Sparse Gaussian Markov Random Field Mixtures for Anomaly Detection

Tsuyoshi Idé
IBM Research
T. J. Watson Research Center
tide@us.ibm.com

Ankush Khandelwal
University of Minnesota
Department of Computer Science
ankush@cs.umn.edu

Jayant Kalagnanam
IBM Research
T. J. Watson Research Center
jayant@us.ibm.com

***Abstract*—We propose a new approach to anomaly detection from multivariate noisy sensor data. We address two major challenges: To provide variable-wise diagnostic information and to automatically handle multiple operational modes. Our task is a practical extension of traditional outlier detection, which is to compute a single scalar for each sample. To consistently define the variable-wise anomaly score, we leverage a predictive conditional distribution. We then introduce a mixture of Gaussian Markov random field and its Bayesian inference, resulting in a sparse mixture of sparse graphical models. Our anomaly detection method is capable of automatically handling multiple operational modes while removing unwanted nuisance variables. We demonstrate the utility of our approach using real equipment data from the oil industry.**

1. Introduction

Anomaly detection from sensor data is one of the critical applications of data mining. In the standard setting, we are given a data set under a normal operating condition, and we build a statistical model as a compact representation of the normal state. In operation, when a new observation is provided, we evaluate the discrepancy from what is expected by the normal model. This paper focuses on a different anomaly detection scenario where the dataset has multiple normal operating conditions. Moreover, instead of reporting a single anomaly score, the goal is to compute anomaly score for each variable separately.

In spite of the long history of research in statistics, as represented by the classical Hotelling's T^2 theory [1], anomaly detection in modern condition-based monitoring applications is still challenging due to various reasons. Major requirements suggested can be summarized as follows. *First*, an anomaly detection algorithm should be capable of handling nuisance variables, which behave like random noise even under normal conditions. We wish to automatically down-weight such unimportant variables as a result of model training. *Second*, it should be capable of handling dynamic state changes over time. The assumption of single Gaussian distribution in the T^2 theory is sometimes not appropriate. We wish to capture multiple operational modes due to dynamic changes in operational conditions of the system. *Third*, it should be capable of giving actionable or diagnostic information. For that direction, providing *variable-wise* anomaly scores will be a promising approach, instead of giving a single scalar as is the case in most of the traditional outlier detection methods.

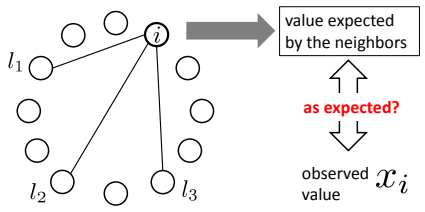

Figure 1. High-level picture of variable-wise anomaly scoring using Gaussian Markov random fields (GMRF). Intuitively, the anomaly score for the i-th variable measures the discrepancy from what is expected by its neighbors ($\{x_{l_1}, x_{l_2}, x_{l_3}\}$ in this case) in the GMRF sense.

To overcome the limitations of the traditional approach, much work has been done in the data mining community. Major approaches include subspace-based methods [2], [3], [4], [5] distance-based methods [6], [7], and mixture models [8], [9], [10]. However, the goal of these approaches is basically to provide a single scalar representing the degree of outlierness of a sample, and it is generally not straightforward to produce variable-wise information. Although the tasks of anomaly analysis [11] and anomaly localization [12] have been proposed recently, they are not readily applicable to our problem of multivariate but variable-wise anomaly scoring.

This paper presents a statistical machine learning approach to anomaly detection that can 1) automatically remove unwanted effects of nuisance variables, 2) handle multiple states of the system, and 3) compute variable-wise anomaly scores. Specifically, we focus on Gaussian Markov Random Fields (GMRF) that provide us a natural way to calculate variable-wise anomaly scores (see Fig. 1). To handle multiple operational modes, we then introduce a mixture of GMRF and propose a novel method to define the conditional distribution from the mixture consistently. Also, to handle nuisance variable, we propose an approach to learning a sparse mixture of the sparse graphical Gaussian model (GGM) [13]. We leverage not only ℓ_1 regularization to achieve sparsity in the variable dependency, but also the automated relevance determination (ARD) mechanism [14]

to achieve sparsity over mixture components. To the best of our knowledge, this is the first work for anomaly detection that extends GMRFs and sparse GGMs to mixtures. Using real sensor data of an oil production compressor, we show that our model is capable of capturing multiple operational conditions and significantly reduce false alerts that have been thought of as unavoidable.

Regarding related work, in the area of image processing, GMRFs have been extensively studied for the purpose of denoising [15], [16]. However, most of them are based on single component GMRFs, not on mixtures. To the best of our knowledge, practical procedures to derive the conditional distribution from GMRF mixtures are not known at least in the context of anomaly detection.

