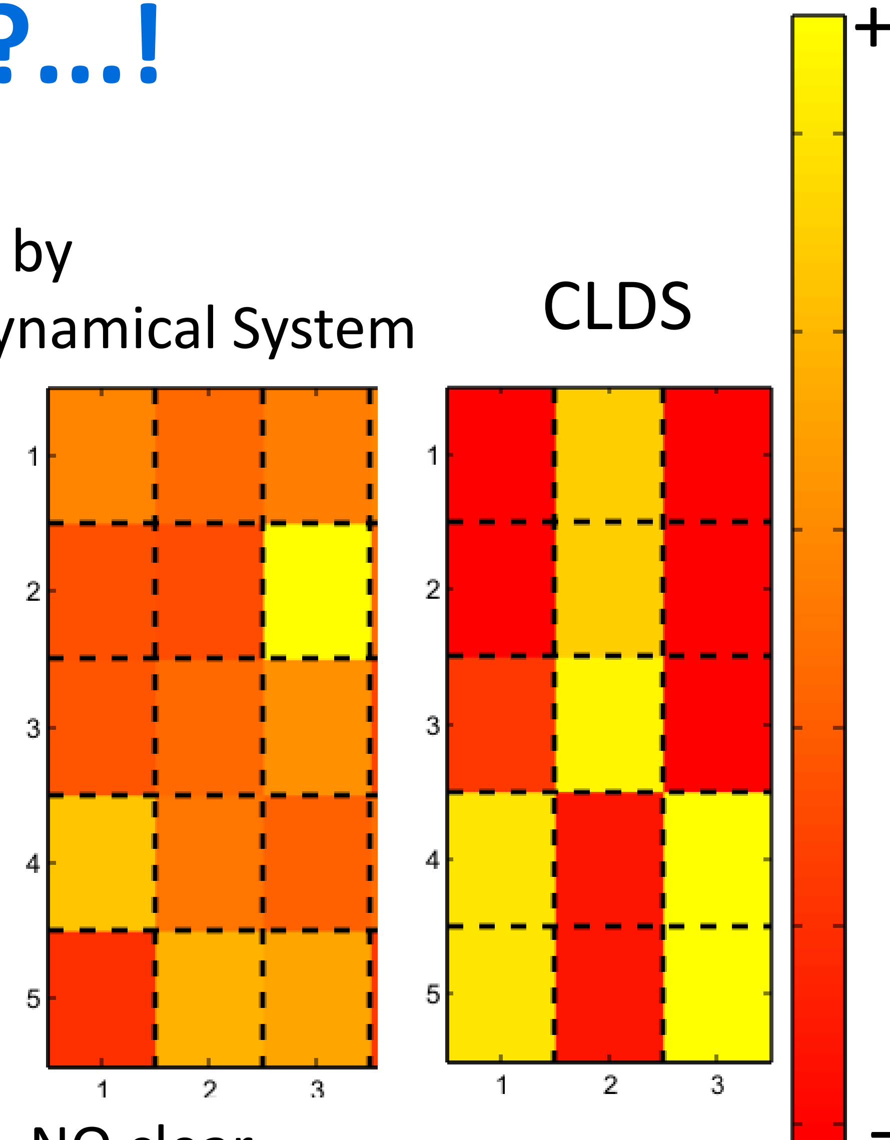
$$C = \begin{pmatrix} 0 & 1 & 0 \\ 0 & i & 0 \\ 0 & 0.866 + 0.5i & 0 \\ 1 & 0 & 1 \\ i & 0 & 0.707 + 0.707i \end{pmatrix}$$

Complex is Simpler?...!

Features by
Linear Dynamical System

CLDS





NO clear
clustering!

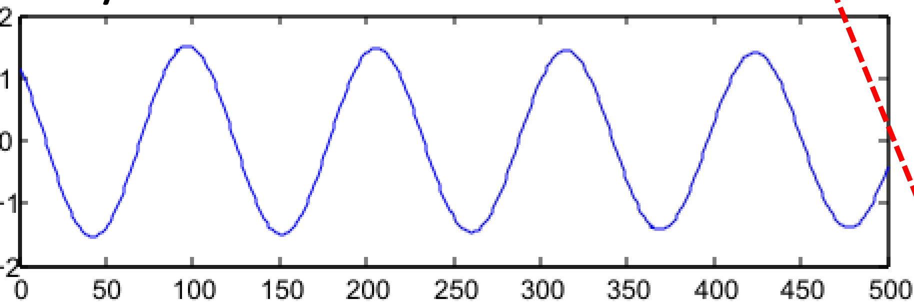
Outline

- Motivation
- Mining w/ Missing Values [Li+ 09, Li+10a]
- Feature Learning for Time Series [Li+10b, Li+11a]
  - Motivation and intuition
  - Complex-valued Linear Dynamical System
  - CLDS Clustering and interpretation
  - Experiments
- Summary of the remaining chapters
- Conclusion and Future Directions

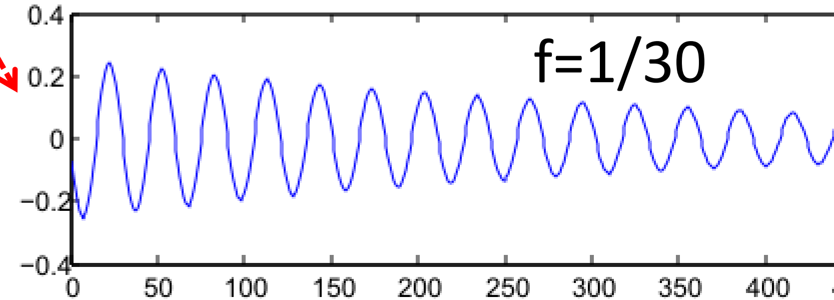
Simple interpretation for “Complex” solution

$$A = \begin{pmatrix} 0.9984 + 0.0571i & 0 & 0 \\ 0 & 0.9980 + 0.0628i & 0 \\ 0 & 0 & 0.9781 + 0.2079i \end{pmatrix}$$

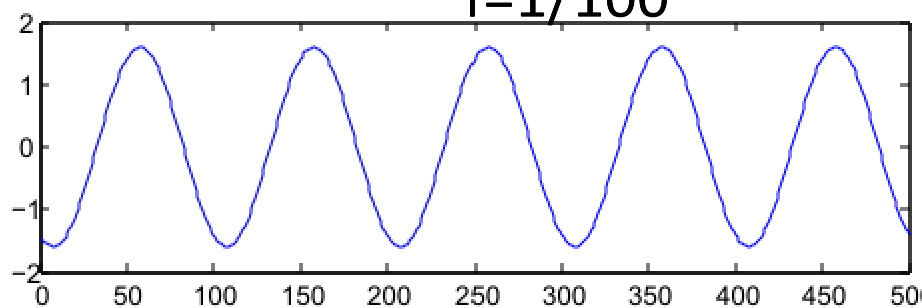
f=1/110



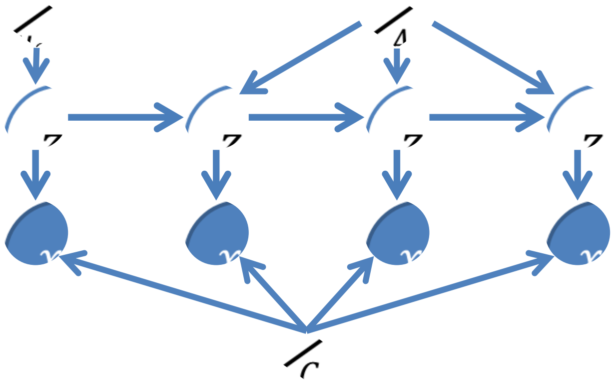
f=1/30



f=1/100



Simple interpretation for “Complex” solution



$$C = \begin{pmatrix} 0 & 1 & 0 \\ 0 & i & 0 \\ 0 & 0.866 + 0.5i & 0 \\ 1 & 0 & 1 \\ i & 0 & 0.707 + 0.707i \end{pmatrix}$$

Take magnitude

Feature Matrix
$F = abs(C)$

0	1	0
0	1	0
0	1	0
1	0	1
1	0	1

CLDS Clustering Algorithm

data: X, k

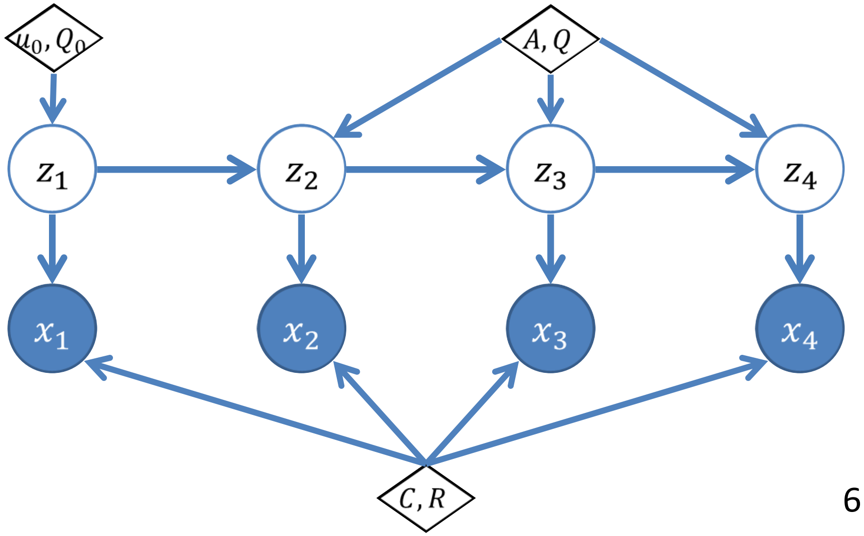
Step 1. θ ← learn diagonal CLDS (X)

