IBM **Research**

Recent advances in sensor data analytics

Tsuyoshi Ide ("Ide-san")
PhD, Senior Technical Staff Member
IBM Thomas J. Watson Research Center

CME 2018 (Shimane, Japan, Sep 7, 2018)

Agenda

- General challenges in industrial sensor data analytics

- Solution examples:
  - Change detection using directional statistics
  - Multi-task multi-modal models for collective anomaly detection
  - Tensorial change analysis

- Discussion: deep learning, Blockchain, and future directions

IBM T. J. Watson Research Center
Center for Computational and Statistical Learning

Accelerated breeding of energy crops

Lead conversion prediction

Fire frontline prediction

Oil field Planning & Control

Joint modeling/clustering for GWAS

Enterprise revenue forecasting

Yield & production modeling

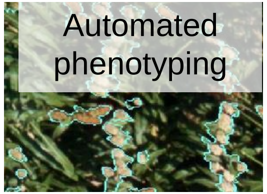
Transfer learning for anomaly detection

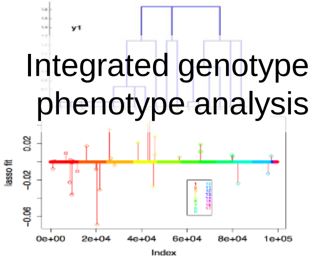
Panicle detection and counting

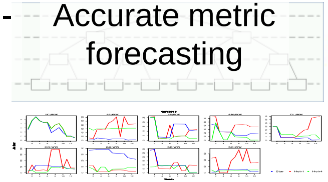
Insurance cost prediction & attribution

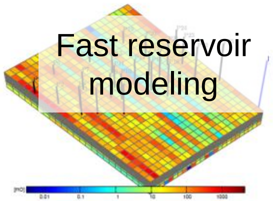
Deep Learning model for reservoir simulator

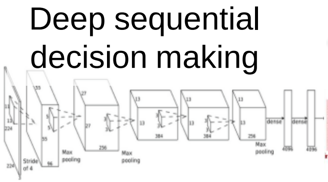
On-going Agenda

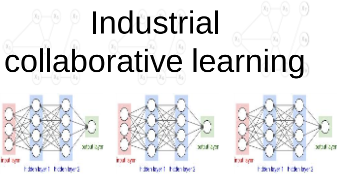

Automated phenotyping


Integrated genotype-phenotype analysis


Accurate metric forecasting


Fast reservoir modeling


Deep sequential decision making


Industrial collaborative learning

Capabilities

Satellite, drone image data analytics

Structural and Adaptive Learning

Structured time series analysis

Integration of ML/statistics with physics

Deep Reinforcement Learning

Distributed learning and Blockchain

Pillar Competencies

Machine learning from sensor data is one of the major research focuses

- Anomaly and change detection is a major topic in sensor data analytics

- Recently published two textbooks (in Japanese)

異常検知と
変化検知

Anomaly Detection
and
Change Detection

井手 剛
杉山 将

Machine Learning Professional Series

MLP 機械学習プロフェッショナルシリーズ

講談社

入門
機械学習による
異常検知
Rによる実践ガイド

Introduction to Anomaly Detection using Machine Learning

井手 剛 著

コロナ社

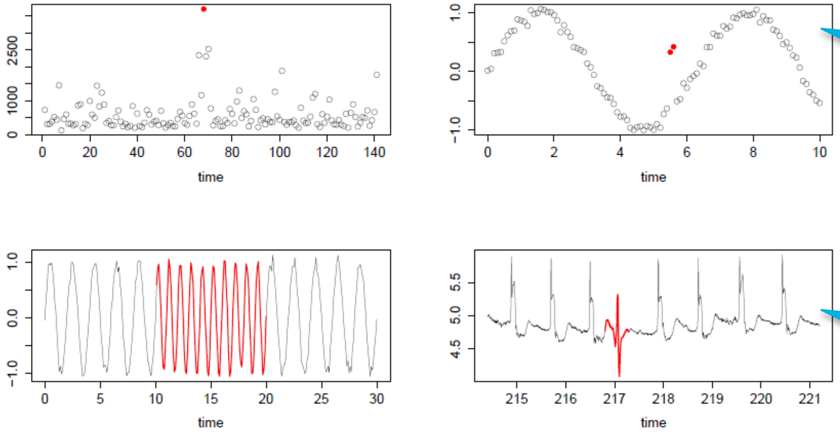
Basics: General problem setting in machine learning

- Supervised learning
  - Given a data set $\{(\boldsymbol{x}^{(1)}, y^{(1)}), \dots, (\boldsymbol{x}^{(N)}, y^{(N)})\}$
  - find the probability distribution of y given **x**: $p(y \mid \boldsymbol{x})$
- Unsupervised learning
  - Given $\{\boldsymbol{x}^{(1)} \dots, \boldsymbol{x}^{(N)}\}$
  - find $p(\boldsymbol{x})$
- Typical assumptions
  - **x** is a vector
  - y is a scalar
  - Samples are independently and identically distributed (i.i.d.)
- What makes sensor data analytics interesting?

General challenges: No “one-size-fits-all” algorithm

- Example in anomaly detection
  - “*Happy families are all alike; every unhappy family is unhappy in its own way*.” - *Anna Karenina*, Leo Tolstoy

Examples of anomalies



outliers (from i.i.d. samples)

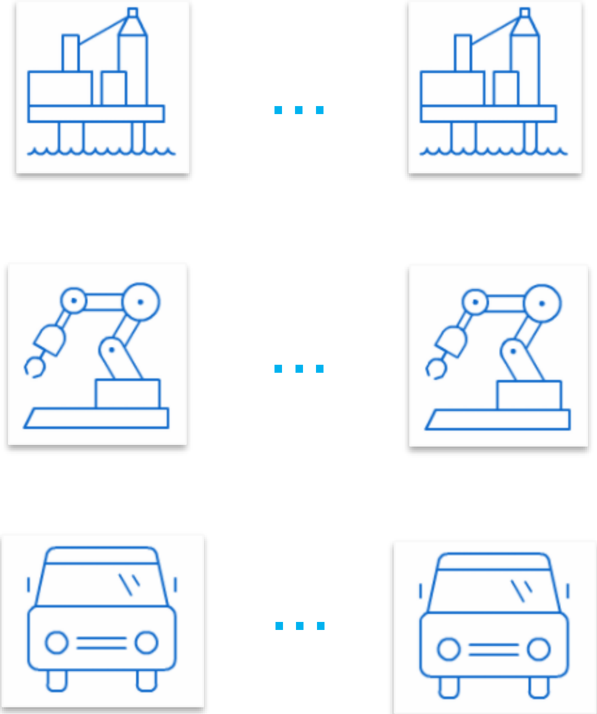
outliers (from auto-correlated samples)

change points

discords

Tsuyoshi Ide and Masashi Sugiyama, Anomaly Detection and Change Detection, Kodansha, 2015 (in Japanese).

General challenges: Business requirements often drive extensions of existing approaches

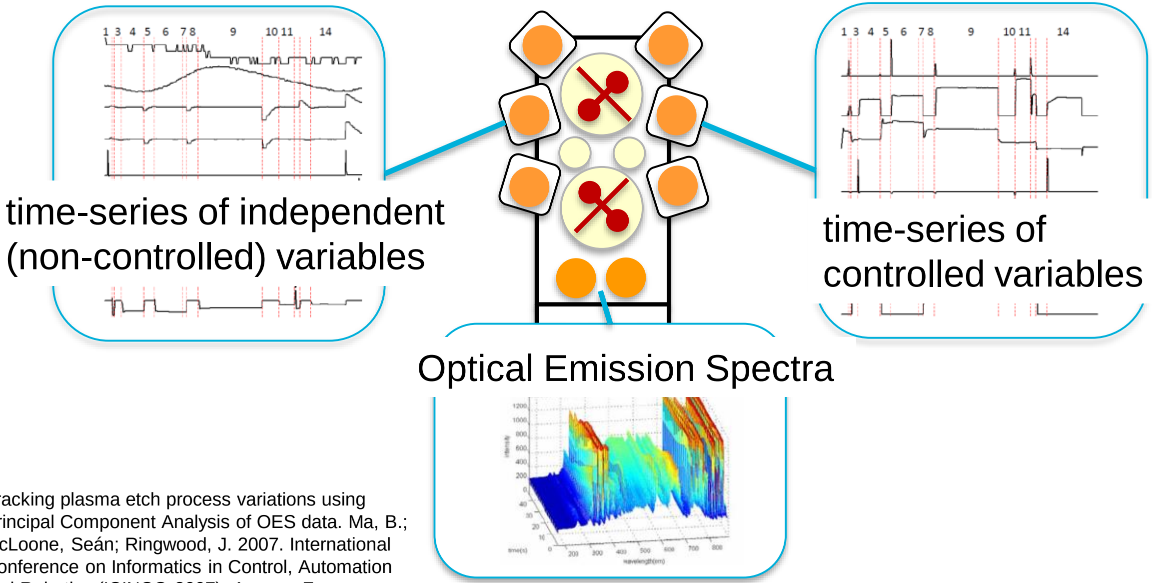


- Example: corporate-level asset management with anomaly detection
  - ○ Typically assets are managed as a cohort
    - ✓ 10s of off-shore oil production systems
    - ✓ 100s of industrial robots
    - ✓ 1000s of electric vehicles in a certain area
  - ○ How can we leverage the commonality between assets to build an anomaly detection solution for individual assets?