2. Problem setting

We are given a training data set $\mathcal{D}$ as

$$\mathcal{D} = \{\boldsymbol{x}^{(t)} \in \mathbb{R}^M \mid t = 1, \ldots, N\}, \tag{1}$$

where N is the number of observations and M is the dimensionality of the samples, or the number of sensors. We represent the dimensions by subscripts and the sample indexes by superscripts, e.g. $x_i^{(n)}$. The training data $\mathcal{D}$ is assumed to be *collected under normal conditions* of the system. One of the major assumptions is that the data generating mechanism may include multiple operational modes and would not be captured by a unimodal model.

Our goal is to compute the *variable-wise* anomaly score for a new sample, $\boldsymbol{x}$. For the i-th variable, it can be generally defined as

$$a_i(\boldsymbol{x}) = -\ln p(x_i \mid \boldsymbol{x}_{-i}, \mathcal{D}), \tag{2}$$

where $p(x_i|\boldsymbol{x}_{-i}, \mathcal{D})$ is the conditional predictive distribution for the i-th variable, given the rest of the variables $\boldsymbol{x}_{-i} \equiv (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_M)^\top$. Intuitively, a_i computes the degree of discrepancy between an observed x_i and what is expected by the rest variables $\boldsymbol{x}_{-i}$ (see Fig. 1).

This definition is a natural extension of Hotelling's T^2, which computes the outlier score with $-\ln \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}, \Sigma)$ up to unimportant constant terms and a prefactor. Here $\mathcal{N}(\cdot \mid \boldsymbol{\mu}, \Sigma)$ denotes the Gaussian distribution with the mean $\boldsymbol{\mu}$ and the covariance matrix Σ. Notice that a_{T^2} is just a single scalar even when $\boldsymbol{x}$ is a multivariate sample. Our task is more general than traditional outlier detection.

3. Gaussian Markov random field mixtures

This section describes how to derive the conditional predictive distribution $p(x_i|\boldsymbol{x}_{-i}, \mathcal{D})$ from a mixture of Gaussian Markov random field, given the generative process of $\boldsymbol{x}$.

3.1. Gaussian Markov random field

For the conditional predictive distribution, we assume the following mixture model:

$$p(x_i|\boldsymbol{x}_{-i}, \mathcal{D}) = \sum_{k=1}^{K} g_k^i(\boldsymbol{x})\, \mathcal{N}\left(x_i \mid u_i^k, w_i^k\right), \tag{3}$$

where $g_k^i(\boldsymbol{x})$ is a function called gating function that is learned from the data (see Eq. (21) and its footnote). Each k specifies a mixture component. Since we are interested in modeling the conditional distribution, unlike standard mixture models, the mixture weights depend on the index i. We also assume that the data generating process of $\boldsymbol{x}$ is described by a K-component Gaussian mixture $p(\boldsymbol{x}|\mathcal{D}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}|\boldsymbol{m}^k, (\mathsf{A}^k)^{-1})$. The means and the precision matrices $\{\boldsymbol{m}^k, \mathsf{A}^k\}$ as well as the optimal number of K are also learned from the data (see Sec. 4), but let us assume that they are given for now.

For the mean u_i^k and the variance w_i^k in Eq. (3), we use a particular form of

$$u_i^k = m_i^k - \frac{1}{A_{i,i}^k} \sum_{l \neq i}^{M} A_{i,l}^k (x_l - m_l^k), \tag{4}$$

$$w_i^k = \frac{1}{A_{i,i}^k}. \tag{5}$$

Gaussian distributions having these expressions are generally called the *Gaussian Markov random field (GMRF)*. The term "Markov" highlights the property that only direct neighbors as defined by nonzero entries of A^k can affect the distribution of x_i (see Fig. 1). For the derivation of the functional form, see Theorem 2.3 in [17]. Note that the problem is trivial when $K = 1$. In this case, the anomaly score (2) is readily given by

$$a_i(\boldsymbol{x})_{K=1} = \frac{1}{2A_{i,i}} [\mathsf{A}(\boldsymbol{x} - \boldsymbol{m})]_i{}^2 - \frac{1}{2} \ln \frac{A_{i,i}}{2\pi}, \tag{6}$$

where we dropped the superscript k and $[\cdot]_i$ denotes the i-th entry of a vector inside the square bracket. This paper is all about how to handle difficulties when $K > 1$.