Step 2. C_m ← abs(C)

Step 3. F ← PCA(C_m)

Step 4. group ← k-means(F, k)

features



Parameter Learning

min $\mathcal{L}(\theta) = \mathbb{E}_{\boldsymbol{Z}|\boldsymbol{X}}[-\log P(\boldsymbol{X}, \boldsymbol{Z}|\theta)]$

$$= \log|\boldsymbol{Q}_0| + \mathbb{E}[(\boldsymbol{z}_1 - \boldsymbol{\mu}_0)^* \boldsymbol{Q}_0^{-1} (\boldsymbol{z}_1 - \boldsymbol{\mu}_0)]$$

$$+ \mathbb{E}[\sum_{n=1}^{N-1} (\boldsymbol{z}_{n+1} - \boldsymbol{A} \cdot \boldsymbol{z}_n)^* \cdot Q^{-1} \cdot (\boldsymbol{z}_{n+1} - \boldsymbol{A} \cdot \boldsymbol{z}_n)] + (N-1)\log|Q|$$

$$+ \mathbb{E}[\sum_{n=1}^{N} (\boldsymbol{x}_n - \boldsymbol{C} \cdot \boldsymbol{z}_n)^* \cdot R^{-1} \cdot (\boldsymbol{x}_n - \boldsymbol{C} \cdot \boldsymbol{z}_n)] + N \log|R|$$

EM algorithm (complex-Fit)

- E-step: compute posterior $P(z_n|x_1, \ldots, x_N)$ and $P(z_n, z_{n+1}|x_1, \ldots, x_N)$
- M-step: update the parameters to optimize $L(\theta)$

Optimizing real-valued functions of complex variables

- With complex variables:
  - $\dfrac{\partial f}{\partial x} = 0$ AND $\dfrac{\partial f}{\partial \bar{x}} = 0$
- EM algorithm (complex-Fit)

$$\frac{\partial}{\partial \boldsymbol{\mu}_0}\mathcal{L} = 0 \quad \frac{\partial}{\partial \overline{\boldsymbol{\mu}_0}}\mathcal{L} = 0 \quad \frac{\partial}{\partial \boldsymbol{Q}_0}\mathcal{L} = 0 \quad \frac{\partial}{\partial \overline{\boldsymbol{Q}_0}}\mathcal{L} = 0$$

$$\frac{\partial}{\partial \boldsymbol{A}}\mathcal{L},\ \frac{\partial}{\partial \overline{\boldsymbol{A}}}\mathcal{L},\ \frac{\partial}{\partial \boldsymbol{Q}}\mathcal{L},\ \frac{\partial}{\partial \overline{\boldsymbol{Q}}}\mathcal{L},\ \frac{\partial}{\partial \boldsymbol{C}}\mathcal{L},\ \frac{\partial}{\partial \overline{\boldsymbol{C}}}\mathcal{L},\ \frac{\partial}{\partial \boldsymbol{R}}\mathcal{L},\frac{\partial}{\partial \overline{\boldsymbol{R}}}\mathcal{L} = 0$$

$$\boldsymbol{a} =(\boldsymbol{Q}^{-1} \circ (\sum_{n=1}^{N-1} \mathbb{E}[\boldsymbol{z}_n \cdot \boldsymbol{z}_n^*])^T)^{-1} \cdot (\boldsymbol{Q}^{-1} \circ (\sum_{n=1}^{N-1} \mathbb{E}[\boldsymbol{z}_{n+1} \cdot \boldsymbol{z}_n^*])^T) \cdot \mathbf{1}$$

$$\begin{aligned}\boldsymbol{Q} =& \frac{1}{N-1} \sum_{n=1}^{N-1} \Big(\mathbb{E}[\boldsymbol{z}_{n+1} \cdot \boldsymbol{z}_{n+1}^*] - \mathbb{E}[\boldsymbol{z}_{n+1} \cdot (\boldsymbol{a} \circ \boldsymbol{z}_n)^*] \\ & - \mathbb{E}[(\boldsymbol{a} \circ \boldsymbol{z}_n) \cdot \boldsymbol{z}_{n+1}^*] + \mathbb{E}[(\boldsymbol{a} \circ \boldsymbol{z}_n) \cdot (\boldsymbol{a} \circ \boldsymbol{z}_n)^*] \Big)\end{aligned}$$

Outline

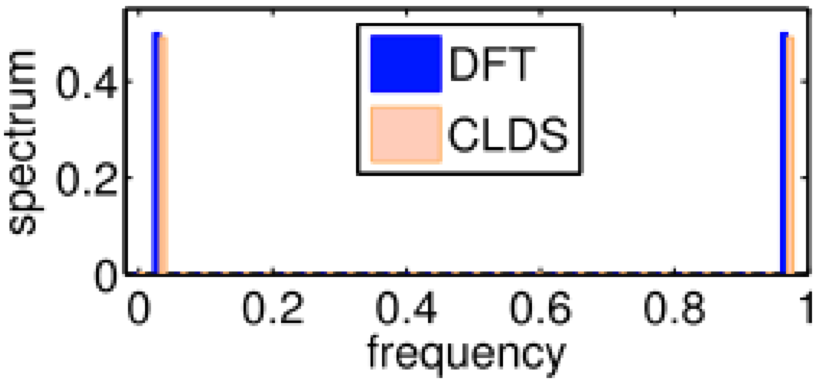
- Motivation
- Mining w/ Missing Values [Li+ 09, Li+10a]
- Feature Learning for Time Series [Li+10b, Li+11a]
  - Motivation and intuition
  - Complex-valued Linear Dynamical System
  - CLDS Clustering and interpretation
  - Experiments
- Summary of the remaining chapters
- Conclusion and Future Directions

