IBM **Research**

Sparse Gaussian Markov Random Field Mixtures for Anomaly Detection

Tsuyoshi Idé (“Ide-san”), Ankush Khandelwal*, Jayant Kalagnanam
IBM Research, T. J. Watson Research Center
(*Currently with University of Minnesota)

Proceedings of the 2016 IEEE International Conference on Data Mining (ICDM 16), Dec. 13-15, 2016, pp.955-960.

Summary: Gaussian mixture + anomaly detection

Newly added features:

- **Principled variable-wise scoring**
- **Mixture extension of sparse structure learning**

Define variable-wise anomaly score as conditional log-loss

- Anomaly score for the *i*-th variable of a new sample $\boldsymbol{x} \in \mathbb{R}^M$

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

conditional predictive distribution

- x_i: the *i*-th variable
- $\boldsymbol{x}_{-i}$: the rest
- D : training data

- c.f. overall score [Yamanishi+ 00]

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

predictive distribution for ***x***

"*a* will be large if ***x*** falls in the area where *p*(***x*** | *D*) is small"

(For ref.) Why negative log p? It reproduces Mahalanobis distance in the single Gaussian case

Gaussian with mean μ and covariance $\mathbf{\Sigma}$

$$\begin{aligned} a(\boldsymbol{x}) &= -\ln \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}, \Sigma) \\ &= \text{const.} + \frac{1}{2} \underbrace{(\boldsymbol{x} - \boldsymbol{\mu})^\top \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu})}_{\text{Mahalanobis distance}} \end{aligned}$$

Mahalanobis distance

Use mixture of Gaussian Markov random fields (GMRF) for the conditional distribution

Variable-wise mixture weight

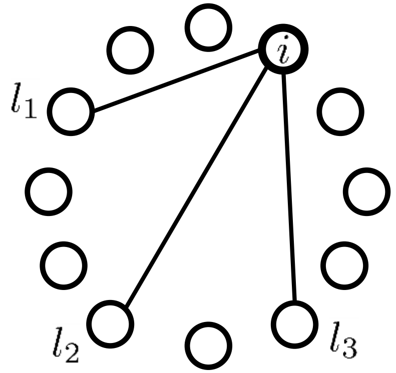
$$p(x_i|\boldsymbol{x}_{-i},\mathcal{D}) = \sum_{k=1}^{K} g_k^i(\boldsymbol{x}) \mathcal{N}\left(x_i \mid u_i^k, w_i^k\right),$$

GMRF

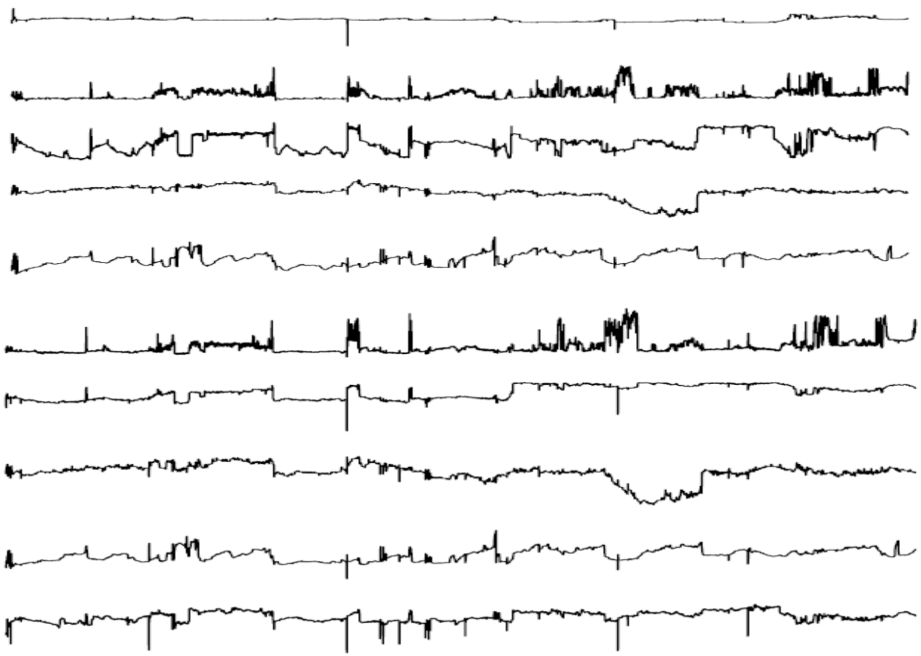
$$u_i^k = m_i^k - \frac{1}{A_{i,i}^k} \sum_{l \neq i}^{M} A_{i,l}^k (x_l - m_l^k)$$

$$w_i^k = \frac{1}{A_{i,i}^k}.$$

- GMRF is characterized as conditional distribution of Gaussian $\mathcal{N}(\boldsymbol{x} \mid \boldsymbol{m}^k, (\mathsf{A}^k)^{-1})$
- GMRF descries dependency among variables