3.2. Variational inference for GMRF mixture

Now let us consider how to find the gating function in Eq. (3) under the assumption that $\{(\boldsymbol{m}^k, \mathsf{A}^k)\}$ are given. With a cluster assignment indicator for the i-th variable, $\boldsymbol{h}^i$, we consider the following model:

$$p(x_i \mid \boldsymbol{x}_{-i}, \boldsymbol{h}^i) = \prod_{k=1}^{K} \mathcal{N}\left(x_i \mid u_i^k, w_i^k\right)^{h_k^i}, \tag{7}$$

$$p(\boldsymbol{h}^i \mid \boldsymbol{\theta}^i) = \prod_{k=1}^{K} \left(\theta_k^i\right)^{h_k^i}, \tag{8}$$

$$p(\boldsymbol{\theta}^i \mid \boldsymbol{\alpha}^i) = \frac{\Gamma(\alpha_1^i) \cdots \Gamma(\alpha_K^i)}{\Gamma(\bar{\alpha}^i)} \prod_{k=1}^{K} (\theta_k^i)^{\alpha_k^i - 1} \tag{9}$$

where $\sum_{k=1}^{K}\theta_k^i = 1$, $\Gamma(\cdot)$ is the gamma function, and $\bar{\alpha} \equiv \sum_{k=1}^{K}\alpha_k^i$ with α_k^i being a hyper parameter treated as a given constant. Alternatively, $p(\boldsymbol{\theta}^i|\boldsymbol{\alpha})$ may be denoted by $\mathrm{Dir}(\boldsymbol{\theta}^i|\boldsymbol{\alpha})$, the Dirichlet distribution. As usual, $h_k^i \in \{0,1\}$ and $\sum_{k=1}^{K} h_k^i = 1$. Based on this model, the complete log likelihood is written as

$$\ln P(\mathcal{D}, \mathsf{H}^i \mid \boldsymbol{\theta}^i) = \sum_{n=1}^{N}\sum_{k=1}^{K} h_k^{i(n)} \ln\left\{\theta_k^i \mathcal{N}(x_i^{(n)}|u_i^k, w_i^k)\right\} - \ln\Gamma(\bar{\alpha}) + \sum_{k=1}^{K}\left\{\ln\Gamma(\alpha_k) + (\alpha_k - 1)\ln\theta_k^i\right\}, \quad (10)$$

where $\boldsymbol{h}^{i(n)}$ is the indicator vector for the n-th sample and H^i is a collective notations for $\{\boldsymbol{h}^{i(n)} \mid n = 1, \ldots, N\}$.

To infer the model, we use the variational Bayes (VB) method [14]. We assume the functional form of the posterior distributions as

$$q(\mathsf{H}^i) = \prod_{n=1}^{N}\prod_{k=1}^{K}\left\{g_k^{i(n)}\right\}^{h_k^{i(n)}}, \quad (11)$$

$$q(\boldsymbol{\theta}^i) = \mathrm{Dir}(\boldsymbol{\theta}^i \mid \boldsymbol{a}^i). \quad (12)$$

VB and point-estimation equations are given as

$$\ln q(\mathsf{H}^i) = \mathrm{c.} + \left\langle \ln P(\mathcal{D}, \mathsf{H}^i \mid \boldsymbol{\theta}^i)\right\rangle_{\boldsymbol{\theta}^i} \quad (13)$$

$$\ln q(\boldsymbol{\theta}^i) = \mathrm{c.} + \left\langle \ln P(\mathcal{D}, \mathsf{H}^i \mid \boldsymbol{\theta}^i)\right\rangle_{\mathsf{H}^i}, \quad (14)$$

where c. symbolically represents a constant. $\langle\cdot\rangle_{\mathsf{H}^i}$ and $\langle\cdot\rangle_{\boldsymbol{\theta}^i}$ represent the expectation by $q(\mathsf{H}^i)$ and $q(\boldsymbol{\theta}^i)$, respectively. Using the well-known result $\left\langle \ln\theta_k^i\right\rangle_{\boldsymbol{\theta}^i} = \psi(a_k^i) - \psi(\bar{a}^i)$, where $\psi(\cdot)$ is the di-gamma function and $\bar{a}^i \equiv \sum_{k=1}^{K} a_k^i$, we can easily derive VB iterative equations as

$$a_k^i \leftarrow \alpha_k + N_k^i, \quad (15)$$

$$\bar{\theta}_k^i \leftarrow \exp\left\{\psi(a_k^i) - \psi(\bar{a}^i)\right\}, \quad (16)$$

$$g_k^{i(n)} \leftarrow \frac{\bar{\theta}_k^i \mathcal{N}(x_i^{(n)}|u_i^k, w_i^k)}{\sum_{l=1}^{K}\bar{\theta}_l^i \mathcal{N}(x_i^{(n)}|u_i^l, w_i^l)} \quad \text{for all } n, \quad (17)$$

$$N_k^i \leftarrow \sum_{n=1}^{N} g_k^{i(n)}, \quad (18)$$

These substitutions are performed until convergence. Repeating over $i = 1, \ldots, M$ and $k = 1, \ldots, K$, we obtain a $M \times K$ matrix $\Theta = [\theta_k^i]$.