DFT as a special case of CLDS

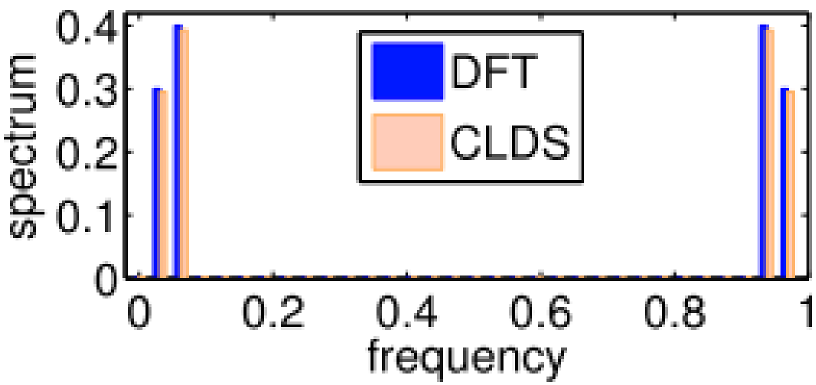
Theorem: For single signal,

If $\boldsymbol{A} = \mathrm{diag}(\exp(\frac{2\pi i}{N} k)), k = 1, \ldots, N$

C will be Fourier spectrum



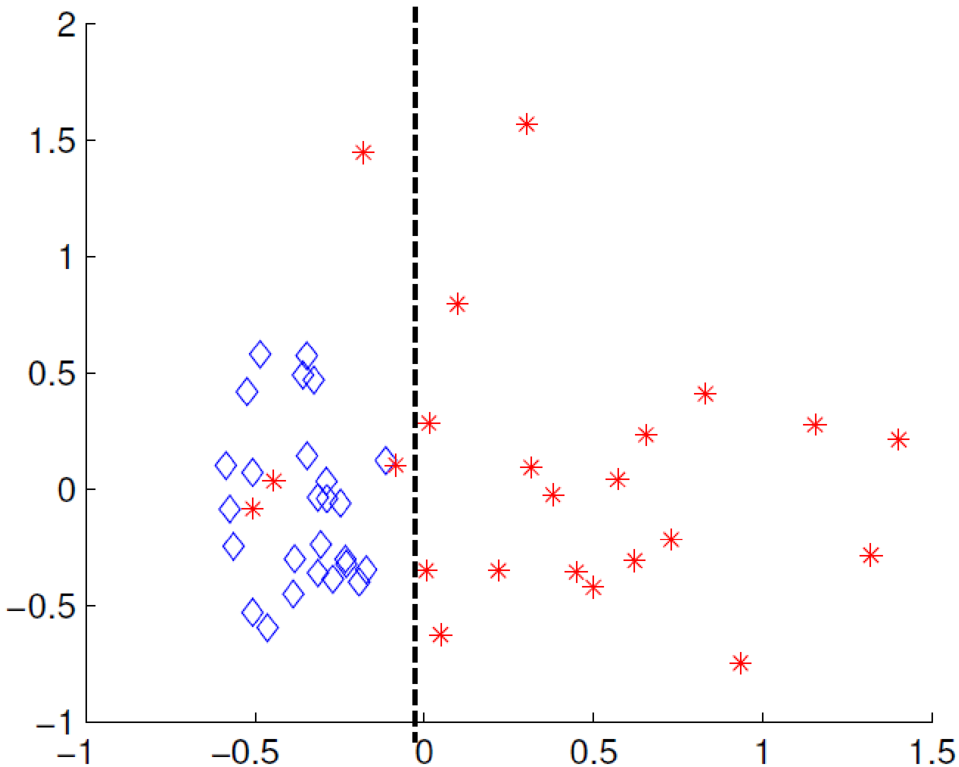
(a) $x = \sin(\frac{2\pi t}{32})$



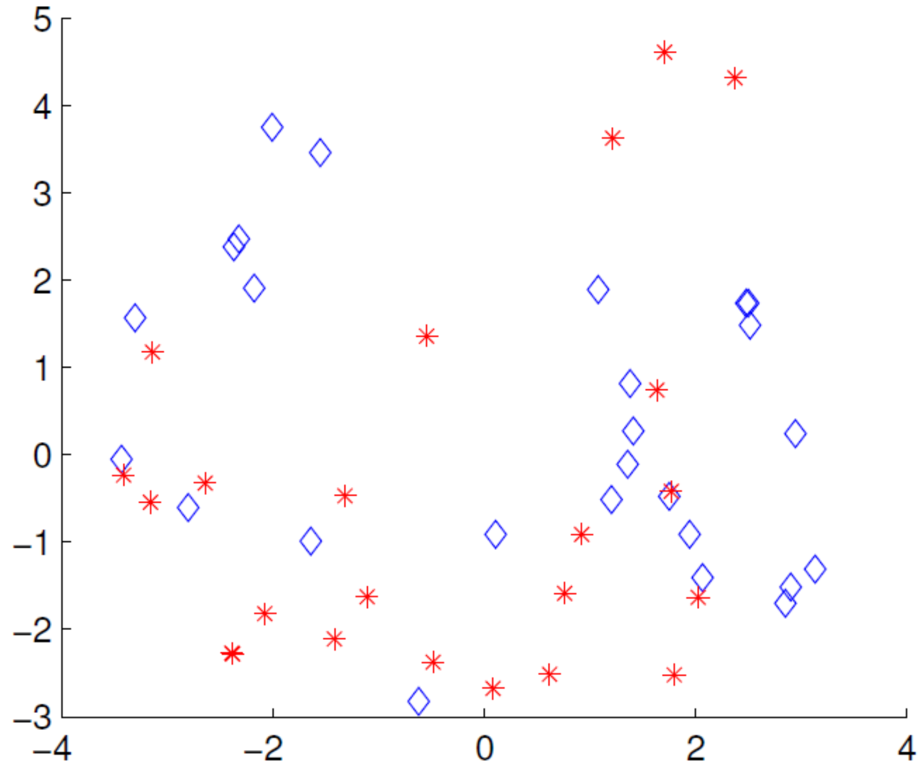
(b) $x = 0.6 \sin(\frac{2\pi t}{32}) + 0.8 \sin(\frac{2\pi t}{16})$

CLDS Clustering Mocap Data

CLDS two features



PCA top 2 components

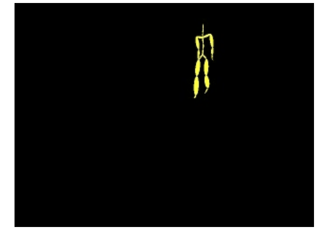


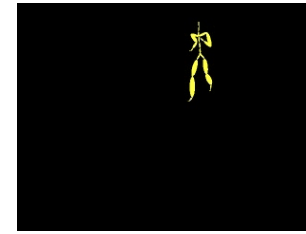
Accuracy = 93.9%

Accuracy = 51.0%

◊ walking motion   ∗ running motion

Results





Conditional Entropy (lower is better)

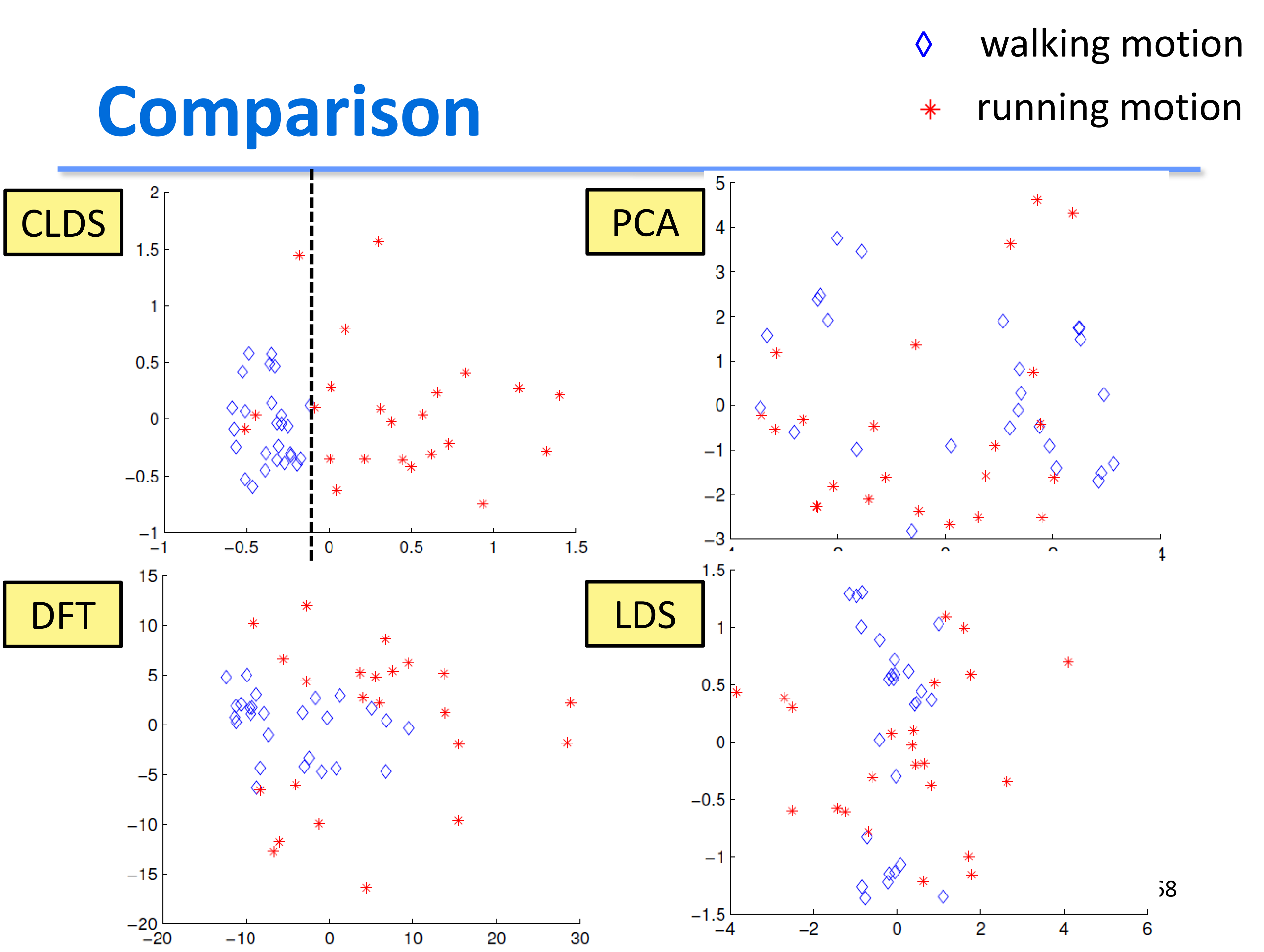
methods	MOCAPPOS S	MOCAPANG S	
CLDS	**0.3786**	**0.1015**	
PCA	0.6818	0.3635	[Bishop 2006]
DFT	0.6143	0.2538	
DTW	0.5707	0.4229	[Gunopulos 2001]
KF	0.6749	0.5239	[Buzan 2004]

– MOCAPPOS (49 motion sequences of marker positions)

– MOCAPANG (33 sequences of joint angles)

- Metric: conditional entropy of the confusion matrix M $S(M) = \sum_{i,j} \frac{M_{i,j}}{\sum_{k,l} M_{k,l}} \log \frac{\sum_k M_{i,k}}{M_{i,j}}$

Comparison



Clustering Network Traffic Streams

BGP data: hierarchical clustering

Outline

- Motivation
- Mining w/ Missing Values [Li+ 09, Li+10a]
- Feature Learning for Time Series [Li+10b, Li+11a]
- Summary of the remaining chapters
- Conclusion and Future Directions

Summary of My Work on Time Series

Pattern discovery

- DynaMMo [Li 09]
- BoLeRO [Li 10a]
- ThermoCast [Li 11b]
- LazinessScore [Li08a]