T. Ide, et al., “Multi-task Multi-modal Models for Collective Anomaly Detection,“ Proc. 2017 IEEE Intl. Conf. Data Mining (ICDM 17), pp.177-186

General challenges: Complex internal structure may exist in one measurement

Example from semiconductor manufacturing (etching)



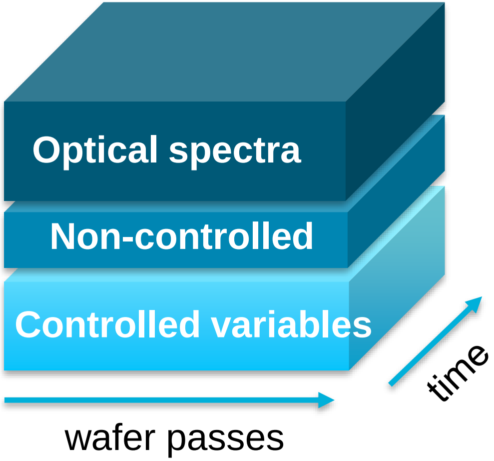
time-series of independent (non-controlled) variables

time-series of controlled variables

Optical Emission Spectra

Tracking plasma etch process variations using Principal Component Analysis of OES data. Ma, B.; McLoone, Seán; Ringwood, J. 2007. International Conference on Informatics in Control, Automation and Robotics (ICINCO 2007), Angers, France.

Each wafer pass is a higher-order tensor, rather than a vector



Optical spectra

Non-controlled

Controlled variables

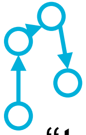
time

wafer passes

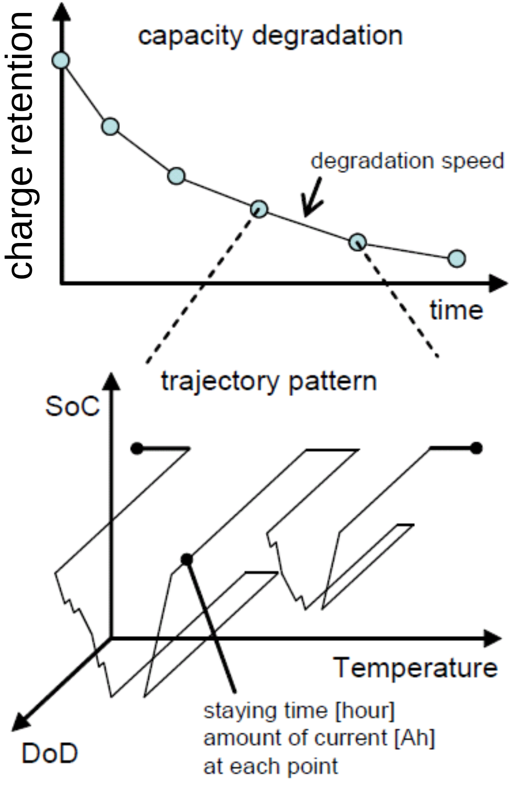
General challenges: Ready-to-use solution to your problem might not even exist

- Example: Charge retention (~ battery life) prediction of electric vehicle batteries
    - Depends on the entire history of battery usage
    - Battery usage is represented as a complex trajectory of a multi-dimensional space

- Charge retention prediction task should be formulated as “trajectory regression”

charge retention $y = f($  $)$

“trajectory”

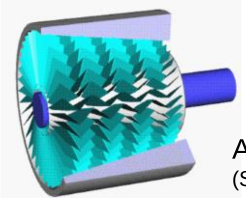


Toshiro Takahashi, Tsuyoshi Ide, “Predicting Battery Life from Usage Trajectory Patterns,” Proc. Intl. Conf. Pattern Recognition (ICPR 2012), pp.2946-2949.

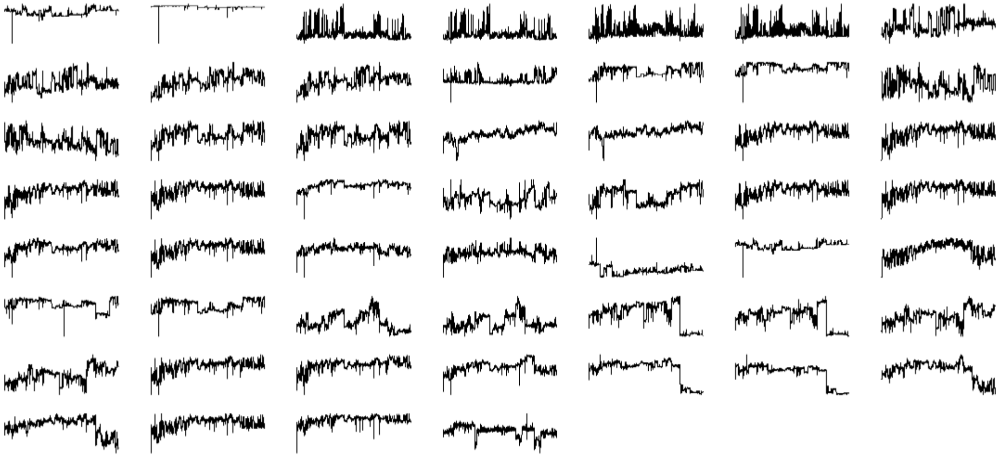
General challenges: Ground truth may not be available. Some degrees of freedom are usually latent

- Example: sensor data of a compressor of oil production system
    - Data taken under a normal operational condition
    - Noisy, nonstationary, heterogeneous, high-dimensional …
- Hard to pinpoint what is indicative of malfunction

(Image: Wikimedia commons)



Axial compressor
(Source: Wikipedia)



Agenda

- General challenges in industrial sensor data analytics
- Solution examples:
  - Change detection using directional statistics (Ide et al., IJCAI 17)
  - Multi-task multi-modal models for collective anomaly detection
  - Tensorial change analysis
- Discussion: deep learning, Blockchain, and future directions

Continuous operation of conveyor systems is critical in the mining industry

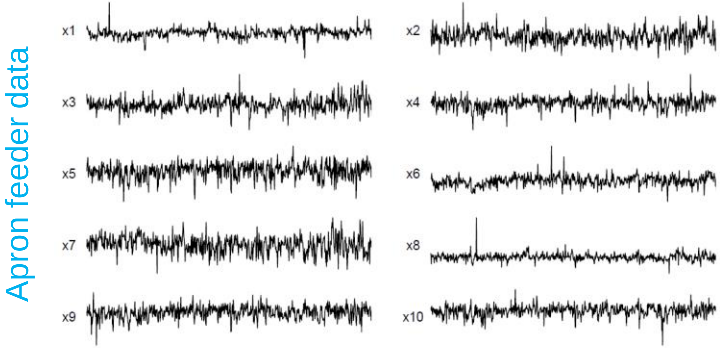
- **Business goal**: Ensure continuous operation of conveyor system (“apron feeder”) by detecting early indications of failures

- **Data**: Physical sensor data from conveyors and motors
  - Every several seconds over ~ 1 year
  - Sensors include: Gearbox temperatures, motor power consumptions, apron speed, etc.

- **Challenge**: Conveyor system is subject to significant fluctuation in load. Hard to characterize the normal operation
  - Mined crude ore never come in a uniform size

(Image: Wikimedia commons)

(Image: Wikimedia commons)



(simulation data)

Problem setting: change detection from multivariate noisy time-series data

- Change = difference between

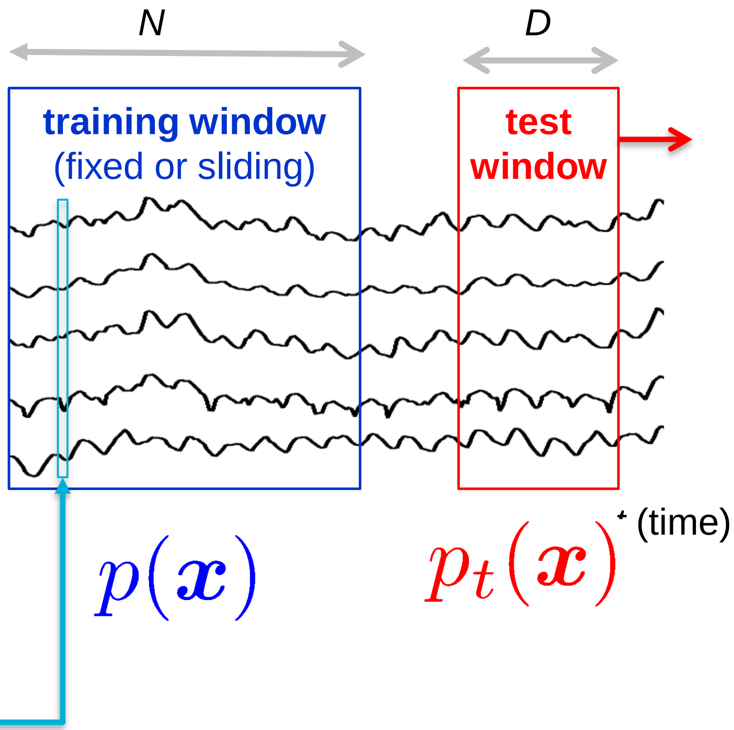
  $p(\boldsymbol{x})$ and $p_t(\boldsymbol{x})$

  - **x**: *M*-dimensional *i.i.d.* observation
  - $p(\mathbf{x})$: p.d.f. estimated from training window
  - $p_t(\mathbf{x})$: p.d.f. estimated from the test window at time *t*

- Assume a sequence of i.i.d. vectors
  - Training data in the training window

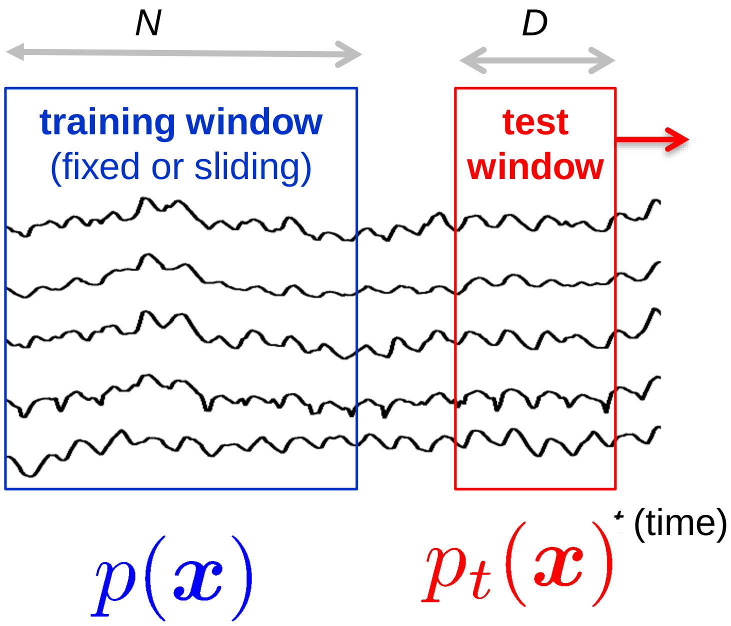
$$\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}, \ldots, \boldsymbol{x}^{(N)}\}$$

time index (or sample index)



Problem setting: change detection from multi-variate noisy time-series data

- Question 1: What kind of model should we use for the probability density?
- Question 2: How can we quantify the difference between the densities?