3.3. Predictive distribution for GMRF mixture

The predictive distribution in Eq. (2) is formally defined as

$$p(x_i|\boldsymbol{x}_{-i}, \mathcal{D}) = \int \mathrm{d}\boldsymbol{h}^i \; q(\boldsymbol{h}^i)\, p(x_i \mid \boldsymbol{x}_{-i}, \boldsymbol{h}^i). \quad (19)$$

To find $q(\boldsymbol{h}^i)$, which is the posterior distribution for the indicator variable associated with a new sample $\boldsymbol{x}$, consider an augmented data set $\mathcal{D} \cup \boldsymbol{x}$. In this case, the log complete likelihood is given by

$$\ln P(\mathcal{D}, \boldsymbol{x}, \mathsf{H}^i, \boldsymbol{h}^i \mid \boldsymbol{\theta}^i) = \ln P(\mathcal{D}, \mathsf{H}^i, \mid \boldsymbol{\theta}^i) + \sum_{k=1}^{K} h_k^i \ln\left\{\theta_k^i \mathcal{N}(x_i|u_i^k, w_i^k)\right\}. \quad (20)$$

Corresponding to this, let the posterior be

$$q(\mathsf{H}^i, \boldsymbol{h}^i) = q(\mathsf{H}^i) \times \prod_{k=1}^{K}(g_k^i)^{h_k^i},$$

from which we get VB iterative equations similar to Eqs. (15)-(18). Although the resulting $\{\theta_k^i\}$ differs from the one obtained using only $\mathcal{D}$, Eq. (18) suggests that the difference is just on the order of $1/N$, which is negligible when $N \gg 1$. Therefore, we conclude that the posterior distribution of a new sample $\boldsymbol{x}$ is given by

$$g_k^i(\boldsymbol{x}) \approx \frac{\bar{\theta}_k^i \mathcal{N}(x_i|u_i^k, w_i^k)}{\sum_{l=1}^{K}\bar{\theta}_l \mathcal{N}(x_i|u_i^l, w_i^l)}. \quad (21)$$

where θ_k^i is the solution of Eqs. (17)-(18) [1].

Finally, using Eqs. (3) and (21), the *variable-wise* anomaly score defined in Eq. (2) is given by

$$a_i(\boldsymbol{x}) = -\ln\sum_{k=1}^{K} g_k^i(\boldsymbol{x})\, \mathcal{N}\left(x_i \mid u_i^k, w_i^k\right). \quad (22)$$

The r.h.s. includes the parameters $\{(\boldsymbol{m}^k, \mathsf{A}^k)\}$ that represent the generative process of $\boldsymbol{x}$. Next section discuss how to get them.

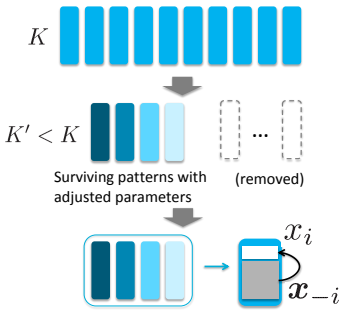

Figure 2. Overview of the `sGMRFmix` algorithm. Starting from K initial patterns that may be redundant, a sparse mixture of sparse graphical models is learned, from which $p(x_i|\boldsymbol{x}_{-i}, \mathcal{D})$ is derived.

4. Sparse mixture of sparse graphical models

To capture multiple operational modes of the system, we assume a Gaussian mixture model for the generative process of $\boldsymbol{x}$. To ensure the capability of removing noisy nuisance variables, we further request that the model should be *sparse*. This section explains to learn *sparse mixture* of *sparse* graphical Gaussian models (see Fig. 2).

1. By construction, $g_k^i(\boldsymbol{x})$ has to be treated as constant when considering the normalization condition of Eq. (3).

4.1. Observation model and priors

We employ a Bayesian Gaussian mixture model having K mixture components. First, we define the observation model by

$$p(\boldsymbol{x} \mid \boldsymbol{z}, \boldsymbol{\mu}, \Lambda) \equiv \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}^k, (\Lambda^k)^{-1})^{z_k}, \tag{23}$$

where $\boldsymbol{\mu}$ and Λ are collective notations representing $\{\boldsymbol{\mu}^k\}$ and $\{\Lambda^k\}$, respectively. Also, $\boldsymbol{z}$ is the indicator variable of cluster assignment. As before, $z_k \in \{0, 1\}$ for all k, and $\sum_{k=1}^{K} z_k = 1$.