Motion capture
Security
Environmental

Feature extraction

- PLiF [Li 10b]
- CLDS [Li 11a]

Motion capture
Network traffic

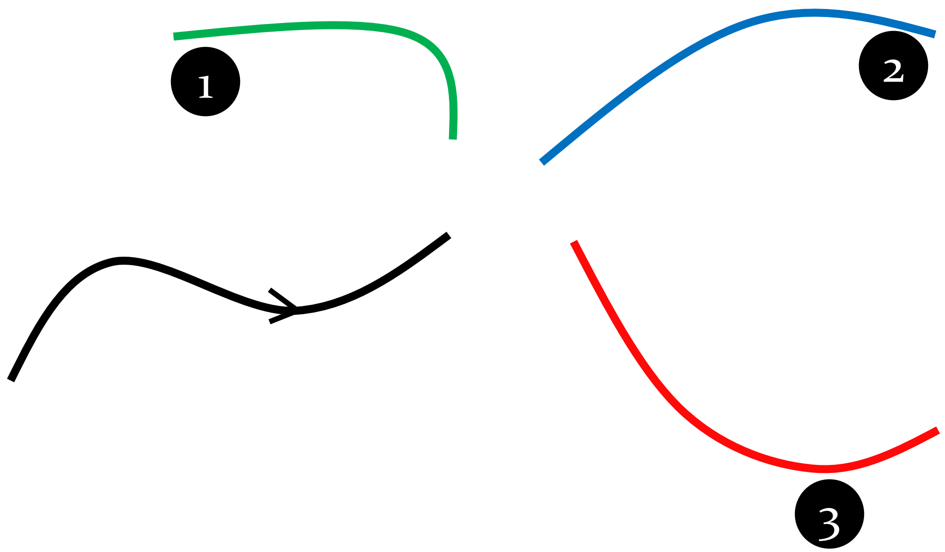
Parallel algorithm

- Cut-And-Stitch [Li 08b]
- WindMine [Sakurai 11]

Datacenter monitoring
web click data

Natural Motion Stitching

- Given two motion-capture sequences that are to be stitched together, how can we assess the goodness of the stitching? [Li et al, Eurographics 08]
- Euclidean will fail



Best stitchable motion?

Intuition and Example

straight moving

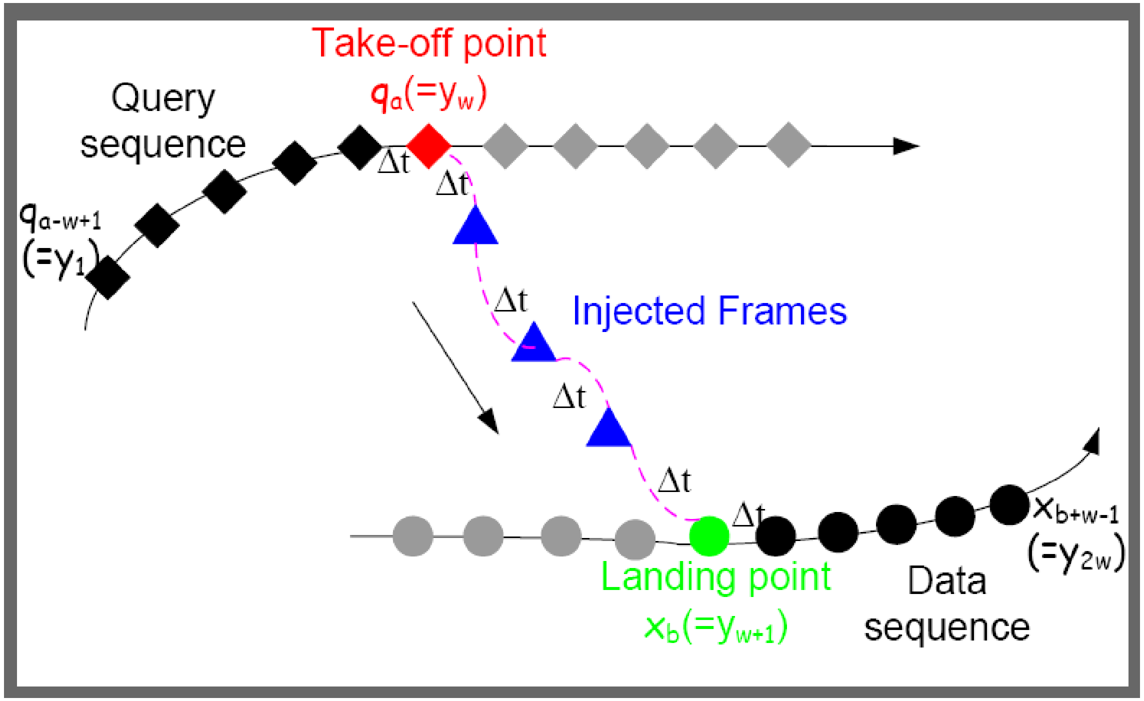
U-Turn

Laziness-score prefer straightforward moving

more results in [Li 2008a]

Laziness Score [Li et al, EG 2008]

- Conjecture: *less human effort → more natural*
- Proposed: use Kalman filters to estimate position, velocity, acceleration → Compute effort/ energy



Summary of My Work on Time Series

Pattern discovery

- ✓ DynaMMo [Li 09]
- ✓ BoLeRO [Li 10a]
- ThermoCast [Li 11b]
- ✓ LazinessScore [Li08a]

Motion capture
Security
Environmental

Feature extraction

- ✓ PLiF [Li 10b]
- ✓ CLDS [Li 11a]

Motion capture
Network traffic

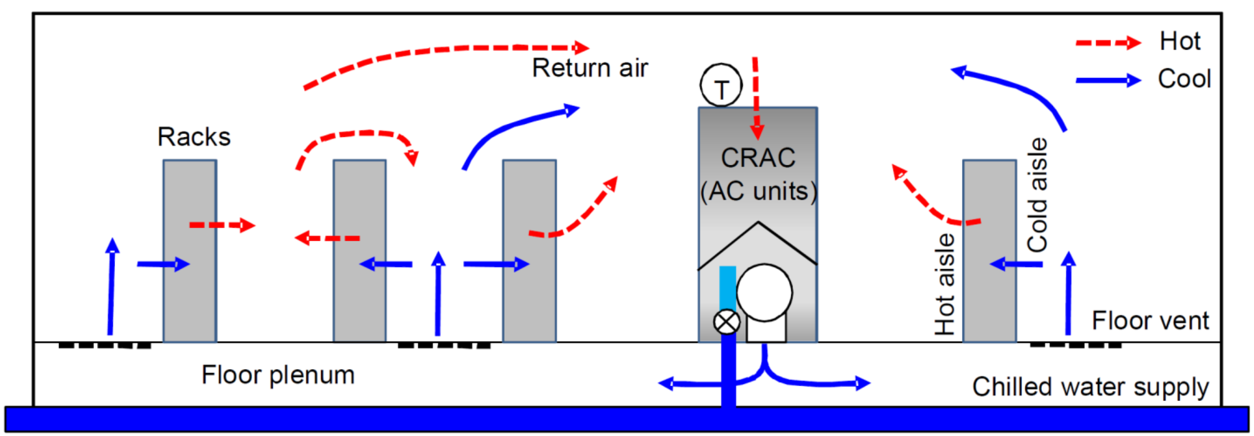
Parallel algorithm

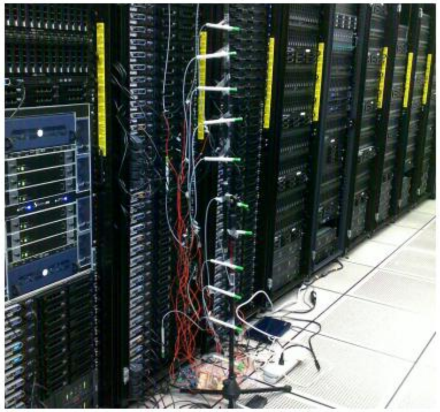
- Cut-And-Stitch [Li 08b]
- WindMine [Sakurai 11]

Datacenter monitoring
web click data

Towards Thermal Aware DC Management

- Data centers are often over provisioned, with ≈40% of energy spent for cooling (total=$7.4B)
- How can we improve energy efficiency in modern multi-MegaWatt data centers?