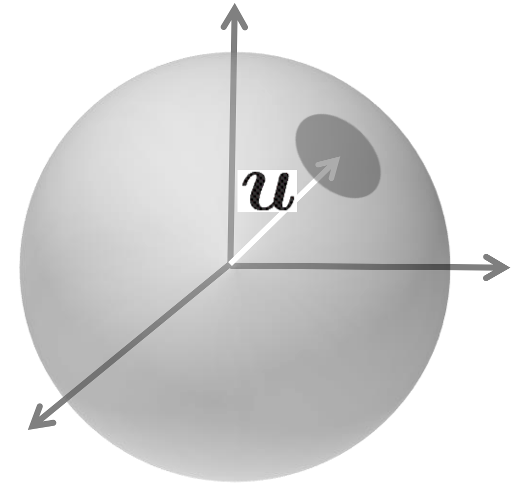


We use von Mises-Fisher distribution to model $p(\boldsymbol{x})$ and $p_t(\boldsymbol{x})$

- vMF distribution: “Gaussian for unit vectors”

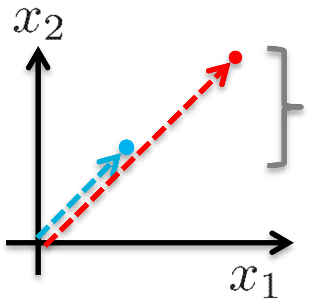
$$p(\boldsymbol{z} \mid \boldsymbol{u}, \kappa) = c_M(\kappa) \exp\left(\kappa \boldsymbol{u}^\top \boldsymbol{z}\right)$$

  - ○ ***z***: random unit vector of ||**z**|| =1
  - ○ ***u***: mean direction
  - ○ κ: “concentration” (~ precision in Gaussian)
  - ○ *M*: dimensionality



- We are concerned only with the direction of observation ***x***:
  - ○ $$\boldsymbol{z} = \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|}$$

- Normalization is always made
- Do not care about the norm

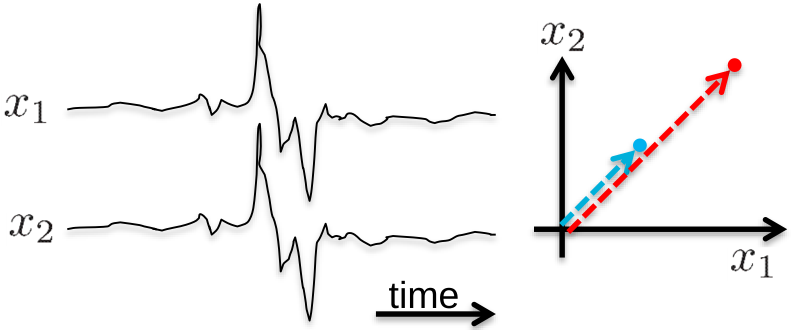


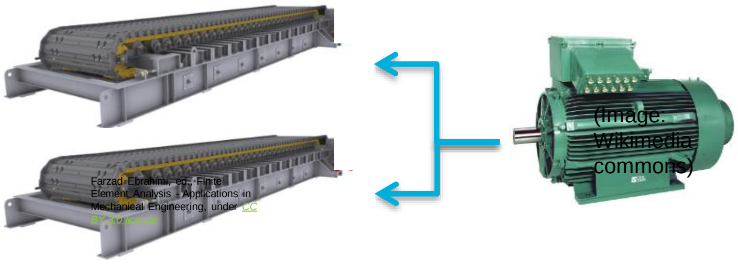
same direction = same input

Normalization is useful to suppress multiplicative noise

- Real mechanical systems often incur multiplicative noise
    - Example: two belt conveyors operated by the same motor

- Normalization of vector is simple but powerful method for noise reduction





Farzad Ebrahimi, ed., Finite Element Analysis - Applications in Mechanical Engineering, under CC BY 3.0 license

(Image: Wikimedia commons)

Mean direction *u* is learned via maximum likelihood. Introduce sample weight to down-weight contaminated ones

- Weighted likelihood function

$\|\boldsymbol{x}^{(n)}\|_2$ (normalization factor)

$$L(\boldsymbol{u}, \kappa) = \sum_{n=1}^{N} w^{(n)} b^{(n)} \{\ln c_M(\kappa) + \kappa \boldsymbol{u}^\top \boldsymbol{z}^{(n)}\}$$

sample weight

- Regularization over sample weights

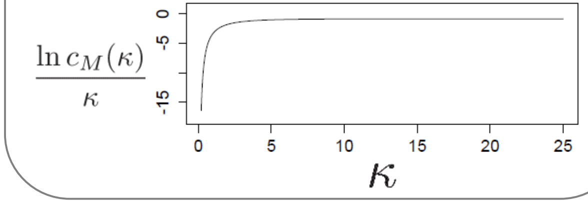
$$R(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \nu\|\boldsymbol{w}\|_1$$

encourage sparsity

- Parameters are learned by solving

$$(\boldsymbol{u}^*, \boldsymbol{w}^*) = \arg\max_{\boldsymbol{u}, \boldsymbol{w}} \{L(\boldsymbol{u}, \kappa) + \lambda R(\boldsymbol{w})\}$$

The term related to κ is less important. κ is treated as a given constant.



Multiple patterns (directions) can be obtained by coupling maximum likelihood equations

- Find orthogonal sequence of the mean direction $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_m$ by coupling the weighted regularized maximum likelihood

$$(\boldsymbol{u}_1^*, \boldsymbol{w}_1^*) = \arg \max_{\boldsymbol{u}_1, \boldsymbol{w}_1} \{L(\boldsymbol{u}_1, \kappa) + \lambda R(\boldsymbol{w}_1)\}$$

$$(\boldsymbol{u}_2^*, \boldsymbol{w}_2^*) = \arg \max_{\boldsymbol{u}_2, \boldsymbol{w}_2} \{L(\boldsymbol{u}_2, \kappa) + \lambda R(\boldsymbol{w}_2)\}$$

$$\vdots$$

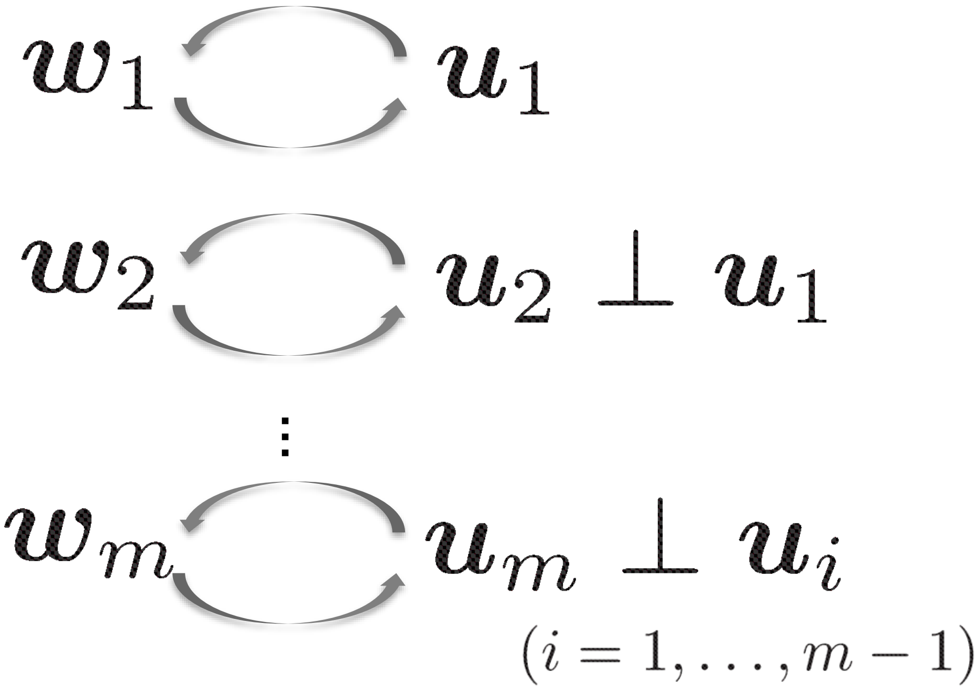
$$(\boldsymbol{u}_m^*, \boldsymbol{w}_m^*) = \arg \max_{\boldsymbol{u}_m, \boldsymbol{w}_m} \{L(\boldsymbol{u}_m, \kappa) + \lambda R(\boldsymbol{w}_m)\}$$