We place the Gauss-Laplace prior on $(\boldsymbol{\mu}^k, \Lambda^k)$ and the categorical distribution on $\boldsymbol{z}$:

$$p(\boldsymbol{\mu}^k, \Lambda^k) \propto \mathrm{e}^{-\frac{\rho}{2}\|\Lambda^k\|_1} \mathcal{N}(\boldsymbol{\mu}^k | \boldsymbol{m}_0, (\lambda_0 \Lambda^k)^{-1}), \tag{24}$$

$$p(\boldsymbol{z}|\boldsymbol{\pi}) = \prod_{k=1}^{K} \pi_k^{z_k} \ \text{ s.t. } \ \sum_{k=1}^{K} \pi_k = 1, \ \pi_k \geq 0, \tag{25}$$

where $\|\Lambda\|_1 = \sum_{i,j} |\Lambda_{i,j}|$. The parameter $\boldsymbol{\pi}$ is determined as a part of the model while $\rho, \lambda_0, \boldsymbol{m}_0$ are given constants. From these equations, we can write down the complete likelihood as

$$\begin{aligned} P(\mathcal{D}, \mathsf{Z}, \Lambda \,|\boldsymbol{\mu}, \boldsymbol{\pi}) \equiv & \prod_{k=1}^{K} p(\boldsymbol{\mu}^k, \Lambda^k) \\ & \times \prod_{n=1}^{N} p(\boldsymbol{z}^{(n)}|\boldsymbol{\pi}) p(\boldsymbol{x}^{(n)} \mid \boldsymbol{z}^{(n)}, \boldsymbol{\mu}, \Lambda), \end{aligned} \tag{26}$$

where Z is a collective notation for $\{\boldsymbol{z}_k^{(n)}\}$.

4.2. Variational Bayes inference

Since the Laplace distribution is not the conjugate prior of Gaussian, exact inference is not possible. We again use the VB method based on the categorical distribution for the posterior of Z and the *Gauss-delta* distribution for the posterior of $(\boldsymbol{\mu}, \Lambda)$:

$$q(\mathsf{Z}) = \prod_{n=1}^{N} \prod_{k=1}^{K} (r_k^{(n)})^{z_k^{(n)}}, \tag{27}$$

$$q(\boldsymbol{\mu}, \Lambda) = \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{\mu}^k | \boldsymbol{m}^k, (\lambda^k \Lambda^k)^{-1}) \delta(\Lambda^k - \bar{\Lambda}^k), \tag{28}$$

where $\delta(\cdot)$ is Dirac's delta function. We combine VB analysis for $\{\mathsf{Z}, \boldsymbol{\mu}, \Lambda\}$ with point estimation for the mixture weight $\boldsymbol{\pi}$. As shown in [18], this leads to a sparse solution (i.e. $\pi_k = 0$ in many k's) through the ARD mechanism.

By expanding $\langle \ln P(\mathcal{D}, \mathsf{Z}, \Lambda \mid \boldsymbol{\pi}, \boldsymbol{\mu}) \rangle_{\Lambda, \boldsymbol{\mu}}$, it is straightforward to obtain the VB iterative equation for $\{r_k^{(n)}\}$:

$$\ln r_k^{(n)} \leftarrow \ln \left\{ \pi_k \, \mathcal{N}(\boldsymbol{x}^{(n)} \mid \boldsymbol{m}^k, (\bar{\Lambda}^k)^{-1}) \right\} - \frac{M}{2\lambda^k} \tag{29}$$

$$r_k^{(n)} \leftarrow \frac{r_k^{(n)}}{\sum_{l=1}^{k} r_l^{(n)}}. \tag{30}$$

Similarly, for the other variables including point-estimated $\boldsymbol{\pi}$, we have the VB solution as

$$N^k \leftarrow \sum_{n=1}^{N} r_k^{(n)}, \quad \pi_k \leftarrow \frac{N^k}{N}, \tag{31}$$

$$\bar{\boldsymbol{x}}^k \leftarrow \frac{1}{N^k} \sum_{n=1}^{N} r_k^{(n)} \boldsymbol{x}^{(n)}, \tag{32}$$

$$\Sigma^k \leftarrow \frac{1}{N^k} \sum_{n=1}^{N} r_k^{(n)} (\boldsymbol{x}^{(n)} - \bar{\boldsymbol{x}}^k)(\boldsymbol{x}^{(n)} - \bar{\boldsymbol{x}}^k)^\top, \tag{33}$$

$$\lambda^k \leftarrow \lambda_0 + N^k, \ \ \boldsymbol{m}^k \leftarrow \frac{1}{\lambda^k}(\lambda_0 \boldsymbol{m}_0 + N^k \bar{\boldsymbol{x}}^k), \tag{34}$$

$$\mathsf{Q}^k \leftarrow \Sigma^k + \frac{\lambda_0}{\lambda^k} (\bar{\boldsymbol{x}}^k - \boldsymbol{m}_0)(\bar{\boldsymbol{x}}^k - \boldsymbol{m}_0)^\top, \tag{35}$$

$$\bar{\Lambda}^k \leftarrow \arg\max_{\Lambda^k} \left\{ \ln |\Lambda^k| - \mathrm{Tr}(\Lambda^k \mathsf{Q}^k) - \frac{\rho}{N^k} \|\Lambda^k\|_1 \right\}. \tag{36}$$

These VB equations are computed for $k = 1, \ldots, K$ and repeated until convergence. Notice that the VB equation for $\bar{\Lambda}^k$ preserves the original ℓ_1-regularized GGM formulation [13]. We see that the fewer samples a cluster have, the more the ℓ_1 regularization is applied due to the ρ/N^k term.