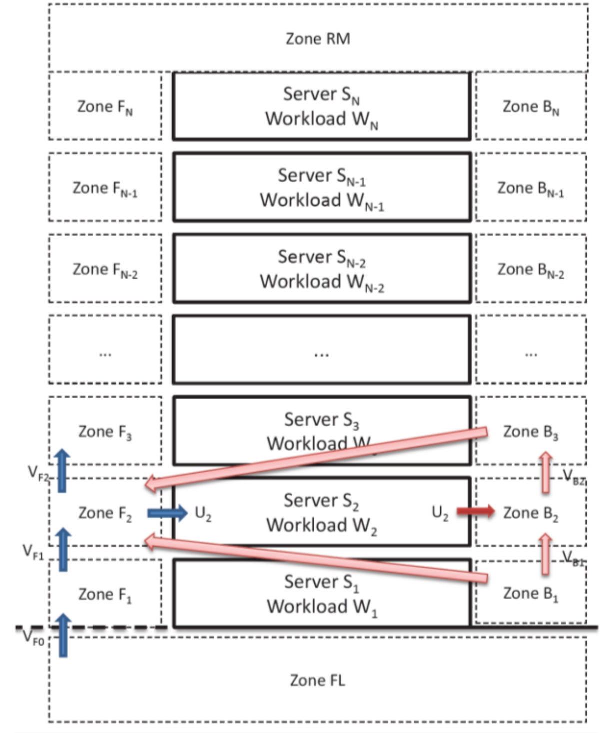


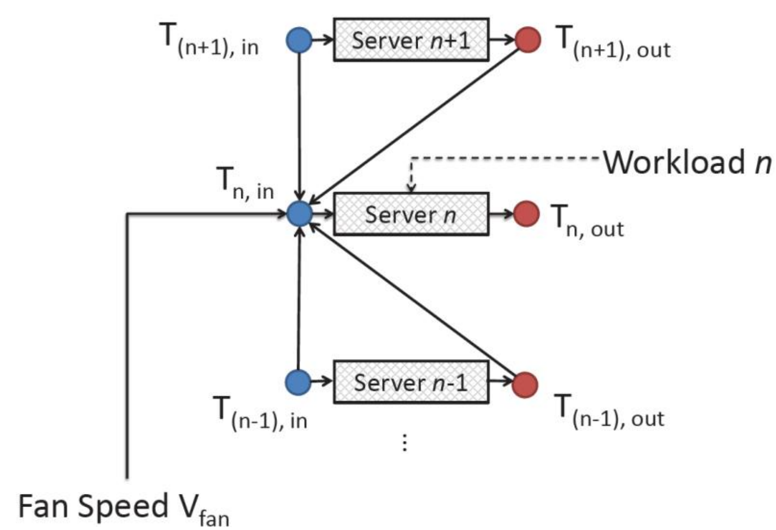


JHU data center with Genomote

ThermoCast [Li *et al*, KDD 2011]

- Given: intake temperatures, outtake temperatures, workload for each server , and floor air speed
- Goal: forecasting temperature distribution and thermal aware placement of workload

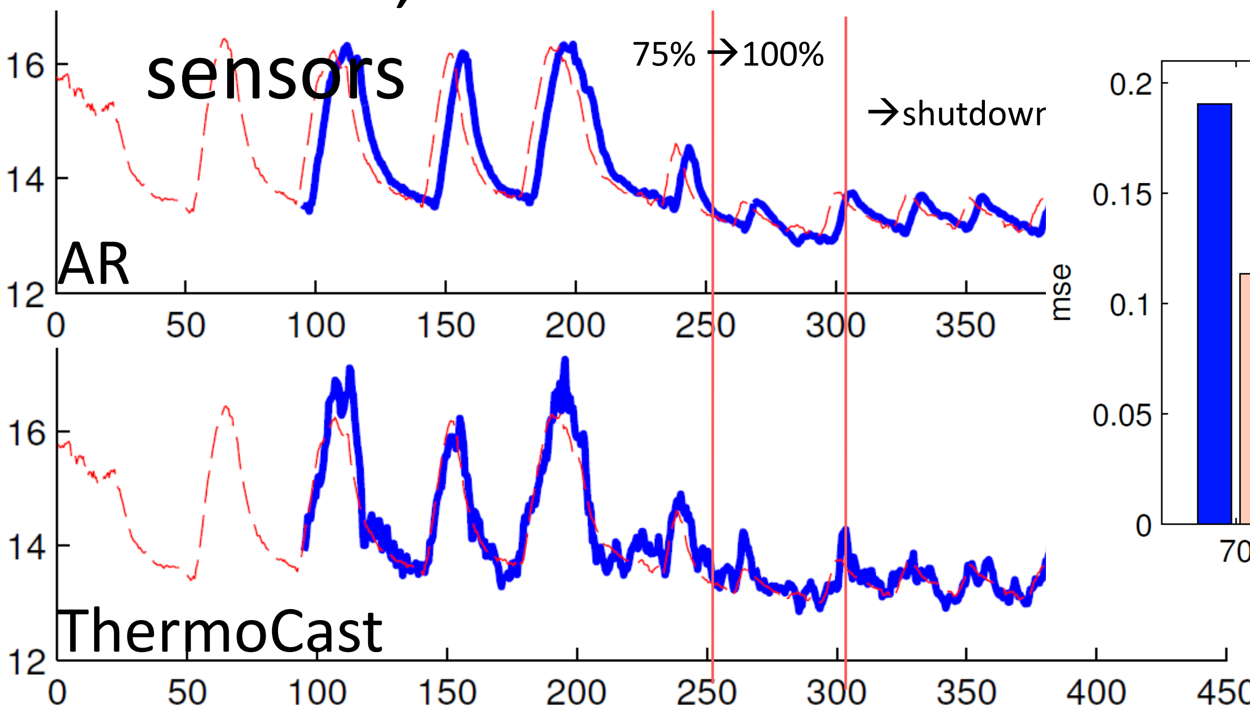


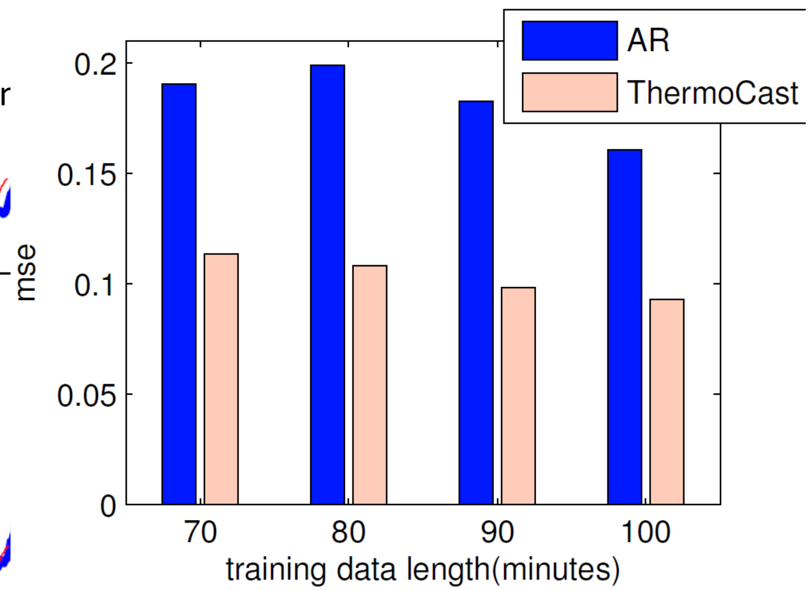


ThermoCast Results

- Q1: How accurately can a server learn its local thermal dynamics for prediction? 2x better

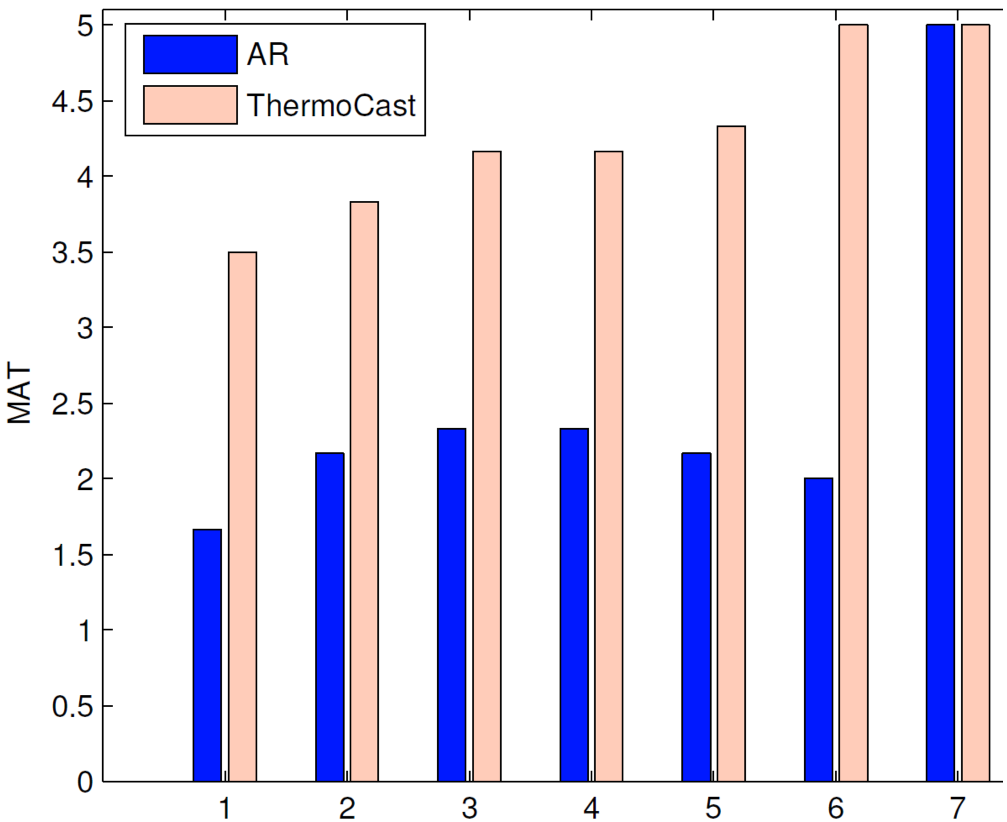
Tested in JHU data center with 171 1U servers, instrumented with a network of 80 sensors