Mixture model with i.i.d. assumption on data is a practical compromise: Compressor data example

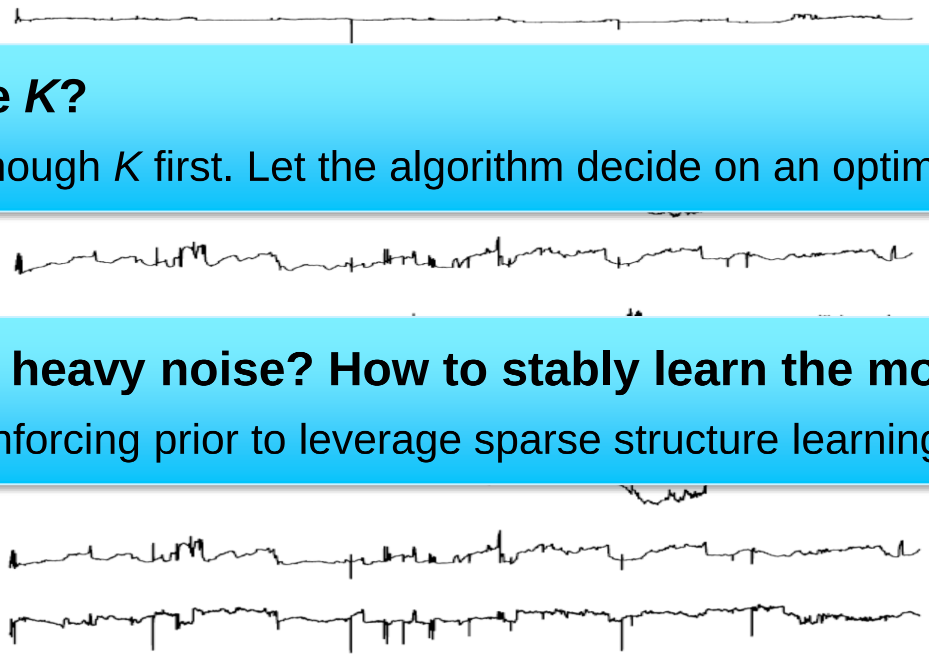


This is **normal** operation data

It looks piece-wise stationary (with heavy noise)

Time-series modeling looks too hard

Tackling noisy data of complex systems: Strategy of designing inference algorithm



How to choose *K*?

- Give a large enough *K* first. Let the algorithm decide on an optimal value.

How to handle heavy noise? How to stably learn the model?

- Use sparsity-enforcing prior to leverage sparse structure learning

Two-step approach to GMRF mixture learning: Model

$$p(x_i|\boldsymbol{x}_{-i},\mathcal{D}) = \sum_{k=1}^{K} g_k^i(\boldsymbol{x}) \mathcal{N}\left(x_i \mid u_i^k, w_i^k\right),$$

Step 1: Find GMRF parameters

- Observation model

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

- Priors:
  Gauss-Laplace for (μ^k, Λ^k)
  Categorical for $\{z_k\}$ (each sample)

Step 2: Find variable-wise weights given GMRF parameters

- Observation model

$$p(x_i \mid \boldsymbol{x}_{-i}, \boldsymbol{h}^i) = \prod_{k=1}^{K} \mathcal{N}\left(x_i \mid u_i^k, w_i^k\right)^{h_k^i}$$

- Priors:
  Categorical-Dirichlet for $\{h_k^i\}$ (each sample)

Two-step approach to GMRF mixture learning: Inference

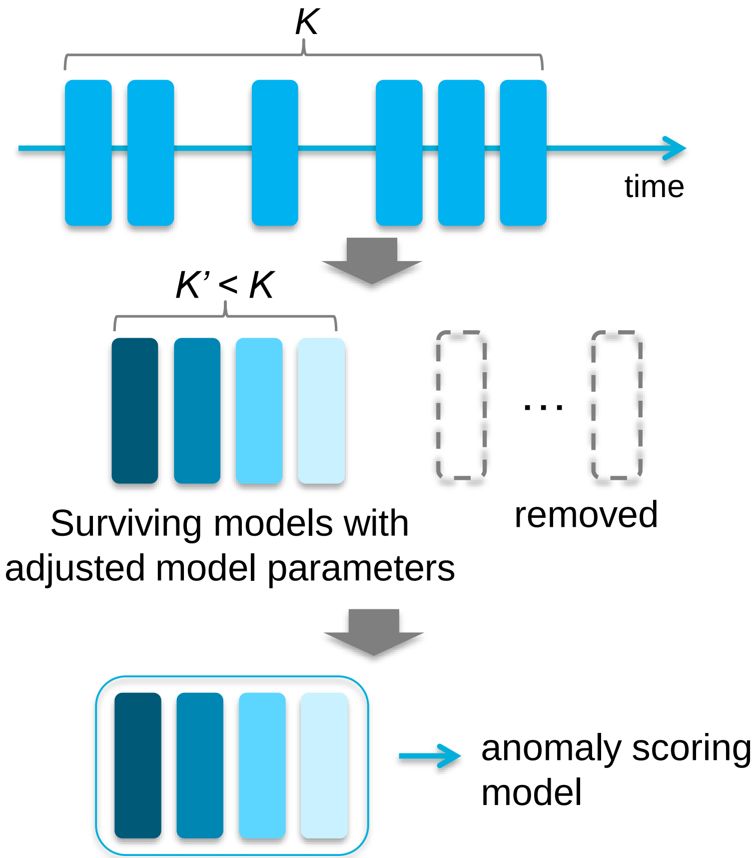
- Use variational Bayes (VB) for the 1st and 2nd steps
- The 1st step achieves sparsity over variable dependency and mixture components
  - Variable dependency: (iteratively) solve graphical lasso [Friedman+ 08]