Finally, the predictive distribution is given by

$$\begin{aligned} p(\boldsymbol{x}|\mathcal{D}) &= \sum_{k=1}^{K} \pi_k \int \mathrm{d}\boldsymbol{\mu}^k \int \mathrm{d}\Lambda^k \, \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}^k, (\Lambda^k)^{-1}) q(\boldsymbol{\mu}^k, \Lambda^k), \\ &= \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{m}^k, (\mathsf{A}^k)^{-1}), \end{aligned} \tag{37}$$

where $\mathsf{A}^k \equiv \frac{\lambda^k}{1+\lambda^k} \bar{\Lambda}^k$. This is the one we assumed in Sec. 3.

5. Algorithm summary

Algorithm 1 gives a high-level summary of `sGMRFmix` (sparse GMRF mixture) algorithm. The first stage (Sec. 4), `sparseGaussMix`, starts with a large enough number of K and identifies major dependency patterns from the data. In the context of industrial condition-based monitoring, initialization of $\{\boldsymbol{m}^k, \bar{\Lambda}^k\}$ can be naturally done by disjointly partitioning the data along the time axis as $\mathcal{D} = \mathcal{D}^1 \cup \ldots \cup \mathcal{D}^K$ and apply e.g. the graphical lasso algorithm [13] on each, as illustrated in Fig. 2. After the initialization, the VB iteration can start with $\pi_k = \frac{1}{K}$ and $\lambda^k = \pi_k N$, as well as with $\lambda_0 = 1, \boldsymbol{m}_0 = \boldsymbol{0}$ if no prior information is available.

The second stage (Sec. 3), `GMRFmix`, determines the gating function $g_k^i(\boldsymbol{x})$ for an arbitrary input sample $\boldsymbol{x}$ through the resulting $\bar{\theta}_k^i$'s to define the anomaly score in Eq. (22). For $\boldsymbol{\alpha}$, it is reasonable to choose $\alpha_k = 1$ for k's with $\pi_k \neq 0$ and zero otherwise. Regarding ρ, an optimal value should be determined together with the threshold on the anomaly score, so the performance of anomaly detection is maximized. One reasonable performance metric is the F-measure between the accuracies separately computed for the normal and anomalous samples.

Algorithm 1 The `sGMRFmix` algorithm

Input: $\mathcal{D}, \rho, \boldsymbol{\alpha}$.
Output: $\{\boldsymbol{m}^k, \lambda^k, \bar{\Lambda}^k\}, \{\bar{\theta}_k^i\}$.
$\{\pi_k, \boldsymbol{m}^k, \mathsf{A}^k\} = \texttt{sparseGaussMix}(\mathcal{D}, \boldsymbol{m}_0, \lambda_0, \rho)$.
$\{\bar{\theta}_k^i\} = \texttt{GMRFmix}(\{\pi_k, \boldsymbol{\mu}^k, \mathsf{A}^k\}, \boldsymbol{\alpha})$.

6. Experimental results

This section presents experimental results of the proposed algorithm. Methods compared are as follows.

`single` [11] is essentially the same as the $K = 1$ version of the proposed algorithm with $\lambda_0 = 0$. The same ρ value is used as `GMRFmix`.

`sPCA` [19] computes the i-th anomaly score via

$$a_i(\boldsymbol{x})_{\text{sPCA}} \equiv |x_i - \boldsymbol{e}_i^\top \mathsf{U}\mathsf{U}^\top \boldsymbol{x}|,$$

where $\boldsymbol{e}_i$ is the i-th basis vector and $\mathsf{U} \equiv [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{K'}]$ is the matrix of K' principal components computed by the sparse principal component analysis (sPCA) [20]. The same values of ρ and K' as `GMRFmix` are used for the ℓ_1 regularization coefficient and U, respectively.

`autoencoder` trains a sparse autoencoder [21] with one hidden layer based on the normalized input as $x_i \leftarrow \frac{x_i - \min_i}{\max_i - \min_i}$, where $\max_i$ and $\min_i$ are the maximum and minimum values of the i-th variable over the *training* data, respectively. The anomaly score is simply defined as

$$a_i(\boldsymbol{x})_{\text{autoendcoder}} \equiv |x_i - \hat{x}_i|, \tag{38}$$

where $\hat{x}_i$ is the output of the i-th output neuron. The input, hidden and output layers have the same number of neurons. The value of ℓ_1 and ℓ_2 regularization parameters (β and λ in [21]) are determined by cross-validation on the averaged reconstruction error on the training data.