ThermoCast Results

- Q2: How long ahead can ThermoCast forecast thermal alarms? 2x faster



	Baseline	ThermoCast
Recall	62.8%	71.4%
FAR	45%	43.1%
MAT	2.3min	**4.2 min**

FAR=false alarm rate
MAT=mean look-ahead time

Contributions and Impact

- Predictability: a hybrid approach to integrate the thermodynamics and sensor data
- Scalable learning/training thanks to the zonal thermal model
- Real data and instrument in a data center with practical workload
- Projected impact: can handle extra 26% workload (e.g. PUE 1.5 → PUE 1.4)

Summary of My Work on Time Series

Pattern discovery

- ✓ DynaMMo [Li 09]
- ✓ BoLeRO [Li 10a]
- ✓ ThermoCast [Li 11b]
- ✓ LazinessScore [Li08a]

Motion capture
Security
Environmental

Feature extraction

- ✓ PLiF [Li 10b]
- ✓ CLDS [Li 11a]

Motion capture
Network traffic

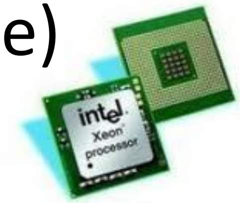
Parallel algorithm

- Cut-And-Stitch [Li 08b]
- WindMine [Sakurai 11]

Datacenter monitoring
web click data

Parallel learning for LDS

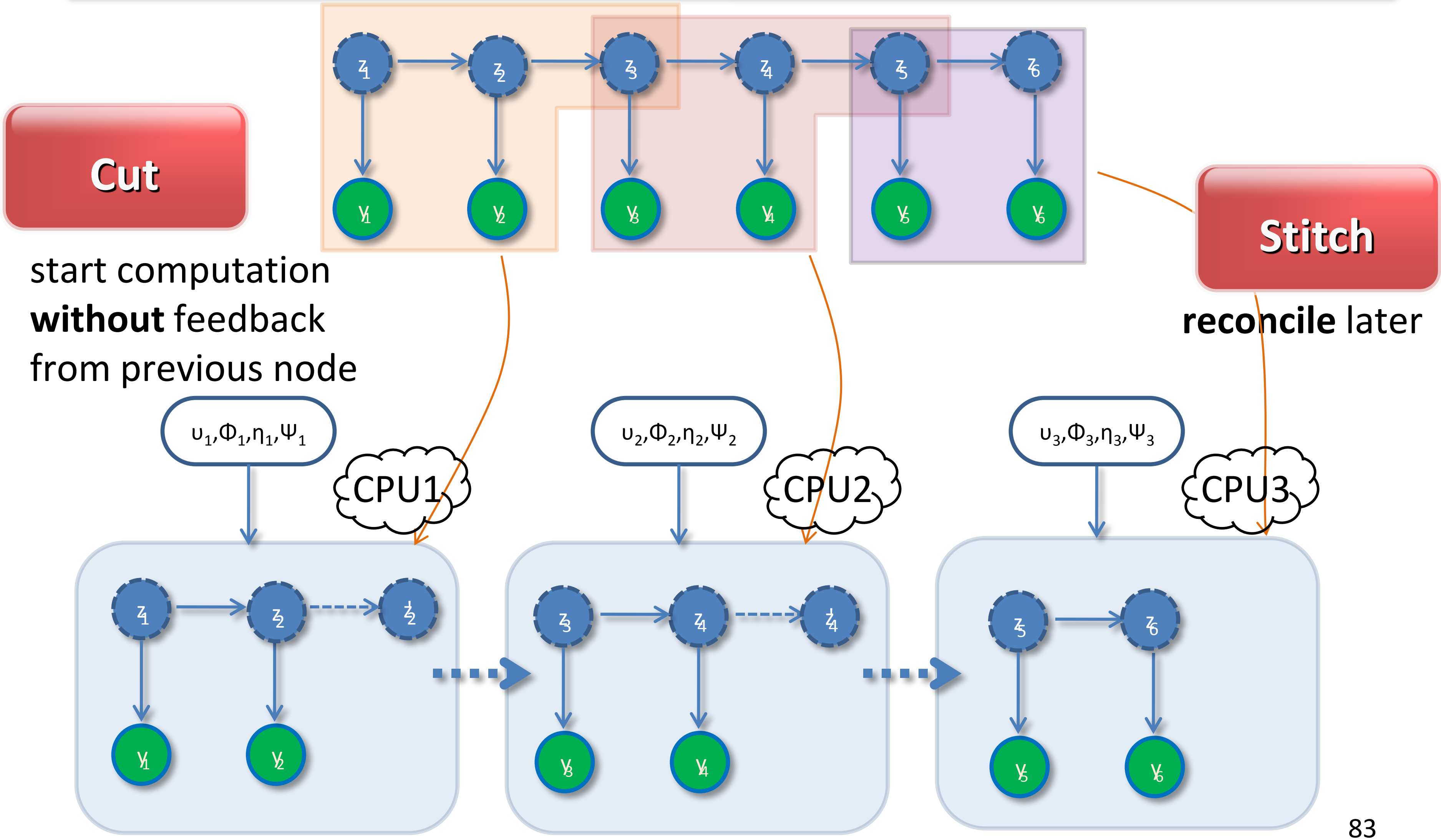
- Problem:
  - Learning LDS on multicore (SMP)
- Goal: ~ linear speed up
- Assumption:
  - *Shared memory architecture* (e.g. multi-core)



- Test environment
  - NCSA SGI Altix, 512 1.6GHz Itanium2 processors, 3TB of total memory (ccNUMA)
  - PSC SGI Altix, with 768 cores, 1.5 TB total memory

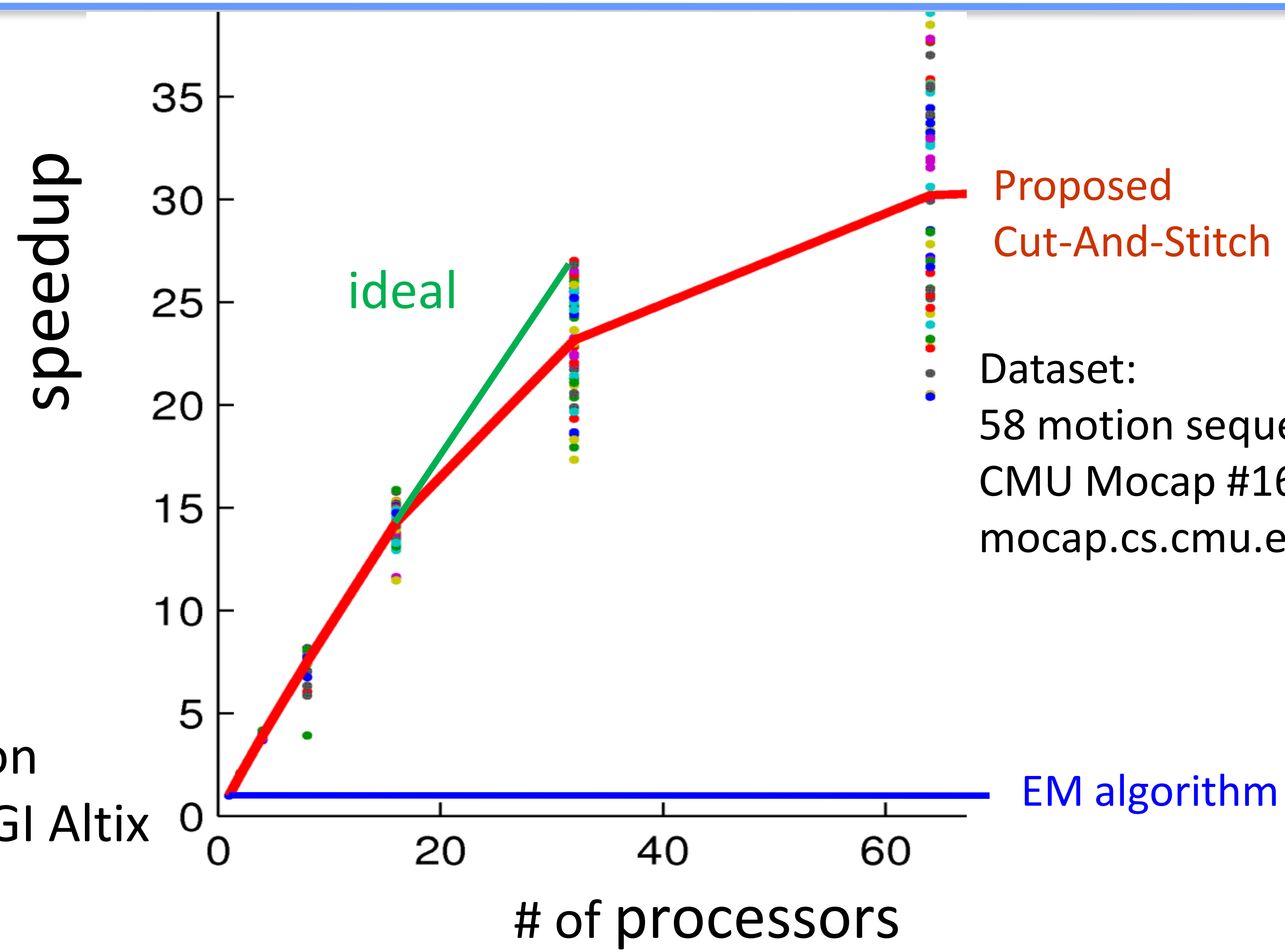
[Li et al, KDD 2008]