$$\bar{\Lambda}^k \leftarrow \arg\max_{\Lambda^k} \left\{ \ln |\Lambda^k| - \mathrm{Tr}(\Lambda^k \mathrm{Q}^k) - \frac{\rho}{N^k} \|\Lambda^k\|_1 \right\}.$$

  - Mixture components: (iteratively) point-estimated for ARD (automated relevance determination) [Corduneanu+ 01]
- Details → paper

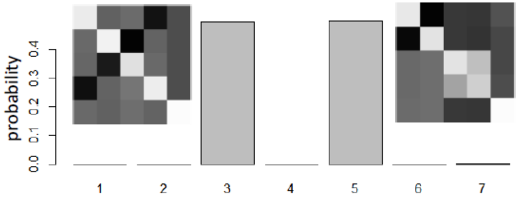
Overview of the approach (for multivariate noisy sensor data)

- Initialize
  - Randomly pick time-series blocks with a large enough K
  - Run graphical lasso separately to initialize $\{(\mu^k, \Lambda^k)\}$
- Step 1
  - Iteratively update $\{(\mu^k, \Lambda^k)\}$ and
  - remove clusters with zero weight
- Step 2
  - Compute variable-wise mixture weights
  - Produce anomaly scoring model

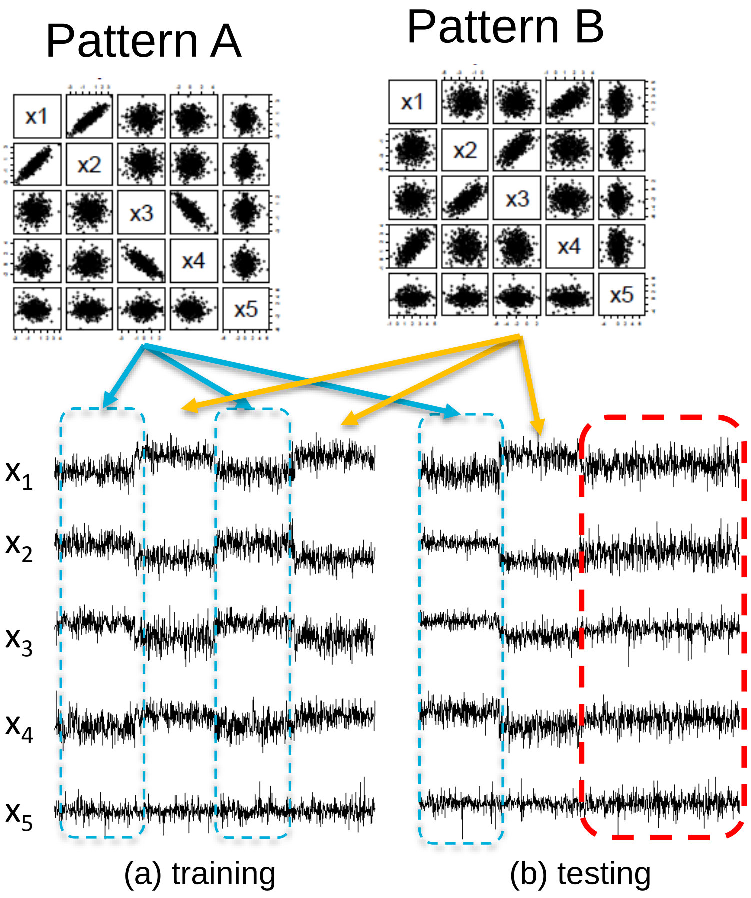


Results: Synthetic data
(see paper for real application)

- Data generated
  - Training: A-B-A-B
  - Testing: A-B-(anomaly)
- Results
  - Successfully recovered 2 major patterns starting from K=7