6.1. Synthetic data: illustration

We synthetically generated a data set by adding t-distributed random noise to Gaussian distributed samples whose correlation structures are shown in Fig. 3. By shifting the mean, we created a sequence of A-B-A-B for the training data and A-B-Anomaly for the testing data. Both data have 1 000 samples, as shown in Fig. 4, where the Anomaly patten is highlighted with the dashed line. To initialize the model, we used a $K = 7$ disjoint partitioning (see Sec. 5 for the detail). Figure 5 shows the learned model. We see that the distinctive patterns A and B are automatically discovered without specifying the ground truth cluster number, thanks to the ARD mechanism.

With the trained model, we computed the anomaly score on the testing data and evaluated the AUC (area under the curve) based on the classification accuracies separately computed for negative and positive samples. The accuracies are defined on the negative and positive labels given to the first and second half of the testing data, respectively. Table 1 clearly shows that the proposed model outperforms the alternative.

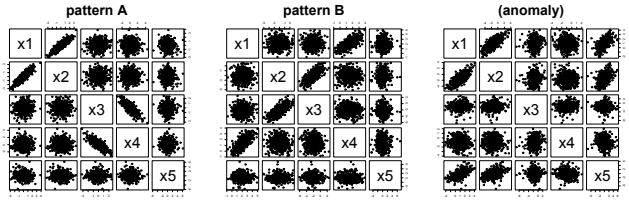

Figure 3. Synthetic Pattern A, Pattern B, and Anomaly.

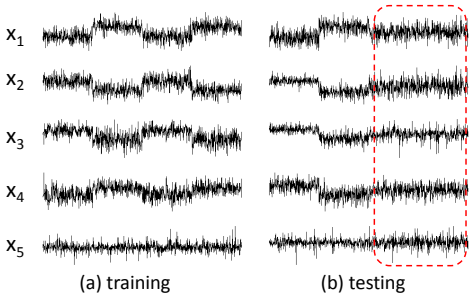

Figure 4. Synthetic training and testing data.

6.2. Real application: offshore oil production

We applied the proposed method to the task of early anomaly detection of a compressor of offshore oil production. Figure 6 shows simulated examples out of $M = 53$ sensor signals (acceleration, pressure, flow rate, etc.) over about one month. Apparently, the system irregularly makes transitions to different trends under heavy spike-like noise. See [22] for more details about challenges in the condition-based monitoring in the oil industry.

To train the model, we used reference data under normal operating conditions over about one year selected by domain experts. For `sGMRFmix`, we partitioned the data into $K = 21$ disjoint subsets for initialization. We also trained the alternative methods. Cross-validated parameters for the sparse autoencoder are $\lambda = 10^{-8}$ and $\beta = 10^{-8}$ in the notation of [21]. Figure 7 shows $\{\pi_k\}$ computed by `sparseGaussMix` with $\rho = 0.1$. Thanks to the ARD mechanism, the algorithm automatically discovered 4 major patterns ($K' = 13$ in total).

Using the trained model, we computed the variable-wise anomaly score on testing data including a few real failures that were unable to be detected by an existing monitoring system. Figure 8 presents the distribution of a_{14} over the testing data. This is a flow rate variable and was confirmed to be involved in the physical failure process related to pump surge. In Fig. 8 (a), we see that the anomaly score of the pre-failure window is significantly higher than the other period while the separation is not very clear in (b)-(c).

TABLE 1. AUC VALUES FOR THE SYNTHETIC DATA.

`sGMRFmix`	`single`	`sPCA`	`autoencoder`
0.72	0.52	0.63	0.57

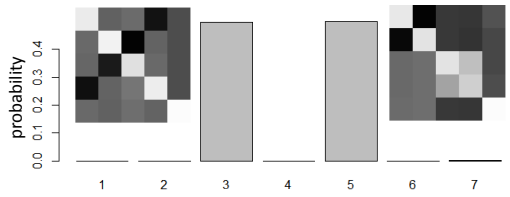

Figure 5. Converged mixture weights $\{\pi_k\}$ and the corresponding precision matrices for the synthetic data.

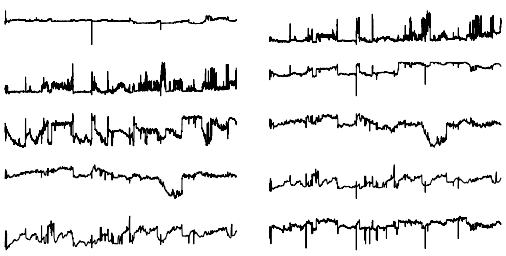

Figure 6. Compressor data under normal operating condition.