Cut-And-Stitch: Intuition



Implemented using OpenMP, details in [Li+ 2008b]

Cut-And-Stitch: Near Linear Speedup



Dataset:
58 motion sequences
CMU Mocap #16
mocap.cs.cmu.edu,

tested on
NCSA SGI Altix

more results in [Li+2008b]

Summary of My Work on Time Series

Pattern discovery

- ✔ DynaMMo [Li 09]
- ✔ BoLeRO [Li 10a]
- ✔ ThermoCast [Li 11b]
- ✔ LazinessScore [Li08a]

Motion capture
Security
Environmental

Feature extraction

- ✔ PLiF [Li 10b]
- ✔ CLDS [Li 11a]

Motion capture
Network traffic

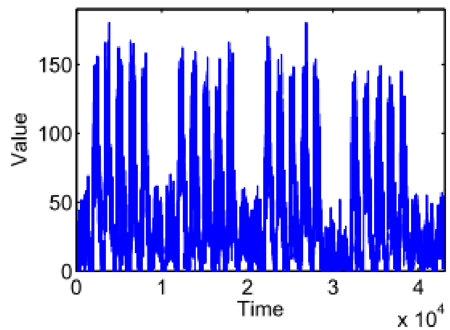
Parallel algorithm

- ✔ Cut-And-Stitch [Li 08b]
- WindMine [Sakurai 11]

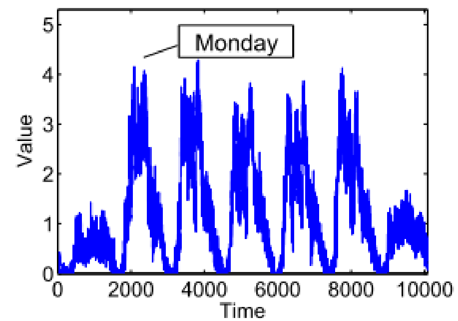
Datacenter
monitoring
web click data

WindMine

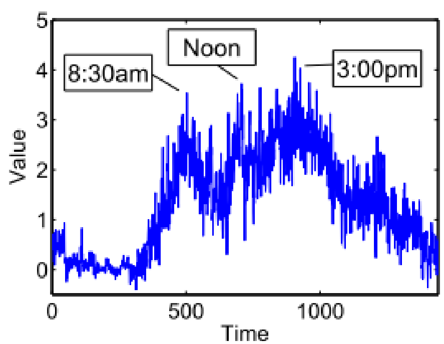
- Goal: find patterns and anomalies from user-click streams



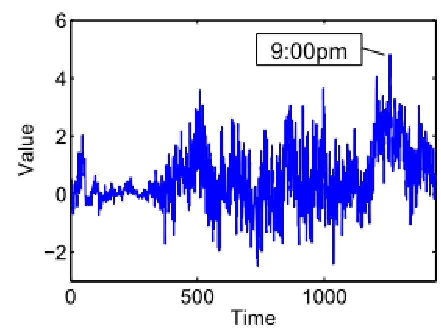
Web-click sequence



Weekly component



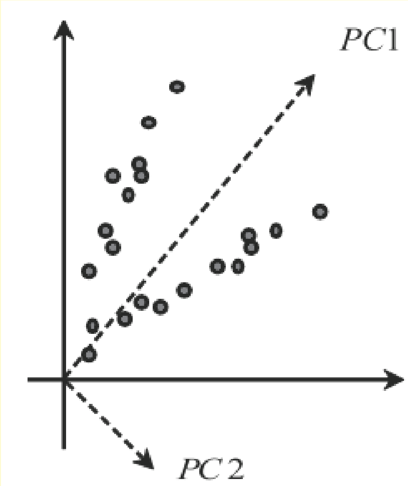
Weekday component



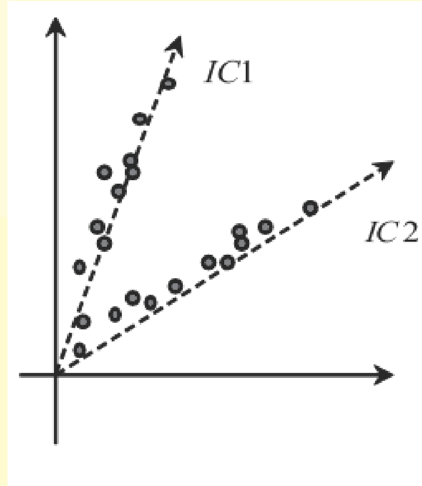
Weekend component

WindMine

- Key technique:
  - Automatic windowing + ICA + parallel/distributed



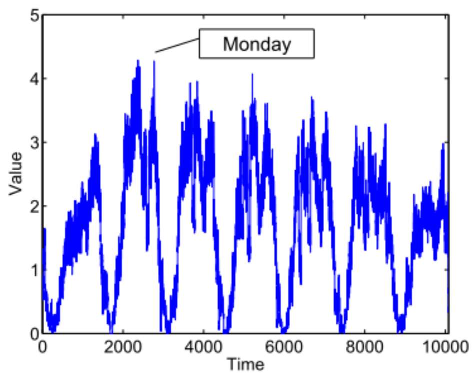
PCA



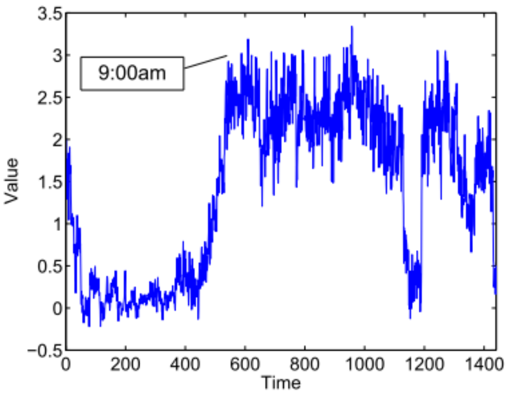
ICA

Discoveries by WindMine

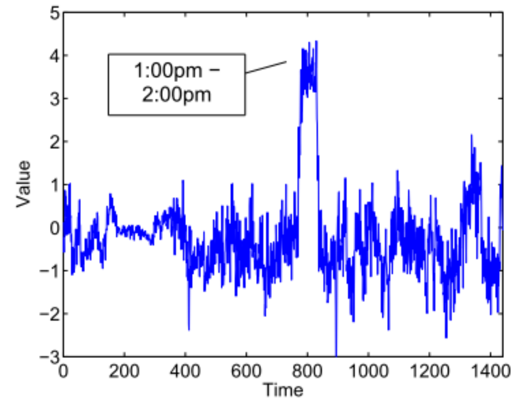
Job website



(a) Weekly pattern

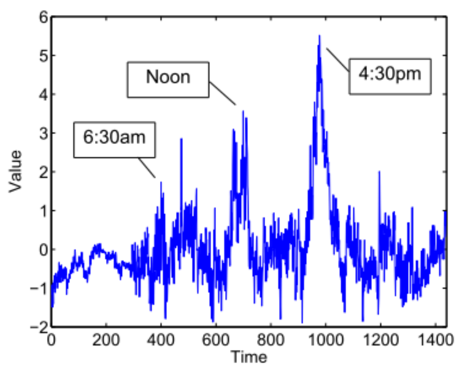


(b) Daily pattern



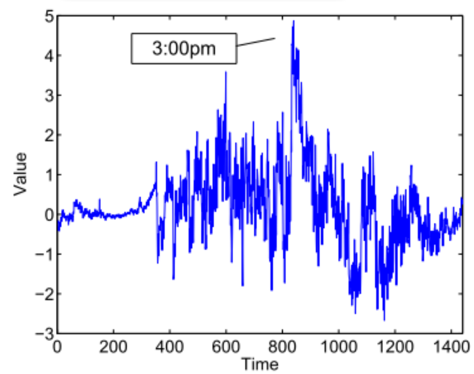
(c) Weekday additional pattern

weather



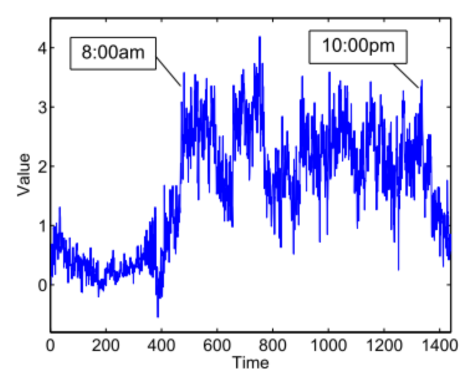
(d) Weather news

kids



(b) Kids

health



(e) Health

Summary of My Work on Time Series

Pattern discovery

- DynaMMo [Li 09]
- BoLeRO [Li 10a]
- ThermoCast [Li 11b]
- LazinessScore [Li08a]

Motion capture
Security
Environmental

Feature extraction

- PLiF [Li 10b]
- CLDS [Li 11a]

Motion capture
Network traffic

Parallel algorithm

- Cut-And-Stitch [Li 08b]
- WindMine [Sakurai 11]

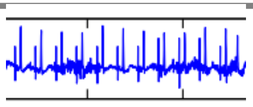
Datacenter monitoring
web click data

Outline

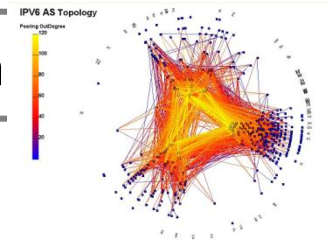
- Motivation
- Mining w/ Missing Values [Li+ 09, Li+10a]
- Feature Learning for Time Series [Li+10b]
- Other relevant work
- Conclusion and Future Directions

Why Mining Time Series?