Orthogonality condition

$$\boldsymbol{u}_i^\top \boldsymbol{u}_j = \delta_{i,j}$$

Kronecker delta

Iterative sequential algorithm for the coupled maximum likelihood



$$\boldsymbol{w}_1 \quad \boldsymbol{u}_1$$

$$\boldsymbol{w}_2 \quad \boldsymbol{u}_2 \perp \boldsymbol{u}_1$$

$$\vdots$$

$$\boldsymbol{w}_m \quad \boldsymbol{u}_m \perp \boldsymbol{u}_i$$

$$(i = 1, \ldots, m-1)$$

- For each *i*, $\boldsymbol{w}_i$ and $\boldsymbol{u}_i$ are solved iteratively until convergence
- Analytic solution exists in each step
- Results in very simple fixed point equations

Derived fixed-point iteration algorithm

- Example: *i* =1

Given $\boldsymbol{w}_1$, solve

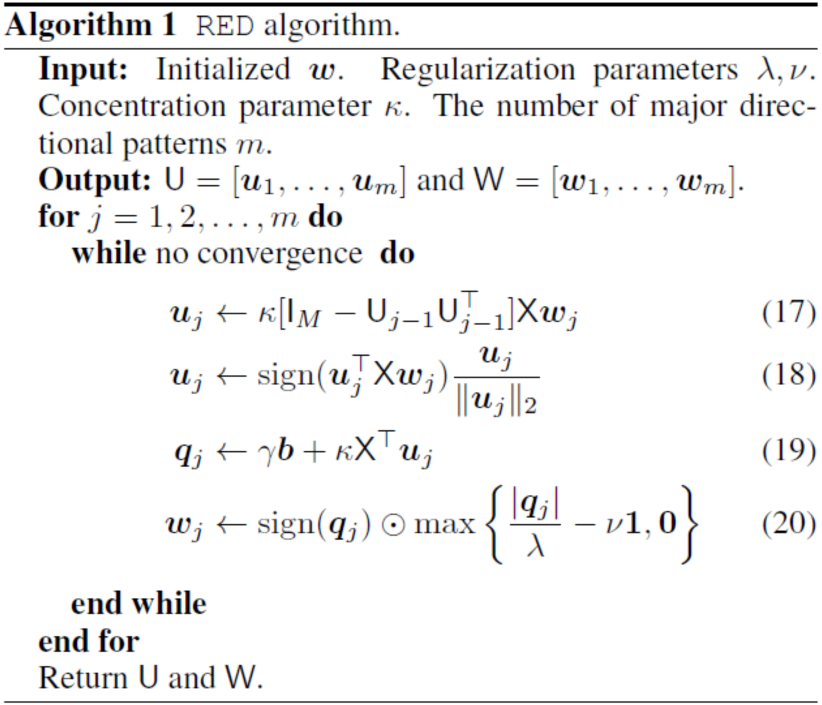
$$\max_{\boldsymbol{u}_1}\{\kappa \boldsymbol{u}_1^\top \mathsf{X} \boldsymbol{w}_1\} \quad \text{s.t.} \quad \boldsymbol{u}_1^\top \boldsymbol{u}_1 = 1$$

Given $\boldsymbol{u}_1$, solve

$$\min_{\boldsymbol{w}_1}\left\{\frac{1}{2}\|\boldsymbol{w}_1 - \frac{\boldsymbol{q}}{\lambda}\|_2^2 + \nu\|\boldsymbol{w}_1\|_1\right\}$$

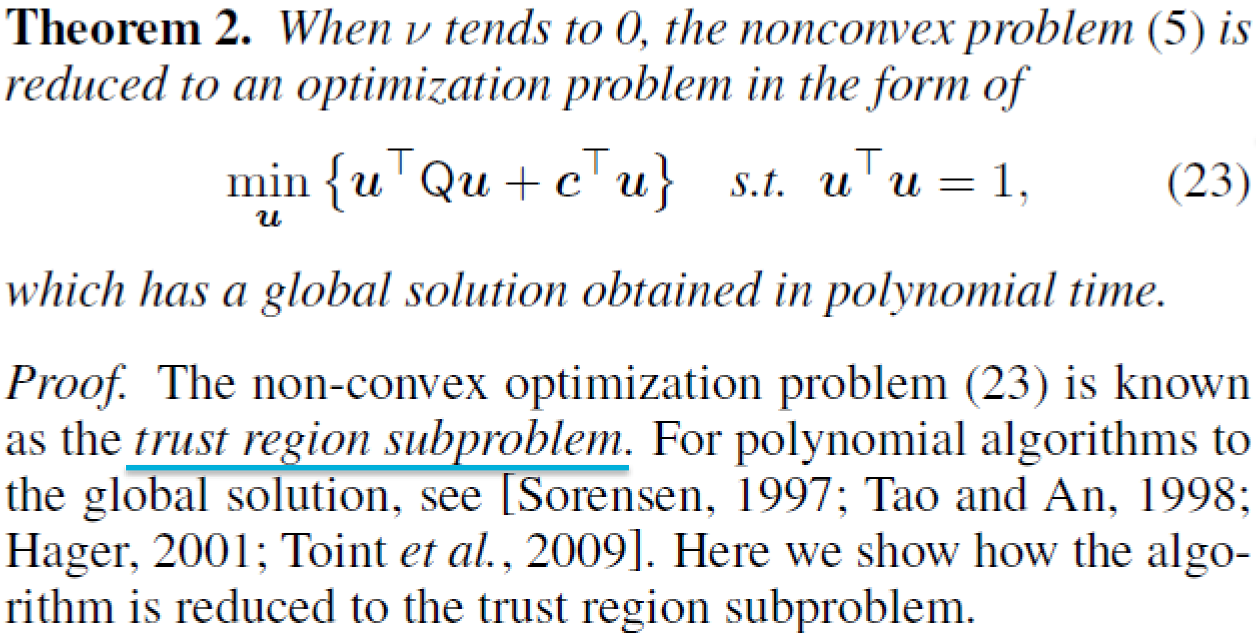
$$\boldsymbol{q} \equiv \ln c_M \boldsymbol{b} + \kappa \mathsf{X}^\top \boldsymbol{u}_1$$

This Lasso problem is solved analytically



```
Algorithm 1 RED algorithm.
Input: Initialized $\boldsymbol{w}$. Regularization parameters $\lambda, \nu$. Concentration parameter $\kappa$. The number of major directional patterns $m$.
Output: $\mathsf{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m]$ and $\mathsf{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m]$.
for $j = 1, 2, \ldots, m$ do
  while no convergence do
    $\boldsymbol{u}_j \leftarrow \kappa[\mathsf{I}_M - \mathsf{U}_{j-1}\mathsf{U}_{j-1}^\top]\mathsf{X}\boldsymbol{w}_j$    (17)
    $\boldsymbol{u}_j \leftarrow \mathrm{sign}(\boldsymbol{u}_j^\top \mathsf{X} \boldsymbol{w}_j) \frac{\boldsymbol{u}_j}{\|\boldsymbol{u}_j\|_2}$    (18)
    $\boldsymbol{q}_j \leftarrow \gamma \boldsymbol{b} + \kappa \mathsf{X}^\top \boldsymbol{u}_j$    (19)
    $\boldsymbol{w}_j \leftarrow \mathrm{sign}(\boldsymbol{q}_j) \odot \max\left\{\frac{|\boldsymbol{q}_j|}{\lambda} - \nu \mathbf{1}, \mathbf{0}\right\}$    (20)
  end while
end for
Return U and W.
```

Theoretical property: The algorithm is reduced to the "trust-region subproblem" in $\nu \rightarrow 0$



Theorem 2. *When ν tends to 0, the nonconvex problem (5) is reduced to an optimization problem in the form of*

$$\min_{\boldsymbol{u}} \{\boldsymbol{u}^\top \mathsf{Q}\boldsymbol{u} + \boldsymbol{c}^\top \boldsymbol{u}\} \quad s.t. \;\; \boldsymbol{u}^\top \boldsymbol{u} = 1, \tag{23}$$

which has a global solution obtained in polynomial time.

Proof. The non-convex optimization problem (23) is known as the *trust region subproblem*. For polynomial algorithms to the global solution, see [Sorensen, 1997; Tao and An, 1998; Hager, 2001; Toint *et al.*, 2009]. Here we show how the algorithm is reduced to the trust region subproblem.