  - Achieved better performance in anomaly detection (in terms of AUC)



Conclusion

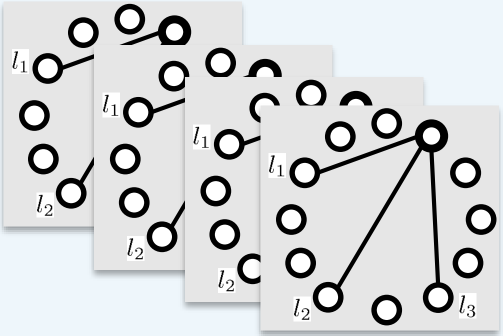
- Proposed a new outlier detection method, the sparse GMRF mixture
- Our method is capable of handling multiple operational states in the normal condition and variable-wise anomaly scores.
- Derived variational Bayes iterative equations based on the Gauss-delta posterior model

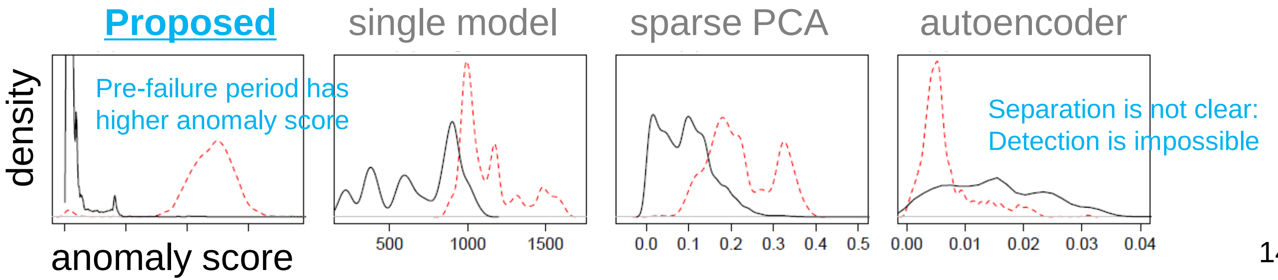
Thank you!

Results: Detecting pump surge failures

- Computed anomaly score for x_{14}, which is a flow-rate variable, based on the normal state model learned
- Compared anomaly score between the pre-failure region (24h) and several normal periods
  - Black: normal
  - red: pre-failure period
- Clearly outperformed alternative methods including neural network (autoencoder)

Meta model learned





Leveraging variational Bayes method for inference

- Assumption: posterior distribution is factorized

$$p(\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{\mu}^1, \ldots, \Lambda^1, \ldots) = \prod_{n=1}^{N} q(\boldsymbol{z}^{(n)}) \prod_{k=1}^{K} q(\boldsymbol{\mu}^k, \Lambda^k)$$

- Posterior is determined so that the KL divergence between the factorized form and the full posterior
    - Full posterior is proportional to the complete likelihood

Two-step approach to GMRF mixture learning: Inference

- Use variational Bayes (VB) for the 1st and 2nd steps
- The 1st step achieves sparsity over variable dependency and mixture components
  - Variable dependency: (iteratively) solve graphical lasso
  - Mixture components: point-estimated for ARD (automated relevance determination)
    - ✓ [Corduneanu-Bishop 01]

VB iteration for the 1st step

$$N^k \leftarrow \sum_{n=1}^{N} r_k^{(n)}, \quad \pi_k \leftarrow \frac{N^k}{N},$$

Point-estimated cluster weight

$$\bar{\boldsymbol{x}}^k \leftarrow \frac{1}{N^k} \sum_{n=1}^{N} r_k^{(n)} \boldsymbol{x}^{(n)},$$

$$\Sigma^k \leftarrow \frac{1}{N^k} \sum_{n=1}^{N} r_k^{(n)} (\boldsymbol{x}^{(n)} - \bar{\boldsymbol{x}}^k)(\boldsymbol{x}^{(n)} - \bar{\boldsymbol{x}}^k)^\top,$$

$$\lambda^k \leftarrow \lambda_0 + N^k, \quad \boldsymbol{m}^k \leftarrow \frac{1}{\lambda^k}(\lambda_0 \boldsymbol{m}_0 + N^k \bar{\boldsymbol{x}}^k),$$

$$\mathrm{Q}^k \leftarrow \Sigma^k + \frac{\lambda_0}{\lambda^k}(\bar{\boldsymbol{x}}^k - \boldsymbol{m}_0)(\bar{\boldsymbol{x}}^k - \boldsymbol{m}_0)^\top,$$

$$\bar{\Lambda}^k \leftarrow \arg\max_{\Lambda^k} \left\{ \ln|\Lambda^k| - \mathrm{Tr}(\Lambda^k \mathrm{Q}^k) - \frac{\rho}{N^k} \|\Lambda^k\|_1 \right\},$$

graphical lasso [Friedman+ 08]