7. Conclusion

We have proposed a new outlier detection method, the sparse GMRF mixture, that is capable of handling multiple operational modes in the normal condition and variable-wise anomaly scores. We derived variational Bayes iterative equations based on the Gauss-delta posterior model.

References

[1] T. W. Anderson, *An Introduction to Multivariate Statistical Analysis*, 3rd ed. Wiley-Interscience, 2003.

[2] T. Idé and H. Kashima, "Eigenspace-based anomaly detection in computer systems," in *Proc. ACM SIGKDD Intl. Conf. Knowledge Discovery and Data Mining*, 2004, pp. 440–449.

[3] H. Ringberg, A. Soule, J. Rexford, and C. Diot, "Sensitivity of pca for traffic anomaly detection," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 35, no. 1. ACM, 2007, pp. 109–120.

[4] L. Xiong, X. Chen, and J. Schneider, "Direct robust matrix factorizatoin for anomaly detection," in *Data Mining (ICDM), 2011 IEEE 11th International Conference on*. IEEE, 2011, pp. 844–853.

[5] D. Blythe, P. von Bunau, F. Meinecke, and K.-R.Muller, "Feature extraction for change-point detection using stationary subspace analysis," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 4, pp. 631–643, 2012.

[6] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: identifying density-based local outliers," *ACM SIGMOD Record*, vol. 29, no. 2, pp. 93–104, 2000.

[7] G. O. Campos, A. Zimek, J. Sander, R. J. Campello, B. Micenková, E. Schubert, I. Assent, and M. E. Houle, "On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study," *Data Mining and Knowledge Discovery*, pp. 1–37, 2016.

[8] K. Yamanishi, J. Takeuchi, G. Williams, and P. Milne, "On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms," in *Proc. the Sixth ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, 2000, pp. 320–324.

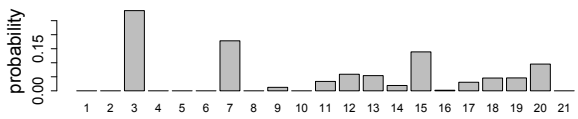

Figure 7. Converged mixture weights $\{\pi_k\}$ for the compressor data.

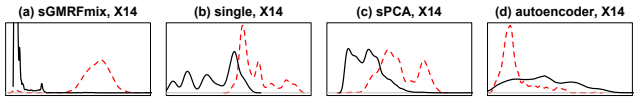

Figure 8. Density plot of the anomaly scores in arbitrary scales. The dashed lines represent the distribution in the 24-hour pre-failure window.

[9] S. Hirai and K. Yamanishi, "Detecting changes of clustering structures using normalized maximum likelihood coding," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD 12, 2012, pp. 343–351.

[10] L. I. Kuncheva, "Change detection in streaming multivariate data using likelihood detectors," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 5, pp. 1175–1180, 2013.

[11] T. Idé, A. C. Lozano, N. Abe, and Y. Liu, "Proximity-based anomaly detection using sparse structure learning," in *Proc. of 2009 SIAM International Conference on Data Mining (SDM 09)*, pp. 97–108.

[12] R. Jiang, H. Fei, and J. Huan, "Anomaly localization for network data streams with graph joint sparse PCA," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2011, pp. 886–894.

[13] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, 2008.

[14] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer-Verlag, 2006.

[15] S. M. Schweizer and J. M. F. Moura, "Hyperspectral imagery: clutter adaptation in anomaly detection," *IEEE Transactions on Information Theory*, vol. 46, no. 5, pp. 1855–1871, Aug 2000.

[16] L. Shadhan and I. Cohen, "Detection of anomalies in texture images using multi-resolution random field models," *Signal Processing*, vol. 87, pp. 3045–3062, 2007.

[17] H. Rue and L. Held, *Gaussian Markov Random Fields: Theory and Applications*, ser. CRC Monographs on Statistics & Applied Probability. Chapman & Hall, 2005.

[18] A. Corduneanu and C. M. Bishop, "Variational bayesian model selection for mixture distributions," in *Artificial intelligence and Statistics*, vol. 2001. Morgan Kaufmann Waltham, MA, 2001, pp. 27–34.

[19] R. Jiang, H. Fei, and J. Huan, "A family of joint sparse pca algorithms for anomaly localization in network data streams," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 11, pp. 2421–2433, 2013.

[20] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *Journal of Computational and Graphical Statistics*, vol. 15, pp. 265–286, 2006.

[21] A. Ng, "Sparse autoencoder," *CS294A Lecture notes*, vol. 72, pp. 1–19, 2011.

[22] S. Natarajan and R. Srinivasan, "Multi-model based process condition monitoring of offshore oil and gas production process," *Chemical Engineering Research and Design*, vol. 88, no. 5, pp. 572–591, 2010.