Useful to initialize the iterative algorithm

Change score as parameterized Kullback-Leibler divergence

- With extracted directions, define the change score at time *t* as

vMF distribution

vMF distribution

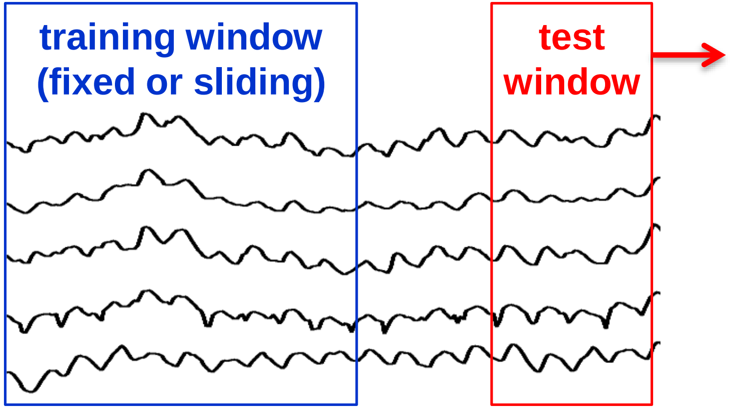
$$a^{(t)} = \min_{\boldsymbol{f},\boldsymbol{g}} \int \mathrm{d}\boldsymbol{x}\, \mathcal{M}(\boldsymbol{x}|\mathsf{U}\boldsymbol{f},\kappa) \ln \frac{\mathcal{M}(\boldsymbol{x}|\mathsf{U}\boldsymbol{f},\kappa)}{\mathcal{M}(\boldsymbol{x}|\mathsf{U}^{(t)}\boldsymbol{g},\kappa)}$$

vMF dist.

$$\boldsymbol{f}^\top \boldsymbol{f} = 1,\ \boldsymbol{g}^\top \boldsymbol{g} = 1$$

- Concisely represented by the top singular value of $\mathsf{U}^\top \mathsf{U}^{(t)}$

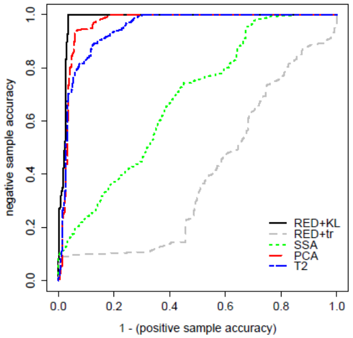
$\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m$ $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r$



$\mathsf{U} \equiv [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m]$ $\mathsf{U}^{(t)} \equiv [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r]$

Experiment: Failure detection of ore belt conveyors

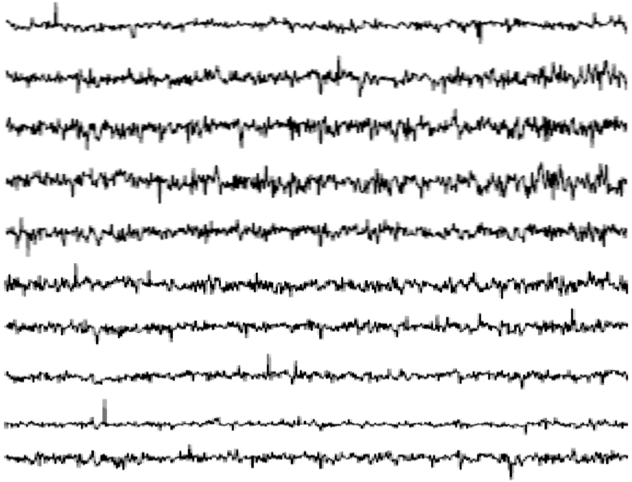
- vMF formulation successfully suppressed very noisy non-Gaussian noise of multiplicative nature
- ~40% of samples were automatically excluded from the model
- Better than alternatives
  - PCA, Hoteling T^2
  - Stationary subspace analysis [Blythe et al., 2012]



Learned sample weights

~ 40% samples have zero weight: automated sample size reduction

Training data

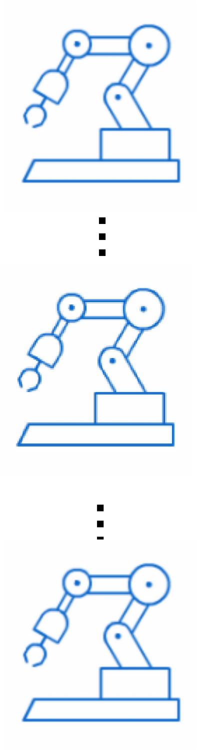


(simulation data)

Agenda

- General challenges in industrial sensor data analytics
- Solution examples:
  - Change detection using directional statistics
  - Multi-task multi-modal models for collective anomaly detection (Ide et al, ICDM 17)
  - Tensorial change analysis
- Discussion: deep learning, Blockchain, and future directions

Wish to build a collective monitoring solution



System 1
(in New
Orleans)

System *s*

System *S*
(in New York)

- You have many similar but not identical industrial assets

- You want to build an anomaly detection model for each of the assets

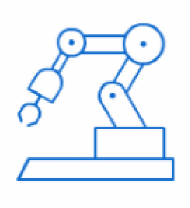
- Straightforward solutions have serious limitations
  - 1. Treat the systems separately. Create each model individually
    - ✓ Suffers from lack of fault examples
  - 2. Build one universal model by disregarding individuality
    - ✓ Model fit is not good

Practical requirements: Need to capture both commonality and individuality

System 1
(in New
Orleans)

⋮

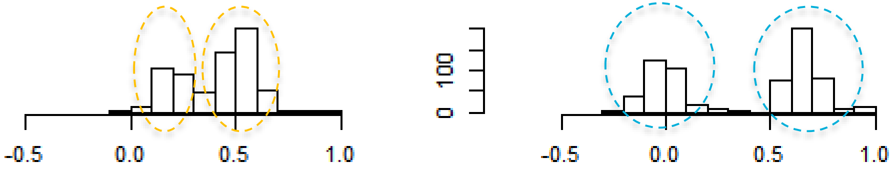
System *s*



⋮

System *S*
(in New York)

- Capture both individuality and commonality
- Automatically capture multiple operational states
  - Real-world is not single-peaked / single-modal



- Be robust to noise
- Be highly interpretable for diagnosis purposes

Formalizing the problem as multi-task density estimation for anomaly detection

Data

Prob. density

Anomaly score

System 1
(in New
Orleans)

$\{\boldsymbol{x}^{1(n)} \in \mathbb{R}^M\}$

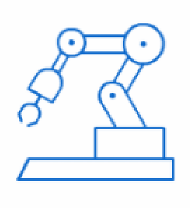
$p^1(\boldsymbol{x}^1 \mid \mathcal{D})$

↑ all data

$a^1(\boldsymbol{x}^1)$

⋮

System *s*



$\{\boldsymbol{x}^{s(n)} \in \mathbb{R}^M\}$

multi-task learning (MTL)

$p^s(\boldsymbol{x}^s \mid \mathcal{D})$

$a^s(\boldsymbol{x}^s)$

- overall
- variable-wise

⋮

System *S*
(in New York)

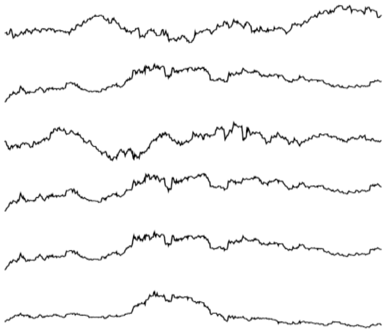
$\{\boldsymbol{x}^{S(n)} \in \mathbb{R}^M\}$

$p^S(\boldsymbol{x}^S \mid \mathcal{D})$

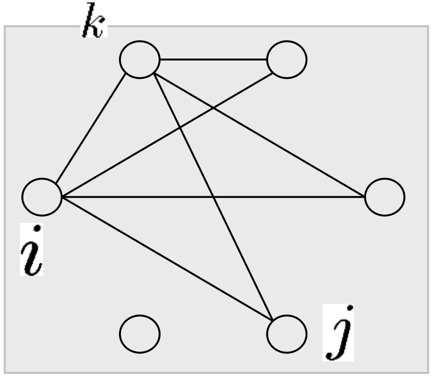
$a^S(\boldsymbol{x}^S)$

Use Gaussian graphical model (GGM)-based anomaly detection approach as the basic building block

Multi-variate data



Sparse graphical model



$$\max_{\Lambda} \{ \ln\det\Lambda - \mathrm{tr}(\Sigma\Lambda) - \rho||\Lambda||_1 \}$$

sample covariance

training data

Anomaly score

$$a(\boldsymbol{x}) =$$

$$-\ln p(\boldsymbol{x} \mid \mathcal{D})$$

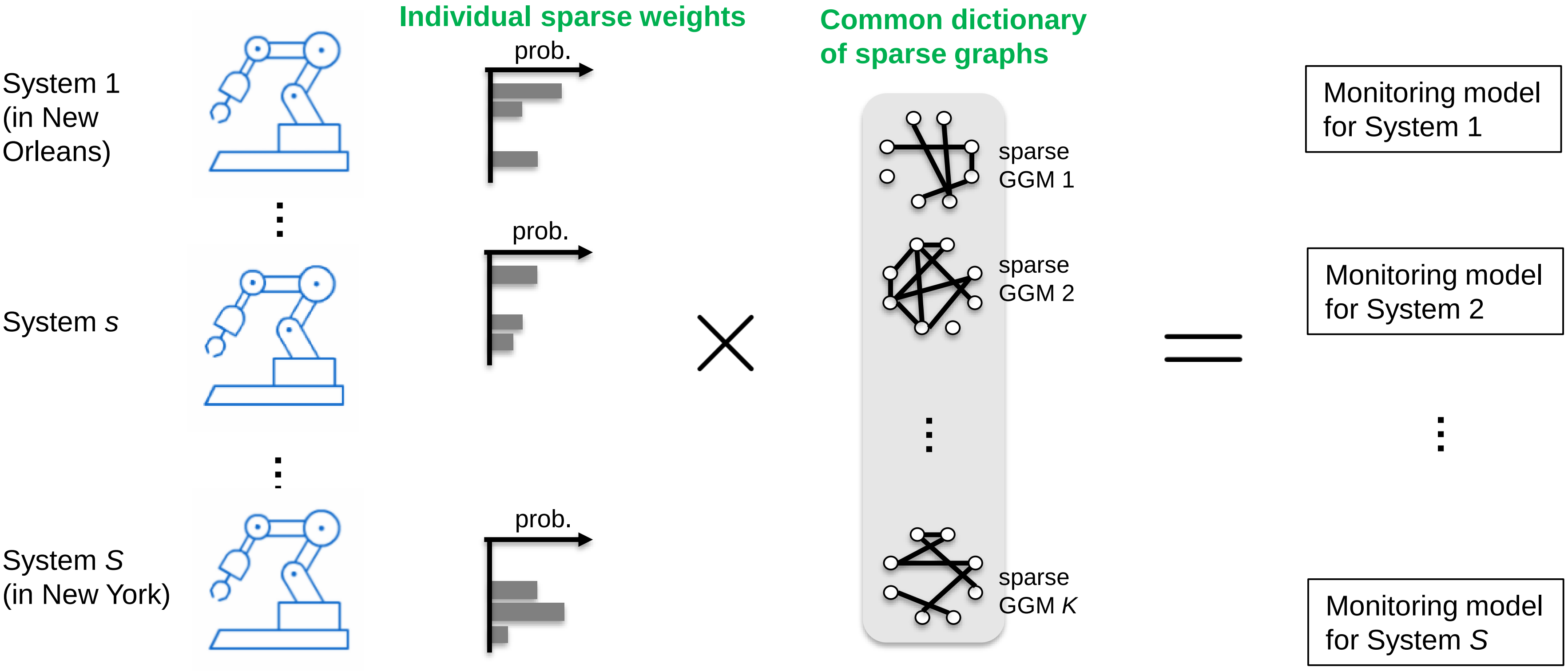
Overall score

$$-\ln p(x_i \mid \boldsymbol{x}_{-i}, \mathcal{D})$$

Variable-wise score

[Ide+ SDM09] [Ide+ ICDM16]

Basic modeling strategy: Combine common pattern dictionary with individual weights



Individual sparse weights

Common dictionary of sparse graphs

System 1
(in New Orleans)

prob.