- Motion Capture (game $57 billion,'09 & in movie)

- Data center monitoring and control ($7.4B power )
- Health informatics (e.g. physiological signals) 
- Environmental monitoring (e.g. drinking water)

- Computer network security & anomaly detection

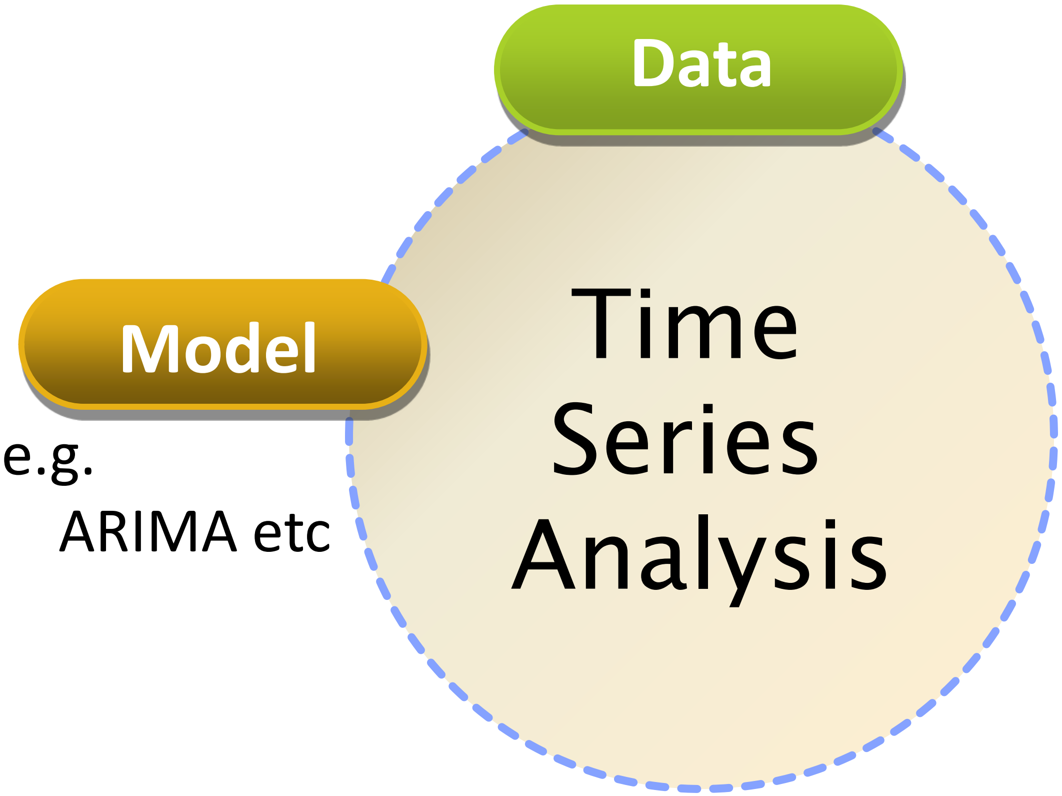


……

Mining problems in the thesis

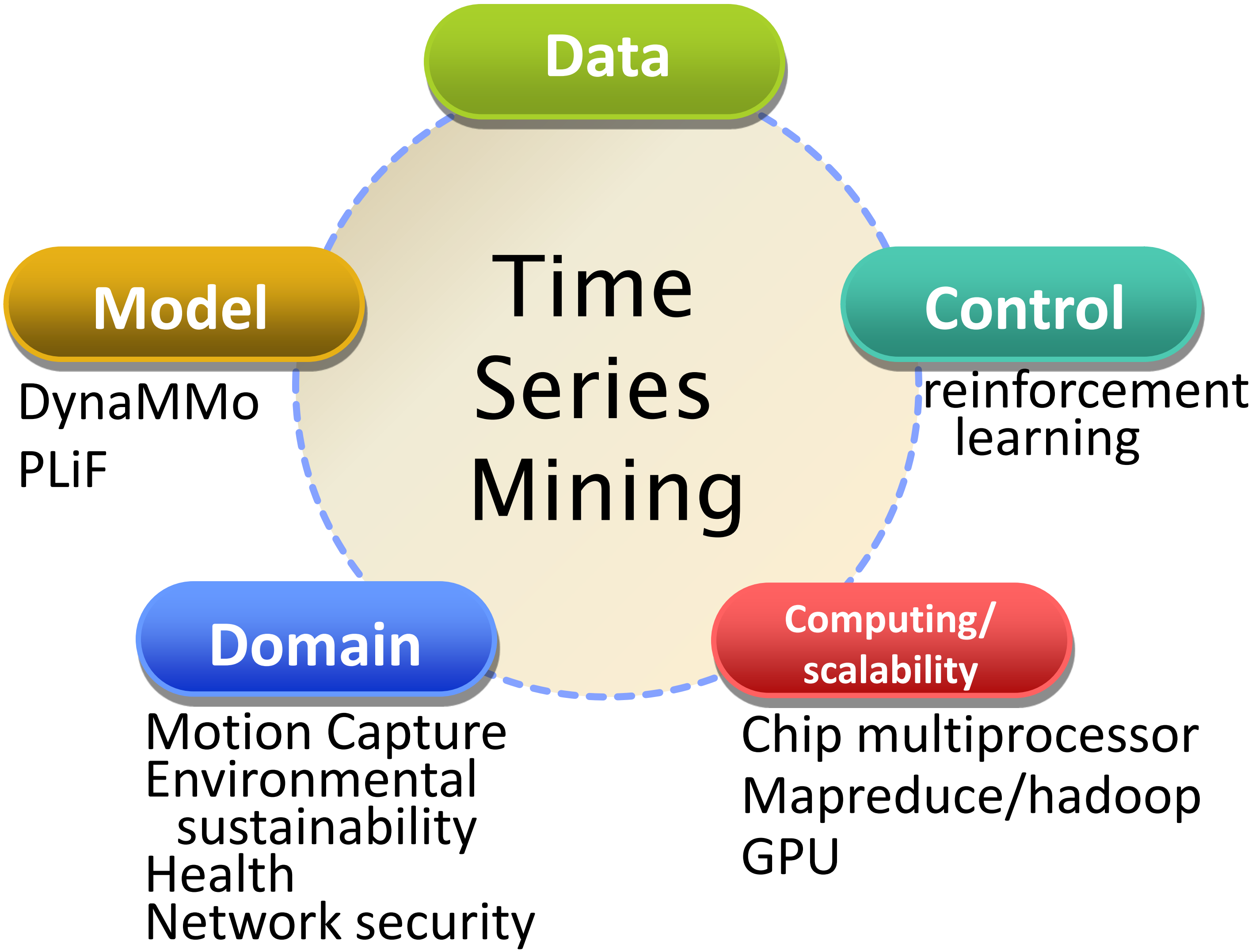
1. Forecasting and imputation (chap 3)
2. Summarization and anomaly (chap 3, 4)
3. Feature, clustering and similarity (chap 4, 5)
4. Parallel and scalability (chap 6, 7, 8)
5. Applications (chap 8, 9, 10, 11)

Traditional View



e.g.
ARIMA etc

What's next?


Data
Time Series Mining
Model
DynaMMo
PLiF
Control
reinforcement learning
Domain
Motion Capture
Environmental sustainability
Health
Network security
Computing/ scalability
Chip multiprocessor
Mapreduce/hadoop
GPU

Thesis overview

Pattern discovery

- DynaMMo [Li 09]
- BoLeRO [Li 10a]
- ThermoCast [Li 11b]
- LazinessScore [Li08a]

Feature extraction

- PLiF [Li 10b]
- CLDS [Li 11a]

Parallel algorithm

- Cut-And-Stitch [Li 08b]
- WindMine [Sakurai 11]

Contributions:

1. Most accurate missing value recovery/summarization
2. Most effective clustering on TS
3. Fast algorithms: linear to length
4. Parallel algorithms: linear speed up on multicore