⋮

System *s*

prob.

⋮

System *S*
(in New York)

prob.

×

sparse GGM 1

sparse GGM 2

⋮

sparse GGM *K*

=

Monitoring model for System 1

Monitoring model for System 2

⋮

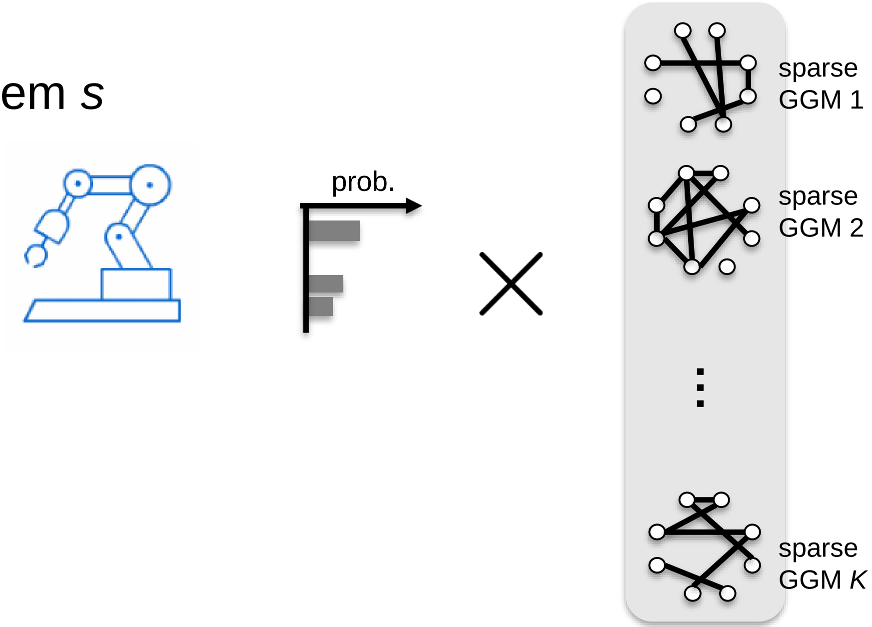
Monitoring model for System *S*

GGM=Gaussian Graphical Model

Basic modeling strategy: Resulting model will be a sparse mixture of sparse GGM

Monitoring model for System s

System *s*



GGM=Gaussian Graphical Model

Gaussian mixture

$$= \sum_{k=1}^{K} \pi_k^s \, \mathcal{N}(\boldsymbol{x}^s \mid \boldsymbol{\mu}^k, (\Lambda^k)^{-1})$$

Sparse mixture weights

(= automatic determination of the number of patterns)

Sparse Gaussian graphical model

Propose a Bayesian multi-task model with two sparsity-enforcing priors

- Observation model (for the *s*-th task)
  - Gaussian mixture with task-dependent weight

$$\prod_{k=1}^{K} \mathcal{N}(\boldsymbol{x}^s \mid \boldsymbol{\mu}^k, (\Lambda^k)^{-1})^{z_k^s}$$

- Sparsity enforcing priors (non-conjugate)
  - Laplace prior for the precision matrix
  - Bernoulli prior for the mixture weights

$$p(\Lambda^k) = \left(\frac{\rho}{4}\right)^{M^2} \exp\left(-\frac{\rho}{2}\|\Lambda^k\|_1\right)$$

$$p(\boldsymbol{\pi}^s) = p_0^{\|\boldsymbol{\pi}^s\|_0}(1-p_0)^{G-\|\boldsymbol{\pi}^s\|_0}$$

- Conjugate prior on $\{\boldsymbol{\mu}^k\}$ and $\{\boldsymbol{z}^s\}$

$$p(\boldsymbol{\mu}^k \mid \Lambda^k) = \mathcal{N}(\boldsymbol{\mu}^k | \boldsymbol{m}^0, (\lambda_0 \Lambda^k)^{-1})$$

$$p(\boldsymbol{z}^s | \boldsymbol{\pi}^s) = \prod_{k=1}^{K} (\pi_k^s)^{z_k^s}$$

Maximizing log likelihood using variational Bayes combined with point-estimation

- Complete log likelihood

$$L = \underbrace{\sum_{s=1}^{S}\sum_{n=1}^{N_s}\sum_{k=1}^{K} \ln \mathcal{N}(\boldsymbol{x}^{s(n)} \mid \boldsymbol{\mu}^k)^{z^{s(n)}}}_{\text{Likelihood by the obs. model}} + \underbrace{\sum_{k=1}^{K} \mathrm{Lap}(\Lambda^k \mid \rho)p(\boldsymbol{\mu}^k \mid \Lambda^k) + \sum_{s=1}^{S} z^{s(n)} \ln \pi_k^s + \sum_{s=1}^{S} \ln p(\boldsymbol{\pi}^s)}_{\text{Prior distributions}}$$

- Use VB for $\{\boldsymbol{\mu}^k\}, \{\boldsymbol{z}^{s(n)}\}$
- Use point-estimate for $\{\Lambda^k\}, \{\boldsymbol{\pi}^s\}$
  - Results in two convex optimization problems

Maximizing log likelihood using variational Bayes combined with point-estimation

- Update sample weights
- Update cluster weights
- Update precision matrices
- Update other parameters

Use new semi-closed form solution

$$\max_{\boldsymbol{\pi}^s} \left\{ \sum_{k=1}^{K} c_k^s \ln \pi_k^s - \tau \|\boldsymbol{\pi}^s\|_0 \right\} \quad \text{s.t.} \quad \|\boldsymbol{\pi}^s\|_1 = 1.$$

The ratio of samples assigned to the *k*-th cluster

Solved by graphical lasso [Friedman 08]

$$\max_{\Lambda^k} \left\{ \ln |\Lambda^k| - \mathrm{Tr}(\Lambda^k \mathsf{Q}^k) - \frac{\rho}{N_k} \|\Lambda^k\|_1 \right\}$$

total # of samples assigned to the *k*-th cluster

Solving the L_0-regularized optimization problem for mixture weights

- Conventional VB approach without L_0 regularization on π_k^s is problematic
  - Claimed to get a sparse solution [Corduneanu+ 01]
  - But mathematically π_k^s cannot be zero due to logarithm

$$\max_{\boldsymbol{\pi}^s} \left\{ \sum_{k=1}^{K} c_k^s \ln \pi_k^s \right\}$$

$$\text{s.t.} \quad \|\boldsymbol{\pi}^s\|_1 = 1.$$

- We re-formalized the problem as a convex mixed-integer programming
  - A semi-closed form solution can be derived (→ see paper)

$$\max_{\boldsymbol{\pi}^s, \boldsymbol{y}^s} \sum_{k=1}^{K} \{c_k^s \ln \pi_k^s - \tau y_k^s\} \quad \text{s.t.} \quad \sum_{k=1}^{K} \pi_k^s = 1,$$

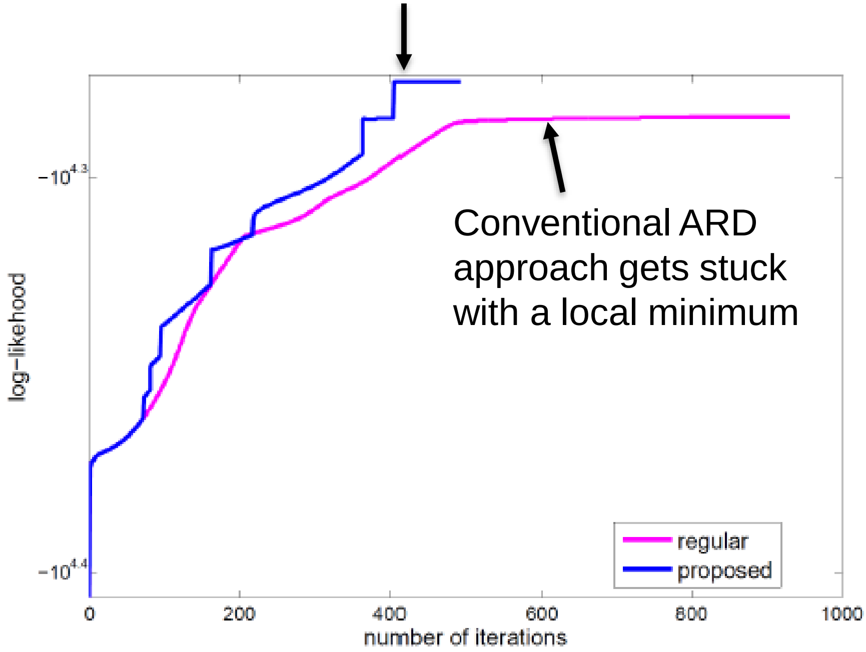
$$y_k^s \geq \pi_k^s - \epsilon, \quad y_k^s \in \{0, 1\} \quad \text{for} \quad k = 1, \ldots, K,$$

Comparison with possible alternatives

		Interpretability	Noise reduction	Fleet-readiness	Multi-modality
Our work [Ide et al. ICDM 17]		**Yes**	**Yes**	**Yes**	**Yes**
(single) sparse GGM	[Ide et al. SDM 2009, Ide et al. ICDM 2016]	**Yes**	**Yes**	No	No
Gaussian mixtures	[Yamanishi et al., 2000; Zhang and Fung, 2013; Gao et al., 2016]	Limited	Limited	No	**Yes**
Multi-task sparse GGM	[Varoquaux et al., 2010; Honorio and Samaras, 2010; Chiquet et al., 2011; Danaher et al., 2014; Gao et al., 2016; Peterson et al., 2015].	**Yes**	**Yes**	**Yes**	No
Multi-task learning anomaly detection	[Bahadori et al., 2011; He et al., 2014; Xiao et al., 2015]	No	(depends)	**Yes**	No

Experiment (1): Learning sparse mixture weights

- Conventional ARD approach sometimes gets stuck with local minima
  - ARD = automatic relevance determination
  - Often less sparse than the proposed convex L_0 approach
- Typical result of log likelihood vs VB iteration count →



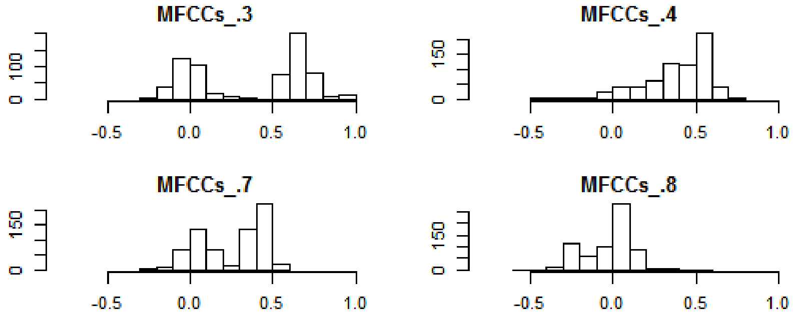
Proposed convex L_0 approach gives better likelihood

Conventional ARD approach gets stuck with a local minimum

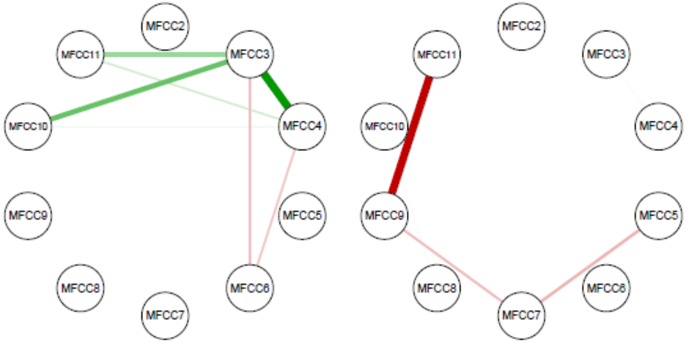
Experiment (2): Learning GGMs and detecting anomalies

- "Anuran Calls" (frog voice) data in UCI Archive
    - Multi-modal (multi-peaked)
    - Voice signal + attributes (species, etc.)
- Created 10-variate, 3-task dataset
    - Use the species of "Rhinellagranulosa" as the anomaly
- Results
    - Two non-empty GGMs are automatically detected starting from K=9
    - Clearly outperformed single-modal MTL alternative in anomaly detection
        - ✓ Group graphical lasso, fused graphical lasso

Example of variable-wise distribution



Automatically learned GGMs



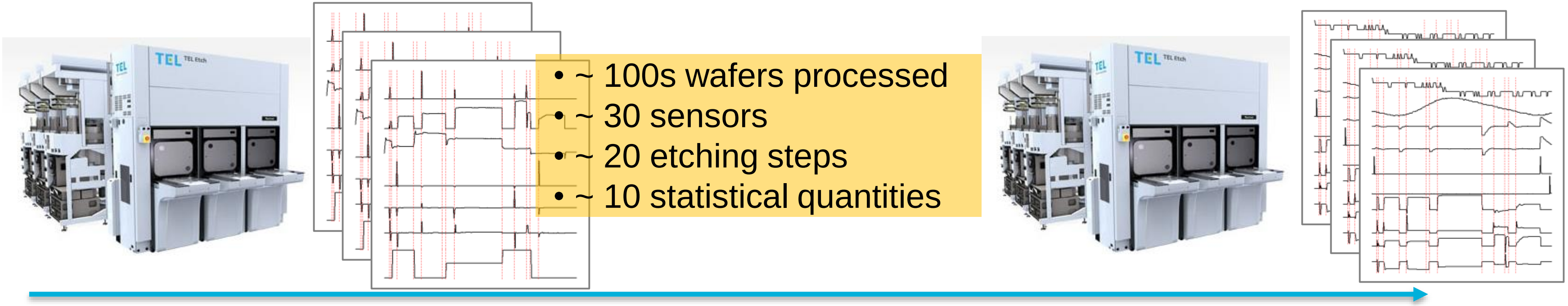
Agenda

- General challenges in industrial sensor data analytics
- Solution examples:
  - Change detection using directional statistics
  - Multi-task multi-modal models for collective anomaly detection
  - Tensorial change analysis
- Discussion: deep learning, Blockchain, and future directions

Developing a system for change diagnosis when input data is a tensor (multi-way array)

- Real application example: Condition-based monitoring of reactive ion etching tool
  - Tools deteriorate over time due to debris in the etching chamber
  - Degradation process is implicit and subtle. Quantification is challenging

- Basic problem setting: Compare a test period with a reference period to explain what really is the difference in terms of observable variables



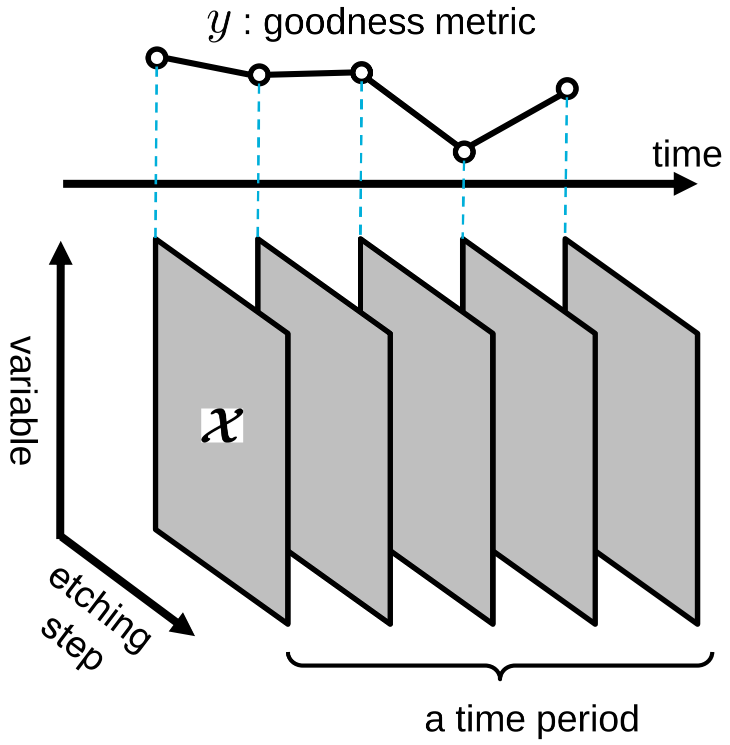
- ~ 100s wafers processed
- ~ 30 sensors
- ~ 20 etching steps
- ~ 10 statistical quantities

“golden period” (or reference period)

test period in question

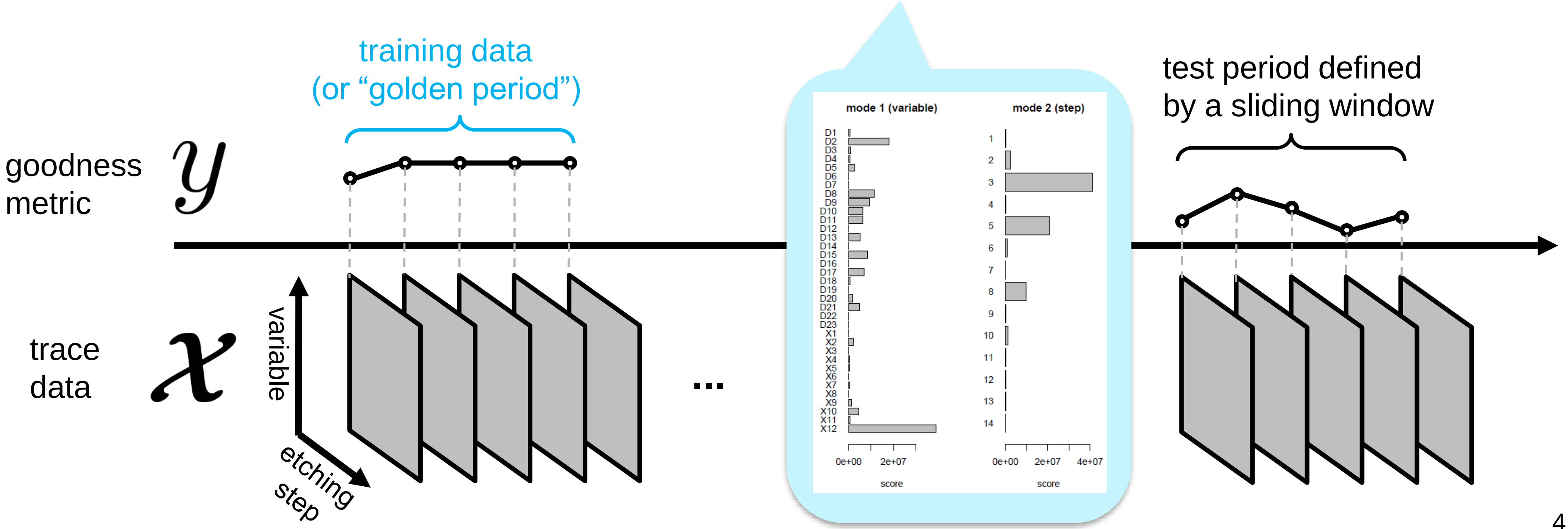
The input is a tensor (multi-way array) associated with a goodness metric

- Semiconductor etching example
    - y: (one of) quality measurements
        - ✓ electric resistance, line widths, ...
    - X: “trace data” (sensor recordings)
        - ✓ pressure, temperature, electric current, ...
- One etching round of trace data is most naturally represented as a tensor (multi-way array)
    - Typically 3-way array
        - ✓ variable x etching step x statistics used x time
        - ✓ variable x etching step x etching metal layer
    - Often summarized as 2-way tensor by e.g. taking the mean over time in each step



The task addressed: (1) Detect a change in X-y relationship. (2) Explain which mode/dimension is most responsible

- (1) Compute the anomalousness of a single or a set of etching round(s) in a test period
- (2) Compute the responsibility of the dimensions of each mode that explains the anomalousness of the test period



Technical challenges

Tensor regression is not well-studied

- Regression is the task to learn a function y = f(X) from training data
- Existing techniques mainly use vectorization of tensors

Probabilistic prediction is even harder

- Non-subjective change scoring requires probabilistic prediction.
- Existing probabilistic tensor regression methods are impractical

Vectorized probabilistic model cannot be the solution

- Not very interpretable – it destroys the tensor structure of the input

Tensorial change diagnosis framework using probabilistic tensor regression algorithm

Agenda

- General challenges in industrial sensor data analytics
- Solution examples:
  - Change detection using directional statistics
  - Multi-task multi-modal models for collective anomaly detection
  - Tensorial change analysis
- Discussion: deep learning, Blockchain, and future directions

Does deep learning mean the end of journey? Probably not.
Factors that make deep learning work

Well-defined and well-accepted task

No need to tell why

Huge amount of <u>labeled</u> training data

Typically needs millions labeled samples

Minimum uncertainty in data representation

Pixels, words, Mel-filterbank

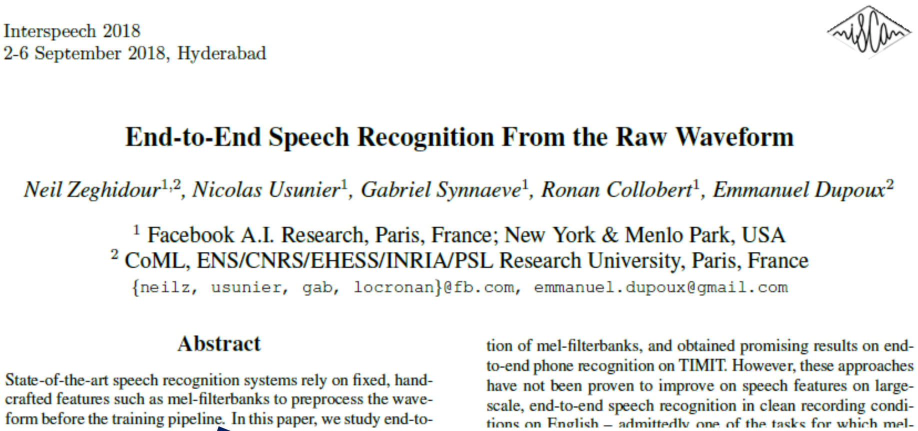
- Good applications meeting these criteria
  - Image recognition
  - (Some of) natural language processing
  - Speech recognition
- How about industrial dynamic systems?
  - Interesting research topic

One caveat: Automated feature learning from noisy senor signal is still challenging

- Image recognition and NLP (natural language processing) are an ideal area for deep learning
  - Huge annotated datasets exist
  - Established preprocess method

- A little secret in speech recognition: State-of-the-art deep-learning-based systems use handcrafted features

- The situation will be much tougher in general industrial sensor data analytics



Interspeech 2018
2-6 September 2018, Hyderabad

End-to-End Speech Recognition From the Raw Waveform

Neil Zeghidour[1,2], *Nicolas Usunier*[1], *Gabriel Synnaeve*[1], *Ronan Collobert*[1], *Emmanuel Dupoux*[2]

[1] Facebook A.I. Research, Paris, France; New York & Menlo Park, USA
[2] CoML, ENS/CNRS/EHESS/INRIA/PSL Research University, Paris, France
`{neilz, usunier, gab, locronan}@fb.com, emmanuel.dupoux@gmail.com`

Abstract

State-of-the-art speech recognition systems rely on fixed, handcrafted features such as mel-filterbanks to preprocess the waveform before the training pipeline. In this paper, we study end-to-

tion of mel-filterbanks, and obtained promising results on end-to-end phone recognition on TIMIT. However, these approaches have not been proven to improve on speech features on large-scale, end-to-end speech recognition in clean recording conditions on English – admittedly one of the tasks for which mel-

> "State-of-the-art speech recognition systems rely on fixed, handcrafted features such as mel-filterbanks to preprocess the waveform before the training pipeline"

Implications for sensor data analytics

- Deep learning (especially RNN such as LSTM) will be a powerful tool when
  - we know how to read the data (and thus a good amount of labeled training data exists)
  - we know limitations of linear models (state-space models)
  - we have a lot of GPU!

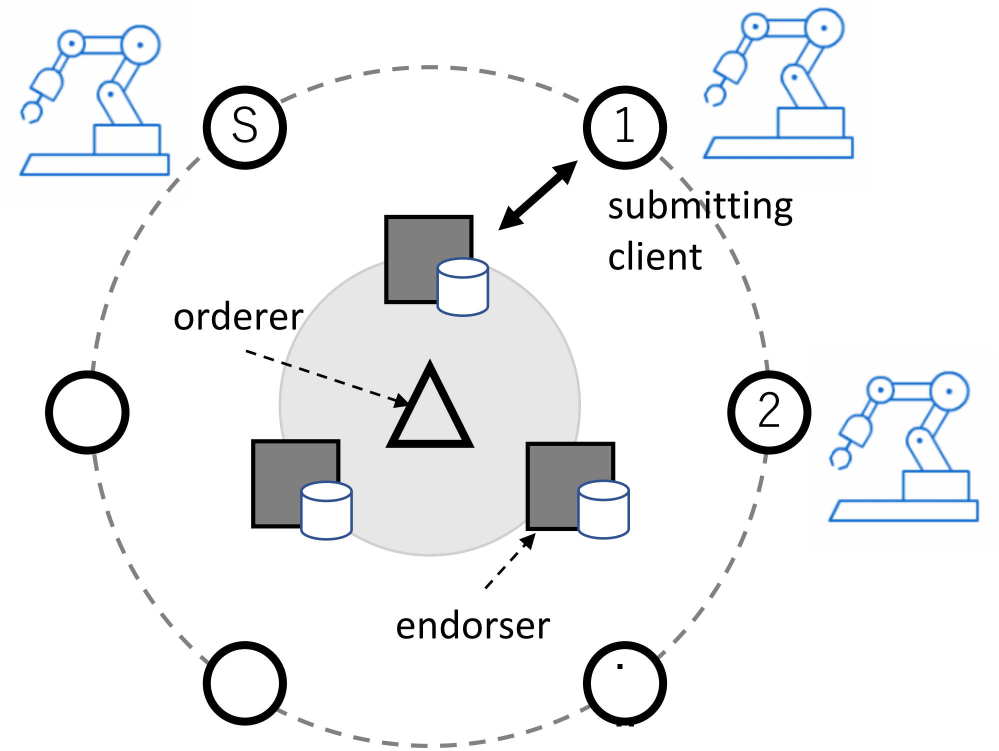
Discussion: Will Blockchain bring in any value on sensor data analytics?

- What is Blockchain?
    - Distributed decentralized database characterized by a hash chain data structure and a consensus algorithm
- Blockchain generations
    - 1st generation (Bitcoin)
        - ✓ De-centralized, secure platform for money transfer
    - 2nd generation (Ethereum, Hyperledger, Corda)
        - ✓ Extended to handle general business transactions beyond money transfer
- Expected to be a useful platform for IoT (internet-of-things) systems
    - “Device democracy”

Discussion: Will Blockchain bring in any value on sensor data analytics?

- Blockchain should be generalized as a collaborative learning platform
  - “Blockchain 3.0”
  - The particular hash chain data structure can be viewed as just one instance of implementation

- Example: privacy-preserving multi-task learning on Blockchain



Summary and ongoing work

- Industrial sensor data have many interesting features that call for new machine learning formulation
- Introduced a few recent works on anomaly detection
  - Change detection using directional statistics
  - Multi-task anomaly detection algorithm
  - Tensorial change analysis
- Ongoing/future work
  - Prediction/anomaly detection from novel data types
    - ✓ tensors, functions, graphs, trajectories, events, etc.
  - Multi-x / cross-x learning
    - ✓ multi-task, view, domain, modality
  - Deep learning for dynamic systems

Thank you!